



Materiały Informatyka 1

Opracował: mgr inż. Woźniak Piotr

1. Konwersja liczby w systemie 10 na 2 (przykład)

$57_{(10)}$ – liczba konwertowana

$?_{(2)}$ - liczba po konwersji

a) metoda I

Liczba dzielona jest n-razy przez liczbę bazową (w tym przypadku 2), aż do 0.

i	operacja	zamiana postaci	reszta
1	$57/2 = 28.5$	czyli $56/2 + 1 = 28 + 1$	1
2	$28/2 = 14$	czyli $28/2 + 0 = 14 + 0$	0
3	$14/2 = 7$	czyli $14/2 + 0 = 7 + 0$	0
4	$7/2 = 3.5$	czyli $7/2 + 1 = 3 + 1$	1
5	$3/2 = 1.5$	czyli $2/2 + 1 = 2 + 1$	1
6	$1/2 = 0.5$	czyli $0/2 + 1 = 0 + 1$	1
7	$0/2$	KONIEC	0

Liczb 57 w systemie 2 to reszta 0111001 pobierana od dołu do góry.

UWAGA: liczy się najstarsza jedynek. Zera nie mają znaczenia, czyli 0010 jest równe 10.

b) metoda II

Poszukiwanie największej, potęgi liczby 2 ($2^0, 2^1, 2^2, 2^3, 2^4 \dots = 1, 2, 4, 8, 16 \dots$)
mniejszej lub równej konwertowanej liczbie.

Zapisujemy:

8	7	6	5	4	3	2	1
128	64	32	16	8	4	2	1
0	0	0	0	0	0	0	0

największą liczbą mniejszą lub równą 57 jest 32 (zamieniamy na 1),

0	0	1	0	0	0	0	0
---	---	----------	---	---	---	---	---

$57 - 32 = 25$ największą liczbą mniejszą lub równą 25 jest 16 (zamieniamy na 1),

0	0	1	1	0	0	0	0
---	---	---	----------	---	---	---	---

$25 - 16 = 9$ największą liczbą mniejszą lub równą 9 jest 8 (zamieniamy na 1),

0	0	1	1	1	0	0	0
---	---	---	---	---	---	---	---

$9 - 8 = 1$ największą liczbą mniejszą lub równą 1 jest 1 (zamieniamy na 1),

0	0	1	1	1	0	0	1
---	---	---	---	---	---	---	---



Wynik to **00111001**, czyli **111001**₍₂₎.

UWAGA: Przy przeszukiwaniu zaczynamy od najstarszego bitu (lewa strona).

2. Konwersja liczby w systemie 2 na 10 (przykład)

10010₍₂₎ – liczba konwertowana

?₍₁₀₎ - liczba po konwersji

Zapisujemy kolejne potęgi liczby 2:

7	6	5	4	3	2	1	0
2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
128	64	32	16	8	4	2	1
0	0	0	1	0	0	1	0

wynik to suma mnożenia (potęgi dwójki i składowej liczby) kolejnych:

$$2^7 * 0 + 2^6 * 0 + 2^5 * 0 + 2^4 * 1 + 2^3 * 0 + 2^2 * 0 + 2^1 * 1 + 2^0 * 0 =$$

$$128 * 0 + 64 * 0 + 32 * 0 + 16 * 1 + 8 * 0 + 4 * 0 + 2 * 1 + 1 * 0$$

Wynik to **18**₍₁₀₎.

3. Konwersja dla liczb systemu 10 na system 8 i 16.

70₍₁₀₎ – liczba konwertowana

?₍₈₎ - liczba po konwersji

Liczba dzielona jest n-razy przez liczbę bazową (w tym przypadku 8), aż do 1.

i	operacja	zamiana postaci	reszta
1	70/8 = 8.75	czyli 64/8 + 6 = 8 + 6	6
2	8/8 = 1	czyli 8/8 + 0 = 1 + 0	0
3	1/8	czyli 1/8 + 0 = 0 + 1	1
4	0/8	KONIEC	0

Wynik to **106**₍₈₎.

Podobnie postępujemy przy konwersji na system 16.

?₍₁₆₎ - liczba po konwersji

i	operacja	zamiana postaci	reszta
1	$70/16 = 4$	czyli $64/16 + 0 = 4 + 0$	6
2	$4/16 = 1$	czyli $0/16 + 4 = 0 + 4$	4
3	$0/16$	KONIEC	0

Wynik to $40_{(16)}$.

4. Konwersja dla liczby 10 na dowolny system

$29_{(10)}$ – liczba konwertowana

$?_{(3)}$ - liczba po konwersji

UWAGA: liczbę bazową (3) można zmienić dla konwersji na inne systemy.

Liczba dzielona jest n-razy przez liczbę bazową (w tym przypadku 3), aż do 1.

i	operacja	zamiana postaci	reszta
1	$29/3 = 9.666...$	czyli $27/3 + 2 = 9 + 2$	2
2	$9/3 = 3$	czyli $9/3 + 0 = 3 + 0$	0
3	$3/3$	czyli $3/3 + 0 = 1 + 0$	0
4	$1/3$	czyli $0/3 + 1 = 0 + 1$	1
5	$0/3$	KONIEC	

Wynik to $1002_{(3)}$.

5. Konwersja dla liczby dowolnego systemu na 10

$1002_{(3)}$ – liczba konwertowana

$?_{(10)}$ - liczba po konwersji

Wynik to suma mnożenia (potęgi bazy i składowej liczby) kolejnych:

$$3^3 * 1 + 3^2 * 0 + 3^1 * 0 + 3^0 * 2 = 27 + 2 = 29$$

Wynik to $29_{(10)}$.

6. Konwersja dla liczby systemu 8 i 16 na 2

$121_{(10)}$ – liczba konwertowana

$?_{(8)}$ - liczba po konwersji

Liczba $121_{(10)}$ to $1111001_{(2)}$.

0	1	1	1	1	0	0	1
---	---	---	---	---	---	---	---

zapis ósemkowy wymaga 3 bitów (000 do 111),
dzielimy liczbę na części.

0	0	1	1	1	1	0	0	1
---	---	---	---	---	---	---	---	---

UWAGA: jeżeli brakuje pozycji do 3 dopisujemy 0 po lewej stronie.

Wynik to:

- $001_{(2)} = 1_{(8)}$
- $111_{(2)} = 7_{(8)}$
- $001_{(2)} = 1_{(8)}$

Ostatecznie: $171_{(8)}$

Podobnie w przypadku (*pobiera się 4 bity*):

$121_{(10)}$ – liczba konwertowana

$?_{(16)}$ - liczba po konwersji

0	1	1	1	1	0	0	1
---	---	---	---	---	---	---	---

Wynik to:

- $0111_{(2)} = 7_{(16)}$
- $1001_{(2)} = 9_{(16)}$

UWAGA: dla wyników spoza zakresu 9 liczb wykorzystuje się litery (A-10, B-11, C-12, D-13, E-14, F-15).

Ostatecznie: $79_{(16)}$

7. Konwersja dla liczby 2 na systemy 8 i 16

$11000101_{(2)}$ – liczba konwertowana

$?_{(8)}$ - liczba po konwersji

Podobnie jak w przykładzie 4 dzielimy liczbę na bloki.

0	1	1	0	0	0	1	0	1
---	---	---	---	---	---	---	---	---

Następnie zamieniamy bloki na system 8.

- $011_{(2)} = 3_{(8)}$
- $000_{(2)} = 0_{(8)}$
- $101_{(2)} = 5_{(8)}$

Wynik to: $305_{(8)}$

$11000101_{(2)}$ – liczba konwertowana

$?_{(16)}$ - liczba po konwersji

1	1	0	0	0	1	0	1
---	---	---	---	---	---	---	---

Następnie zamieniamy bloki na system 16.

- $1100_{(2)} = C_{(16)}$
- $0101_{(2)} = 5_{(16)}$

Wynik to: $C5_{(16)}$

8. Standardu IEEE 754 na liczbę(10)

11000010001010001000000000000000 ₍₇₅₄₎ – liczba konwertowana
?₍₁₀₎ - liczba po konwersji

11000010001010001000000000000000

1 – znak (1-liczba ujemna)

10000100 – cecha

010100010000000000000000 - mantysa

dla znaku:

1 = -1 (*minus*)

dla cechy:

10000100 _(BIAS=127) = 132 – 127 = 5

dla mantysy:

01.010100010000000000000000

1	1/2	1/4	1/8	1/16	1/32	1/64	1/128	1/256	
1	0	1	0	1	0	0	0	1	...

$1 + 1/4 + 1/16 + 1/256 = 1 + 64/256 + 16/256 + 1/256 = 1 * 81/256$

dla całości:

$(-1)^1 * 2^5 * 337/256 = -32 * 337/256 = 42.125$

9. Liczbę(10) na standard IEEE 754

12.65 ₍₁₀₎ – liczba konwertowana

?₍₇₅₄₎ - liczba po konwersji

- 12 = 1100

0.65	x2	1
0.3	X2	0
0.6	X2	1
0.2	X2	0
0.4	X2	0
0.8	X2	1
0.4	X2	0
0.8	X2	1
...	...	...

- 1100.101001010101...
- 1.100101001010101... x 2³
- 127 _(BIAS=127) + 3 = 130 = 10000010
- 10000010100101001010101...

Wynik to: 01000001010010100101010101010101

10. Operacje arytmetyczne i logiczne:

1. Przykładowe operacje logiczne:

$$0 \text{ AND } 0 = 0$$

$$0 \text{ AND } 1 = 0$$

$$1 \text{ AND } 1 = 1$$

$$1 \text{ AND } 0 = 0$$

$$1001 \text{ AND } 1000 = 1000$$

$$000 \text{ AND } 111 = 000$$

$$0 \text{ OR } 0 = 0$$

$$0 \text{ OR } 1 = 1$$

$$1 \text{ OR } 1 = 1$$

$$1 \text{ OR } 0 = 1$$

$$1001 \text{ OR } 1000 = 1001$$

$$000 \text{ OR } 111 = 111$$

$$\text{NOT } 10101 = 01010$$

2. Przykładowe operacje arytmetyczne:

- $\sim 10010 = 01101$
- $11001 + 1 = 11010$
- $11111 - 10 = 11101$

11. Kodowanie liczb (10) na ZM, ZU1, ZU2 (przykład z zapisem 8-mio bitowym)

-10₍₁₀₎ – liczba konwertowana

?_(ZM) - liczba po konwersji

$$10_{(10)} = 00001010_{(2)}$$

$$10_{(10)} = 00001010_{(ZM)}$$

$$-10_{(10)} = 10001010_{(ZM)}$$

$$10_{(10)} = 00001010_{(ZU1)}$$

$$-10_{(10)} = 11110101_{(ZU1)}$$

$$10110101_{(2)} = 181_{(10)}$$

$$10110101_{(ZM)} = -53_{(10)}$$

$$10110101_{(ZU1)} = -74_{(10)}$$

$$10110101_{(ZU2)} = -75$$

$$10_{(10)} = 00001010_{(ZU1)}$$

$$-10_{(10)} = 11110110_{(ZU2)}$$

UWAGA: Konwersja systemu 10 na ZM polega na zarezerwowaniu najstarszego bit-u dla znaku 1(-), 0(+).

UWAGA: Konwersja systemu 10 na ZU1 polega na negowaniu 7 najmłodszych bit-ów postaci ZM w przypadku liczby ujemnej. W przypadku liczby dodatniej nie następuje negacja.

UWAGA: Konwersja systemu 10 na ZU2 polega na dodaniu do postaci ZU1 bit-a w przypadku liczby ujemnej.

Dodatkowe materiały:

1. http://towarzystwo.edu.pl/assets/prace_matematyczne/sm_14_kaczowka_makowski_rajs_szlubowski.pdf
2. https://eduinf.waw.pl/inf/alg/006_bin/0022.php
3. <https://www.h-schmidt.net/FloatConverter/IEEE754.html>
4. https://eduinf.waw.pl/inf/alg/006_bin/0016.php
5. https://eduinf.waw.pl/inf/alg/006_bin/0017.php
6. https://eduinf.waw.pl/inf/alg/006_bin/0018.php
7. <https://www.convertworld.com/pl/liczebnik/trojkowy-system-liczbowy.html>