

Chapitre 3

Les réseaux logiques programmables FPGA

3.1 Classification des circuits numériques

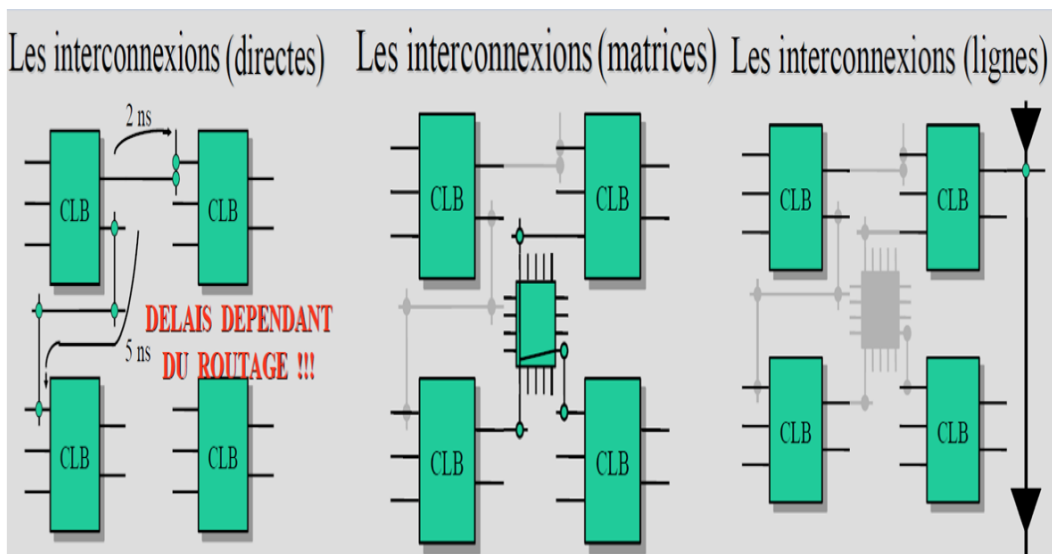


FIGURE 3.1 – Classification des circuits numériques

3.2 Les types d'architectures des FPGA

Les FPGA sont assimilables à des ASIC programmables par l'utilisateur.

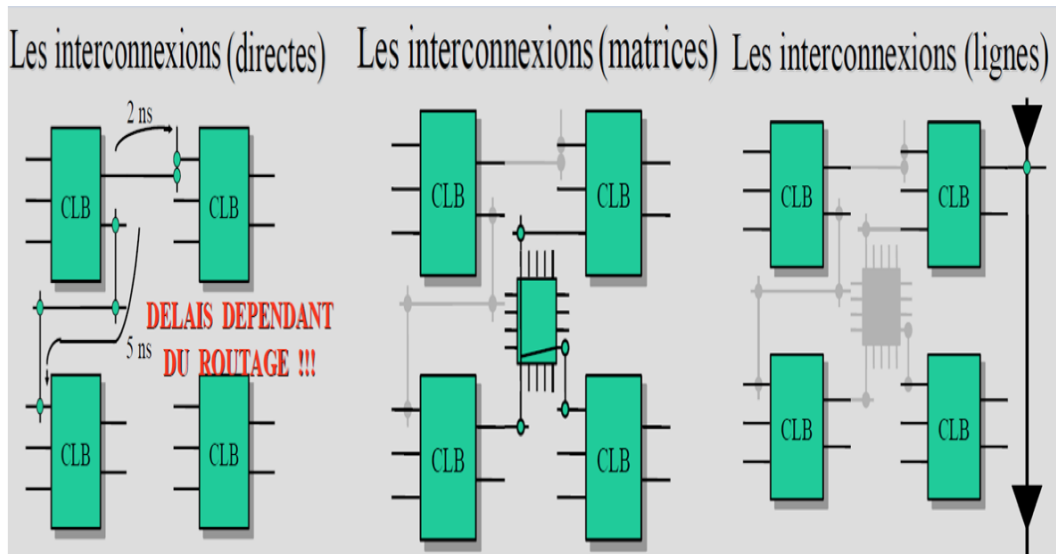


FIGURE 3.2 – Interconnexions entre les blocs logiques

3.2.1 Architecture de type îlots de calcul

: L'architecture de type "îlots de calcul" est l'un des types d'architectures que l'on peut trouver dans les FPGA (Field-Programmable Gate Arrays). Les FPGA sont des dispositifs électroniques programmables qui permettent aux concepteurs de circuits électroniques de mettre en œuvre des fonctions logiques personnalisées en programmant l'interconnexion des blocs logiques et des éléments de traitement du FPGA.

L'architecture en îlots de calcul, parfois appelée "architecture basée sur des grilles", est l'une des approches d'organisation interne des FPGA. Elle se caractérise par la division du FPGA en une série d'îlots ou de blocs de calcul, chaque îlot étant capable de réaliser des opérations logiques et arithmétiques. Ces îlots sont généralement interconnectés via un réseau d'interconnexion qui permet aux données de circuler entre eux.

Voici quelques caractéristiques clés de l'architecture en îlots de calcul dans les FPGA :

1. Isolation des îlots : Chaque îlot de calcul est relativement autonome et peut être configuré pour effectuer des opérations spécifiques. Cela permet une parallélisation efficace des calculs et une utilisation optimale des ressources du FPGA.
2. Réseau d'interconnexion : Les îlots de calcul sont reliés par un réseau d'interconnexion qui permet de faire circuler les données entre eux. La conception de ce réseau d'interconnexion est essentielle pour garantir des performances optimales du FPGA.
3. Flexibilité : L'architecture en îlots de calcul permet une grande flexibilité dans la

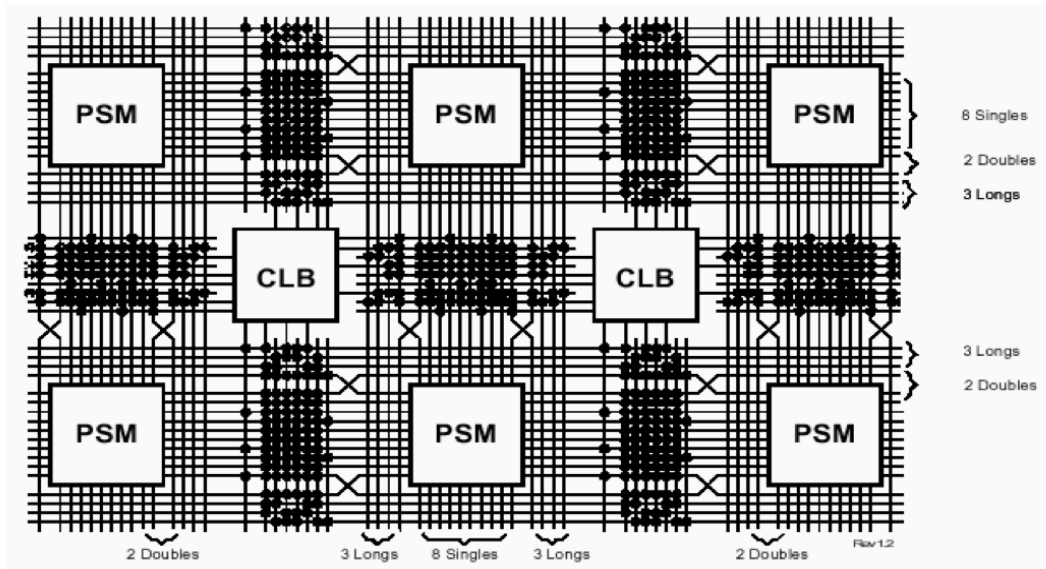


FIGURE 3.3 – Routage

conception des circuits électroniques. Les concepteurs peuvent organiser les îlots en fonction des besoins de leur application.

4. **Traitement parallèle** : Cette architecture est bien adaptée aux applications nécessitant un traitement parallèle intensif, telles que le traitement du signal, la vision par ordinateur et l'apprentissage automatique.

5. **Efficacité énergétique** : En raison de la capacité à activer uniquement les îlots nécessaires pour une tâche donnée, cette architecture peut être plus économe en énergie que d'autres architectures FPGA.

Il est important de noter que les FPGA peuvent adopter différentes architectures en fonction de leurs fabricants et de leurs modèles. Les architectures en îlots de calcul ne sont qu'une des nombreuses approches possibles pour organiser les ressources internes d'un FPGA. D'autres architectures courantes incluent les architectures basées sur des matrices de portes logiques (LUT), les architectures hybrides, etc. Le choix de l'architecture dépendra des exigences spécifiques de votre application.

3.2.2 Architecture de type hiérarchique

: L'architecture de type hiérarchique est une autre approche courante pour organiser les ressources internes des FPGA (Field-Programmable Gate Arrays). Cette architecture

est souvent utilisée pour simplifier la conception et la gestion des circuits programmés dans un FPGA, en particulier pour des projets complexes. Voici comment fonctionne l'architecture de type hiérarchique :

1. Hiérarchie des modules : Dans une architecture de type hiérarchique, la conception du circuit est divisée en différents modules ou blocs fonctionnels, qui sont hiérarchiquement organisés. Chaque module peut être conçu de manière indépendante pour effectuer une fonction spécifique. Ces modules peuvent ensuite être combinés pour créer un circuit électronique plus complexe.
2. Réutilisation des conceptions : Cette approche permet de réutiliser des modules conçus précédemment. Par exemple, si vous avez conçu un module de traitement d'image pour une application, vous pouvez le réutiliser dans une autre application sans avoir à tout refaire. Cela peut considérablement accélérer le processus de conception.
3. Facilité de gestion : La hiérarchie des modules facilite la gestion de projets FPGA complexes. Chaque module peut être conçu, testé et validé séparément, ce qui simplifie le débogage et la maintenance.
4. Optimisation : Chaque module peut être optimisé indépendamment en fonction de ses besoins spécifiques, ce qui peut améliorer l'efficacité de la conception globale.
5. Simplicité de la conception : Pour les concepteurs, cette approche permet de se concentrer sur des parties spécifiques du circuit, ce qui peut rendre la conception plus gérable.
6. Découpage logique : L'architecture hiérarchique découpe logiquement le FPGA en différentes zones ou niveaux, ce qui facilite la gestion des ressources et de l'interconnexion entre les modules.

L'architecture hiérarchique est particulièrement utile pour les projets FPGA de grande envergure, tels que les systèmes embarqués complexes, les processeurs personnalisés et les applications de traitement de données intensives. Cela permet de simplifier la conception, de faciliter la maintenance et de réduire le temps de développement.

En résumé, l'architecture de type hiérarchique dans les FPGA organise la conception en modules hiérarchiques indépendants, ce qui simplifie la gestion et la conception de projets complexes. Elle favorise la réutilisation de conceptions et l'optimisation des modules individuels.

Les FPGA (Field-Programmable Gate Arrays) sont des dispositifs programmables qui offrent une grande flexibilité pour la conception de circuits électroniques personnalisés.

Voici un aperçu des différents éléments d'un FPGA, des caractéristiques des FPGA modernes et des domaines d'application, ainsi que des critères de choix associés :

3.3 Les différents éléments des FPGA :

1. Circuit Configurable : Le cœur d'un FPGA est le circuit configurable, qui se compose de blocs logiques configurables (CLB) interconnectés. Les CLBs sont des matrices de portes logiques programmables.
2. Blocs Logiques (CLB) : Les blocs logiques sont composés de portes logiques programmables (généralement des LUTs, ou "Look-Up Tables") et de bascules. Les CLBs permettent de mettre en œuvre la logique combinatoire et les bascules de stocker des états ou des valeurs.
3. Blocs d'Entrée/Sortie (IOB) : Les blocs d'entrée/sortie sont responsables de l'interface entre le FPGA et le monde extérieur. Ils permettent de connecter des signaux d'entrée et de sortie à des broches du FPGA.
4. Interconnexions Programmables : Les interconnexions programmables permettent de relier les CLBs, les IOBs et d'autres ressources du FPGA. La configuration de ces interconnexions détermine comment les signaux circulent dans le FPGA.
5. Gestionnaire d'Horloge : Les FPGA modernes intègrent souvent des gestionnaires d'horloge qui facilitent la distribution de signaux d'horloge à travers le FPGA, ce qui est essentiel pour synchroniser les opérations.
6. Réseau Mémoire SRAM : Les FPGA disposent d'une mémoire SRAM (Static Random-Access Memory) qui permet de stocker la configuration du FPGA. Cette mémoire peut être reprogrammée pour modifier le comportement du FPGA.

3.4 Caractéristiques des FPGA modernes

1. Blocs de Petits Multiplieurs : Les FPGA actuels incluent souvent des blocs spécialisés pour les opérations de multiplication, ce qui les rend adaptés à des applications telles que le traitement du signal et l'apprentissage automatique.
2. Blocs DSP : Les FPGA comportent des blocs de traitement numérique du signal (DSP) qui sont optimisés pour les opérations mathématiques et de traitement du signal.

3. Blocs de Cœurs de Processeurs : Certains FPGA intègrent des cœurs de processeurs personnalisables, ce qui permet d'exécuter des tâches de calcul intensif.

3.5 Critères de Choix :

Les critères de choix pour un FPGA dépendent de l'application spécifique, mais ils peuvent inclure :

1. Puissance de Calcul : La capacité de traitement, y compris les blocs DSP, les multiplieurs, et les cœurs de processeurs.

Consommation d'Énergie : La consommation électrique du FPGA doit être compatible avec l'application.

Nombre d'Entrées/Sorties : Le nombre d'IOBs disponibles pour les connexions est crucial.

Taille et Complexité du Circuit : Certains projets nécessitent des FPGA plus grands ou plus complexes.

Coût : Le budget joue un rôle important dans le choix du FPGA. Caractéristiques de l'Horloge : La gestion de l'horloge est importante pour la synchronisation.

3.6 Domaines d'Applications :

Les FPGA trouvent des applications dans de nombreux domaines, notamment :

Télécommunications et Réseaux

Traitement du Signal et Audio/Vidéo

Aérospatiale et Défense

Automatisation Industrielle

Calcul Haute Performance (HPC) et Accélération Matérielle

Informatique Embarquée

Systèmes de Vision par Ordinateur

Cryptographie et Sécurité

En résumé, les FPGA offrent une flexibilité exceptionnelle pour la conception de circuits électroniques personnalisés, et leur choix dépend des besoins spécifiques de l'application, des critères de performance, de puissance, de coût et de taille. Les FPGA modernes intègrent des ressources spécifiques pour certaines applications, ce qui les rend encore plus polyvalents.

3.7 Les technologies des éléments programmables

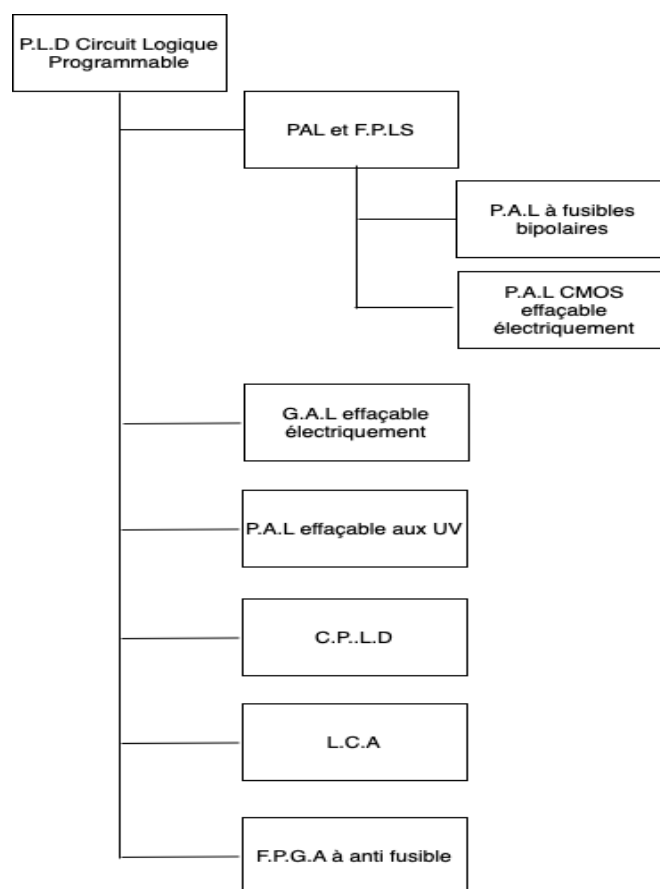


FIGURE 3.4 – Matrices AND programmée

Les éléments programmables sont des composants électroniques qui peuvent être programmés pour réaliser différentes fonctions logiques. Il existe plusieurs technologies pour les éléments programmables, chacune ayant ses avantages et ses limitations. Voici quelques-unes des technologies les plus courantes :

Dispositifs programmables en lecture seule (PROM) : Les PROM sont des dispositifs programmables qui peuvent être utilisés pour stocker des données permanentes, telles que des tables de recherche ou des programmes. Les données sont programmées lors de la fabrication du dispositif et ne peuvent pas être modifiées.

Dispositifs programmables effaçables en série (EPROM) : Les EPROM sont des dispositifs programmables qui peuvent être effacés en exposant le dispositif à une lumière ultraviolette intense pendant un certain temps. Cela permet de reprogrammer le dispositif avec de nouvelles données. Les EPROM sont utilisés pour stocker des données qui

doivent être mises à jour périodiquement, telles que des microprogrammes.

Dispositifs programmables effaçables et reprogrammables en série (EEPROM) : Les EEPROM sont des dispositifs programmables similaires aux EPROM, mais qui peuvent être effacés et reprogrammés électriquement. Les EEPROM sont plus pratiques que les EPROM car elles peuvent être reprogrammées sans avoir besoin d'être retirées du circuit.

Dispositifs logiques programmables (PLD) : Les PLD sont des dispositifs qui peuvent être programmés pour réaliser des fonctions logiques spécifiques. Ils sont souvent utilisés pour concevoir des circuits logiques personnalisés tels que des décodeurs, des multiplexeurs et des compteurs. Les PLD incluent les circuits programmables enregistrables (PAL) et les matrices logiques programmables (PLA).

Dispositifs logiques programmables complexes (CPLD) : Les CPLD sont des circuits logiques programmables plus grands et plus complexes que les PLD. Ils sont capables de réaliser des fonctions logiques plus complexes que les PLD et peuvent être utilisés pour concevoir des systèmes électroniques plus sophistiqués.

Réseaux de portes programmables par champ (FPGA) : Les FPGA sont des dispositifs programmables qui permettent de concevoir des circuits électroniques très complexes. Ils sont composés de blocs logiques programmables et d'interconnexions programmables qui peuvent être configurés pour réaliser pratiquement n'importe quelle fonction logique. Les FPGA sont utilisés dans de nombreuses applications, notamment les systèmes embarqués, les équipements de test et les ordinateurs haute performance.

En somme, il existe plusieurs technologies pour les éléments programmables, chacune ayant ses avantages et ses limitations. Les choix de technologies dépendent des besoins spécifiques de l'application et des exigences de performance.

3.8 Architecture des FPGA

3.9 Présentation des CP (Circuits programmables type PLA, CPLD)

Les circuits programmables sont des circuits intégrés conçus pour être programmés pour réaliser des fonctions logiques spécifiques. Les circuits programmables les plus cou-

3.9. PRÉSENTATION DES CP (CIRCUITS PROGRAMMABLES TYPE PLA, CPLD) 51

rants sont les circuits programmables enregistrables (PAL), les matrices logiques programmables (PLA) et les circuits logiques programmables complexes (CPLD).

Les PLA (Matrices Logiques Programmables) : Les PLA sont des circuits programmables qui permettent de réaliser des fonctions logiques combinatoires. Ils contiennent une matrice de portes logiques programmables et un réseau de connecteurs programmables. Les entrées sont connectées à des lignes d'entrée programmables et les sorties sont connectées à des lignes de sortie programmables. Les PLA peuvent être programmés pour réaliser pratiquement n'importe quelle fonction logique combinatoire en utilisant des termes d'entrée et de sortie programmables. Les PLA ont été largement utilisés dans les applications électroniques, mais ont été remplacés par des technologies plus avancées comme les CPLD et les FPGA.

Les CPLD (Circuits logiques programmables complexes) : Les CPLD sont des circuits programmables plus avancés que les PLA. Ils sont capables de réaliser des fonctions logiques plus complexes que les PLA, en utilisant une matrice de blocs logiques programmables et une matrice de connecteurs programmables. Les blocs logiques sont des circuits combinatoires ou séquentiels programmables qui peuvent être configurés pour réaliser une fonction logique spécifique. Les connecteurs programmables relient les blocs logiques et les entrées/sorties du circuit. Les CPLD sont utilisés pour concevoir des circuits logiques personnalisés tels que des décodeurs, des multiplexeurs et des compteurs.

Les Circuits Programmables Enregistrables (PAL) : Les circuits programmables enregistrables (PAL) sont des circuits intégrés programmables qui permettent de réaliser des fonctions logiques combinatoires. Ils contiennent une matrice de portes logiques programmables et un réseau de connecteurs programmables. Les PAL peuvent être programmés en utilisant un ensemble de termes de sortie programmables qui définissent la sortie en fonction des entrées. Les PAL sont souvent utilisés dans les applications électroniques qui nécessitent des fonctions logiques combinatoires simples.

En somme, les circuits programmables sont des circuits intégrés qui permettent de réaliser des fonctions logiques spécifiques. Les types les plus courants sont les circuits programmables enregistrables (PAL), les matrices logiques programmables (PLA) et les circuits logiques programmables complexes (CPLD). Ces circuits peuvent être programmés pour répondre aux besoins spécifiques d'une application, ce qui les rend très utiles pour la conception de systèmes électroniques personnalisés.

3.10 Structure des FPGA & ASICs

Les FPGA (Field Programmable Gate Arrays) et les ASICs (Application-Specific Integrated Circuits) sont deux types de circuits intégrés conçus pour réaliser des fonctions logiques spécifiques. Bien qu'ils aient des fonctionnalités similaires, leur structure et leur fonctionnement diffèrent.

Structure des FPGA : Les FPGA sont des circuits intégrés programmables qui contiennent une matrice de blocs logiques programmables (CLB) et une matrice de connexions programmables. Les blocs logiques programmables peuvent être configurés pour réaliser des fonctions logiques spécifiques, tandis que la matrice de connexions programmables permet de connecter les blocs logiques entre eux et aux entrées/sorties du circuit. Les FPGA sont configurés par l'utilisateur pour réaliser des fonctions logiques personnalisées. La structure d'un FPGA est donc très flexible et peut être adaptée à différentes applications.

Structure des ASICs : Les ASICs sont des circuits intégrés conçus pour réaliser une fonction logique spécifique. Contrairement aux FPGA, les ASICs sont conçus pour une application spécifique et ne peuvent pas être reprogrammés une fois qu'ils sont fabriqués. La structure des ASICs est donc optimisée pour la fonction spécifique qu'ils sont censés accomplir. Les ASICs sont conçus pour offrir des performances et une efficacité énergétique optimales pour une application donnée.

En résumé, la principale différence entre les FPGA et les ASICs réside dans leur structure. Les FPGA sont des circuits programmables et modifiables, tandis que les ASICs sont des circuits intégrés optimisés pour une application spécifique. Les FPGA sont très flexibles et peuvent être utilisés pour une grande variété d'applications, tandis que les ASICs sont plus spécialisés et sont conçus pour offrir des performances et une efficacité énergétique optimales pour une application particulière.

3.11 Architecture générale

L'architecture générale des FPGA et des ASICs est similaire, mais elle comporte également des différences importantes.

Architecture des FPGA : Les FPGA sont constitués de blocs logiques programmables (CLB), de blocs mémoire et de matrice de connexions programmables (PIM). Les blocs

logiques programmables peuvent être configurés pour réaliser différentes fonctions logiques, comme des portes logiques, des multiplexeurs, des registres, etc. Les blocs mémoire peuvent être utilisés pour stocker des données. La matrice de connexions programmables permet de connecter les blocs logiques entre eux et aux entrées/sorties du circuit.

Architecture des ASICs : Les ASICs sont conçus pour une application spécifique, ils peuvent donc être très différents les uns des autres en termes d'architecture. Cependant, la plupart des ASICs sont constitués d'une ou plusieurs unités de traitement, de mémoire et d'interfaces de communication. Les unités de traitement peuvent être des processeurs, des DSP (Digital Signal Processors), des blocs de traitement de signal, etc. La mémoire peut être utilisée pour stocker des données et des instructions. Les interfaces de communication peuvent être utilisées pour échanger des données avec d'autres circuits.

En résumé, la principale différence entre l'architecture des FPGA et des ASICs est que les FPGA sont des circuits programmables et modifiables, tandis que les ASICs sont conçus pour une application spécifique. Les FPGA sont constitués de blocs logiques programmables, de blocs mémoire et de matrice de connexions programmables, tandis que les ASICs peuvent comporter des unités de traitement, de mémoire et d'interfaces de communication en fonction de l'application.

3.12 Blocs logiques programmables

Les blocs logiques programmables (CLB) sont des éléments clés des FPGA (Field Programmable Gate Arrays). Ils sont utilisés pour réaliser des fonctions logiques spécifiques dans un circuit FPGA. Les blocs logiques programmables sont constitués de plusieurs éléments logiques de base, tels que des portes AND, OR et NOT. Ces éléments logiques peuvent être programmés pour réaliser des fonctions logiques plus complexes.

Les blocs logiques programmables contiennent également des éléments de mémoire, tels que des bascules ou des registres. Cela permet de stocker les résultats intermédiaires des calculs et de les utiliser dans les fonctions logiques ultérieures. Les blocs logiques programmables peuvent également contenir des multiplexeurs, des démultiplexeurs et des comparateurs pour réaliser des opérations plus complexes.

En général, les blocs logiques programmables peuvent être configurés pour réaliser différentes fonctions logiques. Ils peuvent être programmés par l'utilisateur pour réaliser des

fonctions spécifiques en utilisant des langages de description matérielle tels que VHDL ou Verilog. Les blocs logiques programmables sont une caractéristique clé des FPGA, car ils permettent une grande flexibilité dans la conception de circuits numériques. Les concepteurs de circuits peuvent utiliser des blocs logiques programmables pour créer des circuits personnalisés pour répondre aux besoins spécifiques d'une application donnée.

3.13 Terminologies

Les CPLD (Complex Programmable Logic Devices) sont des circuits programmables qui permettent aux concepteurs de circuits de créer des circuits logiques personnalisés pour répondre aux besoins spécifiques d'une application donnée. Les CPLD sont constitués de plusieurs blocs logiques programmables (PLB) qui sont interconnectés par des lignes de communication programmables.

Voici quelques termes couramment utilisés en relation avec les CPLD :

Bloc logique programmable (PLB) : Un bloc logique programmable est un élément logique de base du CPLD. Il est utilisé pour réaliser des fonctions logiques spécifiques et est configuré pour fonctionner en conjonction avec d'autres blocs logiques programmables.

Macrocell : Une macrocell est un groupe de blocs logiques programmables qui est utilisé pour réaliser une fonction logique spécifique. Une macrocell peut contenir des éléments de mémoire, tels que des bascules ou des registres, pour stocker les résultats intermédiaires des calculs.

LUT (Look-Up Table) : Une LUT est un élément logique de base du bloc logique programmable. Elle peut être programmée pour réaliser une fonction logique spécifique et peut être configurée pour utiliser jusqu'à quatre entrées.

Flip-Flop : Un flip-flop est un élément de mémoire qui est utilisé pour stocker un seul bit d'information. Les flip-flops sont souvent utilisés dans les macrocells pour stocker les résultats intermédiaires des calculs.

Product Term : Un terme de produit est une fonction logique AND programmable qui est utilisée pour créer des fonctions logiques plus complexes. Les termes de produit sont souvent utilisés dans les CPLD pour réaliser des fonctions logiques spécifiques.

Bus de configuration : Un bus de configuration est une interface qui permet de programmer les CPLD. Il est utilisé pour télécharger la configuration du circuit dans le CPLD.

En général, les CPLD sont utilisés pour des applications où la complexité de la logique est

relativement faible et où une faible consommation d'énergie est importante. Les CPLD sont également utilisés pour des applications où la flexibilité est importante et où la conception de circuits personnalisés est nécessaire.

3.14 Blocs de mémoire intégrée

Les FPGA (Field Programmable Gate Arrays) sont des circuits intégrés programmables qui permettent aux concepteurs de circuits de créer des circuits numériques personnalisés pour répondre aux besoins spécifiques d'une application donnée. L'un des éléments clés des FPGA est la mémoire intégrée, qui est utilisée pour stocker les données et les résultats intermédiaires des calculs.

Il existe plusieurs types de mémoire intégrée dans les FPGA, chacun avec ses propres avantages et limitations. Voici les types de mémoire intégrée les plus couramment utilisés en FPGA : RAM (Random Access Memory) : La RAM est une mémoire volatile qui peut être lue et écrite à tout moment. Elle est utilisée pour stocker les données et les résultats intermédiaires des calculs. Les FPGA contiennent souvent plusieurs types de RAM, tels que la RAM bloc, la RAM distribuée et la RAM DSP.

ROM (Read Only Memory) : La ROM est une mémoire non volatile qui est utilisée pour stocker des données qui ne changent pas souvent. Elle peut être programmée en usine ou par l'utilisateur pour stocker des données telles que des tables de recherche ou des instructions de programme.

Flash : La mémoire flash est une mémoire non volatile qui est utilisée pour stocker la configuration du FPGA. Elle est programmée en utilisant un processeur spécialisé appelé programmeur de FPGA.

EEPROM (Electrically Erasable Programmable Read Only Memory) : L'EEPROM est une mémoire non volatile qui peut être programmée et effacée électriquement. Elle est utilisée pour stocker des données qui doivent être conservées même lorsque l'alimentation est coupée.

Cache : Les FPGA peuvent également contenir des caches de mémoire, qui sont utilisés pour stocker les données fréquemment utilisées pour améliorer les performances du circuit.

En général, la mémoire intégrée dans les FPGA est utilisée pour stocker des données et

des résultats intermédiaires des calculs. La RAM est souvent utilisée pour stocker les données et les résultats intermédiaires des calculs, tandis que la ROM est utilisée pour stocker des données qui ne changent pas souvent. La mémoire flash est utilisée pour stocker la configuration du FPGA et l'EEPROM est utilisée pour stocker des données qui doivent être conservées même lorsque l'alimentation est coupée. Les caches de mémoire sont utilisés pour améliorer les performances du circuit en stockant les données fréquemment utilisées.

3.15 Exemples de constructeurs Altera et Xilinx

Altera et Xilinx sont deux des principaux fabricants de FPGA et proposent une large gamme de produits pour répondre aux besoins des concepteurs de circuits numériques. Voici quelques exemples de produits de ces deux fabricants :

Altera :

FPGA Intel Stratix 10 : la gamme la plus récente de FPGA d'Intel/Altera, offrant des performances et une densité accrues pour les applications de traitement de données à grande échelle. FPGA Cyclone V : une famille de FPGA économiques et économes en énergie, adaptés aux

applications grand public, industrielles et militaires.

CPLD MAX V : une famille de CPLD (Complex Programmable Logic Device) d'entrée de gamme offrant des performances élevées et une faible consommation d'énergie. Xilinx :

FPGA Virtex UltraScale+ : la gamme la plus avancée de FPGA de Xilinx, offrant des performances, une densité et une évolutivité supérieures pour les applications de traitement de données les plus exigeantes.

FPGA Zynq-7000 : une famille de FPGA avec processeurs ARM intégrés, permettant aux concepteurs de combiner des fonctions de traitement programmables avec des capacités de traitement de données avancées.

CPLD CoolRunner-II : une famille de CPLD d'entrée de gamme offrant une faible consommation d'énergie et une grande densité de logique.

Ces exemples ne sont pas exhaustifs et les deux fabricants proposent de nombreux autres produits pour répondre aux besoins des concepteurs de circuits numériques.

La méthodologie de conception de circuits électroniques peut varier en fonction de la complexité du circuit, de la technologie utilisée et des ressources disponibles. Voici une méthodologie de conception générale, suivie des méthodes spécifiques pour les circuits à faibles densités d'intégration (LDI) et les circuits à haute densité d'intégration (HDI), ainsi que des informations sur les outils de développement et les compilateurs associés.