

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique



UNIVERSITE DE MSILA
FACULTE DE TECHNOLOGIE
DEPARTEMENT D'ELECTRONIQUE



MEMOIRE DE MASTER

DOMAINE : SCIENCES ET TECHNOLOGIE
FILIERE : GENIE ELECTRIQUE
OPTION : INSTRUMENTATION ET MAINTENANCE
INDUSTRIEL

Thème

**Conception et réalisation d'une carte de développement
FPGA**

Présenté par :
SALMI Chawki

Encadré par :
Mr. BENARROUDJ Marouane

N° d'ordre : 2012/ / 85/125/

Promotion : JUIN 2012

Dédicace

Après de longues années d'études et de travail, sachant l'importance de l'aide des êtres qui n'aiment, je voudrai humblement leurs, dédier ce modeste travail tout en avant qu'ils méritent le meilleur qui soit.

Je dédie ce travail :

à ma cher mère

qui a fait tant de sacrifice pour m'élever et m'instruire et qui ma encourage tout le long de mon parcours scolaire et académique.

à mon cher père

ma encourager et suer et a tant travail pour pouvoir m'instruire. tout en lui souhaitant bonne santé et longue vie.

à mes cher enseignants qui mont dirigé et aider et surtout soutenu.

à tout mes ami et camarade d'études

REMERCIEMENTS

Nous sommes reconnaissant tout premièrement à dieu tout puissant qui nous à guidé et fourni les moyens humains et matériaux pour achever notre parcours avec ce présent travail en lui demandant avec humilité de nous aider a faire d'autres progrès.

Nous remercions beaucoup MR, BENARROUDJ MAROUANE, notre encadreur, qui nous a guide, dirige et fourni la documentation et les ressources.

Nous remercions l'administration de notre institut d'électronique.



SOMMAIRE

Introduction générale..... 01

CHAPITRE I

Circuits programmables FPGA

Introduction	02
1- Définition Circuits programmables FPGA	03
2- Structure des FPGA's	04
2-1- Les CLBs (Configurable Logic Bloc)	04
2-2- Les IOBs (Input Output Bloc)	05
2-2-1- Configuration en entrée	06
2-2-2- Configuration en sortie	06
2-3- Les interconnexions	07
2-3-1- Les interconnexions à usage général	07
2-3-2- Les interconnexions directes	08
2-3-3- Les longues lignes	08
2-3-2-Performances des interconnexions	09
3- Classification des FPGAs	09
3-1- FPGAs à SRAM	09
3-2- FPGAs à Anti-fusible	10
3-3- EPROM/EEPROM	11
4- Approvisionnements de tension	12
5- Exemple de IOSTANDARD	12
6- Détection des pannes dans les circuits FPGA	13
7- Les avantages du FPGA	14
Conclusion	15

CHAPITRE II

SPARTAN 3E

Introduction	16
1- Quelques Familles des FPGAs de XILINX	17
2- Caractéristiques	17
3- Marquage de l'emballage	18
4- Architecture interne d'un circuit SPARTAN 3E	19
4-1- Blocs logiques configurables	20
4-2- Les IOBs (Input/ Output Bloc)	22
4-2-1- Normes de signal de SelectIO	24
4-3- Les multiplicateurs	25
4-3-1- Opération	26

4-4- Global Clock	28
4-4-1- Différences ressources de Global Clock	28
4-5- Le bloc RAM	29
5- Resistances Pull-Up et Pull-Down	30
6- Contrôle de taux (Slew Rate) et force du driver	31
7- IOBs a organisé en banques	31
8- Règles d'opérations bancaires d'entrée-sortie	32
9- Processus de configuration	33
10- Considérations pour les broches HSWAP, M [2:0], et VS [2:0]	33
11- Configuration Mode	37
11-1- Mode JTAG	37
11-2- SPI mode Serial flash	38
11-3- Mode maître série	39
11-4- BPI Mode en parallèle Flash	39
11-5- Mode esclave parallèle	41
11-6- MODE ESCLAVE SERIE	42
12- Alimentation	43
Conclusion:	44

CHAPITRE III

Périphériques

Introduction	45
Périphérique de la Carte :	45
1- Écran d'affichage LCD :	45
1-1 Compatibilité de tension	46
1-2- Contrôleur d'affichage à cristaux liquides	46
1-3- Données d'écriture à l'affichage	46
2- Port d'affichage VGA :	47
2-1- Signale et synchronisation pour un affichage 640x480 VGA 60 hertz	48
3- Portes série RS-232 :	51
4- Connecteur PS/2 :	52
4-1- Clavier :	53
4-2- Souris :	54
4-3- Compatibilité de tension	55
5- Mémoire Flash Parallèle	56
5-1- Caractéristiques	57
6- Mémoire Flash série SPI :	58
6-1- Caractéristiques	59
7- DDR SDRAM	60
7-1- Caractéristiques	60
8- LED Switches Interface :	61
9- M2 LECTEUR CARTE MEMOIRE :	62
Conclusion	63

CHAPITRE IV

CONCEPTION ET REALISATION

Introduction	64
1- Le besoin d'évolution	64
2- Le processus de conception d'un circuit	65
2-1- Etude de marché	65
2-2- Evaluation du concept	65
2-3- Spécifications et Architecture	66
2-4- Qualification	66
2-5- Production	66
2-6- L'activité de validation	66
3- PCB :	67
4- Méthode de Câblage	69
5- Règles D'alimentations :	70
6- Réalisation :	72
6-1- Programmeur JTAG :	72
7- Bibliothèque FPGA	78
8- Contraint File	78
9- Programmation des FPGA's	79
9-1- Technique de programmation des FPGA's	79
10- Program de test	82
Conclusion	84
Conclusion générale	85

Liste Des Figures Et Des Tableaux

BIBLIOGRAPHIE

Introduction Générale

Il y a de cela quelques années, la réalisation des montages électroniques impliquait l'utilisation d'un nombre important de circuits intégrés logiques, ce qui avait pour conséquence une mise en œuvre complexe, un prix de revient important et un circuit imprimé de taille.

Pour diminuer et minimiser ces circonstances, ces dernières années ont vu l'apparition d'un nouveau type de circuit intégré à savoir, les circuits programmables FPGA qu'ils sont introduits sur le marché en 1985 et devenus des composants incontournables. Aujourd'hui, ces puces sont devenues les concurrents principaux des ASIC et des DSP en combinant certains avantages des deux et en bénéficiant.

En conséquence Cette année On a fait le choix de travailler avec des FPGA. Actuellement, la solution existante est le FPGA spartan 3E, de la série spartan de XILINX. Cette série propose des FPGA à un prix raisonnable, offrant un bon compromis entre une bonne performance et un niveau d'intégration moyen

La première partie consiste à étudier d'une manière générale des circuits FPGA pour mieux exploiter notre sujet. Nous présentons le circuit FPGA SPARTAN 3E dans une seconde partie. La troisième partie est consacrée à l'étude des périphériques implantés sur la carte. Dans la dernière partie, nous présentons la conception de la carte de test et le programmeur JTAG avec le logiciel de conception électronique Altium Designer. Nous présentons également dans cette partie, la librairie qui facilite l'utilisation et l'interfaçage de notre carte de développement dans Altium Designer.

CHAPITRE I

Circuits programmables

FPGA

Introduction :

Il y a de cela quelques années, la réalisation des montages électroniques impliquait l'utilisation d'un nombre important de circuits intégrés logiques, ce qui avait pour conséquence une mise en œuvre complexe, un prix de revient important et un circuit imprimé de taille.

Pour diminuer et minimiser ces circonstances, ces dernières années ont vu l'apparition d'un nouveau type de circuit intégré à savoir, les circuits programmables FPGA qu'ils sont introduits sur le marché en 1985 et devenus des composants incontournables, entre autres sur les cartes de traitement de signal et d'images. Aujourd'hui, ces puces sont devenues les concurrents principaux des ASIC et des DSP en combinant certains avantages des deux et en bénéficiant, comme ses rivales, de l'évolution de la densité d'intégration qui double tous les 18 mois selon la fameuse loi de Moore. Évoluer à ce rythme entraîne toutefois son lot de défis. Ces défis touchent notamment l'étape de test suivant la fabrication de ces circuits de haute intégration [1].

Les progrès technologiques continus dans le domaine des circuits intégrés ont permis la réduction des coûts, de la consommation, et c'est maintenant un lieu commun d'affirmer que les circuits intégrés spécifiques d'une application ont permis une réduction de la taille des systèmes numériques ainsi que la réalisation de circuits de plus en plus complexes, tout en améliorant leurs performances et leur fiabilité.

Aujourd'hui les techniques de traitement numérique occupent une place majeure dans tous les systèmes électroniques modernes grand public, professionnels ou de défense. De plus, les techniques de réalisation de circuits spécifiques, tant dans les aspects matériels (composants reprogrammables, circuits pré caractérisés et bibliothèques de macro fonctions) que dans les aspects logiciels (placement routage, synthèse logique) font désormais de la microélectronique une des bases indispensables pour la réalisation de systèmes numériques performants. Elle impose néanmoins une méthodologie de développement en CAO (Conception Assistée par Ordinateur) très structurée.

Le progrès de ces technologies permet de faire des composants toujours plus rapides et à plus haute intégration, ce qui permet de programmer des applications importantes.

1- Définition Circuits programmables FPGA :

Les FPGAs (Field programmable Gate Array) ce sont des circuits intégrés prédéfinis programmables par l'utilisateur [1] [2]. Ils constituent une évolution des circuits logiques programmables CPLDs. Ils se composent d'une matrice de blocs logiques programmables entourée de blocs d'entrées/sorties, et l'ensemble est relié par un réseau d'interconnexion programmable [2]. on distingue plusieurs types de lignes définies par leur longueur relative. On trouve en effet [2] :

-Les interconnexions à usage général qui sont composées de segments verticaux et horizontaux qui entourent chaque CLB et qui peuvent être reliés entre eux par une matrice de commutation.

-Les lignes directes fournissant des chemins entre les CLB adjacents et entre les CLB et les cellules d'entrée- sortie.

- Les lignes longues qui sont des lignes verticales et horizontales qui n'utilisent pas de matrice de commutation. Elles parcourent toutes les zones d'interconnexion. Elles sont utilisées pour véhiculer les signaux qui doivent parcourir de long trajet. Ces lignes conviennent pour véhiculer les signaux d'horloge.

Trois technologies différentes sont utilisées pour programmer les différentes ressources d'un FPGA : les antifusibles, la mémoire flash et la mémoire SRAM. Dans ce projet, nous cibons les FPGA utilisant la troisième technologie, en particulier le Spartan 3E de la compagnie Xilinx. Ainsi, les interrupteurs programmables dirigeant la propagation des signaux sur les interconnexions sont réalisés par des transistors MOS dont l'état est contrôlé par des cellules de mémoire SRAM. De la même manière, la définition des fonctions contenues dans les CLB est mémorisée dans ce genre de cellules. Par conséquent, toute la Configuration d'un FPGA basé sur cette technologie est contenue dans des cellules SRAM [1].

2- Structure des FPGA's

Structurés sous forme de matrices, les FPGA sont composés d'éléments logiques de base, Constitués de portes logiques, présentes physiquement sur le circuit. Ces portes sont reliées par un ensemble d'interconnexions modifiables : d'où l'aspect programmable du circuit.

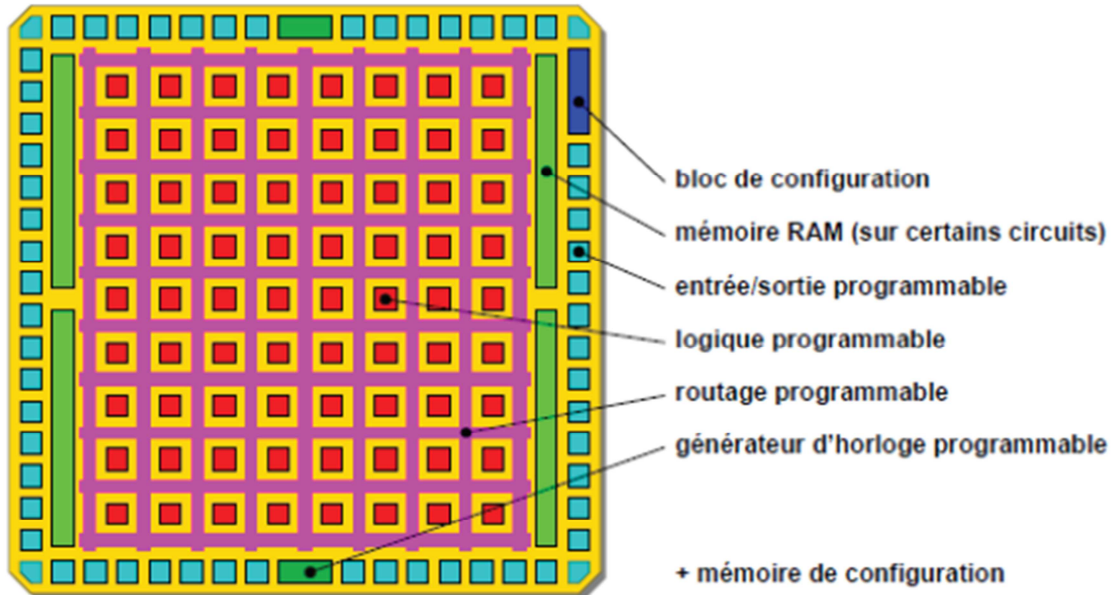


Figure I.1: Structure d'un circuit FPGA [3].

FPGA est structuré de :

2-1- Les CLBs (Configurable Logic Bloc):

Les blocs logiques ou éléments logiques configurables sont les éléments déterminants des performances du FPGA et contenant les fonctions logiques combinatoires et séquentielles.

- La partie combinatoire permet de réaliser des fonctions de complexité moyenne avec des portes classiques ET, OU et NON de deux à une dizaine d'entrées.

- La partie séquentielle comporte une ou deux bascules généralement de type D.

Compte tenu du nombre d'éléments logiques et de leur structure, leur association permet de réaliser tous les types de bascule. L'intérêt est de créer des mémoires élémentaires à un bit. Suivant le fabricant du circuit, ces blocs contiennent un nombre différent de portes logiques et de bascules à l'intérieur d'un bloc comme le montre la Figure 2.

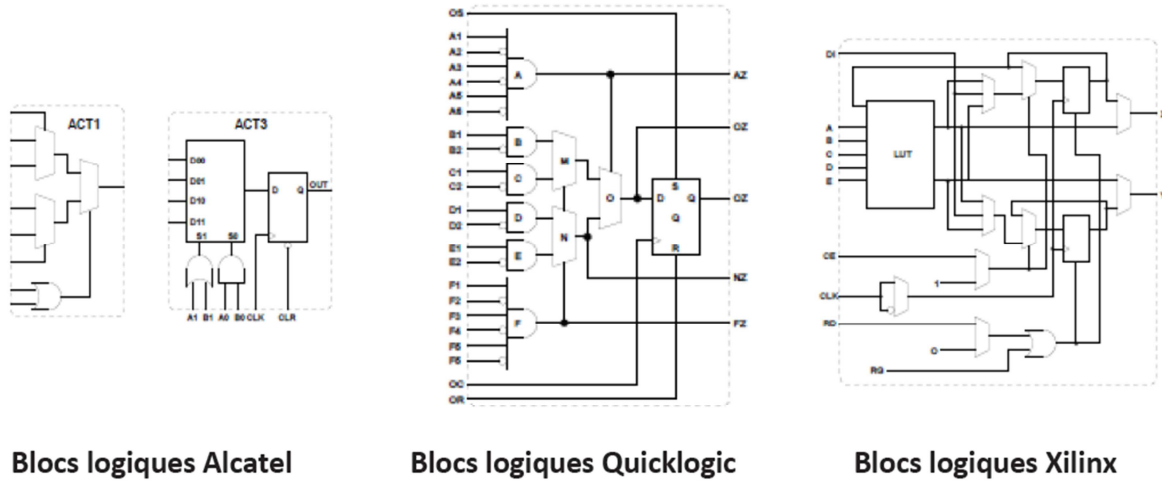


Figure I.2: Exemple de blocs logiques de différents fabricants [3].

2-2- Les IOBs (Input Output Bloc):

La figure I.3 présente la structure de ce bloc. Ces blocs entrée/sortie permettent l'interface entre les broches du composant FPGA et la logique interne développée à l'intérieur du composant. Ils sont présents sur toute la périphérie du circuit FPGA. Chaque bloc IOB contrôle une broche du composant et il peut être défini en entrée, en sortie, en signaux bidirectionnels ou être inutilisé (haute impédance).

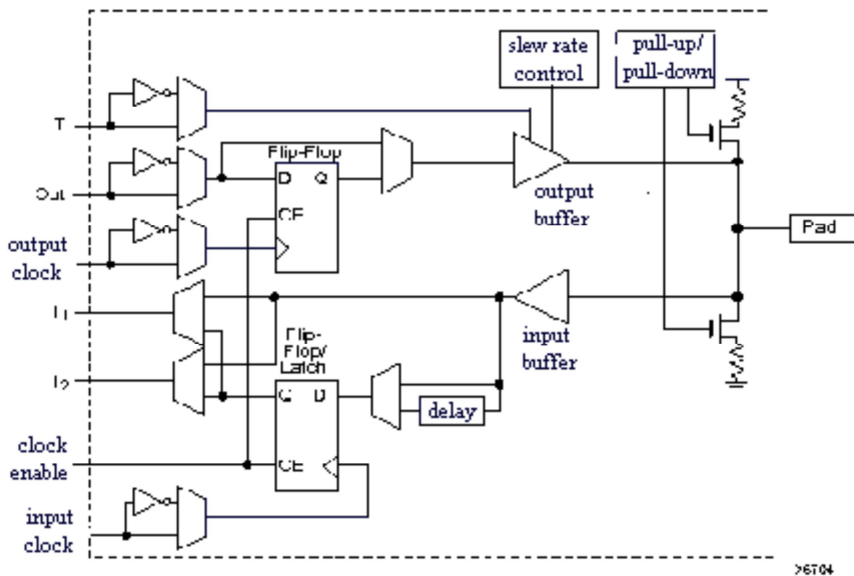


Figure I.3: Input Output Block (IOB) [4].

2-2-1- Configuration en entrée:

Premièrement, le signal d'entrée traverse un buffer qui selon sa programmation peut détecter soit des seuils TTL ou soit des seuils CMOS. Il peut être routé directement sur une entrée directe de la logique du circuit FPGA ou sur une entrée synchronisée. Cette synchronisation est réalisée à l'aide d'une bascule de type D, le changement d'état peut se faire sur un front montant ou descendant. De plus, cette entrée peut être retardée de quelques nanosecondes pour compenser le retard pris par le signal d'horloge lors de son passage par l'amplificateur. Le choix de la configuration de l'entrée s'effectue grâce à un multiplexeur (program controlled multiplexer). Un bit positionné dans une case mémoire commande ce dernier [4].

2-2-2- Configuration en sortie:

Nous distinguons les possibilités suivantes :

- inversion ou non du signal avant son application à l'IOB,
- synchronisation du signal sur des fronts montants ou descendants d'horloge,
- mise en place d'un " pull-up " ou " pull-down " dans le but de limiter la consommation des entrées sorties inutilisées,
- signaux en logique trois états ou deux états. Le contrôle de mise en haute impédance et la réalisation des lignes bidirectionnelles sont commandés par le signal de commande Out Enable lequel peut être inversé ou non. Chaque sortie peut délivrer un courant de 12mA. Ainsi toutes ces possibilités permettent au concepteur de connecter au mieux une architecture avec les périphériques extérieurs.

2-3- Les interconnexions:

Les connexions internes dans les circuits FPGA sont composées de segments métallisés. Parallèlement à ces lignes, nous trouvons des matrices programmables réparties sur la totalité du circuit, horizontalement et verticalement entre les divers CLB. Elles permettent les connexions entre les diverses lignes, celles-ci sont assurées par des transistors MOS dont l'état est contrôlé par des cellules de mémoire vive ou RAM. Le rôle de ces interconnexions est de relier avec un maximum d'efficacité les blocs logiques et les entrées/sorties afin que le taux d'utilisation dans un circuit donné soit le plus élevé possible.

Pour parvenir à cet objectif, Xilinx propose trois sortes d'interconnexions selon la longueur et la destination des liaisons. Nous disposons :

D'interconnexions à usage général, D'interconnexions directes,

De longues lignes.

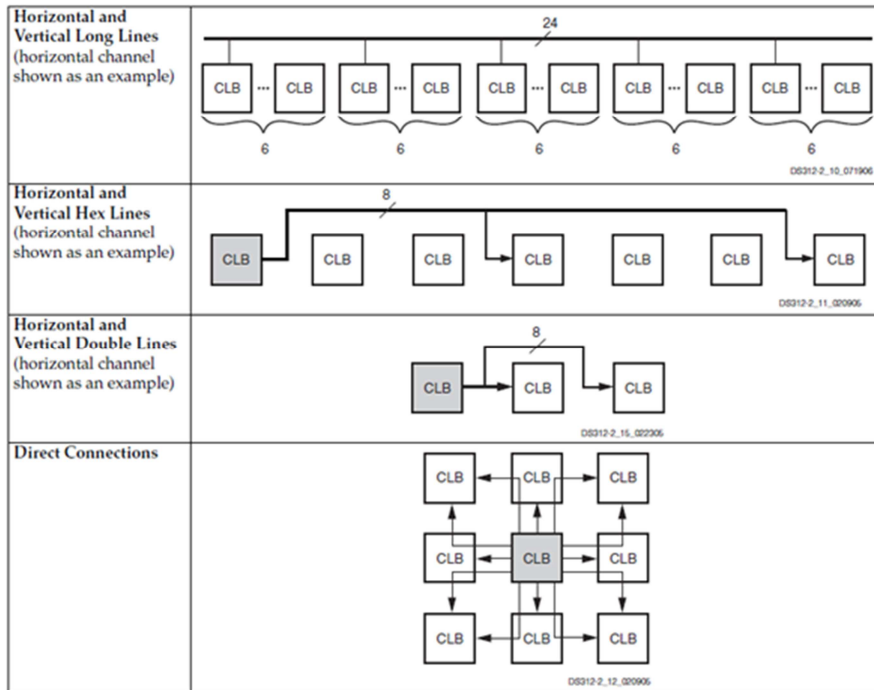


Figure I.4: Types d'interconnexion entre deux CLB d'interconnexion adjacente [4].

2-3-1- Les interconnexions à usage général [4] :

Ce système fonctionne en une grille de cinq segments métalliques verticaux et quatre segments horizontaux positionnés entre les rangées et les colonnes de CLB et de l'IOB.

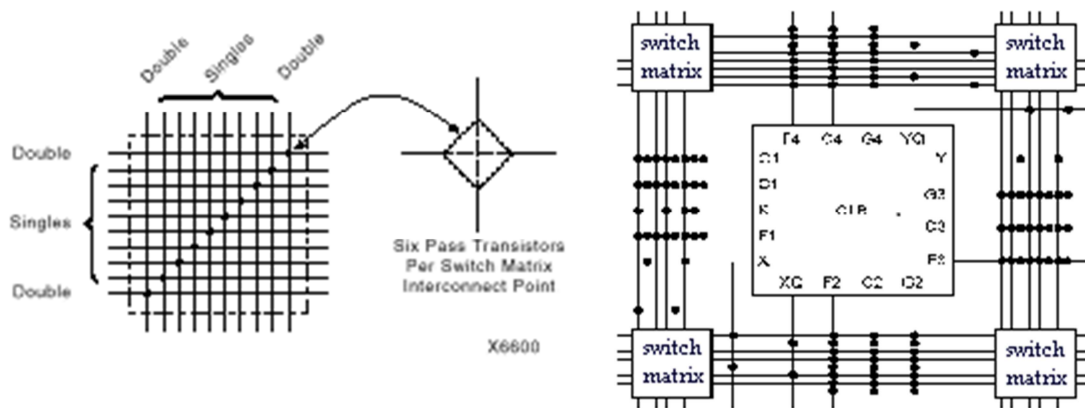


Figure I.5: Connexions à usage général et détail d'une matrice de commutation [4].

Des aiguilleurs appelés aussi matrices de commutation sont situés à chaque intersection. Leur rôle est de raccorder les segments entre eux selon diverses configurations, ils assurent ainsi la communication des signaux d'une voie sur l'autre. Ces interconnexions sont utilisées pour relier un CLB à n'importe quel autre. Pour éviter

que les signaux traversant les grandes lignes ne soient affaiblis, nous trouvons généralement des buffers implantés en haut et à droite de chaque matrice de commutation.

2-3-2- Les interconnexions directes :

Ces interconnexions permettent l'établissement de liaisons entre les CLB et les IOB avec un maximum d'efficacité en termes de vitesse et d'occupation du circuit. De plus, il est possible de connecter directement certaines entrées d'un CLB aux sorties d'un autre.

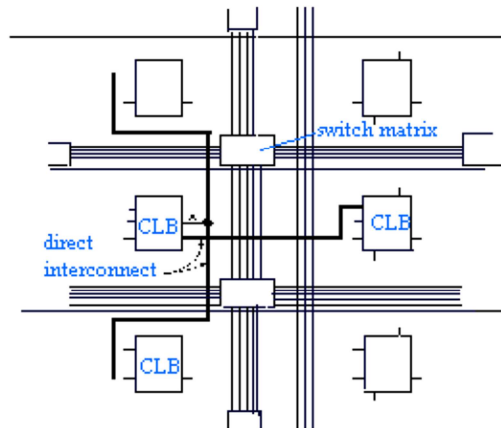


Figure I.6: Les interconnexions directes [4].

Pour chaque bloc logique configurable, la sortie X peut être connectée directement aux entrées C ou D du CLB situé au-dessus et les entrées A ou B du CLB situé au-dessous. Quant à la sortie Y, elle peut être connectée à l'entrée B du CLB placé immédiatement à sa droite. Pour chaque bloc logique adjacent à un bloc entrée/sortie, les connexions sont possibles avec les entrées I ou les sorties O suivant leur position sur le circuit.

2-3-3- Les longues lignes :

Les longues lignes sont de longs segments métallisés parcourant toute la longueur et la largeur du composant, elles permettent éventuellement de transmettre avec un minimum de retard les signaux entre les différents éléments dans le but d'assurer un synchronisme aussi parfait que possible. De plus, ces longues lignes permettent d'éviter la multiplicité des points d'interconnexion.

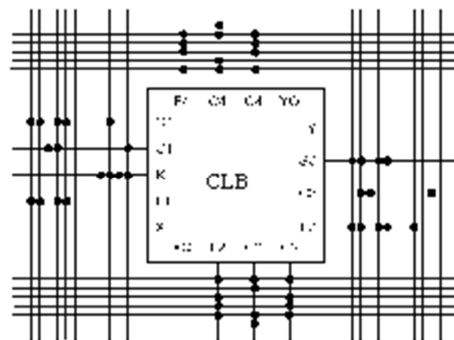


Figure I.7: Les longues lignes [4].

2-3-2-Performances des interconnexions:

Les performances des interconnexions dépendent du type de connexions utilisées. Pour les interconnexions à usage général, les délais générés dépendent du nombre de segments et de la quantité d'aiguilleurs employés. Le délai de propagation de signaux utilisant les connexions directes est minimum pour une connectique de bloc à bloc. Quant aux segments utilisés pour les longues lignes, ils possèdent une faible résistance mais une capacité importante. De plus, si on utilise un aiguilleur, sa résistance s'ajoute à celle existante.

3- Classification des FPGAs

On distingue trois classes de FPGAs selon la technique de réalisation des PIP (points de connexion) :

3-1- FPGAs à SRAM

La structure de base d'un FPGA de type SRAM est très complexe. Le point de connexion entre les différentes cellules est un ensemble de transistors CMOS de commutation commandés par des cellules de mémoire vive (RAM) [5].

Le FPGA à SRAM peut être considéré comme étant constitué de deux couches superposées :

Une couche comportant toutes les ressources du FPGA (blocs d'entrées-sorties, CLB, réseaux de connexion).

Une couche comportant la commande de chaque point de connexion. Le point de commande est une cellule de type mémoire vive (SRAM) et qui peut être assimilée à une bascule D. L'ensemble des points de commande est un grand registre à décalage.

Afin d'établir les connexions souhaitées et obtenir le système programmé, il suffit de charger le registre à décalage.

Ce type de circuit logique programmable possède l'avantage d'être souple dans son utilisation du fait que le point de commande est une mémoire vive, ainsi qu'un effacement beaucoup plus rapide en comparaison avec les CPLDs à EPROM ou EEROM [6].

Cependant, il possède un inconvénient majeur qui réside dans le chargement de la configuration (charger la mémoire vive) à la mise sous tension. En effet, chaque absence d'alimentation se traduit par la perte du contenu du registre à décalage. Ce

problème a été résolu par la dotation de chaque dispositif d'une logique d'auto-chargement après une mise sous tension. Tout cela est réalisé par l'introduction d'une mémoire morte de type EEPROM généralement à accès série.

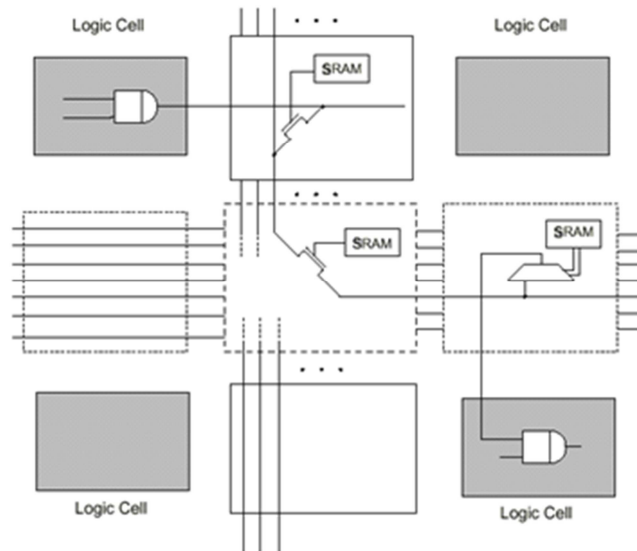


Figure I.8: Technologie SRAM

3-2- FPGAs à Anti-fusible

Ces circuits, contrairement aux FPGA à SRAM, sont plus récents étant donné que leur première commercialisation effective remonte seulement à l'année 1990. L'architecture globale des FPGA à anti-fusible est analogue à celle à SRAM avec un certain nombre de blocs d'entrées/sorties répartis tout autour de la puce, des blocs logiques placés au centre de celle-ci et des lignes d'interconnexion, mais la similitude s'arrête là [6].

En effet, ce type de circuit contrairement aux FPGA à SRAM utilise des cellules très simples et ces dernières sont reliées aux lignes d'interconnexion par ce qu'on appelle des anti-fusibles. Ces moyens de connexion, tant qu'ils ne sont pas programmés, isolent grâce à une fine couche d'oxyde deux segments conducteurs placés sur deux niveaux différents. Afin de réaliser la connexion entre ces deux segments, il suffit d'appliquer une impulsion de haute tension (21Volts) calibrée en temps (moins de 5ms), ce qui engendre un trou dans la couche d'oxyde et les deux lignes conductrices se retrouvent ainsi relié [5].

Ce type de processus, utilisé avec perforation de l'oxyde, implique l'impossibilité de revenir en arrière ; c'est-à-dire que la liaison une fois établie ne peut plus être chargée, elle est irréversible.

Un FPGA à anti-fusible n'est donc programmable qu'une seule fois. Cependant, cette irréversibilité des moyens de connexion (anti-fusible) est compensée par leur surface réduite (ils sont minuscules). Ils peuvent donc être très nombreux sur la puce, ce qui accroît la souplesse d'interconnexion interne et facilite le routage.

Un état anti-fuse réside en un état d'haute impédance. Il peut être programmé dans un état de faible impédance ou état "fused". Il s'agit d'une technologie moins chère que la SRAM, elle permet d'atteindre des vitesses plus élevées et occupe moins de place sur le circuit. Par contre, un tel FPGA ne peut être programmé qu'une seule fois. Performances électriques supérieures à la technologie SRAM (minimisation des effets RC due au faible surface). Résistance de l'ordre d'une centaine d'ohm et effet capacitif de quelques femto farads.

Quelques exemples de FPGA utilisant la technologie anti-fuse:

- Actel série EX
- Actel série SX-A

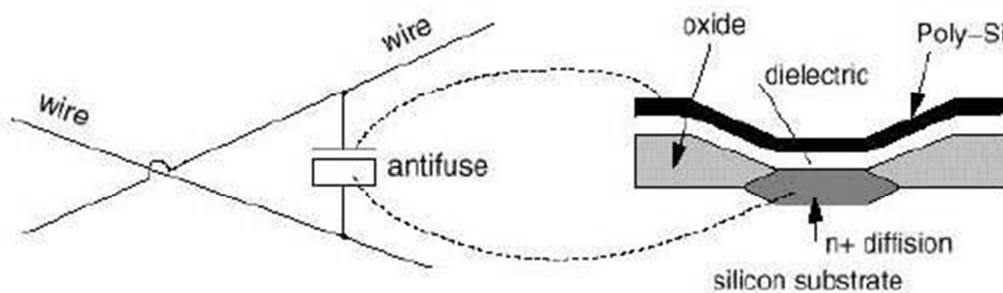


Figure I.9: Technologie anti-Fuse [4].

3-3- EPROM/EEPROM:

Cette méthode est la même que celle utilisée dans les mémoires EPROM. L'EEPROM est reprogrammable. La puce fonctionne seule. La surface moyenne et les caractéristiques électriques sont semblables à la SRAM.

Exemples de datasheet de FPGA utilisant la technologie EEPROM: Actel PROASIC.

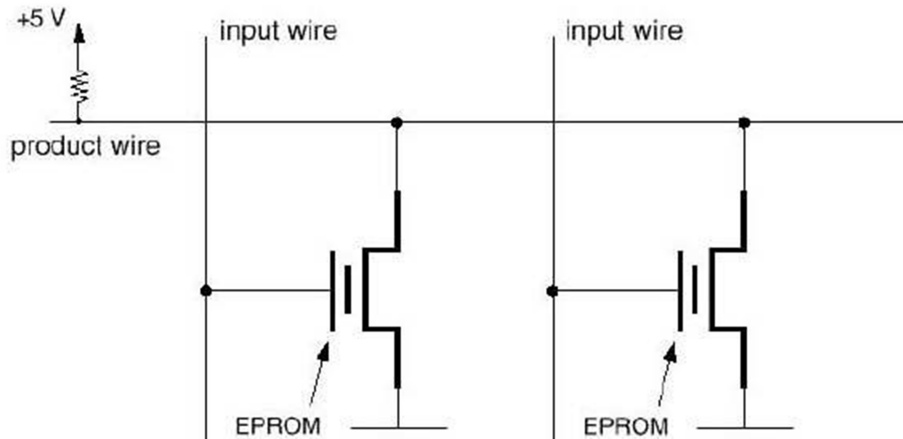


Figure I.10: Technologie EPROM/EEPROM [4].

Résumé :

Nom:	Reprogrammable:	Volatile:	Technologie:
Fuse	Non	Non	Bipolaire
EPROM	Oui, en dehors du circuit	Non	UVC MOS
EEPROM	Oui, dans le circuit	Non	EECMOS
SRAM	Oui, dans le circuit	Oui	CMOS
Antifuse	Non	Non	CMOS

Tableau I.1: Résumé des différentes technologies [4].

4- Approvisionnements de tension

Les FPGAs Spartan-3E ont multiples approvisionnement de tension, suivant les indications du tableau.

Il y a deux entrées d'approvisionnement pour des fonctions de logique interne, VCCAUX et VCCINT.

Chacun des quatre banques d'entrée-sortie à un VCCO qui actionne les driver I/O de la banque d'Associé. Toutes les connexions de VCCO à un spécifique I/O Bank doivent être reliées et doivent se relier à la même tension

Altera	Stratix II	Stratix	Cyclone II	Cyclone
V _{CCINT} (Core)	1.2V ± 5% @ 1A to 6A	1.5V ± 5% @ 1A to 10A	1.2V ± 5% @ 500mA to 5A	1.5V ± 5% @ 500mA to 5A
V _{CCO} (Vo)	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 500mA to 6A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 500mA to 6A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 500mA to 6A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 500mA to 6A
V _{CCPD} (Aux)	3.3V ± 5% @ 300mA	-	-	-

Xilinx	Virtex-5	Virtex-4FX, SX, LX	Virtex-II Pro	Virtex-II	Virtex-E	Virtex	Spartan-3, -3E, -3L	Spartan-IIE	Spartan-II
V _{CCINT} (Core)	1V ± 5% @ 200mA to 5A	1.2V ± 5% @ 200mA to 5A	1.5V ± 5% @ 200mA to 12A	1.5V ± 5% @ 200mA to 12A	1.8V ± 5% @ 200mA to 7A	2.5V ± 5% @ 200mA to 7A	1.2V ± 5% @ 200mA to 5A	1.8V ± 5% @ 200mA to 3A	2.5V ± 5% @ 200mA to 2A
V _{CCO} (Vo)	3.3V, 2.5V, 1.8V, 1.5V and/or 1.2V ± 5% @ 50mA to 4A	3.3V, 2.5V, 1.8V, 1.5V and/or 1.2V ± 5% @ 50mA to 4A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 50mA to 5A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 50mA to 5A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 500mA to 5A	3.3V, 2.5V and/or 1.5V ± 5% @ 50mA to 5A	3.3V, 3.0V, 2.5V, 1.8V, 1.5V and/or 1.2V ± 5% @ 50mA to 4A	3.3V, 2.5V, 1.8V and/or 1.5V ± 5% @ 50mA to 750mA	3.3V, 2.5V and/or 1.5V @ 50mA to 500mA
V _{CCAUX} (Aux)	2.5V ± 5% @ 300mA	2.5V ± 5% @ 300mA	2.5V ± 5% @ 300mA	3.3V ± 5% @ 300mA	-	-	2.5V ± 5% @ 300mA	-	-

Tableau I.2: les besoins d'alimentation pour chaque type d'FPGA [7].

5- Exemple d'IOSTANDARD :

La figure ci-dessus fournit une comparaison entre l'entrée IOSTANDARD CMOS basse tension, et les familles logiques TTL.

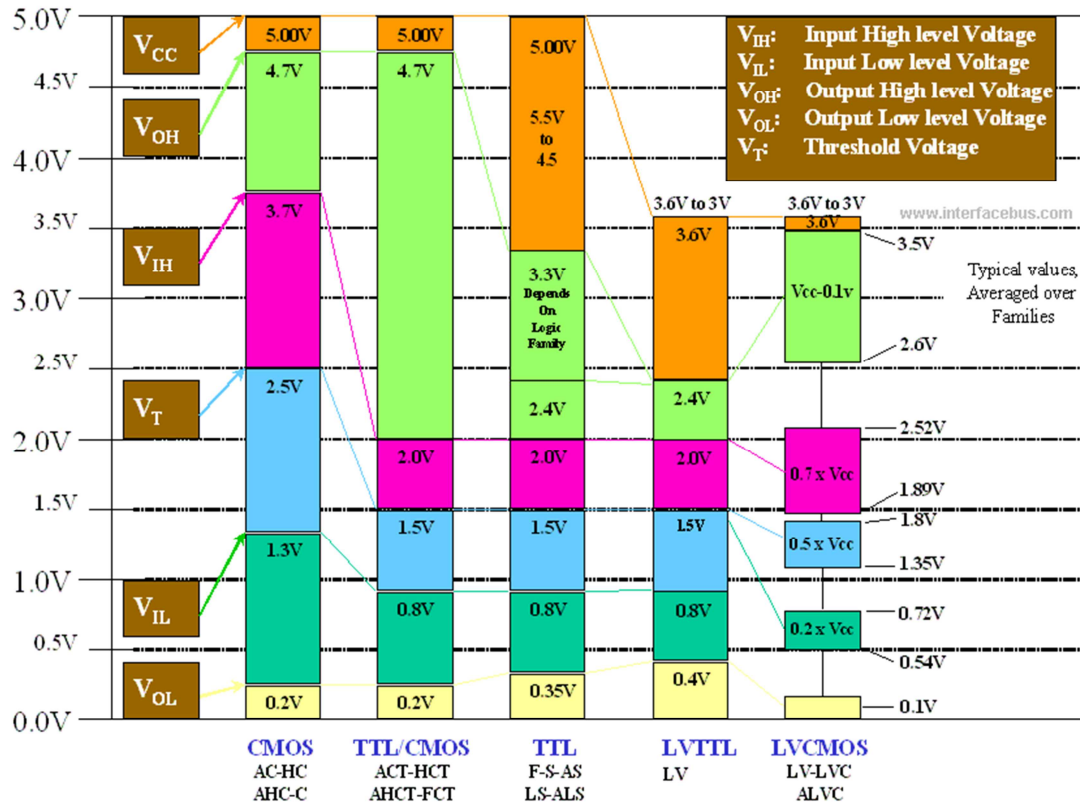


Figure I.11: Comparaison de l'entrée et de sortie I/O CMOS et LVTTL [8].

6- Détection des pannes dans les circuits FPGA [1] :

Depuis l'introduction des circuits FPGA sur le marché, des chercheurs se sont intéressés à la détection des pannes qui peuvent affecter ces circuits. De ce fait, on trouve dans la littérature plusieurs articles qui abordent ce sujet. Les approches proposées peuvent être classées en trois catégories.

La première catégorie utilise certains aspects du test fonctionnel des ASIC adaptés aux FPGA, catégorie souvent appelée test dépendant de l'application. La deuxième approche exploite la programmable de ces circuits pour tester les ressources.

Cette approche est connue comme le test indépendant de l'application.

La troisième méthode consiste à l'application des tests IDDQ sur la puce qui peut être appliqué également dépendamment et indépendamment des applications.

7- Les avantages du FPGA [3] :

Ces notions sur le FPGA donnent des indications quant à l'intérêt de son utilisation dans le cadre du projet. Voici les avantages clés qui ont fait qu'on a s'est tourné vers cette solution.

- Un circuit reprogrammable : L'avantage du FPGA est de pouvoir être reprogrammable contrairement aux circuits intégrés de type ASIC. Ce qui rend cette solution modulable et donne la possibilité de modifier le programme générique de base afin de le rendre spécifique au circuit utilisé. Une solution de validation utilisant le FPGA peut alors convenir à beaucoup de projets et donc diffusée à plusieurs équipes.
- Un investissement rentable dans la durée : Cela est dû à sa reprogrammation, ce qui implique une réutilisation à destination d'autres projets, malgré un prix à l'achat supérieur à un circuit ASIC.
- Une Reprogrammation quasi-instantanée du circuit. Une fois le programme validé cela ne prend que quelques minutes à l'implémenter. A titre de comparaison, la fabrication d'un circuit ASIC peut prendre plusieurs semaines.

Cependant, le FPGA n'est pas le seul composant reprogrammable du marché. Le DSP « Digital Signal Processor », processeur de signal numérique, permet également d'émuler un montage numérique.

Le DSP est programmable grâce au langage C, le FPGA utilise quant à lui le VHDL. Le format de description machine est généré automatiquement par les logiciels de développement des concepteurs.

Il est possible alors de vérifier les circuits sans avoir à les concevoir par nous-mêmes. L'importation de leur fichier dans notre programme est suffisante.

Par ailleurs, le FPGA peut disposer d'un DSP sous forme d'IP incluse dans le système du circuit.

Conclusion :

Dans ce chapitre, nous avons présenté l'architecture des circuits configurables de type FPGA et dont nous précisons ici les principaux avantages :

- Le premier argument est la souplesse de programmation qui permet l'emploi conjoint d'outils de schématisation aussi bien que l'exploitation d'un langage de haut niveau tel VHDL.

Ce qui permet de multiplier les essais, d'optimiser de diverses manières l'architecture développée, de vérifier à divers niveaux de simulation la fonctionnalité de cette architecture.

- Le second argument est évidemment la nouvelle possibilité de reconfiguration dynamique partielle ou totale d'un circuit ce qui permet d'une part, une meilleure exploitation du composant, une réduction de surface de silicium employé et donc du coût, et d'autre part, une évolutivité assurant la possibilité de couvrir à terme des besoins nouveaux sans nécessairement repenser l'architecture dans sa totalité. L'un des points forts de la reconfiguration dynamique est effectivement de permettre de reconfigurer en temps réel en quelques microsecondes tout ou partie du circuit, c'est à dire de permettre de modifier la fonctionnalité d'un circuit en temps quasi réel.

Ainsi le même CLB pourra à un instant donné être intégré dans un processus de filtrage numérique d'un signal et l'instant d'après être utilisé pour gérer une alarme. On dispose donc quasiment de la souplesse d'un système informatique qui peut exploiter successivement des programmes différents, mais avec la différence fondamentale qu'ici il ne s'agit pas de logiciel mais de configuration matérielle, ce qui est infiniment plus puissant.

- Notons enfin que ces circuits n'ont pas vocation à concurrencer les super calculateurs, mais plutôt à offrir une alternative en fonction de critères comme l'encombrement, les performances et le prix, et sont de ce fait bien adaptés à des applications de qualité dans le domaine des systèmes ambulatoires ou nomades.

- Enfin il semble que de plus en plus fréquemment les concepteurs de circuits ASIC préfèrent passer par l'étape intermédiaire d'un FPGA ce qui est moins risqué économiquement, puis une fois que le modèle FPGA est au point, il est alors relativement aisé de le retranscrire dans une architecture de type pré diffusé ou pré caractérisés. Ce que tous les fondeurs de silicium savent effectivement faire pour en faire un circuit réellement personnalisé et confidentiel.

Le FPGA n'étant évidemment pas un circuit très sécurisé sur le plan de la confidentialité puisqu'il suffit d'analyser le contenu de la ROM associée pour remonter à la schématique imaginée.

Chaque FPGA à des caractéristique spécifié dépanadant au fabriquant et au série. Pour la fabrication du Carte FPGA il faut suivre des normes bien spécifié par le Constricteur Alors pour notre cas il indispensable de faire une étude sur notre circuit FPGA Du XILINX série Spartan 3E XC3S500E .

CHAPITRE III

SPARTAN 3E

Introduction

La famille SPARTAN 3E a été spécifiquement conçue pour répondre aux besoins de volume élevé et pour les applications grand public sensibles aux coûts électroniques [9].

Cette famille offre une densité allant de 100.000 à 1.6 millions de portes logiques programmables définissant cinq sous familles, Ce type de FPGA s'appuie sur le succès de la famille antérieure SPARTAN 3 et elle est différenciée par rapport à cette dernière par une augmentation du nombre de IOBs, réduisant ainsi considérablement le coût des cellules [9].

Ces nouvelles fonctionnalités améliorent les fonctionnalités du système et réduisent considérablement le coût de la configuration.

Les nouvelles caractéristiques améliorent des performances système et réduisent le coût de configuration. Ces améliorations de Spartan-3E FPGA, avec la technologie transmatrice avancée de 90 nanomètre, fournissent plus de fonctionnalité et de largeur de bande par dollar que n'était pas avant possible, fixant de nouvelles normes dans l'industrie de logique programmable.

En raison de leur exceptionnellement coût bas, Spartan-3E FPGAs sont idéalement adaptés à un large éventail d'applications d'électronique grand public, y compris l'accès à bande large, les réseaux des maisons, l'affichage/projection, et le matériel de télévision numérique.

La famille de Spartan-3E est une alternative supérieure au masque programmé ASICs. FPGAs évitent le coût initial élevé, les cycles de développement prolongés, et l'inflexibilité inhérente d'ASICs conventionnel. En outre, les liaisons de programmable de FPGA conçoivent des mises à jour dans le domaine sans le remplacement de matériel nécessaire, une impossibilité avec ASICs.

1- Quelques Familles des FPGAs de XILINX :

La firme XILINX est le plus grand fournisseur des circuits logiques programmables. Dès le début des années 80, elle a mis à la disposition très grande gamme de FPGA en commençant par XC 2000, suivie par XC 3000, XC 4000, XC 5000, XC 6000, SPARTAN et arrivant enfin à VIRTEX, septième génération de FPGA (1999) (voir tableau ci-dessous) [9].

Famille	Porte	E/S	Bascules	RAM bits	CLB
XC 2000	0.6 - 1.5K	58 - 74	122 - 174	0	64-100
XC3000	1.3 - 9K	64 - 176	256 - 1320	0	64-484
XC4000	2 - 180K	80 - 448	360 - 7168	3K - 100K	100-3136
XC6200	9K - 100K	192 - 512	2K - 16K	0	2K - 16K
SPARTAN 3	50K - 5000K	124 - 633	-	12K - 520K	192 - 8,32k
SPARTAN 3E	100K - 1600K	108 - 376	-	15K - 231K	240 - 3688
VIRTEX	58K - 1M	180 - 512	1728 - 27K	32K - 131K	384-6144
VIRTEX II	40K - 10M	88 - 1108	256 - 61K	72K - 3.4M	64-15360

Tableau II.1 : Tableau récapitulative des différentes familles de XILINX [9].

Dans notre travail, on s'intéressera uniquement au FPGA de type SPARTAN 3E.

2- Caractéristiques Du SPARTAN 3E[9]:

- coût Très bas, solution performante pour des applications à fort débit
- Technologie transformatrice prouvée du 90 nanomètre avancé
- Multi-tension, 3.3V, 2.5V, 1.8V, 1.5V, et 1.2V Broches multistandard d'interface de Select IO Normes asymétriques de signal de LVCMOS, LVTTTL, HSTL, et SSTL
- Jusqu'à 376 bornes d'entrée-sortie ou 156 paires de signal différentiel
- Vitesse de transfert des données de 622+ Mb/s par entrée-sortie
- début avec RDA SDRAM jusqu'à 333 Mb/s
- 18 x 18 multiplicateurs augmentés avec la canalisation facultative

- IEEE 1149.1/1532 JTAG port programmant/corrigent
- Jusqu'à 648 Kbits du bloc rapide RAM
- Jusqu'à 231 Kbits de RAM distribué efficace
- Synthèse de fréquence, multiplication, division
- Haute résolution déphaseuse de plage de fréquence de la taille (5 mégahertz à plus de 300 mégahertz)
- Huit horloges globales plus huit horloges supplémentaires par chaque moitié de dispositif .
- Accomplissez le logiciel de Xilinx ISE® et de WebPACK™
- MicroBlaze™ et noyaux de processeur inclus par PicoBlaze™
- 32-/64-bit entièrement conforme 33 mégahertz de PCI supporté (66 Mégahertz dans des quelques dispositifs).

3- Marquage de l'emballage [9]:

En compte 3 types de Package

Boîtiers BGA, CP132 et CPG132, QFP.

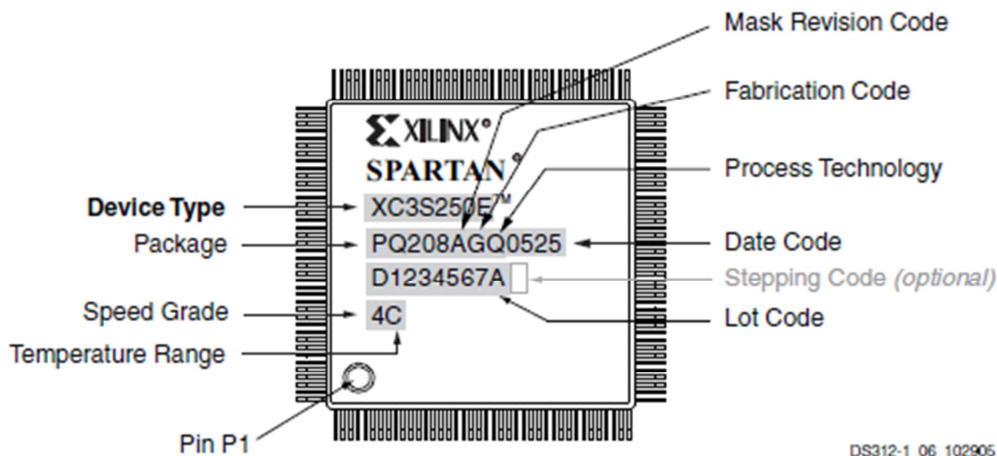


Figure II. 1: Boîtier QFP Spartan-3E Marquage Exemple

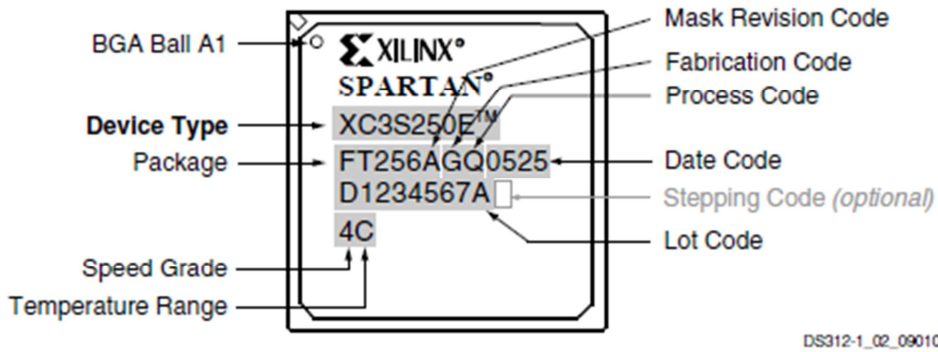


Figure II.2: Spartan-3E package BGA Marquage Exemple

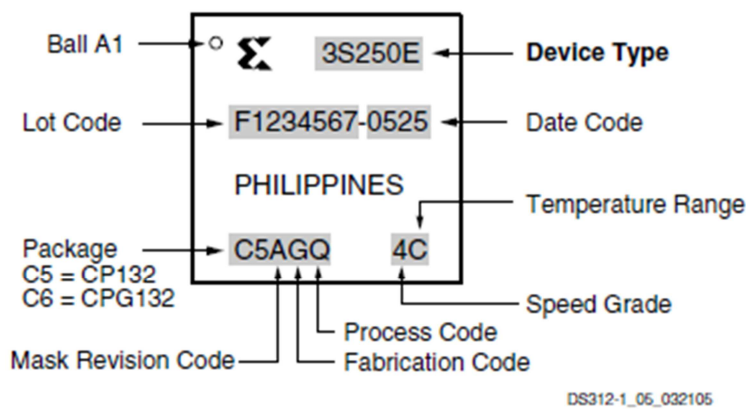


Figure II.3: Spartan-3E CP132 et CPG132 marquage de l'emballage Exemple

4- Architecture interne d'un circuit SPARTAN 3E

L'architecture d'un circuit SPARTAN 3 E se compose de cinq éléments programmables fondamentaux. Ces éléments sont organisés comme indiqué par la figure ci- dessous [9].

- Blocs logiques configurables (CLBs).
- Blocs d'entrées/sorties programmables (IOBs).
- Blocs mémoire RAM
- Multiplieurs de 18 bits * 18 bits.
- GLOBAL CLOCK

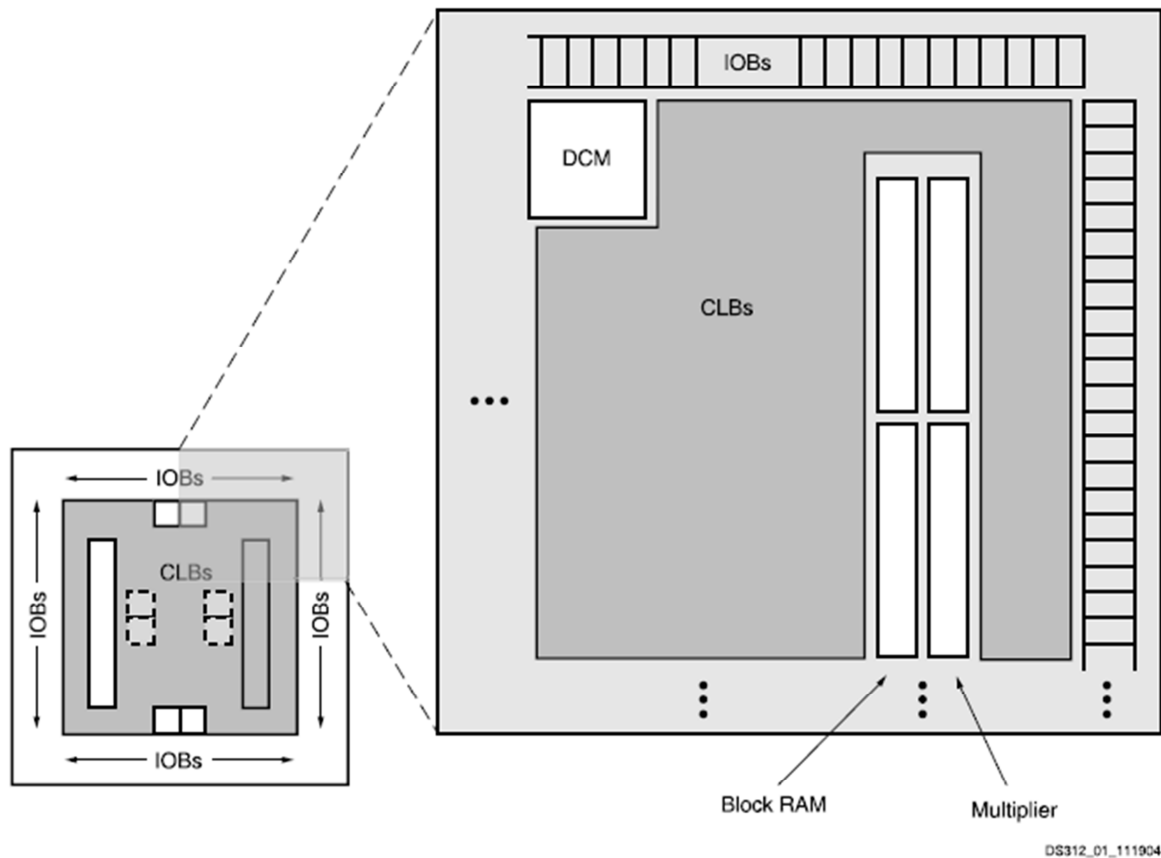


Figure II.4: Architecture d'une SPARTAN 3E [9] .

4-1- Blocs logiques configurables :

Le concept du FPGA est basé sur l'utilisation de BLC. Dans la terminologie de la compagnie Xilinx, ces blocs sont appelés CLB (Configurable Logic Block). Ces blocs constituent la principale ressource pour la mise en œuvre de la logique combinatoire ou séquentielle. Dans le Spartan 3E, chaque BLC contient quatre tranches (slices), et chaque tranche contient principalement deux générateurs de fonctions ou tables de conversion (Look-Up Table, LUT) et deux bascules.

- LUT : Chaque LUT est une mémoire de 16 bits qui permet entre autres de créer n'importe quelle fonction logique combinatoire à 4 variables d'entrée. Ces mémoires sont rapides et elles ont l'avantage d'être situées au cœur de la fonction à réaliser;
- bascules : La sortie des générateurs de fonctions peut être connectée directement à une sortie de la tranche ou bien être mise en mémoire par deux bascules D. Ces deux bascules ont la même horloge (CLK), le même signal de validation (CE) et la même logique de mise à 0 ou de mise à 1 asynchrone (SR). Les deux bascules peuvent être utilisées indépendamment ou à la suite des générateurs de fonctions.

Chaque CLB est composé de quatre cellules logiques (Slices) réparties en tranches identiques de deux slices chacune (paires) avec deux retenues indépendantes et une chaîne de décalage commune [9].

Ils sont disposés en un réseau régulier de lignes et de colonnes, comme le montre la figure (I-8).

Chaque slice contient essentiellement [9] :

- Un générateur de fonction à quatre entrées réalisé à l'aide d'une LUT (Look Up Table) à quatre entrées, d'une mémoire RAM à 16 bits ou d'un registre à décalage à 16 bits.
- Deux multiplexeurs F5MUX, FiMUX.
- Deux éléments de mémoire et de la logique arithmétique.

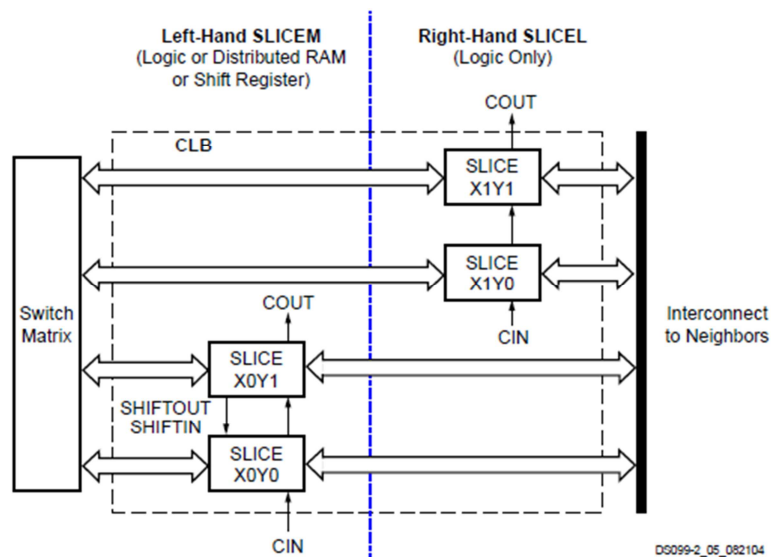


Figure II.5 : Architecture d'un CLB pour une SPARTAN 3E [10].

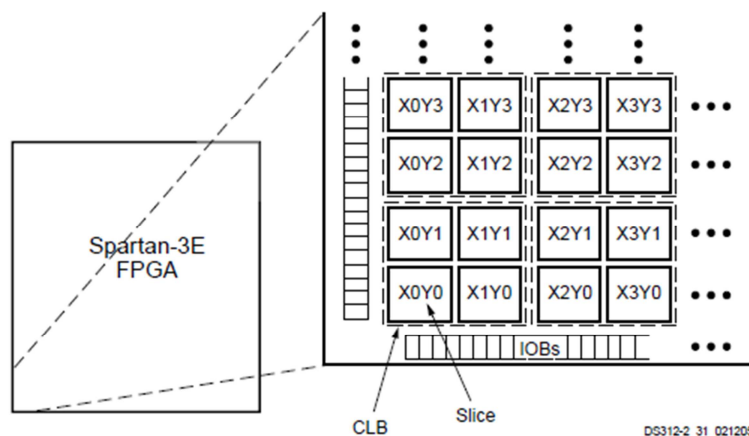


Figure II.6 : Disposition des CLBs dans un FPGA SPARTAN 3E [10]

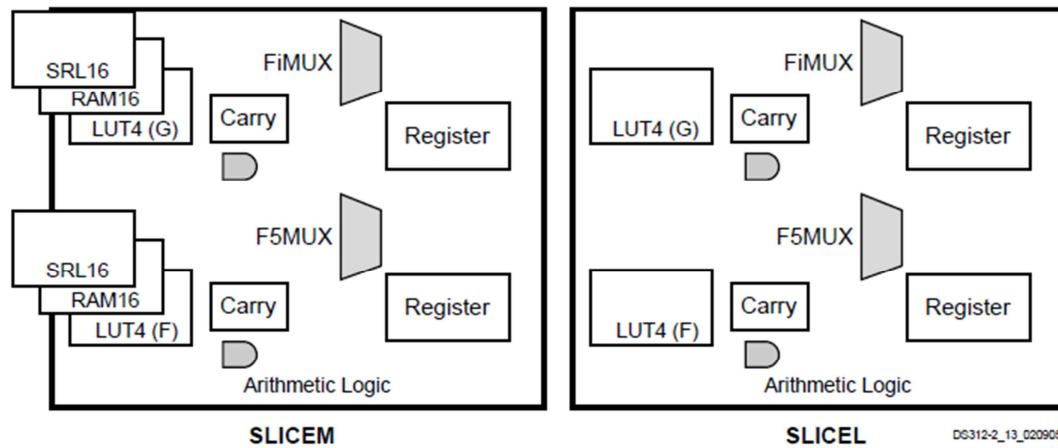


Figure II.7: Architecture d'une slice [9].

4-2- Les IOBs (Input/ Output Bloc) :

Les ports d'entrées/sorties, appelés IOB (Input-Output Blocks), sont configurables et répartis sur toute la périphérie du boîtier, Ils constituent l'interface entre les bornes du circuit et les CLB. Ils sont présents sur toute la périphérie du circuit FPGA.

Chaque bloc IOB assure l'interface entre une broche d'entrée/sortie du boîtier et la logique interne contrôle une broche du composant et peut être défini en entrée, en sortie, en signaux bidirectionnels ou être inutilisé. on distingue trois signaux principaux:

- le signal d'entrée : Il traverse un amplificateur qui, selon sa programmation, peut détecter divers standards (TTL, CMOS, etc.). Ce signal est amené vers les CLB (via les interconnexions) soit directement, soit à travers une paire de bascules D. Il peut également être retardé de quelques nanosecondes en passant par un élément de retard programmable;
- le signal de sortie : Ce signal peut être optionnellement inversé à l'intérieur de l'IOB et sortir directement sur la broche, ou bien être mis en mémoire par une paire de bascules D;
- le signal en logique trois états : Il contrôle la mise en haute impédance et la réalisation des lignes bidirectionnelles. Ce signal transporte les données à partir de la logique interne du FPGA par le biais d'un multiplexeur qui offre la possibilité d'insérer une paire de bascules D. Ainsi, ces possibilités offrent une communication facile et contrôlable avec les périphériques extérieurs.

Chaque deux IOBs peuvent être utilisés comme une paire différentielle toujours connectée à la matrice de commutation, comme le montre la figure ci-dessous.

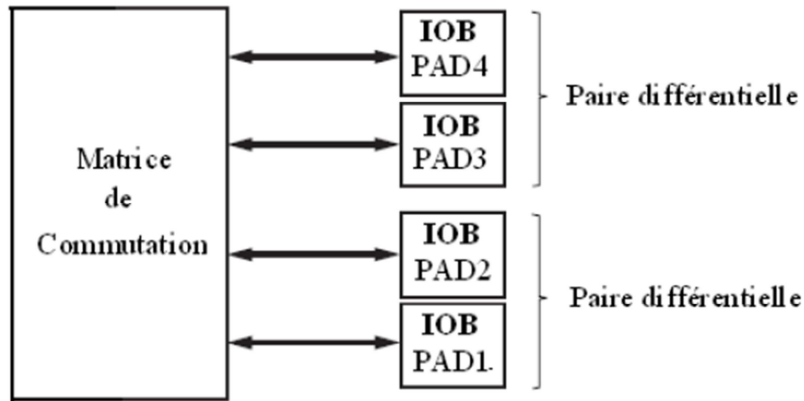


Figure II.8 : Disposition des IOBs [2].

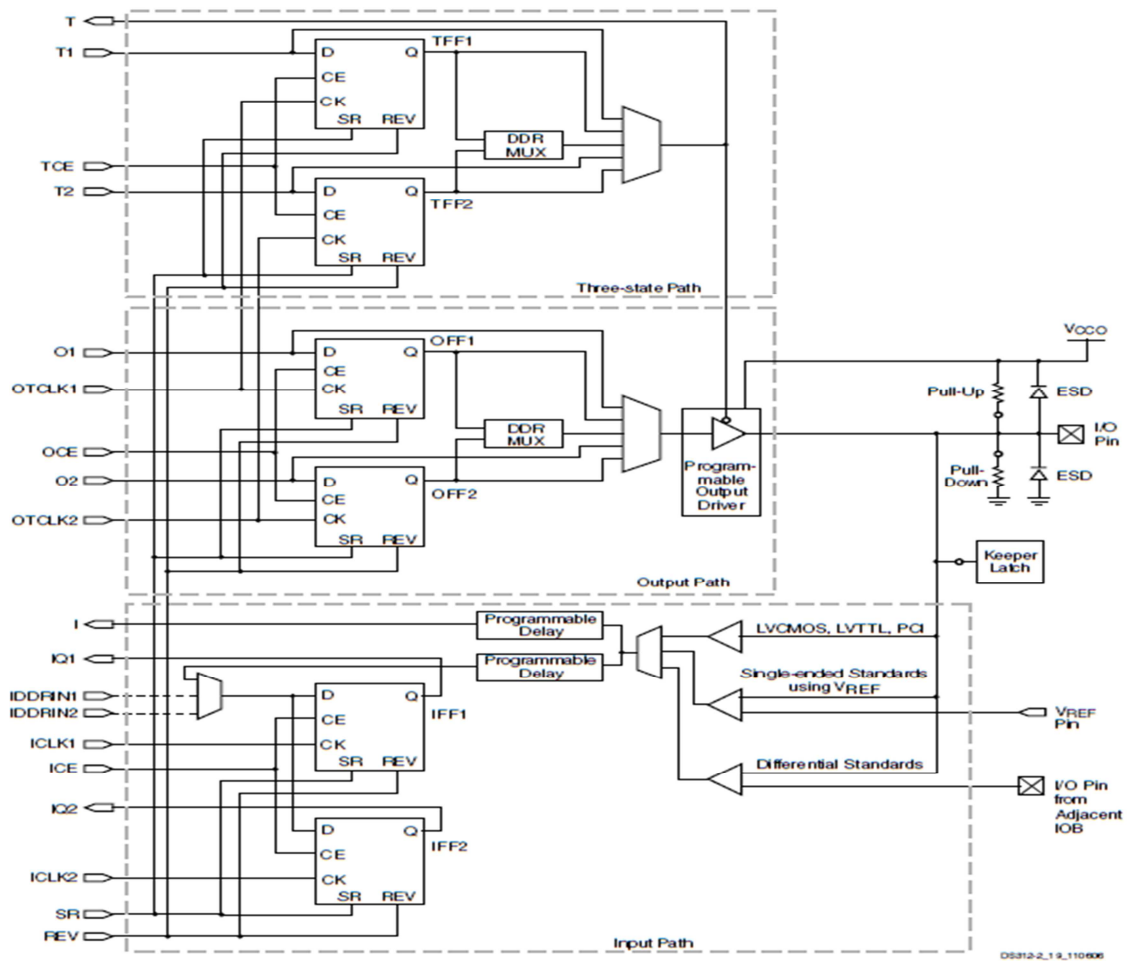


Figure II.9 : Schéma simplifié IOB [9].

4-2-1- Normes de signal de SelectIO

Les entrées et sorties de Spartan-3E supporte large éventail de normes de signalisation (tableau II.2 et tableau II.3). La majorité de l'I/Os peut également être employée pour former des paires différentielles pour soutenir des normes différentielles de signalisation (tableau II.3).

Broche IO STANDARD Driver	VCCO Alimentation / Compatibilité					Conditions d'entrée	
	1.2V	1.5V	1.8V	2.5V	3.3V	VREF	carte Termination Voltage VTT
LVTTL	-	-	-	-	Input/Output	N/R	N/R
LVC MOS33	-	-	-	-	Input/Output	N/R	N/R
LVC MOS25	-	-	-	Input/Output	Input	N/R	N/R
LVC MOS18	-	-	Input/Output	Input	Input	N/R	N/R
LVC MOS15	-	Input/Output	Input	Input	Input	N/R	N/R
LVC MOS12	Input/Output	Input	Input	Input	Input	N/R	N/R
PCI33_3	-	-	-	-	Input/Output	N/R	N/R
PCI66_3	-	-	-	-	Input/Output	N/R	N/R
HSTL_I_18	-	-	Input/Output	Input	Input	0.9	0.9
HSTL_III_18	-	-	Input/Output	Input	Input	1.1	1.8
SSTL18_I	-	-	Input/Output	Input/Output	Input	0.9	0.9
SSTL2_I	-	-	-	Input/Output	Input	1.25	1.25

Tableau II.2 : Broche IO STANDARD Driver Compatibilité [9]

Differential IO STANDARD	V _{cco}		
	1.8V	2.5V	3.3V
LVDS_25	Input	Input, On-chip Differential Termination, Output	Input
RSDS_25	Input	Input, On-chip Differential Termination, Output	Input
MINI_LVDS_25	Input	Input, On-chip Differential Termination, Output	Input
LVPECL_25	Input	Input	Input
BLVDS_25	Input	Input, Output	Input
DIFF_HSTL_I_18	Input, Output	Input	Input
DIFF_HSTL_III_18	Input, Output	Input	Input
DIFF_SSTL18_I	Input, Output	Input	Input
DIFF_SSTL2_I	Input	Input, Output	Input

Tableau II.3: différentiel compatibilité de la Banque IOSTANDARD [9]

4-3- Les multiplicateurs :

Ce sont des multiplicateurs à chiffre signé. Chaque entrée est sur 18 bits et la sortie sur Le circuit SPARTAN 3E possède aussi deux mémoires organisées en blocs de 4096 bits, situées de part et d'autres de la matrice CLB. Il offre également un bloc DCM qui permet de contrôler les décalages du signal d'horloge à l'intérieur du FPGA [2].

Les dispositifs de Spartan-3E fournir 4 à 36 multiplicateur dédié blocs par appareil. Les multiplicateurs sont regroupés avec les la RAM bloc en une ou deux colonnes en fonction de dispositif de densité. Voir Disposition des blocs de RAM sur Die pour plus de détails sur l'emplacement de ces blocs et leur connectivité.

Device	Dedicated Multipliers
XC3100E	4
XC3S250E	12
XC3S500E	20
XC3S1200E	28
XC3S1600E	36

Tableau II.4: nombre de multiplicateurs Pour chaque type du SPARTAN-3E [10].

4-3-1- Opération

Les blocs multiplicateurs exercent principalement des complément à deux multiplication numérique, mais peut également effectuer d'autres moins applications évidentes, telles que le stockage de données simple et le décalage. Les blocs de Spartan-3E multiplicateurs ont plus caractéristiques au-delà de celles prévues dans le Spartan-3 FPGA.

Chaque multiplicateur effectue le principe de fonctionnement $P = A \times B$, où «A» et «B» sont de 18 bits mots en complément à deux forme, et «P» est le plein de précision de 36 bits produit, également deux à deux complètent la forme. Les entrées de 18 bits représentent des valeurs allant à partir d'-131,07210 +131,07110 avec un produit résultant allant de -17,179,738,11210 à +17,179,869,18410.

Des opérations de multiplication plus larges sont exécutés par combinant les multiplicateurs dédiés et tranche basée sur la logique de toute combinaison ou viables par partage de temps d'un multiplicateur unique.

Effectuer la multiplication non signée en limitant les entrées à la plage positive. Attacher le bas bit le plus significatif et représentent la valeur non signé dans les 17 autres moins-bits significatifs.

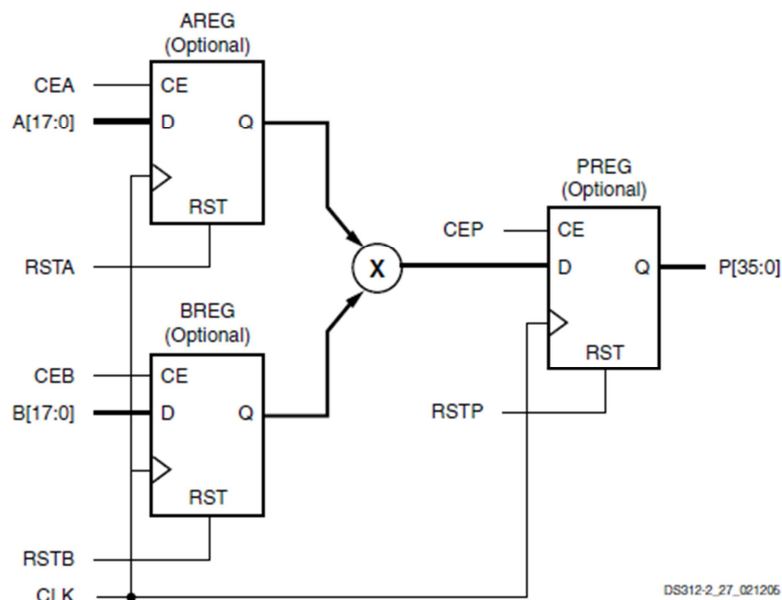


Figure II.10: Principe des ports et fonctions des blocs multiplicateurs [10].

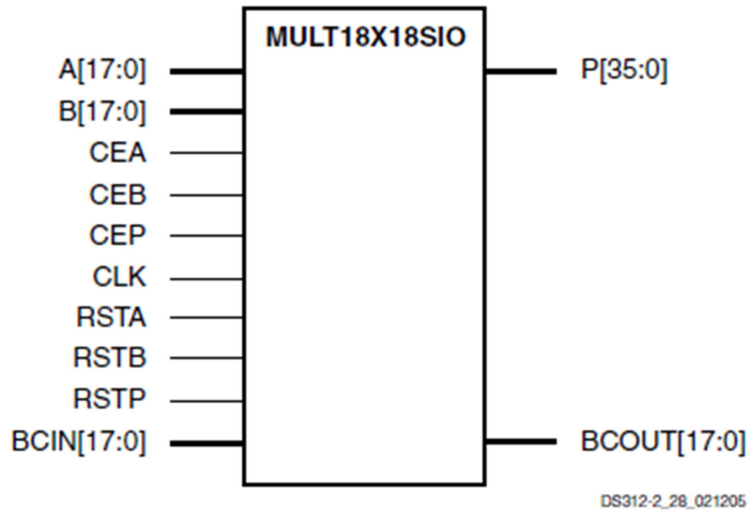


Figure II.11: MULT18X18SIO Primitive [10].

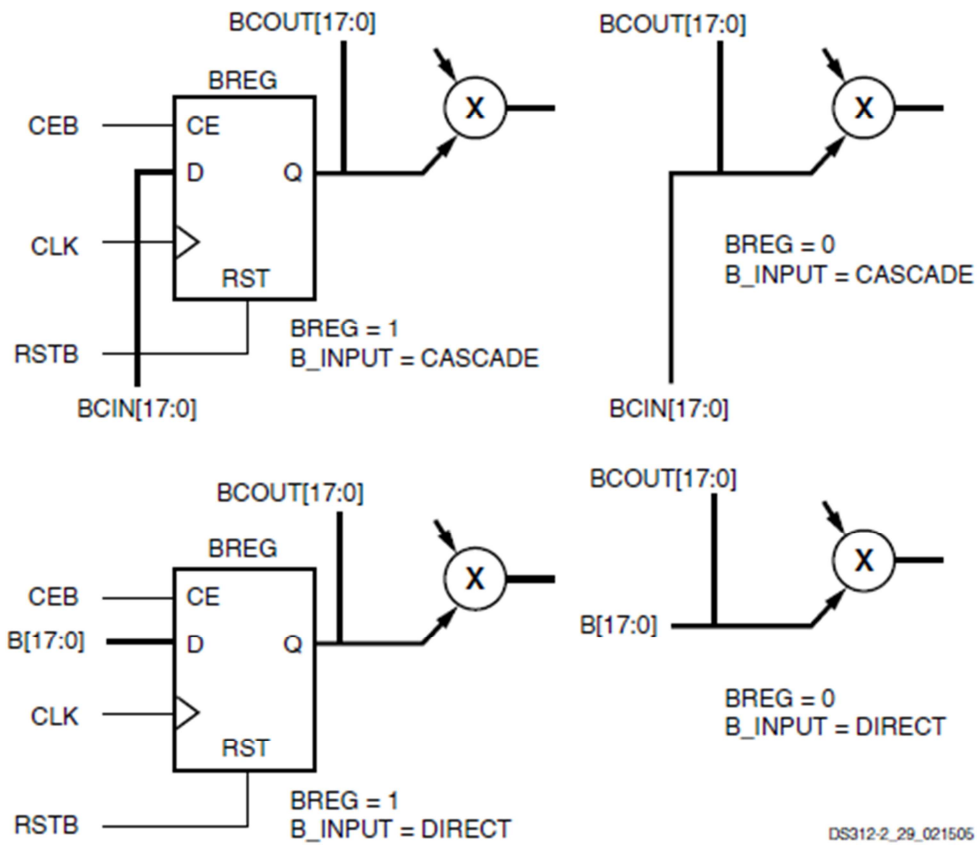


Figure II.12: Quatre configurations de l'entrée B [10].

4-4- Global Clock [10] :

Chaque génération de FPGA Spartan-3 propose huit à grande vitesse et à faible obliquité des ressources Global Clock pour optimiser les performances. Ces ressources sont utilisées automatiquement par les outils Xilinx.

Même si la fréquence d'horloge est relativement lente, il est toujours important d'utiliser les ressources Global Clock de routage pour éliminer tout risque de dangers de synchronisation.

Il est important de comprendre comment définir et tirer le meilleur parti de ces ressources.

4-4-1- Différences ressources de Global Clock [10] ;

Les FPGA Spartan-3E et étendue de la famille Spartan-3A ont identiques horloge globale ressources, avec huit entrées Global Clock et huit autres horloges sur les coté droit et gauche. L'original de la famille Spartan-3 ne propose que les huit entrées d'horloge Global.

Bien que les ressources d'horloge et de routage sont similaires, il y aura des différences temporaires entre chaque plate-forme et entre densités différentes dans une plate-forme.

Les ressources Global d'horloge composée de trois composantes :

GCLK Horloge global Plots d'entrée, multiplexeurs BUFGMUX horloge globale, et de routage d'horloge global. Voir Figure 2-1.

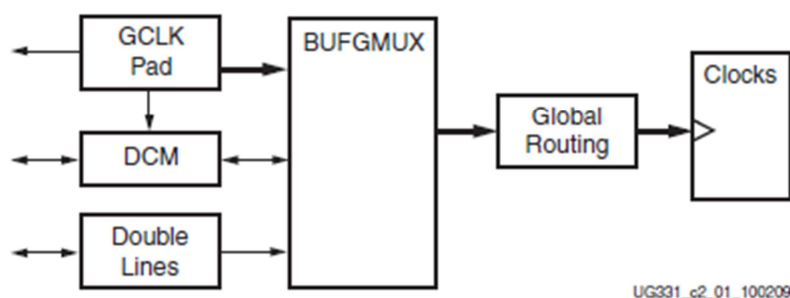


Figure II.13: Vue d'ensemble de connexions d'horloge [10]

Le détail des infrastructures Spartan-3E et étendue de la famille Spartan-3A horloge est montré dans la Figure II.14.

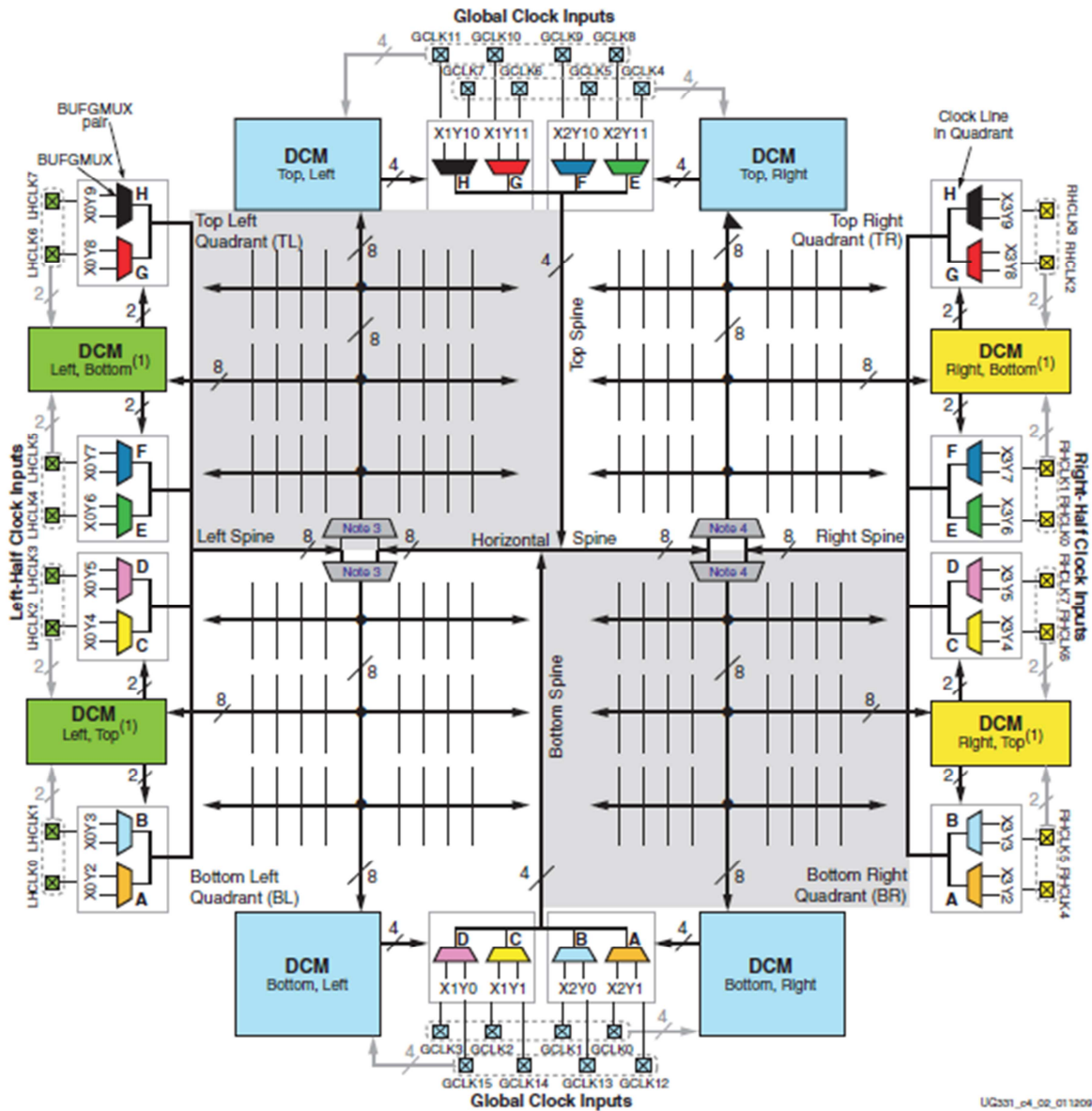


Figure II.14: Spartan-3E Structure horloge interne [10].

4-5- Le bloc RAM [10] :

Tous les FPGA Spartan-3 génération disposent de multiples RAM bloc, organisées en colonnes.

La quantité totale de bloc de mémoire RAM dépend de la taille du FPGA Spartan-3E génération en tant montré dans le tableau II.5.

Device	RAM Columns	RAM Blocks Par Column	Total RAM Blocks	Total RAM Bits	Total RAM Kbits
XC3S100E	1	4	4	73,728	72k
XC3S250E	2	6	12	221,184	216k
XC3S500E	2	10	20	368,640	360k
XC3S1200E	2	14	28	516,096	504k
XC3S1600E	2	18	36	663,552	648k

Tableau II.5: Ram block Pour chaque FPGA SPARTAN 3 [10].

5- Resistances Pull-Up et Pull-Down [9]:

Les résistances Pull-Up et Pull-Down à l'intérieur de chaque IOB forcent sur option une entrée-sortie de flottement ou broche réservée à l'entrée à un état déterminé. Les résistances Pull-Up et Pull-Down sont généralement appliquées à I/Os inutilisé, entrées, et sorties de trois-état, mais peuvent être employées sur n'importe quelle I/O ou IP Broche. Les Pull-Up résistance connectent un IOB à VCCO par une résistance. La valeur de résistance dépend de la tension de VCCO ,La résistance Pull-Down connecte un IOB à la terre avec une résistance.

Par défaut, les résistances Pull-Down connectent a toutes les I/O ou IP inutilisée et les peuvent alternativement être placées au PULLUP ou FLOAT. Pour changer l'arrangement inutilisé de protection d'entrée-sortie, placez l'option UnusedPin du générateur de Bitstream au PULLUP ou FLOAT.

Pendant la configuration un niveau bas de logique sur la broche de HSWAP active les résistances Pull-Up sur toutes les I/O

Voltage Vcco	Valeur du Resistance de Pull-up
VCCO = 3.0 à 3.45V	2.4 à 10.8 kΩ
VCCO = 2.3 à 2.7V	2.7 à 11.8 kΩ
VCCO = 1.7 à 1.9V	4.3 à 20.2 kΩ

Tableau II.6: Valeur du Resistance de Pull-up [9].

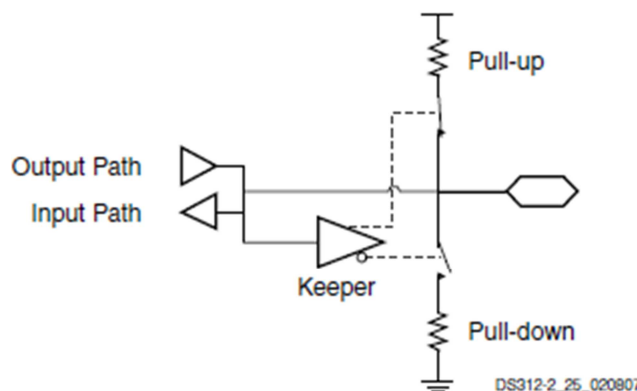


Figure II.15: Circuit Keeper [9].

6- Contrôle de taux (Slew Rate) et force du driver [9]:

Chaque IOB a un Slew-Rate contrôle qui mise le taux de commutation pour des sorties LVCMOS et LVTTL. L'attribut SLEW commande le taux de groupe et peut être placé SLOW (standard) or FAST.

Chaque LVCMOS et LVTTL soutient jusqu'à six forces différentes de courant driver suivant les indications du tableau 4

Pour ajuster l'intensité du courant, l'attribut DRIVE est mise à la force désirée : 2, 4, 6, 8, 12, et 16.

Le défaut de IOSTANDARD est LVCMOS25, SLOW slew-rate, et DRIVE 12 mA.

IOSTANDARD	Output Drive Current (mA)					
	2	4	6	8	12	16
LVTTL	√	√	√	√	√	√
LVCMOS33	√	√	√	√	√	√
LVCMOS25	√	√	√	√	√	-
LVCMOS18	√	√	√	√	-	-
LVCMOS15	√	√	√	-	-	-
LVCMOS12	√	-	-	-	-	-

Tableau II.7: Courant de sortie Programmable [9].

La limitation du Courant indique que chaque Broche a 7 états pas 2 alors 2 broches comme exemple a $7^2=49$ états.

Ce qui est un avantage très Puissant.

7- IOBs a organisé en banques

L'architecture de Spartan-3E organise IOBs en quatre banques d'entrée-sortie suivant les indications du schéma 13. Chaque banque maintient séparément VCCO et VREF pour chaque banque. Référez-vous au tableau 6 et au tableau 7 pour des conditions de VCCO et de VREF.

En travaillant avec des dispositifs de Spartan-3E, la plupart des normes differential d'entrée-sortie de différer sont compatibles et peuvent être combinées au sein de n'importe quelle banque indiquée. Chaque banque peut soutenir n'importe quels deux des normes différentielles suivantes : LVDS_25, MINI_LVDS_25, et RSDS_25.

8- Règles d'opérations bancaires d'entrée-sortie

En assignant I/Os aux banques, ces règles de VCCO doivent être suivies :

1. Toutes les broches de VCCO sur FPGA doivent être reliées même si une banque est inutilisée.
2. Toutes les lignes de VCCO associées au sein d'une banque doivent être placées au même niveau de tension.
3. Les niveaux de VCCO employés par toutes les normes assignées à l'I/Os de n'importe quelle banque indiquée doivent convenir. Les contrôles de logiciel de développement de Xilinx pour ceci. Le tableau II.12.
4. Si une banque n'a aucune condition de VCCO, reliez VCCO à une tension disponible, telle que 2.5V ou 3.3V.

Si les normes l'un des assignées aux entrées de la banque emploient VREF, alors on doit observer les règles supplémentaires suivantes :

1. Toutes les Broches de VREF doivent être reliées au sein d'une banque.
2. Toutes les Broches de VREF liées à la banque doivent être placées au même niveau de tension.
3. Les niveaux de VREF employés par toutes les normes assignées aux entrées de la banque doivent convenir.

Les contrôles de logiciel de développement de Xilinx pour ceci. Le tableau II.2 et II.3 décrit comment les différentes normes emploient l'alimentation de VREF.

Si VREF n'est pas exigé pour polariser les seuils de commutation d'entrée, toutes les goupilles associées de VREF au sein de la banque peuvent être utilisées comme utilisateur I/Os ou goupilles d'entrée.

9- Processus de configuration

Le fonctionnement d'un FPGA Spartan-3E est définie par le chargement des données de configuration spécifiques à l'application dans FPGA, similaire à la façon dont la fonction d'un microprocesseur, Pour les FPGA, ce processus configuration utilise un sous-ensemble des broches de périphériques, dont certains sont dédiés à la configuration; autres broches ne sont que des emprunté et retourné à l'application générale en tant I/O broches après la configuration finie.

Spartan-3E FPGA offrent plusieurs options de configuration afin de minimiser l'impact de la configuration sur la conception globale du système.

Dans certains modes de configuration, le FPGA génère une horloge et se charge à partir d'une source de mémoire externe, soit en série soit via byte-wide (un trajet d'octets de données à l'échelle).

10- Considérations pour les broches HSWAP, M [2:0], et VS [2:0]

La broche HSWAP, les broches de sélection de mode M [2:0], et les broches de la variante-sélection Les niveaux sur le M [2:0] broches et VS [2:0] broches sont échantillonnées lorsque la broche INIT_B des rendements élevés.

Trois broches FPGA M2, M1, M0 et permet de sélectionner le mode de configuration souhaité. Les paramètres du mode broches apparaissent dans le tableau 44.

Lors du démarrage de la configuration la sortie INIT_B le FPGA va Haute. Après le FPGA complète de configuration, les broches sont disponibles en mode utilisateur en tant que I/O.

	maître en série	SPI	BPI	Parallèle esclave	série esclave	JTAG
M[2:0] mode pin settings	<0:0:0>	<0:0:1>	<0:1:0>=Up <0:1:1>=Down	<1:1:0>	<1:1:1>	<1:0:1>
Data width	Serial	Serial	Byte-wide	Byte-wide	Serial	Serial
Configuration memory source	Xilinx Platform Flash	Industry-standard SPI serial Flash	Industry-standard parallel NOR Flash ou Xilinx parallel Platform Flash	Any source via microcontroller, CPU, Xilinx parallel Platform Flash , etc.	Tout source microcontroller, CPU, Xilinx Platform Flash , etc.	Tout source microcontroller, CPU, System ACE™ CF , etc.
Clock source	oscillateur interne	oscillateur interne	oscillateur interne	externe clock on CCLK pin	externe clock on CCLK pin	externe clock on TCK pin
Total I/O pins utilisé pendant la configuration	8	13	46	21	8	0
Mode de configuration pour downstream daisy-chained FPGAs	Slave Serial	Slave Serial	Slave Parallel	Slave Parallel or Memory Mapped	Slave Serial	JTAG
Stand-alone FPGA applications (no external download host)	√	√	√	Utilisation possible XCFxxP Platform Flash, en option génère CCLK	Utilisation possible XCFxxP Platform Flash, en option génère CCLK	
Uses low-cost, industry-standard Flash		√	√			
Supports optional MultiBoot, multi-configuration mode			√			

Tableau II.8: les différents Modes de Configuration [8].

La broche HSWAP définit si FPGA broches I/O ont une résistance de pull-up connectée à leur VCCO associé alimentation broches lors de la configuration ou non, comme le montre le tableau 48. HSWAP doit être valide au début de la configuration et rester constante pendant le processus de configuration.

HSWAP Value	Description
0	Résistances pull-up se connecter à l'alimentation VCCO associé pour toutes les broches d'E / S utilisateur I / O ou à double usage lors de la configuration. Résistances pull-up sont actifs jusqu'à ce la configuration complète.
1	Résistances pull-up désactivé lors de la configuration. Tout utilisateur d'E / S ou à double usage broches E / S sont dans un état haute impédance.

Tableau II.9: Comportement HSWAP [8].

HSWAP Value	I/O Pull-up Resistors during Configuration	Required Resistor Value to Define Logic Level on HSWAP, M[2:0], or VS[2:0]	
		High	Low
0	Enabled	Tiré par un haut interne résistance pull-up à la fourniture VCCO associé. Pas de pull-up externe résistance est nécessaire.	Tirée vers le bas en utilisant un de taille appropriée résistance pull-down à la masse. Pour une interface 2.5V ou 3.3V: R < 560 Ω. Pour une interface 1.8V: R < 1,1 kOhms.
1	Disabled	Tiré Haute aide d'un 3,3 à 4.7 kOhms résistance à la fourniture VCCO associé.	Tirée vers le bas à l'aide de 3,3 à 4,7 kOhms résistance à GND.

Tableau II.10: pull-up ou pull-down pour les valeurs HSWAP, M [2:0], et VS [2:0] [8].

VS [2:0] doivent avoir des valeurs logiques valides et stables au début de la configuration, il ne sont utilisés que dans le mode de configuration SPI.

VS2	VS1	VS0	SPI Read Command	Dummy Bytes	SPI Serial Flash Vendor	SPI Flash Family	iMPACT Programming Support
1	1	1	FAST READ (0x0B) (see Figure 53)	1	STMicroelectronics (ST)	M25Pxx M25PExx/M45PExx	Yes
					Atmel	AT45DB 'D'-Series Data Flash AT26 / AT25 ⁽¹⁾	Yes
					Intel	S33	
					Spansion (AMD, Fujitsu)	S25FLxxxA	
					Winbond (NexFlash)	NX25 / W25	
					Macronix	MX25Lxxxx	
					Silicon Storage Technology (SST)	SST25LFxxxA SST25VFxxxA	
					Programmable Microelectronics Corp. (PMC) AMIC Technology	Pm25LVxxx A25L	
Eon Silicon Solution, Inc.	EN25						
1	0	1	READ (0x03) (see Figure 53)	0	STMicroelectronics (ST)	M25Pxx M25PExx/M45PExx	Yes
					Spansion (AMD, Fujitsu)	S25FLxxxA	
					Winbond (NexFlash)	NX25 / W25	
					Macronix	MX25Lxxxx	
					Silicon Storage Technology (SST)	SST25LFxxxA SST25VFxxxA SST25VFxxx	
					Programmable Microelectronics Corp. (PMC)	Pm25LVxxx	
1	1	0	READ ARRAY (0xE8) (see Figure 54)	4	Atmel Corporation	AT45DB DataFlash (use only 'C' or 'D' Series for Industrial temperature range)	Yes
Autres			Réserver				

Tableau II.11: Codes Variant Select pour Divers SPI Serial PROM flash [8].

11- Configuration Mode [11] :

On compte plusieurs Configuration mode pou FPGA SPARTAN 3E :

11-1- Mode JTAG

Le FPGA Spartan-3E a dédié quatre fils IEEE 1149.1/1532 port JTAG qui est toujours disponible en tout temps le FPGA est alimenté et quelles que soient les paramètres du mode de broches.

Toutefois, lorsque les broches du mode de FPGA sont fixés pour JTAG mode (M [2:0] <= 1:0:1>), le FPGA attend d'être configuré via le port JTAG, après une mise sous tension en cas d'événement ou lorsque PROG_B est affirmé. Sélection du mode JTAG désactive simplement les autres modes de configuration. Pas d'autres broches nécessaires dans la configuration.

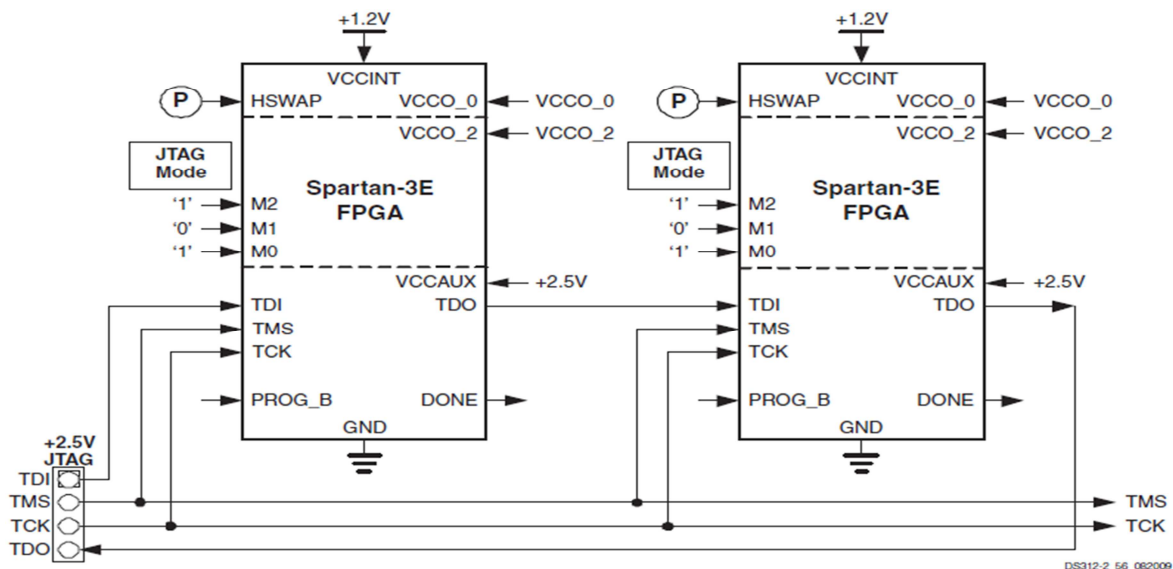


Figure II.16: JTAG Configuration Mode [11].

Spartan-3E FPGA	4-Bit Revision Code		28-Bit Vendor/Device Identifier
	Step 0	Step 1	
XC3S100E	0x0	0x1	0x1C 10 093
XC3S250E	0x0	0x1	0x1C 1A 093
XC3S500E	0x0 0x2	0x4	0x1C 22 093
XC3S1200E	0x0 0x1	0x2	0x1C 2E 093
XC3S1600E	0x0 0x1	0x2	0x1C 3A 093

Tableau II.12: Spartan-3E Identificateurs de périphériques JTAG [11].

11-3- Mode maître série

En mode maître série (M [2:0] <= 0:0:0), le FPGA Spartan-3E se configure à partir d'une plate-forme Xilinx ci-joint, Flash PROM, comme illustré sur la figure II.18.

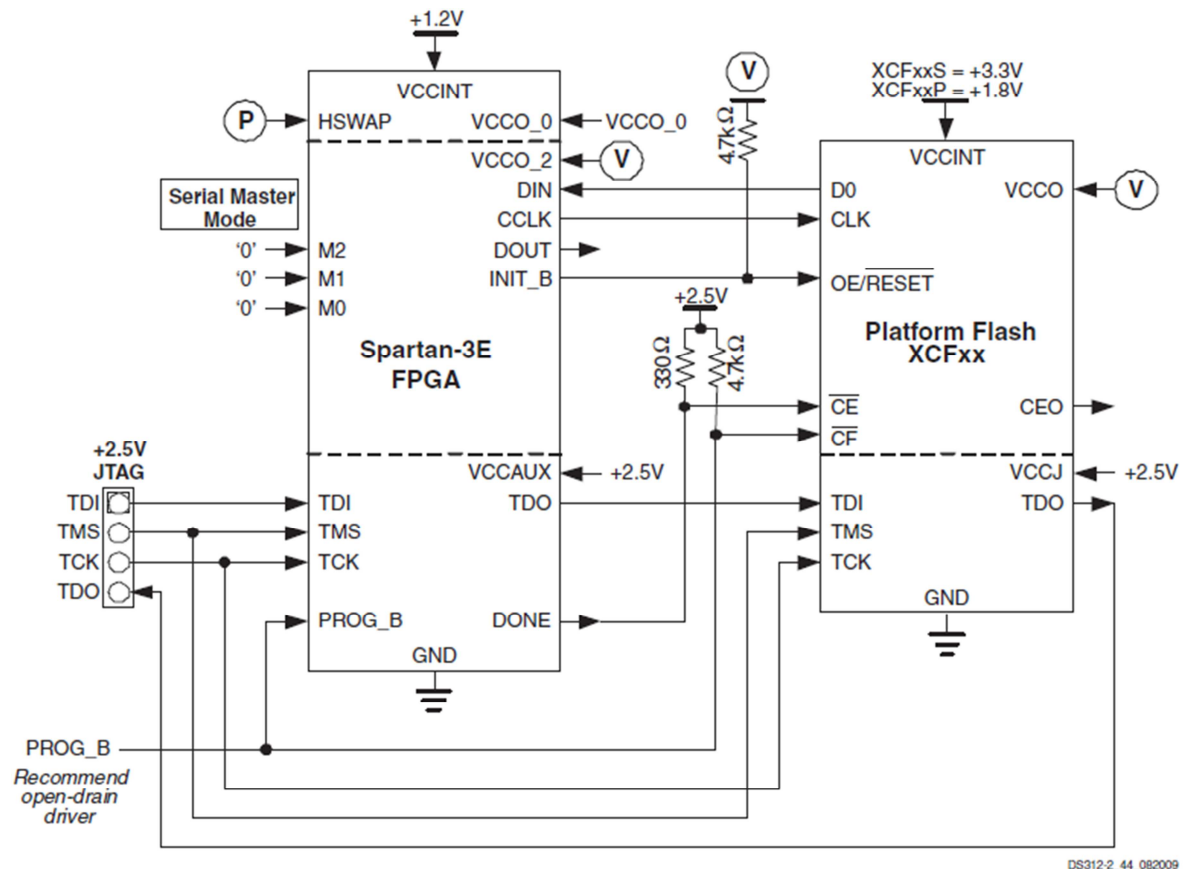


Figure II.18: Mode maître série en utilisant la plate-forme Flash PROM [11].

Le FPGA fournit un signal de sortie CCLK de son oscillateur interne à la plate-forme Flash PROM. En réponse, la plate-forme Flash PROM fournit série de données à l'entrée du FPGA DIN,

HSWAP doit rester à un niveau logique constant tout au long Configuration FPGA.

11-4- BPI Mode en parallèle Flash

Dans le mode (BPI) mode (M [2:0] = <0:1:0> ou <0:1:1>), un FPGA Spartan-3E se configure à partir d'un parallèle NOR flash PROM, comme illustré dans la figure II.19.

Le FPGA génère jusqu'à quelques lignes d'adresse de 24 bits pour accéder à un flash monté en parallèle. sauf les FPGA du Spartan-3E XC3S250E et XC3S500E génère 24 lignes d'adresses FPGA ,dans le paquet TQ144 et CP132 Seuls les 20 lignes d'adresses sont générées.

Lorsque vous utilisez le paquet VQ100, le mode de BPI n'est pas disponible lors de l'utilisation en parallèle Flash NOR, mais est pris en charge en utilisant la plate-forme parallèle Flash (XCFxxP).

Au configuration, la valeur de la broche M0 Mode détermine Common le FPGA génère des adresses.

Lorsque M0 = 0, le FPGA génère des adresses à partir de 0 et s'incrémente l'adresse indiquée sur chaque front descendant du CCLK.

Inversement, lorsque M0 = 1, le FPGA génère aborde à partir de 0xFF_FFFF (tous ceux) et décrémente L'adresse sur chaque front descendant du CCLK.

M2	M1	M0	Start Address	Addressing
0	1	0	0	Incrementing
		1	0xFF_FFFF	Decrementing

Tableau II.13: Adressage BPI contrôle [11].

Le contrôle HSWAP doit demeurer à un niveau logique constant tout au long de configuration FPGA Le B RDWR et CSI_B doit être bas tout au long de la configuration processus.

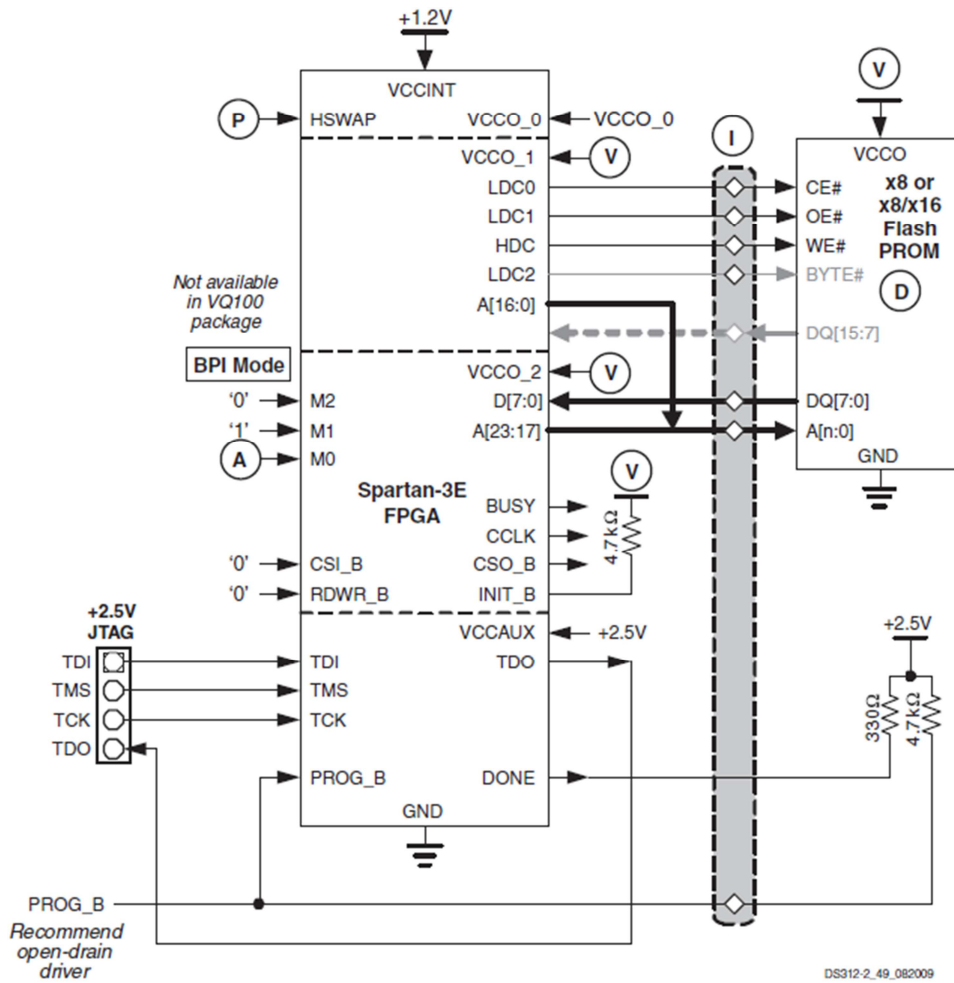


Figure II.19: Mode de l'interface périphérique (BPI) [11].

11-5- Mode ESCLAVE PARALLÈLE :

En mode parallèle esclave ($M [2:0] \leq 01:01:00$), un hôte externe, tel qu'un microprocesseur ou un microcontrôleur, écrit les données de configuration d'octets à l'échelle dans le FPGA,

en utilisant un type Interface de périphérique comme le montre la Figure II.20.

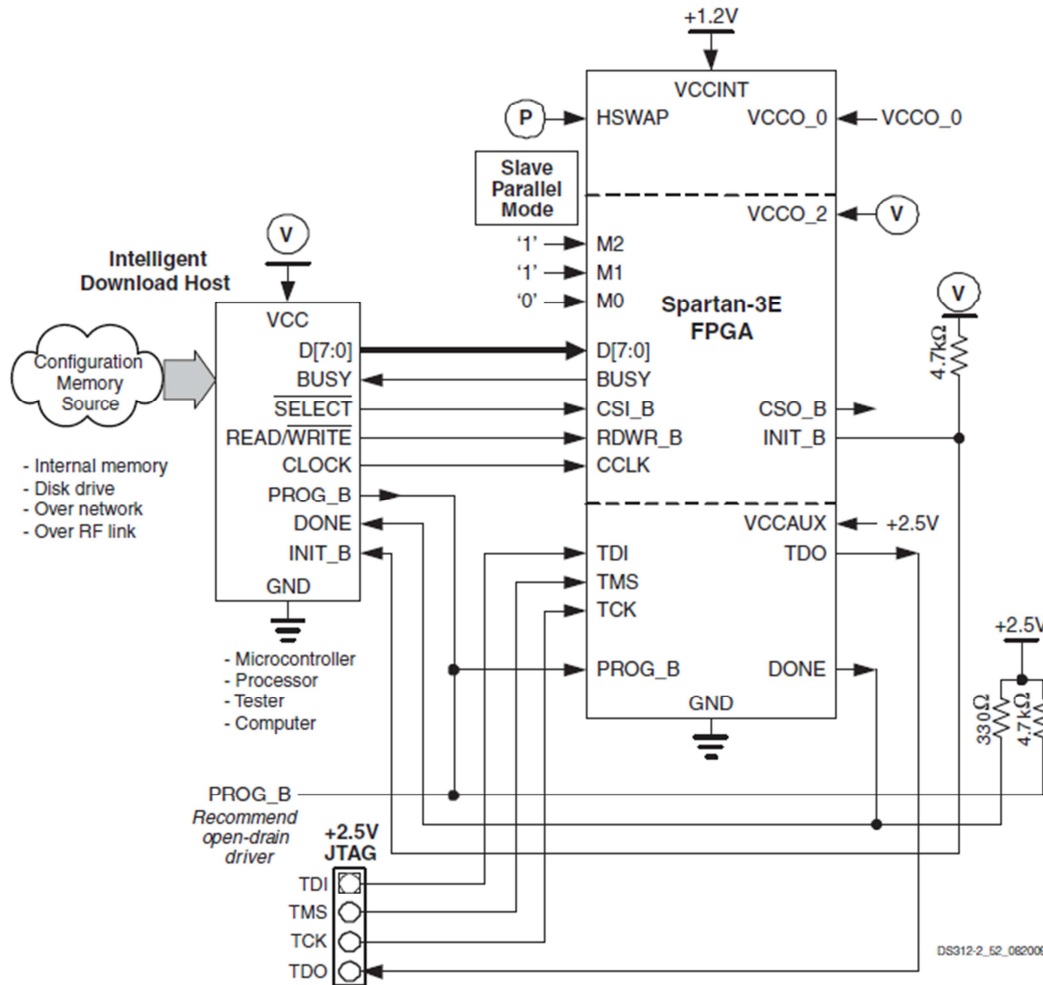


Figure II.20: Mode de configuration esclave parallèle [11].

11-6- MODE ESCLAVE SERIE :

En mode esclave série (M [2:0] <= 1:1:1), un hôte externe tel qu'un microprocesseur ou un microcontrôleur écrit les données de configuration série dans le FPGA, en utilisant l'interface série synchrone illustré à la figure II.21.

Les données de configuration en série est présenté sur la broche d'entrée du FPGA DIN avec le temps d'envoi suffisamment avant chaque front montant de l'entrée d'horloge généré extérieurement CCLK.

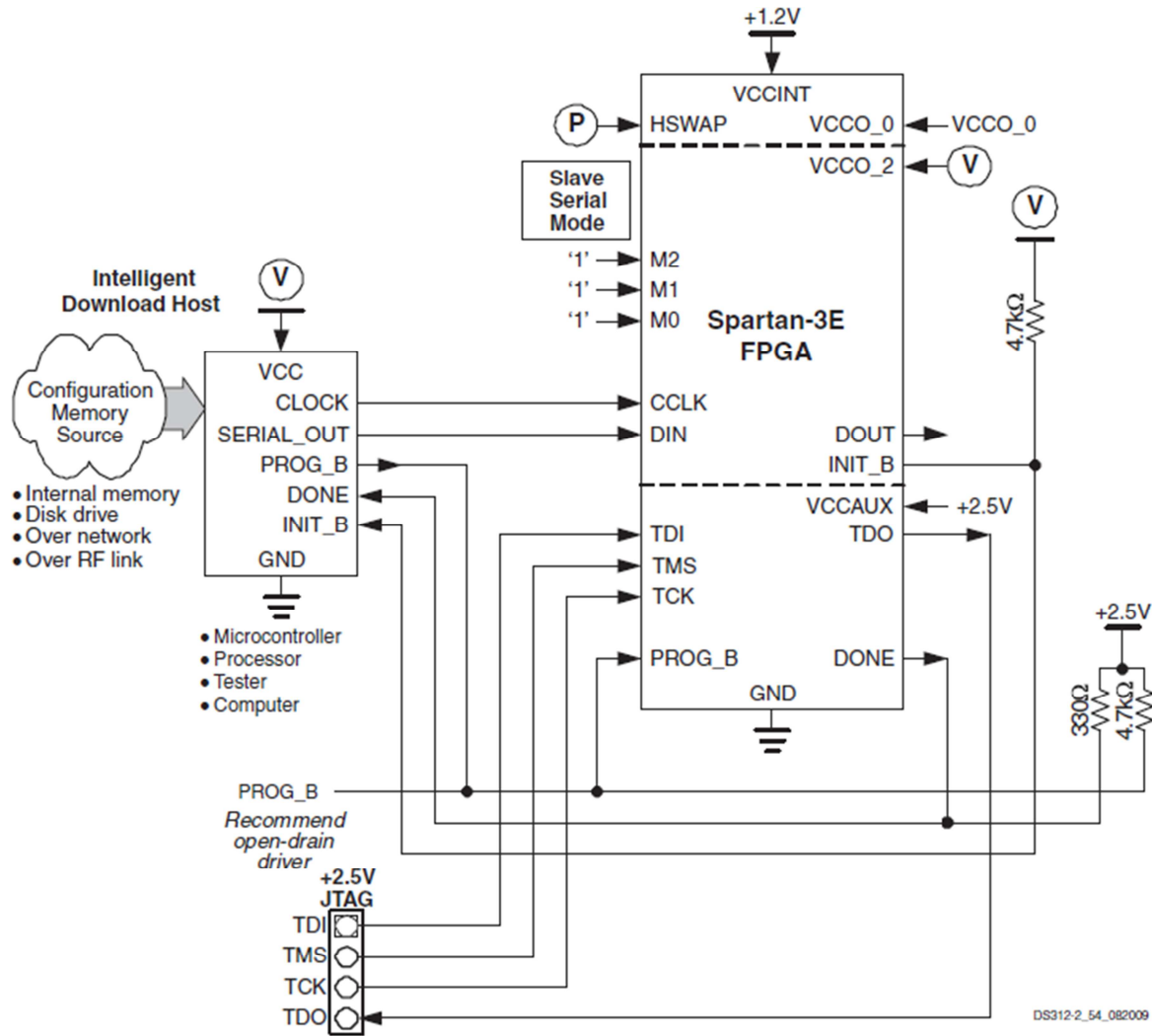


Figure II.21: Slave Serial Configuration [11].

12- Alimentation

Spartan-3 FPGA a de multiples entrées d'alimentation de tension, comme indiqué dans le tableau II.14.

Il ya deux entrées d'alimentation pour les fonctions logiques internes, VCCINT et VCCAUX.

Chacun des quatre I/O banques dispose d'une entrée séparée VCCO, Toutes les connexions Vcco du banque doit être connecté et doit se connecter à la même tension.

Alimentation d'entrée	Description	Tension d'alimentation nominale
V_{CCINT}	tension d'alimentation Interne de base. alimente toutes les fonctions logiques internes, tels que CLB, bloc de mémoire RAM, et les effets multiplicateurs.	1.2V
V_{CCAUX}	Tension d'alimentation auxiliaire. Alimente les gestionnaires Digital Clock (DCMS), les drivers de différentiels, Broches de configuration dédiés, l'interface JTAG.	2.5V
VCCO_0	Alimente les Broches I/O du Banque 0.	sélectionnable, 3.3V, 2.5V, 1.8, 1.5V, ou 1.2V
VCCO_1	Alimente les Broches I/O du Banque 1, En Byte-Wide Interface (BPI) Mode périphérique flash parallèle, se connecte à la même tension que la mémoire Flash PROM.	sélectionnable, 3.3V, 2.5V, 1.8, 1.5V, ou 1.2V
VCCO_2	Alimente les Broches I/O du Banque 2, Se connecte à la même tension que la source de configuration FPGA. (SPI).	sélectionnable, 3.3V, 2.5V, 1.8, 1.5V, or 1.2V
VCCO_3	Alimente les Broches I/O du Banque 3.	sélectionnable, 3.3V, 2.5V, 1.8, 1.5V, or 1.2V

Tableau II.14: Les alimentations du FPGA SPARTAN 3E [9].

Conclusion:

FPGA SPARTAN 3E présente des avantages très puissants par rapport à les séries Précédents.

Dans ce Chapitre on voit clairement que FPGA a besoin de Certain Périphérique pour interfacé alors il indispensable qu'on fait une étude des Périphérique pour l'avoir Connecté à FPGA

CHAPITRE III

Périphériques

Introduction :

Avant d'intégrer un périphérique il nous faut une connaissance au moins d'une manière générale de son fonctionnement et ces broches et comment on doit le connecté .

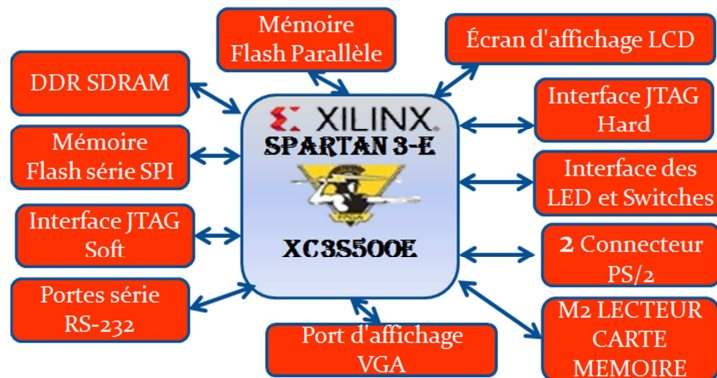


Figure III.1 : Schéma synoptique de FPGA avec les périphériques

Tous ces informations en le trouve sur internet soit pour les Connecteurs ou les composants électriques qui sont connue par :

- L'entreprise de fabrication
- Numéro de série
- Type de Package

Les Caractéristiques de chaque composant ce trouve dans son catalogue (datasheet).

Sur ce chapitre on dicte les Périphérique de la Carte :

1- Écran d'affichage LCD :

Notre Carte inclut un afficheur (LCD). FPGA commande l'affichage à cristaux liquides par l'intermédiaire de l'interface de 4 bit de données

représentée sur le schéma se dessus. Bien que l'affichage à cristaux liquides soutienne une interface à 8 bits de données, notre carte emploie

4-bit de données connectent pour rester compatibles

avec des conseils de développement de Xilinx

et pour réduire le nombre du broches utilisé.

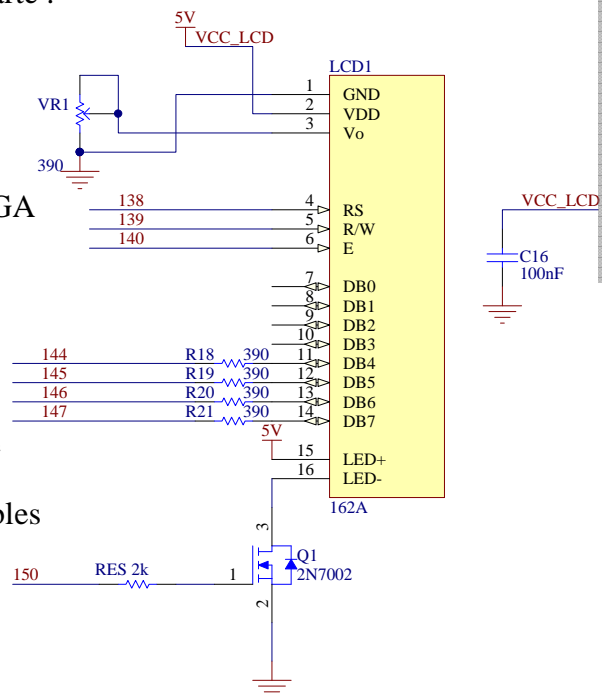


Figure III.2 : Interface de Spartan-3E FPGA avec LCD

1-1- Compatibilité de tension

LCD est alimenté par +5V. Les signaux d'entrée-sortie de FPGA sont alimenté par 3.3V. Cependant, les niveaux de sortie de FPGA sont identifiés en tant que niveaux bas ou élevés valides de logique par l'affichage à cristaux liquides. Le contrôleur d'affichage LCD accepte des niveaux de signal de 5V TTL et les sorties de 3.3V LVCMOS fournies par FPGA répondent aux exigences de niveau de tension de 5V TTL.

La plupart des applications traitent l'affichage LCD comme un périphérique d'écriture seulement et non jamais lu de l'affichage, mais on a mais des résistances série de 390Ω sur les lignes de données qui empêchent surcharger sur les broches de FPGA quand LCD conduit une valeur logique élevée.

LCD conduit les lignes de données quand LCD_RW est haut.

1-2- Contrôleur de LCD :

LCD de 2 x 16 caractères a un contrôleur interne de graphiques de Sitronix ST7066U qui est fonctionnellement équivalent avec les dispositifs suivants.

- Samsung S6A0069X ou KS0066U
- Hitachi HD44780
- SMOS SED1278

1-3- Données d'écriture à l'affichage

Pour écrire des données à l'affichage, spécifiez l'adresse de début, suivie d'un ou plusieurs valeurs de données.

Contraintes d'emplacement d'UCF

```
NET "LCD_E" LOC = "P140" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
NET "LCD_RS" LOC = "P138" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
NET "LCD_RW" LOC = "P139" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
```

```
NET " LCD_DB<4>" LOC = "P144" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
NET " LCD_DB <5>" LOC = "P145" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
NET " LCD_DB <6>" LOC = "P146" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
NET " LCD_DB <7>" LOC = "P147" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW = SLOW ;
```

2- Port d'affichage VGA :

Notre Carte inclut un port d'affichage VGA DB15. Ce port Reliée directement à un moniteur de PC ou LCD utilisant un câble standard de moniteur.

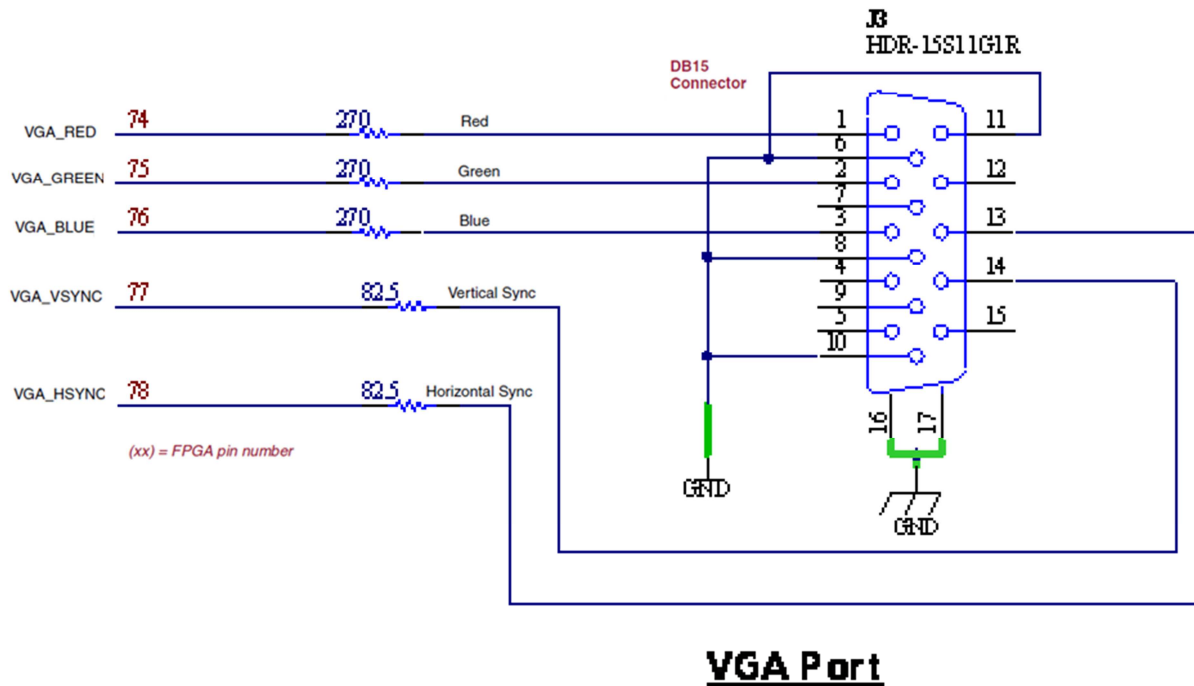


Figure III.3 : Schéma du Câblage du Port VGA avec FPGA Spartan-3E

Le Spartan-3E FPGA conduit directement les cinq signaux de VGA par l'intermédiaire des résistances. Chaque ligne de couleur a une série de résistances, avec un bit chacun pour VGA_RED, VGA_GREEN, et VGA_BLUE.

La résistance de série, en combinaison avec 75Ω établi dans le câble de VGA, s'assure que les signaux de couleur demeurent dans la gamme 0 - 0.7V.

Les signaux VGA_HSYNC et VGA_VSYNC utilisant les niveaux standards de guidage de l'entrée-sortie LVTTTL ou LVCMOS33. Conduisez les signaux VGA_RED, VGA_GREEN, et VGA_BLUE haut ou bas pour produire les huit couleurs montrées dans le tableau III.1.

VGA_ROUGE	VGA_VERT	VGA_BLUE	Couleur
0	0	0	Noire
0	0	1	Blue
0	1	0	Vert
0	1	1	Cyan
1	0	0	Rouge
1	0	1	Magenta
1	1	0	Jaune
1	1	1	Blanc

Tableau III.1 : codes couleurs de l'affichage 3-Bit [12].

La synchronisation de signal de VGA est spécifiée, et vendue par l'association de normes d'électronique visuelle (VESA). Le système de VGA et les informations suivants de synchronisation est fourni comme un exemple de la façon dont FPGA pourrait conduire le moniteur de VGA avec une résolution de 640*480. Pour des informations plus précise ou sur les plus hautes fréquences de VGA, référez-vous aux documents disponibles sur le site Web de VESA ou d'autres sites Web de l'électronique [12].

2-1- Signale et synchronisation pour un affichage de 640x480 VGA 60 hertz :

Dans un affichage de tube, les formes d'onde actuelles traversent les bobines aux champs magnétiques de produit qui braquent des faisceaux d'électrons à transversal la surface de visualisation dans un modèle de trame, horizontalement de gauche à droite et verticalement de haut en bas.

Suivant les indications du figure III.3, l'information est seulement montrée quand le faisceau se déplace le direction-gauche en avant pour redresser et dessus fond-et pas pendant au temps où le faisceau retourne de nouveau au bord gauche ou supérieur de l'affichage.

Une grande partie du temps potentiel d'affichage est donc perdu en masquant des périodes où le faisceau est remis à zéro et stabilisé pour commencer un nouveau passage horizontal ou vertical d'affichage.

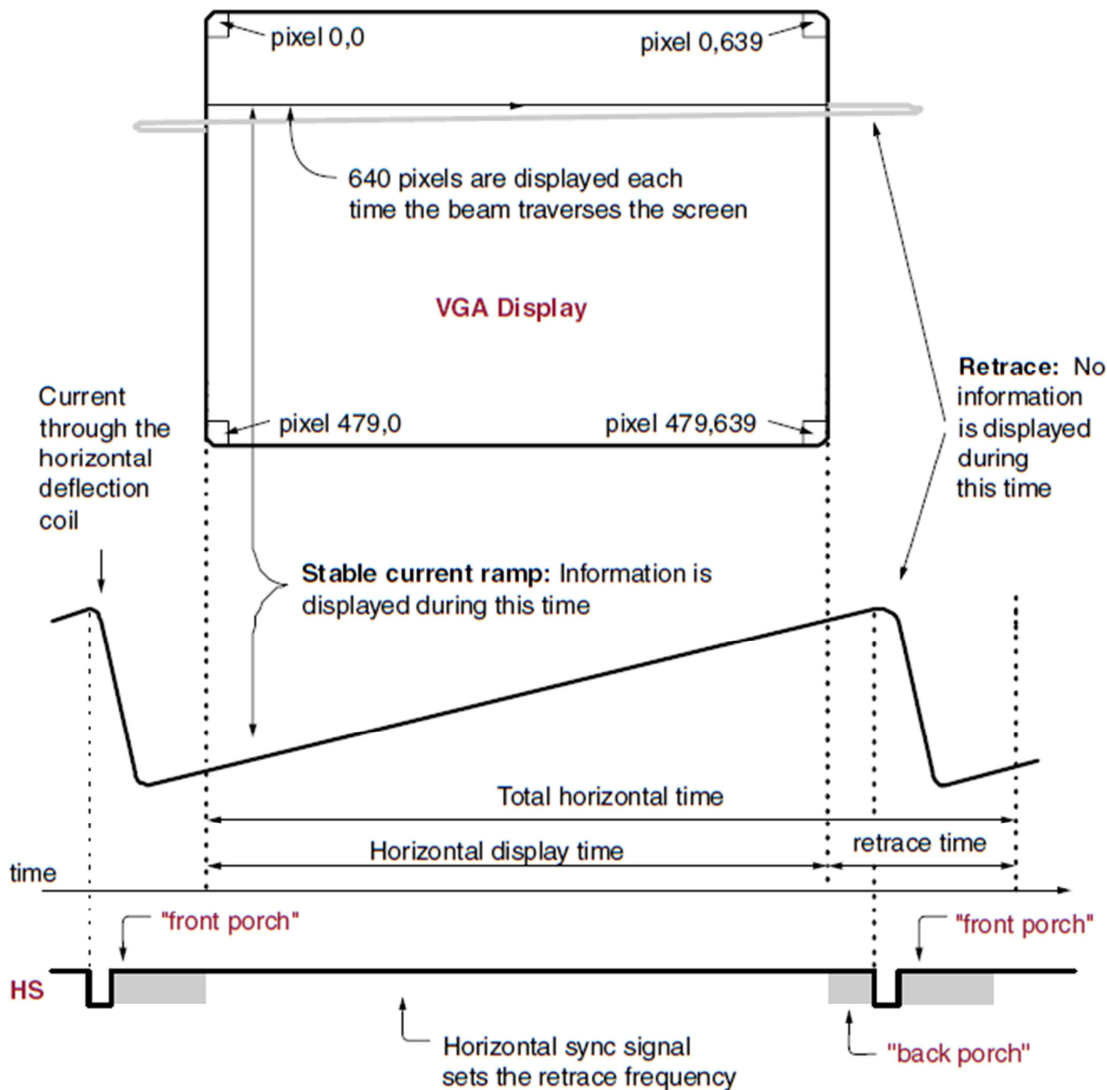


Figure III.4 : synchronisation d'affichage [12].

La résolution d'affichage définit la taille des faisceaux, la fréquence auxquels le faisceau trace à travers l'affichage, et de la fréquence à laquelle le faisceau d'électrons est modulé.

Le contrôleur produit les impulsions de synchronisation niveau TTL qui placent la fréquence à laquelle le courant traverse les bobines de débattement, et il s'assure que le signal de pixel ou vidéo est appliqué aux canons électroniques au temps correct.

Suivant les indications de la figure III.3, le contrôleur de VGA produit des signaux de synchronisation horizontale (HS) et de synchronisation verticale (VS) de et coordonne la livraison des signaux vidéo sur chaque horloge de pixel. L'horloge de pixel définit le temps disponible pour montrer un pixel d'information.

Le signal VS définit la fréquence de régénération de l'affichage, ou la fréquence à laquelle toute l'information sur l'affichage est refaite.

Notre carte FPGA emploie trois bits par pixel, produisant une des huit couleurs possibles montrées dans le tableau III.1.

$$3\text{Bit} * 640 * 480 * 60 = 55296000 = 55,296 \text{ Mhz/S}$$

Pour notre cas on a une Horloge de 27Mhz alors il faut qu'on intègre un multiplicateur de fréquence à l'intérieur du FPGA qui on peut l'aller avec jusqu'à 162 Mhz ou on déminer la fréquence d'affichage

$$F = 27000000 / 3 * 640 * 480 = 29,296875 = 29 \text{ Hz}$$

Le microprocesseur pico Blaze devise la fréquence sur 4 alors on trouve une fréquence d'affichage = $7,32421875 \text{ Hz} = 7 \text{ Hz}$

Contraintes d'emplacement d'UCF

```
NET "VGA_RED" LOC = "P74" | IOSTANDARD = LVTTTL | DRIVE = 8 | SLEW = FAST ;
NET "VGA_GREEN" LOC = "P75" | IOSTANDARD = LVTTTL | DRIVE = 8 | SLEW = FAST ;
NET "VGA_BLUE" LOC = "P76" | IOSTANDARD = LVTTTL | DRIVE = 8 | SLEW = FAST ;
NET "VGA_HSYNC" LOC = "P77" | IOSTANDARD = LVTTTL | DRIVE = 8 | SLEW = FAST ;
NET "VGA_VSYNC" LOC = "P78" | IOSTANDARD = LVTTTL | DRIVE = 8 | SLEW = FAST ;
```

3- Portes série RS-232 :

notre carte a un porte série RS-232 DB9 femelle DCE qui se relie directement au connecteur de porte série disponible sur la plupart des PCs par l'intermédiaire d'un câble série. Le modem, les commutateurs de genre, ou les câbles de croisement ne sont pas exigés.

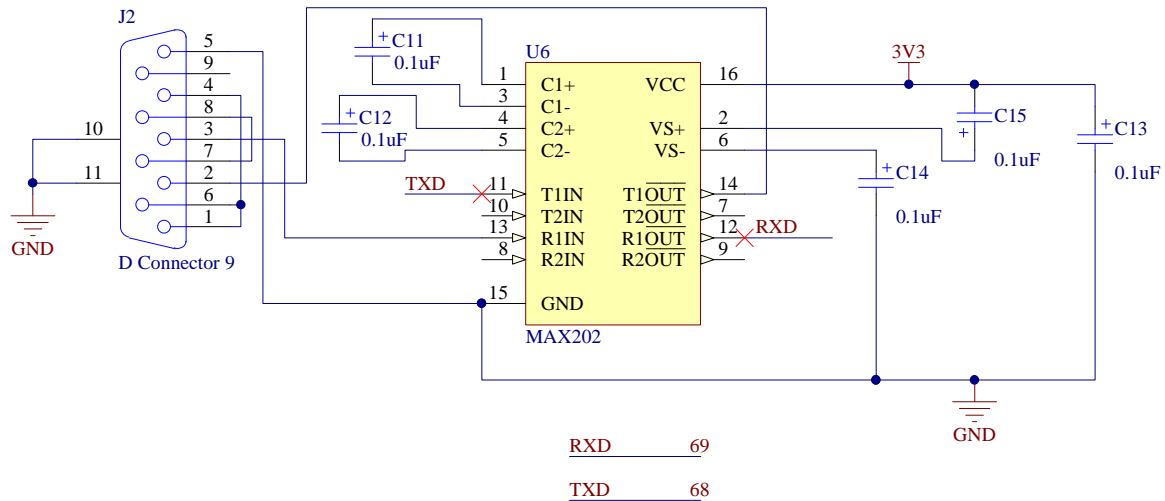


Figure III.5 : Schéma du Câblage du Portes série RS-232 .

La Figure III.5 montre la connexion entre FPGA et le connecteur DB9.

FPGA fournit des données périodiques utilisant LVTTTL ou LVCMOS niveaux au dispositif de maxime, qui convertit la valeur logique en niveau de tension RS-232 approprié. De même, le dispositif de maxime convertit les données d'entrée RS-232 périodiques en niveaux de LVTTTL pour FPGA.

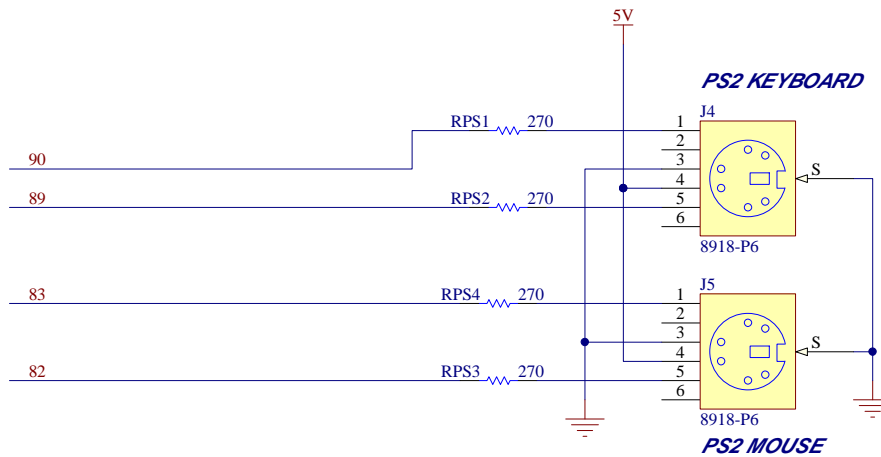
On peut ajouter une résistance de série entre la broche de maxime et du RXD de FPGA qui se protège contre des conflits accidentels de logique.

Contraintes d'emplacement d'UCF

```
NET "RS232_RXD" LOC = "P69" | IOSTANDARD = LVTTTL ;
NET "RS232_TXD" LOC = "P68" | IOSTANDARD = LVTTTL | DRIVE = 8 | SLEW = SLOW ;
```

4- Connecteur PS/2 :

Notre Carte inclut un 2 connecteur de 6-pin PS/2 de la souris et clavier



PS/2 Ports

Figure III.6 : Schéma du Câblage des connecteurs PS/2

la souris et le clavier de PC utilisent le port PS/2 série à deux fils pour communiquer avec un serveur (le FPGA dans ce cas). La bus PS/2 inclut l'horloge et les données.

PS/2 DIN Pin	Signal	FPGA Pin
1	DATA (PS2_DATA)	G13
2	Reserved	G13
3	GND	GND
4	+5V	—
5	CLK (PS2_CLK)	G14
6	Réservé	G13

Tableau III.2 : PS/2 connecteur Brochages [12] .

La souris et le clavier conduisent la bus avec des synchronisations identiques de signal et chacun des deux emploient les mots de 11 bits qui incluent un peu de début, d'arrêt et d'imparité. Cependant, les paquets de données sont organisés différemment pour une souris et un clavier. En outre, l'interface de clavier permet les transferts des données bidirectionnels ainsi le dispositif de centre serveur peut illuminer les LED sur le clavier [12].

La synchronisation de bus PS/2 apparaît sur le tableau III.3 et le schéma III.5. L'horloge et les signaux de données sont seulement conduits quand les transferts des données se produisent ; autrement ils sont jugés dans l'état inactif (logique hauts).

La synchronisation définit des conditions de signal pour des communications de souris-à-FPGA et des communications bidirectionnelles de clavier. Suivant les indications du Figure III.6.

Symbol	Parameter	Min	Max
T_{CK}	Clock High or Low Time	30 μ s	50 μ s
T_{SU}	Data-to-clock Setup Time	5 μ s	25 μ s
T_{HLD}	Clock-to-data Hold Time	5 μ s	25 μ s

Tableau III.3 : Synchronisation de bus PS/2 [12].

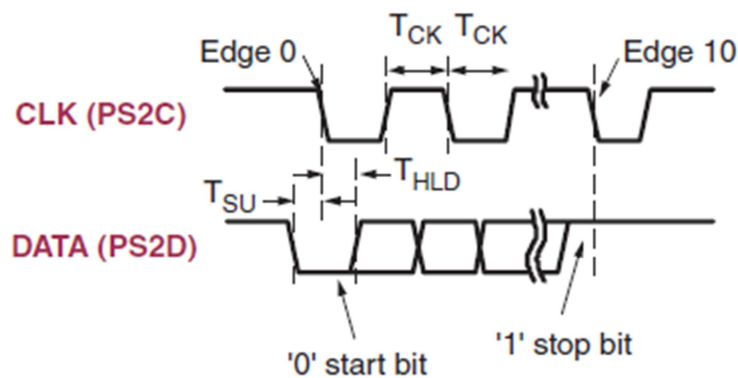


Figure III.7 : Formes d'onde de synchronisation de bus PS/2 [12].

4-1- Clavier :

Le clavier emploie des conducteurs d'ouvert-collecteur de sorte que le clavier ou FPGA puisse conduire les deux fils

Chaque clé a un code de balayage simple et unique qui est envoyé toutes les fois que la clé correspondante est pressée. Les codes de balayage pour la plupart des clés apparaissent sur la figure III.6 .

Si la clé est pressée et tenue, le clavier envoie à plusieurs reprises au code de balayage chaque 100 ms ou ainsi. Quand une clé est lâchée, le clavier envoie un code de la clé- « F0 », suivi du code de balayage de la clé sortie.

ESC 76	F1 05	F2 06	F3 04	F4 0C	F5 03	F6 0B	F7 83	F8 0A	F9 01	F10 09	F11 78	F12 07	↑ E0 75	
~ 0E	1! 16	2@ 1E	3# 26	4\$ 25	5% 2E	6^ 36	7& 3D	8* 3E	9(46	0) 45	-_ 4E	=+ 55	Back Space ← 66	→ E0 74
TAB 0D	Q 15	W 1D	E 24	R 2D	T 2C	Y 35	U 3C	I 43	O 44	P 4D	[{ 54]} 5B	↵ 5D	← E0 6B
CapsLock 58	A 1C	S 1B	D 23	F 2B	G 34	H 33	J 3B	K 42	L 4B	:: 4C	"" 52	Enter ↵ 5A	↓ E0 72	
⇧ Shift 12	Z 1Z	X 22	C 21	V 2A	B 32	N 31	M 3A	,< 41	>. 49	/? 4A	⇧ Shift 59			
Ctrl 14	Alt 11	Space 29						Alt E0 11	Ctrl E0 14					

Figure III.8 : Codes de balayage du clavier PS/2 [12].

FPGA peut également envoyer des commandes et des données au clavier. Le tableau III.4 fournit une liste courte de quelques commandes employées souvent.

Command	Description																
ED	Activer / désactiver le Num Lock, Caps Lock, Scroll Lock . Le clavier accuse réception d'un «ED» de commande en répondant par un "FA", après l'hôte envoie un autre octet pour définir le statut LED.																
	<table border="1"> <tr> <td>7</td> <td>6</td> <td>5</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td colspan="5">Ignoré</td> <td>Caps Lock</td> <td>Num Lock</td> <td>Scroll Lock</td> </tr> </table>	7	6	5	4	3	2	1	0	Ignoré					Caps Lock	Num Lock	Scroll Lock
7	6	5	4	3	2	1	0										
Ignoré					Caps Lock	Num Lock	Scroll Lock										
EE	Echo. Sur réception d'une commande echo, le clavier répond avec le code de balayage même "EE".																
F3	Régler la vitesse de balayage de code de répétition. Le clavier accuse réception d'un "F3" en renvoyant un "FA", après l'hôte envoie un deuxième octet pour fixer le taux de répétition.																
FE	Renvoyer. Sur réception d'une commande, le clavier envoie le dernier balayage code sent.																
FF	Réinitialiser. Réinitialise le clavier.																

Tableau III.4 : Commandes de clavier PS/2 communes [12].

4-2- Souris :

Une souris une fois déplacée produit deux signaux d'horloge et de données; autrement, ces signaux demeurent hauts, indiquant l'état inactif. Chaque fois que la souris est déplacée, la souris envoie trois mots de 11 bits au FPGA. Chacun des mots de 11 bits contient un ` 0' bit de départ, suivi de 8 bits d'informations (LSB d'abord), suivi d'un peu d'imparité, et terminé avec un ` 1' bit d'arrêt. Chaque transmission de données contient 33 bits totaux.

Les trois zones d'information à 8 bits contiennent des données de mouvement suivant les indications du figure III.7. Les données sont valides au bord en baisse de l'horloge, et la période d'horloge est de 20 à 30 kilohertz.

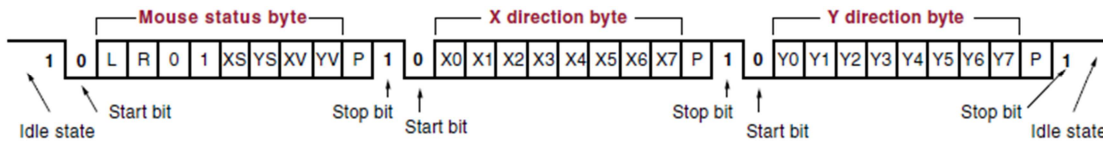


Figure III.9 : Mouse Transaction [12].

Une souris de PS/2-style utilise un système relatif, où le déplacement de la souris vers la droite produit d'une valeur positive dans le domaine de X, et le déplacement vers la gauche produit d'une valeur négative. De même, pour le déplacement sur Y,

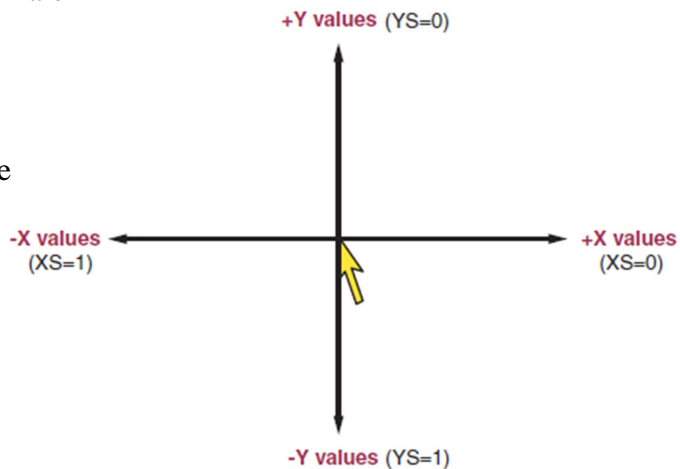
Les bits de XS et YS dans l'octet de statut définit le signe de chaque valeur, où un '1' indique une valeur négative.

L'importance des valeurs de X et de Y représentent le taux de mouvement de souris. Plus la valeur est grande, plus la souris se déplace rapidement.

Les bits de XV et de YV dans l'octet de statut indique quand les valeurs de X ou de Y dépassent leur valeur maximale

Figure III.10 : système coordinatrice

relatif de La souris [12] .



4-3- Compatibilité de tension

Le port PS/2 sur notre carte Spartan-3E est alimenté par 5V , Bien que le Spartan-3E FGA ne soit pas un dispositif 5V il utilise pour notre cas 3V3 qui est le maximum, il peut communiquer avec un dispositif 5V utilisant les résistances série comme limiteur de courant.

Contraintes d'emplacement d'UCF

```
NET "PS2A_CLK" LOC = "P82" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;
NET "PS2A_DATA" LOC = "P83" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;
NET "PS2A_CLK" LOC = "P89" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;
NET "PS2A_DATA" LOC = "P90" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;
```

5- Mémoire Flash Parallèle

Notre carte inclut 128 Mbit (16 Moctet) Intel Mémoire Flash Parallèle M29W160EB. On prend un exemple le schéma câblage du mémoire flash à FPGA 3 E

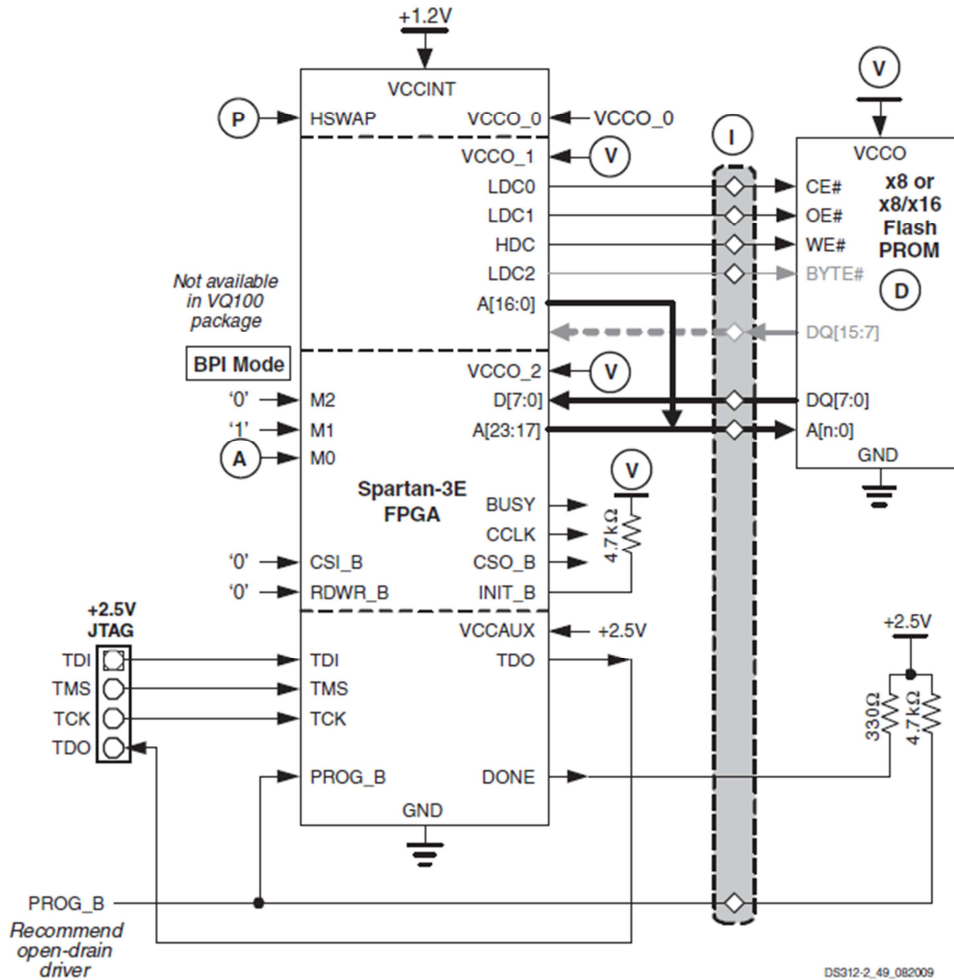


Figure III.11 : Connexions à mémoire flash parallèle [11]

sur la conception pour que FPGA peuvent être configuré a partir du mémoire flash parallèle Les broches suivants de la mémoire flash doit être connectée a des broche bien spécifié du FPGA

LDC0, LDC1, LDC2, HDC, D [7:0], A [23:0]

Le total: 35 Broches

Il est difficile de l'accorder ou peut être impossible pour notre cas à cause de la limite de notre technologie, a cause de sa on a choisie des autres broches pour PROM Flash mais FPGA ne peut pas être configurer au mode esclave parallèle ,mais il peut fournir de

diverses fonctions :

- emmagasines Code de processeur de MicroBlaze ensuite transfère le code dans la mémoire SDRAM avant d'exécuter le code.
- Stockage des données non-volatiles de FPGA.

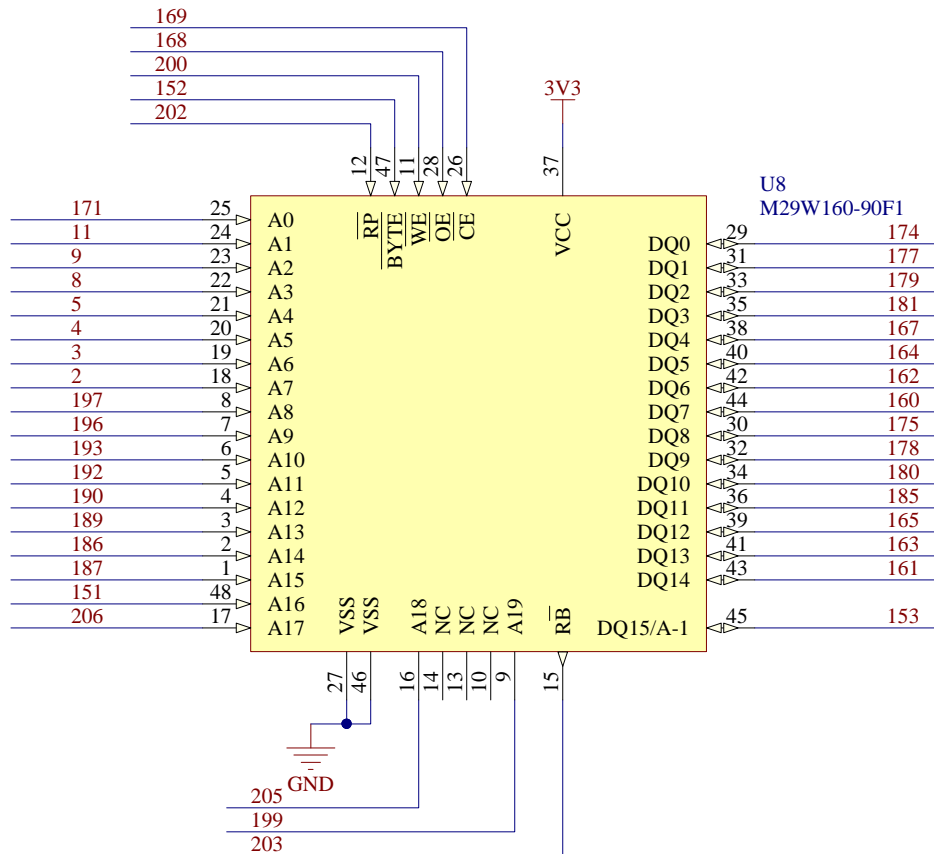


Figure III.12 : Schéma du Câblage du Mémoire Flash Parallèle

5-1- Caractéristiques [13]:

- VCC = 2.7V à 3.6V Temps d'accès: 70, 90ns
- 16 Moctet de Mémoire Flash
- Temps de programmation 10µs par Byte
- 35 blocs de mémoire
- Un code de sécurité de 64 bits
- Faible consommation d'énergie (mise en veille automatique)
- 100.000 de programmation / effacement par CYCLES

6- Mémoire Flash série SPI :

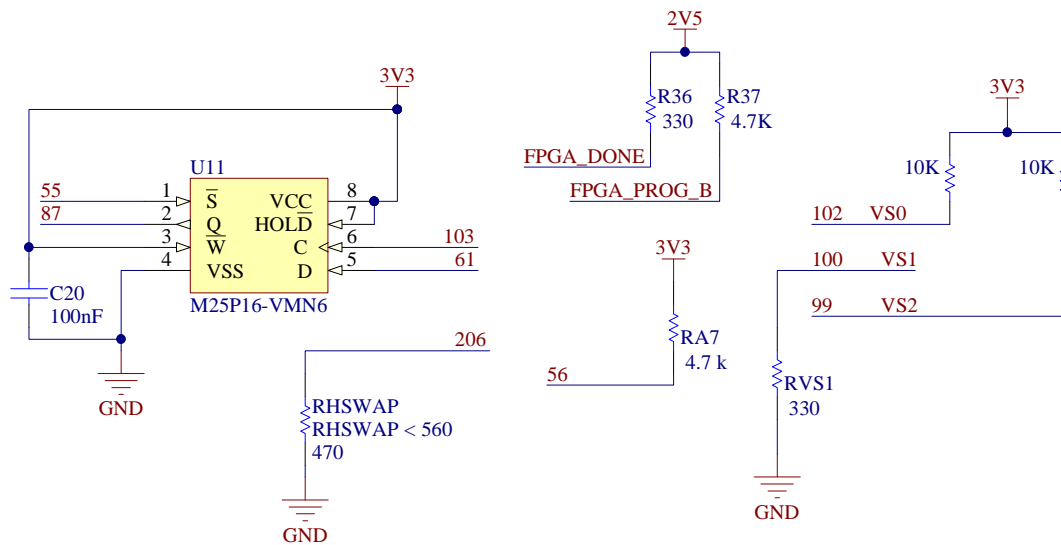


Figure III.13 : Schéma du Câblage du Mémoire Flash série SPI

Notre Carte inclut une mémoire série Flash de STMicroelectronics M25P16 16 Mbit SPI, utilisé dans plusieurs applications.

Une nouvelle caractéristique de FPGA Spartan-3E Mémoire Flash SPI est également disponible à FPGA après configuration pour un grand choix de buts, comme :

- Stockage de données non-volatile simple
- Stockage pour des codes d'identificateur, des numéros de série, des IP address, etc.
- Stockage de code de processeur de MicroBlaze .

Signal	FPGA Pin	Direction	Description
SPI_MOSI	P61	FPGA->SPI	Serial data: Master Output, Slave Input
SPI_MISO	P87	FPGA<-SPI	Serial data: Master Input, Slave Output
SPI_SCK	P103	FPGA->SPI	Clock
SPI_SS_B	P55	FPGA->SPI	Asynchronous, active-Low slave select input

Tableau III.5 : Signaux d'interface instantané de SPI [12]

Pour configurer FPGA en mode SPI, les broches de mode de FPGA doivent être placées convenablement (M [2:0] <= 0:0:1), et SPI doit contenir une image de configuration valide.

6-1- Caractéristiques [14]:

- 16 Mbit de mémoire Flash
- Programme des pages (jusqu'à 256 octets) en 0,64 ms
- Effacer un Secteur (512 Kbit) en 0,6 s
- Effacement totale (16 Mbit) en 13 s
- 2,7 V à 3,6 V de tension d'alimentation unique
- 75 MHz Horloge taux (maximum)
- Consommation faible en mode veille 1 uA .
- JEDEC standard à deux octet de signature .
- Plus de 100.000 Erase / Programme cycles par secteur
- Protection en écriture du matériel: taille de la zone protégée définie par trois bits (BP0, BP1 et BP2)
- Plus de 20 années la conservation des données

Contraintes d'emplacement d'UCF

```
NET "SPI_MISO" LOC = "P87" | IOSTANDARD = LVCMOS33 ;
```

```
NET "SPI_MOSI" LOC = "P61" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 6 ;
```

```
NET "SPI_SCK" LOC = "P103" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 6 ;
```

```
NET "SPI_SS_B" LOC = "P55" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 6 ;
```

7- DDR SDRAM :

Notre Carte inclut une SDRAM de 128 Mbit (32M x 4) K4S281632F

Description générale [15] :

Le K4S281632F est 134,217,728 bits de RAM synchrone à haut débit dynamique organisé comme 4 x 2.097.152 mots de 16 bits, fabriqués avec une haute performance de SAMSUNG

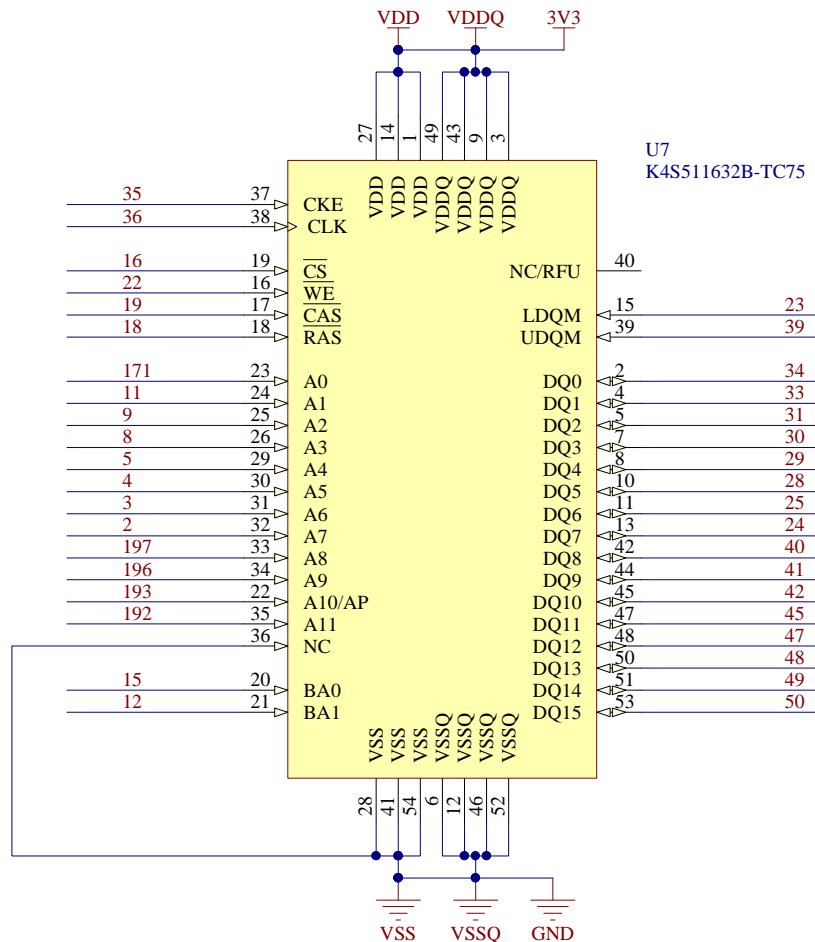


Figure III.14 : Schéma du Câblage du SDRAM

7-1- Caractéristiques [15]:

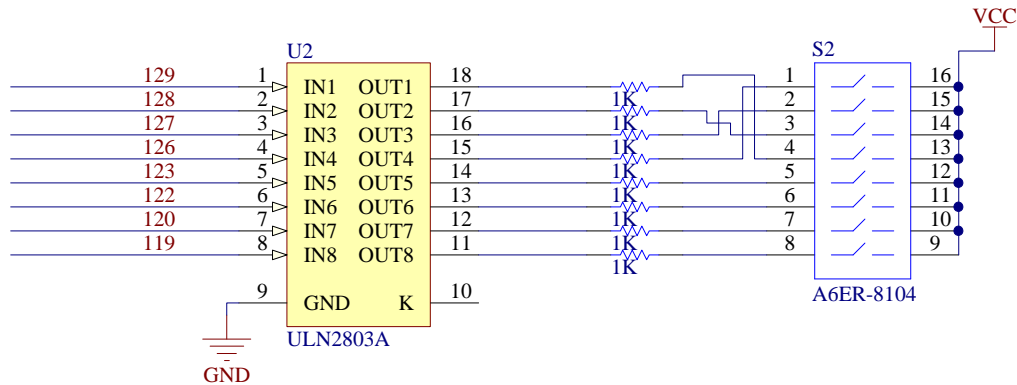
- 128 Mbit de mémoire
- JEDEC standard 3.3V alimentation
- LVTTTL compatible avec les adresses multiplexé
- 4 Bank d'opération
- Toutes les entrées sont échantillonnées au front montant de l'horloge.
- Autorafrâchissements
- période de rafraîchissement de 64ms

8- LED Switches Interface :

Les LED's Consomme un Courant Elevée Pour $V_{cc} = 9V$.

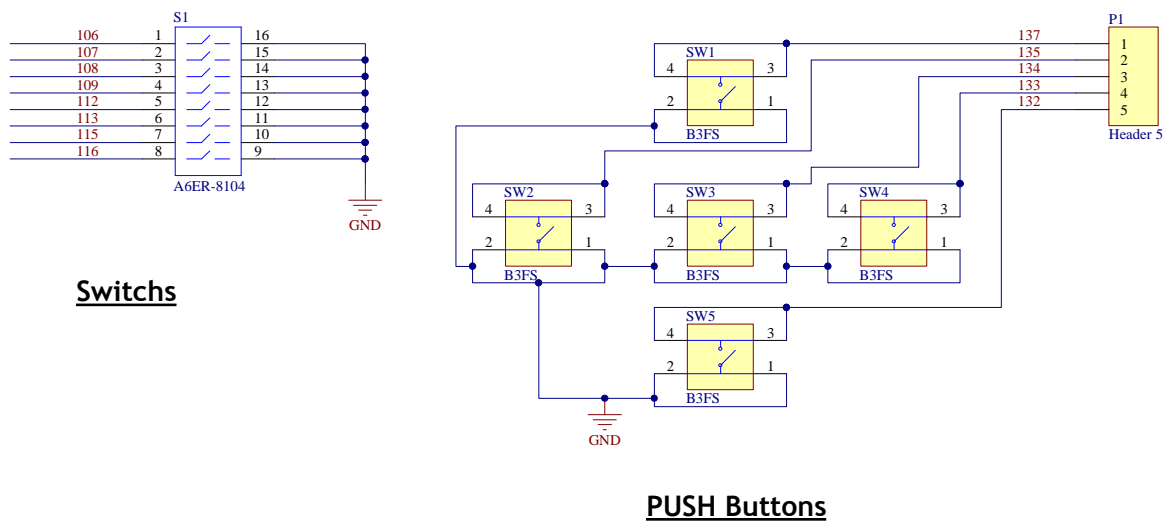
$$I_{LED's} = (9-1.2)/1k = 78 \text{ mA}$$

Alors on a ajouté un « display (LED) drivers » ULN2803AFW qui a une impédance d'entrée de 2,7 kΩ et il peut supporte un courant de sortie de 500 mA dans chaque Broche [16], on l'ajoute pour éviter l'échauffement du FPGA et du régulateur du 3V3 ou 2V5.



LEDs

Figure III.15 : Schéma du Câblage du LEDs



Switchs

PUSH Buttons

Figure III.16 : Schéma du Câblage du Switchs et PUSH Buttons

Il faut activer Les résistances du PULL-UP interne du FPGA pour chaque Broche

Pour Convertie Les états des Switch et Buttons Poussoirs du Z ou 0(GND) a 1 OU 0 Logique.

9- M2 LECTEUR CARTE MEMOIRE :

On a vérifié la position du
Vcc, Vss, NC1, NC2

sur des lecteurs de carte mémoire

Et on a aussi mesuré Vcc

qui l'on trouve = 3.3 V

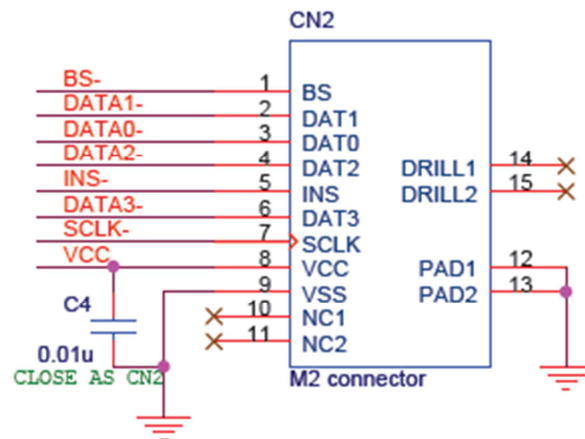


Figure III.17 : Schéma du Câblage du M2 [17]

INS : si une carte M2 est insérée il va t être court circuité au VSS sinon reste libre haute impédance

DATA [3..0] = BUS de Données

BS : Bank sélectionné

SCLK : Clock

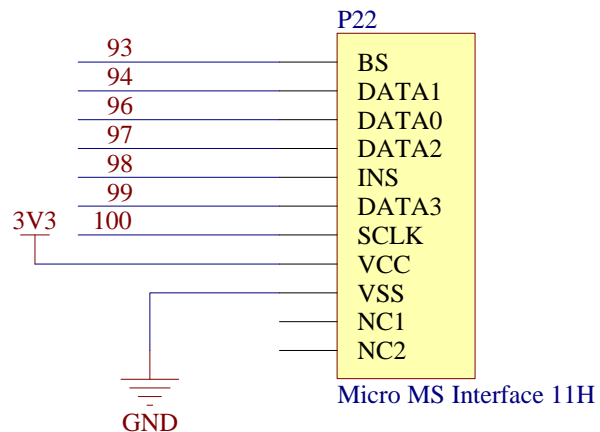


Figure III.18 : Schéma du Câblage du M2

Contraintes d'emplacement d'UCF : contraintes d'UCF pour le M2

NET "INS" LOC = "P98" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

NET "BS" LOC = "P93" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

NET "SCLK" LOC = "P100" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

NET "DATA0" LOC = "P96" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

NET "DATA1" LOC = "P94" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

NET "DATA2" LOC = "P97" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

NET "DATA3" LOC = "P99" | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = SLOW ;

Conclusion

L'étude des périphérique nous donne une idée de son fonctionnement et nous aider

A le réparer en cas où il ne fonctionne pas.

On Connue bien les besoin de chaque Composant, Pour l'alimentation on distingue deux tension 3V3 et 5V

Après d'avoir fait ces études on est prêt pour passer à la Conception ensuite la réalisation du Carte.

CHAPITRE IV

CONCEPTION ET

REALISATION

Introduction :

Pour pouvoir être programmé, le FPGA a besoin d'une carte de développement. C'est un environnement de test et de conception. Elle est présentée au figure 14 et:

- Des éléments de test comme le logiciel, permettent d'interagir avec le circuit. Le but étant de tester si le FPGA répond correctement à des instructions simples. C'est idéal pour commencer à déboguer un montage. Ces interactions se font à l'aide de commandes envoyées par exemple par l'actionnement d'un bouton poussoir. Le FPGA peut renvoyer une information sous la forme d'une séquence d'allumage de LED.
- Un connecteur pour la programmation présent pour brancher l'interface de programmation (JTAG).
- Une mémoire Flash externe qui reprogramme de façon autonome le FPGA.
- Un LCD Pour affiché des Textes
- Le circuit FPGA.

Tout cet environnement implique la présence d'une alimentation générale de la carte de 5V.

1- Le besoin d'évolution :

Le besoin de prototypage de circuit est de plus en plus présent aux laboratoires, ce qui nécessite une capacité d'intégration toujours plus grande des circuits programmables.

Les FPGA de la série Cyclone à présenter des limites de performances liées à un nombre trop faible d'éléments logiques qu'ils contiennent. Ainsi ils ne pourront pas prétendre suivre l'évolution très longtemps. Par conséquent, afin d'anticiper les besoins futurs, la recherche d'un FPGA de plus haute performance et de plus haute intégration a été indispensable. C'est la série de XILINX qui propose la plus grande capacité d'intégration et des performances très élevées.

L'évaluation de leur caractéristique est basée sur le nombre d'éléments logiques présents dans le circuit. Ce chiffre découle directement de la technologie utilisée. Par ailleurs, la performance est elle aussi un critère important. Elle est définie par le nombre de multiplieurs embarqués. Enfin le dernier critère notable est la taille de la mémoire interne, car c'est qui elle permet d'inclure un DSP et un processeur ainsi que les montages numériques conçus.

La communication avec le PC se fait via le protocole RS232. Ces communications sont gérées par le processeur présent dans la FPGA.

2- Le processus de conception d'un circuit :

Plusieurs métiers sont nécessaires aux différentes étapes de conception du produit. Le but étant de mettre en production le circuit afin qu'il puisse intégrer les plateformes clients.



Figure IV-1 : le flot de conception d'un projet

2-1- Etude de marché :

Tout commence soit par une étude de marché dans le cas d'anticipation des demandes soit par l'étude d'une demande en réponse à un appel d'offre du client. Ce sont les commerciaux qui cherchent les nouveaux projets en étudiant l'état du marché actuel et en suivant les stratégies de la compagnie. C'est au responsable de la division que revient la décision d'accepter ou non le projet. Pour cela il s'appuie sur une étude plus approfondie de la concordance avec les produits existants de l'entreprise.

2-2- Evaluation du concept :

C'est la phase de l'étude de faisabilité du circuit. Le circuit est conceptualisé et les caractéristiques demandées sont étudiées en se basant sur ce qui existe. A partir de là, il va être possible de prédéterminer les ressources et les compétences nécessaires à la mise en oeuvre du projet.

2-3- Spécifications et Architecture :

Dans le cas de la faisabilité du projet, un cahier des charges est alors rédigé. C'est durant cette phase d'architecture que les stratégies techniques sont validées pour mener à bien le projet. De plus, c'est à Figure 3 : Processus de conception d'un circuit. Ce moment que les ressources matérielles et humaines sont allouées. Une prévision temporelle et budgétaire est également présentée pendant cette phase.

2-3- Conception et implémentation :

C'est pendant cette phase que les circuits sont conçus par les concepteurs puis validés par le laboratoire dans lequel s'est effectué le stage. Le circuit peut alors passer à l'étape de caractérisation. Cette étape dresse un rapport complet de toutes les caractéristiques du produit.

2-4- Qualification :

Ce sont des tests de conformité effectués à grandes échelles pour répondre à une norme de qualité du produit. Il est par exemple demandé de vérifier la durée de vie d'un circuit.

2-5- Production :

Le circuit atteint alors une maturité suffisante pour être envoyé en production. Cette étape clôturera la phase de développement.

2-6- L'activité de validation :

Nous venons de voir que la validation fait partie intégrante de l'étape de conception d'un circuit.

La mission du laboratoire peut être scindée en deux parties : avant et après la réception des Prototypes à valider (en violet sur le graphe suivant). Au moment de la réception des documents de spécification, la personne qui est en charge de ce nouveau projet étudie le fonctionnement du circuit et son domaine d'application. Cette étude spécifie les besoins en termes de solution de validation.

C'est alors que débute la phase de conception de ces outils.

Ces derniers doivent être fonctionnels à la réception des prototypes.

Commence donc la seconde phase de l'activité : la validation du circuit. Il s'agit de vérifier le fonctionnement des prototypes qui sont testés « physiquement » grâce aux solutions développées pendant la première phase.

Tant que les résultats de ces tests ne sont pas satisfaisants, le concepteur devra corriger son schéma et relancer la production de nouvelles versions de prototypes.

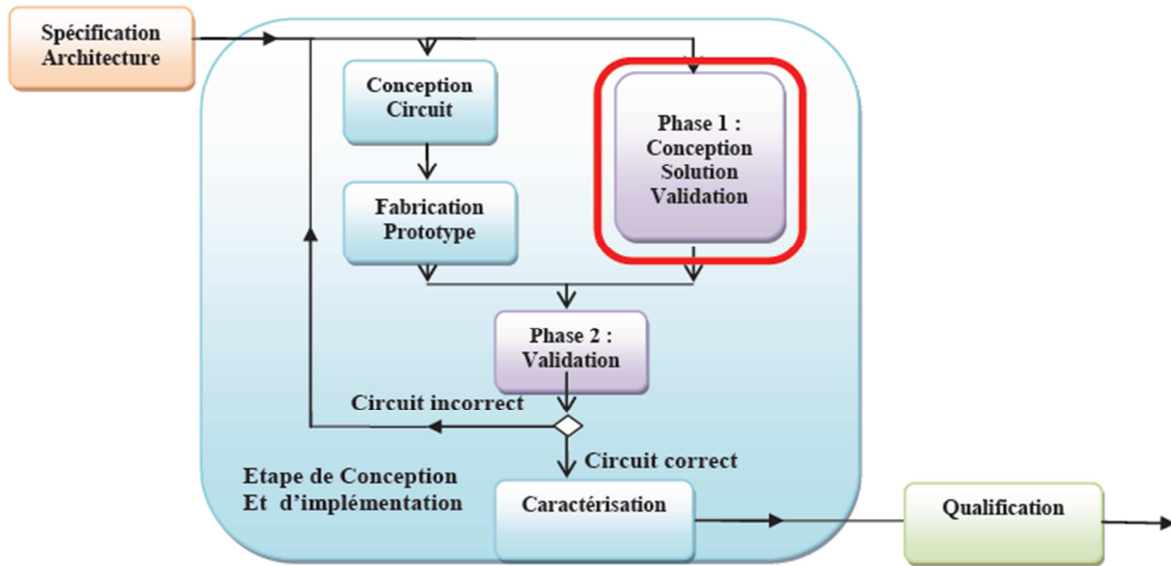


Figure IV-2 : l'organisation des activités

3- PCB :

Avant le câblage il faut qu'on crée une Bibliothèque des composants qui contient ces dimensions précis. Les dimensions de chaque Composant se trouvent à son guide (datasheet), On utilise la moyenne de chaque dimension pour qu'on trouve la valeur typique.

On prend comme exemple : Mémoire Flash

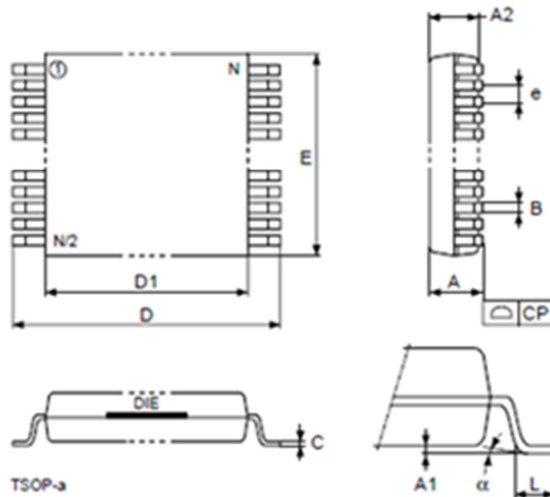


Figure IV-3 : schéma représente la forme du Package TSOP48 12*20mm du mémoire Flash M29W160EB [13].

Symbol	mm			inches		
	Typ	Min	Max	Typ	Min	Max
A			1.20			0.0472
A1		0.05	0.15		0.0020	0.0059
A2		0.95	1.05		0.0374	0.0413
B		0.17	0.27		0.0067	0.0106
C		0.10	0.21		0.0039	0.0083
D		19.80	20.20		0.7795	0.7953
D1		18.30	18.50		0.7205	0.7283
E		11.90	12.10		0.4685	0.4764
e	0.50	-	-	0.0197	-	-
L		0.50	0.70		0.0197	0.0279
<		0°	5°		0°	5°
N	48			48		
CP			0.10			0.0039

Tableau IV-1: Dimension du Package TSOP48 du
Mémoire Flash M29W160EB [13].

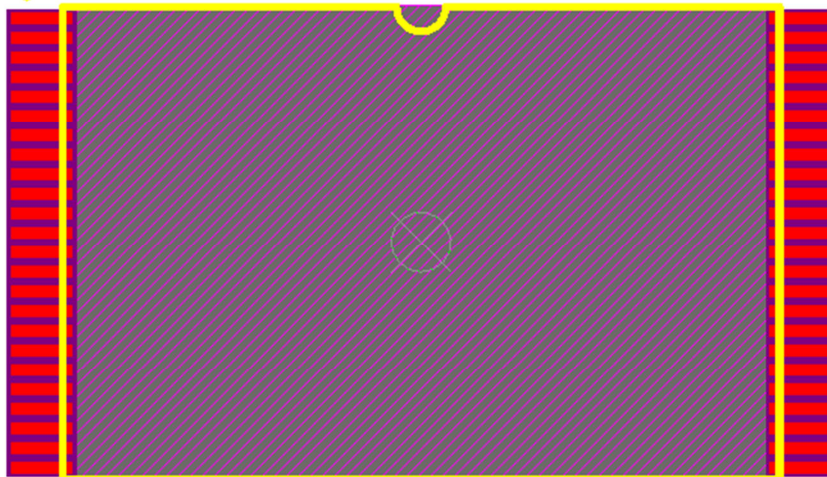


Figure IV-4 : PCB du M29W160EB (sur Altium)

4- Méthode de Câblage

Le but de cette étape est de schématiser le montage avec les composants externes et les connecteurs utilisés.

Le placement-routage : Cette étape est effectuée par un sous-traitant (ALTIUM DESIGNER). La complexité du projet demande une expertise importante quant à la prise en compte de la compatibilité électromagnétique entre les éléments de la carte.

Une bonne maîtrise du placement-routage permet de :

Une intégration de tous ces éléments sur une surface limitée pour diminuer l'effet du bruit.

Nombre minimale des trous et d'Eviter de mettre les trous au-dessus des Composants.

On compte pour FPGA «XC3S250E PQ208» 208 AVEC UN MAXIMUM DE I/O 158 Broches qui sont classé comme suite

Bank 0 : 30 I/O Broches

Bank 1 : 32 I/O Broches

Bank 2 : 31 I/O Broches

Bank 3: 33 I/O Broches

Le câblage la plus difficile était du SDRAM et Mémoire Flash

Pour SDRAM on compte 38 : 16 I/O et 22 inputs

Et mémoire Flash 42 : 16 I/O 25 inputs et 1 output

Les Broches de inputs et I/O des Composant doit être relié avec des broches I/O de FPGA

Les Broches d'output du Composant peut être relié avec des Broches I/O ou input de FPGA

Le nombre total = 80

Pour déminer le nombre des broches on a associé le même Bus d'adresse pour le mémoire Flash et SDRAM

$80 - 12 = 68$ $33 \text{ du Bank3} + 30 \text{ du Bank0} + 5 \text{ du Bank1} = 68$

D'après le câblage de tous les composants on résulte le schéma du Circuit imprimé

5- Règles D'alimentations :

Chaque composant supporte deux intervalles de tension

Domaine nominal d'emploi : conditions normales d'utilisations

Domaine de non-destruction : Les caractéristiques des composants ne sont pas modifiées mais il n'est pas opérationnel.

Domaine de destruction : dépassement du domaine de non-destruction.

Le composant risque d'être détruit.

pour que FPGA fonctionne normalement il a besoin comme alimentation du Vccint , Vccaux , Vcco il faut qu' il ne dépasse pas l'intervalle +-5% [7].

Symbole	Description	Minimum	Nominal	Maximum	Unité
Vccint	alimentation intérieure	1.140	1.2	1.260	V
Vccaux	alimentation auxiliaire	2.375	2.5	2.625	V
Vcco	Alimentation du driver de la sortie	1.1	-	3.465	V

Tableau IV-2: Les limites Opérationnel des alimentations FPGA

Symbole	Description	Minimum	Nominal	Maximum	Unité
Vccint	alimentation intérieure	-0.5	1.2	1.32	V
Vccaux	alimentation auxiliaire	-0.5	2.5	3.00	V
Vcco	Alimentation du driver de la sortie	-0.5	-	3.75	V

Tableau IV-3: Les maximums limites des alimentations FPGA avant la destruction

Et on a besoin aussi d'une alimentation du 5.0V±10% pour LCD et

Les Ports PS /2 : 5V

Pour avoir tous cela on a choisis les régulateurs du tentions suivant

Régulateur 5V [18]:Caractéristique :

$$7V \leq V_{IN} \leq 20V$$

$$1 \text{ mA} \leq I_O \leq 70 \text{ mA} \quad P \leq 0.75W.$$

Tolérance de voltage de sortie au plage de température 5V +-5% : 4.75 V et Maximum : 5.25 V

Limitation interne du courant de court-circuit

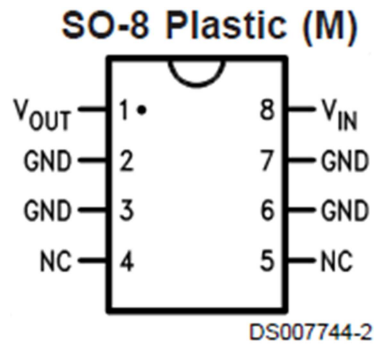


Figure IV-5 : schéma du connections du LM78L05 package SO-8

Régulateur 3V3 [19] : il support un courant maximum du 300 mA et inclut un limiteur de courant interne.

Régulateur 2V5 [20] : il support un courant maximum du 300 mA et inclut un limiteur de courant interne.

Régulateur 1V2 [21] : LM317 avec une résistance de charge qui représente l'impédance maximale de sortie 512 ohm génère 1.2V.

LM317 doit avoir

$1.5\text{mA} \leq I_{\text{OUT}} \leq 300\text{mA}$, $P \leq 625\text{mW}$ et $3\text{V} \leq (V_{\text{IN}} - V_{\text{OUT}}) \leq 13\text{V}$
pour que la tension de sortie soit $1.20 < 1.25 < 1.30\text{V}$

Pour que la tension de sortie soit 1.25V il faut : $I > 1.5\text{mA} \Rightarrow R < 1.2 / 0.0015$

$R < 800\ \Omega$

On a essayé avec une résistance de 510 Ω et on a trouvé une tension de sortie = 1.253 V

Qu'est dans les limites $1.14 < 1.253 < 1.26\text{V}$

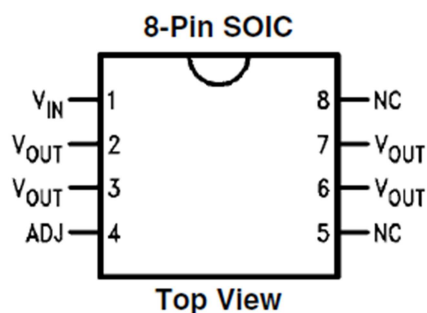
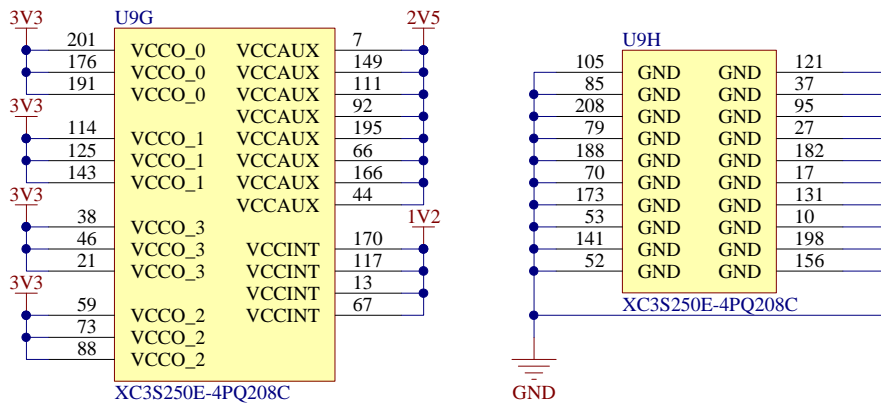


Figure IV-6 : schéma du connections du LM317 package SO-8

Tous les périphériques de la carte fonctionnent avec une alimentation de 3V3 sauf LCD et PS/2



FPGA Power Supply

Figure IV-7: Le schéma d'alimentation FPGA

6- Réalisation :

Comme premier étape ont a réalisé le programmeur JTAG

6-1- Programmeur JTAG :

On a vu que les cartes de FPGA peut être programmé par deux source usb ou JTAG.

Pour l'USB il nous faut qu'on ajoute un circuit FTDI alors on se trouve coincé par la solution JTAG, il existe plusieurs types pour les programmeurs JTAG

USB-JTAG ,RS232-JTAG ,PARRALLEL PORT-JTAG

Il faut aussi que le programmeur soit supporté par le programme qu'on utilise

On A Trouver un qui l'on peut réaliser

Universal JTAG Interface Rev 1.11

Il est auto détectable et supporté par :

- Xilinx sous le nom «Parallèle PORT III»
- Et par Altium Designer



Figure IV-8 : Surface 1 représente les Composant du JTAG

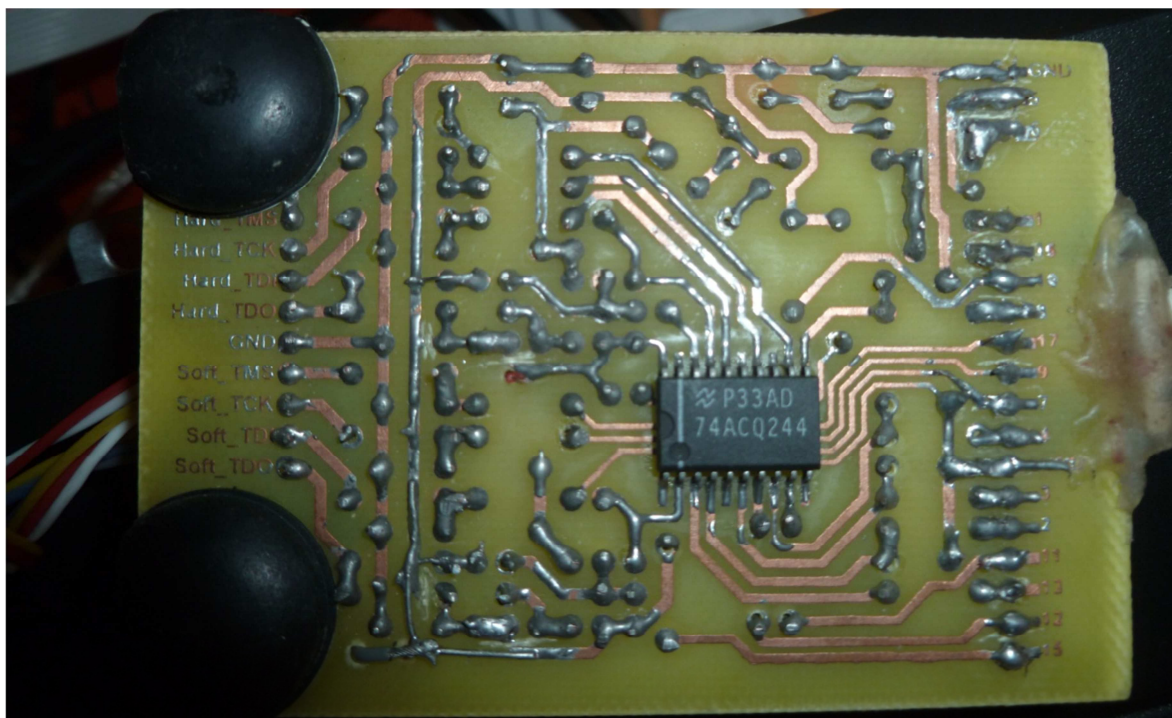


Figure IV-9 : Surface 2 représente le circuit imprimé du JTAG

Universal JTAG Interface Rev 1.11 supporte les FPGA du XILINX et Altera mais on a supprimé la partie de circuit consacré pour Altera .

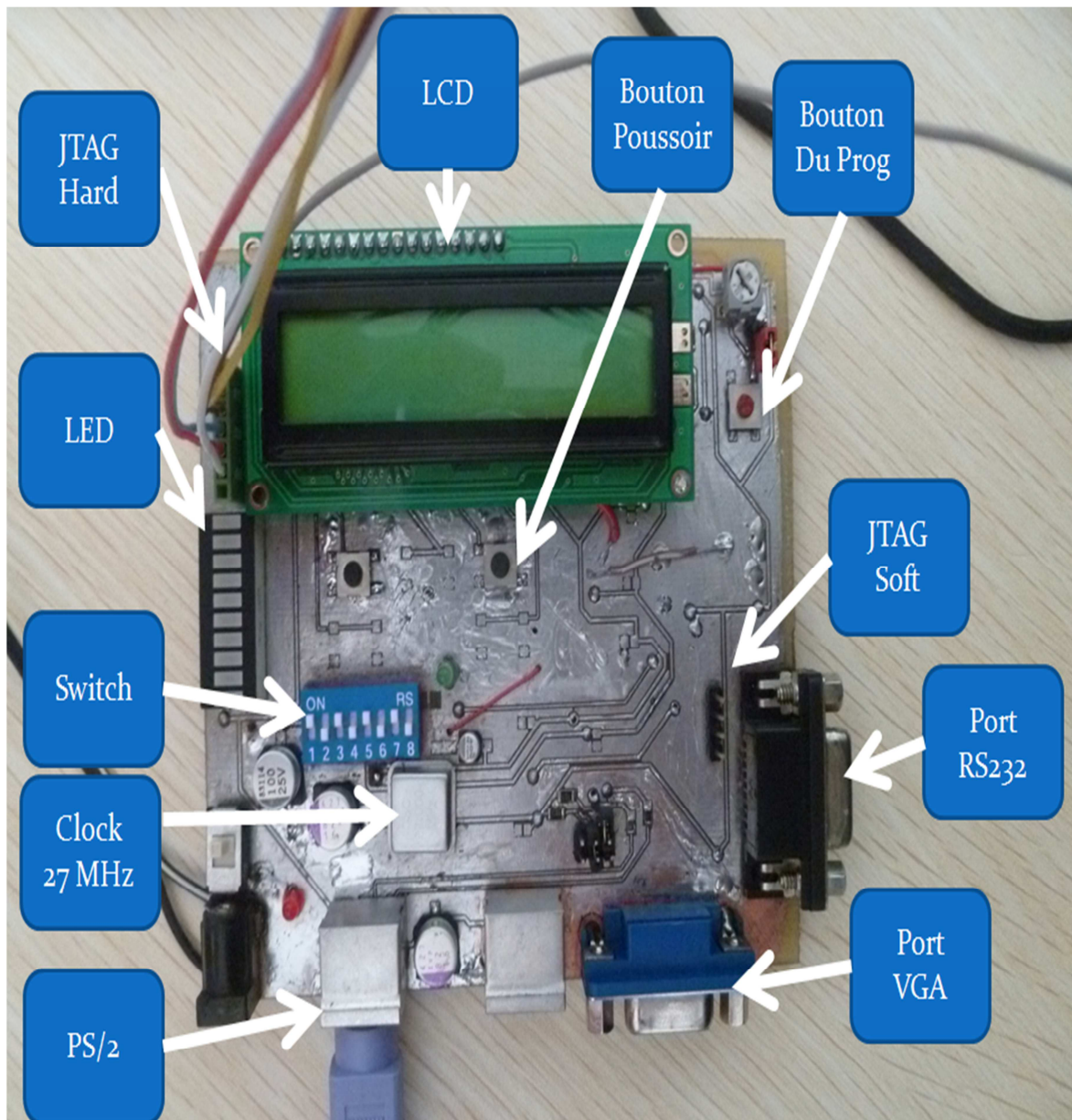


Figure IV-10: le schéma du Circuit imprimé après l'impression Pour la surface 1

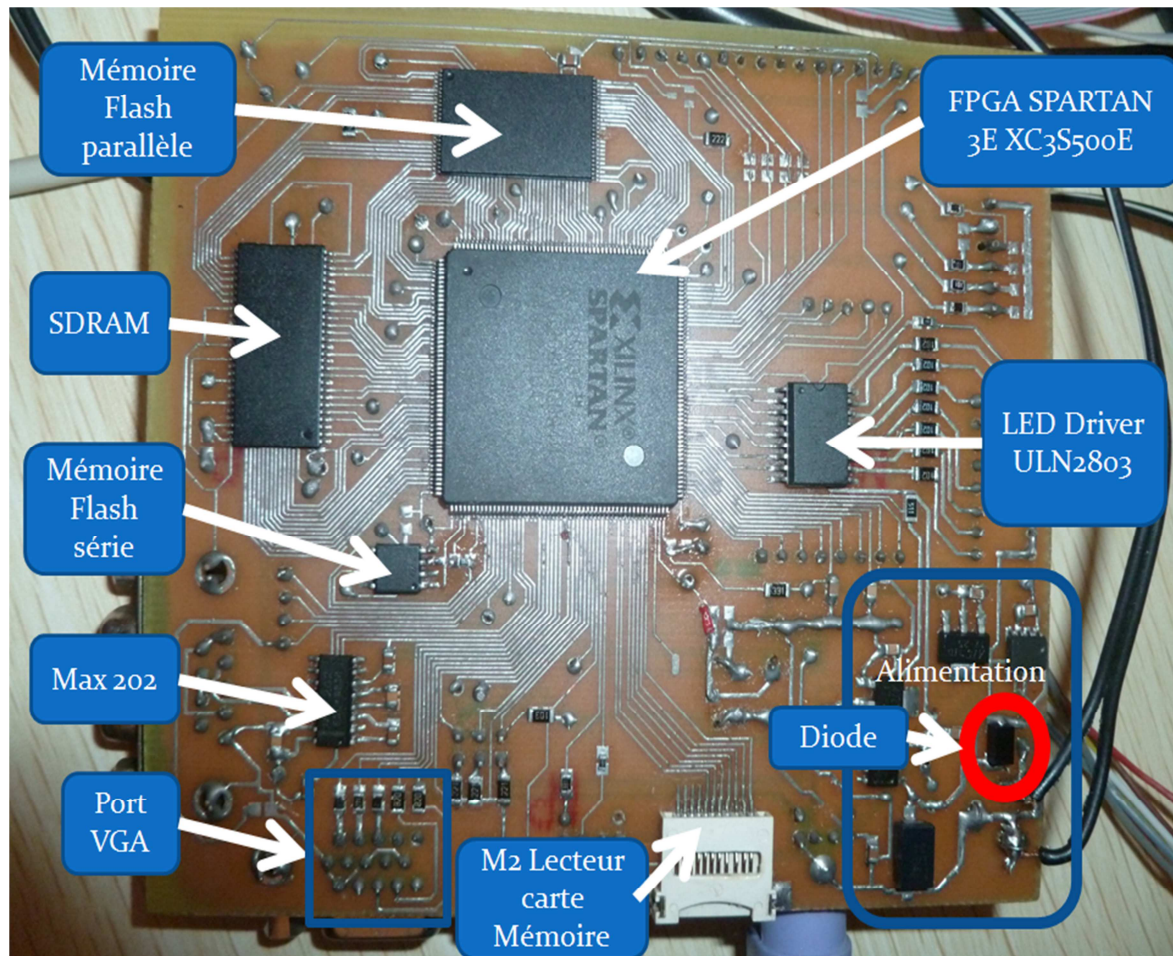


Figure IV-11: le schéma du Circuit imprimé après l'impression Pour la surface 2

On a trouver 3 Broches du Mémoir NOR Flash mall connecté

On a les coupé et branché avec les broche du M2

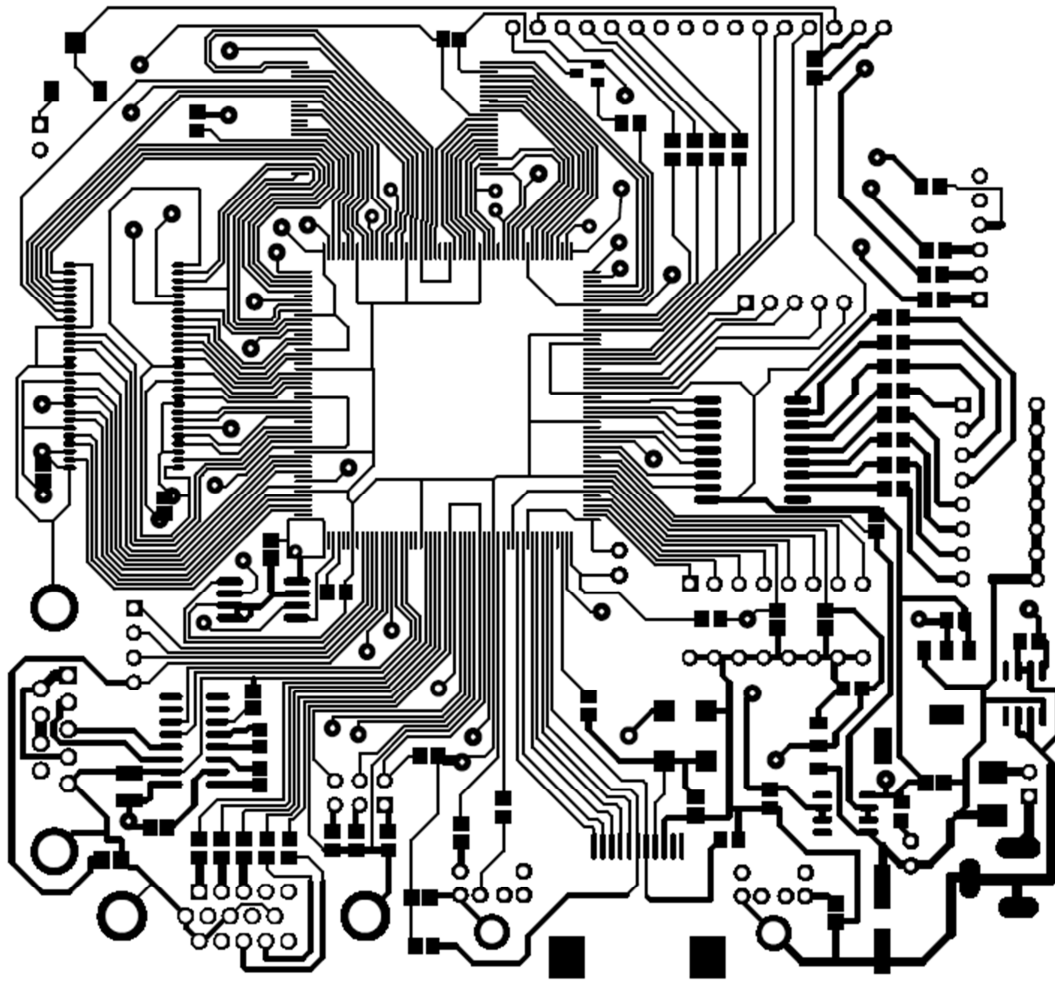


Figure IV-12: le schéma du Circuit imprimé avant l'impression Pour la surface 2

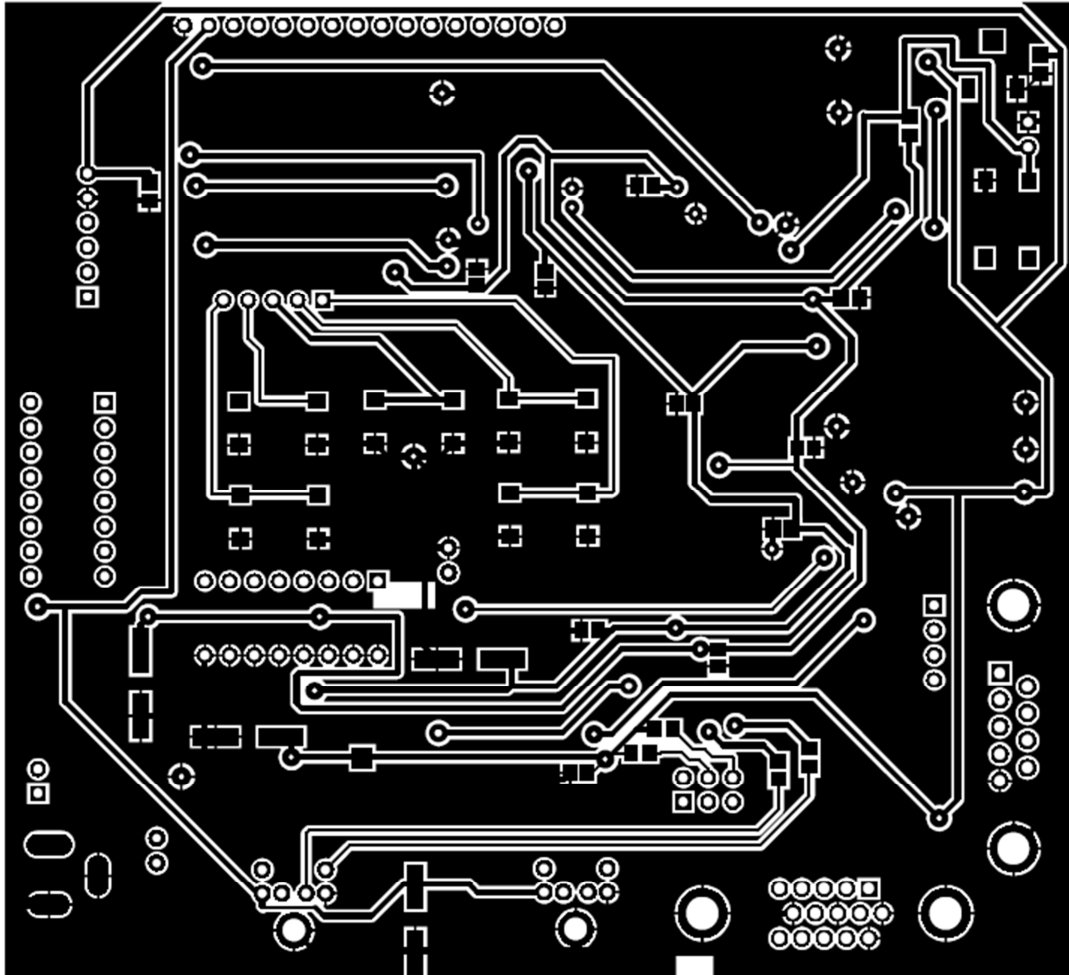


Figure IV-13: le schéma du Circuit imprimé avant l'impression Pour la surface 1

7- Bibliothèque FPGA :

Il Contient les ressources matérielles présentes sur la Carte FPGA avec les modules périphériques qui peuvent être inclus dans la conception FPGA pour faciliter l'interfaçage avec le port externe.

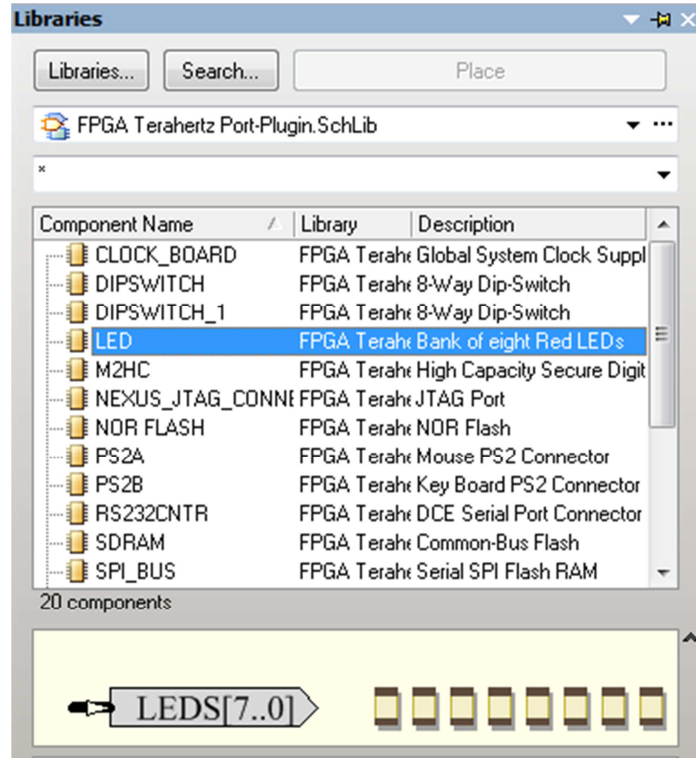


Figure IV-14 : Bibliothèque De la carte FPGA

8- Constraint File :

Il Contient les information du broches utilisé pour chaque variable ,sont mode , iostandard et la force du driver utilisé

Exemple :

```
; RS-232 Serial Connector
```

```
;
```

```
Record=Constraint | TargetKind=Port | TargetId=RS_TX | FPGA_PINNUM=P68
```

```
Record=Constraint | TargetKind=Port | TargetId=RS_RX | FPGA_PINNUM=P69
```

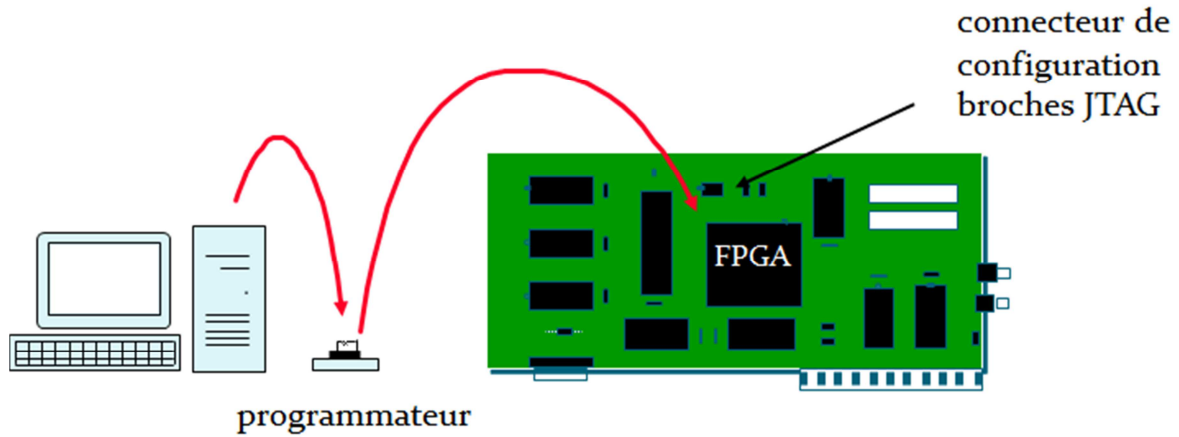



Figure II.16: JTAG Configuration Mode [11].

M [2:0] <= 0:0:1 SPI mode

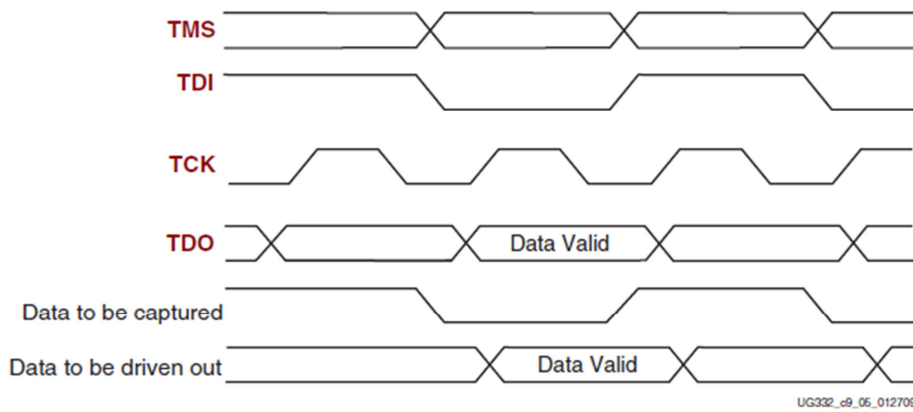
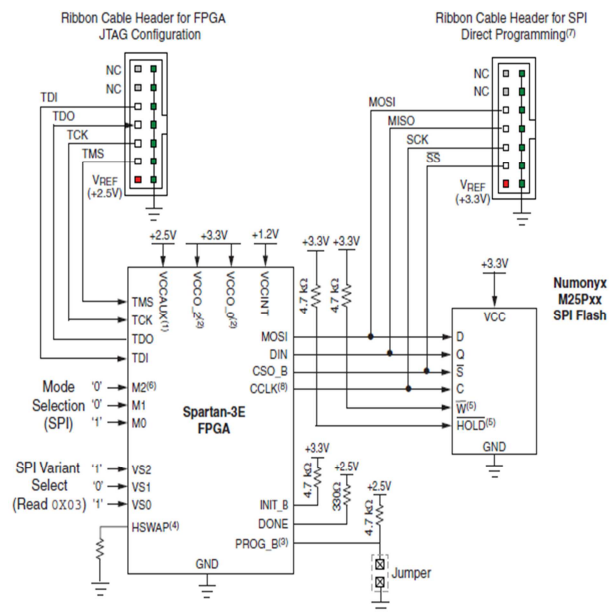


Figure II.17: JTAG signaux durant Configuration

Figure II.18: SPI Flash PROM Interface

Pour le mode READ [22]



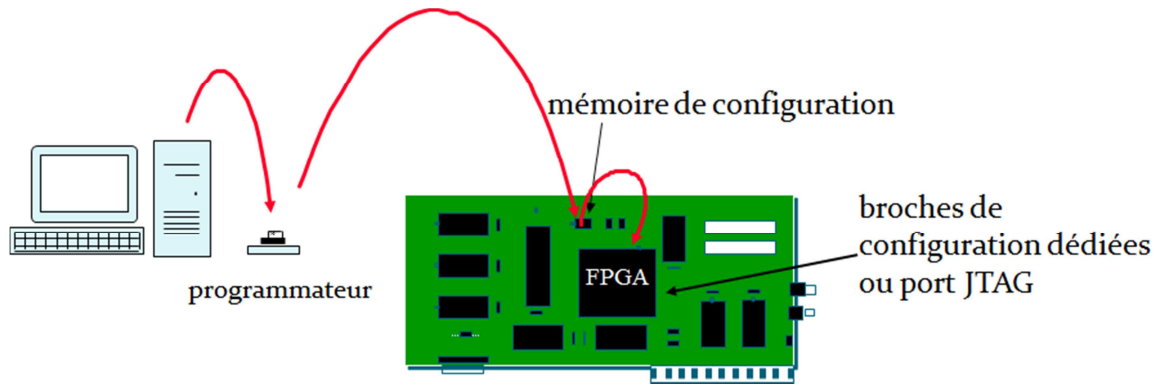


Figure II.19: SPI Configuration Mode [11].

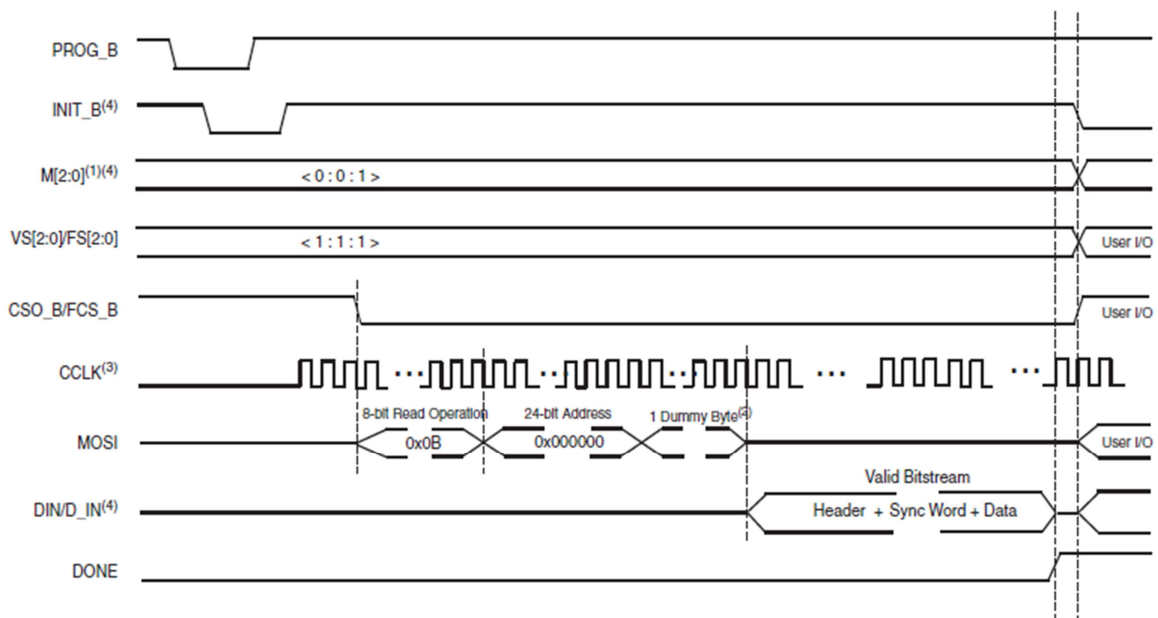


Figure II.20: SPI signaux durant Configuration Pour le mode FAST READ (0x0B) [22]

10- Program de test

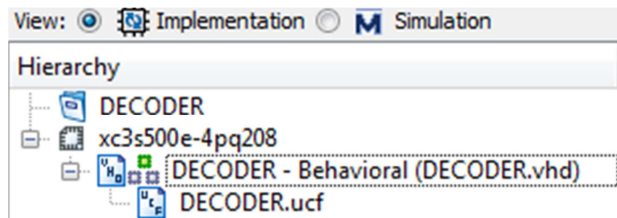
On ouvrir le dossier ISE Designe Tool ensuit Project Navigator puis on créer un nouvelle projet avec les paramètres suivant

Project Settings

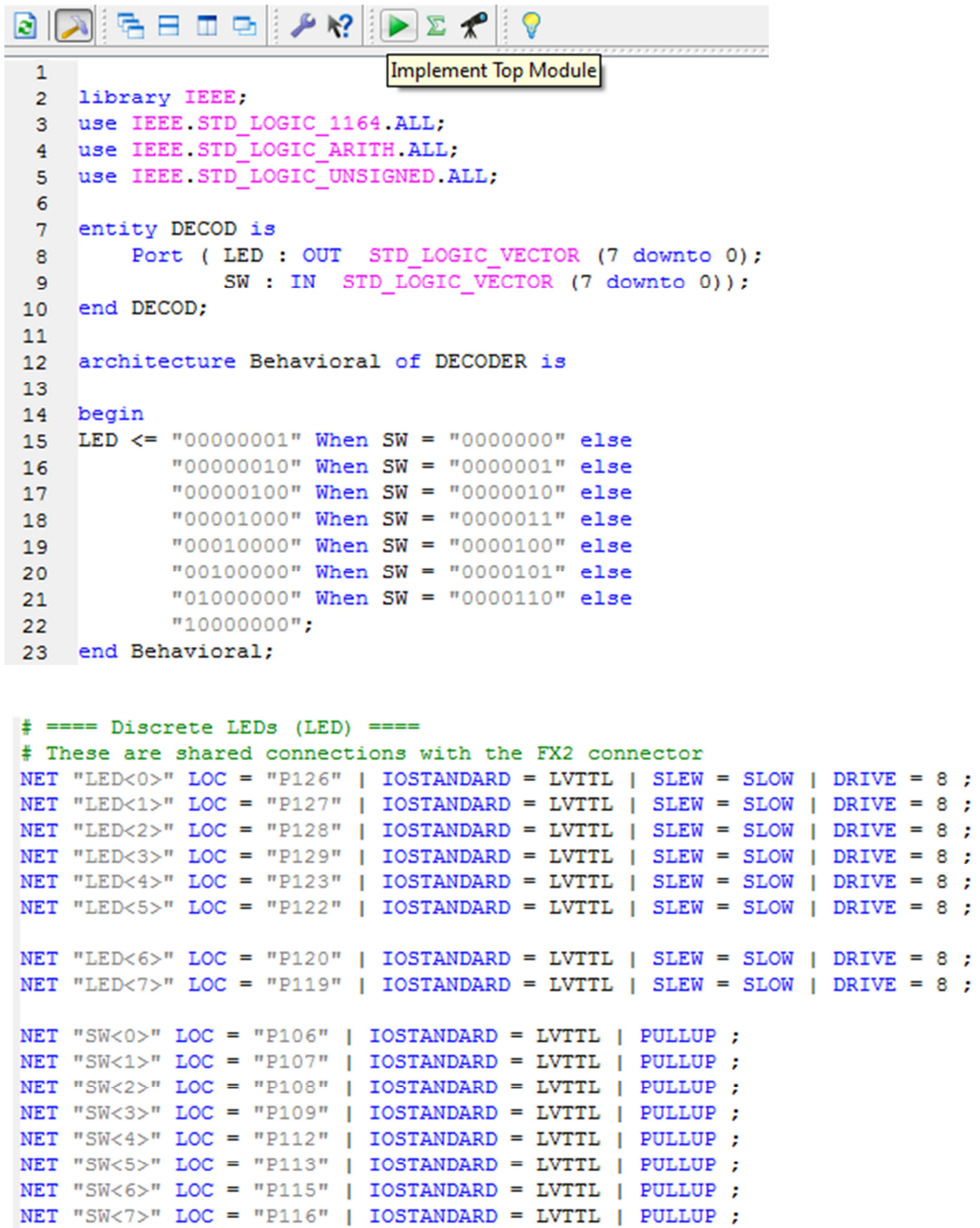
Specify device and project properties.
Select the device and design flow for the project

Property Name	Value
Product Category	All
Family	Spartan3E
Device	XC3S500E
Package	PQ208
Speed	-4
Top-Level Source Type	HDL
Synthesis Tool	XST (VHDL/Verilog)
Simulator	Modelsim-SE VHDL
Preferred Language	VHDL
Property Specification in Project File	Store all values
Manual Compile Order	<input type="checkbox"/>
VHDL Source Analysis Standard	VHDL-93
Enable Message Filtering	<input type="checkbox"/>

Puis on ajoute au projet « VHDL module » et « implementation Contrainst File »



Puis on écrit le programme VHDL
Et le contrainste File



```

1
2 library IEEE;
3 use IEEE.STD_LOGIC_1164.ALL;
4 use IEEE.STD_LOGIC_ARITH.ALL;
5 use IEEE.STD_LOGIC_UNSIGNED.ALL;
6
7 entity DECOD is
8     Port ( LED : OUT  STD_LOGIC_VECTOR (7 downto 0);
9           SW  : IN   STD_LOGIC_VECTOR (7 downto 0));
10 end DECOD;
11
12 architecture Behavioral of DECODER is
13
14 begin
15 LED <= "00000001" When SW = "00000000" else
16       "00000010" When SW = "00000001" else
17       "00000100" When SW = "00000010" else
18       "00001000" When SW = "00000011" else
19       "00010000" When SW = "00001000" else
20       "00100000" When SW = "00001001" else
21       "01000000" When SW = "00001100" else
22       "10000000";
23 end Behavioral;

```

```

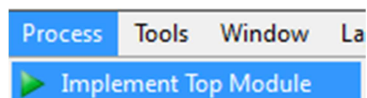
# ===== Discrete LEDs (LED) =====
# These are shared connections with the FX2 connector
NET "LED<0>" LOC = "P126" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
NET "LED<1>" LOC = "P127" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
NET "LED<2>" LOC = "P128" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
NET "LED<3>" LOC = "P129" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
NET "LED<4>" LOC = "P123" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
NET "LED<5>" LOC = "P122" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;

NET "LED<6>" LOC = "P120" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
NET "LED<7>" LOC = "P119" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;

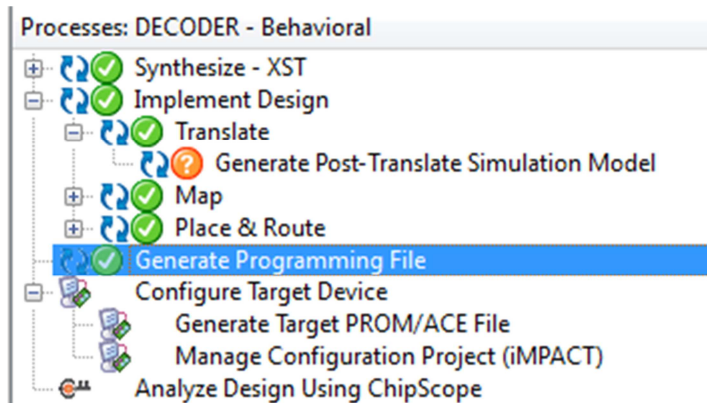
NET "SW<0>" LOC = "P106" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<1>" LOC = "P107" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<2>" LOC = "P108" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<3>" LOC = "P109" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<4>" LOC = "P112" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<5>" LOC = "P113" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<6>" LOC = "P115" | IOSTANDARD = LVTTTL | PULLUP ;
NET "SW<7>" LOC = "P116" | IOSTANDARD = LVTTTL | PULLUP ;

```

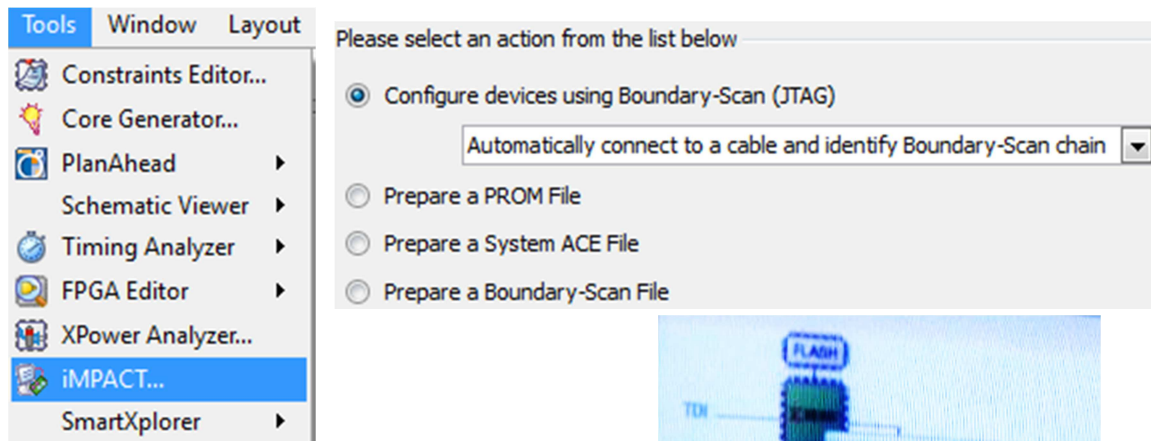
Ensuite on clique sur Process → implement Top Module



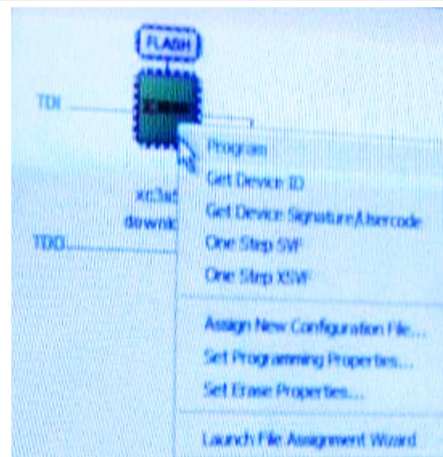
Ensuite on clique sur « Generate Programming File »



Puis on clique sur Tools → iMPACT on crée un nouvelle Project et on choisie Configure device using Boundary Scan



On charge FPGA par le Fichier bit puis on clique Sur Program, Et on attend jusqu' à l'opération Terminé, la led vers s'allume.



Conclusion :

Dans ce chapitre, nous avons présenté les étapes de la conception et réalisation d'un projet.

Le travail difficile était de dessiné les PCB à cause de la précision demandé

Il faut que les dimensions soit exacte, une erreur de 0.1 m peut fauter le PCB du Composant, et même pour le câblage le nombre énormes des broches a câblé

Conclusion général :

Ce projet s'inscrit dans le flot de conception d'une Carte FPGA dont le but est de fournir des outils de validation pour les étudiants. L'approche traditionnelle consiste à proposer une solution permettant la vérification d'un prototype une fois fabriqué. Cependant cette technique présente la nécessité de relancer la fabrication d'un prototype à chaque modification du schéma du circuit.

Le FPGA permet l'émulation de montages numériques. Sa reprogrammation instantanée offre la possibilité de régler en « temps réel » les paramètres afin d'obtenir un schéma fonctionnel avant leur implémentation en circuit final.

La réalisation de cette solution a nécessité d'atteindre différents objectifs :

- Le premier était de maîtriser les outils de conceptions du FPGA et de connaître l'architecture de ce type de circuit. L'étude de ses caractéristiques a montré que la solution actuelle ne pouvait pas suivre l'augmentation de la complexité des circuits à émuler. Le choix d'un FPGA à haute intégration a été inévitable.
- Le deuxième objectif a été d'intégrer ce nouveau FPGA dans l'environnement de test actuel.

Ce qui a conduit à la création d'un nouveau programme de gestion du processeur interne.

Une compréhension approfondie de l'environnement de test et l'étude de la structure du programme FPGA étaient nécessaires pour l'avancement de ce projet.

Dans ce travail, on a associés le bus d'adresse de la mémoire SDRAM avec le bus d'adresse de la mémoire flash mais pour le futur plusieurs broches des Composants peuvent être associés pour avoir la Possibilité d'intégrer Des autres Composant et connecteurs.

Liste Des Figures Et Des Tableaux

Liste Des Figures

Figure I.1: Structure d'un circuit FPGA	04
Figure I.2: Exemple de blocs logiques de différents fabricants	05
Figure I.3: Input Output Block (IOB)	05
Figure I.4: Types d'interconnexion entre deux CLB d'interconnexion adjacente	06
Figure I.5: Connexions à usage général et détail d'une matrice de commutation	07
Figure I.6: Les interconnexions directes	08
Figure I.7: Les longues lignes	08
Figure I.8: Technologie SRAM	10
Figure I.9: Technologie anti-Fuse	11
Figure I.10: Technologie EPROM/EEPROM	11
Figure I.11: Comparaison de l'entrée et de sortie I/O CMOS et LVTTTL	13
Figure II.1: Boîtier QFP Spartan-3E Marquage Exemple	18
Figure II.2: Spartan-3E package BGA Marquage Exemple	19
Figure II.3: Spartan-3E CP132 et CPG132 marquage de l'emballage Exemple	19
Figure II.4: Architecture d'une SPARTAN 3E	20
Figure II.5 : Architecture d'un CLBs pour une SPARTAN 3E [21
Figure II.6 : Disposition des CLBs dans un FPGA SPARTAN 3E	21
Figure II.7: Architecture d'une slice	22
Figure II.8 : Disposition des IOBs	23
Figure II.9 : Schéma simplifié IOB	23
Figure II.10: Principe des ports et fonctions des blocs multiplicateurs	26
Figure II.11: MULT18X18SIO Primitive	27
Figure II.12: Quatre configurations de l'entrée B	27
Figure II.13: Vue d'ensemble de connexions d'horloge	28
Figure II.14: Spartan-3E Structure horloge interne	29
Figure II.15: Gardien de Circuit	30
Figure II.16: JTAG Configuration Mode	37
Figure II.17: SPI Flash PROM Interface pour PROM Support READ (0x03) et FAST_READ (0x0B) Commands	38
Figure II.18: Mode maître série en utilisant la plate-forme Flash PROM	39
Figure II.19: Mode de l'interface périphérique (BPI)	41
Figure II.20: Mode de configuration esclave parallèle	42
Figure II.21: Slave Serial Configuration	43
Figure III.1 : Schéma synoptique de FPGA avec les périphériques	
Figure III.2 : Interface de Spartan-3E FPGA avec LCD	45
Figure III.3 : Schéma du Câblage du Port VGA avec FPGA Spartan-3E	47
Figure III.4 : synchronisation d'affichage [12].	49
Figure III.5 : Schéma du Câblage du Portes série RS-232	51
Figure III.6 : Schéma du Câblage du connecteurs PS/2	52

Figure III.7 : Formes d'onde de synchronisation de bus PS/2	53
Figure III.8 : Codes de balayage du clavier PS/2	54
Figure III.9 : Mouse Transaction	55
Figure III.10 : système coordinatrice relatif de La souris	55
Figure III.11 : Connexions à mémoire flash parallèle	56
Figure III.12 : Schéma du Câblage du Mémoire Flash Parallèle	57
Figure III.13 : Schéma du Câblage du Mémoire Flash série SPI	58
Figure III.14 : Schéma du Câblage du SDRAM	60
Figure III.15 : Schéma du Câblage du LEDs	61
Figure III.16 : Schéma du Câblage du Switchs et PUSH Buttons	61
Figure III.17 : Schéma du Câblage du M2	62
Figure III.18 : Schéma du Câblage du M2	62
Figure IV-1 : le flot de conception d'un projet	65
Figure IV-2 : l'organisation des activités	67
Figure IV-3 : schéma représente la forme du Package TSOP48 12*20mm du mémoire Flash M29W160EB	67
Figure IV-4 : PCB du M29W160EB (sur Altium)	68
Figure IV-5 : schéma du connections du package SO-8	71
Figure IV-6 : schéma du connections du LM317 package SO-8	
Figure IV-7: Le schéma d'alimentation FPGA	72
Figure IV-8 : Surface 1 représente les Composant du JTAG	73
Figure IV-9 : Surface 2 représente le circuit imprimé du JTAG	73
Figure IV-10: le schéma du Circuit imprimé après l'impression Pour la surface 1	74
Figure IV-11: le schéma du Circuit imprimé après l'impression Pour la surface 2	75
Figure IV-12: le schéma du Circuit imprimé avant l'impression Pour la surface 2	76
Figure IV-13: le schéma du Circuit imprimé avant l'impression Pour la surface 1	77
Figure IV-14 : Bibliothèque De la carte FPGA	78
Figure II.15: JTAG Mode Interface	79
Figure II.16: JTAG Configuration Mode	80
Figure II.17: JTAG signaux durant Configuration	80
Figure II.18: SPI Flash PROM Interface Pour le mode READ	80
Figure II.19: SPI Configuration Mode	81
Figure II.20: SPI signaux durant Configuration Pour le mode FAST READ (0x0B)	81

Liste Des Tableaux

Table I.1: Résumé des différentes technologies	12
Tableau I.2: les besoins d'alimentation pour chaque type d'FPGA	12
Tableau II.1 : Tableau récapitulative des différentes familles de XILINX	17
Tableau II.2 : Broche IO STANDARD Driver Compatibilité	24
Tableau II.3: différentiel compatibilité de la Banque IOSTANDARD	25
Tableau II.4: différentiel compatibilité de la Banque IOSTANDARD	25
Tableau II.5: Ram block Pour chaque FPGA SPARTAN 3	30
Tableau II.6: Valeur du Résistance de Pull-up	30
Tableau II.7: Courant de sortie Programmable	31
Tableau II.8: les différents Modes de Configuration	34
Tableau II.9: Comportement HSWAP	35
Tableau II.10: pull-up ou pull-down pour les valeurs HSWAP, M [2:0], et VS [2:0]	35
Tableau II.11: Codes Variant Select pour Divers SPI Serial PROM flash	36
Tableau II.12: Spartan-3E Identificateurs de périphériques JTAG	37
Tableau II.13: Adressage BPI contrôle	40
Tableau II.14: Les alimentations du FPGA SPARTAN 3E	44
Tableau III.1 : codes couleurs de l'affichage 3-Bit	48
Tableau III.2 : PS/2 connecteur Brochages	52
Tableau III.3 : Synchronisation de bus PS/2	53
Tableau III.4 : Commandes de clavier PS/2 communes	54
Tableau III.5 : Signaux d'interface instantané de SPI	58
Tableau IV-1: Dimension du Package TSOP48 du Mémoire Flash M29W160EB	68
Tableau IV-2: Les limites Opérationnel des alimentations FPGA	70
Tableau IV-3: Les maximums limites des alimentations FPGA avant la destruction	70

BIBLIOGRAPHIE

- [1] Haithem KHALED « ADAPTATION DE L'APPROCHE DE TEST CDIDDQ AUX CIRCUITS PROGRAMMABLES FPGA » MONTRÉAL, Québec, Canada, NOVEMBRE 2011.
- [2] CHIBAH AREZKI, « CONCEPTION D'UN CONTROLEUR D'ETAGE DE PUISSANCE PAR FPGA». UNIVERSITE MOULOU D MAMMERI DE TIZI OUZOU, Magister en Electrotechnique Option: Entraînements Electriques,
- [3] Johanna MARIANI « Programmation et Utilisation du FPGA Pour la validation et la vérification De circuits électroniques» RHÔNE-ALPES, France, Février 2011 école d'ingénieur 'CNAM ' Département d' Electronique.
- [4] BOUAMRA Sid-Ahmed, «IMPLÉMENTATION D'UN MODULE DE RECEPTEUR NUMÉRIQUE SUR UN CIRCUIT REPROGRAMMABLE DE TYPE FPGA » Université Dr Yahia Fares de Médéa, 2008/2009 Département d'Electronique Instrumentation,
- [5] Syed Zahid AHMED « eFPGAs: Architectural Explorations, System Integration & a Visionary Industrial Survey of Programmable Technologies » France Université Montpellier 2, École Doctorale Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier.
- [6] TAVENIER, CHRISTIAN, « Circuit logique programmable » Editions DUNOD, 1996.
- [7] Murata « FPGA Power Guide » 2008 Murata Power Solutions.
http://www.mouser.com/catalog/supplier/library/pdf/Murata_FPGA.pdf
- [8] http://www.interfacebus.com/voltage_LV_threshold.html
- [9] XILINX « Spartan-3E FPGA Family » Data Sheet, DS312 (v3.8) August 26, 2009 .
http://www.xilinx.com/support/documentation/data_sheets/ds312.pdf
- [10] XILINX «Spartan-3 Generation FPGA User Guide Extended Spartan-3A, Spartan-3E, and Spartan-3 FPGA Families » UG331 (v1.6) December 3, 2009.
http://www.xilinx.com/support/documentation/user_guides/ug331.pdf
- [11] XILINX «Spartan-3 Generation Configuration User Guide» UG332 (v1.6) October 26, 2009.
http://www.xilinx.com/support/documentation/user_guides/ug332.pdf

- [12] XILINX « Spartan-3E Starter Kit Board User Guide» UG230 (v1.0) March 9, 2006
www.xilinx.com
- [13] M29W160ET-M29W160EB « 16 Mbit (2Mb x8 or 1Mb x16, Boot Block)
3V Supply Flash Memory » STMicroelectronics
<http://www.datasheetcatalog.org/datasheet/stmicroelectronics/9195.pdf>
- [14] M25P16 « 16 Mbit, Low Voltage, Serial Flash Memory With 50 MHz SPI Bus
Interface» STMicroelectronics
<http://www.datasheetcatalog.org/datasheet2/b/0f5y226fwq7kc7g0eg31yofupcpy.pdf>
- [15] K4S281632F « SDRAM 128Mb F-die (x4, x8, x16) » Rev. 1.2 May 2004
SAMSUNG ELECTRONICS
<http://pdf1.alldatasheet.com/datasheet-pdf/view/37061/SAMSUNG/K4S281632F-TC60/+25173UOY1SuOpdxvZvGvGEESMgSLh+/datasheet.pdf>
- [16] ULN2803AFW « 8CH DARLINGTON SINK DRIVER » 2001-07-05
Manufactured by Toshiba Malaysia
<http://www.datasheetcatalog.org/datasheet/toshiba/4015.pdf>
- [17] M2 «MicroMS Extender for MS & MS Duo Schematic » Rev. 1.0 2008/07/07 V1.0
http://www.bplustech.com/pdf/m2ex_schematic.pdf
- [18] LM78L05 « LM78LXX Series Terminal Positive Regulators»2000 National
Semiconductor.
- [19] atc 3hg 17-33 « Terminal Adjustable Regulator » National Semi-conducteur
- [20] atc 3id 17-25 « Terminal Adjustable Regulator » National Semi-conducteur
- [21] LM317L « Terminal Adjustable Regulator » September 14, 2010
National Semi-conducteur
<http://www.ti.com/lit/ds/symlink/lm317l-n.pdf>
- [22] Stephanie Tapp « Configuring Xilinx FPGAs with SPI Serial Flash » XAPP951 (v1.3)
September 23, 2010, XILINX
http://www.xilinx.com/support/documentation/application_notes/xapp951.pdf

Résumé :

Ce travail Consiste a réalisé Une Carte électronique autour du circuit FPGA SPARTAN 3 E de XILINX qui permet de valider les travaux qui rentrant dans le Cadre des TPs du module Logique Programmable du parcours Master .

Abstract:

This work Consists carried out an Electronic board around the FPGA circuit SPARTAN 3E of XILINX which makes it possible to validate work which returning in Framework of TPs of the Programmable Logical module of the course Master degree.

ملخص :

يرتكز هذا العمل على إنجاز لوحة إلكترونية حول " FPGA SPARTAN 3 E XILINX " التي تمكننا من تجريب الأعمال التطبيقية الخاصة بمادة " Logique Programmable " الخاص بالمستوى ماستر