

Langage machine



Année académique 2021-2022

Conversions

Algorithmes de conversions
entre les trois bases utiles

Conversion décimal → binaire

Méthode des divisions successives par 2

149	1
74	0
37	1
18	0
9	1
4	0
2	0
1	1
0	

↑

$149_{10} = 1001\ 0101_2$

Conversion binaire → décimal

$$\begin{array}{rcccccccc} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & {}_2 \\ \hline *2 & | & | & | & | & | & | & | & \\ 2 + 0 & | & | & | & | & | & | & | & \\ \hline *2 & | & | & | & | & | & | & | & \\ 4 + 0 & | & | & | & | & | & | & | & \\ \hline *2 & | & | & | & | & | & | & | & \\ 8 + 1 & | & | & | & | & | & | & | & \\ \hline *2 & | & | & | & | & | & | & | & \\ 18 + 0 & | & | & | & | & | & | & | & \\ \hline *2 & | & | & | & | & | & | & | & \\ 36 + 1 & | & | & | & | & | & | & | & \\ \hline *2 & | & | & | & | & | & | & | & \\ 74 + 0 & | & | & | & | & | & | & | & \\ \hline *2 & | & | & | & | & | & | & | & \\ 148 + 1 & | & | & | & | & | & | & | & \\ \hline & & & & & & & & \\ 148 + 1 & = & 149_{10} \end{array}$$

Conversion binaire → décimal

Méthode conseillée

$$10_2 = 2_{10}$$

$$\begin{array}{rcll} 1 & 0 & 1 & 1_2 = & 1 * 2_{10}^0 & 1 \\ & & & & 1_{10} & \\ & & & + & 1 * 2_{10}^1 & 2 \\ & & & & 2_{10} & \\ & & & + & 0 * 2_{10}^2 & 0 \\ & & & & 4_{10} & \\ & & & + & 1 * 2_{10}^3 & 8 \\ & & & & 8_{10} & \\ & & & & & \hline & & & & & 11_{10} \end{array}$$

Conversion **hexadécimal** → binaire

$$16 = 2^4$$

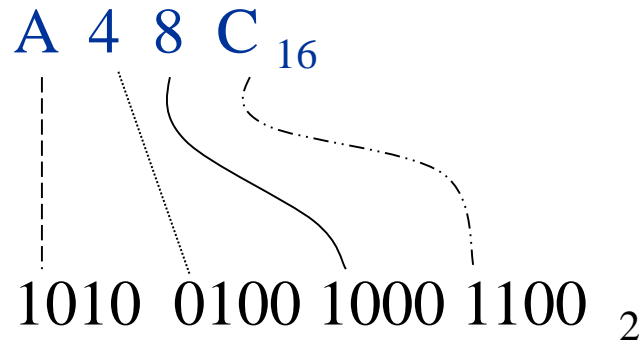
Chaque chiffre hexadécimal

→ 4 chiffres binaires

0_{16}	→	0000_2
1_{16}	→	0001_2
2_{16}	→	0010_2
3_{16}	→	0011_2
4_{16}	→	0100_2
5_{16}	→	0101_2
6_{16}	→	0110_2
7_{16}	→	0111_2
8_{16}	→	1000_2

9_{16}	→	1001_2
A_{16}	→	1010_2
B_{16}	→	1011_2
C_{16}	→	1100_2
D_{16}	→	1101_2
E_{16}	→	1110_2
F_{16}	→	1111_2

Conversion hexadécimal \rightarrow binaire



Remplacer chaque chiffre hexadécimal
par son équivalent binaire (en 4 chiffres)

Remarque : on peut supprimer les 0
en début de nombre

Exemple : 39D₁₆ = X 11 1001 1101₂

Conversion binaire → hexadécimal

$$\begin{array}{cccc} \underline{1010} & \underline{0100} & \underline{1000} & \underline{1100}_2 \\ A & 4 & 8 & C \end{array}$$

$$1010010010001100_2 = A48C_{16}$$

Regrouper les chiffres binaires par 4
(en commençant par la fin du nombre)

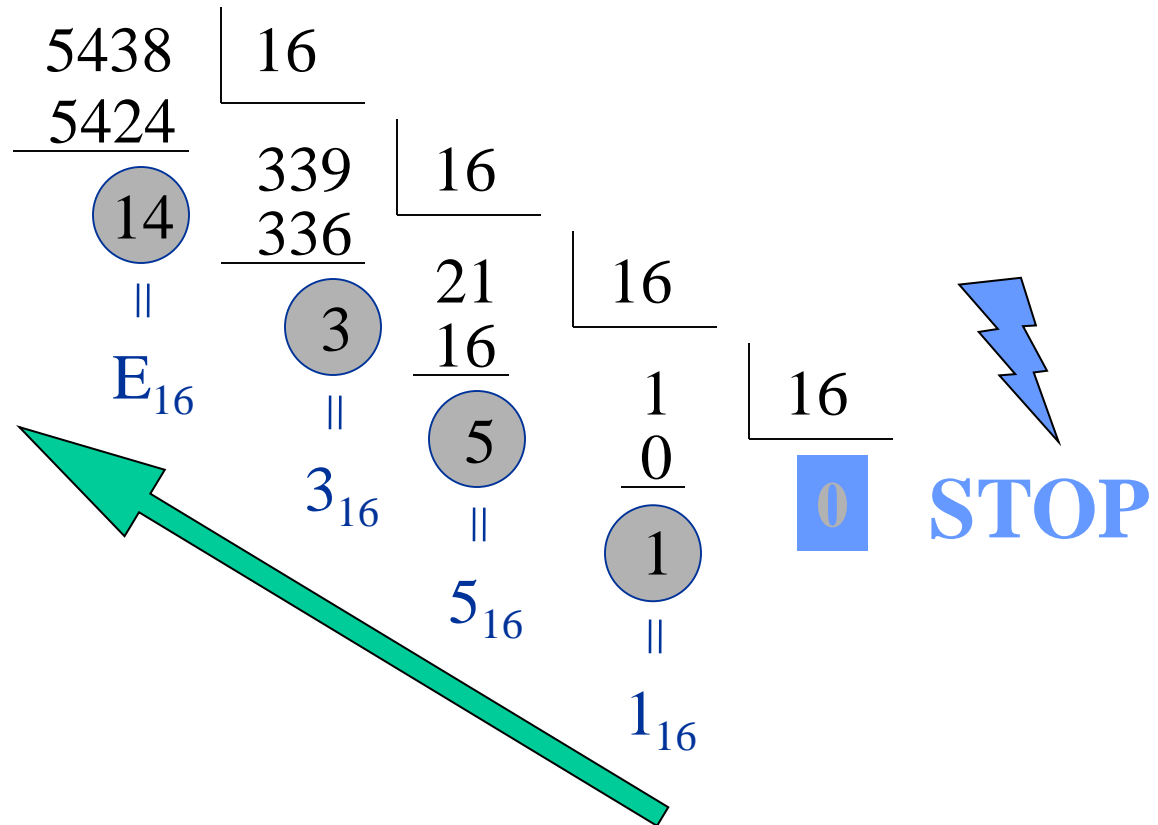
Remplacer chaque groupe de 4 chiffres binaires
par son équivalent hexadécimal

$$\begin{array}{ccc} \text{Exemple :} & \underline{0011} & \underline{1001} & \underline{1011}_2 \\ & 3 & 9 & B \end{array}$$

$$1110011011_2 = 39B_{16}$$

Conversion décimal → hexadécimal

Méthode des divisions successives par 16



Conversion hexadecimal \rightarrow decimal

$$\begin{array}{r} 1 \quad 5 \quad 3 \quad E \quad 16 \\ *16 \\ \hline 16 \quad + \quad 5 \\ \quad *16 \\ \hline 336 \quad + \quad 3 \\ \quad *16 \\ \hline 5424 \quad + \quad 14 = 5438_{10} \end{array}$$

$$153E_{16} = 5438_{10}$$

Conversion hexadécimal → décimal

autre méthode

$$10_{16} = 16_{10}$$

$$\begin{array}{rcll} 1 \ 5 \ 3 \ E_{16} & = & E * 16_{16}^0 & \\ & & 1_{10} & 14 \\ & & + 3 * 16_{16}^1 & \\ & & 16_{10} & 48 \\ & & + 5 * 16_{16}^2 & \\ & & 256_{10} & 1280 \\ & & + 1 * 16_{16}^3 & \\ & & 4096_{10} & \underline{4096} \\ & & & 5438_{10} \end{array}$$

Langage assembleur

NASM 32 bits

MOV

- **MOV copie** le contenu de la source (deuxième) opérande dans la destination (première) opérande
- **Les opérandes** doivent être (quasiment) **toujours de même taille !**
- Exemples où le registre fixe la taille (32 bits)
 - ✓ `MOV eax,0x12345678`
 - ✓ `MOV ebx,0x9ABC`
- Nous verrons les opérandes en mémoire la semaine prochaine

INC et DEC

- **INC** reg32/mem32

- ✓ **Ajoute 1**

- au registre ou à la variable en mémoire,
sur 32 bits

- ✓ reg32/mem32 se note en abrégé r/m32

- ✓ Exemples : INC eax, INC ebx

- **DEC** r/m32

- ✓ **Enlève 1**

- au registre ou à la variable en mémoire,
sur 32 bits

- ✓ Exemples : DEC ecx, DEC edx

ADD

- **ADD** r/m32,reg32
 - ✓ Additionne les deux variables et place le résultat dans la première
 - ✓ $r/m32 = r/m32 + reg32$
- **ADD** reg32,imm32
 - ✓ La seconde variable est une **valeur immédiate** sur 32 bits
 - ✓ Exemple : **ADD** ebx,4

SUB

- SUB r/m32,reg32
 - ✓ Soustrait la seconde variable de la première et place le résultat dans la première
- SUB reg32,imm32
 - ✓ La seconde variable est une **valeur immédiate** sur 32 bits
 - ✓ Exemple SUB edx,2

Ressource externe

- D'autres combinaisons d'opérandes existent... voir la documentation

<http://home.myfairpoint.net/fbkotler/nasmdocc.html>