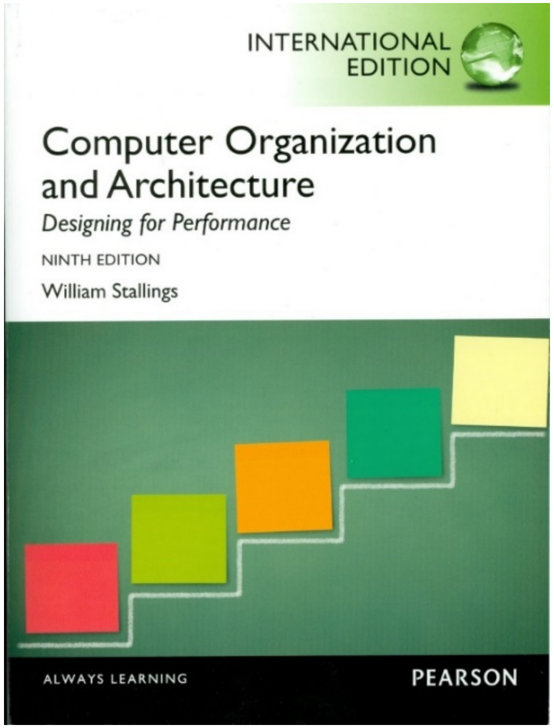
Organizacja i architektura komputerów
(*Computer organization and architecture*)

Uwagi organizacyjne:

- wykład: 30 godzin,
- ćwiczenia: 30 godzin (zaliczenie na ocenę).
- egzamin pisemny: 03.02.2023 r. (piątek) o godzinie 12:00.

1. Pojęcia **organizacji** i **architektury** komputerów. Schemat blokowy komputera.
   - Co to jest **organizacja komputera** (*computer organization*) ?
   - Co to jest **architektura komputera** (*computer architecture*) ?

W. Stallings, Computer Organization and Architecture, Ninth edition, Pearson, 2013.



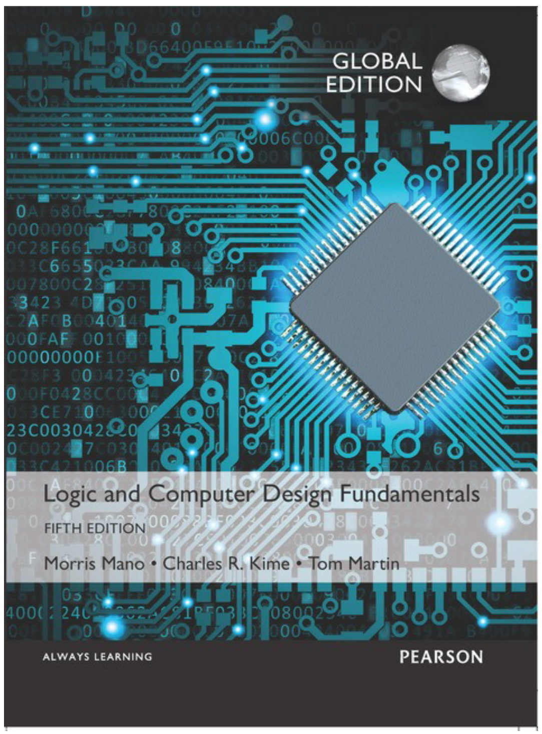
W. Stallings, Organizacja i architektura systemu komputerowego, Wydawnictwa Naukowo-Techniczne, 2000, (dane o oryginale: W. Stallings, Computer Organization and Architecture, Fourth edition, Prentice Hall, 1996):

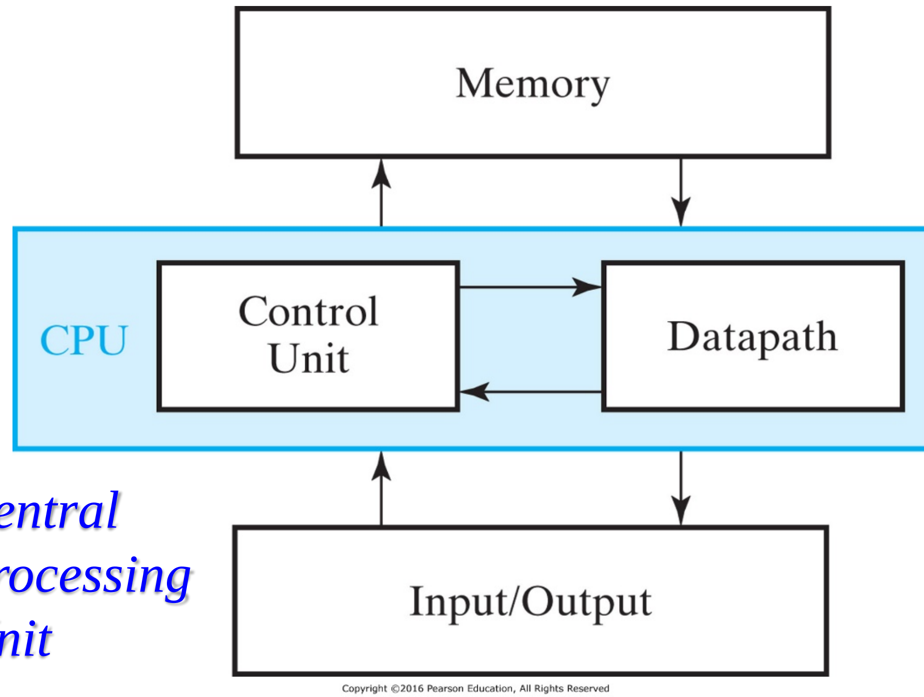
- „Przy opisywaniu systemów komputerowych często czynione jest rozróżnienie między architekturą, a jego organizacją. Chociaż precyzyjne zdefiniowane tych pojęć jest trudne, jednak istnieje zgoda co do zagadnień, których dotyczą.”
- „Na przykład to, czy w komputerze występuje rozkaz mnożenia, jest zagadnieniem projektowania architektury. Zagadnieniem organizacyjnym jest natomiast to, czy ten rozkaz będzie wykonywany przez specjalną jednostkę mnożącą, czy też przez wielokrotne wykorzystanie jednostki sumującej systemu.”
- „Historycznie, a także współcześnie, rozróżnienie między architekturą, a organizacją jest ważne.”
- „Ponieważ jednak organizacja komputera musi być zaprojektowana w celu wdrożenia określonej architektury, dokładne przeanalizowanie organizacji wymaga również szczegółowego zbadania architektury.”

~~1. Pojęcia **organizacji** i **architektury** komputerów.~~ Schemat blokowy komputera.
   - ~~Co to jest **organizacja komputera** (*computer organization*) ?~~
   - ~~Co to jest **architektura komputera** (*computer architecture*) ?~~

Co to jest schemat blokowy komputera ?

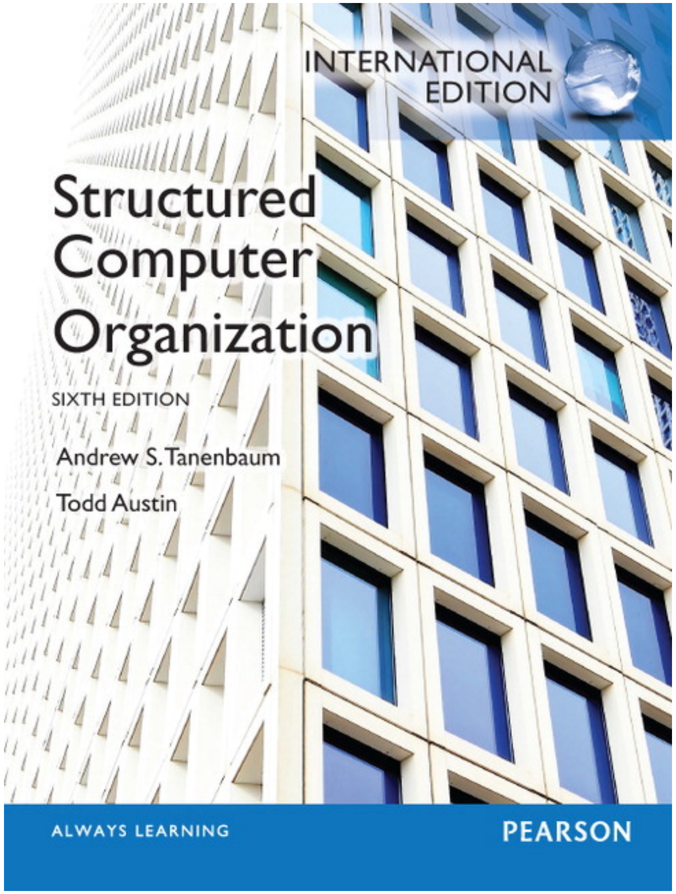
M. Morris Mano, Charles R. Kime, Tom Martin,
Logic and Computer Design Fundamentals,
Fifth edition, Pearson, 2016.





***C**entral*
***P**rocessing*
***U**nit*

A. Tanenbaum, Structured Computer Organization,
Sixth edition, Pearson, 2013.



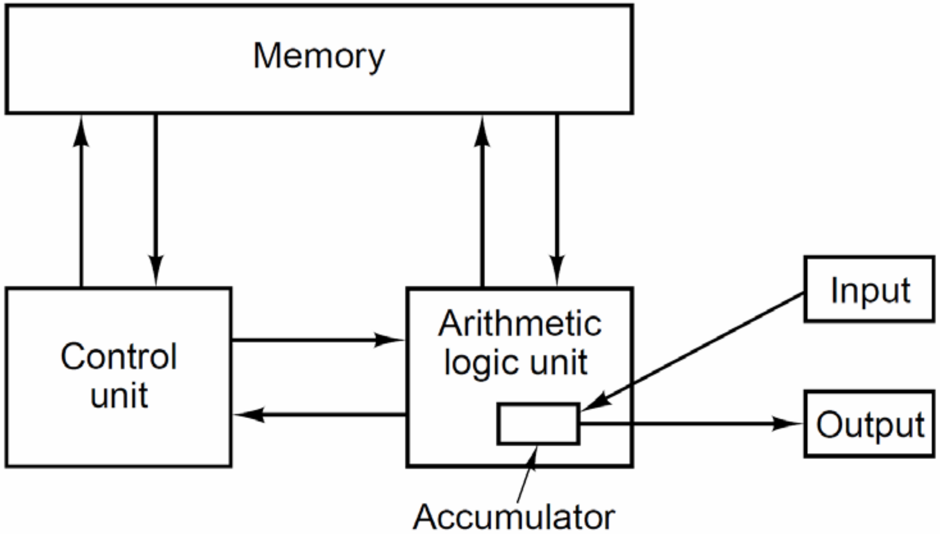
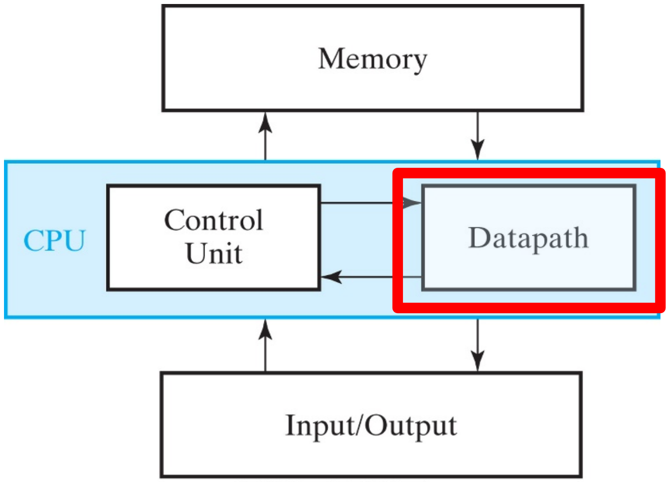


Figure 1-5. The original von Neumann machine.

M. Morris Mano, Charles R. Kime, Tom Martin, Logic and Computer Design Fundamentals, Fifth edition, Pearson, 2016.



Data Path

A. Tanenbaum, Structured Computer Organization, Sixth edition, Pearson, 2013.

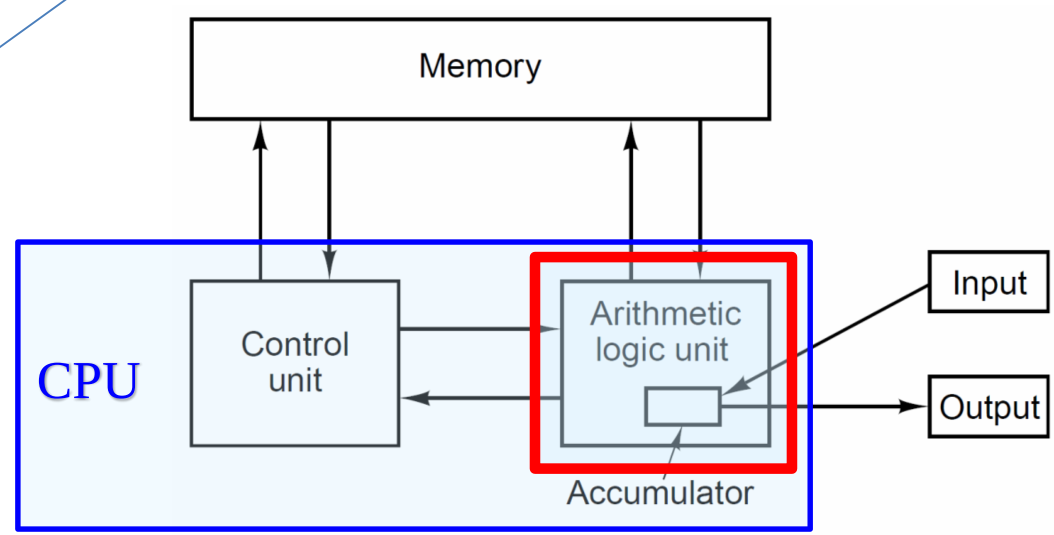


Figure 1-5. The original von Neumann machine.

1. ~~Pojęcia **organizacji** i **architektury** komputerów. Schemat blokowy komputera.~~
2. Systemy liczbowe, konwersja liczb dziesiętnych do innych systemów liczenia, kod znak–moduł, uzupełnienia liczb, kody uzupełnieniowe do 1 i do 2, kod BCD (*Binary Coced Decimal*), kod ASCII (*American Standard Code for Information Interchange*), dodawanie i odejmowanie liczb dwójkowych (binarnych), pojęcie nadmiaru (przepełnienia).

W systemie dziesiętnym ciąg cyfr 123.45 oznacza liczbę o wartości:

$$1\cdot10^2 + 2\cdot10^1 + 3\cdot10^0 + 4\cdot10^{-1}+5\cdot10^{-2} = 100 + 20 + 3 + 0.40 + 0.05 = 123.45$$

$$1\cdot 10^{2} + 2\cdot 10^{1} + 3\cdot 10^{0} + 4\cdot 10^{-1} + 5\cdot 10^{-2} = 100 + 20 + 3 + 0.40 + 0.05 = 123.45$$

Liczba dziesiętna: 1 2 3.4 5

pozycja -2 (10^{-2})

pozycja -1 (10^{-1})

pozycja 0 (10^{0})

pozycja 1 (10^{1})

pozycja 2 (10^{2})

Oznaczając podstawę systemu liczbowego przez p , a cyfry przez a_i , można każdą liczbę N zawierającą n-cyfrową część całkowitą i m-cyfrową część ułamkową, przedstawić w postaci szeregu:

$$N = a_{n-1} \cdot p^{n-1} + a_{n-2} \cdot p^{n-2} + \ldots + a_0 \cdot p^0 + a_{-1} \cdot p^{-1} + a_{-2} \cdot p^{-2} + \ldots + a_{-m} \cdot p^{-m} =$$

$$= \sum_{i=-m}^{n-1} a_i p^i$$

lub w następującej prostszej postaci:

$$N = a_{n-1}\, a_{n-2} \ldots a_0 \,.\, a_{-1}\, a_{-2} \ldots a_{-m}$$

W systemie dziesiętnym podstawa jest równa 10. W systemie dziesiętnym cyframi są: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

W systemie dwójkowym (binarnym) podstawa jest równa 2. W systemie dwójkowym cyframi są: 0 i 1.

W systemie dwójkowym (binarnym) ciąg cyfr: 1001.01 oznacza liczbę dziesiętną o wartości:

$1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = 8 + 0 + 0 + 1 + 0 + 0.25 = 9.25$

Liczba dwójkowa (binarna): 1 0 0 1.0 1

pozycja -2 (2^{-2})

pozycja -1 (2^{-1})

pozycja 0 (2^0)

pozycja 1 (2^1)

pozycja 2 (2^2)

pozycja 3 (2^3)

W celu odróżnienia liczb o odmiennych podstawach stosuje się notacje polegającą na ujęciu zapisu liczby w nawiasy okrągłe ( ) oraz podaniu podstawy systemu liczbowego jako indeksu. Na przykład: $(1001.01)_2 = (9.25)_{10}$

Cyframi w systemie ósemkowym są: 0, 1, 2, 3, 4, 5, 6, 7.

W systemie ósemkowym ciąg cyfr 123.45 oznacza liczbę dziesiętną o wartości:

$$1 \cdot 8^2 + 2 \cdot 8^1 + 3 \cdot 8^0 + 4 \cdot 8^{-1} + 5 \cdot 8^{-2} = 64 + 16 + 3 + 0.5 + 0.078125 = 83.578125$$

Cyframi w systemie szesnastkowym są:
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

W systemie szesnastkowym ciąg cyfr F5A.C8 oznacza liczbę dziesiętną o wartości:

$$15 \cdot 16^2 + 5 \cdot 16^1 + 10 \cdot 16^0 + 12 \cdot 16^{-1} + 8 \cdot 16^{-2} = 3840 + 80 + 10 + 0.75 + 0.03125 =$$
$$= 3930.78125$$

$1 \cdot 8^2 + 2 \cdot 8^1 + 3 \cdot 8^0 + 4 \cdot 8^{-1} + 5 \cdot 8^{-2} = 64 + 16 + 3 + 0.5 + 0.078125 = 83.578125$

Liczba ósemkowa: 1 2 3.4 5

pozycja -2 (8^{-2})

pozycja -1 (8^{-1})

pozycja 0 (8^0)

pozycja 1 (8^1)

pozycja 2 (8^2)

Zamiana liczby dziesiętnej na liczbę dwójkową (binarną).

Zamiana liczby dziesiętnej 25.90625 na liczbę dwójkową (binarną):

Część całkowita: 25

Dlaczego „od dołu do góry” ?

25/2 = 12 reszta = 1
12/2 = 6 reszta = 0
6/2 = 3 reszta = 0
3/2 = 1 reszta = 1
1/2 = 0 reszta = 1

11001

Dlaczego nie „od góry do dołu” ?

$(25)_{10} = (11001)_2$

Część ułamkowa: 0.90625

Dlaczego „od góry do dołu” ?

$0.90625 \cdot 2 = 1.8125$ część całkowita = 1
$0.81250 \cdot 2 = 1.6250$ część całkowita = 1
$0.62500 \cdot 2 = 1.2500$ część całkowita = 1
$0.25000 \cdot 2 = 0.5000$ część całkowita = 0
$0.50000 \cdot 2 = 1.0000$ część całkowita = 1

11101

Dlaczego nie „od dołu do góry” ?

$(0.90625)_{10} = (0.11101)_2$

$(25.90625)_{10} = (11001.11101)_2$

Zamiana liczby dziesiętnej całkowitej N na liczbę o podstawie p.

$N = (a_n\, a_{n-1}\, a_{n-2} \ldots a_3\, a_2\, a_1\, a_0)_p$

$N = a_n \cdot p^n + a_{n-1} \cdot p^{n-1} + a_{n-2} \cdot p^{n-2} + \ldots + a_3 \cdot p^3 + a_2 \cdot p^2 + a_1 \cdot p^1 + a_0$

$(N/p) = a_n \cdot p^{n-1} + a_{n-1} \cdot p^{n-2} + a_{n-2} \cdot p^{n-3} + \ldots + a_3 \cdot p^2 + a_2 \cdot p^1 + a_1 = Q_1$, reszta $\boldsymbol{a_0}$

$(Q_1/p) = a_n \cdot p^{n-2} + a_{n-1} \cdot p^{n-3} + a_{n-2} \cdot p^{n-4} + \ldots + a_3 \cdot p^1 + a_2 = Q_2$, reszta $\boldsymbol{a_1}$

$(Q_2/p) = a_n \cdot p^{n-3} + a_{n-1} \cdot p^{n-4} + a_{n-2} \cdot p^{n-5} + \ldots + a_3 = Q_3$, reszta $\boldsymbol{a_2}$

…

$\boldsymbol{a_n}$

Zamiana liczby dziesiętnej całkowitej N na liczbę o podstawie p.

$N = (a_n\, a_{n-1}\, a_{n-2} \ldots a_3\, a_2\, a_1\, a_0)_p$

$N = a_n \cdot p^n + a_{n-1} \cdot p^{n-1} + a_{n-2} \cdot p^{n-2} + \ldots + a_3 \cdot p^3 + a_2 \cdot p^2 + a_1 \cdot p^1 + a_0$

$(N/p) = a_n \cdot p^{n-1} + a_{n-1} \cdot p^{n-2} + a_{n-2} \cdot p^{n-3} + \ldots + a_3 \cdot p^2 + a_2 \cdot p^1 + a_1 = Q_1$, reszta $\boldsymbol{a_0}$

$(Q_1/p) = a_n \cdot p^{n-2} + a_{n-1} \cdot p^{n-3} + a_{n-2} \cdot p^{n-4} + \ldots + a_3 \cdot p^1 + a_2 = Q_2$, reszta $\boldsymbol{a_1}$

$(Q_2/p) = a_n \cdot p^{n-3} + a_{n-1} \cdot p^{n-4} + a_{n-2} \cdot p^{n-5} + \ldots + a_3 = Q_3$, reszta $\boldsymbol{a_2}$

…

$\boldsymbol{a_n}$ **…**

Zamiana liczby dziesiętnej całkowitej N na liczbę o podstawie p.

$N = (a_n\, a_{n-1}\, a_{n-2} \ldots a_3\, a_2\, a_1\, a_0)_p$

Dlaczego „od dołu do góry” ?

$N = a_n \cdot p^n + a_{n-1} \cdot p^{n-1} + a_{n-2} \cdot p^{n-2} + \ldots + a_3 \cdot p^3 + a_2 \cdot p^2 + a_1 \cdot p^1 + a_0$

$(N/p) = a_n \cdot p^{n-1} + a_{n-1} \cdot p^{n-2} + a_{n-2} \cdot p^{n-3} + \ldots + a_3 \cdot p^2 + a_2 \cdot p^1 + a_1 = Q_1$, reszta $\boldsymbol{a_0}$

$(Q_1/p) = a_n \cdot p^{n-2} + a_{n-1} \cdot p^{n-3} + a_{n-2} \cdot p^{n-4} + \ldots + a_3 \cdot p^1 + a_2 = Q_2$, reszta $\boldsymbol{a_1}$

$(Q_2/p) = a_n \cdot p^{n-3} + a_{n-1} \cdot p^{n-4} + a_{n-2} \cdot p^{n-5} + \ldots + a_3 = Q_3$, reszta $\boldsymbol{a_2}$

…

$\boldsymbol{a_n}$ **…**

Zamiana liczby dziesiętnej ułamkowej F na liczbę o podstawie p.

$$R = (a_n\, a_{n-1}\, a_{n-2} \ldots a_1\, a_0\, .\, a_{-1}\, a_{-2}\, a_{-3} \ldots a_{-m})_p$$

$$F = (.\, a_{-1}\, a_{-2}\, a_{-3} \ldots a_{-m})_p$$

$$F = a_{-1} \cdot p^{-1} + a_{-2} \cdot p^{-2} + a_{-3} \cdot p^{-3} + \ldots + a_{-m} \cdot p^{-m}$$

$$F \cdot p = a_{-1} + a_{-2} \cdot p^{-1} + a_{-3} \cdot p^{-2} + \ldots + a_{-m} \cdot p^{-m+1} = a_{-1} + F_1$$

$$F_1 \cdot p = a_{-2} + a_{-3} \cdot p^{-1} + \ldots + a_{-m} \cdot p^{-m+2} = a_{-2} + F_2$$

$$F_2 \cdot p = a_{-3} + \ldots + a_{-m} \cdot p^{-m+3} = a_{-3} + F_3$$

…

a_{-m}

Zamiana liczby dziesiętnej ułamkowej F na liczbę o podstawie p.

$$R = (a_n\, a_{n-1}\, a_{n-2} \ldots a_1\, a_0\, .\, a_{-1}\, a_{-2}\, a_{-3} \ldots a_{-m})_p$$

$$F = (.\, a_{-1}\, a_{-2}\, a_{-3} \ldots a_{-m})_p$$

$$F = a_{-1} \cdot p^{-1} + a_{-2} \cdot p^{-2} + a_{-3} \cdot p^{-3} + \ldots + a_{-m} \cdot p^{-m}$$

$$F \cdot p = a_{-1} + a_{-2} \cdot p^{-1} + a_{-3} \cdot p^{-2} + \ldots + a_{-m} \cdot p^{-m+1} = \boldsymbol{a_{-1}} + F_1$$

$$F_1 \cdot p = a_{-2} + a_{-3} \cdot p^{-1} + \ldots + a_{-m} \cdot p^{-m+2} = \boldsymbol{a_{-2}} + F_2$$

$$F_2 \cdot p = a_{-3} + \ldots + a_{-m} \cdot p^{-m+3} = \boldsymbol{a_{-3}} + F_3$$

…

$\boldsymbol{a_{-m}}$

Zamiana liczby dziesiętnej ułamkowej F na liczbę o podstawie p.

$$R = (a_n\, a_{n-1}\, a_{n-2} \ldots a_1\, a_0 \,.\, a_{-1}\, a_{-2}\, a_{-3} \ldots a_{-m})_p$$

$$F = (.\, a_{-1}\, a_{-2}\, a_{-3} \ldots a_{-m})_p$$

Dlaczego „od góry do dołu” ?

$$F = a_{-1}\cdot p^{-1} + a_{-2}\cdot p^{-2} + a_{-3}\cdot p^{-3} + \ldots + a_{-m}\cdot p^{-m}$$

$$F\cdot p = a_{-1} + a_{-2}\cdot p^{-1} + a_{-3}\cdot p^{-2} + \ldots + a_{-m}\cdot p^{-m+1} = \boldsymbol{a_{-1}} + F_1$$

$$F_1\cdot p = a_{-2} + a_{-3}\cdot p^{-1} + \ldots + a_{-m}\cdot p^{-m+2} = \boldsymbol{a_{-2}} + F_2$$

$$F_2\cdot p = a_{-3} + \ldots + a_{-m}\cdot p^{-m+3} = \boldsymbol{a_{-3}} + F_3$$

…

$\boldsymbol{a_{-m}}$

$0.345 \times 2 = 0.690$ część całkowita = 0

$0.690 \times 2 = 1.380$ część całkowita = 1

$0.380 \times 2 = 0.760$ część całkowita = 0

$0.760 \times 2 = 1.520$ część całkowita = 1

$0.520 \times 2 = 1.040$ część całkowita = 1

$0.040 \times 2 = 0.080$ część całkowita = 0

$0.080 \times 2 = 0.160$ część całkowita = 0

$0.160 \times 2 = 0.320$ część całkowita = 0

…

W tym przypadku proces mnożenia kolejnych części ułamkowych kończymy wtedy, kiedy liczba wyznaczonych cyfr daje wymaganą dokładność, na przykład:

$$(0.345)_{10} = (0.01011000)_2$$

$0.345 \times 2 = 0.690$ część całkowita $= 0$ $\rightarrow$	0
$0.690 \times 2 = 1.380$ część całkowita $= 1$ $\rightarrow$	1
$0.380 \times 2 = 0.760$ część całkowita $= 0$ $\rightarrow$	0
$0.760 \times 2 = 1.520$ część całkowita $= 1$ $\rightarrow$	1
$0.520 \times 2 = 1.040$ część całkowita $= 1$ $\rightarrow$	1
$0.040 \times 2 = 0.080$ część całkowita $= 0$ $\rightarrow$	0
$0.080 \times 2 = 0.160$ część całkowita $= 0$ $\rightarrow$	0
$0.160 \times 2 = 0.320$ część całkowita $= 0$ $\rightarrow$	0
$0.320 \times 2 = 0.640$ część całkowita $= 0$ $\rightarrow$	
$0.640 \times 2 = 0.160$ część całkowita $= 0$ $\rightarrow$	
$0.640 \times 2 = 0.128$ część całkowita $= 0$ $\rightarrow$	

…

$0.345 \times 2 = 0.690$ część całkowita $= 0 \rightarrow 0$

$0.690 \times 2 = 1.380$ część całkowita $= 1 \rightarrow 1$

$0.380 \times 2 = 0.760$ część całkowita $= 0 \rightarrow 0$

$0.760 \times 2 = 1.520$ część całkowita $= 1 \rightarrow 1$

$0.520 \times 2 = 1.040$ część całkowita $= 1 \rightarrow 1$

$0.040 \times 2 = 0.080$ część całkowita $= 0 \rightarrow 0$

$0.080 \times 2 = 0.160$ część całkowita $= 0 \rightarrow 0$

$0.160 \times 2 = 0.320$ część całkowita $= 0 \rightarrow 0$

Zamiany liczby dziesiętnej 1510.90625 na liczbę ósemkową dokonuje się po uprzednim podzieleniu jej na część całkowitą (1510) i część ułamkową (0.90625) oraz w wyniku dokonania konwersji każdej części oddzielnie:

$$\begin{array}{r|l} 1510 & \\ 188 & 6 \\ 23 & 4 \\ 2 & 7 \\ 0 & 2 \end{array}$$

$$(1510)_{10} = (2746)_8$$

$$\begin{array}{ll} 0.90625 \times 8 = 7.25 & \text{część całkowita} = 7 \\ 0.25000 \times 8 = 2.00 & \text{część całkowita} = 2 \end{array}$$

$$(0.90625)_{10} = (0.72)_8$$

$$(1510.90625)_{10} = (2746.72)_8$$

Zamiany liczby dziesiętnej 1951.65625 na liczbę szesnastkową dokonuje się po uprzednim podzieleniu jej na część całkowitą (1951) i część ułamkową (0.65625) oraz w wyniku dokonania konwersji każdej części oddzielnie:

$$\begin{array}{r|l} 1951 & \\ 121 & (15)_{10} = (F)_{16} \\ 7 & 9 \\ 0 & 7 \end{array} \qquad (1951)_{10} = (79F)_{16}$$

$$\begin{aligned} 0.65625 \times 16 &= 10.5 \quad \text{część całkowita} = (10)_{10} = (A)_{16} \\ 0.50000 \times 16 &= 8.00 \quad \text{część całkowita} = 8 \end{aligned}$$

$$(0.65625)_{10} = (0.A8)_{16}$$

$$(1951.65625)_{10} = (79F.A8)_{16}$$

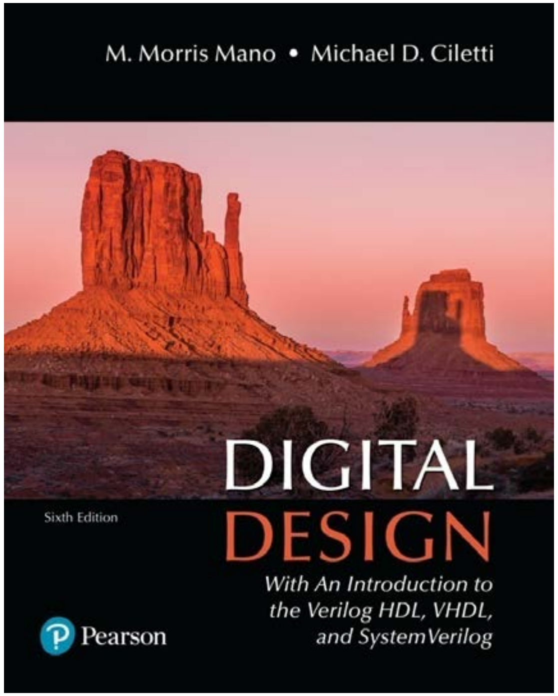
Liczba dwójkowa może być bezpośrednio zamieniona na liczbę ósemkową. W tym celu należy podzielić ją na grupy 3-bitowe, poczynając od kropki w lewo i w prawo, a następnie zastąpić otrzymane grupy odpowiadającymi im cyframi ósemkowymi. Na przykład:

$$(010\ 111\ 100\ 110.111\ 010)_2 = (2746.72)_8$$

Liczba dwójkowa może być bezpośrednio zamieniona na liczbę szesnastkową. W tym celu należy podzielić ją na grupy 4-bitowe, poczynając od kropki w lewo i w prawo, a następnie zastąpić otrzymane grupy odpowiadającymi im cyframi szesnastkowymi. Na przykład:

$$(0111\ 1001\ 1111.1010\ 1000)_2 = (79F.A8)_{16}$$

M. Morris Mano, Michael D. Ciletti, Digital Design, With an Introduction to the Verilog HDL, VHDL, and SystemVerilog, Pearson, 6th Edition, 2018.



Decimal (base 10)	**Binary (base 2)**	**Octal (base 8)**	**Hexadecimal (base 16)**
00	0000	00	0
01	0001	01	1
02	0010	02	2
03	0011	03	3
04	0100	04	4
05	0101	05	5
06	0110	06	6
07	0111	07	7
08	1000	10	8
09	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Reprezentacja liczb całkowitych:

- kod znak-moduł (*sign-magnitude*),
- kod znak-uzupełnienie do 1 (kod U1), *1's complement*, *ones' complement*,
- kod znak-uzupełnienie do 2 (kod U2), *2's complement*, *two's complement*.

Reprezentacja liczb całkowitych:

- kod znak-moduł,
- kod znak-uzupełnienie do 1 (kod U1),
- kod znak-uzupełnienie do 2 (kod U2).

Reprezentacja liczby całkowitej w kodzie znak-moduł

- Liczba dodatnia: 0 na pierwszym bicie (bicie znaku).
- Liczba ujemna: 1 na pierwszym bicie (bicie znaku).
- Na przykład: 3 w zapisie znak-moduł: 0 011
- Na przykład: -3 w zapisie znak-moduł: 1 011

Zapis dziesiętny	Zapis znak-moduł
3	0 011
2	0 010
1	0 001
0	0 000
0	1 000
-1	1 001
-2	1 010
-3	1 011

Reprezentacja liczb całkowitych.

- ~~Kod znak-moduł,~~
- Kod znak-uzupełnienie do 1 (kod U1),
- Kod znak-uzupełnienie do 2 (kod U2).

- Jak otrzymać liczbę w kodzie U1 ?
- Jak otrzymać liczbę w kodzie U2 ?

Istnieją 2 rodzaje uzupełnień dla każdego systemu liczbowego o podstawie $\boldsymbol{p}$:

- uzupełnienie do $\boldsymbol{p-1}$
- uzupełnienie do $\boldsymbol{p}$.

Uzupełnienie do $\boldsymbol{p-1}$ dla n-cyfrowej liczby N o podstawie p jest zdefiniowane jako:

$$\boldsymbol{p^n - 1 - N}$$

Dla liczb dziesiętnych $\boldsymbol{p-1} = 9$, dla liczb dwójkowych $\boldsymbol{p-1} = 1$.

Uzupełnieniem do 9 liczby $(3456)_{10}$ jest

$$\boldsymbol{p^n - 1 - N} = (10^4 - 1) - 3456 = 9999 - 3456 = 6543$$

Uzupełnieniem do 1 liczby $(1001)_2$ jest

$$\boldsymbol{p^n - 1 - \mathbf{N}} = (2^4 - 1) - 1001 = 1111 - 1001 = 0110$$

Uzupełnienie do 1 liczby dwójkowej uzyskuje się w wyniku odjęcia każdej jej cyfry od 1. Odejmując cyfry dwójkowe od 1 otrzymujemy $1 - 0 = 1$ lub $1 - 1 = 0$. W obu przypadkach cyfra otrzymana w wyniku odjęcia od 1 jest negacją jej wartości. Stąd, w celu otrzymania uzupełnienia do 1 danej liczby dwójkowej należy zanegować wszystkie jej cyfry.

Na przykład, uzupełnieniem do 1

liczby dwójkowej:	1	0	1	0	1	0	0	1
jest:	0	1	0	1	0	1	1	0

Uzupełnienie do ***p*** dla *n*-cyfrowej liczby *N* o podstawie ***p*** jest zdefiniowane jako:

$$\boldsymbol{p^n - N}$$

Wyrażenie $p^n - N$ możemy zapisać w postaci:

$$p^n - N = [\, p^n - 1 - N \,] + 1$$

co oznacza, że uzupełnienie do p liczby N otrzymujemy przez dodanie 1 do uzupełnienia do $p - 1$ liczby N.

Jeszcze raz: uzupełnienie do $p - 1$ dla n-cyfrowej liczby N o podstawie p jest zdefiniowane jako:

$$p^n - 1 - N$$

Reprezentacja liczby całkowitej w kodzie znak-uzupełnienie do 2:

- liczba dodatnia jest reprezentowana tak jak w zapisie znak-moduł,
- liczbę ujemną można otrzymać w następujący sposób (to nie jest definicja): negowany jest każdy bit reprezentujący liczbę dodatnią, a następnie do tak zanegowanego ciągu bitów dodaje się 1.

Reprezentacja liczby całkowitej w kodzie znak-uzupełnienie do 2:

- 3 w zapisie znak-uzupełnienie do 2: 0 011
- -3 w zapisie znak-uzupełnienie do 2 można otrzymać w następujący sposób: negowany jest każdy bit liczby 3, to znaczy negowany jest każdy bit w ciągu bitów 0011. Po negacji otrzymujemy 1100. Następnie do 1100 dodajemy 1 otrzymując 1101.

Uzupełnienie do 2 można również otrzymać pozostawiając, od prawej strony do lewej, wszystkie zera i pierwszą jedynkę bez zmian i negując pozostałe cyfry (to nie jest definicja i należy udowodnić, że tak uzyskany wynik jest poprawny).

Pozostawiając w liczbie 10010100, od prawej strony do lewej, wszystkie zera i pierwszą jedynkę bez zmian, a następnie negując pozostałe cyfry

	1	0	0	1	0	1	0	0
	negacja					bez zmian		
otrzymujemy jej uzupełnienie do 2 równe:	0	1	1	0	1	1	0	0

M. Morris Mano, Michael D. Ciletti, Digital Design, With an Introduction to the Verilog HDL, VHDL, and SystemVerilog, Pearson, 6th Edition, 2018.

Decimal	Signed-2's Complement	Signed-1's Complement	Signed Magnitude
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
+4	0100	0100	0100
+3	0011	0011	0011
+2	0010	0010	0010
+1	0001	0001	0001
+0	0000	0000	0000
−0	—	1111	1000
−1	1111	1110	1001
−2	1110	1101	1010
−3	1101	1100	1011
−4	1100	1011	1100
−5	1011	1010	1101
−6	1010	1001	1110
−7	1001	1000	1111
−8	1000	—	—

Zaletą zapisu znak-uzupełnienie do 2, w porównaniu z pozostałymi zapisami, jest istnienie tylko jednej reprezentacji 0 (zera).

M. Morris Mano, Michael D. Ciletti, Digital Design, With an Introduction to the Verilog HDL, VHDL, and SystemVerilog, Pearson, 6th Edition, 2018.

Decimal Symbol	BCD Digit
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Kod ASCII
American Standards Code for Information Interchange.

System kodowania znaków. W tablicy znaków ASCII zebrane są używane znaki alfanumeryczne (litery, cyfry i inne znaki) i przyporządkowane im kody liczbowe. Istnieją dwa rodzaje kodu ASCII: 7-bitowy (możliwość zapisu 128 znaków), oraz 8-bitowy (256 znaków, można zdefiniować tam np. polskie znaki narodowe). Kody ASCII są zazwyczaj podawane w postaci liczb w zapisie szesnastkowym.

Znak	Kod binarny ASCII	Kod szesnastkowy ASCII
A	0100 0001	4 1
a	0110 0001	6 1
B	0100 0010	4 2
b	0110 0010	6 2
. . .	. . .	. . .
1	0011 0001	3 1
2	0011 0010	3 2
. . .		

Przykład dodawania dwóch liczb binarnych:

$$\begin{array}{r} \color{blue}{1111} \leftarrow \text{przeniesienie} \\ \hline 1101 \\ 1011 \\ \hline 11000 \end{array}$$

Przykład odejmowania dwóch liczb binarnych:

$$\begin{array}{r} \color{blue}{1} \leftarrow \text{pożyczka} \\ \hline 1101 \\ -1011 \\ \hline 10 \end{array} \qquad \begin{array}{r} 10000 \\ -\quad 11 \\ \hline 1101 \end{array} \qquad \begin{array}{r} 111001 \\ -1011 \\ \hline 101110 \end{array}$$

$$\begin{array}{r} 11101 \\ -10011 \\ \hline 1010 \end{array}$$