

## TD 6 : Opérations sur les bits

Dans ce TD, nous supposons l'existence d'une fonction `AffBin()` qui permet d'afficher un nombre binaire sur 8 bits suivi d'un retour chariot. Par exemple `AffBin(0x75)` affiche 01110101↵.

Exercice 1 : Pour chacun des programmes suivants, trouver ce qui va s'afficher :

- a. 

```
unsigned char a=0xa7;
AffBin (a & 0x11);
AffBin (a | 0xf0);
AffBin (a ^ 0x0F);
AffBin (~a);
```
- b. 

```
unsigned char a=0x30,Mask;
for (Mask=1;Mask!=0;Mask=Mask<<1)
    AffBin (a | Mask);
```
- c. 

```
unsigned char a=0x30,Mask;
for (Mask=0x01;Mask!=0x00;Mask=Mask<<1)
    AffBin (a ^ Mask);
```
- d. 

```
unsigned char a=0x30,Mask;
for (Mask=0x01;Mask!=0x00;Mask=Mask<<1)
    AffBin (a & ~Mask);
```

Exercice 2 : Soit la déclaration suivante :

```
unsigned char var, Mask=0xC3 ;
```

`var` est donnée sous une forme de 8 variables bits : `b7 b6 b5 b4 b3 b2 b1 b0`. Pour chacune des opérations suivantes, donner son résultat :

- Exemple : `~var` → /b7 /b6 /b5 /b4 /b3 /b2 /b1 /b0
- `var & Mask`
- `~( var | Mask)`
- `var ^ Mask`

Exercice 3 : Compléter le tableau suivant :

Action	Forcer un bit à 0		Forcer un bit à 1			
Opérateur	&				^	
Action des bits du masque	0	1	0	1	0	1
	Force le bit à 0	Conserve la valeur initiale				

Exercice 4 : Compléter le programme ci-dessous de façon à ce que les 4 bits de poids forts de a soient forcés à 0 et que les 4 bits de poids faibles soient forcés à 1. Que va afficher le programme ?

```
unsigned char a=0x32;
// A completer
printf ("a=0x%x",a);
```

Exercice 5 : Le port parallèle d'un PC est constitué de différents registres. Le registre d'état décrit ci-dessous permet de connaître l'état de l'imprimante.

	bit 7								bit 0
Poids	128	64	32	16	8	4	2	1	
Nom	/BUSY	/ACK	PE	SELECT	/ERROR	Réservé	Réservé	Réservé	
E/S	Lecture	Lecture	Lecture	Lecture	Lecture	-	-	-	

- /BUSY=0 : l'imprimante est occupée
- /ACK=0 : l'imprimante est prête pour le prochain transfert de données
- PE=1 : l'imprimante n'a plus de papier (PAPER END)
- SELECT=1 : l'imprimante est en ligne
- /ERROR=0 : une erreur est survenue

L'instruction suivante permet de lire l'état de ce registre, le résultat est renvoyé sous forme de unsigned char :

```
inportb(0x379);
```

Ecrire un programme qui attend que l'imprimante soit en ligne. Le programme affiche ensuite: « Printer [OK] ».

Exercice 6 : Comme pour l'exercice 5, le registre de contrôle est détaillé ci-dessous :

	bit 7								bit 0
Poids	128	64	32	16	8	4	2	1	
Nom	Réservé	Réservé	Réservé	IRQ	SLCT-IN	/INIT	AUTO FEED	/STROBE	
E/S	X	X	X	Bidir.	Bidir.	Bidir.	Bidir.	Bidir.	

Le contenu du registre peut être lu et écrit avec les instructions suivantes :

```
inportb(0x37A); // Pour la lecture
outportb (0x37A, DonneeAEcrire); // DonneeAEcrire est de type unsigned char
```

Par exemple outportb (0x37A,0x00) va mettre tous les bits à zéro.

Ecrire un programme qui force le bit /INIT à 0 sans modifier l'état des autres bits (cela provoque l'initialisation de l'imprimante).