UNIVERSITE ZIANE ACHOUR DJELFA

1 ère année MI
Module: Structure Machine

Solutions de la 1éré Serie d'exercices

Exercice 1 :: *donner la valeur décimale du nombre binaire N=1010*
Solution :

$$= 1x2^3 + 0x2^2 + 1x2^1 + 0x2^0 = 8 + 0 + 2 + 0 = 10_{(10)}$$

Exercice 02*: convertir le nombre N=75(10) en nombre Octal.*
Solution
On aura successivement :

75 - 64 ⟶ 1×8^2
11 - 8 ⟶ 1×8^1
3 ⟶ 3×8^0
Donc $N=75_{(10)}= 1 \times 8^2 + 1 \times 8^1 + 3 \times 8^0$
On a donc : $N=75_{(10)} =113_{(8)}$

i	8^i
0	1
1	8
2	64

Exercice 03 *: convertir $Nf=0,1011_{(2)}$ en décimal.*

Solution : nous obtenons :
$Nf=1x2^{-1} + 0x2^{-2} + 1x2^{-3}+ 1x2^{-4} =0.5+0.125+0.0625=0.6875_{(10)}$

Exercice 04 : convertir $Nf=0,85_{(10)}$ en binaire, prendre 04 chiffres après la virgule.

Solution : nous obtenons :
0.85 x 2 = 1,70
0.70 x 2 = 1.4
0.4 x 2 = 0.8
0.8 x 2= 1.6

On écrit de gauche à droite les nombres encadrés pris de haut en bas.
On aura donc $Nf=0,85_{(10)} = 0,1101_{(2)}$

Exercice 05 *:*

Convertir le nombre binaire N=11010101,11(2) en Octal.

Solution :

N=	011	010	101	,	$110_{(2)}$
	3	2	5	,	$6_{(8)}$

Convertir le nombre octal $N=657{,}12_{(8)}$ *en binaire.*

Solution :

Ecrire par blocs de trois bits, la valeur binaire des chiffres du nombre octal. On obtient :

6	5	7	,	1	$2_{(8)}$
110	101	111	,	001	$010_{(2)}$

Convertir le nombre binaire N=11010101101,001(2) en Octal.

Solution :

	0110	1010	1101	,	$0010_{(2)}$
N=	6	A	D	,	$6_{(16)}$

Convertir le nombre Hexadécimal $N=AB0{,}C1_{(16)}$ *en binaire.*

Solution

Ecrire par blocs de quatre bits, la valeur binaire des chiffres du nombre hexadécimal. On obtient :

A	B	0	,	C	$1_{(16)}$
1010	1011	0000	,	1100	$0001_{(2)}$

Exercice 06 :

Peut-on représenter le nombre -8 sur 04 bits en SVA (signe et valeur absolue) ?

Réponse :

Il est impossible de représenter le chiffre -8 sur 4 bits car sa valeur absolue $|-8_{(10)}|$ qui est égale à $1000_{(2)}$ prends déjà 04 bits et donc on aura besoin au minimum de 5 bits pour pouvoir représenter son bit de signe.

Quels sont les nombres qu'on peut représenter sur 04 bits ?

Solution : d'après le tableau 5, on peut représenter sur 4 bits, l'intervalle de nombres entiers : De $[-(2^3 - 1) , (2^3 - 1)]$ soit de $[-7 , +7]$.

Exercice 07 :

Quelle est la valeur décimale du nombre binaire 10110110 représenté en complément à 1?

Solution : =

Le bit de poids fort indique qu'il s'agit d'un nombre négatif. Donc la Valeur décimale - CP1(10110110) = $-(01001001)_2 = -(73)_{10}$

Quels sont les nombres en complément à 1 qu'on peut représenter sur 04 bits ?

Solution:

- Le plus grand nombre positif représentable est donc 0111 ce qui représente $2^3 - 1$ soit +7
- Le plus petit négatif est -0111. Ce qui donne $-(2^3 -1)$ soit -7

Donc, on constate que sur 04 bits, on peut représenter les nombres qui sont dans l'intervalle $[-7_{(10)},+7_{(10)}]$, soit $[-(2^3 - 1) , +(2^3 - 1)]$

Plus généralement, si on travaille sur n bits, l'intervalle des valeurs qu'on peut représenter en CP1 est : $[-(2^{n-1} - 1), +(2^{n-1} - 1)]$.

Exercice 08 :

Quelle est la valeur décimale correspondante au nombre binaire 10110110 représenté en complément à 1?

Quels sont les nombres en complément à 1 qu'on peut représenter sur 04 bits ?

Solution :

Le bit de poids fort indique qu'il s'agit d'un nombre négatif. Donc la Valeur décimale = - CP1(10110110) = - (01001010)2= - (74)10

Quels sont les nombres qu'on peut représenter sur 04 bits ?

Solution :

Donc, d'après le tableau 7, on constate que sur 04 bits, on peut représenter les nombres qui sont dans l'intervalle $[-8_{(10)},+7_{(10)}]$, soit $[-2^3 , +(2^3 - 1)]$

Plus généralement, si on travaille sur n bits, l'intervalle des valeurs qu'on peut représenter en CP2 est : $[-(2^{n-1}), +(2^{n-1} - 1)]$.

Exercice 09:

Effecteur en C1 les opérations suivantes.

- Sur 5 bits, l'opération: 8- 9.

- Sur 5 bits, l'opération: -8 + -9

Solution: s nombres doivent être sur 5 bits y compris le bit de signe

$(+8) = 01000_{(2)}$
$(+9) = 01001_{(2)}$

Le complément à 1 de 01001 est 10110 = - $9_{(10)}$

	0	1	0	0	0	←	diminuende
+	1	0	1	1	0	←	diminuteur
=	1	1	1	1	0		

Le bit du signe =1 → résultat négatif → le résultat = - Cp1(11110).
Dans ce cas, calculons le complément à 1 du résultat

Cp1(11110)=$00001_{(2)}$. Cela veux dire que le résultat = -$1_{(2)}$

Sur 8 bits effectuons l'opération : (-8) + (-9)

Solution :

Dans ce cas chaque nombre est représenté par son complément à 1 :
(+8) = 01000, (-8)= cp1(01000)=10111
(+9) = 01001, (-9)= cp1(01001)=10110

		1	0	1	1	0
1	1	0	1	1	1	
		0	1	1	0	1
	+					1
		0	1	1	1	0

Exercices 10:

Effecter la même opération en complément à 2. (+8) - (+9).

Solution:

Les nombres doivent être sur 5 bits y compris le bit de signe

$(+8) = 01000_{(2)}$

$(+9) = 01001_{(2)}$

Le complément à 1 de 01001 est 10110 = $-9_{(10)}$

```
  0 1 0 0 0   ←— diminuende
+ 1 0 1 1 0   ←— diminuteur
-----------
= 1 1 1 1 0
```

Le bit du signe =1 → résultat négatif → le résultat = - Cp1(11110).

Dans ce cas, calculons le complément à 1 du résultat

Cp1(11110)=$00001_{(2)}$. Cela veux dire que le résultat = $-1_{(2)}$

Exercices 11:

1. Représenter le nombre réel (-6.125) en format virgule fixe (1 bit de signe, 8 bit pour la part entière et 7 bits pour la partie fractionnaire).

2. Quelle est le plus petit nombre positif représentable dans ce forma

3. Quelle est le plus grand nombre positif représentable dans le même form

Solution :

1. Il faut d'abord convertir le nombre en binaire pour pouvoir le représenter en machine. On a : $-6.125_{(10)} = -110{,}001_{(2)}$.On représente donc le nombre selon le format indiqué. On obtient donc : 1|00000110|0010000
2. le plus petit nombre positif est représenté comme suit : 0|00000000|0000001 ce qui donne la valeur $Nmin=2^{-7}$
3. le plus grand nombre positif est représenté comme suit : 0|11111111|1111111

Pour donner l'équivalent en décimale, calculons la partie entière max (PEmax) et la partie fractionnaire max (PFmax).

$PEmax=2^0+2^1+\ldots+2^7= 2^8-1$

$PFmax =2^{-1}+2^{-2}+\ldots+2^{-7}=1- 2^{-7}$

$Nmax=PE\ max=+PF\ max=2^8-1 +1- 2^{-7}= 2^8- 2^{-7}$

Exercices 12:

RRépresenter par la méthode de l'exposant réel :

1- N= - 1010,10 $_{(2)}$

2- N= - 0,0010 $_{(2)}$

Solution :

1- :

On commence par normaliser la mantisse :

N= - $1010,1001_{(2)}$ =$-0.10101001_{(2)}$ x 2^{+4}. (+4 représente le nombre de déplacement de la virgule vers la gauche).

Dans ce cas : M= - $0.10101001_{(2)}$ et Exp= + $4_{(10)}$ =$0100_{(2)}$

1	0	1	0	0	1	0	1	0	1	0	0	1	0	0	0	0
Signe	Exp				Mantisse											

2- :

On commence par normaliser la mantisse :

N= - $0,001001_{(\ (2)}$ =- $0.1001_{(2)}$ x 2^{-2}. (-2 représente le nombre de déplacement de la virgule vers la droite).

Dans ce cas : M=- $0.1001_{(2)}$ et Exp=$-2_{(10)}$

En virgule flottante, les exposants négatifs sont représentés par la méthode du complément à 2

|Exp|= $+2_{(10)}$ =$0100_{(2)}$

Exp=-2=CP2(0100) =$1100_{(2)}$

1	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
Signe (1 bit)	Exp (4 bits)				Mantisse (12 bits)											

Exercie 13 *: Prenons le même exemple précédent.*

Solution :

$N = -0{,}001001_{(2)} = -0.1001_{(2)} \times 2^{-2}$
$M = -0.1001_{(2)}$ et $Exp = -2_{(10)}$

Calculons maintenant la valeur de l'exposant biaisé :

$$Exp\_biaisé = -2 + (2^4/2) = 2 + 8 = 10_{(10)} = 1010_{(2)}$$

Ce nombre est représenté comme suit dans le format suivant : 12 bits pour la mantisse, 4 bits pour l'exposant biaisé et 1 bit pour le signe de la mantisse.

1	1	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0
Signe (1 bit)	Exp_biaisé (4 bits)				Mantisse (12 bits)											

Exercice 14 :
Soient les deux nombres suivants : $N1 = 0{,}001101_{(2)}$, $N2 = 1{,}101_{(2)}$
Effectuer l'opération N1+N2 en format virgule flottante selon le format suivant : (12 bits pour la mantisse, 4 bits pour l'exposant et 1 bit pour le signe de la mantisse).

N1=0,001101(2) , N2=1, 101(2)

Représenter N1+N2 en format virgule flottante selon le format suivant : (12 bits pour la mantisse, 4 bits pour l'exposant et 1 bit pour le signe de la mantisse).

Solution
On normalise d'abord les mantisses pour cela :
$N1 = 0{,}1101 * 2^{-2}$, $N2 = 0{,}1101 * 2^{1}$
$N1+N2 = 0{,}1101 * 2^{-2} + 0{,}1101 * 2^{1}$
$= 0{,}0001101 * 2^{1} + 0{,}1101 * 2^{1}$
$= 0{,}1110101 * 2^{1}$

Dans ce cas, N1+N2 est représenté comme suit :

0	0	0	0	1	1	1	1	0	1	0	0	0	0	0	0	0
Signe(1 bit)	Exposant (4 bits)				Mantisse (12 bits)											