Interrogation d'informatique (2° trimestre)

SANS CALCULATRICE – 1 heure 30

Nom : ***Classe :.....................................***

Les trois parties sont indépendantes et chacune des parties sera notée sur 20 points : il y aura donc 3 notes sur 20

PARTIE 1

I Compléter le tableau suivant de correspondance entre les systèmes décimal, binaire et hexadécimal :

Décimal	Binaire	Hexadécimal	Décimal	Binaire	Hexadécimal
O			8		
1			9		
2			10		
3			11		
4			12		
5			13		
6			14		
7			15		

II Etablir la correspondance entre les nombres exprimés dans les bases différentes en remplissant les cases vides du tableau :

Décimal	Binaire	hexadécimal
	1010 1010	
	1100 1101 1110	
		1234
		ABCD
257		
32767		

III En système de base 8, peut-on rencontrer le nombre 678 ? Expliquer votre réponse.

IV Effectuer les opérations suivantes (si nécessaire, faire apparaître les retenues) :

En base 2 :

```
  1010 1010        1 0000 0001        1 1010
+        10      +        1011      -      1
=                =                  =
```

en base 16

```
   7FF           ABC          1998
+    1        +  DEF       +     2
=             =            =
```

V En système binaire, comment savoir si un nombre est pair ou impair ?

PARTIE 2

Attention dans cette partie, il ne s'agit pas de SAVOIR (les informations sont données dans l'énoncé) mais de REFLECHIR !

Rem : dans la suite des questions, la base dans laquelle il faut exprimer les résultats est <u>en indice</u>

VI En base 10, avec quatre chiffres, on peut représenter tous les nombres entiers de 0 à 9999 ;

a) en base 2 : avec huit bits :

- quel est le nombre entier le plus petit que l'on puisse représenter ? $X_{Min(2)}$ = ...
- quel est le nombre entier le plus grand que l'on puisse représenter ? $X_{Max(2)}$ = ...
- En déduire l'éventail des nombres en base 2 que l'on puisse ainsi étudier ramené en base 10 :

.................................<= $X_{(10)}$ <=.................................

b) en base 16, avec quatre digits ("chiffres") :

- quel est le nombre entier le plus petit que l'on puisse représenter ? $X_{Min(16)}$ = ...
- quel est le nombre entier le plus grand que l'on puisse représenter ? $X_{Max(16)}$ = ...
- En déduire l'éventail des nombres en base 16 que l'on puisse ainsi étudier ramené en base 10 :

.................................<= $X_{(10)}$ <=.................................

VII En écriture scientifique en base 10, un nombre peut être écrit sous la forme : mantisse x $10^{exposant}$

Exemple le nombre 12,354 peut être écrit sous la forme 1,2354 x 10^1 ou 123,54 x 10^{-1} ou 123540 x 10^{-4} etc.

et il y a une infinité d'écriture suivant l'emplacement donné à la virgule. En informatique, on privilégie l'écriture avec un ***zéro avant la virgule et un chiffre différent de zéro derrière la virgule*** soit l'écriture sur l'exemple ci-dessus :

0,12354 x 10^2 : cette convention permet de "coder" en machine le nombre en deux blocs

un bloc pour la mantisse et un bloc pour l'exposant soit $\boxed{12354}$ $\boxed{+2}$

a) Supposons qu'il y ait cinq chiffres pour la mantisse et 2 chiffres (+ le signe) pour l'exposant pour coder un nombre <u>en base 10 :</u>

- quel est le nombre réel le plus petit que l'on puisse représenter ? $X_{Min(10)}$ = ...
- quel est le nombre réel le plus grand que l'on puisse représenter ? $X_{Max(10)}$ = ...

b) dans l'ordinateur, ce n'est pas la base 10 qui est utilisée mais la base 2 : supposons de même qu'un nombre réel soit représenté par

cinq bits pour la mantisse et 2 bits pour l'exposant (+ 1 bit pour le signe) pour l'exposant :

- quel est le nombre réel le plus petit que l'on puisse représenter ? $X_{Min(2)}$ = ...
- quel est le nombre réel le plus grand que l'on puisse représenter ? $X_{Max(2)}$ = ...
- En déduire l'éventail des nombres en base 10 que l'on puisse ainsi étudier:

.................................<= $X_{(10)}$ <=.................................

Rappels pour la partie III

Opérations booléennes	
NON	NOT
ET	AND
OU	OR
OU_exclusif	XOR
NON-ET	NAND

Ecritures possibles		
NOT (a)	$\overline{a}$	
a AND b	a . b	a ∧ b
a OR b	a + b	a ∨ b
a XOR b	a ⊕ b	a ⊕ b
a NAND b	$\overline{a . b}$	$\overline{a \wedge b}$

PARTIE 3

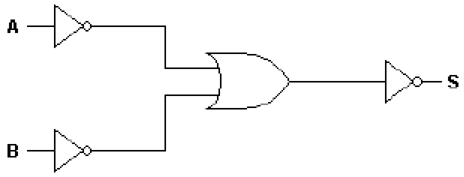
Portes logiques

Exercice I : Rappeler les tables de vérités des deux portes logiques suivantes

Porte NON-ET (ou NAND			*Porte OU exclusif (ou XOR)*		
A	B	S	A	B	S
0	0		0	0	
0	1		0	1	
1	0		1	0	
1	1		1	1	

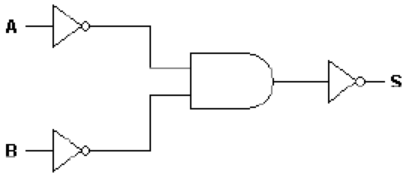
Exercice II Remarque « déterminer l'équation » signifie écrire les opérations sous la forme de l'écriture en grisé du Rappel.

1)
   a. Déterminer l'équation du circuit de la figure suivante :

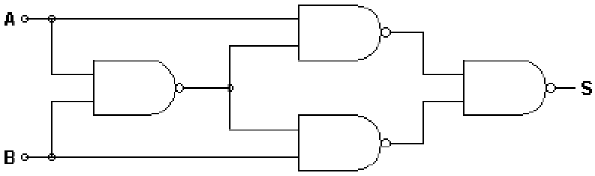


   b. Dresser la table de vérité de ce circuit
   c. Quelle est la fonction logique réalisée et quel est son symbole ?

2) Mêmes questions pour le circuit de la figure suivante :



3) Mêmes questions pour le circuit de la figure suivante :



Expliquer l'intérêt du montage de l'exercice 3)

Exercice 1		
A	B	S
0	0	
0	1	
1	0	
1	1	

Exercice 2		
A	B	S
0	0	
0	1	
1	0	
1	1	

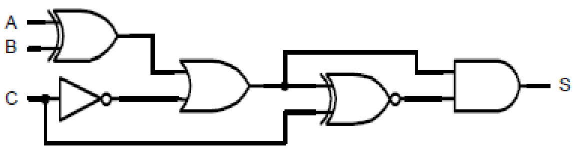
Exercice 3		
A	B	S
0	0	
0	1	
1	0	
1	1	

Fonction Exercice 1 :……………………… …………………… Fonction Exercice 2 :…………………… ………………………

Fonction Exercice 3 :……………………… ……………………………………………………………………

Exercice III

1. Complétez la table de vérité correspondante au circuit logique suivant :



A	B	C	S
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	