

Construction de circuits combinatoires

- Plusieurs critères possibles :
 - Le moins de transistors possible
 - Le moins de consommation d'énergie possible
 - Le plus rapide possible
- Spécification d'un circuit
 - Table de vérité
 - Formule logique

Spécification : Formule Logique

- Constantes : 0, 1
- Variables : x, y, z, \dots
- Opérateurs : +, , .

loi	forme "et"	forme "ou"
identité	$1x = x$	$0 + x = x$
nullité	$0x = 0$	$1 + x = 1$
idempotence	$xx = x$	$x + x = x$
inversion	$x\bar{x} = 0$	$x + \bar{x} = 1$
commutativité	$xy = yx$	$x + y = y + x$
associativité	$(xy)z = x(yz)$	$(x + y) + z = x + (y + z)$
distributivité	$x + yz = (x + y)(x + z)$	$x(y + z) = xy + xz$
absorption	$x(x + y) = x$	$x + xy = x$
de Morgan	$\overline{xy} = \bar{x} + \bar{y}$	$\overline{x + y} = \bar{x} \cdot \bar{y}$

Spécification : Table de vérité

- Pour toute combinaison des entrées, donner la valeur de sortie:

x	y	z	out
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

- Souvent, une spécification complète n'est pas nécessaire:

x	y	z	out
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	-
1	0	0	-
1	0	1	1
1	1	0	0
1	1	1	1

Abréviation de la table de vérité

- Souvent la valeur de sortie est identique pour plusieurs combinaisons d'entrées:

x_1	x_2	x_3	x_4	x_5	y
0	0	-	-	-	1
0	1	-	-	-	0
1	0	-	-	0	0
1	0	-	-	1	1
1	1	0	-	-	1
1	1	1	-	-	0

- Il est possible d'avoir des valeurs de sorties non précisées dans une table abrégée:

x_1	x_2	x_3	x_4	x_5	y
0	0	-	-	-	1
0	1	-	-	-	0
1	0	-	-	0	-
1	0	-	-	1	1
1	1	0	-	-	-
1	1	1	-	-	0

Méthode générale de construction d'un circuit combinatoire

- Si la table de vérité n'existe pas, la construire
- Première couche :
Pour chaque ligne de la table dont la valeur de sortie est 1,
 - mettre une porte non-et, avec une entrée normale pour un 1,
 - et une entrée inversée pour un 0
- Deuxième couche :
Mettre une porte non-et avec en entrée les sorties de la première couche

Exemple de construction générale

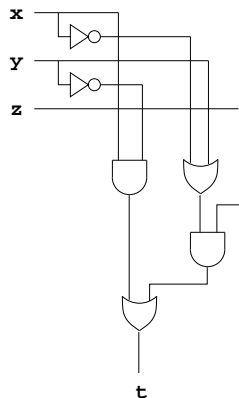
Réaliser un circuit combinatoire à partir de la formule logique:

$$t = x\bar{y} + z(\bar{x} + y)$$

- Table de vérité:

x	y	z	t
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

- le circuit :



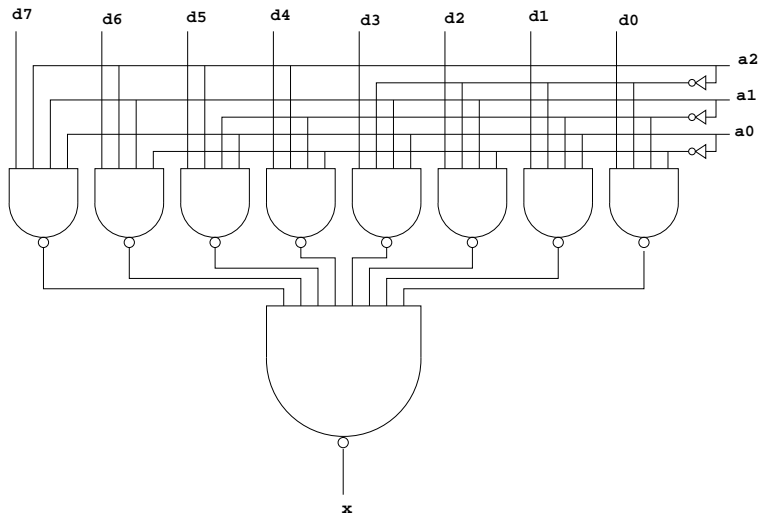
Exemple où la méthode générale est inadaptée

- Circuit avec n entrées "adresses" et 2^n entrées "données"
- L'adresse permet de choisir l'une des entrées "données"
- Table de vérité (abrégée) ($n = 3$):

a_2	a_1	a_0	d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0	out
0	0	0	-	-	-	-	-	-	-	c	c
0	0	1	-	-	-	-	-	-	c	-	c
0	1	0	-	-	-	-	-	c	-	-	c
0	1	1	-	-	-	-	c	-	-	-	c
1	0	0	-	-	-	c	-	-	-	-	c
1	0	1	-	-	c	-	-	-	-	-	c
1	1	0	-	c	-	-	-	-	-	-	c
1	1	1	c	-	-	-	-	-	-	-	c

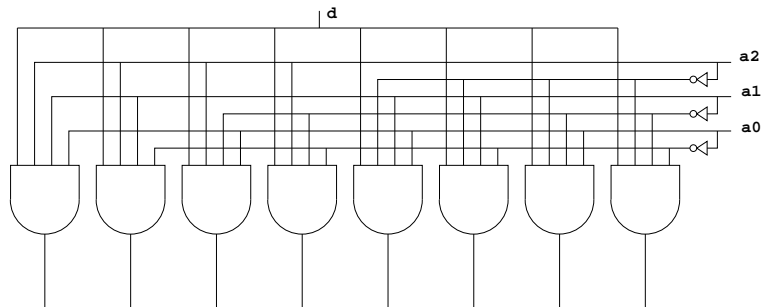
- La table de vérité non abrégée a 2048 lignes dont 1024 avec out = 1
- Donc le circuit construit avec la méthode générale a 1025 portes

Le circuit du multiplexeur



Démultiplexeur

a_2	a_1	a_0	d	x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0
0	0	0	c	0	0	0	0	0	0	0	c
0	0	1	c	0	0	0	0	0	0	c	0
0	1	0	c	0	0	0	0	0	c	0	0
0	1	1	c	0	0	0	0	c	0	0	0
1	0	0	c	0	0	0	c	0	0	0	0
1	0	1	c	0	0	c	0	0	0	0	0
1	1	0	c	0	c	0	0	0	0	0	0
1	1	1	c	c	0	0	0	0	0	0	0



Décodeur

a_2	a_1	a_0	x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

