

Exemples de questions d'examens partiels passés

- Note 1 : on donne en italique des éléments de réponses à **certaines** questions posées.
- Note 2 : la matière couverte dans les examens futurs n'est pas nécessairement la même que lors des examens passés.
- Note 3 : le nombre/type de questions des examens futurs peut changer sans préavis.
- Note 4 : d'autres approches de résolution sont possibles.
- Note 5 : l'information fournie ici l'est **sans garantie aucune** : "à vos risques"

Exemple de questions Examen Partiel 1 -1997

Q1: Convertir le nombre binaire 1101101.11 en décimal, octal, haxadécimal

Réponse : 109.75 décimal, 155.6 octal, 6D.C hex.

Q2: Convertir le nombre décimal 250.72 en binaire (se limiter à 4 chiffres après point):

Réponse : 11111010.1011

Q3: Additionner le nb suivants en binaire complément 2 sur 5 bits. Commentez le résultat : -10 et -8

Réponse 01110 et il y a dépassement car -18 ne s'exprime pas en 5 bits signés.

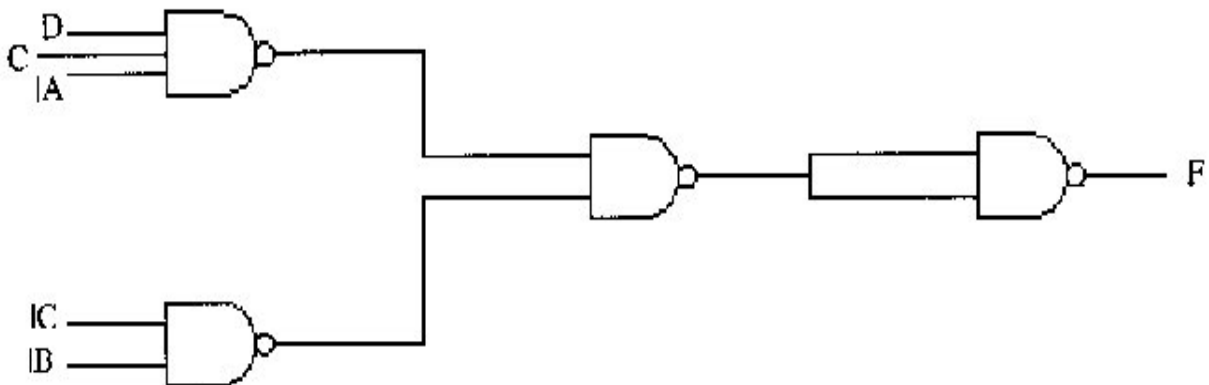
Q4: En employant la technique des tables de Karnaugh, trouvez l'expression POS minimum pour: $F(x,y,z) = (x+y)(x+|y+z)$. Note $|y = y'$ (y barre)

Réponse: $F = (x + z)(x + y)$

Q5: Soit la fonction $F(D,C,B,A) = \text{Somme des minterms } (2,4,5,6,10,13) + X(3,7,9,11,12,15)$
Avec la technique des table de Karnaugh, trouver l'expression POS minimum et réalisez son implantation en NANDs. Tracez le circuit. Justifiez votre réponse

Réponse: $F = (/D + /C + A)(C + B)$

Implantation en NANDs:



Q6: Implantez la fonction $F(D,C,B,A) = \text{Somme des minterms } (2,4,6,13,14) + X(3,7,9,11,12,15)$ dans un Mux 4 à 1. Les variables du Mux seront D et C

Réponse: non disponible

Q7: Implantez la fonction $F(D,C,B,A) = \overline{D}B + DC|A + |D|CB$ dans un nombre minimum de PALs 3 entrées/3 sorties.

Réponse 2 PALs suffisent avec du Retournement. Posons $X = \overline{D}B$ et $Y = DC$ alors:

$$F = X + Y|A + X|C$$

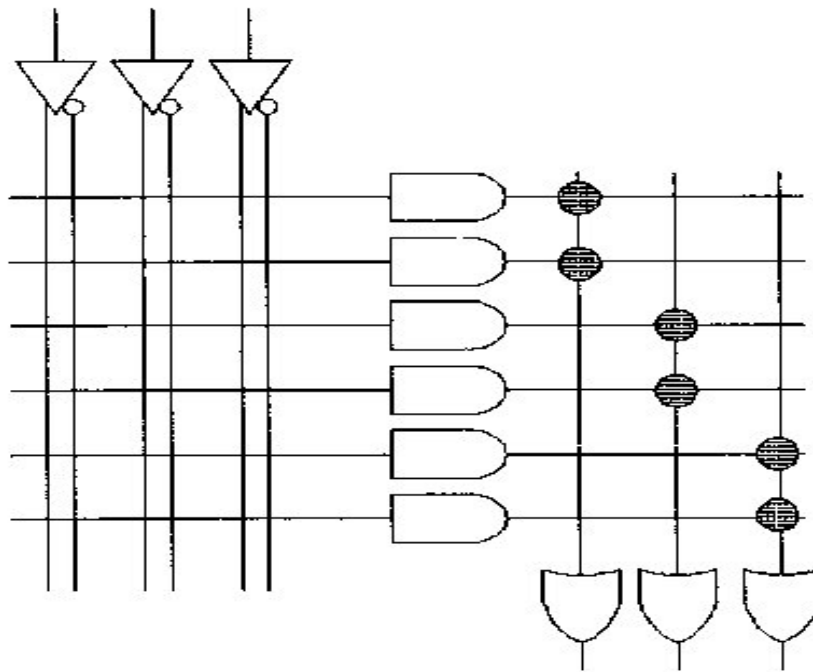
$$F = X(1 + |C) + Y|A$$

$$F = X + Y|A$$

PAL 1 : on synthétise X, Y (entrées du PAL : D, C, B)

PAL 2: on synthétise F (entrées du PAL : X, Y, A)

Chaque PAL réalise la somme de 2 produits:



Exemple de questions Examen Partiel 2 - 1997

Q1: a) Nommez un problème essentiel des latches R-S (répondre en 1 ligne max)?

Élément de réponse: État '11' interdit car alors $Q = \overline{Q}$ par exemple.

b) Nommez une raison justifiant la très grande popularité des flaps flops D ?

Élément de réponse: Plus simple à utiliser/comprendre.

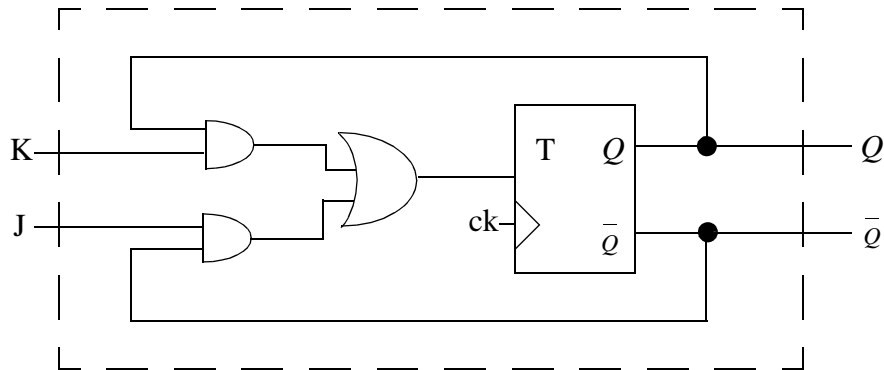
c) Nommez une différence essentielle entre un latch et un flip flop

Latch: sorties suivent entrées si horloge à 1 (level-sensitive clock)

Élément de réponse: FF: changement aux sorties seulement lors des transitions d'horloge.

Q2: Rajouter la logique combinatoire nécessaire et minimale à un flip flop T pour réaliser un flip flop J-K. Justifier votre démarche et dessinez le plan de votre circuit.

Élément de réponse:



Q3: Réaliser la séquence suivante en employant un registre à décalage que vous construirez avec trois flaps flops D individuels.

* La séquence se répètera indéfiniment: 101 -> 010 ...

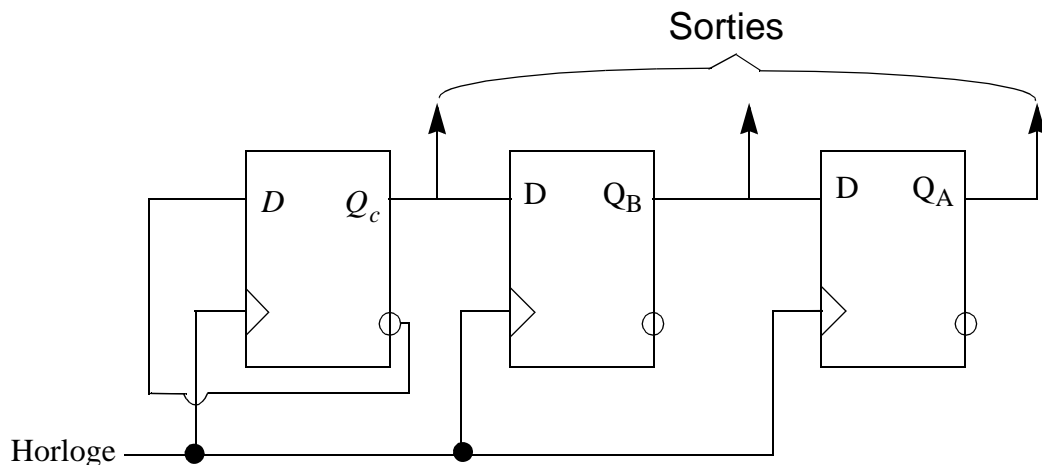
* Votre séquenceur doit être auto-correcteur.

* Faites le diagramme d'état complet de votre design.

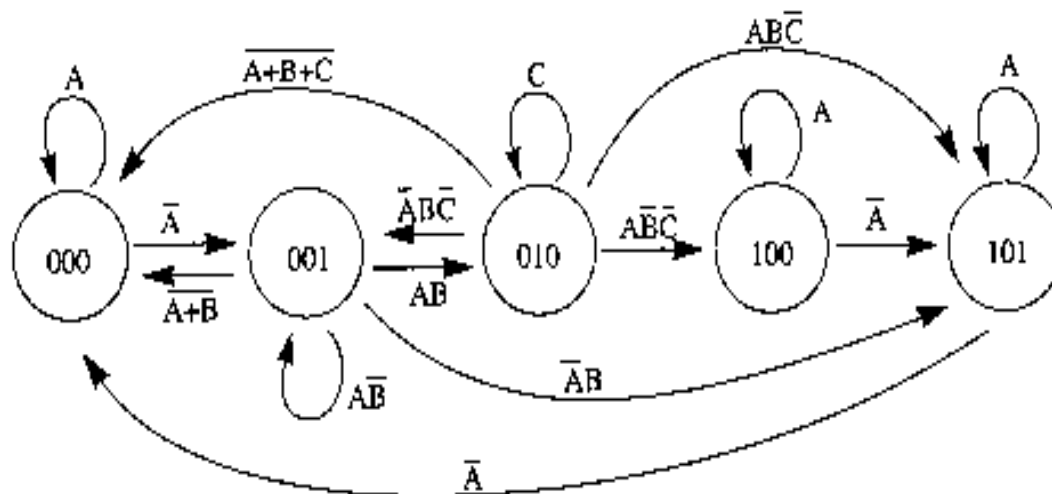
* Dessinez le plan de votre circuit.

* Justifier votre démarche.

Élément de réponse:



Q4 : Soit le diagramme d'état suivant : (A,B,C, sont des signaux externes):



Établissez la table état présent - état futur (present state - next state) en employant la méthode des variables conditionnelles. Justifier votre démarche. Simplifiez vos expressions. Nommez les 3 bits Qc Qb Qa (avec Qc = MSB).

Élément de réponse:

Table PS -> NS

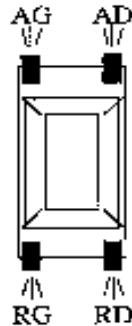
État	PS Present State			NS Next State		
	Q _C	Q _B	Q _A	Q _C	Q _B	Q _A
0	0	0	0	0	0	\bar{A}
1	0	0	1	$\bar{A}B$	AB	$A \oplus B$
2	0	1	0	$A\bar{C}$	C	$\bar{B}C$
3	1	0	0	1	0	\bar{A}
4	1	0	1	A	0	A

Q5: Séoul (Corée, mars 1986). Vous êtes ingénieur chez Hyundai Motor et votre tâche consiste à réaliser le séquenceur des clignotants et feux de détresse (hasard) de la nouvelle PONY 1987 qui sera bientôt vendue au Québec! Les signaux d'entrées auxquels vous avez accès sont les suivants:

- * **CG** - si CG = 1 -> clignote à gauche
- * **CD** - si CD = 1 -> clignote à droite
- * **HA** - si HA = 1 -> feux de détresse (hasard)
- * Une **Horloge de 1 Hz**

La PONY 1987 a 4 ampoules dédiées aux clignotants et feux de détresse :

* **AG** (avant gauche), **AD** (avant droit), **RG** (arrière gauche) et **RD** (arrière droit)
(voir dessin ci-contre)



Les séquences de clignotage sont les suivantes:

- *clignote à gauche : AG et RG: allumés (1 s), éteints (1 s), allumés (1 s),...
- *clignote à droite : AD et RD: allumés (1 s), éteints (1 s), allumés (1 s), ...
- * feux de détresse (hasard) : AG, RG, AD, RD: allumés (1 s), éteints (1 s), allumés (1 s),...

On vous demande de réaliser ce séquenceur (sur 2 bits) pour réaliser les fonctions de : clignote à gauche, clignote à droite et de feux de détresse (hasard) en employant des flaps flops J-K très populaire en Corée !

- * Faites le diagramme d'état complet de votre design
- * Justifier votre démarche
- * Dessinez le plan de votre circuit **en incluant** les sorties pour connexions aux 4 ampoules, on emploiera des LEDs pour fins de démonstration.

Votre design est magnifique, vous avez hâte d'essayer la nouvelle PONY 87!
Élément de réponse :

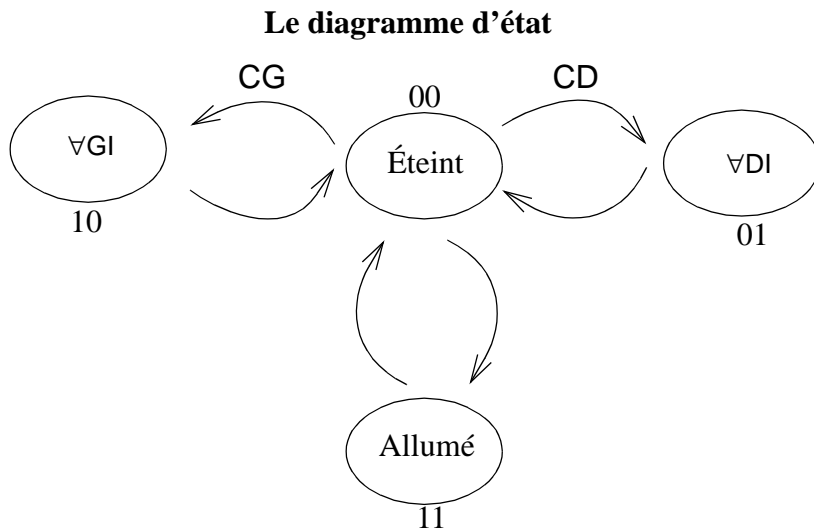
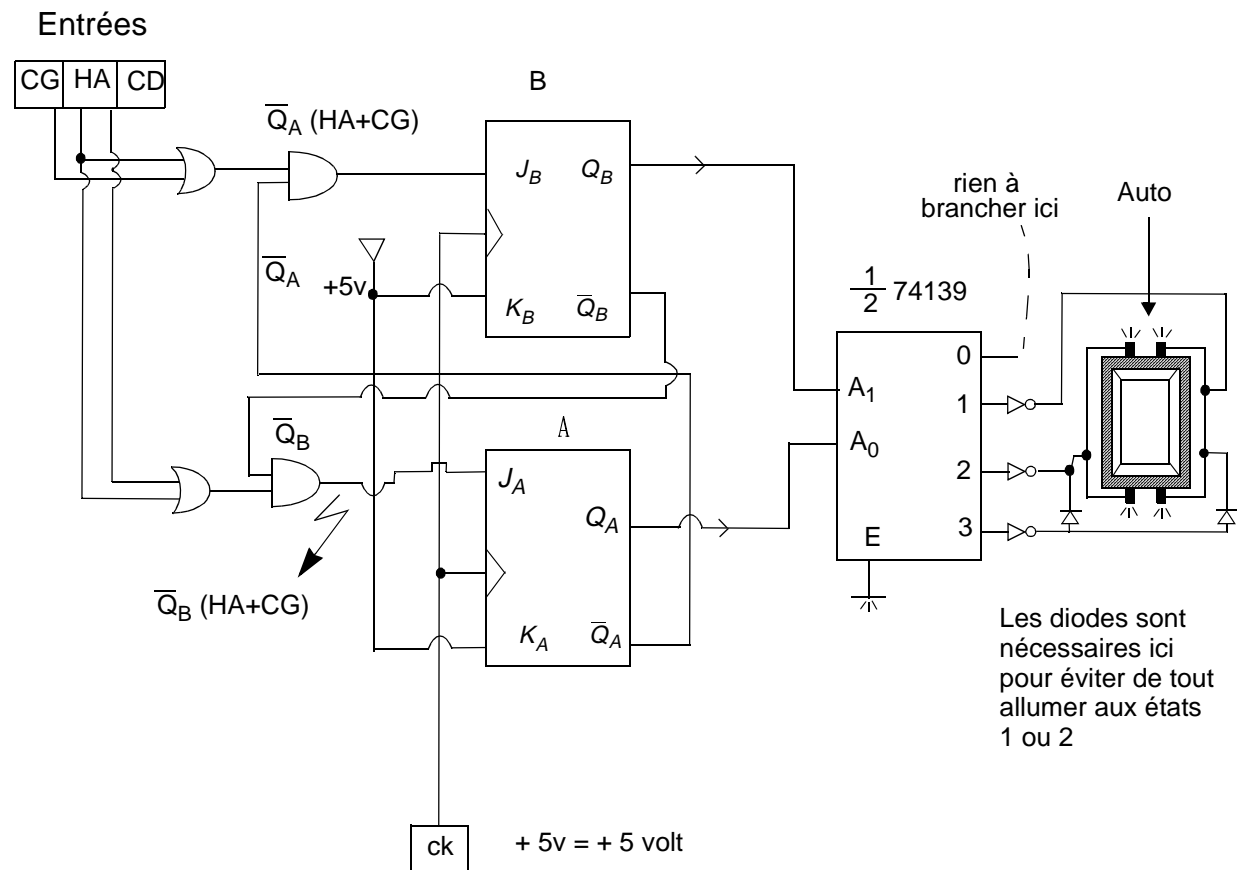


Table État Présent - État futur avec variables conditionnelles

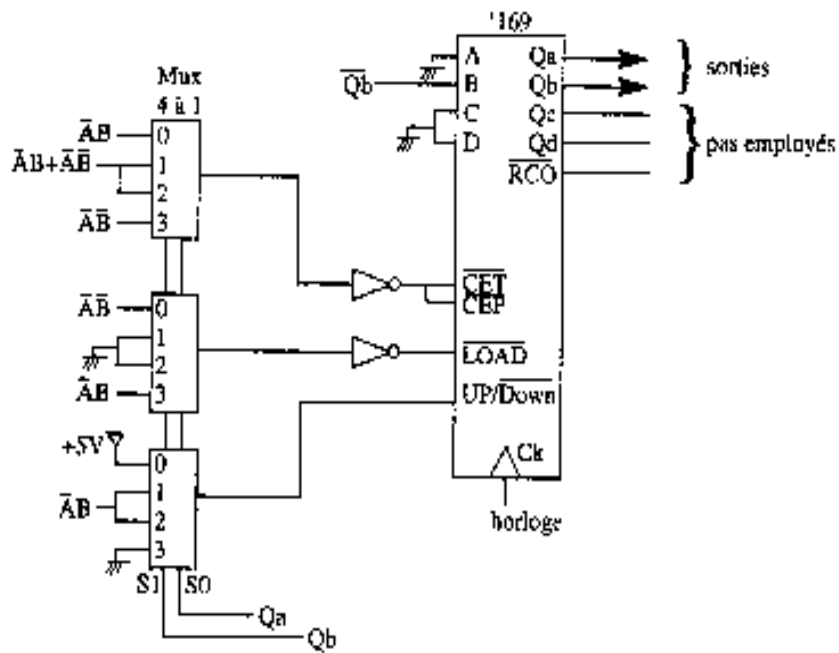
Indice	P	S	N	S	J _B	K _B	J _A	K _A
0	0	0	(CG)	(CD+HA)	(HA+CG)	X	(CD+HA)	X
1	0	1	0	0	0	X	X	1
2	1	0	0	0	X	1	0	X
3	1	1	0	0	X	1	X	1
	Q _B	Q _A	Q _B	Q _A				

Pour état 00 : (HA + CG), (CD + HA)

Schéma complet :



Q6: Séour (Corée, novembre 1998). Vous êtes maintenant ingénieur d'expérience (senior engineer) chez Hyundai Motor. vous savez que le *reverse engineering* est un outil d'espionnage puissant! vos contacts à détroit (USA) viennent de vous faire parvenir le schéma suivant (où A et B sont des signaux d'entrées externes):

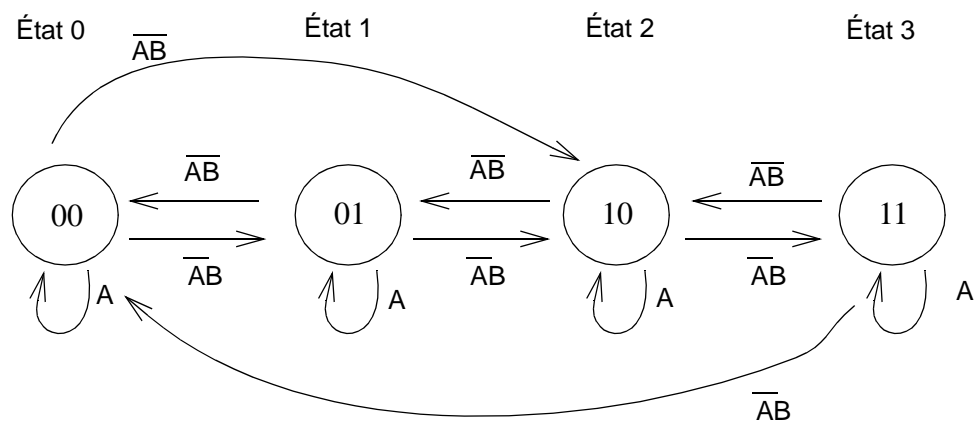


Vous déterminez le diagramme d'état correspondant à ce circuit car vous voudriez vous en servir pour le nouveau modèle de PONY... 1990!

* On vous demande donc de déterminer le diagramme d'état de ce circuit.

*Justifier votre démarche.

Élément de réponse :



Exercice supplémentaire résolu

Faire le design d'une machine distributrice de jus (1\$) et de chip (0.75\$). La machine n'accepte que les 25 cents et les 1\$ (huard). On emploiera un 74169. La machine fonctionne comme suit :

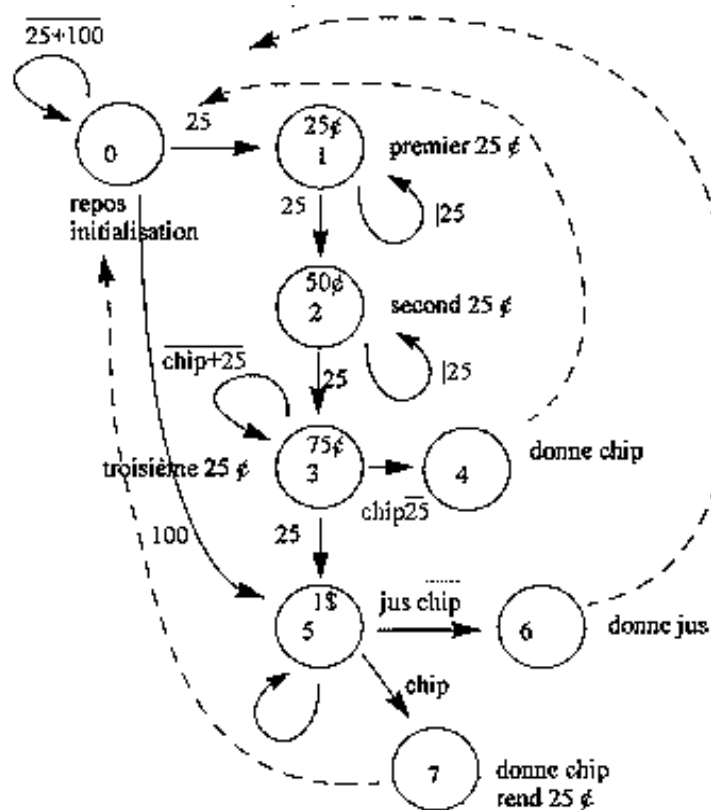
- 1- le client spécifie son choix (1 jus ou 1 chip)
- 2- le client introduit la monnaie.
- 3- si le client fournit 1 \$ pour le chip, on lui rend 0.25\$

Les signaux disponibles sont les suivants :

en entrée	remarques	en sortie
25	on introduit 0.25 \$	rend 0.25 \$
100	on introduit 1 \$	donne jus
chip	on veut 1 sac de chips	donne chip
jus	on veut 1 jus	

Solution

À partir de l'énoncé, on établit le diagramme d'état suivant :



On aura 3 FF ici car il y a 8 états possibles de #0 à #7 (000 à 111)

Les direction d'état sont établies comme suit :

*état #0

Valeur des variables		Expressions booléennes pour les directions possibles			Remarque
100	25	25	100	$25 + 100$	
0	0	0	0	1	On reste sur place # 0
0	1	1	0	0	On saute à l'état 2 (25¢ entré)
1	0	0	1	0	On saute à l'état 5 (1\$ entré)
1	1	Impossible ici		0	

Note : il faut toujours que les directions possibles soient mutuellement exclusives et couvrent toutes les possibilités.

* état #3

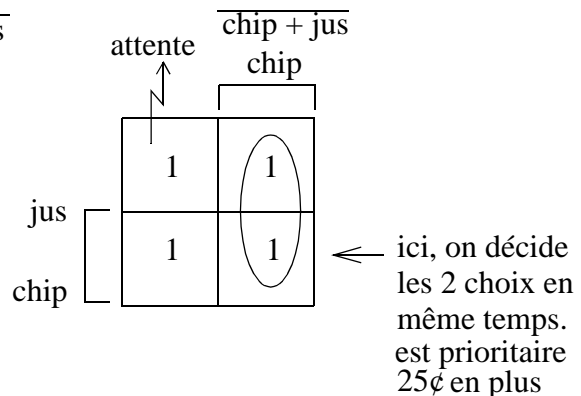
Valeur des variables		Expressions booléennes pour les directions possibles			
25	chip	25	Chip	$\overline{25}$ Chip + 25	
0	0	0	0	1	<div> <div> <div>chip25</div> <div>chip</div> </div> <div> <div>1</div> <div>1</div> </div> <div> <div>1</div> <div>1</div> </div> </div> <div> <div>on reste sur place # 3</div> <div>on va à # 4</div> <div>on va à # 5</div> </div> <div> <div>25</div> <div>remarque</div> </div>
0	1	0	1	0	
1	0	1	0	0	
1	1	1	0	0	

* état #5

On établit :

Variables				
Jus	Chip	Chip	Jus Chip	$\overline{\text{Chip} + \text{jus}}$
0	0	0	0	1
0	1	1	0	0
1	0	0	1	0
1	1	1	0	0

les 3 cas couvrent toutes les possibilités et sont mutuellement exclusives



Dressons la Table PS -> NS (present state à next state)

	PS			NS			Remarque
	Q _C	Q _B	Q _A	Q _C	Q _B	Q _A	
0-	0	0	0	100	0	25 + 100	
1-	0	0	1	0	25	$\overline{25}$	Q _C Q _B Q _A = D _C D _B D _A
2-	0	1	0	0	1	25	FF "D" ici
3-	0	1	1	Chip + 25	$\overline{\text{Chip} + 25}$	$\overline{\text{Chip} + 25}$	
4-	1	0	0	0	0	0	Donne chip
5-	1	0	1	1	Chip + jus	$\overline{\text{Jus} + \text{chip}}$	
6-	1	1	0	0	0	0	Donne jus
7-	1	1	1	0	0	0	Donne chip, remet 25¢

Q_CQ_BQ_A

√ toujours 0

* État 0 : États possibles ici :

0	$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$	0	-> $\overline{25 + 100}$
0	$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$	1	-> si 25
1	$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$	1	-> si 100

(100) (0) (25+100)

* État 1 : états possibles ici :

$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	0	1	-> si $\overline{25}$
$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	1	0	-> si 25

(0) (25) ($\overline{25}$)

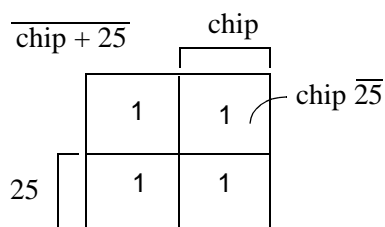
* État 2 : États possibles ici :

$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	0	-> si $\overline{25}$
$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	1	-> si 25

Note : $25 + \overline{25} = 1$

(0) (1) (25)

* État 3 : États possible ici :



0	1	1	-> $\overline{\text{chip} + 25}$, i.e. chip=25=i
1	0	1	-> 25, i.e. 25 = "1"
1	0	0	-> chip $\overline{25}$, i.e. chip = "1" et 25

(25 + chip $\overline{25}$) $[(\overline{\text{chip} + 25}) + 25] = \overline{\text{chip} + 25}$

↓

$\overline{\text{chip} + 25}$

* État 4 : États possibles : 000

* État 5 : États possibles

		chip	
		1	1
jus		1	1

1	0	1	->	$\overline{chip + jus} \cdot$, i.e. chip = jus = 0
1	1	1	->	$chip \cdot$, i.e. chip = "1"
1	1	0	->	$jus \overline{chip} \cdot$, i.e. jus = "1", chip = 0

(1) ↓

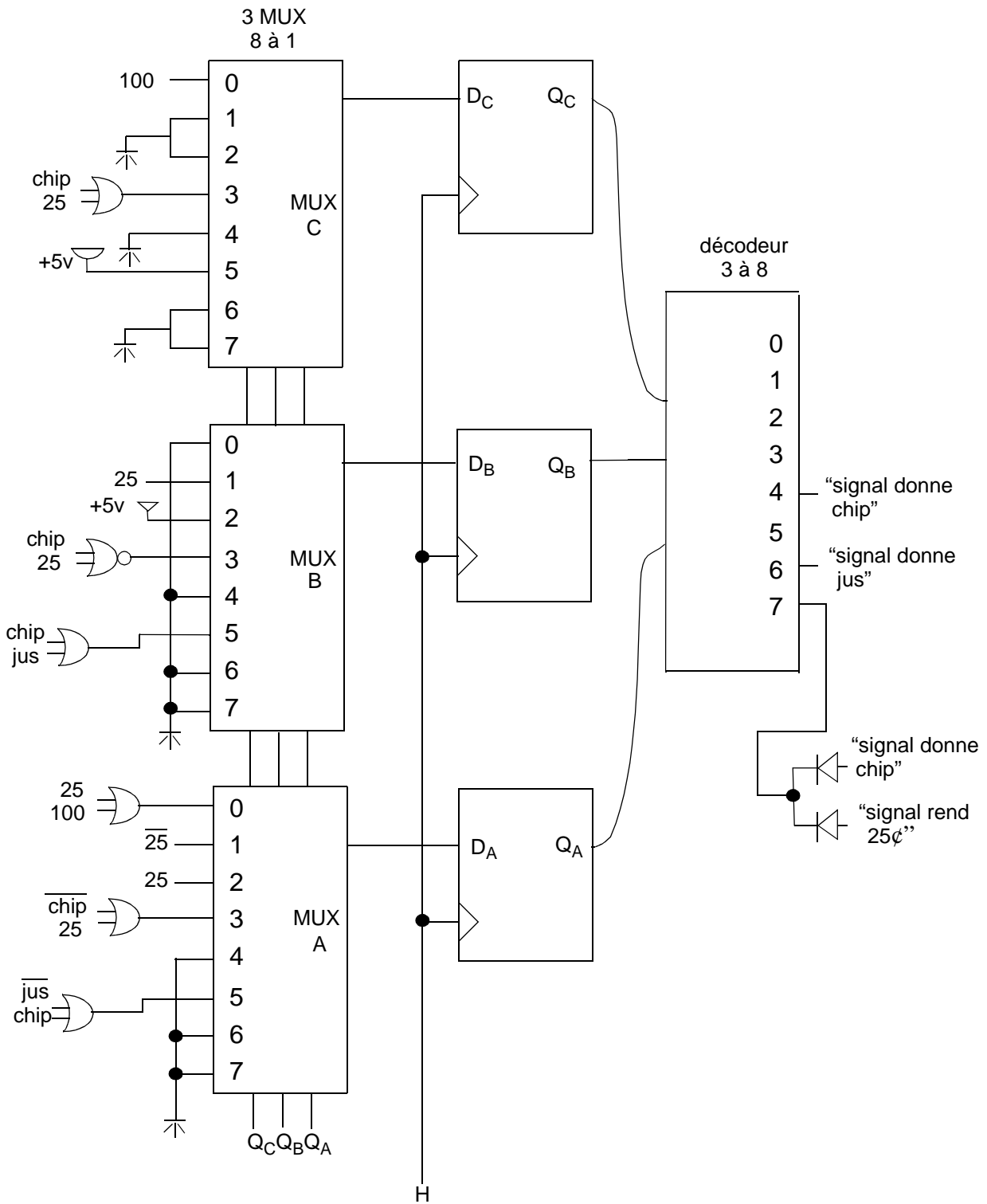
$(chip + \overline{chip} + jus) = \overline{jus} + chip$

$chip + jus \overline{chip} = chip + jus$

* État 6 : États possibles : 000

* État 7 : États possibles : 000

Réalisation avec 3 multiplexeurs 3 à 8 :

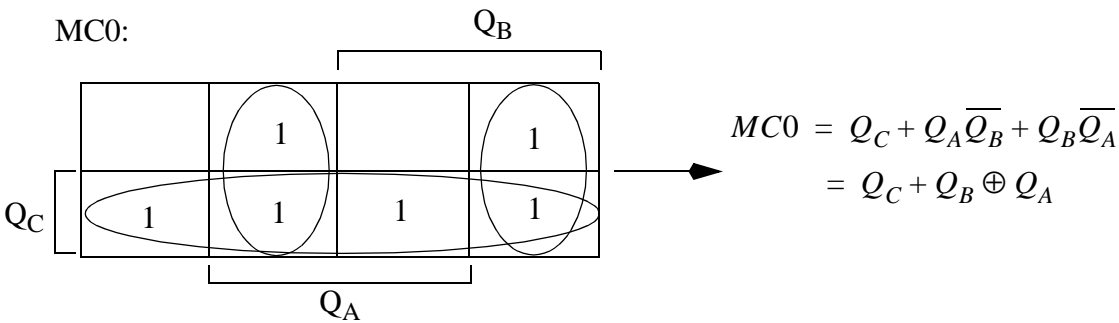
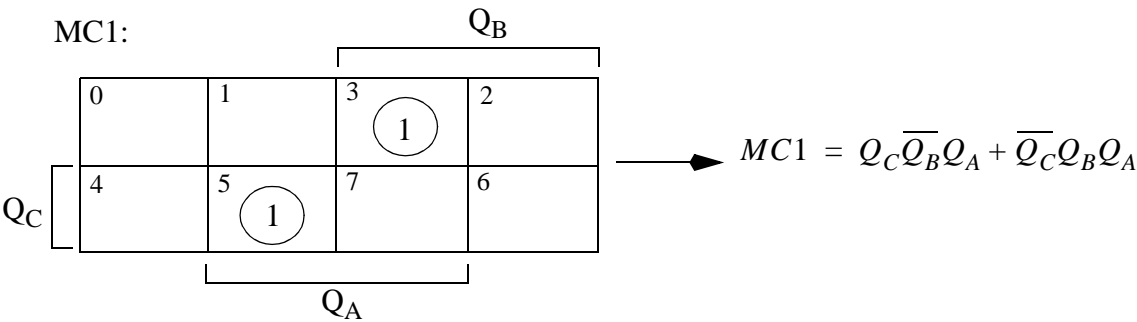


MUX C 4 entrées différentes => simplifiable

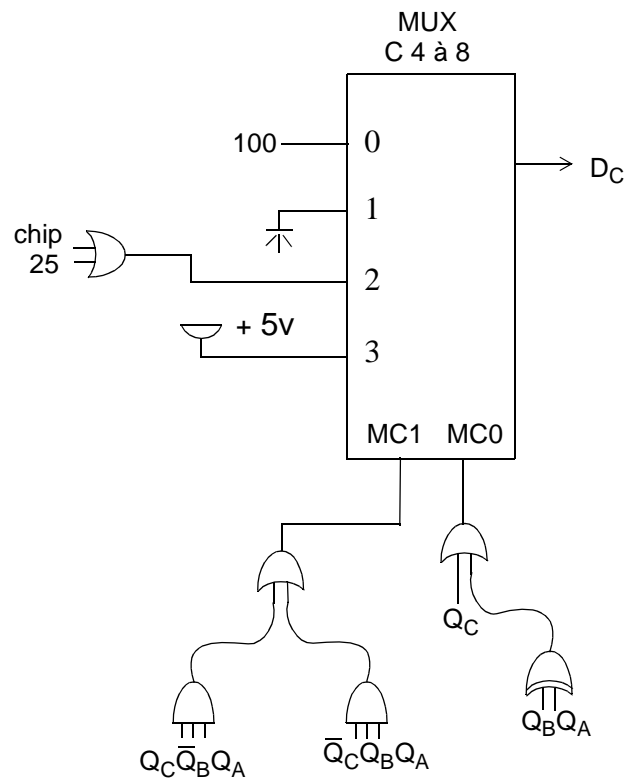
Simplifions MUX C avec MUX 4 à 1 plutôt :

- MUX C 4 entrées différentes => simplifiable
- MUX B 5 entrées différentes => non-simplifiable
- MUX A 6 entrées différentes => non-simplifiable

État	Q _C	Nouvel état	M _C	M _C
0	100	0	0	0
1	0	1	0	1
2	0	1	0	1
3	Chip + 25	2	1	0
4	0	1	0	1
5	1	3	1	1
6	0	1	0	1
7	0	1	0	1
QcQbQa				



Le MUX C (8 à 1) est remplacé par un MUX 4 à 1 :



Bon succès à tous et à toutes !