

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE

*Université de Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj*

*Faculté des Sciences et de la technologie*

*Département d'Electronique*

# *Mémoire*

*Présenté pour obtenir*

LE DIPLOME DE MASTER

FILIERE : **Electronique.**

**Spécialité : Microélectronique.**

Par

- **Delloul Chahrazed**
- **Bechami Dounia**

*Intitulé*

*Réalisation d'un registre à décalage PISO (parallel input serial output) en  
technologie CMOS 90nm.*

*Par la commission d'évaluation composée de :*

| <i>Nom &amp; Prénom</i>       | <i>Grade</i> | <i>Qualité</i>    | <i>Etablissement</i> |
|-------------------------------|--------------|-------------------|----------------------|
| <i>Mm.Fahima.Fares</i>        | <i>MCA</i>   | <i>Présidente</i> | <i>Univ-BBA</i>      |
| <i>M.Lyes.Dib</i>             | <i>MCB</i>   | <i>Encadreur</i>  | <i>Univ-BBA</i>      |
| <i>M.Abdelouahab.Djemouai</i> | <i>MCB</i>   | <i>Examineur</i>  | <i>Univ-BBA</i>      |

*Année Universitaire 2021/2022*

# Dédicace

**Chahrazed :**

**C'est avec un grand plaisir que je tiens à dédié ce modeste travail :**

**A l'être le plus cher de ma vie, la source de mes efforts et la lumière de notre maison, ma mère « Zohra ».**

**A celui qui a fait de moi une femme instruite, cultivée, autonome, et ambitieuse, mon papa chéri « Kamel », le pilier de la maison.**

**A celles qui me poussent toujours vers l'avant, qui m'épaulent, qui m'aiment d'un amour pur et sincère, et qui m'aident toujours à accomplir mes rêves, mes sœurs « Amel » et « Imene », vous êtes le plus beau cadeau que dieu m'a offert.**

**A mes beaux-frères « Abdelhak » et « Fateh » que je considère comme des vrais frères et que je respecte beaucoup, grâce à vous, je n'ai jamais ressenti d'infériorité dans ma vie. C'est vrai que je n'ai pas de frère, mais le bon dieu m'a béni avec votre fratrie, je suis toujours reconnaissante pour cette bénédiction.**

**A mes neveux et ma petite nièce, « Ismail », « Ayoub », « Youcef », et « Myriam », les chouchous de la famille, que dieu leurs réserve tout le bonheur du monde.**

**Dounia :**

**C'est avec un grand plaisir que je tiens à dédié ce modeste travail :**

**À la chose la plus précieuse que j'ai au monde, maman et papa, je vous aime tellement, et mille merci pour le soutien, l'aide et tout.**

**A ma grand-mère, mes sœurs, mes tantes, mes cousines, tous mes amis et tous ceux qui m'ont soutenu de près ou de loin, merci.**

# Remerciement

Tout d'abord, je remercie dieu le tout puissant de nous avoir donné la chance et la volonté d'entamer et de terminer ce projet.

Ce travail ne serait pas aussi riche et n'aurait pas pu avoir le jour sans l'aide de l'encadrement de **Mr Lyes Dib**, merci beaucoup pour votre grande conscience professionnelle, la qualité de votre encadrement exceptionnel, votre patience, et votre disponibilité durant notre préparation de ce mémoire.

Une reconnaissance très spéciale aux membres de jury, **Mr.Abdelouahab.Djemouai** et **Mm.Fahima.Fares**, pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail.

Je souhaiterais remercier également tout nos professeurs pour le temps que chacun d'eux m'a accordé, et plus globalement pour toutes les informations, références bibliographiques, réflexions, corrections...que chacun m'a apporté et qui ont nourrit ce travail.

Et pour terminer, je n'oublierai pas de remercier le Département d'électronique de l'Université Mohamed El Bachir El Ibrahimi.

# Table des matières

Dédicace

Remerciement

Liste des figures

Liste des tableaux

Résumé

Introduction générale \_\_\_\_\_ 1

## Chapitre 1 : Généralités sur la technologie CMOS

1.1 Introduction \_\_\_\_\_ 3

1.2 Historique de la technologie CMOS \_\_\_\_\_ 4

1.3 Les interrupteurs \_\_\_\_\_ 5

1.3.1 Interrupteur MOS \_\_\_\_\_ 5

1.3.2 Interrupteur CMOS (porte de transmission) \_\_\_\_\_ 7

1.3.3 Mode de fonctionnement d'une porte de transmission \_\_\_\_\_ 7

1.3.4 Les avantages d'une porte de transmission \_\_\_\_\_ 8

1.4 Les différentes portes logiques CMOS \_\_\_\_\_ 8

1.4.1 L'Inverseur CMOS (NOT) \_\_\_\_\_ 8

1.4.2 Porte NAND \_\_\_\_\_ 9

1.4.3 Porte NOR \_\_\_\_\_ 10

1.4.4 Les portes NAND et NOR à trois entrées \_\_\_\_\_ 11

1.4.5 Les portes AND et OR \_\_\_\_\_ 12

1.4.5.1 La porte AND \_\_\_\_\_ 12

1.4.5.2 La porte OR \_\_\_\_\_ 13

1.4.6 La porte XOR \_\_\_\_\_ 14

1.4.7 La porte XNOR \_\_\_\_\_ 15

1.5 Conclusion \_\_\_\_\_ 15

## Chapitre 2 : Registres à décalage

2.1 Introduction \_\_\_\_\_ 16

2.2 Architecture d'un registre à décalage \_\_\_\_\_ 17

2.3 Structure d'un registre à décalage \_\_\_\_\_ 18

|   |  |    |
|---|--|----|
| 2.4                                       | Les types des registres à décalage   | 18 |
| 2.4.1                                     | Registre SISO : Entrée série – Sortie série  | 18 |
| 2.4.2                                     | Registre SIPO : Entrée série-sortie parallèle                                      | 20 |
| 2.4.3                                     | Registre PISO : Entrée en parallèle - Sortie en série                              | 21 |
| 2.4.4                                     | Registre PIPO : Parallel In - Parallel Out   | 22 |
| 2.5                                       | Comment fonctionne un registre à décalage ???                                      | 23 |
| 2.6                                       | Exemples d'applications  | 25 |
| 2.7                                       | Caractéristiques d'un registre   | 26 |
| 2.8                                       | Conclusion   | 27 |
| <b>Chapitre 3 : Layout et simulations</b> |  |    |
| 3.1                                       | Introduction au MICROWIND  | 28 |
| 3.2                                       | Conception et simulation   | 28 |
| 3.2.1                                     | Inverseur CMOS   | 28 |
| 3.2.2                                     | Multiplexeur   | 30 |
| 3.2.3                                     | Structure et analyse du comportement de la bascule D flip-flop                     | 33 |
| 3.2.4                                     | La porte logique NAND  | 35 |
| 3.2.5                                     | Bascule D Flip Flop réalisée par les portes logiques NAND +<br>INVERSEURS          | 36 |
| 3.2.6                                     | Layout complet du registre à décalage PISO (entrées parallèles, sorties<br>séries) | 38 |
| 3.3                                       | Conclusion   | 42 |
| Conclusion générale                       |  | 43 |
| Références bibliographique                |  |    |

# Liste des figures

## Chapitre 1 :

|  |    |
|--|----|
| Figure 1.1 : Vue en coupe des transistors CMOS de type n et de type p réalisés par trois différents procédés de fabrication CMOS. La vue en coupe (a) montre les deux types de transistors dans une technologie CMOS à caisson n, (b) montre ces transistors dans une technologie CMOS à caisson p et (c) dans une technologie CMOS à double caisson. .... | 4  |
| Figure 1.2 : Schéma d'une section transversale de la technologie CMOS. ....  | 5  |
| Figure 1.3 : NMOS comme porte logique. ....  | 5  |
| Figure 1.4 : PMOS comme porte logique. ....  | 6  |
| Figure 1.5 : Réponse transitoire d'un NMOS passant. ....   | 6  |
| Figure 1.6 : Porte de transmission ou interrupteur CMOS, schéma électrique et symbole. ....  | 7  |
| Figure 1.7 : Symbole et implémentation (Inverseur). ....   | 9  |
| Figure 1.8 : Symbole et implémentation (NAND). ....  | 10 |
| Figure 1.9 : Symbole et implémentation (NOR). ....   | 11 |
| Figure 1.10 : Symbole et implémentation (NAND à 3 entrées). ....   | 11 |
| Figure 1.11 : Symbole et implémentation (NOR à trois entrées). ....  | 12 |
| Figure 1.12 : Symbole et implémentation (AND). ....  | 12 |
| Figure 1.13 : Symbole et implémentation (OR). ....   | 13 |
| Figure 1.14 : Symbole de la porte XOR. ....  | 14 |
| Figure 1.15 : Implémentations (XOR). ....  | 14 |
| Figure 1.16 : Symbole de la porte XNOR. ....   | 15 |

## Chapitre 2 :

|   |    |
|---|----|
| Figure 2.1 : Déplacements des données dans les registres à décalage. ....   | 17 |
| Figure 2.2 : Architecture d'un registre à décalage. ....  | 17 |
| Figure 2.3 : Structure d'un registre à décalage. ....   | 18 |
| Figure 2.4 : Un registre à décalage SISO à quatre bits. ....  | 18 |
| Figure 2.5 : Un arrangement pour une lecture non destructive en ajoutant des portes logiques. ....                                  | 19 |
| Figure 2.6 : Registre SIPO en série de quatre bits en parallèle. ....   | 20 |
| Figure 2.7 : Processus du décalage interne du registre SIPO. ....   | 21 |
| Figure 2.8 : Un registre à décalage PISO à quatre bits. ....  | 22 |
| Figure 2.9 : Registre à décalage du type PIPO à quatre bits. ....   | 23 |
| Figure 2.10 : Circuit dans lequel la LED est allumée ou éteinte selon la position de l'interrupteur. ....                           | 24 |
| Figure 2.11 : Circuit dans lequel la LED est commandée à travers une bascule D synchrone. ....                                      | 24 |
| Figure 2.12 : Circuit dans lequel l'information sur l'état de l'interrupteur est transmise à la LED à travers quatre bascules. .... | 25 |

## Chapitre 3 :

|  |    |
|--|----|
| Figure 3.1: Interface du logiciel MICROWIND. ....  | 28 |
| Figure 3.2: Symbole et implémentation (CMOS). .... | 29 |

|  |    |
|--|----|
| Figure 3.3 : Layout d'un inverseur CMOS.....                                       | 29 |
| Figure 3.4: Résultat de la simulation d'un inverseur CMOS.....                     | 30 |
| Figure 3.5: Symbole du multiplexeur 2 vers 1. ....                                 | 30 |
| Figure 3.6: Layout du multiplexeur 2 vers 1. ....                                  | 31 |
| Figure 3.7 : Résultat de simulation du multiplexeur 2 vers 1. ....                 | 31 |
| Figure 3.8 : Porte de transmission. ....   | 32 |
| Figure 3.9 : Layout de la porte de transmission.....                               | 32 |
| Figure 3.10 : Résultat de simulation de la porte de transmission. ....             | 33 |
| Figure 3.11 : Structure d'une bascule D à déclenchement sur front montant. ....    | 33 |
| Figure 3.12 : Layout de la bascule maître esclave. ....                            | 34 |
| Figure 3.13 : Résultat de la simulation d'une bascule D maître esclave .....       | 35 |
| Figure 3.14 : Layout de la porte NAND. ....  | 35 |
| Figure 3.15 : Résultat de la simulation de la porte NAND. ....                     | 36 |
| Figure 3.16 : Schéma complet de la bascule D maître/esclave.....                   | 36 |
| Figure 3.17 : Layout de la bascule maître esclave. ....                            | 37 |
| Figure 3.18 : Résultat de la simulation d'une bascule D maître esclave. ....       | 37 |
| Figure 3.19 : Un registre à décalage PISO à quatre bits. ....                      | 38 |
| Figure 3.20 : Layout du registre à décalage PISO. ....                             | 39 |
| Figure 3.21 : Résultat de la simulation du layout du registre PISO pour 1010 ..... | 40 |
| Figure 3.22 : Résultat de la simulation du layout du registre PISO pour 0011 ..... | 40 |
| Figure 3.23 : Résultat de la simulation du layout du registre PISO pour 0110.....  | 41 |
| Figure 3.24 : Résultat de la simulation du layout du registre PISO pour 0111.....  | 41 |

# Liste des tableaux

|   |    |
|---|----|
| Tableau 1 : Table de vérité de l'inverseur. _____   | 9  |
| Tableau 2 : Table de vérité de la porte NAND. _____   | 10 |
| Tableau 3 : Table de vérité de la porte NOR. _____  | 11 |
| Tableau 4 : Table de vérité de la porte AND. _____  | 13 |
| Tableau 5 : Table de vérité de la porte OR. _____   | 13 |
| Tableau 6 : Table de vérité de la porte XOR. _____  | 14 |
| Tableau 7 : Table de vérité de la porte XNOR. _____   | 15 |
| Tableau 8 : Tableaux qui montrent le décalage à gauche de la valeur binaire (1001) dans le registre (SISO). _____ | 19 |
| Tableau 9 : Tableaux qui montrent la réinitialisation à zéro des bascules. _____                                  | 19 |
| Tableau 10 : Tableaux qui montrent le chargement et le déplacement des données. _____                             | 20 |
| Tableau 11 : La façon dont le nombre binaire de quatre bits 1001 est décalé vers les sorties Q du registre. _____ | 21 |

# Résumé

Le registre à décalage PISO est un registre très utilisé dans plusieurs domaines d'application. Ce registre nous permet d'écrire en parallèle et lire en série, c'est un registre mixte, il est utilisé pour le stockage temporelle de l'information, la conversion parallèle-série, la multiplication par une puissance de 2 (Décalage vers la gauche), la division par une puissance de 2 (Décalage vers la droite)...etc. On le retrouve par exemple dans : les UART (universal asynchronous receiver/transmitter) et les modems. Donc l'étude de ce type de registre est très importante.

Le but de ce travail est l'étude et la réalisation du Layouts (dessins de masque) du registre à décalage PISO en utilisant le logiciel MICROWIND en technologie CMOS 90nm. Donc nous avons réalisé des Layouts pour les blocs qui constituent ce registre : Les quatre bascules D maître/esclave, et les trois multiplexeurs 2 vers 1. Des simulations de ces Layouts ont été faites également de chaque bloc d'une manière séparé. Les résultats obtenus confirment le bon fonctionnement de ces Layouts réalisés. A la fin de ce travail, un Layout complet du registre PISO a été fait avec succès en connectant les différents blocs les uns aux autres.

**Mots clés :** PISO, uart, registre, layout, dessin de masque, microwind, cmos, 90nm, bloc, bascule D, multiplexeur.

# Abstract

The PISO shift register is a register widely used in several fields of application, this register allows us to write in parallel and read in series, it is a mixed register, it is used for the temporal storage of information, the conversion series-parallel, multiplication by a power of 2 (Offset to the left), division by a power of 2 (Offset to the right)... .etc. It can be found for example in: UART (universal asynchronous receiver/transmitter) and modems. So the study of this type of registry is very important.

The purpose of this work is the study and realization of the Layouts (mask drawings) of the PISO shift register using the MICROWIND software in 90nm CMOS technology. So we made Layouts for the blocks that make up this register: The four master/slave D flip-flop, and the three multiplexers 2 to 1. Simulations of these Layouts were also made of each block in a separate way. The results obtained confirm the correct functioning of these layouts. At the end of this work, a complete layout of the PISO register was successfully done by connecting the different blocks to each other.

**Keywords :** PISO, uart, shift register, layout, mask drawings, microwind, cmos, 90nm, block, D flip-flop, multiplexer.

## ملخص

السجل PISO هو سجل يستخدم على نطاق واسع في العديد من مجالات التطبيق، وهذا السجل يسمح لنا بالكتابة بالتوازي والقراءة بالتسلسل، وهو سجل مختلط، ويستخدم للتخزين الزمني للمعلومات، وسلسلة التحويل المتوازية، والضرب بقوة 2 (عوض إلى اليسار)، تقسيم بقوة 2 (عوض إلى اليمين)... الخ. يمكن العثور عليه على سبيل المثال في UART (جهاز استقبال/جهاز إرسال عالمي غير متزامن) وأجهزة مودم. لذا فإن دراسة هذا النوع من السجلات مهمة للغاية.

والغرض من هذا العمل هو دراسة وتنفيذ التخطيطات (رسومات الأقمعة) لسجل PISO باستخدام برنامج MICROWIND في تقنية CMOS 90 نانومتر. لذلك قمنا بعمل تخطيطات للكتل التي تشكل هذا السجل: القلابات الأربعة السيد/السيد D، و الميكتيبليكسور الثلاثة 2 إلى 1. كما تم إجراء محاكاة لهذه التخطيطات لكل كتلة بطريقة منفصلة. تؤكد النتائج التي تم الحصول عليها الأداء الصحيح لهذه التخطيطات. في نهاية هذا العمل، تم إجراء تخطيط كامل لسجل PISO بنجاح من خلال ربط الكتل المختلفة ببعضها البعض.

**الكلمات المفتاحية :** PISO, Uart , السجل , التخطيطات , رسومات الأقمعة , cmos , microwind , 90 نانومتر , الكتل , القلابات D , الميكتيبليكسور .

# **Introduction générale**

# Introduction générale

## Introduction générale

Au cours des dernières années, la technologie CMOS a rapidement sommé le champ des circuits intégrés analogues et mixtes (analogique et numérique). L'utilisation de ce type de technologie permet d'augmenter la complexité des circuits conçus et en général de diminuer aussi la consommation requise tout en améliorant les performances des fonctions réalisées.

Un registre à décalage est un circuit séquentiel constitué d'un ensemble de  $N$  bascules, commandées par le même signal d'horloge, afin de stocker et de déplacer les données à  $N$  bits qu'il reçoit à ses entrées. La capacité d'un registre à stocker les données rend ce composant comme un élément de mémorisation important. [1]

La propriété de décalage d'un registre permet le déplacement des données de l'entrée vers la sortie du registre de différentes manières.

Les registres sont largement utilisés dans les systèmes de traitement numérique (les ordinateurs par exemple) pour réaliser des opérations : de mémorisation provisoire (mémoire tampon), de décalage, de rotation, ...

Il y'a plusieurs types de registres à décalage, mais dans notre étude on va fixer comme but, la réalisation (Conception et simulation) d'un registre à décalage PISO (Parallèle input serial output) qui a plusieurs fonctions très importantes, il s'utilise dans les liaisons séries, et il forme la base des modems et des UART (Universal Asynchronous Receiver/Transmitter), en utilisant la technologie CMOS 90nm et le logiciel de simulation MICROWIND.

On a choisit ce thème parce qu'on voudra bien renforcer nos connaissances par rapport à ce domaine, on trouve que c'est la vraie microélectronique, c'est un thème très impressionnant.

Ce mémoire est organisé en trois chapitres :

- Dans le premier chapitre, nous allons parler sur l'historique de la technologie CMOS, aussi, on va consacrer ce dernier à l'étude des interrupteurs CMOS ainsi qu'une simple présentation par rapport aux différentes portes logiques CMOS : AND, NAND, OR, NOR, NON, XOR, XNOR.

# Introduction générale

- Le deuxième chapitre est réservé à l'étude des différents types des registres à décalage : SISO « Serial input serial output », SIPO « Serial input parallèle output », PISO « Parallèle input serial output », PIPO « Parallèle input parallèle output » en se basant principalement sur leurs structures et leurs applications.
- Pour le troisième chapitre, nous allons réaliser les **Layout** et les **simulations** des blocs suivants : Inverseur, porte de transmissions, multiplexeur. Ensuite, on va rassembler les inverseurs et les portes de transmissions dans un seul Layout pour obtenir une bascule D maître/esclave, et pour terminer, on va rassembler quatre bascules maître/esclave avec trois multiplexeurs 2 vers 1 pour obtenir notre registre de décalage PISO.

Et bien sûr, notre étude sera achevée par une conclusion générale.

# **Chapitre 1**

## **Généralités sur la technologie CMOS**

# Généralités sur la technologie CMOS

## 1.1 Introduction :

La technologie CMOS est une technologie destinée au développement des systèmes à très haute échelle d'intégration (Ou VLSI pour Very-Large-Scale Integration en anglais), qui est une technologie de circuit intégré (CI) dont la densité d'intégration permet de supporter plus de 100 000 composants électroniques sur une même puce électronique. [2]

Grâce aux propriétés des transistors MOS complémentaires (notées CMOS pour Complementary Metal Oxyde Silicon), cette technologie permet de réaliser des circuits à faible coût et à basse consommation. Cet avantage lui a permis d'être reconnue comme la technologie de pointe la plus avancée et la plus maîtrisée dans le domaine de la micro-électronique.

Parmi les types des technologies CMOS, on peut citer les trois qui sont les plus connues : la technologie CMOS à caisson n, la technologie CMOS à caisson p, et les technologies CMOS à double caisson. Une technologie CMOS à caisson n utilise un substrat en silicium de type p dans lequel est formé un caisson en silicium de type n. Des transistors pMOS sont alors réalisés dans ce caisson et des transistors nMOS dans le substrat. Une technologie CMOS à caisson p, permet de réaliser des transistors pMOS sur le substrat de type n et des transistors nMOS dans un caisson p. Enfin, la technologie CMOS à double caisson utilise un substrat en silicium de type quelconque, sur lequel sont déposés des caissons de type N et des caissons de type P.

Les transistors NMOS et PMOS qui peuvent être réalisés par ces technologies sont utilisés pour former des fonctions analogiques ou numériques ; une coupe de ces transistors est donnée sur la figure 1.1. Les procédés de fabrication utilisés dans chaque technologie CMOS pour réaliser ces transistors diffèrent d'un fabricant à un autre et évoluent continuellement pour réaliser des composants plus performants et plus rapides [3].

# Généralités sur la technologie CMOS

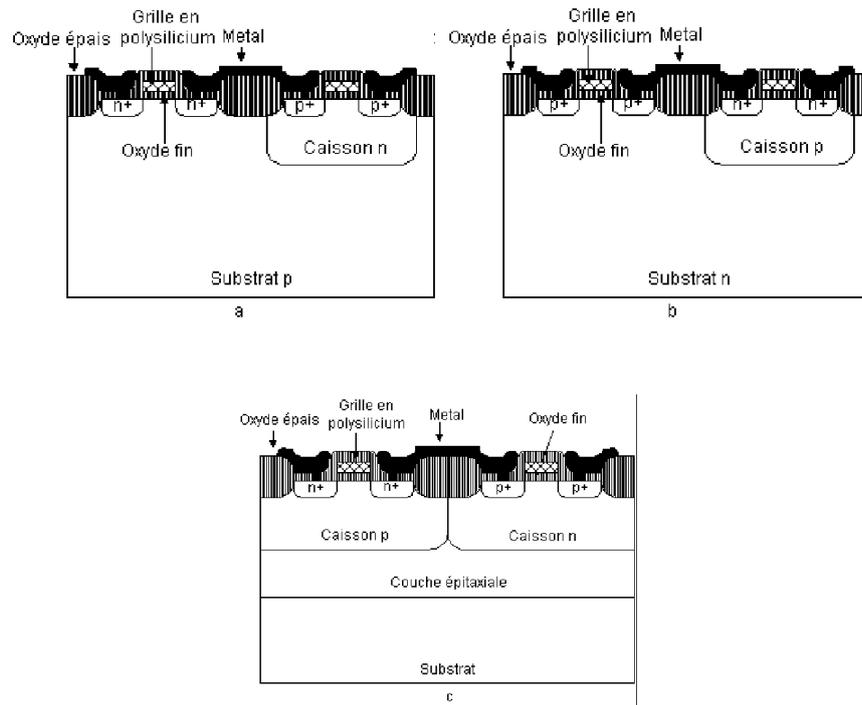


Figure 1.1 : Vue en coupe des transistors CMOS de type n et de type p réalisés par trois différents procédés de fabrication CMOS. La vue en coupe (a) montre les deux types de transistors dans une technologie CMOS à caisson n, (b) montre ces transistors dans une technologie CMOS à caisson p et (c) dans une technologie CMOS à double caisson. [3]

## 1.2 Historique de la technologie CMOS :

Dès 1930, Julius Edgar Lilienfeld de l'Université de Leipzig déposait un brevet dans lequel il décrivait un élément assez proche du transistor MOS et qui aurait pu constituer le premier transistor de l'histoire. Il fallut cependant attendre le début des années 1960, pour voir apparaître les premiers dispositifs MOS puis CMOS industriels, dont le développement avait été rendu possible par les progrès enregistrés dans le domaine des transistors bipolaires et en particulier la résolution des problèmes d'interface oxyde-semi conducteur. [4]

Aujourd'hui le transistor MOS constitue, par sa simplicité de fabrication et ses dimensions, l'élément fondamental des circuits intégrés. [5]

# Généralités sur la technologie CMOS

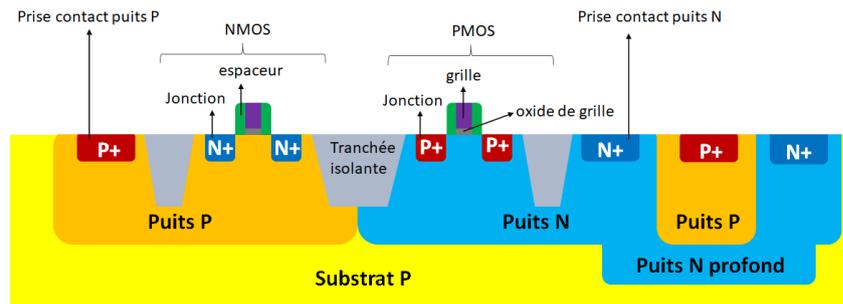


Figure 1.2 : Schéma d'une section transversale de la technologie CMOS. [6]

## 1.3 Les interrupteurs :

### 1.3.1 Interrupteur MOS :

On peut utiliser le MOS comme interrupteur : un NMOS est ON lorsque l'entrée est 1, est OFF quand l'entrée est 0. Un PMOS est ON lorsque l'entrée est 0, et OFF lorsque l'entrée est 1. [8] [9]

\*Un NMOS est un meilleur pour brancher une sortie à la masse (GND) qu'à VDD. Dans le premier cas (figure à gauche), si l'entrée est 1, le condensateur ( $C_L$ ) peut seulement se charger jusqu'à  $V_{DD} - V_{TN}$ , (et l'effet du substrat ne peut pas être négligé). Par contre, si la source du NMOS est reliée à la masse (GND) (figure à droite), le condensateur ( $C_L$ ) peut se décharger complètement lorsque l'entrée est 1. [7]

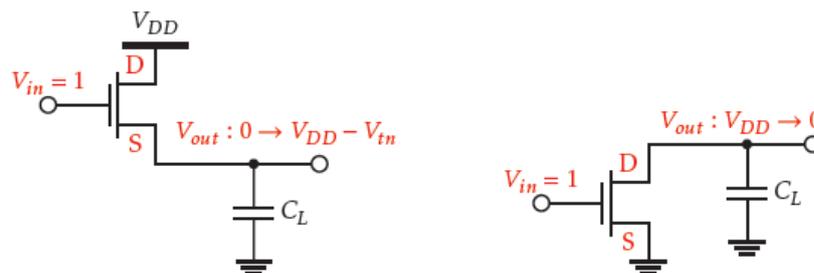


Figure 1.3 : NMOS comme porte logique. [7]

\*Un PMOS est un meilleur pour brancher la sortie à VDD. Dans le premier cas (figure à gauche), le condensateur peut se charger complètement jusqu'à VDD, tandis que le deuxième cas (figure à droite), le condensateur peut se décharger à VTP.

# Généralités sur la technologie CMOS

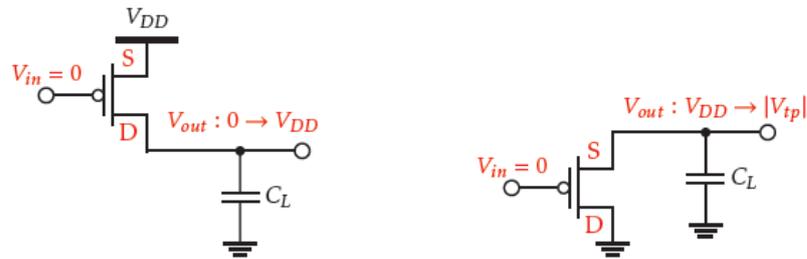


Figure 1.4 : PMOS comme porte logique. [7]

Pour logique passante (effet interrupteur), on utilise un transistor pour passer (ou non) une entrée à une sortie (ou non). On peut utiliser la source et le drain comme des entrées (c'est-à-dire l'entrée n'est pas nécessairement à la grille du transistor). Généralement, si on souhaite utiliser un transistor comme interrupteur, on choisit le NMOS que PMOS, parce que le NMOS est plus rapide que PMOS.

Si on prend l'exemple d'un NMOS en logique passante (interrupteur) qui sert d'entrée d'un inverseur CMOS complémentaire. La sortie (x) peut seulement se charger jusqu'à  $V_{DD}-V_{TN}$ .

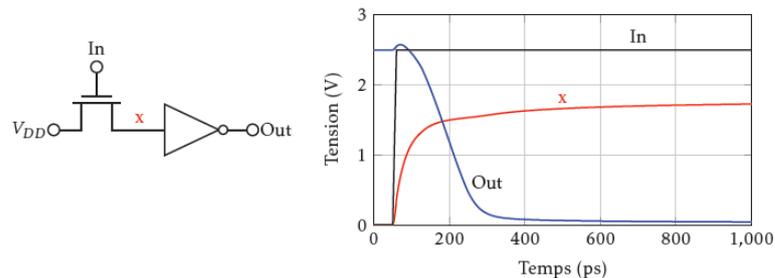


Figure 1.5 : Réponse transitoire d'un NMOS passant.

On ne peut donc pas utiliser plusieurs transistors passant en série : une sortie haute serait détectée comme un 0 après seulement quelques portes. La sortie d'un transistor passant ne doit pas alimenter la grille d'un autre transistor passant.

Pour régler le problème du transistor passant (en logique passant, interrupteur), on peut rajouter un PMOS en parallèle avec le NMOS. Il sert à faire passer correctement un signal à  $V_{DD}$  à la sortie. Donc on obtient un interrupteur de type CMOS complémentaire dite porte de transmission (en anglais Transmission Gate TG). [7]

# Généralités sur la technologie CMOS

## 1.3.2 Interrupteur CMOS (porte de transmission) :

L'interrupteur logique est peut-être réalisé facilement en technologie CMOS en association un transistor NMOS et transistor PMOS en parallèle, commandée par des tensions de grille complémentaire.

Lorsqu'il est actif, le transistor NMOS assure la transmission du signal logique 0 sans dégradation. Lorsqu'il est actif, le transistor PMOS assure la transmission du signal logique 1 sans dégradation. Lorsque les deux transistors sont bloqués, l'interrupteur est ouvert. [7]

## 1.3.3 Mode de fonctionnement d'une porte de transmission :

D'un point de vue logique la porte de transmission présente deux modes de fonctionnement normal :

- Si  $C=0$ ,  $CB=1$ , l'interrupteur est ouvert,
- Si  $C=1$ ,  $CB=0$ , l'interrupteur est fermé.

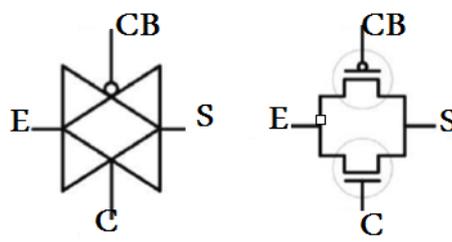


Figure 1.6 : Porte de transmission ou interrupteur CMOS, schéma électrique et symbole.

Dans le cas où ( $CB \neq C$ ) l'un des transistors est passant et l'autre bloqué, la porte de transmission est globalement passante, mais son fonctionnement est dégradé d'un point de vue électrique :

- Lorsque  $C = CB = 1$  (tension VDD), seul le transistor NMOS peut être passant. Il est en régime ohmique si un 0 logique (tension  $V_{SS}=0V$ ) est appliqué en entrée de l'interrupteur et alors  $S = E$ . Mais dans le cas où la tension VDD est appliquée en entrée, le transistor est en régime saturé (à la limite du blocage) et la tension maximale en sortie vaut  $V_{DD}-V_{TN}$ , le niveau de sortie est alors dégradé.

# Généralités sur la technologie CMOS

- Lorsque  $C = CB = 0$  (tension VSS), seul le transistor PMOS peut être passant. Il est en régime ohmique lorsqu'un 1 logique (tension VDD) est appliqué en entrée de l'interrupteur, mais bloqué si on applique VSS en entrée. Il faut appliquer une tension au moins égale à  $-V_{TP}$  pour que l'interrupteur devienne passant et que  $S = E$ .

L'association des deux types de transistors, avec des commandes complémentaires, permet de masquer le défaut de chacun d'eux. Les signaux appliqués à l'entrée de l'interrupteur CMOS sont restitués en sortie sans dégradation.

## 1.3.4 Les avantages d'une porte de transmission :

- Peut transmettre le niveau logique 1 et 0 sans dégradation.
- Déduire le nombre de transistor dans les circuits logiques. Par exemple, une porte XOR à 2 entrées réalisées avec des portes de transmission ne nécessite que 6 transistors, comparativement à 12 en CMOS complémentaire.

## 1.4 Les différentes portes logiques CMOS :

### 1.4.1 L'Inverseur CMOS (NOT) :

On associe sur un même substrat deux transistors complémentaires, un canal N et un canal P (Canaux diffusés) dont les grilles sont reliées.

L'inverseur CMOS est constitué de deux transistors, un nMOS et un pMOS, connectés en série entre l'alimentation qui constitue le 1 logique et la masse qui représente le 0 logique.

Si l'entrée est au niveau logique 1 ( $E = +V_{cc}$ ) le transistor P1 (canal P) est bloqué et le transistor N2 (Canal N) est saturé. Le potentiel de la sortie S est donc égal à 0 qui correspond au niveau logique 0.

Au contraire, quand l'entrée est au niveau logique 0 ( $E=0$ ), le transistor P1 est saturé et le transistor N2 est bloqué. La sortie S est au potentiel  $+V_{cc}$ .

On réalise ainsi un inverseur logique. En dehors des périodes de transition un seul transistor est passant et de ce fait aucun courant ne circule entre  $V_{cc}$  et la masse. C'est pour cette raison

# Généralités sur la technologie CMOS

que la consommation des circuits est aussi faible. Lors des périodes de transition, il apparaît une impulsion de courant entre Vcc et la masse. Elle est due à la charge des condensateurs équivalents aux entrées des transistors suivants et des capacités parasites des connexions. Si la consommation est pratiquement nulle en basse fréquence, elle croît avec la vitesse de fonctionnement du circuit. [10]

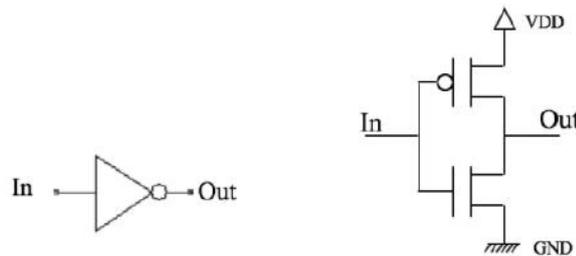


Figure 1.7 : Symbole et implémentation (Inverseur).

| In | Out |
|----|-----|
| 0  | 1   |
| 1  | 0   |

Tableau 1 : Table de vérité de l'inverseur.

Le transistor NMOS est passant si sa Grille est à 1.

Le transistor PMOS est passant si sa Grille est à 0.

## 1.4.2 Porte NAND :

C'est la plus simple à réaliser du point de vue technologique. La porte NAND prend en entrée deux valeurs 0 ou 1. La sortie vaut 0 si les deux entrées valent 1 et elle vaut 1 si au moins une des deux entrées vaut 0. La table de vérité et le schéma d'implémentation sont présentés ci-dessous :

# Généralités sur la technologie CMOS

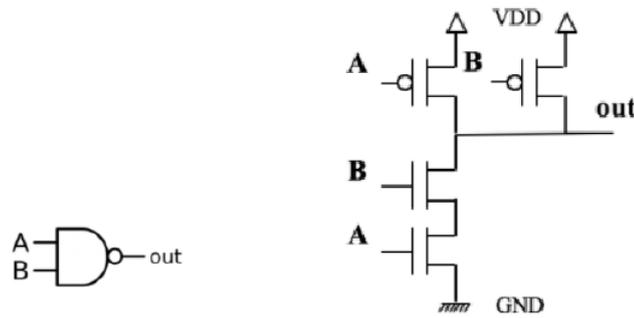


Figure 1.8 : Symbole et implémentation (NAND).

| A | B | $\overline{A \cdot B}$ |
|---|---|------------------------|
| 0 | 0 | 1                      |
| 0 | 1 | 1                      |
| 1 | 0 | 1                      |
| 1 | 1 | 0                      |

Tableau 2 : Table de vérité de la porte NAND.

On associe deux P en parallèle avec deux N en série. La sortie est à 0 uniquement quand les deux N sont saturés.

Pour réaliser une porte à n entrées, on associe n P en parallèle avec n N en série.

Il est possible de réaliser toutes les fonctions logiques en utilisant uniquement des NAND.

## 1.4.3 Porte NOR:

On associe deux N en parallèle avec deux P en série. La sortie est à 1 uniquement quand les deux P sont saturés. Pour réaliser une porte à n entrées, on associe n N en parallèle avec n P en série.

Le circuit pour réaliser la porte NOR en logique CMOS est donné sur la figure ci-dessous. Il est constitué de quatre transistors dont deux nMOS et deux pMOS. C'est le circuit dual du circuit de la porte NAND. Les deux transistors pMOS qui étaient en parallèles dans le circuit de la porte NAND sont en série dans le circuit de la porte NOR. Au contraire, les deux

# Généralités sur la technologie CMOS

transistors nMOS qui étaient en série dans le circuit de la porte NAND sont en parallèles dans le circuit de la porte NOR. [11]

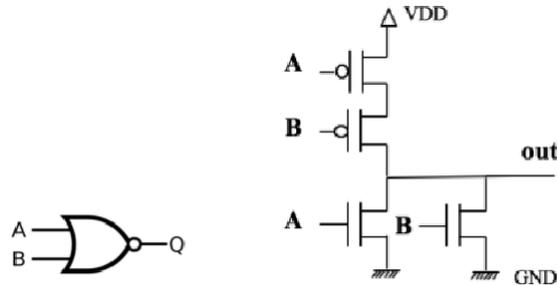


Figure 1.9 : Symbole et implémentation (NOR).

| A | B | $\overline{A + B}$ |
|---|---|--------------------|
| 0 | 0 | 1                  |
| 0 | 1 | 0                  |
| 1 | 0 | 0                  |
| 1 | 1 | 0                  |

Tableau 3 : Table de vérité de la porte NOR.

## 1.4.4 Les portes NAND et NOR à trois entrées :

Dans le même esprit que les portes à deux entrées, celles à trois entrées sont composées de 3 transistors PMOS et trois transistors NMOS connectés en parallèle ou en série, comme indiqué sur la figure ci-dessous :

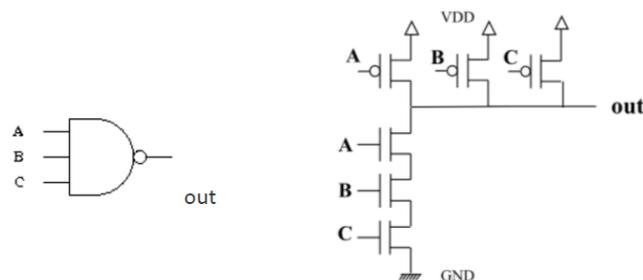


Figure 1.10 : Symbole et implémentation (NAND à 3 entrées).

# Généralités sur la technologie CMOS

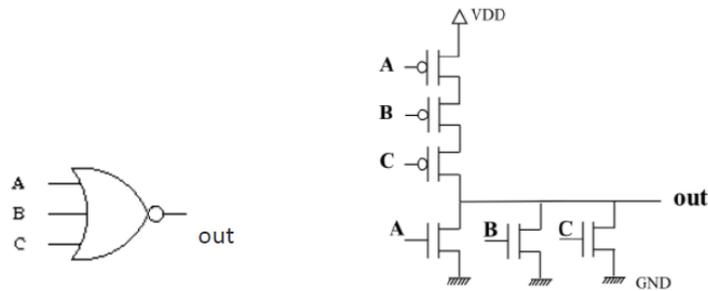


Figure 1.11 : Symbole et implémentation (NOR à trois entrées).

## 1.4.5 Les portes AND et OR :

Les circuits des portes AND et OR sont respectivement obtenus en combinant un circuit de la porte NAND et NOR avec un inverseur. Les portes AND et OR requièrent, étrangement, plus de transistors que les portes NAND ou NOR. En effet on remarque sur les schémas des figures suivantes, que chacune de ces portes contient 6 transistors.

### 1.4.5.1 La porte AND :

La porte AND prend en entrée deux valeurs 0 ou 1. La sortie vaut 1 si les deux entrées valent 1 et elle vaut 0 sinon. On fait suivre la porte NAND d'un inverseur. La table de vérité et le schéma d'implémentation sont présentés ci-dessous :

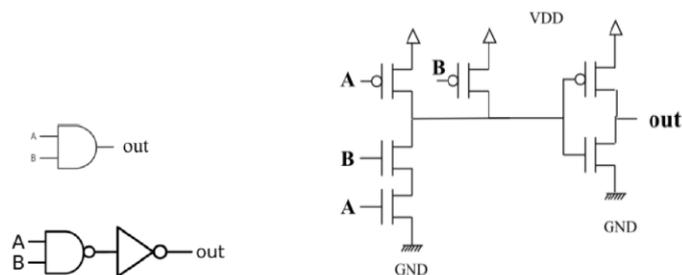


Figure 1.12 : Symbole et implémentation (AND).

# Généralités sur la technologie CMOS

| A | B | A.B |
|---|---|-----|
| 0 | 0 | 0   |
| 0 | 1 | 0   |
| 1 | 0 | 0   |
| 1 | 1 | 1   |

Tableau 4 : Table de vérité de la porte AND.

## 1.4.5.2 La porte OR :

La porte OR prend en entrée deux valeurs 0 ou 1. La sortie vaut 0 si les deux entrées valent 0 et elle vaut 1 sinon. On fait suivre la porte NOR d'un inverseur. La table de vérité et le schéma d'implémentation sont présentés ci-dessous :

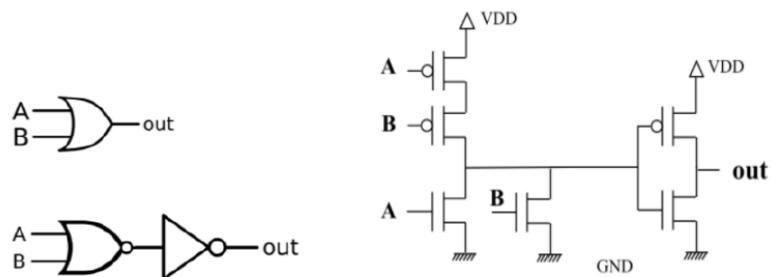


Figure 1.13 : Symbole et implémentation (OR).

| A | B | A+B |
|---|---|-----|
| 0 | 0 | 0   |
| 0 | 1 | 1   |
| 1 | 0 | 1   |
| 1 | 1 | 1   |

Tableau 5 : Table de vérité de la porte OR.

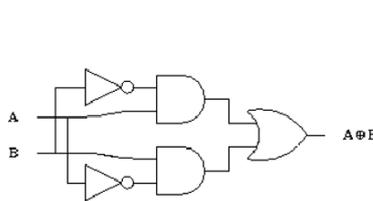
# Généralités sur la technologie CMOS

## 1.4.6 La porte XOR :

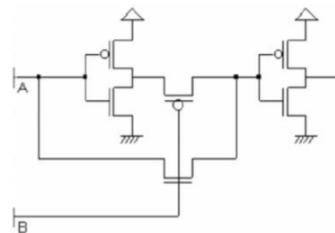
La porte XOR permet de réaliser la fonction ou exclusif qui s'avère très utile pour construire les additionneurs. C'est un assemblage de portes logiques (2 inverseurs, 2 AND et 1 OR) qui nous conduit à un circuit à 22 transistors (11 NMOS et 11 PMOS). Si on souhaite économiser un nombre important de transistors, la structure la plus économe sera celle à 6 transistors. Malheureusement, cette économie se fera au détriment de la qualité des signaux de sortie. En effet, avec un simulateur analogique, vous remarquerez que vos signaux n'atteignent pas de bons niveaux logiques.



Figure 1.14 : Symbole de la porte XOR.



Design complexe (22 MOS)



Design optimisé (6 transistors MOS)

Figure 1.15 : Implémentations (XOR).

| A | B | $A \oplus B$ |
|---|---|--------------|
| 0 | 0 | 0            |
| 0 | 1 | 1            |
| 1 | 0 | 1            |
| 1 | 1 | 0            |

Tableau 6 : Table de vérité de la porte XOR.

# Généralités sur la technologie CMOS

## 1.4.7 La porte XNOR :

La porte XNOR (EXNOR ou même NXOR) est une porte logique numérique dont la fonction est le complément logique de la porte XOR. La porte est surnommée « Porte d'équivalence » car la sortie vaut 1 si les deux entrées sont identiques et elle vaut 0 sinon.

Le symbole et la table de vérité de la porte XNOR sont illustrés ci-dessous.

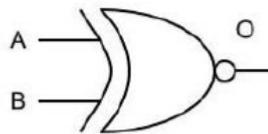


Figure 1.16 : Symbole de la porte XNOR.

| A | B | A $\oplus$ B |
|---|---|--------------|
| 0 | 0 | 1            |
| 0 | 1 | 0            |
| 1 | 0 | 0            |
| 1 | 1 | 1            |

Tableau 7 : Table de vérité de la porte XNOR.

## 1.5 Conclusion :

Dans ce chapitre, nous avons donné une généralité sur la technologie CMOS ainsi que son historique, nous avons parlé sur les interrupteurs MOS et CMOS, également sur le mode de fonctionnement d'une porte de transmission, et on a achevé par une étude détaillée sur les portes logiques CMOS.

# **Chapitre 2**

## **Registres à décalage**

## 2.1 Introduction :

Les registres à décalage sont des circuits logiques séquentiels, principalement pour le stockage de données numériques. Ils sont un groupe de bascules connectés dans une chaîne de sorte que la sortie d'une bascule devient l'entrée de la bascule suivante. Les données peuvent être écrites/lues en même temps (parallèle) ou une après l'autre (série). Le nombre de bits du registre correspond au nombre de cellules mémoire (nombre de bascule D ou JK) du registre. Tous les flip-flops sont pilotés par une horloge commune, et tous sont réglés ou réinitialisés simultanément. Un registre à décalage permet de stocker et de déplacer les données à N bits qu'il reçoit à ses entrées. La capacité d'un registre à stocker les données rend ce composant comme un élément de mémorisation important. La propriété de décalage d'un registre permet le déplacement des données de l'entrée vers la sortie du registre de différentes manières. [12]

Dans ce chapitre, on va citer les types de base des registres à décalage qui seront étudiés, tels que :

- ❖ Les registres à entrées parallèles et sorties parallèles : PIPO (Parallel IN-Parallel OUT).
- ❖ Les registres à entrées parallèles et sorties séries : PISO (Parallel IN-Serial OUT).
- ❖ Les registres à entrées séries et sorties parallèles : SIPO (Serial IN- Parallel OUT).
- ❖ Les registres à entrées séries et sorties séries : SISO (Serial IN- Serial OUT).

Ensuite, nous allons connaître le principe de fonctionnement de chaque type.

La figure ci-dessous illustre les différents déplacements des données dans un registre à décalage.

# Registres à décalage

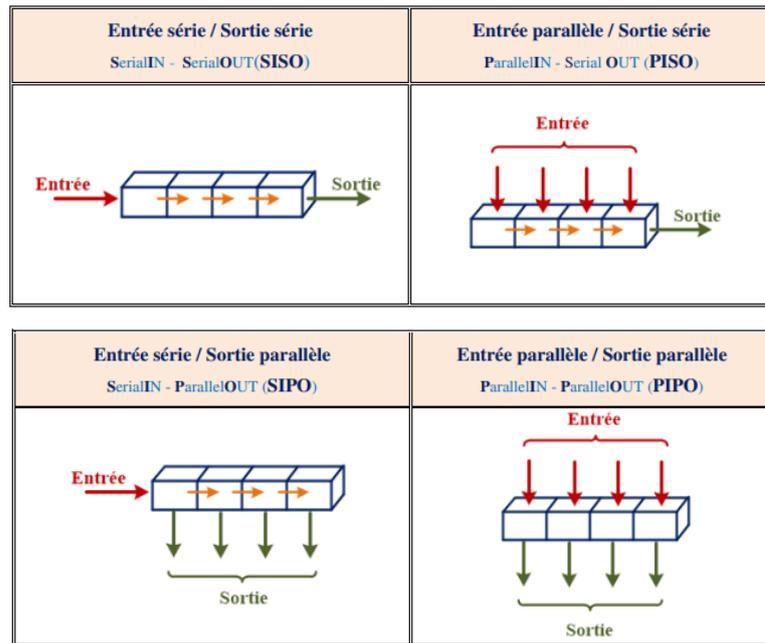


Figure 2.1: Déplacements des données dans les registres à décalage.

Ces quatre types peuvent être classés en deux catégories : les registres de mémorisation (tampon) et les registres à décalage.

## 2.2 Architecture d'un registre à décalage :

Le registre à décalage est construit tous simplement par l'assemblage des bascules D (Latch)  
Donc il est simple à réaliser.

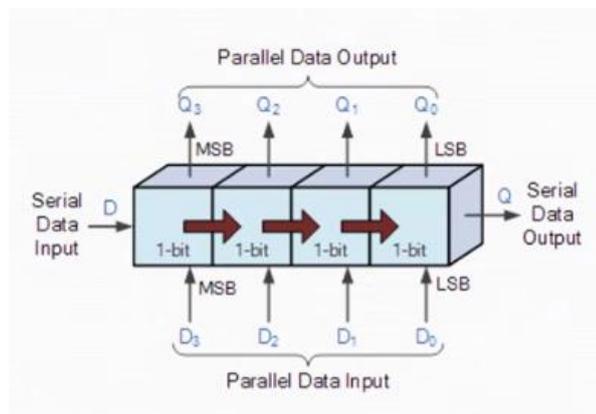


Figure 2.2 : Architecture d'un registre à décalage.

## 2.3 Structure d'un registre à décalage :

La structure du registre à décalage va dépendre de la façon dont on introduit les bits d'information, c'est à dire du chargement du registre. Ce chargement peut être sous forme série, c'est à dire les bits sont introduits les uns après les autres ou sous forme parallèle, c'est-à-dire simultanément.

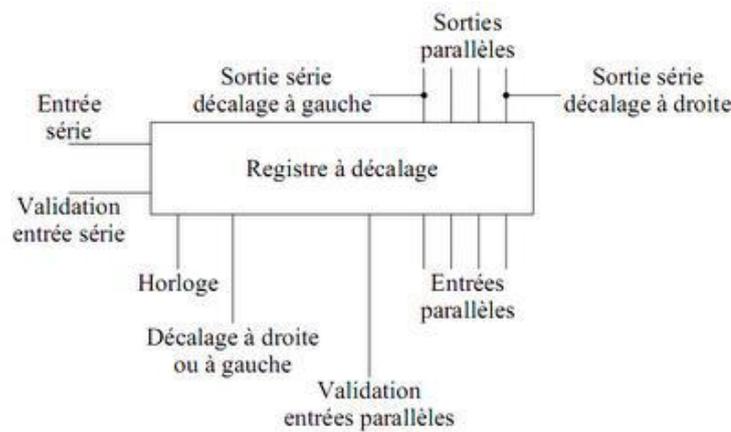


Figure 2.3 : Structure d'un registre à décalage.

## 2.4 Les types des registres à décalage :

### 2.4.1 Registre SISO : Entrée série – Sortie série :

Un registre à décalage basique à quatre bits peut être construit à l'aide de quatre bascules D, comme indiqué ci-dessous. Le fonctionnement du circuit est le suivant : Le registre est d'abord effacé, forçant les quatre sorties à zéro. Les données d'entrée sont ensuite appliquées séquentiellement à l'entrée D du premier flip-flop à gauche (FFA). Pendant chaque impulsion d'horloge, un bit est transmis de gauche à droite.

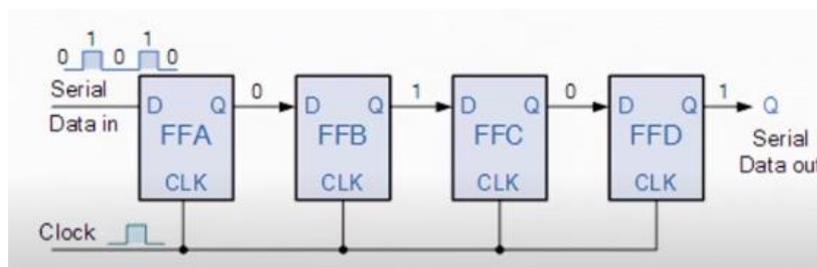


Figure 2.4 : Un registre à décalage SISO à quatre bits.

# Registres à décalage

**Exemple** : on prend la valeur binaire (1001)

Voilà ce qui se passe dans le registre (SISO) : exemple d'un décalage à gauche



Tableau 8 : Tableaux qui montrent le décalage à gauche de la valeur binaire (1001) dans le registre (SISO).

Afin de retirer les données du registre, ils doivent être décalés en série. Cela peut être fait de manière destructrice ou non-destructive. Pour une lecture destructive, les données d'origine sont perdues et à la fin du cycle de lecture, toutes les bascules sont réinitialisées à zéro.

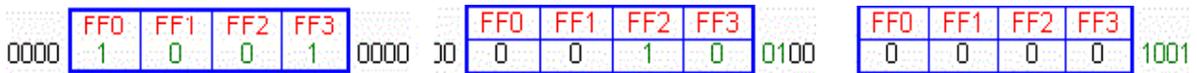


Tableau 9 : Tableaux qui montrent la réinitialisation à zéro des bascules.

Pour éviter la perte de données, un arrangement pour une lecture non destructive peut être effectué en ajoutant deux portes ET, une porte OU et un onduleur au système. La construction de ce circuit est illustrée ci-dessous.

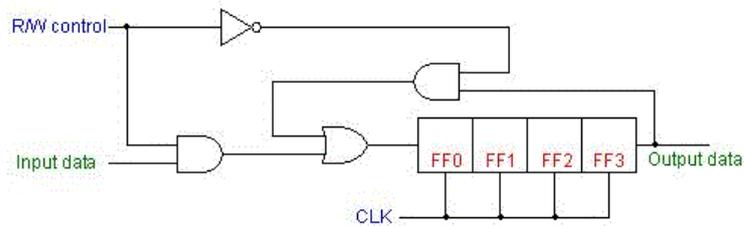


Figure 2.5 : Un arrangement pour une lecture non destructive en ajoutant des portes logiques.

Les données sont chargées dans le registre lorsque la ligne de commande est HAUTE (C'est à dire ÉCRITURE). Les données peuvent être déplacées hors du registre lorsque la ligne de commande est BAS (c'est-à-dire READ). Ceci est montré dans les tableaux ci-dessous.

# Registres à décalage

|      |     |     |     |     |      |
|------|-----|-----|-----|-----|------|
| READ | FF0 | FF1 | FF2 | FF3 |      |
|      | 1   | 0   | 0   | 1   | 1001 |

|      |     |     |     |     |      |
|------|-----|-----|-----|-----|------|
| READ | FF0 | FF1 | FF2 | FF3 |      |
|      | 0   | 1   | 1   | 0   | 0100 |

|      |     |     |     |     |      |
|------|-----|-----|-----|-----|------|
| READ | FF0 | FF1 | FF2 | FF3 |      |
|      | 0   | 0   | 1   | 1   | 0010 |

|       |     |     |     |     |  |
|-------|-----|-----|-----|-----|--|
| CLEAR | FF0 | FF1 | FF2 | FF3 |  |
| 1001  | 0   | 0   | 0   | 0   |  |

|       |     |     |     |     |  |
|-------|-----|-----|-----|-----|--|
| WRITE | FF0 | FF1 | FF2 | FF3 |  |
| 1001  | 0   | 0   | 0   | 0   |  |

|       |     |     |     |     |      |
|-------|-----|-----|-----|-----|------|
| WRITE | FF0 | FF1 | FF2 | FF3 |      |
|       | 1   | 0   | 0   | 1   | 0000 |

Tableau 10 : Tableaux qui montrent le chargement et le déplacement des données.

## 2.4.2 Registre SIPO : Entrée série-sortie parallèle :

Pour ce type de registre, les bits de données sont entrés en série de la même manière que celle décrite dans la section précédente. La différence est la façon dont les bits de données sont retirés du registre. Une fois les données stockées, chaque bit apparaît sur sa ligne de sortie respective, et tous les bits sont disponibles simultanément. Une construction d'un registre en série de quatre bits en parallèle est illustrée ci-dessous.

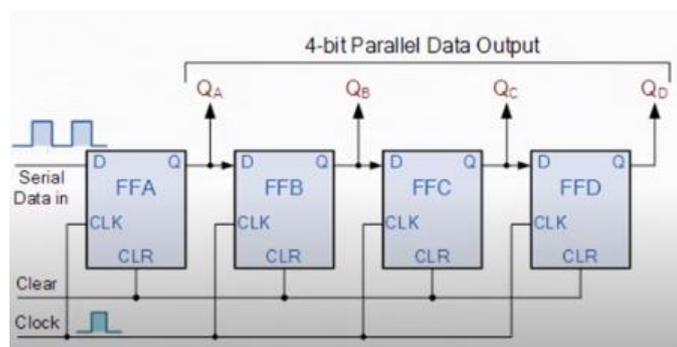


Figure 2.6 : Registre SIPO en série de quatre bits en parallèle.

Dans les tableaux ci-dessous, on peut voir comment le nombre binaire de quatre bits 1001 est décalé vers les sorties Q du registre.

# Registres à décalage

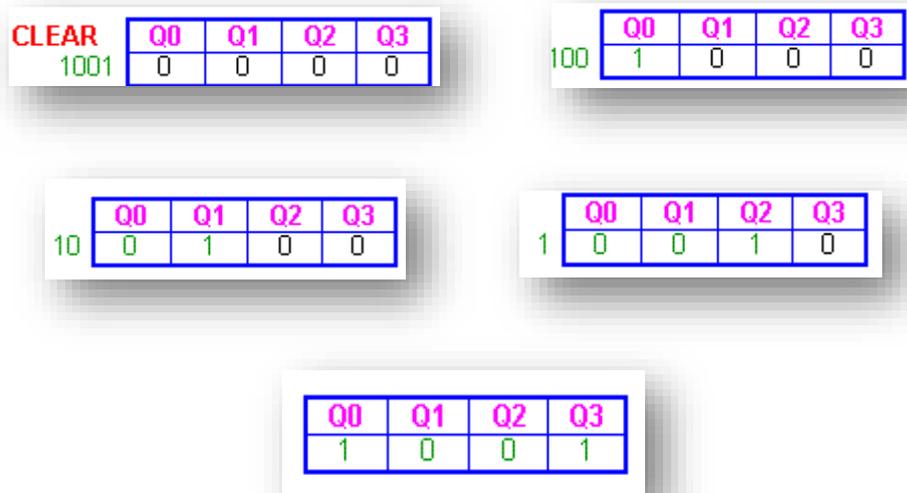


Tableau 11 : La façon dont le nombre binaire de quatre bits 1001 est décalé vers les sorties Q du registre.

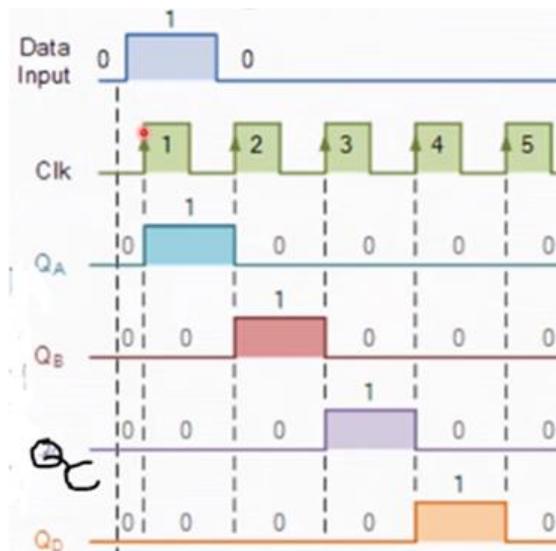


Figure 2.7 : Processus du décalage interne du registre SIPO.

## 2.4.3 Registre PISO : Entrée en parallèle - Sortie en série :

Ce registre à décalage permet une entrée parallèle et génère une sortie série, c'est pourquoi on parle de registre à décalage parallèle en sortie série (PISO). Le circuit du registre de décalage parallèle en sortie série (PISO) est illustré dans la figure 2.8, ce circuit peut être construit avec quatre bascules D, où le signal CLK est connecté directement à tous les FF. Cependant, les

# Registres à décalage

données d'entrée sont connectées séparément à chaque FF à l'aide d'un multiplexeur à chaque entrée de FF.

Dans ce cas il faut appliquer les données aux entrées synchrones  $D_i$ . En fonction de l'ordre de chargement (écriture)  $W$ , chaque bascule recopie l'entrée  $E_i$  ou bien la sortie de la bascule  $i-1$ , à chaque front d'horloge. Il faut donc utiliser un multiplexeur 2 vers 1 à l'entrée  $D_i$  de chacune des bascules 1 à  $n-1$  comme le montre le circuit ci-dessous qui représente un registre 4 bits.

$$D_0 = E_0$$

$$D_i = Z_i = Q_{i-1} \overline{W} + E_i W$$

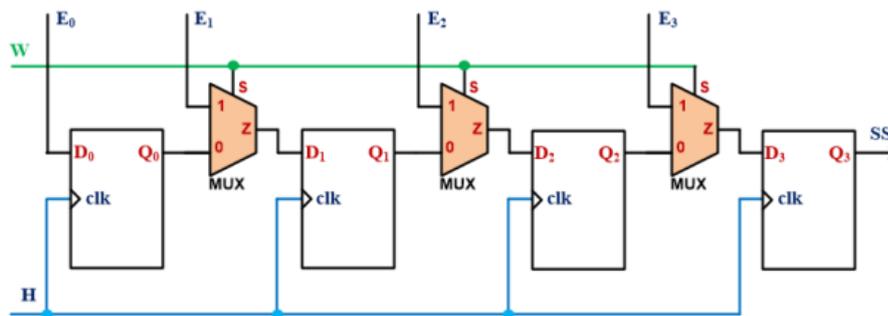


Figure 2.8 : Un registre à décalage PISO à quatre bits.

## 2.4.4 Registre PIPO : Parallel In - Parallel Out :

Pour les registres à décalage parallèles en parallèle, tous les bits de données apparaissent sur les sorties parallèles immédiatement après l'entrée simultanée des bits de données. Le circuit suivant est un registre à décalage parallèle IN parallèle OUT à quatre bits construit par des bascules D.

# Registres à décalage

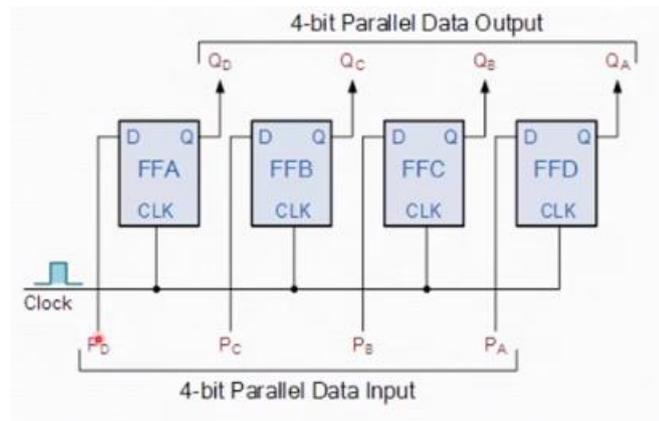


Figure 2.9 : Registre à décalage du type PIPO à quatre bits.

Les D sont les entrées parallèles et les Q sont les sorties parallèles. Une fois le registre reçoit l'impulsion d'horloge, toutes les données aux entrées D apparaissent aux sorties Q correspondantes simultanément.

## 2.5 Comment fonctionne un registre à décalage ???

- Dans cette partie, nous allons examiner les circuits à décalage appelés aussi registres à décalage.
- Ils sont très utilisés comme circuit de temporisation, comme circuit de mémoire et de traitement de l'information.
- Une application importante des registres à décalage est la transmission série de données logiques. [13]

Pour comprendre le fonctionnement des registres à décalage, en particulier le registre à entrée série et sortie série, nous allons prendre en considération le circuit très simple de la figure suivante :

# Registres à décalage

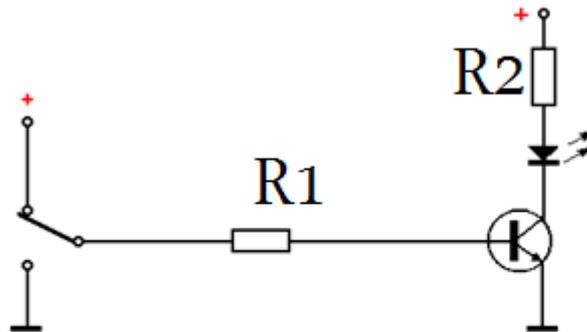


Figure 2.10 : Circuit dans lequel la LED est allumée ou éteinte selon la position de l'interrupteur.

Nous déduisons, en observant cette figure, que la LED s'allume si l'interrupteur est commuté vers le haut, ce qui fait conduire le transistor.

Par contre, la LED est éteinte si l'interrupteur est commuté vers le bas ce qui bloque le transistor.

Ainsi, la LED suit fidèlement et immédiatement les commandes provenant de l'interrupteur ; chaque ouverture ou fermeture de celui-ci produit un effet instantané sur l'état de la LED.

Autrement dit, l'information sur l'état de l'interrupteur est transmise immédiatement (mis à part le temps de commutation du transistor).

Observons maintenant le circuit ci-dessous. Une bascule synchrone de type D est intercalée entre l'interrupteur et la LED.

# Registres à décalage

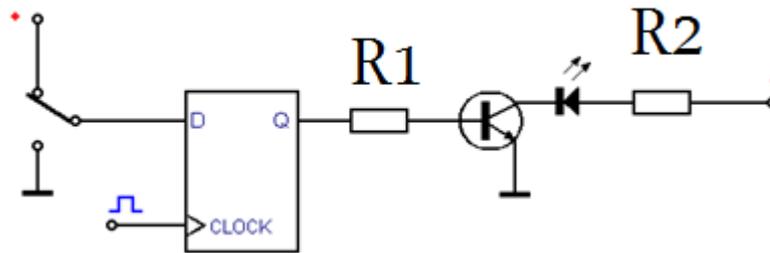


Figure 2.11 : Circuit dans lequel la LED est commandée à travers une bascule D synchrone.

Dans ce cas, l'information sur l'état de l'interrupteur n'est pas transmise immédiatement à la LED, car il est nécessaire d'appliquer une impulsion sur l'entrée CLOCK.

La sortie Q de la bascule se met à l'état déterminé par la position de l'interrupteur chaque fois qu'une impulsion est appliquée sur l'entrée CLOCK.

Nous sommes ainsi renseignés sur l'état de l'interrupteur avec un certain retard.

Modifions à présent le circuit en faisant passer le nombre des bascules à quatre comme illustré à la figure suivante :

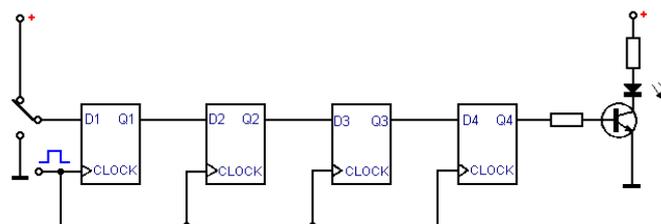


Figure 2.12 : Circuit dans lequel l'information sur l'état de l'interrupteur est transmise à la LED à travers quatre bascules.

Dans ce nouveau circuit, les bascules sont reliées en cascade ; la sortie de l'une est reliée à l'entrée de la suivante.

Les entrées CLOCK, par contre, sont toutes reliées entre elles. Ainsi, une unique entrée d'horloge commande les quatre bascules simultanément.

# Registres à décalage

Puisque quatre bascules sont intercalées entre l'interrupteur et la LED, il faut donc quatre impulsions d'horloge pour transmettre l'information sur l'état de l'interrupteur de l'entrée à la sortie du circuit où se trouve la LED.

La première impulsion transfère l'information de l'entrée à la sortie de la première bascule, la seconde la transmet à la sortie de la deuxième bascule et ainsi de suite jusqu'à la quatrième.

L'information se décale donc en se propageant de l'entrée de la première bascule à la sortie de la quatrième bascule au bout de quatre impulsions d'horloge.

Le circuit de la figure 2.12 constitue un registre à décalage.

## 2.6 Exemples d'applications :

**SISO** : L'information que l'on veut introduire dans le registre est présentée à l'entrée de la première bascule. Lors d'une impulsion d'horloge, le bit d'information est introduit dans le registre, et tous les autres bits sont décalés. Le bit qui était mémorisé dans la dernière bascule est perdu s'il n'est pas stocké ou réinséré dans la structure d'une manière quelconque. Les registres SISO sont utilisés pour réaliser des lignes à retard numériques. Le délai entre l'entrée de l'information dans le registre et sa sortie dépend du nombre de bascules et de la fréquence d'horloge.

**PIPO** : En décalant tous les bits d'un nombre binaire vers la droite ou vers la gauche, on divise ou on multiplie le nombre par 2. Un registre PIPO peut donc être utilisé pour effectuer des calculs (multiplication ou division par une puissance de 2). Il suffit d'opérer le nombre adéquat de décalages vers la gauche ou la droite entre le moment où l'on introduit les bits dans le registre et le moment où on les récupère.

**PISO et SIPO** : Ces deux types de registres sont utilisés dans les liaisons série ; ils forment la base des UART et des modems. Imaginons que l'on veuille transmettre une information entre deux ordinateurs distants de quelques mètres ou dizaines de mètres. Transmettre l'information sous forme « parallèle » nécessiterait au moins 9 fils (8 pour les 8 bits, un pour la masse), sans compter les fils supplémentaires pour le dialogue entre les ordinateurs. Il est plus simple d'employer un registre PISO pour envoyer les bits

# Registres à décalage

constituant chaque octet que l'on désire transmettre en une suite de 8 bits apparaissant l'un après l'autre sur une seule ligne. Au bout de la ligne, un registre SIPO reçoit les bits qui arrivent à la queue-leu-leu et reconstitue des octets qui sont transmis à l'ordinateur de destination. [14]

## 2.7 Caractéristiques d'un registre :

Tout registre est caractérisé par :

La capacité : nombre de bits du mot binaire qu'il peut mémoriser.

Le mode d'écriture ou de chargement: dépend du nombre d'entrées :

- Écriture série : génération bit par bit, avec transmission par un seul fil conducteur.
- Écriture parallèle : génération globale du mot de n bits, avec transmission par un bus de n bits (n fils conducteurs).

Le mode de lecture:

- Lecture série : exploitation bit par bit du mot (une seule sortie).
- Lecture parallèle : exploitation globale du mot (n sorties).

## 2.8 Conclusion :

Dans ce chapitre, nous avons présenté les différents types des registres à décalage ainsi que leurs caractéristiques, nous avons également donné des exemples d'application par rapport à ces registres, et on a expliqué leur principe de fonctionnement.

# **Chapitre 3**

## **Layout et simulations**

## 3.1 Introduction au MICROWIND :

Le programme MICROWIND permet à l'utilisateur de concevoir et de simuler un circuit intégré au niveau de la description physique. Le package contient une bibliothèque de circuits intégrés logiques et analogiques à visualiser et à simuler. MICROWIND comprend toutes les commandes d'un éditeur de masques ainsi que des outils originaux jamais réunis auparavant dans un seul module (vue de processus 2D et 3D, Verilog compiler, tutorial sur les dispositifs MOS). On peut accéder à la simulation de circuit en appuyant sur une seule touche. L'extraction électrique de notre circuit est automatiquement effectuée et le simulateur analogique produit immédiatement des courbes de tension et de courant. [11]

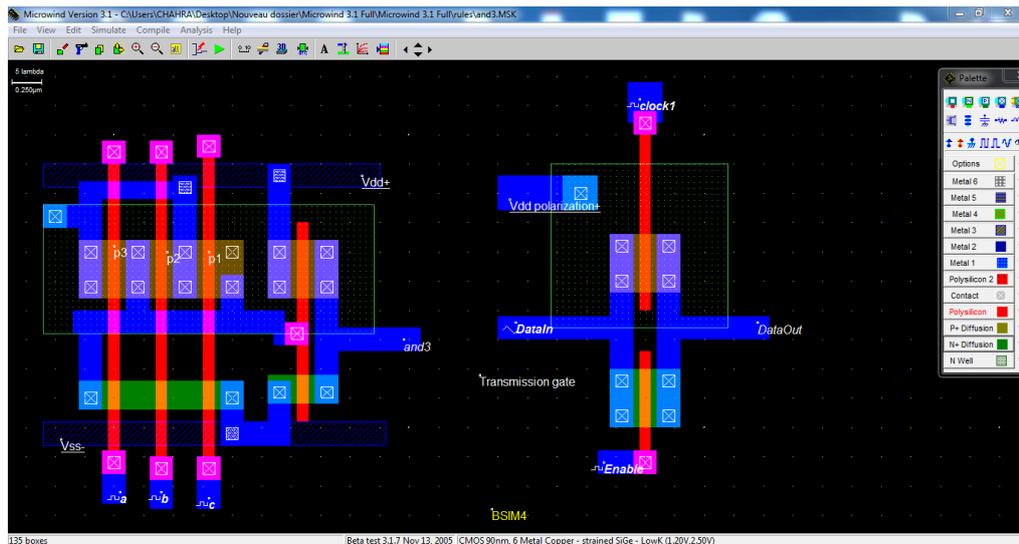


Figure 3.1 : Interface du logiciel MICROWIND. [11]

## 3.2 Conception et simulation :

### 3.2.1 Inverseur CMOS :

L'inverseur CMOS est composé d'un transistor NMOS et d'un transistor PMOS qui est monté en tête-bêche ; ou les grilles des deux transistors sont reliées entre elles et avec l'entrée. Il est considéré comme l'élément de base de tous les circuits intégrés utilisés pour l'implémentation des circuits numériques/analogique.

# Layout et simulations

Il y'a deux états logiques correspondent respectivement à l'état haut (tension la plus élevée Vdd) et l'état bas (tension la plus basse, soit  $V_{ss}=0$ ). Généralement, l'état haut correspond au 1 de l'algèbre de Boole et l'état bas au 0. L'état haut est obtenu en sortie lorsque l'interrupteur de type p est fermé et l'interrupteur de type NMOS est ouvert. Ceci est obtenu, compte tenu du mode de fonctionnement des transistors N et PMOS indiqué par la figure ci-dessous, lorsque l'entrée est à l'état bas. L'état bas est obtenu en sortie lorsque l'interrupteur NMOS est fermé et l'interrupteur PMOS est ouvert (entrée à l'état haut). Les deux états haut et bas en sortie sont obtenus via un interrupteur fermé. Ils correspondent donc exactement aux deux tensions Vdd et  $V_{ss}=0$  fournies par l'alimentation. On constate également qu'un seul interrupteur est fermé à la fois, ce qui signifie qu'il n'y a pas de courant circulant entre les deux bornes de l'alimentation. Dans son principe, la logique CMOS statique ne dissipe pas de puissance en statique, c'est-à-dire lorsque les signaux d'entrée ne varient pas. [16]

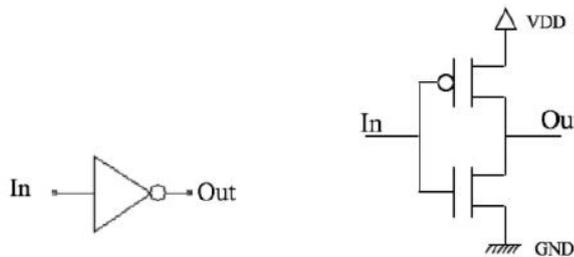


Figure 3.2 : Symbole et implémentation (CMOS).

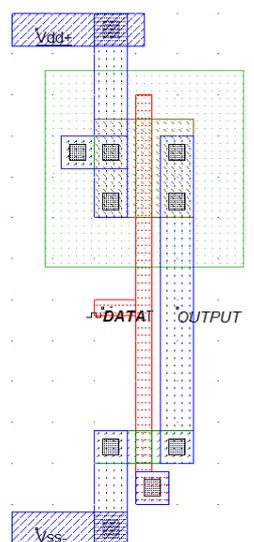


Figure 3.3 : Layout d'un inverseur CMOS.

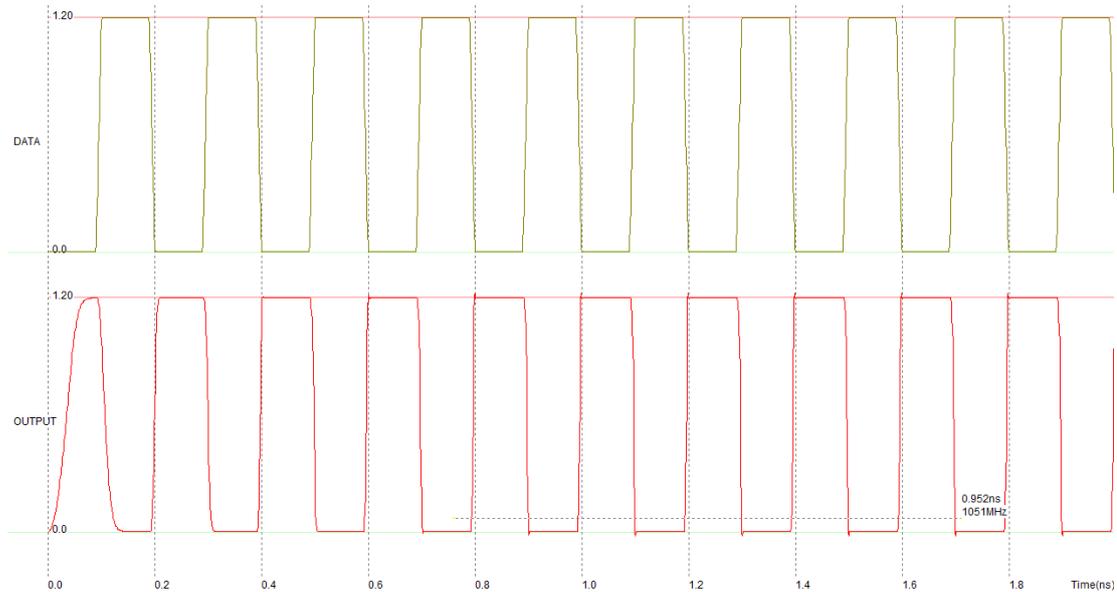


Figure 3.4 : Résultat de la simulation d'un inverseur CMOS.

## 3.2.2 Multiplexeur :

Le multiplexeur 2 vers 1 est la base de tous les multiplexeurs, il est composé par deux portes de transmissions et un inverseur. Il possède deux entrées comme information, une entrée de commande, et une seule sortie (figure ci-dessous). Le signal de commande sélectionne ou choisit une entrée parmi les deux, qui va transmettre à la sortie. On peut dire que c'est un sélecteur. La figure ci-dessous représente le Layout du multiplexeur 2 vers 1, on trouve que I1 et I2 sont des signaux d'entrée (deux informations), S est une commande qui selecte entre I1 et I2. OUT est la sortie du multiplexeur. Le Layout du multiplexeur et le résultat de sa simulation sont représentés ci-dessous. On remarque que le signal de sortie prend parfois la forme de I1 et parfois la forme de I2 selon l'état du signal de commande S. [17]

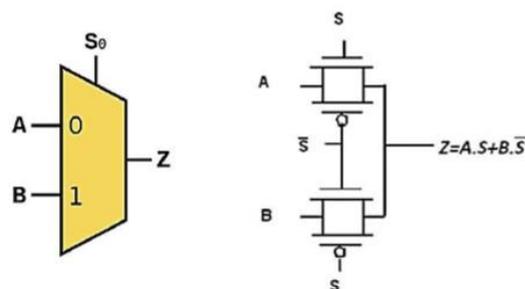


Figure 3.5 : Symbole du multiplexeur 2 vers 1.

# Layout et simulations

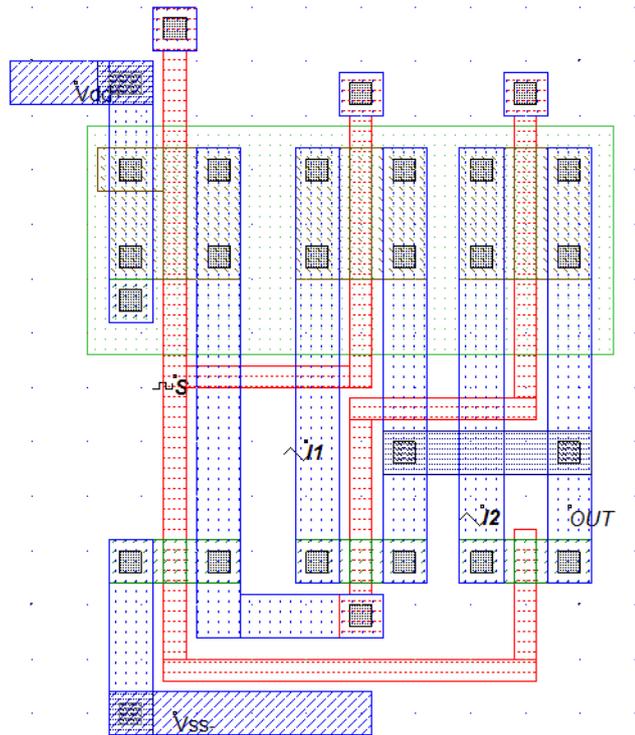


Figure 3.6 : Layout du multiplexeur 2 vers 1.

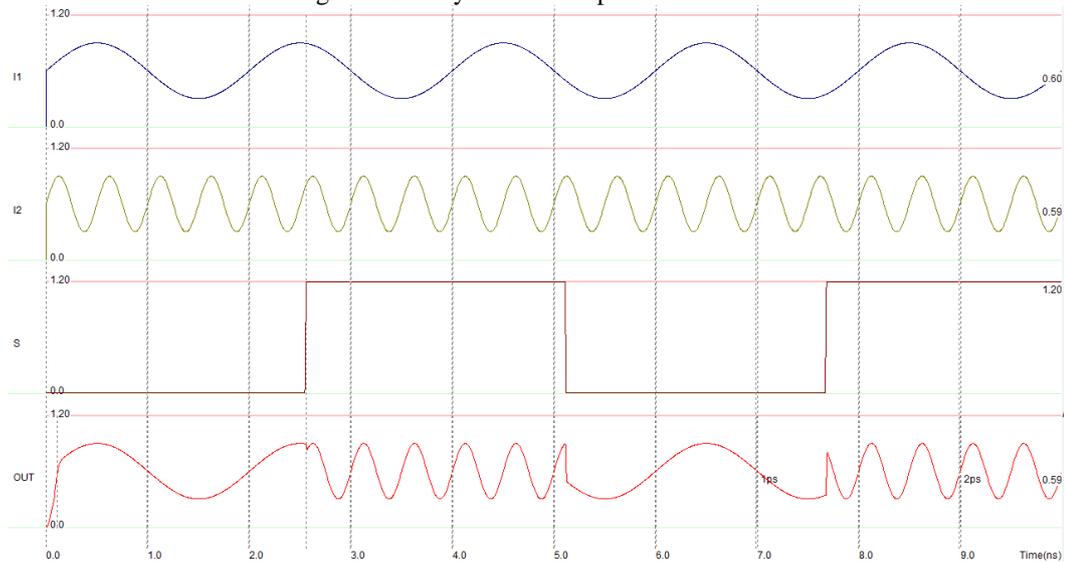


Figure 3.7 : Résultat de simulation du multiplexeur 2 vers 1.

Il est noté que ces multiplexeurs sont construits par des portes de transmission (transmission Gate (TG) en anglais). Cette dernière est utilisée pour corriger la déformation des signaux utilisant les interrupteurs classiques comme le transistor MOS individuel. L'avantage majeur de l'utilisation de ces portes est de minimiser le nombre de transistors dans le circuit logique.

# Layout et simulations

Par exemple, une porte XOR a 2 entrées réalisée avec des portes de transmission ne nécessite que 6 transistors, comparativement à 12 en CMOS complémentaire.

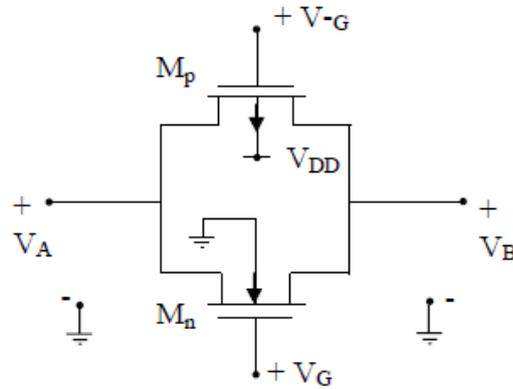


Figure 3.8 : Porte de transmission.

On remarque d'après le résultat de simulation de la porte de transmission présenté ci-dessous que cette porte laisse uniquement une seule partie du signal à transmettre vers la sortie gouvernée par des commandes complémentaires. On remarque aussi que la forme du signal de sortie n'est pas déformée, c'est-à-dire que les signaux d'entrée sortie sont les mêmes.

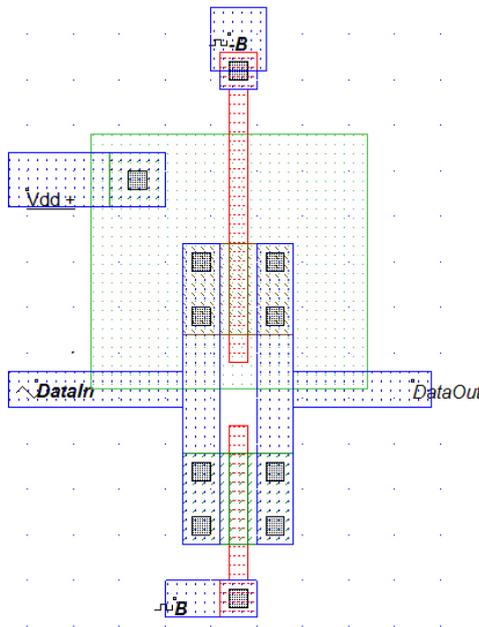


Figure 3.9 : Layout de la porte de transmission.

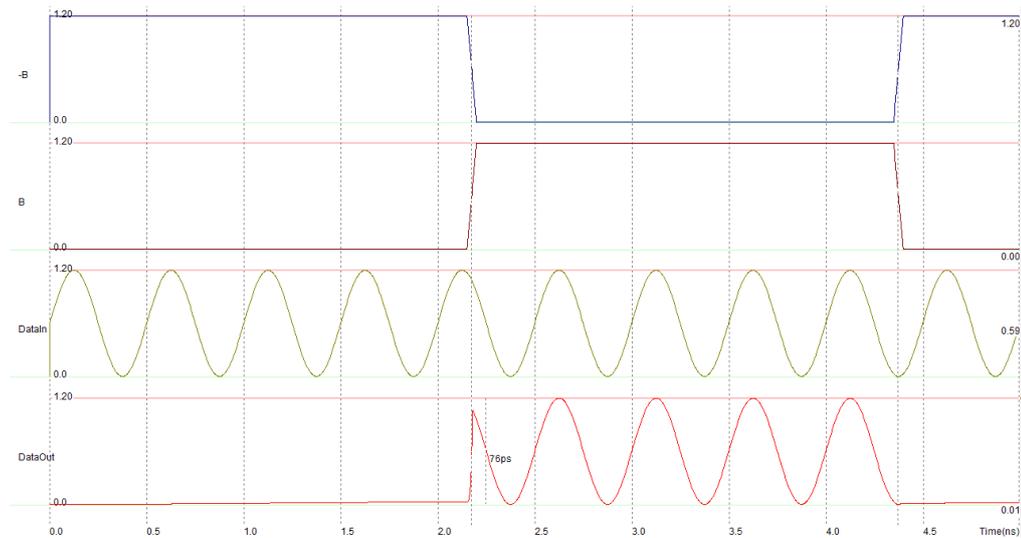


Figure 3.10 : Résultat de simulation de la porte de transmission.

### 3.2.3 Structure et analyse du comportement de la bascule D flip-flop (maître/esclave) réalisée par des portes de transmissions: [18]

La réalisation la plus courante des bascules à déclenchement sur front fait appel à une structure dite maître-esclave qui utilise deux latches, cette structure contient quatre portes de transmissions et quatre inverseurs. On a utilisé les portes de transmissions pour minimiser le nombre des transistors utilisé dans le circuit logique. Par exemple, une porte XOR à 2 entrées réalisées avec des portes de transmission ne nécessite que 6 transistors, comparativement à 12 en CMOS complémentaire. On appelle ça, la logique semi statique.

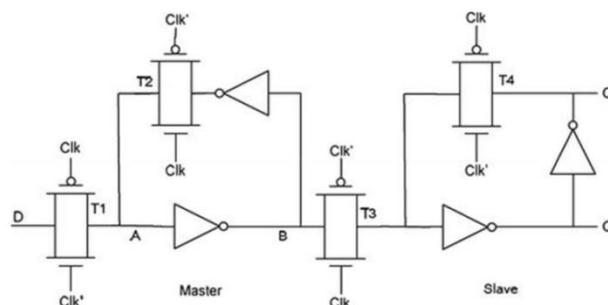


Figure 3.11 : Structure d'une bascule D à déclenchement sur front montant.

Le mode de fonctionnement de chacune des deux latches est lié à l'état de l'horloge CK:

CK = 0

# Layout et simulations

La bascule L1 est en mode transparent et la bascule L2 est verrouillée. Ainsi, la sortie de L1 suit l'entrée D, à une inversion près, mais la sortie de L2 reste bloquée.

CK = 1

Lorsque CK passe à 1, la bascule L1 se verrouille, et mémorise la valeur de D alors présente à l'entrée. D'autre part, la bascule L2 passe en mode transparent et affiche la valeur de D mémorisée par L1 sur la sortie Q.

Pour résumer, la bascule L1 est chargée de faire l'acquisition, sur niveau bas de CK, de la valeur de D à mémoriser. Elle est appelée bascule d'enregistrement, d'acquisition ou bascule maître. La bascule L2 a pour rôle d'afficher sur les sorties Q et Q barre la valeur acquise par L1, lorsque CK passe à 1. Elle est appelée bascule d'affichage ou bascule esclave. Puisque, lorsque CK = 1 ou CK = 0, l'une des deux latches est bloquée, un changement d'état sur D ne peut pas être répercuté sur les sorties. La valeur présente sur l'entrée de donnée D n'est recopiée en sortie qu'à l'instant où CK passe de 0 à 1. Tout se passe donc comme si la copie de D sur la sortie Q avait lieu au moment du front montant de CK.

Dans le cas où les commandes des interrupteurs sont inversées par rapport au schéma précédent, la bascule effectue la copie de D sur les fronts descendants de l'horloge.

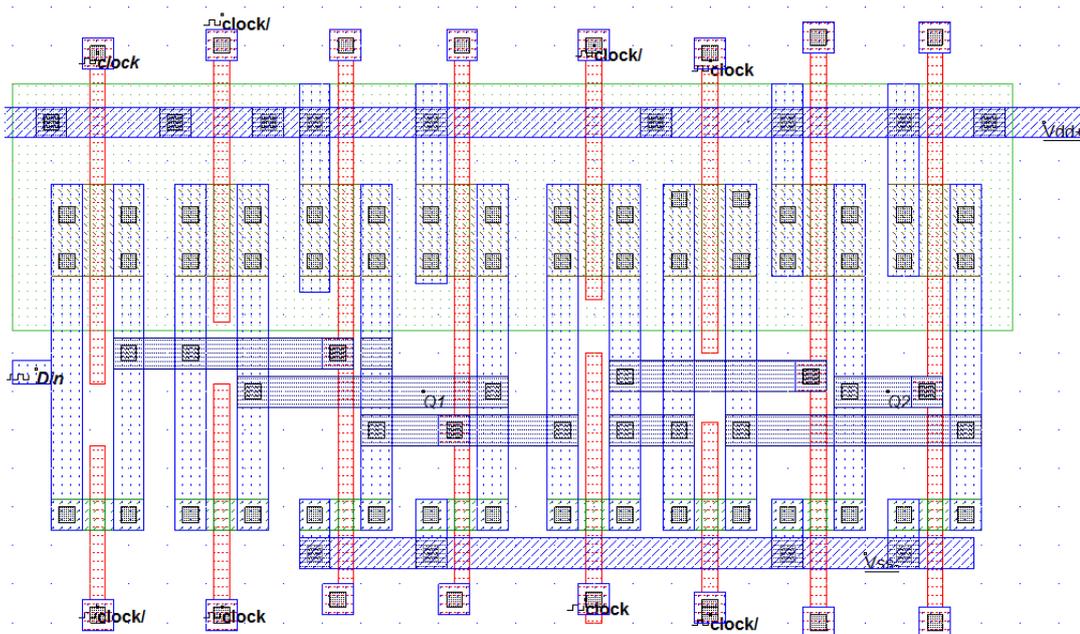


Figure 3.12 : Layout de la bascule maître esclave.

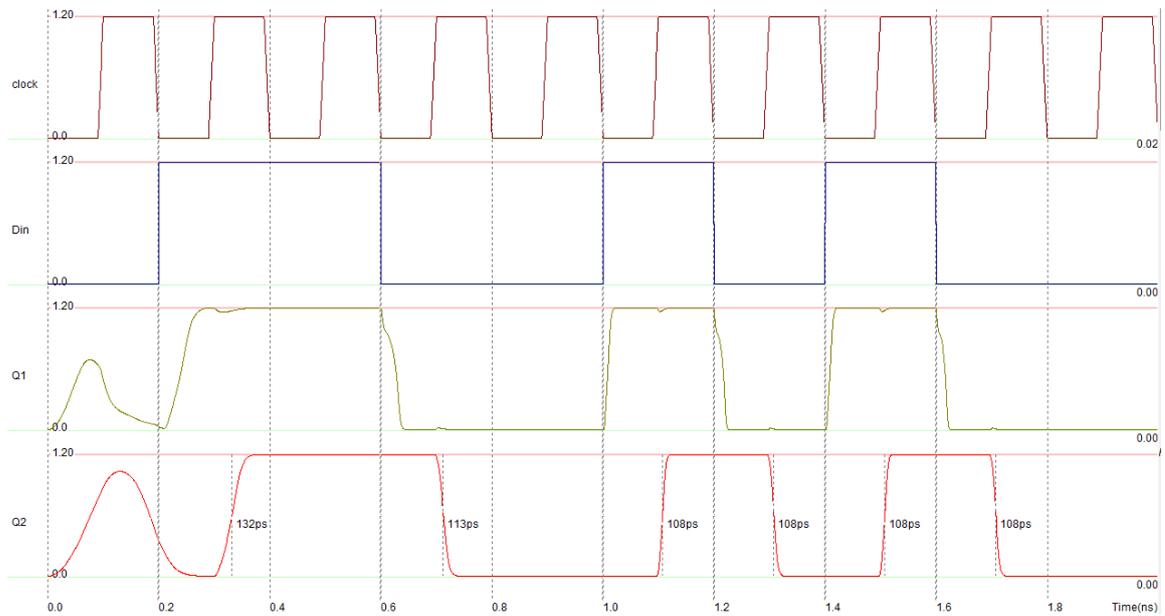


Figure 3.13 : Résultat de la simulation d'une bascule D maître esclave

## 3.2.4 La porte logique NAND :

Ci-dessous, c'est le Layout et le résultat de la simulation de la porte logique NAND en CMOS, on a utilisé cette porte pour définir une deuxième méthode de la réalisation d'une bascule D Flip Flop (maître/esclave) (8 portes NAND avec trois portes NON). La porte NAND prend en entrée deux valeurs 0 ou 1. La sortie vaut 0 si les deux entrées valent 1 et elle vaut 1 si au moins une des deux entrées vaut 0.

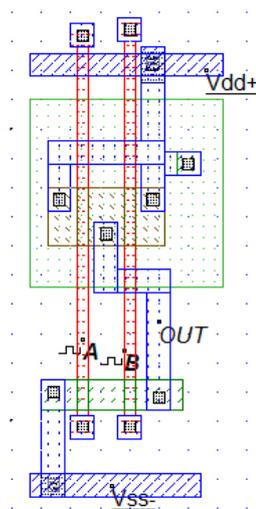


Figure 3.14 : Layout de la porte NAND.

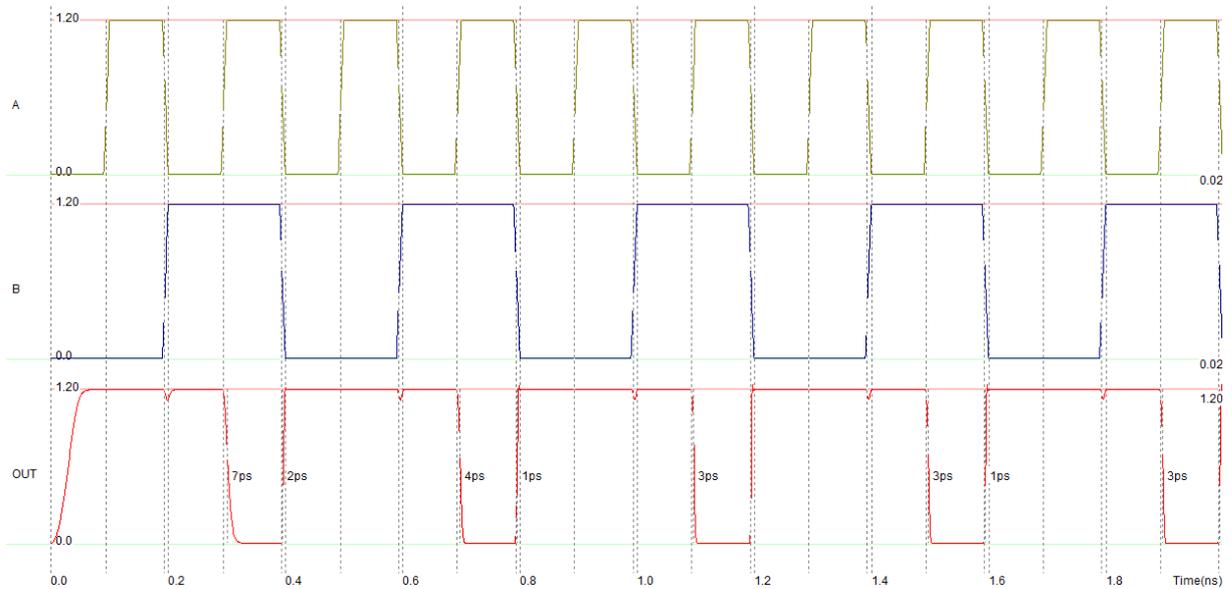


Figure 3.15 : Résultat de la simulation de la porte NAND.

## 3.2.5 Bascule D Flip Flop (maître/esclave) réalisée par les portes logiques NAND + INVERSEURS :

On a réalisé la bascule D (maître/esclave) par huit portes NAND et trois inverseurs, on tient compte que ce circuit contient 35 inverseurs en tout ce qui provoque une perte d'énergie. On appelle ça la logique statique (en utilisant des portes logiques). Le schéma complet de cette bascule est illustré dans la figure 3.16 et son dessin de masque est présenté dans la figure 3.17.

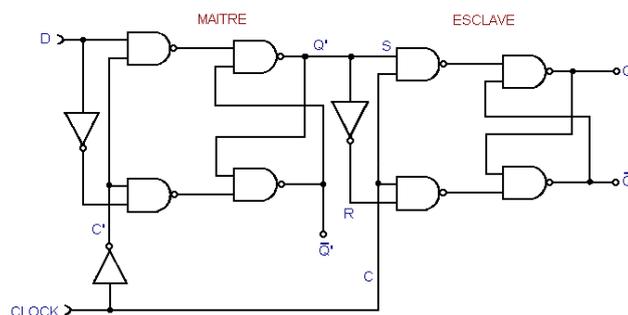


Figure 3.16 : Schéma complet de la bascule D maître/esclave..

# Layout et simulations

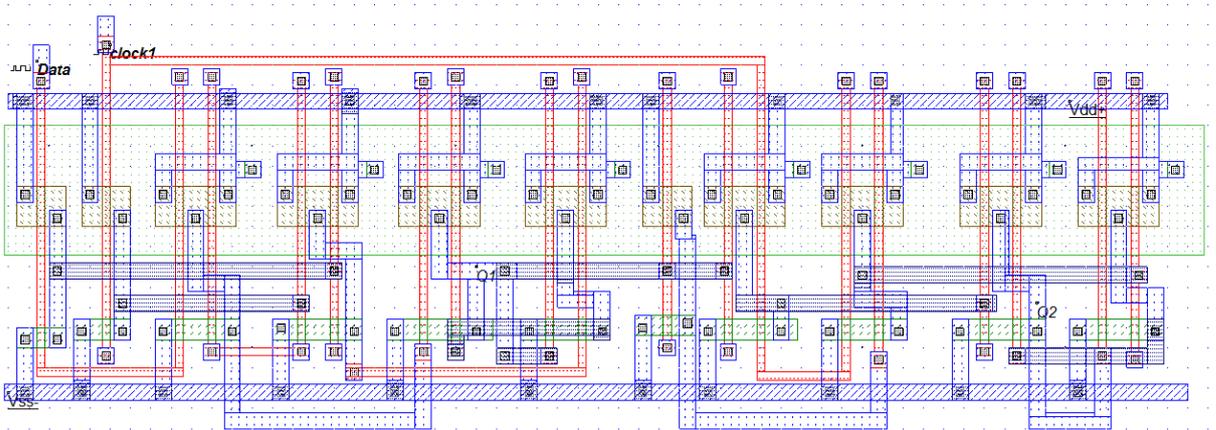


Figure 3.17 : Layout de la bascule maitre esclave.

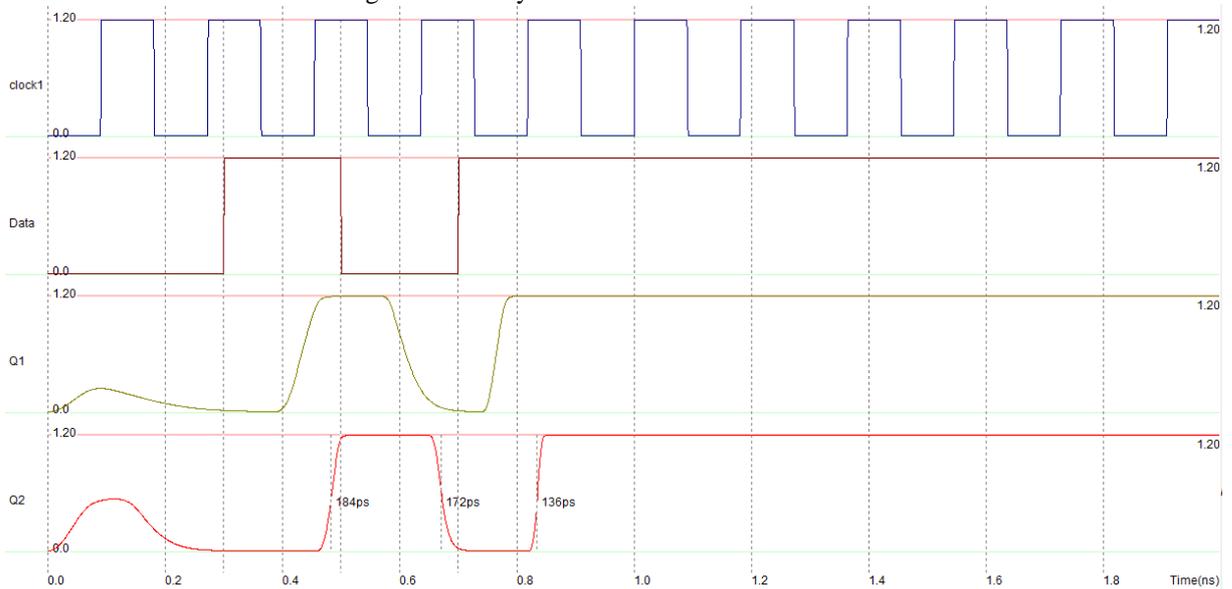


Figure 3.18 : Résultat de la simulation d'une bascule D maitre esclave.

La figure 3.18 représente le résultat de simulation de la bascule D (M/E), on remarque que ce résultat est équivalent au résultat précédent.

## 3.2.6 Layout complet du registre à décalage PISO (entrées parallèles, sorties séries) :

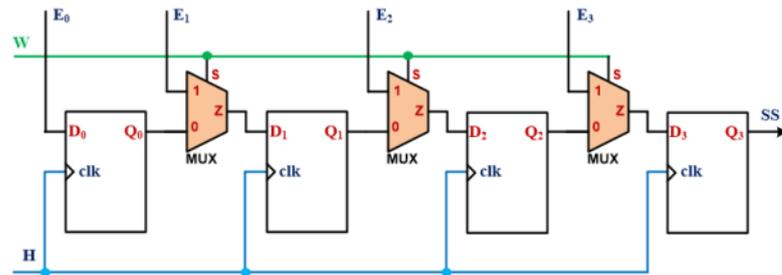


Figure 3.19 : Un registre à décalage PISO à quatre bits.

La figure 3.20 représente le registre à décalage PISO réalisé en utilisant la technologie CMOS 90 nm. Nous avons connecté l'entrée E0 avec l'entrée de la première bascule et les sorties Q0, Q1, et Q2 de chaque bascule sont connectées avec une des deux entrées du multiplexeur 2 vers 1, et la deuxième entrée du multiplexeur est liée avec les entrées E1, E2 et E3. Les trois multiplexeurs utilisés ont le même signal de commande W, et les quatre bascule D (maitre/esclave) sont commandées par le même signal d'horloge H.

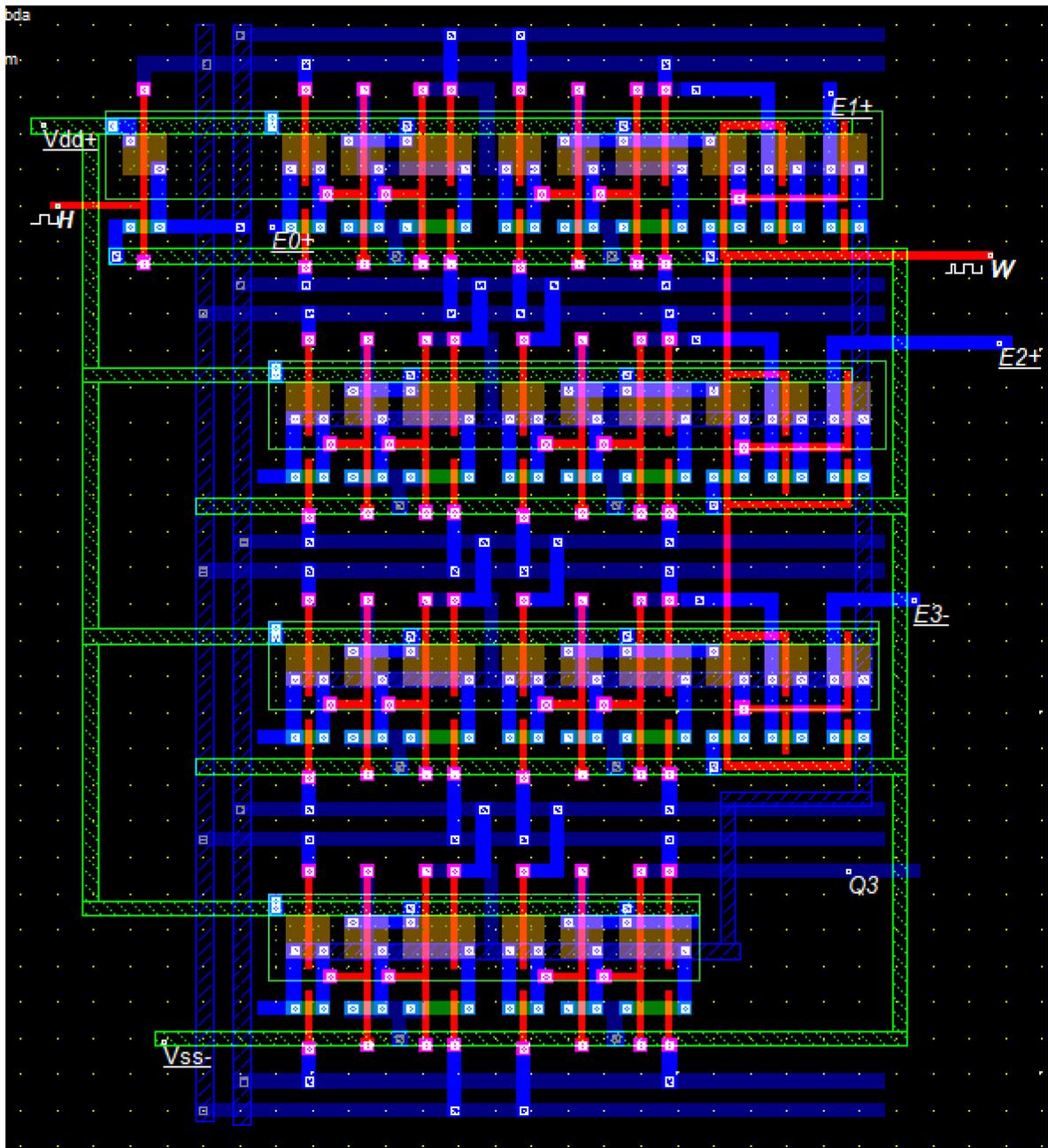


Figure 3.20 : Layout du registre à décalage PISO.

Les 4 figures ci-dessous représentent la simulation obtenue de ce **Layout** pour différents cas pour les entrées en parallèles E3, E2, E1, et E0. On remarque qu'il y a vraiment un décalage en série à la sortie Q3 de la dernière bascule D maître-esclave, après bien sûr le chargement de ces bits en parallèle. Le chargement et le décalage sont commandés par le signal W :

# Layout et simulations

Lorsque le signal de commande (W) soit égal à un, le registre charge les quatre bits E3, E2, E1 et E0 en parallèle dans les quatre bascules, et quand W=0, le registre commence à décaler ces bits en série à chaque front montant.

## ❖ Cas de E3E2E1E0 =1010

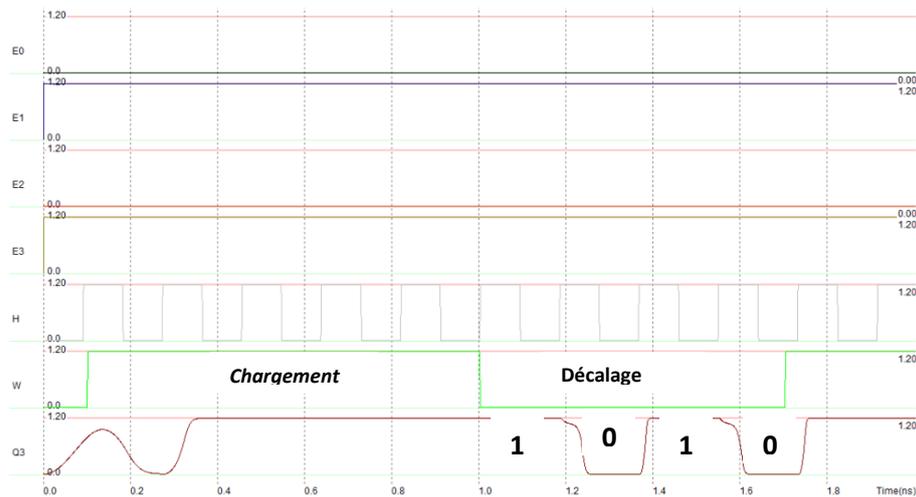


Figure 3.21 Résultat de la simulation du layout du registre PISO pour 1010.

## ❖ Cas de E3E2E1E0 =0011 :

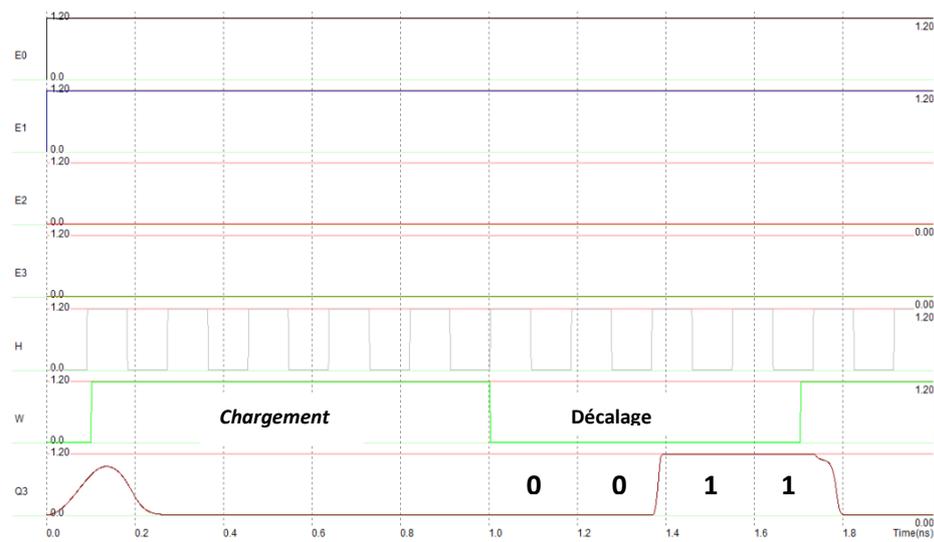
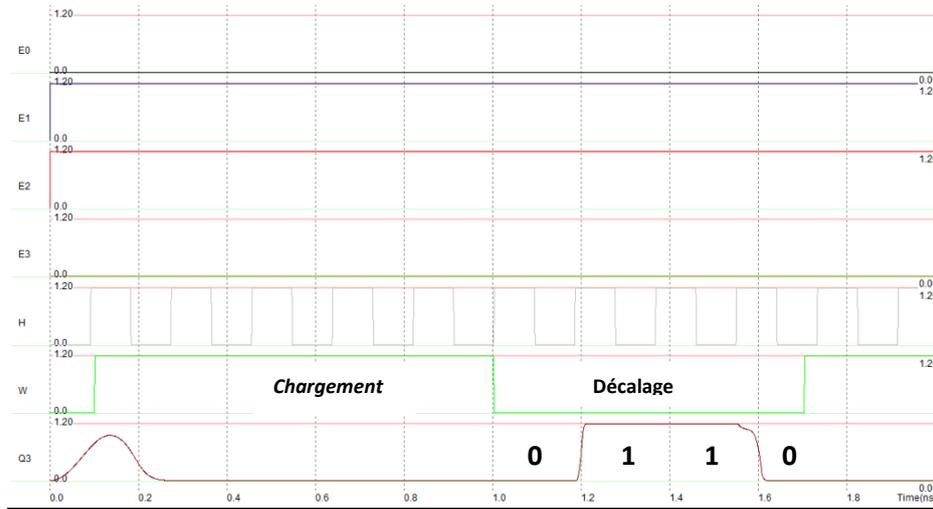


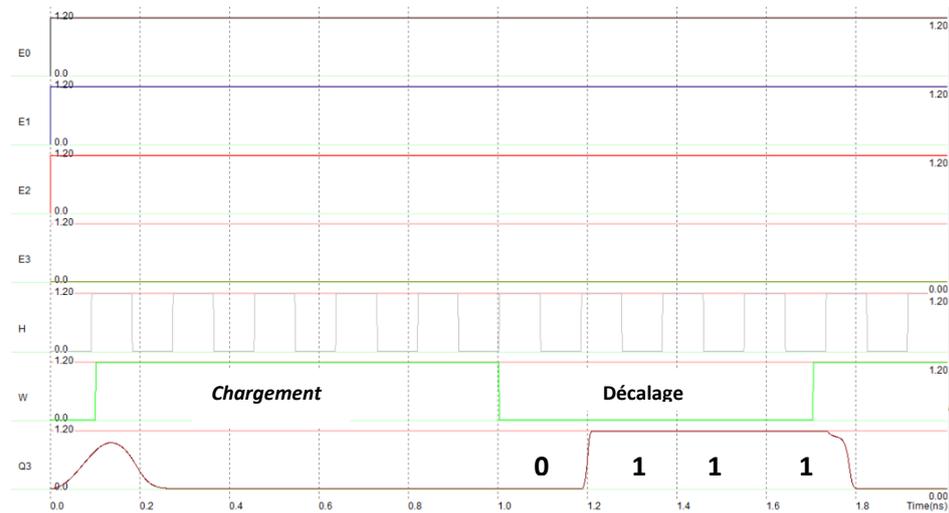
Figure 3.22 Résultat de la simulation du layout du registre PISO pour 0011.

❖ Cas de  $E3E2E1E0 = 0110$  :



*Figure 3.23* Résultat de la simulation du layout du registre PISO pour **0110**

❖ Cas de  $E3E2E1E0 = 0111$  :



*Figure 3.24* Résultat de la simulation du layout du registre PISO pour **0111**.

Après avoir ces simulations, on constate bien clairement que le **Layout** de ce registre de décalage fonctionne d'une manière juste et très précise.

## **3.3 Conclusion :**

Dans ce chapitre, nous avons présenté tous les blocs qui composent le registre à décalage PISO. Nous avons fait les Layout (en utilisant la technologie CMOS 90 nm) et les simulations de ces blocs, la bascule D (maitre/esclave) et le multiplexeur 2 vers 1. Et finalement on a trouvé un bon fonctionnement de ces derniers. Ensuite nous les avons rassemblés dans un seul Layout.

# **Conclusion générale**

# Conclusion générale

Ce modeste travail nous a permis d'assimiler des nouvelles connaissances à propos des types des registres à décalage. Il nous a permis également de comprendre le principe de fonctionnement de chaque registre et de savoir les domaines d'application de chacun d'eux.

Une description dans le chapitre 1, d'un des principaux fonctionnements du transistor MOSFET qui est la modélisation sous forme d'interrupteurs. Une explication approfondie à propos des portes logiques CMOS qui nous a aidées à accomplir le reste de notre travail.

Une présentation dans le chapitre 2 des différents registres à décalage, nous a permis de comprendre plus profondément les différents blocs constituant ces derniers, comme les bascules D maître/esclave, et les multiplexeurs 2 vers 1 pour le cas d'un registre PISO qui est notre thème d'étude.

Des Layouts et des simulations ont été faites dans le chapitre 3, en utilisant un logiciel très connu dans le domaine de conception des circuits analogiques et numériques ou mixte qui est appelé Microwind, des différents circuits constituant le registre à décalage PISO qui sont : le multiplexeur, et les bascules D maître esclave.

Dans le chapitre 3, nous avons réussi, à apprendre et à comprendre plusieurs choses de dessins de masques (Layout) des différents circuits réalisés dans ce travail en utilisant le Microwind. Parce qu'à partir de ces dessins de masques et leurs simulations correctes, les étapes technologiques peuvent se commencer pour fabriquer les circuits intégrés désirés.

Pour conclure, ce travail nous a permis de :

- Comprendre le principe de fonctionnement des circuits numériques tels que : les portes logiques, les circuits combinatoire, les circuits séquentiels bien sur au niveau d'abstraction physique.
- Connaitre les différents types des registres à décalage.
- Connaitre les différents blocs constituant le registre à décalage PISO.
- Réaliser le dessin de masque (Layout) pour chaque bloc, et comprendre leurs principes de fonctionnement.
- Rassembler les différents blocs pour réaliser le registre à décalage PISO dans un seul circuit.

## Les références :

[1] Mr. Mebarki Zakaria Mr.Bouakili Adel « Etude comparative et simulation des miroirs de courant intégrés en technologie CMOS », mémoire master, Université Mohamed El Bachir El Ibrahimi de Bordj Bou Arreridj, Algérie, 2016.

[2][https://fr.wikipedia.org/wiki/Int%C3%A9gration\\_%C3%A0\\_tr%C3%A8s\\_grande\\_%C3%A9chelle](https://fr.wikipedia.org/wiki/Int%C3%A9gration_%C3%A0_tr%C3%A8s_grande_%C3%A9chelle)

[3] [www.abcelectronique.com](http://www.abcelectronique.com) "Technologie Bipolaire".

[4] J. R. Davis, "Instabilities in Mos Devices".

[5] Jean-Baptiste Waldner, "Nano-informatique et Intelligence Ambiante – Inventer l'Ordinateur du XXIème Siècle".

[6][https://fr.wikipedia.org/wiki/Complementary\\_metal\\_oxide\\_semi-conductor#/media/Fichier:CMOS\\_cross\\_section\\_FR.png](https://fr.wikipedia.org/wiki/Complementary_metal_oxide_semi-conductor#/media/Fichier:CMOS_cross_section_FR.png)

[7] Mm.BOUALI SIHEM, Mm.BELAIFA SOURIA « Réalisation du layout d'un convertisseur numérique-analogique en technologie CMOS 0.1 micro mètre », mémoire de master, université Mohamed El Bachir El Ibrahimi de Bordj Bou Arreridj, Algérie, 2018.

[8] Benoit Gosselin "caractéristiques de la conception des portes logique et des portes de transmissions" Conception des cours, Université Laval,pp.2011.

[9] C.Toumazou, J.B. Hughes, and N.C. Batters by. "Switched-currents an analogue Technique for digital technology".Peter Peregrinus, pp. 1994

[10][http://ressources.univlemans.fr/AccesLibre/UM/Pedago/physique/02/electro/logiCmos.html#:~:text=Inverseur%20Cmos%20%3A&text=On%20associe%20sur%20un%20m%C3%A4me,\(canal%20N\)%20est%20satur%C3%A9](http://ressources.univlemans.fr/AccesLibre/UM/Pedago/physique/02/electro/logiCmos.html#:~:text=Inverseur%20Cmos%20%3A&text=On%20associe%20sur%20un%20m%C3%A4me,(canal%20N)%20est%20satur%C3%A9).

# Références bibliographiques

- [11] Mr.BOULAHBAL MOHAMED LAMINE, Mr.BOULAARES HICHEM « Etude et simulation d'un multiplieur 4 bits », mémoire de master, université SAAD DAHLAB de BLIDA, Algérie, 2020.
- [12] Circuits logiques séquentiels, registres à décalage, TRABELSI HICHEM, Université virtuelle de Tunis.
- [13] [https://www.electronique-et-informatique.fr/Digit/Digit\\_8T.php](https://www.electronique-et-informatique.fr/Digit/Digit_8T.php)
- [14] <https://fr-academic.com/dic.nsf/frwiki/1420018>
- [15] Mr.BOULAHBAL MOHAMED LAMINE, Mr.BOULAARES HICHEM « Etude et simulation d'un multiplieur 4 bits », mémoire de master, université SAAD DAHLAB de BLIDA, Algérie, 2020.
- [16] R Steven, « delta-sigma Data converters theory, design, and simulation », IEEE circuits & systems society, sponsor, the institute of electrical and electronics engineers, New York, 1997.
- [17] Mm.Daba Mouna Rim, Mm.Djahnit Mouna « Etude et conception d'un capteur d'image APS en technologie CMOS», mémoire de master, université Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj, Algérie, 2021.
- [18] Mr.Abdelghani Dendouga, « Caractérisation et layout d'une bascule C2MOS différentielle faible puissance en technologie CMOS 0.35 $\mu$ m », MAGISTER EN MICROELECTRONIQUE, institut d'Electronique-Université de Batna, Algérie.