Plan

- Les Systèmes de Numération
- Fonctions et Circuits Logiques
- Simplification des Fonctions Logiques
- Les Différents Codes

Fonctions et Circuits Logiques

- Définition
- Algèbre de commutation ou algèbre de Boole
- Fonction logique
- Circuits combinatoires SSI & MSI

Définitions

- Élément logique
  - 2 éléments logiques notés « 0 » et « 1 »
  - Le symbole « 1 » désigne une action comme une lampe s'allume, la porte s'ouvre …
  - Le symbole « 0 » indique généralement l'absence d'action
- Variable logique ou booléenne X
  - Une variable logique ou booléenne est une grandeur qui ne peut prendre que 2 états « 0 » ou « 1 »
  - Domaine de définition $B_2 = \{0,1\}$
  - Si X est une variable booléenne, on a
    - $X \neq 0$ si et seulement si $X = 1$
    - $X \neq 1$ si et seulement si $X = 0$

Définitions

Opérateurs logiques élémentaires

Inversion (Not) ou Complémentation

$B_2 \rightarrow B_2$

Notation : $\overline{X}$

X	$\overline{X}$
0	1
1	0

Opération OU (OR) ou Union

$B_2 \text{ x } B_2 \rightarrow B_2$

Notation : $X \cup Y$ ou $\mathrm{X+Y}$

X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1

Opération ET (AND) ou Intersection

$B_2 \text{ x } B_2 \rightarrow B_2$

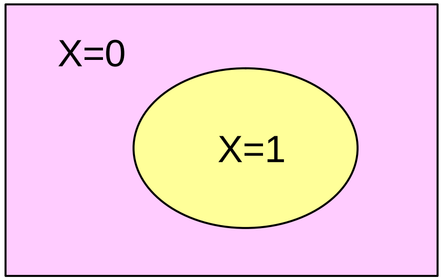
Notation : $X \cap Y$ ou $\mathrm{X.Y}$ ou XY

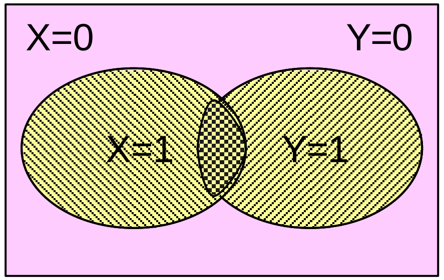
X	Y	X.Y
0	0	0
0	1	0
1	0	0
1	1	1

Définitions

Diagramme de Venn

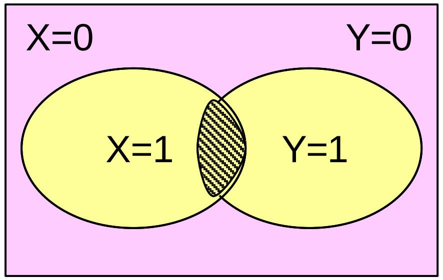
- Les valeurs d'une variable booléenne X peuvent être représentées par 2 régions d'un plan délimitées par une courbe fermée.



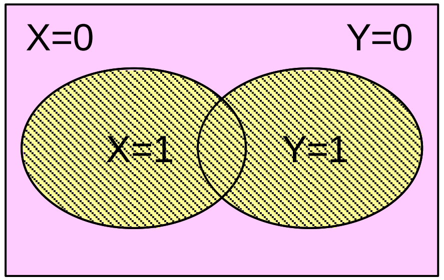


- Cas de 2 variables booléennes X et Y → 2 domaines

X.Y



X + Y



- 3 variables booléennes → 3 domaines

Fonctions et Circuits Logiques

- Définition
- Algèbre de commutation ou algèbre de Boole
- Fonction logique
- Circuits combinatoires SSI & MSI

Lois fondamentales de l'algèbre de Boole

- L'algèbre de commutation ou algèbre de Boole est le système algébrique constitué de l'ensemble B_2 et des opérations ET, OU, PAS.

Axiomes de l'algèbre de boole

Fermeture	A.B Variable logique définie par la table ET A+B Variable logique définie par la table OU
Commutativité	$A.B = B.A$ $A+B = B+A$
Associativité	$A.(B.C) = (A.B).C$ $A+(B+C) = (A+B)+C$
Distributivité	$A.(B+C) = A.B + A.C$ $A+(B.C) = (A+B).(A+C)$ Différent algèbre classique
Complémentarité	$A+\overline{A} = 1$ $A.\overline{A} = 0$
Idempotence	$A+A = A$ $A.A = A$
Identités remarquables	$1.A = A$ $1+A = 1$ $0. A = 0$ $0+A = A$

Théorème de De Morgan

- Théorème 1
  - La négation d'un produit de variables est égale à la somme des négations des variables

$$\overline{A.B.C} = \overline{A} + \overline{B} + \overline{C}$$

- Théorème 2
  - La négation d'une somme de variables est égale au produit des négations des variables

$$\overline{A+B+C} = \overline{A}\,.\,\overline{B}\,.\,\overline{C}$$

Fonctions et Circuits Logiques

- Définition
- Algèbre de commutation ou algèbre de Boole
- Fonction logique
- Circuits combinatoires SSI & MSI

Définitions

- Une fonction logique de n variables $x_1, \ldots, x_n$ est une application qui a toute combinaison de

$$\text{n variables} \in B_2^n \rightarrow \text{un élément} \in B_2$$

- Une fonction logique ne peut prendre que 2 états 0 ou 1

- Le nombre de fonctions que l'on peut créer avec n variables est 2^{2^n} puisqu'à chacune des 2^n combinaisons de variables, on peut faire correspondre les valeurs 0 ou 1

Fonction complètement définie

- Une fonction logique est complètement définie quand on connaît sa valeur 0 ou 1 pour toutes les combinaisons possibles des variables.
- Ces combinaisons sont au nombre de 2^n pour n variables

Exemple

- Soit une fonction de 3 variables f(x,y,z)
- $2^3 = 8$ combinaisons
- Répertorier les combinaisons dans l'ordre croissant de 0 à 2^n-1

X	Y	Z	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Fonction incomplètement définie

- Une fonction logique est incomplètement définie quand sa valeur est indifférente ou non spécifiée pour certaines combinaisons de variables
- On note sa valeur par X ou Æ

Exemple

- Soit une fonction de 3 variables g(x,y,z)
- $2^3 = 8$ combinaisons dont 4 combinaisons indéfinies

X	Y	Z	g
0	0	0	X
0	0	1	0
0	1	0	0
0	1	1	X
1	0	0	X
1	0	1	1
1	1	0	1
1	1	1	X

Formes Canoniques des Fct. Logiques

Première Forme Canonique (1)

- Appelé aussi
  - Produels de Produits
  - Forme canonique disjonctive
- S'exprime sous la forme d'une somme de produits

- Écriture à partir de la table de vérité
  - Repérer dans la table de vérité les combinaisons x, y, z pour lesquelles la fonctions vaut 1
  - Pour ces combinaisons, faire le produit des variables en affectant le symbole ¾ aux variables dont l'état est 0. On obtient les monômes de la fonction
  - Faire la somme de tous les monômes

Formes Canoniques des Fct. Logiques
Première Forme Canonique (2)

- Exemple
  - soit la fonction f tel que
    - f = 1 si la majorité des variables sont à 1
    - f = 0 sinon

Réalisation de la table de vérité

X	Y	Z	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

1) Recherche les cas où la fonction vaut 1

X	Y	Z	f
0	0	0	0
0	0	1	0
0	1	0	0
0	**1**	**1**	**1**
1	0	0	0
1	**0**	**1**	**1**
1	**1**	**0**	**1**
1	**1**	**1**	**1**

2) Écriture des monômes

- Remarque
  - Comment obtenir le complément de f ?

X	Y	Z	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$\overline{f}(x,y,z) = \overline{x}.\overline{y}.\overline{z} + \overline{x}.\overline{y}.z + \overline{x}.y.\overline{z} + x.\overline{y}.\overline{z}$$

Formes Canoniques des Fct. Logiques
Première Forme Canonique (4)

- Écriture de la table de vérité à partir de f canonique
  - dresser la table de vérité à n variables
  - les combinaisons correspondantes à un monômes de f seront affectées à l'état 1, les autres à l'état 0
- Exemple

$$f(x,y,z) = \overline{x}.\overline{y}.\overline{z} + x.\overline{y}.z + x.y.z$$

X	Y	Z	f
0	**0**	**0**	**1**
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	**0**	**1**	**1**
1	1	0	0
1	**1**	**1**	**1**

Formes Canoniques des Fct. Logiques

Deuxième Forme Canonique (1)

- Appelé aussi
  - Produits de Produels
  - Forme canonique conjonctive
- S’exprime sous la forme d’un produit de sommes

- Écriture à partir de la table de vérité
  - Repérer les combinaisons pour lesquelles l’état de f est 0
  - Pour ces combinaisons, faire la sommes des variables en affectant le symbole ¾ aux variables dont l’état est 1
  - Faire le produit des sommes

Formes Canoniques des Fct. Logiques

Deuxième Forme Canonique (2)

- Exemple

1) Recherche les cas où la fonction vaut 0

X	Y	Z	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

2) Écriture des monômes

$$x + y + z$$

$$x + y + \overline{z}$$

$$x + \overline{y} + z$$

$$\overline{x} + y + z$$

Formes Canoniques des Fct. Logiques
Deuxième Forme Canonique (3)

- Écriture de la table de vérité à partir de f
  - Dresser la table
  - Pour chaque terme somme de f, prendre la combinaison faisant apparaître
    - un 0 pour une variable directe
    - un 1 pour une variable inverse notée ¾
  - Affecter 0 à f pour ces combinaisons, 1 à f pour les autres
- Exemple

X	Y	Z	f
0	0	0	1
0	**0**	**1**	**0**
0	1	0	1
0	**1**	**1**	**0**
1	0	0	1
1	0	1	1
1	**1**	**0**	**0**
1	1	1	1

$$f(x,y,z) = (x+y+\overline{z}).(x+\overline{y}+\overline{z}).(\overline{x}+\overline{y}+z)$$

Fonctions d'une seule variable Booléenne

- On peut former 2^{2^1} fonctions, soit 4 fonctions
- Ces fonctions sont appelées monoïdes

x	f_0	f_1	f_2	f_3
0	0	0	1	1
1	0	1	0	1

$$\left.\begin{array}{l} f_0 = 0 \\ f_3 = 1 \end{array}\right\} \text{fonctions constantes}$$

$f_1 = x$ c'est la variable elle-même

$f_2 = \overline{x}$ c'est le complément de la variable noté NON ou NOT

Fonctions de deux variables Booléennes

- On peut former $2^{2^{2}}$ fonctions, soit 16 fonctions

x	y	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Fonction à une seule variable

$f_0 = 0 \qquad f_3 = x \qquad f_5 = y$

$f_{15} = 1 \qquad f_{12} = \bar{x} \qquad f_{10} = \bar{y}$

Opérateurs fondamentaux

$f_7 = x + y$ OU

$f_1 = x.y$ ET

Autres fonctions

$f_6 = x.\bar{y} + \bar{x}.y = x \oplus y$ Fonction OU exclusif ou comparateur d'inégalité

$f_9 = x.y + \bar{x}.\bar{y} = x \Theta y$ Fonction Identique ou comparateur d'identité

$f_8 = \bar{x}.\bar{y} = \overline{x + y}$ Fonction NON OU ou NOR

$f_{14} = \bar{x} + \bar{y} = \overline{x.y}$ Fonction NON ET ou NAND

Fonctions et Circuits Logiques

- Définition
- Algèbre de commutation ou algèbre de Boole
- Fonction logique

- Circuits combinatoires SSI & MSI

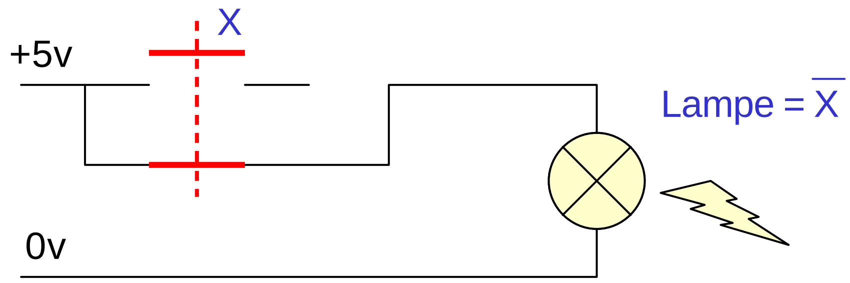
Circuits SSI (Small Scale Integration)

Portes Logiques Élémentaires (1)

Porte NON, PAS ou Inverseur (NOT)

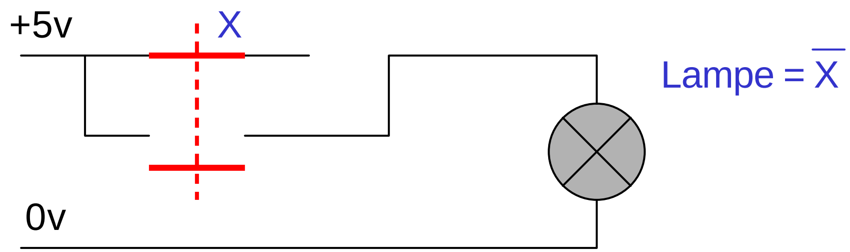
Électricité

Cas 1



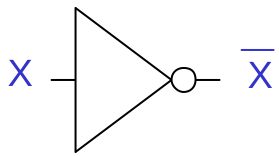
X = 0

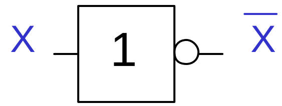
Cas 2



X = 1

Symboles Électroniques

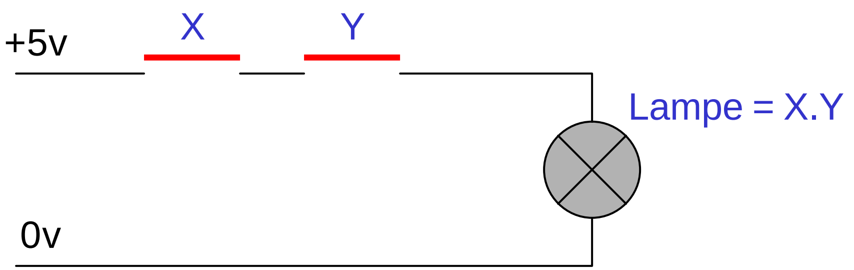




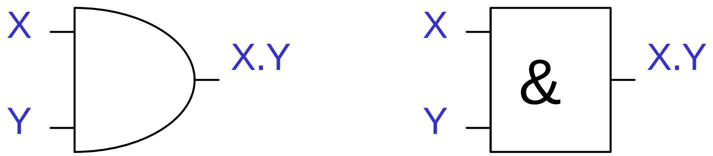
Circuits SSI (Small Scale Integration)
Portes Logiques Élémentaires (2)

Porte ET (AND)

Électricité

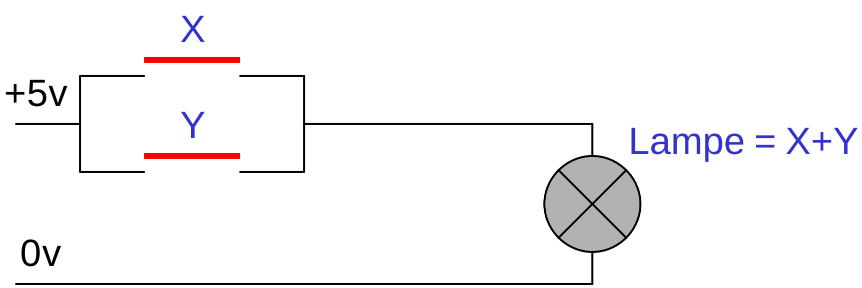


Symboles Électroniques

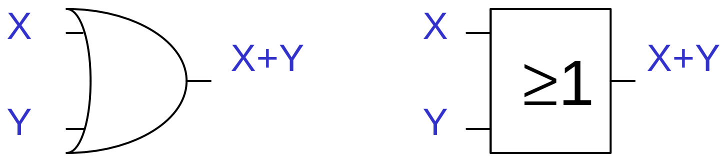


Porte OU (OR)

Électricité



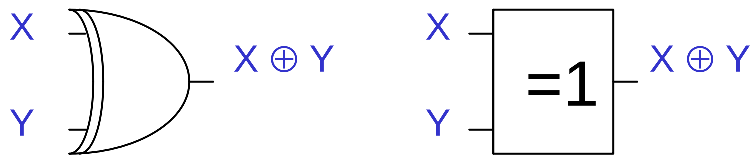
Symboles Électroniques



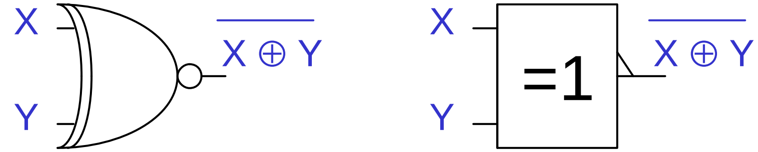
Circuits SSI (Small Scale Integration)

Portes Logiques de Base (1)

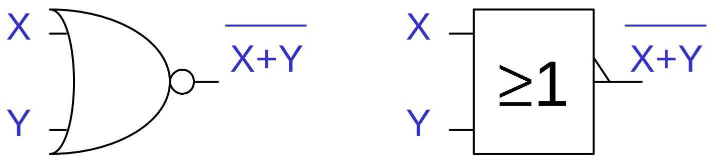
Porte OU Exclusif (EXOR)



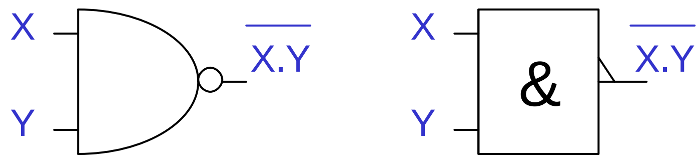
Porte OU Exclusif Complémenté (EXNOR)



Porte NON OU (NOR)

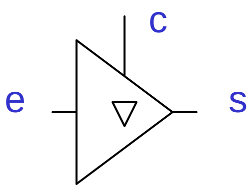


Porte NON ET (NAND)

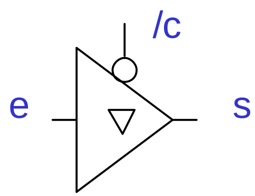


Circuits SSI (Small Scale Integration)
Portes Logiques de Base (2)

Circuits 3 états (TRISTATE)



c = 0 alors s=haute impédance (z)
c = 1 alors s=e



c = 1 alors s=haute impédance (z)
c = 0 alors s=e

e ___ c=0 ___ s=z (haute impédance)

e ___ c=1 ___ s=e

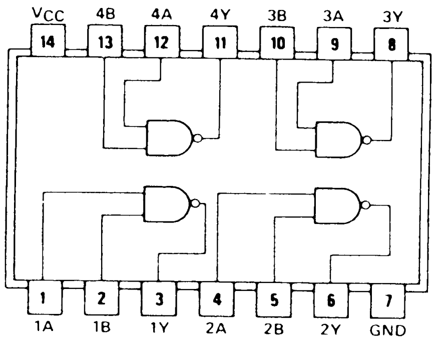
e ___ /c=1 ___ s=e

e ___ /c=0 ___ s= z (haute impédance)

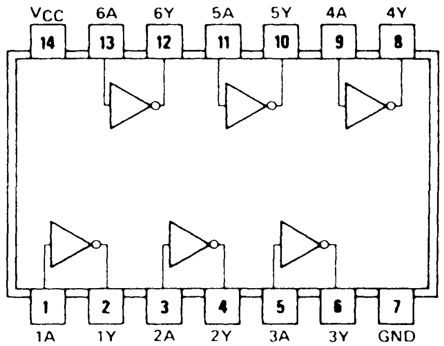
Circuits SSI (Small Scale Integration)

Exemples

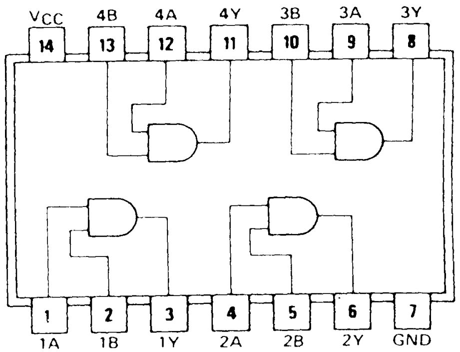
NAND (7400)



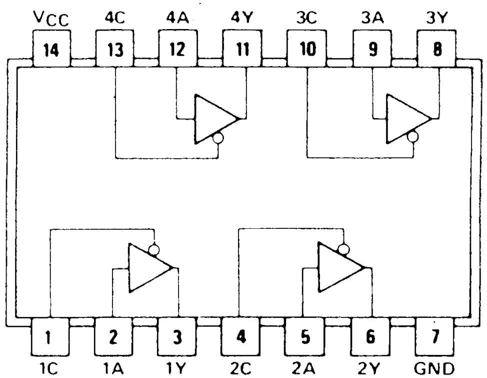
NOT (7404)



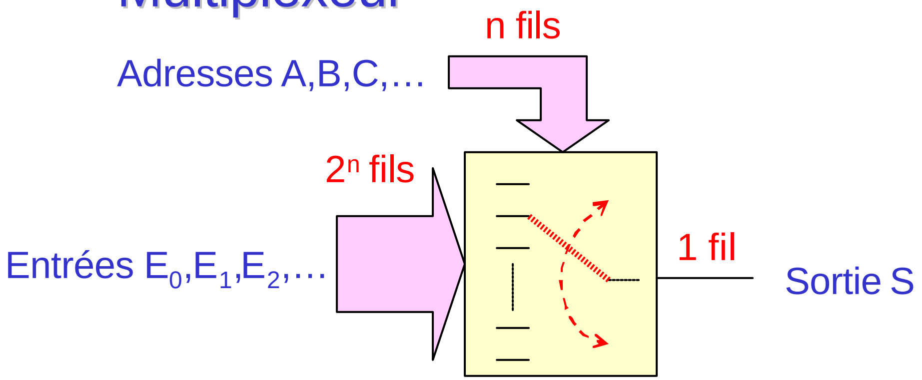
OR (7408)



Buffer Tristate (74126)



Multiplexeur

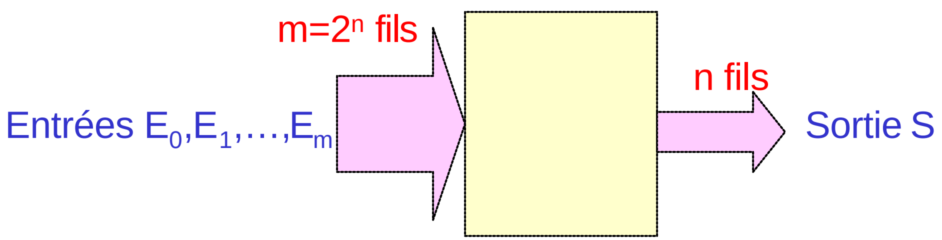


Exemple

Multiplexeur 4 → 1

- 4 entrées = 2^2
- n=2 donc 2 fils d'adresse A et B
- 1 sortie (toujours vrai)

Encodeur



E_{m-1}	…	E_1	E_0	S_n	…	S_1	S_0	
0	…	0	1	0	…	0	0	0
0	…	1	0	0	…	0	1	1
⋮	⋮	⋮		⋮	⋮	⋮		
1	…	0	0	1	…	1	1	m-1

Exemples

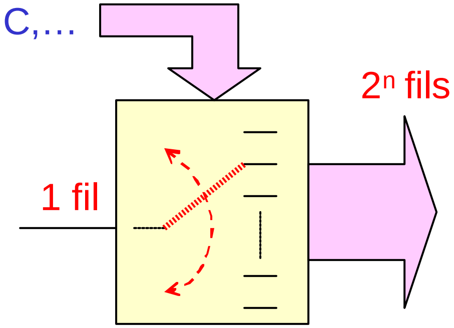
Encodeur 8 → 3
- 8 entrées
- 3 sorties

Encodeur 10 → 4

Circuits MSI (Medium Scale Integration)

Demultiplexeur & Décodeur

Demultiplexeur

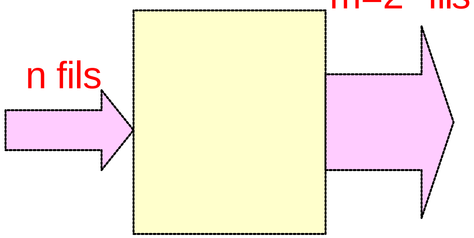


Exemple

Demultiplexeur 1 $\rightarrow$ 8

- 1 entrée (toujours vrai)
- 8 sorties ou 2^3 sorties
- n=3 donc 3 fils d'adresse A,B et C

Décodeur



	E_n	…	E_1	E_0	S_{m-1}	…	S_1	S_0
0	0	…	0	0	0	…	0	1
1	0	…	0	1	0	…	1	0
	⋮		⋮	⋮	⋮		⋮	⋮
m-1	1	…	1	1	1	…	0	0

Exemples

Décodeur 3 $\rightarrow$ 8

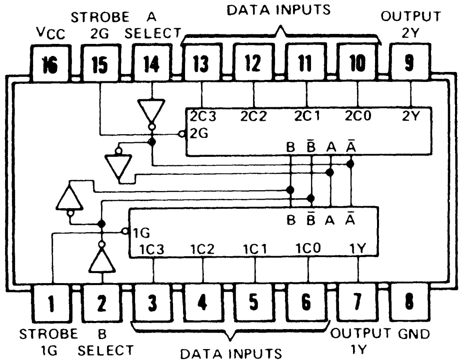
- 3 entrées
- 8 sorties

Décodeur 4 $\rightarrow$ 10

Circuits MSI (Medium Scale Integration)

Exemples

Multiplexeur 4 vers 1 (74153)

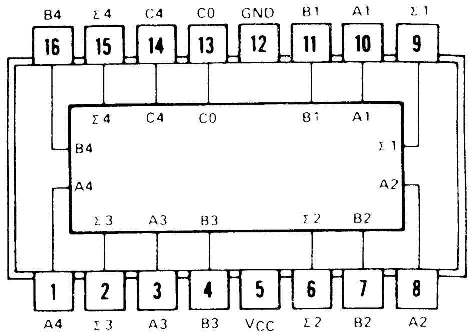


FUNCTION TABLE

SELECT INPUTS		DATA INPUTS				STROBE	OUTPUT
B	A	C0	C1	C2	C3	G	Y
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
L	H	X	L	X	X	L	L
L	H	X	H	X	X	L	H
H	L	X	X	L	X	L	L
H	L	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

Select inputs A and B are common to both sections.
H = high level, L = low level, X = irrelevant

Additionneur 4 bits (7483)



FUNCTION TABLE

INPUT				OUTPUT					
				WHEN C0 = L / WHEN C2 = L			WHEN C0 = H / WHEN C2 = H		
A1 / A3	B1 / B3	A2 / A4	B2 / B4	Σ1 / Σ3	Σ2 / Σ4	C2 / C4	Σ1 / Σ3	Σ2 / Σ4	C2 / C4
L	L	L	L	L	L	L	H	L	L
H	L	L	L	H	L	L	L	H	L
L	H	L	L	H	L	L	L	H	L
H	H	L	L	L	H	L	H	H	L
L	L	H	L	L	H	L	H	H	L
H	L	H	L	H	H	L	L	L	H
L	H	H	L	H	H	L	L	L	H
H	H	H	L	L	L	H	H	L	H
L	L	L	H	L	H	L	H	H	L
H	L	L	H	H	H	L	L	L	H
L	H	L	H	H	H	L	L	L	H
H	H	L	H	L	L	H	H	L	H
L	L	H	H	L	L	H	H	L	H
H	L	H	H	H	L	H	L	H	H
L	H	H	H	H	L	H	L	H	H
H	H	H	H	L	H	H	H	H	H

high level, L = low level