

Pracownia komputerowa

Rafał J. Wysocki

Instytut Fizyki Teoretycznej, Wydział Fizyki UW

9 stycznia 2012

Kontakt

- <http://www.fuw.edu.pl/~rwys/pk>
- rwys@fuw.edu.pl
- tel. 22 55 32 263
- ul. Hoża 69, pok. 142 (dawniej 216)

Materiał na ćwiczenia

- 1 Pliki i katalogi w systemie Linux.
 - Drzewo katalogowe.
 - Operacje na plikach i katalogach, linia poleceń (*ang. command line*).
 - Prawa dostępu.
 - Przeszukiwanie drzewa katalogowego.
 - Archiwizowanie danych.
 - Procesy.
- 2 Pakiet biurowy OpenOffice.org.
 - Arkusz kalkulacyjny.
 - Procesor tekstu.
 - Edytor równań.
 - Łączenie dokumentów, eksport i import danych.
- 3 Wizualizacja danych.
 - qtiplot
 - gnuplot

Plan wykładu

- Zasada działania i budowa komputera.
- Binarny zapis danych, reprezentacje liczb i znaków.
- Struktura i budowa pamięci, rodzaje pamięci.
- Masowe składowanie danych, systemy plikowe i drzewa katalogowe.
- Przestrzeń dyskowa, pliki i katalogi.
- Standardowe formaty plików.
- Cyfrowy zapis dźwięku i obrazu.
- Przesyłanie danych na odległość i sieci komputerowe.

Zasada działania komputera

Podstawowe składniki (każdego) komputera

- 1 Procesor (*ang. processor*) — może być więcej, niż jeden.
- 2 Pamięć (*ang. memory*).
- 3 Urządzenia wejścia/wyjścia (*ang. Input/Output devices*).

Zasada działania komputera

Podstawowe składniki (każdego) komputera

- 1 Procesor (*ang. processor*) — może być więcej, niż jeden.
- 2 Pamięć (*ang. memory*).
- 3 Urządzenia wejścia/wyjścia (*ang. Input/Output devices*).

Zadaniem procesora jest wykonywanie **programów**, które składają się z symboli zwanych **rozkazami** (*ang. instruction*), odczytywanych z pamięci.

Zasada działania komputera

Podstawowe składniki (każdego) komputera

- 1 Procesor (*ang. processor*) — może być więcej, niż jeden.
- 2 Pamięć (*ang. memory*).
- 3 Urządzenia wejścia/wyjścia (*ang. Input/Output devices*).

Zadaniem procesora jest wykonywanie **programów**, które składają się z symboli zwanych **rozkazami** (*ang. instruction*), odczytywanych z pamięci.

Rozkazy reprezentują **bardzo proste** operacje do przeprowadzenia. Zwykle dopiero wykonanie wielu rozkazów powoduje zauważalny skutek.

Zasada działania komputera

Podstawowe składniki (każdego) komputera

- 1 Procesor (*ang. processor*) — może być więcej, niż jeden.
- 2 Pamięć (*ang. memory*).
- 3 Urządzenia wejścia/wyjścia (*ang. Input/Output devices*).

Zadaniem procesora jest wykonywanie **programów**, które składają się z symboli zwanych **rozkazami** (*ang. instruction*), odczytywanych z pamięci.

Rozkazy reprezentują **bardzo proste** operacje do przeprowadzenia. Zwykle dopiero wykonanie wielu rozkazów powoduje zauważalny skutek.

Każdy rozkaz jest odczytywany z określonego miejsca (lokacji) w pamięci i może wyznaczać położenie (w pamięci) następnego rozkazu (domyślnie kolejne rozkazy są położone w pamięci jeden za drugim).

Wykonywanie rozkazów

Przykład: obliczamy 3^5

Lokacja	Zawartość
1	Zapisz 3 w lokacji 7
2	Zapisz 3 w lokacji 8
3	Zapisz 4 w lokacji 9
4	Pomnóż zawartość lokacji 8 przez zawartość lokacji 7 i zapisz wynik w lokacji 8
4	Zmniejsz zawartość lokacji 9 o 1
5	Jeżeli zawartość lokacji 9 jest różna od 0, wróć do rozkazu w lokacji 4
6	
7	
8	
9	

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).
- 3 Czy maszyna ENIAC była skonstruowana zgodnie z definicją von Neumanna?

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).
- 3 Czy maszyna ENIAC była skonstruowana zgodnie z definicją von Neumanna?
 - Nie.

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).
- 3 Czy maszyna ENIAC była konstruowana zgodnie z definicją von Neumanna?
 - Nie.
- 4 W którym roku powstała maszyna ENIAC?

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).
- 3 Czy maszyna ENIAC była konstruowana zgodnie z definicją von Neumanna?
 - Nie.
- 4 W którym roku powstała maszyna ENIAC?
 - 1946. Ostatecznie wyłączono ją w 1955 roku.

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).
- 3 Czy maszyna ENIAC była konstruowana zgodnie z definicją von Neumanna?
 - Nie.
- 4 W którym roku powstała maszyna ENIAC?
 - 1946. Ostatecznie wyłączono ją w 1955 roku.
- 5 W którym roku został zaprojektowany pierwszy „prawdziwy” komputer?

Quiz historyczny

- 1 Od czego pochodzi nazwa „komputer”?
 - Od nazwy maszyny ENIAC (*ang. Electronic Numerical Integrator And Computer*).
- 2 Kto sformułował „współczesną” definicję komputera?
 - John von Neumann (maszyna von Neumanna).
- 3 Czy maszyna ENIAC była konstruowana zgodnie z definicją von Neumanna?
 - Nie.
- 4 W którym roku powstała maszyna ENIAC?
 - 1946. Ostatecznie wyłączono ją w 1955 roku.
- 5 W którym roku został zaprojektowany pierwszy „prawdziwy” komputer?
 - 1946, była to maszyna IAS (*ang. Institute for Advanced Study*). Budowę zakończono w 1951 roku.

Zapis binarny, bity

Komputery przechowują rozkazy, a także liczby i znaki (ogólnie informacje każdego rodzaju), w postaci ciągów cyfr 0 i 1, zwanych **cyframi binarnymi** lub **bitami** (*ang. Binary digiT*).

Zapis binarny, bity

Komputery przechowują rozkazy, a także liczby i znaki (ogólnie informacje każdego rodzaju), w postaci ciągów cyfr 0 i 1, zwanych **cyframi binarnymi** lub **bitami** (*ang. Binary digiT*).

Cyfrы binarne łatwo jest przechowywać

- 1 Weźmy dowolny układ o dwóch (łatwo rozróżnialnych) stanach.
- 2 Oznaczmy te stany jako „↓” i „↑”.
- 3 Wprowadzając układ w stan „↓” zapisujemy cyfrę 0.
- 4 Wprowadzając układ w stan „↑” zapisujemy cyfrę 1.
- 5 Zbiór takich układów może reprezentować ciąg cyfr 0 i 1.

Zapis binarny, bity

Komputery przechowują rozkazy, a także liczby i znaki (ogólnie informacje każdego rodzaju), w postaci ciągów cyfr 0 i 1, zwanych **cyframi binarnymi** lub **bitami** (*ang. Binary digiT*).

Cyfrы binarne łatwo jest przechowywać

- 1 Weźmy dowolny układ o dwóch (łatwo rozróżnialnych) stanach.
- 2 Oznaczmy te stany jako „↓” i „↑”.
- 3 Wprowadzając układ w stan „↓” zapisujemy cyfrę 0.
- 4 Wprowadzając układ w stan „↑” zapisujemy cyfrę 1.
- 5 Zbiór takich układów może reprezentować ciąg cyfr 0 i 1.

Z takich układów zbudowana jest pamięć komputera.

Zapis binarny, bity

Komputery przechowują rozkazy, a także liczby i znaki (ogólnie informacje każdego rodzaju), w postaci ciągów cyfr 0 i 1, zwanych **cyframi binarnymi** lub **bitami** (*ang. Binary digiT*).

Cyfrы binarne łatwo jest przechowywać

- 1 Weźmy dowolny układ o dwóch (łatwo rozróżnialnych) stanach.
- 2 Oznaczmy te stany jako „↓” i „↑”.
- 3 Wprowadzając układ w stan „↓” zapisujemy cyfrę 0.
- 4 Wprowadzając układ w stan „↑” zapisujemy cyfrę 1.
- 5 Zbiór takich układów może reprezentować ciąg cyfr 0 i 1.

Z takich układów zbudowana jest pamięć komputera.

Ciągi cyfr binarnych (bitów) są nazywane **słowami** (*ang. word*).

Reprezentacje liczb i znaków

Liczby

- 1 Reprezentacja „naturalna” – nieujemne liczby całkowite.
- 2 Reprezentacje „umowne” – liczby ujemne, liczby niecałkowite.

Reprezentacje liczb i znaków

Liczby

- 1 Reprezentacja „naturalna” – nieujemne liczby całkowite.
- 2 Reprezentacje „umowne” – liczby ujemne, liczby niecałkowite.

Znaki

Tylko reprezentacje „umowne” – **zbiory znaków** (*ang. character set*).

- ASCII (m. in. $1000001_{bin} = A$, $1000010_{bin} = B$ itd.)
- Strony kodowe, standardy ISO-8859, Unicode.

Reprezentacje liczb i znaków

Liczby

- 1 Reprezentacja „naturalna” – nieujemne liczby całkowite.
- 2 Reprezentacje „umowne” – liczby ujemne, liczby niecałkowite.

Znaki

Tylko reprezentacje „umowne” – **zbiory znaków** (*ang. character set*).

- ASCII (m. in. $1000001_{bin} = A$, $1000010_{bin} = B$ itd.)
- Strony kodowe, standardy ISO-8859, Unicode.

Dane (*ang. data*)

Liczby i znaki w zapisie binarnym.

Typy danych

Dla słowa N -bitowego

$$\underbrace{1}_{b_{N-1}} \underbrace{1}_{b_{N-2}} 011010 \dots 010111 \underbrace{0}_{b_1} \underbrace{1}_{b_0}$$

- b_j – bit (cyfra binarna) na pozycji $j = 0, 1, \dots, N - 1$
- **Waga** bitu odpowiada jego pozycji w słowie:
 - b_0 – najmniej znaczący (najmłodszy) bit.
 - b_{N-1} – najbardziej znaczący (najstarszy) bit.

Typy danych

Dla słowa N -bitowego

$$\underbrace{1}_{b_{N-1}} \underbrace{1}_{b_{N-2}} 011010 \dots 010111 \underbrace{0}_{b_1} \underbrace{1}_{b_0}$$

- b_j – bit (cyfra binarna) na pozycji $j = 0, 1, \dots, N - 1$
- **Waga** bitu odpowiada jego pozycji w słowie:
 - b_0 – najmniej znaczący (najmłodszy) bit.
 - b_{N-1} – najbardziej znaczący (najstarszy) bit.

Typ danych (*ang. data type*)

Określa **rozmiary** danych (np. jaka liczba bitów ma być wykorzystywana do zapisania znaku) oraz **interpretację** zapisu binarnego (tzn. jakie ma być znaczenie poszczególnych bitów).

Reprezentacja bezznakowa dla liczb całkowitych

b – nieujemna liczba całkowita

$$b = \sum_{j=0}^{N-1} b_j 2^j = b_{N-1} 2^{N-1} + b_{N-2} 2^{N-2} + \dots + b_1 2^1 + b_0 2^0$$

Reprezentacja bezznakowa dla liczb całkowitych

b – nieujemna liczba całkowita

$$b = \sum_{j=0}^{N-1} b_j 2^j = b_{N-1} 2^{N-1} + b_{N-2} 2^{N-2} + \dots + b_1 2^1 + b_0 2^0$$

Typy danych dla reprezentacji bezznakowej

$N = 8$: od 0 do $255 = 2^8 - 1$

$N = 16$: od 0 do $65535 = 2^{16} - 1$

$N = 32$: od 0 do $2^{32} - 1$

$N = 64$: od 0 do $2^{64} - 1$

$N = 128$: od 0 do $2^{128} - 1$

Reprezentacja bezznakowa dla liczb całkowitych

b – nieujemna liczba całkowita

$$b = \sum_{j=0}^{N-1} b_j 2^j = b_{N-1} 2^{N-1} + b_{N-2} 2^{N-2} + \dots + b_1 2^1 + b_0 2^0$$

Typy danych dla reprezentacji bezznakowej

$N = 8$: od 0 do $255 = 2^8 - 1$

$N = 16$: od 0 do $65535 = 2^{16} - 1$

$N = 32$: od 0 do $2^{32} - 1$

$N = 64$: od 0 do $2^{64} - 1$

$N = 128$: od 0 do $2^{128} - 1$

Działania przeprowadza się na danych **tego samego typu**.

Arytmetyka w reprezentacji bezznakowej – przykłady

$$10010101 + 01001110 = 11100011 \quad (149 + 78 = 227)$$

	0	0	1	1	1	0	0	
<i>a</i>	1	0	0	1	0	1	0	1
<i>b</i>	0	1	0	0	1	1	1	0
<i>a + b</i>	1	1	1	0	0	0	1	1

Arytmetyka w reprezentacji bezznakowej – przykłady

$$10010101 + 01001110 = 11100011 \quad (149 + 78 = 227)$$

	0	0	1	1	1	0	0
<i>a</i>	1	0	0	1	0	1	0
<i>b</i>	0	1	0	0	1	1	1
<i>a + b</i>	1	1	1	0	0	0	1

$$10110111 + 01101011 = 100100010 \quad (183 + 107 = ?)$$

1	1	1	1	1	1	1	1
<i>a</i>	1	0	1	1	0	1	1
<i>b</i>	0	1	1	0	1	0	1
<i>a + b</i>	0	0	1	0	0	0	1

Dodawanie w reprezentacji bezznakowej

Wprowadzamy funkcje:

$$PAR(x, y, z) = (x + y + z) \% 2$$

$$MAJ(x, y, z) = (x + y + z) / 2$$

gdzie $x, y, z \in \{0, 1\}$, a znak / oznacza dzielenie z pominięciem reszty.

Dodawanie w reprezentacji bezznakowej

Wprowadzamy funkcje:

$$PAR(x, y, z) = (x + y + z) \% 2$$

$$MAJ(x, y, z) = (x + y + z) / 2$$

gdzie $x, y, z \in \{0, 1\}$, a znak $/$ oznacza dzielenie z pominięciem reszty.

Definiujemy ciąg $c_0 = 0, c_j = MAJ(a_{j-1}, b_{j-1}, c_{j-1})$ dla $j = 1, 2, \dots, N$.

Dodawanie w reprezentacji bezznakowej

Wprowadzamy funkcje:

$$PAR(x, y, z) = (x + y + z) \% 2$$

$$MAJ(x, y, z) = (x + y + z) / 2$$

gdzie $x, y, z \in \{0, 1\}$, a znak $/$ oznacza dzielenie z pominięciem reszty.

Definiujemy ciąg $c_0 = 0, c_j = MAJ(a_{j-1}, b_{j-1}, c_{j-1})$ dla $j = 1, 2, \dots, N$.

Wtedy c_j jest **przeniesieniem** (*ang. carry*) z pozycji $(j - 1)$ na pozycję j podczas dodawania, więc mamy $(a + b)_j = PAR(a_j, b_j, c_j)$ dla $j = 0, 1, \dots, N - 1$ oraz $(a + b)_N = c_N$.

Własności reprezentacji bezznakowej

- Tylko liczby całkowite nieujemne.
- Naturalna arytmetyka.
- Możliwość wystąpienia **przepełnienia** (*ang. overflow*) przy dodawaniu.
- Problem z odejmowaniem liczby większej od mniejszej.

Reprezentacja uzupełnienia do 2 dla liczb całkowitych

b – liczba całkowita (może być ujemna)

$$b = -b_{N-1}2^{N-1} + \sum_{j=0}^{N-2} b_j 2^j = -b_{N-1}2^{N-1} + b_{N-2}2^{N-2} + \dots + b_1 2^1 + b_0 2^0$$

Reprezentacja uzupełnienia do 2 dla liczb całkowitych

b – liczba całkowita (może być ujemna)

$$b = -b_{N-1}2^{N-1} + \sum_{j=0}^{N-2} b_j 2^j = -b_{N-1}2^{N-1} + b_{N-2}2^{N-2} + \dots + b_1 2^1 + b_0 2^0$$

Typy danych dla reprezentacji uzupełnienia do 2

$N = 8$: od $-2^7 = -128$ do $127 = 2^7 - 1$

$N = 16$: od $-2^{15} = -32768$ do $32767 = 2^{15} - 1$

$N = 32$: od -2^{31} do $2^{31} - 1$

$N = 64$: od -2^{63} do $2^{63} - 1$

$N = 128$: od -2^{127} do $2^{127} - 1$

Reprezentacja uzupełnienia do 2 dla liczb całkowitych

b – liczba całkowita (może być ujemna)

$$b = -b_{N-1}2^{N-1} + \sum_{j=0}^{N-2} b_j 2^j = -b_{N-1}2^{N-1} + b_{N-2}2^{N-2} + \dots + b_1 2^1 + b_0 2^0$$

Typy danych dla reprezentacji uzupełnienia do 2

$N = 8$: od $-2^7 = -128$ do $127 = 2^7 - 1$

$N = 16$: od $-2^{15} = -32768$ do $32767 = 2^{15} - 1$

$N = 32$: od -2^{31} do $2^{31} - 1$

$N = 64$: od -2^{63} do $2^{63} - 1$

$N = 128$: od -2^{127} do $2^{127} - 1$

Działania przeprowadza się na danych **tego samego typu**.

Arytmetyka w reprezentacji uzupełnienia do 2 – przykłady

$$10110111 + 01101011 = (1)00100010 \quad (-73 + 107 = 34)$$

(1)	1	1	1	1	1	1	1	1
<i>a</i>	1	0	1	1	0	1	1	1
<i>b</i>	0	1	1	0	1	0	1	1
<i>a + b</i>	0	0	1	0	0	0	1	0

Bit z przeniesienia nie jest uwzględniany przy zapisywaniu wyniku.

Arytmetyka w reprezentacji uzupełnienia do 2 – przykłady

$$10110111 + 01101011 = (1)00100010 \quad (-73 + 107 = 34)$$

(1)	1	1	1	1	1	1	1	1
<i>a</i>	1	0	1	1	0	1	1	1
<i>b</i>	0	1	1	0	1	0	1	1
<i>a + b</i>	0	0	1	0	0	0	1	0

Bit z przeniesienia nie jest uwzględniany przy zapisywaniu wyniku.

$$00110111 + 01101011 = 10100010 \quad (55 + 107 = ?)$$

	1	1	1	1	1	1	1	1
<i>a</i>	0	0	1	1	0	1	1	1
<i>b</i>	0	1	1	0	1	0	1	1
<i>a + b</i>	1	0	1	0	0	0	1	0

Własności reprezentacji uzupełnienia do 2

- Tylko liczby całkowite (mogą być ujemne).
- Dodawanie i odejmowanie jak dla reprezentacji bezznakowej.
- Możliwość wystąpienia **przepełnienia** (w innych okolicznościach, niż dla reprezentacji bezznakowej).
- Przy danej liczbie bitów największa wartość jest o połowę mniejsza od największej wartości dla analogicznego typu danych w reprezentacji bezznakowej.

Reprezentowanie liczb niecałkowitych

Obserwacja

Dowolną liczbę rzeczywistą można przedstawić w postaci:

$$r = \pm S_r \times 2^{W_r}$$

gdzie $S_r \in [0, 2)$ oraz W_r jest liczbą całkowitą.

Reprezentowanie liczb niecałkowitych

Obserwacja

Dowolną liczbę rzeczywistą można przedstawić w postaci:

$$r = \pm S_r \times 2^{W_r}$$

gdzie $S_r \in [0, 2)$ oraz W_r jest liczbą całkowitą.

Idea reprezentacji zmiennoprzecinkowej (*ang. floating point*)

Słowo reprezentujące liczbę można podzielić na trzy części:

- 1 Znak – 1 bit.
- 2 Wykładnik (W_r) – zwykle od ok. 1/5 do ok. 1/4 bitów.
- 3 Mantysa (*ang. mantissa, significand*) lub cecha – pozostałe m bitów (np. $S_r = 1 + b_{m-1} \cdot 2^{-1} + b_{m-2} \cdot 2^{-2} + b_{m-3} \cdot 2^{-3} + \dots$).

Zmiennoprzecinkowe reprezentacje liczb

Istnieje wiele reprezentacji dla tej samej długości słowa

- 1 Dokładność zależy od długości (liczby bitów) mantysy.
- 2 Zakres wartości zależy od długości (liczby bitów) wykładnika.
- 3 Przy przeprowadzaniu operacji wyniki **zaokrągla się** tak, aby można je było zapisać z pomocą wybranej liczby bitów mantysy i wykładnika.

Problem niezgodności między różnymi reprezentacjami został rozwiązany poprzez wprowadzenie międzynarodowej normy – standard IEEE 754.

Zmiennoprzecinkowe reprezentacje liczb

Istnieje wiele reprezentacji dla tej samej długości słowa

- 1 Dokładność zależy od długości (liczby bitów) mantysy.
- 2 Zakres wartości zależy od długości (liczby bitów) wykładnika.
- 3 Przy przeprowadzaniu operacji wyniki **zaokrągla się** tak, aby można je było zapisać z pomocą wybranej liczby bitów mantysy i wykładnika.

Problem niezgodności między różnymi reprezentacjami został rozwiązany poprzez wprowadzenie międzynarodowej normy – standard IEEE 754.

Postać znormalizowana

Zakłada się, że najbardziej znaczący bit mantysy jest jedyneką i nie jest on przechowywany. Wówczas jednak występuje problem reprezentacji dla zera.

Wartości specjalne według standardu IEEE 754

Reprezentacje 0 i ∞

- 1 Liczba 0 jest reprezentowana przez słowo, w którym wszystkie bity mantysy i wykładnika są zerami (**dwie reprezentacje liczby 0**).
- 2 $\pm\infty$ jest reprezentowana przez słowo, w którym wszystkie bity wykładnika są jedynkami a wszystkie bity mantysy – zerami.

Wartości specjalne według standardu IEEE 754

Reprezentacje 0 i ∞

- 1 Liczba 0 jest reprezentowana przez słowo, w którym wszystkie bity mantysy i wykładnika są zerami (**dwie reprezentacje liczby 0**).
- 2 $\pm\infty$ jest reprezentowana przez słowo, w którym wszystkie bity wykładnika są jedynkami a wszystkie bity mantysy – zerami.

Niedomiar i postać zdenormalizowana

- 1 Obszar niedomiaru odpowiada liczbom, które mają zbyt małą wartość bezwzględną, aby można było je przedstawić w postaci znormalizowanej.
- 2 **Zasada:** Jeżeli pole wykładnika zawiera ciąg samych zer, to najbardziej znaczący bit mantysy jest **zerem** (nie jest on przechowywany) i przyjmuje się, że **wykładnik ma minimalną wartość**.

Reprezentacja o pojedynczej precyzji (*ang. single precision*)

$$r = \pm S_r \times 2^{W_r}$$

- 1 Słowo 32-bitowe.
- 2 b_{31} = znak.
- 3 $b_{30} \dots b_{23} = W_r$ (od -126 do 127).
- 4 Postać znormalizowana:

$$S_r = 1 + \sum_{j=1}^{23} b_{23-j} 2^{-j}$$

- 5 Postać zdenormalizowana:

$$W_r = -126, \quad S_r = \sum_{j=1}^{23} b_{23-j} 2^{-j}$$

Reprezentacja o podwójnej precyzji (*ang. double precision*)

$$r = \pm S_r \times 2^{W_r}$$

- 1 Słowo 64-bitowe.
- 2 $b_{63} = \text{znak}$.
- 3 $b_{62} \dots b_{52} = W_r$ (od -1022 do 1023).
- 4 Postać znormalizowana:

$$S_r = 1 + \sum_{j=1}^{52} b_{52-j} 2^{-j}$$

- 5 Postać zdenormalizowana:

$$W_r = -1022, \quad S_r = \sum_{j=1}^{52} b_{52-j} 2^{-j}$$

Precyzja dla zmiennoprzecinkowych reprezentacji liczb

NaN (*ang. Not a Number*)

Dla reprezentacji zmiennoprzecinkowej – ciąg bitów nie reprezentujący liczby (W_r zawiera ciąg jedynek, S_r zawiera co najmniej jedną jedynekę).

Precyzja dla zmiennoprzecinkowych reprezentacji liczb

NaN (*ang. Not a Number*)

Dla reprezentacji zmiennoprzecinkowej – ciąg bitów nie reprezentujący liczby (W_r zawiera ciąg jedynek, S_r zawiera co najmniej jedną jedynekę).

Cyfry znaczące (*ang. significant digits*)

Bierzemy pod uwagę tylko pewną liczbę cyfr o największych wagach.

- 24 cyfry dwójkowe dla pojedynczej precyzji (postać znormalizowana).
- 53 cyfry dwójkowe dla podwójnej precyzji (postać znormalizowana).

Precyzja dla zmiennoprzecinkowych reprezentacji liczb

NaN (*ang. Not a Number*)

Dla reprezentacji zmiennoprzecinkowej – ciąg bitów nie reprezentujący liczby (W_r zawiera ciąg jedynek, S_r zawiera co najmniej jedną jedynkę).

Cyfry znaczące (*ang. significant digits*)

Bierzemy pod uwagę tylko pewną liczbę cyfr o największych wagach.

- 24 cyfry dwójkowe dla pojedynczej precyzji (postać znormalizowana).
- 53 cyfry dwójkowe dla podwójnej precyzji (postać znormalizowana).

Liczba znaczących cyfr dwójkowych nie przelicza się dokładnie na cyfry dziesiętne.

- Około 7 cyfr dziesiętnych dla pojedynczej precyzji.
- Około 14 cyfr dziesiętnych dla podwójnej precyzji.

Zaokrąglenia dla zmiennoprzecinkowych reprezentacji liczb

Zaokrąglenia (*ang. rounding*)

- 1 Dokładność obliczeń zależy od liczby cyfr znaczących i rzędu wielkości danych (wartości ich wykładników).
- 2 Wyniki obliczeń, których nie można zapisać w danej reprezentacji, muszą być **zaokrąglane** do wartości, które można zapisać.
- 3 Dopuszczalne są różne **metody zaokrąglania** (*ang. rounding modes*):
 - Do najbliższej (dwa warianty).
 - W górę.
 - W dół.
 - W kierunku zera (odcinanie).
- 4 To, która metoda zaokrąglania jest używana, zależy od wykorzystywanego sprzętu i oprogramowania.

Wartości, które można reprezentować

Liczba różnych wartości możliwych do zapisania

- 1 Dla reprezentacji o pojedynczej precyzji: $2^{32} - 2^{24} + 1$ (wykładnik może być ciągiem samych jedynek tylko dla $\pm\infty$ oraz mamy dwie reprezentacje liczby 0).
- 2 Dla 32-bitowego typu danych w reprezentacji uzupełnienia do 2: 2^{32} .
- 3 Dla reprezentacji o podwójnej precyzji: $2^{64} - 2^{53} + 1$ (jak wyżej).
- 4 Dla 64-bitowego typu danych w reprezentacji uzupełnienia do 2: 2^{64} .

Wartości, które można reprezentować

Liczba różnych wartości możliwych do zapisania

- 1 Dla reprezentacji o pojedynczej precyzji: $2^{32} - 2^{24} + 1$ (wykładnik może być ciągiem samych jedynek tylko dla $\pm\infty$ oraz mamy dwie reprezentacje liczby 0).
- 2 Dla 32-bitowego typu danych w reprezentacji uzupełnienia do 2: 2^{32} .
- 3 Dla reprezentacji o podwójnej precyzji: $2^{64} - 2^{53} + 1$ (jak wyżej).
- 4 Dla 64-bitowego typu danych w reprezentacji uzupełnienia do 2: 2^{64} .

Rozmieszczenie wartości możliwych do zapisania na osi liczbowej

- 1 „Gęsto” wokół zera.
- 2 Odstępy między nimi rosną wraz ze wzrostem wartości wykładnika.
- 3 Dla danej wartości wykładnika odstępy zależą od liczby bitów mantysy.

Notacja szesnastkowa dla liczb całkowitych

Zapis szesnastkowy (*ang. hexadecimal notation*)

Dowolną nieujemną liczbę całkowitą można rozłożyć na potęgi liczby 16

$$x = \sum_{j=0}^{N-1} h_j(x)(16)^j$$

gdzie $h_j(x) \in \{0, 1, 2, \dots, 14, 15\}$ są **cyframi szesnastkowymi**.

Notacja szesnastkowa dla liczb całkowitych

Zapis szesnastkowy (*ang. hexadecimal notation*)

Dowolną nieujemną liczbę całkowitą można rozłożyć na potęgi liczby 16

$$x = \sum_{j=0}^{N-1} h_j(x)(16)^j$$

gdzie $h_j(x) \in \{0, 1, 2, \dots, 14, 15\}$ są **cyframi szesnastkowymi**.

Umowa

- 1 Cyfry szesnastkowe od 0 do 9 oznacza się takimi samymi symbolami, jak odpowiadające im cyfry dziesiętne.
- 2 Cyfry szesnastkowe odpowiadające liczbom od 10 do 15 oznacza się literami od 'A' do 'F' lub od 'a' do 'f', odpowiednio.

Związek notacji szesnastkowej z dwójkową

Obserwacja

Każda cyfra w notacji szesnastkowej odpowiada czterem bitom w zapisie binarnym tej samej liczby.

0	0000	1	0001	2	0010	3	0011
4	0100	5	0101	6	0110	7	0111
8	1000	9	1001	A	1010	B	1011
C	1100	D	1101	E	1110	F	1111

Związek notacji szesnastkowej z dwójkową

Obserwacja

Każda cyfra w notacji szesnastkowej odpowiada czterem bitom w zapisie binarym tej samej liczby.

0	0000	1	0001	2	0010	3	0011
4	0100	5	0101	6	0110	7	0111
8	1000	9	1001	A	1010	B	1011
C	1100	D	1101	E	1110	F	1111

Przykłady

$1111\ 1111_{bin} = FF_{hex} = 255$, $1111\ 1111\ 1111\ 1111_{bin} = FFFF_{hex} = 65535$,
 $1011\ 1101_{bin} = BD_{hex} = 189$, $1010\ 0011\ 1001\ 0101_{bin} = A395_{hex} = ?$,
 $1001\ 0010_{bin} = 92_{hex} = 146$, $0001\ 0110\ 0100\ 0111_{bin} = 1647_{hex} = ?$,

Czym są zbiory znaków?

Zbiór znaków (*ang. character set*)

System, zgodnie z którym symbole graficzne z pewnego zbioru (znaki) są reprezentowane w określony sposób przez ciągi bitów (słowa). Definiuje także przyporządkowanie symbolom określonych wartości liczbowych (**kody znaków**).

Czym są zbiory znaków?

Zbiór znaków (*ang. character set*)

System, zgodnie z którym symbole graficzne z pewnego zbioru (znaki) są reprezentowane w określony sposób przez ciągi bitów (słowa). Definiuje także przyporządkowanie symbolom określonych wartości liczbowych (**kody znaków**).

Kodowanie (*ang. encoding*)

Zastępowanie symboli z określonego zbioru (np. graficznych) symbolami z innego zbioru (np. liczby) zgodnie z ustalonymi zasadami. **Nie ma na celu ukrywania informacji.**

Czym są zbiory znaków?

Zbiór znaków (*ang. character set*)

System, zgodnie z którym symbole graficzne z pewnego zbioru (znaki) są reprezentowane w określony sposób przez ciągi bitów (słowa). Definiuje także przyporządkowanie symbolom określonych wartości liczbowych (**kody znaków**).

Kodowanie (*ang. encoding*)

Zastępowanie symboli z określonego zbioru (np. graficznych) symbolami z innego zbioru (np. liczby) zgodnie z ustalonymi zasadami. **Nie ma na celu ukrywania informacji.**

Istnieją „niecyfrowe” systemy o takim charakterze

- 1 Alfabet Morse'a (*ang. Morse code*).
- 2 Pismo punktowe Braille'a.

Zbiór znaków ASCII

ASCII (American Standard Code for Information Interchange)

- 1 Najstarszy standardowy system reprezentowania znaków z pomocą ciągów bitów (słów).
- 2 Zbiór znaków określający reprezentacje binarne oraz (w związku z tym) liczbowe dla 128 znaków (kody od 0 do 127) – 7 bitów.
- 3 Obejmuje litery angielskiego alfabetu (wielkie i małe), cyfry, znaki przestankowe i symbole matematyczne, symbole specjalne (np. \$) i tak zwane **znaki sterujące** (*ang. control characters*).

Zbiór znaków ASCII

ASCII (American Standard Code for Information Interchange)

- 1 Najstarszy standardowy system reprezentowania znaków z pomocą ciągów bitów (słów).
- 2 Zbiór znaków określający reprezentacje binarne oraz (w związku z tym) liczbowe dla 128 znaków (kody od 0 do 127) – 7 bitów.
- 3 Obejmuje litery angielskiego alfabetu (wielkie i małe), cyfry, znaki przestankowe i symbole matematyczne, symbole specjalne (np. \$) i tak zwane **znaki sterujące** (*ang. control characters*).

Znaki sterujące (kody ASCII od 0 do 31)

Reprezentują operacje, np. przejście do następnego wiersza (*ang. line feed*) lub przesunięcie głowicy drukującej na początek wiersza (*ang. carriage return*) albo przesunięcie jej o jedną pozycję wstecz (*ang. backspace*). Wykorzystywane m. in. do formatowania tekstu.

Zbiór znaków ASCII – ograniczenia

Zbiór znaków ASCII nie wystarcza do wszystkich zastosowań

- 1 Litery akcentowane.
- 2 Litery z innych alfabetów (np. cyrylica, alfabet grecki).
- 3 Znaki nie będące literami (np. symbole matematyczne).
- 4 Języki, w których nie używa się liter.

Zbiór znaków ASCII – ograniczenia

Zbiór znaków ASCII nie wystarcza do wszystkich zastosowań

- 1 Litery akcentowane.
- 2 Litery z innych alfabetów (np. cyrylica, alfabet grecki).
- 3 Znaki nie będące literami (np. symbole matematyczne).
- 4 Języki, w których nie używa się liter.

Dodatkowo zakładano, że każdy znak będzie zajmował taką samą przestrzeń na wydruku lub na ekranie terminala (tzn. do drukowania każdego znaku był przeznaczony prostokąt o określonej wysokości i szerokości jednakowej dla wszystkich znaków).

Rozszerzenia ASCII

Bajt (*ang. byte*)

Słowo 8-bitowe (ciąg 8 bitów).

Rozszerzenia ASCII

Bajt (*ang. byte*)

Słowo 8-bitowe (ciąg 8 bitów).

Zasada „1 znak – 1 bajt”

- 1 ASCII wymaga stosowania 7 bitów do zapisywania jednego znaku.
- 2 Dla współczesnych komputerów 1 bajt jest podstawową jednostką pojemności pamięci.
- 3 W „naturalnym” zapisie znaków ASCII każdy znak zajmuje 1 bajt, ale tylko najmniej znaczących 7 bitów ma ustalone znaczenie.
- 4 Dla znaków ASCII najbardziej znaczący bit w bajcie jest zerem.
- 5 Pozostają do wykorzystania bajty, dla których najbardziej znaczący bit jest jedyką.

Strony kodowe IBM

Strona kodowa (*ang. code page*)

- 1 Zbiór znaków, w którym znaki o kodach 0 ... 127 są zgodne z ASCII.
- 2 Pozostałe kody oznaczają znaki spoza zbioru ASCII.

Strony kodowe IBM

Strona kodowa (*ang. code page*)

- 1 Zbiór znaków, w którym znaki o kodach 0 ... 127 są zgodne z ASCII.
- 2 Pozostałe kody oznaczają znaki spoza zbioru ASCII.

Strony kodowe IBM

- 1 Zaprojektowane dla zgodności ze sprzętem („znakowe” tryby działania kart graficznych).
- 2 Przykłady:
 - 850 – Multilingual (Latin-1): języki zachodnioeuropejskie.
 - 852 – Multilingual (Latin-2): języki środkowo- i wschodnioeuropejskie.
 - 855 – Cyrylica.

Strony kodowe Microsoft

Strony kodowe „ANSI”

- 1 Zaprojektowane na podstawie (wątpliwej autentyczności) projektu strony kodowej, która później została przekształcona w standard ISO-8859-1.
- 2 Przykłady:
 - 1250 – Latin-2: języki środkowo- i wschodnioeuropejskie.
 - 1251 – Cyrylica.
 - 1252 – Latin-1: języki zachodnioeuropejskie.

Strony kodowe Microsoft

Strony kodowe „ANSI”

- 1 Zaprojektowane na podstawie (wątpliwej autentyczności) projektu strony kodowej, która później została przekształcona w standard ISO-8859-1.
- 2 Przykłady:
 - 1250 – Latin-2: języki środkowo- i wschodnioeuropejskie.
 - 1251 – Cyrylica.
 - 1252 – Latin-1: języki zachodnioeuropejskie.

Niezgodność ze standardami ISO-8859

Strony kodowe Microsoft są niezgodne ze standardowymi stronami kodowymi ISO, z którymi przez długi czas konkurowały.

Standardy ISO-8859

ISO-8859-1

- Znaki wykorzystywane w językach zachodnioeuropejskich.
- Brak znaków „akcentowanych” z języka polskiego.

ISO-8859-2

- Znaki wykorzystywane w językach środkowoeuropejskich.
- Kody 128 ... 255 przypisane innym znakom, niż w ISO-8859-1.

ISO-8859-15

Rewizja ISO-8859-1 wprowadzająca znak waluty euro.

Standardy ISO-8859

ISO-8859-1

- Znaki wykorzystywane w językach zachodnioeuropejskich.
- Brak znaków „akcentowanych” z języka polskiego.

ISO-8859-2

- Znaki wykorzystywane w językach środkowoeuropejskich.
- Kody 128 ... 255 przypisane innym znakom, niż w ISO-8859-1.

ISO-8859-15

Rewizja ISO-8859-1 wprowadzająca znak waluty euro.

Zasada „1 znak – 1 bajt” powoduje problemy z przenoszeniem tekstów między systemami wykorzystywanymi w różnych krajach.

Standard Unicode

ISO/IEC 10646:2003, Universal Character Set (UCS)

- 1 Koniec zasady „1 znak – 1 bajt”.
- 2 Znaki reprezentowane przez kody wielobajtowe.
- 3 „Rozszerzenie” ASCII (znaki ASCII odpowiadają kodom 0 . . . 127).
- 4 Brak zgodności z ISO-8859 i stronami kodowymi IBM oraz Microsoft.

Standard Unicode

ISO/IEC 10646:2003, Universal Character Set (UCS)

- 1 Koniec zasady „1 znak – 1 bajt”.
- 2 Znaki reprezentowane przez kody wielobajtowe.
- 3 „Rozszerzenie” ASCII (znaki ASCII odpowiadają kodom 0 . . . 127).
- 4 Brak zgodności z ISO-8859 i stronami kodowymi IBM oraz Microsoft.

UTF-8 (8-bit Unicode Transformation Format)

- Jednobajtowe reprezentacje znaków ASCII.
- Dwubajtowe reprezentacje znaków z języków europejskich.
- Najbardziej popularny format Unicode.

Standard Unicode

ISO/IEC 10646:2003, Universal Character Set (UCS)

- 1 Koniec zasady „1 znak – 1 bajt”.
- 2 Znaki reprezentowane przez kody wielobajtowe.
- 3 „Rozszerzenie” ASCII (znaki ASCII odpowiadają kodom 0 . . . 127).
- 4 Brak zgodności z ISO-8859 i stronami kodowymi IBM oraz Microsoft.

UTF-8 (8-bit Unicode Transformation Format)

- Jednobajtowe reprezentacje znaków ASCII.
- Dwubajtowe reprezentacje znaków z języków europejskich.
- Najbardziej popularny format Unicode.

Dzięki Unicode możliwe jest tworzenie **uniwersalnych** dokumentów tekstowych (tzn. tekstów wyświetlanych wszędzie tak samo).

Struktura pamięci głównej

Lokacje (*ang. location*)

- 1 Jednakowe rozmiary (przeważnie 8 bitów).
- 2 Ustalona kolejność bitów.
- 3 Wykorzystywane w grupach do przechowywania dłuższych słów.
 - Wartość całkowita 32-bitowa – 4 kolejne lokacje.
 - Wartość zmiennoprzecinkowa o podwójnej precyzji – 8 kolejnych lokacji.
- 4 Numeracja – adresy (*ang. address*).

Struktura pamięci głównej

Lokacje (*ang. location*)

- 1 Jednakowe rozmiary (przeważnie 8 bitów).
- 2 Ustalona kolejność bitów.
- 3 Wykorzystywane w grupach do przechowywania dłuższych słów.
 - Wartość całkowita 32-bitowa – 4 kolejne lokacje.
 - Wartość zmiennoprzecinkowa o podwójnej precyzji – 8 kolejnych lokacji.
- 4 Numeracja – adresy (*ang. address*).

Przestrzeń adresowa pamięci (*ang. memory address space*)

Zbiór wszystkich możliwych adresów lokacji dla danego komputera (procesora).

Jednostki pojemności pamięci

1 bajt (1 B), słowo 8-bitowe

Podstawowa jednostka pojemności pamięci.

Jednostki pojemności pamięci

1 bajt (1 B), słowo 8-bitowe

Podstawowa jednostka pojemności pamięci.

Jednostki pojemności pamięci

1 KiB (1 kilobajt) = 1024 B

1 MiB (1 megabajt) = 1024 KiB

1 GiB (1 gigabajt) = 1024 MiB

1 TiB (1 terabajt) = 1024 GiB

Jednostki pojemności pamięci

1 bajt (1 B), słowo 8-bitowe

Podstawowa jednostka pojemności pamięci.

Jednostki pojemności pamięci

1 KiB (1 kilobajt) = 1024 B

1 MiB (1 megabajt) = 1024 KiB

1 GiB (1 gigabajt) = 1024 MiB

1 TiB (1 terabajt) = 1024 GiB

W innym kontekście mogą być stosowane inne jednostki

Np. pojemność dysków twardych często podaje się w GB lub TB, gdzie

$$1 \text{ GB} = 10^9 \text{ B}$$

$$1 \text{ TB} = 10^{12} \text{ B}$$

Problem *little endian* i *big endian*

Jak zapisać w pamięci słowo dwubajtowe

little endian – mniej znaczący bajt zapisuje się pod **mniejszym** adresem.

big endian – mniej znaczący bajt zapisuje się pod **większym** adresem.

Problem *little endian* i *big endian*

Jak zapisać w pamięci słowo dwubajtowe

little endian – mniej znaczący bajt zapisuje się pod **mniejszym** adresem.

big endian – mniej znaczący bajt zapisuje się pod **większym** adresem.

Każde z rozwiązań ma wady i zalety, ale nie można przesądzić które z nich jest lepsze.

Problem *little endian* i *big endian*

Jak zapisać w pamięci słowo dwubajtowe

little endian – mniej znaczący bajt zapisuje się pod **mniejszym** adresem.

big endian – mniej znaczący bajt zapisuje się pod **większym** adresem.

Każde z rozwiązań ma wady i zalety, ale nie można przesądzić które z nich jest lepsze.

Architektura komputera decyduje o tym, które z nich jest wykorzystywane.

Problem *little endian* i *big endian*

Jak zapisać w pamięci słowo dwubajtowe

little endian – mniej znaczący bajt zapisuje się pod **mniejszym** adresem.

big endian – mniej znaczący bajt zapisuje się pod **większym** adresem.

Każde z rozwiązań ma wady i zalety, ale nie można przesądzić które z nich jest lepsze.

Architektura komputera decyduje o tym, które z nich jest wykorzystywane.

W niektórych przypadkach kolejność dla danych zmiennoprzecinkowych może być inna, niż dla danych całkowitych.

Pamięć RAM

RAM (*ang. random access memory*)

Pamięć ulotna, niezależny dostęp do każdej lokacji.

statyczna (*ang. static RAM*) – zbudowana z bramek logicznych.

dynamiczna (*ang. dynamic RAM*) – zbudowana z kondensatorów.

Pamięć RAM

RAM (*ang. random access memory*)

Pamięć ulotna, niezależny dostęp do każdej lokacji.

statyczna (*ang. static RAM*) – zbudowana z bramek logicznych.

dynamiczna (*ang. dynamic RAM*) – zbudowana z kondensatorów.

Odświeżanie (*ang. refresh*)

Konieczne w przypadku dynamicznych pamięci RAM, polega na okresowym odczytywaniu zapisanych bitów.

Pamięć RAM

RAM (*ang. random access memory*)

Pamięć ulotna, niezależny dostęp do każdej lokacji.

styczna (*ang. static RAM*) – zbudowana z bramek logicznych.

dynamiczna (*ang. dynamic RAM*) – zbudowana z kondensatorów.

Odświeżanie (*ang. refresh*)

Konieczne w przypadku dynamicznych pamięci RAM, polega na okresowym odczytywaniu zapisanych bitów.

Schowki (*ang. cache*)

Zbudowanie całej pamięci komputera ze statycznych pamięci RAM byłoby zbyt kosztowne – stosuje się tymczasowe przechowywanie najbardziej potrzebnych danych i rozkazów procesora w takich pamięciach.

Struktura pamięci

Z punktu widzenia procesora (logiczna)

Pamięć jest ciągiem lokacji o kolejnych adresach („płaska” struktura).

Struktura pamięci

Z punktu widzenia procesora (logiczna)

Pamięć jest ciągiem lokacji o kolejnych adresach („płaska” struktura).

„Dziura” w pamięci (*ang. memory hole*)

Obszar w przestrzeni adresowej pamięci, któremu nie są przypisane żadne lokacje w fizycznej pamięci.

Struktura pamięci

Z punktu widzenia procesora (logiczna)

Pamięć jest ciągiem lokacji o kolejnych adresach („płaska” struktura).

„Dziura” w pamięci (*ang. memory hole*)

Obszar w przestrzeni adresowej pamięci, któremu nie są przypisane żadne lokacje w fizycznej pamięci.

Moduł pamięci (*ang. memory module*)

Urządzenie elektroniczne zawierające pewną liczbę tzw. **kości pamięci** (*ang. memory chip*), w których są ulokowane **komórki pamięci** (*ang. memory cell*).

Fizyczna struktura pamięci

Matryca pamięci (*ang. memory array*)

Prostokątna struktura, w której ulokowane są komórki pamięci o pojemności 1 bitu. Odczyt i zapis odbywa się poprzez wybór **wiersza** (*ang. row*) i **kolumny** (*ang. column*) w odpowiedniej kolejności.

Fizyczna struktura pamięci

Matryca pamięci (*ang. memory array*)

Prostokątna struktura, w której ulokowane są komórki pamięci o pojemności 1 bitu. Odczyt i zapis odbywa się poprzez wybór **wiersza** (*ang. row*) i **kolumny** (*ang. column*) w odpowiedniej kolejności.

Kontroler pamięci (*ang. memory controller*)

Urządzenie umożliwiające komunikację między procesorem (oraz urządzeniami I/O) i modułami pamięci (identyfikacja fizycznych komórek pamięci na podstawie adresów „liniowych”).

Fizyczna struktura pamięci

Matryca pamięci (*ang. memory array*)

Prostokątna struktura, w której ulokowane są komórki pamięci o pojemności 1 bitu. Odczyt i zapis odbywa się poprzez wybór **wiersza** (*ang. row*) i **kolumny** (*ang. column*) w odpowiedniej kolejności.

Kontroler pamięci (*ang. memory controller*)

Urządzenie umożliwiające komunikację między procesorem (oraz urządzeniami I/O) i modułami pamięci (identyfikacja fizycznych komórek pamięci na podstawie adresów „liniowych”).

Magistrala pamięci (*ang. memory bus*)

System połączeń między kontrolerem pamięci i modułami pamięci wraz z określonymi zasadami przesyłania danych.

Pamięci DRAM typu DDR, DDR2 i DDR3

DDR (*ang. double data rate*)

Technologia umożliwiająca przeprowadzanie operacji (np. z udziałem modułów pamięci) dwukrotnie w ciągu każdego cyklu zegara magistrali (do 1600 MB/s).

Pamięci DRAM typu DDR, DDR2 i DDR3

DDR (*ang. double data rate*)

Technologia umożliwiająca przeprowadzanie operacji (np. z udziałem modułów pamięci) dwukrotnie w ciągu każdego cyklu zegara magistrali (do 1600 MB/s).

DDR2

DDR z dwukrotnie większą częstotliwością taktowania magistrali, przy **nie zmienionej** częstotliwości taktowania matrycy pamięci (do 3200 MB/s).

Pamięci DRAM typu DDR, DDR2 i DDR3

DDR (*ang. double data rate*)

Technologia umożliwiająca przeprowadzanie operacji (np. z udziałem modułów pamięci) dwukrotnie w ciągu każdego cyklu zegara magistrali (do 1600 MB/s).

DDR2

DDR z dwukrotnie większą częstotliwością taktowania magistrali, przy **nie zmienionej** częstotliwości taktowania matrycy pamięci (do 3200 MB/s).

DDR3

DDR z czterokrotnie większą częstotliwością taktowania magistrali, przy **nie zmienionej** częstotliwości taktowania matrycy pamięci (do 6400 MB/s).

Pamięci ROM, PROM i EPROM

ROM (*ang. read-only memory*)

Pamięć tego typu zawiera informacje (bity) zakodowane na stałe (najczęściej w postaci matrycy o prostokątnym kształcie).

Pamięci ROM, PROM i EPROM

ROM (*ang. read-only memory*)

Pamięć tego typu zawiera informacje (bity) zakodowane na stałe (najczęściej w postaci matrycy o prostokątnym kształcie).

PROM (*ang. programmable ROM*)

Pamięć typu ROM, w której można „wyzerować” niektóre bity przepuszczając przez nie odpowiednio silny prąd (powodujący zniszczenie połączenia) – operacja jednorazowa!

Pamięci ROM, PROM i EPROM

ROM (*ang. read-only memory*)

Pamięć tego typu zawiera informacje (bity) zakodowane na stałe (najczęściej w postaci matrycy o prostokątnym kształcie).

PROM (*ang. programmable ROM*)

Pamięć typu ROM, w której można „wyzerować” niektóre bity przepuszczając przez nie odpowiednio silny prąd (powodujący zniszczenie połączenia) – operacja jednorazowa!

EPROM (*ang. electrically programmable ROM*)

Pamięć typu ROM, w której można zapisać stan poszczególnych bitów, a później go zmienić (wymaga stosowania światła nadfioletowego do „kasowania” zapisu).

Pamięci EEPROM

Zasada działania pamięci EPROM

- 1 Transystor polowy MOSFET (*ang. metal–oxide–semiconductor field-effect transistor*).
- 2 Dodatkowa bramka, tzw. **bramka pamiętająca** (*ang. floating gate*), między bramką **sterującą** (*ang. control gate*) i kanałem tranzystora.
- 3 Ładunek na bramce pamiętającej ekranuje pole wytwarzane przez bramkę sterującą i zmienia sposób działania tranzystora.

Pamięci EEPROM

Zasada działania pamięci EPROM

- 1 Transystor polowy MOSFET (*ang. metal–oxide–semiconductor field-effect transistor*).
- 2 Dodatkowa bramka, tzw. **bramka pamiętająca** (*ang. floating gate*), między bramką **sterującą** (*ang. control gate*) i kanałem tranzystora.
- 3 Ładunek na bramce pamiętającej ekranuje pole wytwarzane przez bramkę sterującą i zmienia sposób działania tranzystora.

EEPROM (*ang. electrically programmable and erasable ROM*)

Pamięć nieulotna, której zawartość można zmieniać przykładając odpowiednie napięcie elektryczne.

Programowanie i kasowanie pamięci EEPROM

Programowanie – dwie metody

- 1 Przyłożenie odpowiedniego napięcia między źródłem lub drenem i bramką sterującą powoduje „tunelowanie” elektronów przez barierę potencjału między kanałem i bramką pamiętającą.
- 2 Odpowiednio duży prąd między źródłem i drenem tranzystora powoduje, że część elektronów ma wystarczającą dużo energii, by pokonać barierę potencjału między kanałem i bramką pamiętającą.

Programowanie i kasowanie pamięci EEPROM

Programowanie – dwie metody

- 1 Przyłożenie odpowiedniego napięcia między źródłem lub drenem i bramką sterującą powoduje „tunelowanie” elektronów przez barierę potencjału między kanałem i bramką pamiętającą.
- 2 Odpowiednio duży prąd między źródłem i drenem tranzystora powoduje, że część elektronów ma wystarczającą dużo energii, by pokonać barierę potencjału między kanałem i bramką pamiętającą.

Kasowanie

Wykorzystuje zjawisko „tunelowania” elektronów przez barierę potencjału między kanałem i bramką pamiętającą, ale napięcie przykłada się w odwrotnym kierunku w stosunku do operacji programowania (wykasowane komórki zawierają bit o wartości 1).

Pamięci *flash*

Rodzaj pamięci EEPROM, w których kasowanie odbywa się **blokami** (tzn. trzeba kasować wiele komórek **jednocześnie**).

Pamięci *flash*

Rodzaj pamięci EEPROM, w których kasowanie odbywa się **blokami** (tzn. trzeba kasować wiele komórek **jednocześnie**).

Pamięci *flash* NOR i NAND

- 1 Różnią się sposobem ułożenia komórek w matrycy (przypominającym sposób podłączenia w układzie cyfrowym bramek NOR i NAND, odpowiednio).
- 2 W pamięciach NAND każdy blok składa się z pewnej liczby tzw. **stron** (*ang. page*) i każdą stronę trzeba programować w całości (kasuje się całe bloki).
- 3 Pamięci NAND są tańsze, mają większą żywotność i pozwalają na uzyskanie większych gęstości upakowania komórek.

Zapis magentyczny

Twarde dyski (*ang. hard disk drive*)

- 1 Talerze (*ang. platter*).
- 2 Głowice (*ang. head*) zapisująco-odczytujące.
- 3 Namagnesowanie materiału na powierzchni talerza w odpowiedni sposób powoduje przepływ prądu w głowicy przesuwaną się nad talerzem.
- 4 Zapis „prostopadły” (*ang. perpendicular recording*).

Zapis magentyczny

Twarde dyski (*ang. hard disk drive*)

- 1 Talerze (*ang. platter*).
- 2 Głowice (*ang. head*) zapisująco-odczytujące.
- 3 Namagnesowanie materiału na powierzchni talerza w odpowiedni sposób powoduje przepływ prądu w głowicy przesuwanej się nad talerzem.
- 4 Zapis „prostopadły” (*ang. perpendicular recording*).

Taśmy magnetyczne

- Zapis **sekwencyjny**.
- Niższy koszt na 1 bit przechowywanych danych niż dla twardych dysków.
- Zastosowania głównie w archiwizacji danych.

Zapis optyczny

Dyski optyczne – pierwsza generacja (od 1982)

- CD (*ang. Compact Disc*)
- CD-R (*ang. CD-Recordable*)
- CD-RW (*ang. CD-Rewritable*)

Od ok. 650 MB do ok. 700 MB danych na jednym dysku.

Zapis optyczny

Dyski optyczne – pierwsza generacja (od 1982)

- CD (*ang. Compact Disc*)
- CD-R (*ang. CD-Recordable*)
- CD-RW (*ang. CD-Rewritable*)

Od ok. 650 MB do ok. 700 MB danych na jednym dysku.

Dyski optyczne – druga generacja (od 1995-1999)

- DVD (*ang. Digital Versatile Disc*)
- DVD-R (*ang. DVD-Recordable*)
- DVD±RW (*ang. DVD-Rewritable*)
- DVD-RAM

Od ok. 2,7 GB do ok. 9 GB danych na jednym dysku.

Dyski optyczne III i IV generacji

Trzecia generacja (od 2006-2007)

- Blu-ray Disc (BD), ok. 25 GB na jednej warstwie
- Mini Blu-ray Disc (Mini-BD), ok. 8 GB na jednej warstwie
- Blu-ray Disc recordable (BD-R, BD-RE)

Wykorzystuje lasery z azotkiem galu (GaN) o długości fali 405 nm.

Dyski optyczne III i IV generacji

Trzecia generacja (od 2006-2007)

- Blu-ray Disc (BD), ok. 25 GB na jednej warstwie
- Mini Blu-ray Disc (Mini-BD), ok. 8 GB na jednej warstwie
- Blu-ray Disc recordable (BD-R, BD-RE)

Wykorzystuje lasery z azotkiem galu (GaN) o długości fali 405 nm.

Czwarta generacja (technologie w stadium rozwoju)

- HVD (*ang. Holographic Versatile Disc*)
- LS-R (*ang. Layer-Selection-Type Recordable Optical Disk*)
- PCD (*ang. Protein-coated disc*)

Potencjalnie możliwe przechowywanie ok. 1 TB (lub więcej) danych na jednym dysku.

Urządzenia do składowania danych

Składowanie (*ang. storage*)

Przechowywanie przez długi (nieokreślony) czas.

Urządzenia do składowania danych

Składowanie (*ang. storage*)

Przechowywanie przez długi (nieokreślony) czas.

Urządzenia składujące (*ang. storage devices*)

Urządzenia przeznaczone do składowania danych.

Urządzenia do składowania danych

Składowanie (*ang. storage*)

Przechowywanie przez długi (nieokreślony) czas.

Urządzenia składujące (*ang. storage devices*)

Urządzenia przeznaczone do składowania danych.

Nośnik danych (*ang. data carrier*)

Część urządzenia składującego, w której (lub na której) fizycznie zapisywane są dane. Może być wymienna (np. CD, DVD, *Compact Flash*) lub wbudowana na stałe (np. twarde dyski).

Urządzenia do składowania danych

Składowanie (*ang. storage*)

Przechowywanie przez długi (nieokreślony) czas.

Urządzenia składujące (*ang. storage devices*)

Urządzenia przeznaczone do składowania danych.

Nośnik danych (*ang. data carrier*)

Część urządzenia składującego, w której (lub na której) fizycznie zapisywane są dane. Może być wymienna (np. CD, DVD, *Compact Flash*) lub wbudowana na stałe (np. twarde dyski).

Układy odczytująco-zapisujące

Współpracują z nośnikami danych.

Komunikacja z urządzeniami składającymi

Podział na **bloki** (*ang. block*), urządzenia **blokowe** (*ang. block devices*)

- 1 Rozmiary bloku od 512 B do 4 KiB.
- 2 Odczyt i zapis **całych bloków**.
- 3 Położenie danych określane z pomocą **sektorów** (*ang. sector*) o ustalonych numerach.
- 4 Ustalona, całkowita liczba bloków w każdym sektorze.

Komunikacja z urządzeniami składającymi

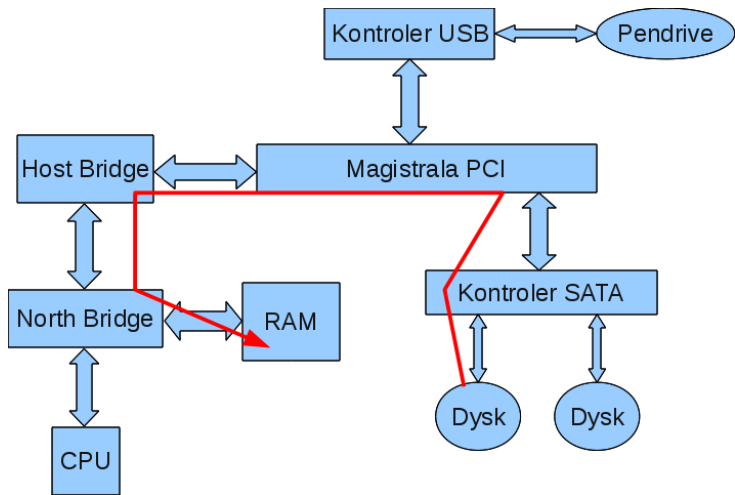
Podział na **bloki** (*ang. block*), urządzenia **blokowe** (*ang. block devices*)

- 1 Rozmiary bloku od 512 B do 4 KiB.
- 2 Odczyt i zapis **całych bloków**.
- 3 Położenie danych określane z pomocą **sektorów** (*ang. sector*) o ustalonych numerach.
- 4 Ustalona, całkowita liczba bloków w każdym sektorze.

Systemy połączeń, zwane **magistralami** (*ang. bus*)

- PCI (*ang. Peripheral Component Interconnect*), PCI Express, HyperTransport
- SCSI (*ang. Small Computer System Interface*), SAS (*ang. Serial Attached SCSI*), SATA (*ang. Serial AT Attachment*)
- FireWire, USB (*ang. Universal Serial Bus*)

Komunikacja z urządzeniami składającymi w PC



Urządzenia blokowe i systemy plikowe

Posługiwanie się numerami sektorów nie byłoby wygodne dla ludzi.

Urządzenia blokowe i systemy plikowe

Posługiwanie się numerami sektorów nie byłoby wygodne dla ludzi.

System plikowy (*ang. file system*)

Pozwala na (stosunkowo) łatwe lokalizowanie składowanych danych z użyciem nazw i struktury **drzewa katalogowego** (*ang. directory tree*).

Urządzenia blokowe i systemy plikowe

Posługiwanie się numerami sektorów nie byłoby wygodne dla ludzi.

System plikowy (*ang. file system*)

Pozwala na (stosunkowo) łatwe lokalizowanie składowanych danych z użyciem nazw i struktury **drzewa katalogowego** (*ang. directory tree*).

Metadane (*ang. metadata*)

Dane wykorzystywane do opisu położenia, rozmiarów oraz innych własności danych zapisywanych przez użytkowników, czyli **danych użytecznych**. Rozmiary oraz organizacja i sposób zapisu metadanych zależą od rodzaju systemu plikowego.

Urządzenia blokowe i systemy plikowe

Posługiwanie się numerami sektorów nie byłoby wygodne dla ludzi.

System plikowy (*ang. file system*)

Pozwala na (stosunkowo) łatwe lokalizowanie składowanych danych z użyciem nazw i struktury **drzewa katalogowego** (*ang. directory tree*).

Metadane (*ang. metadata*)

Dane wykorzystywane do opisu położenia, rozmiarów oraz innych własności danych zapisywanych przez użytkowników, czyli **danych użytecznych**. Rozmiary oraz organizacja i sposób zapisu metadanych zależą od rodzaju systemu plikowego.

Każdy rodzaj systemu plikowego wymaga innych metod odczytu i zapisu danych oraz odczytu i odświeżania metadanych.

Podział systemów plikowych

Różne systemy plikowe dla różnych technologii składowania

- Twarde dyski rotacyjne – np. ext2/ext3/ext4, NTFS, FAT.
- Twarde dyski rotacyjne i **SSD** (*ang. Solid-State Disk*) – np. btrfs.
- Pamięci *flash* – np. JFFS2.
- Dyski optyczne – np. ISO9660, UDF.

Podział systemów plikowych

Różne systemy plikowe dla różnych technologii składowania

- Twarde dyski rotacyjne – np. ext2/ext3/ext4, NTFS, FAT.
- Twarde dyski rotacyjne i **SSD** (*ang. Solid-State Disk*) – np. btrfs.
- Pamięci *flash* – np. JFFS2.
- Dyski optyczne – np. ISO9660, UDF.

Różne systemy plikowe dla różnych zastosowań

- Macierze dyskowe – np. XFS, JFS.
- Klastry (*ang. cluster*) – np. GFS2, OCFS2.
- Sieciowe systemy plikowe – np. NFS, CIFS.

Bufory dyskowe

Operacje zapisu i odczytu dla urządzeń blokowych są czasochłonne

- Fizyczne manipulowanie nośnikami danych wymaga czasu.
- Operacje dotyczą całych bloków danych.
- Dane są przesyłane przez systemy połączeń o różnej przepustowości.

Bufory dyskowe

Operacje zapisu i odczytu dla urządzeń blokowych są czasochłonne

- Fizyczne manipulowanie nośnikami danych wymaga czasu.
- Operacje dotyczą całych bloków danych.
- Dane są przesyłane przez systemy połączeń o różnej przepustowości.

Czas dostępu (*ang. access time*)

Średni czas, po jakim dane trafiają do pamięci RAM (lub na nośnik przy zapisie), licząc od momentu zlecenia przeprowadzenia operacji.

Bufory dyskowe

Operacje zapisu i odczytu dla urządzeń blokowych są czasochłonne

- Fizyczne manipulowanie nośnikami danych wymaga czasu.
- Operacje dotyczą całych bloków danych.
- Dane są przesyłane przez systemy połączeń o różnej przepustowości.

Czas dostępu (*ang. access time*)

Średni czas, po jakim dane trafiają do pamięci RAM (lub na nośnik przy zapisie), licząc od momentu zlecenia przeprowadzenia operacji.

Bufor dyskowy (*ang. disk buffer*) lub schowek dyskowy (*ang. disk cache*)

Obszar pamięci RAM wykorzystywany do tymczasowego przechowywania danych pochodzących z systemu plikowego w celu skrócenia czasu dostępu do nich.

Zapisywanie zmian z opóźnieniem

Zapis zwrotny (*ang. write back*)

Technika polegająca na przechowywaniu **zmodyfikowanych** danych pochodzących z systemu plikowego **w buforze dyskowym** tak długo, jak to jest możliwe i zapisywaniu ich na nośniku wtedy, kiedy jest to konieczne lub kiedy można na to poświęcić czas.

Zapisywanie zmian z opóźnieniem

Zapis zwrotny (*ang. write back*)

Technika polegająca na przechowywaniu **zmodyfikowanych** danych pochodzących z systemu plikowego **w buforze dyskowym** tak długo, jak to jest możliwe i zapisywaniu ich na nośniku wtedy, kiedy jest to konieczne lub kiedy można na to poświęcić czas.

- 1 W pamięci odwzorowuje się strukturę logiczną danych użytecznych i metadanych.
- 2 Przy modyfikacji zmieniają się tylko **kopie przechowywane w pamięci**.
- 3 Dane i metadane z pamięci są przenoszone na nośnik w blokach, kiedy jest to wygodne (**zapis zwrotny**).
- 4 Występuje opóźnienie między modyfikacją i pojawieniem się zmian na nośniku danych.

Problem spójności metadanych

Przy nagłym wyłączeniu komputera tracimy zawartość buforów dyskowych.

Problem spójności metadanych

Przy nagłym wyłączeniu komputera tracimy zawartość buforów dyskowych.

Możliwe jest naruszenie spójności metadanych podczas przeprowadzania złożonych operacji, np. podczas usuwania pliku:

- 1 Usuwanie plików w dwóch krokach:
 - Najpierw trzeba usunąć nazwę z katalogu.
 - Później oznacza się bloki zajmowane przez plik jako „wolne”.
- 2 Jeśli między pierwszym i drugim krokiem wystąpi awaria (np. zasilania), bloki zajmowane przez plik nie zostaną zwolnione.

Naprawa takich problemów zwykle wymaga przejrzania **wszystkich metadanych** w systemie plikowym i doprowadzenia ich do spójnego stanu (jest to **bardzo** czasochłonne).

Systemy plikowe z żurnalem (*ang. journaling file systems*)

Żurnal (*ang. journal*)

Specjalny obszar dysku, w którym z **wyprzedzeniem** zapisuje się zmiany do wprowadzenia.

Systemy plikowe z żurnalem (*ang. journaling file systems*)

Żurnal (*ang. journal*)

Specjalny obszar dysku, w którym z **wyprzedzeniem** zapisuje się zmiany do wprowadzenia.

- W przypadku awarii żurnal jest odczytywany i zapisane w nim zmiany są wprowadzane we „właściwym” systemie plikowym.
- Zmiany z żurnala muszą być albo wprowadzone w całości, albo wcale nie wprowadzone (aby „właściwe” metadane były zawsze spójne).

Systemy plikowe z żurnalem (*ang. journaling file systems*)

Żurnal (*ang. journal*)

Specjalny obszar dysku, w którym z **wyprzedzeniem** zapisuje się zmiany do wprowadzenia.

- W przypadku awarii żurnal jest odczytywany i zapisane w nim zmiany są wprowadzane we „właściwym” systemie plikowym.
- Zmiany z żurnala muszą być albo wprowadzone w całości, albo wcale nie wprowadzone (aby „właściwe” metadane były zawsze spójne).

Nadal w użyciu są systemy plikowe **bez** żurnala

- FAT (**Uwaga:** używany z pendrive'ami, kartami *Compact Flash* itp.!).
- ext2
- Sieciowe systemy plikowe.

Montowanie i odmontowywanie systemu plikowego

Montowanie (*ang. mount*) systemu plikowego

- 1 Jądro systemu przygotowuje się do operowania danym systemem plikowym.
 - Ładowanie sterownika do pamięci (jeśli trzeba).
 - Odczyt metadanych i tworzenie buforów dyskowych.
 - Wprowadzanie zmian z żurnala (jeśli trzeba).
- 2 Wskazany katalog w drzewie katalogowym staje się **punktem zaczepienia** (*ang. mount point*) nowego systemu plikowego.

Montowanie i odmontowywanie systemu plikowego

Montowanie (*ang. mount*) systemu plikowego

- 1 Jądro systemu przygotowuje się do operowania danym systemem plikowym.
 - Ładowanie sterownika do pamięci (jeśli trzeba).
 - Odczyt metadanych i tworzenie buforów dyskowych.
 - Wprowadzanie zmian z żurnala (jeśli trzeba).
- 2 Wskazany katalog w drzewie katalogowym staje się **punktem zaczepienia** (*ang. mount point*) nowego systemu plikowego.

Odmontowywanie (*ang. unmount*) systemu plikowego

Odłączanie systemu plikowego od drzewa katalogowego:

- 1 Zapis zwrotny buforów dyskowych i wprowadzanie zmian z żurnala.
- 2 Punkt zaczepienia systemu plikowego z powrotem staje się „zwykłym” katalogiem.

Pliki

Plik (*ang. file*)

Zbiór danych użytecznych w systemie plikowym.

Pliki

Plik (*ang. file*)

Zbiór danych użytecznych w systemie plikowym.

Własności pliku

- 1 Rozmiary mogą zmieniać się w czasie.
 - Liczba zajmowanych bloków może się zmieniać (musi być co najmniej jeden).
- 2 Z punktu widzenia użytkownika stanowi **ciąg bajtów**.
 - Bajtom wchodzącym w skład pliku można nadać kolejne numery.
 - Pierwszy bajt w pliku ma numer 0 (konwencja).
- 3 Nie musi zajmować ciągu sektorów o kolejnych numerach.
 - „Między” sektorami zajmowanymi przez plik mogą znajdować się sektory zajmowane przez inne pliki lub sektory niewykorzystane.
- 4 Jest identyfikowany przez **ścieżkę** (*ang. path*).

Katalogi

Katalog (*ang. directory*), folder

Lista nazw plików, katalogów i innych obiektów z informacjami umożliwiającymi znalezienie zajmowanych przez nie sektorów.

Katalogi

Katalog (*ang. directory*), folder

Lista nazw plików, katalogów i innych obiektów z informacjami umożliwiającymi znalezienie zajmowanych przez nie sektorów.

- 1 Katalogi (podobnie, jak pliki) są identyfikowane z pomocą ścieżek.
- 2 Aby ustalić które sektory zajmuje plik (lub katalog), trzeba znaleźć katalog z jego nazwą.
- 3 Ścieżka jest zbiorem nazw katalogów, które trzeba kolejno odczytywać, by w końcu dotrzeć do informacji o położeniu pliku.
 - Informacje o położeniu następnego katalogu na ścieżce znajdują się w poprzednim katalogu.
 - Informacje o położeniu pierwszego katalogu na ścieżce znajdują się w **katalogu głównym** (*ang. root directory*), którego nazwą jest /.

Własności katalogów i plików

Podkatalog (*ang. subdirectory*)

Katalog, którego nazwa jest zapisana w innym katalogu, zwanym **katalogiem nadrzędnym** (*ang. parent directory*) w stosunku do niego.

Własności katalogów i plików

Podkatalog (*ang. subdirectory*)

Katalog, którego nazwa jest zapisana w innym katalogu, zwanym **katalogiem nadrzędnym** (*ang. parent directory*) w stosunku do niego.

Katalogi **nie zawierają** informacji o własnościach plików, takich jak prawa dostępu, czas ostatniej modyfikacji itp.

- 1 Własności plików, poza nazwą, są przechowywane poza katalogami.
- 2 Własności te zależą od rodzaju systemu plikowego.
- 3 Systemy plikowe mogą nakładać ograniczenia na nazwy plików i ich rozmiary.
 - Nie każdy ciąg znaków może być poprawną nazwą pliku lub katalogu w każdym systemie plikowym (może to zależeć od zbioru znaków).
 - Zwykle systemy plikowe nakładają górne ograniczenie na rozmiary pojedynczego pliku.

Przeność danych

Dane zapisane w plikach powinny być możliwe do odczytania w różnych systemach operacyjnych i z użyciem różnych programów.

- 1 Ułatwia to komunikację.
- 2 Pozwala użytkownikom komputerów korzystać z oprogramowania pochodzącego z różnych źródeł (np. od różnych dostawców).
- 3 Teoretycznie gwarantuje możliwość odtworzenia informacji zawartych w plikach w dowolnie odległej przyszłości.
 - Programy użyte do stworzenia tych plików mogą być wówczas niedostępne.
- 4 Stymuluje rozwój oprogramowania (do pewnego stopnia).
 - Tworzone są nowe programy służące do tych samych celów (np. oprogramowanie biurowe, przeglądarki WWW) i każdy z nich może być użyty do odczytu i modyfikacji tego samego pliku z danymi.

Pliki tekstowe

Plik tekstowy (*ang. text file*)

Zawiera dane zakodowane w postaci kodów znaków, które powinny być interpretowane zgodnie z ustalonym zbiorem znaków.

- 1 Tzw. **znaki sterujące** (*ang. control characters*) odpowiadają specjalnym kodom określającym sposób **formatowania wydruku** (np. znaki ASCII o kodach 0 – 31).
- 2 Pliki tekstowe nie zawierają informacji o tym jaki zbiór znaków jest odpowiedni do interpretacji ich zawartości (program drukujący tekst lub użytkownik powinien to określić).

Pliki tekstowe mają ograniczone zastosowania

- 1 Mogą zawierać tylko tekst (brak grafiki, tabel, odnośników itp.).
- 2 Nie zawierają informacji o kroju i wielkości czcionki, kolorach itp.

Pliki metatekstowe

Plik metatekstowe

Zawiera dane w postaci tekstu (tzn. kodów znaków jak dla „zwykłego” pliku tekstowego), z których część może być interpretowana jako **znaczniki** (*ang. tag*) określające m. in.

- Oczekiwany wygląd dokumentu (np. krój i wielkość czcionki itp.).
- Jego powiązania z innymi dokumentami (np. odnośniki).
- Zbiór znaków, zgodnie z którym mają być interpretowane kody znaków w dokumencie.

Pliki metatekstowe

Plik metatekstowe

Zawiera dane w postaci tekstu (tzn. kodów znaków jak dla „zwykłego” pliku tekstowego), z których część może być interpretowana jako **znaczniki** (*ang. tag*) określające m. in.

- Oczekiwany wygląd dokumentu (np. krój i wielkość czcionki itp.).
- Jego powiązania z innymi dokumentami (np. odnośniki).
- Zbiór znaków, zgodnie z którym mają być interpretowane kody znaków w dokumencie.

Reguły tworzenia i interpretacji znaczników określają metajęzyki

- HTML (*ang. HyperText Markup Language*)
- XML (*ang. Extensible Markup Language*)
- SGML (*ang. Standard Generalized Markup Language*)

Grafika rastrowa i wektorowa

Grafika rastrowa (*ang. raster graphics*)

Obraz jest reprezentowany przez (ogólnie) prostokątną siatkę punktów o różnych kolorach, czyli tzw. **pikseli** (*ang. pixel, picture element*). Oczka tej siatki **nie muszą** mieć jednakowej szerokości i wysokości.

Grafika rastrowa i wektorowa

Grafika rastrowa (*ang. raster graphics*)

Obraz jest reprezentowany przez (ogólnie) prostokątną siatkę punktów o różnych kolorach, czyli tzw. **pikseli** (*ang. pixel, picture element*). Oczka tej siatki **nie muszą** mieć jednakowej szerokości i wysokości.

Mapa bitowa, bitmapa (*ang. bitmap*)

Struktura danych reprezentująca obraz w postaci gotowej do wydrukowania (lub wyświetlenia). Kolor każdego piksela reprezentuje określona liczba bitów danych.

Grafika rastrowa i wektorowa

Grafika rastrowa (*ang. raster graphics*)

Obraz jest reprezentowany przez (ogólnie) prostokątną siatkę punktów o różnych kolorach, czyli tzw. **pikseli** (*ang. pixel, picture element*). Oczka tej siatki **nie muszą** mieć jednakowej szerokości i wysokości.

Mapa bitowa, bitmapa (*ang. bitmap*)

Struktura danych reprezentująca obraz w postaci gotowej do wydrukowania (lub wyświetlenia). Kolor każdego piksela reprezentuje określona liczba bitów danych.

Grafika wektorowa (*ang. vector graphics*)

Obraz jest reprezentowany z pomocą **instrukcji**, które mają być wykonane w celu utworzenia go (np. „narysuj linię prostą”, „narysuj koło” itp.). Poszczególne punkty identyfikuje się podając ich **współrzędne** (wektory).

Kolory w grafice rastrowej

RGB (*ang. red, green, blue*)

Kolor piksela określa się podając natężenie każdej z trzech podstawowych barw światła (czerwony, zielony, niebieski). Zwykle jeden piksel jest reprezentowany przez 3 lub 4 bajty danych. Reprezentacja odpowiednia dla monitorów i wyświetlaczy LCD, OLED itp.

Kolory w grafice rastrowej

RGB (*ang. red, green, blue*)

Kolor piksela określa się podając natężenie każdej z trzech podstawowych barw światła (czerwony, zielony, niebieski). Zwykle jeden piksel jest reprezentowany przez 3 lub 4 bajty danych. Reprezentacja odpowiednia dla monitorów i wyświetlaczy LCD, OLED itp.

CMYK (*ang. cyan, magenta, yellow, black*)

Kolor piksela określa się podając natężenie każdego z czterech kolorów „farb” (turkusowy, purpura, żółty, czarny). Reprezentacja bardziej odpowiednia do drukowania na papierze.

Formaty plików dla grafiki rastrowej

- BMP (*ang. bitmap*)
 - Istnieją co najmniej 2 formaty plików graficznych o takiej nazwie.
- PBM (*ang. Portable Bitmap*)
 - Używany w systemach wykorzystujących środowisko X Windows.
- TIFF (*ang. Tagged Image File Format*), ISO 12639
- GIF (*ang. Graphics Interchange Format*), kompresja LZW
 - 8 bitów na piksel, 24-bitowa przestrzeń barw (palety referencyjne).
 - Od 1994 w „szarej strefie” z powodu patentu na algorytm kompresji (Unisys).
- PNG (*ang. Portable Network Graphics*), ISO/IEC 15948:2004, kompresja bezstratna
- JPEG (*ang. Joint Photographic Experts Group*), ISO/IEC 10918, kompresja stratna

Grafika wektorowa i dokumenty „drukowane”

Nie tylko grafika, także dokumenty tekstowe itp. (praktycznie wszystkie rodzaje dokumentów).

Grafika wektorowa i dokumenty „drukowane”

Nie tylko grafika, także dokumenty tekstowe itp. (praktycznie wszystkie rodzaje dokumentów).

- PostScript (PS) – język programowania wykorzystywany jako język opisu stron (*ang. page description language*) dla ploterów, drukarek itp. (1982).
- EPS (*ang. Encapsulated PostScript*) – kod w języku PostScript z dołączonym „podglądem” w niskiej rozdzielczości.
- PDF (*ang. Portable Document Format*), ISO/IEC 32000-1:2008 – opis strony z „dodatkami”, jak odnośniki (*ang. link*) itp., („przetworzony” PostScript).
- SVG (*ang. Scalable Vector Graphics*)
- DjVu (bardzo dobra kompresja zeskanowanych dokumentów).

Dokumenty z możliwością wprowadzania zmian

Zawartość pliku reprezentuje dokument

- Sformatowany tekst.
- Ewentualnie grafika, tabele itp.
- Zawiera informacje o sposobie wyświetlania lub drukowania.
- Może być skompresowany.

Dokumenty z możliwością wprowadzania zmian

Zawartość pliku reprezentuje dokument

- Sformatowany tekst.
- Ewentualnie grafika, tabele itp.
- Zawiera informacje o sposobie wyświetlania lub drukowania.
- Może być skompresowany.

W przeszłości istniało wiele konkurencyjnych zamkniętych formatów

- 1 Brak przenośności dokumentów dokuczliwy dla użytkowników.
- 2 Niezgodności między różnymi wersjami tego samego programu (!).
- 3 Nieformalnym standardem stały się formaty z MS Office 97:
 - DOC – dokumenty tekstowe.
 - XLS – arkusze kalkulacyjne.
 - PPT – prezentacje.

Open Document Format

ISO/IEC 26300:2006

Open Document Format for Office Applications (OpenDocument) v1.0.

Open Document Format

ISO/IEC 26300:2006

Open Document Format for Office Applications (OpenDocument) v1.0.

Zestaw otwartych formatów plików, opartych na XML (z kompresją), dla różnych rodzajów dokumentów:

- ODT – dokumenty tekstowe.
- ODS – arkusze kalkulacyjne.
- ODP – prezentacje.
- ODG – rysunki (grafika wektorowa).
- ODF – wzory matematyczne.

Open Document Format

ISO/IEC 26300:2006

Open Document Format for Office Applications (OpenDocument) v1.0.

Zestaw otwartych formatów plików, opartych na XML (z kompresją), dla różnych rodzajów dokumentów:

- ODT – dokumenty tekstowe.
- ODS – arkusze kalkulacyjne.
- ODP – prezentacje.
- ODG – rysunki (grafika wektorowa).
- ODF – wzory matematyczne.

Większość dostępnych na rynku pakietów biurowych wspiera ODF.

- Dostępna jest „wtyczka” dla Microsoft Office.

Zapis w oparciu o „nuty”

MIDI (*ang. Musical Instrument Digital Interface*), 1982

Zapis dźwięku poprzez określenie wysokości, natężenia, czasu trwania i rodzaju instrumentu oraz efektów takich jak wibrato, pogłos itp. (określony został także protokół komunikacyjny dla urządzeń odtwarzających dźwięk).

Zapis w oparciu o „nuty”

MIDI (*ang. Musical Instrument Digital Interface*), 1982

Zapis dźwięku poprzez określenie wysokości, natężenia, czasu trwania i rodzaju instrumentu oraz efektów takich jak wibrato, pogłos itp. (określony został także protokół komunikacyjny dla urządzeń odtwarzających dźwięk).

Sterowanie urządzeniami (takimi jak syntezatory i maszyny perkusyjne) poprzez przesyłanie do nich **komunikatów** (*ang. message*) określających dźwięki do „zagrania”.

Zapis w oparciu o „nuty”

MIDI (*ang. Musical Instrument Digital Interface*), 1982

Zapis dźwięku poprzez określenie wysokości, natężenia, czasu trwania i rodzaju instrumentu oraz efektów takich jak wibrato, pogłos itp. (określony został także protokół komunikacyjny dla urządzeń odtwarzających dźwięk).

Sterowanie urządzeniami (takimi jak syntezatory i maszyny perkusyjne) poprzez przesyłanie do nich **komunikatów** (*ang. message*) określających dźwięki do „zagrania”.

Każde urządzenie dysponuje zestawem brzmień odpowiadających różnym instrumentom do wyboru. Mogą one być **syntetyczne** (np. tworzone poprzez złożenie wielu różnych fal dźwiękowych z generatora) lub oparte o **próbki** (*ang. sample*), czyli zarejestrowane brzmienia „prawdziwych” instrumentów.

Pliki MIDI

`.mid`, `.smf` (*ang. standard MIDI file*)

Pliki z zapisem dźwięku zakodowanym zgodnie ze standardem MIDI (zawartość takiego pliku stanowi zapis ciągu komunikatów, które należy przesłać do syntezy w celu odtworzenia „nagrania”).

Pliki MIDI

.mid, .smf (ang. standard MIDI file)

Pliki z zapisem dźwięku zakodowanym zgodnie ze standardem MIDI (zawartość takiego pliku stanowi zapis ciągu komunikatów, które należy przesłać do syntezy w celu odtworzenia „nagrania”).

Sposób odtworzenia zapisu odczytanego z pliku zależy od syntezy:

- 1 Wysokość, natężenie i czas trwania każdej „nuty” powinny być jednakowe dla wszystkich urządzeń zgodnych z MIDI.
- 2 Brzmienie (barwa itd.) jest ogólnie zależne od danego urządzenia.

Pliki MIDI

.mid, .smf (ang. standard MIDI file)

Pliki z zapisem dźwięku zakodowanym zgodnie ze standardem MIDI (zawartość takiego pliku stanowi zapis ciągu komunikatów, które należy przesłać do syntezy w celu odtworzenia „nagrania”).

Sposób odtworzenia zapisu odczytanego z pliku zależy od syntezy:

- 1 Wysokość, natężenie i czas trwania każdej „nuty” powinny być jednakowe dla wszystkich urządzeń zgodnych z MIDI.
- 2 Brzmienie (barwa itd.) jest ogólnie zależne od danego urządzenia.

Dźwięk wytwarzany przez urządzenia MIDI zwykle znacząco różni się od brzmienia „analogowych” instrumentów muzycznych.

Próbkowanie i zapis dźwięku

Próbkowanie (*ang. sampling*)

Technika polegająca na rejestracji natężenia sygnału (np. dźwięku) w ustalonej skali i w ustalonych odstępach czasu.

Próbkowanie i zapis dźwięku

Próbkowanie (*ang. sampling*)

Technika polegająca na rejestracji natężenia sygnału (np. dźwięku) w ustalonej skali i w ustalonych odstępach czasu.

Sygnał zarejestrowany w ten sposób można później odtworzyć poprzez wytwarzanie odpowiedniego natężenia (np. powodując odpowiednie wychylenie membrany głośnika) w takich samych odstępach czasu, w jakich rejestrowane były próbki.

Próbkowanie i zapis dźwięku

Próbkowanie (*ang. sampling*)

Technika polegająca na rejestracji natężenia sygnału (np. dźwięku) w ustalonej skali i w ustalonych odstępach czasu.

Sygnał zarejestrowany w ten sposób można później odtworzyć poprzez wytwarzanie odpowiedniego natężenia (np. powodując odpowiednie wychylenie membrany głośnika) w takich samych odstępach czasu, w jakich rejestrowane były próbki.

Twierdzenie Nyquista-Shannona (o próbkowaniu)

Sygnał ciągły może być ponownie wiernie odtworzony z sygnału dyskretnego, jeśli był próbkowany z częstotliwością co najmniej dwa razy większą od granicznej częstotliwości swego widma (tzw. **warunek Nyquista**).

Cyfrowy zapis dźwięku

Obserwacja

Ponieważ ludzie słyszą dźwięki o częstotliwościach **co najwyżej 20 kHz**, więc na ich potrzeby wystarczy próbkować dźwięk z częstotliwością 40 kHz (na jeden kanał stereo), wówczas bowiem (zgodnie z twierdzeniem Nyquista-Shannona) po jego odtworzeniu **nie będą w stanie** odróżnić go od oryginału.

Cyfrowy zapis dźwięku

Obserwacja

Ponieważ ludzie słyszą dźwięki o częstotliwościach **co najwyżej** 20 kHz, więc na ich potrzeby wystarczy próbkować dźwięk z częstotliwością 40 kHz (na jeden kanał stereo), wówczas bowiem (zgodnie z twierdzeniem Nyquista-Shannona) po jego odtworzeniu **nie będą w stanie** odróżnić go od oryginału.

Zapis na płytach CD

Wykorzystuje powyższą obserwację oraz stosuje próbki o 16-bitowej rozdzielczości (tzn. rozróżnia się 65536 poziomów natężenia dźwięku i każda próbka zajmuje 2 bajty danych) na każdy kanał stereo.

Formaty WAV i MP3

WAV (*ang. wave form audio format*)

Format zapisu plików audio, w którym wykorzystuje się próbki takie, jak dla płyt CD (Microsoft, IBM). W nieskompresowanej postaci zajmuje około 1,5 Mb/s dla jakości CD.

Formaty WAV i MP3

WAV (*ang. wave form audio format*)

Format zapisu plików audio, w którym wykorzystuje się próbki takie, jak dla płyt CD (Microsoft, IBM). W nieskompresowanej postaci zajmuje około 1,5 Mb/s dla jakości CD.

Znaczące zmniejszenie rozmiarów plików „dźwiękowych” można uzyskać praktycznie tylko przy zastosowaniu **stratnej kompresji**.

Formaty WAV i MP3

WAV (*ang. wave form audio format*)

Format zapisu plików audio, w którym wykorzystuje się próbki takie, jak dla płyt CD (Microsoft, IBM). W nieskompresowanej postaci zajmuje około 1,5 Mb/s dla jakości CD.

Znaczące zmniejszenie rozmiarów plików „dźwiękowych” można uzyskać praktycznie tylko przy zastosowaniu **stratnej kompresji**.

MP3 (*ang. MPEG-1/2 Audio Layer-3*)

Format zapisu dźwięku poddanego stratnej kompresji (Fraunhofer-Institut für Integrierte Schaltungen, 1991) z wykorzystaniem modelu psychoakustycznego (dynamicznie zmienia się rozdzielczość próbek lub częstotliwość próbkowania tak, aby nie można było zauważyć różnicy).