

Pour fabriquer le circuit qui sert l'affichage des chiffres de 0 à 9 en 7 segments, on a procédé de la manière suivante:

- On place quatre interrupteurs avec leurs étiquettes
- Câblage au cas par cas et utilisation de portes ET (exemple: 0 ET 1 ET 0 ET 0 pour 4)
- On place sept portes OU pour sept segments: s'il y a plus de cinq câbles, on place deux portes OU devant chaque porte OU
- Chaque chiffre a des segments en communs avec d'autres (exemple: 0 et 7 ont un segment A, celui du haut)
- On relie aux portails OU correspondants, on met des diodes et leurs étiquettes
- On peut ensuite utiliser tout ça comme sous-circuit

On utilise le sous-circuit dans un nouveau circuit qui servira comme modèle (test 7 segments).

#### Pour le demi-additionneur bit+bit et la retenue:

- On met deux interrupteurs (+étiquettes)
- On en ajoute un troisième (au cas où il y a 1+1 avec une retenue de 1)
- Au cas par cas, on câble et on utilise des portes ET et OU en suivant les règles de l'addition de deux bits:

$$0+0=0$$

$$0+1=1$$

$$1+0=1$$

$$1+1=0 \text{ retenue de } 1$$

$$1+1+(\text{retenue de } 1)=1 \text{ retenue de } 1$$

- On a les deux sorties, utilisation comme sous-circuit pour les additions

#### Au niveau de l'affichage de 4 bits (5 bits maximum pour le résultat):

- 5 entrées et 8 sorties (4 par 7 segments)
- On peut entrer jusqu'à 30 en binaire
- Utilisation d'un splitter
- On divise l'entrée par 10 (division euclidienne)
- On multiplie par 6 (décalage de 6)
- Somme du résultat précédent et de l'entrée

#### Addition 4 bits+4 bits:

On sait qu'il existe le module additionneur, mais ici on détaille le calcul en utilisant le sous-circuit bit+bit et retenue.

- On procède colonne par colonne
- Le bit 3 est initialisé avec la constante 0, après on lui relie à chaque fois la retenue (0 ou 1), et en dernier la retenue devient le bit de sortie tout à gauche
- Le modèle de test 7 segments est mis 6 fois, pour les trois nombres

#### Soustraction 4 bits+4 bits:

Il y a eu quelques problèmes pour pouvoir faire la soustraction détaillée, donc nous sommes passés par le soustracteur intégré.

### Add3 et BCD conversion affichage entier de 4 chiffres:

Initialement, il était prévu de continuer dans la logique de l'affichage 4 bits (.../10\*6\*...) sauf que ça ne marchait pas. Après plusieurs recherches nous avons trouvé la technique dite du décimal codé binaire (BCD, Binary Coded Decimal) qui permet de convertir un nombre binaire en une nouvelle version pour plusieurs affichages 7 segments.

Add3 est un convertisseur, qui suit cette table:

entrée:	sortie:
0000	0000
0001	0001
0010	0010
0011	0011
0100	0100
0101	1000
0110	1001
0111	1010
1000	1011
1001	1100

BCD utilise de nombreux sous-circuits add3 arrangés en escalier, pour avoir le résultat voulu.

### 4 chiffres + 4 chiffres et 4 chiffres – 4 chiffres

Suivent le même principe que pour 4 bits.