

	<p style="text-align: center;"><b>Architektura systemów komputerowych</b></p> <p style="text-align: center;">dr inż. Piotr Zielniewicz e-mail: piotr.zielniewicz@cs.put.poznan.pl</p>

	<p><b>Literatura</b></p> <ul style="list-style-type: none"> <li>■ P. Metzger, <i>Anatomia PC</i>, Helion, 2007</li> <li>■ J. Biernat, <i>Architektura komputerów</i>, Oficyna Wydawnicza Politechniki Wrocławskiej, 1999</li> <li>■ W. Stallings, <i>Organizacja i architektura systemu komputerowego</i>, WNT Warszawa, 2003</li> <li>■ R. Pełka, <i>Mikrokontrolery</i>, WKŁ Warszawa, 2000</li> <li>■ J. Sibigroth, <i>Zrozumieć małe mikrokontrolery</i>, BTC Warszawa, 2003</li> <li>■ C. Petzold, <i>Kod</i>, WNT Warszawa, 2002</li> </ul>
	2

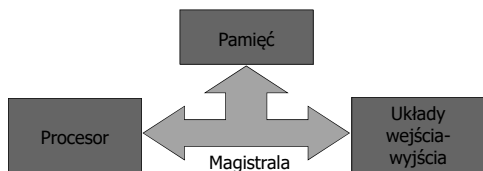
	<p><b>Zarys historii komputerów</b></p>
	<ul style="list-style-type: none"> <li>■ 1945: ENIAC (<i>Electronic Numerical Integrator And Computer</i>) <ul style="list-style-type: none"> <li>– konstrukcja oparta na lampach próżniowych</li> <li>– arytmetyka w systemie dziesiętnym</li> <li>– pamięć dwudziestu 10-cyfrowych liczb</li> <li>– ręczne programowanie z użyciem przełączników i kabli krosujących</li> <li>– szybkość przetwarzania: 5000 operacji dodawania na sekundę</li> </ul> </li> <li>■ 1946: EDVAC (<i>Electronic Discrete Variable Automatic Computer</i>) <ul style="list-style-type: none"> <li>– zastosowanie systemu binarnego</li> <li>– zastosowanie pamięci do przechowywania danych oraz kodu programu</li> </ul> </li> </ul>
	3

	<p><b>Zarys historii komputerów</b></p>
	<ul style="list-style-type: none"> <li>■ 1951: UNIVAC (<i>Universal Automatic Computer</i>) <ul style="list-style-type: none"> <li>– pierwszy komputer dostępny w handlu (określany terminem „mózg elektronowy”)</li> <li>– szybkość przetwarzania: ponad 20 000 operacji na sekundę</li> <li>– zastosowania: obsługa powszechnego spisu ludności z 1950 roku, prognozowanie wyników wyborów prezydenckich w 1952 roku</li> </ul> </li> <li>■ Koniec lat 50-tych: DEC PDP-1, IBM seria 7000 <ul style="list-style-type: none"> <li>– konstrukcje oparte na tranzystorach</li> <li>– bardziej złożone jednostki arytmetyczno-logiczne i sterujące</li> <li>– pamięci ferrytowe</li> <li>– rozbudowane urządzenia peryferyjne</li> <li>– 50-krotny wzrost prędkości w stosunku do komputerów pierwszej generacji</li> </ul> </li> </ul>
	4

	<p><b>Zarys historii komputerów</b></p>
	<ul style="list-style-type: none"> <li>■ Połowa lat 60-tych: DEC PDP-8, IBM System 360 <ul style="list-style-type: none"> <li>– konstrukcja oparta na układach scalonych</li> <li>– pamięci półprzewodnikowe</li> <li>– koncepcja minikomputera</li> <li>– struktura magistralowa</li> </ul> </li> <li>■ 1971: Mikroprocesor Intel 4004 <ul style="list-style-type: none"> <li>– wszystkie elementy procesora zintegrowane w jednym układzie (2300 tranzystorów)</li> <li>– 4-bitowa szerokość szyny danych</li> </ul> </li> <li>■ 1974: Mikroprocesory Intel 8080, Motorola 6800 <ul style="list-style-type: none"> <li>– mikroprocesory ogólnego przeznaczenia</li> <li>– 8-bitowa szerokość szyny danych</li> <li>– 16-bitowa szerokość szyny adresowej</li> </ul> </li> </ul>
	5

	<p><b>Zarys historii komputerów</b></p>
	<ul style="list-style-type: none"> <li>■ 1977: Mikrokomputer Apple II</li> <li>■ 1978, 1979: Mikroprocesory Intel 8086, 8088 <ul style="list-style-type: none"> <li>– 16-bitowa szerokość szyny danych</li> <li>– 20-bitowa szerokość szyny adresowej</li> <li>– wbudowane rozkazy mnożenia i dzielenia</li> </ul> </li> <li>■ 1981: Mikrokomputer IBM PC</li> <li>■ 1984: Mikrokomputer Apple Macintosh</li> <li>■ 1985: Mikroprocesor Intel 80386 <ul style="list-style-type: none"> <li>– 32-bitowa szerokość szyny danych</li> <li>– 32-bitowa szerokość szyny adresowej</li> <li>– wsparcie dla systemów wielozadaniowych</li> </ul> </li> </ul>
	6

Klasyczna koncepcja komputera
<ul style="list-style-type: none"> <li>■ Elementy funkcjonalne komputera: <ul style="list-style-type: none"> <li>– Procesor, jednostka centralna, CPU (<i>Central Processing Unit</i>)</li> <li>– Pamięć operacyjna</li> <li>– Układy wejścia-wyjścia</li> </ul> </li> <li>■ Schemat architektury:</li> </ul>



7

Klasyczna koncepcja komputera
<ul style="list-style-type: none"> <li>■ Elementy komputera komunikują się ze sobą poprzez magistrale: danych, adresową i sterującą</li> <li>■ Magistrale pracują w sposób synchroniczny, co oznacza, że sygnały na nich występujące, mogą się pojawiać wyłącznie w ściśle określonych momentach wyznaczonych przez tzw. <u>cykl zegarowy</u></li> <li>■ Przykład pobierania zawartości komórki pamięci operacyjnej przez procesor: <ul style="list-style-type: none"> <li>– procesor wysyła adres żądanej komórki na magistralę adresową, a po odczekaniu odpowiedniej liczby cykli zegarowych odczytuje z magistrali dane</li> <li>– pamięć dekoduje przesłany adres po czym umieszcza zawartość zaadresowanej komórki na magistrali danych w odpowiednim momencie i na odpowiednim długi czas</li> </ul> </li> </ul>

8

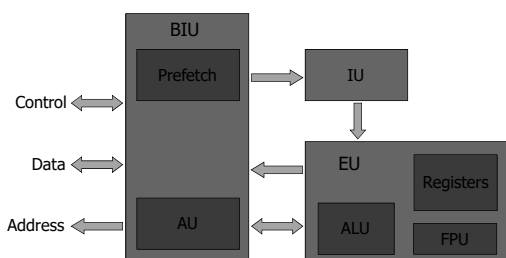
Klasyczna koncepcja komputera
<ul style="list-style-type: none"> <li>– jeżeli pamięć operacyjna jest zbyt wolna w stosunku do szybkości pracy procesora i nie może wysłać danej w odpowiednio krótkim czasie, to informuje o tym procesor, wysyłając odpowiedni sygnał (WAIT) przez magistralę sterującą</li> <li>– procesor wydłuża wówczas okres oczekiwania na daną o dodatkowy cykl zegarowy i w tak określonym momencie odczytuje ją z magistrali danych</li> <li>■ Opisany cykl pracy procesora nazywa się <u>cyklem maszynowym</u> i jest on związany ze zmianą adresu przesyłanego przez magistralę adresową</li> <li>■ Cykle maszynowe nie są jednakowej długości, choć zawsze są wielokrotnością cyklu zegarowego</li> </ul>

9

Klasyczna koncepcja komputera
<ul style="list-style-type: none"> <li>■ Komunikacja między procesorem a układami wejścia-wyjścia odbywa się w postaci analogicznych cykli maszynowych</li> <li>■ Komunikacja między urządzeniami wejścia-wyjścia a pamięcią operacyjną także przebiega w postaci podobnych cykli maszynowych w przypadku transmisji przez kanał bezpośredniego dostępu do pamięci</li> <li>■ Cykl pracy magistrali komputera składający się z pewnej liczby cykli maszynowych, w którym następuje wykonanie pełnego rozkazu procesora nazywa się <u>cyklem rozkazowym</u></li> </ul>

10

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Schemat blokowy procesora:</li> </ul>



11

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Komunikacja procesora ze światem zewnętrznym odbywa się za pośrednictwem układu sterowania magistralami BIU (<i>Bus Interface Unit</i>), w którego skład wchodzi jednostka adresowania AU (<i>Addressing Unit</i>) i podręczna kolejka rozkazów (<i>Prefetch</i>)</li> <li>■ Jednostka adresowania ma za zadanie wytworzenie na magistrali adresowej adresu, pod którym znajdują się wymieniane z procesorem dane lub pobierane rozkazy</li> <li>■ Pobierane rozkazy umieszczane są z wyprzedzeniem w podręcznej kolejce rozkazów, której uzupełnianie odbywa się automatycznie</li> </ul>

12

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Dekodowanie rozkazów pobieranych z kolejki rozkazów odbywa się w bloku dekodera instrukcji IU (<i>Instruction Unit</i>)</li> <li>■ Rozkodowane instrukcje przekazywane są do układu wykonawczego EU (<i>Execution Unit</i>), który w zależności od rodzaju rozkazu wydaje odpowiednie polecenia do jednostki adresującej oraz jednostki arytmetyczno-logicznej ALU (<i>Arithmetic-Logic Unit</i>) lub jednostki zmiennoprzecinkowej FPU (<i>Floating Point Unit</i>)</li> <li>■ Wykonywane rozkazy mogą dotyczyć manipulacji na zawartości wewnętrznych rejestrów procesora (<i>Registers</i>), jak też komórek pamięci operacyjnej</li> </ul>

13

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Zadaniem jednostki arytmetyczno-logicznej jest wykonywanie bezpośrednich operacji na danych tj. dodawanie, odejmowanie, porównywanie itd.</li> <li>■ W przypadku rozkazów operujących na danych zmiennoprzecinkowych ich przetwarzaniem zajmuje się wyspecjalizowana jednostka zmiennoprzecinkowa</li> <li>■ Nie istnieje ścisły podział pamięci operacyjnej na obszary danych i rozkazów – w każdej komórce pamięci operacyjnej mogą znajdować się zarówno podlegające przetwarzaniu dane, jak i przeznaczone do wykonania rozkazy</li> </ul>

14

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Ponieważ większość działań procesora sprowadza się do operacji na stosunkowo wolnej pamięci RAM (<i>Random Access Memory</i>), to w celu zwiększenia efektywności procesor wyposażony jest w rejestry – szybką pamięć wewnętrzną</li> <li>■ Wykonanie jakiegokolwiek operacji na pamięci sprowadza się do przesłania zawartości komórki pamięci do rejestru, wykonania operacji i odesłania wyniku do pamięci</li> <li>■ Rejestry można podzielić na trzy funkcjonalne grupy: <ul style="list-style-type: none"> <li>– rejestry ogólnego przeznaczenia (np. AX, BX, CX, DX)</li> <li>– rejestry segmentowe (np. CS, DS, SS)</li> <li>– rejestry statusowe i sterujące (np. FLAGS, IP)</li> </ul> </li> </ul>

15

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Zadaniem rejestrów ogólnego przeznaczenia jest przechowywanie danych</li> <li>■ W praktyce każdy z tych rejestrów jest przeznaczony do wykonywania określonych zadań np. rejestr AX (akumulator) w operacjach arytmetycznych i logicznych, BX do adresowania pamięci, CX jest licznikiem w pętlach, natomiast DX umożliwia przekazywanie i odbieranie danych z portów wejścia-wyjścia</li> <li>■ Rejestry segmentowe służą do przechowywania: <ul style="list-style-type: none"> <li>– CS (<i>Code Segment</i>) - adresu segmentu kodu do wykonania</li> <li>– DS (<i>Data Segment</i>) - adresu segmentu danych</li> <li>– SS (<i>Stack Segment</i>) - adresu segmentu stosu</li> </ul> </li> </ul>

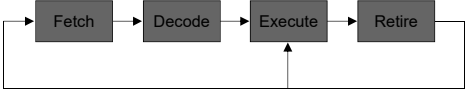
16

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Zadaniem rejestru stanu FLAGS jest kontrolowanie pracy procesora</li> <li>■ W rejestrze tym znajdują się m.in. następujące znaczniki stanu (flagi): <ul style="list-style-type: none"> <li>– <u>Znak</u> (<i>Sign Flag</i>) - bit znaku wyniku ostatniej operacji arytmetycznej</li> <li>– <u>Zero</u> (<i>Zero Flag</i>) - flaga ustawiana jeżeli wynik operacji jest równy zero</li> <li>– <u>Przeniesienie</u> (<i>Carry Flag</i>) - flaga ustawiana jeżeli wynikiem operacji arytmetycznej jest przeniesienie</li> <li>– <u>Przepelnienie</u> (<i>Overflow</i>) - flaga ustawiana jeżeli w czasie wykonywania operacji arytmetycznej wystąpiło przepelnienie</li> </ul> </li> </ul>

17

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>■ Rejestr wskaźnika rozkazów IP (<i>Instruction Pointer</i>) zawiera przemieszczenie aktualnie wykonywanej instrukcji względem początku segmentu kodu</li> <li>■ O tym że zawartość komórki pamięci operacyjnej zostanie przez procesor zinterpretowana jako rozkaz do wykonania decyduje fakt jej zaadresowania przez wskaźnik (licznik) rozkazów</li> <li>■ W cyklu rozkazowym procesora możemy wyróżnić następujące cztery fazy: <ul style="list-style-type: none"> <li>– <u>Faza pobrania rozkazu</u> (<i>Fetch</i>), w której następuje pobranie rozkazu z podręcznej kolejki rozkazów i jeżeli zachodzi taka potrzeba, uruchomienie procedury uzupełniania kolejki</li> </ul> </li> </ul>

18

Ogólna zasada działania procesora
<ul style="list-style-type: none"> <li>– <u>Faza dekodowania rozkazu</u> (<i>Decode</i>), w której określone są operacje do wykonania oraz obliczane adresy efektywne argumentów (jeżeli występują)</li> <li>– <u>Faza wykonania</u> (<i>Execute</i>), w której pobierane są z pamięci argumenty rozkazu oraz wykonywane operacje ustalone w fazie dekodowania</li> <li>– <u>Faza zapisania wyników</u> (<i>Retire, Write-Back</i>), w której następuje zapisanie wyniku w miejscu określonym w kodzie rozkazu oraz ustawienie znaczników rejestru statusowego</li> </ul>  <pre> graph LR     Fetch --&gt; Decode     Decode --&gt; Execute     Execute --&gt; Retire     Retire --&gt; Fetch </pre>
19

Obsługa urządzeń zewnętrznych
<ul style="list-style-type: none"> <li>■ Jednym z zadań procesora jest wymiana informacji z urządzeniami zewnętrznymi (stacja dyskiek, karta graficzna, klawiatura itd.)</li> <li>■ Urządzenie zewnętrzne komunikuje się z układem wejścia-wyjścia za pośrednictwem interfejsu, tworząc wspólnie tzw. urządzenie peryferyjne</li> <li>■ Podstawowymi funkcjami realizowanymi przez układ wejścia-wyjścia jest: <ul style="list-style-type: none"> <li>– komunikacja z urządzeniem zewnętrznym (głównie przez interfejs równoległy lub szeregowy), polegająca na przesyłaniu danych, wysyłaniu do niego rozkazów sterujących i odczytywaniu informacji o jego stanie</li> </ul> </li> </ul>
20

Obsługa urządzeń zewnętrznych
<ul style="list-style-type: none"> <li>– komunikacja z procesorem, obejmująca dekodowanie adresu układu, wymianę danych, przesyłanie rozkazów i informacji o stanie układu</li> <li>– buforowanie i formatowanie danych</li> <li>– wykrywanie i informowanie o błędach</li> </ul> <ul style="list-style-type: none"> <li>■ Przestrzeń adresowa układów wejścia-wyjścia może być: <ul style="list-style-type: none"> <li>– odwzorowana w pamięci (układy wejścia-wyjścia i komórki pamięci współdzielą tę samą przestrzeń adresową)</li> <li>– izolowana (układy wejścia-wyjścia i komórki pamięci korzystają z niezależnych przestrzeni adresowych)</li> </ul> </li> </ul>
21

Obsługa urządzeń zewnętrznych
<ul style="list-style-type: none"> <li>■ Do typowych układów wejścia-wyjścia zaliczamy: <ul style="list-style-type: none"> <li>– układy szeregowej i równoległej transmisji danych</li> <li>– timery (układy pomiaru czasu, zliczania lub generowania zdarzeń)</li> <li>– zegary czasu rzeczywistego</li> <li>– układy przetworników analogowo-cyfrowych i cyfrowo-analogowych</li> </ul> </li> <li>■ Sterowanie pracą urządzenia zewnętrznego polega na zaprogramowaniu sterującego nim układu wejścia-wyjścia, czyli wpisaniu do odpowiadającego mu portu określonej wartości, która wymusza takie a nie inne zachowanie urządzenia</li> </ul>
22

Obsługa urządzeń zewnętrznych
<ul style="list-style-type: none"> <li>■ Przesyłanie danych między urządzeniem peryferyjnym a pamięcią operacyjną może odbywać się w trybie programowanego wejścia-wyjścia PIO (<i>Programmed Input/Output</i>) lub w trybie bezpośredniego dostępu do pamięci DMA (<i>Direct Memory Access</i>)</li> <li>■ W trybie programowanego wejścia-wyjścia całym procesem przesyłania danych zajmuje się procesor</li> <li>■ W trybie bezpośredniego dostępu do pamięci procesor przechodzi w stan wysokiej impedancji, a na czas transmisji danych kontrolę nad magistralami komputera przejmuje układ DMA</li> </ul>
23

Obsługa urządzeń zewnętrznych
<ul style="list-style-type: none"> <li>■ Urządzenie może poinformować procesor o zmianie swojego stanu zgłaszając tzw. przerwanie (<i>interrupt</i>)</li> <li>■ Typową sytuacją powodującą zgłoszenie przerwania jest zajście zdarzenia zewnętrznego lub zakończenie zadania „złego” urządzeniu przez procesor</li> <li>■ Zgłoszenie przerwania przez urządzenie powoduje zawieszenie wykonywania bieżącego programu, wykonanie procedury obsługi przerwania a następnie powrót do wykonywania bieżącego programu</li> <li>■ Alternatywną metodą sprawdzania stanu urządzenia jest jego cykliczne przepytanie przez procesor w oczekiwaniu na spełnienie warunków koniecznych do realizacji transmisji (<i>polling</i>)</li> </ul>
24

Podstawowe własności i klasyfikacje pamięci
<ul style="list-style-type: none"> <li>W typowym systemie komputerowym można zauważyć hierarchię podsystemów pamięci – niektóre z nich umiejscowione są bezpośrednio w procesorze, inne tworzą główną pamięć operacyjną, jeszcze inne są podłączane do urządzeń wejścia-wyjścia stanowiąc pamięć pomocniczą</li> <li>Podstawową własnością pamięci jest jej <u>pojemność</u> wyrażana najczęściej w bajtach lub w słowach</li> <li>Innym parametrem związanym z pojemnością jest tzw. <u>jednostka transferu</u> (<i>transfer unit</i>) będąca liczbą bitów jednocześnie odczytywanych z pamięci lub do niej zapisywanych</li> </ul>

25

Podstawowe własności i klasyfikacje pamięci
<ul style="list-style-type: none"> <li>W zależności od <u>sposobu dostępu</u> pamięci możemy podzielić na: <ul style="list-style-type: none"> <li>pamięci o dostępie sekwencyjnym, w których dostęp do danych (najczęściej w postaci rekordów) jest możliwy w określonej sekwencji liniowej (przykład: pamięć taśmowa)</li> <li>pamięci o dostępie bezpośrednim, w których możliwy jest bezpośredni dostęp do bloku danych (dzięki mechanizmowi adresacji bloku), natomiast dostęp do konkretnej danej w bloku odbywa się sekwencyjnie (przykład: pamięć dyskowa)</li> <li>pamięci o dostępie swobodnym, w których poszczególne dane mają swój unikatowy adres i dostęp do nich nie zależy od sekwencji poprzednio wykonanych operacji dostępu (przykład: pamięć operacyjna)</li> </ul> </li> </ul>

26

Podstawowe własności i klasyfikacje pamięci
<ul style="list-style-type: none"> <li>pamięci o dostępie skojarzeniowym, będący rodzajem dostępu swobodnego, w których adresowanie danych odbywa się na zasadzie badania zgodności (porównywania) wybranych bitów słowa z zawartością pamięci (przykład: pamięć podręczna)</li> <li>Kolejną istotną własnością pamięci jest jej <u>wydajność</u>, określana m.in. przez następujące parametry: <ul style="list-style-type: none"> <li>czas dostępu – czas od chwili doprowadzenia adresu do momentu udostępnienia danych</li> <li>czas cyklu pamięci – czas jaki musi upłynąć między dwiema operacjami dostępu do pamięci</li> <li>szybkość przesyłania – szybkość z jaką dane mogą być wprowadzane do pamięci lub z niej wyprowadzane</li> </ul> </li> </ul>

27

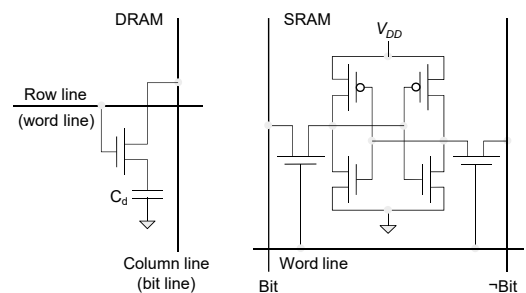
Podstawowe własności i klasyfikacje pamięci
<ul style="list-style-type: none"> <li>W zależności od <u>technologii wykonania</u> pamięci możemy podzielić na półprzewodnikowe, optyczne i magnetyczne</li> <li>Pamięci półprzewodnikowe możemy podzielić na następujące typy: <ul style="list-style-type: none"> <li>pamięci o dostępie swobodnym RAM (<i>Random Access Memory</i>)</li> <li>nieulotne pamięci o dostępie swobodnym NVRAM (<i>Non-Volatile RAM</i>)</li> <li>pamięci stałe ROM (<i>Read Only Memory</i>)</li> <li>programowalne pamięci stałe PROM (<i>Programmable ROM</i>)</li> <li>wymazywalne pamięci stałe EPROM (<i>Erasable and Programmable ROM</i>)</li> </ul> </li> </ul>

28

Podstawowe własności i klasyfikacje pamięci
<ul style="list-style-type: none"> <li>elektrycznie wymazywalne pamięci stałe EEPROM (<i>Electrically Erasable and Programmable ROM</i>)</li> <li>pamięci błyskawiczne (błyskowe) FLASH</li> <li>Pamięci o dostępie swobodnym RAM można podzielić na: <ul style="list-style-type: none"> <li>pamięci statyczne SRAM (<i>Static RAM</i>), w których komórki pamięci zbudowane są z dwustanowych przerzutników (konstruowanych z bramek logicznych)</li> <li>pamięci dynamiczne DRAM (<i>Dynamic RAM</i>), w których komórki pamięci zbudowane są z układów przechowujących ładunek elektryczny (konstruowanych z kondensatora i tranzystora) i wymagających okresowego odświeżania ładunku w celu zachowania danych</li> </ul> </li> </ul>

29

Podstawowe własności i klasyfikacje pamięci
<ul style="list-style-type: none"> <li>Schematyczny diagram komórek pamięci RAM:</li> </ul>



30

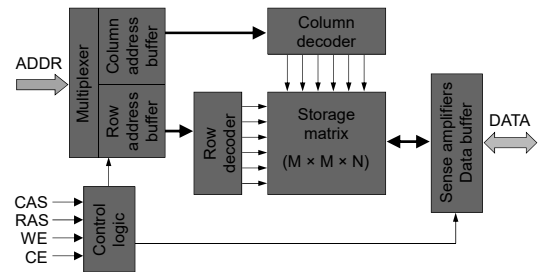
## Pamięć operacyjna

- Pamięć operacyjna typowego komputera zbudowana jest z układów pamięci DRAM
- W układzie pamięci DRAM można wyróżnić m.in. następujące linie sygnałowe:
  - adresy słowa  $A_0 \div A_{m-1}$  (m linii adresowych pozwalających na zaadresowanie maksymalnie  $2^{2^m}$  słów)
  - dane słowa  $D_0 \div D_{n-1}$  (n linii danych)
  - uaktywnienie układu CE (*Chip Enable*)
  - zezwolenie na zapis WE (*Write Enable*)
  - zezwolenie na odczyt OE (*Output Enable*)
  - wybór adresu wiersza RAS (*Row Address Strobe*)
  - wybór adresu kolumny CAS (*Column Address Strobe*)

31

## Pamięć operacyjna

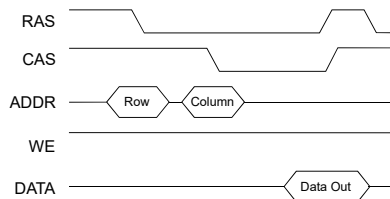
- Organizacja wewnętrzna układu pamięci DRAM:



32

## Pamięć operacyjna

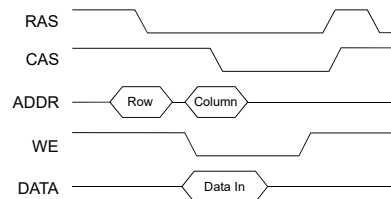
- Cykl odczytu pamięci DRAM działającej w trybie PM (*Page Mode*):



33

## Pamięć operacyjna

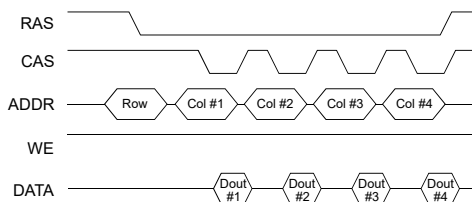
- Cykl zapisu pamięci DRAM działającej w trybie PM (*Page Mode*):



34

## Pamięć operacyjna

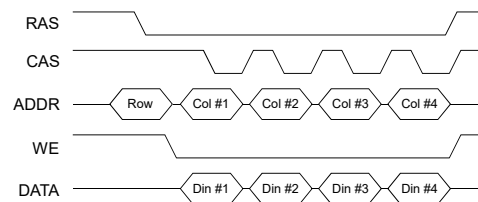
- Cykl odczytu pamięci DRAM działającej w trybie FPM (*Fast Page Mode*):



35

## Pamięć operacyjna

- Cykl zapisu pamięci DRAM działającej w trybie FPM (*Fast Page Mode*):



36

Pamięć operacyjna	
■	Cykl odczytu pamięci DRAM działającej w trybie pakietowym ( <i>Burst Mode</i> ):
	37

Pamięć operacyjna	
■	Cykl zapisu pamięci DRAM działającej w trybie pakietowym ( <i>Burst Mode</i> ):
	38

Pamięć operacyjna	
■	Cechą charakterystyczną układów pamięci DRAM, wynikającą z zasady jej działania jest konieczność okresowego odświeżania komórek pamięci tj. uzupełniania ładunków elektrycznych w tych komórkach pamięci, które przechowują wartość logiczną „1”
■	W celu prawidłowego funkcjonowania mechanizmu odświeżania konieczne jest cykliczne pobudzenie (adresowanie) poszczególnych wierszy matrycy pamięci z częstotliwością nie mniejszą, niż to wynika z tzw. czasu cyklu odświeżania ( <i>Refresh Time</i> )
	39

Pamięć operacyjna	
■	Operacja odświeżania pamięci DRAM realizowana jest przez układy logiczne będące albo elementem systemu komputerowego (kontroler pamięci) albo stanowiące integralną część samego układu pamięci
■	Istnieją cztery podstawowe sposoby odświeżania pamięci DRAM: <ul style="list-style-type: none"> <li>– odświeżanie sygnałem RAS (<i>RAS Only</i>)</li> <li>– odświeżanie CAS przed RAS (<i>CBR, CAS-before-RAS</i>)</li> <li>– odświeżanie ukryte (<i>Hidden</i>)</li> <li>– autoodświeżanie (<i>Self-refresh</i>)</li> </ul>
	40

Pamięć operacyjna	
■	Cykl odświeżania sygnałem RAS ( <i>RAS-Only</i> ) pamięci DRAM:
	41

Pamięć operacyjna	
■	Podstawowym, stosowanym obecnie typem pamięci dynamicznej jest pamięć SDRAM ( <i>Synchronous DRAM</i> ), która różni się od typowej pamięci DRAM sposobem sterowania matrycą pamięci oraz techniką dostępu
■	Cechą charakterystyczną układu pamięci DRAM jest <u>synchronizowanie</u> wszystkich sygnałów sterujących specjalnym przebiegiem zegarowym (CK), co ułatwia współpracę z magistralami systemu komputerowego i pozwala na zwiększenie szybkości jej działania
■	Pamięć SDRAM jest optymalizowana do realizacji przesyłania pakietowego, ale dostęp do niej jest także możliwy w trybie pojedynczych cykli
	42

Pamięć operacyjna
<ul style="list-style-type: none"> <li>■ Układ pamięci SDRAM zbudowany jest z niezależnych od siebie banków (najczęściej dwóch lub czterech), które adresowane są naprzemiennie, co przyspiesza proces dostępu do danych</li> <li>■ W układ pamięci SDRAM wbudowany jest licznik odświeżania i układy logiczne, które pozwalają na jego autonomiczne odświeżanie</li> <li>■ Przed użyciem pamięć SDRAM wymaga zaprogramowania przez wpisanie do specjalnego rejestru (<i>Mode Register</i>) informacji konfiguracyjnych m.in. parametrów czasowych, liczby przesyłanych słów w pojedynczym cyklu pakietowym</li> </ul>

43

Pamięć operacyjna
<ul style="list-style-type: none"> <li>■ W celu uzyskania określonej liczby jednorazowo przesyłanych bitów danych układy pamięci łączone są w tzw. moduły (np. SIMM, DIMM)</li> <li>■ W przypadku modułów DIMM informacje o jego strukturze wewnętrznej niezbędne do prawidłowego zaprogramowania układów SDRAM pamiętane są w niewielkiej (256 B) pamięci stałej SPD (<i>Serial Presence Detect</i>)</li> <li>■ W przypadku konstrukcji modułów złożonych z dużej liczby układów pamięci może okazać się konieczne buforowanie sygnałów sterujących modulem</li> </ul>

44

Pamięć operacyjna
<ul style="list-style-type: none"> <li>■ Odmianą pamięci SDRAM są pamięci DDR (<i>Double Data Rate</i>), które w każdym taktie zegara przekazują dwa słowa danych</li> <li>■ Do układu pamięci DDR przekazywane są dwa komplementarne przebiegi zegarowe (CLK/<math>\bar{\text{CLK}}</math>), których przecięcie wyznacza moment kluczowania</li> <li>■ Pamięć DDR wykorzystuje dodatkowo dwukierunkowy sygnał strobuujący DQS, który w cyklu odczytu generowany jest przez układ pamięci, natomiast w cyklu zapisu przez kontroler pamięci</li> <li>■ Dane są odczytywane z pamięci w środku impulsu DQS, a zapisywane do niej na zboczach tego impulsu</li> </ul>

45

Pamięć operacyjna
<ul style="list-style-type: none"> <li>■ Innym typem stosowanych obecnie układów pamięci są pamięci RDRAM (<i>Rambus DRAM</i>), które do komunikacji z otoczeniem wykorzystują dedykowaną magistralę Rambus przesyłającą dane w trybie pakietowym</li> <li>■ Pojedynczy kanał magistrali Rambus pozwala na dołączenie maksymalnie 32 układów pamięci</li> <li>■ Magistrala Rambus jest bardzo szybka (taktowanie 400 MHz) magistralą o szerokości 16 bitów danych (podzielonych na dwie 8-bitowe szyny: DQA i DQB), co pozwala na transmisję danych z szybkością 1.6 GB/s</li> </ul>

46

Pamięć operacyjna
<ul style="list-style-type: none"> <li>■ Informacja w pamięci RDRAM przechowywana jest w typowych blokach pamięci SDRAM zorganizowanych w niezależne banki (zwykle od 4 do 32) o szerokości najczęściej 128 bitów</li> <li>■ Układy logiczne pamięci RDRAM przesyłają takie porcje danych z/do magistrali Rambus wykorzystując bardzo szybkie multipleksery i demultipleksery</li> <li>■ Do synchronizacji przesyłanych danych magistrala Rambus wykorzystuje dwie pary komplementarnych sygnałów zegarowych CTM (<i>Clock to Master</i>) i CFM (<i>Clock from Master</i>)</li> </ul>

47

Pamięć operacyjna
<ul style="list-style-type: none"> <li>■ Dane są odczytywane z układu pamięci w trakcie narastającego i opadającego zbocza sygnału CTM, a zapisywane do niego w trakcie narastającego i opadającego zbocza sygnału CFM</li> <li>■ Układ pamięci RDRAM wyposażony jest w 8-bitową magistralę adresową, w której 3 bity przypadają na adres wiersza, a 5 bitów na adres kolumny</li> <li>■ Przed użyciem układ pamięci RDRAM wymaga zaprogramowania tj. wpisania do jego wewnętrznych rejestrów sterujących odpowiednich informacji konfiguracyjnych (przesłanych z wykorzystaniem linii sterujących magistrali Rambus)</li> </ul>

48

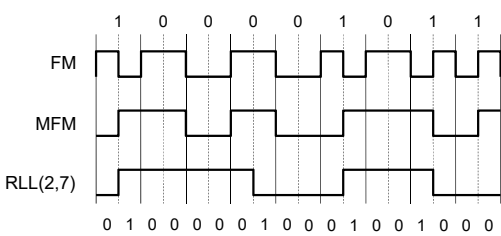


	Pamięć operacyjna
	<ul style="list-style-type: none"> <li>■ Z uwagi na możliwość wystąpienia przekłamań układy pamięci i ich kontrolery wyposaża się dodatkowo w mechanizmy, których celem jest wykrycie albo wykrycie i korekcja błędów</li> <li>■ Najprostszym stosowanym mechanizmem wykrywającym błędy jest tzw. kontrola parzystości</li> <li>■ Realizacja kontroli parzystości wymaga zapamiętania dodatkowego bitu, którego wartość dobierana jest tak, aby liczba „jedynek” w słowie danych była parzysta (operacja XOR)</li> <li>■ Kontrola parzystości pozwala na wykrycie dowolnej nieparzystej liczby błędnych bitów danych</li> </ul>
	49

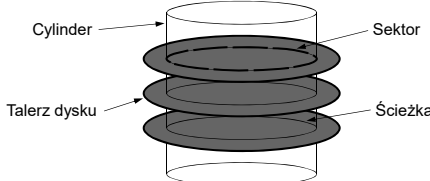
	Pamięć operacyjna
	<ul style="list-style-type: none"> <li>■ Bardziej zaawansowanym mechanizmem kontrolnym są tzw. kody korekcyjne ECC (<i>Error Correction Code</i>) bazujące głównie na tzw. kodowaniu Hamminga</li> <li>■ Najprostsza wersja kodu Hamminga polega na generowaniu bitów parzystości dla odpowiednio dobranych grup bitów danych</li> <li>■ Przykładowo korekcja jednego bitu w słowie 64-bitowym wymaga dodatkowych 7 bitów kontrolnych</li> <li>■ Stosowane obecnie 8-bitowy kod ECC w modułach pamięci DIMM pozwala na wykrycie przekłamań dowolnych dwóch bitów oraz na korekcję dowolnego pojedynczego bitu w 64-bitowym słowie danych</li> </ul>
	50

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Nośnikiem w pamięciach dyskowych jest krążek (dysk) z tworzywa pokrytego warstwą magnetyczną</li> <li>■ W niewielkiej odległości nad/pod obracającym się dyskiem umieszczona jest na ruchomym ramieniu głowica odczytująco-zapisująca</li> <li>■ W trakcie zapisu przepływający przez głowicę prąd elektryczny wytwarza pole magnetyczne trwale namagnesowujące przemieszczający się nad/pod głowicą nośnik</li> <li>■ W trakcie odczytu przemieszczający się nad/pod głowicą nośnik wytwarza zmienne pole magnetyczne wzbudzające w głowicy prąd elektryczny</li> </ul>
	51

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Ruch obrotowy dysku pozwala na dostęp do informacji zlokalizowanej w dowolnym miejscu koncentrycznego okręgu czyli pierścienia na dysku</li> <li>■ Dostęp do kolejnej „porcji danych” (kolejnego pierścienia na dysku) wymaga przesunięcia ramienia, na którym zamocowana jest głowica czyli tzw. pozycjonowania głowicy</li> <li>■ Aby w prawidłowy sposób odtworzyć zapisane dane (zinterpretować impulsy z głowicy odczytującej) zapisywana na nośniku informacja użyteczna musi zostać „wzbogacona” o informację synchronizującą wynikającą z przyjętego <u>sposobu kodowania</u></li> </ul>
	52

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Przegląd sposobów kodowania informacji:</li> </ul>  <p>Diagram illustrating three digital signal waveforms (FM, MFM, RLL(2,7)) for the data sequence 1 0 0 0 0 1 0 1 1. The RLL(2,7) waveform is shown with a label 'Kody RLL' and an arrow pointing to it.</p>
	53

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Nośnik w pamięci dyskowej (dysk) może być niewymienny (np. w dysku twardym) lub wymienny (<i>removable</i>) (np. w stacji dyskietek)</li> <li>■ Warstwa magnetyczna może być umieszczona po obu stronach nośnika (dysk dwustronny) lub tylko po jednej stronie (dysk jednostronny)</li> <li>■ W pamięci dyskowej może znajdować się jeden dysk lub ich większa liczba zamocowana na wspólnej osi</li> <li>■ Mechanizm głowicy może być kontaktowy, mieć ustaloną odległość (przerwę) od nośnika lub też odległość ta może być zmienna w wyniku oddziaływań aerodynamicznych</li> </ul>
	54

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Przyjęta konstrukcja pamięci dyskowej powoduje, że dane organizowane są <u>cylindry</u> będące zbiorem koncentrycznych okręgów o tym samym promieniu zwanych <u>ścieżkami</u>, które dzielone są na mniejsze fragmenty (bloki danych) zwane <u>sektorami</u></li> </ul> 
	55

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Początek każdej ścieżki oznaczony jest: <ul style="list-style-type: none"> <li>– ciągiem bajtów synchronizujących (<i>SYNC</i>)</li> <li>– znacznikiem adresowym indeksu (<i>IAM</i>)</li> </ul> </li> <li>■ W każdym sektorze można wyszczególnić: <ul style="list-style-type: none"> <li>– ciąg bajtów synchronizujących (<i>SYNC</i>)</li> <li>– znacznik adresowy (<i>IDAM</i>)</li> <li>– identyfikator (adres) sektora (<i>ID</i>)</li> <li>– znacznik adresowy danych (<i>DAM</i>)</li> <li>– blok danych sektora</li> <li>– pola kodu ECC lub CRC (<i>Cyclic Redundancy Check</i>)</li> </ul> </li> <li>■ Znacznik początku ścieżki oraz poszczególne sektory rozdzielone są od siebie przerwami (<i>GAP</i>)</li> </ul>
	56

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Adres fizyczny sektora na dysku (tzw. adres CHS) określają trzy parametry: <ul style="list-style-type: none"> <li>– numer cylindra (<i>Cylinder</i>)</li> <li>– numer głowicy (<i>Head</i>)</li> <li>– numer sektora (<i>Sektor</i>)</li> </ul> </li> <li>■ Czas dostępu do danych na dysku zależy od: <ul style="list-style-type: none"> <li>– czasu przeszukiwania (<i>seek time</i>) związanego z pozycjonowaniem głowicy nad ścieżką</li> <li>– czasu opóźnienia obrotowego (<i>rotational latency</i>) związanego z przemieszczeniem się głowicy nad sektor</li> </ul> </li> <li>■ Gęstość zapisu danych na dysku zależy od: <ul style="list-style-type: none"> <li>– szerokości ścieżki</li> <li>– przyjęty sposób kodowania danych</li> </ul> </li> </ul>
	57

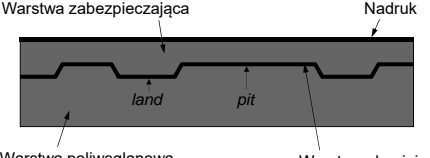
	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ W celu poprawy wydajności podsystemu dyskowego, a w szczególności zwiększenia jego niezawodności i szybkości działania stosuje się macierze dyskowe RAID (<i>Redundant Array of Independent Disks</i>)</li> <li>■ Macierz RAID zbudowana jest z wielu połączonych ze sobą dysków fizycznych, które widoczne są przez system operacyjny jako jeden dysk logiczny</li> <li>■ Dane zapisywane w macierzy rozpraszane są na wiele dysków, co gwarantuje odzyskanie danych w przypadku uszkodzenia dysku</li> <li>■ W praktyce stosuje się różne poziomy RAID oznaczane zwykle jako RAID 0 do RAID 5</li> </ul>
	58

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Poziom RAID 0 (<i>striping</i>) <ul style="list-style-type: none"> <li>– dane dzielone są na tzw. paski, które rozmieszczane są cyklicznie na kolejnych dyskach macierzy</li> <li>– brak redundancji danych</li> <li>– duża szybkość odczytu i zapisu danych</li> <li>– pojemność macierzy jest równa sumie pojemności wszystkich dysków wchodzących w jej skład</li> </ul> </li> <li>■ Poziom RAID 1 (<i>mirroring</i>) <ul style="list-style-type: none"> <li>– dane są paskowane i następnie odwzorowywane na dwóch odrębnych dyskach fizycznych</li> <li>– średnia szybkość odczytu, mała szybkość zapisu danych</li> <li>– pojemność macierzy jest równa połowie pojemności wszystkich dysków wchodzących w jej skład</li> </ul> </li> </ul>
	59

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Poziom RAID 2 <ul style="list-style-type: none"> <li>– dane są paskowane (na paski o rozmiarze mniejszym niż w RAID 0 i RAID 1) i rozmieszczane cyklicznie na kolejnych dyskach danych macierzy</li> <li>– na pozostałych dyskach (tzw. dyskach redundancyjnych) umieszczone zostają informacje korekcyjne, utworzone w oparciu o kod Hamminga na podstawie odpowiednich bitów z każdego dysku danych</li> <li>– liczba dysków redundancyjnych jest proporcjonalna do logarytmu liczby dysków danych</li> <li>– duża szybkość odczytu i zapisu (równoległy dostęp do wielu dysków)</li> <li>– bardzo wysoki koszt</li> </ul> </li> </ul>
	60

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>■ Poziom RAID 3 <ul style="list-style-type: none"> <li>– dane są paskowane (na paski o podobnym rozmiarze jak w RAID 2) i rozmieszczane cyklicznie na kolejnych dyskach danych macierzy</li> <li>– na dysku redundancyjnym (tylko jednym niezależnie od rozmiarów macierzy) umieszczone zostają informacje korekcyjne, utworzone w oparciu o bity parzystości wyliczone dla odpowiednich bitów z każdego dysku danych</li> <li>– duża szybkość odczytu i zapisu (równoległy dostęp do wielu dysków)</li> </ul> </li> <li>■ Poziom RAID 4 <ul style="list-style-type: none"> <li>– dane są paskowane (na paski o dużym rozmiarze) i rozmieszczane cyklicznie na kolejnych dyskach danych macierzy</li> </ul> </li> </ul>
	61

	Pamięć dyskowa
	<ul style="list-style-type: none"> <li>– na dysku redundancyjnym (tylko jednym niezależnie od rozmiarów macierzy) umieszczone zostają paski informacji korekcyjnych, utworzone w oparciu o bity parzystości wyliczone dla odpowiednich pasków z każdego dysku danych</li> <li>– dyski działają niezależnie od siebie, co pozwala na obsługę wielu żądań wejścia-wyjścia równolegle</li> <li>– średnia szybkość odczytu, mała szybkość zapisu</li> <li>■ Poziom RAID 5 (<i>striping with parity</i>) <ul style="list-style-type: none"> <li>– organizacja zbliżona do RAID 4</li> <li>– paski parzystości są rozproszone na wszystkie dyski w macierzy</li> <li>– średnia szybkość odczytu i zapisu, ale lepsza niż w RAID 1</li> </ul> </li> </ul>
	62

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ W dyskach CD-ROM (<i>Compact Disk ROM</i>) nośnikiem jest przezroczysta warstwa poliwęglanowa z wytłoczonymi wgłębieniami, która pokryta jest cienką warstwą aluminium i zabezpieczona od góry powłoką lakieru</li> <li>■ Przekrój poprzeczny nośnika CD-ROM:</li> </ul>
	
	63

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ Promień lasera odbijając się od warstwy aluminium w obszarze zwanym landem ulega mniejszemu rozproszeniu niż odbijając się w obszarze zwanym pitem</li> <li>■ Po odbiciu od nośnika promień lasera pada przez półprzepuszczalny pryzmat i układ soczewek na światłoczułe detektory, których zadaniem jest: <ul style="list-style-type: none"> <li>– przetworzenie zmian natężenia padającego światła na sygnał elektryczny (sygnał danych)</li> <li>– wytworzenie sygnałów sterujących układem ogniskowania wiązki</li> <li>– wytworzenie sygnałów sterujących układem pozycjonowania wiązki</li> </ul> </li> </ul>
	64

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ Podczas odczytu promień lasera przemieszcza się nad spiralną ścieżką z szybkością zależną od prędkości wirowania płyty</li> <li>■ W praktyce stosuje się dwie metody odczytu danych: <ul style="list-style-type: none"> <li>– ze stałą prędkością liniową CLV (Constant Linear Velocity)</li> <li>– ze stałą prędkością kątową CAV (Constant Angular Velocity)</li> </ul> </li> <li>■ Informacje na płycie CD-ROM kodowane są metodą EFM (<i>Eight to Fourteen Modulation</i>), w której każdy zapisywany bajt zastępowany jest ciągiem 14 bitów dobranym w taki sposób, aby dwie kolejne jedyńki były rozdzielone nie mniej niż dwoma i nie więcej niż dziesięcioma zerami (co oznacza 2 pity na bajt)</li> </ul>
	65

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ Dane na płycie CD-ROM zorganizowane są fizycznie w ramki o następującej strukturze: <ul style="list-style-type: none"> <li>– ciąg 27 bitów synchronizujących</li> <li>– bajt sterujący</li> <li>– dwa 16-bajtowe bloki (każdy złożony z ciągu 12 bajtów danych i 4 bajtów kontrolnych)</li> </ul> </li> <li>■ Grupa połączonych 98 ramek tworzy logiczny sektor, na który składa się: <ul style="list-style-type: none"> <li>– ciąg 12 bajtów synchronizujących</li> <li>– 4 bajtowy nagłówek</li> <li>– blok 2048 bajtów danych</li> <li>– blok 288 bajtów kontrolno-korekcyjnych</li> </ul> </li> </ul>
	66

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ Pojedynczy sektor zawiera 2048 bajtów danych zapisanych w trybie 1 (z korekcją błędów) lub 2336 bajtów danych zapisanych w trybie 2 (bez korekcji błędów)</li> <li>■ Bloki bajtów kontrolno-korekcyjnych połączone z odpowiednim rozlokowaniem (przeplotem) kolejnych ramek umożliwiają korekcję błędów metodą CIRC (<i>Cross-Interleave Reed-Solomon Coding</i>), która pozwala na skorygowanie ciągu około 450 błędnych bajtów danych</li> <li>■ Typowa płyta CD-ROM o średnicy 12 centymetrów może pomieścić 650-700MB danych</li> </ul>

67

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ Poza płytami CD-ROM przeznaczonymi wyłącznie do odczytu coraz większe znaczenie mają nośniki pozwalające na jednokrotny (CD-R) lub wielokrotny (CD-RW) zapis danych</li> <li>■ Warstwa poliwęglanowa w tego typu płytach pokryta jest specjalną substancją, która mętnieje (zmienia postać z krystalicznej na amorficzną) pod wpływem naświetlania promieniem laserem o zwiększonej mocy (co w konsekwencji powoduje zmniejszenie natężenia wiązki odbitej w trakcie odczytu)</li> <li>■ Substancja stosowana w płytach CD-RW może na powrót zmienić postać na krystaliczną</li> </ul>

68

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ Powszechnie stosowanym rodzajem pamięci optycznej o dużej pojemności przeznaczonym do odczytu są płyty DVD-ROM (<i>Digital Versatile Disk ROM</i>)</li> <li>■ Dzięki zastosowaniu lasera o mniejszej długości fali (co pozwoliło dwukrotnie zwiększyć gęstość zapisu i zmniejszyć odległości pomiędzy ścieżkami) oraz dwuwarstwowej struktury nośnika (pierwsza warstwa nośnika jest półprzezroczysta) uzyskano możliwość zapisania wielokrotnie większej ilości informacji niż na płytach CD-ROM, pomimo zastosowania nośnika tej samej wielkości</li> </ul>

69

	Pamięć optyczna
	<ul style="list-style-type: none"> <li>■ W trakcie odczytu wiązka lasera jest ogniskowana odpowiednio na pierwszej lub drugiej warstwie nośnika</li> <li>■ Specyfikacja DVD-ROM dopuszcza stosowanie płyt: <ul style="list-style-type: none"> <li>– jednostronnych jednowarstwowych o pojemności 4.7GB</li> <li>– jednostronnych dwuwarstwowych o pojemności 8.5GB</li> <li>– dwustronnych jednowarstwowych o pojemności 9.4GB</li> <li>– dwustronnych dwuwarstwowych o pojemności 17GB</li> </ul> </li> <li>■ Format danych zapisanych na płycie DVD-ROM jest zbliżony do formatu danych na płycie CD-ROM, lecz z uwagi na zwiększoną gęstość zapisu został ulepszony mechanizm korekcji błędów</li> </ul>

70

	Pamięć magnetoptyczna
	<ul style="list-style-type: none"> <li>■ Nośnikiem informacji w dyskach magnetoptycznych jest materiał, w którym następuje zmiana polaryzacji światła pod wpływem pola magnetycznego</li> <li>■ Odczyt danych polega na oświetleniu nośnika wiązką spolaryzowanego światła o odpowiednio dobranej płaszczyźnie polaryzacji</li> <li>■ Po odbiciu od nośnika płaszczyzna ta ulega skręceniu o kąt zależny od stopnia jego namagnesowania, a następnie przez półprzepuszczalny pryzmat i filtr polaryzacyjny (którego zadaniem jest tłumienie wiązki odbitej od obszaru nienamagnesowanego) trafia na układ detektora</li> </ul>

71

	Pamięć magnetoptyczna
	<ul style="list-style-type: none"> <li>■ Zapis danych polega na wstępnym namagnesowaniu nośnika, a następnie na jego oświetleniu wiązką światła laserowego o dużej energii, co powoduje miejscowe podgrzanie nośnika powyżej tzw. punktu Curie i w konsekwencji jego roznamagnesowanie</li> <li>■ Typowe dyski magnetoptyczne są najczęściej wymienne i mają pojemność rzędu kilkuset MB</li> <li>■ Wadą tego typu pamięci jest wysoki koszt zarówno nośnika jak i samego napędu</li> </ul>

72

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>■ Zadaniem interfejsu EIDE (<i>Enhanced Integrated Device Electronics</i>) będącego rozszerzeniem interfejsu IDE jest podłączenie do systemu komputerowego takich urządzeń zewnętrznych jak dyski twarde, napędy optyczne i magnetoptyczne</li> <li>■ Cechą charakterystyczną obsługiwanych przez ten interfejs urządzeń jest to, że zintegrowano w nich mechanizm napędu wraz z układem sterującym (kontrolerem) odpowiedzialnym m.in. za pozycjonowanie głowic, kodowanie/dekodowanie sygnału, kontrolę i korekcję błędów, translację adresów oraz przesyłanie danych</li> </ul>

73

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>■ Standardy interfejsu IDE (AT-BUS): <ul style="list-style-type: none"> <li>– 1994: Standard ATA-1 (<i>Advanced Technology Attachment, AT Attachment</i>) <ul style="list-style-type: none"> <li>■ transmisja danych w trybie PIO</li> <li>■ prędkość transmisji do 4 MB/s</li> <li>■ obsługa najwyżej dwóch dysków twardych</li> </ul> </li> <li>– 1996: Standard ATA-2 <ul style="list-style-type: none"> <li>■ transmisja danych w trybach PIO i DMA</li> <li>■ prędkość transmisji do 16.6 MB/s</li> </ul> </li> <li>– 1997: Standard ATA-3 <ul style="list-style-type: none"> <li>■ system autodiagnostyki S.M.A.R.T. (<i>Self Monitoring, Analysis and Reporting Technology</i>)</li> </ul> </li> </ul> </li> </ul>

74

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>– 1998: Interfejs EIDE (Fast ATA), standard ATA/ATAPI-4 <ul style="list-style-type: none"> <li>■ transmisja danych w trybach PIO, DMA i Ultra-DMA</li> <li>■ prędkość transmisji do 33 MB/s</li> <li>■ interfejs logiczny ATAPI (<i>AT Attachment Packet Interface</i>) pozwalający na obsługę takich urządzeń jak napędy CD-ROM, streamery itp.</li> <li>■ obsługa do czterech urządzeń zewnętrznych</li> </ul> </li> <li>– 2000: Standard ATA/ATAPI-5 (Ultra ATA/100) <ul style="list-style-type: none"> <li>■ prędkość transmisji do 100 MB/s</li> </ul> </li> <li>– Standard ATA/ATAPI-6 (Ultra ATA/133) <ul style="list-style-type: none"> <li>■ prędkość transmisji do 133 MB/s</li> <li>■ 48-bitowy model adresowania LBA</li> <li>■ funkcje AAM (<i>Automatic Acoustic Management</i>)</li> </ul> </li> </ul>

75

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>■ Magistrala interfejsu IDE połączona jest z magistralą systemową przez kontroler IDE (<i>IDE Host-Adapter</i>), którego zadaniem jest dekodowanie adresu oraz buforowanie sygnałów na magistralach</li> <li>■ Standardowy interfejs IDE pozwala na dołączenie do kontrolera dwóch urządzeń: nadrzędnego (<i>Master</i>) i podrzędnego (<i>Slave</i>), wykorzystując w tym celu 40-, 50- lub 80-żyłowy przewód o długości do 46 cm</li> <li>■ W przypadku interfejsu EIDE wykorzystuje się dwa kanały IDE: IDE 0 (<i>Primary</i>) oraz IDE 1 (<i>Secondary</i>), dzięki czemu możliwe jest podłączenie do systemu czterech urządzeń</li> </ul>

76

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>■ Kanały kontrolera EIDE są od siebie całkowicie niezależne tzn. do komunikacji z systemem wykorzystują oddzielne przerwania i adresy wejścia-wyjścia rejestrów kontrolno-sterujących CBR (<i>Control Block Registers, Command Block Registers</i>): <ul style="list-style-type: none"> <li>– IDE 0: IRQ 14, Porty 1F0÷1F7, 3F6</li> <li>– IDE 1: IRQ 15, Porty 170÷177, 376</li> </ul> </li> <li>■ Przesyłanie danych pomiędzy pamięcią operacyjną a urządzeniem zewnętrznym może odbywać się w trybach: <ul style="list-style-type: none"> <li>– programowanego wejścia-wyjścia: PIO-0 (3.33 MB/s), PIO-1 (5.22 MB/s), PIO-2 (8.33 MB/s), PIO-3 (11.11 MB/s) lub PIO-4 (16.66 MB/s)</li> </ul> </li> </ul>

77

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>– bezpośredniego dostępu do pamięci: <ul style="list-style-type: none"> <li>■ Singleword DMA 0 (2.08 MB/s), Singleword DMA 1 (4.16 MB/s), Singleword DMA 2 (8.33 MB/s)</li> <li>■ Multiword DMA 0 (4.16 MB/s), Multiword DMA 1 (13.33 MB/s), Multiword DMA 2 (16.66 MB/s)</li> <li>■ Ultra DMA 1 (25.00 MB/s), Ultra DMA 2 (33.33 MB/s), Ultra DMA 3 (44.44 MB/s), Ultra DMA 4 (66.64 MB/s), Ultra DMA 5 (99.96 MB/s), Ultra DMA 6 (133.28 MB/s)</li> </ul> </li> <li>■ Komunikacja z urządzeniami IDE realizowana jest za pomocą rozkazów należących do: <ul style="list-style-type: none"> <li>– zbioru podstawowego (<i>General Feature Set</i>)</li> <li>– zbioru rozkazów trybu pakietowego (<i>Packet Command Feature Set</i>)</li> </ul> </li> </ul>

78

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>■ Specyfikacja ATA/ATAPI pozwala na obsługę urządzeń implementujących podstawowe (<i>Power Management</i>) lub zaawansowane (<i>Advanced Power Management</i>) funkcje zarządzania poborem mocy</li> <li>■ Podstawowy mechanizm zarządzania poborem mocy definiuje cztery możliwe tryby pracy urządzenia: <ul style="list-style-type: none"> <li>– tryb gotowości (<i>Active</i>)</li> <li>– tryb jałowy (<i>Idle</i>)</li> <li>– tryb czuwania (<i>Standby</i>)</li> <li>– tryb uśpienia (<i>Sleep</i>)</li> </ul> </li> <li>■ W trybie zaawansowanym możliwe jest ustawienie jednego z 254 możliwych poziomów poboru energii</li> </ul>

79

Interfejs IDE (EIDE)
<ul style="list-style-type: none"> <li>■ Urządzenia IDE obsługujące protokół ATAPI mogą implementować następujące mechanizmy: <ul style="list-style-type: none"> <li>– System zabezpieczeń (Security Mode) chroniący dostęp do urządzenia mechanizmem dwóch haseł: administratora (<i>master password</i>) i użytkownika (<i>user password</i>)</li> <li>– System autodiagnostyki S.M.A.R.T (<i>Self Monitoring, Analysis and Reporting Technology</i>) pozwalający na bieżącą kontrolę krytycznych parametrów pracy dysku, co zapewnia wczesne wykrywanie problemów i minimalizowanie skutków awarii</li> <li>– System zarządzania parametrami akustycznymi (<i>Automatic Acoustic Management</i>) pozwalający na wybór „kompromisu” pomiędzy szybkością pracy napędu, a emitowanym przez niego poziomem hałasu</li> </ul> </li> </ul>

80

Interfejs Serial ATA (SATA)
<ul style="list-style-type: none"> <li>■ Interfejs szeregowy Serial ATA służy do podłączenia do systemu komputerowego takiej samej klasy urządzeń zewnętrznych jak interfejs EIDE</li> <li>■ Cechy interfejsu Serial ATA: <ul style="list-style-type: none"> <li>– przesyłanie danych z szybkością od 1.5 Gb/s (SATA I)</li> <li>– programowa zgodność ze standardem ATA/ATAPI</li> <li>– niższy pobór mocy (niższe napięcie magistrali)</li> <li>– topologia gwiazdy (każde urządzenie połączone jest z kontrolerem własnym przewodem)</li> <li>– zwiększona odporność na błędy (transmisja dwukierunkowa w systemie różnicowym)</li> <li>– możliwość zmiany konfiguracji przyłączonych urządzeń w trakcie pracy (<i>Hot Plugging</i>)</li> </ul> </li> </ul>

81

Interfejs Serial ATA (SATA)
<ul style="list-style-type: none"> <li>■ Interfejs SATA wykorzystuje 7-żyłowy przewód połączeniowy (cztery linie sygnałowe plus trzy linie masy) o długości mniejszej niż 1m</li> <li>■ Dane fizycznie przesyłane interfejsem Serial ATA zorganizowane są w 32 bitowe słowa, kodowane regułą 8b/10b (NRZ), co zapobiega jednostronnej polaryzacji kabla</li> <li>■ Wymiana informacji pomiędzy kontrolerem Serial ATA i urządzeniem zewnętrznym odbywa się za pomocą: <ul style="list-style-type: none"> <li>– symboli (<i>Primitives</i>)</li> <li>– ramek FIS (<i>Frame Information Structure</i>)</li> </ul> </li> </ul>

82

Interfejs Serial ATA (SATA)
<ul style="list-style-type: none"> <li>■ Symbol jest pojedynczym słowem, którego zadaniem jest przesłanie: <ul style="list-style-type: none"> <li>– informacji o stanie magistrali i urządzenia</li> <li>– sygnatury oznaczającej początek (SOF) lub koniec (EOF) przesyłanej ramki</li> <li>– znaku synchronizującego (SYNC) w stanie spoczynkowym (<i>Idle</i>) magistrali</li> </ul> </li> <li>■ Ramka FIS ma zmienną długość i jest zbudowana z: <ul style="list-style-type: none"> <li>– nagłówka (<i>Header</i>) o długości 1 słowa zawierającego m.in. sygnaturę (<i>FIS Type</i>) i długość ramki</li> <li>– bloku danych użytkowych o długości do 2048 słów</li> <li>– kodu CRC o długości 1 słowa</li> </ul> </li> </ul>

83

Interfejs Serial ATA (SATA)
<ul style="list-style-type: none"> <li>■ Ramki FIS służą do przesyłania danych użytkowych, diagnostycznych i rozkazów programujących rejestry trybu pracy SRB (<i>Shadow Register Block</i>)</li> <li>■ Specyfikacja Serial ATA definiuje symetryczny (z potwierdzeniem) protokół transmisji danych (zestaw reguł opisujących proces przesyłania danych), według którego, każda odebrana ramka wymaga przesłania zwrotnie symbolu potwierdzającego poprawny odbiór ramki (ROK) lub sygnalizującego błąd transmisji (RERR)</li> <li>■ Transfer danych po magistrali Serial ATA może odbywać się zarówno w trybie PIO, jak i DMA</li> </ul>

84

	Interfejs Serial ATA (SATA)
	<ul style="list-style-type: none"> <li>■ Rozwój interfejsu SATA: <ul style="list-style-type: none"> <li>– SATA I: przepustowość 1.5 Gb/s (ok. 180 MB/s)</li> <li>– SATA II: przepustowość 3 Gb/s (ok. 375 MB/s), kolejki zadań NCQ (<i>Native Command Queuing</i>), możliwość przyłączenia wielu urządzeń do jednego portu dzięki powielaczom portów (<i>Port Multipliers</i>), redundancja połączeń dzięki selektorom portów (<i>Port Selectors</i>)</li> <li>– SATA III: przepustowość 6 Gb/s (ok. 750 MB/s), ulepszony mechanizm NCQ, tryb izochorycznego transferu danych</li> <li>– eSATA (external SATA): zewnętrzny port SATA 3Gb/s do podłączania urządzeń zewnętrznych na odległość do 2 m</li> <li>– xSATA: rozwinięcie eSATA do transmisji na odległość do 8 m</li> </ul> </li> </ul>
	85

	Interfejs SCSI
	<ul style="list-style-type: none"> <li>■ Standard SCSI (<i>Small Computer System Interface</i>) powstał w 1982 roku dla potrzeb obsługi urządzeń zewnętrznych (m.in. dysków twardych, napędów taśmowych, czytników CD-ROM, skanerów) przez minikomputery i wydajne stacje robocze</li> <li>■ Magistrala SCSI została opracowana jako szybka, uniwersalna, elastyczna i niezawodna magistrala równoległa obsługująca wiele urządzeń (także różnych producentów) przyłączonych do jednego kontrolera i montowanych zarówno wewnątrz, jak i na zewnątrz obudowy komputera</li> </ul>
	86

	Interfejs SCSI
	<ul style="list-style-type: none"> <li>■ Standardy interfejsu SCSI: <ul style="list-style-type: none"> <li>– Standard SCSI-1 (1986) <ul style="list-style-type: none"> <li>■ magistrala danych o szerokości 8 bitów</li> <li>■ adresowanie do 8 urządzeń</li> <li>■ prędkość transmisji do 5 MB/s</li> <li>■ długość przewodu połączeniowego do 6m</li> </ul> </li> <li>– Standard SCSI-2 (1990) <ul style="list-style-type: none"> <li>■ magistrala danych o szerokości 16 lub 32 bitów (Wide SCSI)</li> <li>■ adresowanie do 16 urządzeń</li> <li>■ szybkość transmisji do 10 MB/s (tryb Fast SCSI), 20 MB/s (tryby Fast Wide SCSI i Ultra SCSI) lub 40 MB/s (tryby Wide Ultra SCSI i Ultra2 SCSI)</li> </ul> </li> </ul> </li> </ul>
	87

	Interfejs SCSI
	<ul style="list-style-type: none"> <li>■ długość przewodu połączeniowego do 25m</li> <li>– Standard Ultra3 (Ultra160) SCSI , Ultra4 (Ultra320) SCSI <ul style="list-style-type: none"> <li>■ wykorzystanie obydwu zboczy sygnału taktującego pozwala na maksymalny transfer 160 MB/s (320 MB/s)</li> <li>■ możliwość korzystania z różnych interfejsów fizycznych pełna kontrola integralności przesyłanych danych kodami CRC (Ultra320 SCSI)</li> <li>■ obsługa protokołu pakietowego (Ultra320 SCSI)</li> <li>■ adresowanie do 32 urządzeń</li> <li>■ wprowadzenie protokołu diagnostycznego jakości łącza</li> <li>■ przyspieszenie arbitrażu w dostępie do magistrali</li> </ul> </li> <li>– Standard Ultra640 SCSI <ul style="list-style-type: none"> <li>■ maksymalny transfer 640 MB/s</li> </ul> </li> </ul>
	88

	Interfejs SCSI
	<ul style="list-style-type: none"> <li>■ Każde urządzenie podłączone do magistrali SCSI ma przydzielony unikalny identyfikator ID (liczba z zakresu 0÷7 lub 0÷15) określający adres i priorytet urządzenia</li> <li>■ Najczęściej urządzeniem o najwyższym priorytecie (identyfikatorze) jest kontroler SCSI (<i>Host Adapter</i>)</li> <li>■ Długość przewodu połączeniowego elektrycznej wersji magistrali SCSI wynosi: <ul style="list-style-type: none"> <li>– do 6m w przypadku magistrali pojedynczej (<i>single-ended</i>)</li> <li>– do 25m w przypadku magistrali różnicowej (<i>differential</i>)</li> <li>– do 12m w przypadku magistrali niskonapięciowej różnicowej LVD (<i>low-voltage differential</i>)</li> </ul> </li> </ul>
	89

	Interfejs SCSI
	<ul style="list-style-type: none"> <li>■ Urządzenia do magistrali przyłączone są łańcuchowo (równolegle), przy czym końce magistrali muszą być wyposażone w terminatory tj. mostki rezystorowe zapewniające dopasowanie falowe</li> <li>■ Szerokość magistrali SCSI zależy od jej realizacji – w przypadku jej elektrycznej wersji (SCSI-2) wynosi odpowiednio: <ul style="list-style-type: none"> <li>– 50 linii sygnałowych dla 8-bitowej magistrali danych</li> <li>– 68 linii sygnałowych dla 16-bitowej magistrali danych</li> </ul> </li> <li>■ Magistrala SCSI pozwala na niezależną komunikację między różnymi parami dołączonych do niej urządzeń w ramach tzw. kanałów logicznych</li> </ul>
	90

Interfejs SCSI
<ul style="list-style-type: none"> <li>■ Specyfikacja SCSI definiuje protokół wymiany informacji przez magistralę, obejmujący następujące fazy pracy: <ul style="list-style-type: none"> <li>– Faza wolnej magistrali (<i>Bus Free</i>), w której żadne urządzenie nie korzysta z magistrali</li> <li>– Faza arbitrażu (<i>Arbitration</i>), w której dokonywany jest wybór urządzenia (inicjującego lub docelowego), które przejmie kontrolę nad magistralą</li> <li>– Faza selekcji (<i>Selection</i>), w której urządzenie inicjujące dokonuje wyboru urządzenia docelowego (przesyłając na magistralę identyfikator swój i urządzenia docelowego)</li> <li>– Faza rozkazowa (<i>Command</i>), w której przesyłane są rozkazy do urządzenia docelowego</li> </ul> </li> </ul>

91

Interfejs SCSI
<ul style="list-style-type: none"> <li>– Faza reselekcji (<i>Reselection</i>), w której następuje nawiązanie połączenia przez urządzenie docelowe z urządzeniem inicjującym (faza taka ma miejsce najczęściej po wykonaniu przez urządzenie docelowe rozkazu, w trakcie wykonania którego magistrala była zwolniona)</li> <li>– Faza przesyłania danych (<i>Data</i>), w której następuje przesyłanie danych między urządzeniami</li> <li>– Faza przesyłania raportu (<i>Status</i>), w której przesyłana jest informacja o sposobie zakończenia wykonywanej operacji</li> <li>– Faza przesyłania wiadomości (<i>Message</i>), w której przesyłane są (dwukierunkowo) informacje pomocnicze</li> <li>– Faza zgłaszania przesłania wiadomości (<i>Attention</i>), w której urządzenie inicjujące zgłasza żądanie wysłania wiadomości</li> <li>– Faza zerowania (<i>Reset</i>), w której następuje zerowanie wszystkich urządzeń na magistrali</li> </ul>

92

Interfejs SCSI
<ul style="list-style-type: none"> <li>■ Przesyłanie danych magistralą SCSI może odbywać się w trybie asynchronicznym lub synchronicznym</li> <li>■ Rozkazy SCSI przesyłane są w tzw. blokach CDB (<i>Command Data Block</i>) o długości 6, 10 lub 12 bajtów, zawierających odpowiednio: <ul style="list-style-type: none"> <li>– kod operacji</li> <li>– parametry rozkazu</li> <li>– bajt kontrolny</li> </ul> </li> <li>■ Urządzenie docelowe wyposażone jest najczęściej w kolejkę rozkazów, dzięki której możliwe jest bardziej efektywne wykorzystanie magistrali</li> </ul>

93

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Interfejs IEEE-1284 jest standardem przesyłania informacji między urządzeniami za pośrednictwem łącza równoległego</li> <li>■ Specyfikacja standardu IEEE-1284 definiuje pięć trybów pracy łącza równoległego: <ul style="list-style-type: none"> <li>– tryb kompatybilny (<i>Compatibility Mode</i>)</li> <li>– tryb półbajtowy (<i>Nibble Mode</i>)</li> <li>– tryb bajtowy PS/2 (<i>Byte Mode</i>)</li> <li>– tryb EPP (<i>Enhanced Parallel Port</i>)</li> <li>– tryb ECP (<i>Extended Capabilities Port</i>)</li> </ul> </li> <li>■ Pierwsze trzy z wymienionych trybów pracy określa się także jako tryb SPP (<i>Standard Parallel Port</i>)</li> </ul>

94

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Interfejs IEEE-1284 pracujący w trybie SPP wykorzystuje oprócz 8-bitowej magistrali danych, następujące linie sygnałowe: <ul style="list-style-type: none"> <li>– wejściowy sygnał błędu drukarki: ERR (<i>Error</i>)</li> <li>– wyjściowy sygnał strobujący dane: STR (<i>Strobe</i>)</li> <li>– wejściowy sygnał potwierdzenia: ACK (<i>Acknowledge</i>)</li> <li>– wejściowy sygnał zajętości: BSY (<i>Busy</i>)</li> <li>– wejściowy sygnał braku papieru: PERR (<i>Paper Error</i>)</li> <li>– wejściowy sygnał gotowości (stanu „on-line”): SEL (<i>Select</i>)</li> <li>– wyjściowy sygnał wysuwu wiersza: ALF (<i>Auto Line Feed</i>)</li> <li>– wyjściowy sygnał zerowania (inicjowania) drukarki: INI (<i>Initialization</i>)</li> <li>– wyjściowy sygnał wyboru drukarki: DSL (<i>Device Select</i>)</li> </ul> </li> </ul>

95

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Tryb kompatybilny zwany także standardowym lub trybem Centronics realizuje jednokierunkową transmisję danych z prędkością do 200 kB/s</li> <li>■ Tryb półbajtowy pozwala na dwukierunkową transmisję danych, w której do odbierania danych wykorzystuje się linie BSY, ACK, PERR i SEL połączone z magistralą danych przez multiplexer sterowany sygnałem STR</li> <li>■ Tryb bajtowy realizuje dwukierunkową transmisję danych opartą na wykorzystaniu dwukierunkowego portu danych (<i>Bi-Directional Data Port</i>)</li> </ul>

96



Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Dostęp do łącza równoległego pracującego w trybie SPP odbywa się za pośrednictwem: <ul style="list-style-type: none"> <li>– rejestru danych (<i>Data Register</i>), umożliwiającego zapis/odczyt danych z magistrali danych interfejsu</li> <li>– rejestru stanu (<i>Status Register</i>), pozwalającego m.in. na odczyt stanu linii BSY, ACK, PERR, SEL i ERR</li> <li>– rejestru sterującego (<i>Control Register</i>), pozwalającego m.in. na aktywowanie dwukierunkowej pracy portu danych oraz na ustawianie stanu linii DSL, INI, ALF i STR</li> </ul> </li> <li>■ Implementacja protokołów transmisji danych w trybach SPP (polegająca m.in. na wytworzeniu właściwych sekwencji sygnałów sterujących) musi być zrealizowana programowo</li> </ul>

97

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Tryb EPP pozwala na realizację dwukierunkowej transmisji danych z prędkością do 2 MB/s</li> <li>■ Protokół wymiany danych w trybie EPP wykorzystuje następujące linie sygnałowe: <ul style="list-style-type: none"> <li>– sygnał kierunku transmisji: WRITE (linia STR trybu SPP)</li> <li>– sygnał wyzwolenia przerwania: INTERRUPT (linia ACK trybu SPP)</li> <li>– sygnał zakończenia cyklu EPP: WAIT (linia BSY trybu SPP)</li> <li>– sygnał strobojący dane: DATA STROBE (linia ALF trybu SPP)</li> <li>– sygnał strobojący adres: ADDR STROBE (linia DSL trybu SPP)</li> <li>– sygnał zerowania (inicjowania): RESET (linia INI trybu SPP)</li> </ul> </li> </ul>

98

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Dostęp do łącza równoległego pracującego w trybie EPP odbywa się za pośrednictwem zestawu rejestrów SPP, rozszerzonego o: <ul style="list-style-type: none"> <li>– dwukierunkowy rejestr danych (<i>Data Register</i>)</li> <li>– dwukierunkowy rejestr adresowy (<i>Address Register</i>)</li> </ul> </li> <li>■ Wytworzenie właściwych sekwencji sygnałów sterujących związanych z realizacją protokołu transmisji danych w trybie EPP realizowane jest automatycznie przez kontroler łącza równoległego</li> <li>■ Transmisję danych i/lub adresów realizuje się przez ich bezpośrednie wpisanie do odpowiednich rejestrów EPP</li> </ul>

99

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Tryb ECP pozwala na realizację dwukierunkowej transmisji danych z prędkością do 2 MB/s</li> <li>■ Protokół wymiany danych w trybie ECP wykorzystuje m.in. następujące linie sygnałowe: <ul style="list-style-type: none"> <li>– sygnał zegarowy: HostCLK (linia STR trybu SPP)</li> <li>– sygnał zegarowy: PeriphCLK (linia ACK trybu SPP)</li> <li>– sygnał sterujący przesyłaniem danych/rozkazów: PeriphACK (linia BSY trybu SPP)</li> <li>– sygnał sterujący przesyłaniem danych/rozkazów: HostACK (linia ALF trybu SPP)</li> <li>– sygnały zmiany kierunku transmisji: RevREQ (linia INI trybu SPP) oraz AckREV (linia PERR trybu SPP)</li> </ul> </li> </ul>

100

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Tryb ECP pozwala na korzystanie z: <ul style="list-style-type: none"> <li>– przesyłania danych z kontrolera w trybie DMA</li> <li>– dodatkowego (najczęściej 16-stopniowego) rejestru buforowanego FIFO</li> <li>– kompresji przesyłanych danych algorytmem RLE (<i>Run Length Encoding</i>)</li> <li>– definiowania i adresowania do 128 kanałów logicznych</li> </ul> </li> <li>■ Dostęp do łącza równoległego pracującego w trybie ECP odbywa się za pośrednictwem zestawu rejestrów SPP, rozszerzonego m.in. o: <ul style="list-style-type: none"> <li>– rejestr adresowy FIFO (<i>ECP Address FIFO</i>)</li> <li>– rejestry danych FIFO (<i>Data FIFO</i>)</li> <li>– rejestr sterujący ECR (<i>Extended Control Register</i>)</li> </ul> </li> </ul>

101

Interfejs IEEE-1284
<ul style="list-style-type: none"> <li>■ Wytworzenie właściwych sekwencji sygnałów sterujących związanych z realizacją protokołu transmisji danych w trybie ECP realizowane jest automatycznie przez kontroler łącza równoległego</li> <li>■ Interfejs IEEE-1284 korzysta z poziomów logicznych TTL, a długość kabla połączeniowego (niezależnie od trybu pracy) nie powinna przekraczać 5 m</li> <li>■ Gniazdo przyłączeniowe interfejsu IEEE-1284 może mieć 25 pinów (DB-25) lub 36 pinów (Centronics)</li> </ul>

102

Interfejs RS-232C
<ul style="list-style-type: none"> <li>■ Interfejs RS-232C (V24) jest standardem przesyłania informacji między dwoma urządzeniami za pośrednictwem asynchronicznego łącza szeregowego</li> <li>■ Dane przesyłane tego typu łączem są zorganizowane w ramki SDU (<i>Serial Data Unit</i>), składające się z: <ul style="list-style-type: none"> <li>– znacznika początku ramki (bitu startu o wartości 0)</li> <li>– 5, 6, 7 lub 8 bitów danych</li> <li>– opcjonalnego bitu kontroli parzystości</li> <li>– znacznika końca ramki (1, 1½ lub 2 bity stopu o wartości 1)</li> </ul> </li> <li>■ Przed nawiązaniem połączenia oba urządzenia muszą mieć ustawione takie same parametry transmisji tj. częstotliwość taktowania i format ramki</li> </ul>

103

Interfejs RS-232C
<ul style="list-style-type: none"> <li>■ Odbiornik w celu zsynchronizowania swojego zegara z zegarem nadajnika próbuje stan linii odbiorczej z częstotliwością wielokrotnie wyższą od częstotliwości z jaką przesyłane są dane</li> <li>■ Przekształcaniem danych z postaci równoległej na szeregową i odwrotnie zajmuje się układ UART (<i>Universal Asynchronous Receiver-Transmitter</i>)</li> <li>■ Transmisja danych może odbywać się z maksymalną prędkością 56 kb/s (UART 8250) lub 115200 kb/s (UART 16450) w trybie: <ul style="list-style-type: none"> <li>– jednokierunkowym (simpleksowym)</li> <li>– dwukierunkowym (półdupleksowym i duplexowym)</li> </ul> </li> </ul>

104

Interfejs RS-232C
<ul style="list-style-type: none"> <li>■ Standard RS-232C definiuje protokół przesyłania danych między terminalem DTE (<i>Data Terminal Equipment</i>) i modemem DCE (<i>Data Communication Equipment</i>), wykorzystujący następujące linie sygnałowe: <ul style="list-style-type: none"> <li>– dane wysyłane: TxD (<i>Transmitted Data</i>)</li> <li>– dane odbierane: RxD (<i>Received Data</i>)</li> <li>– żądanie rozpoczęcia nadawania: RTS (<i>Request To Send</i>)</li> <li>– potwierdzenie sygnału RTS (zgłoszenie gotowości odbioru danych z terminala): CTS (<i>Clear To Send</i>)</li> <li>– gotowość modemu na przyjęcie danych: DSR (<i>Data Set Ready</i>)</li> <li>– gotowość terminala: DTR (<i>Data Terminal Ready</i>)</li> </ul> </li> </ul>

105

Interfejs RS-232C
<ul style="list-style-type: none"> <li>– odbiór fali nośnej: DCD (<i>Data Carrier Detect</i>)</li> <li>– sygnał wywołania abonenta: RI (<i>Ring Indicator</i>)</li> <li>■ Para sygnałów RTS/CTS jest odpowiedzialna za sterowanie przepływem danych (oraz za sterowanie kierunkiem transmisji w trybie półdupleksowym)</li> <li>■ Para sygnałów gotowości DTR/DSR odpowiada za utrzymanie nawiązanego połączenia</li> <li>■ Interfejs RS-232C do reprezentowania stanów logicznych wykorzystuje napięcia z zakresu od -15 do +15 V (poziom logiczny „0” odpowiada napięciu na linii sygnałowej od +3 do +15 V, natomiast poziom logiczny „1” odpowiada napięciu od -3 do -15 V)</li> </ul>

106

Interfejs RS-232C
<ul style="list-style-type: none"> <li>■ Typowe gniazdo przyłączające interfejsu RS-232C ma 25 (DB-25) lub 9 pinów (DB-9)</li> <li>■ Maksymalna odległość między nadajnikiem i odbiornikiem nie przekracza 200 m</li> <li>■ Istnieje możliwość wymiany danych pomiędzy dwoma urządzeniami typu DTE połączonych tzw. kablem modemu zerowego (<i>null modem cable</i>)</li> <li>■ Obsługa łącza szeregowego może być realizowana przez: <ul style="list-style-type: none"> <li>– przerwania zgłaszane przez układ UART</li> <li>– okresowe testowanie przez procesor rejestru stanu transmisji układu UART</li> </ul> </li> </ul>

107

Magistrala USB
<ul style="list-style-type: none"> <li>■ Magistrala USB (<i>Universal Serial Bus</i>) jest standardem łącza szeregowego przeznaczonym do podłączania szerokiej gamy urządzeń peryferyjnych</li> <li>■ Cechy magistrali USB: <ul style="list-style-type: none"> <li>– prędkość transmisji: 1.5 Mb/s (<i>Low Speed</i>), 12 Mb/s (<i>Full Speed</i>), 480 Mb/s (<i>High Speed</i>) lub 4.8 Gb/s (<i>Super Speed</i>)</li> <li>– obsługa przez jeden kontroler (<i>Host Adapter</i>) do 127 urządzeń USB</li> <li>– topologia połączeń typu gwiazda (o maksymalnie siedmiu poziomach) realizowana przez koncentratory USB (<i>Hub</i>)</li> <li>– maksymalna długość kabla połączeniowego wynosi 5 m (maksymalna odległość między urządzeniem i kontrolerem nie może przekroczyć 35 m)</li> </ul> </li> </ul>

108

Magistrala USB
<ul style="list-style-type: none"> <li>– możliwość zmiany konfiguracji podłączonych urządzeń w trakcie pracy (<i>Hot-Plugging</i>)</li> <li>– automatyczny przydział identyfikatorów (adresów) do urządzeń (<i>Auto-ID</i>)</li> <li>– możliwość zasilania podłączonych urządzeń kablem USB (obciążenie do 500 mA dla USB 2.0, 900mA dla USB 3.0)</li> </ul> <ul style="list-style-type: none"> <li>■ Dane kodowane metodą NRZI (<i>Non Return to Zero Invert</i>) przesyłane są parą przewodów sygnałowych: D+ i D- w sposób różnicowy (napięcie od 0 do 3.6 V)</li> <li>■ Kabel połączeniowy magistrali USB to czterożyłowa skrętka (w USB 3.0 – ośmiożyłowa) zakończona złączami standardowymi (płaskim - A i kwadratowym - B) lub złączami miniaturowymi (mini, micro)</li> </ul>

109

Magistrala USB
<ul style="list-style-type: none"> <li>■ Wymiana informacji magistralą USB odbywa się w pakietach przez wirtualne kanały przesyłowe (<i>Pipes</i>) tworzone na żądanie przez kontroler USB</li> <li>■ Każdy kanał przesyłowy może wykorzystywać jeden z czterech trybów transmisji: <ul style="list-style-type: none"> <li>– tryb asynchroniczny nie gwarantujący określonej przepustowości (<i>Bulk Transfer</i>), który przeznaczony jest do obsługi podstawowych urządzeń zewnętrznych (np. drukarka, skaner)</li> <li>– tryb synchroniczny o gwarantowanej szerokości pasma (<i>Isochronous Transfer</i>), który przeznaczony jest do obsługi urządzeń przesyłających duże porcje danych z określoną prędkością (np. kamera)</li> </ul> </li> </ul>

110

Magistrala USB
<ul style="list-style-type: none"> <li>– tryb natychmiastowy (<i>Interrupt Transfer</i>), który przeznaczony jest do obsługi urządzeń asynchronicznych przesyłających niewielkie porcje danych (np. myszka, klawiatura)</li> <li>– tryb sterujący (<i>Control Transfer</i>), który przeznaczony jest do konfigurowania nowo podłączonych do magistrali urządzeń</li> </ul> <ul style="list-style-type: none"> <li>■ Wszystkie przesyłane magistralą USB dane transmitowane są w kolejności od najmłodszego do najstarszego bitu (bajta)</li> <li>■ Każdy przesyłany pakiet rozpoczyna się 8 lub 32-bitową sekwencją synchronizującą (<i>SYNC</i>), po której przekazywany jest 8-bitowy identyfikator pakietu (<i>PID</i>) określający m.in. rodzaj pakietu</li> </ul>

111

Magistrala USB
<ul style="list-style-type: none"> <li>■ Specyfikacja USB definiuje następujące rodzaje pakietów: <ul style="list-style-type: none"> <li>– żeton (<i>Token</i>) służący do określania adresu urządzenia USB (7 bitów ADDR + 4 bity ENDP), kierunku przepływu pakietów danych oraz testowania aktywności urządzenia</li> <li>– pakiet danych (<i>Data</i>) służący do przesyłania danych w blokach o długości 0÷1024 bajtów zabezpieczonych 16-bitowym kodem CRC</li> <li>– pakiet potwierdzający (<i>Handshake</i>) służący do przekazania raportu o przesłanych pakietach danych czy informacji o aktywności urządzenia</li> <li>– pakiet specjalny (<i>Special</i>) służący do sterowania przepływem danych w magistrali, do której podłączone są urządzenia o różnej prędkości transmisji</li> </ul> </li> </ul>

112

Magistrala FireWire
<ul style="list-style-type: none"> <li>■ Magistrala FireWire (i.Link, IEEE-1394) jest odmianą łącza szeregowego służącą do podłączania szerokiej gamy urządzeń peryferyjnych (przede wszystkim audiowizualnych)</li> <li>■ Cechy magistrali FireWire: <ul style="list-style-type: none"> <li>– transmisja w trybie asynchronicznym z prędkością 100 Mb/s (S100), 200 Mb/s (S200) lub 400 Mb/s (S400)</li> <li>– negocjowana minimalna gwarantowana przepustowość pasma (<i>Guaranteed Bandwidth</i>)</li> <li>– automatyczny przydział numerów (adresów) do urządzeń (<i>Auto-ID</i>)</li> <li>– możliwość zmiany konfiguracji podłączonych urządzeń w trakcie pracy (<i>Hot-Plugging</i>)</li> </ul> </li> </ul>

113

Magistrala FireWire
<ul style="list-style-type: none"> <li>– automatyczne dopasowywanie końców magistrali impedancją falową</li> <li>– fizyczna topologia połączeń typu drzewo (logiczna magistrala) złożona z kaskadowo połączonych łańcuchów (drzew) węzłów (<i>Nodes</i>) spajanych pomostami (<i>Bridges</i>)</li> <li>– maksymalna głębokość pojedynczego drzewa wynosi 16</li> <li>– 16-bitowe adresowanie węzłów (pojedynczy pomost pozwala na dołączenie do 63 węzłów)</li> <li>– 10-bitowe adresowanie pomostów</li> <li>– maksymalna długość kabla połączeniowego wynosi 4.5 m (maksymalna odległość między urządzeniem i kontrolerem nie może przekroczyć 72 m)</li> <li>– możliwość zasilania urządzeń izolowanych (maksymalne obciążenie do 1.5 A)</li> </ul>

114

	Magistrala FireWire
	<ul style="list-style-type: none"> <li>■ Sześciżyłowy kabel połączeniowy magistrali FireWire (2 pary ekranowanej skrętki + 2 linie zasilające) zakończony jest jednym z dwóch typów gniazd: 6 (1394-1995) lub 4 pinowym (1394a-2000)</li> <li>■ Pierwsza para skrętek służy do przesyłania strumienia danych, druga do przesyłania impulsów synchronizujących (transmisja typu simplex)</li> <li>■ Wymiana danych odbywa się poprzez kanały wirtualne służące zarówno do realizacji klasycznej transmisji asynchronicznej jak i transmisji z gwarantowaną szerokością pasma (<i>Isochronous Data Transport</i>)</li> </ul>
	115

	Magistrala FireWire
	<ul style="list-style-type: none"> <li>■ Najnowsza wersja standardu IEEE-1394b wprowadza następujące rozszerzenia: <ul style="list-style-type: none"> <li>– tryby prędkości transmisji: 400 Mb/s, 800 Mb/s, 1600 Mb/s i 3200 Mb/s</li> <li>– kodowania danych regułą 8/10, co pozwala na równoczesne korzystanie z obu par skrętek sygnałowych do transmisji danych</li> <li>– zwiększenie maksymalnej długości kabli połączeniowych do 100 m wykorzystując okablowanie strukturalne kat. 5</li> <li>– możliwość realizacji połączeń na bazie innych mediów (np. światłowodu)</li> <li>– wprowadzenie nowych typów kabli połączeniowych (9-żyłowych) i nowych złączy</li> <li>– optymalizacja protokołu transmisji</li> </ul> </li> </ul>
	116

	Magistrala Thunderbolt
	<ul style="list-style-type: none"> <li>■ Magistrala Thunderbolt opracowana przez firmę Intel jest odmianą szybkiego łącza szeregowego służącego do podłączania urządzeń wymagających bardzo dużej prędkości przesyłania danych, np. monitorów o wysokiej rozdzielczości, kontrolerów RAID</li> <li>■ W założeniu Thunderbolt ma zastąpić wszystkie wykorzystywane obecnie magistrale komunikacyjne, w tym: USB, FireWire i HDMI</li> <li>■ Thunderbolt powstał z połączenia DisplayPort (DP) i zewnętrznej magistrali PCI-Express x4 (PCIe)</li> <li>■ Do podłączania urządzeń wykorzystuje przyłączy mini DisplayPort (mDP)</li> </ul>
	117

	Magistrala Thunderbolt
	<ul style="list-style-type: none"> <li>■ Cechy magistrali Thunderbolt: <ul style="list-style-type: none"> <li>– dwukierunkowy transfer danych z szybkością 10 Gb/s (Thunderbolt 1), 20 Gb/s (Thunderbolt 2), 40 Gb/s (planowane w Thunderbolt 3) w każdą stronę</li> <li>– planowana możliwość przesyłania danych z szybkością 100 Gb/s wykorzystując łącze światłowodowe</li> <li>– możliwość równoczesnego łączenia do 6 urządzeń</li> <li>– obsługa standardu plug&amp;play</li> <li>– obsługa wielu protokołów komunikacyjnych</li> <li>– wbudowany mechanizm QoS</li> <li>– maksymalna długość kabli połączeniowych wynosi 3m (60m dla światłowodu)</li> <li>– możliwość bezpośredniego zasilania podłączanych urządzeń o poborze mocy do 10W (do 100W w Thunderbolt 3)</li> </ul> </li> </ul>
	118

	Architektura procesora
	<ul style="list-style-type: none"> <li>■ Począwszy od lat 60-tych procesory budowane były w myśl zasady CISC (<i>Complex Instruction Set Computer</i>) cechującej się tendencją do rozszerzania listy rozkazów o nowe rozkazy, realizujące coraz bardziej złożone zadania (np. przesłania blokowe, operacje na łańcuchach)</li> <li>■ Lista rozkazów typowego procesora CISC obejmuje nawet kilkaset rozkazów cechujących się zmienną długością i czasem wykonania od jednego do kilkudziesięciu taktów zegara</li> <li>■ Na pojedynczy rozkaz składa się sekwencja kilku do kilkunastu tzw. mikrooperacji</li> </ul>
	119

	Architektura procesora
	<ul style="list-style-type: none"> <li>■ Analiza kodu generowanego przez kompilatory języków wysokiego poziomu pozwoliła na wyciągnięcie następujących wniosków: <ul style="list-style-type: none"> <li>– za 80% czasu wykonywania programu odpowiada 20% instrukcji (reguła 80/20)</li> <li>– dominującą grupą rozkazów stanowią proste instrukcje przypisania, instrukcje warunkowe (IF, LOOP) oraz wywołania i powroty z podprogramów</li> <li>– większość używanych argumentów stanowią zmienne skalarne (głównie zmienne lokalne w procedurach)</li> </ul> </li> <li>■ Powyższe obserwacje doprowadziły do powstania koncepcji procesora o zredukowanej liczbie rozkazów RISC (<i>Reduced Instruction Set Computer</i>)</li> </ul>
	120

	Architektura procesora
	<ul style="list-style-type: none"> <li>■ Kluczowymi elementami koncepcji RISC jest: <ul style="list-style-type: none"> <li>– nieduży zestaw stosunkowo prostych rozkazów o stałej długości, pozwalający na znaczące uproszczenie konstrukcji jednostki sterującej i układu dekodera instrukcji oraz na zwiększenie szybkości przetwarzania rozkazów</li> <li>– wielopoziomowa pamięć podręczna usprawniająca pobieranie zwiększonej w stosunku do procesorów CISC liczby rozkazów</li> <li>– duża liczba rejestrów wewnętrznych umożliwiająca optymalizację odwołań do argumentów (zwłaszcza lokalnych)</li> <li>– optymalizacja potoków wykonawczych z uwagi na duży udział w kodzie rozkazów rozgałęzień warunkowych oraz wywołań i powrotów z podprogramów</li> </ul> </li> </ul>
	121

	Architektura procesora
	<ul style="list-style-type: none"> <li>■ Procesory oparte na architekturze RISC (np. Intel i860, Sun SPARC, IBM RS/6000, Motorola MC88000) cechują się pewnymi wspólnymi własnościami: <ul style="list-style-type: none"> <li>– wykonywanie większości rozkazów w jednym cyklu zegara</li> <li>– prosty format i stała długość rozkazu (dostosowana do szerokości magistrali danych)</li> <li>– jawne określanie rejestrów jako argumentów w kodzie rozkazu</li> <li>– mała liczba prostych trybów adresowania (głównie adresowanie natychmiastowe, bezpośrednie i rejestrowe)</li> </ul> </li> <li>■ Wiele obecnie konstruowanych procesorów łączy w sobie cechy architektur CISC i RISC (np. Intel 80960, Pentium 4, PowerPC 601)</li> </ul>
	122

	Przetwarzanie potokowe
	<ul style="list-style-type: none"> <li>■ Przetwarzanie pojedynczego rozkazu można dekomponować na szereg etapów (faz), które realizowane są przez niezależne obwody procesora</li> <li>■ Strategia przetwarzania potokowego (<i>Pipeline</i>) polega na zorganizowaniu procesu przetwarzania rozkazów w sposób zbliżony do linii montażowej w zakładzie produkcyjnym</li> <li>■ Podobnie jak linia montażowa potok zawiera rozkazy znajdujące się w różnym stadium przetwarzania, co prowadzi do zwiększenia szybkości ich przetwarzania</li> <li>■ Podstawowym parametrem mającym wpływ na efektywność potoku jest jego głębokość</li> </ul>
	123

	Przetwarzanie potokowe
	<ul style="list-style-type: none"> <li>■ Dla przykładu 6-etapowy potok (z jednostkowymi czasami trwania poszczególnych etapów), który wykorzystuje następującą dekompozycję przetwarzania rozkazu: <ul style="list-style-type: none"> <li>– Faza pobierania rozkazu (FI)</li> <li>– Faza dekodowania rozkazu (DI)</li> <li>– Faza obliczania argumentów (CO)</li> <li>– Faza pobierania argumentów (FO)</li> <li>– Faza wykonania rozkazu (EI)</li> <li>– Faza zapisania wyniku (WR)</li> </ul> </li> </ul> <p>pozwała na skrócenie czasu przetwarzania pięciu rozkazów z 30 do 10 cykli zegara</p>
	124

	Przetwarzanie potokowe
	<p>The diagram shows a 6-stage pipeline with stages FI, DI, CO, FO, EI, and WR. Five instructions (Rozkaz #1 to #5) are processed in parallel. Rozkaz #1 starts at cycle 1 and ends at cycle 6. Rozkaz #2 starts at cycle 2 and ends at cycle 7. Rozkaz #3 starts at cycle 3 and ends at cycle 8. Rozkaz #4 starts at cycle 4 and ends at cycle 9. Rozkaz #5 starts at cycle 5 and ends at cycle 10. The total execution time for all five instructions is 10 clock cycles.</p>
	125

	Przetwarzanie potokowe
	<ul style="list-style-type: none"> <li>■ Czynniki obniżające wydajność przetwarzania potokowego: <ul style="list-style-type: none"> <li>– różny czas trwania poszczególnych etapów prowadzi do powstawania opóźnień i najczęściej wymusza stosowanie dodatkowych buforów</li> <li>– niewszystkie etapy mogą być zawsze realizowane równolegle, z uwagi na możliwość wystąpienia konfliktów w dostępie do zasobów (np. fazy FI, FO, WR z poprzedniego przykładu mogą wymagać dostępu do pamięci)</li> <li>– wystąpienie przerwania, błędu strony lub naruszenie ochrony zaburza sekwencję przetwarzanych rozkazów i wymusza czyszczenie całego potoku</li> <li>– wystąpienie instrukcji rozgałęzienia warunkowego może prowadzić do unieważnienia rozkazów wewnątrz potoku</li> </ul> </li> </ul>
	126

Przetwarzanie potokowe
<ul style="list-style-type: none"> <li>■ Sposoby postępowania w przypadku rozgałęzień: <ul style="list-style-type: none"> <li>– pobieranie rozkazu docelowego z wyprzedzeniem (pobieranie rozkazu docelowego razem z rozkazem następującym po rozgałęzieniu)</li> <li>– zwielokrotnienie strumienia (jednoczesne przetwarzanie kilku ścieżek wykonawczych)</li> <li>– stosowanie bufora pętli (zapamiętywanie w niewielkiej, szybkiej pamięci ostatnio pobranych rozkazów i sprawdzanie czy cel rozgałęzienia znajduje się wewnątrz bufora)</li> <li>– opóźnianie rozgałęzienia (zmiana porządku wykonywanych rozkazów tak, aby rozgałęzienie występowało później)</li> <li>– przewidywanie rozgałęzień (predykcja wyniku rozgałęzienia w oparciu o kod operacji, zapamiętaną historię rozkazu lub tablicę historii rozgałęzień)</li> </ul> </li> </ul>

127

Przetwarzanie potokowe
<ul style="list-style-type: none"> <li>■ Najczęściej stosowanym obecnie rozwiązaniem jest technika przewidywania biegu programu (<i>Branch Prediction</i>) w oparciu o bufor historii rozgałęzień BTB (<i>Branch Target Buffer</i>)</li> <li>■ Tablica BTB jest szybką pamięcią asocjacyjną zawierającą od kilkuset do kilku tysięcy rekordów przechowujących: adres instrukcji rozgałęzienia, adres instrukcji docelowej oraz pewną liczbę bitów rejestrujących przebieg (historię) przetwarzania</li> <li>■ Efektywność przewidywania zależy od stosowanego algorytmu predykcji oraz wielkości bufora BTB i waha się od ok. 65% do prawie 90%</li> </ul>

128

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Technika przetwarzania superskalarnego polega na zastosowaniu kilku działających równolegle potoków przetwarzających dane skalarne, co pozwala na wykonywanie więcej niż jednego rozkazu w jednym cyklu zegara</li> <li>■ Podstawową trudnością w realizacji przetwarzania superskalarnego jest wzajemna współzależność rozkazów wykonywanych w poszczególnych potokach, wynikająca m.in. z rozgałęzień oraz uzależnień i konfliktów w dostępie do zasobów</li> <li>■ Dla przykładu procesor Pentium wykorzystuje dwa potoki wykonawcze, oznaczane odpowiednio U i V</li> </ul>

129

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Dzięki równoległemu wykonywaniu rozkazów, możliwe jest wykonywanie instrukcji całkowito-liczbowych w obu potokach w jednym cyklu rozkazowym</li> <li>■ Potoki skojarzone z obiema kolejkami rozkazów nie są równouprawnione: <ul style="list-style-type: none"> <li>– potok U pozwala na wykonywanie dowolnej instrukcji z zestawu rozkazów procesorów 80x86</li> <li>– potok V umożliwia realizację jedynie tzw. instrukcji prostych (rozkazy operujące na danych całkowitych oraz rozkazy zamiany danych w rejestrach), które realizowane są całkowicie sprzętowo, bez udziału mikroprogramu wbudowanego w pamięć ROM procesora</li> </ul> </li> </ul>

130

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Rozkazy przeznaczone dla potoku U i V przesuwają się synchronicznie przez poszczególne stopnie przetwarzania</li> <li>■ Jeżeli rozkaz w jednym z potoków przejdzie określony stopień przetwarzania przed zakończeniem tej samej fazy w drugim potoku, to wówczas oczekuje on na drugi rozkaz, tak aby oba rozkazy razem mogły przejść do kolejnej fazy przetwarzania</li> <li>■ Procesor jest w stanie z każdego potoku równocześnie odwoływać się do danych umieszczonych w wewnętrznej pamięci podręcznej</li> </ul>

131

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Uzyskanie zrównoleglenia przetwarzania wymaga tzw. „parowania rozkazów” tj. łączenia ich w pary w taki sposób, aby mogły być wykonywane równolegle przez oba potoki</li> <li>■ Instrukcje podlegające łączeniu w pary można podzielić na trzy grupy: <ul style="list-style-type: none"> <li>– grupę UV stanowią instrukcje, które połączone w pary mogą być wykonywane w obu potokach</li> <li>– grupę PU stanowią instrukcje wykonywane tylko w potoku U połączone w pary z odpowiednimi instrukcjami z potoku V</li> <li>– grupę PV stanowią instrukcje (wykonywane w obu potokach), które połączone w pary z instrukcjami ze zbioru UV i PU będą wykonywane w potoku V</li> </ul> </li> </ul>

132

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Próba wykonania pewnych par instrukcji w sposób równoległy może prowadzić do pojawienia się konfliktów lub uzależnień (które w konsekwencji utrudniają lub wręcz uniemożliwiają łączenie takich instrukcji w pary): <ul style="list-style-type: none"> <li>– RAR (<i>Read After Read</i>) np. <code>add bx, ax</code> <code>mov cx, ax</code></li> <li>– RAW (<i>Read After Write</i>) np. <code>mov ax, bx</code> <code>mov [mem], ax</code></li> <li>– WAR (<i>Write After Read</i>) np. <code>mov bx, ax</code> <code>add ax, cx</code></li> <li>– WAW (<i>Write After Write</i>) np. <code>mov ax, [mem]</code> <code>mov ax, bx</code></li> </ul> </li> </ul>
133

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Uzależnienie typu RAR jest wynikiem konfliktu podczas próby jednoczesnego odczytu rejestru lecz można je stosunkowo łatwo usunąć poprzez zdublowanie portu odczytu rejestru</li> <li>■ Wynikiem uzależnienia typu RAW jest wystąpienie opóźnienia związanego z oczekiwaniem na wynik wcześniejszej operacji</li> <li>■ Uzależnienia typu WAR i WAW wynikają z konfliktu podczas próby jednoczesnego zapisu (lub odczytu i zapisu) do rejestru i można im zapobiec stosując technikę przemianowywania rejestrów (<i>Register Renaming</i>)</li> </ul>
134

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Technika przemianowywania rejestrów polega na wykorzystaniu puli dodatkowych rejestrów i ich dynamicznym mapowaniu do rejestrów dostępnych programowo</li> <li>■ Po zdekodowaniu rozkazu, którego wynik ma być docelowo umieszczony w rejestrze, następuje przydzielenie z puli rejestru, na którym będzie wykonywana operacja i zaktualizowanie tablicy zamienników RAT (<i>Register Allocation Table</i>)</li> <li>■ Ponieważ rejestr dostępny programowo może być mapowany na kilka rejestrów z puli, pozwala to na zrównoleglenie wykonywania rozkazów</li> </ul>
135

Przetwarzanie superskalarne
<ul style="list-style-type: none"> <li>■ Dla przykładu sekwencja rozkazów (WAR): <code>mov bx, ax</code> <code>add ax, cx</code> po podstawieniu: <code>ax -&gt; R0</code> <code>bx -&gt; R1</code> <code>cx -&gt; R2</code> i przemianowaniu: <code>ax -&gt; R3</code> zostaje przekształcona w sekwencję: <code>R1 := R0</code> <code>R3 := R0 + R2</code></li> </ul>
136

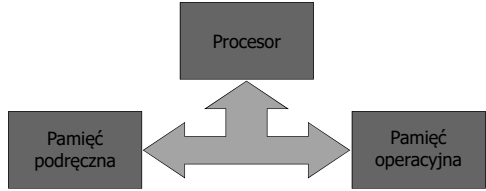
Dynamiczne wykonywanie instrukcji
<ul style="list-style-type: none"> <li>■ Dynamiczne wykonywanie instrukcji jest techniką pozwalającą na wykonywanie wejściowej sekwencji instrukcji w dowolnej kolejności (<i>out-of-order execution</i>) w celu uniknięcia konfliktów w dostępie do zasobów oraz powstawania opóźnień podczas potokowego przetwarzania instrukcji</li> <li>■ Dynamiczne wykonywanie instrukcji oparte jest na: <ul style="list-style-type: none"> <li>– zaawansowanej predykcji rozgałęzień (<i>deep branch prediction</i>)</li> <li>– dynamicznej analizie przepływu danych (<i>dynamic data flow analysis</i>)</li> <li>– spekulatywnym wykonywaniu rozkazów (<i>speculative execution</i>)</li> </ul> </li> </ul>
137

Dynamiczne wykonywanie instrukcji
<ul style="list-style-type: none"> <li>■ Dynamiczna analiza przepływu danych polega na analizowaniu w czasie rzeczywistym sekwencji instrukcji przetwarzanych przez procesor w celu wykrycia i określenia zależności pozwalających na ich optymalne uszeregowanie w jednostkach wykonawczych</li> <li>■ Spekulatywne wykonywanie instrukcji polega na wykonywaniu instrukcji znajdujących się za jeszcze nierozstrzygniętą instrukcją warunkową, przy czym wyniki tychże instrukcji są zapamiętywane w specjalnym bloku rejestrów tymczasowych</li> </ul>
138

	<b>Dynamiczne wykonywanie instrukcji</b>
	<ul style="list-style-type: none"> <li>■ Typowa sekwencja dynamicznego wykonania instrukcji przebiega następująco: <ul style="list-style-type: none"> <li>– utworzenie przez układ MIS (<i>Microcode Instruction Sequencer</i>) sekwencji mikrooperacji do wykonania</li> <li>– uzupełnienie sekwencji mikrooperacji informacją o ich oryginalnej kolejności</li> <li>– przydzielenie do mikrooperacji rejestrów tymczasowych</li> <li>– pobranie mikrooperacji do wykonania przez układ przydziału zasobów RS (<i>Reservation Station</i>) w takiej kolejności, która zapewnia najbardziej optymalne ich wykonanie</li> <li>– odtworzenie przez układ RU (<i>Retirement Unit</i>) właściwej kolejności zrealizowanych mikrooperacji i przepisanie wyników z rejestrów tymczasowych do rejestrów dostępnych programowo</li> </ul> </li> </ul>
	139

	<b>Pamięć podręczna</b>
	<ul style="list-style-type: none"> <li>■ Celem stosowania pamięci podręcznej (<i>Cache</i>) jest zwiększenie wydajności podsystemu pamięci wynikające z zasady lokalności, w myśl której: <ul style="list-style-type: none"> <li>– zdecydowana większość kodu przetwarzana jest w sposób sekwencyjny (za wyjątkiem instrukcji skoków oraz wywołań i powrotów z procedur), co oznacza, że zajmuje on spójny i ciągły blok w pamięci</li> <li>– znaczna część obliczeń dotyczy przetwarzania zmiennych prostych, tablic i rekordów, które położone są w pamięci stosunkowo blisko siebie</li> </ul> </li> <li>■ Cechą charakterystyczną pamięci podręcznej jest bardzo krótki czas dostępu, co pozwala na efektywne buforowanie wolniejszej pamięci operacyjnej RAM</li> </ul>
	140

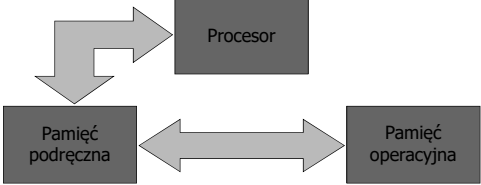
	<b>Pamięć podręczna</b>
	<ul style="list-style-type: none"> <li>■ Typowa pamięć podręczna jest systemem wielopoziomowym złożonym najczęściej z dwóch (L1+L2) lub trzech (L1+L2+L3) kaskadowo połączonych układów pamięci statycznych o coraz większym czasie dostępu i coraz większej pojemności</li> <li>■ Pamięć podręczna L1 ma zwykle pojemność od kilku do kilkudziesięciu KB i na ogół służy do rozdzielnego buforowania danych i instrukcji</li> <li>■ Rodzaje architektur systemów z pamięcią podręczną: <ul style="list-style-type: none"> <li>– konwencjonalna (<i>Look-Aside</i>)</li> <li>– szeregową (<i>Look-Through</i>)</li> <li>– dwuszynową (<i>Backside</i>)</li> </ul> </li> </ul>
	141

	<b>Pamięć podręczna</b>
	<ul style="list-style-type: none"> <li>■ Umiejscowienie pamięci podręcznej w architekturze konwencjonalnej (<i>Look-Aside</i>):</li> </ul>
	
	142

	<b>Pamięć podręczna</b>
	<ul style="list-style-type: none"> <li>■ Pamięć podręczna typu <i>Look-Aside</i> dołączana jest do systemu przez magistralę systemową i pracuje z wyznaczoną przez nią częstotliwością taktowania</li> <li>■ Kontroler pamięci podręcznej monitoruje wszystkie odwołania procesora do pamięci, aby sprawdzić, czy żądana dana znajduje się w pamięci podręcznej</li> <li>■ W przypadku „trafienia” (<i>cache hit</i>) tj. znalezienia żądanej danej, procesor przerywa cykl magistrali i informuje podsystem sterujący pamięcią operacyjną, aby zignorował żądanie</li> <li>■ W przypadku „chybienia” (<i>cache miss</i>), cykle magistrali są wykonywane w normalny sposób</li> </ul>
	143

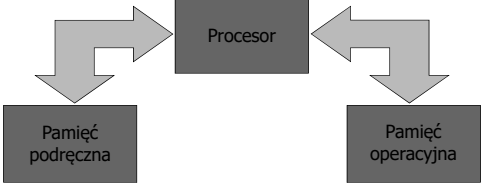
	<b>Pamięć podręczna</b>
	<ul style="list-style-type: none"> <li>■ Zalety stosowania pamięci typu <i>Look-Aside</i>: <ul style="list-style-type: none"> <li>– niski koszt wynikający z prostej konstrukcji</li> <li>– polepszony czas dostępu do danych, które nie znajdują się w pamięci podręcznej</li> </ul> </li> <li>■ Podstawową wadą stosowania pamięci typu <i>Look-Aside</i> jest brak izolowania magistrali procesora od magistrali systemowej, co prowadzi do: <ul style="list-style-type: none"> <li>– niemożności zredukowania stopnia wykorzystania magistrali systemowej</li> <li>– braku możliwości współbieżnego wykonywania operacji (dostęp do pamięci podręcznej realizowany jest przez magistralę systemową)</li> </ul> </li> </ul>
	144



	Pamięć podręczna
	<ul style="list-style-type: none"> <li>Umiejscowienie pamięci podręcznej w architekturze szeregowej (<i>Look-Through</i>):</li> </ul> 
	145

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>Po zainicjowaniu cyklu odczytu pamięć typu <i>Look-Through</i> przeszukuje swój katalog w celu rozstrzygnięcia, czy zawiera kopię żądanej przez procesor danej</li> <li>W przypadku „trafienia” dana wysyłana jest do procesora z zerowymi cyklami oczekiwania magistrali, a magistrala systemowa pozostaje dostępna dla innych urządzeń (np. dla danych przesyłanych w trybie DMA)</li> <li>W przypadku „chybienia” następuje odczyt danej z pamięci operacyjnej (przez magistralę systemową)</li> </ul>
	146

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>Zalety stosowania pamięci typu <i>Look-Through</i>: <ul style="list-style-type: none"> <li>zredukowanie stopnia wykorzystania magistrali systemowej</li> <li>możliwość równoczesnego (współbieżnego) wykonywania operacji na dwóch magistralach znajdujących się w jednym systemie</li> <li>wykonywanie operacji zapisu w zerowych cyklach oczekiwania magistrali</li> <li>możliwość taktowania magistrali lokalnej inną częstotliwością niż magistrali systemowej</li> </ul> </li> <li>Wady stosowania pamięci typu <i>Look-Through</i>: <ul style="list-style-type: none"> <li>wyższy koszt wynikający z większej złożoności układu</li> <li>wolniejszy niż w przypadku pamięci <i>Look-aside</i> odczyt danych, które nie znajdują się w pamięci podręcznej</li> </ul> </li> </ul>
	147

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>Umiejscowienie pamięci podręcznej w architekturze dwuszynowej (<i>Backside</i>):</li> </ul> 
	148

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>Pamięć podręczna typu <i>Backside</i> podłączona jest do procesora dedykowaną magistralą BSB (<i>Back Side Bus</i>), która jest całkowicie niezależna od magistrali pamięciowej FSB (<i>Front Side Bus</i>)</li> <li>Rozwiązanie to pozwala na swobodny wybór częstotliwości pracy pamięci podręcznej</li> <li>Podobnie jak w przypadku pamięci typu <i>Look-Through</i> obie magistrale mogą działać współbieżnie</li> <li>Pamięć typu <i>Backside</i> może być wkomponowana w strukturę samego procesora (Pentium Pro) lub umieszczona w jego pobliżu (Pentium III)</li> </ul>
	149

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>Jeżeli dane w pamięci operacyjnej różnią się od odpowiadających im danych w pamięci podręcznej mamy do czynienia z utratą spójności pamięci podręcznej</li> <li>Przyczyną takiej sytuacji może być „trafienie” zapisu w pamięć podręczną lub zapisanie danych do pamięci operacyjnej przez urządzenie zewnętrzne</li> <li>Strategie zapisu zapewniające spójność pamięci podręcznej: <ul style="list-style-type: none"> <li>Zapis jednoczesny (<i>Write-Through</i>)</li> <li>Zapis buforowany (<i>Buffered Write-Through</i>)</li> <li>Zapis opóźniony (<i>Write-Back</i>)</li> </ul> </li> </ul>
	150

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>■ W strategii <i>Write-Through</i> każdy zapis do pamięci podręcznej powoduje natychmiastowe wykonanie cyklu zapisu do pamięci operacyjnej, co gwarantuje stałe utrzymywanie aktualnych danych w pamięci operacyjnej</li> <li>■ Operacja zapisu w pamięci operacyjnej realizowana jest niezależnie od tego, czy nastąpiło „trafienie” czy „chybienie” z zapisem do pamięci podręcznej</li> <li>■ Implementacja tej metody jest bardzo efektywna i prosta, cechuje się jednak małą wydajnością, gdyż każda operacja zapisu powoduje odwołanie się do wolnej pamięci operacyjnej</li> </ul>
	151

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>■ Strategia <i>Buffered Write-Through</i> polega na zapamiętywaniu przez kontroler pamięci podręcznej realizowanych operacji zapisu w specjalnym buforze, co umożliwia ich późniejsze zrealizowanie w pamięci operacyjnej bez wpływu na pracę procesora</li> <li>■ Podstawową zaletą tej metody jest brak cykli oczekiwania magistrali zarówno w przypadku „trafienia” jak i „chybienia” zapisu</li> <li>■ Ten rodzaj zapisu powoduje, że inne urządzenie wymagające dostępu do magistrali nie może przeprowadzać na niej żadnych operacji do chwili zakończenia operacji zapisu</li> </ul>
	152

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>■ Strategia <i>Write-Back</i> zwiększa całkowitą wydajność systemu, gdyż dokonuje aktualizacji pamięci operacyjnej tylko wtedy, gdy jest to konieczne</li> <li>■ Aktualizacja pamięci operacyjnej ma miejsce w przypadku gdy: <ul style="list-style-type: none"> <li>– urządzenie zewnętrzne inicjuje operację odczytu lub zapisu tych słów pamięci, które zawierają „przeterminowane” dane</li> <li>– słowo pamięci podręcznej, które zawiera zmodyfikowane dane, ma zostać nadpisane danymi z pamięci operacyjnej</li> </ul> </li> <li>■ Słowa pamięci podręcznej są oznaczane w lokalnym katalogu w momencie kiedy są aktualizowane przez procesor</li> </ul>
	153

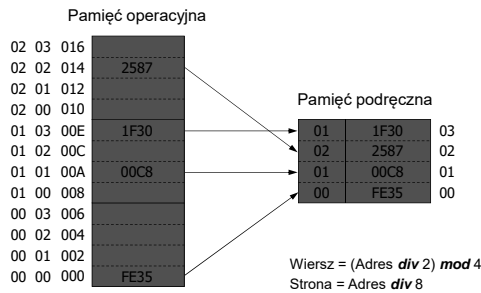
	Pamięć podręczna
	<ul style="list-style-type: none"> <li>■ W momencie zapisu lub odczytu do lub z pamięci operacyjnej, podsystem pamięci podręcznej musi monitorować magistralę systemową aby sprawdzić, czy są to odwołania do słów pamięci oznaczonych jako zmodyfikowane</li> <li>■ Zachowanie spójności pamięci podręcznej (zwłaszcza w systemach wieloprocessorowych) oparte jest na protokole MESI, który określa stan każdego wiersza pamięci podręcznej danych jako: <ul style="list-style-type: none"> <li>– zmodyfikowany (<i>Modified</i>)</li> <li>– wyłączony (<i>Exclusive</i>)</li> <li>– współdzielony (<i>Shared</i>)</li> <li>– nieważny (<i>Invalid</i>)</li> </ul> </li> </ul>
	154

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>■ Pamięć podręczna zorganizowana jest w wiersze, których szerokość odpowiada najczęściej szerokości magistrali danych</li> <li>■ Pamięć operacyjna dzielona jest na strony o rozmiarze równym liczbie wierszy pamięci podręcznej</li> <li>■ Informacja o tym, które wiersze pamięci operacyjnej są odwzorowane w pamięci podręcznej znajduje się w katalogu pamięci podręcznej TAG-RAM</li> <li>■ Strategie odwzorowywania pamięci operacyjnej: <ul style="list-style-type: none"> <li>– Odwzorowanie bezpośrednie (<i>Direct Mapped</i>)</li> <li>– Odwzorowanie skojarzeniowe (<i>Fully Associative</i>)</li> <li>– Odwzorowanie sekwencyjno-skojarzeniowe (<i>Set Associative</i>)</li> </ul> </li> </ul>
	155

	Pamięć podręczna
	<ul style="list-style-type: none"> <li>■ Odwzorowanie bezpośrednie polega na takim sposobie odwzorowywania pamięci operacyjnej na pamięć podręczną, że n-ty wiersz każdej strony pamięci operacyjnej może zostać odwzorowany tylko w n-tym wierszu pamięci podręcznej</li> <li>■ Zaletą tej metody jest duża szybkość wyszukiwania informacji (tylko jedna operacja porównania) oraz prosta konstrukcja układu</li> <li>■ Ponieważ każdy wiersz pamięci operacyjnej ma stałą lokalizację w pamięci podręcznej, prowadzi to do ciągłej wymiany zawartości pamięci podręcznej, a tym samym do obniżenia wydajności systemu</li> </ul>
	156

## Pamięć podręczna

### ■ Przykład odwzorowania bezpośredniego:



157

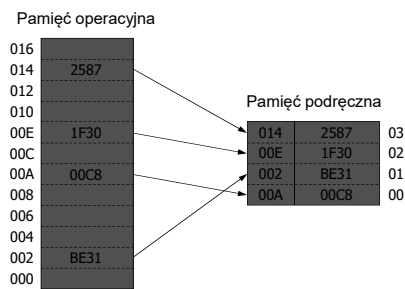
## Pamięć podręczna

- Odwzorowanie skojarzeniowe pozwala na ładowanie każdego wiersza pamięci operacyjnej do dowolnego wiersza pamięci podręcznej, co eliminuje największą wadę odwzorowania bezpośredniego
- W tego typu odwzorowaniu nie występuje podział pamięci operacyjnej na strony – wszelkie operacje wykonywane są na wierszach identyfikowanych przez adresy
- Podstawową wadą tej metody jest konieczność przeszukania całego katalogu TAG-RAM aby zbadać, czy żądany wiersz pamięci operacyjnej został odwzorowany w pamięci podręcznej

158

## Pamięć podręczna

### ■ Przykład odwzorowania skojarzeniowego:



159

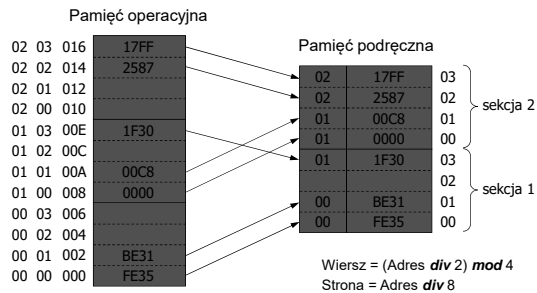
## Pamięć podręczna

- Odwzorowanie sekcyjno-skojarzeniowe jest metodą, która łączy zalety odwzorowania bezpośredniego i skojarzeniowego
- Cechą charakterystyczną tego typu podejścia jest podział pamięci podręcznej na 2, 4 lub 8 sekcji (*2-Way-Set*, *4-Way-Set*, *8-Way-Set*) o rozmiarze równym wielkości strony pamięci operacyjnej
- Ponieważ pamięć operacyjna odwzorowywana jest na każdą sekcję w sposób bezpośredni, wyszukiwanie odpowiedniej informacji w pamięci podręcznej wymaga przeprowadzenia co najwyżej 8 porównań (jedno porównanie na sekcję)

160

## Pamięć podręczna

### ■ Przykład dwudrożnego odwzorowania sekcyjno-skojarzeniowego (*2-Way-Set*):



161

## Pamięć podręczna

- Wspólną cechą odwzorowania skojarzeniowego i sekcyjno-skojarzeniowego jest konieczność wyboru tego wiersza pamięci podręcznej, który zostanie zastąpiony przez nowo ładowany wiersz z pamięci operacyjnej
- W tym celu kontroler pamięci podręcznej implementuje jeden z następujących algorytmów:
  - FIFO (*First In First Out*), który do zastąpienia wybiera wiersz najdłużej pozostający w pamięci podręcznej
  - LFU (*Least Frequently Used*), który do zastąpienia wybiera najrzadziej używany wiersz pamięci podręcznej
  - LRU (*Least Recently Used*), który do zastąpienia wybiera najmniej ostatnio używany wiersz pamięci podręcznej

162

Przegląd architektury IA-32
<ul style="list-style-type: none"> <li>■ Architektura IA-32 jest 32-bitową architekturą procesorów firmy Intel, będącą rozwinięciem architektury procesorów rodziny 80x86</li> <li>■ Każdy procesor z rodziny IA-32 cechuje zachowanie wstecznej zgodności programowej (nowszy typ procesora może wykonywać kod utworzony dla starszego typu procesora)</li> <li>■ Typowymi przedstawicielami linii produktów opartej na architekturze IA-32 są procesory, które wyewoluowały z procesorów Pentium II, takie jak: Pentium III, Pentium 4, Pentium 4 HT (<i>Hyper Threading</i>), Intel Xeon, Pentium M</li> </ul>

163

Przegląd architektury IA-32
<ul style="list-style-type: none"> <li>■ Ewolucja procesorów linii IA-32: <ul style="list-style-type: none"> <li>– Intel 8086 (1978): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 29 tys.</li> <li>■ częstotliwość taktowania: 8 MHz</li> <li>■ magistrala danych: 16 bitów</li> <li>■ przestrzeń adresowa: 1 MB</li> <li>■ podręczna kolejka rozkazów</li> </ul> </li> <li>– Intel 80286 (1982): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 134 tys.</li> <li>■ częstotliwość taktowania: 12.5 MHz</li> <li>■ przestrzeń adresowa: 16 MB</li> <li>■ tryb chroniony (<i>protected mode</i>)</li> </ul> </li> </ul> </li> </ul>

164

Przegląd architektury IA-32
<ul style="list-style-type: none"> <li>– Intel 80386 (1985): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 275 tys.</li> <li>■ częstotliwość taktowania: 20 MHz</li> <li>■ magistrala danych: 32 bity</li> <li>■ przestrzeń adresowa: 4 GB</li> <li>■ tryb wirtualny procesora 8086 (<i>virtual 8086 mode</i>)</li> <li>■ jednostka stronicowania pamięci (<i>paging unit</i>)</li> </ul> </li> <li>– Intel 80486 (1989): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 1.2 mln</li> <li>■ częstotliwość taktowania: 25 MHz</li> <li>■ pamięć podręczna: 8 KB</li> <li>■ jednostka zmiennoprzecinkowa FPU</li> </ul> </li> </ul>

165

Przegląd architektury IA-32
<ul style="list-style-type: none"> <li>– Intel Pentium (1993): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 3.1 mln</li> <li>■ częstotliwość taktowania: 60 MHz</li> <li>■ magistrala danych: 64 bity (wewn. 128/256 bitów)</li> <li>■ pamięć podręczna: 8 KB kod + 8 KB dane (MESI)</li> <li>■ dwa potoki wykonawcze całkowitoliczbowe</li> <li>■ wsparcie dla systemów dwuprocesorowych</li> <li>■ technologia MMX</li> </ul> </li> <li>– Intel Pentium Pro (1995): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 5.5 mln</li> <li>■ częstotliwość taktowania: 200 MHz</li> <li>■ przestrzeń adresowa: 64 GB</li> <li>■ pamięć podręczna: 16 KB (L1), 256/512 KB (L2)</li> </ul> </li> </ul>

166

Przegląd architektury IA-32
<ul style="list-style-type: none"> <li>■ mikro-architektura superskalarna P6 (3 instrukcje/takt zegara; 5 jednostek wykonawczych: 2 x Integer, 2 x FPU, 1 x MIU)</li> <li>– Intel Pentium II (1997): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 7 mln</li> <li>■ częstotliwość taktowania: 266 MHz</li> <li>■ pamięć podręczna: 32 KB (L1), 256/512 KB (L2)</li> <li>■ technologia MMX</li> </ul> </li> <li>– Intel Pentium III (1999): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 8.2÷28 mln</li> <li>■ częstotliwość taktowania: 500 MHz</li> <li>■ pamięć podręczna: 32 KB (L1), 256/512/1024 KB (L2)</li> <li>■ technologia SSE (<i>Streaming SIMD Extensions</i>)</li> </ul> </li> </ul>

167

Przegląd architektury IA-32
<ul style="list-style-type: none"> <li>– Intel Pentium 4 (2000): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 42 mln</li> <li>■ częstotliwość taktowania: 1.5 GHz</li> <li>■ pamięć podręczna: 12.000 pops (ETC), 8 KB (L1), 256 KB (L2)</li> <li>■ Intel NetBurst Micro-Architecture</li> <li>■ technologia SSE2</li> </ul> </li> <li>– Intel Xeon (2002): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 55 mln</li> <li>■ częstotliwość taktowania: 2.2 GHz</li> <li>■ pamięć podręczna: 12.000 pops (ETC), 8 KB (L1), 512 KB (L2)</li> <li>■ technologia HT (<i>Hyper Threading</i>)</li> </ul> </li> </ul>

168

Przegląd architektury IA-32	
<ul style="list-style-type: none"> <li>– Intel Pentium 4 HT (2002): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 55 mln</li> <li>■ częstotliwość taktowania: 3.06 GHz</li> <li>■ pamięć podręczna: 12.000 μops (ETC), 8 KB (L1), 512 KB (L2)</li> <li>■ technologia HT</li> </ul> </li> <li>– Intel Pentium M (2003): <ul style="list-style-type: none"> <li>■ liczba tranzystorów: 77 mln</li> <li>■ częstotliwość taktowania: 1.6 GHz</li> <li>■ pamięć podręczna: 32 KB kod + 32 KB dane (L1), 1 MB (L2)</li> <li>■ technologia SSE2</li> <li>■ technologia APM (<i>Advanced Power Management</i>)</li> </ul> </li> </ul>	169

Przegląd architektury IA-32	
<ul style="list-style-type: none"> <li>■ Podstawowe środowisko wykonawcze procesorów IA-32 tworzy: <ul style="list-style-type: none"> <li>– Zestaw rejestrów podstawowych, który zawiera: <ul style="list-style-type: none"> <li>■ osiem 32-bitowych rejestrów ogólnego przeznaczenia (EAX, EBX, ECX, EDX, EBP, ESI, EDI, ESP), których młodsze 16-bitów jest odwzorowywanych na rejestry 16-bitowe procesorów 8086 i 80286 (AX, BX, CX, DX, BP, SI, DI, SP); ponadto rejestry AX, BX, CX i DX są dodatkowo mapowane na pary rejestrów 8-bitowych (AH, AL, BH, BL, CH, CL, DH, DL)</li> <li>■ sześć 16-bitowych rejestrów segmentowych (CS, DS, ES, FS, GS, SS)</li> <li>■ jeden 32-bitowy rejestr statusowy (EFLAGS)</li> <li>■ jeden 32-bitowy rejestr wskaźnika rozkazów (EIP)</li> </ul> </li> </ul> </li> </ul>	170

Przegląd architektury IA-32	
<ul style="list-style-type: none"> <li>– Zestaw rejestrów jednostki zmiennie-przecinkowej (FPU), który zawiera m.in.: <ul style="list-style-type: none"> <li>■ osiem 80-bitowych rejestrów danych zorganizowanych w postaci stosu: ST(0)÷ST(7)</li> <li>■ jeden 16-bitowy rejestr sterujący (FPU Control Word)</li> <li>■ jeden 16-bitowy rejestr stanu (FPU Status Word)</li> <li>■ jeden 16-bitowy rejestr słowa wyróżników (FPU Tag Word)</li> <li>■ jeden 48-bitowy rejestr wskaźnika rozkazu (FPU Instruction Pointer)</li> <li>■ jeden 48-bitowy rejestr wskaźnika argumentu (FPU Operand Pointer)</li> </ul> </li> <li>– Zestaw ośmiu 64-bitowych rejestrów technologii MMX (MM0÷MM7), współdzielonych z rejestrami danych FPU</li> </ul>	171

Przegląd architektury IA-32	
<ul style="list-style-type: none"> <li>– Zestaw rejestrów technologii SSE i SSE2, który zawiera: <ul style="list-style-type: none"> <li>■ osiem 128-bitowych rejestrów danych (XMM0÷XMM7)</li> <li>■ jeden 32-bitowy rejestr kontrolno-statusowy (MXCSR)</li> </ul> </li> <li>– Zestaw rejestrów pomocniczych, który zawiera m.in.: <ul style="list-style-type: none"> <li>■ rejestry sterujące pracą procesora (CR0÷CR4)</li> <li>■ rejestry podsystemu zarządzania pamięcią (GDTR, LDTR, IDTR, TR)</li> <li>■ rejestry uruchomieniowe (DR0÷DR7)</li> <li>■ rejestry zaawansowanego kontrolera przerwań APIC (ISR, TMR, IRR, ESR)</li> <li>■ rejestry wspomagające testowanie procesora (TR3÷TR7)</li> </ul> </li> </ul>	172

IA-32: Tryby pracy procesora	
<ul style="list-style-type: none"> <li>■ Procesor może pracować w jednym z następujących trybów pracy: <ul style="list-style-type: none"> <li>– trybie rzeczywistym (<i>Real Mode, Real-address Mode</i>)</li> <li>– trybie chronionym (<i>Protected Mode</i>)</li> <li>– trybie zarządzania SMM (<i>System Management Mode</i>)</li> </ul> </li> <li>■ Procesor działając w trybie rzeczywistym (np. po włączeniu zasilania czy aktywacji sygnału Reset) odpowiada funkcjonalnie procesorowi 8086</li> <li>■ W tym trybie pracy 20-bitowy adres fizyczny tworzony jest na podstawie 16-bitowego adresu segmentu i 16-bitowego przesunięcia w segmencie (<math>Address = 16 * Segment + Offset</math>)</li> </ul>	173

IA-32: Tryby pracy procesora	
<ul style="list-style-type: none"> <li>■ Tryb chroniony jest podstawowym trybem pracy procesora, zapewniającym ochronę niezależnych zadań wykonywanych współbieżnie w środowisku wielozadaniowym (<i>multi-tasking environment</i>)</li> <li>■ Jedną z możliwości oferowanych przez tryb chroniony jest udostępnianie dla każdego z wykonywanych zadań niezależnego środowiska procesora 8086 (<i>Virtual-8086 Mode</i>)</li> <li>■ Tryb wirtualny procesora 8086 pozwala na uruchamianie programów pisanych dla trybu rzeczywistego w środowisku wielozadaniowym (chronionym)</li> </ul>	174

	IA-32: Tryby pracy procesora
	<ul style="list-style-type: none"> <li>Tryb zarządzania SMM pozwala na realizację funkcji systemowych wysokiego poziomu, takich jak zarządzanie energią czy ochrona zasobów przed nieuprawnionym dostępem</li> <li>Funkcjonowanie procesora w trybie zarządzania SMM kontrolowane jest wyłącznie przez oprogramowanie systemowe zawarte w pamięci stałej w sposób transparentny dla innych programów</li> <li>Typowym zastosowaniem tego trybu pracy jest implementacja systemu zarządzania zasilaniem, który może wprowadzać procesor i wybrane urządzenia zewnętrzne w stan „uśpienia” w okresie bezczynności</li> </ul>
	175

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>Procesory linii IA-32 mogą adresować maksymalnie do 64 GB (<math>2^{36}</math>) pamięci fizycznej (najmniejszą adresowalną jednostką pamięci fizycznej jest bajt)</li> <li>Mechanizm zarządzania pamięcią wirtualną procesora (<i>virtual memory management mechanism</i>) odwzorowuje przestrzeń adresową wykonywanego programu w przestrzeń pamięci fizycznej wykorzystując jeden z trzech modeli pamięci: <ul style="list-style-type: none"> <li>model płaski (<i>flat model</i>)</li> <li>model segmentowy (<i>segment model</i>)</li> <li>model trybu rzeczywistego procesora 8086 (<i>real-address model</i>)</li> </ul> </li> </ul>
	176

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>W modelu płaskim pamięć widoczna dla programu jest ciągła i liniowa (kod, dane oraz stos programu rezydują w jednej przestrzeni adresowej)</li> <li>Maksymalny rozmiar udostępnianej programowi przestrzeni liniowej wynosi 4 GB (<math>2^{32}</math>)</li> <li>W modelu segmentowym pamięć widoczna jest dla programu jako grupa niezależnych obszarów pamięci zwanych segmentami (najczęściej kod, dane oraz stos programu rezydują w niezależnych segmentach)</li> <li>W tym modelu program może zaadresować maksymalnie 16383 segmenty różnego typu i o różnym rozmiarze (nie większym jednak niż 4 GB)</li> </ul>
	177

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>Adres logiczny (<i>far pointer</i>) w modelu segmentowym składa się z dwóch elementów: 16-bitowego selektora segmentu (<i>segment selector</i>) i 32-bitowego przesunięcia w segmencie (<i>offset</i>)</li> <li>Selektor segmentu stanowi indeks do globalnej (GDT) lub lokalnej (LDT) tablicy deskryptorów opisujących poszczególne segmenty</li> <li>Deskryptor segmentu (<i>segment descriptor</i>) jest strukturą przechowującą jego typ, granularność, rozmiar, lokalizację w pamięci oraz prawa dostępu</li> <li>Adresacja wewnątrz pojedynczego segmentu jest liniowa</li> </ul>
	178

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>Selektory segmentów przechowywane są w rejestrach segmentowych (CS, DS, ES, FS, GS i SS)</li> <li>Przesunięcia wewnątrz segmentów przechowywane są najczęściej w rejestrach ogólnego przeznaczenia (np. EBX i ESI dla DS, EDI dla ES, EBP i ESP dla SS)</li> <li>Zaletą stosowania modelu segmentowego jest możliwość poprawienia niezawodności programu (systemu operacyjnego) dzięki: <ul style="list-style-type: none"> <li>odseparowaniu od siebie programów wykonywanych współbieżnie w środowisku wielozadaniowym</li> <li>rozdzieleniu logicznie niezależnych „fragmentów” programu (np. stosu od danych)</li> </ul> </li> </ul>
	179

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>Model trybu rzeczywistego odwzorowuje mechanizm adresowania pamięci procesora 8086</li> <li>Tryb ten wykorzystuje specjalną implementację modelu segmentowego polegającą na podziale pamięci na nakładające się wzajemnie i przesunięte o 16 B segmenty o stałym rozmiarze 64 KB</li> <li>Maksymalny rozmiar udostępnianej programowi przestrzeni adresowej wynosi 1 MB (<math>2^{20}</math>)</li> <li>Fizyczny adres w pamięci wyznaczany jest na podstawie 16-bitowego adresu segmentu (<i>segment</i>) i 16-bitowego przesunięcia w segmencie (<i>offset</i>)</li> </ul>
	180

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>■ Począwszy od procesora 80386 odwzorowanie adresu logicznego na adres fizyczny (w modelach płaskim i segmentowym) może być realizowane z wykorzystaniem mechanizmu stronicowania pamięci</li> <li>■ Jednostka stronicowania (<i>Paging Unit</i>) dzieli liniową przestrzeń adresową na strony (<i>pages</i>) o stałym rozmiarze (4 KB, 2 MB lub 4 MB), które są następnie odwzorowywane w pamięci wirtualnej złożonej z pamięci operacyjnej i pamięci dyskowej</li> <li>■ Maksymalny rozmiar adresowalnej w ten sposób pamięci wirtualnej wynosi 64 TB</li> </ul>
	181

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>■ W przypadku strony o rozmiarze 4 KB interpretacja adresu liniowego jest następująca: <ul style="list-style-type: none"> <li>– Najstarsze 10 bitów adresu stanowi indeks w katalogu stron (<i>Page Directory</i>) przechowującym wskaźniki do tablic stron; adres fizyczny katalogu stron znajduje się w rejestrze CR3</li> <li>– Kolejnych 10 bitów adresu stanowi indeks w tablicy stron (<i>Page Table</i>) przechowującej fizyczny adres początku strony</li> <li>– Pozostałe 12 bitów adresu traktowane jest jako przesunięcie wewnątrz wybranej strony</li> </ul> </li> <li>■ W przypadku strony o rozmiarze 4 MB katalog stron (indeksowany 10 bitami adresu) przechowuje adres fizyczny początku strony, pozostałe 22 bity adresu liniowego stanowi przesunięcie wewnątrz strony</li> </ul>
	182

	IA-32: Zarządzanie pamięcią
	<ul style="list-style-type: none"> <li>■ Jeżeli wyznaczona w powyższy sposób strona nie znajduje się w pamięci operacyjnej, to jej adres zostaje zapisany w rejestrze CR2 i generowany jest wyjątek błędu strony (<i>page fault</i>)</li> <li>■ Zadanie wczytania brakującej strony z dysku spoczywa na systemie operacyjnym (procedurze obsługi tego wyjątku)</li> <li>■ W celu zwiększenia efektywności mechanizmu stronicowania procesor wyposażony jest w pamięci podręczne TLB (<i>Translation Lookaside Buffer</i>) przechowującą zawartość ostatnio używanych (maksymalnie 128) tablic stron</li> </ul>
	183

	IA-32: Technologia MMX
	<ul style="list-style-type: none"> <li>■ Technologia MMX (<i>MultiMedia eXtension</i>) ma celu zwiększenie efektywności wykonywania programów (głównie multimedialnych), które cechują się powtarzaniem takich samych, prostych operacji na długich ciągach krótkich danych</li> <li>■ Idea leżąca u podstaw technologii MMX stanowi krok w stronę przetwarzania równoległego SIMD (<i>Single Instruction Multiple Data</i>) – zamiast kolejnego przetwarzania pojedynczych jednostek informacji, można te same dane przetworzyć szybciej, pobierając i wykonując rozkazy na kilku porcjach informacji na raz</li> </ul>
	184

	IA-32: Technologia MMX
	<ul style="list-style-type: none"> <li>■ Procesor implementujący technologię MMX wyposażony jest w osiem 64-bitowych rejestrów: MM0÷MM7, pozwalających na przechowywanie nowych typów danych (tzw. danych spakowanych): <ul style="list-style-type: none"> <li>– Packet Bytes (8 x 8 bitów)</li> <li>– Packed Words (4 x 16 bitów)</li> <li>– Packet Doublewords (2 x 32 bity)</li> <li>– Quadword (1 x 64 bity)</li> </ul> </li> <li>■ Jednoczesne wykonywanie operacji na danych spakowanych umożliwia zestaw dodatkowych 57 rozkazów typu SIMD wykorzystujących tzw. arytmetykę nasycenia</li> </ul>
	185

## IA-32: Technologia MMX

- W arytmetyce nasycenia nie występuje sygnalizacja wystąpienia przekroczenia zakresu (w wyniku pakowania lub wykonywania operacji) lecz zastąpienie wyniku wartością maksymalną lub minimalną dla danego typu danych
- Dla przykładu obliczanie dwóch iloczynów skalarnych wektorów może zostać zastąpione pojedynczym rozkazem PMADDWD:

The diagram illustrates the PMADDWD instruction. It shows two 4-element vectors,  $[a_0, a_1, c_0, c_1]$  and  $[b_0, b_1, d_0, d_1]$ , being multiplied element-wise. The result is a single 2-element vector  $[a_0 \cdot b_0 + a_1 \cdot b_1, c_0 \cdot d_0 + c_1 \cdot d_1]$ .

186

	IA-32: Technologia SSE (SSE2, ...)
	<ul style="list-style-type: none"> <li>■ Technologia SSE (<i>Streaming SIMD Extensions</i>) rozszerza ideę jednoczesnego przetwarzania grupy danych pojedynczym rozkazem (znaną z technologii MMX) na dane w formacie zmiennoprzecinkowym</li> <li>■ Procesor implementujący technologię SSE (SSE 2) wyposażony jest w osiem 128-bitowych rejestrów XMM0÷XMM7 oraz rejestr kontrolno-sterujący MXCSR</li> <li>■ Zaletą takiego rozwiązania jest uniezależnienie się od rejestrów zmiennoprzecinkowych FPU</li> <li>■ Największą wadą tej technologii jest brak zgodności z wcześniejszymi typami procesorów</li> </ul>
	187

	IA-32: Technologia SSE (SSE2, ...)
	<ul style="list-style-type: none"> <li>■ Lista rozkazów SSE zawiera 72 rozkazy obejmujące: <ul style="list-style-type: none"> <li>– rozkazy przetwarzające grupy danych zmiennoprzecinkowych (<i>SIMD Floating Point</i>)</li> <li>– rozkazy przetwarzające pojedyncze dane zmiennoprzecinkowe (<i>Single Scalar</i>)</li> <li>– rozkazy sterujące przepływem danych (<i>Memory Streaming Instructions</i>)</li> <li>– rozkazy rozszerzające technologię MMX (<i>SIMD-INT</i>)</li> <li>– rozkazy sterujące (<i>Control</i>)</li> </ul> </li> <li>■ Typowym zastosowaniem technologii SSE jest obróbka grafiki 3D oraz pełnoekranowa kompresja (dekompresja) MPEG-2 w czasie rzeczywistym</li> </ul>
	188

	Magistrale systemowe
	<ul style="list-style-type: none"> <li>■ Magistrala jest zbiorem pewnych linii sygnałowych umożliwiającym takie połączenie współpracujących elementów, aby każdy z nich mógł komunikować się ze wszystkimi pozostałymi</li> <li>■ Podstawowe cechy magistral: <ul style="list-style-type: none"> <li>– Rodzaj (specjalistyczna, multipleksowana)</li> <li>– Metoda arbitrażu (centralny, rozproszony)</li> <li>– Koordynacja czasowa (synchroniczna, asynchroniczna)</li> <li>– Szerokość magistrali (adres, dane)</li> <li>– Rodzaj transferu danych (zapis, odczyt, odczyt-modyfikacja-zapis, odczyt-po-zapisie, blokowy)</li> </ul> </li> </ul>
	189

	Magistrale systemowe
	<ul style="list-style-type: none"> <li>■ Magistrala systemowa jest komponentem zapewniającym komunikację między podstawowymi podzespołami systemu komputerowego (procesor, pamięć, układy wejścia-wyjścia)</li> <li>■ W stosowanych obecnie magistralach systemowych możemy wyróżnić trzy grupy linii sygnałowych: <ul style="list-style-type: none"> <li>– linie adresowe (tworzące tzw. magistralę adresową)</li> <li>– linie danych (tworzące tzw. magistralę danych)</li> <li>– linie sterujące (tworzące tzw. magistralę sterującą)</li> </ul> </li> <li>■ Typowym rozwiązaniem zwiększającym wydajność systemu komputerowego jest stosowanie struktur wielomagistralowych o określonej hierarchii</li> </ul>
	190

	Magistrale systemowe
	<ul style="list-style-type: none"> <li>■ W zależności od przeznaczenia można wyróżnić wiele rodzajów magistral: <ul style="list-style-type: none"> <li>– magistrala pamięciowa</li> <li>– magistrala międzyukładowa (wewnętrzna)</li> <li>– magistrala rozszerzająca (zewnętrzna)</li> <li>– magistrala lokalna</li> </ul> </li> <li>■ Zadaniem magistrali pamięciowej (<i>memory bus</i>) jest przesyłanie informacji między procesorem a pamięcią operacyjną i/lub pamięcią podręczną</li> <li>■ Magistrala międzyukładowa (<i>interconnect bus</i>) służy do wymiany informacji między poszczególnymi układami otoczenia procesora (<i>chipsets</i>)</li> </ul>
	191

	Magistrale systemowe
	<ul style="list-style-type: none"> <li>■ Magistrala rozszerzająca (<i>expansion bus</i>) służy do wymiany informacji między urządzeniami wejścia-wyjścia a resztą systemu komputerowego</li> <li>■ Najczęściej konstrukcja tego typu magistrali musi spełniać warunki zgodności z wcześniejszymi rozwiązaniami i zapewnić poprawną współpracę z szerokim zakresem urządzeń peryferyjnych</li> <li>■ Magistrala lokalna (<i>local bus</i>) jest odmianą magistrali rozszerzającej przeznaczoną do obsługi urządzeń wymagających bardzo szybkiej transmisji danych (np. dysków twardych, kart graficznych, kontrolerów SCSI)</li> </ul>
	192



Magistrale systemowe
<ul style="list-style-type: none"> <li>■ Cechą charakterystyczną magistrali lokalnej jest bezpośrednie korzystanie z sygnałów sterujących procesora oraz praca z prędkością wyznaczoną przez zegar systemowy</li> <li>■ Przegląd standardów magistral rozszerzających: <ul style="list-style-type: none"> <li>– ISA (<i>Industry Standard Architecture</i>) <ul style="list-style-type: none"> <li>■ standard magistrali oparty na rozwiązaniu z IBM PC/AT</li> <li>■ przepustowość 8.33 MB/s (taktowanie 8.33 MHz)</li> <li>■ 16-bitowa szyna danych</li> <li>■ kontroler DMA oparty na dwóch połączonych kaskadowo układach 8237A</li> <li>■ kontroler przerwain sprzętowych oparty na dwóch połączonych kaskadowo układach 8259A</li> </ul> </li> </ul> </li> </ul>

193

Magistrale systemowe
<ul style="list-style-type: none"> <li>– MCA (<i>Micro Channel Architecture</i>) <ul style="list-style-type: none"> <li>■ brak zgodności ze standardem ISA</li> <li>■ 32-bitowa szyna danych</li> <li>■ przepustowość 20 MB/s (taktowanie 10 MHz)</li> <li>■ dostęp do pamięci taktowany częstotliwością zegara procesora</li> <li>■ wsparcie dla systemów wieloprocesorowych (centralny procesor na płycie głównej tzw. <i>Host CPU</i> może być wspomagany przez 16 procesorów umieszczonych na kartach rozszerzających; dla potrzeb komunikacji między procesorami, uwzględnienia ich priorytetów, przydziału odcinków czasowych dla dysponowania magistralami, dedykowano specjalną 4-bitową szynę sterującą)</li> </ul> </li> </ul>

194

Magistrale systemowe
<ul style="list-style-type: none"> <li>■ kontroler DMA obsługujący osiem 32-bitowych kanałów (brak emulacji 8- i 16-bitowych trybów pracy układu 8237A; wszystkie kanały mogą być jednocześnie aktywne)</li> <li>■ kontroler przerwain sprzętowych wyzwalanych poziomem pozwalający dzięki przydziałowi wielu urządzeń do jednego kanału na obsługę do 255 urządzeń</li> <li>■ kontroler przerwain sprzętowych obsługujący do 255 urządzeń (przydział jednego kanału wielu urządzeniom; przerwanie wyzwalane poziomem)</li> <li>– EISA (<i>Extended Industry Standard Architecture</i>) <ul style="list-style-type: none"> <li>■ zgodność ze standardem ISA</li> <li>■ 32-bitowa szyna danych</li> <li>■ przepustowość 33 MB/s (taktowanie 8.33 MHz)</li> </ul> </li> </ul>

195

Magistrale systemowe
<ul style="list-style-type: none"> <li>■ dostęp do pamięci realizowany jest z pełną częstotliwością zegara procesora</li> <li>■ możliwość symulowania przez kontroler magistrali starego, 16-bitowego trybu pracy</li> <li>■ wsparcie dla systemów wieloprocesorowych (kontroler magistrali umożliwia hierarchiczno-priorytetowy sposób dostępu wielu procesorów do magistrali systemowej)</li> <li>■ kontroler DMA obsługujący siedem 8-, 16- lub 32-bitowych kanałów DMA (rotacyjny system przydziału kanałów; pełne 32-bitowe adresowanie umożliwiające transfer bloków o wielkości do 4 GB)</li> <li>■ kontroler przerwain sprzętowych obsługujący przerwania wyzwalane poziomem</li> </ul>

196

Magistrale systemowe
<ul style="list-style-type: none"> <li>– VESA Local Bus (<i>Video Electronics Standards Association</i>) <ul style="list-style-type: none"> <li>■ 32-bitowa szyna danych</li> <li>■ przepustowość do 120 MB/s</li> <li>■ praktycznie obsługa tylko dwóch urządzeń: karty graficznej i kontrolera dysków</li> <li>■ silna zależność od sprzętu (magistrala wykorzystuje część niebuforowanych sygnałów procesora 80486)</li> </ul> </li> <li>– PCI (<i>Peripheral Component Interconnect</i>) <ul style="list-style-type: none"> <li>■ 32/64-bitowa szyna danych</li> <li>■ przepustowość 133/266/533 MB/s (taktowanie 33/66 MHz)</li> <li>■ odczyt i zapis w trybie pakietowym (<i>burst</i>)</li> <li>■ obsługa do 32 urządzeń (wielofunkcyjnych)</li> </ul> </li> </ul>

197

Magistrale systemowe
<ul style="list-style-type: none"> <li>■ możliwość współpracy do 256 magistral PCI w jednym systemie komputerowym (<i>PCI-to-PCI bridges</i>)</li> <li>■ możliwość obsługi magistrali ISA (<i>PCI-to-ISA bridge</i>)</li> <li>■ niezależność od typu procesora (komunikacja z procesorem i pamięcią realizowana jest za pośrednictwem specjalnego układu - <i>PCI bridge</i>)</li> <li>■ wbudowany mechanizm autokonfiguracji</li> <li>■ urządzenia dołączane do magistrali mogą należeć do jednej z dwóch kategorii: urządzeń inicjujących transmisję – inicjatorów (<i>initiators</i>), które zarządzają magistralą oraz urządzeń podporządkowanych (<i>slaves</i>) potrafiących jedynie transmitować dane; transmisja danych może odbywać się między dwoma inicjatorami lub inicjatorem a urządzeniem podporządkowanym</li> </ul>

198

Magistrale systemowe
<ul style="list-style-type: none"> <li>■ Magistrala umożliwia współdzielenie każdej z czterech linii zgłoszenia przerwania (INTA#÷INTD#) przez kilka urządzeń, lecz nie określa sposobu ich podłączenia do kontrolera przerw systemu</li> <li>■ urządzenia PCI mogą pracować z napięciem 5V lub 3.3V</li> </ul> <p>– AGP (<i>Accelerated Graphics Port</i>)</p> <ul style="list-style-type: none"> <li>■ magistrala AGP dedykowana do obsługi układu graficznego jest rozszerzeniem standardu PCI 2.1</li> <li>■ magistrala AGP może pracować w jednym z czterech trybów: <ul style="list-style-type: none"> <li>– 1x (częstotliwość 66MHz, przepustowość 266 MB/s)</li> <li>– 2x (wykorzystanie obu zboczy sygnału zegarowego 66MHz, przepustowość 532 MB/s, obniżenie napięcia z 5V do 3.3V)</li> </ul> </li> </ul>

199

Magistrale systemowe
<ul style="list-style-type: none"> <li>– 4x (dwie transakcje na jeden cykl zegara, przepustowość 1064 MB/s, obniżenie napięcia do 1.5V) – specyfikacja AGP 2.0</li> <li>– 8x (zmiany w protokole transmisji, przepustowość 2.1 GB/s, obniżenie napięcia do 0.8V, zgodność z trybem 4x) – specyfikacja AGP 3.0</li> </ul> <ul style="list-style-type: none"> <li>■ Aby wykorzystać pełne pasmo przenoszenia 32-bitowej magistrali AGP do przesyłania danych kontroler graficzny może opcjonalnie wykorzystać pomocniczą, 8-bitową magistralę SBA (<i>Sideband Addressing</i>) do niezależnego przesyłania adresów i rozkazów</li> </ul> <p>– PCI Express (PCIe, PCI-E)</p> <ul style="list-style-type: none"> <li>■ magistrala połączeniowa typu punkt-punkt (kontroler – urządzenie PCI-E, najczęściej karta graficzna)</li> </ul>

200

Magistrale systemowe
<ul style="list-style-type: none"> <li>■ magistrala PCI-E wykorzystuje od 1 do 32 linii sygnałowych determinujących jej przepustowość (500 MB/s na jedną linię w trybie <i>full-duplex</i> v2.0) oraz rozmiar złącza połączeniowego</li> <li>■ przepustowość magistrali PCI-E w zależności od jej wariantu wynosi od 250 MB/s (x1) do 16 GB/s (x16)</li> <li>■ maksymalny pobór mocy urządzenia podłączonego do magistrali wynosi: 75 W (v1.0), 150 W (v2.0) lub 300 W (v3.0)</li> </ul>

201

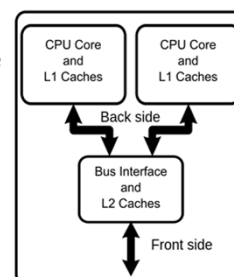
Procesory wielordzeniowe
<ul style="list-style-type: none"> <li>■ Procesor wielordzeniowy posiada więcej niż jeden tzw. rdzeń fizyczny tj. jednostkę odpowiedzialną za przetwarzanie danych złożoną z wielu jednostek wykonawczych</li> <li>■ Wprowadzenie procesorów wielordzeniowych miało na celu zwiększenie efektywności przetwarzania przez procesor danych poprzez ominięcie ograniczeń konstrukcyjnych procesorów jednordzeniowych: <ul style="list-style-type: none"> <li>– częstotliwość taktowania procesora i związana z nią ilość wydzielanego ciepła (<i>Thermal Design Power</i>) – w praktyce ok. 4GHz</li> <li>– liczba instrukcji wykonywanych przez procesor w jednym cyklu zegara IPC (<i>Instructions Per Cycle</i>) – w praktyce 3-4</li> </ul> </li> </ul>

202

Procesory wielordzeniowe
<ul style="list-style-type: none"> <li>■ Współczesne procesory wielordzeniowe oparte są na znanej z systemów wieloprocesorowych architekturze SMP (<i>Symmetric Multi Processing</i>), która pozwala na symetryczne przetwarzanie danych jednocześnie na kilku procesorach</li> <li>■ Procesory funkcjonujące w architekturze SMP komunikują się ze sobą za pomocą wspólnej magistrali systemowej, a przydział procesorów dla poszczególnych wątków (<i>threads</i>) jest realizowany przez system operacyjny</li> <li>■ Zaletą architektury SMP jest jej prosta sprzętowa implementacja oraz prosty model programowy</li> </ul>

203

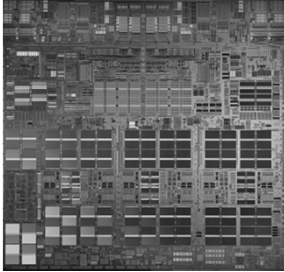
Procesory wielordzeniowe
<ul style="list-style-type: none"> <li>■ Każdy z fizycznych rdzeni procesora wielordzeniowego jest identyczny i komunikuje się z pozostałymi rdzeniami za pomocą wspólnej wewnętrznej magistrali systemowej</li> <li>■ Liczba rdzeni zależy od konstrukcji procesora wielordzeniowego</li> <li>■ Procesory wielordzeniowe mogą pracować w systemach wieloprocesorowych</li> </ul>



204

## Procesory wielordzeniowe

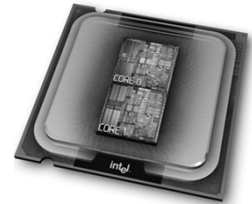
- Pierwszy na świecie produkowany seryjnie procesor wielordzeniowy IBM Power 4 zawierał dwa rdzenie 64-bitowe oparte na architekturze RISC (2001)



205

## Procesory wielordzeniowe

- Pierwszymi produkowanymi seryjnie (2005) procesorami wielordzeniowymi zgodnymi z architekturą x86 były dwurdzeniowe procesory AMD serii Opteron oraz Pentium D
- Pentium D zawierał dwa układy krzemowe (procesory) umieszczone we wspólnej obudowie, co było źródłem problemów komunikacyjnych na magistrali FSB



206

## Procesory wielordzeniowe

- W procesorze wielordzeniowym wiele wątków rozkładanych jest równomiernie na poszczególne rdzenie, jednakże bardziej złożone zadania, które nie wymagają obliczeń na kilku wątkach równocześnie wykonają się szybciej na pojedynczym rdzeniu procesora o wyższej częstotliwości taktowania
- Teoretycznie w architekturze SMP wydajność procesora posiadającego  $n$  rdzeni jest  $n$ -krotnie większa od procesora o jednym rdzeniu
- Z uwagi na ograniczenia związane z przesyłaniem danych praktyczna wydajność procesorów wielordzeniowych jest jednak o 20-40% mniejsza

207

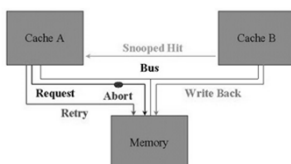
## Procesory wielordzeniowe

- Ograniczenia architektury SMP w procesorach wielordzeniowych:
  - skończona skalowalność, wynikająca z "rywalizacji" poszczególnych rdzeni o zasoby (pamięć operacyjną i zasoby urządzeń wejścia-wyjścia)
  - przepustowość magistrali systemowej, za pomocą której poszczególne rdzenie komunikują się ze sobą oraz ze wspólną pamięcią podręczną (*cache*)
  - szybkość działania mechanizmu zapewniającego spójność danych we wspólnej pamięci podręcznej
  - szeregowanie zadań

208

## Procesory wielordzeniowe

- We współczesnych procesorach wielordzeniowych każdy z rdzeni dysponuje prywatną pamięcią *cache* L1, prywatną lub współdzieloną pamięcią L2 oraz współdzieloną pamięcią *cache* L3 (jeśli występuje)
- Zapewnienie spójności danych w pamięciach *cache* jest realizowane sprzętowo wykorzystując tzw. protokół spójności (np. protokół MESI)



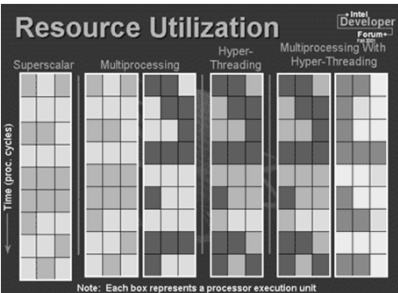
209

## Procesory wielordzeniowe

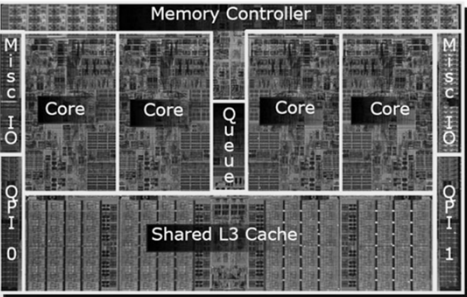
- Według firmy Intel graniczna liczba rdzeni x86 umieszczonych w jednym procesorze, która oferuje istotny wzrost wydajności to maksimum 16 rdzeni (łącznie 64 rdzenie w systemie wieloprocessorowym)
- Według firmy AMD graniczna liczba rdzeni x86 umieszczonych w jednym procesorze nie powinna przekraczać 24 rdzeni (łącznie 96 rdzeni w systemie wieloprocessorowym)

210

	Procesory wielordzeniowe
	<ul style="list-style-type: none"> <li>HyperThreading (hiperwątkowość), HT – technologia zwiększania wydajności wykonywanych współbieżnie zadań polegająca na udostępnianiu dla systemu operacyjnego dwóch logicznych rdzeni (procesorów wirtualnych) przypadających na jeden rdzeń fizyczny</li> </ul> <p>Implementacja HT polega na zduplikowaniu obwodów procesora (rdzenia) wykorzystywanych do przechowywania stanów procesów, zachowując niezmienioną liczbę jednostek wykonawczych</p> <p>HT pozwala na zwiększenie stopnia wykorzystania jednostek wykonawczych procesora – na jednym rdzeniu fizycznym wykonywane są dwa wątki</p>
	211

	Procesory wielordzeniowe
	<ul style="list-style-type: none"> <li>Wykorzystanie jednostek wykonawczych procesora:</li> </ul>  <p>Note: Each box represents a processor execution unit</p> <p>źródło: <a href="http://www.pcstats.com/articleview.cfm?articleID=1302">www.pcstats.com/articleview.cfm?articleID=1302</a></p>
	212

	Procesory wielordzeniowe
	<ul style="list-style-type: none"> <li>HT jest stosowany od 2002r w procesorze Xeon Foster MP, a następnie w Pentium 4 HT, Petium Extreme Edition, Intel Atom, a od 2008r w procesorach opartych na mikroarchitekturze Core</li> <li>Według firmy Intel implementacja technologią HT wymaga ok. 5% większej powierzchni płytki półprzewodnika, oferując wzrost wydajności w granicach 15-30% w porównaniu z procesorami bez technologii HT</li> <li>HT została skrytykowana przez konsorcjum ARM za nadmierne zużycie energii (do 46% większe niż w procesorach dwurdzeniowych)</li> </ul>
	213

	Procesory wielordzeniowe
	<ul style="list-style-type: none"> <li>Struktura procesora Intel Core i7:</li> </ul> 
	214

	Procesory wielordzeniowe
	<ul style="list-style-type: none"> <li>Intel Turbo Boost – technika zwiększająca wydajność procesora wtedy, kiedy nie ma potrzeby korzystania ze wszystkich rdzeni</li> </ul> <p>W takiej sytuacji częstotliwość taktowania części rdzeni procesora jest automatycznie zwiększana powyżej częstotliwości nominalnej, umożliwiając bardziej wydajne przetwarzanie zmniejszonej liczby potoków instrukcji</p> <p>Częstotliwość przetaktowania rdzeni jest ograniczona przede wszystkim przez maksymalną dopuszczalną ilość wydzielanego w procesorze ciepła</p>
	215

	Procesory wielordzeniowe
	<ul style="list-style-type: none"> <li>Macro-Ops Fusion – technika pozwalająca na łączenie dwóch instrukcji x86 w jedną całość – mikrooperację, która jest następnie wykonywana w jednym cyklu zegara (np. rozkaz skoku następujący po rozkazie porównania)</li> <li>Virtualization Technology – technika, dzięki której pojedynczy procesor może zachowywać się tak, jakby był kilkoma procesorami działającymi niezależnie i równolegle, umożliwiając pracę kilku systemów (operacyjnych) jednocześnie</li> </ul> <p>Każdy z działających systemów ma do dyspozycji własny „procesor wirtualny”</p>
	216