Cours 2 : Boole, portes logiques...

En ce cours, on apprendra six choses :

- Tables de vérité
- Portes logiques
- Algèbre de Boole
- PdS, SdP
- Diagrammes temporelles
- Bascule et Bistable

N'inquietez-vous pas. Seulement l'algèbre est un peu compliqué.

Tables de vérité

On utilise des tables de vérité pour décrire des fonctions logiques :

variables fonctions

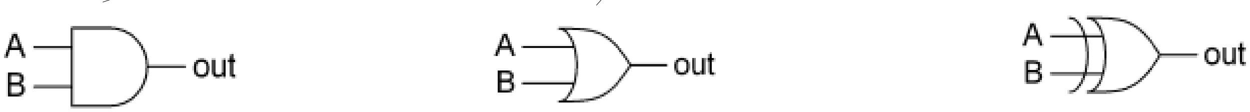
a	b	$a \cdot b$	$a+b$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Ça marche bien jusqu'à 3 ou 4 variables. Plus que ça, une page est vite remplie !

Portes logiques (positives)

On vient juste de voir ET et OU. L'autre est OU-EXCLUSIF :

a	b	a·b	a+b	a⊕b
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

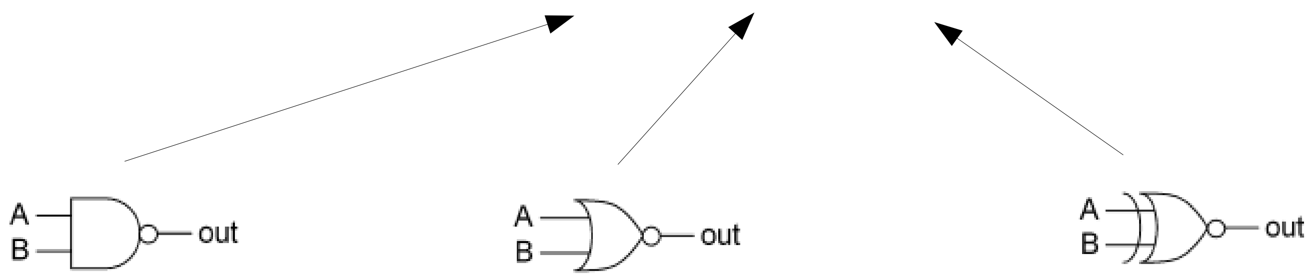


ET OU OUX

Portes logiques (négatives)

Maintenant, les mêmes portes à l'inverse :

a	b	$\overline{a \cdot b}$	$\overline{a + b}$	$\overline{a \oplus b}$
0	0	1	1	1
0	1	1	0	0
1	0	1	0	0
1	1	0	0	1



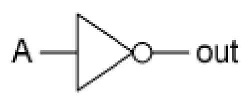
NET NOU NOUX

(La boule indique inversion)

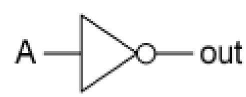
Oops, on a oublié une porte !

a	b	$\overline{a}$	$\overline{b}$
0	0	1	1
0	1	1	0
1	0	0	1
1	1	0	0

(OOPS = Inverseur)



OOPS



OOPS

Mais Jeff, on a vu seulement 6 portes à deux entrées, mais il-y-a 2^4 = 16 combinaisons possibles :

Quelles combinaisons ne sont pas couverts ?

Algèbre de (George) Boole

Bon, on connait toutes les portes, mais quoi faire avec ? On a besoin d'un système formel de logique binaire : c'est l'algèbre de Boole. Voilà les axiomes :

$$0 \cdot 0 = 0$$
$$1 \cdot 1 = 1$$

$$0 + 0 = 0$$
$$1 + 1 = 1$$

$$0 \cdot 1 = 1 \cdot 0 = 0$$
$$1 + 0 = 0 + 1 = 1$$

Si $x = 0$, d'abord $!x = 1$

Si $x = 1$, d'abord $!x = 0$

Ces axiomes sont simples, on vient de les voir !

Théorèmes (équations à une variable)

OK, il faut penser un peu pour ces théorèmes, mais ils ne sont pas difficiles non plus :

(N.B. : x est une variable)

$$x \cdot 0 = 0$$
$$x \cdot 1 = x$$

$$x + 0 = x$$
$$x + 1 = 1$$

$$x \cdot x = x$$
$$x + x = x$$

$$x \cdot !x = 0$$
$$x + !x = 1$$

$$!(!x) = x$$

Il faut être capable de faire ces simplifications sans penser...

Propriétés (avec leurs propres titres)

Réveillez-vous ! Il faut penser un peu :

$x \cdot y = y \cdot x$ $x + y = y + x$	→ *Commutativité*
$x \cdot (y \cdot z) = (x \cdot y) \cdot z$ $x + (y + z) = (x + y) + z$	→ *Associativité*
$x \cdot (y + z) = x \cdot y + x \cdot z$ $x + y \cdot z = (x + y) \cdot (x + z)$	→ *Distributivité*
$x + x \cdot y = x$ $x \cdot (x + y) = x$	→ *Absorption*
$x \cdot y + x \cdot !y = x$ $(x + y) \cdot (x + !y) = x$	→ *Combinaison*
$x + !x \cdot y = x + y$ $x \cdot (!x + y) = x \cdot y$	→ *Yada, yada, yada*

(N.B. : ET a précédence sur OU)

Théorème DeMorgan

Réveillez-vous ! <u>ÇA C'EST TOUJOURS SUR L'EXAMEN</u> !!!

$$\overline{x \cdot y} = \overline{x} + \overline{y}$$

$$\overline{x + y} = \overline{x} \cdot \overline{y}$$

Ça nous laisse changer façilement entre PdS et SdP (à venir).

Minterms n'Maxterms

a = Jeff boit de l'Alcool, b = Jeff mange de la Bouffe
c = Jeff fume une Cigarette

a	b	c	Minterm	Maxterm
0	0	0	$m_0 = !a \cdot !b \cdot !c$	$M_0 = a + b + c$
0	0	1	$m_1 = !a \cdot !b \cdot c$	$M_1 = a + b + !c$
0	1	0	$m_2 = !a \cdot b \cdot !c$	$M_2 = a + !b + c$
0	1	1	$m_3 = !a \cdot b \cdot c$	$M_3 = a + !b + !c$
1	0	0	$m_4 = a \cdot !b \cdot !c$	$M_4 = !a + b + c$
1	0	1	$m_5 = a \cdot !b \cdot c$	$M_5 = !a + b + !c$
1	1	0	$m_6 = a \cdot b \cdot !c$	$M_6 = !a + !b + c$
1	1	1	$m_7 = a \cdot b \cdot c$	$M_7 = !a + !b + !c$

Étonnant !?! Ça sert à quelque chose ? ->

Somme de Produits

a = Jeff boit de l'Alcool, b = Jeff mange de la Bouffe
c = Jeff fume une Cigarette, M = Jeff devient Malade :-(

a	b	c	M	Minterm
0	0	0	**1**	$m_0 = !a \cdot !b \cdot !c$
0	0	1	**1**	$m_1 = !a \cdot !b \cdot c$
0	1	0	0	$\cancel{m_2 = !a \cdot b \cdot !c}$
0	1	1	**1**	$m_3 = !a \cdot b \cdot c$
1	0	0	0	$\cancel{m_4 = a \cdot !b \cdot !c}$
1	0	1	0	$\cancel{m_5 = a \cdot !b \cdot c}$
1	1	0	0	$\cancel{m_6 = a \cdot b \cdot !c}$
1	1	1	**1**	$m_7 = a \cdot b \cdot c$

Jeff devient Malade = $m_0 + m_1 + m_3 + m_7$

(N.B. Les valeurs de la colonne M ont été <u>choisies</u> par Jeff)

Produit de Sommes

a = Jeff boit de l'Alcool, b = Jeff mange de la Bouffe
c = Jeff fume une Cigarette, M = Jeff devient Malade

a	b	c	M	Maxterm
0	0	0	1	$\cancel{M_0 = a + b + c}$
0	0	1	1	$\cancel{M_1 = a + b + !c}$
0	1	0	**0**	$M_2 = a + !b + c$
0	1	1	1	$\cancel{M_3 = a + !b + !c}$
1	0	0	**0**	$M_4 = !a + b + c$
1	0	1	**0**	$M_5 = !a + b + !c$
1	1	0	**0**	$M_6 = !a + !b + c$
1	1	1	1	$\cancel{M_7 = !a + !b + !c}$

Jeff devient Malade = $M_2 \cdot M_4 \cdot M_5 \cdot M_6$

Un exemple qui vous concerne

a = ______________________________, b = ______________________________

c = ______________________________, R = ______________________________

a	b	c	R
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

R =

Circuit :

De Morgan : PdS <-> SdP

On utilise le théorème De Morgan pour changer de forme :

Théorèmes :

$$x \cdot y = \overline{\overline{x} + \overline{y}}$$

$$x + y = \overline{\overline{x} \cdot \overline{y}}$$

Exemples :

$$
\begin{aligned}
M_0 \cdot M_2 \cdot M_4 \cdot M_6 &= \overline{\overline{M_0} + \overline{M_2} + \overline{M_4} + \overline{M_6}} \\
&= \overline{m_0 + m_2 + m_4 + m_6} \\
&= m_1 + m_3 + m_5 + m_7
\end{aligned}
$$

$$
\begin{aligned}
m_1 + m_3 + m_5 + m_7 &= \overline{\overline{m_1} \cdot \overline{m_3} \cdot \overline{m_5} \cdot \overline{m_7}} \\
&= \overline{M_1 \cdot M_3 \cdot M_5 \cdot M_7} \\
&= M_0 \cdot M_2 \cdot M_4 \cdot M_6
\end{aligned}
$$

Voyez-vous le pattern ?

Je me souviens

Certain que vous êtes maintenant super-excités puisque vous pouvez concevoir des circuits logiques combinatoires. :-)

Mais qu'est-ce-qui se passe quand vous aurez besoin de s'en souvenir des valeurs intermédiares ou des valeurs anciens des entrées ? :- l

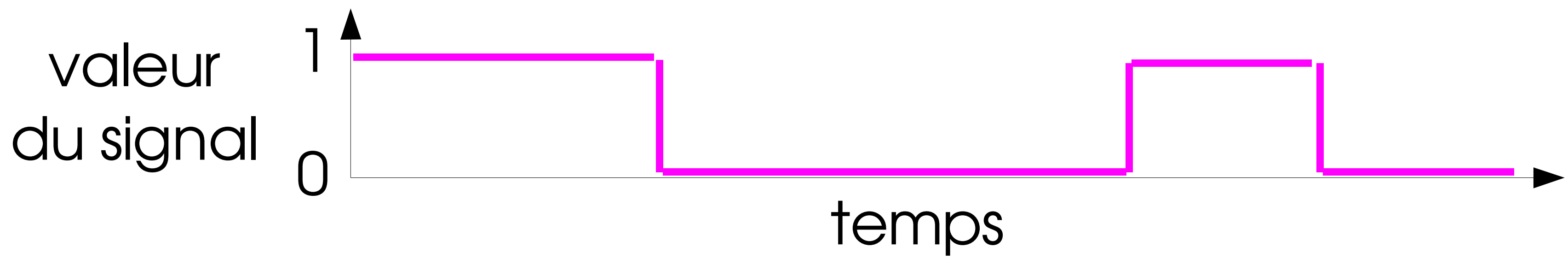
On vous présente alors les deux mémoires de base, les bascules et les bistables...

(Une très bonne reférence pour ces deux derniers est : http://www.play-hookey.com/digital/ . C'est super-cool !)

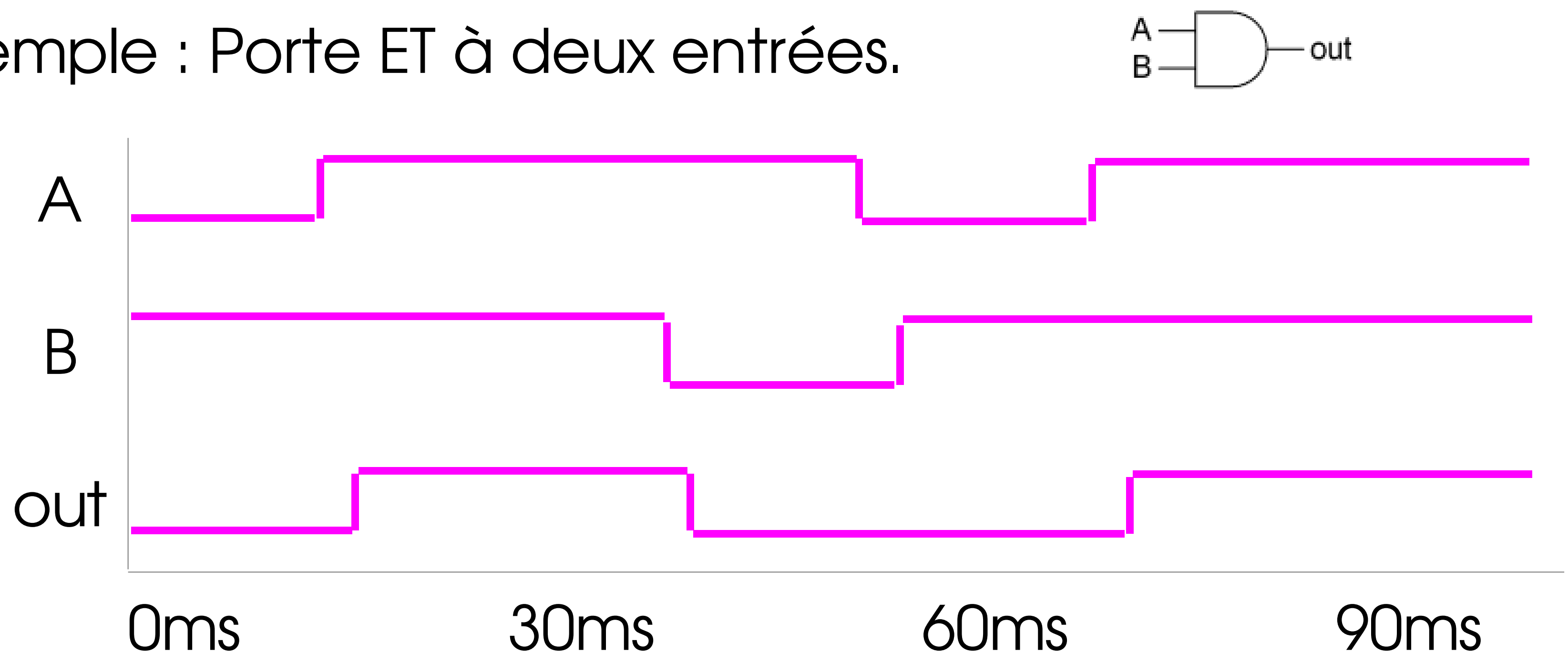
...mais en premier on introduit les diagrammes temporelles.

Diagrammes temporelles

L'axe X représente le temps, l'axe Y représente le valeur :



Exemple : Porte ET à deux entrées.

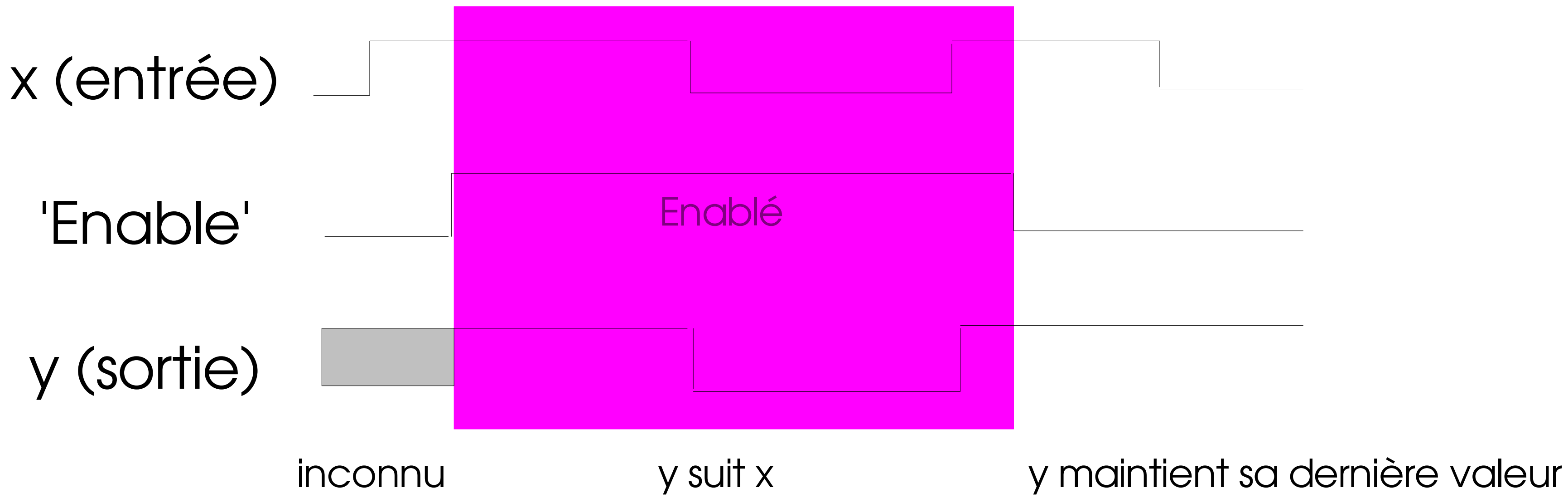


Bistable

In English: Latch

Mémoire asynchrone (ne réagit pas selon une horloge)

Suit son entrée quand c'est enablé :



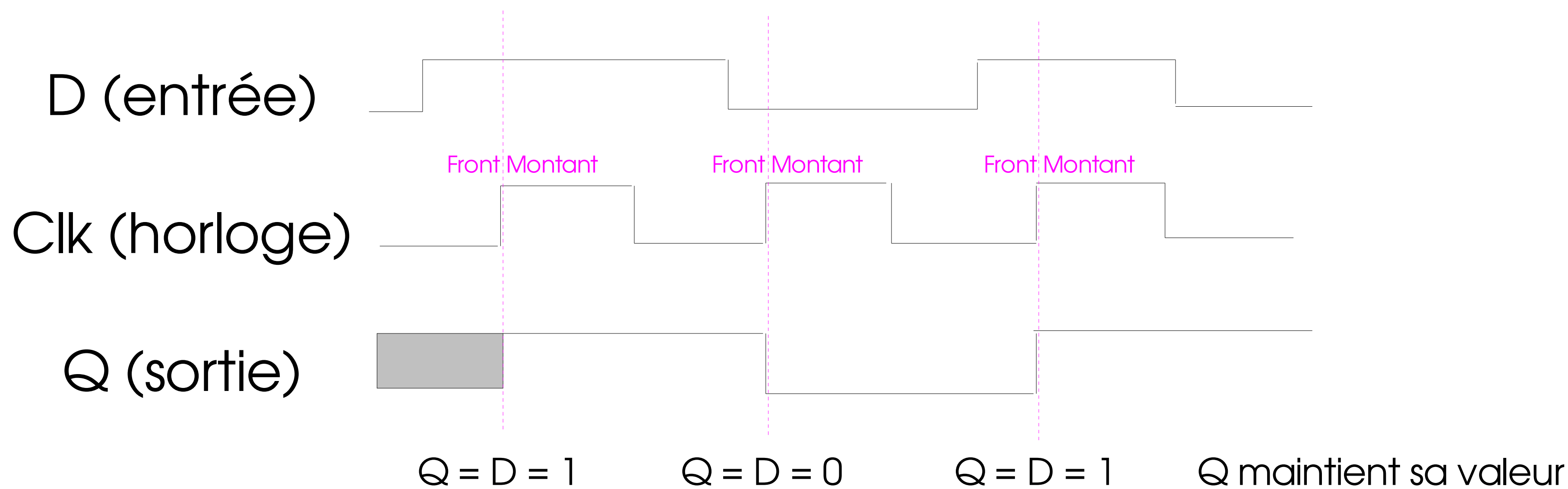
(N.B. Regardez http://www.play-hookey.com/digital/d_nand_latch.html)

Bascule

In English: Flip-flop

Mémoire synchrone (réagit selon une horloge)

Capture son entrée lorsq'un front montant de l'horloge :



(N.B. Regardez http://www.play-hookey.com/digital/d_nand_flip-flop.html)

Plus sur les mémoires plus tard...

On verra plus sur les bistables et bascules plus tard. On ne les présente maintenant que pour introduire les diagrammes temporelles.

Quand-même, souvenez-vous bien de leur comportement temporel. C'est une question populaire d'intra. ;-)

Les bascules nous serviront pour la conception de circuits séquentiels, ce qui est présenté dans la deuxième partie du cours.

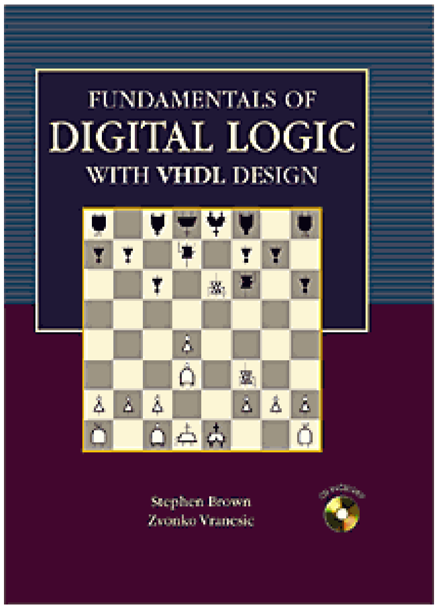
Sommaire (par Monsieur T)

Faut maîtriser les tables de verité et les diagrammes temporelles.

Faut connaître l'algèbre de Boole comme George Boole lui même !

Faut connaître les portes logiques et être capable de faire des circuits combinatoires avec.

On s'amuse bien à faire les devoirs



Lecture courant : 2.1 – 2.6, 7.1 - 7.9

Lecture à venir : 3.1 – 3.5, 3.8

Exercices : Chapitre 2, site web

(réponses sur le site web du cours)

Lecture optionelle : Polycopié '99 2.1 – 2.3

Monsieur T sait si vous avez fait vos devoirs !!!

Concours

a	b	c	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

1) Écrivez f comme la somme de ses minterms.
2) Utilisez 1) pour écrire f en termes de a, b et c.
3) Simplifiez 2) en se servant de l'algèbre de Boole.
4) Dessinez le circuit en utilisant des portes logiques. (Vous pouvez utiliser des portes ET et OU à plus que deux entrées)

Le design (correct) utilisant le moindre portes logiques gagnera. Bonne chance !