

Circuits et architectures logiques

Electronique numérique

2021 – durée 1h30 – Matthieu Denoual

CORRIGÉ

Consignes : les documents et calculatrices sont autorisés. Les réponses seront données sur ces feuilles à l'intérieur des espaces prévus à cet usage.

Partie 1 : Numération et codage [/5]

Exercice 1 [/2.5] : Convertissez en base 2 (**16 bits** virgule fixe **Q12,4**) le nombre : -621,17.
Expliquer votre démarche et calculez l'**erreur de représentation**.

nombre négatif, conversion en 3 étapes

1- On représente la valeur absolue du nombre soit 621,17

	<i>partie entière div. succ.</i>	<i>partie fractionnaire mult. succ.</i>						
$\begin{array}{r} 621 \quad \quad 16 \\ 13 \quad \quad 38 \quad 16 \\ \quad \quad \quad 6 \quad 2 \end{array}$	$621 \rightarrow 26D_h$ $621 \rightarrow 001001101101$	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: right;">0,17</td> <td style="text-align: left;">0,72</td> </tr> <tr> <td style="text-align: right;">x 16</td> <td style="text-align: left;">x 16</td> </tr> <tr style="border-top: 1px solid black;"> <td style="text-align: right;">2,72</td> <td style="text-align: left;">11,52</td> </tr> </table> <p style="text-align: right; margin-top: 10px;"> $0,17 \rightarrow 0,2B_h$ arrondi $0,17 \rightarrow 0,3_h$ $0,17 \rightarrow 0,0011$ </p>	0,17	0,72	x 16	x 16	2,72	11,52
0,17	0,72							
x 16	x 16							
2,72	11,52							

2- Complément à « 1 » de chacun des bits

0010 0110 1101 0011
1101 1001 0010 1100

3- On ajoute un bit à « 1 » tout à droite

1101 1001 0010 1101

+1

L'erreur est introduite lors de l'arrondi : $0,0011 \rightarrow 0,1875$

$$\text{erreur} = |0,1875 - 0,17| = 0,0175 < \text{erreur}_{\max} = q/2 = 2^{-5} = 0,03125$$

* les nombres binaires seront représentés en complément à deux sur **16 bits virgule fixe Q_{12,4}**.

Rappel représentation $Q_{m,k}$ sur N bits: $b_{m+k-1}b_{m+k-2} \dots b_k b_{k-1} \dots b_2 b_1 b_0$; $N = m+k$.

Exercice 2 [/2.5] : Codez la valeur A en format 32 bits virgule flottante selon la norme IEEE 754 rappelée en bas de page. $A = -40\,468,48$

$$A = (-1)^1 \times 1,235 \times 2^{15}$$

$$E - 127 = 15 \rightarrow E = 142 \rightarrow 1000\,1110$$

0,235	0,76	0,16	0,56	0,96	0,36
x 16	x 16	x 16	x 16	x 16	x 16
3,76	12,16	2,56	8,96	15,36	5,76
0011	1100	0010	1000	1111	0101
					011

	S	E7	E0	F22		F0
A	1 1000 1110 0011 1100 0010 1000 1111 011					

Rappel : représentation en virgule flottante suivant la norme IEEE 754. La valeur X est représentée suivant la forme : $X = (-1)^S \cdot 2^{E-127} \cdot 1, F$.

X s'écrit alors en binaire virgule flottante : $\underbrace{S}_{\text{signe}} \underbrace{e_7 e_6 \dots e_1 e_0}_{E} \underbrace{f_{22} f_{21} \dots f_2 f_1 f_0}_{F}$; E et F sont codés en binaire non signé.

Exercice bonus [+2] : À combien de décimales significatives correspond la valeur codée en format 32 bits virgule flottante selon la norme IEEE 754 pour la valeur π ? **Expliquez votre raisonnement.**

C'est la résolution du format virgule flottante (erreur de représentation) qui limite la résolution du nombre représenté.

L'erreur maximum de représentation est $q/2$, soit $\frac{2^{-23}}{2}$. L'erreur correspondant en décimal est $\frac{10^{-d}}{2}$, avec d le nombre de décimales.

erreur décimale > erreur binaire

$$\frac{10^{-d}}{2} > \frac{2^{-23}}{2} \cdot 2 \quad (1,F,2)$$

$$d < 6,6$$

Le format 32 bits virgule flottante conduit à des nombres avec 6 décimales significatives.

Par exemple pour π , on représente : $\left(\left\lceil a \left[\left(\frac{\pi}{2} - 1 \right) \times 2^{23} \right] / 2^{23} \right\rceil + 1 \right) \times 2 = 3,141592741 \dots$ au lieu de 3,14159265

$a[\]$: arrondi à l'unité

Partie 2 : synthèse combinatoire [/7]

Exercice 3 [/5] - Fonction coût minimal

L'objectif de cet exercice est la synthèse du circuit logique combinatoire de la fonction coût minimal à N entrée.

Fonction coût minimal: lorsque toutes les entrées sont à niveau bas '0', les sorties sont à niveau bas '0'. Lorsqu'une ou plusieurs entrées sont à niveau haut '1', la sortie correspondant à l'entrée de plus faible indice est à niveau haut '1'. Le tableau 1 présente l'exemple de la fonction minimum pour un circuit à 3 entrées.

Pour la combinaison 110 en entrée la fonction sort 010, car E_1 est l'entrée à '1' d'indice le plus petit.

Tableau 1: Table de vérité de la fonction minimum pour 3 entrées

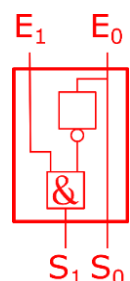
Entrées			Sorties		
E_2	E_1	E_0	S_2	S_1	S_0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	0	1
1	0	0	1	0	0
1	0	1	0	0	1
1	1	0	0	1	0
1	1	1	0	0	1

Question 3.1 [/3]

Établir les équations d'un circuit logique combinatoire pour la fonction coût minimal à 2 entrées. Implémenter le avec des portes ET, OU, NON.

La fonction à 2 entrées correspond à la table surlignée en haut pour E_1, E_0 en entrée et S_1, S_0 en sortie.

On voit directement : $S_0 = E_0$ et $S_1 = E_1 \cdot \overline{E_0}$



Question 3.2 [2]

Proposer une structure modulaire (illustration en Figure 1), du type de celle utilisée pour les additionneurs, permettant de généraliser le circuit de fonction minimale pour N entrées.

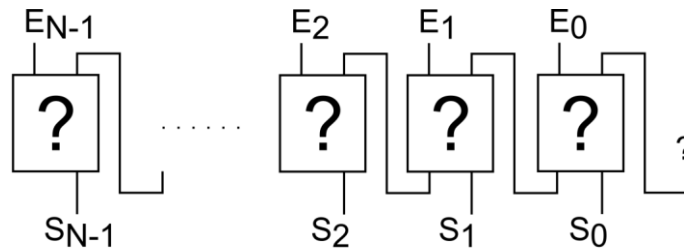


Figure 1 : Structure modulaire pour le circuit de fonction minimale.

On a un bit de rang minimal au rang N si on a un bit à 1 au rang N et qu'il n'y a eu aucun bit à 1 dans les rangs inférieurs (0 à N-1).

Quand un bit de rang minimal est généré, il doit bloquer toutes les générations de bits de rang minimal des rangs supérieurs.

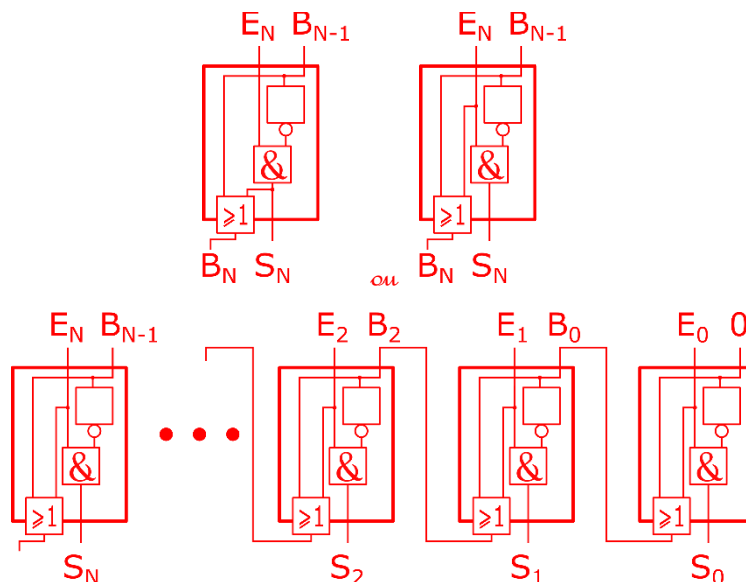
Bit de rang minimal si bit à 1 et pas de bit à 1 avant avec B_N une fonction logique active si un bit à 1 dans les rangs jusqu'au rang N :

$$S_N = \overline{B_{N-1}} \cdot E_N$$

B_N est active si bit minimal au rang N (S_N) ou s'il y a eu un bit minimal avant (B_{N-1}), la fonction B_N bloque la génération de bit minimal si elle est active :

$$B_N = B_{N-1} + S_N \text{ ou } B_N = B_{N-1} + E_N$$

Cellule élémentaire de la structure modulaire :



Exercice 4 [2] – Porte logique CMOS

Rappel: d'un point de vue physique, les portes logiques sont réalisées à partir de transistors. Plusieurs types de transistors peuvent être employés, mais le plus couramment utilisé est celui des transistors Métal Oxyde Semiconducteur plus connu sous le sigle transistor MOS. La fabrication des portes logiques s'appuie alors sur l'utilisation de deux types de transistors, les transistors NMOS et les transistors PMOS. Ces transistors sont utilisés comme des interrupteurs et chacun de ces transistors est caractérisé par les niveaux logiques qui le rendent assimilable à un interrupteur ouvert ou bien à un interrupteur fermé. La figure 2 reprend les configurations possibles.

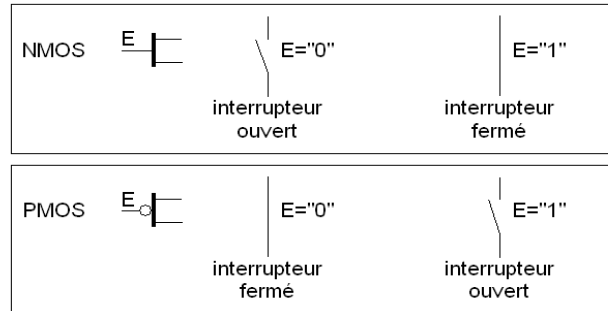
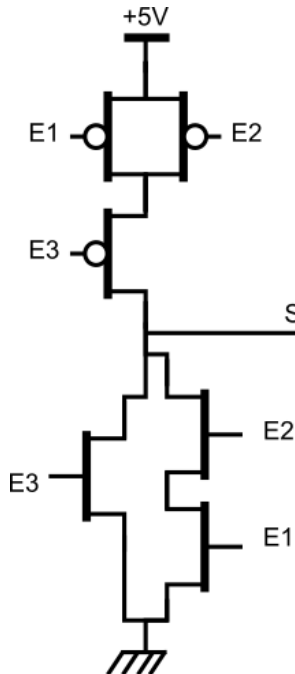


Figure 2: Modèles des transistors MOS (E entrée de commande)

À partir de ces informations, déterminez pour le schéma ci-après la fonction logique réalisée.



S Fonction logique à 3 entrées

E_3	E_2	E_1	S
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

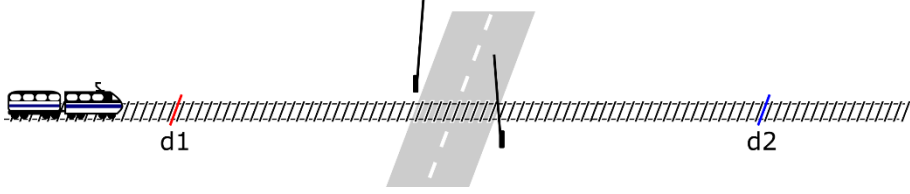
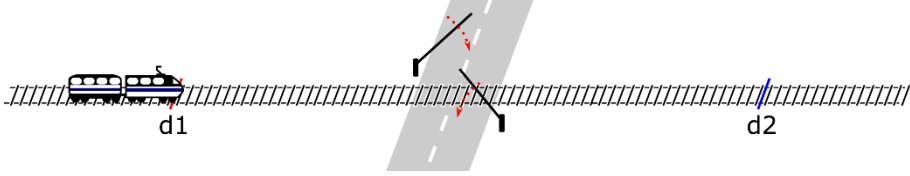
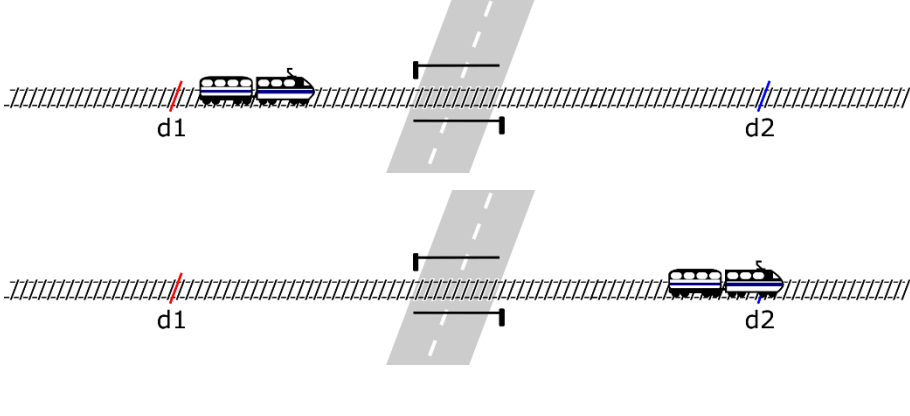
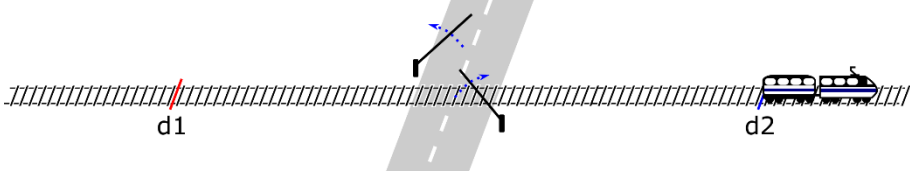
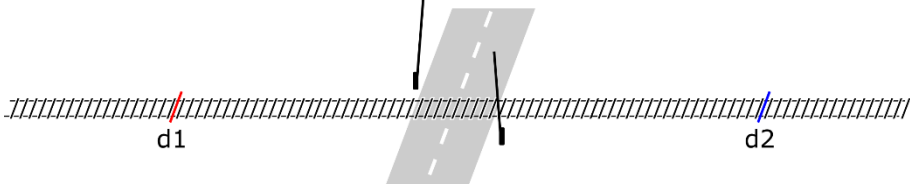
$S = \overline{E_3}(\overline{E_2} + \overline{E_1})$

Partie 3 : Synthèse logique séquentielle

Exercice 5 [7/8] Machine à états finis pour la gestion d'une barrière ferroviaire.

Le tableau 2 illustre le fonctionnement des barrières ferroviaires dont vous devez concevoir le système de contrôle sous forme de machine à états finis. On considère une direction unique pour le passage du train, qu'un seul train circule à la fois et que la longueur du train est plus petite que la distance entre les détecteurs utilisés pour activer les barrières.

Tableau 2 : Illustration du fonctionnement des barrières ferroviaires.

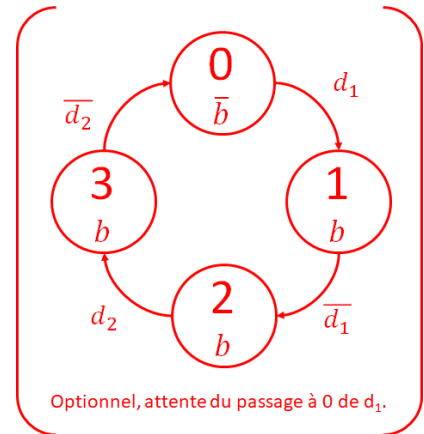
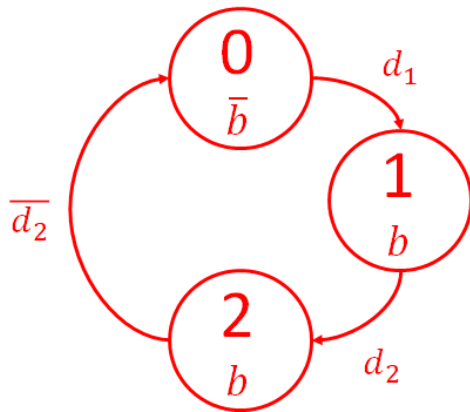
<p>Par défaut les barrières sont relevées.</p>	
<p>Lorsqu'un train est détecté (détecteur d1), les barrières s'abaissent.</p>	
<p>Les barrières sont baissées tant que le train n'a pas entièrement dépassé le détecteur d2.</p>	
<p>Lorsque le train a passé le détecteur d2, la commande pour relever les barrières est générée.</p>	
<p>Retour à la situation par défaut avec les barrières relevées.</p>	

Les réponses aux questions seront formulées sur les pages suivantes.

Question 5.1 : Traduisez le cahier des charges sous forme d'un diagramme d'état. Précisez si nécessaire les hypothèses ou conventions que vous adoptez. /2

Question 5.2 : Dessinez la structure globale de la machine à états finis de type Moore qui implémentera le diagramme d'état de la question 5.1. /2

Question 5.1 :



0 : état par défaut, barrières ouvertes pas de train.

1 : un train a été détecté par d_1 ($d_1=1$).

2 : le train est détecté par d_2 ($d_2=1$) mais n'a pas encore dépassé le détecteur. Quand l'arrière du train passe le détecteur ($d_2=0$), on repasse à l'état 0.

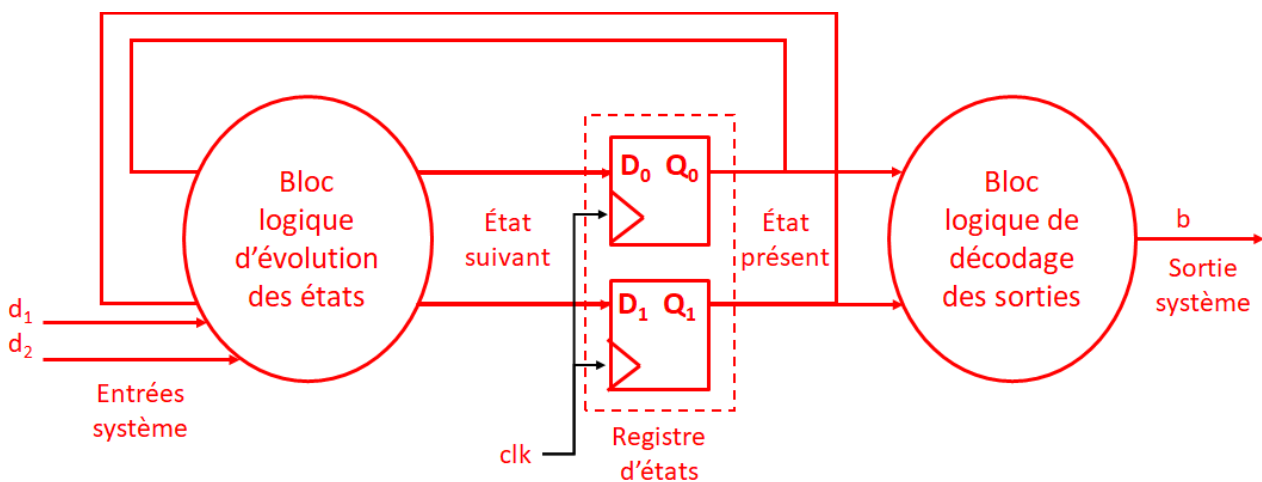
b : barrière fermée,

\bar{b} : barrière ouverte.

Question 5.2 :

3 états → 2 bascules

Machine d'état à 3 états, 2 entrées d_1 et d_2 et 1 sortie b .



Question 5.3 : Synthétisez la machine d'état. La synthèse ira jusqu'aux expressions algébriques. Veillez à décrire votre démarche et vos résultats intermédiaires. /4

Table de vérité du bloc logique d'évolution des états

	entrées systèmes		état présent			état suivant		
	d_1	d_2	q_1	q_0		D_1	D_0	
hypothèse	0	0	0	0	0	0	0	0
trains	1	X	0	0	0	1	0	1
suffisamment	X	0	0	1	1	1	0	1
espaces	X	1	0	1	1	2	1	0
pas 2	X	1	1	0	2	2	1	0
trains à ce	0	0	1	0	2	0	0	0
suivre.	X	X	1	1	3	0	0	0

Table de Karnaugh

D_1

$d_1 d_2$	$q_1 q_0$	00	01	11	10
00		0	0	0	0
01		0	1	0	1
11		0	0	0	0
10		0	0	0	0

$D_1 = \bar{d}_2 \cdot \bar{q}_1 \cdot q_0 + d_2 \cdot q_1 \cdot q_0$

$D_1 = d_2 \cdot q_0 + d_2 \cdot q_1 \cdot q_0$

D_0

$d_1 d_2$	$q_1 q_0$	00	01	11	10
00		0	0	0	0
01		0	0	0	0
11		0	0	0	0
10		1	1	0	0

$D_0 = \bar{d}_2 \cdot \bar{q}_1 \cdot q_0 + d_1 \cdot \bar{q}_1$

$D_0 = \bar{d}_2 \cdot q_0 + d_1 \cdot \bar{q}_1$

état 3 → état suivant dépend de d_2 si $d_2 = 10$ si $d_2 = 01$

Table de vérité du bloc logique de décodage des sorties

état présent	sortie
$Q_1 Q_0$	B
0 0	0 → barrière levée
0 1	1
1 0	1
1 1	3 → barrière baissée sécurité ou avant retour 0.

$b = Q_1 + Q_0$