

Examen ARC 1 - Dur

Documents et calculatrices

Exercice 1 : Représentation des nombres

1. Donner la valeur décimale des entiers suivants, la base précisée.

- 1011011 et 101010 en binaire
- 1BE en hexadécimal
- 7210 en octal

2. Coder le nombre 21,37 en base 2 et 8.

3. Combien d'entiers positifs peut-on coder en binaire pour représenter 65 563 entiers différents en binaire ?

4. Soit un ordinateur dont les mots mémoire sont composés de mémoire. Un entier étant codé sur un mot, combien simultanément ? Quelle est la plus grande valeur mémoriser, cette valeur étant représentée par son code binaire ?

5. Coder en binaire sur un octet les entiers 105 et 21 puis les coder. Vérifier que le résultat sur un octet est correct.

6. Dans la représentation des nombres en virgule flottante IEEE 32 bits, chaque nombre est codé sur 4 octets, qui composent les trois champs suivants : le signe (S), l'exposant (E), et la mantisse (M). La relation suivante est vérifiée : $E = 2^M - 127$.

31	30	23	22	
S	E			0
M				

Reliez par des flèches les nombres à leur codage IEEE 32 bits

256	01000111100000000000000000000000
32768	00111111100000000000000000000000
-32	01000011111100000000000000000000
1.9375	11000000100000000000000000000000
-4	11000010000000000000000000000000

Exercice 2 : Représentation DCB

Le "Décimal Codé Binaire" (DCB) est un code de représentation des nombres dans le système décimale est codé sur un groupe de 4 bits.

Exemple : 1664 se code 0001 0110 0110 0100 en DCB

1. Comparaison des représentations DCB et binaire

2. Donner les représentations binaire et DCB des nombres 12 et 185.

3. Combien de nombres peut-on représenter sur 8 bits en binaire ? en DCB ?

4. Quel est l'avantage principal du codage DCB ? et son inconvénient majeur ?

2. Additionneur 1 chiffre DCB

On a à notre disposition un additionneur binaire 4 bits classique (du type 7483 TTL) avec retenues entrante et sortante, et les portes logiques de base. La retenue entrante est mise à 0. Le but est de construire un additionneur 1 chiffre DCB à partir de ces composants.

- Rappelez les règles d'addition des nombres en DCB.
- Illustrez ces règles en faisant les 3 additions suivantes : 6+2, 6+5 ; 8+8
- Donner l'équation de sortie d'un circuit qui va détecter quand la somme calculée par un additionneur binaire 4 bits est fautive en DCB. Ce circuit a comme entrées les 4 bits de la somme (S_3, S_2, S_1, S_0) et la retenue sortante (R_{out}) de l'additionneur. Sa sortie E vaut 1 s'il y a erreur et 0 sinon.

Vous commenterez par remplir la table de vérité de ce circuit, avant d'exprimer E.

R_{out}	S_3	S_2	S_1	S_0	E
0	0	0	0	0	0
.....					

- En vous servant du résultat de la question précédente, et en utilisant deux additionneurs 4 bits (dont les retenues de sorties sont R_{out1} et R_{out2}) et les composants logiques de base (ET, OU, ...) dessinez un additionneur DCB qui prend deux chiffres DCB en entrée et qui produit un chiffre DCB et une retenue R_{DCB} en sortie.

3. Compteur

On désire réaliser un circuit logique qui "compte" de 0 à 14 (et retour à 0) au format DCB.

- Donner la table de vérité

Q (mot de n bits)	$Q += D$ (mot de n bits)
.....

- Combien faudra-t-il de bascules D pour réaliser ce circuit ?

Exercice 5 : Additionneur série

On veut réaliser un additionneur série 8 bits. On suppose que le résultat est représentable sur 8 bits. Le résultat de l'opération doit être stocké afin d'être lu directement (lecture de tous les bits simultanément à la fin de l'addition). On dispose des composants suivants :

- Registres 8 bits : Entrée série et parallèle / Sortie parallèle (les registres fonctionnent avec une horloge active sur front montant, une entrée Load permet le chargement synchrone, une entrée Enable=0 valide le décalage série à chaque front d'horloge)
- Additionneur binaire 4 bits avec retenue d'entrée et de sortie
- Bascules D

1. Combien de composants de chaque sorte faut-il utiliser ?
2. Rappeler les équations de l'additionneur binaire complet 1 bit.
3. Réaliser le schéma de l'additionneur en représentant les différents composants correctement reliés entre eux, et en expliquant en 10 lignes maximum le fonctionnement de l'additionneur, lorsque l'on veut additionner deux nombres A ($A_7 \dots A_0$) et B ($B_7 \dots B_0$).

CHAP 10