

Exercice 1 – Changements de base

Donnez, dans les cases libres, la conversion des nombres proposés dans les autres bases de numération. Tous les nombres proposés sont des entiers positifs.

Base 2	Base 8	Base 10	Base 16
1001 0010	222	146	92
1 0011 0010	462	306	132
1010 1011	253	171	AB
1100 0010 0000	6 040	3 104	C20
1 1111 1111	777	511	1FF
1111	17	15	F

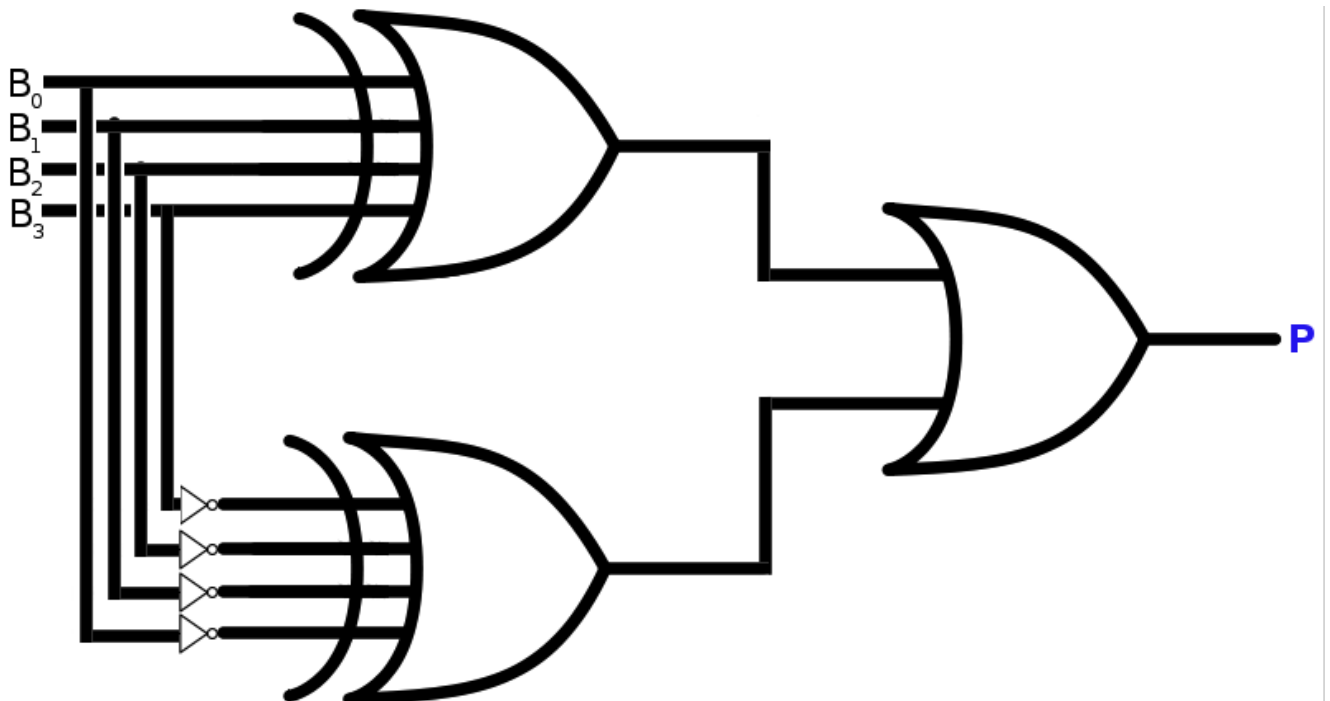
Exercice 2 – Logique

Nous cherchons à construire une fonction permettant d'ajouter automatiquement un bit de parité à un 'mot' mémoire de 4 bits. On notera le mot $B_0B_1B_2B_3$.

1. Ecrivez, en fonction de ces quatre valeurs d'entrée, la table de vérité de la fonction.

B₀	B₁	B₂	B₃	P
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

2. Dessinez le circuit associé.



Exercice 3 – Distance de Hamming

Soit le code acceptant l'ensemble de mots suivants : {0000000000, 0000011111, 1111100000, 1111111111}.

1. Quelle est la distance de Hamming de ce code ?

La distance de Hamming d'un code est la distance minimale entre deux mots valides du code, soit, ici, $d=5$.

2. Combien peut-il détecter d'erreurs ?

Il peut détecter $d-1$ erreurs, soit 4.

3. Combien peut-il en corriger ?

Il peut corriger au plus $(d-1)/2$ erreurs, soit, ici, 2 erreurs.

4. On reçoit le mot 0000011001. Quel mot va être considéré ?

Le mot le plus proche, soit 0000011111.