Circuits Reconfigurables Robustes

Jean-Max Dutertre

UNIVERSITE MONTPELLIER II

SCIENCES ET TECHNIQUES DU LANGUEDOC

T H E S E

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE MONTPELLIER II

Discipline : Electronique, Optronique et Systèmes
Formation Doctorale : Systèmes Automatiques et Microélectroniques
Ecole Doctorale : Information, Structures et Systèmes

présentée et soutenue publiquement

par

Jean-Max DUTERTRE

Le 30 octobre 2002

Circuits Reconfigurables Robustes

JURY

M. BERTRAND Y., Professeur, Université Montpellier II	, Président
M. FOUILLAT P., Professeur, ENSEIRB, Université Bordeaux I	, Rapporteur
M. NICOLAIDIS M., Directeur de Recherche CNRS, iRoC Technologies	, Rapporteur
M. CATHEBRAS G., Maître de Conférences, Université Montpellier II	, Examinateur
M. MICHEL J., Maître de Conférences, Université Strasbourg I	, Examinateur
M. ROCHE F.M., Maître de Conférences, Université Montpellier II	, Directeur de Thèse

Au terme de ce travail de thèse effectué au Département Microélectronique du Laboratoire d'Informatique, de Robotique et de Microélectronique de l'Université Montpellier II, je tiens à remercier :

Monsieur Gaston CAMBON, Professeur, Directeur du LIRMM de m'avoir accueilli au sein du Laboratoire.

Messieurs Christian LANDRAULT et Michel RENOVELL, Directeurs de Recherche CNRS, responsables successifs du Département Microélectronique de m'avoir fourni des conditions optimales à la poursuite de cette étude.

Monsieur Yves BERTRAND, Professeur à l'Université Montpellier II, qui m'a fait l'honneur de présider le jury de cette Thèse.

Monsieur Pascal FOUILLAT, Professeur à l'ENSEIRB, Université Bordeaux I et Monsieur Michaël NICOLAIDIS, Directeur de Recherche CNRS, Directeur Scientifique de iRoC Technologies, qui m'ont fait l'honneur d'accepter d'être les rapporteurs du mémoire.

Monsieur Guy CATHEBRAS, Maître de Conférences à l'Université Montpellier II et Monsieur Jacques MICHEL, Maître de Conférences à l'Université Strasbourg I, pour leur participation au Jury .

Monsieur Fernand-Michel ROCHE, Maître de Conférences à l'Université Montpellier II, pour m'avoir encadré et avoir dirigé mes recherches tout au long de ces trois années. Je le remercie également pour le choix du sujet de ce travail, que j'ai trouvé passionnant.

J'adresse une nouvelle fois mes remerciements à Monsieur Yves BERTRAND, pour son rôle dans mon choix du LIRMM et également pour la confiance qu'il n'a cessé de me témoigner comme moniteur dans le cadre des enseignements qu'il m'avait confié en DEUG STPI . Que Monsieur Guy CATHEBRAS, trouve également ici l'expression de ma gratitude pour sa disponibilité dans le règlement de mes soucis d'utilisation des outils de CAO ; le circuit que nous avons réalisé lui doit une grande part.

Je tiens aussi à assurer de toute mon amitié mes collègues thésards du LIRMM : Serge, mon ex–colocataire qui m’a vivement incité à venir à Montpellier; Frédéric et Philippe, et également Xavier et Yannick, pour les parties homériques disputées ensembles. Isabelle qui a partagé notre box ; Julien, Olivier, Vincent, Philippe (le 2éme) et encore Yannick les "Star men" ; Marianne et Ben, Régis, Wence, Pascal, Alain, Gilles, David et tous les autres qui m’ont rendu si agréables ces trois années.

Merci également à tous les personnels du LIRMM que j’ai côtoyés avec beaucoup de plaisir.

Enfin je remercie Isabelle pour son aide, son soutien et ses relectures lors de la rédaction de ce manuscrit.

Table des matières

I

Introduction générale

Introduction générale.

Les premiers circuits intégrés reconfigurables de type FPGA (Field Programmable Gate Array) ont été commercialisés par la société Xilinx en 1985. Depuis cette date d'autres acteurs sont apparus sur un marché qui n'a cessé de croître. Au fur et à mesure que la complexité des FPGA s'est développée, leurs possibilités d'emploi se sont accrues, jusqu'à concurrencer sérieusement les circuits spécifiques pour de petits volumes de production (jusqu'à quelques milliers de composants). C'est pourquoi on envisage de plus en plus de développer l'utilisation des FPGA dans le cadre des applications aéronautiques et spatiales. Ils apportent une grande facilité d'emploi et permettent une réduction du coût de développement des électroniques embarquées. Les FPGA reconfigurables à base de SRAM (Static Random Access Memory) apportent en outre la possibilité de faire évoluer in situ la fonctionnalité de ces circuits, ce qui ouvre des perspectives de maintenance à distance que n'autorisent pas les circuits spécifiques.

Mais les circuits intégrés utilisés dans les domaines aéronautiques et spatiaux sont confrontés à un environnement radiatif complexe qui perturbe leur fonctionnement et qui peut même parfois devenir destructif [BIN75]. Aussi, tout un travail de recherche est poursuivi depuis une vingtaine d'années afin d'assurer un durcissement des circuits intégrés vis à vis de ces phénomènes. Depuis quelques années ce champ d'étude profite d'un intérêt croissant lié à la mise en évidence d'erreurs d'origine radiative au niveau de la surface terrestre [ZIE96], [NOR96]. L'augmentation de leur probabilité d'apparition accompagne l'évolution de la technologie des circuits intégrés vers des gravures sans cesse plus fines.

Le travail de recherche présenté dans ce manuscrit de thèse est consacré au développement de solutions de durcissement applicables aux circuits reconfigurables à base de SRAM. Nous nous sommes plus particulièrement intéressé à une famille d'effets radiatifs : les effets singuliers, en considérant leur action sur des circuits réalisés dans une technologie standard de type " bulk CMOS ".

Après avoir souligné dans une première étape les spécificités propres à l'architecture des FPGA à base de SRAM nous décrivons les principaux phénomènes afin de mettre en évidence leurs effets sur les circuits reconfigurables.

Nous présentons ensuite les principaux types de durcissement existants et leurs caractéristiques. Et nous exposons finalement les méthodes de durcissement que nous avons développées.

Deux approches sont proposées. La première repose sur une restructuration des cellules de base (inverseurs et éléments de mémorisation) au niveau de l'agencement de leurs transistors, tandis que la deuxième est basée sur l'utilisation d'un algorithme de détection et de correction d'erreurs.

Enfin, avant de conclure, nous présentons la conception d'un circuit de test réalisé afin de valider expérimentalement ce travail.

II

Position du problème

II.A. Les FPGA à base de SRAM.

L'objectif de cette partie est de présenter l'architecture des FPGA à base de SRAM et de mettre en évidence ses particularités par rapport aux circuits intégrés spécifiques classiques.

A.1 – Présentation et architecture des FPGA.

A.1.1. Intérêt des FPGA.

Les circuits programmables de type FPGA (Field Programmable Gate Arrays) sont constitués de blocs logiques élémentaires et de réseaux d'interconnexions pouvant être configurés par l'utilisateur. Celui-ci a la possibilité d'implanter une fonctionnalité donnée dans ces circuits sans avoir à se préoccuper des différentes étapes de sa fabrication. La complexité des circuits implantés dans les FPGA leur permet aujourd'hui de concurrencer les circuits spécifiques pour de petits volumes de production (jusqu'à quelques milliers d'unités). Ils sont donc très bien adaptés à une utilisation pour des applications de type spatial. Le recours aux ASIC (Application Specific Integrated Circuits) impose en effet des temps de développement et de fabrication de l'ordre de plusieurs mois et ce pour un coût élevé. Les FPGA autorisent la réalisation de circuits ayant une complexité équivalente en des temps et pour des coûts plus réduits. En outre, de nombreux FPGA permettent une reconfiguration à volonté et donc une évolution des circuits au cours de leur vie. Ce qui nous amène à nous intéresser de plus près à leur architecture.

A.1.2. Couche de configuration et couche opérative.

Pour décrire les FPGA on peut avoir recours à un partitionnement symbolique de leur architecture selon deux couches : la couche opérative et la couche de configuration.
La couche opérative comprend les différents éléments assurant la réalisation de la fonction programmée : les blocs logiques élémentaires, les réseaux d'interconnexion et les entrées / sorties.

La couche de configuration regroupe tous les éléments nécessaires à la programmation du circuit. Il existe plusieurs technologies de programmation des FPGA.
Certains fabricants, comme Actel par exemple, ont recours à une programmation par antifusibles. C'est la plus économe en terme de surface mais les circuits sont alors configurés une fois pour toute.

L'utilisation d'une technologie à base d'EPROM (Eraseable Programmable Read Only Memory) permet de reprogrammer plusieurs fois un FPGA. Cependant l'effacement d'une configuration impose une exposition du circuit à un rayonnement ultraviolet. Une reprogrammation à distance du circuit n'est donc pas envisageable. Cette limitation a été levée par l'apparition de technologies à base d'EEPROM (Electrically Eraseable Programmable Read Only Memory) qui permettent une reprogrammation électrique des composants in situ. Mais les principaux fabricants de FPGA pour des applications aéronautiques et spatiales (Actel et Xilinx entre autres) ne proposent pas encore de circuits reconfigurables basés sur cette technologie dans le cadre de ces applications.

Nous avons fait le choix de nous intéresser plus particulièrement aux FPGA à base de SRAM (Static Random Access Memory). Ces FPGA présentent le grand avantage d'être programmables et reconfigurables à volonté in situ. Ils offrent ainsi la possibilité intéressante d'avoir une fonctionnalité modifiable à distance. C'est cependant la technologie la plus gourmande en surface. Un autre inconvénient de ces FPGA est la volatilité de leur configuration, celle-ci est perdue à chaque mise hors tension du circuit. La configuration du circuit doit être stockée dans un élément extérieur au FPGA (PROM, EEPROM, disque dur, etc.) et chargée à chaque mise sous tension.

Les parties suivantes présentent plus en détail les couches opérative et de configuration des FPGA à base de SRAM (cf. figure 1).

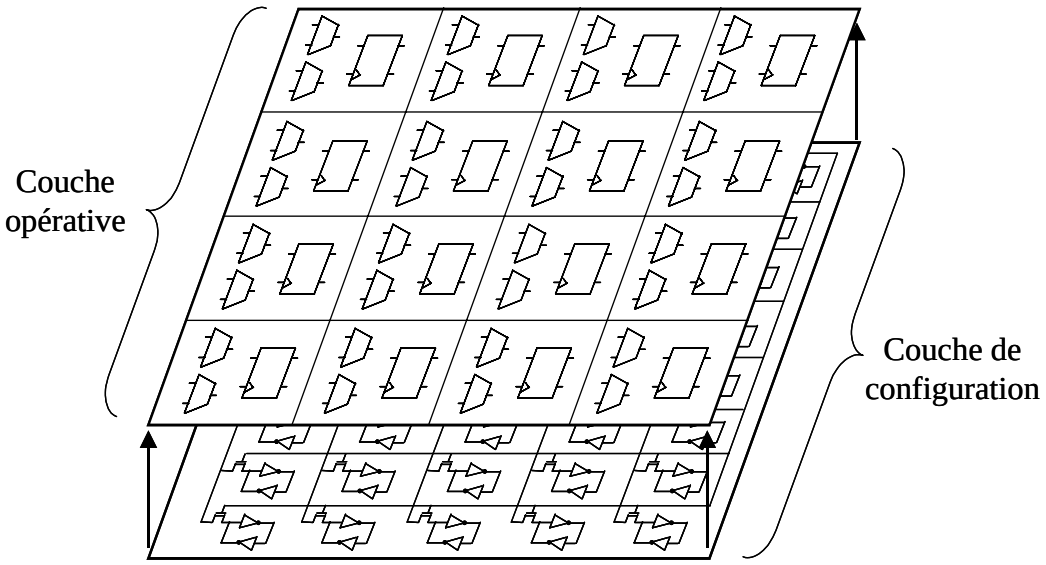


Figure 1 : Partitionnement symbolique d'un FPGA en couches de configuration et opérative

Cette partition en deux couches est uniquement symbolique et formelle. Elle n'a pas de réalité physique, car au niveau du substrat les deux couches sont étroitement entremêlées.

A.2 – La couche opérative.

Dans une représentation symbolique la couche opérative regroupe les différents éléments configurables nécessaires à l'implantation d'un circuit logique dans un FPGA à l'exclusion des parties dédiées à sa programmation. Il s'agit des blocs logiques, des réseaux d'interconnexion et des entrées / sorties du circuit. La figure 2 présente le principe d'architecture retenu par de nombreux constructeurs (Xilinx et Lattice).

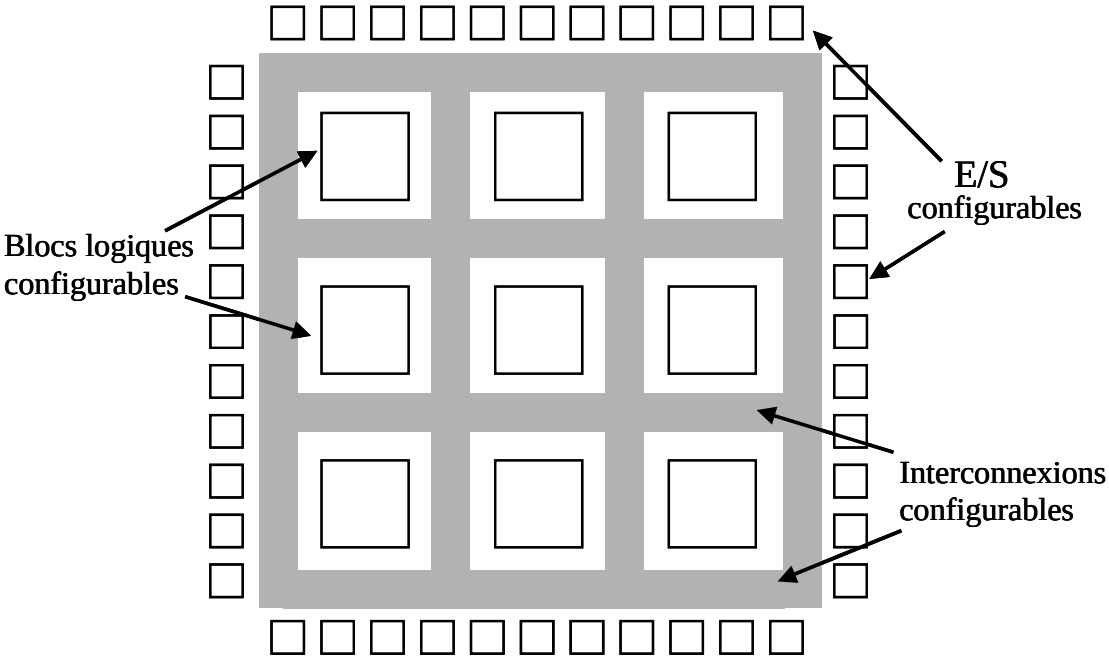


Figure 2 : Architecture des FPGA symétriques.

Il s'agit d'une architecture symétrique. Les blocs logiques configurables sont rangés sous la forme d'un tableau régulier et symétrique. Ils sont entourés d'un réseau d'interconnexions configurables horizontales et verticales qui assurent l'échange des données entre les blocs logiques mais aussi avec l'extérieur (entrées / sorties primaires du circuit). Cette architecture est la plus répandue. A côté de celle-ci une architecture dite hiérarchique est utilisée par des constructeurs tels que Actel et Altera. Il existe de grandes similitudes entre ces deux architectures, aussi nous nous intéresserons plus spécifiquement à l'architecture symétrique pour présenter la couche opérative des FPGA.

A.2.1. Les cellules logiques.

La couche opérative d'un FPGA comporte un grand nombre de cellules logiques élémentaires configurables. Ces cellules comportent des éléments de logique purement combinatoire et des éléments de logique séquentielle afin de permettre la réalisation de fonctions complexes. Pour illustrer notre propos considérons le schéma de principe d'un bloc logique configurable utilisé dans les FPGA de la série Virtex-E de la société Xilinx (Figure 3).

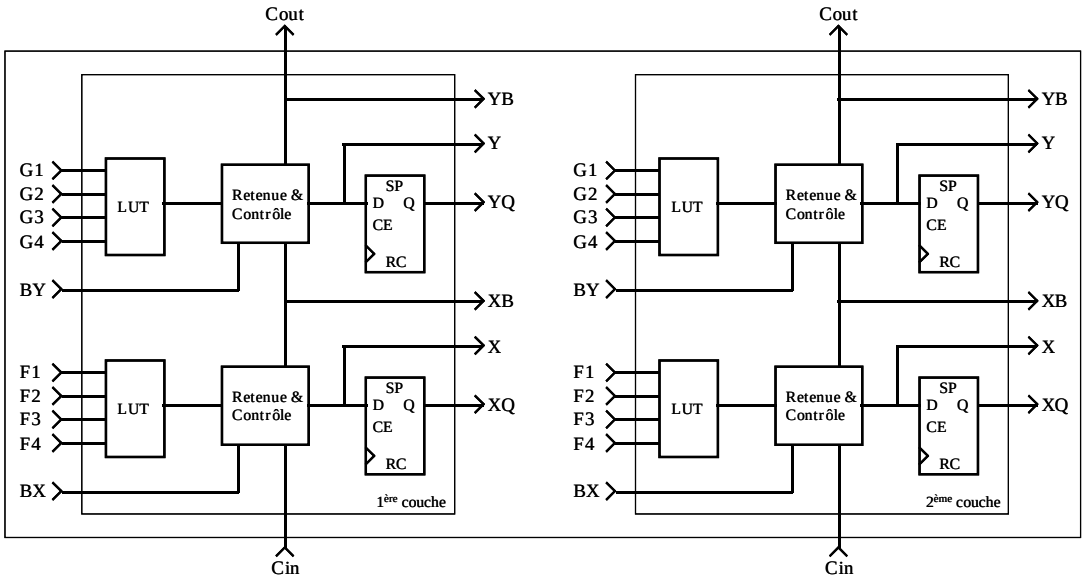


Figure 3 : Bloc logique configurable d'un FPGA Xilinx Virtex-E.

Ce bloc logique est constitué de deux couches logiques, chacune d'entre elles regroupant deux cellules logiques élémentaires. Dans une cellule logique on trouve une partie purement combinatoire comprenant une table d'allocation de données (notée LUT pour Look Up Table), un module permettant la propagation d'une retenue et plusieurs multiplexeurs non représentés. On trouve également un élément séquentiel : une flip-flop de type D dont les fonctionnalités sont configurables.

En fonction de la programmation, la donnée d'entrée de la D flip-flop provient au choix de la table d'allocation ou directement d'une entrée de la cellule logique. La sortie de la table d'allocation peut être aiguillée directement sur une sortie, sur l'entrée de la flip-flop et sur la sortie de retenue sans exclusivité. Une cellule logique peut ainsi être utilisée comme un élément purement combinatoire, purement séquentiel ou à la fois combinatoire et séquentiel. L'association de ces cellules via les diverses interconnexions d'un FPGA permet de réaliser des circuits aussi complexes que ceux proposés sous forme de circuits spécifiques.

Nous allons maintenant nous intéresser aux cellules élémentaires de ces blocs logiques : les multiplexeurs, les tables d'allocation et les D flip-flop.

Les multiplexeurs

Le bloc logique présenté figure 3 comporte onze multiplexeurs à deux entrées. Ce chiffre met en évidence leur importance dans les FPGA, et ce quel que soit le constructeur considéré.

Le multiplexeur le plus simple comporte deux entrées de données, une entrée de sélection et une sortie. La fonction logique réalisée correspond à un aiguillage. Selon la valeur de l'entrée de sélection la sortie est égale à l'une des entrées de données. La figure 4 présente le schéma symbolique, la table de vérité et une réalisation à base de portes de transmission d'un multiplexeur à deux entrées.

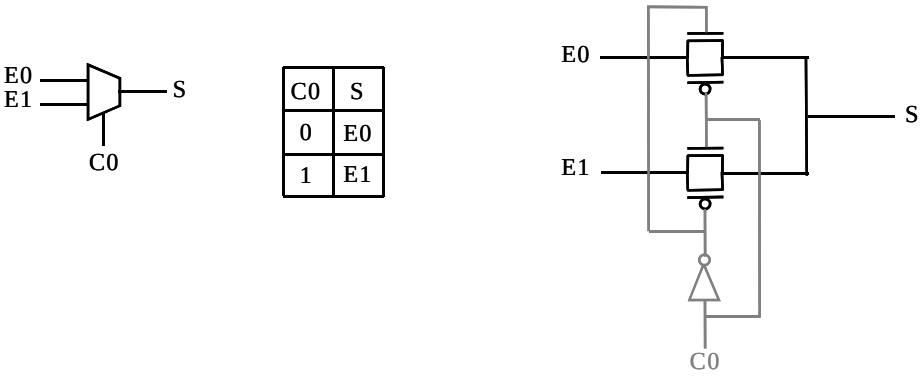


Figure 4 : Architecture d'un multiplexeur à deux entrées.

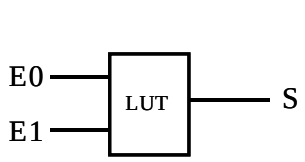
Lorsque l'entrée de sélection (C0) est à 0, la sortie S prend la valeur de l'entrée E0 et réciproquement lorsque C0 est à 1, S prend la valeur de E1.

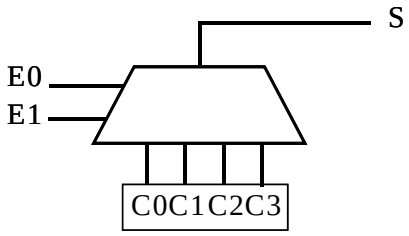
Sur le même principe on réalise des multiplexeurs à 2^n entrées de données, n étant le nombre d'entrées de sélection nécessaires pour sélectionner l'entrée à aiguiller vers la sortie.

La majorité des multiplexeurs utilisés dans les FPGA ont comme entrée de sélection un bit de configuration. Ils sélectionnent alors toujours la même entrée de donnée. A titre d'exemple, l'entrée D de la flip-flop est reliée à la sortie d'un multiplexeur à deux entrées, l'une correspond à la sortie de la table d'allocation et l'autre à une entrée de la cellule logique. Ainsi, en fonction de la configuration, la partie combinatoire de la cellule logique sera ou non utilisée.

Les tables d'allocation

Le rôle d'une table d'allocation est de permettre l'implantation de fonctions logiques de plusieurs variables. Le nombre de variables correspond au nombre d'entrées. Ainsi les tables d'allocation (ou LUT) du bloc logique configurable de la figure 3 permettent d'implanter directement des fonctions de quatre variables. Ce nombre peut être accru en les interconnectant. Le principe de fonctionnement d'une table d'allocation repose sur l'utilisation de multiplexeurs. Si on considère une fonction de deux variables, il faut avoir recours à un multiplexeur à quatre entrées (2^2). Son symbole, son schéma de principe et sa table de vérité sont donnés figure 5.





E0	E1	S
0	0	C0
0	1	C1
1	0	C2
1	1	C3

Figure 5 : Table d'allocation de deux variables.

Les entrées (E0 et E1) ou variables de la fonction sont connectées aux entrées de sélection du multiplexeur. Et, les quatre entrées de données sont connectées à des bits de configuration (C0, C1, C2 et C3). Prenons l'exemple de la fonction S = E0 ⊕ E1 dont la table de vérité est rappelée figure 6.

E0	E1	S
0	0	0
0	1	1
1	0	1
1	1	0

Figure 6 : Table de vérité de la fonction ou exclusif

Pour implanter cette fonction, il faut fixer, lors de la phase de programmation du FPGA, les valeurs de ces bits de configuration de sorte que les tables de vérité des figures 5 et 6 concordent. Dans ce cas, on aura C0=0, C1=1, C2=1 et C3=0.

D'une façon générale, pour réaliser une fonction de n variables on utilise un multiplexeur à 2^n entrées de données.

Les D flip-flop.

Les flip-flop de type D utilisées dans les FPGA sont également configurables. Parmi les fonctionnalités disponibles les plus courantes ; on peut citer : une entrée de validation du signal d'horloge et des entrées d'initialisation et de mise à 1 synchrones ou bien asynchrones selon la configuration. Leur architecture est détaillée au II.C.2.3 qui traite de leur susceptibilité aux phénomènes radiatifs.

Nous venons de présenter les différents constituants des blocs logiques configurables. On les retrouve dans la quasi totalité des FPGA existants. Nous allons maintenant présenter la façon dont ils sont interconnectés.

A.2.2. Les interconnexions.

Dans les FPGA à architecture symétrique on peut distinguer une composante d'interconnexion locale et une composante d'interconnexion globale. La composante locale a pour rôle d'interconnecter les éléments des blocs logiques en fonction de la configuration et d'assurer la communication entre les blocs logiques directement adjacents sans avoir recours à la composante globale. Cette dernière assure la circulation des données à l'échelle du FPGA entre les blocs logiques éloignés.

Au niveau local, on trouve donc des interconnexions configurables au sein des blocs logiques dans et entre les cellules logiques élémentaires. Elles sont réalisées au moyen de multiplexeurs, de buffers trois états et de portes passantes (un simple NMOS ou bien une porte de transmission associant un NMOS et un PMOS en parallèle). On trouve également un groupe d'interconnexions vers les blocs logiques directement adjacents selon le même principe. Il s'agit d'interconnexions courtes présentant de faibles retards si on les compare aux interconnexions à longue distance.

Au niveau global, on trouve un réseau d'interconnexions horizontales et verticales à l'échelle du circuit situé entre les blocs logiques (cf. le réseau grisé de la figure 2). L'échange des données entre les blocs logiques et ces interconnexions globales se fait par l'intermédiaire de Cellules d'Interconnexions Locales (CIL). L'aiguillage des données au sein du quadrillage d'interconnexions globales est assuré par des Cellules d'Interconnexions Globales (CIG) placées à chacune de ces intersections selon le schéma synoptique de la figure 7.

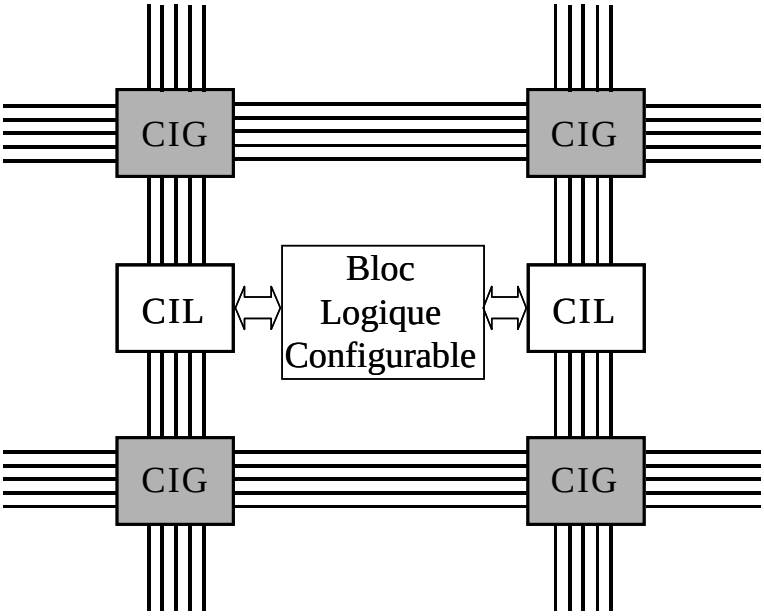


Figure 7 : Réseau d'interconnexions global.

Ces cellules (CIL et CIG) sont constituées principalement de multiplexeurs et de portes de transmission à un ou deux transistors. On trouve également de nombreux buffers et buffers trois états afin de régénérer régulièrement les signaux le long des interconnexions.
Il existe enfin, en général, dans les FPGA, un ou plusieurs réseaux d'interconnexions configurables dédiées aux signaux d'horloge. Ils comportent principalement des inverseurs et des buffers trois états.

A.2.3. Les entrées / sorties.

Les cellules d'entrée / sortie des FPGA sont des circuits propriétaires multifonctions relativement complexes. Elles peuvent être configurées au choix pour servir d'entrée ou de sortie primaire au circuit. Les données peuvent être délivrées directement ou bien via des éléments de stockage comme des D flip-flops ou de simples bascules D. De nombreuses fonctionnalités sont offertes, on peut citer des entrées de mise à 1 ou à 0 pour les bascules, des entrées de validation du signal d'horloge, etc. Ces cellules comportent également des buffers trois états, des circuits de "pull-up" et de "pull-down" et des circuits de protection contre les décharges électrostatiques.
Les cellules d'entrée / sortie des FPGA sont des éléments spécifiques complexes dont l'architecture exacte est difficile à connaître, aussi nous ne les envisagerons pas dans cette étude qui se veut plus générale.

A.3 – La couche de configuration.

Nous avons fait le choix de restreindre notre étude à celle des FPGA à base de SRAM. La couche de configuration de ces circuits regroupe les points mémoire chargés de mémoriser leur configuration ainsi que les registres et la logique dédiés à la programmation. Après avoir présenté l'architecture des points mémoire, nous nous intéressons à leur ordonnancement et au principe de configuration mis en œuvre.

A.3.1. Les points mémoires de configuration.

Les points mémoire retenus pour la configuration des FPGA sont des SRAM que nous noterons CSRAM pour Configuration Static Random Access Memory. La configuration des FPGA est donc volatile, elle doit être chargée à chaque mise sous tension.

Chaque CSRAM permet de stocker un unique bit de configuration, un FPGA en comporte donc un grand nombre. Ainsi, par exemple, la configuration d'une table d'allocation d'une fonction de quatre variables requiert seize (4^2) bits de configuration, soit seize CSRAM. A titre d'exemple, on peut citer un FPGA de la série Virtex de Xilinx, le XCV200E qui comporte 1 442 016 CSRAM afin de configurer ses 1176 blocs logiques. Aussi, l'architecture retenue pour les CSRAM est la plus simple possible afin de limiter la taille de la couche de configuration. C'est une architecture à cinq transistors présentée figure 8 qui répond à ce critère.

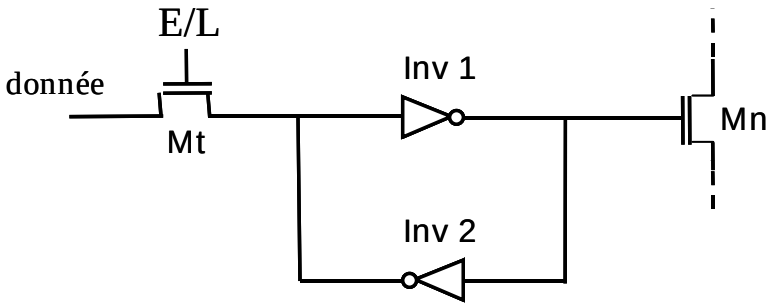


Figure 8 : Cellule mémoire de configuration (CSRAM)

Une CSRAM est constituée de deux inverseurs rebouclés (Inv1 et Inv2) et d'un transistor d'accès (Mt) de type NMOS. Le transistor d'accès permet l'écriture et éventuellement la relecture du bit de configuration. Dans cet exemple le bit de configuration stocké commande un NMOS (Mn) appartenant à la couche opérative.

A.3.2. Ordonnancement des CSRAM et principe de la configuration.

Physiquement, les CSRAM sont réparties régulièrement parmi les éléments de la couche opérative. Mais il est plus simple de décrire leur ordonnancement sous la forme d'un tableau bidimensionnel présenté figure 9.

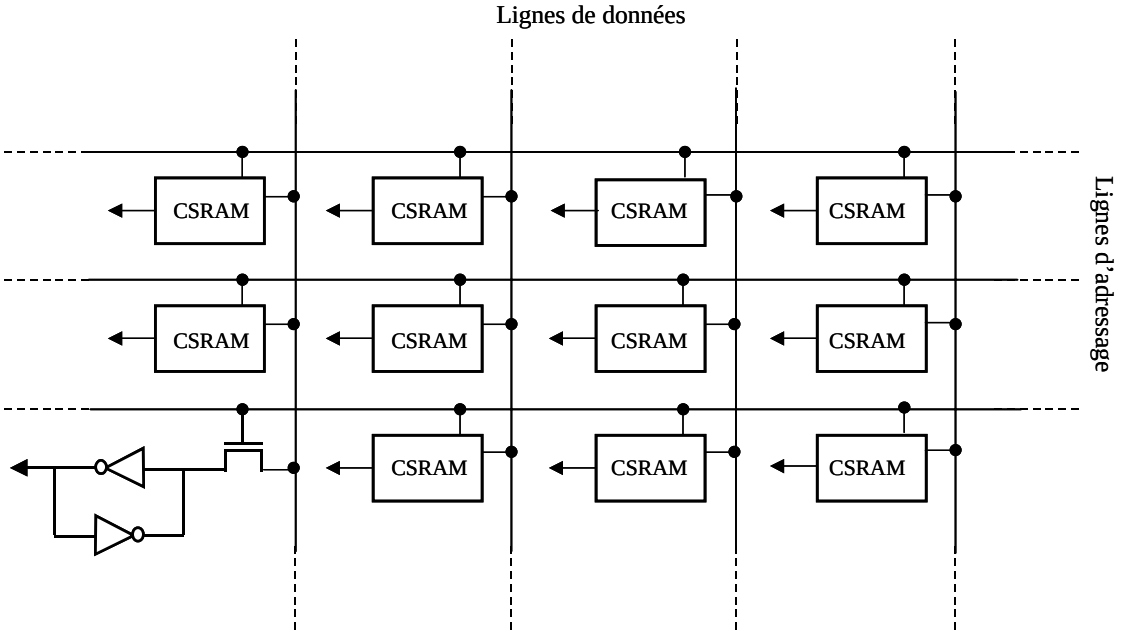


Figure 9 : Tableau bidimensionnel de CSRAM

Les CSRAM d'une même colonne partagent la même ligne de données et les CSRAM d'une même ligne partagent la même ligne d'adressage. Selon ce principe une ligne complète de CSRAM peut être écrite ou lue lorsque les transistors d'accès sont commandés en mode passant par la mise à un de leur ligne d'adressage. Les bits de configuration sont alors lus ou écrits sur l'ensemble des lignes de données.

Ce schéma de principe est repris de façon plus complète par le modèle de couche de configuration proposé par Michinishi [MIC97], il est présenté figure 10.

La couche de configuration comprend un tableau bidimensionnel de n colonnes et m lignes, constitué par les CSRAM ; un Registre à Décalage de Données (RDD) de longueur n dans lequel les bits de configuration sont chargés en série par mots de n bits ; un Registre à Décalage des Adresses (RDA) de longueur m qui permet de sélectionner la ligne de CSRAM activée en écriture ou en lecture ; et une logique non représentée chargée de gérer le processus de configuration.

Il y a deux modes de fonctionnement lors de la configuration d'un FPGA : l'écriture et la lecture.

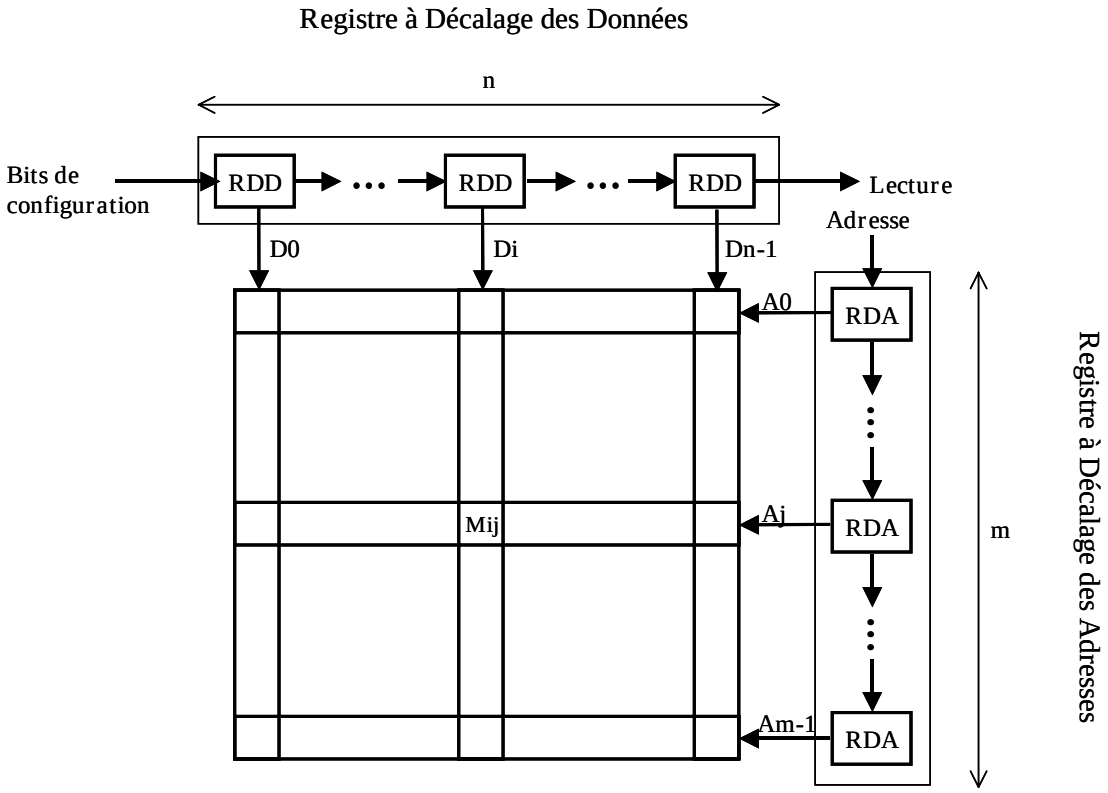


Figure 10 : Couche de configuration

Lors de la phase d'écriture, les bits de configuration sont chargés en série dans le RDD. Dès qu'un mot complet de n bits est positionné dans le RDD, il est écrit en parallèle, via les n lignes de données, dans les CSRAM d'une même ligne. Puis, un nouveau mot est chargé et les différentes lignes de CSRAM sont écrites successivement. L'écriture d'une ligne est validée par le RDA qui rend passant simultanément tous les transistors d'accès des CSRAM d'une même ligne. La configuration est donc chargée séquentiellement sous forme de mots qui sont écrits successivement dans les différentes lignes.

La phase de lecture permet de vérifier que la configuration du circuit a été chargée sans erreur. Elle fonctionne selon un principe similaire à l'écriture. Le premier mot est chargé dans le RDD (la ligne complète étant validée par le RDA) et celui-ci fonctionne ensuite comme un registre à décalage afin de récupérer sur une sortie série tous les bits du mot. Puis un second mot est chargé dans le RDD et ainsi de suite jusqu'à ce que toute la configuration soit lue. Dans la pratique, seules les signatures caractéristiques des différents mots sont envoyées en sortie lors de la relecture.

La gestion du RDD et du RDA nécessite la présence d'une logique de configuration non représentée figure 10, nous ne nous y intéresserons pas dans l'immédiat.

A.4 – Conclusion

Après avoir présenté l'architecture des FPGA à base de CSRAM et le principe de leur configuration, nous sommes à même de pointer leurs spécificités par rapport aux circuits spécifiques classiques.

La principale d'entre elles est l'existence d'une couche de configuration comportant jusqu'à plusieurs millions de points mémoire de configuration pour les plus complexes. C'est ce qui les distingue le plus radicalement des autres familles de circuits intégrés. Nous verrons dans les parties suivantes comment l'intégrité des bits de configuration stockés par les CSRAM est menacée par la présence d'un environnement radiatif.

La deuxième principale spécificité de ces circuits provient du grand nombre d'interconnexions qui peuvent être fragilisées par un environnement hostile.

Et enfin, leur logique combinatoire est constituée principalement par des multiplexeurs sensibles également aux radiations.

Les dernières générations de FPGA permettent d'implanter des circuits d'une grande complexité ce qui leur permet de concurrencer les ASIC à des coûts très avantageux pour les petites séries. Le marché des composants spatiaux et aéronautiques dans une moindre mesure, est caractérisé par de petits volumes d'où l'intérêt des FPGA pour ces applications dans un contexte de réduction des budgets de ces secteurs.

L'utilisation de FPGA à base de SRAM ouvre également la possibilité de reconfigurer les circuits à distance et donc de les faire évoluer ou de corriger des erreurs de conception qui n'auraient pas été détectées avant leur mise en orbite.

Aussi, nous nous attacherons dans la suite de ce manuscrit, après avoir décrit l'impact des phénomènes radiatifs sur les FPGA, à proposer des solutions de durcissement aux radiations afin d'améliorer leur fiabilité.

II.B. Les phénomènes radiatifs.

L'espace proche est parcouru par des particules cosmiques (ions lourds, photons) et solaires (photons, électrons, protons). Electrons et photons sont piégés au voisinage de la terre et dégradent de façon régulière les circuits intégrés (CI) placés en orbite. Les rayonnements cosmiques, principalement les ions lourds, sont à l'origine d'erreurs logiques et de pannes (parfois destructives) dans ces circuits. Nous nous attacherons dans cette partie à décrire, pour les technologies CMOS, les mécanismes à l'origine des phénomènes radiatifs et leurs conséquences.

B.1 – Origine des phénomènes radiatifs.

Les particules d'origine spatiale qui traversent les circuits intégrés interagissent avec les matériaux les constituant, il y a transfert d'une partie (ou de la totalité) de l'énergie des particules incidentes vers les matériaux considérés. Ce transfert est caractérisé par son caractère ionisant ou non.

Si le transfert est non ionisant, l'échange d'énergie se fait par choc avec les atomes du matériau, il crée alors des défauts dans le réseau cristallin du semi-conducteur, entraînant une modification de ses propriétés électriques. Si la particule incidente est assez énergétique (par exemple des protons d'origine solaire en orbite ou des neutrons dans l'atmosphère terrestre), il peut y avoir production d'ions lourds secondaires ou de particules alpha par initiation d'une réaction nucléaire locale ou par effet de recul. Ces ions seront à leur tour susceptibles d'interagir avec la matière selon les mécanismes décrits dans cette partie.

En cas de transfert de type ionisant, la particule perd son énergie en générant des paires électron – trou le long de sa trajectoire. Dans le cadre des applications spatiales, c'est ce type de phénomène qui a l'impact le plus significatif sur le fonctionnement des CI. Une partie des paires électron – trou générées se recombine aussitôt sans effet notable, les charges restantes, en fonction de leur devenir, sont susceptibles d'altérer les circuits intégrés.

Une partie de ces charges peut finir piégée dans les oxydes de grille au niveau de l'interface Si – SiO_2 des grilles et dans les oxydes de champ. Elles entraînent ainsi une dégradation des propriétés électriques des transistors qui se traduit par une dérive des tensions de seuil, l'apparition de courants de fuite, l'augmentation de la consommation statique des composants, etc. Ces effets sont progressifs et cumulatifs, ils se manifestent au bout d'un certain temps d'irradiation, on les qualifie " d'effets de dose cumulée".

Les paires électron – trou résiduelles présentes dans le substrat peuvent aussi créer, sous certaines conditions, des impulsions de courant. Elles sont à l'origine d'évènements singuliers au sein des circuits. On distinguera principalement, parmi ces effets singuliers, l'amorçage de structures thyristor parasites (ou Latch Up) et la génération de transitoires de tension dans la partie logique des circuits.

B.2 – Les effets de dose cumulée.

Les effets de dose sont crées par l'exposition cumulative des circuits intégrés à des particules radiatives telles que les photons, les ions, les protons et les électrons de haute énergie. Elles créent par interaction avec les oxydes de silicium (SiO_2) des paires électron - trou. La majorité de ces paires se recombine sans conséquences pour le circuit, mais une partie d'entre elles est à l'origine des effets de dose. Ils se manifestent au niveau des oxydes de grille des transistors et des oxydes de champ.

Deux mécanismes sont à l'œuvre dans les oxydes de grille.

Le premier est le piégeage d'une partie des trous ayant échappés à la recombinaison au niveau de l'interface entre l'oxyde de grille et le substrat de silicium. Il se caractérise par une dérive négative des tensions de seuil des transistors. Ce mécanisme est actif dès les faibles doses d'irradiation. Il existe également un mécanisme de piégeage des électrons mais il est peu significatif.

Le deuxième mécanisme est mal connu, il correspond à la création d'états d'interface entre l'oxyde de grille et le substrat. Plusieurs interprétations différentes coexistent [BEA95]. Lorsque les transistors sont en conduction, les états d'interfaces sont chargés, négativement pour les NMOS et positivement pour les PMOS. Ce qui induit une dérive positive de la tension de seuil des NMOS et une dérive négative de la tension de seuil des PMOS.
Dans le cas des NMOS, les effets de piégeage des trous et de création d'états d'interface sont opposés, c'est ce dernier qui s'impose finalement pour les irradiations les plus importantes. Dans le cas des PMOS, ces deux effets sont cumulatifs. Les conditions d'irradiation et l'existence de phénomène de guérison entraînent une cinétique complexe de ces effets [LER97]. L'effet de dose dans les oxydes de grille entraîne l'apparition de courants de fuite, l'augmentation de la consommation statique, parfois même le blocage en conduction des transistors et finalement une perte de la fonctionnalité du circuit concerné.

Le phénomène de piégeage des trous concerne également les oxydes de champ. L'accumulation de charges positives piégées à proximité d'un substrat de type p est

susceptible de créer des canaux n entre les diffusions d'un même transistor, voire entre celles de deux transistors. Les interconnexions du circuit font alors office de grille, on se trouve en présence de structures transistor parasites. La figure 1 illustre ce phénomène au niveau de la partie de l'oxyde de champ proche d'un transistor, appelée bec d'oiseau.

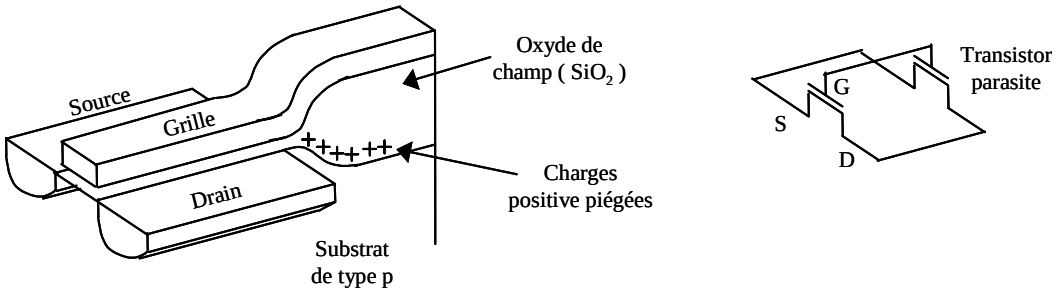


Figure 1 : Formation d'un transistor parasite à proximité du bec d'oiseau

La présence des charges positives piégées à proximité du substrat favorise la création d'un canal parasite de type n entre la source et le drain du transistor. Sur le dessin, le canal est commandé par la grille mais une interconnexion du circuit pourrait également jouer ce rôle. Les transistors parasites créés sont à l'origine de courants de fuite. Il en résulte une augmentation de la consommation du circuit pouvant aller jusqu'à un échauffement dommageable.

La notion de dose totale, ou dose absorbée, permet de quantifier l'importance de l'irradiation d'un composant. On la note D, c'est le rapport de l'énergie déposée par les radiations ionisantes (ΔE_d), par unité de masse (Δm) du matériau considéré :

$$D = \Delta E_d / \Delta m$$

L'unité courante est le Rad. Un ordre de grandeur de la dose absorbée par un circuit en orbite géostationnaire en dix années avoisine 50 krads.

Plusieurs approches de durcissement aux effets de dose ont été proposées. On peut citer : l'utilisation d'oxyde de silicium très purs (les impuretés favorisant le piégeage des trous et la création d'états d'interfaces), la réduction de l'épaisseur d'oxyde de grilles (ce qui va dans le sens de l'évolution technologique), la modification du dessin des transistors (comme la mise en place d'anneaux de garde réduisant les courants de court circuit), etc. Ainsi de nombreux fabricants proposent aujourd'hui des circuits intégrés pour les applications spatiales garantis pour fonctionner sans altération notable au delà d'une dose totale de 100 krads. Aussi nous

avons laissé de côté dans notre étude les effets de dose, ceux-ci nous semblant moins critiques que les autres effets radiatifs que nous allons maintenant détailler.

B.3 – Les effets singuliers.

On regroupera sous le terme " d'effets singuliers " les différents effets que peut avoir l'impact d'une unique particule radiative en un point d'un circuit intégré.
Nous nous attacherons à décrire les effets de Latch Up et les effets transitoires avant de préciser une hypothèse que nous avons retenue pour notre travail, celle de l'événement unique.

B.3.1. Le Latch Up.

Il existe dans les circuits intégrés CMOS des structures thyristor (PNPN) parasites. C'est l'amorçage de ces thyristors parasites induit par le passage d'une particule radiative dans le substrat (ion lourd ou proton) qui est appelé Latch Up. Il en résulte un fort appel de courant, potentiellement destructif pour le circuit, entre les alimentations.
La figure 2 met en évidence la structure du thyristor parasite d'un inverseur CMOS. Le 2.a donne la vue en coupe de l'inverseur et le 2.b la schématique d'assemblage des transistors bipolaires NPN et PNP et des résistances de substrat (Rs1, Rs2) et de puits (Rp1 et Rp2) associées à cette structure.

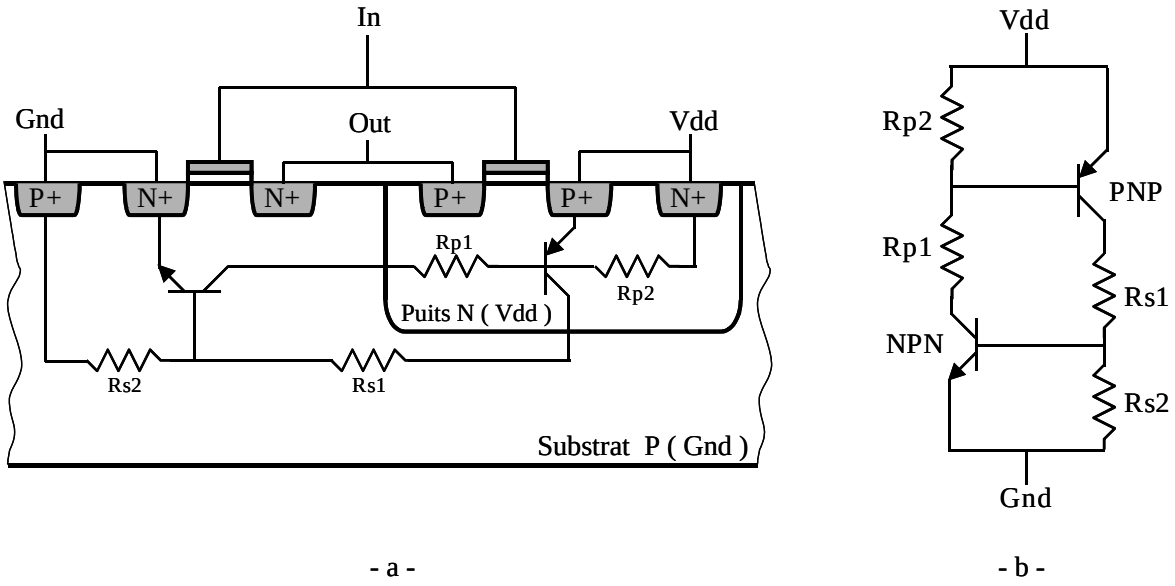


Figure 2 : Structure thyristor parasite d'un inverseur CMOS.

Nous avons vu que le passage d'une particule radiative crée le long de sa trajectoire des paires électron – trou. Ces charges créent des impulsions de courant susceptibles d'amorcer la

structure thyristor. Les effets destructifs se manifestent lorsqu'un état de verrouillage maintenu est atteint et permet la conduction d'un fort courant. C'est à dire lorsque :

- les transistors parasites sont polarisés en direct,
- le produit des gains des transistors est supérieur à un,
- le courant passant dans la structure dépasse le courant de maintien.

Plusieurs approches ont été proposées pour résoudre ce problème. Parmi celles-ci l'ajout de circuits de coupure de l'alimentation lorsqu'un fort courant est détecté, mais cette approche a l'inconvénient d'entraîner la perte de toutes les informations du circuit.
Il existe aussi des approches technologiques de durcissement au Latch Up, ainsi le recours à une technologie CMOS SOI, par exemple, élimine la structure thyristor parasite, etc.
Le phénomène de Latch Up reste néanmoins préoccupant et les familles de composants doivent être caractérisées rigoureusement avant toute utilisation en milieu radiatif.
Une étude plus détaillée de ce phénomène sort du cadre du présent travail.

B.3.2. Les effets transitoires.

Il s'agit plus exactement de transitoires de courant créés par le passage d'une particule ionisante à travers une zone sensible du circuit.

Dans la majorité des cas, les paires électron-trou générées se recombinent sans perturber le circuit. Mais en présence d'un champ électrique, les électrons et les trous sont balayés dans deux sens opposés et donnent naissance à ce que nous appellerons un courant d'aléa. Les champs électriques à l'origine de ce phénomène sont localisés au niveau des jonctions PN polarisées en inverse du circuit, plus exactement dans la zone de charge d'espace (ZCE) séparant les semi-conducteurs de type P et N. Par exemple, entre la diffusion dopée N+ d'un transistor NMOS et le substrat dopé P (la ZCE est plus étendue du côté du substrat P du fait d'un moindre dopage comparé à la diffusion dopée N+). La figure 3 illustre le mécanisme de collecte des charges à l'origine du courant d'aléa.

La trajectoire de la particule ionisante incidente est dessinée ainsi que les paires électron-trou générées. Cet apport de charges entraîne une extension en forme d'entonnoir (la zone de funneling sur la figure) de la ZCE. Les charges induites dans la zone préalablement déplétée et dans son extension (funneling) sont balayées quasi instantanément sous l'action du champ électrique et créent un pic de courant, c'est la composante instantanée de la figure 4.

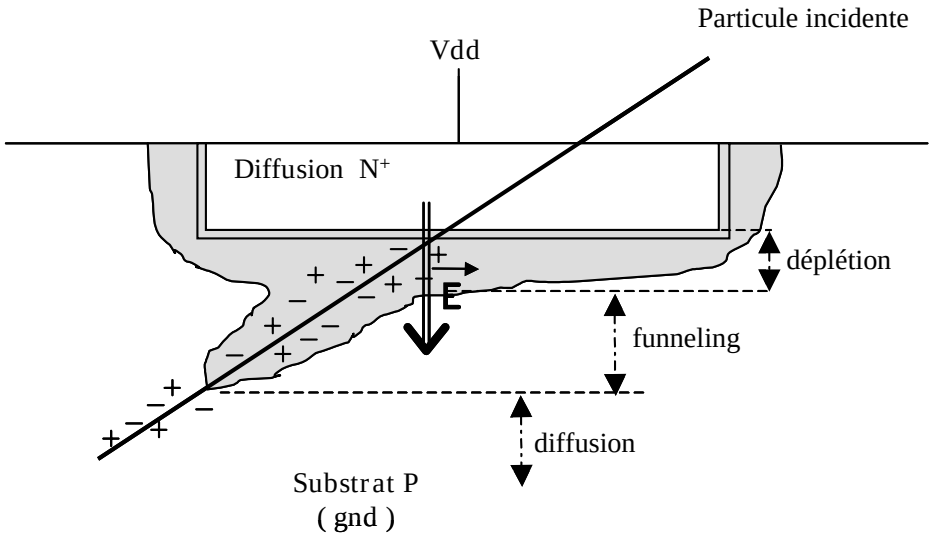


Figure 3 : Création d'un courant transitoire dans une jonction PN polarisée en inverse.

Le pic de courant peut atteindre une valeur Imax de quelques milliampères. Une partie des charges est également collectée par diffusion et contribue au courant d'aléa. Cet effet est moins rapide, il est à l'origine d'une composante retardée qui peut durer quelques centaines de picosecondes. La figure 4 donne une allure du courant d'aléa induit par le passage de la particule.

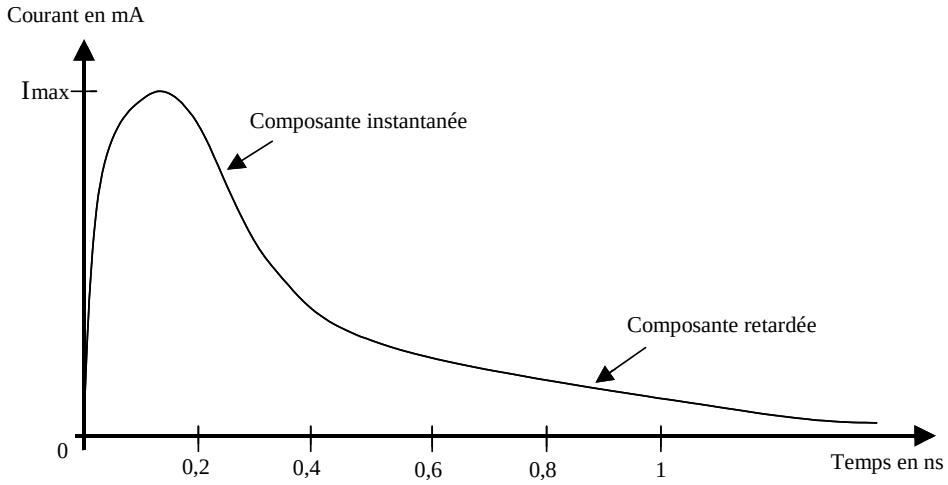


Figure 4 : Impulsion de courant générée par le passage d'une particule ionisante.

Son amplitude maximale et sa durée exacte dépendent de nombreux facteurs : le dopage des semi-conducteurs, l'énergie cédée par la particule, sa trajectoire, etc. On admet couramment une amplitude de quelques milliampères et une durée de quelques centaines de picosecondes.

Il est possible de simuler électriquement ce courant d'aléa avec une source de courant connectée entre la diffusion concernée et le substrat. L'impusion de courant que nous utiliserons par la suite est représenté figure 5.

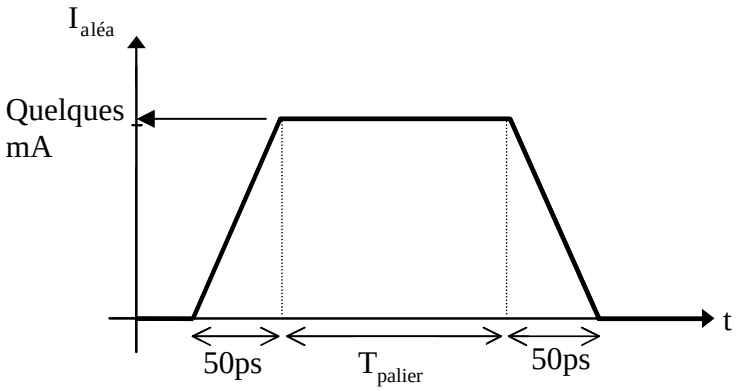


Figure 5 : Simulation du courant d'aléa.

Il s'agit d'une impulsion trapézoïdal avec des temps de montée et de descente de cinquante picosecondes et une durée de palier (Tpalier) programmable. Cette forme de courant d'aléa est relativement grossière mais elle permet d'étudier la susceptibilité des circuits aux effets transitoires.

Les zones sensibles des CI sont les jonctions PN polarisées en inverse. Elles sont nombreuses et leur localisation varie en fonction du niveau logique haut ou bas des nœuds du circuit. L'étude d'un inverseur CMOS réalisé sur un substrat de type P permet d'introduire l'étude de leur localisation de façon simple. Si l'entrée de l'inverseur est au niveau logique haut alors son NMOS est passant, son PMOS est bloqué et sa sortie au niveau bas. La source et le drain du NMOS (de type N+) ont un potentiel nul tout comme le substrat P, il n'y a pas de champ électrique caractérisant une zone sensible au niveau de ces diffusions. Le PMOS réalisé dans un puits N est lui bloqué. Sa source de type P+ est au potentiel haut comme le puits, elle n'est pas sensible. Par contre son drain de type P+ étant connecté à la sortie de l'inverseur est au même potentiel nul. La jonction PN drain-puits est donc polarisée en inverse, c'est une zone sensible. En suivant un raisonnement similaire dans le cas ou l'entrée de l'inverseur est au niveau bas, on en conclut la présence d'une zone sensible (jonction PN polarisée en inverse) au niveau du drain du NMOS bloqué.

Deux conclusions peuvent être tirées de cette rapide étude de l'inverseur :

- la localisation des zones sensibles varie en fonction des niveaux logiques,
- un inverseur possède une zone sensible localisée au niveau du drain du transistor bloqué.

La figure 6 illustre ces deux règles.

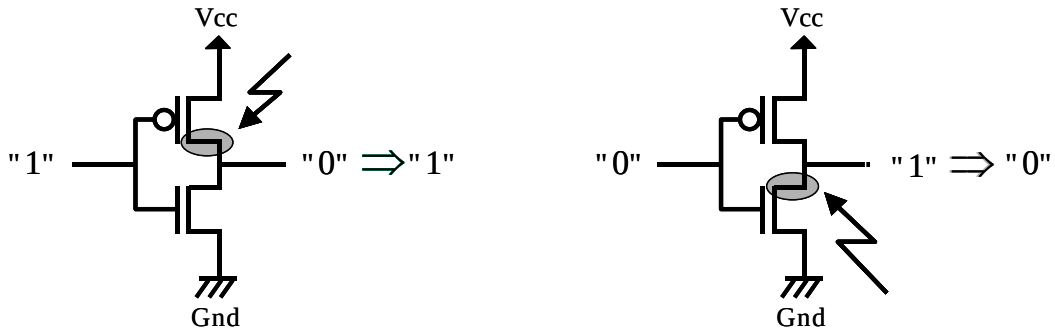


Figure 6 : Zones sensibles d'un inverseur.

Les zones sensibles sont localisées par les parties grisées. Dans le premier cas, si une particule ionisante traverse le drain du PMOS bloqué (flèche sur le dessin), il y a génération d'un courant d'aléa qui "tire" transitoirement le niveau logique de la sortie de l'inverseur vers le niveau haut. Dès la fin de la collection des charges générées par la particule, le courant d'aléa cesse et la sortie retrouve un niveau logique bas. Un phénomène de "guérison" se met en place, le NMOS est toujours passant, il permet la restauration du potentiel de la sortie. Réciproquement, sous l'action d'un courant d'aléa, la sortie d'un inverseur ayant un niveau logique haut en sortie peut être forcée à zéro pendant une période transitoire.

La figure 7 présente le résultat de la simulation électrique d'un transitoire sur un inverseur.

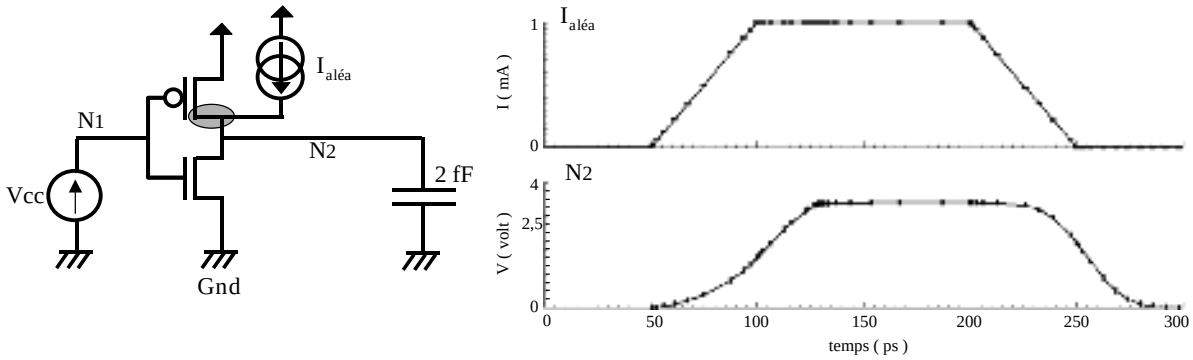


Figure 7 : Simulation d'un courant d'aléa sur le drain du NMOS d'un inverseur.

Sa sortie est connectée à un condensateur de capacité deux femtofarads et son potentiel d'entrée est fixé à Vcc. C'est donc le drain du PMOS qui est sensible. Le courant d'aléa (Ialéa) est simulé par une source de courant, l'impulsion de courant a une amplitude de un milliampère et un palier de cent picosecondes.
Le courant d'aléa impose le passage au niveau haut du potentiel du nœud de sortie N2 de l'inverseur. Mais dès qu'il cesse, la charge injectée est évacuée via le NMOS toujours passant.

La localisation des zones sensibles ne se réduit pas aux inverseurs. Considérons un circuit de technologie CMOS réalisé sur un substrat de type P polarisé à la masse et possédant un puits N polarisé à Vdd. Sur ce circuit, les NMOS sont gravés sur le substrat P et les PMOS dans le puits N. Dès lors, la présence d'un niveau logique haut (Vdd) sur le drain ou la source d'un NMOS polarise en inverse la jonction PN correspondante et similairement la présence d'un niveau logique bas sur le drain ou la source d'un PMOS polarise la jonction PN correspondante en inverse. Ainsi d'une façon plus générale :

- la localisation des zones sensibles varie en fonction des niveaux logiques des nœuds du circuit,
- le drain (ou la source) d'un NMOS est sensible au niveau haut,
- le drain (ou la source) d'un PMOS est sensible au niveau bas.

En conclusion , un transitoire de courant généré par le passage d'une particule ionisante induit un transitoire de tension au niveau du nœud connecté à la zone sensible. Ce transitoire est susceptible de se propager dans les parties logiques du circuit et d'être à l'origine d'erreurs logiques. Les mécanismes d'apparition de ces erreurs sont détaillés dans la partie 2.3.

B.3.3. L'hypothèse de l'événement unique.

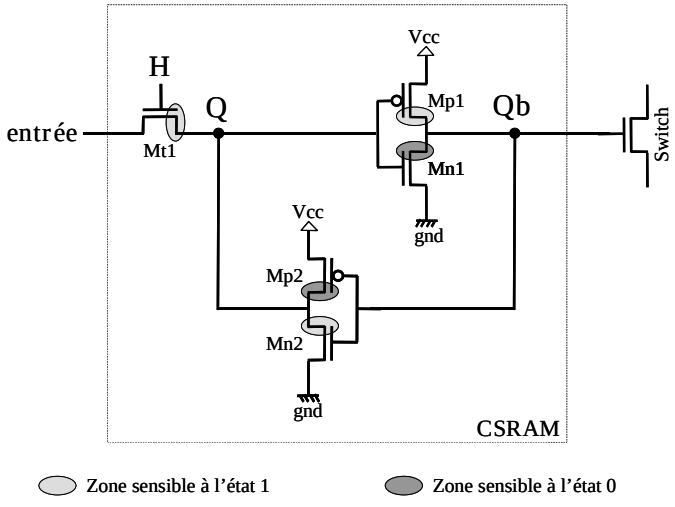
Les effets singuliers ont un caractère statistique. Leur occurrence dépend de facteurs non déterministes liés aux phénomènes radiatifs (flux des particules radiatives, énergie, localisation de l'impact, etc.). Ils apparaissent de façon irrégulière et aléatoire. Nous ferons l'hypothèse qu'ils sont suffisamment espacés dans le temps pour que l'on puisse considérer qu'un seul effet transitoire peut se produire simultanément. C'est cette hypothèse de l'événement unique qui nous permettra de proposer dans la troisième partie des solutions de durcissement à leurs effets.

II.C. Radiations et FPGA.

Cette partie traite des effets des évènements singuliers (aléas logiques et transitoires) sur les circuits reconfigurables. Selon leurs conséquences, on peut classer ces effets en deux familles. Une première famille regroupe les effets propres aux seuls circuits reconfigurables à base de SRAM. Elle concerne principalement les aléas logiques dans la couche de configuration. Et une deuxième famille d'effets singuliers, plus générale, est consacrée à leurs conséquences communes aux FPGA et à l'ensemble des circuits intégrés opérant en milieu radiatif. Le principal objectif de cette partie est d'identifier les différentes faiblesses des FPGA et de hiérarchiser leur importance.

C.1 – Les effets singuliers de la couche de configuration.

La principale caractéristique des circuits reconfigurables est la présence d'une couche de configuration, décrite au II.A.1.2 Son rôle est de mémoriser la fonctionnalité du circuit (fonctions logiques programmées, routage des interconnexions, etc.). Cette couche de configuration est constituée de points mémoire, les CSRAM (Configuration Static Random Access Memory), réalisées avec cinq transistors. Ils possèdent plusieurs zones sensibles à l'impact d'une particule radiative, dont la localisation varie en fonction du niveau logique des nœuds du circuit. La figure 1 rappelle l'architecture d'une CSRAM et donne l'emplacement de ces différentes zones sensibles en fonction de l'état du point mémoire. Nous définissons cet état par le niveau logique du nœud Q, ainsi nous considérons que le point mémoire est dans l'état 1 quand Q=1 avec Qb=0, et respectivement à l'état 0 quand Q=0 avec Qb=1.



	W (µm)	L (µm)
Mt1	1,5	0,25
Mn1	1	0,25
Mn2	0, 5	0,25
Mp1	1,7	0,25
Mp2	0,85	0,25

Tableau 1

Figure 1 : Zones sensibles d'une CSRAM (Etat défini par le niveau de Q).

On localise rapidement les zones sensibles des deux inverseurs (les drains des transistors bloqués) en appliquant les règles vues au paragraphe B.3.2 de la partie II. Il y a également une zone sensible à l'état 1 au niveau de la jonction Q du transistor d'accès Mt1 (NMOS avec le potentiel haut sur sa diffusion). On se limite ici au cas où la CSRAM est en mode mémoire, le transistor d'accès Mt1 est bloqué (H=0). Dès lors, si un événement radiatif entraîne l'inversion du bit de configuration du point mémoire on dira qu'il s'est produit un aléa logique statique. Par opposition, on parlera d'aléa logique dynamique si l'inversion se produit au moment de l'écriture, ce cas sera étudié en détail dans la partie C.2.3. Le choix de restreindre, dans un premier temps, l'étude au mode mémoire est motivé par le fait que la durée de la phase d'écriture des CSRAM est marginale lors du fonctionnement d'un FPGA. De plus, par construction, le nœud d'entrée des CSRAM est peu sensible aux aléas car il possède une capacité importante (nous verrons dans la partie III.A qu'il s'agit d'une méthode de durcissement parfois utilisée).

La simulation électrique de la réponse d'un point mémoire à un transitoire de courant induit par le passage d'une particule radiative permet de comprendre très simplement le mécanisme d'inversion de son état logique.

Pour réaliser les simulations qui vont suivre, un layout de la CSRAM de la figure 1 a été dessiné avec les tailles de transistors données par le tableau 1. La technologie choisie est une technologie commerciale CMOS 0,25 µm (DKhcmos7 de la société ST Microelectronics) et les capacités parasites ont été extraites. Considérons que la CSRAM de la figure 1 est dans l'état 1 (Q=1 avec Qb=0) en mode mémoire (H=0). Le nœud Qb est alors sensible au niveau du drain du PMOS Mp1, il est susceptible de subir un transitoire de courant entraînant un passage de son potentiel vers le niveau haut. On simule alors un aléa en injectant une impulsion de courant de forme triangulaire avec des temps de montée et de descente égaux à 50 ps et une intensité maximale de 2 mA. Le résultat de cette simulation est présenté figure 2.

On peut suivre, pas à pas, le mécanisme de l'aléa sur les courbes. Sous l'action du courant d'aléa le potentiel du nœud Qb passe du niveau logique bas au niveau logique haut. Dès lors, le deuxième inverseur, commandé par Qb, impose à son tour une transition de sa sortie, le nœud Q, du niveau haut vers le niveau bas. Le potentiel du nœud Q étant désormais à sa valeur basse, il impose un niveau logique haut sur la sortie du premier inverseur : le nœud Qb. Ainsi, même lorsque le courant d'aléa cesse, le nœud Qb conserve un niveau logique haut.

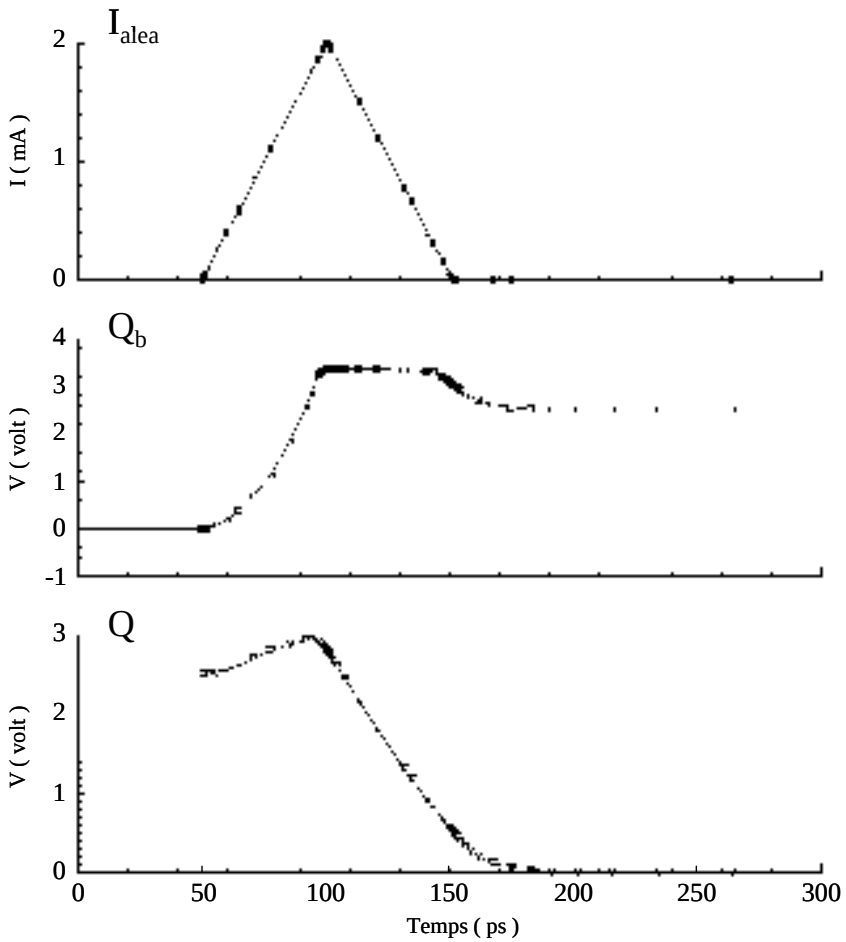


Figure 2 : Simulation d'un aléa sur le nœud Qb.

L'aléa s'est propagé dans le point mémoire, celui-ci a basculé, il est maintenant "verrouillé" dans un nouvel état stable : l'état 0 (Q=0 avec Qb=1). Le bit d'information stocké par la CSRAM a été inversé, elle a subi un aléa logique statique.

La simulation précédente correspond au cas où la particule radiative est suffisamment énergétique pour générer un courant induit susceptible d'entraîner un aléa logique. Ce n'est pas toujours vrai. En effet, pour un courant induit plus faible, la perturbation de potentiel subie par Qb peut être insuffisante pour provoquer un basculement définitif de la CSRAM. La figure 3 illustre cette possibilité, pour un courant d'aléa dont l'intensité maximale a été abaissée à 1,2 mA.

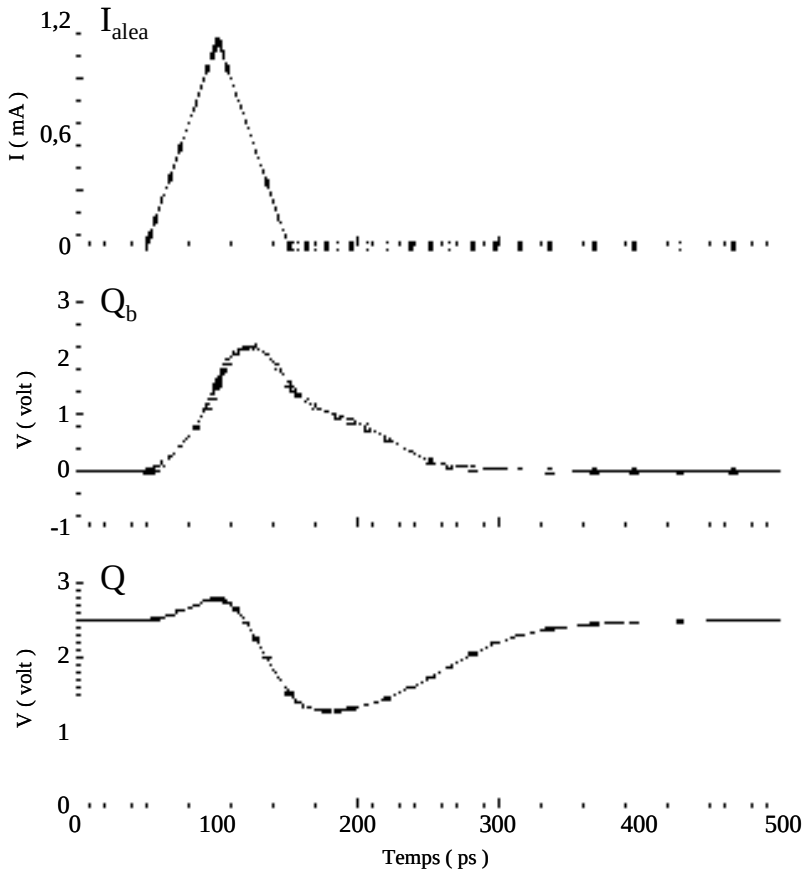


Figure 3 : Simulation d'un aléa non générateur d'erreur sur le nœud Qb.

Dans ce deuxième cas le courant d'aléa n'est pas suffisamment important pour entraîner le basculement du point mémoire. La croissance du potentiel du nœud Qb est plus lente que dans le cas précédent, ce qui retarde d'autant la chute de potentiel du nœud Q commandée par le deuxième inverseur. La CSRAM n'a pas le temps de basculer avant la fin de l'aléa (100 ps), elle retrouve finalement son niveau logique après correction des perturbations.

L'intensité maximale choisie (1,2 mA) correspond au cas limite en deçà duquel il n'y a pas occurrence d'un aléa logique. Pour un circuit de caractéristiques données, cette valeur critique au delà de laquelle il y a basculement de la CSRAM dépend de la durée et de la forme du courant d'aléa. Pour des durées différentes d'injection du courant d'aléa, cette amplitude critique varie, c'est donc une valeur difficile à manipuler pour rendre compte de la sensibilité aux aléas logiques d'un nœud particulier. Il est plus simple de considérer une notion introduite pour la première fois par [DOD95], la notion de charge critique. Elle correspond à la partie de la charge déposée par une particule radiative qui est collectée par le nœud sensible. Cette fraction de la charge est à l'origine du courant d'aléa. Elle est calculée en intégrant le courant d'aléa sur sa durée :

$$Q_{critique} = \int_{début\ aléa}^{fin\ aléa} I_{alea}(t)\,dt$$

La charge critique prend en compte la forme du courant d'aléa, sa durée et son intensité.
Lorsque la charge Q à l'origine du courant d'aléa est inférieure ou égale à la charge critique $Q_{critique}$ il n'y a pas apparition d'un aléa logique, au contraire, lorsque la charge Q est supérieure à la charge critique il y a occurrence d'un aléa.
Ainsi, dans le cas d'un aléa vers 1 sur le nœud Qb de la CSRAM précédemment analysée on obtient une charge critique $Q_{critique}$ = 0,06 pC. Cette valeur permet principalement de comparer la sensibilité du nœud Qb avec les sensibilités respectives des autres nœuds du point mémoire. Le tableau 2 donne les charges critiques de tous les nœuds de la CSRAM.

	Aléa vers 1	Aléa vers 0
Q	0,03 pC	0,025 pC
Qb	0,06 pC	0,045 pC

Tableau 2 : Charge critique des différents nœuds

Pour un nœud donné les simulations électriques effectuées ne montrent pas de différence notable entre les aléas vers 1 (PMOS) et les aléas vers 0 (NMOS). Les PMOS semblent légèrement moins sensibles. La sensibilité relative des nœuds Q et Qb varie, elle, du simple au double. Le nœud Qb est moins sensible du fait d'une largeur de grille plus importante des transistors du premier inverseur (Mp1 et Mn1). Le courant de son transistor passant permet de compenser une plus grande partie du courant d'aléa injecté, avec comme effets de limiter l'évolution du potentiel de Qb et de raccourcir le transitoire de tension correspondant.

La notion de charge critique a été développée dans le cadre de simulations en trois dimensions plus proche de la réalité physique qu'une simulation des courants d'aléa au moyen d'une source de courant. Une telle modélisation ne saurait donc être remise en cause par des simulations purement électriques. Or, nous avons effectivement constaté pour des impulsions de courant d'amplitude et de durée différentes mais correspondants à une même charge, l'occurrence ou non d'un aléa logique. Ce résultat met en évidence les limitations de la modélisation du phénomène d'aléa avec une source de courant. Nous devrions donc chercher un modèle plus approprié. Cependant, le modèle "source de courant" permet de comparer

simplement la susceptibilité aux aléas de différents noeuds. Nous continuerons donc à l'utiliser, en gardant à l'esprit ses limites.

La couche de configuration des circuits reconfigurables à base de CSRAM est susceptible de subir des aléas logiques. L'inversion d'un bit de configuration peut suffire à modifier la fonctionnalité du circuit. Pour la rétablir, il est alors nécessaire de mettre le FPGA hors fonction et de charger à nouveau la configuration. Le grand nombre de CSRAM de la couche de configuration augmente d'autant les risques.

C.2 – Vulnérabilité de la couche logique aux effets singuliers.

La partie précédente a mis en évidence le phénomène de l'aléa logique d'un point mémoire, dû à la propagation d'une perturbation électrique à travers ses inverseurs. Nous allons voir maintenant que de telles perturbations peuvent se propager le long d'un nombre d'inverseurs bien plus grand et à travers la logique combinatoire ; on parlera alors de la propagation de transitoires de tension ou SET (Single Event Transient).

Ces transitoires affectent les différents éléments présents dans les FPGA : les arbres d'horloge, les interconnexions, les tables d'allocation et les multiplexeurs. Un transitoire, c'est à dire la perturbation momentanée d'un niveau logique, n'a une incidence sur le fonctionnement du circuit que si l'erreur propagée atteint une entrée asynchrone (l'horloge, la remise à zéro d'une bascule par exemple) ou bien est enregistrée par une flip-flop. Dans ce dernier cas, on parlera d'aléa logique dynamique.

C.2.1. Les arbres d'horloge.

La migration vers les technologies submicroniques permet à des transitoires, autrefois filtrés par les temps de traversée des portes logiques, de se propager dans les circuits [BAZ97] [MAV00]. Dans le cas d'un arbre d'horloge, les conséquences de l'apparition d'un transitoire sont particulièrement importantes, car celui-ci se propage dans les branches situées en aval de son point d'apparition. La figure 4 représente la structure d'arbre d'horloge que nous avons utilisée pour mettre en évidence ce phénomène.

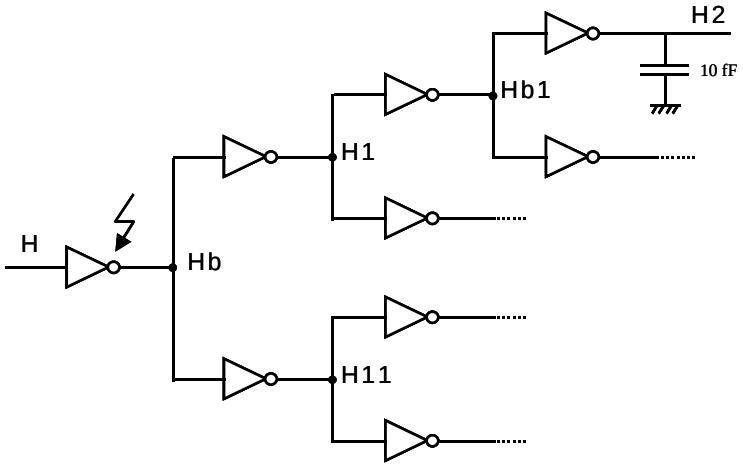


Figure 4 : Arbre d'horloge.

Une simulation électrique[1] a été effectuée pour une fréquence d'horloge de 200 MHz et un transitoire injecté au niveau du nœud Hb (flèche brisée) par une impulsion de courant (I_{alea}) d'amplitude 2 mA, de temps de montée et de descente de 50 ps et un plateau de 100 ps. La figure 5 met en évidence la propagation du transitoire le long de la branche supérieure de l'arbre d'horloge.

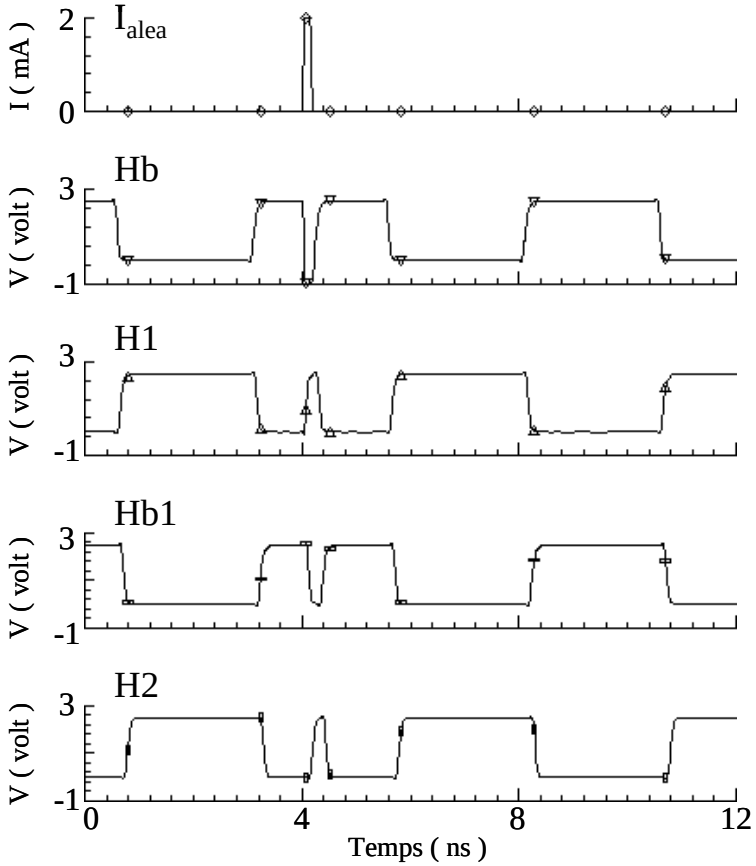


Figure 5 : Propagation d'un transitoire le long de l'arbre d'horloge.

Le transitoire se propage sans atténuation le long de la chaîne d'inverseurs, de Hb à H2 via H1 et Hb1, et c'est également le cas de toutes les branches en aval de Hb. Mavis et Al. montrent qu'un transitoire d'une durée de l'ordre de 100 ps se propage à travers une chaîne d'inverseurs de technologie 0,25 µm sans être atténué [MAV00].

Le fonctionnement de toute la partie du circuit commandée par les branches en aval de Hb va avoir son fonctionnement perturbé (non respect des temps de pré-positionnement et de maintien, fronts d'horloge inattendus, etc.) et va générer des erreurs.

[1] La simulation électrique a été effectuée pour une technologie CMOS 0,25 µm (Dkhcmos7 de la société ST Microelectronics) après extraction des capacités parasites.

Les circuits programmables comptent aussi un grand nombre d'interconnexions. Celles-ci, particulièrement les interconnexions à longue distance, possèdent plusieurs buffers. La vulnérabilité de ces buffers est similaire à celle des inverseurs. Les interconnexions des FPGA sont donc elles aussi très sensibles à la propagation des transitoires. Les conséquences en sont différentes selon qu'elles aboutissent, ou non, à des entrées asynchrones. Dans le cas de la commande d'une remise à zéro (asynchrone), par exemple, l'apparition d'une erreur sera immédiate. En revanche dans le cas où l'interconnexion aboutit à une partie logique, les conséquences en seront diverses comme nous le montrons dans les parties suivantes.

C.2.2. Multiplexeurs et tables d'allocation.

La partie logique des FPGA est principalement constituée de multiplexeurs et de tables d'allocation. Ces sous-circuits sont susceptibles de donner naissance à des erreurs transitoires et de les propager.

Considérons le cas d'un multiplexeur à quatre entrées (MUX4) dont l'architecture au niveau transistor est rappelée figure 6. Les zones sensibles aux aléas (indifféremment pour les aléas vers 1 ou vers 0) ont été indiquées pour une configuration particulière des niveaux d'entrée (E0, E2 et E3 à la masse et E1 à vdd) et des deux bits de configuration (C0 à vdd et C1 à la masse).

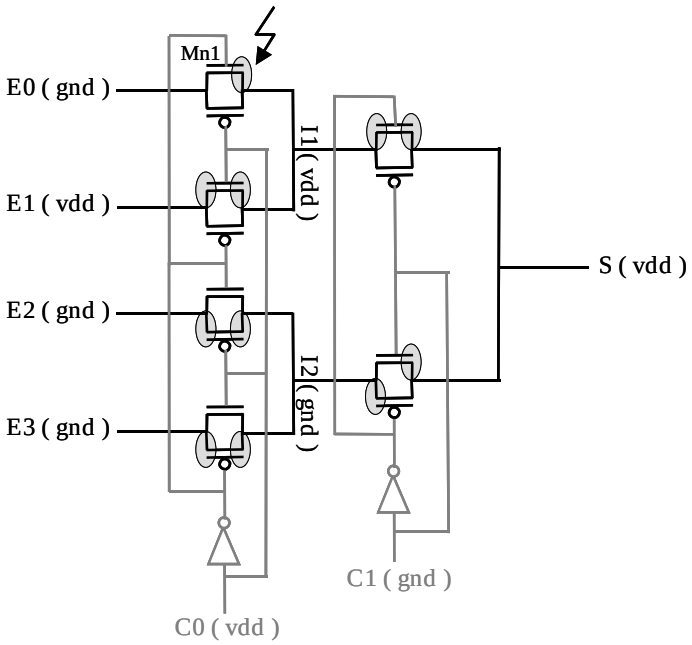


Figure 6 : Localisation des zones sensibles du MUX4 pour un état donné.

Le résultat de la simulation électrique[1] d'un transitoire au niveau de la jonction sensible du transistor Mn1 (flèche brisée de la fig. 6) est représentée figure 7, pour un transitoire de courant (I_{alea}) d'amplitude 2 mA, de temps de montée et de descente 50 ps pour un plateau de 100 ps.

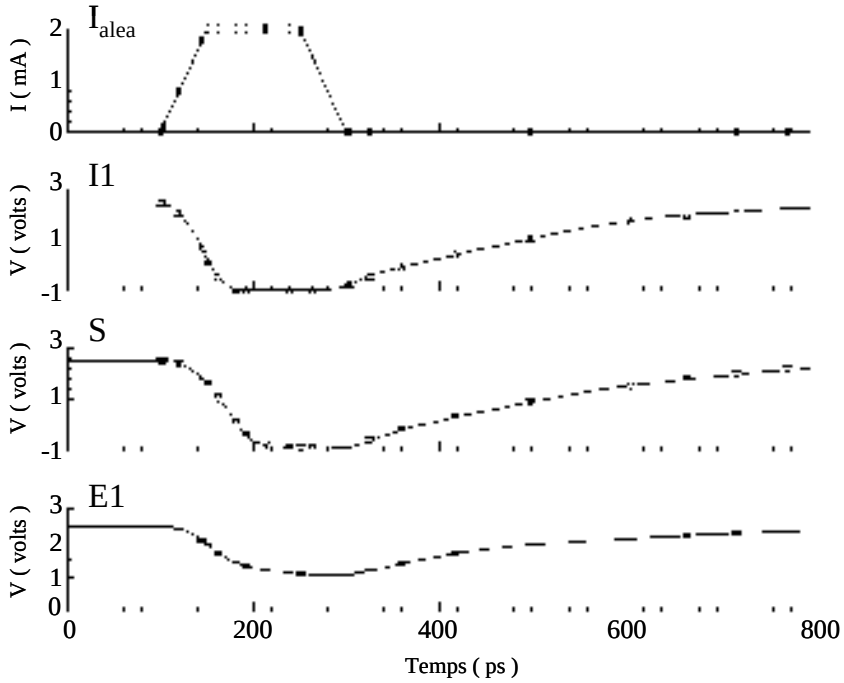


Figure 7 : Propagation d'un transitoire dans un MUX4.

La tension du nœud I1 passe du niveau haut au niveau bas sous l'effet de l'aléa. Et le transitoire se propage jusqu'au nœud de sortie S, dont le niveau logique subit une transition pleine d'échelle du niveau haut au niveau bas. Le niveau logique correct (haut) n'est restauré qu'environ 400 ps après la fin du courant d'aléa. A noter que le potentiel du nœud E1 est lui aussi perturbé, les transitoires se propagent aussi bien vers l'amont que vers l'aval à travers les "pass gates".

L'architecture des tables d'allocation est similaire à celle des multiplexeurs. On peut les considérer comme des multiplexeurs dont les entrées de données sont connectées directement à des CSRAM et les entrées de sélection à la logique en amont. Elles sont donc tout aussi susceptibles que les multiplexeurs de donner naissance à des erreurs transitoires et de les propager.

Mais les transitoires que nous venons d'étudier sont aussi à l'origine d'erreurs permanentes quand ils atteignent des entrées asynchrones ou bien dans certaines conditions lorsqu'ils parviennent aux entrées des flip-flops. Nous présentons ce dernier cas ci-après.

C.2.3. Flip-flops.

Dans les circuits programmables, on rencontre de nombreuses bascules du type Maître - Esclave, comme les D flip-flops. Elles sont, en général, situées entre deux blocs logiques (multiplexeurs, tables d'allocation, etc.). Leur rôle est de mémoriser l'information le temps nécessaire à son traitement (une période d'horloge). Les valeurs mémorisées forment une image de l'état logique du circuit à un instant donné. Ainsi, si l'une d'entre elles subit une erreur, le circuit passe dans un état logique erroné et il peut s'écouler un temps important avant le rétablissement d'un état correct.

Le schéma de principe d'une flip-flop simplifiée (sans présélection et sans remise à zéro) commandée par les fronts montant de l'horloge H est donné figure 8. Elle est constituée de deux bascules D (le maître et l'esclave) et d'un inverseur.

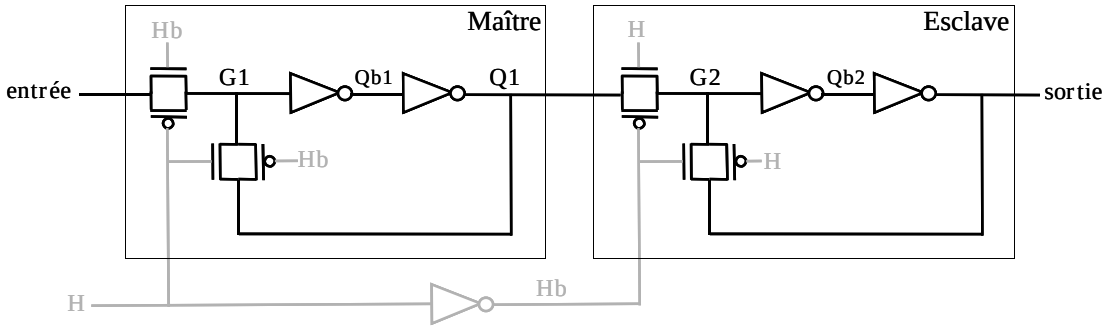


Figure 8 : Schéma de principe d'une flip-flop.

La susceptibilité des flip-flops aux aléas a deux origines : les bascules D qui la composent et la logique qui les entoure. On distingue trois types d'aléa : l'aléa logique statique, il correspond au moments où l'horloge est stable ; l'aléa logique dynamique, il se manifeste lors des fronts d'horloge ; et les transitoires de tension sur l'horloge.

L'aléa logique statique concerne directement les bascules D constituant la flip-flop. Il se produit lorsque la bascule D est en mode mémoire (transistors d'accès bloqués et transistors de la boucle de rétroaction passants), c'est le cas de l'esclave pour H=0 et le cas du Maître pour H=1. Une bascule D a une architecture un peu plus complexe que celle d'un point mémoire mais elle est basée sur le même principe : l'utilisation de deux inverseurs rebouclés. Ainsi, le mécanisme de l'aléa statique est quasi identique à celui décrit au II.C.1.

L'aléa logique dynamique est lié aux transitions de l'horloge ou plus exactement à la conjonction d'un transitoire et d'un front d'horloge. Le C.2.2. a mis en évidence la propagation des transitoires dans les parties logiques et les interconnexions d'un circuit. Ces transitoires lorsqu'ils atteignent une flip-flop peuvent entraîner, ou non, une altération du bit enregistré. Ainsi, on appelle aléa logique dynamique, la mémorisation d'un bit de donnée incorrect par la flip-flop, due à l'arrivée d'un transitoire sur son entrée. Pour les bascules D, cela se produit si le transitoire atteint l'entrée au moment du passage de l'horloge du niveau bas au niveau haut (passage du mode transparence au mode mémoire). Un aléa dynamique dépend donc étroitement de l'instant où le transitoire atteint la bascule, on appelle fenêtre de vulnérabilité l'intervalle de temps pendant lequel un transitoire peut induire la mémorisation d'une erreur. Sa durée correspond à peu près à la somme des temps de pré-positionnement et de maintien des données de la bascule.

Pour illustrer la dépendance entre l'instant d'arrivée du transitoire et le front d'horloge, un circuit constitué d'une table d'allocation de deux variables connectée à l'entrée d'une flip-flop a été réalisé. Deux transitoires de tension vers le niveau haut créés par un générateur d'impulsion de courant et caractérisés par une amplitude de 2 mA, des temps de montée, de descente et de plateau de 50 ps, ont été injectés par simulation au sein de la table d'allocation de façon à atteindre l'entrée de la flip-flop : le premier à un instant t =1,3ns et le second à l'instant t =3,8ns. Les résultats de la simulation électrique sont donnés figure 9.

Dans le premier cas, l'aléa atteint l'entrée de la flip-flop pour H=0. La bascule Maître est en mode transparence. Elle transmet la perturbation jusqu'à sa sortie,Q1, qui est aussi l'entrée de la bascule Esclave. Cette dernière étant en mode mémoire (Hb=1), son état logique n'est pas altéré par le transitoire, elle conserve donc l'état 1. Les effets du transitoire disparaissent avant l'apparition du front montant de l'horloge H. Celui-ci passé, le Maître délivre à nouveau un niveau logique bas en entrée de l'esclave. Survient alors le front montant de H, le Maître passe en mode mémoire, à l'état 0, et l'Esclave en mode transparence, il transmet donc un niveau logique bas sur la sortie de la flip-flop. Il s'agit bien de la valeur attendue pour un fonctionnement correct sans erreur. Le transitoire était trop éloigné du front de l'horloge pour induire un aléa logique dynamique, il n'a eu aucun effet sur le fonctionnement normal du circuit.

Dans le deuxième cas, l'arrivée du transitoire sur le nœud d'entrée de la flip-flop est proche du front montant de l'horloge. Sous l'effet de la perturbation, le nœud Q1 est passé au niveau haut lorsque le front montant de l'horloge se produit. Le Maître mémorise l'état logique erroné 1 au lieu de l'état logique 0 correspondant au fonctionnement correct. On

retrouve un niveau haut en sortie de la flip-flop (l'Esclave est passé en mode transparence). Le transitoire a été capturé et mémorisé par la bascule Maître, il y a eu aléa logique dynamique.

Le phénomène de l'aléa dynamique est aussi susceptible de se produire sur les fronts descendants de l'horloge. Dans ce cas c'est l'esclave qui mémorise le transitoire créé en amont au niveau de la logique ou du Maître.

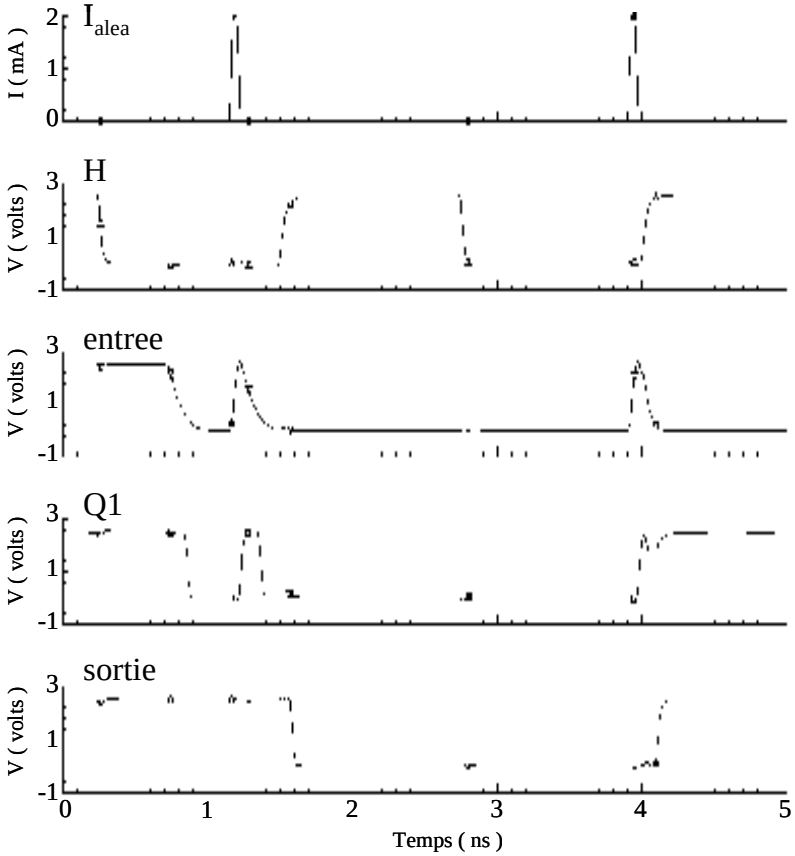


Figure 9 : Simulation d'aléas dynamiques.

Les conditions d'apparition d'un aléa logique dynamique sont délicates à cerner. Elles nécessitent la simultanéité d'un front d'horloge et d'une propagation de transitoire jusqu'à l'entrée d'une bascule. Ce phénomène pouvait être considéré comme peu probable pour des technologies microniques. Mais le passage au submicronique a favorisé d'une part la propagation et l'augmentation du nombre des transitoires et d'autre part un accroissement des fréquences d'horloge. La multiplication du nombre de fronts d'horloge augmente la fréquence d'apparition des conditions d'aléa dynamique [BUZ97]. Aussi, le nombre d'aléas dynamiques augmente avec la fréquence de fonctionnement d'un circuit alors que le nombre d'aléas statiques reste stable.

Les transitoires de tension sont également susceptibles de générer des erreurs de fonctionnement par l'intermédiaire de l'horloge des D flip-flops. Lorsqu'une bascule D est en mode mémoire, tout transitoire sur son horloge entraîne son passage en mode écriture pendant toute sa durée. Dès la fin de celui-ci, elle retourne en mode mémoire. Mais elle mémorise désormais la valeur logique présente sur son entrée pendant le transitoire, cette valeur peut être différente de celle précédemment mémorisée. Un transitoire de tension sur l'horloge est donc susceptible d'altérer l'état des bascules. Les simulations électriques suivantes illustrent ce mécanisme.

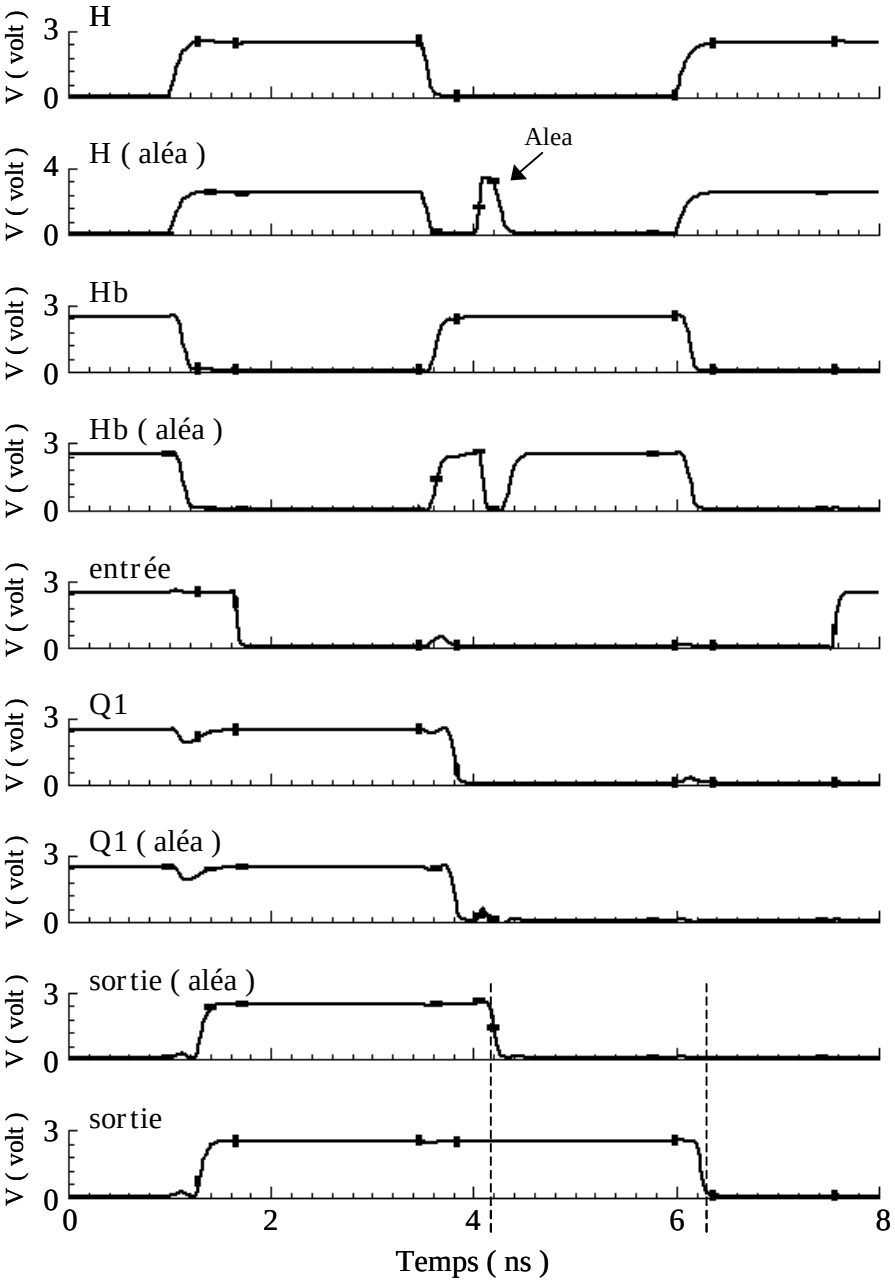


Figure 10 : Effets d'un transitoire de tension sur l'horloge

Les courbes comparatives de la figure 10 présentent successivement le cas du fonctionnement normal et le cas d'un transitoire de tension vers le niveau haut sur l'horloge H (simulé par injection d'un courant : amplitude 2 mA, temps de montée et de descente 50 ps, et durée du plateau 100 ps).
Au moment où le transitoire débute, la bascule Maître est en mode écriture avec Q1=0 tandis que la bascule Esclave est en mode mémoire, ici à l'état 1. Dans le cas d'un fonctionnement normal, la bascule Esclave conserve l'état 1 jusqu'à son passage en mode écriture à l'instant t=6ns dès que H=1. Mais la simulation montre que si un transitoire de tension vers le niveau haut en H est présent à l'instant t=4ns, l'Esclave passe prématurément en mode écriture et à l'état 0. Dès la fin du transitoire, il repasse en mode mémoire, mais, il mémorise désormais l'état 0. Les lignes en pointillées de la figure 10 pointent cette différence. Le transitoire a imposé une commutation prématurée de la sortie vers le niveau bas. Il en résulte un non respect des contraintes temporelles au risque de provoquer la propagation d'une erreur vers l'aval du circuit.

Nous venons de décrire les mécanismes qui conduisent à l'apparition d'erreurs de fonctionnement dans les flip-flops de la couche logique des FPGA utilisés en milieu radiatif. La conséquence de ces erreurs est la délivrance de résultats faux en sortie et une fiabilité de fonctionnement douteuse de ces flip-flops. Le nombre important de flip-flops justifie donc une opération de durcissement systématique afin de pouvoir envisager une utilisation en milieu hostile.

C.3 - Conclusion

La partie II.C a mis en évidence les principales conséquences des événements singuliers sur les circuits reconfigurables. Deux grandes familles d'effets se dégagent.

La première se distingue par des aléas apparaissant dans la couche de configuration des circuits. C'est le point de blocage essentiel à l'utilisation des FPGA à base de SRAM en milieu radiatif. Le grand nombre de CSRAM et leur sensibilité élevée aux aléas rend très probable une altération de la configuration initiale du circuit. La perte de fonctionnalité impose alors une reprogrammation du circuit ; qui ne sera plus fonctionnel avant qu'elle ne soit effectuée. Il s'agit ici d'erreurs non destructives ; une fois reprogrammé le circuit devrait être à nouveau fonctionnel. Wang et Al. [WAN99] ont cependant mis en évidence la possibilité d'apparition de micro court-circuits, potentiellement destructifs, au sein des FPGA.

Leur apparition est liée à une modification des interconnexions due à l'inversion de bits de configuration.
De par sa couche de configuration, un circuit reconfigurable non durci ne peut donc présenter une fiabilité d'utilisation satisfaisante dans un contexte radiatif.

La deuxième famille d'effets se rapproche des effets classiques rencontrés dans tous les circuits intégrés, avec une spécificité propre à l'architecture des FPGA (interconnexions nombreuses, recours aux tables d'allocation, etc.). Les événements singuliers dans la couche logique peuvent ne pas nécessiter de reprogrammation du composant pour retrouver un fonctionnement correct et ils n'ont pas d'effets destructifs. Dans ce cas, les dysfonctionnements se limitent à l'apparition de données erronées pendant un temps plus ou moins long. Mais dans le cas d'architecture du type machines d'état, un fonctionnement correct peut ne jamais se rétablir, après passage dans un état interdit, par exemple. Si cette possibilité n'a pas été prise en compte par le concepteur, il deviendra nécessaire de procéder à une réinitialisation de la machine d'état.

Si tous ces défauts potentiels doivent être corrigés, en fonction de la fiabilité attendue du FPGA, ceux qui sont liées à une modification de la configuration doivent l'être impérativement. Le chapitre suivant traite des stratégies de durcissement envisagées.

III

Durcissement des FPGA

III.A. Les solutions actuelles.

Cette partie est consacrée à la présentation des solutions récentes proposées pour " durcir " les circuits intégrés aux aléas logiques et à la propagation des transitoires de tension. Dans un premier temps, nous rappelons les approches générales de durcissement sans considération de la nature du CI concerné. Certaines de ces solutions basées sur des approches technologiques ou de conception peuvent être appliquées aux FPGA. Nous présentons ensuite les techniques de durcissement retenues par les fabricants de FPGA. Et enfin, nous concluons sur les approches que nous proposons et qui sont décrites dans la partie III.B.

A.1 – Les approches générales en durcissement des circuits intégrés.

Nous avons décrit dans les parties précédentes les mécanismes physiques de collection des charges générées par les particules radiatives et la notion de charge critique liée à l'apparition d'un effet singulier. Le durcissement d'un circuit peut être amélioré en diminuant la charge collectée ou bien en augmentant la charge critique nécessaire à l'apparition d'une erreur.

La première approche correspond à un durcissement physique des CI en ayant recours à des technologies de fabrication plus complexes que le process CMOS standard.

La deuxième approche consiste en un durcissement des CI en faisant évoluer leur architecture au niveau des transistors, des cellules de base ou du circuit lui même.

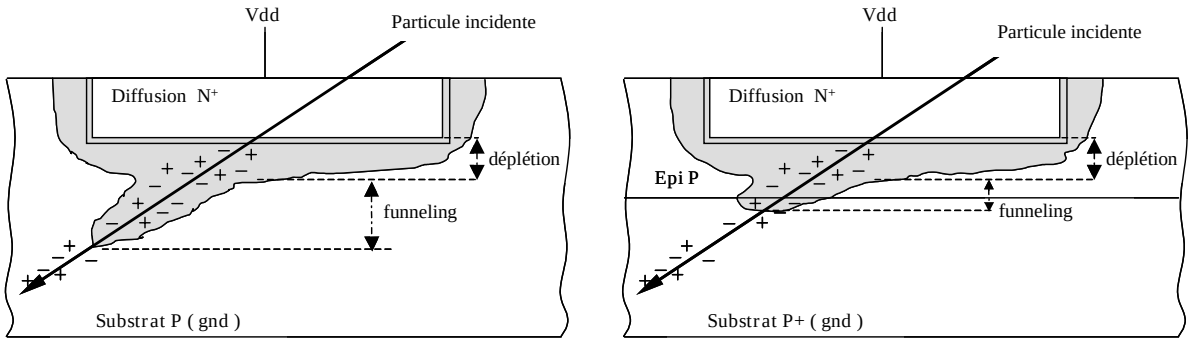
A.1.1. Le durcissement technologique des circuits intégrés.

Le recours à des technologies CMOS plus complexes est qualifié de durcissement " technologique " des CI. Afin de limiter la charge collectée par les nœuds sensibles du circuit, des technologies spécifiques épi CMOS, CMOS SOI et CMOS SOS peuvent être utilisées (ces technologies apportent également des améliorations aux effets de dose et de Latch Up, mais ces aspects sortent du cadre de notre étude).

Epi – CMOS

Une réduction de la charge collectée par le phénomène dit de " funneling " peut être obtenu en limitant l'extension de la zone de charge d'espace. Cette méthode entraîne une réduction de la charge totale collectée au niveau des diffusions sensibles. Le recours à la technologie Epi – CMOS permet d'y parvenir. Les transistors sont alors gravés sur une couche mince épitaxiée peu dopée par rapport à un substrat fortement dopé. La figure 1 met

en évidence cette réduction de l'extension de la ZCE autour d'une diffusion dopée N+ lors du passage de la technologie CMOS classique (fig. 1.a) à la technologie Epi – CMOS (fig. 1.b).



- a - - b -

Figure 1 : Comparaison des effets de funneling pour une technologie bulk – CMOS (- a -) et Epi – CMOS (- b -).

La ZCE s'étend relativement moins dans le substrat dopé P+ situé sous la couche épitaxiée du fait de son fort dopage. Il en résulte une limitation de la collection de charge.
La technologie Epi – CMOS réduit la susceptibilité aux évènements singuliers mais ne la supprime pas.

CMOS – SOI / CMOS – SOS

Les technologies CMOS sur isolant (ou CMOS – SOI pour Silicon On Insulator) ou sur saphir (ou CMOS – SOS pour Silicon On Sapphire) permettent également de réduire la charge collectée. Nous laisserons de côté l'approche CMOS – SOS qui, bien que postérieure, est une sous famille de la technologie CMOS – SOI aujourd'hui majoritaire.
En CMOS – SOI il existe une isolation diélectrique totale entre les transistors individuels qui sont réalisés sur une couche isolante (SiO_2), avec pour effet de réduire le volume de silicium dans lequel l'irradiation peut générer des porteurs.
La figure 2 donne le schéma de principe de la réalisation des transistors pour les deux variantes à film épais (- a -) ou film mince (- b -). Dans la variante à film épais les transistors sont réalisés par gravure des diffusions dans des "îlots" de silicium dopés P ou N. Pour la variante à film mince, les diffusions des transistors touchent directement la couche isolante et encadrent la zone active de silicium.

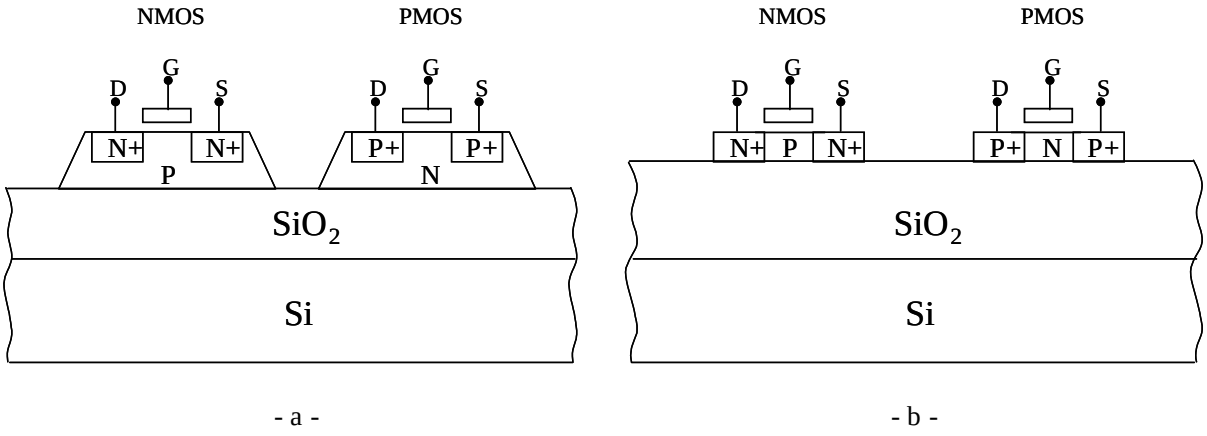


Figure 2 : Schéma de principe de réalisation des transistors dans les technologies à film épais (- a -) et à film mince (- b -).

La réduction du volume de génération des porteurs dus à l'irradiation et la disparition (quasi disparition pour les films épais) des composantes de funneling est manifeste sur ces schémas. La technologie CMOS – SOI permet de réduire la susceptibilité des circuits intégrés aux évènements singuliers de plusieurs ordres de grandeurs sans cependant les faire disparaître.

Le durcissement technologique des circuits intégrés apporte un gain significatif pour la réduction du nombre des évènements singuliers. La technologie Epi – CMOS est aujourd'hui la plus répandue pour les applications spatiales grâce à un coût moins élevé que l'approche CMOS – SOI .

A.1.2. Le durcissement par conception des circuits intégrés.

La deuxième grande famille de techniques de durcissement des circuits intégrés aux effets singuliers est le durcissement par conception. Elle regroupe des approches s'appliquant aussi bien à l'échelle des transistors qu'à l'échelle du système dans son ensemble. Une partie de ces techniques a pour but d'augmenter la charge critique ou charge tolérable par les points sensibles du circuit (jeu sur le dimensionnement, augmentation des capacités, redondance fonctionnelle). Elles sont nombreuses à concerner les éléments de mémorisation (découplage résistif, restructuration de leur architecture au niveau des transistors). Une autre partie s'applique à un niveau plus global, elles ont pour finalité de permettre au circuit de fonctionner correctement malgré l'apparition périodique d'erreurs d'origine radiative (redondance massive, utilisation de codes détecteurs et correcteurs d'erreurs).

A.1.2.1. Redimensionnement et augmentation des capacités.

Un surdimensionnement des transistors des portes logiques se traduit, pour une même charge collectée, par un transitoire moins abrupt sur les nœuds sensibles. Il permet également une évacuation plus rapide des charges générées via les transistors passants et donc un retour plus rapide au potentiel initial correct.

De façon similaire, une augmentation de la capacité des nœuds sensibles (grâce à un jeu sur le dessin des transistors par exemple) entraîne une augmentation de la charge critique, en effet, la collection de charge nécessaire pour atteindre le potentiel des seuils de transition des portes logiques augmente en proportion.

Mais, ces deux approches ont pour inconvénient une dégradation de la vitesse de fonctionnement du circuit. Elles sont antagonistes avec l'évolution technologique qui va vers une intégration sans cesse plus poussée des circuits et une augmentation de leur fréquence de fonctionnement. En conséquence, elles sont rarement acceptables.

A.1.2.2. Durcissement par restructuration des éléments de mémorisation.

L'aléa logique a été un des premiers effets radiatifs observés dans les satellites [BIN75]. D'où un intérêt et un effort constant de nombreuses équipes de recherche depuis une vingtaine d'années pour le durcissement des éléments de mémorisation qu'il affecte. Ainsi, plusieurs évolutions de l'architecture des bascules et des points mémoires ont été proposées. Une des plus anciennes est le découplage résistif des boucles de rétroaction des SRAM. Depuis, un grand nombre d'architectures alternatives basées sur l'introduction de redondances dans les bascules ont également été présentées. Nous nous proposons maintenant de décrire quelques exemples significatifs de durcissement par restructuration des éléments de mémorisation.

Le durcissement par découplage résistif.

Une des premières approches de durcissement par restructuration a été l'introduction d'un découplage résistif dans les boucles de rétroaction des SRAM [AND82]. L'insertion de résistances en poly silicium (notées R sur le schéma de principe de la figure 3) entre les nœuds Q et Qb sensibles aux aléas et les nœuds de commande G et Gb des inverseurs permet un filtrage RC des effets transitoires.

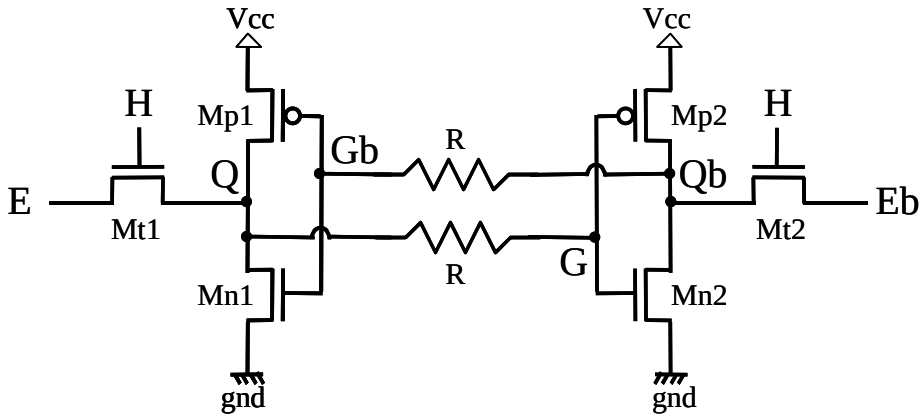


Figure 3 : Durcissement par découplage résistif d'une SRAM.

Pour illustrer ce mécanisme de durcissement, prenons le cas où la SRAM est en mode statique (H=0) et où elle mémorise l'état logique 1 sur les nœuds Q et G et l'état logique 0 sur les nœuds Qb et Gb. Au niveau du premier inverseur, c'est le drain du transistor NMOS bloqué Mn1 qui est sensible. Dans l'hypothèse où cette zone sensible est traversée par une particule ionisante il en résulte un passage du potentiel de Q vers le niveau bas. Le filtrage passe bas RC introduit par R retarde alors la propagation de l'erreur vers G. Si la perturbation du potentiel de Q cesse avant que le potentiel de G n'ait atteint le seuil de basculement du deuxième inverseur, alors, la SRAM conserve son information, l'aléa n'a pas de conséquence sur l'état logique " permanent " du point mémoire. Dans le cas contraire la cellule bascule malgré le durcissement. Un même raisonnement peut être développé pour l'ensemble des zones sensibles et des niveaux logiques possibles.

Le durcissement est efficace tant que la durée de la perturbation d'origine radiative est inférieure au " temps de réponse " du point mémoire.

Cette méthode est fréquemment employée pour durcir les SRAM, elle est encore d'actualité [ROC00]. Cependant elle ne garantit qu'une immunité à seuil pour les aléas logiques et elle possède deux principaux inconvénients :

- une dégradation des temps d'écriture d'autant plus importante que l'on cherche à filtrer les aléas les plus longs,
- et une variation de la qualité du durcissement avec la température due au coefficient de température négatif du poly silicium.

Le besoin d'autres approches de durcissement s'est fait sentir.

Depuis la fin des années quatre-vingt de nombreuses propositions de durcissement des bascules par restructuration de leur architecture ont été proposées [ROC88], [CAN91], [WHI91], [LIU92], [BES93], [VEL94], [SAL99] avec chacune leurs qualités et leurs inconvénients. Elles ont en commun un accroissement de leur complexité comparée aux architectures standards et l'introduction de redondances dans le stockage de l'information. Ce dernier point limite l'efficacité de leur durcissement au cadre des évènements radiatifs uniques (cf. partie II.B.3.3). Nous avons fait le choix de détailler deux architectures récentes pour illustrer ce principe, celle de la cellule DICE proposée par [CAL96] et celle de la S/DH Latch proposée par [MON99].

La cellule mémoire DICE (Dual Interlocked Storage Cell).

Le durcissement aux aléas logiques de la cellule DICE est basé sur deux principes :

- le stockage de l'information sur quatre nœuds (contre deux pour une bascule D classique),
- et la correction du niveau logique du nœud perturbé par un passage de particule ionisante grâce au niveau correct conservé par les autres noeuds.

Son architecture est donnée figure 4.

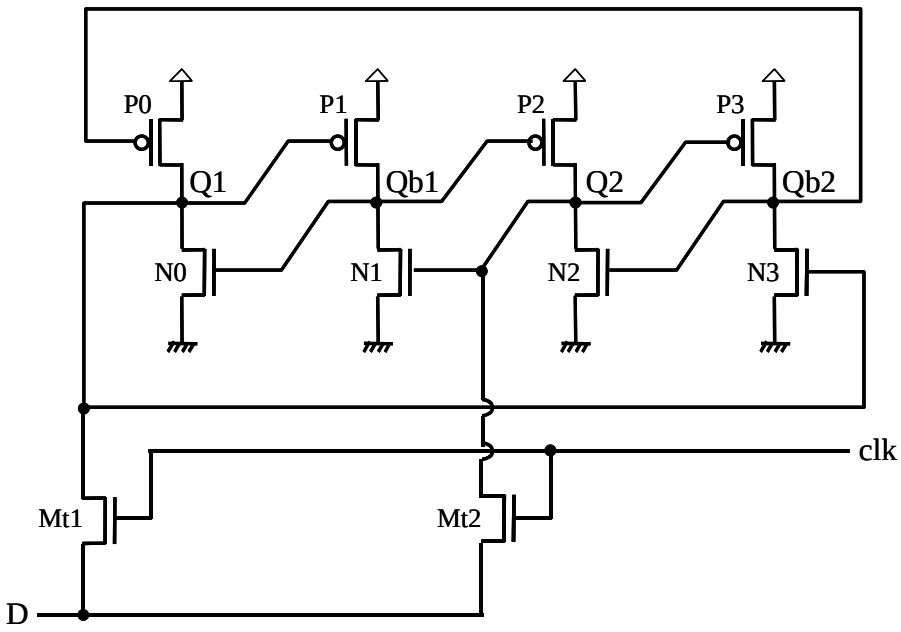


Figure 4 : Architecture de la cellule mémoire DICE.

Lorsque la cellule DICE est en mode mémoire les nœuds Q1 et Q2 ont le même niveau logique, complémentaire de celui des nœuds Qb1 et Qb2. L'écriture est assurée en parallèle

via les transistors d'accès Mt1 et Mt2 commandés par le signal d'horloge (clk). Il semble que l'on retrouve dans cette cellule quatre inverseurs, mais le câblage des signaux de commande des transistors est modifié. La grille des PMOS est connectée sur le nœud de stockage situé à leur gauche et à leur droite pour les NMOS assurant ainsi le durcissement.
Dans un pire cas, la corruption de l'état d'un de ces nœuds entraîne au maximum une erreur sur un nœud de stockage complémentaire. Les deux autres nœuds de stockage passent en mode haute impédance et conservent leurs niveaux logiques inchangés. Dès la fin de l'aléa, ils commandent le retour des nœuds corrompus au niveau logique initial correct.
Les simulations électriques réalisées [CAL96] confirment la validité du durcissement mais nous n'avons pas connaissance de la publication de résultats expérimentaux.
Cette cellule souffre d'un défaut commun à la majorité des approches par restructuration, à savoir un surcoût en surface, il est ici de l'ordre de 90% par rapport à la cellule standard.
Mais, son principal défaut provient de la limitation de son durcissement aux seuls aléas logiques statiques. La cellule DICE n'est pas durcie aux aléas logiques dynamiques mis en évidence au II.C.2.3 contrairement à la S/DH Latch présentée ci-après.

La S/DH Latch.

La S/DH Latch [MON98] remplit les fonctions d'une bascule D, et elle est durcie aux aléas logiques statiques et dynamiques. Son architecture au niveau transistors est donnée figure 5.

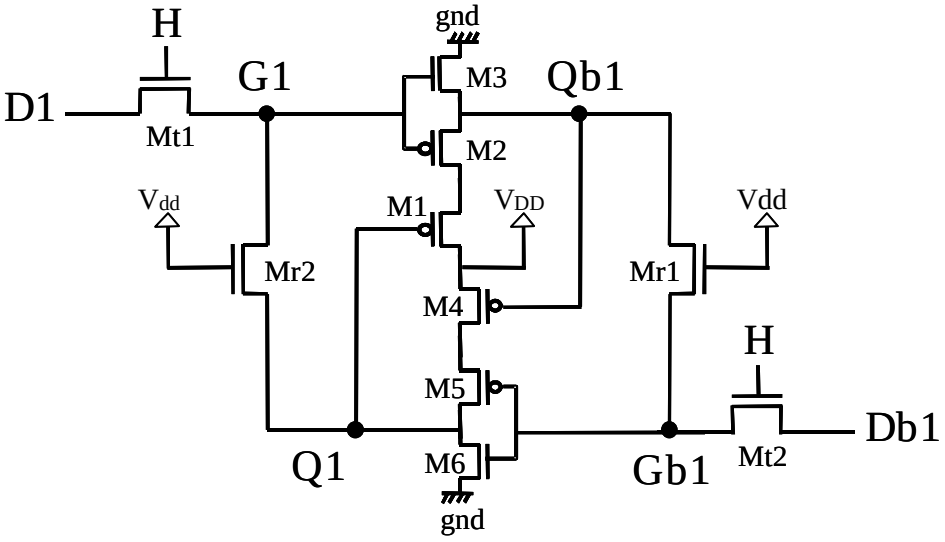


Figure 5 : Architecture de la S/DH Latch.

Elle possède deux transistors d'accès Mt1 et Mt2 commandés par l'horloge H et reliés à deux entrées complémentaires D1 et Db1. On y retrouve également deux inverseurs constitués pour le premier par les transistors M2 et M3, et pour le second par les transistors M5 et M6. L'originalité de cette architecture provient :

- des transistors M1 et M4 qui permettent l'alimentation des deux inverseurs par la tension haute Vdd,
- des transistors NMOS Mr1 et Mr2, toujours passants, qui permettent un découplage résistif entre les nœuds Q1 et G1 et entre les nœuds Qb1 et Gb1.

Il en résulte à la fois une redondance de l'information (stockée sur les couples G1 - Q1 et Gb1 - Qb1) et un mécanisme de correction des aléas, tous deux basés sur les principes de durcissement suivants :

- le découplage résistif permis par Mr1 et Mr2,
- la mise en haute impédance de la sortie des inverseurs lorsque leur entrée subit un aléa.

Considérons dans un premier temps le cas du durcissement statique (H=0) et faisons l'hypothèse que la cellule est dans un état tel que Q1=G1=1 avec Qb1=Gb1=0. En appliquant les règles du II.B.3.2, on localise les zones sensibles relatives aux nœuds Q1, G1 et Qb1. Etudions maintenant chacune de ces possibilités en détail lorsque le passage d'une particule radiative ionisante induit une inversion de niveau logique.

Dans le cas où G1 passe au niveau logique bas, M3 se bloque et M2 devient passant, tandis que M1 demeure bloqué. Le nœud Qb1 passe en haute impédance, il conserve (ainsi que Gb1) un niveau logique bas. Ces deux nœuds assurent alors (via leur commande sur M4, M5 et M6) le maintien de Q1 au niveau haut. Dès la fin de l'aléa, Q1 réimpose un niveau haut sur G1.

Le deuxième cas est celui, où Q1 passe au niveau logique bas ; M1 devient alors passant. Mais M2 étant bloqué (G1 est au niveau haut), le niveau de Qb1 n'est pas perturbé. La propagation de l'erreur du nœud Q1 vers le nœud G1 est bloquée par le filtrage RC introduit par le transistor résistif Mr2. On retrouve le mécanisme décrit pour la SRAM durcie par découplage résistif. Le durcissement est efficace si la durée de la perturbation est inférieure au temps de réaction de la cellule.

Le troisième cas se présente lorsque Qb1 passe au niveau logique haut. On retrouve alors le principe d'un filtrage résistif de l'aléa entre les nœuds Qb1 et Gb1 exposé dans le cas précédent.

De façon similaire, si la cellule mémorise des états logiques opposés (Q1=G1=0 avec Qb1=Gb1=1) les mêmes mécanismes sont à l'œuvre pour garantir son durcissement.
Le durcissement dynamique de la cellule est assuré lors de l'écriture (H=1) par l'utilisation des deux entrées de données complémentaires D1 et Db1.

Considérons l'exemple où la cellule est en passe de mémoriser (par un retour de H à 0) des niveaux logiques tels que Q1=G1=1 avec Qb1=Gb1=0, imposés par D1=1 et Db1=0. Deux possibilités d'aléas logiques dynamiques se présentent alors : celle de la propagation d'un transitoire de tension vers le niveau bas jusqu'au nœud G1 via D1 et Mt1 et celle de la propagation d'un transitoire de tension vers le niveau haut jusqu'au nœud Gb1 via Db1 et Mt2.

Le premier cas est similaire à celui d'un aléa vers le niveau bas induit directement sur G1 en statique et décrit précédemment.

Le deuxième cas ne peut pas se ramener à l'étude effectuée en statique car il n'existait pas de possibilité d'aléa vers le niveau haut en Gb1. Dans ce cas, c'est le NMOS Mt2 qui permet un filtrage RC efficace du transitoire de tension.
La description des cas symétriques est identique.
Ainsi, la S/DH Latch assure une réelle amélioration du durcissement aux aléas logiques statiques et dynamiques comme l'ont montré les simulations électriques et les tests expérimentaux sous irradiation d'ions lourds réalisés [MON99b].
D'après [MON99], le surcoût en surface de la S/DH Latch par rapport à une bascule D classique est de l'ordre de 30% ce qui est un très bon résultat dans le cadre du durcissement par restructuration.

Cependant, il nous est apparu que l'utilisation de transistors résistifs de type NMOS entraînait une dégradation des niveaux logiques hauts des nœuds G1 et Gb1 du fait d'une chute de tension à leurs bornes. La S/DH Latch ayant été développée dans une technologie dont la tension d'alimentation est de cinq volts, ce problème ne s'est pas révélé critique. Mais le passage à des technologies dont les tensions d'alimentation sont divisées par deux, voire trois, réduit les marges de sécurité et l'efficacité du durcissement résistif. Ainsi, dans le cas d'un aléa du niveau haut vers le niveau bas en Q1, le potentiel de G1 étant déjà proche du seuil de basculement des transistors M2 et M3, la qualité du durcissement résistif peut être mise en défaut pour les aléas les plus longs [DUT01].

A.1.2.3. Techniques de durcissement par redondance.

Les techniques de durcissement par redondance sont bien adaptées au durcissement vis à vis des aléas logiques. On distingue deux familles : la redondance spatiale et la redondance temporelle.

Redondance spatiale.

La technique de redondance spatiale la plus complète est celle de la redondance triple avec vote majoritaire. Elle consiste à implanter trois fois la même fonction logique et à comparer leurs sorties. Dans le cadre de l'hypothèse de l'événement unique, une seule de ces trois fonctions est susceptible de délivrer une sortie erronée à la fois. Un module de vote majoritaire est ajouté à la suite de leurs sorties, son rôle est de sélectionner sur sa sortie la réponse majoritaire, c'est à dire de filtrer la sortie invalide. La figure 6 donne le schéma de principe de cette architecture.

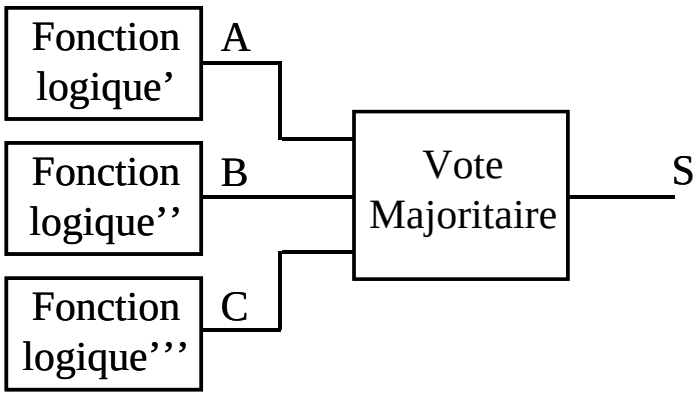
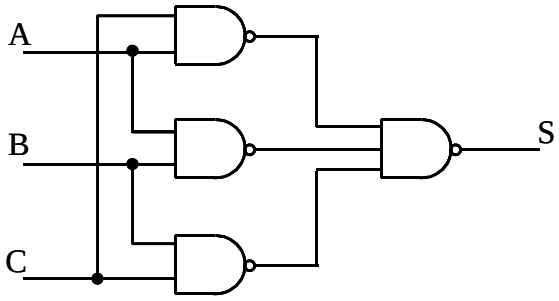


Figure 6 : Principe de l'architecture de redondance triple avec vote majoritaire.

Tant que le niveau logique d'un seul des nœuds A, B ou C est altéré, on retrouve sur la sortie S du module de vote un niveau logique correspondant au fonctionnement valide du circuit. La figure 7 donne une implantation possible du module de vote majoritaire sous forme de portes non-ET.

Il est à noter que la sortie du module de vote est elle même sensible aux effets transitoires. Pour en limiter les conséquences, il est possible d'avoir recours à plusieurs modules de votes en parallèle. Il est également nécessaire de s'assurer, pour les fonctions séquentielles, que les erreurs induites dans les parties redondantes sont corrigées. Sous peine de rendre inefficace le vote majoritaire si une nouvelle erreur se produit dans celles-ci. Une

solution consiste, par exemple , à introduire des modules de vote dans les boucles de rétroaction.



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

Figure 7 :Architecture et table de vérité d'un module de vote majoritaire.

Le principal inconvénient de cette technique de durcissement est lié à un surcoût en surface très important, dû au triplement de la logique et à l'ajout des modules de vote.
On peut envisager de réduire en partie ce surcoût en utilisant une redondance simple (la logique est seulement dupliquée) et une porte OU exclusif selon le principe de la figure 8.

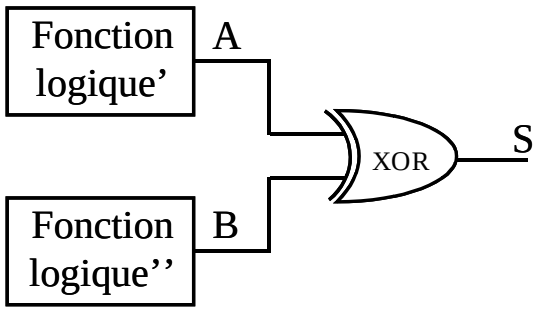


Figure 8 : Redondance simple et détection d'erreur.

La porte OU-exclusif (XOR) est utilisée en comparateur, elle permet de détecter l'occurrence d'une erreur dans une des fonctions logiques dupliquées. Cette approche présente néanmoins une limitation car elle ne permet pas de corriger les erreurs, mais seulement de les détecter. Il appartient au concepteur du circuit d'avoir prévu la procédure d'évitement avec interruption du fonctionnement du circuit et sa reprise à un stade antérieur non fautif après la disparition de l'erreur.

Le surcoût en surface reste important, bien plus de 100% en comptant la logique ajoutée dédiée à la gestion des interruptions consécutives à la détection d'erreurs. Aussi, il a été récemment proposé de substituer à la redondance spatiale une redondance temporelle beaucoup moins gourmande en surface.

La redondance temporelle.

Le principe de la redondance temporelle consiste à transposer dans la domaine temporel la contrainte spatiale imposée par la redondance triple. Cette approche fait l'objet d'un brevet d'invention [NIC99].

Au lieu de comparer les sorties de trois modules fonctionnels identiques pour éliminer les erreurs radiatives, on peut envisager d'échantillonner la sortie d'un unique bloc fonctionnel à trois instants différents. La comparaison des trois échantillons via un module de vote majoritaire permet de filtrer les erreurs, pourvu que la durée qui sépare leurs instants de capture respectifs soit supérieure à la durée d'un éventuel transitoire de tension. C'est sur ce principe, illustré figure 9, qu'est basé le principe de la redondance temporelle présenté par [MAV98] et [MAV00].

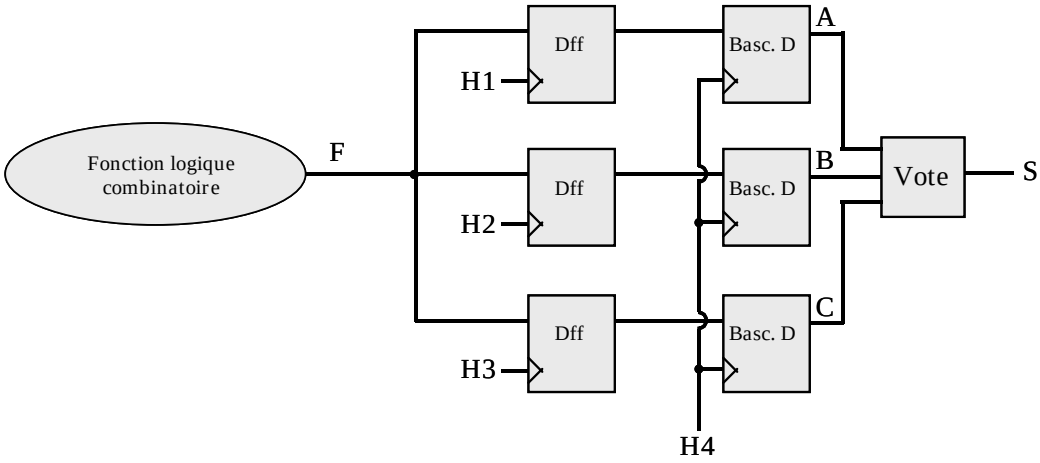


Figure 9 : Principe de la redondance temporelle.

Dans l'exemple simplifié de la figure 9, la sortie d'une fonction logique combinatoire est reliée aux entrées de trois D flip-flops. Les sorties des trois flip-flops sont connectées aux entrées de trois bascules D qui commandent, via leurs sorties, un module de vote majoritaire. Le chronogramme des signaux d'horloge de période T, dont les fronts sont successivement décalés d'une durée δ (tel que $\delta=T/4$) est donné figure 10.

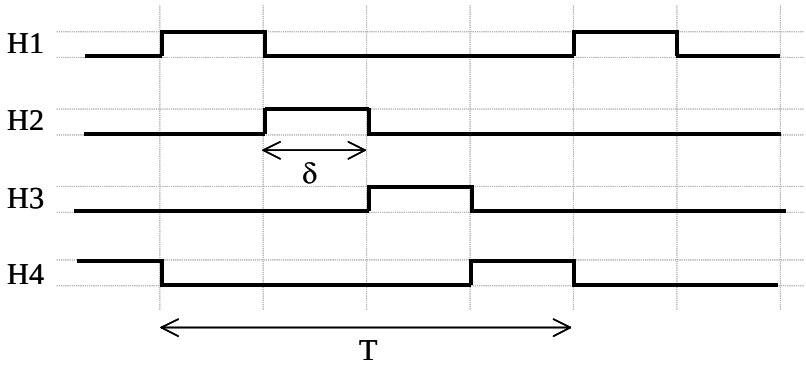


Figure 10 : Chronogramme des signaux d'horloge.

H1, H2 et H3 commandent la capture de la sortie F de la fonction combinatoire à trois instants décalés d'un intervalle de temps δ. Puis, H4 commande la transmission simultanée de ces trois échantillons au module de vote via les bascules D. Ainsi, tout transitoire de tension d'une durée inférieure à δ affectant F ne peut pas être mémorisé par plus d'une D flip-flop et est filtré par le vote majoritaire. Cette technique ne nécessite pas l'utilisation d'éléments durcis. En effet, tout aléa logique statique qui changerait l'état d'une des flip-flops ou bascules D serait éliminé par le voteur, et tout évènement transitoire qui se produit au niveau du voteur serait à son tour filtré en aval par redondance temporelle.

Cette technique est très séduisante, elle permet un gain en surface important par rapport à la redondance spatiale et elle assure un durcissement efficace aux aléas logiques et aux transitoires. Cependant, elle impose l'utilisation d'une horloge dont la période doit être quatre fois plus grande que la durée maximale des transitoires auxquels elle est durcie. Si on considère une durée d'aléa δ=500 ps, on obtient une fréquence maximale d'horloge limitée à 500 MHz. La contrainte spatiale a été convertie en contrainte temporelle.

[ANG00] présente une approche basée sur une redondance temporelle simple (deux échantillons seulement sont capturés) plus économe en surface et qui permet à la période d'horloge d'approcher δ, la durée d'un éventuel transitoire. Mais une faute éventuelle est seulement détectée, elle n'est pas corrigée.

A.1.2.4. Codes détecteurs et correcteurs d'erreurs.

Les codes détecteurs et correcteurs d'erreurs sont fréquemment utilisés dans les mémoires et lors des échanges de données entre les CI. On peut citer entre autre : le contrôle de la parité, le code Berger, le code de Hamming, le code Reed-Solomon, etc. Ils permettent

de détecter et de corriger une ou plusieurs erreurs dans des groupes de bits en fonction de leur complexité. Cependant, nous n'avons pas connaissance de leur utilisation et de leur efficacité vis à vis des phénomènes radiatifs.

A.1.3. Conclusion sur les approches générales.

Les techniques de durcissement que nous venons d'évoquer ont un coût financier :

- les process Epi - CMOS et CMOS SOI ne bénéficient pas des effets d'échelle du process bulk CMOS standard et ont un coût plus élevé,
- les techniques de durcissement par redondance fonctionnelle imposent des surfaces additionnelles qui peuvent être importantes (particulièrement pour la redondance triple),
- les temps de développement sont allongés par la difficulté de l'utilisation des bibliothèques de cellules standards, etc.

Elles entraînent également souvent une dégradation des performances temporelles et une augmentation de la puissance consommée.

Et, malgré toutes ces contraintes, elles ne garantissent pas une immunité totale aux effets singuliers. Il n'existe pas aujourd'hui de solution optimale à ces problèmes.

Les constructeurs de FPGA qui s'intéressent aux marchés spatiaux et aéronautiques, se sont inspirés de ces approches pour améliorer la fiabilité de leurs circuits comme nous allons le voir dans la partie suivante.

A.2 – Etat de l'art des techniques de durcissement proposées par les fabricants de FPGA.

Cette partie est consacrée à la revue des techniques de durcissement proposées par les fabricants de FPGA. Son objectif est de les évaluer afin de guider notre réflexion dans la mise au point de nos propres solutions. Deux sociétés, Xilinx et Actel, sont particulièrement actives sur le marché des applications spatiales des FPGA, nous nous limiterons aux approches qu'elles proposent. La société Xilinx commercialise des FPGA programmables à base de SRAM, elle est donc confrontée au double problème du durcissement des couches opérative et de configuration. La société Actel propose des circuits programmables par antifusibles, elle ne s'est donc intéressée qu'au durcissement de la couche opérative. Ces sociétés utilisent toutes deux une technologie Epi – CMOS qui permet de limiter les effets de dose et de réduire l'occurrence d'évènements singuliers. Ces derniers ne disparaissent cependant pas complètement. Aussi, ces sociétés ont-elles également recours à des techniques de

durcissement par conception. Dans un premier temps, nous présentons les solutions proposées pour le durcissement de la couche opérative, puis ensuite, les approches de Xilinx pour le durcissement de la couche de configuration. Et enfin, nous tirerons des conclusions sur ces approches.

A.2.1. Durcissement de la couche opérative.

Le durcissement de la seule couche opérative n'est pas suffisant. Nous avons vu dans la partie II.C.1 que toute modification de la couche de configuration d'un FPGA entraîne une modification de sa fonctionnalité et par conséquent la nécessité d'une reconfiguration (les FPGA à anti- fusibles ne sont pas concernés par ce problème). Cependant, les approches proposées par les fabricants reposent principalement sur l'utilisation de la redondance triple avec vote majoritaire, ce qui a pour conséquence de permettre au circuit de continuer à fonctionner correctement en attendant la correction de la configuration comme nous allons le voir après avoir présenté ces approches.

En règle générale, les FPGA durcis pour les applications spatiales reprennent l'architecture des circuits commerciaux classiques sans évolution notable. Seul Actel propose une approche de durcissement par restructuration des D flip-flops de sa famille RT54SX-X de FPGA [ACT01]. La restructuration utilisée est basée sur une redondance triple avec vote majoritaire, l'architecture de ces D flip-flops durcies est donnée figure 11.

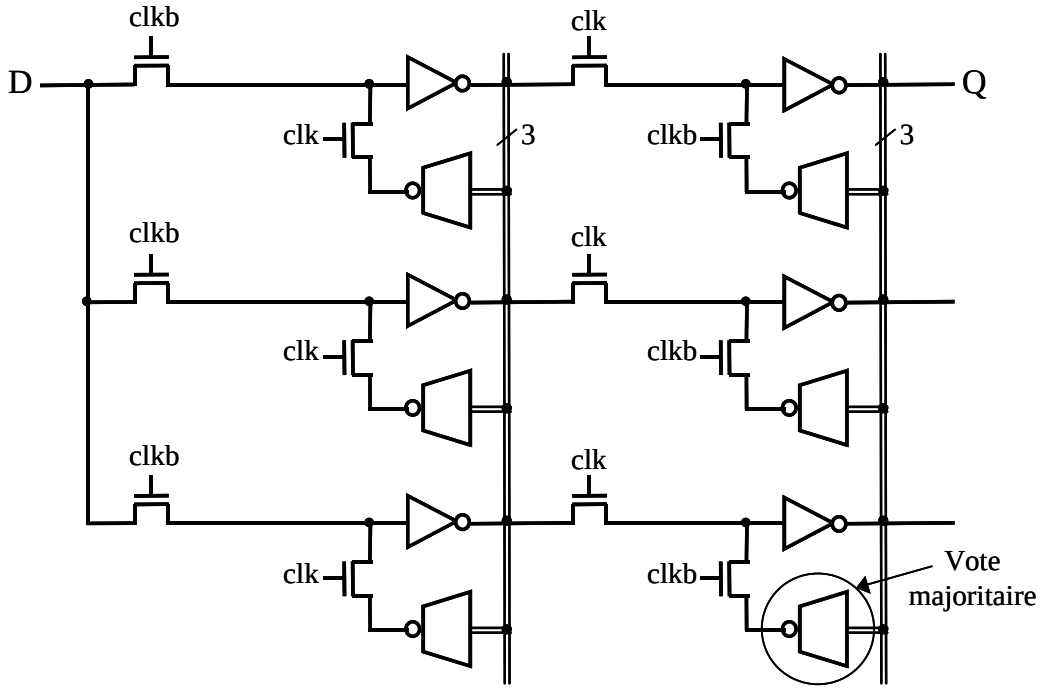


Figure 11 : D flip-flop durcie par restructuration (Actel RT54SX-S).

Elles sont constituées de trois D flip-flops connectées en parallèle à une même entrée D, à un même signal d'horloge clk et à son complémentaire clkb. La sortie Q est prise sur une des bascules D esclave. Le vote est intégré dans chacune des bascules grâce à une porte de vote majoritaire inverseuse. Cette architecture permet d'éviter l'accumulation d'erreurs dans les bascules car le vote intervient dans toutes les boucles de rétroaction. Cette D flip-flop est durcie contre tout aléa logique statique survenant dans une de ses bascules D. Mais, la technique de redondance utilisée n'est pas complète, elle n'assure pas le durcissement vis à vis des aléas logiques dynamiques tels que nous les avons décrits au II.C.2.3. En effet, si un transitoire de tension entraîne la propagation d'une erreur sur D simultanément au front descendant de clkb, celle-ci sera capturée par les trois bascules maître, mettant ainsi en défaut l'efficacité des modules de vote majoritaire. De façon similaire, un transitoire de tension sur clk ou clkb peut entraîner la mémorisation d'une valeur erronée par les trois bascules maître ou esclave. Cette approche permet de réduire le nombre d'erreurs d'origine radiative mais n'est pas suffisante pour garantir un durcissement optimal.

Les sociétés Actel et Xilinx utilisent en général la redondance triple avec vote majoritaire pour durcir les FPGA. L'architecture redondante proposée par Actel pour la couche opérative [ACT97] présente les mêmes faiblesses vis à vis des aléas logiques dynamiques et des transitoires de tension que la D flip-flop durcie que nous venons de considérer. Nous verrons dans la partie III.B.1.4 des simulations électriques les mettant en évidence. En revanche, les techniques d'implantation de la redondance triple avec vote proposées par Xilinx [XIL01] sont plus complètes et prennent en compte la possibilité d'aléas dynamiques. Elles préconisent la triplication de tous les chemins de données, des entrées du circuit jusqu'à ses sorties et l'utilisation de trois signaux d'horloge indépendants. Ainsi, toutes les possibilités de fonctionnement erroné induites en un point unique du circuit sont éliminées. Elles permettent un durcissement efficace de la couche opérative du circuit aux aléas logiques statiques et dynamiques. Les techniques de redondance triple avec vote majoritaire n'assurent un durcissement efficace que tant que la configuration du FPGA reste valide. Mais elles introduisent un temps de latence pendant lequel le circuit continue a fonctionner correctement.

Considérons le schéma de principe de la figure 12 qui montre l'implantation de la redondance triple pour une fonction logique donnée.

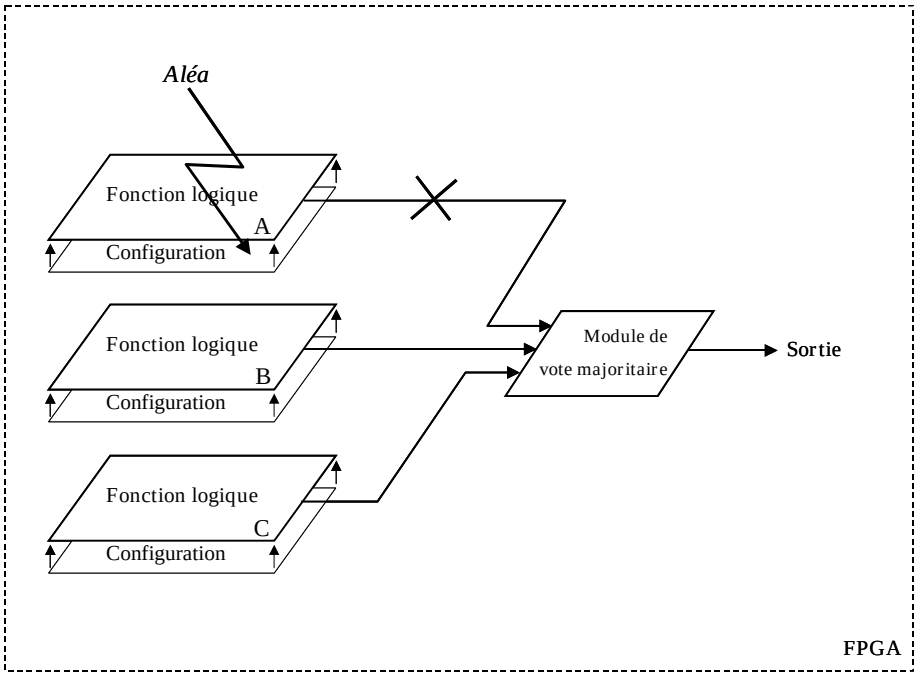


Figure 12 : Application de la redondance triple avec vote dans un FPGA.

Cette même fonction logique est réalisée par trois modules différents A, B et C, disposant chacun de leurs propres CSRAM, et leurs trois sorties correspondantes sont connectées à un module de vote majoritaire. Dans l'hypothèse où un aléa logique modifie un bit de configuration du module A (flèche brisée sur le schéma) et que celui-ci perd sa fonctionnalité, il commence à délivrer des données erronées. Elles sont cependant filtrées par le vote majoritaire, les modules B et C continuant à fonctionner correctement. Ainsi, on retrouve toujours en sortie du module de vote des données correspondant à un fonctionnement correct. Mais désormais, le circuit est vulnérable à tout nouvel évènement singulier frappant les modules B et C et leurs bits de configuration. Si par exemple, un nouvel aléa logique inverse un bit de configuration du bloc B et lui fait perdre sa fonctionnalité, le module de vote sélectionnera la valeur invalide des sorties de A et de B aux dépends de la sortie de C. Le circuit cesse alors de fonctionner correctement. Il existe donc un temps entre l'occurrence des deux aléas dit temps de latence pendant lequel le circuit continue à fonctionner correctement. Si l'erreur dans la configuration de A est corrigée *avant* l'apparition du deuxième aléa, alors, aucune erreur n'apparaît en sortie du circuit. C'est l'existence de ce temps de latence qui est mise à profit par Xilinx pour assurer la correction de la couche de configuration comme nous le verrons dans la partie suivante.

A.2.2. Durcissement de la couche de configuration.

La couche de configuration d'un FPGA peut compter jusqu'à quelques millions de CSRAM. Ces points mémoire de configuration sont très sensibles aux aléas logiques [WAN91], et leur modification entraîne une perte de la fonctionnalité du circuit. Il est nécessaire de reprogrammer le FPGA chaque fois que sa configuration est altérée. Pour éviter la répétition des opérations de reconfiguration dans les milieux les plus radiatifs (lors d'une éruption solaire par exemple), le durcissement de la couche de reconfiguration s'impose.
A notre connaissance, les fabricants de FPGA ne proposent pas de solutions de durcissement par restructuration des CSRAM, probablement à cause du coût en surface qui en résulterait (seul [ROC00] a présenté des CSRAM durcies par découplage résistif sans surface additionnelle).

La solution de durcissement de la couche opérative la plus aboutie est celle développée par Xilinx pour ses FPGA de la famille des Virtex [XIL00], [XIL01]. Ces circuits possèdent deux capacités qui sont exploitées pour détecter et corriger les erreurs, la possibilité de relecture (readback) de l'ensemble des bits de configuration et la possibilité de reconfiguration partielle (partial reconfiguration) d'une partie des CSRAM sans avoir à interrompre le fonctionnement du FPGA. La détection des erreurs peut être effectuée par une relecture de l'ensemble des bits de configuration et une comparaison à ceux contenus dans une mémoire de référence, ou encore, en calculant les signatures d'ensembles de bits groupés en motifs (d'un millier de bits chacun) et en les comparant à celles attendues d'une configuration valide. Une fois l'erreur détectée, le motif est localisé et réécrit grâce à l'aptitude du circuit à une reconfiguration partielle. Xilinx propose également une autre approche appelée " scrubbing ", qui fait l'économie de l'étape de détection des erreurs et consiste à réécrire périodiquement l'ensemble de la configuration du FPGA.
Ces deux approches supposent l'utilisation d'une mémoire de référence externe au FPGA pour stocker la configuration. Mais il faut qu'elle soit elle-même durcie aux aléas logiques, sinon le problème ne serait que déplacé, ce qui exclut l'utilisation d'une mémoire à base de SRAM. Il est possible d'utiliser une PROM (Programmable Read Only Memory) mais le caractère évolutif lié aux FPGA reprogrammables à base de SRAM est alors perdu. Il serait plus efficace d'avoir recours à un FPGA à anti-fusibles. Xilinx propose également des PROM reprogrammables à base de mémoires flash [XIL01b], mais leur capacité de durcissement vis à vis des aléas logiques n'est pas détaillée et leur durcissement aux effets de dose est limité à 40 krad contre 100 krad pour les FPGA de la famille Virtex. On peut, dès lors, s'interroger sur la viabilité à moyen et long terme d'une telle association.

A.2.3. Conclusion sur les approches des fabricants de FPGA.

Les approches proposées par les fabricants de FPGA pour le durcissement de la couche opérative sont toutes basées sur l'utilisation de la redondance triple avec vote majoritaire. Cette solution utilisée conjointement avec des techniques de correction des erreurs de la couche de configuration (reconfiguration partielle ou bien " scrubbing ") assurent un durcissement de bonne qualité vis à vis des aléas singuliers, le circuit continue à fonctionner correctement jusqu'à leur correction.

Cependant, le recours à la redondance triple est coûteux en terme de ressources, il requiert l'utilisation de FPGA plus complexes disposant d'une plus grande capacité d'implantation et entraîne une augmentation non négligeable de la consommation. De plus, le problème de l'utilisation d'une mémoire de référence durcie pour stocker, tout en conservant une possibilité d'évolution, la configuration du FPGA n'est pas résolu.

A.3 – Conclusion.

Nous venons de passer en revue les différentes techniques de durcissement physique et par conception des CI et des FPGA les plus répandus et d'en montrer les limites.

Les techniques de durcissement physique ne faisaient pas partie de notre domaine de travail centré sur l'utilisation de technologies standards. Néanmoins, leur utilisation conjointe avec les approches de durcissement que nous proposons dans la partie suivante ne sont pas incompatibles et permettrait très certainement d'obtenir des résultats encore améliorés.

L'étude des approches de durcissement par conception nous a permis de mettre en évidence l'intérêt et les limites des techniques basées sur la restructuration des éléments de mémorisation. Une grande partie de notre travail s'est donc orienté vers la mise au point d'une bascule D durcie aux aléas logiques statiques et dynamiques et aux transitoires de tension présentant une amélioration de la qualité du durcissement par rapport aux architectures préexistantes.

III.B. Nouvelles approches du durcissement des FPGA.

Cette partie présente les deux pistes suivies lors de cette étude afin d'améliorer la fiabilité des circuits reconfigurables en présence d'évènements singuliers.

Une première approche est basée sur un durcissement par restructuration. Elle propose une nouvelle architecture d'inverseur durci par une restructuration au niveau transistor. Cet inverseur, l'inverseur HZ, trouvera son utilisation dans la réalisation de différents circuits (arbres d'horloge, points mémoires et bascules durcies). Les points mémoire durcis permettront de résoudre les problèmes de vulnérabilité de la couche de configuration détaillés dans le paragraphe précédent. Et l'utilisation conjointe d'arbres d'horloge et de bascules durcis sera appliquée à la mise au point d'une nouvelle architecture alternative à la redondance triple où TMR (Triple Modular Redundancy).

Une deuxième approche s'attaquera à la sauvegarde de l'intégrité de la couche de configuration grâce à l'utilisation d'un algorithme de détection et de correction des erreurs.

B.1 – Durcissement par restructuration.

B.1.1. L'inverseur HZ.

Plusieurs architectures d'inverseurs durcis aux événements singuliers ont été proposées par le passé [CAN91][CAL96][SAL99], mais leur efficacité n'est pas absolue. Nous proposons ici une autre approche [DUT01b], elle est basée sur une restructuration de l'inverseur au niveau des transistors. Dans la suite, le principe de fonctionnement de l'inverseur HZ est tout d'abord détaillé, puis, son durcissement aux transitoires de tension mis en évidence dans le cas d'un arbre d'horloge, et enfin, une comparaison est effectuée avec l'inverseur classique.

Principe

En logique CMOS standard éviter l'émergence de transitoires de tension au niveau des zones sensibles d'un circuit en milieu radiatif apparaît impossible. Ce constat nous a conduit à proposer une structure d'inverseur qui s'accommode de ces transitoires sans générer d'erreurs fonctionnelles. Le principe utilisé repose sur l'utilisation de deux techniques de durcissement : une redondance simple des entrées et des sorties et une mise en haute impédance des deux sorties trois états lors des transitoires. Ces techniques permettent de protéger l'information et de stopper la propagation d'éventuelles erreurs. L'état haute impédance (HZ en abrégé) donne son nom à cet inverseur : l'inverseur HZ. La figure 1 donne le détail de son architecture.

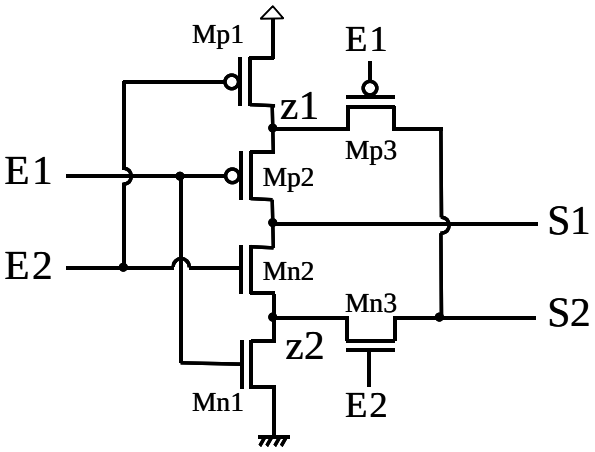


Figure 1 : Schéma de l'inverseur HZ

Constitué de six transistors : 3 PMOS (notés Mp1, Mp2 et Mp3) et 3 NMOS (notés Mn1, Mn2 et Mn3), cet inverseur possède deux entrées (E1 et E2) et deux sorties (S1 et S2). Tant que les deux entrées sont dans le même état logique, l'inverseur HZ fonctionne comme un inverseur classique : ses deux sorties sont dans un même état logique, complémentaire de celui des entrées. Ainsi, lorsque E1=E2=1, les trois PMOS sont bloqués et les trois NMOS passants, ce qui impose S1=S2=0, et respectivement pour E1=E2=0, les trois PMOS sont passants et les trois NMOS bloqués imposant S1=S2=1. Si une erreur atteint une des entrées de l'inverseur HZ, ses sorties passent en état haute impédance. Elles conservent ainsi leur niveau logique, et la propagation de l'erreur est stoppée. Par exemple, si on se place dans le cas où E1=E2=1 impose S1=S2=0 et qu'un transitoire de tension fait passer E1 à l'état bas. Alors, Mp2 et Mp3 deviennent passants et Mn1 se bloque. S1 et S2 ne sont plus reliées ni à vdd ni à la masse, elles sont en état haute impédance. Elles conservent donc un niveau logique bas. L'erreur ne se propage pas au delà de l'inverseur HZ. Lorsque le transitoire est passé et que E1 revient au niveau haut, le circuit retrouve un fonctionnement normal. Par suite, les 3 NMOS passants et les 3 PMOS bloqués imposent S1=S2=0. Tous les transitoires sur les entrées aboutissent à un fonctionnement similaire. La table de vérité de l'inverseur HZ est donnée table 1.

E1	E2	S1	S2
0	0	1	1
0	1	HZ	HZ
1	0	HZ	HZ
1	1	0	0

Table 1 : Table de vérité de l'inverseur HZ

Ainsi, considéré comme élément de transition, la structure de l'inverseur HZ permet de stopper la propagation des erreurs tout en maintenant un niveau logique correct sur ses sorties grâce au passage en haute impédance.
Cependant, l'inverseur HZ possède en propre des zones sensibles aux événements singuliers. Elles sont données figure 2 en fonction du potentiel des entrées-sorties.

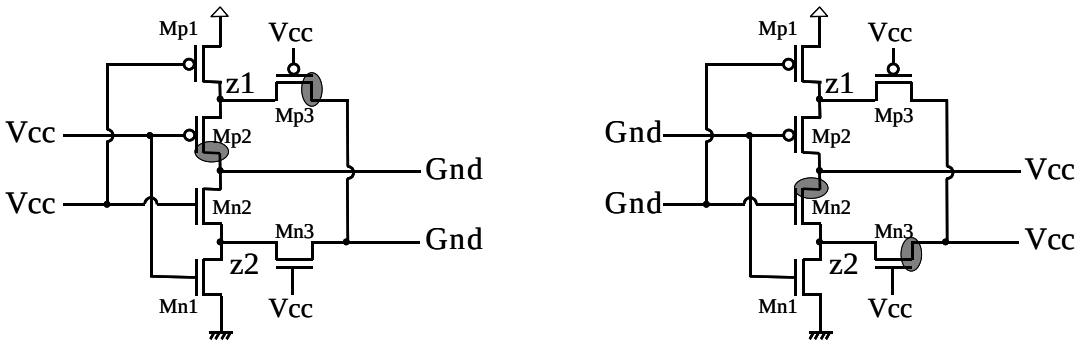


Figure 2 : Zones sensibles de l'inverseur HZ en fonction des entrées

Le durcissement repose aussi sur la généralisation de l'utilisation des inverseurs HZ et sur l'hypothèse que les deux sorties ne peuvent être corrompues simultanément par un même transitoire. Ce dernier résultat, la non propagation d'une erreur d'une sortie à une autre, est obtenu par un jeu adéquat sur les largeurs de canal des transistors. Le transistor Mp1 a un dimensionnement supérieur à ceux de Mp2 et Mp3 qui sont identiques, et de même Mn1 a une largeur de canal plus importante que celles identiques de Mn2 et Mn3.
Si on impose un niveau logique haut aux entrées de l'inverseur durci, alors, ses sorties sont au niveau bas et les zones sensibles sont les drains des deux PMOS bloqués Mp2 et Mp3 (cf. fig. 2). Si une particule radiative traverse le drain sensible de Mp3, elle induit un transitoire de tension vers le niveau haut sur la sortie S2 à laquelle le drain est directement connecté. Pour être en mesure d'affecter également l'état de la sortie S1, ce transitoire doit se propager à travers les deux NMOS passant Mn3 puis Mn2 via le nœud z2. Or, z2 est "collé" à la masse par le transistor passant Mn1 et sa largeur de canal est supérieure à celle de Mn3. La tension de z2 reste donc au niveau bas. Le transitoire ne peut donc pas se propager de S2 en S1 via z2, de même via Mp3 et Mp2 qui sont bloqués. Le même mécanisme est à l'œuvre quels que soient les niveaux logiques des E/S et la sortie erronée. Ainsi, la présence d'un transitoire de tension sur l'une des sorties ne peut pas corrompre le niveau logique de l'autre. Cette propriété et l'efficacité du durcissement aux transitoires des inverseurs HZ sont illustrés par

l'exemple d'application suivant.

Application à un arbre d'horloge

L'inverseur HZ peut être utilisé pour réaliser des interconnexions bufferisées et des arbres d'horloge durcis. C'est à travers ce dernier exemple que nous avons choisi d'illustrer son efficacité. Un arbre d'horloge, figure 3, similaire à celui présenté au 2.3.B.1 a été réalisé au moyen d'inverseurs HZ.

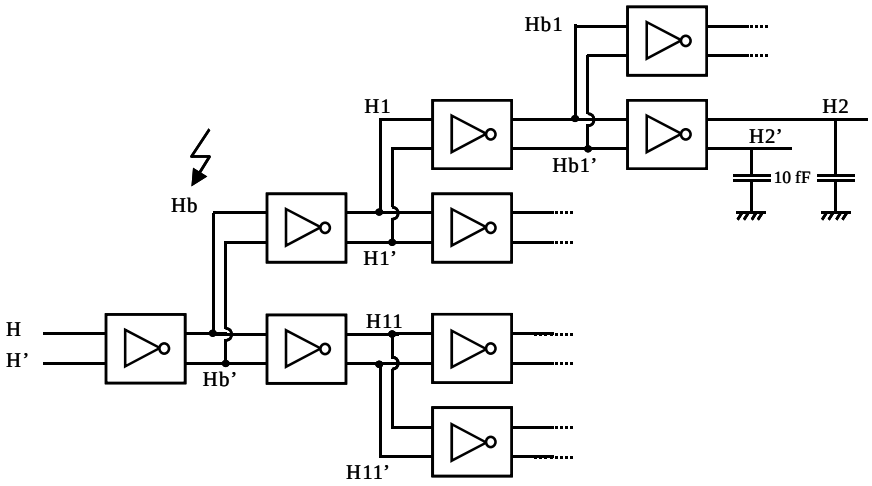


Figure 3 : Arbre d'horloge et localisation du transitoire

Un layout[1] de l'inverseur HZ a été dessiné et les capacités parasites extraites en vue des simulations électriques. La largeur de grille W de ses différents transistors est donnée par la table 2 (cf. fig. 1 pour leurs noms respectifs).

	Mp1	Mp2, Mp3	Mn1	Mn2, Mn3
W (µm)	2	1,7	1,2	1

Table 2 : Largeur de grille des transistors de l'inverseur HZ

Un grand nombre de simulations de transitoires ont été effectuées pour une fréquence d'horloge de 200 MHz. La figure 4 donne les résultats de la simulation d'un transitoire vers le niveau bas sur le nœud Hb (flèche brisée de la fig. 3). Le courant d'aléa injecté a une amplitude de 2 mA, 50 ps de temps de montée et de descente et un plateau de 100 ps.

[1] Une technologie CMOS 0,25 µm commerciale a été utilisée (DKhcmos7 de ST Microelectronics)

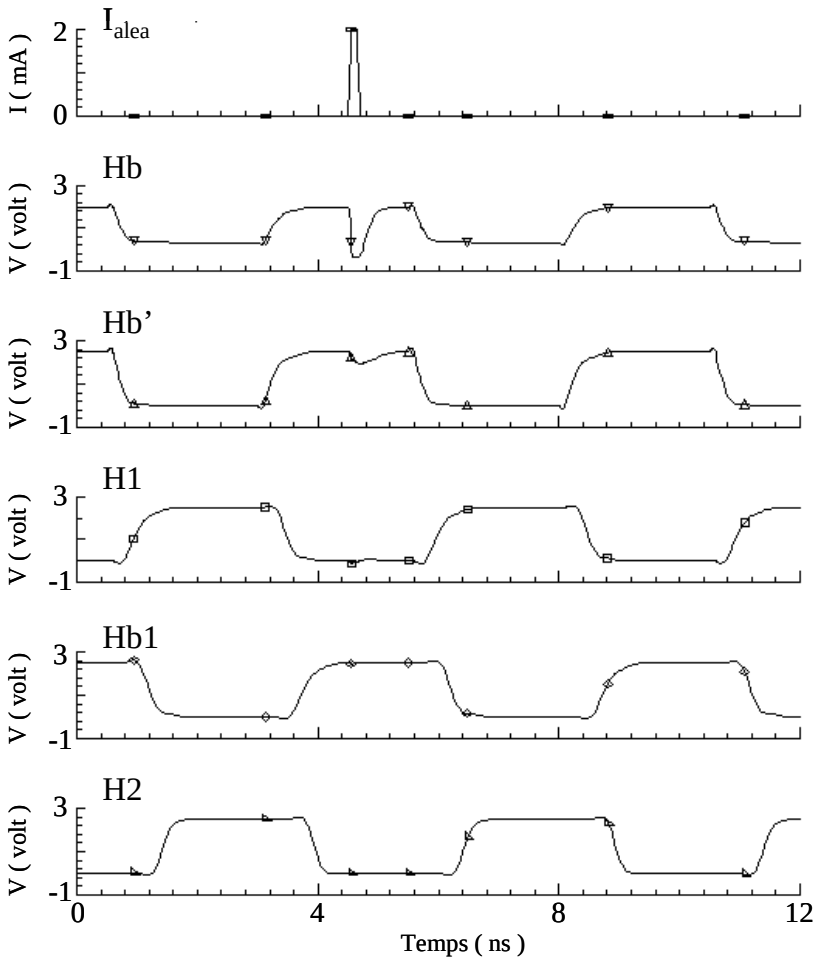


Figure 4 : Transitoire en Hb de l'arbre d'horloge

Pendant toute la durée du transitoire le nœud Hb passe au niveau bas, tandis que le potentiel du nœud Hb' ne subit qu'une très légère décroissance. Hb' reste au niveau logique haut. Le transitoire n'engendre pas d'erreur sur la deuxième sortie du premier inverseur HZ. Ainsi, l'erreur ne se propage pas le long de l'arbre d'horloge. Les nœuds H1, Hb1 et H2 ont des niveaux logiques conformes au fonctionnement normal. Les courbes de potentiel des autres branches de l'horloge ne sont pas représentées, elles sont identiques et exemptes de toute perturbation. Toutes les possibilités de transitoires (vers 0, vers 1, sur Hb et sur Hb') ont été simulées sans que la propagation d'une erreur puisse être mise en évidence.

Une autre série de test a été réalisée en statique sur le même circuit pour des transitoires de courant de 2 mA d'amplitude et des durées de plateau poussées à 5000 ps. Les perturbations n'ont pas été transmises. Cette durée des impulsions dépasse de loin des valeurs réalistes de transitoires, les impulsions les plus longues étant en général donnés pour une durée de l'ordre

de la nanoseconde, mais ils permettent de mettre en évidence l'immunité des inverseurs HZ à toute propagation des SET.

Comparaison avec un inverseur classique

L'inverseur HZ est constitué de six transistors contre deux pour un inverseur classique. Il occupe par conséquent une surface plus importante. La figure 5 présente les layouts respectifs d'un inverseur HZ et d'un inverseur classique réalisé dans la même technologie CMOS 0,25 µm.

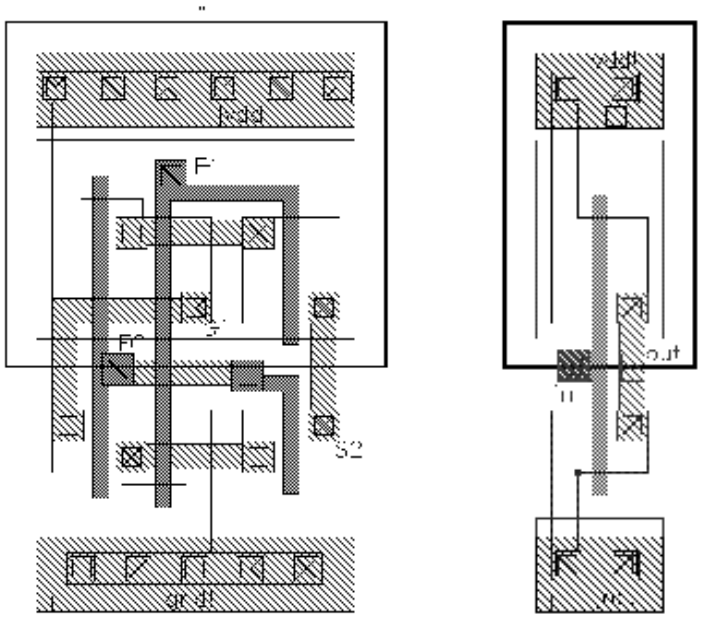


Figure 5 : Layouts de l'inverseur HZ et d'un inverseur standard

L'inverseur HZ a une surface de 45 μm^2 soit deux fois et demi les 18 μm^2 de surface occupée par un inverseur simple. En terme de temps de propagation la comparaison est également au détriment de l'inverseur HZ. Pour mesurer les temps de propagation nous avons utilisé une première chaîne de trois inverseurs standards identiques (avec des largeurs de canal W_{Mp}=1,7 µm pour le PMOS et W_{Mn}=1 µm pour le NMOS) et une deuxième chaîne de trois inverseurs HZ identiques (cf. table 2 pour les tailles de transistors). Le temps de propagation a été mesuré entre les points moyens des fronts des signaux d'entrée et de sortie de l'inverseur central dans chacun des deux cas. Les temps de propagation les plus élevés correspondent aux transitions des sorties du niveau bas vers le niveau haut. La mesure donne 80 ps pour l'inverseur standard et 250 ps pour l'inverseur HZ. La disproportion en terme de surface et de délai peut sembler importante, mais seul l'inverseur HZ est durci. Si on veut obtenir un niveau comparable de durcissement aux transitoires pour des inverseurs classiques,

il faut utiliser des techniques de redondance triple avec vote telles que celles décrites au III.A. On arrive alors à des coûts en surface et en délai bien supérieurs à ceux de l'inverseur HZ comme le met en évidence la table 3.

	Délai (ps)	Surface (μm^2)
Inverseur	80	18
Voteur	230	110
TMR	310	384
Inverseur HZ	250	45

Table 3 : Comparaison inverseur HZ, inverseur et TMR

La table 3 donne les pires cas en terme de retard (pire temps de traversée) et de surface de portes. Sous le nom TMR, on présente la structure logique constituée de voteurs et d'inverseurs permettant le durcissement aux transitoires (cf. fig. 12 de la partie III.A). En la matière, trois inverseurs et trois voteurs sont nécessaires pour atteindre un durcissement égal à celui fourni par l'inverseur HZ. Le délai correspond alors au pire temps de traversée d'un inverseur puis d'un voteur et la surface à celle de trois inverseurs et trois voteurs. L'inverseur HZ permet un gain en délai de 20 % et en surface de 88 %.

En milieu radiatif, l'inverseur HZ apporte une solution au problème des transitoires de tension. Il permet de réaliser des interconnexions et des arbres d'horloge durcis, et ce pour un coût en surface bien moindre que celui nécessaire pour parvenir à un durcissement similaire avec des inverseurs classiques et une architecture de type TMR. Les propriétés de l'inverseur HZ sont utilisées dans la partie suivante pour réaliser un point mémoire durci.

B.1.2. La CSRAM-HZ.

La forte sensibilité des CSRAM aux aléas logiques constitue la principale entrave à l'emploi des circuits reconfigurables en milieu radiatif. Chaque erreur entraîne en effet une modification de la fonctionnalité du circuit et impose une réécriture de la configuration afin de retrouver un fonctionnement normal. Le maintien de l'intégrité de l'information de la couche de configuration peut être obtenu, par exemple, par l'utilisation de points mémoire

durcis. Nous avons réalisé de telles CSRAM durcies en tirant partie des propriétés de l'inverseur HZ vues dans la partie précédente.

Principe

L'utilisation d'inverseurs HZ donne son nom à cette nouvelle cellule : la CSRAM-HZ. Son architecture est calquée sur celle d'un point mémoire classique à cinq transistors, à savoir deux inverseurs rebouclés et un transistor d'accès de type NMOS. La figure 6 donne l'architecture au niveau transistor de la CSRAM-HZ.

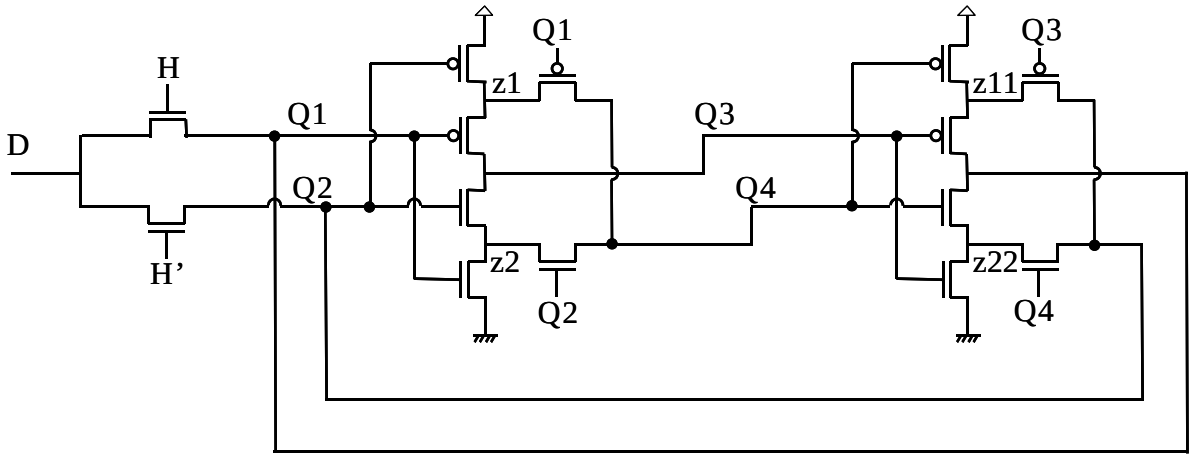


Figure 6 : Architecture de la cellule CSRAM-HZ

On y trouve deux inverseurs HZ rebouclés et deux transistors d'accès de type NMOS afin de conserver le caractère redondant de l'information.

Ce point mémoire possède des zones sensibles susceptibles de donner naissance à des transitoires de tension lorsqu'elles sont traversées par une particule radiative ionique. Ces zones varient en fonction de l'état logique de la CSRAM-HZ selon les règles définies au 2.2.B.2. On dira que cette cellule est dans l'état 1 lorsque Q1=Q2=1 avec Q3=Q4=0 et dans l'état 0 lorsque Q1=Q2=0 avec Q3=Q4=1.

Le durcissement repose sur la redondance double de l'information et sur la non propagation des transitoires. Pour illustrer pas à pas le principe du durcissement, considérons la cellule durcie dans l'état 1, en statique (H=H'=0 entraîne le blocage des transistors d'accès). Et étudions, par exemple, les effets du passage d'un ion lourd qui induit un aléa vers l'état logique bas du nœud Q1. Pendant toute la durée de l'aléa, le nœud Q1 passe transitoirement au niveau logique bas. Et, le nœud Q2 reste au niveau logique haut (la partie précédente a mis en évidence la non propagation d'un transitoire d'une des sorties d'un inverseur HZ vers la seconde). Les nœuds Q1 et Q2 sont les entrées du premier inverseur HZ. Celui-ci ayant alors

ses deux entrées dans des états logiques différents (Q1=0 et Q2=1) impose une mise en haute impédance à ses sorties Q3 et Q4. Les nœuds Q3 et Q4 conservent donc le même niveau logique bas inchangé. Mais Q3 et Q4 sont aussi les entrées du deuxième inverseur. Donc, Q3=Q4=0 impose un forçage de ses sorties au niveau haut. Ainsi, dès que le courant d'aléa qui impose un niveau logique bas sur Q1 cesse, celui-ci retrouve un niveau logique haut conforme à l'état de la CSRAM-HZ. La structure de cette cellule empêche la propagation d'une erreur à plus d'un nœud simultanément et fournit un mécanisme de recouvrement du niveau logique correct.

Le même principe est à l'œuvre indifféremment pour les transitoires vers 1 ou vers 0 et ce quel que soit le nœud Q1, Q2, Q3 ou Q4 concerné.

Nous conclurons donc qu'en statique aucun événement singulier ne peut modifier l'état de la CSRAM-HZ qui est complètement immunisée.

Deux remarques doivent compléter la présentation de la CSRAM-HZ:

- les aléas logiques dynamiques n'ont pas encore été considérés. L'écriture d'un point mémoire représente en effet une fraction très courte de son temps d'utilisation. Après écriture, l'intégrité de la couche de configuration est vérifiée et les éventuelles erreurs corrigées.
- l'effet des transitoires sur les horloges H et H' n'a pas été présenté. Ce cas se ramène en effet à celui de l'aléa statique. Un transitoire sur H, par exemple, peut entraîner le déblocage du transistor d'accès correspondant et la corruption de l'état logique du nœud Q1 par celui du nœud D pendant toute la durée du transitoire. On retrouve effectivement un cas similaire à celui d'un aléa sur Q1 en statique.

Simulation

Une série de simulations électriques permet de confirmer l'efficacité du durcissement de la CSRAM-HZ aux aléas logiques.

CSRAM-HZ	Mp1	Mp2, Mp3	Mn1	Mn2, Mn3	Mt
W (µm) 1er inverseur	2	1,7	1,2	0.9	1,5
W (µm) 2ème inverseur	2,1	1,4	1,2	1	

Table 4 : Largeur de grille des transistors de la CSRAM-HZ

La table 4 donne les tailles de largeur de grille utilisées dans la cellule[1] que nous avons dessinée. La diversité des largeurs de grille utilisées résulte de modifications successives après simulation pour obtenir un durcissement optimal.

Le nœud Q3 a été relié à la grille d'un PMOS afin de représenter le switch commandé par la CSRAM-HZ. Les simulations ont été effectuées après extraction des capacités parasites du layout. Elles mettent en évidence l'efficacité des techniques employées. Nous présentons ici deux simulations typiques parmi toutes celles réalisées.

Dans la première, le point mémoire est à l'état 1 (Q1=Q2=1 avec Q3=Q4=0). Un courant d'aléa engendrant le passage du nœud Q1 au niveau bas est injecté. La durée de l'impulsion de courant a été poussée à 5000 ps, avec des temps de montée et de descente de 50 ps et une amplitude de 2 mA. Le résultat de simulation est donné figure 7.

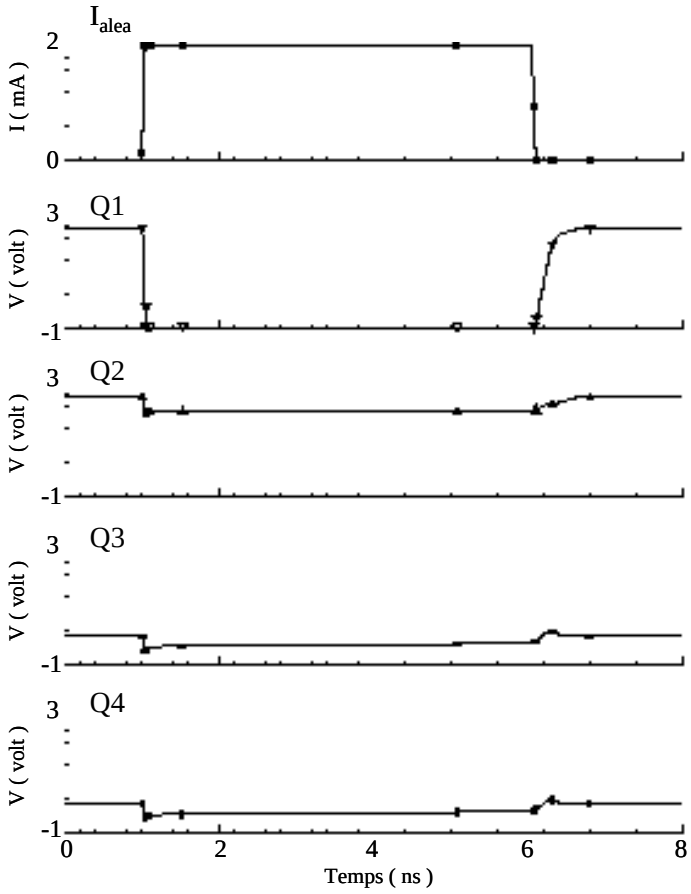


Figure 7 : Aléa vers 0 sur Q1

Dès le début de l'impulsion de courant d'aléa, le potentiel de Q1 passe au niveau bas. Il y restera pendant toute la durée d'injection du courant d'aléa. Le potentiel du nœud Q2 subit

[1] Une technologie CMOS 0,25 µm commerciale a été utilisée (DKhcmos7 de ST Microelectronics)

une légère décroissance, celle-ci n'affecte cependant pas son niveau logique qui reste haut. Les potentiels des nœuds Q3 et Q4 subissent également une petite perturbation due au passage en haute impédance qui n'affecte pas leur niveau logique bas. Pendant toute la durée de l'aléa, l'erreur ne se propage pas à Q2 et comme on peut le constater, le passage en haute impédance de Q3 et Q4 préserve leur état. Ainsi, dès la fin de l'impulsion de courant Q3=Q4=0 impose le retour du nœud Q1 à un niveau logique haut.

La deuxième simulation présentée ci-dessous a été réalisée avec une CSRAM-HZ dans l'état 0. Un courant d'aléa a été injecté au nœud Q1 et il a engendré un passage de son potentiel au niveau haut. L'impulsion de courant considérée a une amplitude de 2 mA, des temps de montée et de descente de 50 ps et à nouveau un plateau de 5000 ps. La figure 8 donne le comportement des potentiels des différents nœuds de la cellule.

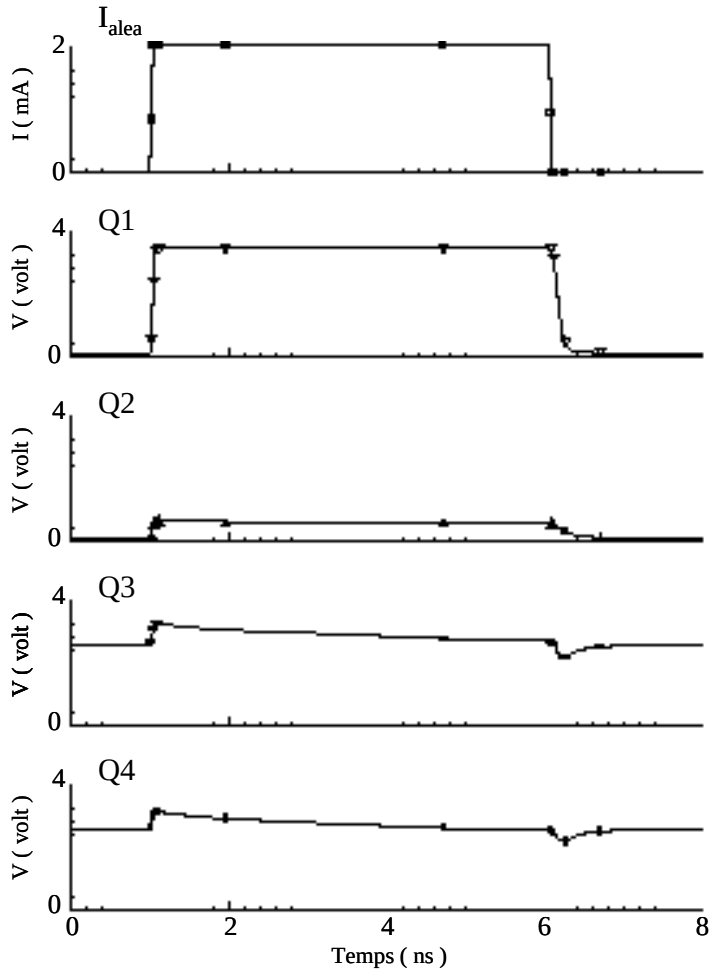


Figure 8 : Aléa vers 1 sur Q1

Elle met en évidence, comme précédemment, la non propagation de l'erreur à un autre nœud que Q1, de même que le retour de Q1 à un niveau logique bas dès la fin de l'impulsion de

courant. Ceci grâce à la préservation des niveaux de potentiel de Q3 et Q4. La CSRAM-HZ a conservé un état correct.
Toutes les possibilités d'aléas ont été testées pour tous les nœuds de cette cellule, et pour des courants d'aléa similaires. Aucune erreur n'a pu être mise en évidence.

La durée d'aléa utilisée pour ces simulations, 5000 ps, est bien au delà des durées rapportées dans la littérature scientifique. Les aléas d'une durée supérieure à la nanoseconde sont considérés comme peu probables. Cette durée ne correspond pas à une réalité physique possible. Mais dans le cas d'une analyse selon le principe du « pire cas » elle illustre l'impossibilité d'occurrence d'un aléa logique dans une CSRAM-HZ. Ce constat est fait sur la base de résultats de simulations électriques.

Comparaison avec une CSRAM classique

La CSRAM-HZ comporte 14 transistors. Soit quasiment le triple d'une CSRAM classique à cinq transistors. Elle occupe en conséquence une surface bien plus importante. La figure 9 donne les layouts respectifs d'une CSRAM-HZ et d'une CSRAM classique.

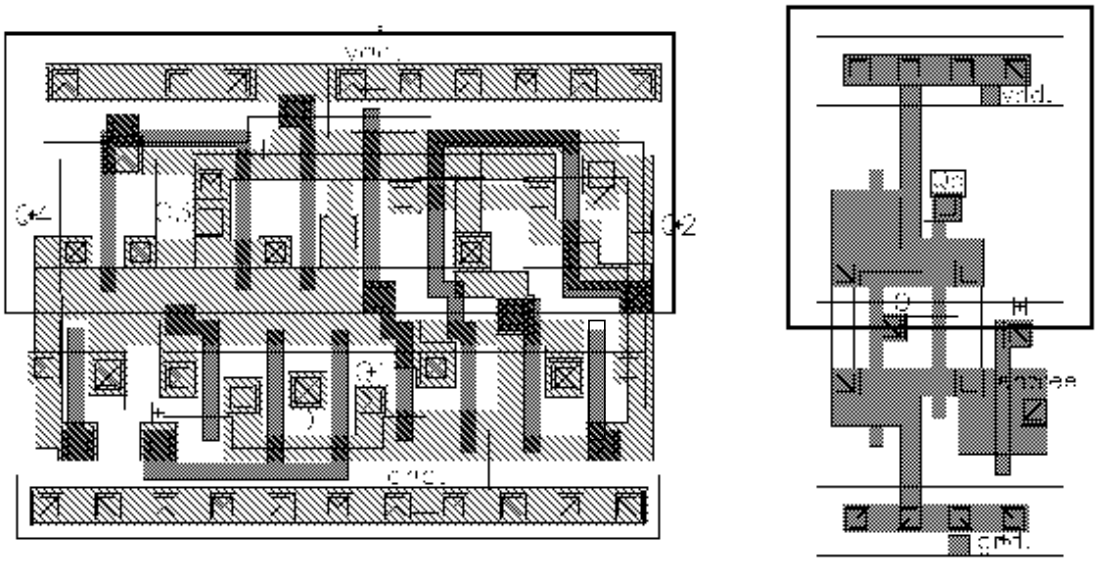


Figure 9 : Layouts d'une CSRAM-HZ et d'une CSRAM classique à 5 transistors

La cellule durcie occupe une surface de 68,6 μm^2 contre 38,5 μm^2 pour le point mémoire standard (ces surfaces pourraient probablement être d'avantage optimisées). Cela représente une surface additionnelle de 78 %. Les résultats de la comparaison entre une CSRAM à cinq transistors (CSRAM5T) et une CSRAM-HZ sont reportés dans la table 5.

	Nb. transistors	Surface (μm^2)	Durci
CSRAM5T	5	38,5	non
CSRAM-HZ	14	68,6	oui

Table 5 : Comparaison CSRAM5T –CSRAM-HZ

La CSRAM-HZ occupe une surface plus importante que la CSRAM5T mais elle apporte une solution au problème des aléas logiques.

Conclusion

L'utilisation de CSRAM-HZ dans la couche de configuration d'un circuit reconfigurable permet de garantir son intégrité vis à vis des aléas logiques. Mais au prix d'un coût en surface important. Un FPGA comporte en effet plusieurs centaines de milliers de CSRAM. Ce coût peut sembler prohibitif, aussi proposons nous, dans la partie B, une deuxième approche de durcissement, bien plus économique en surface, pour la couche de configuration. Un argumentaire permettant le choix entre les deux approches proposées est présenté à la fin de cette partie.

B.1.3. La Latch-HZ.

La bascule D est l'un des principaux éléments constitutifs des circuits intégrés et plus particulièrement des circuits reconfigurables. Aussi, de nombreuses bascules durcies ont elles été proposées par le passé [ROC88], [CAN91], [WHI91], [LIU92], [BES93], [VEL94], [SAL99], [MON99] avec leurs qualités et leurs défauts (cf. partie 3.1.). Nous proposons ici une nouvelle architecture basée sur une restructuration de la bascule au niveau des transistors. Elle reprend les principes utilisés pour la CSRAM-HZ et comporte deux inverseurs HZ [DUT01c]. Ceux-ci lui donnent son nom : la Latch-HZ.

Après avoir donné les principaux mécanismes de durcissement aux aléas logiques, nous rapportons les résultats de simulation électrique confirmant leur efficacité. Et avant de conclure sur l'intérêt et sur les perspectives ouvertes par la Latch-HZ, nous la comparons à une bascule D classique.

Principe

Une bascule D est un élément de mémorisation sensible aux niveaux logiques de l'horloge. La bascule HZ s'inspire de la bascule D classique présentée au II.C.2.3. Cependant, les inverseurs ont été remplacés par des inverseurs HZ et un certain nombre de modifications nécessaires au maintien de la dualité de l'information a été réalisé. Le résultat obtenu est donné figure 10.

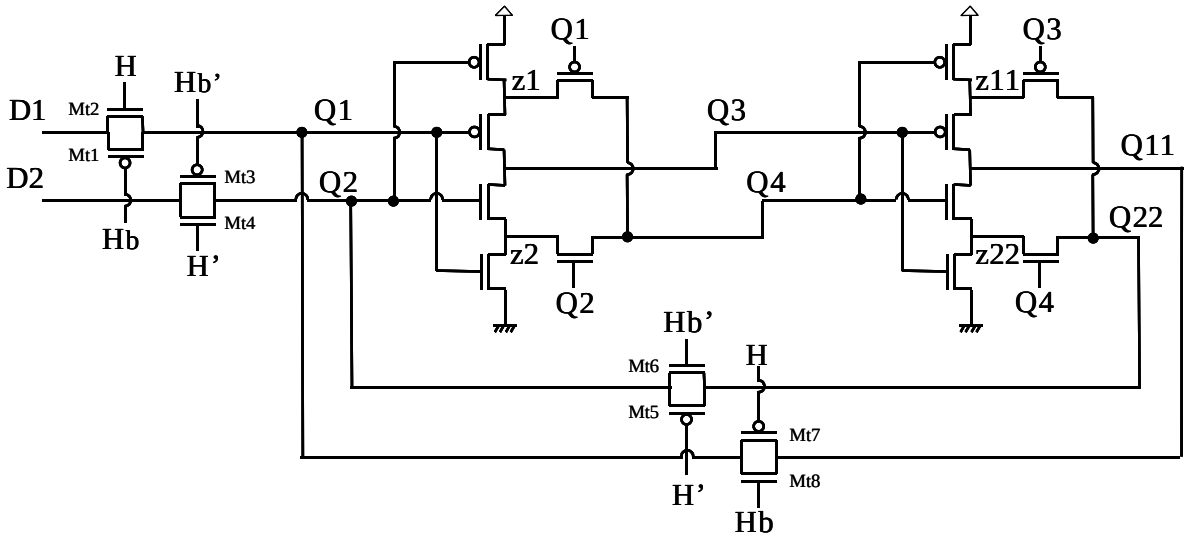


Figure 10 : Architecture au niveau transistor de la Latch-HZ

La bascule durcie HZ compte également quatre transistors d'accès (Mt1, Mt2, Mt3 et Mt4) permettant de connecter les deux entrées de donnée D1 et D2 respectivement aux nœuds Q1 et Q2 et quatre transistors de mémorisation (Mt5, Mt6, Mt7 et Mt8) permettant la fermeture ou l'ouverture de la boucle de rétroaction de la bascule. Ces huit transistors sont commandés par les signaux d'horloge H, H' , Hb et Hb' issus d'une horloge durcie à base d'inverseurs HZ telle que celle présentée précédemment au 3.2.A.1.

La bascule HZ a deux modes de fonctionnement : un mode écriture et un mode mémorisation ; ces deux modes sont définis par les niveaux logiques de l'horloge.

Pour H=H'=1 et Hb=Hb'=0, la Latch-HZ est en mode écriture (les transistors d'accès sont passants et les transistors de mémorisation bloqués) : les données présentes en entrée (D1=D2 en fonctionnement normal) sont écrites dans la bascule. Les nœuds Q1, Q2, Q11 et Q22 prennent le niveau logique des noeuds D1 et D2, tandis que Q3 et Q4 passent dans l'état logique complémentaire.

Pour H=H'=0 et Hb=Hb'=1, la Latch-HZ est en mode mémoire (les transistors d'accès sont bloqués et les transistors de mémorisation passants) . De cette façon, les deux inverseurs HZ

sont rebouclés et l'état logique de la bascule est maintenu dans un état stable jusqu'à la prochaine phase d'écriture.
Le principe de durcissement aux aléas logiques est similaire à celui décrit pour la CSRAM-HZ . Il repose sur une redondance simple de l'information et sur le maintien d'un état logique correct sur deux nœuds de données grâce au passage en haute impédance. Une bascule logique peut subir trois types distincts d'évènements singuliers : aléa statique, aléa dynamique et transitoire de tension sur un signal d'horloge.

Simulations

Un aléa statique est susceptible de se manifester lorsque la bascule HZ est en mode mémoire. Les transistors d'accès sont alors bloqués et les transistors de mémorisation passants (H=H'=0 et Hb=Hb'=1). La configuration de la cellule correspond à celle de la CSRAM-HZ en mode mémoire et les mécanismes de durcissement sont identiques à ceux décris dans la partie A.2. Toutes les possibilités d'aléas ont été testées sur les différents nœuds du circuit. Elles mettent en évidence la même immunité aux aléas statiques. Les impulsions de courant injectées pour une amplitude de 2 mA et pour une durée allant jusqu'à 5000 ps n'ont pas mis en évidence la moindre perte d'information. La Latch-HZ est durcie aux aléas statiques.

Le deuxième type d'aléa susceptible d'affecter une bascule D est l'aléa dynamique. Il se produit lors du passage de la cellule du mode écriture au mode mémoire. Son mécanisme a été présenté en détail pour une bascule D classique dans la partie 2.3.B.3. Il a pour origine un transitoire de tension se propageant sur l'entrée de la bascule au moment du front descendant de l'horloge. La bascule mémorise alors un état erroné. L'utilisation d'entrées duales pour la Latch-HZ permet d'éviter le basculement de la cellule et la mémorisation d'une erreur, à la condition que le transitoire n'affecte qu'une seule des deux entrées. Prenons l'exemple où la Latch-HZ est dans l'état 1 avant de mémoriser l'état 0. Dans ce cas, les entrées de données (D1 et D2) passent à 0 avant le front montant de l'horloge. Lorsque celui-ci survient, la Latch-HZ passe à l'état 0 (Q1=Q2=0 et Q3=Q4=1), état qui est mémorisé lors du front descendant de l'horloge en fonctionnement normal. Si un transitoire de tension se produit en amont dans le circuit avant le front descendant de l'horloge et induit en se propageant un passage de D1 au niveau haut, alors, le nœud Q1 passe lui aussi au niveau haut (les transistors d'accès sont passants). Mais Q2 n'est pas affecté par cette erreur, il reste au niveau bas. Les sorties Q3 et Q4 du premier inverseur HZ passent alors en haute impédance, elles

conservent inchangé le niveau logique haut. Lorsque le front descendant de l'horloge a finalement lieu, les transistors d'accès se bloquent et D1 n'impose plus un niveau haut sur Q1. Au même moment, les transistors de mémorisation deviennent passants. Q3 et Q4 imposent un niveau bas, via le deuxième inverseur HZ et Q11 sur Q1. Celui-ci retrouve donc le niveau logique bas attendu. Le transitoire de tension sur D1 n'a pas entraîné la mémorisation d'un état erroné par la bascule HZ comme cela aurait été le cas pour une bascule D non durcie. Des mécanismes similaires sont à l'œuvre pour durcir la Latch-HZ contre tout aléa dynamique et ce, quelle que soit l'entrée victime d'un transitoire de tension.

Toutes les simulations électriques correspondantes ont été réalisées[2]. La figure 11 présente les courbes de la simulation électrique correspondant au cas décrit précédemment pour une impulsion de courant de 2 mA d'amplitude, 50 ps de temps de montée et de descente et 500 ps de durée. Cette impulsion induit sur D1 un transitoire de tension dirigé vers le niveau haut et situé 200 ps avant le front descendant de l'horloge.

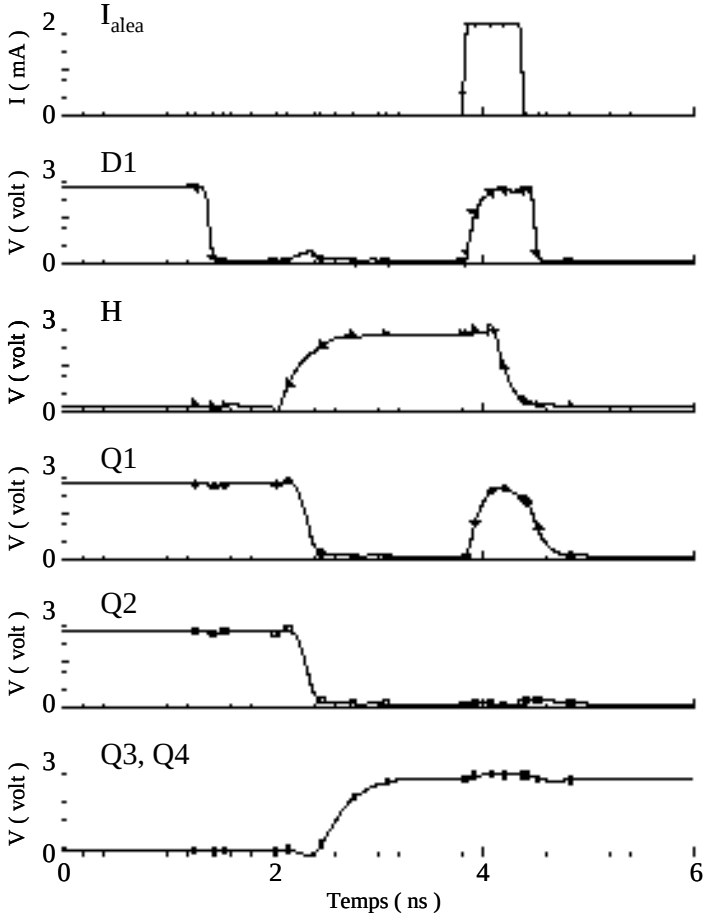


Figure 11 : Simulation d'un aléa dynamique

[2] Une technlogie CMOS 0,25 µm commerciale a été utilisée après extraction des capacités parasites (DKhcmos7 de ST Microelectronics)

On constate effectivement que la présence de ce transitoire de tension se propage sur le nœud Q1. Dans le même temps, les niveaux logiques des nœuds Q2, Q3 et Q4 ne sont pas altérés. Et dès le passage de l'horloge (H) au niveau bas, le nœud Q1 retrouve le niveau logique bas attendu. En effet, les transistors d'accès sont alors bloqués et l'état de D1 n'a plus d'influence sur Q1 et la mise en conduction des transistors de mémorisation assure la correction de l'erreur. La Latch-HZ mémorise bien l'état 0 attendu.
Quel que soit le type d'aléa dynamique considéré, les simulations mettent bien en évidence le durcissement de la Latch-HZ .

Le troisième type d'erreur pouvant affecter une bascule D résulte de la propagation de transitoires de tension dans les arbres d'horloge. Lors du mode mémoire d'une bascule D classique, un transitoire sur son horloge entraîne son passage en mode écriture. Un bit de donnée différent de celui mémorisé peut alors être écrit. Et dès la fin du transitoire, ce bit erroné est mémorisé après retour en mode mémoire. Ce mécanisme d'aléa logique a été décrit en détail au II.C.2.3.
La solution proposée pour le schéma de connexion des signaux d'horloge H, H', Hb et Hb' sur la Latch-HZ ainsi que les propriétés intrinsèques de cette dernière lui permettent de ne pas être affectée par les transitoires de tension de son horloge. En effet, l'utilisation d'un arbre d'horloge à base d'inverseurs HZ (cf. III.B.1.1) dont la partie finale est présentée figure 12 permet de limiter simultanément les conséquences des transitoires à un seul nœud de la Latch-HZ .

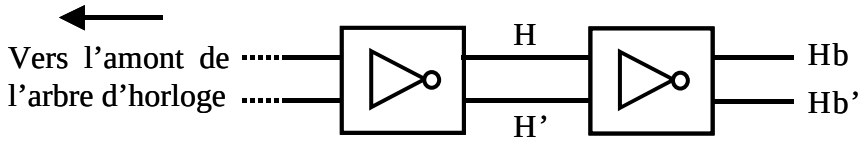


Figure 12 : Génération des signaux d'horloge de la Latch-HZ

Pour présenter le principe de durcissement aux transitoires sur l'horloge, considérons le cas où la Latch-HZ est à l'état 1 (Q1=Q2=1 avec Q3=Q4=0), est en mode mémoire (H=H'=0 et Hb=Hb'=1) et où les entrées D1 et D2 de la bascule sont au niveau bas. Considérons qu'il se produit alors un transitoire de tension vers le niveau haut en H. Le transistor d'accès Mt2 (cf. fig. 10) devient passant et le transistor de mémorisation Mt7 se bloque. Les autres transistors

d'accès et de mémorisation ne sont pas affectés car le transitoire ne peut pas se propager aux nœuds H', Hb et Hb' comme nous l'avons montré au A.1. Le blocage de Mt7 affaiblit la boucle de rétroaction qui assurait le maintien du niveau haut de Q1 et la mise en conduction de Mt2 permet au nœud D1 d'imposer son niveau bas à Q1. Cependant, le niveau logique haut de Q2 n'est pas affecté. Les transistors d'accès qui l'isolent de D2 restent bloqués et les transistors de sa boucle de rétroaction restent passants. Avec Q1 et Q2 dans deux niveaux logiques différents, les sorties du premier inverseur HZ passent en haute impédance et conservent un niveau logique bas inchangé. Dès la fin du transitoire de tension sur H et son retour au niveau bas, Mt2 se bloque à nouveau et Mt7 redevient passant. Ainsi, simultanément, l'influence de D1 sur Q1 disparaît et la boucle de rétroaction est rétablie. Le nœud Q1 retrouve donc rapidement un niveau logique bas, réimposé par le deuxième inverseur HZ via les transistors de mémorisation. Cet exemple met en évidence le principe assurant le durcissement de la Latch-HZ aux transitoires de tension sur l'horloge. Il se vérifie de façon similaire quel que soit le signal d'horloge concerné H,H',Hb ou Hb'.

Une vérification de ce principe a été réalisée par le biais de simulations électriques[2]. La figure 13 présente les principaux signaux permettant d'illustrer le cas décrit précédemment.

Le transitoire de tension simulé sur le nœud H de l'horloge est créé par l'injection d'une impulsion de courant ($I_{aléa}$) d'amplitude 2 mA, de 50 ps de temps de montée et de descente et de 300 ps de temps de plateau. Pendant toute la durée de l'impulsion, H passe effectivement au niveau haut et le nœud H' reste au niveau logique bas. La mise en conduction du transistor d'accès Mt2 qui en résulte entraîne un passage de Q1 au niveau bas (cela est dû au niveau bas de D1). Mais ni Q2, ni Q3 et ni Q4 ne subissent une évolution de leur niveau logique. Dès la fin du transitoire la cause de la perturbation de Q1 disparaît (Mt2 se bloque à nouveau) et un phénomène de guérison (Mt7 qui était bloqué redevient passant), dû au deuxième inverseur HZ assure le retour de Q1 au niveau haut. L'état de la Latch-HZ n'a pas été modifié par le transitoire.

Toute une série de simulations a été réalisée pour toutes les possibilités de transitoires sur les signaux d'horloge. Dans tous les cas la Latch-HZ s'est révélée durcie vis à vis des transitoires sur l'horloge.

[2] Une technlogie CMOS 0,25 µm commerciale a été utilisée après extraction des capacités parasites (DKhcmos7 de ST Microelectronics)

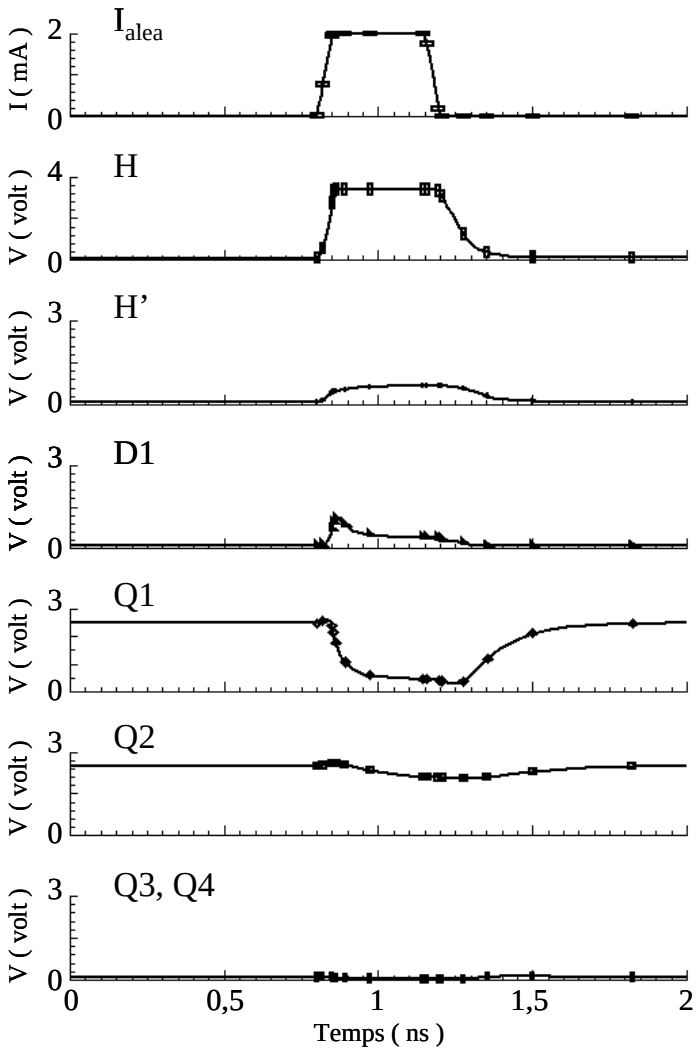


Figure 13 : Aléa sur l'horloge

Comparaison avec une bascule D classique

La Latch-HZ comporte vingt transistors. Tandis qu'une bascule D classique, construite selon le même principe, en compte huit. Cela a un premier impact sur leurs surfaces relatives. La figure 14 présente les layouts respectifs de la Latch-HZ utilisée pour les simulations précédentes et d'une bascule D à huit transistors.

La bascule durcie occupe une surface de 126,5 μm^2 contre 58,5 μm^2 pour la bascule D standard (ces surfaces pourraient probablement être d'avantage optimisées). Cela représente une surface additionnelle de 116 %.

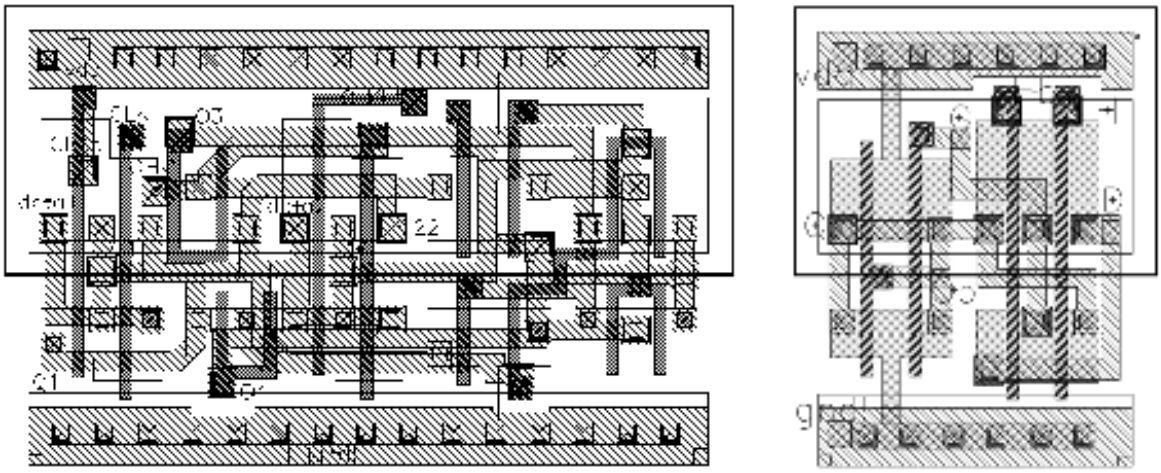


Figure 14 : Layouts de la Latch-HZ et d'une bascule D à huit transistors

La table 6 donne les tailles (en µm) des largeurs de grille retenues pour la Latch-HZ .

Mp1	Mp2, Mp3	Mn1	Mn2, Mn3	Mt1, Mt3	Mt2, Mt4	Mt5, Mt7	Mt6, Mt8
2,2	1,6	1,2	1	2,5	1,5	1,7	1

Table 6 : Largeur de grille des transistors de la Latch-HZ

La comparaison est également au détriment de la Latch-HZ en termes de temps d'écriture. C'est à dire le temps nécessaire pour inverser l'état mémorisé par une bascule après le front d'horloge permettant l'écriture. Dans les deux cas, les simulations électriques réalisées ont montré que le passage de l'état 1 à l'état 0 était le plus long. Ce temps a été mesuré entre le point milieu du front d'horloge et le point milieu des sorties (Q11/Q22 pour la Latch-HZ et Q pour la bascule standard). Les deux bascules étant chargées par des inverseurs. La Latch-HZ est deux fois plus longue à basculer avec un délai de 500 ps dans le pire cas, contre 250 ps pour la bascule D standard. Les différents résultats de la comparaison sont rappelés dans la table 7.

	Nb. transistors	Surface (μm^2)	Délai (ps)	Durci
Bascule Std	8	58,5	250	non
Latch-HZ	20	126,5	500	oui

Table 7 : Comparaison Bascule D – Latch-HZ

Le durcissement de la Latch-HZ se paie d'un surcoût en surface de 116 % et en délai de 100%. Mais elle est durcie vis à vis de tous les types d'aléas possibles, alors que la bascule D est particulièrement vulnérable à ces phénomènes.

Il est possible d'obtenir un durcissement similaire en utilisant plusieurs bascules D classiques et une architecture de type TMR avec vote (cf. partie 3.1.). La figure 15 donne le schéma de principe correspondant. Trois bascules D sont connectées en parallèles à l'entrée de données E et au signal d'horloge H, et leurs trois sorties A,B et C commandent un module de vote majoritaire. Ainsi, si une des trois bascule subie un aléa et délivre une information erronée, celle-ci est éliminée par le voteur.

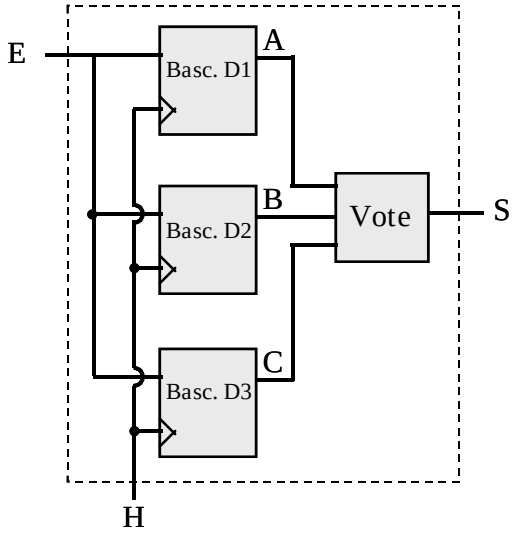


Figure 15 : Principe du durcissement par TMR

Cependant, l'immunité aux aléas logiques obtenue est bien moindre que celle présentée par la Latch-HZ . En effet, le module de vote n'est pas durci, il est sensible aux transitoires de tension. Les aléas dynamiques ou liés à l'horloge ne sont pas pris en compte par cette architecture (usage d'une entrée et d'une horloge uniques).

Et en régime statique, l'éventuelle accumulation d'erreurs dans le triplet de bascules rend inopérant la sélection effectuée par le voteur. La comparaison du surcoût en surface est également au détriment de la TMR. La table 8 donne les surfaces et les délais d'une bascule D, d'un module de vote, de l'implantation de la redondance triple (trois bascules et un voteur) et d'une Latch-HZ .

	Nb. transistors	Surface (μm^2)	Délai (ps)	Durci
Bascule Std	8	58,5	250	non
Voteur	16	110	240	non
Architecture TMR	40	285.5	490	imparfait
Latch-HZ	20	126,5	500	oui

Table 8 : Comparaison TMR – Latch-HZ

L'architecture par redondance triple avec vote entraîne un surcoût en surface de l'ordre de 125% avec un très léger avantage en terme de délai de 2% par rapport à la Latch-HZ, et ce pour un durcissement de bien moindre qualité comme nous venons de le voir.

Conclusion

La Latch-HZ apporte une solution au durcissement des bascules D grâce à une approche par restructuration au niveau transistors. Elle est immunisée aux effets des aléas statiques, dynamiques et des transitoires de tension sur les signaux d'horloge comme l'ont montré de nombreuses simulations. Cette immunité est obtenue au prix d'un doublement de la surface (+ 116%) et du temps d'écriture par rapport à une bascule D classique. Cependant, si la Latch-HZ est comparée à une architecture classique de durcissement telle que la redondance triple avec vote majoritaire, elle présente une amélioration notable du durcissement pour un gain en surface de 125%.

Mais, l'intérêt de la Latch-HZ ne s'arrête pas là, son utilisation permet de mettre au point une nouvelle architecture globale durcie pour les circuits séquentiels : l'architecture HZ. Elle est présentée dans la partie suivante.

B.1.4. L'architecture HZ.

Dans la partie précédente nous avons présenté une bascule D durcie : la Latch-HZ . Ses remarquables propriétés de durcissement aux aléas et aux transitoires de tension ont été mises en évidence. Nous allons maintenant l'utiliser pour réaliser un élément que nous rencontrons fréquemment dans les circuits reconfigurables : une D flip-flop. Cette flip-flop durcie, la flip-flop HZ, nous permet de proposer une nouvelle architecture globale des circuits séquentiels : l'architecture HZ. Nous verrons dans les pages qui suivent que cette architecture est durcie vis à vis des aléas logiques et des transitoires. L'immunité aux évènements singuliers est meilleure que celle obtenue par les techniques de redondance triple et ce pour un coût en surface inférieur.

La flip-flop HZ

Nous nous sommes inspirés d'une architecture classique de D flip-flop, présentée figure 16, constituée de deux bascules D (le Maître et l'Esclave).

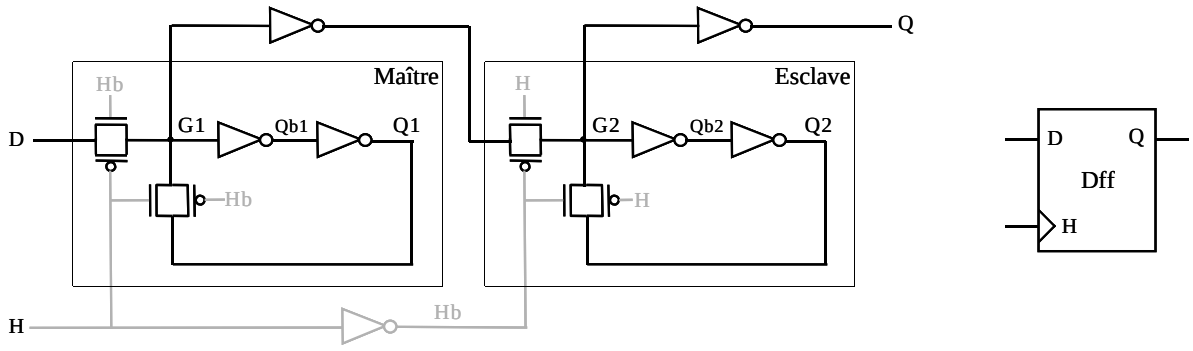


Figure 16 : Architecture de la D flip-flop standard

Trois inverseurs ont également été ajoutés, un afin de générer le signal complémentaire (Hb) de l'horloge (H), et les deux autres pour limiter la dégradation des niveaux logiques des bascules lors des fronts d'horloge.

C'est une structure similaire que nous avons utilisé pour assembler une D flip-flop à base de Latch-HZ : la flip-flop HZ. Son dessin est donné figure 17.

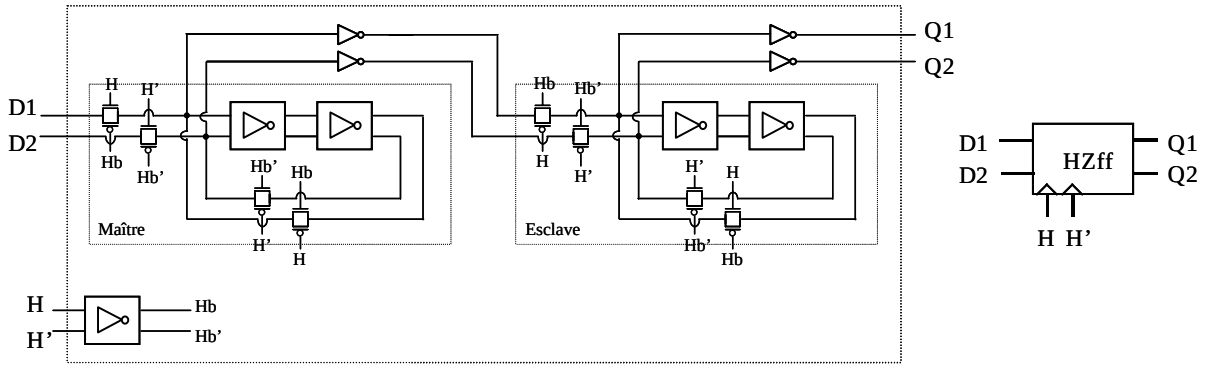


Figure 17 : Architecture de la flip-flop HZ

On retrouve un schéma de principe utilisant une bascule Maître et une bascule Esclave. Un inverseur HZ est utilisé pour générer le complémentaire (Hb et Hb') des signaux d'horloge (H et H'). Quatre inverseurs sont également utilisés pour limiter la dégradation des niveaux logiques des bascules lors des fronts d'horloge.

La flip-flop HZ est susceptible de subir trois type d'aléas logiques : les aléas statiques, dynamiques et les transitoires de tension sur l'horloge. L'utilisation de la Lach-HZ et le respect de la redondance simple des données permettent d'obtenir les mêmes propriétés de durcissement aux aléas statiques et dynamiques que pour la Latch-HZ . Ce résultat est confirmé par toutes les simulations électriques que nous avons réalisées. Elles sont similaires à celles rapportées pour la Latch-HZ dans la partie A.3., aussi nous ne les présentons pas ici. Le durcissement aux transitoires de tension sur les signaux d'horloge est apporté par l'utilisation d'un arbre d'horloge durci à base d'inverseurs HZ , comme cela est le cas également pour la Latch-HZ . Cette nouvelle D flip-flop durcie nous permet de proposer une architecture assurant l'immunité aux événements singuliers des circuits séquentiels.

L'architecture HZ

Jusqu'à présent nous n'avons présenté des solutions de durcissement que pour des éléments particuliers des circuits intégrés. Ceux-ci nous permettent maintenant de proposer une solution globale .

La structure des circuits séquentiels peut être sommairement représentée comme l'alternance de blocs logiques purement combinatoires et de D flip-flop (cf. figure 18).

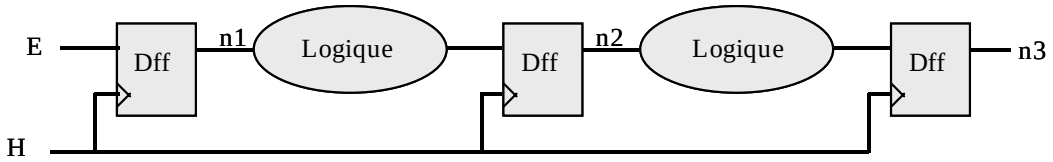


Figure 18 : Structure des circuits séquentiels

Ce type d'architecture est très vulnérable aux différents aléas. Les bascules des D flip-flops sont susceptibles de subir des aléas statiques. Les parties logiques sont sensibles aux transitoires de tension et leur propagation jusqu'aux flip-flops peut entraîner des aléas dynamiques. Et enfin, les signaux d'horloge sont également vulnérables aux transitoires de tension.

Nous proposons une nouvelle architecture des circuits séquentiels utilisant des flip-flops HZ et un arbre d'horloge durci à base d'inverseur HZ, nous l'avons appelée l'architecture HZ. L'utilisation d'éléments durcis et la duplication des parties logiques qui en découle permettent d'obtenir une immunité aux aléas évoqués précédemment, comme diverses simulations électriques le montrent.

Le schéma de principe de l'architecture HZ est donné figure 19.

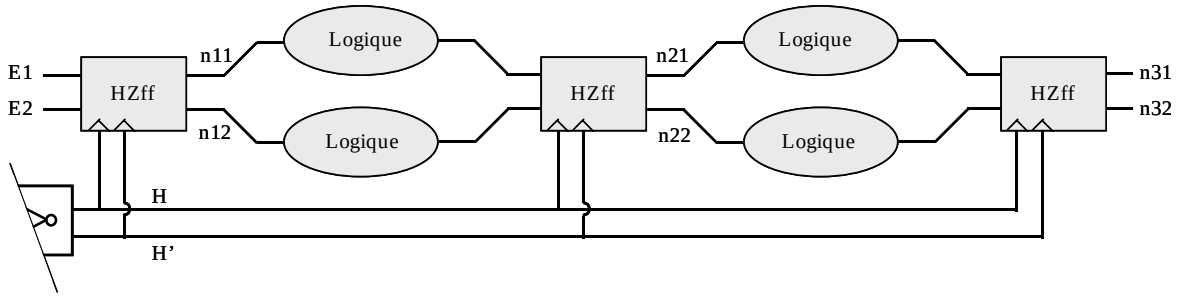


Figure 19 : Architecture HZ des circuits séquentiels

Les D flip-flops de l'architecture classique ont été remplacées par des flip-flops HZ . Pour garantir le maintien de la redondance simple de l'information traitée, les parties logiques sont dupliquées. Et un arbre d'horloge durci à base d'inverseurs HZ est utilisé.

Si on passe en revue les différents éléments utilisés dans l'architecture HZ, on constate qu'elle est effectivement durcie vis à vis de tous les aléas envisageables.

Les transitoires de tension dans l'arbre d'horloge n'entraînent pas d'erreurs au niveau des Latch-HZ comme nous l'avons vu au III.B.1.3.

Les blocs logiques sont également sensibles aux transitoires de tension. Un transitoire peut se propager jusqu'à une des entrées d'une flip-flop HZ . Mais la partie III.B.1.3 a également mis en évidence qu'un aléa dynamique ne pouvait pas en résulter.

Et enfin, les flip-flops HZ sont immunisées aux aléas statiques et dynamiques.

Ainsi, l'architecture HZ est conçue pour fonctionner sans erreurs en environnement radiatif.

Sa principale limitation provient du surcoût en surface nécessaire pour implanter une fonctionnalité par rapport à une architecture classique. Les parties de logique combinatoires sont doublées, l'arbre d'horloge utilisé est plus volumineux et les flip-flops HZ sont plus encombrantes que les D flip-flops classiques. Mais, c'est le prix à payer pour s'affranchir des problèmes liés aux aléas.

Nous allons maintenant la comparer à une architecture durcie par redondance triple pour mettre en évidence son intérêt par rapport à cette dernière.

Comparaison avec le durcissement par redondance triple avec vote majoritaire

La solution de durcissement de la couche logique prônée par les principaux fabricants de FPGA ([ACT97], [XIL01]) est l'utilisation de la redondance triple avec vote. Elle consiste à tripler le nombre de chacun des éléments du circuit (bascules et blocs logiques) et à insérer des modules de vote majoritaire. L'application de cette technique à un circuit séquentiel conduit au schéma de principe présenté figure 20.

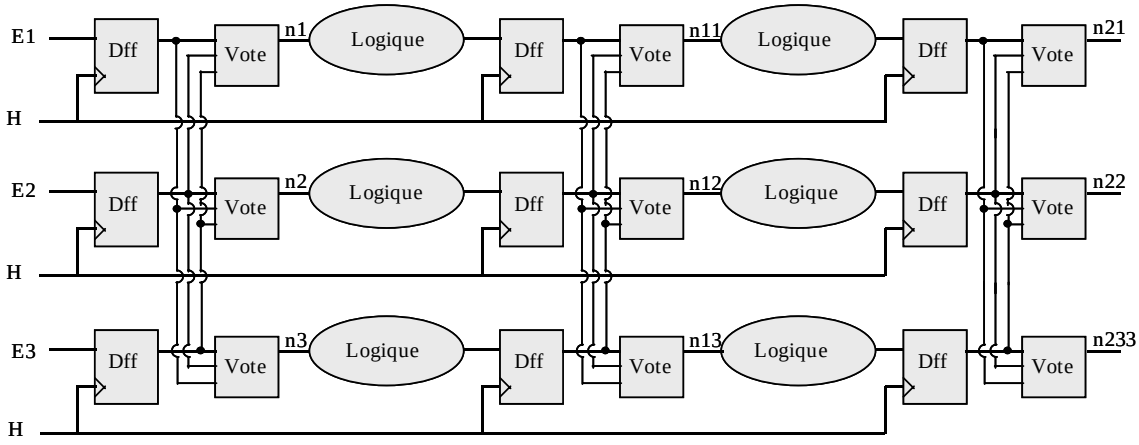


Figure 20 : Structure de circuit séquentiel durci par redondance triple

Les D flip-flops et les blocs logiques triplés fonctionnent alors en parallèle. Un module de vote est ajouté à la sortie de chaque flip-flop. Les trois entrées de chacun de ces modules sont connectées aux trois sorties des flip-flops en parallèle le précédant. Sa sortie est reliée à

l'entrée de la partie logique suivante. Aucun des éléments n'est durci individuellement aux aléas. Le durcissement provient de la triplication et du vote. Ainsi, si une bascule passe dans un état erroné sous l'effet d'un événement singulier, le module de vote qui la suit bloquera la propagation de l'erreur et délivrera toujours un niveau logique correct sur sa sortie, ce niveau logique lui est dicté par ses deux autres entrées toujours valides. Le circuit continue à fonctionner sans générer d'erreur. De façon similaire, un transitoire sur un bloc logique ou un voteur qui se propagerait jusqu'à la D flip-flop suivante peut entraîner un aléa dynamique à son niveau. Mais, on est ramené à l'exemple précédant. L'erreur stockée dans une D flip-fop sera en outre corrigée au plus tard après une durée maximale correspondant à une période de l'horloge. L'étage précédant du circuit n'ayant pas été affecté. Ainsi, les erreurs ne se propagent pas et sont rapidement corrigées après un court temps de latence (une période d'horloge). Il est possible de réduire le nombre de modules de vote utilisé. Mais, le temps de latence pendant lequel une erreur (localisée et sans conséquences sur le fonctionnement correct) est présente dans le circuit avant sa correction est augmenté. Ce temps de latence correspond à une vulnérabilité du circuit vis à vis d'un nouvel aléa. Cette approche ne garantit la correction que d'une seule erreur simultanément dans une même portion du circuit. Si une seconde erreur est générée dans une D flip-flop parallèle à la première, les modules de vote sont alors mis en défaut. Ils reproduisent l'erreur sur leurs sorties, celle-ci étant désormais majoritaire sur leurs entrées. L'erreur n'est plus corrigée. Le choix du compromis entre le nombre d'étages de vote utilisés et le temps de latence acceptable dépend de l'agressivité attendue du milieu d'utilisation du circuit. Nous nous placerons pour notre étude dans le cas où le temps de latence doit être minimal (un voteur derrière chaque D flip-flop).
Malgré ce choix, l'approche par emploi d'une redondance triple n'est pas aussi efficacement durcie que l'architecture HZ. En effet, l'arbre d'horloge utilisé n'est pas durci aux transitoires de tension.

Nous allons mettre cette faiblesse en évidence en présentant la simulation électrique d'un transitoire de tension sur l'horloge d'un circuit simple. Le circuit choisi est un registre à décalage possédant quatre étages de D flip-flops et trois étages de modules de votes. Son schéma est donné figure 21. Les signaux d'entrée (E) et d'horloge (H) sont connectés en parallèle. Les trois sorties (n31, n32 et n33) sont reliées à des capacités de dix femto farad.

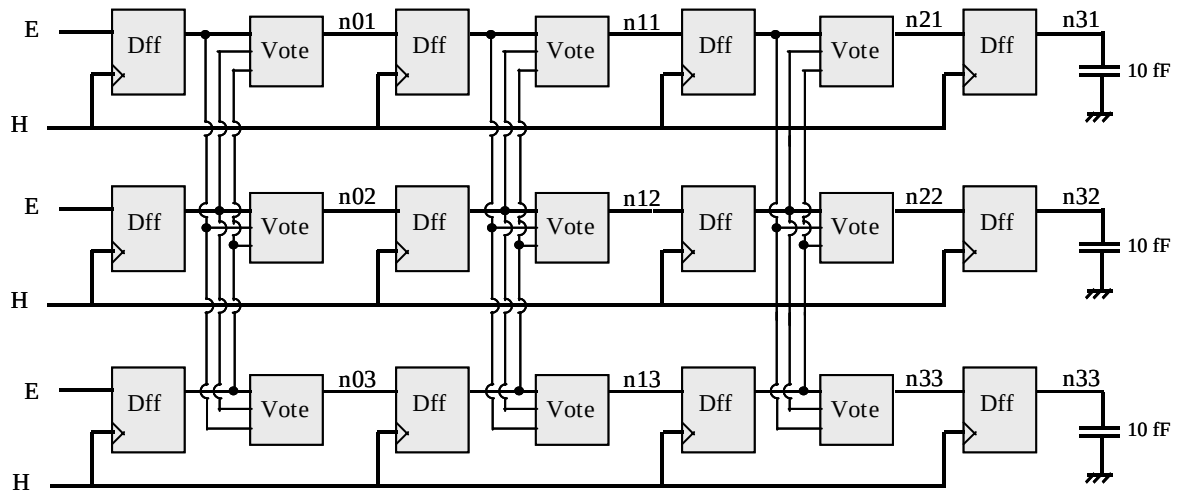


Figure 21 : Registre à décalage et TMR

Pour la simulation électrique[2], la fréquence d'horloge a été fixée à 400 MHz et l'entrée prend alternativement les valeurs 0 et 1. Les résultats de deux simulations sont regroupées figure 22 pour l'horloge, l'entrée et les nœuds n01, n11, n21 et n31 du registre (les tensions correspondants aux nœuds nx2 et nx3 sont identiques à celles du nœud nx1, elles ne sont pas représentées).

Une première simulation à été effectuée sans transitoire de tension d'aucune sorte, elle montre le fonctionnement normal du circuit.

Lors de la deuxième simulation, un transitoire de tension a été généré en amont de H dans l'arbre d'horloge par injection d'une impulsion de courant de 2 mA d'amplitude, 50 ps de temps de montée et de descente et 300 ps de temps de plateau. Il en résulte, au niveau du nœud H, un transitoire de tension vers le niveau haut, intercalé entre deux créneaux. Le transitoire est indiqué par une flèche sur le schéma de la fig. 22.

[2] Une technlogie CMOS 0,25 µm commerciale a été utilisée après extraction des capacités parasites (DKhcmos7 de ST Microelectronics)

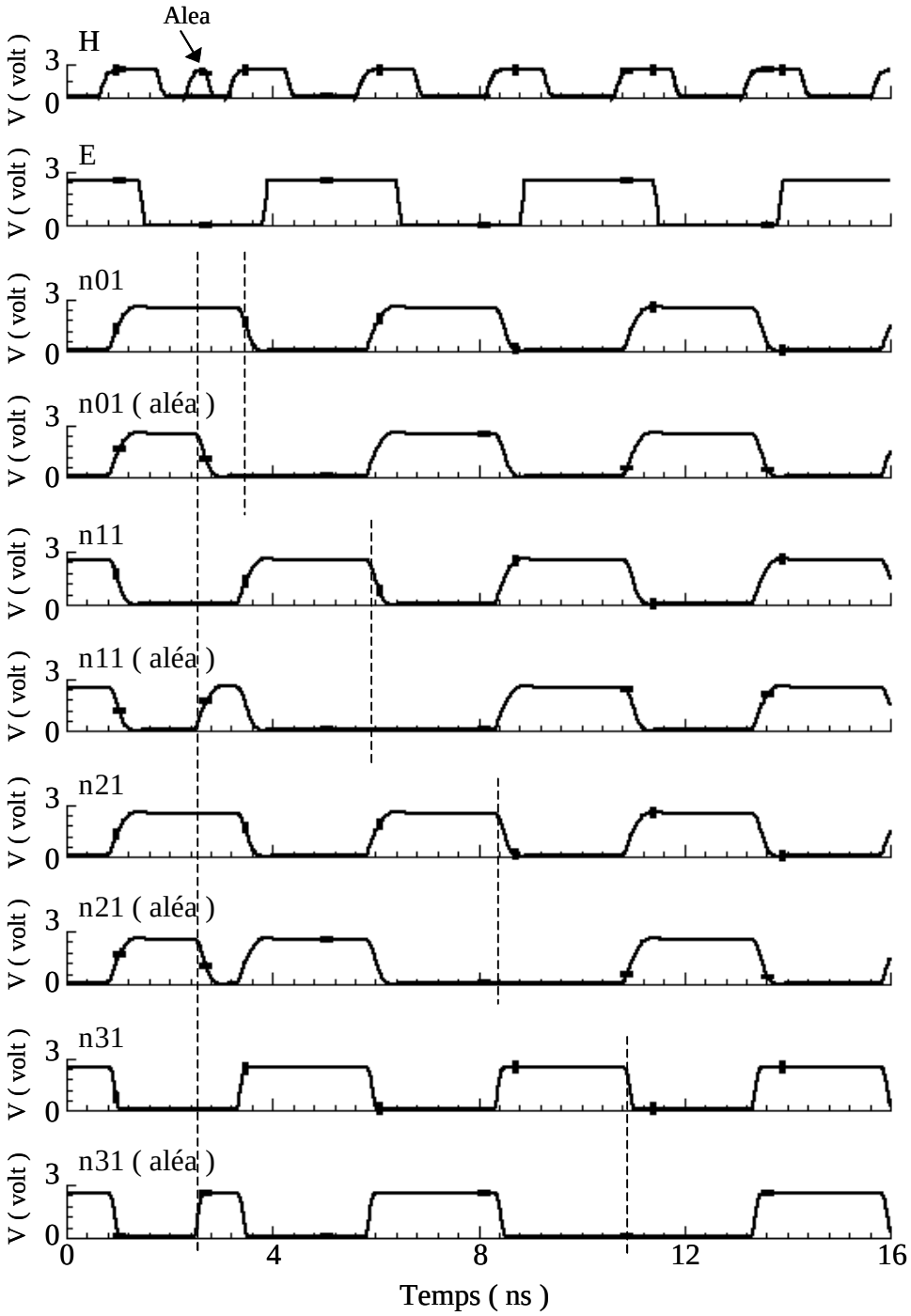


Figure 22 : Effet d'un transitoire sur l'horloge d'un registre à décalage durci par redondance

La perturbation de H entraîne un décalage intempestif des bits dans le registre à décalage. Les traits en pointillé mettent en évidence le temps nécessaire au différents nœuds pour retrouver un niveau logique correct. Cette durée est d'autant plus élevée que le nœud considéré est proche de la fin du registre à décalage.

Ces simulations mettent en évidence la faiblesse du durcissement par redondance triple avec vote majoritaire et la nécessité d'utilisation d'un arbre d'horloge durci.
Un registre à décalage implanté selon les principes de l'architecture HZ et utilisant un arbre d'horloge durci est immunisé à cet effet. Pour confirmation, nous avons également simulé les effets d'un transitoire de tension identique sur le nœud H de l'horloge du registre à décalage présenté figure 23.

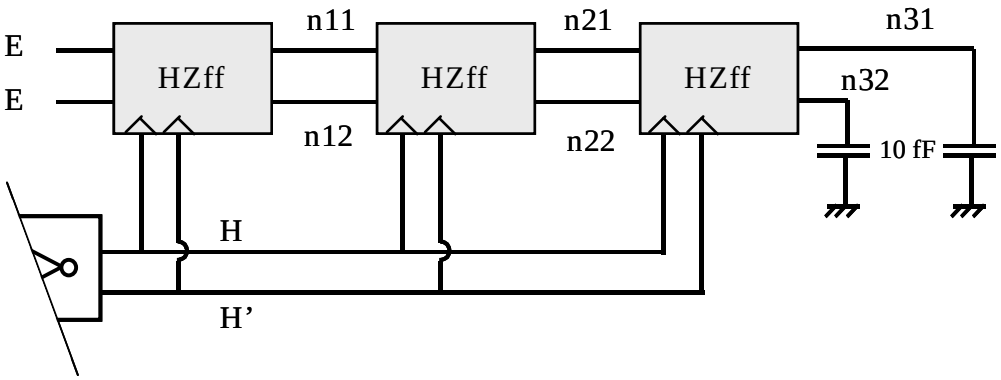


Figure 23 : Application de l'architecture HZ à un registre à décalage

Les résultats de simulation sont présentés figure 24. Le seul signal d'horloge perturbé par le transitoire est le nœud H (pic de tension vers le niveau haut). Le niveau logique de Hb n'est pas altéré. Le nœud n21 passe transitoirement au niveau bas pendant la durée de l'aléa. Cette perturbation est dûe au passage du transistor d'accès Mt1 (commandé par H) de l'esclave de la deuxième flip-flop HZ de l'état bloqué à l'état passant. Les autres T d'accès ne sont pas perturbés. Le nœud n22 conserve bien son niveau haut, ainsi, l'erreur n'est pas susceptible d'affecter la dernière flip-flop HZ . Et, n21 retrouve un niveau logique haut dès la fin du transitoire. L'erreur ne se propage pas et elle ne dure que pendant la durée du transitoire. Le registre à décalage continue à fonctionner correctement.
L'architecture HZ assure l'immunité du circuit aux transitoires de tension affectant l'arbre d'horloge. Ce n'est pas le cas pour le durcissement par redondance.

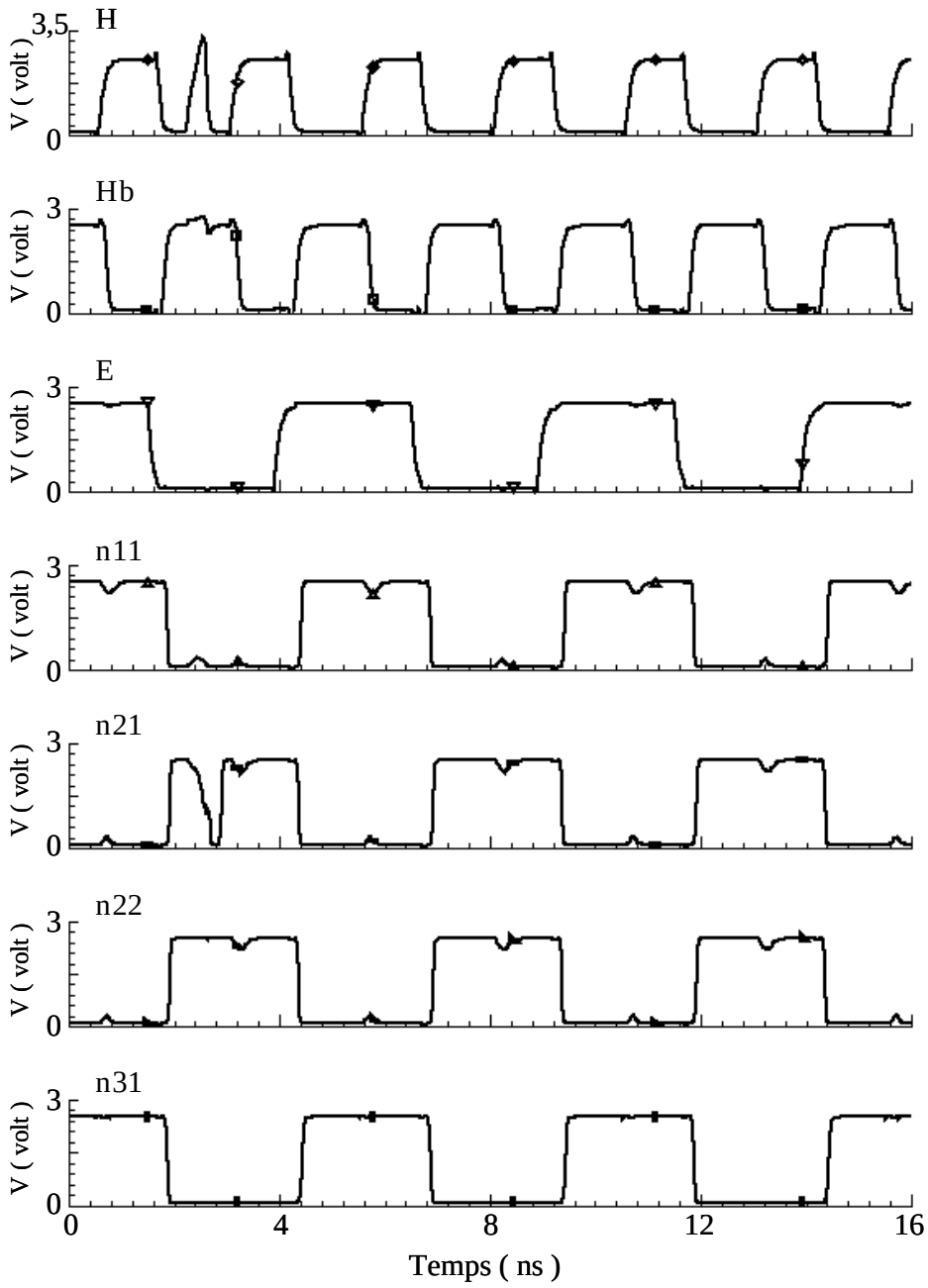


Figure 24 : Effet d'un transitoire de tension sur le nœud H du registre HZ à décalage

La table 9 permet d'établir un comparatif entre les différentes approches en termes de délai et de surface. Le délai considéré pour les flip-flops est celui des bascules D les constituant, pour l'architecture durcie par redondance (TMR), c'est la somme de ceux d'une bascule D et d'un module de vote. Pour le calcul des surfaces des flip-flops, l'architecture retenue est celle des figures 16 et 17. La surface retenue considérée pour l'approche TMR est celle des trois D flip-flops et des trois voteurs nécessaires à son implantation. Le surcoût en surface lié à la

duplication ou à la triplication des blocs logiques combinatoires est mis en évidence par la colonne logique ; il n'a pas d'incidence sur les délais.

	Délai (ps)	Surface (μm^2)	Logique	Durci
D flip-flop	250	171	1	non
Architecture TMR	480	843	3x	imparfait
flip-flop HZ	500	369	2x	oui

Table 9 : Comparaison architecture HZ - TMR

Le durcissement par redondance triple et vote majoritaire a un coût important en termes de délai et de surface. Les contraintes de temps sont augmentées du fait de l'ajout des modules de vote, le délai est presque doublé. Cette approche impose également un surcoût en surface très important (+ 392 %) au niveau des flip-flops et des modules de vote, mais également, un triplement des blocs de logique combinatoire. Et si cette approche garanti un durcissement efficace aux aléas logiques statiques et dynamiques, elle n'est cependant pas adaptée pour durcir un circuit vis à vis des transitoires de tension sur son arbre d'horloge.
L'architecture HZ permet, elle, d'obtenir un durcissement à ces différents types d'aléas (statiques, dynamiques, transitoires). Son intérêt est renforcé par le fait que ce durcissement de bien meilleure qualité est atteint pour un coût bien moindre en surface. L'augmentation de la taille des flip-flops est limitée à 115% et les blocs logiques sont seulement doublés. En terme de délai, l'écart avec la TMR est limité à 4 % .
Ces éléments plaident pour l'utilisation de l'architecture HZ en remplacement du durcissement par redondance et vote majoritaire.

B.1.5. Limitation en fréquence

Lors de la finalisation de l'architecture HZ, il est apparu une possibilité de limitation de la fréquence d'horloge liée aux aléas les plus longs. Nous allons maintenant la mettre en évidence à travers deux exemples de simulation d'une chaîne d'inverseurs HZ et de l'écriture d'une Latch-HZ.
La figure 25 présente la chaîne de trois inverseurs HZ que nous avons simulée ainsi que la localisation du nœud (Q11) choisi pour introduire l'aléa considéré. On fixe en entrée (E) une tension en créneaux de rapport cyclique unité et de fréquence 400 MHz.

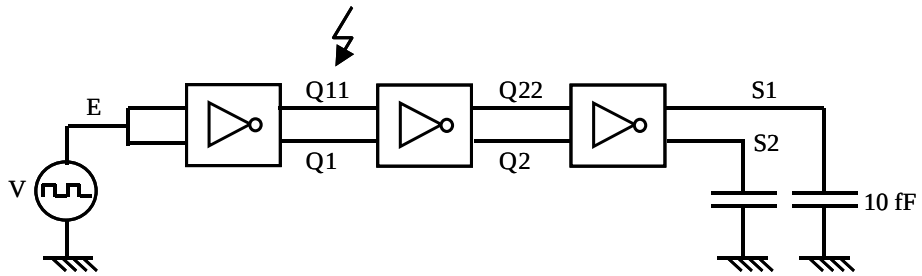


Figure 25 : Localisation de l'aléa sur la chaîne d'inverseurs HZ.

Le courant d'aléa injecté au nœud Q11 crée un transitoire de tension vers le niveau bas. Il n'est cependant susceptible d'exister que si le nœud est sensible à ce type de transitoire, c'est à dire s'il est au niveau logique haut. Nous avons vu dans un exemple similaire au A.1 que si le transitoire se produit après un temps correspondant au temps de transition de l'inverseur HZ après le front montant, il n'a aucune incidence sur l'intégrité du signal transmis en aval.
Cependant, si le transitoire affecte le nœud Q11 juste après son front montant et avant que l'inverseur HZ suivant ait transmis cette évolution sur ses sorties, il peut apparaître un retard. En effet, lorsque les deux entrées d'un inverseur HZ (ici Q1 et Q11) ont des niveaux logiques différents, ses sorties (ici Q2 et Q22) passent en haute impédance et conservent leur niveau logique inchangé jusqu'à la fin de l'aléa. C'est ce phénomène que mettent en évidence les résultats de simulation présentés dans la partie gauche de la figure 26 pour une durée d'aléa $t_{aléa}$ = 400 ps.

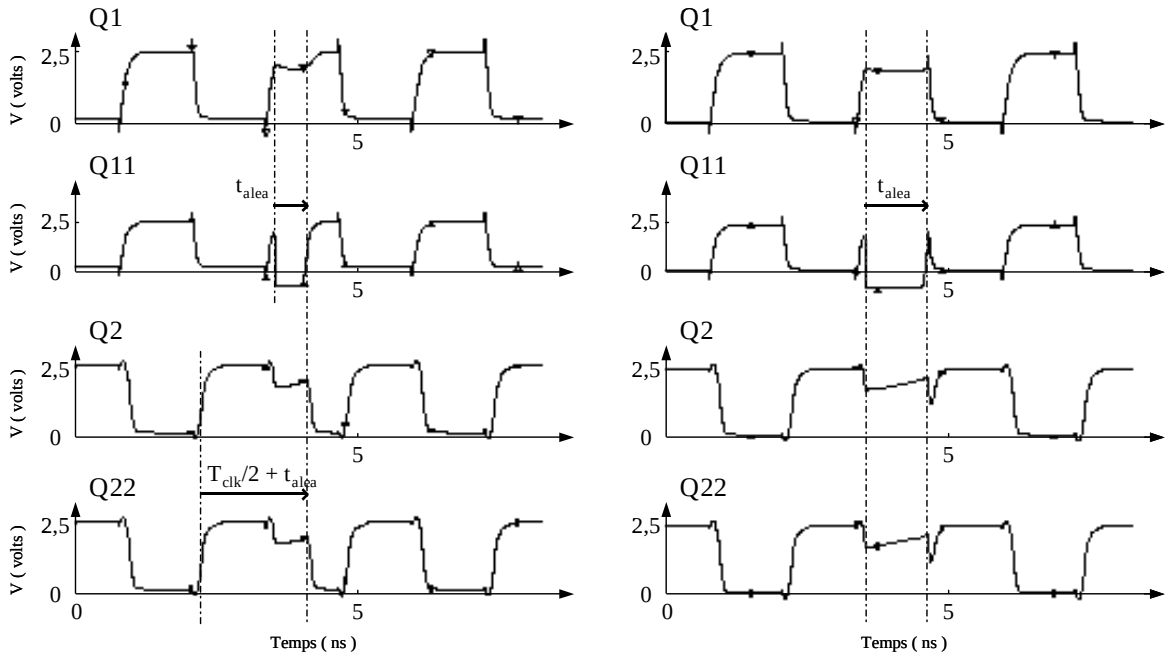


Figure 26 : Retard et disparition de fronts liés à un aléa.

On constate effectivement qu'un aléa en Q11 commençant juste après un front montant retarde de sa durée $t_{aléa}$ les fronts descendants de Q2 et Q22. La durée du créneau haut de Q2 et Q22 précédant l'aléa est portée à $T_{clk} / 2 + t_{aléa}$. Le temps entre le front descendant post aléa et le front montant suivant est réduit à $T_{clk} / 2 - t_{aléa}$. Cette réduction, si elle intervient dans un arbre d'horloge commandant des flip-flops, est susceptible d'entraîner un non respect des temps de propagation et de mémorisation et donc de créer des erreurs.
Dans un pire cas, pour une durée d'aléa élevée, on peut supposer que ce phénomène puisse conduire à la disparition pure et simple de la propagation d'un front montant et d'un front descendant. C'est ce que montre les simulations de la partie droite de la figure 27 pour une durée d'aléa $t_{aléa}$ = 1 ns. Si cela se produit dans une branche d'un arbre d'horloge, une partie du circuit en aval du nœud subissant l'aléa restera figée tandis que le reste du circuit évoluera normalement, créant ainsi de nombreuses erreurs.
Pour éviter ces conséquences néfastes et respecter les contraintes temporelles, on est alors contraint de limiter la fréquence de façon à ce qu'un éventuel retard sur les fronts d'horloge soit sans effet.

Similairement, ce phénomène est susceptible de se produire lors de l'écriture d'une Latch-HZ. Nous avons utilisé pour les simulations présentées figure 27 une Latch-HZ (cf. fig. 10) initialement à l'état 1 (Q1=Q2=1 avec Q2=Q3=0) et dans laquelle on veut mémoriser l'état 0. La fréquence d'horloge a été fixée à 500 MHz. Pour ce faire, les deux entrées de la Latch-HZ (D1 et D2) passent au niveau bas peu avant que les signaux d'horloge (H, H', Hb et Hb') n'autorisent l'écriture et également pendant toute sa durée. La partie gauche de la figure 27 présente la superposition des simulations électriques réalisées en l'absence d'aléa et en présence d'un aléa de 500 ps vers le niveau haut peu avant l'écriture au nœud d'entrée D2 (les courbes correspondantes sont en gras et marquées "aléa" pour lever toute ambiguïté).
Pour une durée de courant d'aléa $t_{aléa}$ = 500 ps, on constate au niveau des nœuds Q3 et Q4 un retard à l'écriture du même ordre. Ce retard est dû au premier inverseur HZ de la Latch-HZ dont les sorties ne peuvent évoluer que lorsque ses deux entrées ont le même niveau logique. Dans ce cas, le transitoire de tension sur D2 est transmis à Q2 qui reste au niveau haut pendant toute la durée de l'aléa alors que D1 est passé au niveau bas dès le début de l'écriture. Q3 et Q4 ne passent finalement au niveau haut que lorsque l'aléa a pris fin et que le rétablissement de D2 au niveau bas a été transmis à Q2. Ce retard est susceptible de perturber la propagation de l'information dans la partie aval du circuit à cause d'une éventuelle violation des contraintes liées aux temps de propagation.

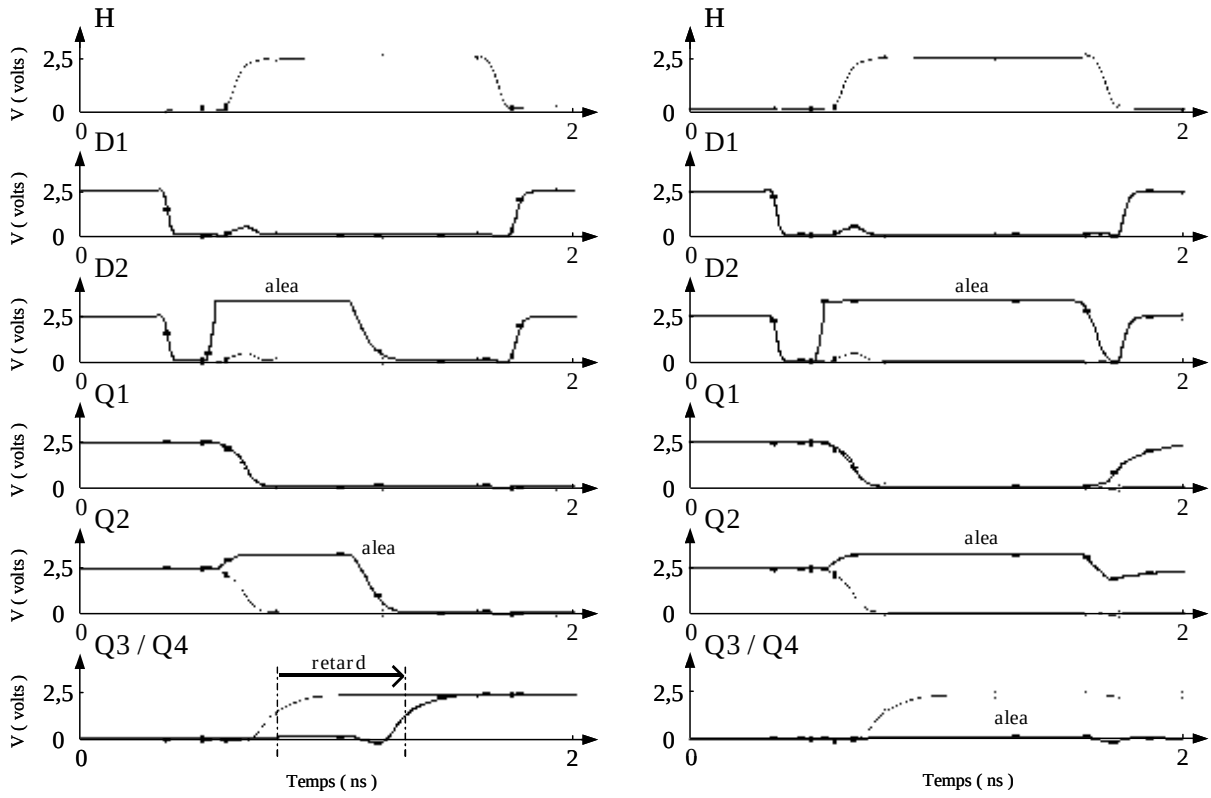


Figure 27 : Retard et non écriture d'une Latch-HZ liés à un aléa.

Dans un pire cas, pour une durée de courant d'aléa élevée, l'écriture peut être empêchée. C'est ce que montrent les résultats de simulation électrique présentés dans la partie droite de la figure 27. Les chronogrammes des tensions des principaux nœuds y sont représentés en l'absence d'aléa et en présence d'un aléa de durée $t_{aléa}$ = 1000 ps (en gras). Dans ce dernier cas on constate que l'écriture de la Latch-HZ n'est pas effectuée, elle reste finalement à l'état 1 (Q1=Q2=1 avec Q3=Q4=0).

Malgré tout, l'existence d'un transitoire de tension d'une durée de l'ordre de la nano-seconde semble improbable. C'est une possibilité qu'il importe de vérifier expérimentalement (par exemple en utilisant le circuit test présenté partie IV).

Cependant, c'est un phénomène qui risque de se manifester à des fréquences plus élevées pour des aléas plus courts , ce qui imposera probablement une limitation de la fréquence.

Il est à noter que cette limitation concerne une grande partie des bascules durcies par restructuration présentées précédemment [BES93], [CAL96], [SAL99], [MON98].

B.1.6. Conclusion

Dans cette partie nous avons introduit une nouvelle approche du durcissement des circuits intégrés par restructuration au niveau transistor. La mise au point d'un inverseur durci, l'inverseur HZ , nous a permis de réaliser un arbre d'horloge durci et une bascule D durcie, la Latch-HZ . Leur utilisation conjointe assure l'immunité aux aléas logiques statiques et dynamiques et aux transitoires de tension sur l'horloge. Ces éléments sont utilisés dans la nouvelle architecture durcie des circuits séquentiels que nous avons proposé : l'architecture HZ. Cette approche globale assure un durcissement complet au prix d'une augmentation des contraintes en délai et en surface. Il apparaît également une possible limitation de la fréquence d'horloge de cette architecture qui reste à vérifier expérimentalement. Mais, elle apporte néanmoins, un gain en surface par rapport à l'approche du durcissement par redondance triple avec vote majoritaire et aussi un durcissement plus complet prenant en compte les transitoires de tension sur l'arbre d'horloge.

B.2 – Durcissement par détection et correction d'erreurs de la couche de configuration.

La deuxième approche de durcissement de la couche de configuration des FPGA que nous avons développée est basée sur l'utilisation d'un code bidirectionnel ([HAR00], [ELI55]) de détection et de correction des erreurs. Son objectif est de permettre la correction des aléas logiques des CSRAM sans avoir recours à un élément de mémorisation extérieur au FPGA et sans interrompre la tâche effectuée par le circuit.

Dans un premier temps, nous présentons le principe de détection – correction des erreurs utilisé et les hypothèses de travail que nous avons retenues. Sa mise en œuvre implique des modifications de l'architecture de la couche de configuration que nous détaillons dans un second temps. Puis, nous vérifions la validité de notre méthode en passant en revue les erreurs radiatives susceptibles de frapper la couche de configuration, avant de conclure.

B.2.1. Principe de la détection – correction d'erreur par test de la parité.

Lorsque l'on considère un mot de n bits, la façon la plus simple de détecter une faute (ou plus exactement un nombre impair de fautes) dans ce mot est de tester sa parité. Un bit de parité correspondant à la somme logique de ces n bits est ajouté au mot. Le test de sa parité consiste à calculer la parité des n bits et à en comparer le résultat avec le bit de parité. Si ils sont différents c'est qu'au moins un des n bits du mot, ou bien le bit de parité lui-même, a été inversé. Mais, c'est seulement l'existence d'une éventuelle erreur qui est détectée, l'utilisation d'un simple bit de parité ne permet pas de la localiser.

Or , nous avons vu dans la partie 2.1 que l'on pouvait ranger les CSRAM de la couche de configuration d'un FPGA sous la forme d'un tableau bidimensionnel de n colonnes et de m lignes [MIC97].

Dès lors, si l'on ajoute à chacune de ces colonnes un bit de parité contenu dans une CSRAM supplémentaire, le test de la parité d'une colonne permet de localiser la présence, ou non, d'une erreur en son sein. Et si chaque ligne du tableau de CSRAM est également dotée d'un bit de parité, il est alors possible en testant leur parité d'y détecter une éventuelle erreur. Ces deux informations, les localisations de la colonne et de la ligne fautives, une fois croisées permettent de repérer précisément la CSRAM dont le bit de configuration a été altéré. La correction d'une erreur, après qu'elle ait été détectée et localisée, est très simple à effectuer avec des mémoires binaires : il suffit d'inverser le bit fautif.

C'est ce principe du test de la parité des colonnes et des lignes que nous avons choisi de développer pour assurer la détection et la correction des erreurs de la couche de configuration. Cependant, cette technique est mise en défaut si, par exemple, deux fautes apparaissent simultanément dans la même colonne ou la même ligne, car sa parité n'est alors pas modifiée ce qui empêche la localisation précise des fautes, et donc leur correction. Le test bidirectionnel de la parité n'est efficace que dans le cadre de l'hypothèse de travail suivante : à savoir pour des fautes uniques suffisamment espacées dans le temps pour que la détection et la correction d'une faute soient réalisées avant l'apparition d'une autre faute.
L'implantation de cette méthode de durcissement impose une évolution de l'architecture de la couche de configuration du circuit. La figure 28 présente cette évolution par rapport à la figure 10 du 2.1.

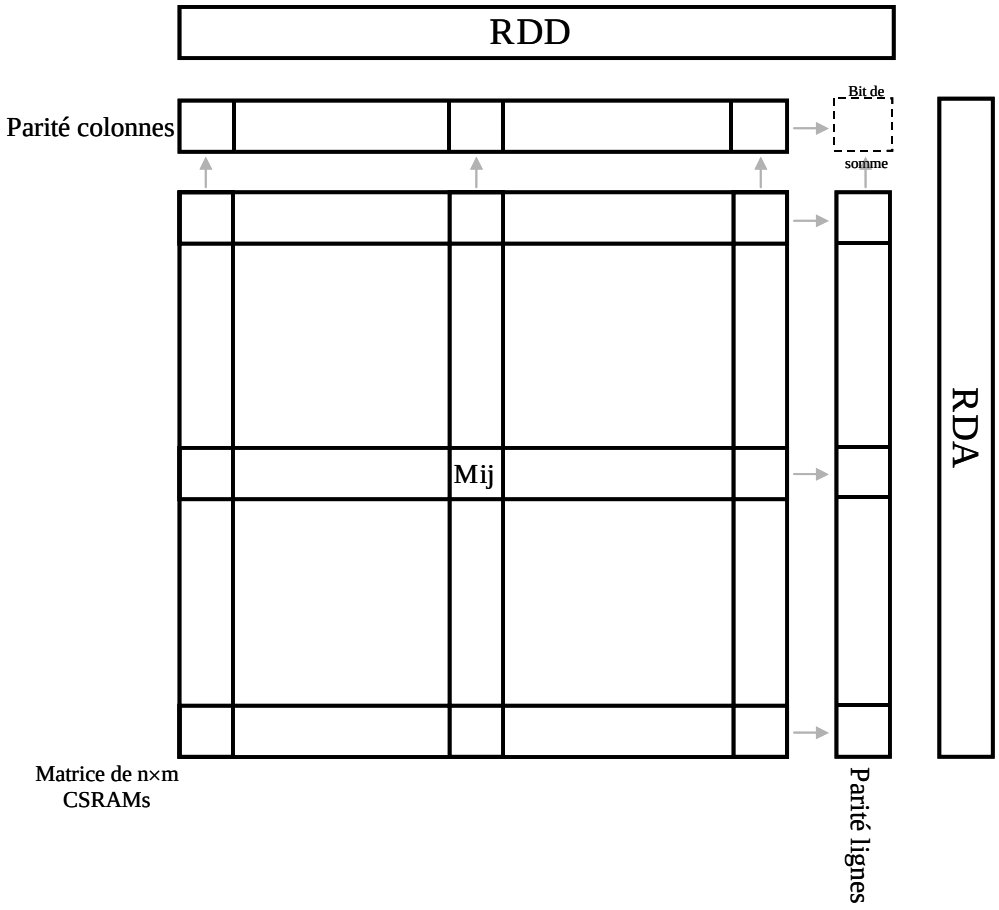


Figure 28 : Schéma de principe de l'application d'un code de parité bidirectionnel à la détection des erreurs de la couche de configuration d'un FPGA.

On retrouve une matrice de n×m CSRAM (n colonnes et m lignes) qui contient la configuration de la partie opérative et les Registres à Décalage de Données (RDD) et

d'Adresses (RDA) servant aux opérations de lecture et d'écriture de la couche de configuration et qui désormais vont également être exploités lors du test des parités. Une ligne de n CSRAM, appelée " parité colonnes ", est ajoutée. Son rôle est de stocker les bits de parité de chacune des colonnes correspondantes (flèches grises). Une colonne de m CSRAM, appelée " parité lignes ", est également ajoutée. Son rôle est de stocker les bits de parité de chacune des lignes correspondantes. Et enfin une dernière CSRAM, nommée " bit de somme ", est ajoutée. Elle contient le bit de parité de l'ensemble des CSRAM de la matrice (leur somme logique). Il correspond également à la somme logique (la parité) des bits contenus dans parité colonnes et dans parité lignes, ainsi, il peut être utilisé indifféremment pour le test de la parité de l'une ou l'autre.

Le test de la parité d'une ligne consiste à charger ses n bits de configuration et son bit de parité dans le RDD et à calculer la somme logique de ces n+1 bits. Si le résultat de cette somme logique est 0 cela signifie que la parité de la ligne s'accorde bien avec son bit de parité (0+0=0 ou 1+1=0) et que la ligne ne contient pas d'erreur. Si le résultat est 1, c'est qu'il y a désaccord entre la parité de la ligne et son bit de parité (0+1=1 ou 1+0=1), et c'est donc qu'une erreur a provoqué l'inversion d'un des n bits de la ligne ou bien l'inversion de son bit de parité (celui-ci est également stocké dans une CSRAM, il est donc également susceptible de subir un aléa logique). Il est ainsi possible de localiser une éventuelle ligne fautive.

De façon similaire, le test de la parité d'une colonne s'effectue en chargeant ses m bits ainsi que le bit de parité correspondant dans le RDA et en calculant la somme logique de ces m+1 bits. Un résultat égal à 0 permet de conclure à l'absence d'une faute et un résultat égal à 1 à la présence d'une faute dans la colonne (bit de parité y compris).

La présence du bit de somme est nécessaire pour effectuer le test de la parité des bits de parité des colonnes et des lignes.

Ainsi, par test successif de la parité des lignes et des colonnes, il est possible de localiser avec précision une faute unique, que celle-ci se trouve dans la matrice de n×m CSRAM, dans parité colonnes, dans parité lignes ou au niveau du bit de somme.

Afin de limiter au maximum l'ampleur des modifications et des ajouts dans la couche de configuration, nous avons choisi d'organiser les différentes étapes de la détection et de la correction des fautes de la façon suivante :

1. test successif de la parité des colonnes :
   - si une faute est détectée, l'emplacement de la colonne fautive est noté et on passe au test de la parité des lignes,
   - si aucune faute n'est détectée, on reprend au début le test de la parité des colonnes.
2. test successif de la parité des lignes :
   - si une faute est détectée, on passe à la phase de correction,
   - si aucune faute n'est détectée après le test des lignes, on revient au test des colonnes.
3. correction de la faute :
   - la ligne fautive est réécrite en prenant soin d'inverser le bit erroné correspondant à la colonne fautive,
   - puis on revient au test des colonnes.

La mise en place de cette technique nécessite des modifications de l'architecture de la couche de configuration, elles sont détaillées dans la partie suivante.

B.2.2. Modifications de l'architecture de la couche de configuration.

Nous avons fait le choix d'utiliser au maximum les éléments de la couche de configuration préalablement dédiés à son écriture et à sa lecture (RDD, RDA, logique de configuration, etc.) pour mettre en place les tests de parité et la correction des fautes, ceci afin de limiter la surface additionnelle.

Les RDD et RDA sont utilisées pour le test de la parité des colonnes et des lignes ce qui nécessite l'ajout de fonctionnalités. Pour les seules opérations d'écriture et de lecture de la configuration, le RDD devait être à même de charger les bits d'une ligne (avant l'écriture ou après la lecture) et le RDA devait assurer l'adressage des lignes. Désormais le RDD doit également permettre l'adressages des colonnes et le RDA le chargement des bits d'une colonne. Il est donc nécessaire d'ajouter des lignes d'adressage des colonnes commandées par le RDD et des lignes de données reliées au RDA pour gérer toutes ces opérations. Et également, un transistor d'accès supplémentaire (Mt' dans la figure suivante) au niveau de chaque CSRAM afin d'éviter des conflits entre les lignes d'adressage et de données. La figure 29 permet de mettre en évidence les modifications apportées à l'architecture du tableau de points mémoire par rapport à la première version présentée figure 9 de la partie 2.1.

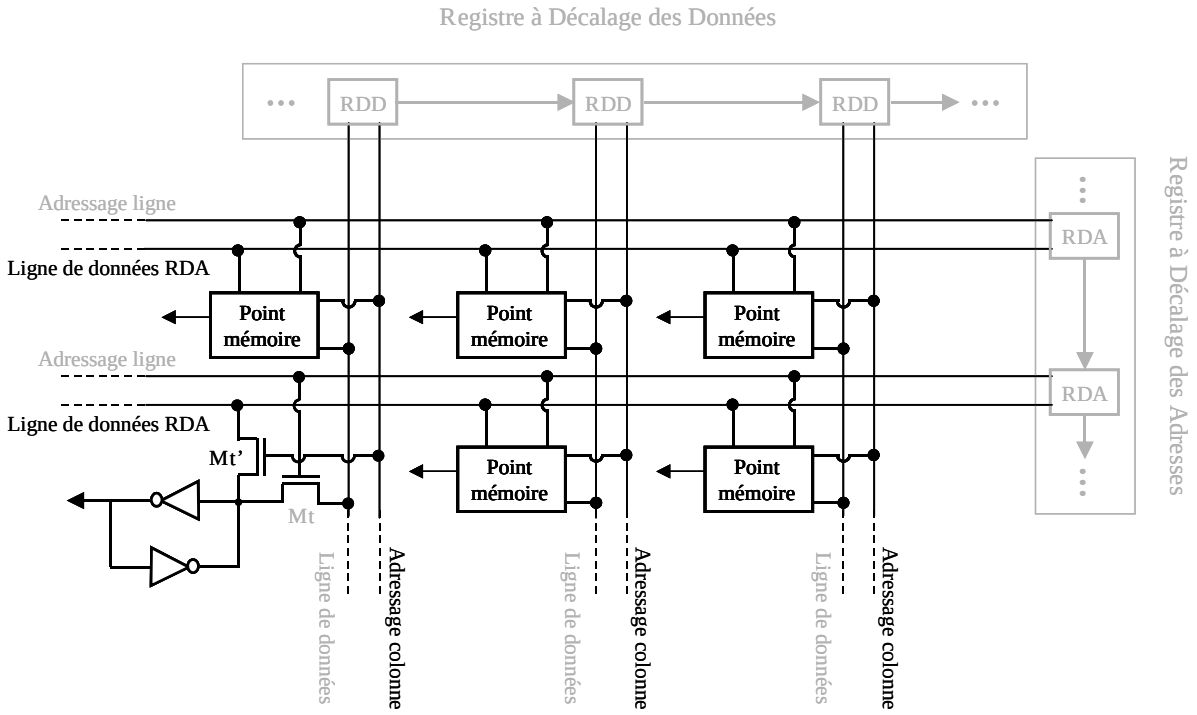


Figure 29 : Modifications de l'architecture du tableau de points mémoire.

Une logique de calcul de la parité à base de portes ou-exclusif est également ajoutée aux RDD et RDA, ainsi qu'une logique permettant l'inversion du bit fautif avant la réécriture de la ligne correspondante.

La gestion de tous ces éléments est plus complexe à réaliser que précédemment ce qui impose l'utilisation d'une logique de configuration et de détection – correction des erreurs plus évoluée.

Il est à noter que le flot de configuration du FPGA est lui aussi modifié pour inclure les bits de parité des colonnes et des lignes et le bit de somme.

Toutes les modifications que nous venons de présenter introduisent de nouvelles sensibilités aux aléas logiques. La partie suivante est consacrée en partie à leur étude.

B.2.3. Gestion et validité du contrôle de la parité et de la correction.

La gestion des tests successifs de la parité des colonnes, des lignes et des corrections est effectuée par une machine d'états principale commandant l'activation de machines d'états secondaires chargées des opérations élémentaires (adressage, écriture, etc.). Dans une première approche, la machine d'états principale peut être limitée à quatre états codés sur

deux bits :

- l'état init " 00 ", il correspond à l'activation de la machine d'états principale, aucune action ne lui est associée,
- l'état ctrl_col " 01 ", il correspond à l'activation de la machine d'états secondaire chargée du contrôle de la parité des colonnes,
- l'état ctrl_lign " 10 ", il correspond à l'activation de la machine d'états secondaire chargée du contrôle de la parité des lignes,
- l'état correction " 11 ", il correspond à l'activation de la machine d'états secondaire chargée de la correction d'une éventuelle faute dans le tableau de CSRAM.

La figure 30 donne le schéma bulle explicitant le fonctionnement de la machine d'états principale.

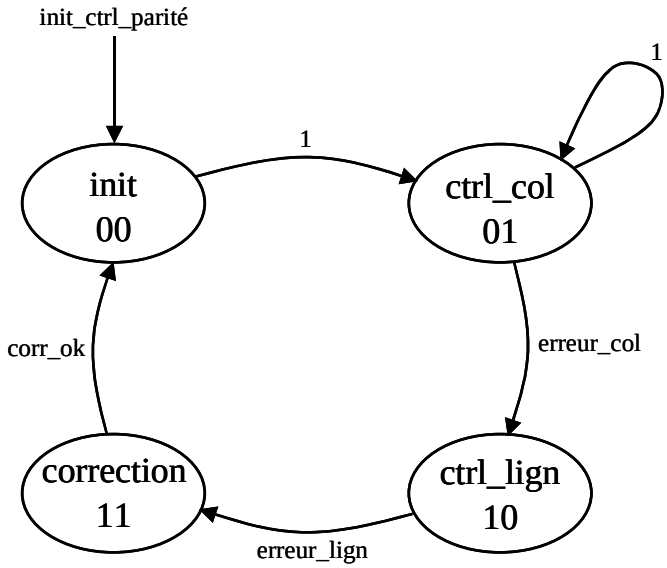


Figure 30 : Schéma bulle, en première approche, de la machine d'états principale commandant le contrôle de la parité.

Le signal init_ctrl_parité commande l'activation de la machine d'états principale (MeP) (passage dans l'état init). Elle permet également de la désactiver à tout instant.
La phase d'initialisation correspond à une temporisation d'une période d'horloge, le passage de la MeP dans l'état ctrl_col est automatique.
L'état ctrl_col correspond à l'activation de la MeS qui gère le contrôle successif de la parité des colonnes. Cet état est rebouclé sur lui-même, le test des colonnes ne peut être interrompu que dans deux cas de figure. Le premier cas correspond à la désactivation de la MeP via le signal init_ctrl_parité. Le deuxième cas correspond à la détection d'une faute de parité sur une

colonne, la MeS correspondante commande alors le passage de la MeP dans l'état ctrl_lign via le signal erreur_col (après avoir également commandé la mémorisation de la colonne fautive).
L'état ctrl_lign correspond à l'activation de la MeS qui gère le contrôle successif de la parité des lignes. Une fois la ligne fautive localisée (celle-ci est alors stockée dans le RDD), elle commande le passage de la MeP dans l'état correction via le signal erreur_lign.
L'état correction correspond à l'activation de la MeS qui gère la correction de la faute détectée. Cette correction est effectuée en réécrivant la ligne fautive avec les valeurs contenues dans le RDD tout en inversant le bit correspondant à la colonne fautive. Une fois la correction effectuée, la MeS de correction commande le retour de la MeP dans l'état init via le signal corr_ok. Le processus de détection – correction des fautes reprend alors son cours.

La MeP que nos venons de décrire permet de détecter et de corriger efficacement tout aléa logique unique localisé dans le tableau de CSRAM (y compris au niveau des différents bits de parité). Cependant, d'autres évènements radiatifs peuvent apparaître à différents emplacements de la couche de configuration (RDD, RDA, MeP, etc.) et compromettre sa validité. Nous allons maintenant étudier leur impact et faire évoluer la MeP en conséquence. Mais cette étude doit être effectuée en gardant à l'esprit l'hypothèse de travail que nous avons posé au paragraphe B.1 ; celle de fautes uniques espacées dans le temps. Aussi, nous ne considérerons que les évènements singuliers frappant la couche de configuration à l'exclusion du tableau de CSRAM et en supposant que celui-ci ne contient aucune faute. Ainsi, ces évènements ne peuvent survenir que lorsque la MeP est dans les états init et ctrl_col. Ce choix est motivé par le fait que notre méthode de test des parités n'est efficace que pour des fautes uniques dans les bits de configuration. La prise en compte d'une deuxième faute localisée dans le MeP, par exemple, n'est pas utile. En effet, elle aurait statistiquement plus de probabilité d'apparaître au niveau des CSRAM (du fait de leur grand nombre) et par la même d'empêcher toute correction. Il est inutile de durcir la logique de gestion du test des parités au delà de sa propre capacité à corriger les fautes de la matrice de points mémoire.

La MeP est elle même sensible aux évènements singuliers radiatifs. Ils sont susceptibles de modifier l'état logique des bascules utilisées pour coder ces quatre états. Le pire cas de figure correspond à un passage de l'état ctrl_col codé par les bits 01 à l'état correction codé par les bits 11 qui déclencherait la réécriture intempestive d'une ligne par les valeurs stockées dans le

RDD et donc l'apparition d'une multitude de fautes incorrigibles dans la configuration. La solution consiste à coder les états de la MeP sur trois bits en faisant correspondre l'état init au code 000, l'état ctrl_col au code 001, l'état ctrl_lign au code 010 et l'état correction au code 111. L'état correction ne peut alors plus être atteint à partir des états init et ctrl_col par le fait d'un aléa logique. Il reste cependant quatre états interdits (100, 110, 101 et 0111) qui ne peuvent être atteint en fonctionnement normal. Pour éviter que la MeP ne puisse passer dans l'un de ces états et y reste bloquée du fait d'un aléa inversant un de ses trois bits de codage des états, ils conduisent automatiquement à un retour de la MeP dans l'état init après une période d'horloge.

Les fautes survenant lorsque la MeP est dans l'état init sont sans conséquences car les différents registres et compteurs utilisés lors des états suivants sont réinitialisés avant utilisation.

Il reste à étudier les fautes survenant lorsque la MeP est dans l'état ctrl_col. Celles localisées dans le RDD et le RDA sont susceptibles d'avoir des conséquences mettant en défaut le durcissement. Lors du test de la parité des colonnes, le RDD est utilisé pour l'adressage successif des colonnes lors de leur lecture par le RDA. Toutes les flip-flops du RDD sont alors à 0, à l'exception de la flip-flop correspondant à la colonne sélectionnée qui est à 1. Si un aléa logique provoque le passage à 1 d'une deuxième flip-flop, lors de la validation de la lecture, les CSRAM de deux colonnes seront connectées simultanément aux même lignes de données reliées au RDA, avec comme conséquence de nombreux conflits logiques pouvant entraîner l'inversion de plusieurs bits de configuration. Pour éviter ce phénomène on peut tirer parti de la parité des bits contenus par le RDD qui est impaire en fonctionnement normal lors de l'adressage. Si elle est paire, c'est qu'un aléa logique a corrompu l'adressage des colonnes et qu'il ne faut pas en valider la lecture. Cet avertissement est délivré au MeP via le signal erreur_RDD, la solution consiste à le faire retourner dans l'état init. Le test normal de la parité des colonnes reprend ensuite son cours. Le RDA est également sensible aux évènements radiatifs. Lors du test de la parité des colonnes, un aléa logique dans l'une de ses flip-flops ou bien un transitoire de tension sont susceptibles de perturber le contrôle de la parité et d'être interprétés comme une faute de parité ce qui entraîne le passage de la MeP dans l'état ctrl_lign. Mais dans ce cas, aucune ligne fautive ne peut être détectée et la MeP décrite précédemment reste bloquée dans l'état ctrl_lign. Pour remédier à ce défaut, il suffit d'interpréter la non détection d'une faute lors du contrôle des lignes comme résultant d'une erreur localisée au niveau RDA et de commander le passage de la MeP dans l'état init via le signal erreur_RDA.

Toutes les modifications que nous venons de décrire conduisent au nouveau schéma bulle de la machine d'états principale présenté figure 31.

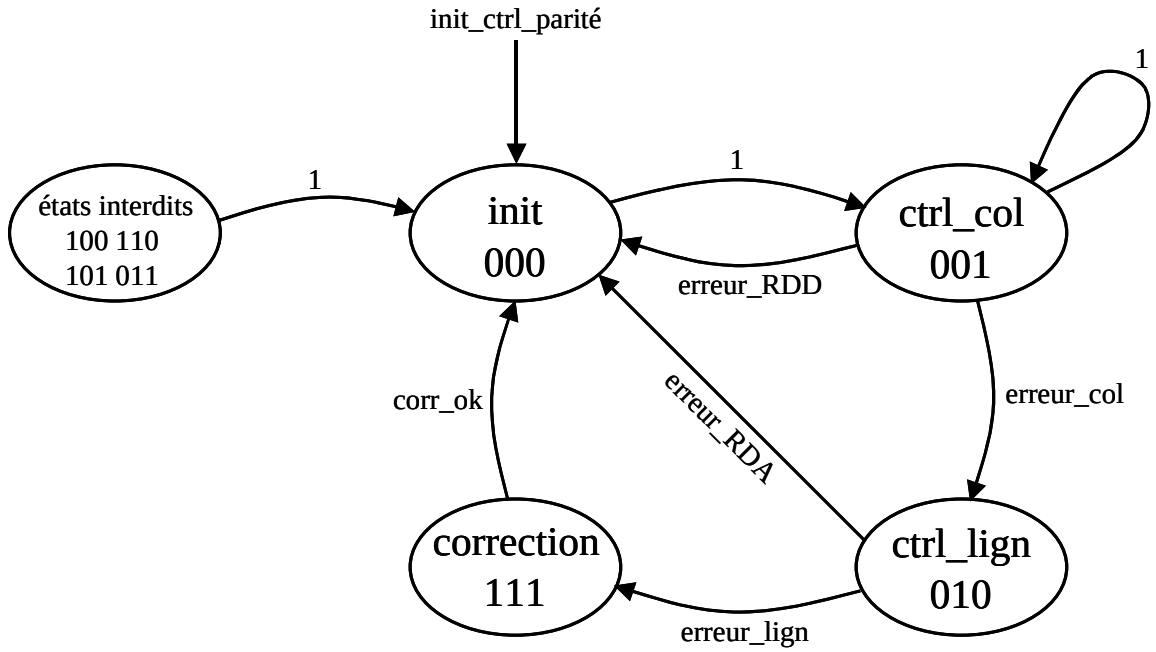


Figure 31 : Schéma bulle final de la machine d'états principale commandant le contrôle de parité.

D'autres sensibilités aux transitoires de tension existent au niveau des différents signaux d'adressage de la couche de configuration, elles ont été éliminées en les faisant dépendre de plusieurs commandes. Aucune erreur simple n'est susceptible d'avoir de conséquences.

Le modèle de couche de configuration que nous proposons a été décrit dans le langage AHDL de la société Altera (une version propriétaire du langage VHDL) et son prototypage a été effectué et validé au moyen d'un FPGA et du logiciel MAX+PLUS II.

B.2.4. Conclusion.

La méthodologie de détection et de correction des fautes de la couche de configuration que nous proposons assure son durcissement aux aléas logiques simples. Elle est opérationnelle de façon continue et permet au FPGA de fonctionner sans interruption malgré l'apparition de fautes parmi ses bits de configuration et ceci sans nécessiter le recours à une mémoire externe au circuit.

Son implantation requiert une surface additionnelle relativement limitée. Les principaux facteurs du surcoût en surface proviennent : du transistor d'accès additionnel ajouté à chaque

CSRAM, des points mémoires utilisés pour stocker les bits de parité, des modifications apportées aux RDD et RDA, et des machines d'états gérant son fonctionnement. Si on considère un tableau de n lignes et n colonnes, le nombre de bit de parité nécessaires est 2n+1. Ce qui correspond, en prenant n=30 pour un tableau d'un millier de CSRAM, à une surface additionnelle de 6,7 % ($2n+1/n^2$). L'augmentation de la surface des RDD et RDA liée au test des parités est de l'ordre de dix pour-cent. Nous avons finalement évalué l'augmentation de la surface de la couche de configuration à une dizaine de pour-cents.

La mise en place de cette méthode de durcissement se traduit également par une augmentation de la puissance consommée. Elle correspond à celle d'un processus de lecture ou d'écriture classique, elle est donc similaire à celle des approches de readback ou de scrubbing proposées par Xilinx. Il est possible de la limiter en réduisant la périodicité des contrôles de la parité, mais l'efficacité du durcissement est alors réduite en conséquence (en espaçant les contrôles, on augmente la probabilité de l'occurrence d'une deuxième faute avant que la première ne soit corrigée).

Cette approche est bien adaptée aux FPGA car les CSRAM sont réparties parmi les éléments de la couche opérative et il est donc peu probable qu'une seule particule radiative puisse induire des fautes multiples dans la couche de configuration. Cette probabilité peut être encore réduite en utilisant des CSRAM durcies ou en répartissant les points mémoire dans plusieurs tableaux entrecroisés de façon à ce que des CSRAM adjacentes n'appartiennent pas au même tableau.

B.3 – Conclusion sur les nouvelles approches du durcissement des FPGA.

Nous avons présenté dans la partie III.B les nouvelles approches de durcissement que nous avons développées afin de durcir les FPGA aux aléas logiques et aux transitoires de tension.

La première piste que nous avons suivie repose sur la mise au point d'inverseurs, de points mémoires et de bascules durcies par restructuration au niveau transistor. Elle nous a permis de mettre au point une architecture durcie originale pour les circuits séquentiels : l'architecture HZ. Elle peut être utilisée pour durcir la couche opérative des FPGA (l'utilisation par la société Actel de D flip-flops durcies spécifiques pour les applications spatiales dans sa famille RT54SX-S de FPGA [ACT01] valide la pertinence de ce type d'approches). Les simulations électriques post layout que nous avons réalisées ont mis en évidence la très bonne qualité du durcissement obtenu vis à vis des aléas logiques statiques, mais également vis à vis des aléas logiques dynamiques et des transitoires de tension. Ces dernières qualités présentent une amélioration notable par rapport à un grand nombre d'approches par restructuration préexistantes qui ne les prenaient pas en compte (à l'exception notable de durcissement par redondance temporelle et des approches proposées par la société Xilinx). L'architecture HZ autorise en outre une économie de surface notable par rapport au durcissement par redondance triple. Il est apparu une possibilité de limitation de la fréquence d'horloge liée aux aléas les plus longs (de l'ordre de la nano-seconde). Ce phénomène reste cependant à observer expérimentalement.
Le point mémoire durci par restructuration que nous avons mis au point est bien adapté au durcissement de la couche de configuration des FPGA. Les simulations électriques ont montré que son état ne pouvait être corrompu même pour des aléas d'une durée peu envisageable (5000 ps).
L'efficacité des principes de durcissement par restructuration que nous avons développés reste cependant à valider expérimentalement. Le circuit que nous avons réalisé dans ce but est présenté dans la partie IV.

La deuxième piste que nous avons explorée repose sur l'utilisation d'un code détecteur et correcteur d'erreur. Elle s'applique au durcissement de la couche de configuration des FPGA.

Elle est basée sur le contrôle de la parité des lignes et des colonnes de points mémoire de configuration d'un FPGA. Elle permet la détection et la correction de toute faute unique parmi les CSRAM, ceci sans interrompre le fonctionnement normal du circuit. La correction est en outre assurée sans avoir recours à une mémoire extérieure au FPGA (elle même éventuellement sensible aux aléas logiques). La principale réserve provient de sa limitation au cas des erreurs uniques. Des erreurs multiples peuvent apparaître lorsque une même particule radiative traverse des zones sensibles appartenant à plusieurs points mémoire (dans le cas de trajectoires rasantes), ou lorsque plusieurs particules radiatives traversent le circuit au même instant. Cependant, dans un FPGA, il est possible d'éloigner les CSRAM les unes des autres, car elles sont reparties parmi la logique de la couche opérative, ce qui limite la probabilité pour une même particule de corrompre plusieurs points mémoires. Les CSRAM peuvent être réparties en plusieurs tableaux pour limiter les possibilités d'erreurs multiples dans le même tableau. Et il est également possible d'utiliser des points mémoire durcis pour obtenir un durcissement optimal.

Les deux approches que nous avons développées permettent de proposer des solutions efficaces pour le durcissement des FPGA comme les simulations réalisées l'ont montré. Leur validation expérimentale permettra de confirmer leur degré de pertinence.

IV

Validation expérimentale

IV. Validation expérimentale.

L'objectif de cette partie est de présenter le circuit test que nous avons réalisé et envoyé en fonderie afin de valider expérimentalement le principe de durcissement par restructuration que nous avons présenté au III.B.1.

Les résultats que nous avons présenté sont issus de simulations électriques post – layout réalisées après extraction des capacités parasites de circuits réalisés dans la technologie CMOS 0,25 µm DKhcmos7 de la société ST Microelectronics. C'est cette technologie que nous avons retenu pour fabriquer le circuit test.
Notre principal objectif est de valider expérimentalement les principes de durcissement que nous avons choisi. Ils reposent comme nous l'avons vu :

- sur une redondance double de l'information,
- et sur le passage en haute impédance des sorties d'un inverseur HZ dont les entrées ont des niveaux logiques opposés.

Ainsi, l'utilisation d'inverseurs HZ nous a permis de réaliser des points mémoire durcis (CSRAM-HZ) et des bascules durcis (Latch-HZ). Nous avons choisi, pour un premier circuit, de vérifier la pertinence du durcissement obtenu pour les CSRAM-HZ.
Les choix architecturaux que nous avons fait concernant le circuit test ont été guidés par les deux types de validation expérimentale que nous avons retenus : un test radiatif sous ions lourds et un test laser.

Le premier test expérimental que nous souhaitons mettre en œuvre est un bombardement du circuit sous ions lourds réalisé dans un accélérateur de particules. Il permettra de tester la susceptibilité des CSRAM-HZ aux aléas logiques créés par le passage des ions lourds à travers leurs zones sensibles. Ce type de test permet de caractériser le durcissement d'un circuit grâce au tracé de sa section efficace en fonction de l'énergie cédée au substrat par la particule incidente. La grandeur la plus couramment utilisée pour quantifier ce transfert d'énergie est le LET (Linear Energy Transfert), c'est l'énergie transférée moyenne par unité de grandeur de trace d'un ion incident, exprimée le plus souvent en $MeV.cm^2/mg$ [BEA95]. La section efficace σ(LET) donne le nombre d'évènements par unité de fluence en fonction du LET des ions. Cette courbe permet de déterminer le LET seuil (c'est à dire le LET

minimum créant une erreur) et la section efficace de saturation (σ_{sat}) qui est la surface totale sensible du circuit.

Les phénomènes radiatifs étant par nature non déterministes, la caractérisation de la CSRAM-HZ nécessite le test simultané d'un grand nombre de points mémoire durcis afin de limiter la durée d'irradiation. Pour ce faire, nous avons choisi de donner au circuit de test la forme d'une couche de configuration de FPGA selon le modèle présenté au II.A.3. Afin de mettre en évidence le durcissement des CSRAM-HZ, nous avons également décidé d'incorporer au circuit un grand nombre de CSRAM classiques à cinq transistors (cf. II.C.1). Enfin, nous avons dessiné deux layouts différents pour la CSRAM-HZ. Le premier correspond au point mémoire durci décrit au III.B.1.2. La figure 1 met en évidence la localisation de ses zones sensibles. Comme nous l'avons vu précédemment, elles dépendent de l'état mémorisé par la CSRAM-HZ. Les diffusions sensibles quand la CSRAM-HZ est à l'état 0 sont marquées d'un « 0 » et celles qui le sont à l'état 1 sont marquées d'un « 1 ».

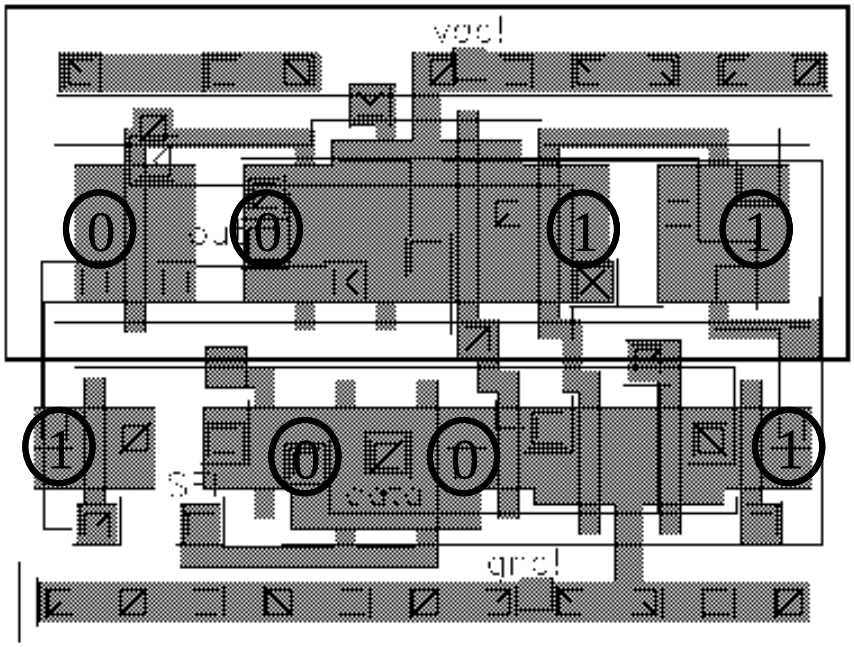


Figure 1 : Layout de la CSRAM-HZ et localisation des diffusions sensibles.

On constate que plusieurs zones sensibles dans le même état sont relativement proches : 1 µm d'écart pour les deux diffusions sensibles à l'état 0 de la partie inférieure du layout et 1,475 µm pour les deux zones sensibles à l'état 1 de sa partie supérieure.

Dès lors, il existerait une possibilité pour que les charges générées dans le silicium par le passage d'une particule radiative soient collectées simultanément par deux zones sensibles,

mettant alors en défaut le durcissement de la CSRAM-HZ. Afin de vérifier l'existence de ce phénomène, et le cas échéant de le quantifier, nous avons dessiné un deuxième layout de CSRAM-HZ dont l'écartement des diffusions sensibles entre elles a été augmenté. Ce layout "aéré" est présenté figure 2.

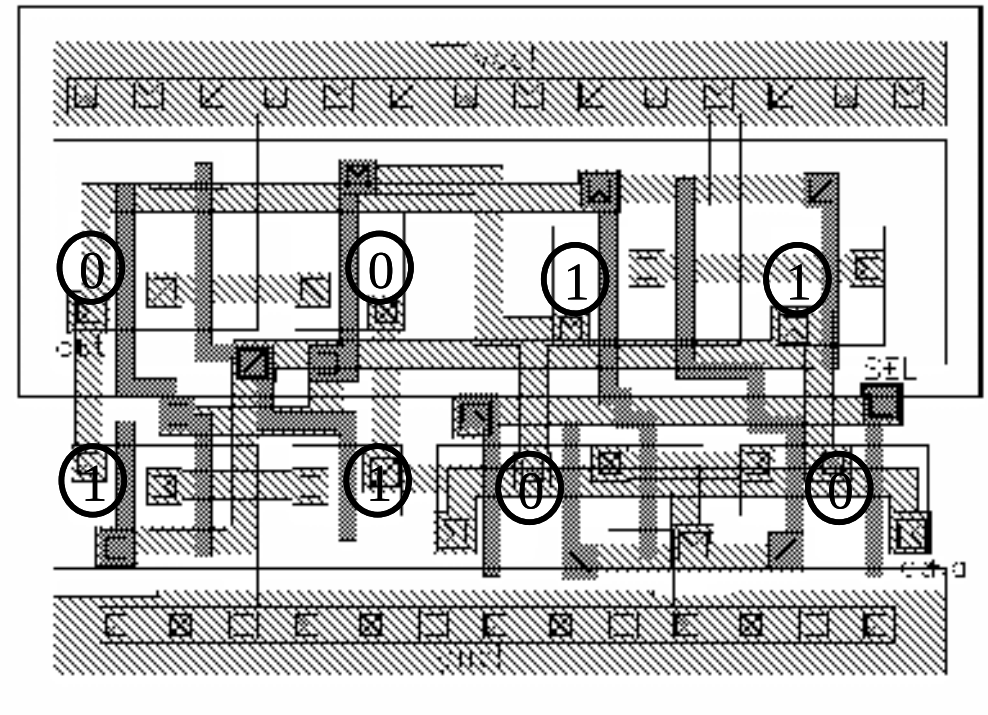


Figure 2 : Layout aéré de la CSRAM-HZ et localisation des diffusions sensibles.

La distance séparant les deux diffusions sensibles à l'état 0 les plus proches a été portée à 2,15 µm et celle séparant les deux zones sensibles à l'état 1 les plus proches à 2,55 µm. Cet éloignement des zones sensibles se fait au prix d'une augmentation de la surface occupée par le layout qui passe de 68,6 μm^2 à 114 μm^2.

Le plan mémoire que nous avons réalisé comporte un tableau de neuf cent points mémoire ayant trente lignes et trente colonnes. Les dix premières lignes sont constituées de CSRAM classiques à cinq transistors, les dix suivantes de CSRAM-HZ dans leur forme compacte et les dix dernières de CSRAM-HZ dans leur forme aérée.
N'ayant pas à notre disposition de bibliothèque de cellules standards dans la technologie choisie, l'ensemble de la logique de configuration (Registres à décalage des données et des adresses, machines d'états, etc.) a été réalisé en "full custom".

Chaque circuit test comporte deux couches de configuration indépendantes afin de palier d'éventuels effets radiatifs destructifs. La figure 3 donne le layout du circuit test.

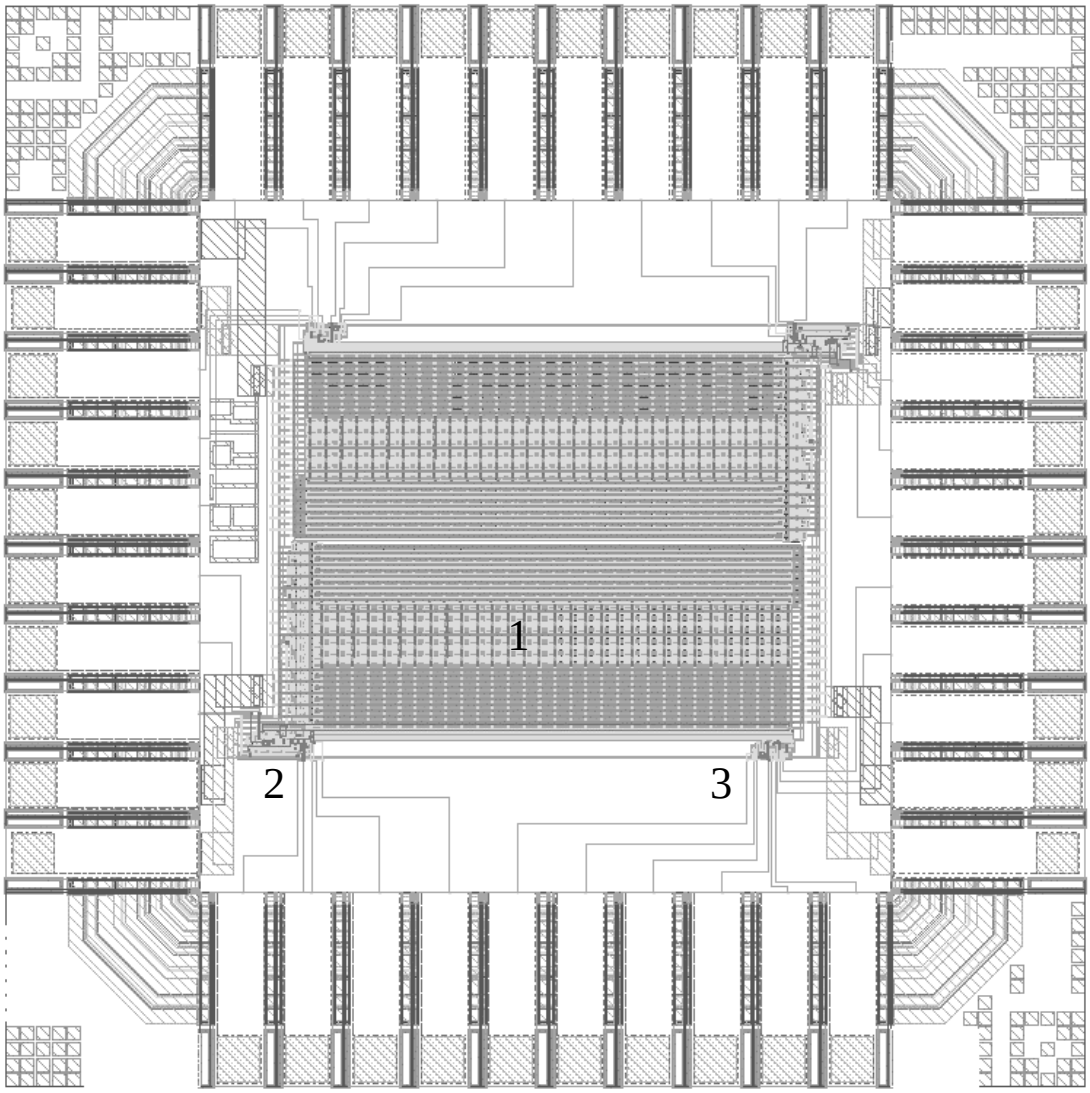


Figure 3 : Layout du circuit test.

Le tableau de points mémoires est localisé par le chiffre 1, le chiffre 2 met en évidence la partie logique commandant les opérations d'écriture et de lecture du plan mémoire et le chiffre 3 indique un bloc logique dédié au test laser que nous allons maintenant évoquer.

La mise en place du test d'un circuit sous irradiation d'ions lourds est coûteuse et les accélérateurs de particules sont souvent difficiles d'accès. En outre, ils ne permettent pas d'obtenir d'informations spatiales concernant les éventuelles parties d'un circuit à l'origine

d'erreurs. Nous avons donc prévu d'incorporer au circuit de test une partie dédiée à un deuxième type de test expérimental : le test laser.

Le passage d'un faisceau laser à travers le silicium provoque, du fait de l'interaction des photons avec le semi-conducteur, la génération de paires électrons-trous. Lorsque ces charges sont générées à proximité de zones sensibles, il peut y avoir apparition d'événements singuliers similaires à ceux créés par le passage de particules radiatives ionisantes. Ainsi, le test laser peut être utilisé pour la caractérisation de la sensibilité aux évènements singuliers d'un circuit intégré [BUC90], c'est un outil complémentaire du test sous ions lourds [MEL94].

Le test laser est facile à mettre en œuvre (pas de problèmes de radiations) et relativement peu coûteux. Dans le cas présent, l'utilisation d'un laser permet d'obtenir des informations spatiales sur la localisation des zones à l'origine d'une erreur, ce que ne permet pas le test dans un accélérateur de particules. La puissance du faisceau laser incident est réglable, elle permet donc de classer les zones sensibles en fonction de leur seuil de sensibilité.

Le module logique dédié au test laser que nous avons implanté comprend une CSRAM à cinq transistors, une CSRAM-HZ compacte et une CSRAM-HZ aérée.

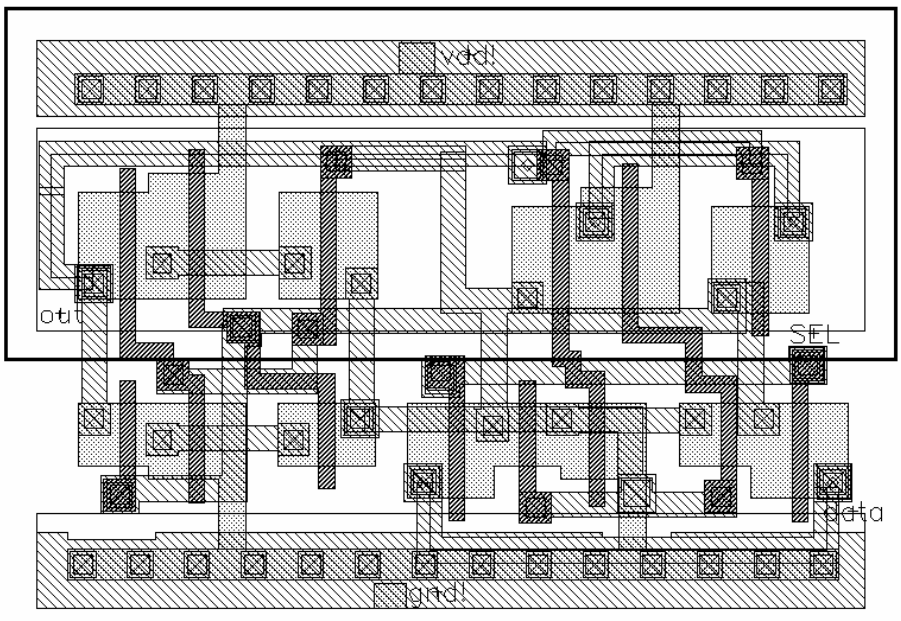


Figure 4 : Layout de la CSRAM-HZ aérée en vue du test laser.

Cependant, le faisceau laser est réfléchi par les pistes métalliques et son spot à un diamètre de l'ordre du micromètre, ce qui risque de fausser les résultats obtenus. Aussi, nous avons

également ajouté trois versions des points mémoire précédents dont le layout a été redessiné de façon à découvrir au maximum les diffusions sensibles en écartant les pistes métalliques. A titre indicatif, la figure 4 donne la nouvelle version de la CSRAM-HZ aérée.
Si on fait correspondre les layouts des figures 2 et 4, on constate effectivement que désormais les diffusions sensibles sont accessibles à un faisceau laser.
Nous attendons du test laser qu'il nous permette de valider rapidement l'efficacité de nos techniques de durcissement, et que dans l'hypothèse ou des défaillances seraient mises en évidence, qu'il nous permette de les localiser précisément et de les quantifier [McM99], [DUT01].

En conclusion, le circuit test que nous avons mis au point doit nous permettre de valider expérimentalement les méthodes de durcissement par restructuration que nous avons développées.

V

Conclusion

V. Conclusion générale.

L'objectif de ce travail de thèse était l'étude et la mise au point de solutions de durcissement aux effets radiatifs singuliers applicables aux circuits reconfigurables à base de SRAM.

Dans un premier temps, nous avons analysé la structure des FPGA à base de SRAM et leur architecture partitionnée en couche de configuration et couche opérative, nous avons envisagé les différentes faiblesses de ces dispositifs suite à la manifestation des phénomènes radiatifs. Après avoir rappelé les solutions actuelles de durcissement, nous avons mis en évidence la grande vulnérabilité des SRAM de configuration et l'importance de leur durcissement et avons proposé deux approches :

La première est une approche de durcissement par restructuration. Elle consiste en de nouvelles architectures d'inverseurs - l'inverseur HZ - et d'éléments de mémorisation durcis en modifiant le nombre et l'agencement de leurs transistors. L'inverseur durci HZ permet en effet de réaliser des arbres d'horloge durcis et une bascule D durcie : la Latch-HZ. Leur utilisation conjointe a donné naissance a une nouvelle architecture durcie aux aléas logiques statiques et dynamiques et à la propagation des transitoires : l'architecture HZ. Cette architecture présente une économie en surface notable si on la compare aux architectures durcies par redondance triple avec vote majoritaire. La méthode est également applicable au durcissement de la couche opérative des FPGA, mais d'une façon plus générale elle est adaptée à la réalisation de circuits séquentiels robustes.
Nous avons également utilisé les inverseurs HZ pour réalisaser des points mémoire durcis : les CSRAM-HZ. Ces cellules permettent de durcir la couche de configuration des FPGA à base de SRAM vis à vis des aléas logiques statiques, levant ainsi la principale limitation à leur utilisation en milieu radiatif. Néanmoins, la CRAM-HZ présente une surface plus importante que la CSRAM classique à cinq transistors, et de ce fait, son emploi est susceptible de s'en trouver limité du fait du très grand nombre de points mémoire présents dans un FPGA. Ceci nous a conduit à développer une méthode de durcissement alternative de la couche de configuration.

La deuxième approche de durcissement est destinée à la seule couche de configuration. Elle est basée sur un code détecteur et correcteur d'erreurs par test de la parité. Elle permet la

correction de toute erreur simple parmi les bits de configuration, sans interruption du fonctionnement du circuit et sans avoir recours à une mémoire externe au FPGA. Le surcoût en surface est limité grâce à la réutilisation de la logique déjà consacrée à la configuration du circuit.

Les résultats obtenus sur le principe d'un durcissement par restructuration sont prometteurs. Ils reposent actuellement sur des simulations électriques pré et post layout. Nous avons donc également développé un circuit test qui nous permettra de valider expérimentalement la robustesse de cette approche par redondance simple et mise en haute impédance. Ce circuit doit faire l'objet de prochaines campagnes de test par faisceau laser et en accélérateur de particules.

Bibliographie

[ACT97] "Design Techniques for Radiation-Hardened FPGAs", Actel Application Note 5192642-0, September 97.

[ACT01] "RT54SX-S Rad Tolerant FPGAs for Space Applications", Actel Data Sheet 5172151-1/3.01 v0.2, Mars 2001.

[AND82] "Single Event Error Immune CMOS RAM", J.L. Andrews et al, Trans. Nuc. Sci., Vol. 29, No. 6, p2040, 1982.

[ANG00] "Les limites technologiques du silicium et tolérance aux fautes", L. Anghel, Thèse de Doctorat, 15 décembre 2000.

[BAZ97] " Attenuation of Single Event Induced Pulses in CMOS Combinational Logic ", M. Baze, S. Buchner, IEEE Trans. on Nuc. Sci., Vol. 44, No. 6, 1997

[BEA95] "Effets de rayonnements sur les composants électroniques : aspects fondamentaux", J. Beaucour, Cours RADECS, Septembre 1995.

[BES93] "Conception de deux points mémoire statiques CMOS durcis contre l'effet des aléas logiques provoqués par l'environnement radiatif spatial", D. Bessot, Thèse de Doctorat, 26 Novembre 1993.

[BIN75] "Satellite Anomalies from Galactic Cosmic Rays", D. Binder, E.C. Smith, A.B. Holman, IEEE Trans. Nucl. Sci., NS 22, p2675, 1975.

[BUC90] "Pulsed Laser-Induced SEU in Integrated Circuits : a Practical Method for Hardness Assurance Testing", S. Buchner, K. Kang, W.J. Stapor, A.B. Campbell, A.R. Knudson, P. McDonald, S. Rivet, IEEE Trans. On Nuc. Sci., Vol. 37, No. 6, 1990.

[BUC97] " Comparison of Error Rates in Combinational and Sequential Logic ", S. Buchner, M. Baze, D. Brown, D. McMorrow, J. Melinger, IEEE Trans. on Nuc. Sci., Vol. 44, No. 6, 1997

[CAL96] " Upset Hardened Memory Design for Submicron CMOS Technology ", T. Calin, N. Nicolaidis, R. Velazco, IEEE Trans. Nuc. Sci., NS-43, n°6, 2875-2878, December 1996

[CAN91] "An SEU Immune Logic Family", J. Canaris, Proceedings of the 3rd NASA Symposium on VLSI Design, 2.3.1-2.3.11, 1991.

[DOD95] "Critical Charge Concepts for CMOS SRAMs", P.E.Dodd, F.W.Sexton, IEEE Trans. On Nuc. Sci., Vol. 42, No. 6, 1995

[DUT01] "Improving the SEU Hard Design using a Pulsed Laser", J.M. Dutertre, F.M. Roche, P. Fouillat, D. Lewis, Radecs, 2001.

[DUT01b] " Robustness of CMOS Circuits Designed for Space and Terrestrial Environment", J.M. Dutertre, F.M. Roche, XVI Conference on Design of Circuits and Integrated Systems, 2001, pp 672-676

[DUT01c] " Integration of Robutness in the Design of a Cell ", J.M. Dutertre, F.M. Roche, G. Cathebras, 11th IFIP International Conference on Very Large Scale Integration, pp 189-193, 2001

[HAR00] "CMOS Memory Circuits", T.P. Haraszti, Kluwer Academic Publishers, 2000.

[HELI55] "Coding for Noisy Channels", P. Helias, IRE Convention Records, Part 4, pp. 37-46, 1955.

[LIU92] "Low Power SEU Immune CMOS Memory Circuits", M.N. Liu, S. Whitaker, IEEE Trans. Nucl. Sci., Vol. 39, No. 6, p1679, 1992.

[MAV98] "Temporally Redundant Latch for Preventing Single Event Disruption in Sequential Integrated Circuits", D.G. Mavis, P.H. Eaton, Technical Report P8111.29, Mission Research Corporation, 1998.

[MAV00] "Development of a Design Methodology for Preventing Single Event Disruptions in Deep Submicron Microcircuits", D.G. Mavis, Draft Final Report, Mission Research Corporation, 2000.

[McM99] "Application of Pulsed Laser for Evaluation and Optimization of SEU-Hard Designs", D. McMorrow, J.S. Melinger, S. Buchner, T. Scott, R.D. Brown, F-26, RADECS99.

[MEL94] "Critical evaluation of the Pulsed Laser Method for Single Event Effects Testing and Fundamental Studies", J.S. Melinger, S. Buchner, D. McMorrow, W.J. Stapor, T.R. Weatherford, A.B. Campbell, IEEE Trans. Nuc. Sci., Vol. 41, No. 6, 1994.

[MIC97] "Testing for the Programming Circuits of LUT-based FPGAs", H. Michinishi, T. Yokohira, T. Okamoto, T. Inoue, H. Fujiwara, IEEE 6th Asian Test Symposium, pp 242-247, November 1997.

[MON98] "Flip-Flop Hardening for Space Applications", T. Monnier, F.M. Roche, G. Cathebras, IEEE International Workshop on Memory Technology, Design, and Testing, p104, 1998.

[MON99] "Durcissement de circuits convertisseurs A/N rapides fonctionnant en environnement spatial", T. Monnier, Thèse de Doctorat, 29 Octobre 1999.

[MON99b] "SEU Testing of a Novel Hardened Register Imlemented Using Standard CMOS Technology", T. Monnier, F.M. Roche, J. Cosculluela, R. Velazco, IEEE Trans. Nuc. Sci. Vol. 46, No. 6, p1440-1444, 1999.

[NIC99] "Circuit Logique Protégé contre des Perturbations Transitoires", Brevet d'invention n° 99/03027 déposé le 9 mars 1999, Inventeur : Michaël Nicolaïdis.

[NOR96] "Single Event Upset at Ground Level", IEEE Trans. Nuc. Sci. Vol. 43, No. 6, p2742-2750, 1996.

[POR99] "Test des Circuits Configurables de Type FPGA à Base de SRAM", J.M. Portal, Thèse de Doctorat, 1999.

[ROC88] "An SEU-Hardened CMOS Data Latch Design", L.R. Rockett, IEEE Trans. Nucl. Sci., Vol. 35, No. 6,p1682, 1988.

[ROC00] "A design based on proven concepts of an SEU-immune CMOS configuration data cell for reprogrammable FPGAs", L.R. Rockett, Microelectronics Journal, No. 32, p99-111, 2001.

[SAL99] " Conception en vue du Durcissement des circuits intégrés numériques aux effets radiatifs ", L. Salager, Thèse de Doctorat, 11 janvier 1999

[VEL94] "Two CMOS Memory Cells Suitable for the Design of SEU Tolerant VLSI Circuits", R. Velazco, D. Bessot, S. Duzellier, R. Ecoffet, S. Koga, IEEE Trans. Nucl. Sci., Vol. 41, No. 6, p2229, 1994.

[VEL96] "SEU-Hardened Storage Cell Validation Using a Pulsed Laser", R. Velazco, T. Calin, M. Nicolaidis, S.C. Moss, S.D. LaLumondière, V.T. Tran, R. Koga, IEEE Trans. Nuc. Sci., Vol. 43, No. 6, 1996.

[WAN99] "SRAM Based Re-programmable FPGA for Space Applications", J.J. Wang, R.B. Katz, J.S. Sun, B.E. Cronquist, J.L. McCollum, T.M. Speers, W.C. Plants, IEEE Trans. Nuc. Sci., Vol. 46, No. 6, December 1999.

[WHI91] "SEU Hardened Memory Cells for a CCSDS Reed Solomon Encoder", S. Whitaker, J. Canaris, K. Liu, IEEE Trans. Nucl. Sci., Vol. 38, No. 6, p1471, 1991.

[XIL00] "Correcting Single-Event Upsets Through Virtex Partial Configuration", Xilinx Application Note, XAPP216, June 2000.

[XIL01] "Triple Module Redundancy Design Techniques for Virtex FPAGAs", Xilinx Application Note, XAPP197, November 2001.

[XIL01b] "QPro XQ18V04 QML In-System Programmable Configuration PROMs", Xilinx Preliminary Product Specification, DS082 (v1.2), November 2001.

[ZIE96] "IBM experiments in soft fails in computer electronics (1978 - 1994)", IBM Journal of Research and Development, Vol. 90, No. 1, p3-18, 1996.

Résumé

Cette thèse est consacrée à l'étude de solutions de durcissement des circuits reconfigurables à base de SRAM aux effets radiatifs singuliers. Un partitionnement symbolique des FPGA en une couche de configuration et une couche opérative a permis de mettre en évidence et de hiérarchiser les erreurs d'origine radiative. C'est l'éventuelle inversion de bits de configuration qui est le principal facteur limitant l'usage des FPGA en milieu radiatif. Après avoir étudié les solutions actuellement retenues, nous présentons deux approches permettant d'assurer leur durcissement.
La première approche est basée sur la restructuration des inverseurs et des éléments de mémorisation au niveau de l'agencement de leurs transistors. Elle permet de durcir efficacement la couche opérative aux effets singuliers. Elle est également adaptée au durcissement de la couche de configuration, mais au prix d'un surcoût en surface important.
La deuxième approche repose sur l'utilisation d'un code détecteur et correcteur d'erreurs par test de la parité. Elle est dédiée au durcissement de la couche de configuration.
Un circuit test est également présenté afin de valider expérimentalement les principes de durcissement par restructuration que nous avons utilisés.

Mots-clés :
Circuits reconfigurables, FPGA, SRAM, Effets Singuliers, Durcissement aux radiations, Restructuration, Conception robuste.

Discipline : Electronique, Optronique et Systèmes

Robust Reconfigurable Circuits

Abstract

This thesis is devoted to the development of Single Event Upset hardness methodologies dedicated to SRAM based FPGA. SEU may alter the FPGA function through induced errors in the configuration memory. This is the major concern about the use of FPGA in radiation environment. Furthermore they affect the user logic in a similar way than classical integrated circuits.
Thanks to restructuration of their transistors arrangement and number, we propose a new inverter and data latch architectures. It allows us to define an SEU proof architecture for user logic hardness. This method is although applicable to harden the configuration memory. However it is area consuming.
So we propose a second methodology dedicated to the configuration memory. It is an Error Correction And Detection algorithm based on parity testing.
Finally we present the test circuit we designed to validate the restructurating approach.

Key words :
Reconfigurable circuit, Robust design, FPGA, SRAM, Single Event Upset, EDAC,Radiation Hardness.

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM)
161, rue Ada, 34392 Montpellier Cedex 05 FRANCE