

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE 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'électronique.

Mémoire

Présenté pour obtenir

LE DIPLOME DE MASTER

FILIERE : ELECTRONIQUE INDUSTRIELLE

Spécialité : INDUSTRIES ELECTRONIQUES

Par

- **Hadji Rabah**
- **Benkhira Rissel**

Intitulé

***Etude et réalisation d'un timer de microprocesseur en
technologie CMOS***

Soutenu le :04/07/2023

Devant le Jury composé de :

<i>Nom & Prénom</i>	<i>Grade</i>	<i>Qualité</i>	<i>Etablissement</i>
<i>M.YOUSFI.Abderrahim</i>	<i>MCB</i>	<i>Président</i>	<i>Univ-BBA</i>
<i>M.DIB.Ilyes</i>	<i>MCB</i>	<i>Encadreur</i>	<i>Univ-BBA</i>
<i>M.DJEMOUAI Abdelouahab</i>	<i>MCA</i>	<i>Examineur</i>	<i>Univ-BBA</i>

Année Universitaire 2022/2023

Remerciement

Tout d'abord, nous remercions Allah le tout-puissant de nous avoir donné le courage et la patience de développer ce travail modeste.

C'est un plaisir pour nous de pouvoir remercier tous ceux qui nous ont aidés de près ou de loin à parvenir.

Notre profonde gratitude et nos sincères remerciements vont à notre encadrant Dr.DIB.Lyes, pour confiance qu'il nous a accordée, ce qui nous a poussés à multiplier nos efforts pour être à la hauteur de son attente. Nous espérons que vous trouverez dans cet ouvrage une sincère gratitude pour toutes les connaissances que vous nous ont transmis durant notre cursus avec autant de dynamisme, de compétence et de rigueur. Notre gratitude va à tous les membres du jury qui nous feront l'honneur d'apprécier notre travail.

Tous les mots restent faibles pour exprimer notre profonde gratitude à nos parents pour le soutien et l'aide précieuse qu'ils nous ont apportée durant nos longues années d'études.

Dédicace

*Je voudrais dédier cet humble travail :
A ma chère mère et mon cher père pour leur aide
et soutien.*

A mon frère et ma sœur.

*A tous mes enseignants, et mes amis
sans oublier mon collègue Kissel qui été à mon
côtés tout au long de ce projet.*

H.RABAH

Dédicace

A mon cher pays : l'Algérie, fidèle à son peuple.

À l'homme qui m'a appris à quel point la patience est un chemin vers le
succès Le

maître et le modèle...! Mon père bien-aimé, que Dieu prolonge son vie.

A celle dont la satisfaction est mon but et mon ambition..... Elle

m'a beaucoup donné et n'a pas attendu merci.... à l'émetteur de

détermination, détermination et volonté... qui ont eu une empreinte
sincère dans

ma vie... Ma mère bien-aimée, que Dieu prolonge sa vie.

Aux Compagnons d'une élégante maison immaculée... Mes frères et
sœurs.

A mes tous mes amis et professeurs et

tous ceux qui m'ont aidé et soutenu dans ce travail.

B. RISSEL

Résumé :

L'objectif principal de ce projet était de concevoir et de mettre en œuvre avec la réalisation des dessins de masques (layout) d'un timer de microprocesseur efficace et précis en technologie CMOS en utilisant le logiciel Microwind. Ce timer permet de réaliser plusieurs fonctions comme le comptage et décomptage des évènements, la synchronisation des signaux et la génération des évènements périodique. Ce timer constituent de 3 blocs principaux ce qui permet de s'agit plusieurs entrées et sorties pour la communication avec les autres périphériques de microprocesseur. Les Layouts de chaque bloc de ce timer a été réalisés en technologie CMOS 90nm, et leurs simulations ont été faites avec succès. On a trouvé le bon résultat entre la simulation et le principe de fonctionnement de chaque bloc. Enfin on a pu regrouper tous les blocs du timer dans un seul Layout.

Abstract:

The main objective of this project was to design and implement with the realization of the drawings of masks (layout) of an efficient and precise microprocessor timer in CMOS technology using the Microwind software. This timer is used to perform several functions such as counting up and down counting of events, synchronization of signals and generation of periodic events. This timer consist of 3 main blocks which allows for multiple inputs and outputs for communication with other microprocessor peripherals. The layouts of each block of this timer were made in 90nm CMOS technology, and their simulations were done successfully. We found the right result between the simulation and the operating principle of each block. Finally we were able to group all the blocks of the timer in a single Layout.

الملخص :

كان الهدف الرئيسي من هذا المشروع هو التصميم والتنفيذ مع تحقيق رسومات الأقنعة (التخطيط) لمؤقت معالج دقيق فعال ودقيق في تقنية CMOS باستخدام برنامج Microwind. يستخدم هذا المؤقت لأداء العديد من الوظائف مثل العد التصاعدي والتنازلي للأحداث ومزامنة الإشارات وتوليد الأحداث الدورية. يتكون هذا المؤقت من 3 كتل رئيسية تسمح بمدخلات ومخرجات متعددة للتواصل مع الأجهزة الطرفية الأخرى للمعالج الدقيق. تم إجراء تخطيطات كل كتلة من هذا المؤقت بتقنية 90 نانومتر CMOS ، وتم إجراء عمليات المحاكاة الخاصة بهم بنجاح. وجدنا النتيجة الصحيحة بين المحاكاة ومبدأ التشغيل لكل كتلة. أخيرًا ، تمكنا من تجميع كل الكتل الخاصة بالمؤقت في تخطيط واحد.

Liste d'abréviations

CMOS: Complementary Metal Oxide Semiconductor

MOSFET: Metal Oxide Semiconductor Field Effect Transistor

ASIC : Application Specific Integrated Circuit

FPGA : Field Programmable Gate Array

PLA : Programmable Logic Array

SSI : Small Scale Integration

MSI : Medium Scale Integration

TTL : Transistor Transistor Logic

DTL : Diode Transistor Logic

CLK : Clock

RTC : Real time clock

PWM : Pulse Width Modulation

DSP : Digital signal processor

LED : Light Emitting Diode

IC : Integrated Circuit

Listes des figures :

Figure 1.1 : Présentation du transistor NMOS.....	5
Figure 1.2 : Présentation du transistor PMOS.	5
Figure 1.3 : CMOS pull up et pull down.	7
Figure 1.4 : schéma et symbole de l'inverseur	7
Figure 1.5 : schéma et symbole de porte NAND	8
Figure 1.6 : schéma CMOS du NAND	9
Figure 1.7 : schéma et symbole de porte NOR	9
Figure 1.8 : schéma CMOS du NOR	10
Figure 1.9 : symbole et schéma CMOS de la porte AND.....	11
Figure 1.10 : symbole et schéma CMOS de la porte OR	11
Figure 1.11 : symbole des portes XOR et XNOR.....	12
Figure 1.12 : Schéma CMOS des portes XOR et XNOR.....	12
Figure 1.13 : schéma bloc de latch SR	13
Figure 1.14 : circuit logique du latch SR	13
Figure 1.15 : le schéma bloc du bascule RS.	14
Figure 1.16 : circuit logique du bascule RS.	14
Figure 1.17 : schéma bloc du bascule JK	15
Figure 1.18 : schéma bloc du bascule T.....	16
Figure 1.19 : schéma bloc du bascule D.....	17
Figure 1.20 : circuit logique du bascule D.....	17
Figure 1.21 : Les impulsions d'horloge positive et négative.....	18
Figure 1.22 : Schémas bloc de bascule D maitre esclave	19
Figure 1.23 : Représentation schématique d'un circuit combinatoire.....	19
Figure 1.24 : schéma bloc d'un multiplexeur 2 vers 1.	20
Figure 1.25 : le schéma logique du multiplexeur 2 vers 1	21
Figure 1.26 : schéma bloc d'un multiplexeur 4 vers 1.	21
Figure 1.27 : le schéma logique du multiplexeur 4 vers 1	22
Figure 1.28 : le schéma logique d'un comparateur d'égalité	23
Figure 1.29 : le schéma logique d'un comparateur d'amplitude.....	24
Figure 2.1 : Schéma bloc d'un compteur asynchrone 3 bit.....	29
Figure 2.2 : Signal PWM avec un rapport cyclique 50 %.....	31
Figure 3.1 : Schéma bloc du timer	35
Figure 3.2 : Schéma logique globale du timer.....	36
Figure 3.3 : Schéma logique de bascule D maitre esclave avec set/reset	38
Figure 3.4 : Layout de bascule D maitre esclave avec set/reset.....	39
Figure 3.5 : Résultat de simulation de bascule D maitre esclave	39
Figure 3.6 : Schéma bloc de multiplexeur 2 vers 1	40
Figure 3.7 : Layout de multiplexeur 2 vers 1.....	40
Figure 3.8 : Résultat de simulation de multiplexeur 2 vers 1	41
Figure 3.9 : Schéma logique de comparateur 3bit	41
Figure 3.10 : Layout de comparateur 3bit.....	42
Figure 3.11 : Résultat de simulation de comparateur 3bit.....	42

<i>Figure 3.12 : Schéma logique de comparateur 7485.....</i>	<i>43</i>
<i>Figure 3.13 : Layout de bloc PWM</i>	<i>43</i>
<i>Figure 3.14 : Résultat et simulation du bloc PWM</i>	<i>44</i>
<i>Figure 3.15 : Layout globale de timer.....</i>	<i>45</i>
<i>Figure 3.16 : Chronogramme compteur.</i>	<i>46</i>
<i>Figure 3.17 : Chronogramme décompteur.</i>	<i>47</i>

Listes des tableaux :

<i>Tableau 1 : table de vérité de porte NAND.....</i>	<i>8</i>
<i>Tableau 2 : table de vérité de porte NOR</i>	<i>9</i>
<i>Tableau 3 : table de vérité de porte AND</i>	<i>11</i>
<i>Tableau 4 : table de vérité de porte OR</i>	<i>11</i>
<i>Tableau 5 : table de vérité des portes XOR et XNOR.....</i>	<i>12</i>
<i>Tableau 6 : table de vérité de latch SR.....</i>	<i>13</i>
<i>Tableau 7 : table de vérité du bascule RS.</i>	<i>15</i>
<i>Tableau 8 : table de vérité du bascule JK.....</i>	<i>16</i>
<i>Tableau 9 : table de vérité du bascule T</i>	<i>16</i>
<i>Tableau 10 : table de vérité du bascule D.....</i>	<i>17</i>
<i>Tableau 11 : table de vérité du multiplexeur 2 vers 1.....</i>	<i>20</i>
<i>Tableau 12 : tableau des fonctions d'un multiplexeur 4 vers 1.</i>	<i>22</i>
<i>Tableau 13 : tableau de vérité de comparateur.....</i>	<i>24</i>
<i>Tableau 14 : Les sorties binaires d'un compteur 3 bit</i>	<i>30</i>
<i>Tableau 15 : Table de vérité de bascule D maitre esclave avec set/reset.....</i>	<i>38</i>

Table des matières

Introduction générale	1
------------------------------	---

Chapitre 1

1.1	Introduction	3
1.2	Histoire sur la microélectronique	3
1.3	Technologie TTL et MOS en mode de commutation	4
1.3.1	La technologie MOS	4
1.3.2	La technologie TTL	5
1.4	La technologie CMOS	6
1.4.1	Les portes logiques CMOS	7
1.4.1.2	Porte NAND	8
1.4.1.3	Porte NOR	9
1.4.1.4	Porte AND et OR	10
1.4.1.5	XOR et XNOR	12
1.4.2	Les circuits séquentiels	13
1.4.2.1	SR latch :	13
1.4.2.2	Bascule RS :	14
1.4.2.3	Bascule JK	15
1.4.2.4	Bascule T	16
1.4.2.5	Bascule D	17
1.4.2.6	Bascule D maitre esclave :	17
1.4.3	Les circuits combinatoires	19
1.4.3.1	Les multiplexeurs	20
1.4.3.2	Les comparateurs	22
1.5	Conclusion	24

Chapitre 2

2.1	Introduction	26
2.2	Les timers	26
2.3	Types des timers à microprocesseur	26
2.3.1	Timers d'intervalle	26
2.3.2	Timers de surveillance (timer chien de garde)	27
2.3.3	Horloges en temps réel (RTC)	27
2.3.4	Timers PWM	27
2.3.5	Compteurs d'événements	27
2.4	Importance des timers à microprocesseur	27

2.5 Applications des timers à microprocesseur _____	28
2.6 Les blocs constituent le timer d'un microprocesseur _____	29
2.6.1 Compteur 3bit _____	29
2.6.2 Bloc PWM _____	30
2.6.3 Bloc contrôleur de modulo _____	31
2.7 Conclusion _____	32
Chapitre 3	
3.1 Introduction _____	34
3.2 Présentation du logiciel Microwind _____	34
3.3 Structure logique global du timer _____	35
3.3.1 Principe de fonctionnement _____	37
3.4 Dessin de masques et simulation des blocs constituent le timer _____	37
3.4.1 Bascule D maitre esclave _____	37
3.4.2 Multiplexeur _____	40
3.4.3 Contrôleur de modulo _____	41
3.4.4 PWM _____	42
3.5 Dessin de masque et simulation globale du timer _____	45
3.5.1 Circuit globale _____	45
3.5.2 Simulation globale _____	46
3.6 Conclusion _____	47
Conclusion générale _____	47
Références	

Introduction générale

Les timers jouent un rôle crucial dans le fonctionnement des microprocesseurs, car ils permettent de mesurer des intervalles de temps précis et de synchroniser différentes opérations. Dans cette étude, nous nous intéressons à la conception et à la réalisation d'un timer en technologie CMOS (Complementary Metal-Oxide-Semiconductor) pour un microprocesseur.

La technologie CMOS est largement utilisée dans l'industrie des semi-conducteurs en raison de sa faible consommation d'énergie, de sa compatibilité avec les circuits intégrés et de sa capacité à fonctionner à des vitesses élevées. La conception d'un timer en utilisant cette technologie offre de nombreux avantages, tels qu'une meilleure précision, une plus grande stabilité et une consommation d'énergie réduite.

Le timer sera capable de compter les impulsions d'horloge et de générer des signaux de temporisation précis pour synchroniser les opérations du microprocesseur.

L'objectif de cette étude est de concevoir un timer capable de générer des intervalles de temps précis, allant de quelques microsecondes à plusieurs heures, selon les besoins du microprocesseur, cela avec l'utilisation des circuits combinatoires tels que le multiplexeurs, comparateurs, et des circuits séquentiels tels que les bascules D maître-esclave et compteurs basés sur la technologie CMOS.

Ce travail est structuré en trois chapitres :

Le premier chapitre fournira l'historique et les généralités sur la technologie CMOS et TTL avec présentation des porte logiques et les circuit séquentielles et combinatoires en technologie CMOS et sont caractéristiques.

Le deuxième chapitre présente les timers, ces types et ces différents blocs avec leur principe de fonctionnement.

Au final nous aboutirons à la réalisation des dessins de masques (**Layout**) des blocs constituent le timer et le schéma globale avec ces résultats de simulation final.

Ce mémoire se termine par une conclusion générale.

CHAPITRE 1 :
Généralités sur la
technologie CMOS

1.1 Introduction

La technologie CMOS est la technologie dominante pour la réalisation des circuits intégrés, elle permet d'augmenter le nombre de transistors par puce selon une loi empirique, la loi de Moore, qui s'applique depuis plus de 50 ans. En 2019, cette technologie permet de réaliser des microprocesseurs ou des processeurs graphiques avec plusieurs dizaines de milliards de transistors. [1]

L'objectif de ce chapitre est de présenter les caractéristiques de la technologie CMOS et ces différents portes logiques ainsi que ces circuits séquentiel et combinatoire.

1.2 Histoire sur la microélectronique

Avec l'invention du transistor en 1947 et du circuit intégré en 1958, constitué de millions de transistors sur une puce. Cette évolution a permis le développement de l'industrie électronique, qui représente environ 20% du marché électronique mondial en 2004. Les techniques de fabrication de transistors et de circuits intégrés impliquent l'utilisation de matériaux semiconducteurs, en particulier le silicium, ainsi que des processus de photolithographie et de dopage. [1]

La passivation et la couverture de nitrure de silicium ont permis de protéger les transistors et les puces de l'air ambiant, sans nécessiter de boîtiers étanches. La société Intel a été fondée en 1968 pour commercialiser des mémoires MOS et a sorti la première mémoire dynamique en 1970. Chaque cellule de mémoire de 1 bit comportait 3 transistors. [1]

L'évolution de la technologie électronique a permis le développement des ordinateurs et des logiciels, et a transformé la société en profondeur. Cette évolution est comparée à la deuxième révolution industrielle, liée au système technique basé sur les moteurs, l'énergie, les métaux, les machines, le transport de matières et de personnes et les industries extractives au XIXe siècle. [1]

Dans les années 1960, les premiers circuits intégrés numériques commerciaux étaient des portes logiques et des bascules élémentaires, qui étaient assemblées pour réaliser des fonctions logiques plus complexes. Au début des années 1970, les premiers microprocesseurs et les premières mémoires denses sont apparus, ce qui a permis de réaliser de nombreuses applications électroniques. Dans les années 1980, les circuits SSI et MSI ont été remplacés

par des PLAs programmables et des circuits ASIC spécialisés. À la fin des années 1980, les réseaux de portes programmables FPGA sont apparus, offrant un temps de développement plus court et une reprogrammable électrique. Aujourd'hui, les cartes électroniques spécialisées sont souvent réalisées en assemblant un microprocesseur, une mémoire, et un FPGA ou un ASIC, qui résument les principaux segments du marché des circuits intégrés. [1]

1.3 Technologie TTL et MOS en mode de commutation

Deux familles se sont développées parallèlement, l'une fondée sur les transistors bipolaires au silicium, l'autre utilisant des transistors à effet de champ de type M.O.S. (Métal - Oxyde - Semi-conducteur). [2]

1.3.1 La technologie MOS

Le transistor MOS est, de loin, le dispositif le plus répandu dans la production actuelle de composants semi-conducteurs, car il est le composant de base de la technologie CMOS (Complementary MOS), qui, à elle seule, englobe plus de 70 % de la production mondiale de circuits intégrés.

Le transistor MOS comporte trois électrodes :

- la source : zone de silicium dopé négativement, diffusée sur un substrat de type P ;
- le drain : identique à la source, il ne s'en différencie que par la tension qui lui est appliquée ;
- la grille : elle est en aluminium et constitue l'électrode de commande. Elle est isolée de la source, du drain et du substrat par une couche de diélectrique en oxyde de silicium SiO_2 . C'est la nature des trois électrodes qui justifie l'appellation M.O.S. La source étant à la masse, appliquons une tension d'une dizaine de volts au drain. En l'absence de polarisation de la grille, les porteurs ne peuvent circuler d'une électrode à l'autre. Bien qu'elle soit isolée, elle va agir par l'intermédiaire du champ électrique qu'elle crée à l'intérieur du substrat, en chassant les trous à une certaine distance de l'oxyde de silicium (la dénomination transistor à effet de champ se trouve ainsi justifiée). Il s'établit sous le diélectrique une zone désertée en porteurs positifs. Simultanément, la grille attire de plus en plus les électrons, porteurs majoritaires de la source et du drain, jusqu'à ce que se constitue un canal conducteur de type N, Donc il existe deux types transistor MOS, type N et type P qui montrons dans les figures 1.1 et 1.2 ci-dessus :

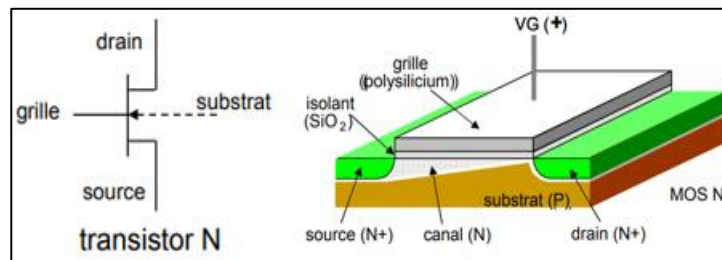


Figure 1.1 : Présentation du transistor NMOS. [15]

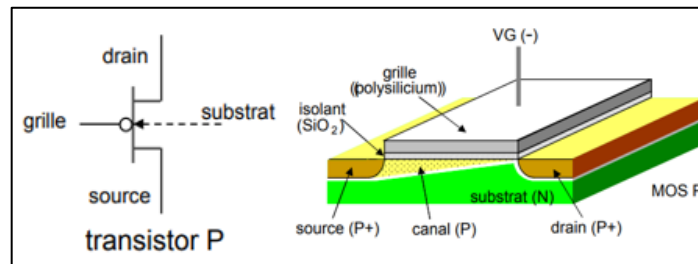


Figure 1.2 : Présentation du transistor PMOS. [15]

On n'observe que celui que le N.M.O.S. à enrichissement. En effet, le canal n'existe pas en l'absence de tension de grille. C'est en polarisant celle-ci qu'on enrichit le substrat en porteurs de type N. Nous rencontrerons plus loin le P.M.O.S. à enrichissement dont le principe est identique, mais la constitution symétrique : substrat N, drain, source et canal P, source et substrat à la masse, V_d et V_g négatives. Les performances des N.M.O.S. sont meilleures que celles des P.M.O.S. surtout en ce qui concerne la rapidité de transmission des signaux. Ceci est dû à la mobilité des électrons, trois fois supérieure à celle des trous. [2]

1.3.2 La technologie TTL

Une famille de logique numérique est constituée d'un ensemble de circuits intégrés ou d'autres blocs élémentaires compatibles qui peuvent être combinés dans diverses façons d'effectuer une série de décisions "oui-non" basées sur la présence ou absence de "oui" et de "non" sur diverses entrées, et éventuellement prise compte de l'historique des "oui" et des "non" précédents. Selon la manière dont vous interconnectez ces blocs logiques, vous pouvez construire un ordinateur, une calculatrice, un système de musique électronique, un voltmètre ou compteur, un écran de lecture de terminal de télévision, un téléviseur couleur générateur de barres de points, des démonstrateurs éducatifs ou l'un des milliers d'autres possibilités. Bien qu'une seule décision "oui-non" en elle-même ne soit généralement pas trop utile, la bonne combinaison de décisions groupées "oui-non" pris ensemble peuvent représenter un nombre,

un mot, une commande, une musique note, un signal de test ou pratiquement tout ce que vous pourriez aimer. [2]

Transistor-Transistor-Logic, également connu sous le nom de TTL inventé par James L. Buie en 1961. Les avantages importants du TTL sont son faible coût (aussi bas que 304 par paquet en quantités uniques sur le marché excédentaire), sa capacité à haut débit (20 MHz typique ; à 125 MHz avec des dispositifs spéciaux), sa capacité d'entraînement modérée et son immunité au bruit, ainsi que la disponibilité à l'échelle de l'industrie de centaines d'appareils différents. Cela vous donne une large sélection de blocs logiques simples et élaborés qui peuvent être directement interconnectés pour produire des fonctions uniques avec moins paquets que pratiquement tout autre système logique actuel. [2]

Cette famille utilise la technologie des transistors bipolaire sature par contre la technologie des transistors à effet de champ (MOS) utilisée par les familles NMOS, PMOS, CMOS. Elle représente une avancée par rapport à la famille de portes logiques DTL (Diode-Transistor Logic) apparue pour la première fois dans la seconde moitié des années 1950. Elle est actuellement en train de disparaître à cause de sa forte consommation d'énergie (par rapport aux circuits CMOS). [2]

1.4 La technologie CMOS

Complementary Metal Oxide Semiconductor, parfois appelé CMOS, est une technologie de fabrication de composants électroniques et, par conséquent, les composants produits à l'aide de cette technologie. La majorité de ces circuits logiques (NAND, NOR, etc.) sont similaires à ceux de la famille logique transistor-transistor (TTL), mais ces circuits constitué d'une paire de transistors MOSFET N et P que nous avons déjà expliqué disposés symétriquement et remplissant la même fonction. [3]

Dans les portes logiques CMOS, une collection de MOSFET de type n est disposée dans un réseau déroulant entre la sortie et le rail d'alimentation basse tension (V_{ss} ou assez souvent la masse). Au lieu de la résistance de charge des portes logiques NMOS, les portes logiques CMOS ont une collection de MOSFET de type p dans un réseau pull-up entre la

sortie et le rail à tension plus élevée (souvent appelé Vdd), comme illustré dans la figure 1.3 ci-dessous.

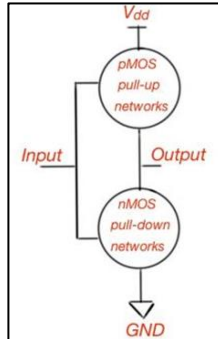


Figure 1.3 : CMOS pull up et pull down.

Ainsi, si un transistor de type p et de type n ont leurs grilles connectées à la même entrée, le MOSFET de type p sera activé lorsque le MOSFET de type n est désactivé, et vice-versa. Les réseaux sont agencés de sorte que l'un soit activé et l'autre désactivé pour n'importe quel modèle d'entrée. [3]

1.4.1 Les portes logiques CMOS

CMOS fait référence à une méthode ou une technologie particulière pour concevoir et construire des circuits. Le mot complémentaire signifie que les paires de transistors NMOS et PMOS sont liées faire des portes logiques.

1.4.1.1 L'inverseur

L'inverseur CMOS est un élément de base pour la conception de circuits numériques. Comme la figure 1.4 s'affiche :

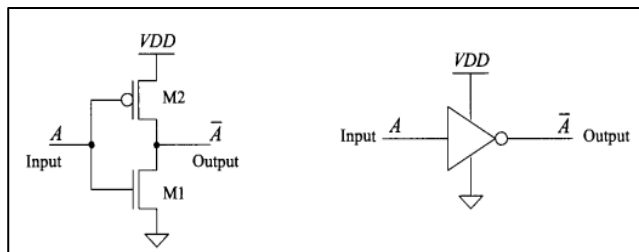


Figure 1.4 : schéma et symbole de l'inverseur

L'inverseur effectue l'opération logique de A à \bar{A} . Lorsque l'entrée de l'onduleur est connectée à la masse, la sortie est tirée vers VDD via le dispositif PMOS M2 (et M1 s'éteint).

Lorsque la borne d'entrée est connectée à VDD, la sortie est tirée vers masse via le dispositif NMOS M1 (et M2 s'éteint). L'inverseur CMOS a un rendement de tension oscille de VDD à la terre contrairement à d'autres familles logiques qui n'atteignent jamais tout à fait le niveau d'alimentation. De plus, la dissipation de puissance statique de l'inverseur CMOS est pratiquement nulle, l'inverseur peut être dimensionné pour donner des capacités d'alimentation et d'absorption égales, et la logique le seuil de commutation peut être définie en modifiant la taille de l'appareil. [4]

1.4.1.2 Porte NAND

La figure 1.5 montre le symbole logique de porte NAND avec tableau 1 de vérité :

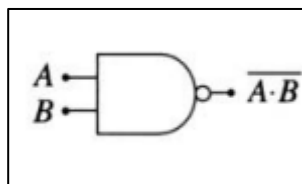


Figure 1.5 : schéma et symbole de porte NAND

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

Tableau 1 : table de vérité de porte NAND

Pour construire un circuit CMOS qui assure cette fonction, nous utiliserons deux paires complémentaires, une pour chacune des entrées A et B, et créons les NFET et PFET en fonction des sorties. Tout d'abord, notez qu'il n'y a qu'un seul cas où la sortie est un 0. Cela se produit lorsque les deux les entrées sont aux valeurs logiques 1. La traduction de cette observation en tensions indique alors que la tension de sortie si et seulement si les deux tensions d'entrée sont élevées, c'est-à-dire, puisque les NFET connectent le nœud de sortie à la masse, cela nécessite que les deux NFET soient connectés en série. Si soit la tension d'entrée est basse, indiquant alors que le nœud de sortie doit être connecté à l'alimentation. Pour répondre à ces cas, nous câblerons les deux PFET en parallèle. Combiner les exigences pour les FET aboutisse au circuit illustré à la Figure 1.6.

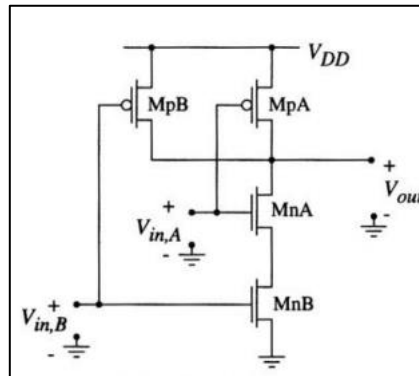


Figure 1.6 : schéma CMOS du NAND

Le fonctionnement logique du circuit peut être vérifié en travaillant en sens inverse. Considérons les NFET connectés en série MnA et MnB. Si les deux et sont élevés, alors ces transistors sont actifs et conduire le courant pendant que les deux PFET sont en coupure. Cela fournit un chemin de conduction solide pour terre et donne une tension de sortie de 0. Cependant, si A ou B est bas (soit individuellement soit en même temps), il n'y a pas de chemin vers la terre ; dans ce cas, au moins un dispositif à canal p conduit à l'alimentation, donnant une valeur de $V_{oh} = V_{dd}$. [4]

1.4.1.3 Porte NOR

La figure 1.7 montre la logique symbole et tableau 2 de vérité de la porte NOR. L'opération se caractérise par le fait qu'un 1 logique à l'une ou l'autre des entrées (ou aux deux entrées) provoque une sortie à 0 logique.

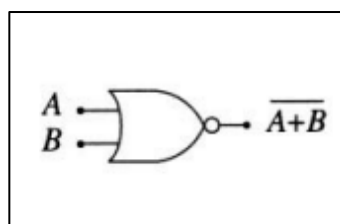


Figure 1.7 : schéma et symbole de porte NOR

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

Tableau 2 : table de vérité de porte NOR

Une porte NOR peut être construite en utilisant deux paires complémentaires comme le montre la figure 1.8.

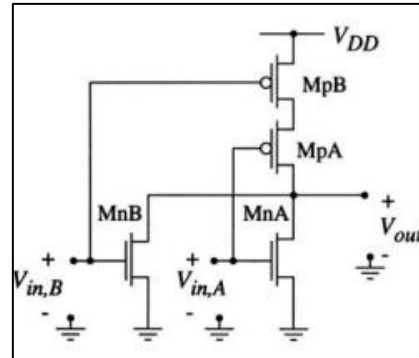


Figure 1.8 : schéma CMOS du NOR

L'entrée A est connectée à MnA et MpA, tandis que B contrôle MnB et MpB. Notez que les NFET sont connectés en parallèle, tandis que les PFET forment une chaîne en série. Pour comprendre le fonctionnement de la grille, nous examinons les états de conduction des transistors pour différentes tensions d'entrée et si alors MnA est activé et MpA est désactivé ; puisque MnA fournit un chemin conducteur depuis le sol à la sortie, le réglage active MnB ON et MpB OFF et également depuis la masse à la sortie, le réglage active MnB et désactive MpB et entraîne également Et, si les deux et sont élevés, alors les deux NFET sont activés et la sortie La tension est La seule combinaison d'entrées qui se produit est lorsque puisque les deux PFET sont activés alors que les deux NFET sont désactivés. [4]

1.4.1.4 Porte AND et OR

Si dans le schéma de la porte NOR, chaque transistor NMOS est remplacé par un transistor PMOS et inversement, on obtient un schéma qui donne théoriquement une porte AND. Pourtant, le circuit de la porte AND n'est pas réalisé de cette manière. Cela provient du fait que la source et le drain des transistors ne jouent pas des rôles complètement symétriques. Pour des raisons de consommation, les connexions avec le 0 sont toujours commandées par des transistors de type n et les connexions avec le 1 par des transistors de type p. Les circuits des portes AND et OR sont respectivement obtenus en combinant un circuit de la porte NAND et NOR avec un inverseur.

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. Le schéma de porte AND et La tableau 3 de vérité est donnée ci-dessous :

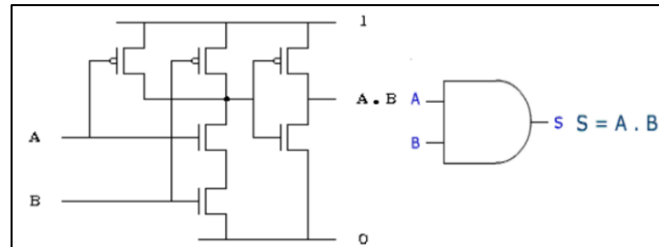


Figure 1.9 : symbole et schéma CMOS de la porte AND

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

Tableau 3 : table de vérité de porte AND

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. Le schéma de porte OR et La tableau 3 de vérité est donnée ci-dessous :

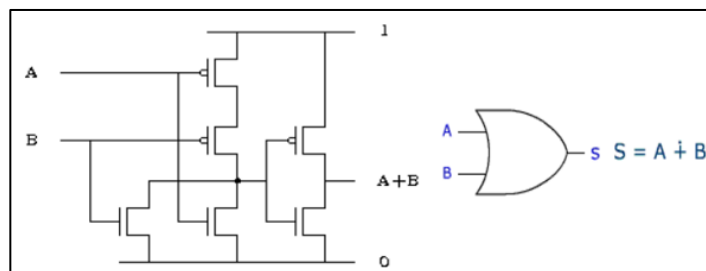


Figure 1.10 : symbole et schéma CMOS de la porte OR

A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

Tableau 4 : table de vérité de porte OR

1.4.1.5 XOR et XNOR

L'opération OU exclusif (XOR) n'est pas une fonction logique primitive, mais elle est utilisée si souvent qu'elle mérite un symbole logique spécial qui lui est propre. Par définition, la sortie est un 1 logique lorsque l'une des entrées est un 1 logique, mais est 0 dans le cas où les deux entrées sont à un état logique 1 simultanément. L'opération XOR est notée en prenant le complément du XOR donne l'opération Exclusive-NOR (XNOR). [6]

La Figure 1.11 montre le symbole logique pour XOR et XNOR avec le tableau 5 de vérité.

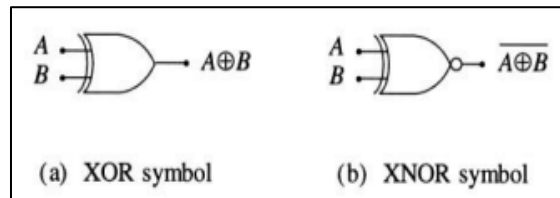


Figure 1.11 : symbole des portes XOR et XNOR

A	B	$A \oplus B$	$\overline{A \oplus B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

Tableau 5 : table de vérité des portes XOR et XNOR

La structuration logique générale discutée ci-dessus peut être utilisée pour créer les portes CMOS illustrées dans la figure 1.12. Les deux circuits utilisent les paires d'entrées et de sorte que des onduleurs sont nécessaires pour générer et à partir des entrées de base A et B.

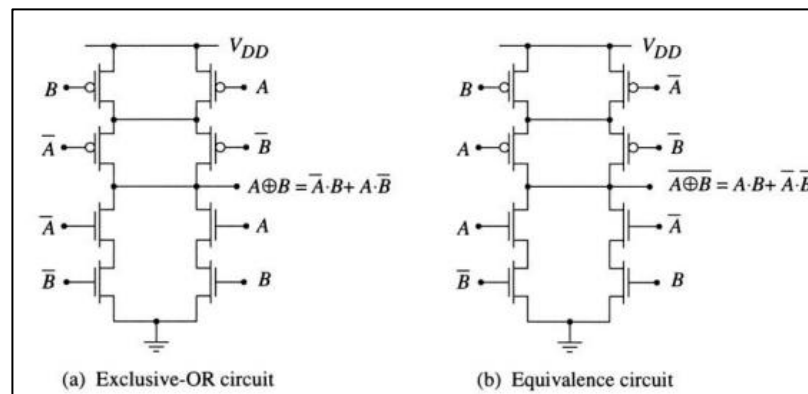


Figure 1.12 : Schéma CMOS des portes XOR et XNOR

1.4.2 Les circuits séquentiels

Les circuits logiques séquentiels contiennent des éléments de mémoire et la sortie dépend de la valeur actuelle de l'entrée et conditions de niveau d'entrée antérieures. Les éléments de base des éléments de mémoire sont des bascules qui peuvent contenir des valeurs binaires tant que l'appareil est alimenté. La sortie d'un logique séquentielle synchrone dépend des sorties des éléments de mémoire et contributions. [5]

1.4.2.1 SR latch :

Le latch S-R est un type de mémoire à deux entrées S (set) et R (reset), deux sorties Q et \bar{Q} et les sorties sont complémentaires les unes des autres comme la figure 1.13 montre :

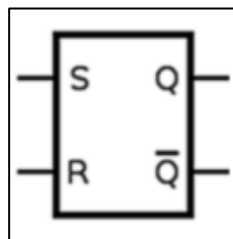


Figure 1.13 : schéma bloc de latch SR

Un latch S-R peut être construit avec les portes NOR ou NAND, la figure 1.14 montre S-R latch utilisant des portes NOR et l'autre schéma avec des portes NAND.

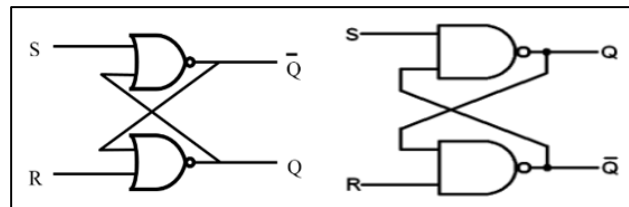


Figure 1.14 : circuit logique du latch SR

Le tableau 6 montre son tableau de vérité :

S	R	Q	\bar{Q}
0	0	1	1
0	1	1	0
1	1	1	0
1	0	0	1
1	1	0	1

Tableau 6 : table de vérité de latch SR.

D'autre part, la sortie de la porte NAND \overline{Q} est 0 si $Q = 1$ et 1 si $Q = 0$. Ainsi, \overline{Q} est également inchangé. Supposons que $S = 1$, $R = 0$ et $CLK = 1$, Cela générera 1 et 0 à les sorties Q et \overline{Q} respectivement. Ainsi, la bascule est mise à 1. Lorsque le l'horloge est à zéro. Cela rendra à son tour les sorties Q et \overline{Q} inchangées.

Les autres conditions du tableau ci-dessus des fonctions peuvent être vérifiées de la même manière. Notez que $S = 1$, $R = 1$ et $CLK = 1$ est une combinaison d'entrées invalides car cela fera les deux sorties et égal à 1. De plus, $Q+$ et $\overline{Q}+$ sont les sorties de la bascule après application de l'horloge (CLK).

Le tableau 7 montre la table de vérité du bascule RS. [5]

S	R	CLK	Q+	$\overline{Q}+$
0	0	1	Q	\overline{Q}
0	1	1	0	1
1	0	1	1	0
1	1	1	?	?
X	X	0	Q	\overline{Q}

Tableau 7 : table de vérité du bascule RS.

1.4.2.3 Bascule JK

La figure 1.17 montre le schéma fonctionnel d'une bascule JK où J, K et l'horloge sont les entrées de la bascule JK :

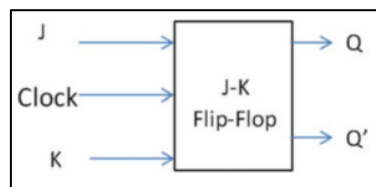


Figure 1.17 : schéma bloc du bascule JK

L'application de la bascule JK est le compteur et la fréquence diviseur. Le tableau 8 montre le tableau des caractéristiques de la bascule JK :

CLK	J	K	Q
↑	0	1	No change
↑	0	1	0
↑	1	1	1
↑	1	0	Complement

Tableau 8 : table de vérité du bascule JK

Les étapes suivantes décrivent les opérations de bascule JK :

- En réglant $J = K = 0$ et en appliquant une impulsion d'horloge à la bascule, la sortie Q ne change pas, si $Q = 0$ alors reste 0, ou si $Q = 1$ alors reste 1.
- En réglant $J = 0, K = 1$ et en appliquant une impulsion d'horloge à la bascule, puis en sortant Q passe à 0.
- En réglant $J = 1, K = 0$ et en appliquant une horloge à la bascule, la sortie Q change à 1.
- En réglant $J = K = 1$ et en appliquant une impulsion d'horloge, la sortie de la bascule est la Complément de la production actuelle ; cela signifie que si $Q = 0$ et en appliquant l'horloge, alors la sortie passe à 1 et si $Q = 1$ et l'application d'une impulsion d'horloge, la sortie sera passée à 0. [5]

1.4.2.4 Bascule T

La bascule T est un cas particulier de bascule JK, et en connectant les entrées J et K de JK bascule ensemble donne une bascule T ; La figure 1.18 montre un schéma fonctionnel de la bascule.

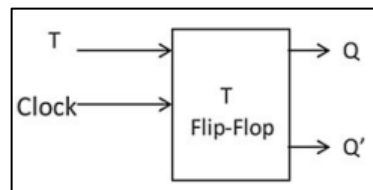


Figure 1.18 : schéma bloc du bascule T

Comme représenté sur la Tableau 9 si $T = 0$ et en appliquant une impulsion d'horloge, alors la sortie de la bascule T ne change pas, et si $T = 1$ et l'application de l'horloge, alors la sortie de la bascule devient le complément de la sortie actuelle.

CLK	T	Q
↑	0	No change
↑	1	Complement

Tableau 9 : table de vérité du bascule T

situation peut être évitée si les sorties des bascules ne changent pas tant que l'impulsion d'horloge ne revient pas à 0. Une Pour y parvenir, il faut s'assurer que les sorties des bascules sont affectées par la transition d'impulsion plutôt que la durée d'impulsion de l'entrée d'horloge. [11]

Pour comprendre ce concept, considérez les impulsions d'horloge illustrées à la figure 1.21 :

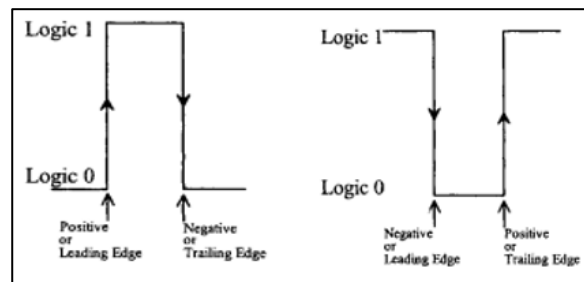


Figure 1.21 : Les impulsions d'horloge positive et négative

Il existe deux types d'impulsions d'horloge : positives et négatif. Une impulsion positive comprend deux transitions : du 0 logique au 1 logique et du 1 logique au 0 logique. Une impulsion négative passe également par deux transitions : 1 logique vers 0 logique et 0 logique à la logique 1.

Supposons qu'une impulsion positive est utilisée comme entrée d'horloge d'une bascule D. Avec l'entrée $D = 1$, la sortie de la bascule deviendra 1 lorsque l'impulsion d'horloge atteindra le 1 logique.

Supposons maintenant que l'entrée D passe à zéro mais que l'impulsion d'horloge est toujours à 1. Cela signifie que la bascule aura une nouvelle sortie, 0. Dans cette situation, la sortie d'une bascule ne peut pas être connecté à l'entrée d'un autre lorsque les deux bascules sont activées simultanément par la même entrée d'horloge. Ce problème peut être évité si la bascule est cadencée soit par le front montant ou descendant plutôt que le niveau de signal de l'impulsion.

La bascule maître-esclave est utilisé pour accomplir cela. La figure 1.22 montrent le schéma bloc et logique d'une bascule D maître-esclave.

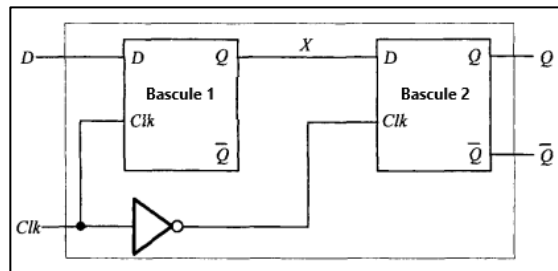


Figure 1.22 : Schémas bloc de bascule D maître esclave [11]

Une bascule maître-esclave contient deux bascules indépendantes. La bascule 1 fonctionne comme une bascule maître et la bascule 2 est esclave. Un inverseur est utilisé pour inverser l'entrée d'horloge de la bascule maître.

Supposons que le CLK est une impulsion positive et l'entrée D du bascule maître (bascule1) est 1 et l'entrée CLK = 1 (front montant). La sortie de l'inverseur sera appliquée un 0 à l'entrée CLK de la bascule esclave (bascule2). Ainsi, bascule 2 est désactivé, la bascule maître transmettra un 1 à sa sortie Q, donc X sera égal à 1.

Au front descendant de l'entrée CLK, l'entrée CLK de la bascule maître est à 0 donc la bascule 1 est désactivé. L'inverseur appliquera un 1 à l'entrée CLK du bascule 2. Ici 1 à l'entrée X (entrée D de bascule 2) sera transféré à la sortie Q de bascule 2. Quand le CLK repasse à 0, la bascule maître est séparée. Cela évite tout changement dans les autres entrées pour affecter la bascule maître. La bascule esclave aura la même sortie que le maître. [11]

1.4.3 Les circuits combinatoires

Un circuit combinatoire est un circuit où la sortie à tout moment ne dépend que de la combinaison actuelle d'entrées à ce moment-là sans tenir compte de l'état passé des entrées.

La figure 1.23 montre la représentation schématique d'un circuit combinatoire généralisé ayant n variables d'entrée et m variables de sortie ou simplement sorties.

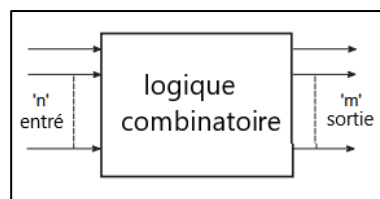


Figure 1.23 : Représentation schématique d'un circuit combinatoire

Le nombre de variables d'entrée étant n , il y a 2^n combinaisons possibles de bits en entrée. Chaque sortie peut être exprimée en termes de variables d'entrée par une expression booléenne. Les circuits logiques combinatoires qui peuvent être utilisés pour effectuer des opérations arithmétiques et connexes.

1.4.3.1 Les multiplexeurs

Un multiplexeur est un circuit combinatoire qui reçoit des informations binaires des lignes de données d'entrée de 2^n et la dirige vers une seule ligne de sortie. La sélection d'une ligne de données d'entrée particulière pour la sortie est déterminée par un ensemble de entrées de sélection. Un multiplexeur 2^n vers 1 a des lignes de données d'entrée de 2^n et n entrées lignes de sélection dont les combinaisons de bits déterminent quelles données d'entrée sont sélectionnées pour la sortie. La figure 1.24 montre le schéma fonctionnel d'un multiplexeur 2 vers 1. [5]

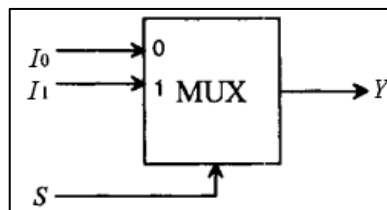


Figure 1.24 : schéma bloc d'un multiplexeur 2 vers 1.

Les deux entrées peuvent être sélectionné par une ligne de sélection, S . Lorsque $S = 0$, la ligne d'entrée 0 (I_0) sera présentée comme sortir. D'autre part, lorsque $S = 1$, la ligne d'entrée 1 (I_1) sera produite à la sortie.

Le tableau 11 montre la table de vérité du multiplexeur 2 vers 1. On peut montrer que $Y = \bar{S}I_0 + SI_1$.

S	I0	I1	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Tableau 11 : table de vérité du multiplexeur 2 vers 1.

La figure 1.25 montre le schéma logique du multiplexeur 2 vers 1 :

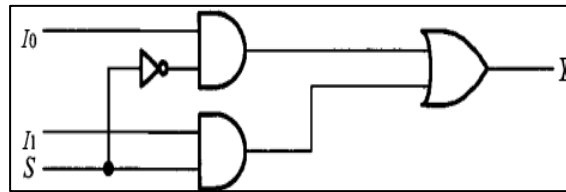


Figure 1.25 : le schéma logique du multiplexeur 2 vers 1

Dans un autre exemple, considérons le schéma fonctionnel du multiplexeur 4 vers 1 illustré à la figure 1.26.

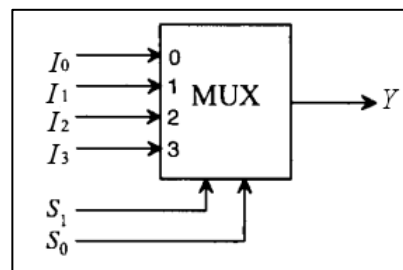


Figure 1.26 : schéma bloc d'un multiplexeur 4 vers 1.

Un multiplexeur 4 vers 1 ligne est illustré à la figure 1.27. Chacune des quatre entrées de données I_0 à I_3 , est appliquée à une entrée d'une porte AND. Les deux entrées de sélection S_1 et S_0 sont décodés pour sélectionner une porte AND particulière. Les sorties des portes AND sont appliquées à une seule porte OR pour fournir la sortie unique. Pour démontrer le fonctionnement du circuit, considérons le cas où $S_1=1$, $S_0 = 0$. La porte AND associée à l'entrée I_2 a deux de ses entrées égales à 1. La troisième entrée est connectée à I_2 et les trois autres portes AND ont au moins une entrée égale à 0, ce qui rend leurs sorties égales à 0. La sortie de la porte OR est maintenant égale à la valeur de I_2 , fournissant ainsi un chemin de l'entrée sélectionnée à la sortie.

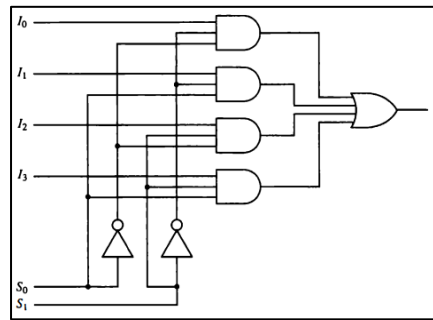


Figure 1.27 : le schéma logique du multiplexeur 4 vers 1

Le multiplexeur de ligne 4 vers 1 de la 1.27 a six entrées et une sortie. Le table de vérité décrivant le circuit nécessite 64 lignes puisque six variables d'entrée peuvent avoir 26 combinaisons binaires. Ce tableau est excessivement long et ne sera pas montré ici.

Une manière plus pratique de décrire le fonctionnement des multiplexeurs est au moyen d'une table de fonctions. Le tableau 12 montre les fonctions de ce multiplexeur :

S1	S0	Y
0	0	I0
0	1	I1
1	0	I2
1	1	I3

Tableau 12 : tableau des fonctions d'un multiplexeur 4 vers 1.

Le tableau 12 montre la relation entre les quatre entrées et la sortie unique en fonction des entrées de sélection S1 et S0. Lorsque les entrées de sélection sont égales à 00, la sortie Y est égale à l'entrée I0, lorsque les entrées de sélection sont égales à 01, l'entrée I1 a un chemin vers la sortie Y, et de même pour les deux autres combinaisons.

1.4.3.2 Les comparateurs

Les comparateurs binaires, également appelés comparateurs numériques ou comparateurs logiques, sont des circuits logiques combinatoires utilisés pour tester si la valeur représentée par un mot binaire est supérieure, inférieure ou égale à la valeur représentée par un autre mot binaire. Deux types de base de comparateur peuvent être utilisés : comparateur d'égalité et comparateurs de magnitude.

Le comparateur d'égalité, tel que celui illustré à la figure 1.28, est le comparateur logique multi bit le plus simple et peut être utilisé pour des circuits tels que des serrures électroniques et des dispositifs de sécurité où un mot de passe binaire composé de plusieurs bits est entré dans le comparateur pour être par rapport à un autre mot prédéfini. [6]

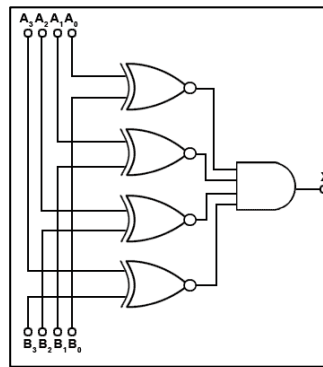


Figure 1.28 : le schéma logique d'un comparateur d'égalité

Dans la figure 1.28, un 1 logique sera présent à la sortie si les deux mots d'entrée correspondent, sinon la sortie reste à 0. Par conséquent, il n'y a qu'une seule combinaison d'entrée qui est correcte, et plus les mots d'entrée possèdent de bits, plus il y a de mauvaises combinaisons possibles. Avec des circuits supplémentaires pour le comptage, une sécurité supplémentaire peut être fournie en limitant le nombre d'essais avant que l'entrée ne soit inhibée. Le circuit du comparateur d'égalité est constitué d'une porte NOR exclusive (XNOR) par paire de bits d'entrée. Si les deux entrées sont identiques (les deux 1 ou les deux 0) une sortie logique 1 est obtenue. Les sorties des portes XNOR sont alors combinées dans une porte AND, dont la sortie sera 1, uniquement lorsque toutes les portes XNOR indiquent des entrées concordantes.

Le comparateur d'amplitude peut également être utilisé pour indiquer l'égalité, mais a deux autres sorties, une qui est logique 1 lorsque A est supérieur B, et une autre qui est logique 1 lorsque est inférieur au B. Comparateurs d'amplitude donc forment la base de la prise de décision dans les circuits logiques. Tout problème logique peut être réduit à une ou plusieurs (parfois plusieurs) décisions oui/non basées sur une paire de valeurs comparées. [6]

Un simple comparateur d'amplitude à 1 bit est illustré à la figure 1.29. La porte 1 produit la fonction $A > B$ et la porte 2 donne $A < B$ tandis que la porte 3 est une porte XNOR donnant une sortie d'égalité.

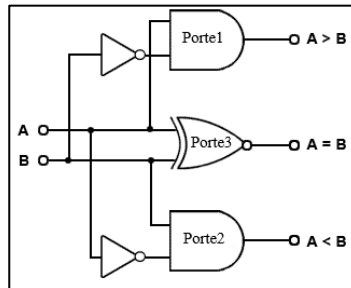


Figure 1.29 : le schéma logique d'un comparateur d'amplitude.

Le tableau 13 présente le tableau de vérité de ce comparateur :

A	B	A > B	A = B	A < B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

Tableau 13 : tableau de vérité de comparateur.

Ce circuit de base pour un comparateur d'amplitude peut être étendu pour n'importe quel nombre de bits, mais plus le circuit doit comparer de bits, plus le circuit devient complexe. Des comparateurs d'amplitude de circuit intégré sont disponibles et peuvent être utilisés pour fournir des comparaisons entre des mots multi-bits.

1.5 Conclusion

Dans ce chapitre, nous avons passé en revue les bases théoriques de la technologie CMOS et les différences portes logiques et les circuits séquentiel et combinatoires et sont principe de fonctionnement. Cette étude théorique est nécessaire avant la conception et la simulation final du circuit.

CHAPITRE 2 :

Les timer

2.1 Introduction

Dans le monde des microprocesseurs, les timers jouent un rôle crucial dans la gestion de diverses tâches et assurent un fonctionnement efficace. Les temporisateurs ou les timers sont des composants essentiels qui permettent un comptage et une synchronisation précis dans les systèmes à microprocesseur.

Dans ce chapitre on va explorer les subtilités des timers à microprocesseur, leurs fonctionnalités et leur importance dans les applications technologiques modernes et les blocs constituant de ces timers.

2.2 Les timers

Les timers à microprocesseur sont des dispositifs matériels intégrés aux microprocesseurs utilisés pour mesurer, contrôler et gérer le temps dans les systèmes informatiques. Ils sont chargés de générer des intervalles de temps précis, de mesurer des durées et de synchroniser des événements dans un système à microprocesseur. Ils sont souvent programmables et offrent un large éventail de fonctionnalités pour prendre en charge diverses applications. Ils peuvent être utilisés pour des applications telles que la génération de signaux d'horloge, la temporisation d'événements, la mesure de la durée d'exécution des programmes, etc. [7]

2.3 Types des timers à microprocesseur

Les timers à microprocesseur peuvent être classés en différents types en fonction de leurs caractéristiques de fonctionnement et de leurs fonctionnalités. Certains types courants de timers à microprocesseur comprennent :

2.3.1 Timers d'intervalle

Ces timers sont conçus pour mesurer avec précision l'intervalle de temps entre deux événements. Ils sont fréquemment utilisés dans des applications telles que la synchronisation d'événements, la génération d'impulsions et les systèmes de contrôle basés sur le temps. [13]

2.3.2 Timers de surveillance (timer chien de garde)

Les timers de chien de garde sont essentielles pour garantir la fiabilité du système et la tolérance aux pannes. Ils surveillent le système à microprocesseur pour tout dysfonctionnement ou comportement anormal et lancent des actions correctives si nécessaire. [7]

2.3.3 Horloges en temps réel (RTC)

Les RTC combinent des fonctionnalités de timer avec des capacités de suivi de la date et de l'heure. Ils sont largement utilisés dans les applications nécessitant un chronométrage précis, telles que l'enregistrement de données, la planification et les systèmes d'alarme. [7]

2.3.4 Timers PWM

Les timers de modulation de largeur d'impulsion (PWM) génèrent des signaux numériques avec des cycles de service variables. Ils sont largement utilisés dans des applications telles que la commande de moteur, la gestion de l'alimentation et la commande d'éclairage. [8]

2.3.5 Compteurs d'événements

Les compteurs d'événements suivent et enregistrent le nombre d'occurrences d'événements spécifiques. Ils trouvent des applications dans des domaines tels que la mesure de fréquence, la détection de position et le traitement numérique du signal. [13]

2.4 Importance des timers à microprocesseur

Les timers à microprocesseur sont des composants vitaux des systèmes électroniques modernes pour plusieurs raisons :

-Synchronisation de précision : les timers fournissent une synchronisation précise et cohérente, permettant des opérations synchronisées au sein de systèmes à microprocesseur. Cette précision est cruciale dans diverses applications qui nécessitent une coordination précise des événements, une synchronisation ou des actions sensibles au facteur temps.

-Gestion des tâches : les timers facilitent le multitâche en gérant les tâches urgentes et en allouant efficacement les ressources système. Ils permettent au microprocesseur de

hiérarchiser les tâches, de planifier les opérations et d'assurer des performances optimales du système.

-Contrôle du système : les timers jouent un rôle important dans le contrôle et l'automatisation du système. Ils permettent au microprocesseur de déclencher des actions ou des événements spécifiques à des intervalles prédéterminés, aidant à automatiser les processus, à surveiller les états du système et à mettre en œuvre des protocoles basés sur le temps.

-Efficacité énergétique : les timers contribuent à l'efficacité énergétique en permettant des techniques de gestion de l'alimentation. Par exemple, un microprocesseur peut utiliser des timers pour passer en mode basse consommation ou ajuster dynamiquement les fréquences d'horloge, réduisant ainsi la consommation d'énergie lorsqu'il est inactif ou pendant les périodes de faible activité. [14]

2.5 Applications des timers à microprocesseur

Les timers à microprocesseur sont largement utilisés dans une large gamme d'applications, y compris, mais sans s'y limiter :

1. Dans les systèmes d'automatisation industrielle, les timers à microprocesseur sont utilisés pour un contrôle précis des processus de fabrication, la synchronisation des opérations et la synchronisation des événements critiques. Ils permettent un séquençage, une planification et une coordination précis de divers composants dans un environnement automatisé.
2. Les systèmes embarqués dépendent fortement des timers à microprocesseur pour les opérations et le contrôle en temps réel. Ces timers aident à gérer les processus en temps réel, la planification des tâches et les opérations sensibles au facteur temps, garantissant le bon fonctionnement des systèmes embarqués utilisés dans des secteurs tels que l'automobile, l'aérospatiale et les dispositifs médicaux.
3. Les temporisateurs à microprocesseur jouent un rôle crucial dans les systèmes de communication en facilitant une synchronisation précise de la transmission des données, la synchronisation des protocoles réseau et un chronométrage précis. [14]

2.6 Les blocs constituent le timer d'un microprocesseur

Les blocs de timer de microprocesseur sont des composants essentiels dans les systèmes informatiques pour la gestion du temps et la synchronisation des opérations. Ces blocs de généralement intégrés dans les microprocesseurs modernes et offrent des fonctionnalités de comptage précis et de temporisation. Un bloc de timer de microprocesseur se compose généralement de plusieurs éléments clés. Voici quelques-uns des blocs couramment présents :

2.6.1 Compteur 3bit

Un compteur est un circuit séquentiel qui permet de dénombrer des impulsions appliquées sur son entrée d'horloge (clock) et de restitué sur ces sorties des informations sous forme binaire. [12]

Un compteur 3 bits est un circuit numérique qui peut compter de 0 à 7, en utilisant trois bascules ou registres. Il est capable de représenter huit états différents, correspondant aux nombres binaires 000, 001, 010, 011, 100, 101, 110 et 111. Le compteur passe incrémentalement d'un état à l'autre de manière cyclique, suivant la séquence de comptage binaire. Chaque bascule représente un bit du compteur et leurs sorties sont connectées pour former la représentation des nombres binaires. [12]

Le compteur 3 bits peut être implémenté à l'aide de divers composants logiques numériques, tels que des bascules ou des registres. Le type de bascule le plus couramment utilisé dans les compteurs est la bascule D, bien que d'autres types tels que les bascules JK ou les bascules T puissent également être utilisés comme illustre dans la figure 2.1 :

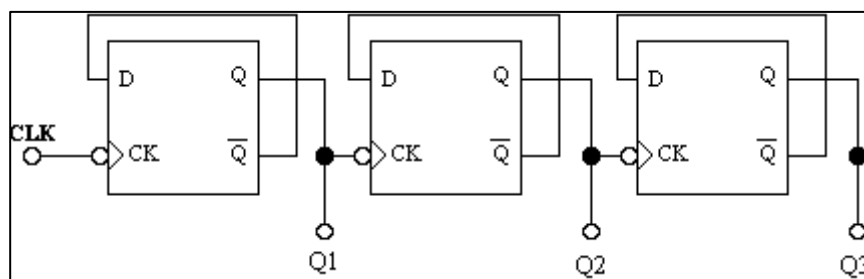


Figure 2.1 : Schéma bloc d'un compteur asynchrone 3 bit.[12]

Le tableau 14 présente un exemple de compteur 3 bits avec ses sorties binaires correspondantes :

Etat	Q2	Q1	Q0	Décimal
000	0	0	0	0
001	0	0	1	1
010	0	1	0	2
011	0	1	1	3
100	1	0	0	4
101	1	0	1	5
110	1	1	0	6
111	1	1	1	7

Tableau 14 : Les sorties binaires d'un compteur 3 bit

2.6.2 Bloc PWM

Le bloc PWM (Pulse Width Modulation) est un composant couramment utilisé dans les systèmes numériques pour générer des signaux analogiques précis et ajustables. PWM est une technique qui permet de contrôler la valeur moyenne d'un signal numérique en faisant varier la largeur de ses impulsions.[9]

Le bloc PWM se compose généralement d'un comparateur et d'un compteur. Le comparateur compare un signal de référence, généralement une rampe ou une sinusoïde, avec un signal modulant. Le signal de modulation détermine le rapport cyclique de la forme d'onde PWM. Le rapport cyclique représente le rapport entre la largeur d'impulsion (la durée pendant laquelle le signal est haut) et la période totale de la forme d'onde. Il est souvent exprimé en pourcentage.

Par exemple, un rapport cyclique de 50 % signifie que le signal est haut pendant la moitié de la période et bas pendant l'autre moitié comme la figure 2.2 montre :

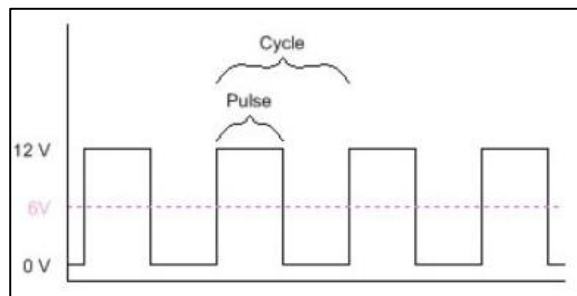


Figure 2.2 : Signal PWM avec un rapport cyclique 50 % [9]

Le compteur du bloc PWM génère le signal de synchronisation ou l'horloge qui détermine la fréquence de la forme d'onde PWM. La fréquence est déterminée par la source d'horloge du compteur et sa valeur. Les blocs PWM sont largement utilisés dans diverses applications, notamment la commande de moteur, les convertisseurs de puissance, les amplificateurs audios et la gradation des LED. En ajustant le rapport cyclique de la forme d'onde PWM, la puissance ou la tension moyenne délivrée à une charge peut être contrôlée. Par exemple, dans les applications de commande de moteur, le signal PWM peut être utilisé pour réguler la vitesse d'un moteur en faisant varier le rapport cyclique. Un cycle de service plus élevé fournit plus de puissance au moteur, ce qui entraîne une augmentation de la vitesse, tandis qu'un cycle de service inférieur réduit la puissance et ralentit le moteur. Les blocs PWM sont implémentés dans le matériel à l'aide de modules PWM dédiés disponibles dans les microcontrôleurs, les processeurs de signaux numériques (DSP) ou les circuits intégrés spécifiques à l'application (ASIC). Ces modules fournissent souvent des fonctionnalités supplémentaires telles que la génération de temps mort, la protection contre les pannes et les options de synchronisation pour améliorer leurs fonctionnalités. En utilisant un bloc PWM, un contrôle précis des signaux analogiques peut être obtenu dans les systèmes numériques, offrant flexibilité et efficacité dans diverses applications. [9]

2.6.3 Bloc contrôleur de modulo

Un contrôleur de modulo permet de régler la valeur maximale de comptage dans un système ou une application. Il est généralement utilisé dans des contextes où il est nécessaire de créer des boucles ou des séquences de valeurs qui se répètent cycliquement. Le contrôleur de modulo fonctionne en utilisant une valeur de compteur qui s'incrémente à chaque itération ou chaque événement. Lorsque le compteur atteint la valeur maximale définie par le

contrôleur de modulo, il est remis à zéro pour recommencer le cycle. Cela permet d'assurer que le comptage reste dans une plage spécifique déterminée par la valeur maximale. [10]

Par exemple, si un contrôleur de modulo est configuré avec une valeur maximale de 10, le compteur passera de 0 à 9 et reviendra ensuite à 0. Ainsi, une séquence de 10 valeurs sera générée de manière répétitive. Cela peut être utile dans de nombreuses situations, telles que le contrôle de l'accès à une ressource, la gestion de tâches périodiques ou la création de motifs cycliques dans une animation ou un affichage.

En ajustant la valeur maximale du contrôleur de modulo, il est possible de définir différentes longueurs de séquence ou de faire varier la périodicité des boucles. Cela offre une flexibilité pour adapter le comportement du système en fonction des besoins spécifiques de l'application.

2.7 Conclusion

Dans ce chapitre, nous avons passé en revue des subtilités des timers à microprocesseur, leurs fonctionnalités et leur importance et les blocs constituant de ces timers. Cette étude théorique est nécessaire avant la conception, la simulation et les tests pratique dans le chapitre suivant qui mettra en œuvre le système proposé.

CHAPITRE 3 :

Simulation et Layout

3.1 Introduction

Dans ce chapitre, nous développerons les étapes nécessaires à la réalisation du système de timer proposé. Dans un premier temps, nous présentons le logiciel nécessaire pour la simulation après les schémas constituant les blocs de timer proposée. A la fin, nous présentons les résultats de simulation et le dessin de masque globale du timer.

3.2 Présentation du logiciel Microwind

Microwind est un logiciel de conception de circuits intégrés (IC) utilisé dans l'industrie de la microélectronique. Il est largement utilisé par les ingénieurs électroniques pour concevoir, simuler et analyser des circuits intégrés à petite échelle.

Ce logiciel est particulièrement utile pour la conception de circuits intégrés CMOS et offre une interface conviviale pour créer et éditer des schémas de circuits. Il permet aux utilisateurs de dessiner des structures de transistors, des interconnexions et des dispositifs sur une puce.

Microwind offre également des outils de simulation qui permettent de vérifier le fonctionnement du circuit avant sa fabrication. Il propose des modèles de transistors et de composants électroniques, ainsi que des simulateurs pour évaluer les performances électriques du circuit, tels que le temps de propagation, la consommation d'énergie et les interférences électromagnétiques.

En utilisant Microwind, les concepteurs de circuits intégrés peuvent optimiser leurs conceptions, effectuer des analyses de sensibilité et explorer différentes configurations pour obtenir les performances souhaitées. Le logiciel facilite également la génération de fichiers de fabrication, tels que les masques photo lithographiques, pour la production des circuits intégrés.

Microwind est un outil puissant pour les concepteurs de circuits intégrés et il est largement utilisé dans l'industrie électronique pour accélérer le processus de développement et améliorer la fiabilité des circuits intégrés.

3.3 Structure logique global du timer

Dans cette section, nous montrons en détail le schéma logique globale de notre timer proposés nécessaires pour réaliser le circuit final.

Ce timer contient les caractéristiques suivantes :

- Comptage et décomptage 3 bit
- Modulo réglable
- Set / Reset
- Sortie PWM

D'un point de vue externe, on peut voir notre circuit intégré comme un composant à part entière de la forme présente dans la figure 3.1 ci-dessus :

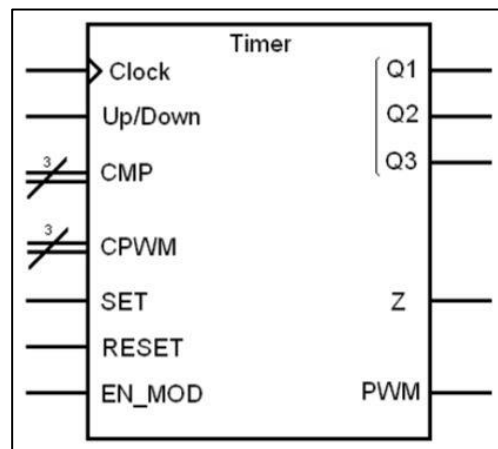


Figure 3.1 : Schéma bloc du timer

Détail des ports :

- Clock : entrée de l'horloge, active sur front montant.
- Up/Down : sélection compteur/décompteur. 0 -> compteur, 1 -> décompteur.
- CMP : valeur de modulo, sur 3 bits.
- CPWM : valeur de seuil pour génération de signal PWM.
- Set : mise à « 111 » du compteur, actif au niveau bas.
- Reset : mise à « 000 » du compteur, actif au niveau haut.

- EN_MOD : activation du modulo variable, actif au niveau haut.
- Q1-Q2-Q3 : sortie du compteur. Bit de poids faible : Q1.
- Z : flag Zéro. Vaut 1 quand la sortie du compteur vaut « 000 ».
- PWM : sortie de signal PWM.

La figure 3.2 présente le schéma logique globale de notre timer :

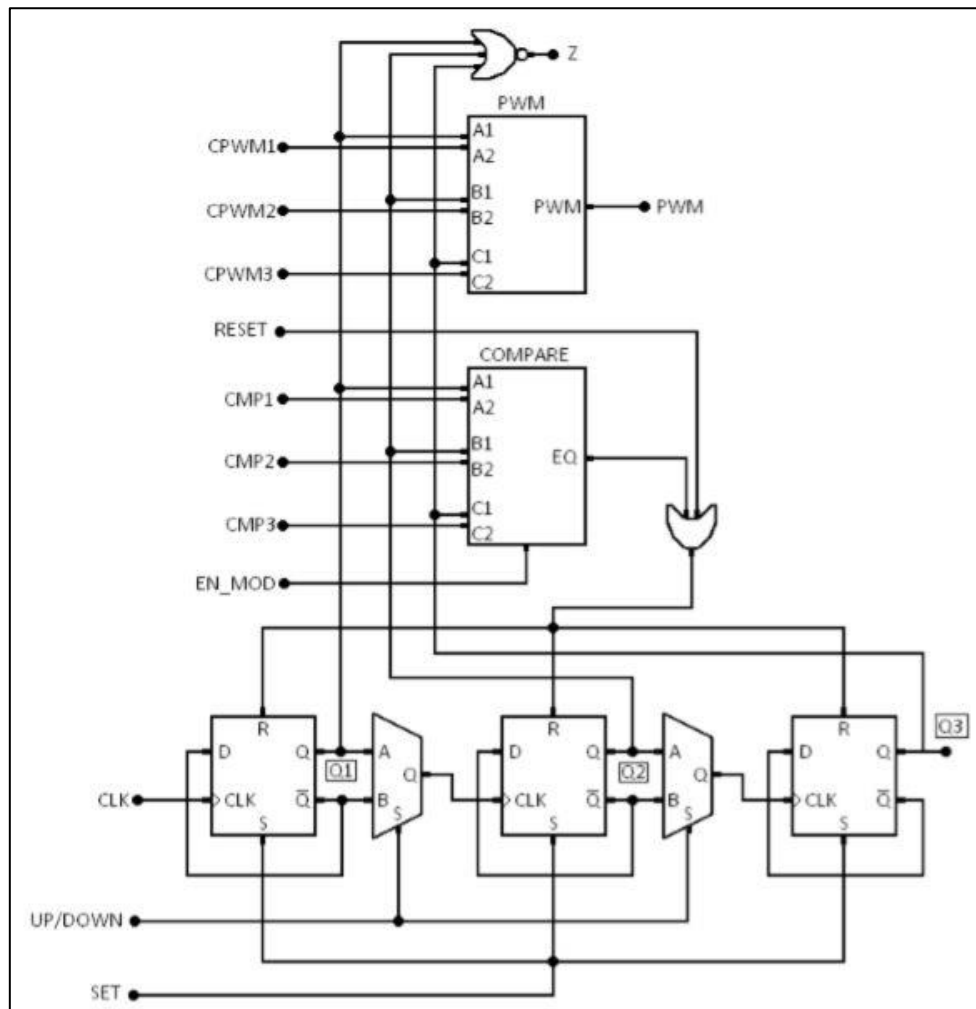


Figure 3.2 : Schéma logique globale du timer

3.3.1 Principe de fonctionnement

Un périphérique timer est un système complexe composé de plusieurs sous-systèmes simples. La partie fondamentale d'un tel périphérique est le comptage, qui, une fois associée à une horloge, lui permet de mesurer le temps.

Dans cette logique nous avons commencé le projet par l'implémentation d'un compteur 3 bits à l'aide de bascules D maître esclave que nous avons déjà expliqué dans le chapitre 1. Ce schéma a ensuite été amélioré afin d'implémenter la fonction de Set.

Dans un second temps, nous avons souhaité ajouter la fonctionnalité de décomptage. Pour

cela nous avons décidé d'intercaler un multiplexeur entre deux bascules D. Ceux-ci possèdent deux voies d'entrées, une sortie, et une voie de sélection.

Ensuite, afin de pouvoir régler la valeur maximum de comptage, nous avons développé un bloc de test d'égalité de deux mots de 3 bits, celui-ci contrôlant la fonctionnalité de Reset des bascules D.

Un bloc semblable a aussi été pensé, celui-ci surveillant l'inégalité de deux mots de 3 bits, et ayant pour but de générer alors un signal PWM entièrement réglable.

Pour finir, afin de coller au plus près des besoins d'un microcontrôleur, une simple logique combinatoire détecte une sortie nulle du Timer, un « flag Z ». Celui-ci peut être utile au microcontrôleur pour le déclenchement d'interruptions.

3.4 Dessin de masques et simulation des blocs constituant le timer

3.4.1 Bascule D maître esclave

Nous utilisons des bascules D maître esclave, actives sur front montant d'horloge, avec les fonctionnalités de Set et de Reset. Mises en cascade, 3 bascules créent un compteur 3 bits asynchrone qui sera la base de notre circuit. Les fonctionnalités de Set et Reset sur chaque bascule permettra le forçage à 1 ou à 0 du compteur.

La figure 3.4 présente layout du bascule D :

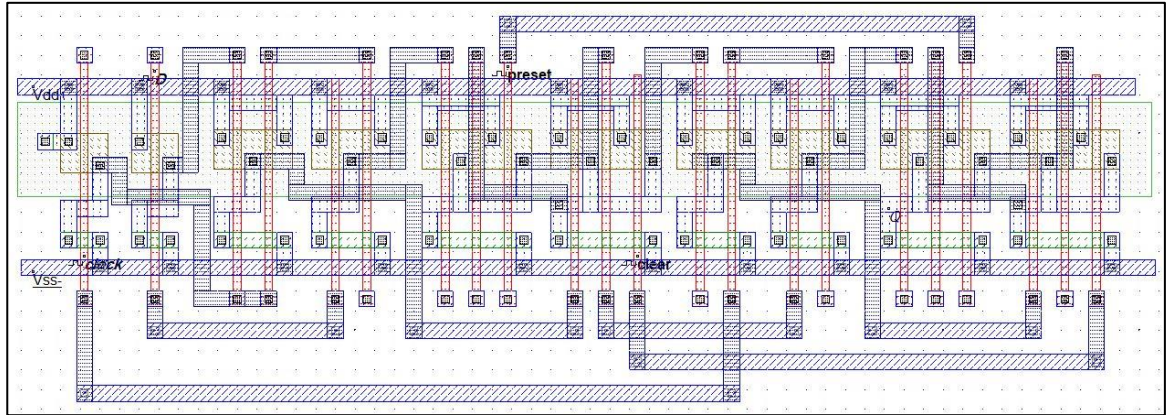


Figure 3.4 : Layout de bascule D maitre esclave avec set/reset

La figure 3.5 présente résultat de simulation de bascule D :

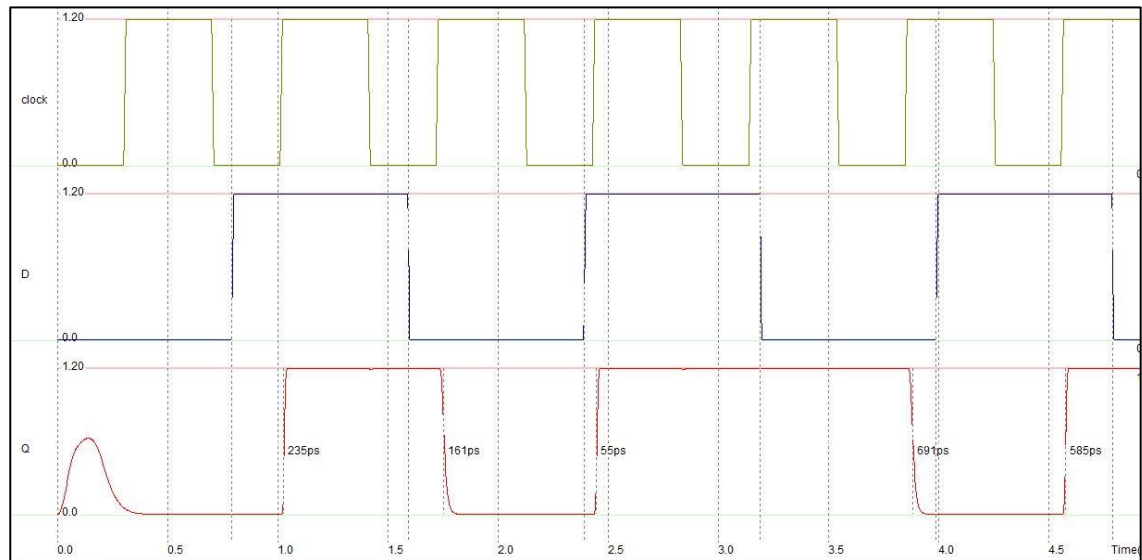


Figure 3.5 : Résultat de simulation de bascule D maitre esclave

3.4.2 Multiplexeur

Le multiplexeur est intercalé entre chaque bascule. Il va sélectionner la sortie Q ou nQ comme entrée d'horloge de la bascule. Lorsque l'horloge est attaquée par le signal Q, on obtient un décompte. Lorsqu'elle est attaquée avec le signal nQ. Les deux multiplexeurs du circuit sont contrôlés par l'entrée « Up/Down ».

La figure 3.6 montre le schéma logique de multiplexeur :

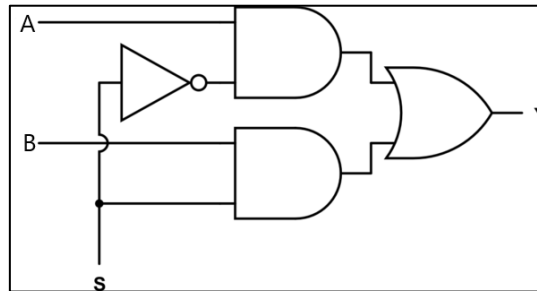


Figure 3.6 : Schéma bloc de multiplexeur 2 vers 1

La figure 3.7 montre layout de multiplexeur 2 vers 1 :

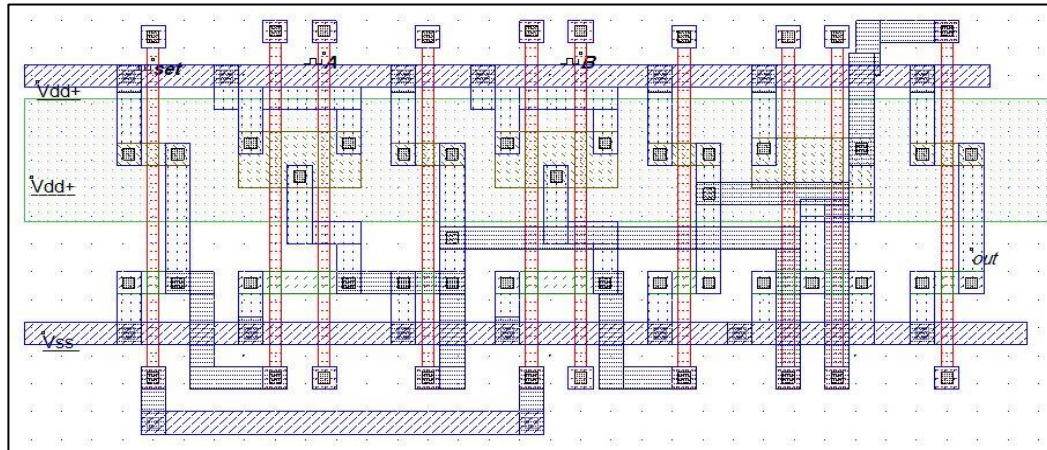


Figure 3.7 : Layout de multiplexeur 2 vers 1

La figure 3.8 montre le résultat de simulation de multiplexeur :

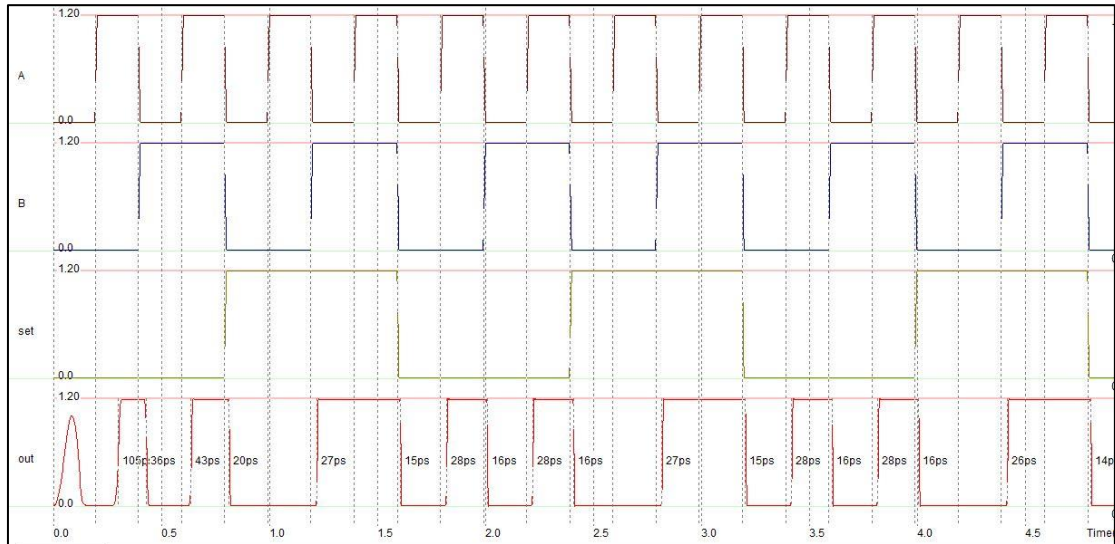


Figure 3.8 : Résultat de simulation de multiplexeur 2 vers 1

3.4.3 Contrôleur de modulo

Le contrôleur de modulo illustré dans la figure 3.9 permet de régler la valeur maximum de comptage. Pour cela la sortie est comparée à 3 bits (CMP, ici C1, C2, C3). Une fois cette valeur égale à la valeur courante du compteur, l'égalité déclenche un Reset du compteur. Pour que cela fonctionne, il faut être en mode « comptage » (Up/Down = 0) et activer le modulo (EN_MOD = 1).

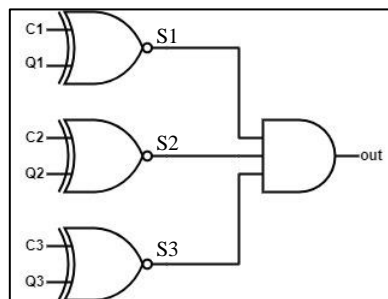


Figure 3.9 : Schéma logique de comparateur 3bit

La figure 3.10 présente layout de contrôleur de modulo :

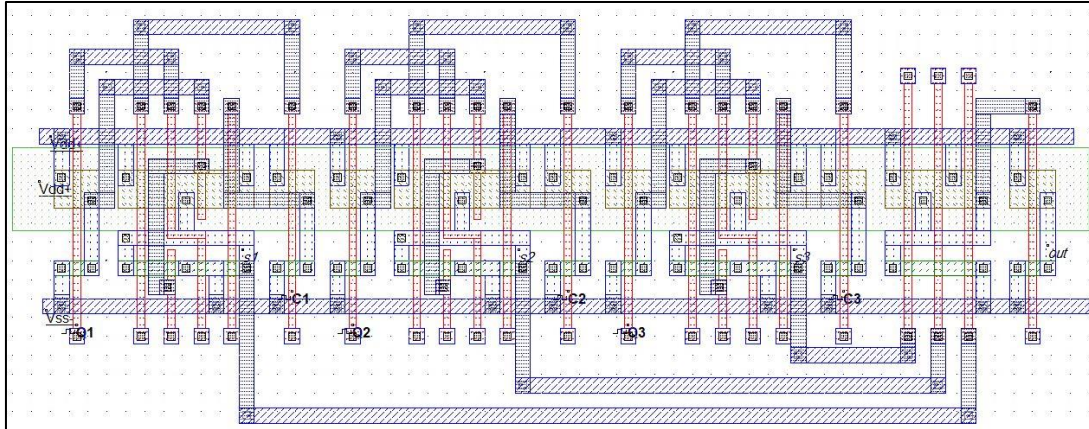


Figure 3.10 : Layout de comparateur 3bit

La figure 3.11 montre le résultat de simulation de contrôleur de modulo :

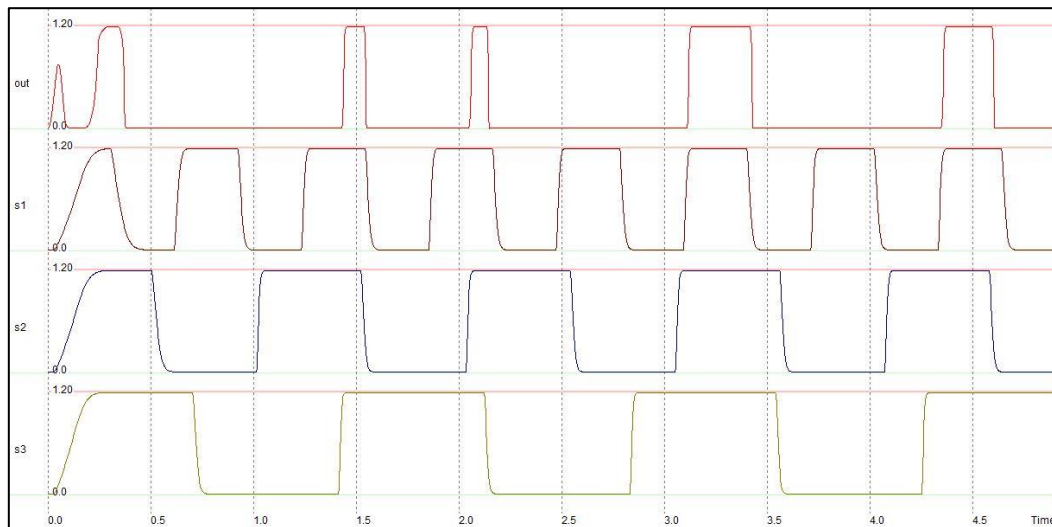


Figure 3.11 : Résultat de simulation de comparateur 3bit

3.4.4 PWM

Le module PWM compare les 3 bits du compteur (Q1 à Q3) à 3 bits de référence (CPWM). Tant que la valeur du compteur est inférieure à la référence, la sortie du PWM est à 1. Sinon elle passe à 0.

Par exemple, pour une valeur de référence à 2, si le compteur est en modulo 8, on aura un rapport cyclique de sortie du PWM à 25%.

Dans un premier temps, nous avons créé un module pour assurer cette fonction. Hélas, nous avons omis quelques aspects de logique élémentaire et de ce fait, notre comparateur, trop simpliste, s'est avéré défectueux.

Voici le schéma dans la figure 3.12 du comparateur que nous aurions dû implémenter juste la sortie $A < B$ de ce comparateur (7485) car ce schéma est très compliqué avec trop grand nombre de portes logiques à implémenter. La fréquence du PWM sera égale à la fréquence de l'horloge divisée par le modulo du compteur.

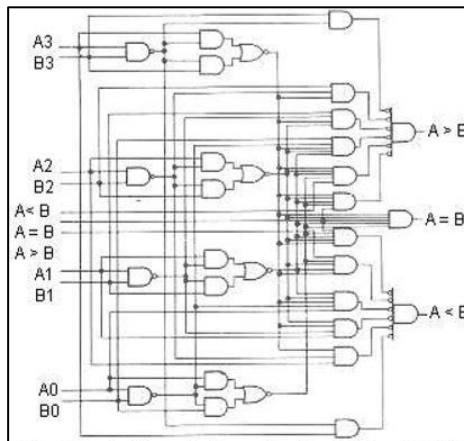


Figure 3.12 : Schéma logique de comparateur 7485

La figure 3.13 présente layout de PWM :

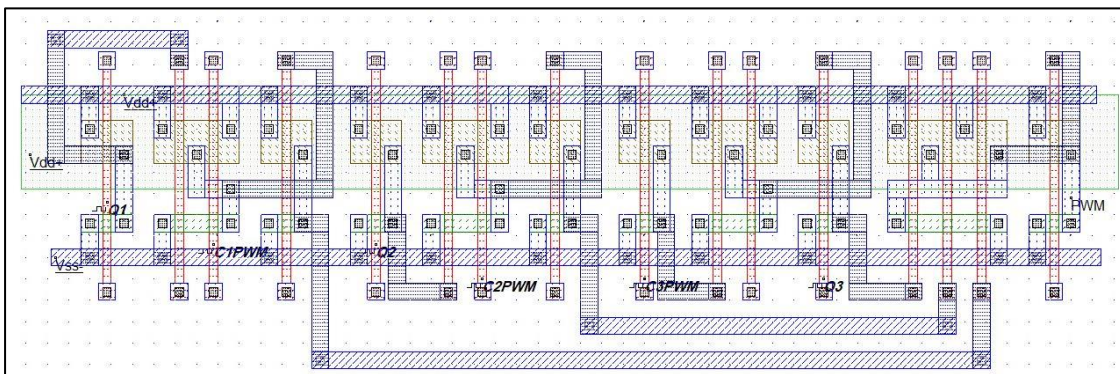


Figure 3.13 : Layout de bloc PWM

La figure 3.14 présente résultat de simulation du bloc PWM :

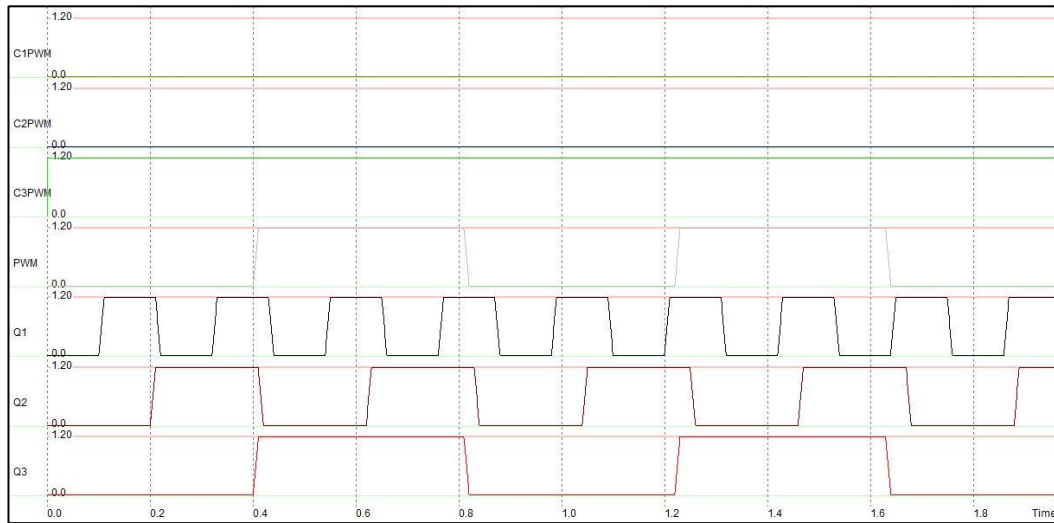


Figure 3.14 : Résultat et simulation du bloc PWM

3.5 Dessin de masque et simulation globale du timer

3.5.1 Circuit globale

La figure 3.15 montre le circuit CMOS globale de timer :

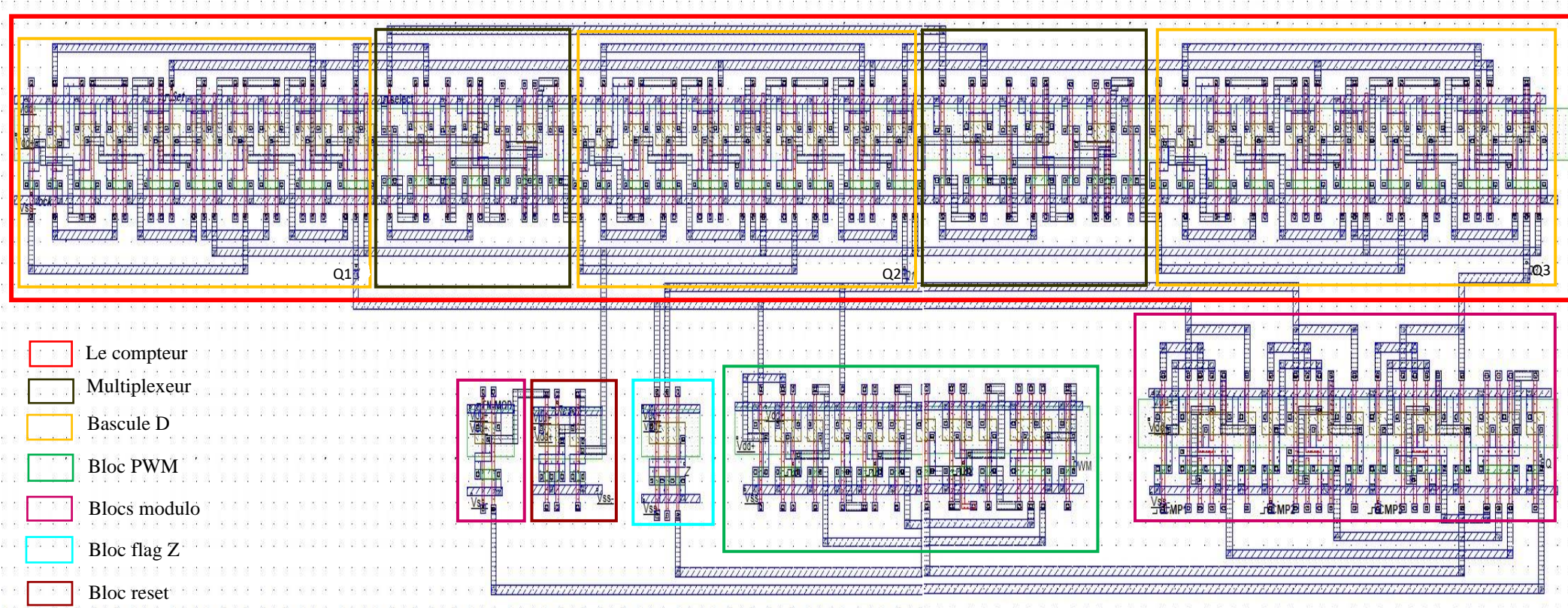


Figure 3.15 : Layout globale de timer

3.5.2 Simulation globale

- La figure 3.16 montre le mod comptage :

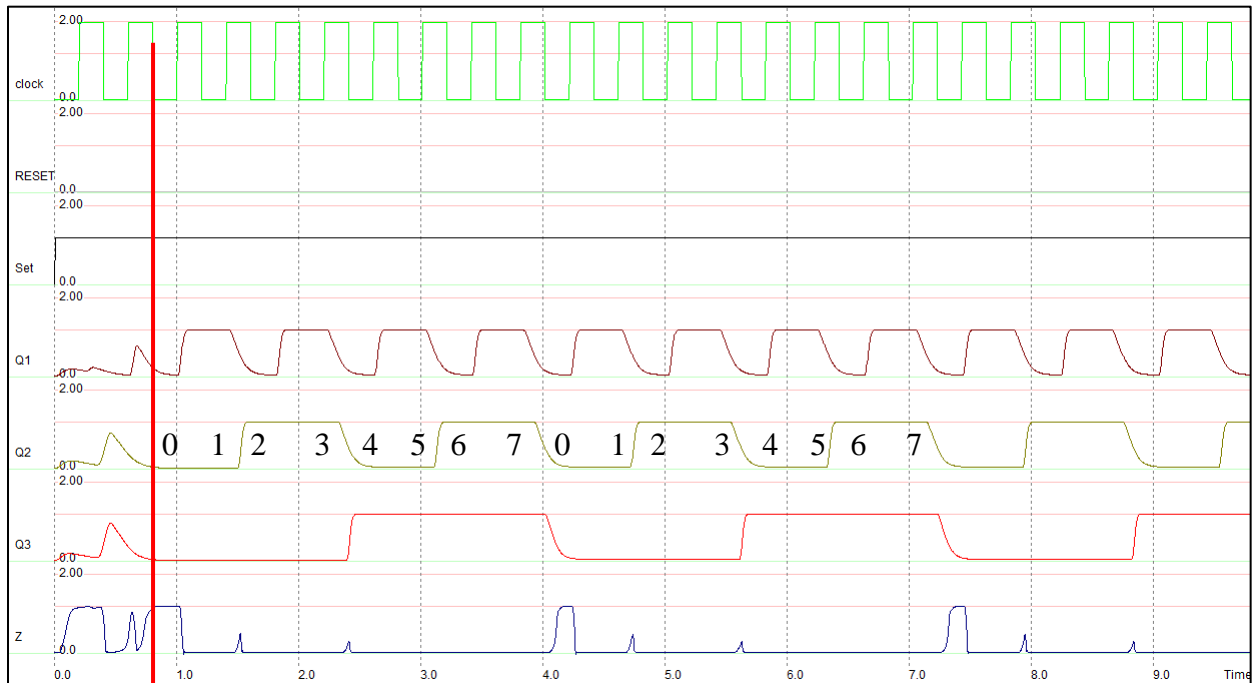


Figure 3.16 : Chronogramme compteur.

Le compteur est en fonctionnement simple, sans modulo. Il compte donc de 0 à 7. On observe le Reset asynchrone en début de chronogramme qui réinitialise les bits Q à 0 et le Set asynchrone en fin de chronogramme qui met les bits Q à 1.

On observe également le flag Z qui se lève lorsque la valeur du compteur est 0. On s'aperçoit qu'il se lève brièvement lors de certains changements de valeur (par exemple lors du passage de 1 à 2 et de 3 à 4). Ce phénomène est dû au temps de propagation du signal à travers les bascules qui ne commutent pas en même temps. Cela paraît gênant, mais vu la très faible durée de ces derniers (de l'ordre de la nanoseconde), en fonctionnement à plus basse fréquence, ces pics sont négligeables (on pourrait même rajouter un filtre passe bas pour les enlever).

- La figure 3.17 montre le mod décomptage :

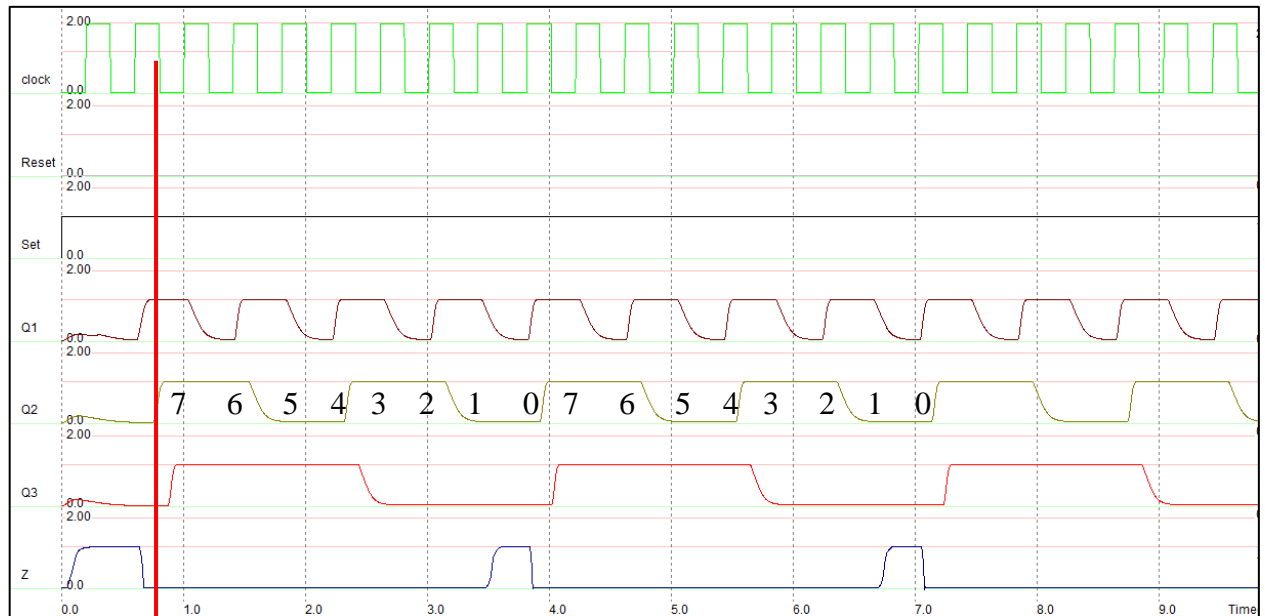


Figure 3.17 : Chronogramme décompteur.

Le décompteur est ici aussi en fonctionnement simple. Il décompte donc de 7 à 0. On observe le Reset asynchrone en début de chronogramme qui réinitialise les bits Q à 0. Cela vient du fait qu'en décomptage, c'est la sortie Q (et non Q en comptage) qui est branchée sur l'entrée CLK de la bascule suivante. Ainsi, après un set, les bits bascule de 111 à 000.

3.6 Conclusion

Dans ce chapitre nous avons présenté tous les blocs constituant le timer d'un microprocesseur et tous les layout avec le logiciel microwind et sont simulation.

Après avoir présenté en détail les différentes étapes de la conception et de la réalisation, nous pouvons conclure que notre objectif était complet les résultats obtenus sont très satisfaits.

Conclusion générale

Notre objectif dans ce présent travail consiste en la conception et la réalisation d'un timer de microprocesseur en technologie CMOS, pour améliorer le contrôle du temps de microprocesseur

Dans le premier chapitre nous avons présenté l'historique de technologie CMOS ainsi les transistors MOS et la technologie TTL. Nous avons également présenté les portes logiques et les différents bascules et les circuits combinatoires sachant que les multiplexeurs et les comparateurs et sont principe de fonctionnement.

Dans la deuxième partie, on a présenté les timers contant généralité, types et sont rôle avec les blocs constituent le timer d'une manière générale.

Au final, nous avons réalisé les dessins de masques et les simulations à des différents blocs du timer, tels que la bascule D maître esclave et le multiplexeur ce qui forment le compteur 3bit, le bloc contrôleur de modulo, le bloc PWM, les blocs reset et flag Z.

Dans ce projet, nous avons appris à maîtriser le logiciel de conception Microwind ainsi que les méthodes d'élaboration de layout, en appliquant nos connaissances sur la physique des semi-conducteurs et la micro-électronique.

Références

- [1] Philippe Matherat. Une histoire de la microélectronique. Engineering school. Ecole nationale supérieure des télécommunications ; depuis 1998, 2007, pp.32.
- [2] Initiation aux circuits intégrés, à la technologie T.T.L. et aux familles M.O.S. par Jean-Michel ROLANDO, Ecole Normale, Bonneville.
- [3] CMOS Cookbook Second Edition. Don Lancaster Revised by Howard M.Berlin. pp 2-3 , p317.
- [4] CMOS Circuit Design, Layout, and Simulation. R.JACOB BARKER. IEEE Press. Chapitre 11 et 12.
- [5] Computer Systems Digital Design, Fundamentals of Computer Architecture and Assembly Language, Ata Elahi. Springer.
- [6] Neil H. E. Weste Macquarie University and The University of Adelaide David Money Harris Harvey Mudd College. CMOS VLSI Design A Circuits and Systems Perspective p 471, p462
- [7] www.ece353.engr.wisc.edu/timers/timers/
- [8] Roy Dong, Implementing Pulse-Width Modulation through MSP430 Timers
- [9] The Common Use of Pulse Width Modulation "PWM" Technique in Power Electronics Safaa Alaa Eldeen Hamza, Faculty of Engineering, Department of Electronic Engineering, MSc in Communication and Data Networks, Sudan-Khartoum
- [10] Design and Implementation of a Modulo Controller for Efficient Division Operations" Auteurs : Ahmed Z. Abid, Fatma Abdelkefi, Mohamed Abid Revue : IEEE Transactions on Computers Année : 2016
- [11] M. MORISSE MANO, Digital Logic and Computer design, p211
- [12] A. SAHA N. MANN, Digital Principals and Logic design, Infinity Science Press, p291
- [13] Ahmed Bensalem, Introduction aux timers des microprocesseurs : concepts et application, Journal de l'Informatique et de ses Applications (JIA), 2017
- [14] Thomas Dubois Sophie Lambert, Étude des performances des timers à microprocesseur pour les applications temps réel", Techniques et Sciences Informatiques, 2021
- [15] François ANCEAU Prof CNAM émérite, La logique, des MOS aux Circuits Intégrés