



Multiplexeurs et Démultiplexeurs (MUX)

Cours Systèmes numérique

Hes·so VALAIS
WALLIS



Haute Ecole d'Ingénierie
Hochschule für Ingenieurwissenschaften

Orientation : Informatique et systèmes de communication (ISC)

Cours : Systèmes numérique (DiD)

Auteur : Christophe Bianchi, François Corthay, Pierre Pompili, Silvan Zahno

Date : 25 août 2022

Version : v2.1

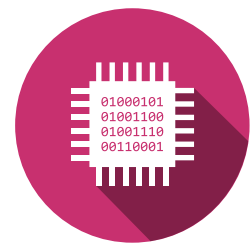


Table des matières

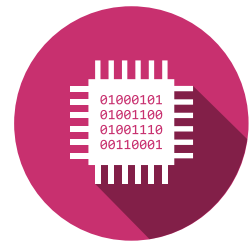
1	Introduction	2
2	Multiplexeur	3
2.1	Fonction logique universelle	3
2.2	Réalisation de multiplexeurs à l'aide de portes logiques de base	4
2.3	Réalisation de multiplexeurs avec une structure en arbre	5
2.4	Réduction de fonctions à base de multiplexeurs	5
3	Démultiplexeur	8
3.1	Réalisation de démultiplexeurs à l'aide de portes logiques de base	8
3.2	Réalisation de démultiplexeurs avec une structure en arbre	9
	Références	10



1 Introduction

Le multiplexeur est un dispositif permettant d'aiguiller l'une de ses entrées vers son unique sortie. Entre autres, le **multiplexeur** peut servir à réaliser une fonction logique quelconque. A l'inverse, un **démultiplexeur** aiguille son entrée vers l'une de ses sorties.

Ce chapitre présente ces 2 opérateurs logiques, leur circuit interne et les possibilités de simplifier des circuits à base de ces opérateurs.



2 Multiplexeur

Le multiplexeur transmet l'une de ses 2^n entrées de données à sa sortie, et ce en fonction de la combinaison de ses n entrées de sélection.

Le multiplexeur se comporte comme un aiguillage.

Exemple : multiplexeur de 2 à 1

Le multiplexeur de 2 à 1 de la figure 1 aiguille l'une de ses 2 entrées de données, x_0 et x_1 , vers sa sortie y , en fonction du signal de sélection sel .



FIGURE 1 – Multiplexeur de 2 à 1

Exemple : multiplexeur de 4 à 1

Le multiplexeur de 4 à 1 de la figure 2 aiguille l'une de ses 4 entrées de données, x_0 à x_3 , vers sa sortie y , en fonction des 2 bits de sélection, s_1 et s_0 .

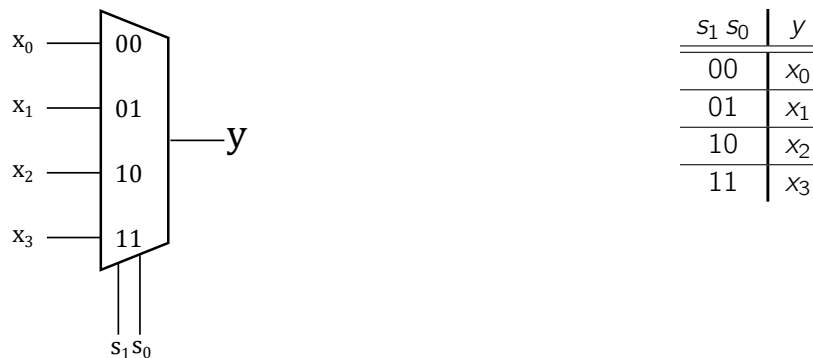


FIGURE 2 – Multiplexeur de 4 à 1

2.1 Fonction logique universelle

Le multiplexeur peut s'utiliser pour réaliser une fonction quelconque.

Pour cela, il suffit de brancher les entrées de la fonction sur les lignes de sélection du multiplexeur et de fixer les entrées de données à '0' ou à '1' selon la valeur correspondante de la fonction pour les entrées sélectionnées.

Le multiplexeur permet ainsi de réaliser électroniquement une table de vérité.



Exemple : fonction à 2 entrées

La figure 3 montre comment réaliser la fonction OU-exclusif de 2 entrées à l'aide d'un multiplexeur à 2 entrées de sélection, et donc à $2^2 = 4$ entrées de données.

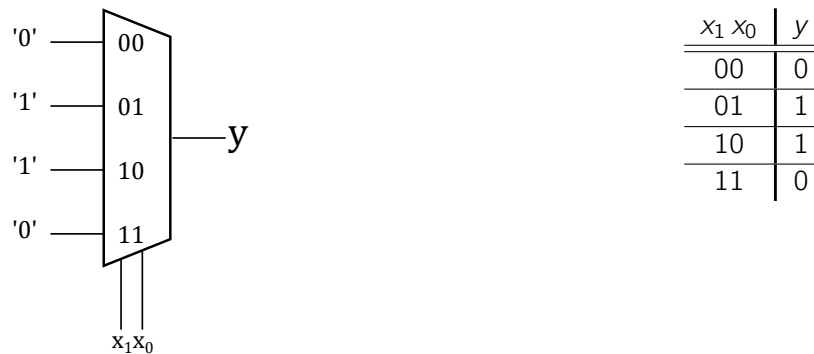


FIGURE 3 – Fonction OU-exclusif à 2 entrées

2.2 Réalisation de multiplexeurs à l'aide de portes logiques de base

Un multiplexeur de 2^n à 1 se réalise avec 2^n portes ET recevant chacune l'une des entrées de données et toutes les entrées de sélection, inversées ou non. Les entrées de sélection inversées ou non se distribuent de manière à ce que seule l'une des portes ET n'ait pas nécessairement sa sortie à '0'. Les sorties des portes ET sont amenées à une porte OU qui réalise la sortie du multiplexeur.

Exemple : multiplexeur de 4 à 1

Le multiplexeur de 4 à 1 se réalise à l'aide de portes ET, OU et d'inverseurs comme présenté à la figure 4.

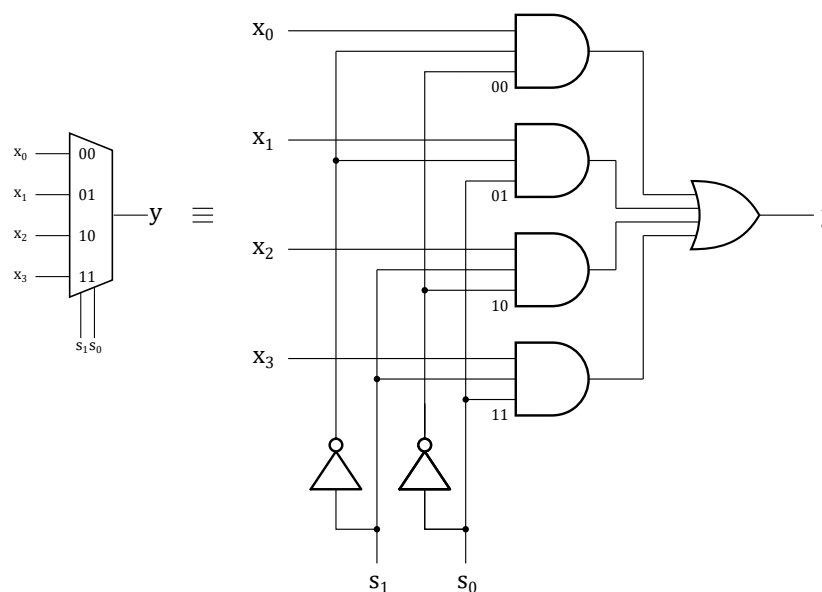


FIGURE 4 – Réalisation d'un multiplexeur de 4 à 1 à l'aide de portes logiques de base



2.3 Réalisation de multiplexeurs avec une structure en arbre

Un multiplexeur de 2^n à 1 se réalise avec une structure en arbre à n étages de multiplexeurs de 2 à 1. Les multiplexeurs de chaque étage sont contrôlés par l'une des variables de sélection.

Exemple : multiplexeur de 4 à 1

Le multiplexeur de 4 à 1 se réalise à l'aide de multiplexeurs de 2 à 1 comme présenté à la figure 5.

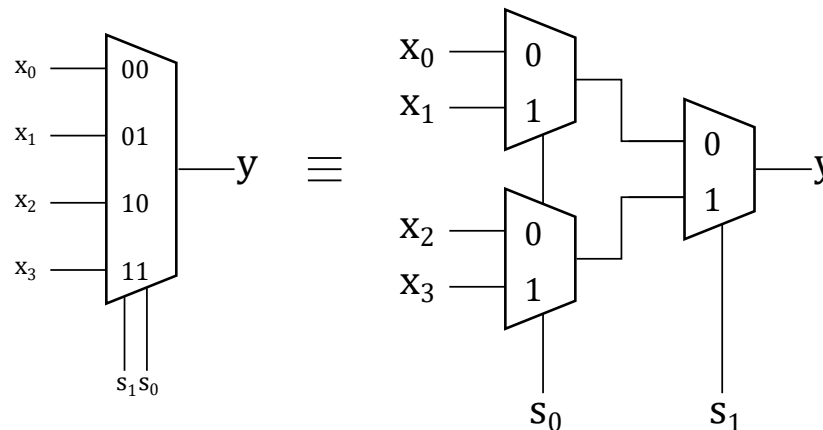


FIGURE 5 – Réalisation d'un multiplexeur de 4 à 1 avec une structure en arbre

2.4 Réduction de fonctions à base de multiplexeurs

Lorsque le multiplexeur est utilisé comme générateur de fonction, il est toujours possible de réduire la taille du circuit en partant de la représentation du multiplexeur sous forme de structure en arbre. Cette réduction se base sur les considérations suivantes :

- un multiplexeur de 2 à 1 dont les 2 entrées sont identiques peut être éliminé et sa sortie connecté à l'une des entrées,
- un multiplexeur de 2 à 1 dont les entrées $x_0 = 0$ et $x_1 = 1$ peut être éliminé et sa sortie connecté à l'entrée de sélection,
- un multiplexeur de 2 à 1 dont les entrées $x_0 = 1$ et $x_1 = 0$ peut être éliminé et sa sortie connecté à l'inverse de l'entrée de sélection.

Cette simplification permet de réduire d'une unité la profondeur de la structure en arbre.

La simplification peut être conduite sur les étages suivants et l'observation du circuit résultant permet aussi de dénombrer des multiplexeurs avec toutes les entrées identiques pour n'en conserver qu'un seul.

Exemple : réduction d'un niveau

La figure 6 présente la réalisation de la fonction majorité de 3 entrées à l'aide d'un multiplexeur à 3 entrées de sélection et donc $2^3 = 8$ entrées de données.

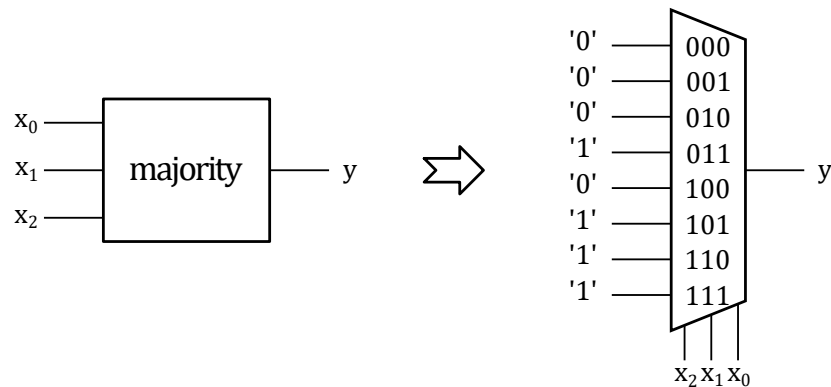
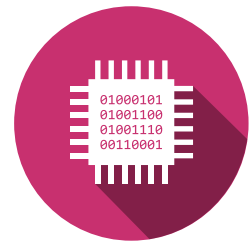


FIGURE 6 – Réalisation de la fonction majorité à l'aide d'un multiplexeur

La figure 7 présente la réduction de ce circuit à un multiplexeur de 4 à 1.

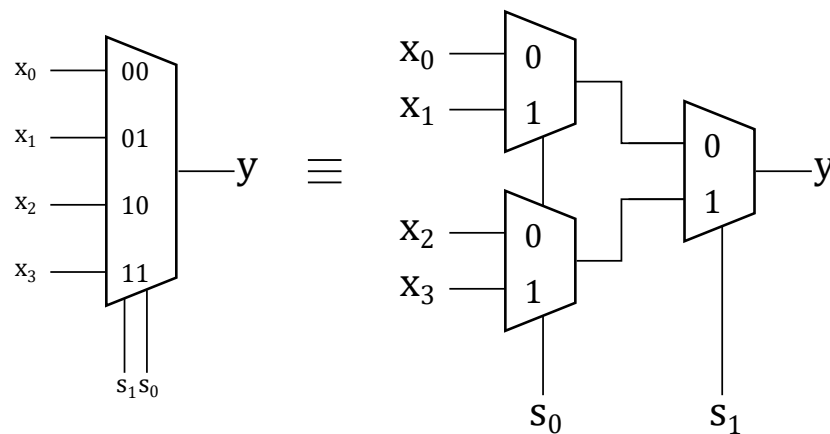


FIGURE 7 – Réduction d'un multiplexeur de 8 à 1 à un multiplexeur de 4 à 1

Exemple : réduction de plusieurs niveaux

La figure 8 présente la simplification d'un multiplexeur de 16 à 1 à entrées constantes en utilisant des multiplexeurs de 2 à 1.

Deux paires de multiplexeurs ont toutes les entrées identiques et un seul multiplexeur a été conservé pour ces groupes. Remarquons que le multiplexeur servant à inverser l'entrée x_0 a été conservé.

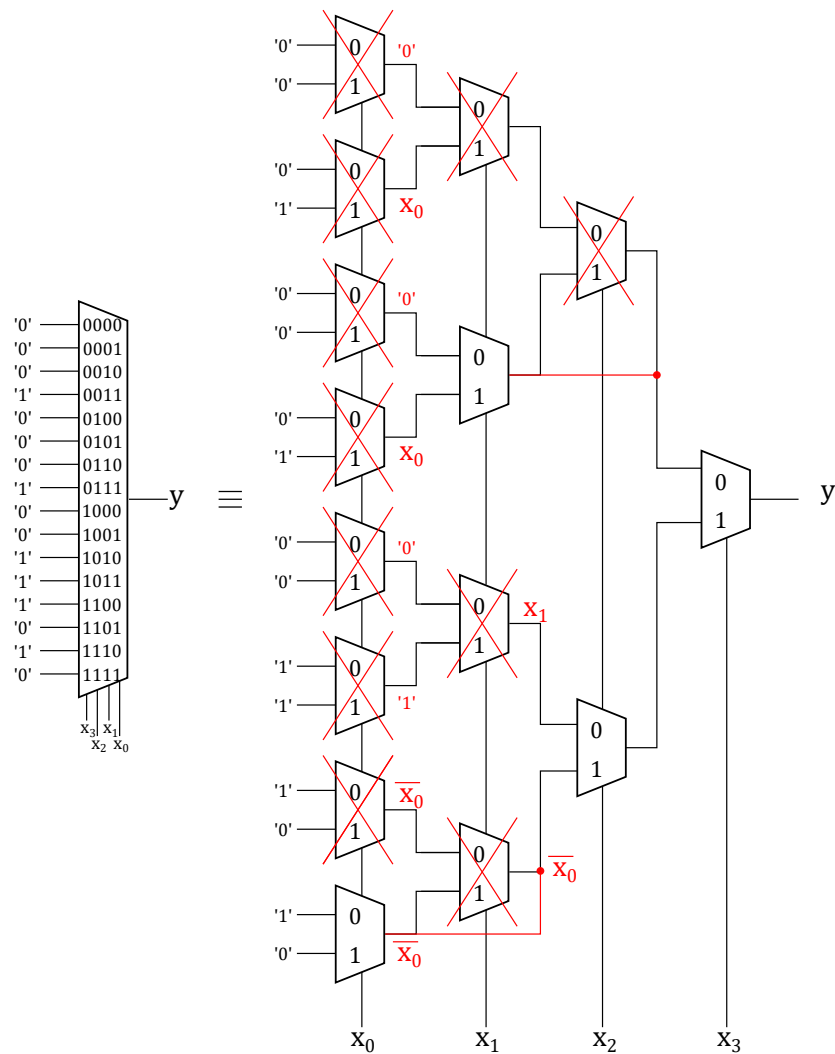
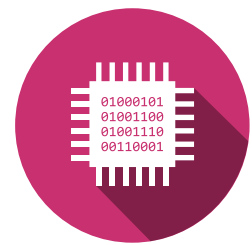


FIGURE 8 – Simplification d'un multiplexeur de 16 à 1 avec des multiplexeurs de 2 à 1



3 Démultiplexeur

Le démultiplexeur transmet son entrée de donnée à l'une de ses 2^n sorties, et ce en fonction de la combinaison de ses n entrées de sélection. Les sorties non sélectionnées sont fixées à '0'.

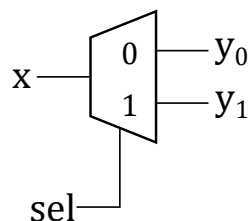
Le démultiplexeur se comporte comme un aiguillage.



Il est trop souvent oublié que les sorties non sélectionnées sont fixées à '0'.

Exemple : démultiplexeur de 1 à 2

Le démultiplexeur de 1 à 2 de la figure 9 aiguille son entrée de donnée, x , vers l'une de ses 2 sorties, y_0 ou y_1 , en fonction du signal de sélection sel . L'autre sortie est mise à '0'.

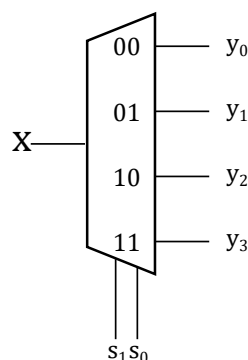


sel	y_1	y_0
0	0	x
1	x	0

FIGURE 9 – Démultiplexeur de 1 à 2

Exemple : démultiplexeur de 1 à 4

Le démultiplexeur de 1 à 4 de la figure 10 aiguille son entrée de donnée, x , vers l'une de ses 4 sorties, y_0 à y_3 , en fonction des 2 bits de sélection, s_1 et s_0 . Les autres sorties sont mises à '0'.



$s_1 s_0$	y_3	y_2	y_1	y_0
00	0	0	0	x
01	0	0	x	0
10	0	x	0	0
11	x	0	0	0

FIGURE 10 – Démultiplexeur de 1 à 4

3.1 Réalisation de démultiplexeurs à l'aide de portes logiques de base

Un démultiplexeur de 1 à 2^n se réalise avec 2^n portes ET recevant chacune l'une des entrées de données et toutes les entrées de sélection, inversées ou non, de manière similaire à la composition d'un multiplexeur.



Exemple : démultiplexeur de 1 à 4

Le démultiplexeur de 4 à 1 se réalise à l'aide de portes ET et d'inverseurs comme présenté à la figure 11.

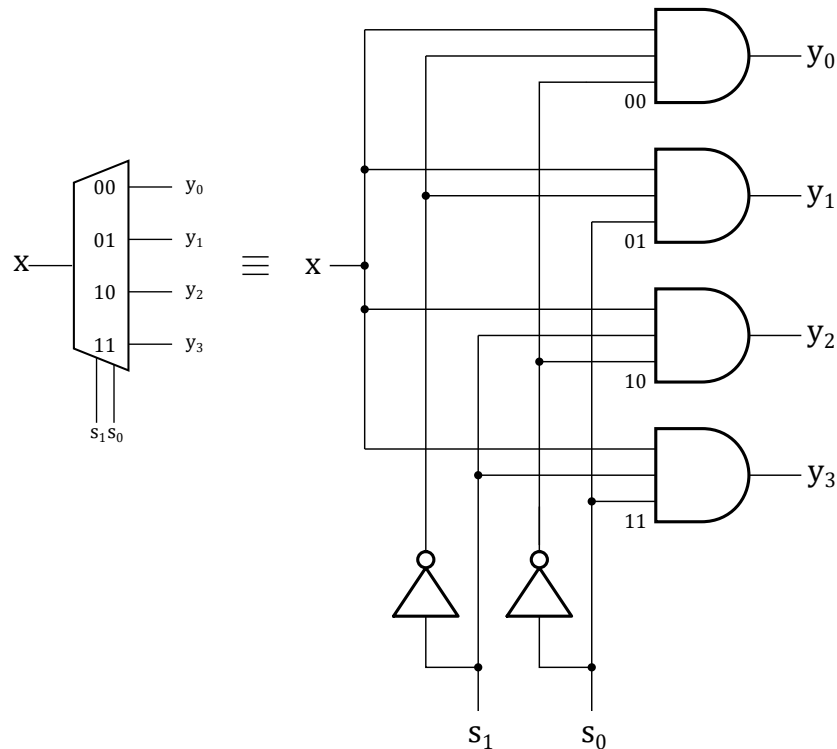


FIGURE 11 – Réalisation d'un démultiplexeur à l'aide de portes logiques de base

3.2 Réalisation de démultiplexeurs avec une structure en arbre

Un démultiplexeur de 1 à 2^n se réalise avec une structure en arbre à n étages de démultiplexeurs de 2 à 1, de manière similaire au multiplexeur.

Exemple : démultiplexeur de 4 à 1

Le démultiplexeur de 4 à 1 se réalise à l'aide de démultiplexeurs de 2 à 1 comme présenté à la figure 12.

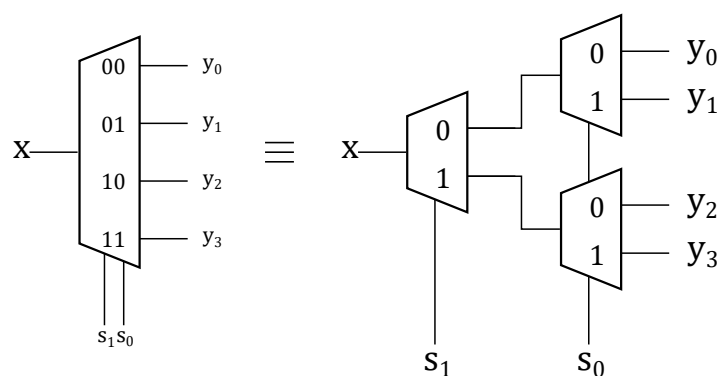
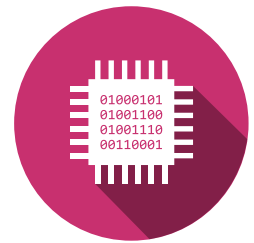


FIGURE 12 – Réalisation d'un démultiplexeur de 1 à 4 avec une structure en arbre



Références

- [1] Michael D. CILETTI et M. Morris MANO. *Digital Design*. second edition. New-Jersey : Prentice-Hall, 2007.
- [2] Marcel GINDRE et Denis ROUX. *Electronique Numérique, Logique Combinatoire et Technologie*. Paris : McGraw-Hill, 1987.
- [3] Daniel MANGE. *Analyse et synthèse des systèmes logiques*. Editions Géorgi. T. Traité d'électricité, volume V. St Saphorin : PPUR presses polytechniques, 1995. 362 p. ISBN : 978-2-88074-045-0. Google Books : [5NSdD4GRl3cC](#).
- [4] John F. WAKERLY. *Digital Design : Principles And Practices*. 3rd edition. Prentice-Hall, 2008. ISBN : 0-13-082599-9.