

**Série de TD N° 8 : Electronique Numérique**

---

1. Donner le logigramme pour simuler la somme, sur 16 bits, des 100 premiers entiers naturels:

```
unsigned short int i, S=0 ;  
for ( i=0 ; i < 100 ; i++)  
S = S + i ;
```

2. La division binaire peut être effectuée par le principe des soustractions successives.

Exemple :  $A / B = 34 / 7$ ,  $Q(\text{quotient}) = 0$ ,  $R(\text{reste}) = 0$

$A - B = 34 - 7 = 27 \Rightarrow Q = 1$ ,  $A - B = 27 - 7 = 20 \Rightarrow Q = 2$ ,  $A - B = 20 - 7 = 13 \Rightarrow Q = 3$ ,

$A - B = 13 - 7 = 6 \Rightarrow Q = 4$ , (arrêt  $Q < B = 7$ )  $\Rightarrow Q(\text{quotient}) = 4$ , le dernier résultat est le  $R(\text{reste}) = 6$ . On remarque que  $A = 34$  à la 1<sup>ère</sup> itération,  $A = 27$  à la 2<sup>ème</sup> itération, ...etc.

En utilisant ce principe, donner le logigramme qui permet de matérialiser la division binaire sur 8 bits:  $A / B$ ,  $A > B$  et  $B \neq 0$ .

Composants à utiliser:

- à  $t = 0$ ,  $A$  est stockée préalablement dans un registre  $R$ , ensuite  $R$  sera utilisé pour le stockage de  $A - B$ ,
- Un additionneur pour effectuer «  $A - B$  »,
- Un compteur  $C$ ,
- Un comparateur pour contrôler l'arrêt du compteur  $C$ .

A la fin de l'opération le registre  $R$  représentera le reste et le compteur  $C$  indiquera le quotient. Indiquer le « Data bits » des composants.

Liaisons : une seule horloge  $CK$  et un minimum de portes logiques.

3. Traduire sous la forme d'un logigramme la séquence d'instructions, ci-dessous, qui représente le calcul factoriel d'un entier  $n$ .

```
Unsigned short int i, n=5, Fact=1;  
for (i = 1 ; i <=n; i++)  
Fact = Fact*i;  
printf(" %d ", Fact);
```

Composants à utiliser :

- à  $t = 0$ , «  $\text{Fact} = 1$  » est stockée préalablement dans un registre  $R$ , ensuite  $R$  stockera «  $\text{Fact} * i$  »,
- Un multiplieur pour effectuer «  $\text{Fact} * i$  »,
- Un compteur  $C$ , à  $t = 0 \rightarrow C = 1$ ,
- Un comparateur pour contrôler l'arrêt du compteur  $C$ .

A la fin de l'opération le registre  $R$  représentera le produit factoriel de  $n$ . Indiquer le « Data bits » des composants.

Liaisons : une seule entrée horloge  $CK$  et un minimum de portes logiques.

#### 4. Addition parallèle et série

Rappeler le logigramme de la somme  $S = A + B$  sur 8 bits en parallèle ensuite donner le logigramme qui effectue cette somme en série. Pour cela, on doit utiliser uniquement additionneur complet d'un bit, deux « shift register » de 8 bits pour le stockage des variables A et B, une bascule D pour le rebouclage « cout » vers « cin » et enfin une « clock : Horloge Ck » pour les registres à décalage et la bascule D. Le résultat de la somme doit être stocké dans B.