Performance and Safety/Security Management in automotive and IoT applications

Kalpana Senthamarai Kannan

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE GRENOBLE ALPES

Spécialité : **Nano Electronique et Nano Technologies**

Arrêté ministériel : 25 mai 2016

Présentée par

Kalpana SENTHAMARAI KANNAN

Thèse dirigée par **Lorena ANGHEL,** Professeur des Universités, Grenoble INP
et codirigée par **Michèle PORTOLAN,** Maître de Conférences, Grenoble INP

préparée au sein du **Laboratoire Techniques de l 'Informatique et de la Microélectronique pour l 'Architecture des systèmes intègres**
dans **l'École Doctorale Electronique, Electrotechnique, Automatique, Traitement du signal (EEATS)**

Management des Performances de Sûreté et de Sécurité pour les Applications Automotives et IoT

Performance and Safety/Security Management in automotive and IoT applications

Thèse soutenue publiquement le **19 juillet 2021**,
devant le jury composé de :

Madame LORENA ANGHEL
Professeur des Universités, GRENOBLE INP, Directrice de thèse
Monsieur, ALBERTO BOSIO
Professeur des Universités à l' ECOLE CENTRALE DE LYON, Rapporteur
Monsieur, PAOLO BERNARDI
Maître de Conférences à POLITECNICO DI TORINO, Rapporteur
Monsieur, ARNAUD VIRAZEL
Maître de Conférences à l' UNIVERSITE DE MONTEPELLIER, Examinateur
Monsieur, GIORGIO DI NATALE
DIRECTEUR DE RECHERCHE, CNRS, LABORATORIE TIMA, Président,

Table of Contents

INTRODUCTION 16

THESIS MOTIVATION 17
SUMMARY 19
THESIS OUTLINE 21

CHAPTER 1: AGING MECHANISM OF DIGITAL CMOS FDSOI TECHNOLOGY 23

A.INTRODUCTION 23
A.1 CMOS Circuit Transistor Aging 23
A.2 Inverter Delay model 26
A.2 Bias Temperature Instability (BTI): 27
A.3 Hot Carrier Injection (HCI): 28
A.4 Time-Dependent Dielectric Breakdown (TDDB): 30
B.SOURCES OF VARIABILITY FOR CMOS DEVICES 30
B.1 Process Variation 31
B.2 Voltage Variation 32
B.3 Temperature Variation 33
C.CMOS LOGIC GATES 34
D.DELAY MONITORS 34
D.1 Embedded In-Situ Monitors 35
E. TRADITIONAL PATH SELECTION AND ISM INSERTION FLOW 37
E.1 Critical path selection 37
E.2. Critical Path Monitoring 40
E.3. In-Situ Monitor (ETI) Insertion flow 41
E.4 Example of In-Situ Monitor (ETI) Insertion 44
F.LOW POWER CMOS DESIGN 45
F.1 Dynamic voltage and frequency scaling: 45
F.2 Adaptive Voltage and Frequency Scaling: 46
G.CONCLUSION 47

CHAPTER II : MACHINE LEARNING ALGORITHMS ...48

A.INTRODUCTION 48
B.MACHINE LEARNING TYPES 49
B.1 Supervised Learning Algorithm 49
B.2 Unsupervised Learning Algorithm 50
B.3 Semisupervised Learning Algorithm 51
B.4 Reinforcement Learning Algorithm 52
C. CHOOSING THE SUITABLE ML ALGORITHM 52
D.MACHINE LEARNING IN EMBEDDED APPLICATION 53
E.EVALUATION OF MACHINE LEARNING ALGORITHM 56
E.1 Mean Square Error (MSE) or Root MSE (RMSE) 56
E.2 Mean Absolute Error (MAE) 56

E.3 Mean Absolute Percentage Error (MAPE) *56*
F. Application of Machine Learning Algorithm 57
G. Conclusions 57

CHAPTER III: MACHINE LEARNING MODEL FOR AGING ESTIMATION 59

A. Introduction 59
B. Prediction Framework for Circuit Aging 59
B.1 Aging Prediction for Reference Gates *60*
B.2 Proposed Machine Learning Framework *64*
C. Experimental Validation 65
C.1 Activity Aware Aging for NAND and NOR gates *65*
C. 2 Activity Aware Aging for Generic Gates *68*
C.3 Switching Activity Extraction for Complex Designs *69*
C.4 Test Case 1: FIR Filter *70*
C.5 Test Case 2: AES Circuit *73*
D. Alternative MVL Algorithms: Random Forest 78
D.1 Introduction *78*
D.2 Random Forest Algorithm *78*
D.3 Advantage and disadvantage of using RF algorithm *78*
D.4 Comparison with Linear Regression *79*
F. Conclusion 80

CHAPTER IV: SYSTEM-LEVEL STRATEGIES AND APPLICATIONS 81

A. Introduction 81
B. Workload-dependent NCP selection 82
C. Aging-Aware Adaptation of Operating Performance Points 83
D. Power Minimization for fixed frequency OPP 86
E. Fine-grain Timing Prediction 89
F. Aging Aware OPP Overclocking 90
G. Maximum Performance with OPP Cap 92
H. Conclusion 93

CHAPTER V: CONCLUSION AND PERSPECTIVES 94

Perspectives 95
List of Publicaitons 96

REFERENCES 97

Table of Figures

Figure 1: Source: Ray Kurzweil, "The singularity is near: when humans transcend biology", P.67, The vikiking Press, 2006. Data points between 2000 and 2012 represent BCA estimates. 16
Figure 2: (a) CMOS Inverter, (b) Propagation Delay of Rising and Fall Time 24
Figure 3: Schematic of the Physical mechanism of NBTI 28
Figure 4: HCI Mechanism 29
Figure 5: General taxonomy of variation [37] 31
Figure 6: The Supply Voltage and threshold voltage in time [39] 32
Figure 7: Full chip temperature Increase profile [42] 33
Figure 8: Razor-I flip-flop[28] 36
Figure 9: Block diagram of canary flip flop monitor [9] 36
Figure 10: Register – Register (Reg - Reg) 37
Figure 11: Primary Input – Register (PI - Reg) 38
Figure 12: Register – Primary Input (Reg - PO) 38
Figure 13: Primary Input – Primary Output (PI - PO) 39
Figure 14: Path Slack of FIR Filter 39
Figure 15: Path slack of AES Circuit 40
Figure 16: Example of a Critical Path Monitoring Technique[45] 40
Figure 17: ETI Insertion methodology flow in digital design [44] 42
Figure 18: Pattern dependence of critical path ranking[50] 43
Figure 19: different path activations scenario based on the workload[50] 43
Figure 20: Critical path of FIR Filter 44
Figure 21: Waveform of FIR Filter 44
Figure 22: Waveform of FIR along with ETI monitor 45
Figure 23: Closed-loop for AVFS schematic [54] 47
Figure 24: Learning System Model[56] 48
Figure 25: Machine Learning Types [57] 49
Figure 26: Pictorial representation of Supervised, Unsupervised and Semi-supervised Learning[59] 51
Figure 27: Major Challenges in ML embedded system[69] 55
Figure 28: Novel Delay Aging Prediction Framework flowchart 61
Figure 29: Linear Regression Model 64
Figure 30: Aging Delay prediction from the limited training set for NAND Gate 66
Figure 31: Aging delay prediction from the limited training set for NOR Gate 67
Figure 32: Delay Degradation comparison between our proposed ML algorithm and SPICE simulation for NAND and NOR gates 67
Figure 33: An aging-induced degradation prediction model for several standard cell gates 69
Figure 34: Full Synopsys flow [83] 70
Figure 35: The FIR Filter VCD file waveform 70
Figure 36: Architecture of the FIR filter 71
Figure 37: RTL (Top-level) view of FIR Filter 71
Figure 38: Schematic view of FIR Filter 71
Figure 39: FIR filter: Stacked bar graph for CP1 72
Figure 40: FIR filter: Stacked bar graph for different CP 73

Figure 41: Architecture of the AES Circuit 74
Figure 42: RTL view of the AES circuit 74
Figure 43: AES circuit critical path aging for PVT (ss28_1.0V_125C) 75
Figure 44: AES circuit critical path aging for PVT (ss28_1.0V_125C) 75
Figure 45: Ranking Variation for Workload 1 76
Figure 46: Ranking Variation for Workload 2 77
Figure 47: Evaluation of machine learning algorithms for complex gates 79
Figure 48: Evaluation of machine learning algorithms for Critical path in FIR Filter 79
Figure 49: Evaluation of machine learning algorithm for Critical path in AES circuit 80
Figure 50: NCP subset evolution over time for two workloads 83
Figure 51: Frequency Vs. Flag_count of all process for fresh and aged critical path 85
Figure 52: Shift in the frequency of different voltages for the slow process 85
Figure 53: Shift in the frequency of different voltages for typical process 86
Figure 54: Flow chart for the fixed frequency algorithm 88
Figure 55: Fixed frequency graph for SS_25C 89
Figure 56: Fine-grain prediction graph with a time window in weeks 90
Figure 57: Fine-grain prediction graph with a time window in months 90
Figure 58: Maximum performance optimization strategy algorithm 92
Figure 59: OPP optimization capped at 1V/1.5 GHz 93

Table 1: Parameters for the delay model[75] 27
Table 2: Pearson's Correlation table 68
Table 3: Process Vs. Frequency 84

Acronyms

CMOS Complementary Metal Oxide Semiconductor

IOT Internet Of Things

FDSOI Fully Depleted Silicon on-chip

PVT Process Voltage Temperature

PVTA Process Voltage Temperature Aging

ML Machine Learning

AI Artifical Intelligence

NBTI Negative Bias Temperature Instability

HCI Hot Carrier Injection

BTI Bias Temperature Instability

TDDB Time-Dependent Dielectric Breakdown

AVFS Adaptive voltage and frequency Scaling

DVFS Dynamic Voltage and Frequency Scaling

ABB Adaptive Body Bias techniques

UDRM User-defined Reliability Model

AES Advanced Encryption Standard

NCP Near Critical Paths

RFA Random Forest Algorithm

SoC System on Chip

ABB Adaptive Body Bias techniques

DAHC The Drain avalanche hot carrier

CHEI Channel hot electron injection

SHCI Substrate hot carrier injection

RDF Random dopant fluctuation

LER Line edge roughness

OTV Oxide thickness variation

AOI Input AND into Input NOR

OAI Input OR into Input NAND

MAJ Majority gate

FA Full Adder

DSTB Double sampling with time borrowing monitors

BICS Built-in current sensors

STA Statistical Timing Analysis

ISM Insitu Monitor

CTS Clock Tree Synthesis

TA Timing Analysis

RMSE Root Mean Square Error

MAE Mean absolute error

MAPE Mean absolute percentage error

FIR Finite Impulse Response

AES Advanced Encryption Standard

VCD Value Change Dump

LR Linear Regression

DC Duty Cycle

LE Logical Effort

OPP Operating Performance Point

Résumé de la thèse en Francais

D'ici 2025, le monde de l'informatique de pointe s'appuiera probablement sur les technologies émergentes des nœuds nanométriques et sur les CMOS les plus avancées et leurs innovations. Depuis 1967, Gordon Moore a prédit l'augmentation du nombre de transistors tous les deux ans [1], une règle qui est toujours valable pour la microélectronique (voir Figure 1). Cette prédiction empirique reste d'une précision surprenante, et a permis d'énormes progrès dans le HPC, les serveurs, les plates-formes de traitement parallèles, les accélérateurs et les capteurs intelligents, en fait tout, d'un smartphone de petite taille à plusieurs gadgets technologiques innovants [2] et ça continue. De nos jours, des transistors encore plus petits et plus avancés sont utilisés dans les puces.

Le concept de l'Internet des Objets (IoT, « Internet of Things ») et les circuits intégrés sont largement utilisés dans l'espace d'aujourd'hui, les automobiles, l'avionique, les soins de santé, l'industrie mobile, etc. Pour rendre possible l'intégration de tels nœuds avancés dans des applications hautes performances, différentes technologies ont été déployées dans le but de contenir les problèmes de dissipation de puissance et de fiabilité tout en maintenant des performances proches des performances existantes. La technologie FDSOI (Fully Depleted Silicon On Insulator) est l'une des technologies utilisées à cette fin. D'autres technologies dites émergentes (PCM, memristive, nanofils, etc.) sont en phase développement pour accompagner la même tendance. Avec les technologies actuelles, la taille d'un transistor pourrait se réduire à quelques nanomètres (par exemple, TSMC produit des processeurs Intel Core i3 sur un nœud de processus de 5 nm en 2021). Cela a été possible grâce à de nombreux processus et technologies de conception innovants ont été utilisés, y compris des matériaux d'interface thermique de soudure et de matrice plus minces pour réduire les coûts et augmenter les performances. En effet, l'efficacité énergétique et le rendement par watt ont été bien améliorés. Cependant, les contraintes de paramètres physiques dans le transistor, les connexions et les couches conduisent à une certaine perte de performances et de fiabilité, et le vieillissement est devenu un réel problème.

La plupart des produits électroniques ne sont pas conçus et optimisés par rapport à leur durée de vie spécifique, qui peut changer de façon importante selon l'application cible. Par exemple, le cas des téléphones portables ou de l'électronique grand public, où il est supposé fonctionner correctement pendant environ 2-3 ans avec une durée de vie maximale de 10 ans. Dans plusieurs autres applications, telles que les applications militaires, avioniques, automobiles et médicales hybrides, plus de 15 à 20 ans de service fiable sont attendus, selon les normes qui régissent leur fonctionnalité. De telles applications doivent maintenir constantes les performances non dégradées des applications au cours d'une durée de vie aussi étendue, malgré le vieillissement du matériel et les différents bruits et erreurs aléatoires.

Au cours de la dernière décennie, plusieurs études de recherche ont été menées sur les défaillances de circuits dans le temps. Il avait inclus plusieurs perspectives, telles que le processus, la tension, la température (PVT, « Process, Voltage, Temperature »), l'impact sur l'environnement d'exploitation, les erreurs logicielles, les défauts de fabrication et les impacts sur les charges de travail, etc. Ainsi, il existe un besoin évident de développer non seulement des techniques de robustesse pour assurer leur fonctionnalité pour la durée de vie cible, mais aussi une infrastructure de test hiérarchique pour des mesures en ligne robustes au sein des circuits et systèmes intégrés. À leur tour, ces mesures en ligne permettront l'auto-test, l'amélioration des fonctionnalités de fiabilité et, en général, la facilitation d'un fonctionnement ininterrompu de hautes performances tout en garantissant les exigences de sécurité. Cependant, ces infrastructures de test doivent être gérées correctement sur puce, pour permettre une adaptation du système suffisamment à l'avance, avant que l'échec réel de la synchronisation ne se produise.

Cette thèse traite d'une solution ambitieuse à ce problème en prédisant le vieillissement des portes dans les chemins critiques en mélangeant les connaissances acquises en intelligence artificielle avec des paramètres de fonctionnement adaptatifs, tels que les tensions et les fréquences.

Motivation de la thèse

L'apprentissage automatique (ML, « Machin » Learning ») est l'une des tendances d'actualité dans le domaine de la science des données. Il est largement utilisé dans l'analyse de données, la classification et la prédiction de diverses données sur la base des données précédemment observées ou des actions ou réactions précédentes selon un algorithme fixe. Le ML permet aux machines de rechercher et d'identifier des informations cachées lorsqu'elles sont exposées à de nouveaux ensembles de données. Aujourd'hui, la plupart des entreprises utilisent l'Intelligence Artificielle (IA) dans de nombreux aspects de leur travail professionnel pour améliorer la qualité et la productivité et créer de nouveaux produits et services basés sur le ML. Par exemple : marketing, services financiers, soins de santé, transports, pétrole et gaz, reconnaissance de la parole et de l'écriture manuscrite, robotique, etc. L'une des subdivisions de l'IA est l'apprentissage en profondeur. Dans la plupart des cas, il est basé sur des réseaux de neurones avec de nombreux neurones empilés en couches et capables de calculer une sortie. L'apprentissage profond est un sous-ensemble de l'apprentissage automatique, qui est également un sous-ensemble de l'IA [3].

Un changement considérable est en train de se produire dans la conception du matériel, poussé par le défi croissant de fournir de bonnes performances et une faible consommation d'énergie sans aucune pénalité en termes de fiabilité et de vieillissement. Cependant, les effets liés au vieillissement augmentent et, par conséquent, la durée de vie d'un circuit intégré est réduite. De nombreuses

recherches ont été menées pour compenser la perte de fiabilité due au vieillissement. Dans des conditions de fonctionnement normales, plusieurs facteurs, tels que la température de tension de procédé (PVT), l'injection de porteurs chauds (HCI, « Hot Carrier Injection »), l'instabilité de la température de polarisation (BTI, « Bias Temperature Instability ») et la rupture diélectrique dépendante du temps (TDDB, « Time-Dependent Dielectric Breakdown ») sont amplifiés dans les technologies nanométriques et peuvent causer des défaillances fonctionnelles et temporelles.

En particulier, le NBTI (Negative BTI) et le HCI provoquent une dégradation plus importante que d'autres sources pour les technologies récentes[4]. Par conséquent, les concepteurs sont plus prudents dans la conception d'un circuit intégré et sont obligés d'introduire une large gamme de bandes de garde de sécurité pour assurer un fonctionnement correct dans différentes sources de conditions de fonctionnement et de dégradation des performances. L'ajout de nombreuses marges de sécurité pessimistes et de moniteurs à l'intérieur de la conception du circuit intégré signifie que la puce finale présentera de sérieuses pertes de performances, avec une augmentation de la surface et du coût de conception. En fait, les marges temporelles doivent être conçues pour prendre en compte toutes les conditions les plus défavorables, qui dans la plupart des cas sont sous-optimales et pas toujours réalistes. De plus, le processus de vieillissement est significativement affecté par le comportement dynamique des circuits [5], ce qui rend l'analyse a priori extrêmement difficile car la charge de travail normale des circuits n'est pas toujours connue lors de la phase de conception de la puce.

De plus, la réduction de la tension d'alimentation n'a pas pu être avancée comme prévu à cause de contraintes d'efficacité énergétique [6]. Les architectures adaptatives de tension et de fréquence (AVFS, « Adaptive Voltage and Frequency Scaling ») ou les techniques Adaptive Body Bias (ABB) sont utilisées depuis la fin des années 2000 pour diminuer les marges de sécurité et compenser les variations [7][8]. Ces méthodes utilisent des moniteurs de performances intégrés insérés à des points stratégiques de la conception pour suivre les fluctuations de synchronisation du circuit, combinés à des tensions adaptatives (VDD et Back Body) et à des schémas de gestion de fréquence pour réduire la consommation d'énergie et éviter les erreurs de synchronisation [8][9]. L'efficacité de ces approches est directement impactée par la qualité et les performances de ces Instruments de Test Embarqués (ETI) : ils doivent être modélisés, caractérisés et utilisés successivement pour les adaptations de l'environnement d'exécution, car l'extraction des données des moniteurs peut être un véritable défi.

En raison de la difficulté d'évaluer correctement tous ces paramètres, y compris l'influence de la charge de travail sur les processus de vieillissement [10], l'apprentissage automatique (ML) est devenu l'une des options possibles pour traiter les données massives obtenues à partir de la simulation et des mesures. Il peut être utilisé pour analyser et prédire des données sur la base des données

précédemment observées et des actions ou réactions précédentes et agir selon un plan fixe. Le ML permet aux machines de rechercher et d'identifier des informations cachées lorsqu'elles sont exposées à de nouveaux ensembles de données.

L'objectif principal de la thèse est de construire un algorithme de Machine Learning capable de prédire la dégradation du délai pour les portes numériques simples et complexes. Pour réaliser la prédiction de dégradation du retard pour l'ensemble du circuit, nous avons commencé avec les portes universelles de base NAND et NOR, pour lesquelles nous pourrions avoir accès à des mesures expérimentales complètes et à des caractérisations dans les technologies FDSOI. En fait, pour les deux portes NAND et NOR à 2 entrées, des simulations SPICE effectuées avec la bibliothèque UDRM (User-Defined Reliability Model) ont été utilisées à des fins de comparaison avec notre modèle. Nous avons formé un framework d'apprentissage automatique (ML) prenant en entrée les principaux paramètres physiques, électriques, environnementaux et topologiques et fournissant une estimation du vieillissement des transistors ou des grilles en termes de dégradation du délai. Ensuite, nous avons comparé les résultats prédits par porte simple avec ceux obtenus par simulation SPICE. Nous avons étendu le travail pour chaque retard de porte, y compris les plus complexes, et les avons appliqués à l'estimation du chemin critique.

Résumé

Les technologies CMOS modernes telles que FDSOI sont affectées par de graves effets de vieillissement qui dépendent des problèmes de niveau physique liés aux technologies nanométriques et à l'environnement du circuit et à son activité d'exécution. Par conséquent, il est difficile d'établir a priori des bandes de garde suffisantes pour les estimations de chemin critique, ce qui conduit généralement à une surestimation importante des délais (et donc à une perte de performances) ou à une durée de vie de fonctionnement trop courte. Dans le même temps, l'apprentissage automatique est l'un des algorithmes tendance pour le traitement des mégadonnées et ses applications associées. Il est pratiquement impossible de traiter des données brutes avec une grande précision. L'approche combinée de conception et de simulation de système numérique comportemental améliore et développe un modèle mathématique pour les performances du système, la fiabilité et les données minimales pour l'apprentissage du comportement du système, du composant ou du sous-système à la fois.

L'objectif principal de la thèse est de construire le modèle mathématique pour chaque porte complexe et les portes de base du circuit numérique entraînées avec des algorithmes d'apprentissage automatique pour résoudre les problèmes de défaillance et de vieillissement de circuit mentionnés ci-dessus. L'efficacité et la précision de la dégradation du délai des portes sont de plus en plus cruciales lors du passage à des technologies telles que le FDSOI 28 nm. Tous nos circuits sont

composés de portes logiques basiques : si on peut prédire leur vieillissement, on pourra donc le prédire pour l'ensemble du circuit. Ainsi, nous partons des portes universelles de base NAND et NOR. Une porte universelle est une porte logique qui peut construire toutes les autres portes logiques.

Dans ce travail, nous avons développé et validé un cadre de vieillissement d'apprentissage automatique en plusieurs étapes : grâce à une analyse théorique, nous avons développé un modèle de vieillissement d'apprentissage automatique pour les portes universelles dans les technologies cibles FDSOI 28n ; Nous avons entraîné le modèle à l'aide des données de portes NAND et NOR fournies par les fondeurs et l'avons validé par rapport à la simulation SPICE avec un taux d'erreur minimum ; Grâce à une approche appelée Effort Logique, nous avons étendu ces résultats à n'importe quelle porte générique ; Nous avons appliqué le cadre de vieillissement résultant à deux circuits de référence, un filtre FIR et un circuit numérique AES (Advanced Encryptions Standard), et avons étudié leur comportement de vieillissement dans différentes conditions et charges de travail. À partir de là, nous avons développé et démontré plusieurs stratégies d'adaptation dynamique de tension et de fréquence tenant compte du vieillissement. Les moniteurs de vieillissement sont souvent utilisés, mais leur placement est essentiel car ils doivent être instanciés sur les chemins quasi critiques (NCP) qui sont plus sujets au vieillissement. Nous avons donc appliqué notre approche pour sélectionner le meilleur sous-ensemble NCP pour l'insertion du moniteur. Les résultats finaux démontrent la validité de l'approche proposée, reproduisant avec précision les données de la littérature avec seulement une fraction des besoins de calcul des anciennes approches basées sur la simulation. Cela nous a permis de proposer des approches dynamiques innovantes basées sur le vieillissement qui peuvent adapter dynamiquement le comportement du système en fonction de son vieillissement, avec des gains significatifs par rapport aux bandes de garde classiques dans le pire des cas.

Contributions de la thèse :

Proposer une méthodologie pour générer un modèle mathématique pour la prédiction du retard induit par le vieillissement des portes complexes en utilisant plusieurs algorithmes de régression linéaire

De nombreuses solutions existent dans la littérature pour surveiller les dégradations de délai dues à la PVT et aux variations induites par le vieillissement. Dans la plupart des cas, les solutions proposées traitent de l'évaluation des contributions individuelles du PVT et des phénomènes induits par le vieillissement sur la fiabilité des circuits à différents niveaux d'abstraction avec un accent particulier sur les techniques au niveau du circuit. D'autres études portent sur des circuits basés sur des capteurs pour surveiller périodiquement ou

pendant l'exécution les violations de synchronisation, avec un certain impact sur la zone du circuit ou les performances du système.

Dans cette étude, nous avons développé une solution adaptée aux conditions de PVT et au vieillissement pour estimer rapidement l'impact des effets PVTA sur les performances du système grâce à un algorithme d'apprentissage automatique entraîné. Pour cela, nous avons établi le modèle mathématique du circuit et un algorithme d'apprentissage automatique pour prédire le vieillissement des circuits numériques en technologie FDSOI. Des données telles que le coin de processus, la tension, la température, le taux de basculement, la probabilité statique, la charge de travail et le temps sont transmises à plusieurs algorithmes de régression linéaire pour la prédiction de base du retard des portes. En plus de cela, le retard des portes complexes et la dégradation du chemin critique sont également obtenus pour différentes cibles de durée de vie.

Développement et évaluation de l'algorithme proposé pour les circuits pouvant fonctionner sous des modes de mise à l'échelle dynamique de tension et de fréquence

Un algorithme a été développé pour incorporer la dynamique induite par les techniques de modification dynamique (DVFS) de tension et de fréquence dans le calcul du retard de chaque porte complexe et chemin critique d'un circuit numérique donné. Avec l'aide de l'algorithme proposé, nous sommes en mesure de prédire avec précision la fin de vie des circuits et donc changer la tension ou la fréquence de fonctionnement en raison d'objectifs de performance/puissance/fiabilité.

Cette caractérisation aide le circuit à fonctionner avec les paramètres de tension et de fréquence optimaux et à anticiper les défaillances temporelles. L'algorithme a été expliqué en détail, ainsi que la méthodologie permettant d'obtenir les résultats de performance/puissance sur des circuits complexes tels que des filtres numériques et des processeurs cryptographiques AES.

Démontrer l'application d'un autre algorithme de ML : algorithme de Forêt Aléatoire (RFA, « Random Forest Algorithm ») et comparaison avec un algorithme de régression linéaire multiple

L'estimation de prédiction de retard proposée peut être utilisée hors ligne pour anticiper les défaillances temporelles potentielles. Nous avons également appliqué le modèle RFA et évalué les résultats de performance par rapport à l'algorithme de régression linéaire multiple. La comparaison montre que l'algorithme de régression linéaire multiple est une approche mieux adaptée et plus simple pour nos données.

Schéma de la thèse

En dehors de cette introduction, le manuscrit est divisé en cinq chapitres.

Le Chapitre 1 : Introduit les bases fondamentales et présente brièvement le mécanisme de vieillissement des nœuds CMOS numériques et leurs effets sur les performances. Tout d'abord, il présente la dégradation des transistors MOS et les mécanismes de dégradation NBTI, HCI et TDDB. Ensuite, il présente les sources de variabilité pour les dispositifs CMOS, ce qu'on appelle la variation de processus, de tension et de température et son impact sur les performances des circuits. Enfin, l'effet de ces phénomènes sur les portes logiques CMOS et les moniteurs de retard sont discutés. Les types de moniteurs In-Situ intégrés et leur importance pour le flux d'insertion sont discutés. Enfin, leur analyse des erreurs de synchronisation est présentée en détail.

Le chapitre 2 : est consacré aux algorithmes d'apprentissage automatique (ML) et leur classification générale telle que les algorithmes supervisés, non supervisés, semi-supervisés et de renforcement est brièvement expliquée. Chaque algorithme ML et ses sous-divisions sont définis avec des exemples. Dans la première partie de ce chapitre, l'état de l'art des algorithmes ML est passé en revue. Leur avantage et inconvénient de chacun de ces algorithmes sont analysés. Ensuite, nous présentons leurs méthodes d'évaluation. Dans la deuxième partie de ce chapitre, les algorithmes de ML dans les applications embarquées avec des innovations récentes sont discutés. Enfin, l'application de l'algorithme ML a été expliquée.

Le Chapitre 3 : Décrit le cadre ML proposé comme le modèle mathématique et les algorithmes de régression linéaire multiple pour prédire les délais de porte logique et de chemin critique dans un circuit numérique donné. Comment choisir l'algorithme approprié pour nos données et l'importance des paramètres utilisés dans notre algorithme ML sont expliqués. En outre, il explique que le cadre ML a été implémenté dans deux circuits numériques différents tels que le filtre FIR et le circuit crypto AES pour explorer les résultats formés et testés. Ensuite, les Chemins Critiques (CP, « Critical Path ») les plus importants sont classés à l'aide de différentes charges de travail. Ceci est utilisé pour analyser les effets du vieillissement afin de connaître l'importance de la variation de classement sur les CP dans différentes circonstances. Cette approche aide à la sélection des CP pour le suivi. Il explique également la méthode d'évaluation expérimentale avec la comparaison de simulation SPICE avec nos résultats d'algorithme ML entraînés sont analysés. Enfin, il démontre l'utilisation du modèle proposé par rapport à l'algorithme de forêt aléatoire et évalue les résultats.

Le Chapitre 4 : Dans ce chapitre, nous proposons un algorithme de changement dynamique de tension et de fréquence (DVFS) adapté aux circuits numériques à utiliser avec la technologie FDSOI. La sélection de Chemin Critique traditionnelle et le flux d'insertion du moniteur in-situ sont expliqués au début du chapitre. Ceci est suivi par une méthodologie de caractérisation de chemin au niveau du circuit pour différents PVT afin d'explorer la défaillance du circuit du système. Le schéma DVFS est démontré à l'aide des implémentations de simulation au niveau du circuit en utilisant deux approches : l'overclocking du point de fonctionnement

tenant compte du vieillissement et les performances maximales avec un point de fonctionnement plafonné sont expliqués en détail. En conséquence, une méthode simplifiée est développée pour estimer la dégradation induite par le vieillissement en suivant à la fois la variation de tension et de fréquence dans le temps.

Le Chapitre 5 : Conclusion et proposition de futures orientations de travail.

Introduction

By 2025, the advanced computing world will probably rely on the most advanced CMOS nanometer nodes emerging technologies and their innovations. Since1967, Gordon Moore has predicted the increasing number of transistors every two years [1], a rule that is still valid in the microelectronics hardware field (see Figure 1). This prediction framework remains impressively accurate, a fact that had made possible tremendous advances in HPC, servers, parallel processing platforms, accelerators, and smart sensors, in fact everything from a small size smartphone to several innovative technology gadgets[2] and it goes on. Nowadays, even smaller and more advanced transistors are used on a microchip.

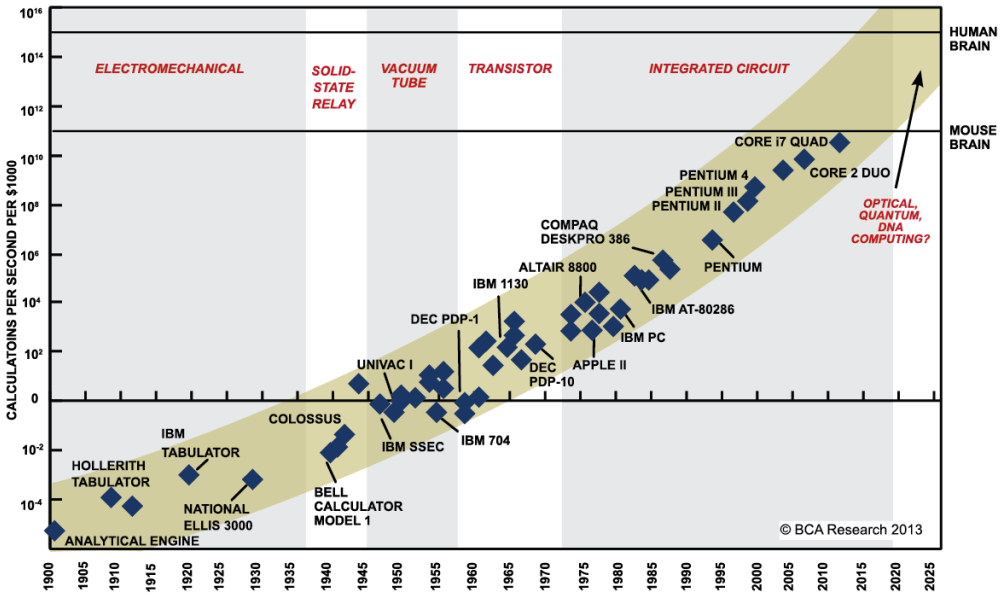


Figure 1: Source: Ray Kurzweil, "The singularity is near: when humans transcend biology", P.67, The vikiking Press, 2006. Data points between 2000 and 2012 represent BCA estimates.

The Internet of Things (IoT) concept and VLSI system applications are widely used in today's space, automobiles, avionics, healthcare, mobile industry and so on. To make possible the integration of such advanced nodes in high-performance applications different flavors of technologies have been deployed in the attempt to contain the power dissipation and reliability issues while maintaining close to existent performances. Fully Depleted Silicon on-chip (FDSOI) Technology is one of the technologies used for this purpose. Other technologies including emerging ones (PCM, memristive, nanofils, etc.) are pointing up to support the same trend. With these current technologies, the size of a transistor could shrink to few tens of nanometers (for example, TSMC produces Intel core i3 CPUs on a 5nm process node in 2021). As many innovative processes and design

technologies have been used, this was possible, including thinner die and solder thermal interface materials to reduce the cost and increase the performance. Indeed, the energy efficiency and the performance per watt were well improved. However, physical parameter constraints in the transistor, wires and layers lead to a certain performance and reliability loss, and aging became a real problem.

Most electronics products are not designed and optimized with respect to their specific lifetime. For example, the case of mobile phones or Consumer electronics, where it is assumed to work properly for about 2-3 years with a maximum lifetime of 10 years. In several other applications, such as military, avionics, automotive, and hybrid medical applications, more than 15-20 years of reliable service is expected, according to standards that are ruling their functionality. Such applications need to keep constant, the non-degraded performance of the applications within such an extended lifetime, despite hardware aging and different random noise and errors.

In the last decade, several research studies have been done related to circuit failure in time. It had included several perspectives, such as process, voltage, temperature (PVT), operating environment impact, soft errors, manufacturing flaws and workloads impacts, etc. Thus, there is an obvious need to develop not only robustness techniques to ensure their functionality for the target lifetime, but also a hierarchical test infrastructure for robust online measurements within integrated circuits and systems. In turn, these online measurements will enable self-testing, enhancing dependability features, and, in general, facilitating uninterrupted high-performance operation while guaranteeing security requirements. However, these test infrastructures need to be managed properly on chip, to allow system adaptation sufficiently ahead of time, before the actual timing failure could happen.

This thesis deals with an ambitious solution to this problem by predicting the aging of gates within critical paths by mixing trained artificial intelligence knowledge with adaptive operating parameters, such as voltages and frequencies.

Thesis Motivation

Machine Learning(ML) is one of the hot topic trends in the data science domain. It is widely used in data analysis, classification and prediction of diverse data based on the previously observed data or previous actions or reactions according to a fixed algorithm. ML enables machines to search and identify hidden information when exposed to new data sets. Today, most companies use Artificial Intelligence (AI) vision in many of aspects of their professional work to improve the quality and productivity and create new products and services based on ML. For example: marketing, financial services, healthcare, transportation, oil and gas, speech and handwriting recognition, robotics, etc. One of the subdivisions in AI

is deep learning. In most cases, it is based on neural networks with numerous neurons stacked in layers and able to compute an output. Deep learning is a subset of machine learning, which is also a subset of AI [3].

A tremendous change is taking place in hardware design, driven by the growing challenge of providing good performances and low power consumption without any penalties in terms of reliability and aging. However, the aging related effects are growing and as a consequence, the lifetime of an IC design is shrinked. A great amount of research has been done to compensate for reliability loss due to aging. Under normal operating conditions, several factors, such as Process Voltage Temperature (PVT), Hot Carrier Injection(HCI), Bias Temperature Instability(BTI), and Time-Dependent Dielectric Breakdown(TDDB) are magnified in nanometric technologies and generate functional and timing failures.

In particular, NBTI and HCI cause more catastrophic degradation than other sources for recent technologies[4]. Therefore, designers are more cautious in designing an IC and are forced to introduce a large range of safety guard-bands to ensure correct operation under different sources of operating conditions and performance degradation. Adding lots of pessimistic safety margins and monitors inside the IC design means that the final chip becomes will present serious losses in performance, with an increase of area and design cost. In fact, timing margins need to be designed to consider all worst-case conditions, which in most of the cases are suboptimal and not always possible. In addition, the aging process is significantly affected by the circuits' dynamic behaviour [5], which makes a-priori analysis extremely difficult as the normal circuit workload is not always known at the chip design phase.

Moreover, scaling down the voltage supply could not be pushed forward as intended for energy efficiency purposes[6]. Adaptive voltage and frequency architectures (AVFS) or Adaptive Body Bias techniques (ABB) have been used since the late 2000s to decrease safety margins and compensate for the variations[7][8]. Such methods use embedded performance monitors inserted at strategic points within the design to track circuit timing fluctuations, combined with adaptive voltages (V_{DD} and Back Body) and frequency management schemes to reduce energy consumption and avoid timing errors[8][9]. These approaches' efficiency is directly impacted by the quality and performances of these Embedded Test Instruments (ETIs): they have to be modeled, characterized, and successively utilized for run-time environment adaptations, as extracting data from the monitors can be a real challenge.

Due to the difficulty of properly assess all these parameters including workload influence on the aging processes[10] Machine Learning(ML) became one of the possible options to deal with massive data obtained from simulation and measures. It can be used to analyze and predict data based on the previously observed data and previous actions or reactions and act according to a fixed plan. ML enables the machines to search and identify hidden information when exposed to new data sets.

The thesis's principal objective is to build up a Machine Learning algorithm able to predict the delay degradation for digital simple and complex gates. To achieve the delay degradation prediction for the whole circuit, we have started with the basic universal gates NAND and NOR, for which we could have access to full experimental measures and characterizations in FDSOI technologies. In fact, for both 2 input NAND and NOR gates, spice simulations performed with the User-defined Reliability Model (UDRM) library were used for comparison with our model. We trained a Machine learning (ML) framework taking as inputs the main physical, electrical, environmental, and topological parameters and providing an estimation of transistor or gate aging in terms of delay degradation. Then, we have compared the simple gate predicted results with those obtained through spice simulation. We extended the work for each gate's delay including the complex ones and applied them in the critical path estimation.

Summary

Modern CMOS technologies such as FDSOI are affected by severe aging effects that depend on physical level issues related to nanometric technologies and the circuit environment and its run-time activity. Therefore, it is challenging to establish a-priori sufficient guard bands for Critical Path estimations, usually leading to large delay overestimation (and loss of performances) or a too-short operating lifetime. At the same time, Machine learning is one of the trending algorithms for big data processing and its related applications. Processing raw data with high accuracy is impractically not possible. Modern machine learning algorithms are intrinsically integrated. The combined behavioral digital system design and simulation approach improves and develops a mathematical model for system performance, reliability, and minimum data for learning the system behavior, component, or subsystem at a time.

The thesis's principal objective is to build up the mathematical model for each complex gate and basic gates of the digital circuit trained with Machine learning algorithms to tackle the afore-mentioned circuit failure and aging problems. The efficiency and accuracy of delay degradation of gates are increasingly crucial when moving to technologies such as 28nm FDSOI. All of our circuits are composed of basic logic gates: if we can predict their aging, we will therefore be able to predict it for the whole circuit. Thus, we start from basic universal gates NAND and NOR. A universal gate is a logic gate that can construct all other logic gates.

In this work, we developed and validated a Machine Learning aging framework through several step: Through theoretical analysis, we developed a Machine Learning Aging Model for Universal Gates in the target FDSOI 28n technologies; We trained the model using foundry-provided NAND and NOR gate data, and validated it against Spice simulation with minimum error rate; Thanks to an

approach called Logical Effort, we extended these results to any generic gate; We applied the resulting Aging Framework to two Reference Designs, a FIR filter and AES (Advanced Encryption Standard) digital circuit, and studied their aging behaviour under different conditions and workload. Starting from this, we developed and demonstrated several aging-aware dynamic voltage and frequency adaption strategies; Aging monitors are often used, but their placement is critical as they need to be instantiated over the Near Critical Paths (NCP) that are more prone to aging. We therefore applied our approach to select the best NCP subset for monitor insertion. The final results demonstrate the validity of the proposed approach, accurately replicating Literature data with just a fraction of the computational needs of legacy simulation-based approaches. This allowed us to propose innovative aging-based dynamic approaches that can dynamically adapt the system's behavior based in its aging, with significant gains if compared to classical worst-case guard bands.

The main contribution of this work is :

Propose a methodology to generate a mathematical model for complex gates aging induced delay prediction by using multiple linear regression algorithms

Many solutions exist in the literature to monitor delay degradations due to PVT and aging induced variations. In most of the cases, the proposed solutions deal with the evaluation of individual contributions of PVT and aging induced phenomena on the reliability of the circuits at different abstraction levels with a particular focus on circuit-level techniques. Other studies deal with sensor-based circuits to monitor periodically or at runtime the timing violations, with a certain impact on the circuit area or the system performance.

In this study, we have developed a PVT and aging-aware solution to quickly estimate the impact of the PVTA effects on the system performances through a trained machine learning algorithm. For that, we established the mathematical model of the circuit and a machine-learning algorithm to predict digital circuit aging in FDSOI technology. Data such as process corner, voltage, temperature, toggle rate, static probability, workload, and time are fed to multiple linear regression algorithms for basic gates delay prediction. Further to that, delay of complex gates and critical path degradation are also obtained for different lifetime targets.

Developed and evaluated the proposed algorithm for circuits that can operate under dynamic voltage and frequency scaling modes

An algorithm was developed to incorporate dynamics induces by voltage and frequency scaling modes in the delay calculation of each complex gate and critical path of a given digital circuit. With the proposed algorithm's help, we are able to accurately predict the end of life for circuits that need to change the voltage or the operating frequency due to performance/power/reliability targets.

This characterization helps the circuit operate with the optimum voltage and frequency parameters and anticipate potential timing-induced failures. The algorithm was explained in detail, together with the methodology of achieving the performance/power results on complex circuits such as digital filters and AES crypto processors.

Demonstrate the application of another ML algorithm: random forest algorithm (RFA) and comparison with multiple linear regression algorithm

The proposed delay prediction estimation can be used offline to anticipate potential timing failures. We have also applied the RFA model and evaluated the performance results compared to the multiple linear regression algorithm. The comparison shows that the multiple linear regression algorithm is a better suitable and more straightforward approach for our data.

Thesis Outline

Apart from this introduction, the manuscript is divided into five chapters.

Chapter1: Introduces the fundamental basics and briefly presents the overview of the aging mechanism of digital CMOS nodes and their effects on performance. Firstly, it presents the degradation of MOS transistors and the NBTI, HCI, and TDDB degradation mechanisms. Then, it presents the sources of variability for CMOS devices the so called process, voltage and temperature variation and its impact on circuit performances. Finally, the effect of these phenomena on CMOS logic gates and delay monitors are discussed. Types of embedded In-Situ monitors and their insertion flow importance are discussed. At last their timing errors analysis are presented in detail.

Chapter 2: is dedicated to the Machine Learning (ML) algorithms and their broad classification such as supervised, unsupervised, semi-supervised and reinforcement algorithms are briefly explained. Each ML algorithms and their sub-divisions are defined with examples. In the first part of this chapter, the state of the art of ML algorithms are reviewed. Their advantage and disadvantage of each of these algorithms are analysed. Then, we present their evaluation methods. In the second part of this chapter, ML algorithms in embedded applications with recent innovations are discussed. Finally, the application of the ML algorithm has been explained.

Chapter 3: Describes the proposed framework ML like the mathematical model and multiple linear regression algorithms to predict logic gate and the critical path delays in a given digital circuit. How to choose the suitable algorithm for our data

and the importance of parameters used in our ML algorithm are explained. Further, it explains that the ML framework was implemented in two different digital circuits such as FIR filter and AES crypto circuit to explore the trained and tested results. Then, the most important critical paths are ranked using different workloads. This is used to analyse the effects of aging to know the importance of ranking variation on CPs under different circumstances. This approach which helps in the selection of CPs for monitoring. Also it explains the experimental evaluation method with SPICE simulation comparison with our trained ML algorithm results are analysed. Finally, it demonstrates the use of the proposed model compared to the random forest algorithm and evaluated the results.

Chapter 4: In this chapter, we propose an algorithm for dynamic voltage and frequency scaling (DVFS) adapted to digital circuits to be used with FDSOI technology. The traditional path selection and In-situ monitor insertion flow are explained at the beginning of the chapter. This is followed by circuit-level path characterization methodology for different PVT to explore the circuit failure of the system. The DVFS scheme is demonstrated using the circuit level simulation implements using two approaches: Aging aware operating point overclocking and maximum performance with capped operating point is explained in detail. As a result, a simplified method is developed to estimate the aging induced degradation by tracking both the voltage and frequency variation over time.

Chapter 5: Conclusion and propose future work directions.

Chapter 1: Aging Mechanism of Digital CMOS FDSOI Technology

A.Introduction

Integrated digital circuits profit from technology over increasing transistor count and complexity in reducing its size. In modern systems, the circuit parameter variation during its lifetime becomes critical. In terms of the ideal case, the constant supply voltage is given to the circuit but in the real case, on-chip variations take place due to glitches, resistivity, fluctuations of supply prominently vary the supply voltages. Design and implement the digital circuits using sub-nanometer CMOS technology. The size of the chip is scaled down prominently than the supply voltage. Electric fields are stronger inside the circuit: this increases the aging effects, so parameters shifts are fast and important. The changes that occur in the device parameter influence whole concert degradation throughout the lifetime period. System on chip (SoC) has been cost-effective 10 years back, but this idea of SoC single chip is not enough to reduce the cost and satisfy consumer quality products. In recent years, there is a demand for cheap and quality Electronics products. Reliability is essential in the quality of the products. The reliability issue in terms of cost, area, and power with respect to performance goals changes from one technology to another technology scale. Significantly, advanced nanometric technology nodes need more attention towards cost-effective solutions.

On the other hand, the optimized design will degrade the reliability of the circuit. This shows that we are in need of low cost, consistent and resilient design method. So, several research types have been done on aging issues faced by CMOS technology[11][12]. The lifetime of an IC depends on many factors, as was pointed out in the introduction section. In this section, we will provide a brief fundamental description of CMOS circuit transistor aging.

A.1 CMOS Circuit Transistor Aging

Complementary MOS (CMOS) is a combination of p-type and n-type MOSFET transistors, sometimes called pull-up and pull-down networks. Scaling down the transistor size tends to provoke more reliability issues due to aging-related physical phenomena within CMOS devices[13]. Significantly, the NMOS device presents more dominant aging issues when compared to a PMOS device. In this

section, the impact of NBTI on standard library components is discussed, focusing on the relationship between gate delays and the threshold voltage, which incorporates NBTI degradation performance.

One of the essential properties of a logic gate is its delay. Reducing delay increases the overall performance of the circuit. On the combinational circuit, the propagation delay between input and output increases or decreases depends on the circuit stress along with time, temperature, VDD, its topology, etc. Under stress conditions, NBTI in PMOS of a pull-up network has more impact in reliability terms [14]. Practical stress patterns of NMOS and PMOS transistors in the same circuit may vary over time and depend on the structure of the circuit and the input pattern of the circuit. Different circuit structures and workloads will result in different shifts in the threshold voltage of PMOS and NMOS devices. Moreover, various shifts will result in different gate delays for different logic gates.

The delay for the CMOS inverter is derived using the α-power law model [14]. The charging and discharging time delay is given by the following equation (2).

$$t_{pHL}, t_{pLH} = \left(\frac{1}{2} - \frac{1 - v_\tau}{1 + \alpha}\right) t_T + \frac{C_L V_{DD}}{2 I_{D0}}, \quad (1)$$

Where v_T, α - Constant, t_T – The input waveform proportional to the transition time from high to low/low to high , C_L – Output capacitance or Load capacitance of CMOS inverter, V_{DD}- Drain saturation voltage, I_{D0} – Drain saturation current, For the short-channel MOSFET case, V_T and α can be 0.2 and 1[14]. Thus, the delay increases when the threshold voltage reaches a high value also depending on the transition time period t_T. Delay degradation for CMOS logic gates has been investigated with an electrical simulation using the Spice wear-out model[14].

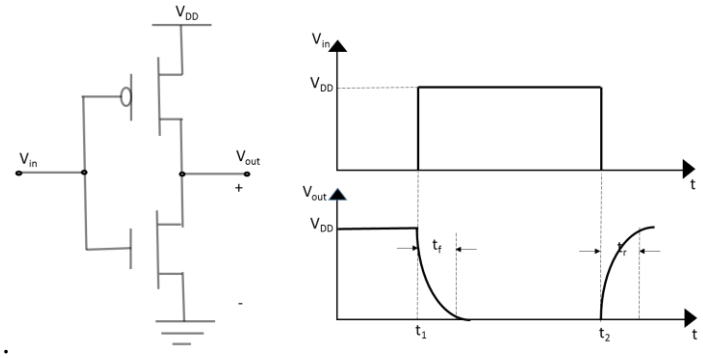


Figure 2: (a) CMOS Inverter, (b) Propagation Delay of Rising and Fall Time

Figure 2 (a) shows the block diagram of the CMOS inverter along with the propagation delay of the rise and fall time output signal. The inverter's rise and

the fall time are noted and plotted in the right-side diagram at times t1 and t2. The supply voltage V_{DD} and the gate to source voltage are either -V_{DD} or 0. The NMOS transistor has input from the ground, and the PMOS transistor has input from $V_{DD.}$ The terminal V_{in} and V_{out} is input and output. When high voltage is given at the input terminal V_{in} of the inverter, the PMOS becomes an open circuit and NMOS switched OFF so the output will be pulled down to V_{SS}. When a low-level voltage applied to the inverter. The NMOS switched OFF and PMOS switched ON. So the output becomes V_{DD} or the circuit is pulled up to V_{DD}. The ON and OFF state of the transistor determines the speed and static power consumption of the transistor. The ON and OFF current ratio of the transistor determines its performance.

According to the α-power law, the delay is inversely proportional to the threshold voltage V_{th}. i.e., the delay is dependent on the threshold voltage [14].

$$\tau \approx \frac{C_L V_{DD}}{I_{D0}} \alpha \frac{V_{DD}}{(V_{DD} - V_{th})^{\alpha}} \tag{2}$$

Using the above equation (2), the delay degradation, along with the threshold voltage, can be formed[15]. Equation (3) gives the relationship between the threshold voltage and the delay degradation.

$$\frac{\Delta \tau_d}{\tau_d} \propto \frac{\alpha \Delta V_{th}}{(V_{DD} - V_{th})^{\alpha}} \tag{3}$$

Thus, the aging of the transistor plays a prominent role in delay degradation. The above equation is applicable for all the basic gates in the circuit one can estimate the delay degradation.

In microelectronics, the effects of aging are an important factor in designing the chip on a nanometre scale. The gate oxide thickness of the transistor is the superior factor of aging phenomena. This thesis concentrates on the aging of each gate in a complex FDSOI technology scale and how it impacts the circuit gradually, leading to the increased circuit failure rate. It is necessary to understand the transistor functioning and the physical phenomena of aging, detailed in this chapter before moving forward.

There is a change of characteristics of transistor age due to the physical parameter of the transistor. The transistor aging phenomena can be classified as Bias Temperature Instability (BTI), Hot Carrier Injection (HCI), and Time-Dependent Dielectric Breakdown (TDDB). The mechanism of HCI and BTI plays a vital role in circuit aging. Thus, it is necessary to know the degradation facts before they arise in the circuit. Each of these aging effects and changes in the physical parameters is discussed in the following sub-sections.

A.2 Inverter Delay model

One of the input parameters is Aging Delay, modeled and expressed into a polynomial expression. However, there will be more than one parameter are involved with aging.

A model for delay estimation of a MOSFET model has been proposed and defined with T. Sakurai and R. Newton's alpha-power law in 1990 [14]. The propagation delay of a cell is expressed by alpha power law as:

$$Delay_{Cell} \; \alpha \, C_{out} \frac{V}{I_d} \tag{1}$$

Where, C_{out}- output load capacitances, v – supply voltage and I_d- Drain current. Equation (1) is the basic equation to derive an equation for an Inverter. From this model, the propagation delay with voltage and temperature is given in equation (2).

$$Delay(V,T) \; \alpha \; C_{tot} * \frac{V}{\mu(T) * (V - V_{th}(T))^{\alpha}} \tag{2}$$

Where μ(T) – carrier mobility, V_{th}(T) – Threshold voltage at temperature T, α – positive constant (carrier velocity saturation), V – The supply voltage for all cells. The delay model, including aging variation, is expressed as in equation (3). In contrast, the same as the delay model and the time parameter is added[75].

$$\begin{aligned} &Delay(V,T,t) \\ &= \; p_{\beta} \; + p_{\mu-1}(T) \; \frac{V}{V \; - \left(pv_{th}(T)\right) + \Delta pv_{th}(V,T,t)^{p_{\alpha}}} \end{aligned} \tag{3}$$

Where, p_{β} , p_{α} are constant while $p_{\mu-1}(T)$ and pv_{th} are exponential temperature dependence and related to transistors mobility and threshold voltage, respectively.

The above equation (3) is for an Inverter delay degradation[74] and it consists of 8 parameters. Each of them is described by the extended below formula.

$$p_{\mu-1}(T) = \; C_1 + k_1 T^{n1} \tag{4}$$

$$p_{vth}(T) = \; C_2 + k_2 T^{n2} \tag{5}$$

$$\Delta_{pvth} = V^{\gamma} * e^{-\frac{E_a}{kT}} * (C_1 * t^{n1} + C_2 * t^{n2}) \tag{6}$$

Where, C_1, C_2, k_1, k_2, - fit parameters for a given technology (i.e., FDSOI 28nm) obtained with a high degree of confidence, γ - voltage acceleration factor, E_a- temperature activation energy, k- Boltzmann's constant. (n_1, n_2) - two different

time exponents. Both HCI and NBTI effects degrade the reliability and performance of the circuit by varying the threshold voltage of a transistor with respect to time. Thus, the final Delay model, including aging variations, is expressed in equation (3).

Each Delay model parameter change from one process corner to another. The three process corners with their corresponding eight parameters shown in Table 1 are already validated and explained [74].

	p_β	C_1	k_1	n_1	C_2	k_2	n_2	p_α
SS	6.96e-11	2.72e-11	6.60e-11	1.88	0.57	1.12e-4	1.30	2.60
TT	6.76e-11	2.66e-11	4.18e-11	1.94	0.46	1.15e-4	1.30	2.68
FF	6.88e-11	2.80e-11	7.98e-11	1.94	0.46	1.15e-4	1.30	2.68

Table 1: Parameters for the delay model[75]

This equations (3) does not include any workload dependent parameter. Workload influence inside the circuit is an important factor in the aging analysis. Thus, how this thesis incorporates this parameter to develop our proposed model is explained in the next sub-section.

A.2 Bias Temperature Instability (BTI):

BTI is a destructive phenomenon that mainly affects the threshold voltage of a transistor and continuous on/off delay of the gate, and the path delay of a circuit. BTI is classified into Negative Bias Temperature Instability (NBTI) and Positive Bias Temperature Instability (PBTI). Among these two, NBTI causes more degradation in the circuit than PBTI.

Negative Bias Temperature Instability (NBTI): It is a serious reliability problem for digital and analog CMOS circuits[16]. It occurs mainly in p-channel MOS devices. Degradation follows a logarithmic relationship with time. The principal parameters affecting NBTI are

- transconductance g_m
- linear drain current $I_{d,lin}$
- saturation current $I_{d,sat}$
- Channel mobility μ_{eff}
- subthreshold slope S
- off current I_{off}
- the threshold voltage V_{th}

Figure 3 [17]shows the physical setup for NBTI exploration[18]. The gate is negatively biased while the source, drain, and substrate are connected to the ground. At a well-known temperature, this condition is applied for a few seconds. So, there are no hot carriers stimulated. NBTI occurs mostly in PMOS transistors

with a negative voltage source at the gate. Thinner oxides are near the gate where non-nitride gate oxides are presented. The trapping mechanism, along with degradation, takes place. When there is stress in the PMOS transistor, the threshold voltage increases linearly because of its positive charges. Thus, it decreases the drain current and damages the gate. At higher negative threshold voltage and the elevated temperature is maintained. A high gate to source voltage V_{GS} and low drain to source voltage V_{DS} stress condition exists. Every defect is caught with time depending on whether it is positive or negative charged.

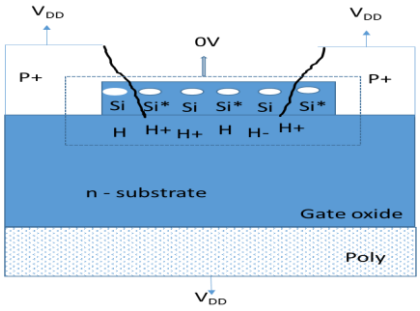


Figure 3: Schematic of the Physical mechanism of NBTI

Positive Bias Temperature Instability (PBTI): It occurs mostly in the NMOS transistor. PBTI causes much lesser degradation compared to NBTI if no high-k or metal gate is used[19]. Trap generation takes place with a combination of pre-existing electron traps in NBTI and PBTI[20]. The gate's material is coated with metal, and high k-oxides reduce the leakage near the gate [21]. So, along with high k-oxides, PBTI is also noted with high care in the future.

A.3 Hot Carrier Injection (HCI):

HCI is one of the reasons for the degradation of CMOS transistors. Switching on or off the Field Effect Transistor, the current reaches the gate's peak value, causing hot electron injection. Thus, an energy-efficient electron in Si/SiO_2 interface increases the collision in the transistor channel. Figure 4 shows the mechanism of HCI in the NMOS device [22][23][24]. When the Gate to source voltage is higher than the threshold voltage, the transistor in ON state and hot carriers are produced under a high electric field, which is accumulated near the drain region. Few are injected into the gate region. There is a slight change or shifting in the threshold voltage, and these hot carriers degrade the dielectrics. Drain and gate are repelled by the holes, which produce a substrate current. Continuous observation of degradation takes place when there is an electric field in a digital CMOS circuit. HCI degradation mainly depends on the slew rate or fan-out parameter in the circuit[25].

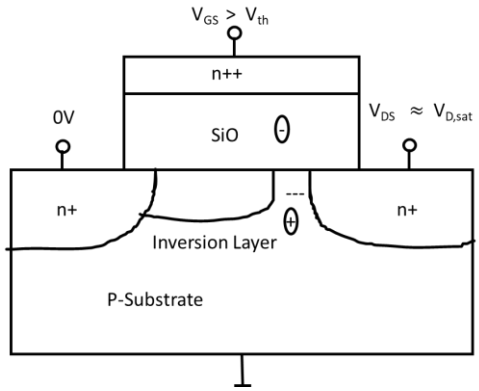


Figure 4: HCI Mechanism

The hot carrier degradation is correlated to the length of the channel, oxide thickness, and supply voltage. There are four hot carrier injection mechanisms[26]. These are

- Drain avalanche hot carrier injection;
- Channel hot-electron injection;
- Substrate hot electron injection;
- Secondary generated hot electron injection.

The Drain avalanche hot carrier (DAHC): Under normal operating temperature, Hot carriers are considered to introduce the worst degradation temperature. At high drain to source voltage and lesser gate to source voltage under stress conditions, the DAHC effect is more significant[27]. A high electric field is applied near the drain region, i.e., the drain voltage is greater than the gate voltage under a non-saturated condition. In turn, the channel carriers produce the drain depletion region. Various research shows that the drain voltage is double the gate voltage times, where there are bad effects occur [28]. The hot carriers' movement hit with a Si lattice atomic, generating an electron-hole pair in the process. This phenomenon is known as impact ionization.

Channel hot electron injection (CHEI): It exists when the gate and the drain voltage is higher than the source voltage. When there is a high gate voltage, the channel carriers move from source to drain or, on occasion, are driven through the gate oxide. The channel hot electrons produce an electron-hole pair known as impact ionization at the drain channel's edge. Thus the substrate collects all the holes and produces the substrate current[29].

Substrate hot carrier injection (SHCI) occurs with a very high positive or negative bias applied at the substrate back[30]. Thus, the carriers leave the substrate and reach the Si-SiO2 interface. This moving process produces more kinetic energy in the depletion area. These hot carriers can overcome the energy barrier at the interface and are injected into the gate oxide, where few are trapped.

The Secondary Generated Hot Carrier Injection: It is a photo-induced generation process[31]. Impact ionization occurs, as mentioned in the CHEI, involving a secondary carrier made by a former incident of impact ionization. The typical condition for impact ionization is that the drain voltage is higher than the gate voltage. This condition is similar to DAHC, but the only difference is that the hot electron injection process occurs at the substrate back bias.

A.4 Time-Dependent Dielectric Breakdown (TDDB):

It is a failure mechanism in MOSFET and a major reliability issue in MOSFET[32]. Under a constant electric field, the strength of the material gets breakdown with time. Thus, the dielectric gets the transition from an insulating phase to a more conductive phase. It is field, temperature, energy, and polarity dependent[33]. The electric field acceleration parameter does not depend on temperature, indicating any correlation between time to failure of TDDB and oxide trapped charges[34]. Under stress conditions, the density function can be approximated as shown in [34] for a particular temperature and voltage.

$$\left(f_s \ln(t) = \frac{1}{\sigma\sqrt{2\pi}(t-t_0)}\left[-\frac{1}{2}\left(\frac{ln(t-t_0)-ln\mu}{\sigma}\right)^2\right]\right) \quad -\infty < \ln t < \infty \tag{4}$$

Where σ- variance, μ - mean, and t_0 – initial failure time, the oxide quality and applied electric field determine the activation energy of TDDB. The lifetime of TDDB depends on its particular temperature range [34]. Scaling down the dimension and usage of low k-material may be problematic for TDDB [35][36].

B.Sources of Variability for CMOS Devices

The current trend of shrinking transistor size and advanced technology nodes increases the digital circuit's performance and speed. Significantly, advanced CMOS-based devices are very much affected by various factors in recent technology nodes. In this section, the source of variability for CMOS devices and how important this variation's impact affects the circuit's performance are as follows. Source of Variations for CMOS device can be sub-divided into two broad categories: Process and the other is Environmental. There are three sources of variability: Process, voltage, temperature, and Aging (PVTA) variations. PVTA variations affect the overall performance and reliability of the circuit, making the temporal behavior of gates and circuits quite non-deterministic.

Furthermore, BTI, HCI, and TBBD degrade the circuit by increasing the voltage threshold of transistors, as discussed in detail in the previous section. Again process can be sub-divided into two they are systematic and non-systematic. Non-

systematic is further sub-divided into two global (Within-die) variations and local (Die-to-Die) variations. Figure 5 shows the general taxonomy of variation in CMOS processes. Each of these variations is discussed further in detail.

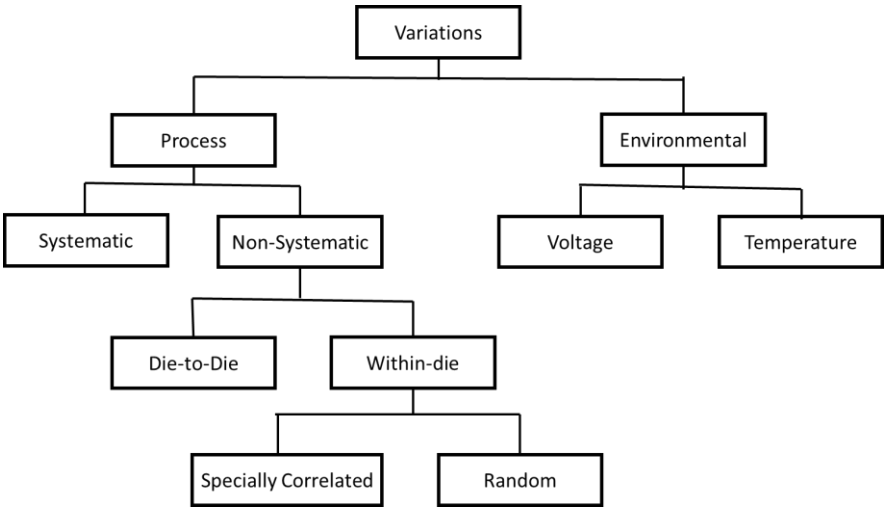


Figure 5: General taxonomy of variation [37]

B.1 Process Variation

Process variation has been increased a lot due to the scaling of process technologies. This impact variation increases the delay and frequency performances of the digital design. These slight delay variations affect the target frequency and yield. After fabrication, the number of chips operated within the target frequency also decreased a lot. The primary source of process variation in CMOS processes are listed below [37]

Random dopant fluctuation (RDF): It is one of the processes which can change the threshold voltage and alter the transistor's properties. This is due to the variation in the impurity concentration. It is one of the local forms of process variation.

Line edge roughness (LER): It is a process of variation on the edges from an ideal form. That is, varying edge features of the semiconductor. Its been one of the issues while using extreme ultraviolet lithography in making IC chips.

Oxide thickness variation (OTV): Control over atom-level interface between silicon and dielectrics of the gate. It also increases the variations in the device mobility and threshold voltage.

Further, the process variation is broadly classified into two Global process and Local processes which are discussed further.

Global Process: It is also called an on-die or inter-die, or die-to-die variation (including die from different wafers and different wafer lots). Global variability

or global process defines changes in the physical parameter of the transistor channel length, width, layer thickness, resistivity, doping density, and body effect[38]. To know the variation of a global process, device characterization is important between current and voltage to model between fast/fast (FF) and slow/slow (SS) cases of a transistor.

Local Process: Local process is also called Intra-die or within-die variation. That is a variation of gate lengths of different devices within the same die. Local process variation can be further classified as spatially correlated variations (SCV) and random or independent variations (RIV). SCV is one of the within-die variation process that behaves as the same kind of characteristics within-die in the particular area as those are located far away from die. RIV variations are statistically liberated from other variations of the device.

B.2 Voltage Variation

In CMOS technology nodes, the supply voltage and threshold voltage play a vital role in the circuit aspects' performance and reliability. There will be many circuit design challenges when the voltage variation takes place. For nanotechnology IC chips decreasing the supply voltage and threshold voltage leads to an increase in the current leakage. Figure 6 shows the International Roadmap for Devices and System (IRDS) prediction of the technology node's supply voltage and threshold voltage [39]. The threshold voltage does not follow the trend of supply voltage, which decreases the leakage current. The effect of CMOS threshold voltage variation on high-performance circuits

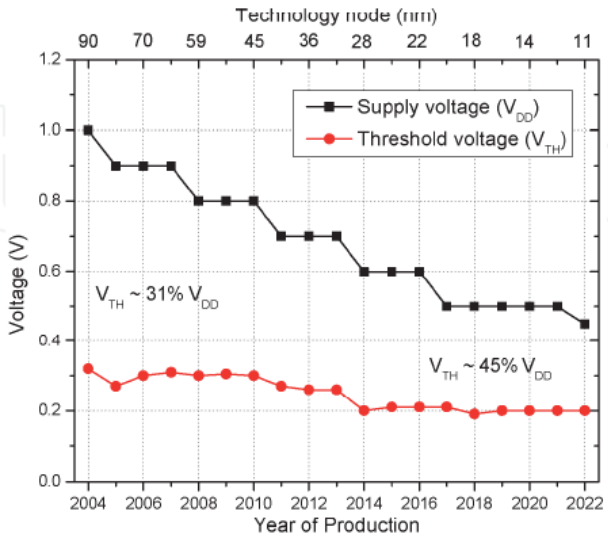


Figure 6: The Supply Voltage and threshold voltage in time [39]

B.3 Temperature Variation

In low-power applications, temperature variation is one of the main variations in CMOS devices. Moreover, the temperature and threshold voltage variation may inverse the N and P transistor current [40]. These may entirely change the circuit's characteristic corners and call it a Temperature Inversion Phenomenon (TIP). The temperature-dependent values are the threshold voltage and the transistor's carrier mobility [40][41]. By increasing the temperature, both threshold voltage and carrier mobility decrease, affecting the circuit timing performance naturally. This leads to sub-threshold leakage current exponentially. Thus, this problem of leakage of a device operating at high temperature. The environmental temperature around us also causes an essential role in fluctuating the die temperature and performances.

The high temperature in the circuit causes slower transistors, higher interconnect resistance and higher subthreshold leakage. Figure 7 shows the IBM chip temperature variations from 0.8C to 30.3C of 0.13nm CMOS technology [42]. The hot spot of the chip varies depending on the circuit's activity and discharge more power leakage. The switching speed of each gate in the critical path increases with the circuit's voltage, temperature, and process variation. The fluctuations in the temperature cause significant variation in the CMOS device characteristics and vary the IC design performances.

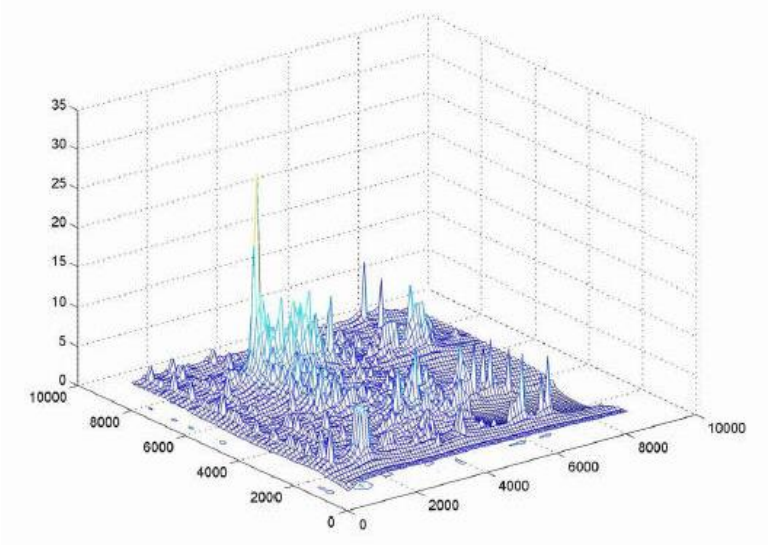


Figure 7: Full chip temperature Increase profile [42]

To consider the impact of PVTA variations become necessary to find an efficient way to overcome these issues. These variations affect the circuit's performance. The voltage variations are due to circuit impedance and temperature variations

produced by different circuit activities. This leads to power dissipation between the circuit blocks. At last, aging effects, particularly BTI and NBTI, causes' gradual degradation of the circuit characteristics and had a significant impact on variations in the circuit.

C.CMOS Logic gates

In an Integrated circuit (IC) design, the system contains many kinds of logic gates and interconnects. Basic logic gates are for example, NOT, AND, OR, NOR, NAND, XOR, and XNOR, but more complex gates have been designed too (AOI (Input AND into Input NOR), OAI (Input OR into Input NAND), MAJ (Majority gate), FA (Full Adder), etc.). Logic gates have n inputs, but in general, they have only one digital output (except the FA gate). Each individual logic gate can be connected to form a combinational or sequential type to produce a different logic gate function from standard gates. Each gate's delay time should be calculated to estimate such a system's ability to operate at the specified frequency. The IC design's size becomes smaller and smaller due to the technology scaling, making larger transistors and interconnects resistances. Thus, larger resistance results in a larger effect on the gate delay, which is important for IC performance. Therefore, advanced models are necessary for calculating the gate delay accurately and efficiently. The delay calculation techniques are summarized in [43]. Delay monitors are used inside the circuit to achieve nanometric on-chip reliability and circuit performance to ensure fault-free operation [44]. Delay monitors and their insertion flow are explained in detail in the following section

D.Delay Monitors

Delay monitors is one of the method used to avoid pre-error or post-error timing faults. The timing errors caused by variability and the physical phenomena degrade the circuit. As aging-induced phenomena influence the performance and Reliability of CMOS digital monitors, or sensors are placed in the middle or at the end of the circuit to measure the system's degradation level and reliability performance. Degradation of physical and electrical parameters in the circuit is the main reason so far. The main researches on monitors for aging-induced effects are listed as follows. Inline resistance faults and struck-open faults inside the circuit can also be monitored and detected [45]. Other defects are monitored through the current degradation Idd_t, which can be detected while measuring the aging of the circuit [46]. For example, Built-in current sensors (BICS) are used for fault detection in CMOS integrated circuits [47]. However, a few types of researches had been done to highlight the lifetime of the circuit.

In this section, solutions for reliable and process variation-aware design of CMOS integrated circuits are discussed in detail. In-Situ Monitors are used either to extracting or to reacting at the circuit delay degradation. This information is converted into a digital domain where the degradation part is identified precisely. Hence, the delay faults are measured, and the necessary mitigation is taken. The circuit's life period is prolonged by taking the necessary actions, and system failure is avoided. In this section, different approaches for the implementations are discussed.

D.1 Embedded In-Situ Monitors

Embedded in-situ monitors are placed (Inserted) at the end of the critical paths to monitor the particular path's error delay and pre-error delay. It is very difficult and costly to directly predict the errors before they arrive in a digital circuit. One possible solution to this problem is the Pre-error approach used in In-Situ Monitors[48]. Consequently, scaling down the voltage supply could not be pushed forward as intended for energy efficiency purposes[6]. In this thesis, embedded monitors with error detection and pre-error detection, along with their examples, are discussed in detail

D.1.1 Embedded monitor with Error Detection

Examples of error-detection embedded monitors are double sampling with time borrowing monitors (DSTB), TDTB monitors, Razor-I and Razor-II, etc. To overcome the challenges in critical path replication approaches, a Razor flip-flop can be used. Here a single path is identified as the critical path to be monitored [45]. Figure 8 shows the Razor flip flop block diagram with main and shadow flip flop in followed to XOR gate. An on-chip timing checker is used to check the critical path timings. A delayed clock is used for both master and shadow flip flop to capture the data and comparisons. The value latched in the main and shadow flip flop may vary due to its scaled supply voltage leads to giving an error signal at the end. Thus, the error rate depends on the supply voltage limits till the point where the error becomes unacceptable. When the design has more critical paths, more shadow latches are needed, which results in efficiency reduction. If the error rate increases, this might affect the overall system performance because of the increased latches.

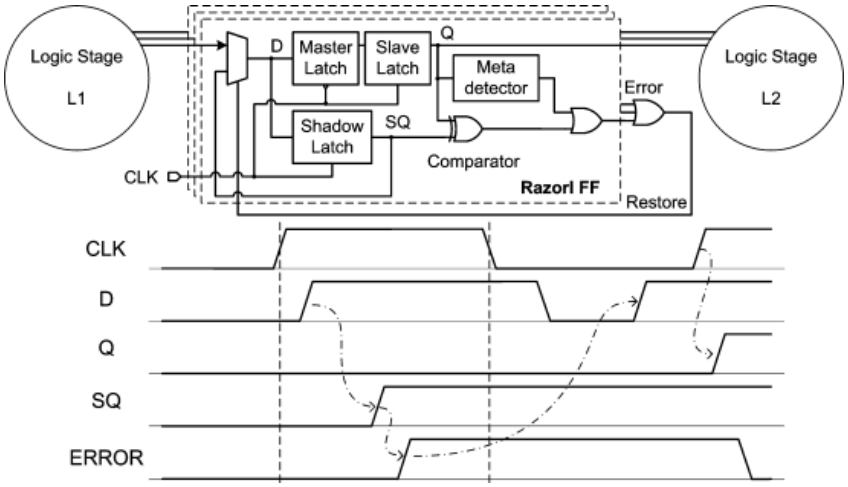


Figure 8: Razor-I flip-flop[28]

D.1.2 Embedded Monitor with pre-error detection

The error detection scheme helps the circuit to predict the circuit failure before it arises. For pre-error detection, embedded monitors are In-situ monitor with buffer delay, In-situ monitor with passive delay, In-situ monitor with master delay, and In-situ monitor with a canary flip flop, etc. Figures 9, show a block diagram of a canary flip flop [9]. The difference in the value between captured and shadow flip flop at the end of the output generator. This warning generator or pre-error flag indicator is used to find the circuit speed, variations, and aging-aware voltage adaptation. This type of flip flop is simple and easy to implement in the circuit to automate the process. These types of monitors are embedded inside the circuit, which is more precise and accurate than the external monitors. The timing variations are captured without any failures.

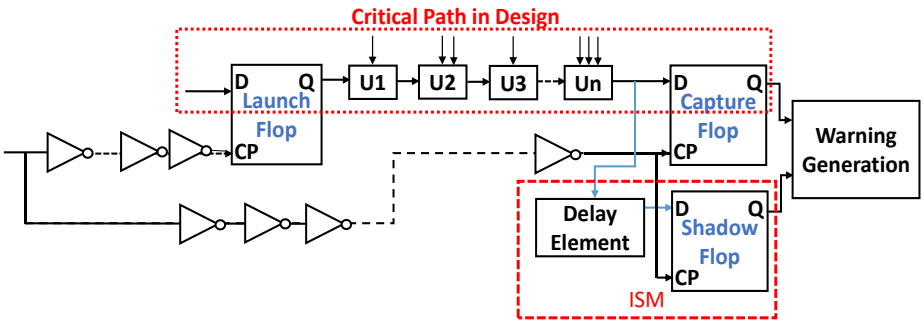


Figure 9: Block diagram of canary flip flop monitor [9]

E. Traditional path selection and ISM Insertion flow

E.1 Critical path selection

Choosing the right subset of Near Critical Path (NCPs) of a circuit to instrument is a complex question, related to both the lifetime utilisation and the optimization processes used at fabrication time. Researches have been done for selecting a critical path for performance optimization [87]. Further, this selection process is not only based on optimization problems but also on Process variations [88]. A circuit is required to operate as fast as possible by monitoring the delay of the CPs of the targeted circuit. The delay of the circuit should not be longer than a given clock period. The slow and fast process is based on the CP delay performance. If the delay of CP is slower than the clock period, its performance are reduced by that margin, which would potentially be exploited.

Before proceeding to the characterization and optimization algorithm, it is necessary that we need to select the critical path selection. In this thesis, worst-case critical paths are considered. After synthesis, all the critical paths inside the circuit are analyzed using the Statistical Timing Analysis (STA) option inside the Design Vision software. Static timing paths are usually noted in four ways:

1.Register – Register (Reg - Reg)
2.Primary Input – Register (PI - Reg)
3.Register – Primary Output (Reg - PO)
4.Primary Input – Primary Output (PI - PO)

Register – Register (Reg - Reg) :

The start point is the clock input pin of the launch Flip Flop and the end point is D input pin of the latch flop as shown in Figure 14. For setup constraints, the goal is to make sure that the start point's delay is at least the setup time of latch flop less than the clock period of the latch clock.

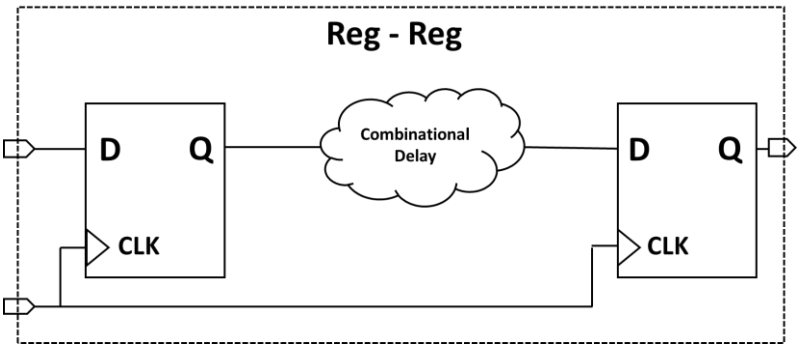


Figure 10: Register – Register (Reg - Reg)

Primary Input – Register (PI - Reg)

The start point will be input ports and an end point will be the D pin of the latch flop. One can assume an input delay from the input port to the combinatorial cloud as shown in Figure 15. The input delay can include pad delay or any net delay.

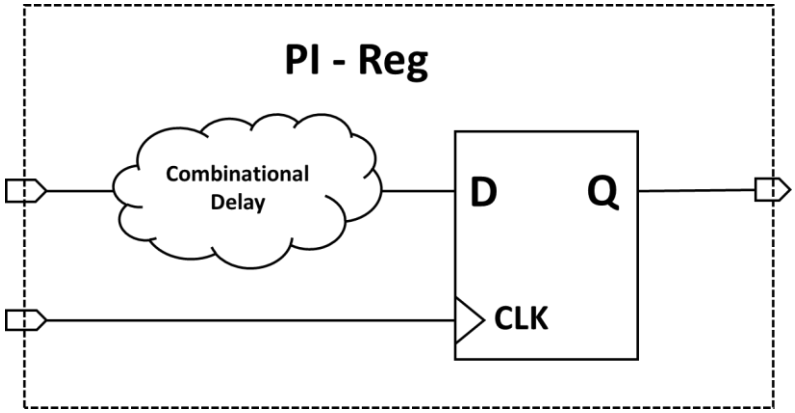


Figure 11: Primary Input – Register (PI - Reg)

Register – Primary Output (Reg - PO)

The start point will be the clock pin of the launch flop and an end point will be the output port. One can assume an output delay from the output of the combinatorial cloud as shown in figure 16. Output delay can include pad delay or the net delay from combinatorial cloud to pad.

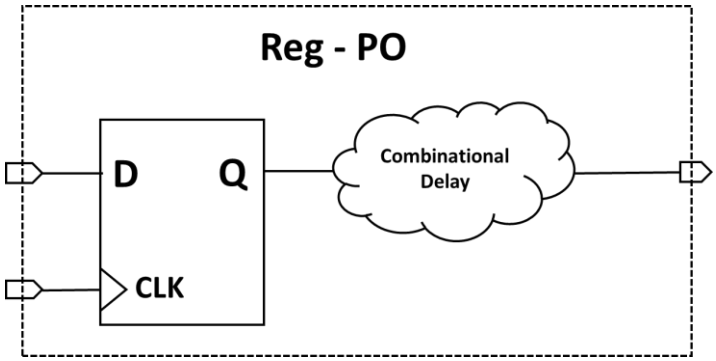


Figure 12: Register – Primary Input (Reg - PO)

Primary Input – Primary Output (PI - PO)

The start point is the input port, whereas the end points are output ports, as shown in figure 17. One can assume input delay for input ports and output delay for

output ports. Typically, the combinatorial paths between inputs and output ports are constrained to meet the minimum and maximum delay constraints.

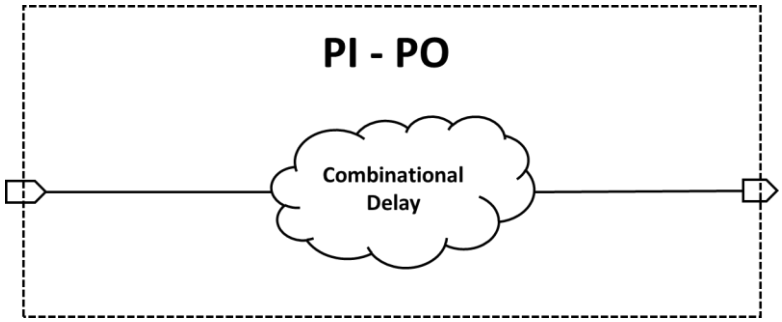


Figure 13: Primary Input – Primary Output (PI - PO)

Figure 14 and Figure 15 show the Path Slack analysis provided by Design Vision: the path slack for most worst-case is noted and highlighted in yellow color, whereas the start and end of the critical path are highlighted in blue . It shows the graph between the number of paths and the slack time. For our project, the worst-case critical path slack is taken was highlighted in yellow color on the left side and marked as blue color on the right-hand side was taken. Here all the worst-case paths are Reg – Reg blocks.

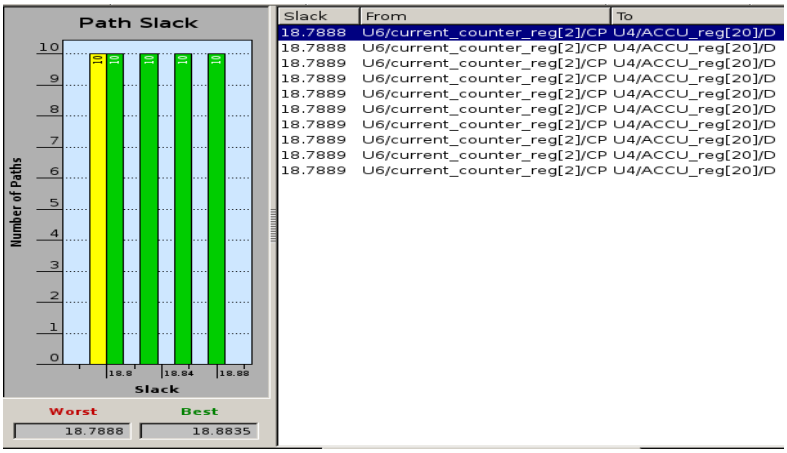


Figure 14: Path Slack of FIR Filter

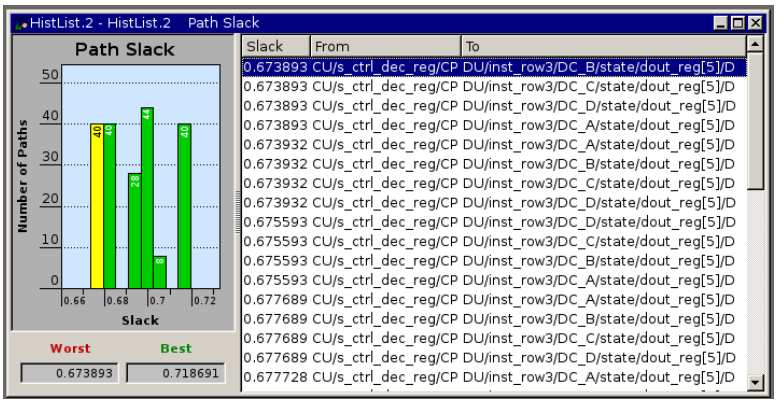


Figure 15: Path slack of AES Circuit

In this thesis, the worst-case critical path are considered and the paths are chosen manually. For the FIR filter, 21 NCP are considered to analyze the performance of the circuit, whereas, from the AES circuit, 150 NCP are taken. The choice of PVT is based on application specifications. These NCPs are considered more important for predicting a circuit's characterization and performance with a limited life cycle. The following section describes the critical path characterization of FDSOI technology.

E.2. Critical Path Monitoring

Large-scale CMOS ASIC and other digital systems are suffering from a power consumption problem. Power saving can be managed by various factors such as PVT, aging, and other electrical, physical parameters. By controlling the supply voltage and frequency, we can achieve saving the power for many devices. Thus ADVS technique can control the voltage and frequency to increase the power efficiency and the performance of the circuit. The voltage and frequency relationship can be determined using critical path analysis of a circuit. We can measure this by using a ring Oscillator or inserting a delay line between the critical path. Figure 16 shows the critical path monitoring technique for AVFS is shown below.

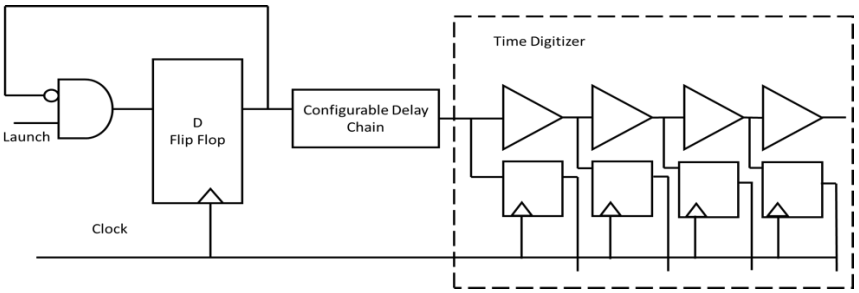


Figure 16: Example of a Critical Path Monitoring Technique[45]

The delay chain comprises different digital gates such as inverters, NAND, NOR gate, wire segments, etc., and the output can be selected from one of the points.

Thus the critical path delay is noted and captured. A configurable delay chain is used in a critical path monitoring technique. During the AVS operation, the start and the end of the clock are captured and stored using a set of the buffer at the end. The AVS system's feedback adjusts the supply voltage to meet the timing constraints between the delay chain and critical path meet margins.

In order to measure the delayed launch, an edge at the start of the clock cycle and then capture the edge at the end of the clock cycle at the output of the delay chain and a set of buffers. Then the buffer outputs that are captured are analyzed to determine the exact positioning of the rising edge. Now the feedback employed by the AVS system continually adjusts the supply voltage such that the delay chain and critical path meet the timing constraints with adequate margins. The adjustment of the supply voltage is made such that the launch, rising edge makes it to a specified buffer stage, which ensures the timing of the delay line.

There are lots of challenges in critical path monitor: Enough safety margin has to be included. Otherwise, there is a chance of a mismatch between delay lines and actual delay path and huge process variation. Thus, in order to maintain a safe operation, an additional delay margin is to be maintained. Selecting a single critical path and placing a monitor becomes very challenging. Sometimes the overall delay being close to each other due to a different combination of logic and interconnect delay paths.

E.3. In-Situ Monitor (ETI) Insertion flow

The efficiency of ETI relies on activity on the paths where they are inserted. The conventional method to insert ETI is to find a list of setup critical paths from static timing analysis and target the worst critical paths for ETI insertion, especially the In-situ monitors. The generic approach is illustrated in Figure 17. The classical Front-end steps are executed with logic synthesis or physical synthesis steps. In the end, a gate netlist is provided as input to the placement and route tool. After placement of gates and of the clock tree synthesis (CTS) and CTS optimization (Setup and hold optimization), timing analysis (TA) is performed. For the chosen functional corner, a decision is made to insert ETIs and to regenerate connectivity and delay calculation on a sub-set of critical paths. It comes up with an updated netlist, timings, and power. The flow is normally executed: detailed routing and optimization (timing, power, IR drop, signal integrity, etc.).

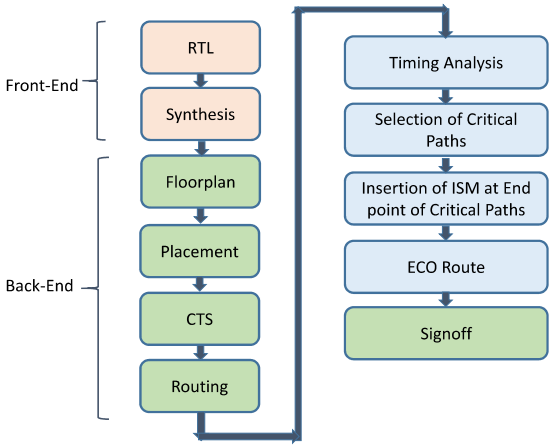


Figure 17: ETI Insertion methodology flow in digital design [44]

Based on this method and for a complex digital design, the number of ETI to be inserted can become rapidly huge. These designs have hundreds of thousands of Flip-Flops, and a careful selection of those FF to be used as ETI has to be done. Also, synthesis tools have tendencies to propose physical gate netlists implementations with well-balanced paths. Therefore, the number of subcritical paths to be monitored can be significant. Even if from the first number of FF we can consider only the subset of FF that are endpoints of these sub-critical paths, the number of the FF can still be quite high, generating significant area overhead and making it challenging to handle ETI alarms at a reasonable time. It is essential to be able to select only meaningful aging, sensitive critical, and subcritical paths to be monitored for setup delay violations.

The delay of a given path may degrade depending on the environmental conditions, time, and the application running on the circuit. Path delay degradation due to aging has a logarithmic relationship with time, being more important at the beginning of the utilization time, and saturating after that. In paper [49], accurate RTL simulation has been performed to extract the endpoint signal probability. Then the aging-induced shifts of the critical paths are estimated for each workload by using BTI-aware static timing analysis. Therefore, the global aging model is built with signal probabilities of the selected endpoint.

The workload has an additional crucial impact on the delay degradation of the path. Different workloads executions degrade the delay of the paths in completely different ways, and as a result, the higher-ranked paths at design time may be different from the critical path after the execution of specific workloads. Figure 18 shows how far a given circuit, the ranking of age paths, changes due to the workload.

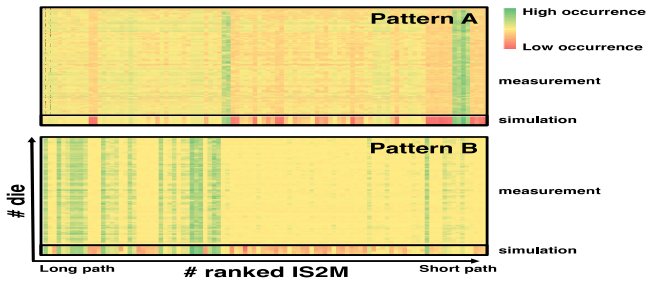


Figure 18: Pattern dependence of critical path ranking[50]

Assuming N workloads are available and can be used to extract the activity profiles of gates and paths in the circuit, we can identify the more sensitive paths to aging-induced degradation delay and which one can become critical near-critical paths at a given moment in the future. The path has to be identified and buy its endpoint individually, as the monitor has to be inserted right at that endpoint Flip-flop. Figure 19 shows different path activations during different workloads that one circuit may experience.

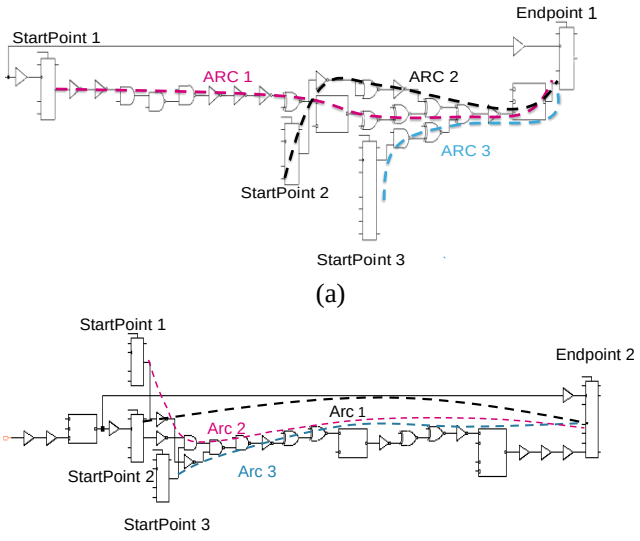


Figure 19: different path activations scenario based on the workload[50]

To detail a bit more this analysis, the following methodology is used:

First, an average activity of the outputs of the gates is extracted from the workload by using a combination of application simulation and activity extraction tools.

1. With this activity, the delay degradation of logic gates due to aging is estimated for the standard logic cells of the library
2. Finally, the full delay degradation can be projected at the targeted-end of-life time,
3. Static Timing Analysis is performed with this new delay degradation data, and the first collection of critical and near-critical paths is obtained.

4. Fan-in cone analysis is performed to identify all the endpoints that cover all the sets of paths.
5. The final set of flip-flops is obtained, and ETI can be inserted at the endpoints

E.4 Example of In-Situ Monitor (ETI) Insertion

The In-situ monitor (ETI) and sensors are used to monitor the digital circuit's aging and its functioning point. At the beginning of our work, we targeted the FIR filter to extract the worst-case critical path using the Synopsis Design Vision tool shown in figure 20 and evaluated the results with a test bench using ModelSim to get the waveform of the circuit as viewed in figure 21. Then, ETI were placed at the end of the worst-case critical path of the FIR filter and placed exactly at the register named 16, 17, 18 as highlighted and separated in yellow colour dash lines in Figure 22. At time t = 0, these 3 monitors are inserted manually. The delayed critical path with no alert signal and the next stage output is noted in register 16.

We aim to predict the aging of the circuit. Thus we need to compare the monitor output alert signal at time t = 0 and t = 1.

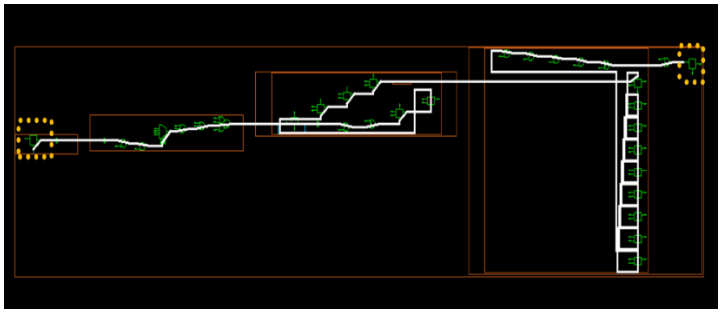


Figure 20: Critical path of FIR Filter

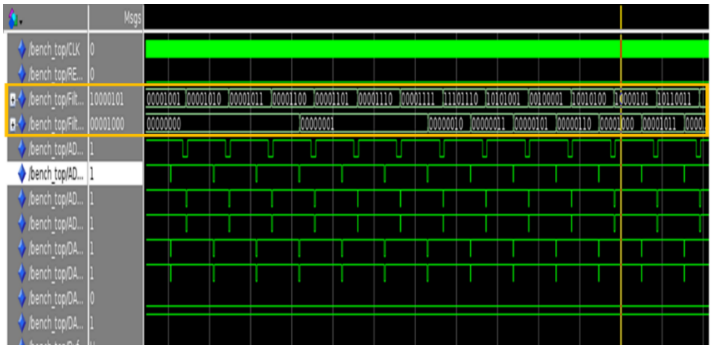


Figure 21: Waveform of FIR Filter

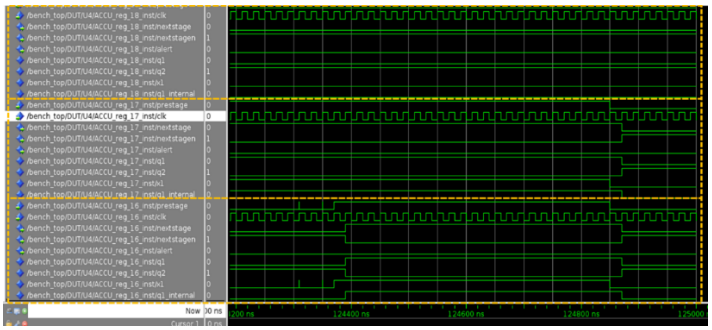


Figure 22: Waveform of FIR along with ETI monitor

This task allowed us to validate that once an Endpoint has been identified thanks to NCP ranking, ETI insertion can be done in a fully mechanical way without any further analysis: even though we stopped at manual insertion this process could be easily automated.

F.Low power CMOS Design

The low power design is a technique to reduce the dynamic and static power consumption of a digital IC. The below equation gives the power components.

$$P_{Total} = \alpha f C_L V_{DD}^2 + t_{sc} V_{DD} I_{peak} + V_{DD} I_{leakage}$$

Where α is the activity factor, f – frequency, t_{sc}- transmission time, C_L- capacitive load, V_{DD}- supply voltage, $I_{leakage}$ – leakage current, I_{peak}- peak current. The power component is related to the activity, frequency, transition time, capacitive load voltage, leakage current, and peak current. Nowadays, many industries and researchers are pushing their boundaries on new techniques to reduce power consumption. Most of the low power techniques have emerged, where the dynamic power is adjusted during the run-time of the system [51][52]. There will be a need for low power design because it affects the overall cost of the systems. The low-power design techniques are discussed in detail below.

There are many advanced techniques for low power design. Among them, two of them are listed below,

1. Dynamic voltage and frequency scaling (DVFS)
2. Adaptive voltage and frequency scaling (AVFS)

F.1 Dynamic voltage and frequency scaling:

DVFS, the name itself, allows the digital circuit to modify the system's frequency and supply voltage. DVFS technique is an effective way to reduce power consumption for the low power circuit. There will be a trade-off between voltage

and frequency operation of the circuits. Lower the voltage will generally run the circuits slowly and there will be a need to check the frequency of operation need to be often rechecked when the voltage reduces. The equation for switching power in CMOS circuit is

$$P = kCFV^2$$

Where, k – percent time switched, C – capacitance, F – frequency, V – voltage. The equation was showing that optimization energy will be adopted by reducing the voltage as low as possible. In reducing the chip size to nanometer processes, the effects of voltage and frequency change over increases the leakage heavily, resulting in a decrease in the optimization energy.

F.2 Adaptive Voltage and Frequency Scaling:

AVFS is a dynamic power minimization technique that changes the supply voltage in coherence with the chip's power supply during run-time. It is also called as closed-loop dynamic power minimization technique. The performance of the chip is directly determined by the run-time and the optimal voltage-frequency correlation. AVFS is used for many applications such as ASIC, microprocessor, and System-on-chip (SoC), etc., Often the chip is supposed to have an NBTI issue while increasing the voltage supply. This problem is solved by using AVFS to contest with the system supplies. The NBTI degradation is seized possible by fixing the sensor in the AVS system[53].

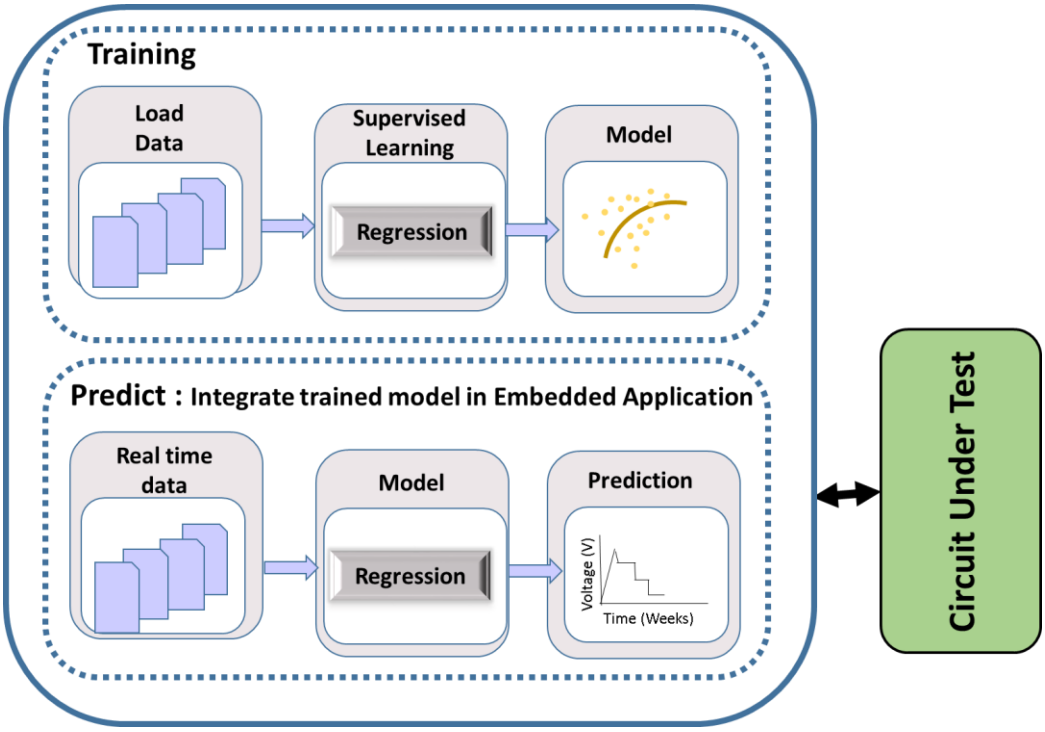


Figure 23: Closed-loop for AVFS schematic [54]

Figure 23 shows the closed-loop for AVFS schematic for our proposed model. Developing real-time analytics for embedded system start with a workflow of training. In the training flow, stored data is used to develop for preprocessing then categorize the data which is numeric or categorical. Analyze and process the data to get ready for finding the right algorithm. After understanding the data, the next step is choosing the right algorithm that is fit for our predictive solution. For our data we chose a Multiple Linear Regression ML algorithm for predicting the adaptive voltage based on the degradation of each worst-case critical path of our target digital circuit. The error flags are predicted by our proposed ML model. Thus, the supply voltage under each particular time period was noted and adjusted based on our predictive algorithm.

G.Conclusion

This chapter introduces aging mechanisms in digital CMOS circuits, namely, BTI, HCI, and TDDB mechanical variations. These three mechanical variations are the major resource for the gradual degradation of the circuit. It also explains the source of variability for CMOS devices and CMOS logic complex gates. This variability also one of the reasons for the changes to the circuit degradation and its performances. Then, CMOS logic complex gates and delay monitor types and their importance are explained. The safety guard bands are imposed on the circuit to prevent the circuit from degradation. But, it is not appropriate for the large circuit. The following chapter discusses the machine learning algorithm and its importance. Further, state of the art is explained with an existing solution of handling the circuit without safety guard-bands to avoid timing closures. This solution increases energy efficiency drastically by avoiding safety guard bands.

Chapter II : Machine Learning Algorithms

A.Introduction

Machine Learning (ML) is fundamental to artificial intelligence. As intelligence needs information, the system needs to gain information, allowing self-learning. When new data is entered, it automatically assisted to learn, grow, change, and develop by themselves. The Learning system model block diagram is shown below in Figure 24. When designing a learning system, we first need to collect our input data. The next stage consists of training and testing, which is essential whenever ML takes place. We need to train and test the ML algorithm with the selected input data.

Real-life examples of ML systems are the Page Ranking algorithm (Example: Google), Recommendation system (Example: Amazon and Netflix), and in-stock exchanges[55] (Example: Prediction using Deep learning), but today they are present in all aspects of our life.

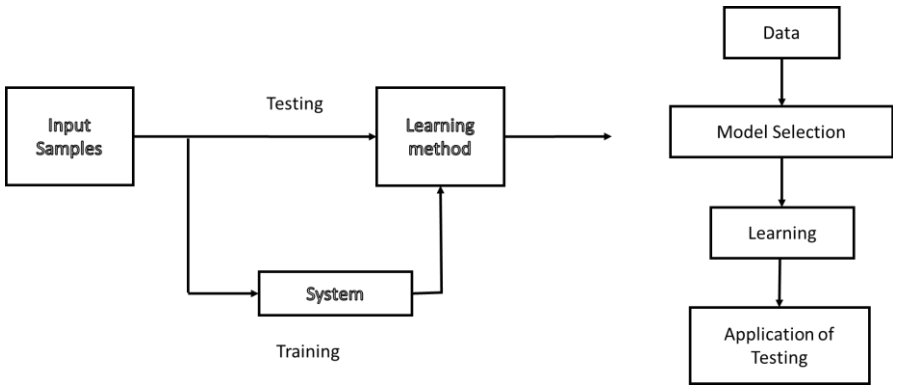


Figure 24: Learning System Model[56]

In an input dataset, a training subset is implemented to build up a model or a function, while a test sub-set (also called a validation) is to validate the model built. Data points in the training set are excluded from the test (validation) set. In Machine Learning, we try to create a model (or a function) to predict the test data. So, we will use the training data to fit the model (by using one of the ML algorithms based on its input data and problem) and testing data to test it. The functions generated are to predict the unknown results, which are actually the test sub-set. So the input dataset is divided into train and test set to check and improve accuracy and precisions by the final system.

The proportion to be divided is completely dependent on the input set and the task to be solved. It is common sense to have 70% of the data for training and the rest for testing, but it is not necessary to be in this proportion. So, assume that we

trained it on 50% data and tested it on the rest 50%. The precision will be different from training it on 90% or so. This is mostly because, in Machine Learning, the bigger the input dataset to train, the better.

Machine learning uses data to generate an internal mathematical calculation that learns to make predictions. It usually defines the common mathematical structures that approximate the distribution of data. An example of this would be weather prediction.

In this chapter, all the learning methods of machine learning are detailed with example and their pros and cons are explained. In addition to describing a specific application of machine learning is illuminated.

B.Machine Learning Types

Machine learning (ML) has different learning styles based on its data problem. In ML, there are two types of grouping: Learning type-based grouping and problem-type-based grouping [57]. There could be many ML algorithms that are possible for the same problem. Thus, finding the right fit for our problem and data would be a great challenge. Figure 25 shows the types of machine learning algorithms with example cases. Each type of ML algorithm is detailed in this thesis.

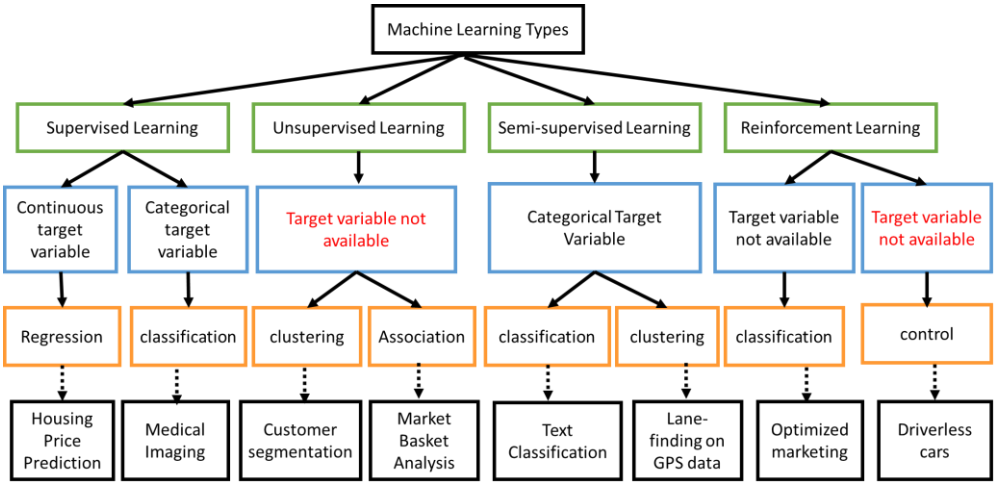


Figure 25: Machine Learning Types [57]

B.1 Supervised Learning Algorithm

In supervised learning, is where you have input data and you know what are their desired outputs variable (classes, or output data). Mainly supervised learning algorithms are used in predictive modeling or in classification. The features or attributes are trained to predict the future from the input data. The predicted new data is based on the previous and historical data sets. Supervised learning algorithms can be broadly classified into two sub-groups.

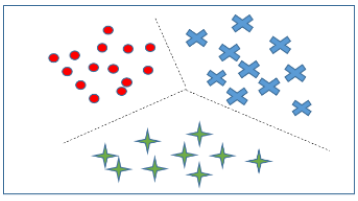
1.Regression algorithm
2.Classification algorithm

The human designer has to identify the problem before applying it to either regression or classification approach. Regression algorithm aims to predict and estimate a continuous quantity. In contrast, the Classification algorithm aims to predict discrete categories [58]. Some supervised learning algorithms applications are search engines, stock market trading, speech recognition, traffic prediction, bioinformatics, spam detection, object-recognition for vision, etc. The pros of the supervised learning algorithm are a simple method and more accurate prediction. The cons are sometimes you could overfit your algorithm easily, large computational time, unwanted data could reduce the accuracy, pre-processing challenges, in case of incorrect data which will make prediction incorrect and useless.

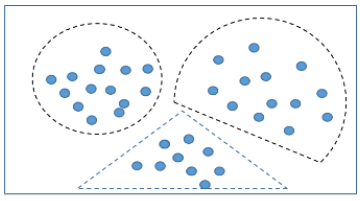
B.2 Unsupervised Learning Algorithm

In opposition to supervised learning is unsupervised learning, i.e., finding input data on its own is called an unsupervised learning task. Clustering, probability distribution estimation, dimension reduction, and handwriting recognition are examples of unsupervised learning[58].

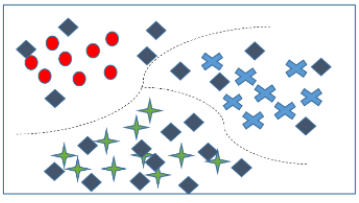
Figure 26 shows the pictorial representation of supervised, unsupervised, and semi-supervised learning systems. It consists of data of different labels which is identified by the algorithms. From the picture, similar grouping features and their classification (i.e., from the figure 26: Dotted, crossed and stared data are grouped) is called supervised learning. By grouping same or similar objects or things else, features that are more frequent together (i.e., data of similar kinds of characters are grouped from unlabeled data) are called unsupervised learning. Semi-supervised learning is the mixture of both supervised and unsupervised learning (i.e., data are labeled and unlabeled with different features) where there is a training input with some of the missing inputs and outputs.



Supervised Learning



UnSupervised Learning



Semi-Supervised Learning

Figure 26: Pictorial representation of Supervised, Unsupervised and Semi-supervised Learning[59]

The main two types of unsupervised learning algorithms are listed below.

1. Clustering algorithm
2. Association rule learning algorithm

The clustering algorithm is to cluster or group input data points to form classes with input data and other external information. Some of the clustering algorithms are hierarchical clustering, k-medoids, k-means, etc. An association rule learning algorithm is used to extract and identify the new patterns from the input data. Some of the association rule algorithms are the Eclat algorithm, à priori algorithm, FP-growth algorithm, etc. The pros of unsupervised learning algorithms automatically split data into groups based on their similarity, identify association mining, and unusual data points detection. The cons are sometimes it's challenging to get accurate information or results from the data. Humans need to spend more time handling and interpreting the data. Spectral properties also change over time and may ruin the classification problem. Unsupervised learning algorithms applications are social network analysis, association mining, climatology, customer segmentation in marketing, to name such a few of them.

B.3 Semisupervised Learning Algorithm

In semi-supervised learning, data can be either classified and unclassified or labeled and unlabelled data. In most of the cases, labeled data are significantly less when compared to the unlabelled data. These labeled datasets allow the algorithm to identify the relationship between your data and give certain information. In case of lacking enough labeled data to produce a precise model, we can use a Semi-supervised learning algorithm to increase the size of your training data. Also, this algorithm helps to label the data and retrain the model with a newly labeled dataset. Figure 26 shows the example of unlabelled and labeled data in semi-supervised learning. The semisupervised learning algorithm can be classified into two types they are

1. Classification Algorithm
2. Clustering Algorithm

The semi-supervised learning-based classification algorithm is used for classification and predictive modeling to observe the input data pattern. The semi-supervised clustering algorithm is used to separate the data set into homogeneous subgroups. It is applied to partially labeled and unlabeled data. The semi-supervised learning-based clustering algorithm works on the inter and intra-cluster similarities of the input data. The application of a semi-supervised learning algorithm is a text document classifier, natural language processing, web crawling, document processing, and modern genetics, etc.

B.4 Reinforcement Learning Algorithm

Reinforcement learning is used to observe the collected information from the environment. It is a method that receives a delayed reward the next time to evaluate its previous action. It was mostly used in games. There are two kinds of reinforcement learning one is positive, and the other is negative. Positive reinforcement learning increases the strength, frequency and maximizes the performance of the model. Negative reinforcement learning should have stopped or avoided behaving negatively and defines the model's minimum performance. Choosing the best method from large rewards is possible using this algorithm. The reinforcement learning algorithm is classified into two they are

1. Classification Algorithm
2. Control Algorithm

The reinforcement classification algorithm varies with the supervised learning algorithm, where the input data is mapped with output data by trained labeled features [60]. Whereas reinforcement learning control algorithm is used for many automatic controlled processes such as reinforcement in combination with feedback controllers developed to heating and cooling buildings[61]. Applications of reinforcement learning algorithms are aircraft control and robot motion control, data processing and machine learning, business planning, robotics for industrial automation, traffic light control, smart sensing, the computer played board games (chess, Go), robotic hand, self-driving cars, etc.

C. Choosing the suitable ML algorithm

It is necessary to deal with our data practically applied to ML learning problems. Below are the steps and methods to identify the right ML algorithm among different data types. They are [62]

1.Categorize the problem
2.Understand your data
3.Find the available algorithm
4.Implement the ML algorithm
5.Optimize hyperparameters

Categorize the problem: First, we need to categorize the data by their input and output. For input labeled data, it’s a supervised learning problem. If it’s input unlabelled data, it’s an unsupervised problem. If the outcome imply an objective function by communicate with an environment, it’s a reinforcement learning problem. If the output of the model is a number, its’ a regression problem. If it is a class output model, its’ a classification problem. If set of input groups is output model, it’s a clustering problem.

Understand your data: The raw data or raw material is essential in the entire process of analysis. Understanding the insight information inside the data plays a

vital role in selecting the correct algorithm for the valid problem. A limited set of data are dealt with some algorithms whereas few algorithms can work with a massive amount of sample data. Some algorithm works with categorical while other work with numerical input data. In this process of understanding the data, the first step is to analyse the data, the second step is to process the data, and the final step is to transform the data.

Find the available algorithm: After succeeding categorization and understanding your data, the next step is to identify the suitable algorithm practically possible to implement concerning to time. Few factors which affect the choice of a model are the accuracy, Interpretability, complexity, scalability, the training, testing and prediction time of the model, at last need to analyse it meets the output goal or not. From the available algorithms of supervised machine learning, we found that the multiple linear regression and Random forest model meet all the choices of our requirements to meet the higher prediction accuracy in supervised regression machine learning algorithms.

Implement the ML algorithm: Implement the data by building a suitable ML model based on our input and output datasets. Another method of approach is to apply the different subgroups of data to the same algorithm. Finally, validate the data with the same algorithm for verification. Thus, we implemented our input data with two different regression algorithms such as Multiple linear regression and Random forest algorithms. Finally, we validated the data with both algorithms and compared their results.

Optimize hyper parameters: Grid search, random search, and Bayesian optimization are the method used to optimize hyper parameters [62].

D.Machine Learning in Embedded Application

ML permits the electronics systems to gain knowledge from the present and previous data to make predictions, and compute values, or interfere in the control steps. These highly performance-intensive applications are usually performed on computers and cloud servers. Nowadays, one of the challenges would be to directly implement machine learning on embedded devices with the help of well-chosen light algorithms and devoted CPUs. Embedded systems for machine learning applications are used to accomplish various responsibilities. Moreover, the rise of IoT applications is one of the reasons for the evaluation of embedded machine learning algorithms. Such embedded machine learning IoT applications could be found in [63] where the authors present an embedded sensor board, or in speech recognition or audio analysis [64][65] (e.g., Apple Siri and Amazon Alexa) embedded applications, monitoring applications [66][67], robotics, network applications [68], drone navigation, etc.

The main key idea of an ML model is input data availability. ML model's main function is model building, and the other is the inference of new information or new data adaptation of the output [69]. For example, data can come from the

embedded camera, microphone, or sensors, etc. The inference operation is using a model to make a prediction on new data. There will be two possibilities: 1. Inference on the cloud, and 2. Inference on edge. Inference on the cloud requires network bandwidth, latency issues, and cloud computes costs and sometimes may undergo security threats. In contrast, Inference on edge has increased privacy and security, faster response time and throughput, lower power, and don't need internet connectivity, hence low power.

The ML is a complex algorithm that uses lots of computations to train a model. But, computations on embedded devices are limited with the amount of memory and compute power available. The embedded system refers to the special computing processing system such as IoT applications. Different types of ML models require an additional amount of memory and time to make predictions. For example, single decision trees have a faster prediction speed. They require a small amount of memory whereas, the nearest neighbor methods have a slower prediction speed and require a medium amount of memory. We need to make a clever decision when determining which model has to be used on a given embedded device.

Most embedded systems are programmed in low-level languages such as C/C++ language, but usually ML is programmed with high-level interpreted languages such as Matlab, Python and R. After that, requirements of the ML model system are considered, such as the available memory and the model type and complexity. Sometimes, the memory size is too small, or the model will take a too long time to produce an on-chip prediction, making the system not adapted to real-time operation. Therefore we need to try other types of models to meet the hardware requirements. Depending on user IoT applications, a designer must carefully consider which tactics may be appropriate for a given hardware consideration, network connections, and budget, which are all key factors to be considered for an embedded design decision.

This thesis's key objective is to explain the real-time problems and solutions of a machine learning algorithm on embedded IoT applications. ML programming can be a hard problem for embedded environments, where memory, energy, clock, and power are very constrained.

Figure 27 shows the major challenges in ML embedded systems [69]. It was grouped into six major groups: execution time of the ML and its dependencies of the memory and data representation, the memory size vs. model size and its inherent characteristics, the power consumption envelope and all considerations related to the power budget, the accuracy of the model computation vs. noise or data representation, the health of the system in terms of susceptibility to internal errors and intrusions, and finally the flexibility and scalability of a given solution. There is a trade-off between these metrics to compensate for the energy performance of the embedded system. Characteristics of ML-based embedded system design such as memory, speed, size, cost, energy, time are well balanced to get better performance. The highly optimized and efficient systems frequently drive embedded applications.

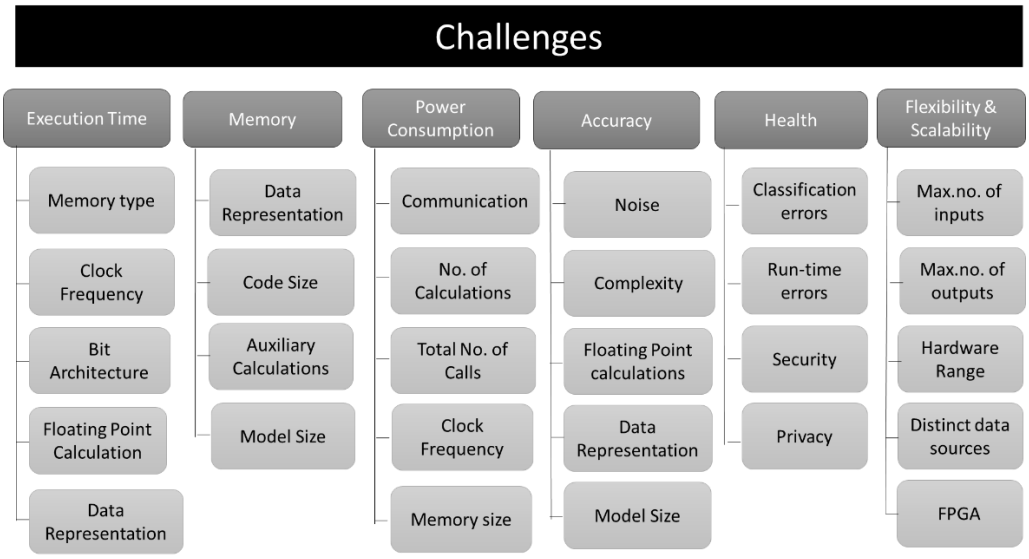


Figure 27: Major Challenges in ML embedded system[69]

ML inferences impact a wide range of markets and devices, especially low-power microcontrollers and power-constrained devices for IoT applications. These devices can often only consume milliwatts of power, and therefore not achieve the traditional power requirements of cloud-based approaches. ML can be enabled on these IoT endpoints by performing inference on-device, delivering greater responsiveness, security, and privacy while reducing network energy consumption, latency, and bandwidth usage.

Processor options for ML workloads:

ML model can be located on two different computing processes: Cloud and Edge-device. Cloud processing is done in data farms or servers(e.g., compute, storage, and networking application services). Edge processing is done on local devices (e.g., sensors and applications). An ML model is a representation or an approximation of a pattern. A system-on-chip is often quite a complex thing, and there are different specialized processors or slightly tuned processors to deal with a lot of different activities. It contains multiple compute engines such as a common processor or Central processing unit (CPU), Graphical Processing Unit (GPU), Digital signal processor (DSP), Accelerators, etc. Choosing the best processor for running ML demands accuracy and response time varies by user cases. It also considers the cost of silicon, area, or power. It is also about the amount of processing one needs to do with the processor and the concerns in terms of silicon area in power. For example, considering several ML workloads and many ML user cases, the observed trend is to push them to smaller and smaller workloads (detection with incredibly low power uses Cortex M3). Speech recognition requires a bit more processing. Visual requires further processing elements to fit with good processor options, real-time requirements and important bandwidth.

E.Evaluation of Machine Learning Algorithm

Performance evaluation (error measures) of ML algorithm is a vital part of any task. In ML experiments, error measurements are used to compare the trained model predictions with the actual data (observed) from the testing data set. Based on the algorithm, the error evaluation method varies. Especially prediction models are essential for evaluating how much the data deviate from observation to access the chosen methods' quality. Most of the survey [70][71][72] says that there are three most essential strategies for performance metric are listed below

1. Mean Square Error (MSE) or Root MSE (RMSE)
2. Mean absolute error (MAE)
3. Mean absolute percentage error (MAPE)

E.1 Mean Square Error (MSE) or Root MSE (RMSE)

The RMSE is one of the error measures of the average magnitude. The equation for the RMSE is the difference between forecast and observed values, squared and averaged over the sample. At last, the square root of the average is taken. In relation, it gives a high weight to large errors. Thus, this RMSE error means it is useful when a system is exposed to large errors. It can range from 0 to infinity. The lower the MSE value represents, the better the model is.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2$$

$$RMSE = \sqrt{MSE(\hat{\theta})} = \sqrt{E((\hat{\theta}) - \theta)^2}$$

n- predicted sample data points and Y – the vector of observed values of the variable $\hat{Y}$ – the predicted values. Where, $\hat{\theta}$ – estimated parameter

E.2 Mean Absolute Error (MAE)

MAE is another method for metric evaluating performance. Instead of squared error, it takes only the absolute value of the difference between the actual and predicted value. If we have lots of outliers, we should prefer MAE to RMSE. If we do not have a lot of outliers, then RMSE should be the preferred choice.

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n}$$

E.3 Mean Absolute Percentage Error (MAPE)

MAPE is also known to be the mean absolute percentage deviation (MAPD), is a calculation of the prediction truth of a forecasting method in statistics. It is usually

expressed as a ratio of the difference between the actual value and the forecast value and the actual value.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

Where A_t is the actual value and F_t is the forecast value. Sometimes, it is reported as a percentage. The absolute value is summed up for each point in the forecast with respect to time and divided by the number of fitted points n.

F.Application of Machine Learning Algorithm

Machine Learning is a combination of computer science and statistics. Since the ML algorithm's evaluation, it made human life easier by calculating and evaluating various tasks. Despite the rapid development of ML, intelligence also plays a fundamental role between humans and machines in doing their chores. The era of intelligence had begun in the middle of the twentieth century. Since then, the artificial intelligence branch of computer science has advanced fast. Some examples of the application of machine learning algorithms are discussed in this paper [73].

The recent research in machine learning focuses on natural language processing, image processing, pattern recognition, etc. All these researches focus on collecting the data based on humans' sense of knowledge and then process the data with artificial intelligence tools to make predictions. Validation and evaluation are needed to know the machine learning algorithm's learned model is worthy or not.

G.Conclusions

This chapter showed that there are many different types of machine learning (ML) problems and algorithms. More importantly, no one algorithm is best suited for all situations. Each application and each input data set has different issues and therefore requires other solutions. We need to choose a suitable algorithm based on its pros and cons. It also depends on the training data's size to get reliable predictions, accuracy and interpretability of the output, speed of training time, linearity, and features.

Choosing a suitable ML algorithm is also the most important and essential step in this thesis. In the context of this thesis, the input data available to us is continuous to time and related to regression problems. Moreover, the research focuses on generating time-series predictions based on a reduced set of reference data but with the possibility of exploiting a theoretical analysis of the problem itself. For these reasons, we chose a supervised machine learning algorithm to deal with our data. We will first develop the framework using Linear Regression, and then we

will explore changing the algorithm which is best suitable for our data based on the evaluation method of RMSE value, which had been debated in this chapter, section C. The lower the value of RMSE, the better the model is.

Chapter III: Machine Learning Model for Aging Estimation

A.Introduction

In Chapter 1, we highlighted the complexity of the physical phenomena related to circuit aging and their correlation with circuit activity, which makes a-priori evaluation extremely difficult. In Chapter 2, an overview of Machine Learning algorithms allowed us to identify how these approaches can help solve our problem. Using monitors and sensors, we can only sense the delay and other transistor features at a given time t. Moreover, it is not possible to insert monitors on all critical paths or to measure all the time. This would not be cost-effective, it would occupy too much area and consume a lot of power. Similarly, a precise analytical analysis of the aging effects would be much too complex in computing terms, and require too much information to be effective.

Therefore, we need to estimate the effect of aging from a limited set of data and with reduced computational effort, but still obtain data that is reliable and precise enough to take mitigation action before delay faults occur. ML algorithms are perfectly suited for this task. ML is a growing technology, which has been playing with a vast and huge amount of data. Modern deep learning and ML algorithms can now give approximate and accurate results

In this Chapter, we present our original contribution: based on the Theoretical analysis of Chapter 1, we will build a Machine Learning model able to predict the aging of FDSOI circuits based on activity parameters measurable in simulation or during run time. The goal is to be able to efficiently estimate the aging of the Most Critical Paths and predict how aging will affect them so that it is possible to take appropriate mitigation measure at both design and run time.

This chapter is organized as follows: first, Section B will provide the details about the implementation of Machine Learning, allowing the prediction of aging for any given Critical Path based on both experimental data for a subset of reference gates and activity measures. Section C first validates the prediction capabilities of the MVL framework against foundry data for the reference gates, and then applies to it two target designs: a Finite Impulse Response (FIR) filter and an AES cryptographic module. Lastly, Section D explores the impact of the MVL algorithm on the framework by replacing Linear Regression with Random Forest.

B.Prediction Framework for Circuit Aging

This chapter offers a new methodology for offline estimation of aging-related delay variation in a digital circuit, applicable from the gate to the circuit level.

Aging degradation takes into account both BTI and HCI efforts, calculating aging-induced degradation under different PVT and activity conditions of the propagation delay for each logic gate. While SPICE simulations for reliability estimation of digital circuit degradation are available for given Operating Points, physical aging models are difficult to apply for online estimates [74]. Therefore, we need an online estimation of the critical path delay with a minimum error rate. We aim to model each gate's propagation delay individually and sum it up to get the critical path's total delay.

This is done in 3 steps: First, we will develop an offline prediction framework, which we will validate against known data coming from SPICE simulations; Second, we will use the validated framework to predict data for gates or Operating Points (PVT and Activity) for which simulation data is not available; Last, we will adopt this framework for online estimation.

B.1 Aging Prediction for Reference Gates

In this chapter, we focus on the Offline Prediction framework. The propagation delay prediction structure can be split into two: Aging Delay Prediction and Logical Effort Conversion. Aging Delay Prediction takes into account aging-induced degradation and propagation delay concerning PVT (Process Voltage Temperature) for gates for which validation SPICE aging simulation data exists. The aim to obtain a verified prediction framework able to provide accurate predictions for any given OPP for the selected gates. Logical Effort Conversion is an abstraction model that allows conversion of delays computed for a reference gate (an Inverter)to other more complex gates. Therefore, we developed a Logical Effort framework, which we validated thanks to SPICE simulation data obtained from Eldo UDRM (User-Defined Reliability Model) API. By composing these two steps (Aging Delay Prediction + Logical Effort Conversion), we can predict aging delay for any given gate, as detailed in the following Sections.

We introduce an approach and a framework to predict the desired lifetime of all types of generic gates as well as the critical path (CP) delay of a digital circuit. A novel Delay aging prediction framework flow chart is illustrated in Figure 28. The overall framework begins with the input parameters such as Delay Aging and logical effort. The proposed methodology has been validated by the universal gates and compared with the SPICE simulation with 0 to 1 percentage error rate. This has been taken as a reference model data, and we further extend our work to model the delayed aging to other standard gates. Critical paths are extracted from our target digital circuit design (FIR Filter). The numerical data that we got from our model is taken as training and testing input features for the supervised learning prediction algorithm called a Linear Regression. Finally, the CP delay aging, along with its desired lifetime, was successfully predicted.

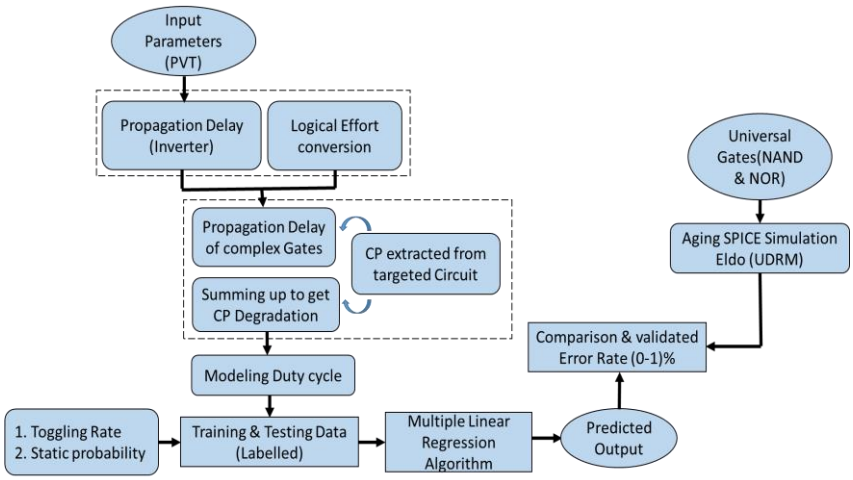


Figure 28: Novel Delay Aging Prediction Framework flowchart

In the following sections, we present the physical and analytical models describing aging's effect on gate Delay and used inside our Prediction framework.

B.1.1 Duty Cycle

The workload and aging dependence on Duty cycle (DC) for a standard cell is defined as a fraction of pulse width to the total period and is usually expressed as a percentage. In digital design, a set of standard cell timing library files of their delay values are available to the designer. A specific aged timing library file with the input signal probabilities of 50% was available along with these files, using a reference. Starting from the delay for 50% activity, the delay from other DC can be computed using equation (7)[76],

$$Delay(DC) = Delay(0.5) * \frac{\tanh(x^{\alpha})}{\tanh(1)} \quad (7)$$

Where,

DC: Duty cycle or input signal probability at the inputs of standard cells

X: stands for the expression DC/(1-DC)

α: Cell dependent fit parameter where, α = function (input slope, output capacitance)

DC of each gate varies when we consider the activity of a circuit: the impact of workload during aging of a standard cells is an important factor. In order to consider it the activity of a each individual gate is extracted during simulation is extracted by dumping a VCD file of gate toggling. The delay of each gates can then be calculated by means of equation (12) and the α static probability value which is extracted from the VCD file.

B.1.2 Extension to Generic Gates: Logical Effort

In 1991, Ivan Sutherland and Bob Sproull invented a new term called "Logical Effort" a method used to estimate delay in a CMOS circuit. The normalized delay of a gate was derived in the following way.

$$d_{abs} = d.\,\tau \tag{8}$$

The normalized delay of a gate can be expressed in the unit as τ. In typical 28nm Technology and below process τ is in Pico seconds. The standardized delay can be expressed in two ways: Parasitic Delay (p) and Stage effort (f) [78]. The parasitic delay, which is an intrinsic delay where the gate has no driving loads, and the stage effort depends on the loads.

$$d = f + p \tag{9}$$

$$f = gh \tag{10}$$

$$d = gh + P_{inv} \tag{11}$$

Where f – stage effort and p – parasitic delay. The stage effort can be further split into two: Logical Efforts (g) and Electrical Effort (h), which can be defined in equation (11).

$$h = \frac{C_{out}}{C_{in}} \tag{12}$$

$$P_{inv} = \frac{Output\ capacitance\ of\ a\ complex\ gate}{Capacitance\ if\ input\ of\ the\ inverter} \tag{13}$$

The exact modelling of the propagation delay from the device level to the circuit level model is quite complicated for a 28nm FDSOI technology. Thus, we introduce a novel model to get an accurate and approximate propagation delay of generic gates with an error rate of 0 to 2%. The novel aging Delay model for the gate is estimated and proposed by a combination of Inverter and Logical effort (LE) delay, as expressed in equations (14) & (15).

$$d_g = (Inverter\ Delay) * (Logical\ Effort)_g \tag{14}$$

$$Delay(V,T,t)_g = \left(p_\beta + p_{\mu-1}(T)\ \frac{V}{V - \left(pv_{th}(T)\right) + \Delta pv_{th}(V,T,t)^{p_\alpha}}\right) * (LE)_g \tag{15}$$

Equation (11) consists of 8 parameters and logical effort, which is already discussed in section (B). The degradation of each gate in the circuit is estimated with the equation (11). Thus estimation of the delay of the generic gate is possible. This method of evaluation of each gates of a standard cell delay is effective with switching activity along with any corner analysis with accuracy.

In the framework of this thesis, we applied Logical Effort to extend delay predictions from the reference fully characterized NAND/NOR gates to the other cells for which we did not have any reference data.

For example: Equation (7) is adopted to get the delay of each gates. For two input NAND gate,

$$Delay(0.5) = Inverter\ Delay\ [Eq\ (3)] * Logical\ effort\ of\ NAND\ gate$$

$$x = DC/(1 - DC)$$

$$DC = static\ probability\ of\ NAND\ gate$$

$$Delay(NAND) = Delay(0.5) * \frac{\tanh(x^{\alpha})}{\tanh(1)}$$

By substituting all the above values in equation (7). We can get the delay of NAND gate mathematically. In such a way, we extended our mathematical model to all other complex gates.

From this data set, we can automate the process of prediction for gates and CPs using a well-known approach called the multiple linear regression algorithm. The linear regression conversion model is explained in the following section.

B.1.3 Critical Path Aging as a sum of Individual gates

The thesis's objective is to analyse the degradation of complex digital circuits due to aging and environmental conditions, and most notably its effect on the Near Critical Paths which limit the maximum working frequency. As each NCP is in fact a sequence of individual gates, we propose to compute the aging effect of the whole path as the sum of contributions from each individual gate composing it.

Aging, environmental conditions and workloads determine the failure mode of the device. Aging-induced degradation for individual gates is investigated with our proposed model.

This step will allow us to observe the effect of aging at the system level in terms of Near Critical Path evolution and distribution. This information will then be used

to define mitigation strategies which will be adapted to both the circuit aging and the target usage model, as explained in Chapter 4.

B.2 Proposed Machine Learning Framework

The ML framework for our model is given by equation (16). Y is the output, f is the prediction function, and x is the aging model's feature.

$$y = f(x) \tag{16}$$

We chose a multiple linear regression algorithm to predict the transistor's aging because we need to reconstruct a tendency starting from a limited set of measurement points. The methodology is shown in Figure 29. First, we selected a set of Training features based on the theoretical analysis: process, voltage , temperature, delay of each complex gates (from equation (7)), toggle rate (extracted from the VCD and SAIF file), Static probability (extracted from the VCD and SAIF file), workload parameter and time. These features populate the Model, which is then Trained using a subset of the data available from technological characterization [77]. The Learned Model is then Tested using the data not used for training. As per ML best practice, the ratio between Training and Test data is 70/30.

To proceed with our target design, follow the procedure which is explained below. The algorithm will examine patterns in the data to correlate with the output results. After training, the algorithm which motivates to predict it for testing new data inputs.

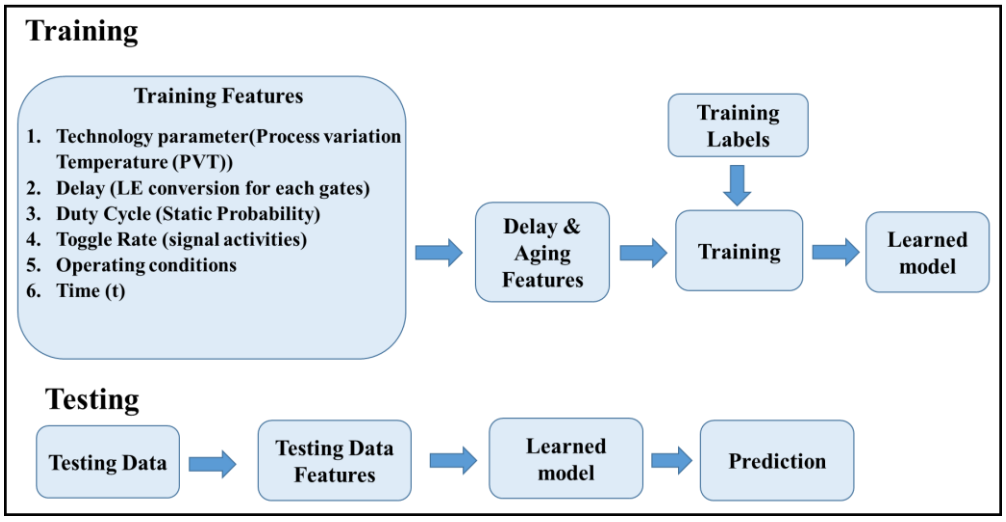


Figure 29: Linear Regression Model

B.2.1 Predictive Modeling

Predictive modeling uses statistical and analytical techniques to predict outcomes[79] based on input data. Using past and present features, future data will be expected or predicted. Predictive modeling is also referred to as predictive

analytics. One of the difficult tasks in predictive modeling is collecting exact or sound data before developing an algorithm to work on. There are various algorithm methods for the prediction model[80]. This thesis uses two different algorithms to compare their impact in the final results: linear regression and random forest

B.2.2 Multiple Linear Regression Algorithm

Linear Regression (LR) takes into account the relationship between dependent and independent variables, for which it creates a generalized continuous function [81]. LR model can be expressed with an equation (14) concerning variables x and y.

$$y = \alpha + \beta_1 * x_1 + \beta_2 * x_2 + \beta_3 * x_3 + \cdots + \beta_n * x_n \quad (14)$$

Where, y is the gate delay, α is y-intercept, β1, β2, β3,….. βn, are the coefficients of gate parameters and X1, X2, X3,……,Xn are feature of gates (e.g. Voltage, Temperature, Time, Toggling rate, slope, load, corners, Input Activity). A regression with more than one variable is called multiple regression. The aim of LR it to minimize the error sum of squared errors (SSE) between observed and predicted results[80].

C.Experimental Validation

In this Section, we will validate our Machine Learning algorithm by applying the methods detailed in Chapter 3-B. To reach this goal, we need to test each step carefully to predict the data from independent sources.

This section reproduced the same structure as the previous one, giving experimental results for each theoretical framework. Section 1 details the setup and results of the ML Prediction algorithm with respect to experimental data, while Section 2 reports the application of Logical Effort to extend the model to generic gates. Section 3 introduces the methodology to extract Switching Activity for complex systems, which is then applied to two reference designs (an FIR filter and an AES crypto processor) in Sections 4 and 5, respectively.

The first step is to test each individual gate of a standard cell if Before proceeds with our target design. We started with the prediction of universal gates and compared their results with SPICE simulation data with 1 percent error difference. Followed by predicting each individual gate as explained in detail in the following sections.

C.1 Activity Aware Aging for NAND and NOR gates

As explained before, it is theoretically possible to obtain precise Aging Delay figures by performing Spice simulations on the gates' low-level models. Unfortunately, these simulations are too computationally intensive and require sensitive information, which only founders have. As a result, simulation data is

usually scarce and limited to a few gate types in a typical setup. We had access to data from the FDSOI 28nm foundry for our ML model, providing detailed measures for NAND and NOR gates[77] for relevant setups. All the data used in our simulation were extracted from Eldo with library User-defined Reliability Model (UDRM) API. We used this data as input to the Methodology of Figure 29 and Figure 30, obtained a Learned Model for using Linear Regression. Figures 31 and 32 compare the Model prediction results (left-hand side) with the reference SPICE data (right-hand side) for Rising-to-Falling and Falling-to-Rising delays for NAND and NOR gates, respectively.

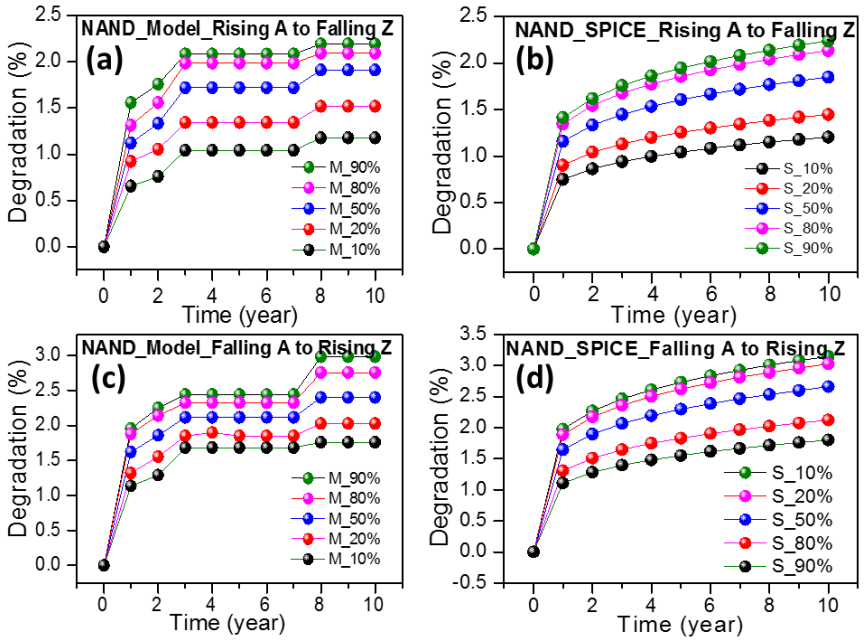


Figure 30: Aging Delay prediction from the limited training set for NAND Gate

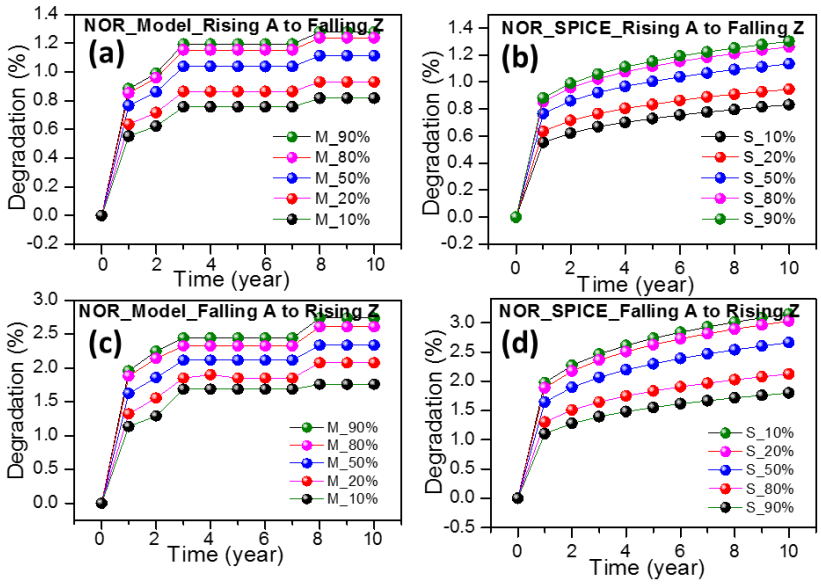


Figure 31: Aging delay prediction from the limited training set for NOR Gate

All experiments have been done for the same Process Voltage Temperature (PVT) point: SS, 1V, 125°C, and each curve plots results for different Duty Cycles, from 10% to 90%. The Model is clearly replicating the same tendency, but an objective validation can only come by computing the Root Mean Square Error (RMSE). The lower the RMSE, the better our model is. Figure 32 plots RSME for NAND and NOR gates for DC at 10% and 90%: is all points, RSME is lower than 1%. This is the same for all the other DC sets, not plotted here for simplicity.

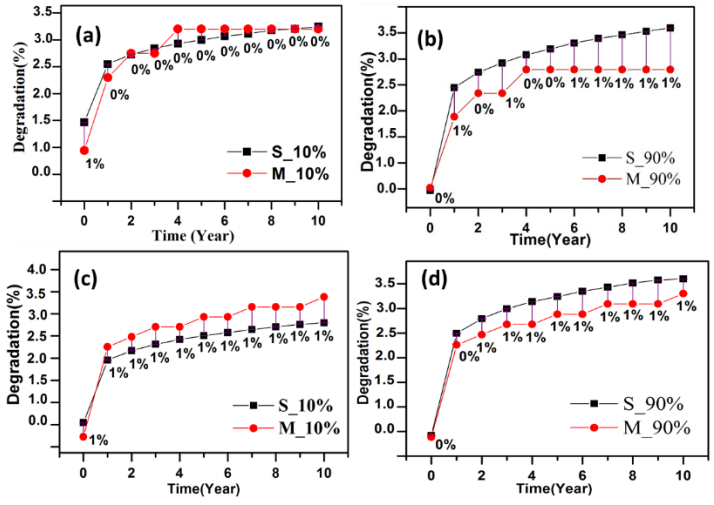


Figure 32: Delay Degradation comparison between our proposed ML algorithm and SPICE simulation for NAND and NOR gates

These results validate our Machine Learning Framework's capability to reliably predict the aging of the reference NAND and NOR gates.

Pearson's Correlation:

It was developed by Karl Pearson. It is a method to measure the statistical relationship between two continuous variables. It is also called as "Pearson's r" or "Bivariate correlation" or "Peasrson product-moment correlation coefficient". The resultant value or the correlation coefficient range is in between -1 and 1.

Table 2 shows the Gate and its corresponding Pearson's correlation between SPICE and our modelled data. There is a positive correlation between two data points. These correlations are far away from 0 value. Thus, the relationship between two data points are stronger.

Gate_Duty Cycle	Correlation
NAND_10%	0.9645
NAND_90%	0.9818
NOR_10%	0.9958
NOR_90%	0.9952

Table 2: Pearson's Correlation table

C. 2 Activity Aware Aging for Generic Gates

As introduced in chapter 3 B.2, Logical Effort conversion is an effective way to convert delay data from an Inverter gate to other gates. During the digital design phase, the standard cell library files (FDSOI28nm technology) are available to the designer, providing both the input and hidden output capacitances needed by the method and the gate's topology. We can deduce the logical and electrical effort for each gate from these library files. Therefore, we apply equation (11) to get the individual gate delay and finally compute each gate's aging delay separately. Thus the ground data is extracted.

The results reported for NOR/NAND gates are trained and kept as historical data. These historical data are trained along with each gate's ground data to predict other gates. By combining these two steps, we are finally able to obtain Aging Delay for any given gate. Figure 23 plots the degradation delay for 10 years for ten different gates: INV, NAND, NAND4AB (4-input NAND), AND, FA (Full Adder), OAI (OR-AND-Invert), NOR, NOR3 (3-input NOR), NOR2A (2-input NOR with one input as Inverted), and XOR2. The delay degradation structure looks quite similar, with the degradation of around 0 to 5%.

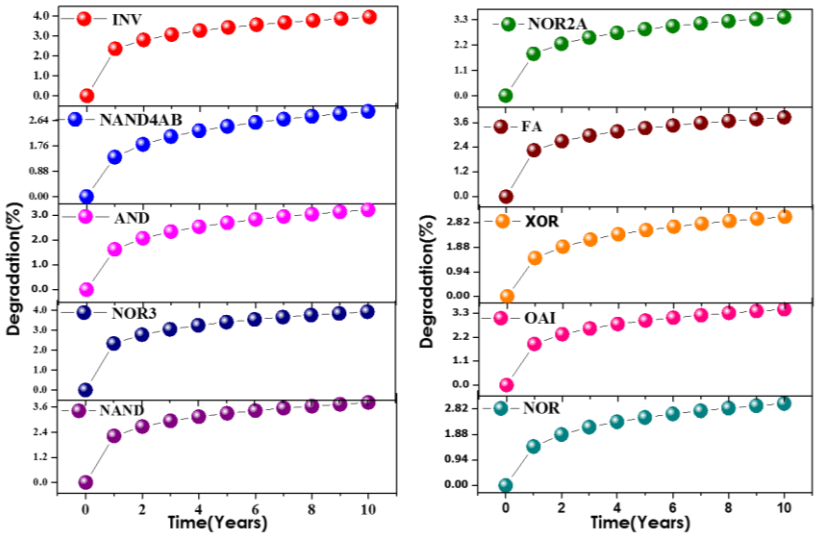


Figure 33: An aging-induced degradation prediction model for several standard cell gates

The results depicted in Figure 33 demonstrate our capability of extending the prediction to generic gates in our target technology. Even though we do not have reference results for all these gates, they are obtained by combining the ML prediction framework validated and the Logical Effort approach proved in literature [14]. This allows us to obtain reasonable predictions even in the absence of full foundry data.

C.3 Switching Activity Extraction for Complex Designs

The Switching Activity of each gate and net is comprised of two parameters: Static probability and Transition rate [82]. The signal's expected state is referred to as static probability and the number of transitions per unit time is called transition rate or toggle rate. There will be two transitions with each cycle one is rising and the other is falling signal.

We simulated our designs using Siemens (formally Mentor) Modelsim simulator and recorded signal activity as a VCD (Value Change Dump) file for this experiment. We then applied Synopsys's back-end flow by first extracting Switching Activity from the VCD file into a SAIF (Switching Activity Interchange Format) file. Primetime PX then merges these measurements with Synthesis information from Design Compiler to compute Static Activity and Transition Rates for all observed Nodes. Figure 34 depicts the full Synopsys flow: the only difference is the usage of VCS instead of Modelsim for simulation. The obtained VCD file, illustrated in Figure 35, is the same regardless of the chosen RTL simulator.

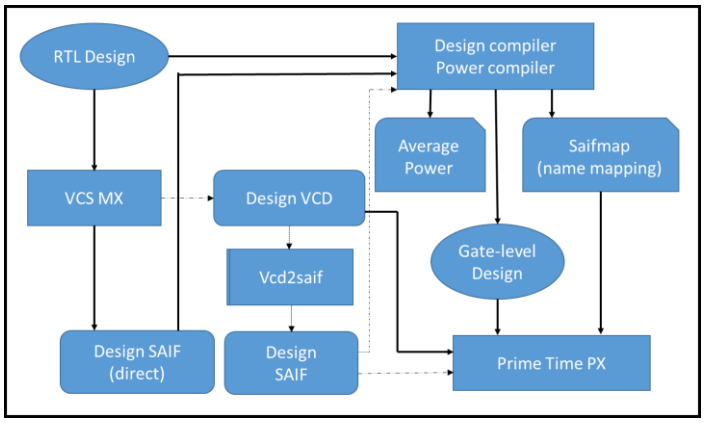


Figure 34: Full Synopsys flow [83]

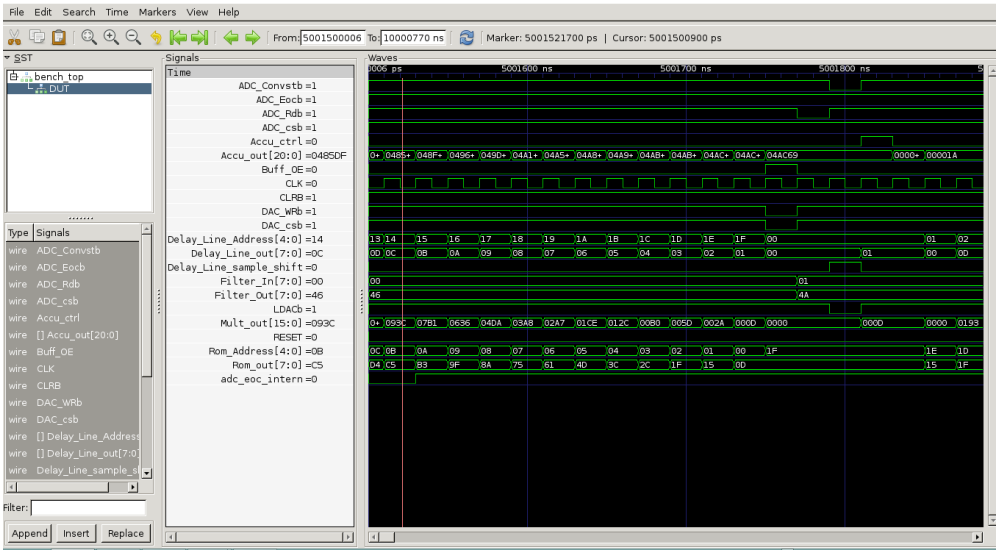


Figure 35: The FIR Filter VCD file waveform

C.4 Test Case 1: FIR Filter

To validate the workflow of our proposed Machine learning algorithm we first applied it to a simple design, a Finite Impulse Response (FIR) filter. It is a pretty standard system, as shown in its Architecture diagram in Figure 36. The RTL (Register Transfer Level) and schematics are depicted in Figure 37 and Figure 38.

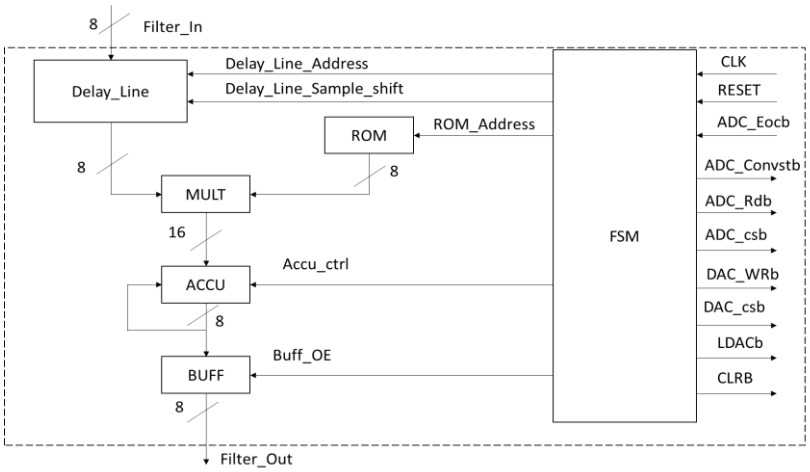


Figure 36: Architecture of the FIR filter

Figure 37: RTL (Top-level) view of FIR Filter

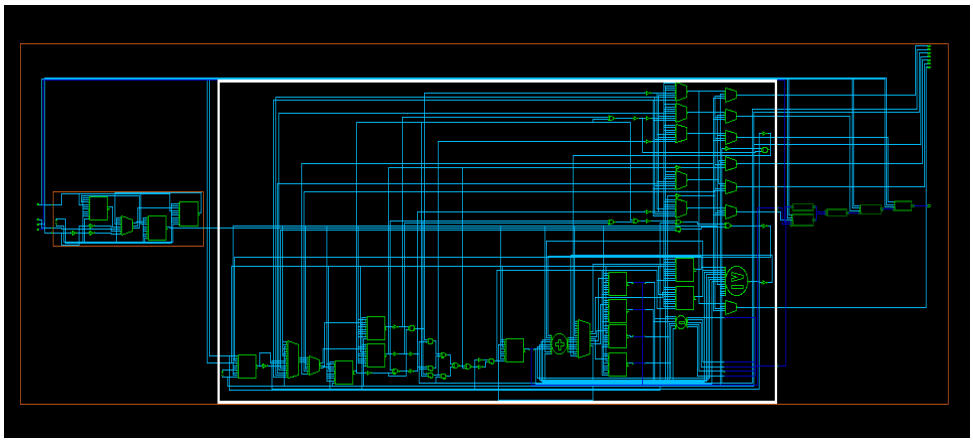


Figure 38: Schematic view of FIR Filter

C.4.1 Experimental results for FIR Filter

For the FIR filter, we applied our ML prediction framework to estimate the aging of its 20 Near-Critical Paths. Figure 29 depicts the delay degradation for the Critical Path over time (x-axis), with a stack bar chart highlighting the contribution of each gate. Stacked bar charts help to notice changes at the gate level that are likely to have the most influence on individual CPs. This chart helps to compare the total delay and notice sharp changes at the gate delay level that are likely to have the most influence on toggling activity. In this case, we can see how the aging is uniformly distributed among the gates: this is because for the FIR filter, the Critical Path resides in the Multiplier/Accumulator (labeled MULT and ACC in Figure 26) on the same data path. Therefore there are no big changes in activity among the gates.

Figure 30 depicts the aging in terms of Degradation for the 21 Near-Critical Paths, with CP1 being the most Critical. We can observe that the Most Critical Paths are also the ones that age faster: one more this is easily explained by an architectural analysis: each one of the NCPs is related to a bit in the Accumulator, CP1 being the Less Significant Bit. Of course, LSBs are more active than MSB, so it is reasonable that aging is correlated with the CP ranking.

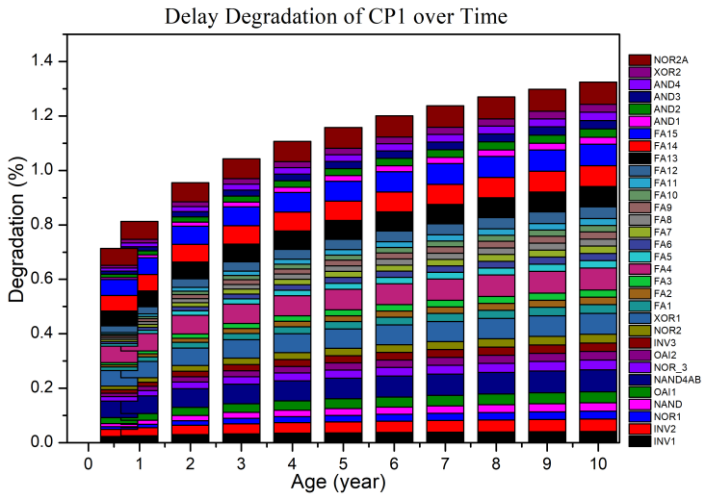


Figure 39: FIR filter: Stacked bar graph for CP1

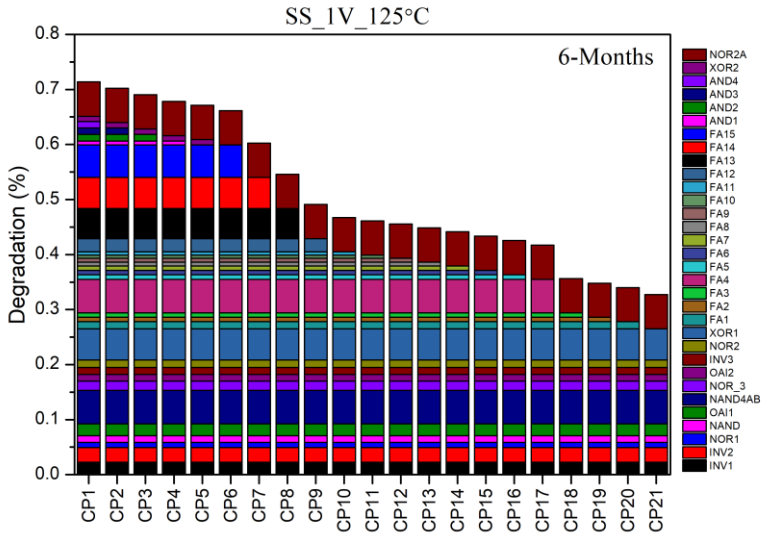


Figure 40: FIR filter: Stacked bar graph for different CP

These results validate our setup's capability to produce good results which, thanks to the FIR filter's simplicity, we were able to explain through an architectural analysis.

C.5 Test Case 2: AES Circuit

The first FIR circuit design was chosen because while it is complex enough to debug the Workflow, it is still simple enough to allow manual interpretation of its aging results.

As a complete use case, we selected an AES (Advanced Encryption Standard) crypt-processor performing a set of encryption and decryption operations. It is broadly used in wireless security, processor security, file encryption, etc. The architecture of the AES circuit is shown in Figure 41. The RTL view of the circuit in Figure 42. This particular implementation has been developed to validate the AES code while reducing pin count: a Decryption operation always follows an Encryption Operation, and the result is compared with the original data and only the comparison results are presented as outputs. The AES circuit is interesting not only for the above-mentioned practical applications but also because it is fairly complex with a lot of balanced Near-Critical Paths whose aging is extremely difficult to predict using conventional means.

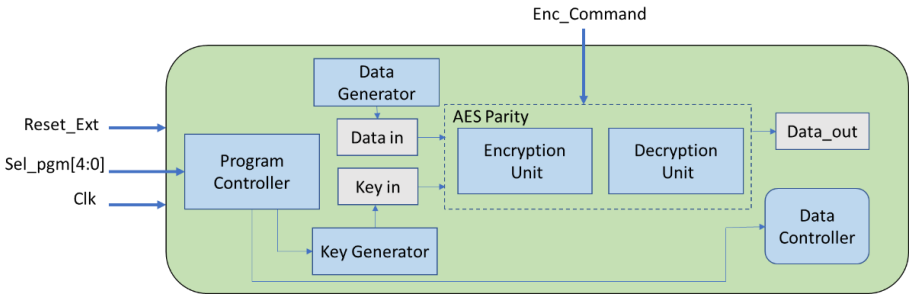


Figure 41: Architecture of the AES Circuit

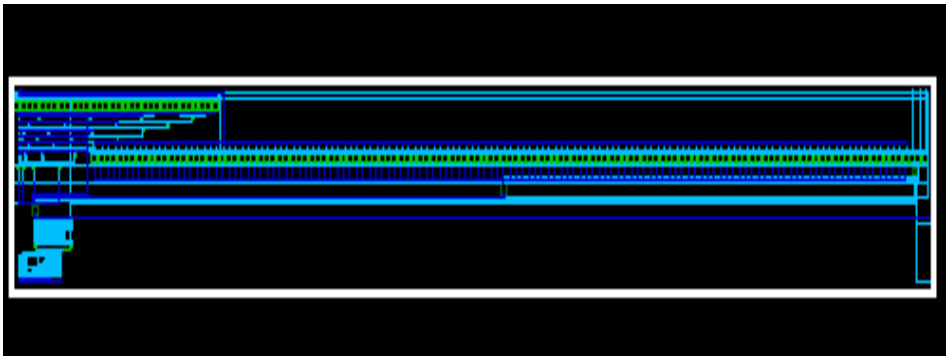


Figure 42: RTL view of the AES circuit

C.5.1 Experimental results for AES Circuit

Figure 43 and 44 shows the AES circuit aging variation for the 150 worst-case critical paths for 6-months and 1-year degradation for PVT (ss28_1.0V_125° C) bar graph. It exposes the percentage of activity of each gate inside each critical path that can be visualized clearly. From this graph, we can see that the NOR gate is more active than other gates, and the percentage rate of degradation also higher

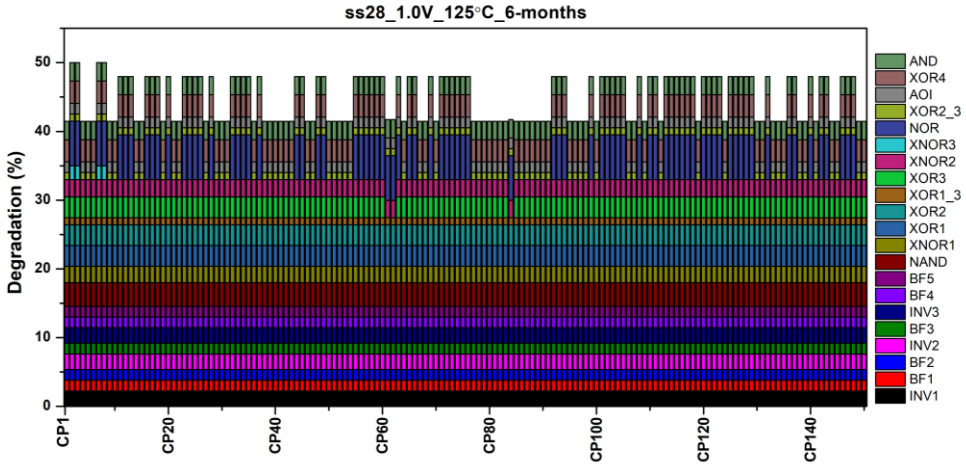


Figure 43: AES circuit critical path aging for PVT (ss28_1.0V_125C)

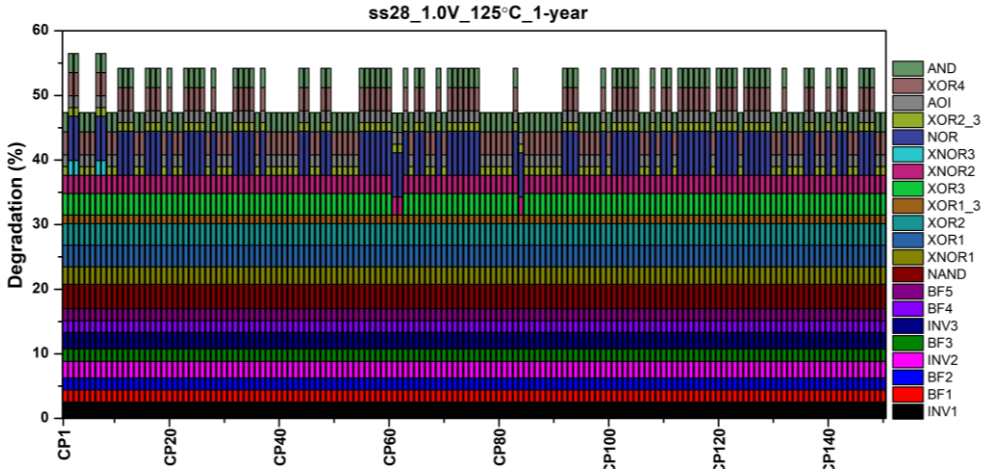


Figure 44: AES circuit critical path aging for PVT (ss28_1.0V_125C)

C.5.2 Activity-Aware Critical path Ranking Variation

CP ranking variations for test case 2: AES circuit is taken to rank 150 NCP for different PVT conditions. Ranking based on the delay of each critical path. Figure 44 shows the ranking variation for the fresh delay, 6 months and 1-year ranking variation for workload 1. The two workloads are given to the data in and key in of the circuit. They have been carefully selected to provoke uniform and extreme functionality of AES circuit. The Data_in, Key_in, and their corresponding values are given below.

Data_in1 <= X"3243f6a8885a308d313198a2e0370734";

Data_in2 <= X"00112233445566778899aabbccddeeff";

Key_in1 <= X"2b7e151628aed2a6abf7158809cf4f3c";

Key_in2 <= X"000102030405060708090a0b0c0d0e0f";

The following observations are noted.

The X-axis shows the 150 NCP, and the y-axis stands for the ranking. The first row of the graph stands for fresh delay variation. It is almost ranking linearly. The second row of the graph showing ranking variation for 6 months. The third row of the graph showing ranking variation for 1 year. We ranked the CPs based on the delay variation from fresh delay to age 6 months and 1 year. All the three graph shows different ranking it is because of the PVTA variation. The red color bars are marked to show up the top 10 highest delay among 150 NCP. It is interesting to see how PVT profiles and toggling activity of each gate have strongly impacted

aging, resulting even in raking inversions of the critical path concerning time faults are more prone to occur on NCPs with strong DC activity than on less active CPs identified by a static timing analysis.

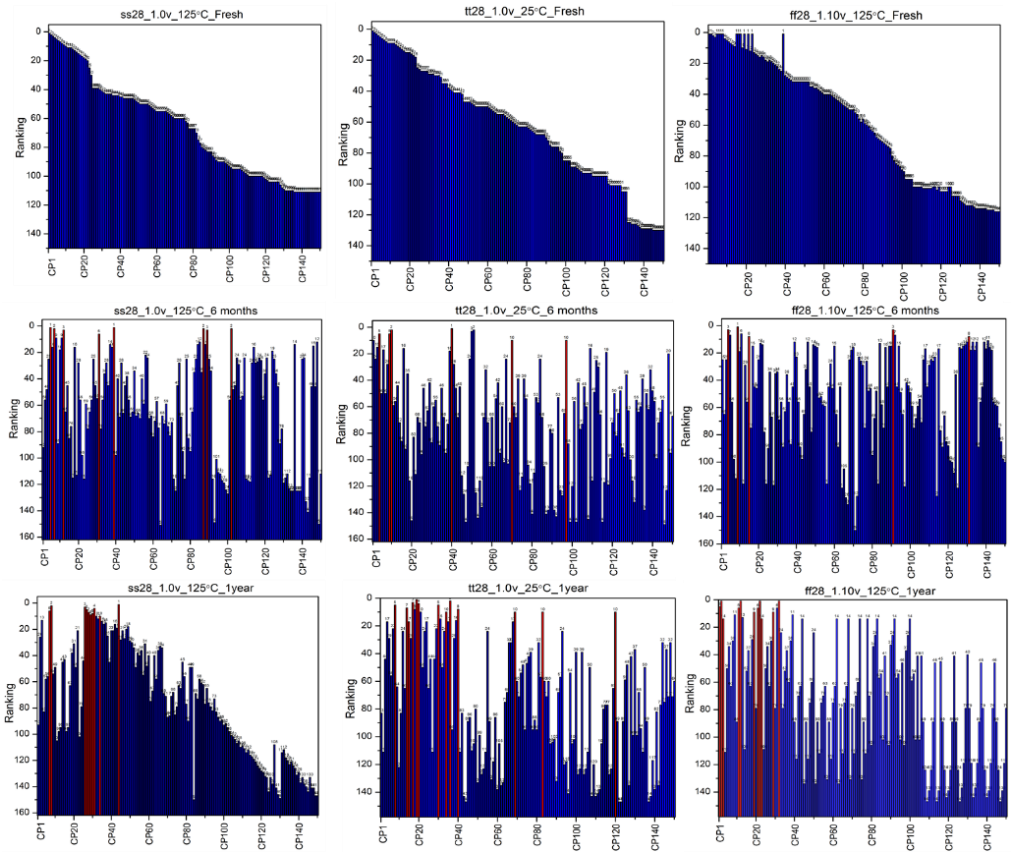


Figure 45: Ranking Variation for Workload 1

Each Graph looks different because the voltage and temperature variation impact the chart showing many variations in ranks. Figure 45 shows the ranking variation for workload 2 given to the data in and key in the circuit, which is different from workload 1. The following observations are noted. The above-mentioned five points are applied to this workload 2. The graph looks different from figure 44 because the workload variation affects the ranking of the NCPs. Even if we change the workloads, we can notice that only a subset of NCPs are essential to monitor.

Therefore, our method can identify the NCPs that will be more critical during the circuit lifetime and take counter-measures early in the design phase to alter the PVT profiles' changes. We can detect the most important NCP to fix the monitor before ahead. This proposed methodology is technology-dependent.

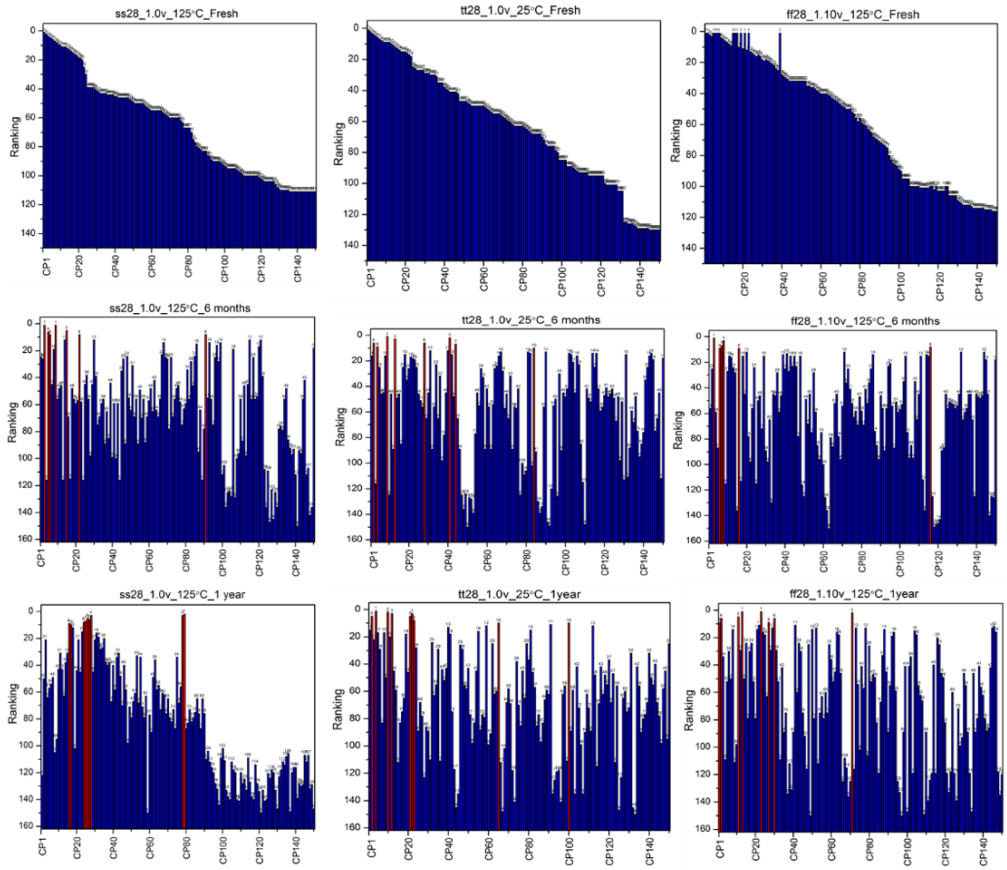


Figure 46: Ranking Variation for Workload 2

As NCPs age differently and their absolute difference is small, Ranking Inversions might happen, i.e., paths that were not critical at Time 0 can become problematic after some time due to aging. This phenomenon reported in [84] by the authors of the paper is extremely difficult to observe and predict using traditional flows because it depends on both low-level physical phenomena and high-level setups such as the workload. The computational complexity of such simulations and their analysis is by itself a show-stopper. On the other hand, our ML framework is extremely lightweight and we proved its ability to efficiently predict Path Aging depending on aging time. As the ordering of paths on the X-axis is unchanged, it is the effect of workload on path aging is clear. Path distribution is almost chaotic: each gate aged depending on its activity and the aging profile is extremely different between the two workloads.

This observed phenomenon is one of the greatest drawbacks of Monitor Insertion flow. In each aged distribution, we highlighted in red the 10 Most Critical Paths on which Aging Monitors should be inserted. From this figure, it is obvious to understand that a choice made at Time 0 based on STA evaluation would not be coherent with an Aged system, making most inserted monitors unnecessary.

D.Alternative MVL Algorithms: Random Forest

D.1 Introduction

Many algorithms have been developed for prediction problems in machine learning. One of the predictive machine learning algorithms is the random forest (RF) algorithm. RF algorithm was developed by Tin Kam Ho [85] in 1995. RF is a supervised learning algorithm used for classification, prediction, and regression problems[86]. It is also known to be a tree-based machine learning algorithm. The trees in random forests are in parallel to run. It trained multiple decision trees and combined them to acquire more precise and steady predictions. One of the advantages is that it can be used in both prediction and regression problems. Let's look at the random forest in regression and prediction view. While expanding the tree in the random forest increases the model unpredictability.

The remaining sections define the random forest algorithm framework and metric evaluation results compared to multiple linear regression algorithms. The advantage of our model is concluded with one of the algorithms at the end.

D.2 Random Forest Algorithm

RF algorithm is one of the most used and supervised learning algorithms for prediction. Therefore, we analyzed the impact of our model prediction provided by the same digital circuit on the RF algorithm and compared it with a multiple linear regression algorithm. This thesis's proposed circuit-level model can be used either on-line and off-line to estimate the digital circuit degradation effects.

It is a meta-estimator that combines the results of multiple predictions. Multiple decision trees are constructed during the training of random forests. RF is great and precise. It also solves many problems, including linear and non-linear features. The issue related to our data is linear to time. Thus, the linear random forest method is applied here to see the prediction accuracy. Finally, the RMSE of our model is compared with a random forest algorithm and multiple linear regression prediction to know which algorithm is best for prediction.

We aim to predict the delay of the critical path in the digital circuit based on the PVTA of our model. It's a regression problem to solve this. The RF algorithm is used via the scikit-learn python library and machine learning pipeline. The dataset is collected and trained to get the results. The error metrics for regression RMSE (Root mean square error), MSE (Mean squared Error), and Mean Absolute error (RBE) are used to evaluate the RF regression model. The error value is lower or higher value defines the accuracy of the model. The lower the value, the better is our model.

D.3 Advantage and disadvantage of using RF algorithm

There are a few advantages and disadvantages of using the RF algorithm that are discussed here. Each tree inside the bunch of trees is trained on a subset of data.

Thus, therefore, the overall efficiency of the RF algorithm is not biased. It's a stable algorithm even if new datasets are added. It does not affect the impact of the algorithm. Numerical features are work well for the RF algorithm. There will no problem when data has missing values or it had not been scaled well. The complexity of the algorithm is a major disadvantage. It requires much more time to train due to their complexity.

D.4 Comparison with Linear Regression

The metric evaluation of each complex gate is calibrated and validated with multiple linear regression algorithms and random forest algorithms. Figure 47 shows the bar chart plotted between different complex gates inside our targeted complex gates and our model's error rate. For metric evaluation, both algorithms are proposed with our model. bar chart shows the RMSE, MSE, and MAE of each complex gate separately and makes a comparison between the two algorithms. It can be seen that the overall error rate for Multiple linear regression is smaller than the random forest algorithm. The results demonstrate that our method can determine the relative performance of the LR algorithm with high accuracy compared with the RF algorithm.

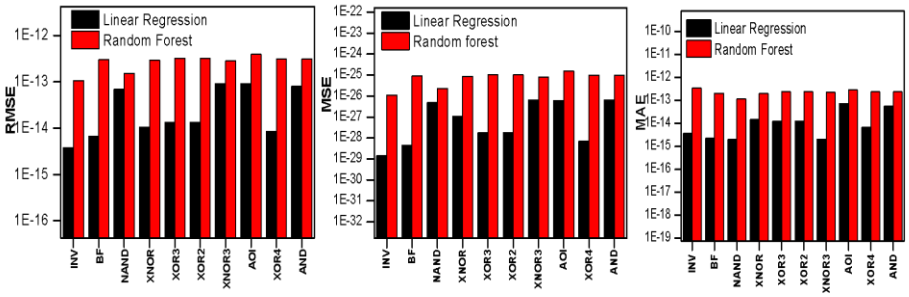


Figure 47: Evaluation of machine learning algorithms for complex gates

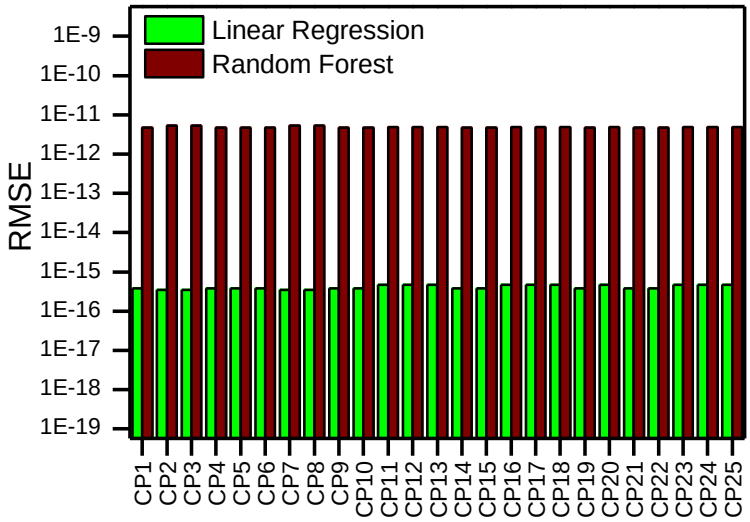


Figure 48: Evaluation of machine learning algorithms for Critical path in FIR Filter

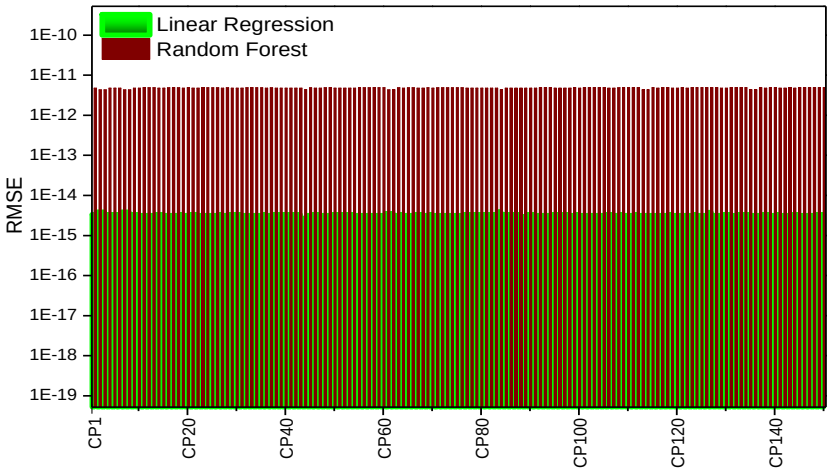


Figure 49: Evaluation of machine learning algorithm for Critical path in AES circuit

Figure 48 and 49 shows the evaluation of two different machine learning algorithm for FIR filter and AES circuit critical path analysis and comparison. Linear Regression, despite being simpler, performs systematically better than Random Forest in terms of RMSE for in all test cases, from simple gates to a complex target like AES. This proves that in the context of our prediction framework the impact of the ML algorithms itself is low, and validates our decision to select Linear Regression.

F. Conclusion

In this chapter, we proposed an ML algorithm that includes a mathematical model to predict the aging of the two different digital circuits, and applied it to circuit level simulation results for different operating points. Then, we evaluated the results with SPICE simulation and compared the error differences. Finally, we compared the results obtained with the simple Linear Regression Model with the more complex ML algorithm Random forest algorithm, demonstrating that the results are independent from the chosen ML approach.. This validates out choice of the Multiple Linear Regression algorithm as a simple yet effective method for our ground data.

Chapter IV: System-Level Strategies and Applications

A.Introduction

This chapter demonstrates the use of our model on a system-level application. In recent years, many works focus on the PVTA issues in the digital circuit. Some of them [12][53][10] are analyzing the degradation of the circuit during the run time. The reliability of the circuit depends mainly on its PVTA of the IC design. Usually, the chip is designed to operate under a different constraint even under worst-case processes, voltage, and temperature conditions. The peak PVTA computation, in combination with the core, memories, architecture, and interconnections, decides the digital circuit's operating frequency. The total number of gates and workload finally determines the maximum frequency and voltage of the circuit.

Chapter 3 gave an explanation about the proposed state of the art of our work implemented in two different digital circuits and evaluated results. In this chapter, the built model was used and implemented in low-power CMOS design techniques such as dynamic voltage and frequency scaling. This work introduces a design scheme that improves critical path energy optimization for a fixed performance and maximum performance optimization of the digital circuit. To achieve this scheme, we developed an algorithm based on our mathematical model and linear regression algorithm. The digital circuit voltage levels include a significant margin to deal with the worst-case critical path process variation, temperature variation, workload induced voltage variation provided with an energy-saving performance.

It also generalizes the worst-case critical path energy optimization scheme by providing details of the methodology, prior comparison, Fmax (maximum frequency) tracking accuracy, and power-saving realized by this method in 28nm technology. We also report PVTA with BTI & HCI aging, a novel approach to actively calibrate the platform energy optimization unique to each digital circuit. The proposed algorithm is used to determine the optimal voltage needed to support the frequency under operating conditions. These optimization techniques, which are applied offline and check during the circuit's run-time, are possible.

This chapter proposes and demonstrates some cases of application of the methodology presented in chapter 3. The proposed algorithm models are used for both off-line and on-line estimating the circuit degradation. This model can perform reliability simulations of complex gates and critical paths in a digital circuit. However, few paths under different operating conditions are possible due

to the required simulation time. Section B and C discuss the critical path selection and their insertion flow inside our target designs. Section D explains the novel dynamic voltage and frequency to track the maximum operating frequency. The circuit lifetime can be calculated, taking into account the actual condition of operation. The novel method of Maximum optimization algorithm and capped performance optimization algorithm using our model in detail.

B. Workload-dependent NCP selection

As highlighted in the previous Section, the key point for ETI Monitor insertion is the correct selection of the subset of NCPs to modify. However, the results of Section III and most notably in Figures 44 and 45 demonstrate that NCP ranking changes over time based on the actual workload: a selection made at time 0 exclusively from STA ranking as presented in Figures 44 and 45 might be suboptimal: paths that were deemed critical might age slower and therefore instrumenting them would be useless. On the other hand, some paths that were not instrumented might age faster and become critical and potentially lead to timing errors.

In this section we applied our ML approach to age the target AES circuit under two different workload, and selected the 10% NCP at different times. We then chose to represent these subset in a Venn diagram form. A Venn diagram is used to show the relationship between two different things or finite groups of things. Circles that overlap have a commonality, while circles that do not overlap do not share those traits. It is a great way to visualize informative comparisons between data sets. .

Figure 50 shows the Venn diagram for fresh, 3 years, 6 years, and 10 years of predicted degradation of the AES circuit under two different workloads. Each circle from the figure represent the 10% NCPs at a given aging time. In both cases, out of 63 NCPs selected at Time 0, only a subset of roughly 70% will still be critical regardless of aging. The remaining 30% changes based on both aging time and workload, and will be missed by traditional STA-based path selection.

Another interesting result is that depending on the aging time, and therefore the expected system lifetime the NCP subset changes: for instance, a system aiming for a 3-year mission will have to monitor a different set of Paths than a system aiming for 10-year lifetime.

Modern technologies are more sensitive to PVTA variations. Accurate and inexpensive performance monitoring for different variability is difficult to do. There are no standard solutions for finding the critical CPs for two different workloads. This Venn diagram will help the designer decide which CP to concentrate on determining the essential CPs to monitor the global and local variations. This type of Venn diagram approach can resolve the critical path

selection method. It is a simple and effective solution to the 28-nanometer technology scaled system on a chip.

Figure 50: NCP subset evolution over time for two workloads

C. Aging-Aware Adaptation of Operating Performance Points

In the era of large-scale integrated digital circuits in a single system-on-chip and fabricated in continuously shrinking nanometer technology nodes. It is essential to test the circuit and ensure it is operating in a fault-free system or not. Today there will be a vast amount of cost involved in testing semiconductors, and designers facing many complex problems and challenges. In the meantime, advanced and innovative techniques are needed to manage the new failure mechanisms under PVTA constraints. The absolute value of NCPs is more important as mentioned in the before section. The delay of worst-case NCPs determines the Operating Performance Point (OPP) of a system.

OPPs are usually represented as a couplet of Voltage/Frequency and play a major role in both defining the System performances and avoiding delay faults during circuit runtime. To simplify, the higher the Frequency, the more performing a system will be, but at the same time it will be more exposed to Delay Faults. Raising the Voltage will make transistor switch faster and therefore reduce the system's susceptibility to Delay faults, but it will boost its power consumption.

Delay faults occur when the propagation delay exceeds the working frequency period. ETI inserted at the end of the CPs will raise a flag when such faults occur.

Thus, it is necessary for the circuit to maintain the flag count to zero. Unluckily, it is not only dependent on the OPP but also the PVTA conditions. In order to consider all the conditions of PVTA and flag rise of 150 CPs are considered using our model to be monitored in this work.

This work focuses on the characterization methodologies that will help detect the accuracy and resolution of issues that may arise due to voltage failure. We extracted the fresh delay from our targeted AES circuit and used the Design Vision tool to compare it with our model. We set up the PVT (SS_1V_25^{o}C) clock constraint for 500 MHz and simulated to get the 150 near worst-case critical path delays. Further, the delays are added by 70 ps to reach the failure of the circuit [77]. The frequency range is bounded for different process corners. We identified the frequency making the first flag rise depending on the process : the results are list in Table 2. At the end we compared these results with STMicroelectronics SPICE simulation and the error difference is nearly 1% [77].

Process	Frequency (MHz)
SS (slow-slow)	694.4444
FF (fast-fast)	909.0909
TT (typical-typical)	833.3333

Table 3: Process Vs. Frequency

After validating our model's fresh delay, we aged our circuit to 12 years to observe frequency shift. Figure 51 shows the AES circuit's flag characterization for all three processes, 1 volt, 25^{o}C temperature for a fixed frequency of 500 MHz. The graph is plotted in-between frequency and flag count of 150 near the worst-case critical path of the circuit with our mathematical come trained linear regression model. The impact of corners is observed easily on this graph. The variation in fresh and aged frequency drift explains the importance of PVTA. Thus, the aging critical path of a circuit can affect the frequency directly and the circuit's performance. The frequency drift between fresh and aged is very near to the failure of the circuit.

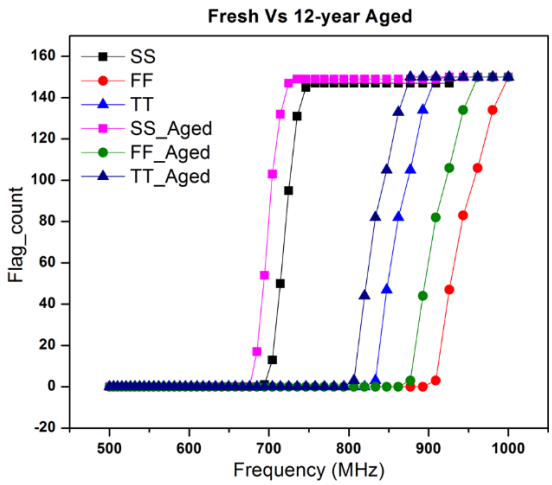


Figure 51: Frequency Vs. Flag_count of all process for fresh and aged critical path

Figures 52 and 53 show the impact of Operating Voltage on the frequency shift for a fixed process (slow-slow and typical-typical) respectively. As expected, raising the Operating Voltage is an effective way to reduce the flag count (and therefore the timing violations) for an aged circuit while maintaining the same frequency. The older the circuits, the higher the voltage.

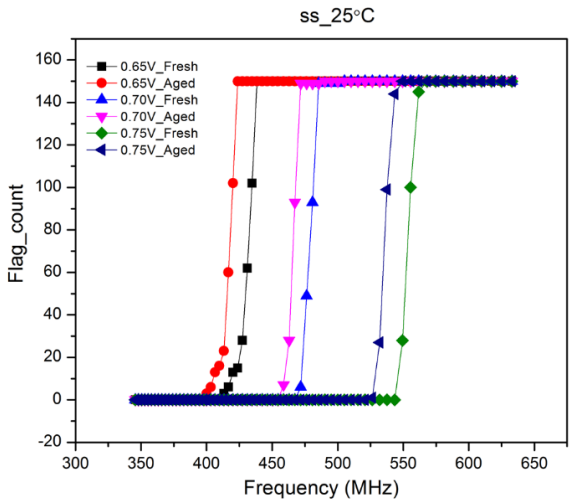


Figure 52: Shift in the frequency of different voltages for the slow process

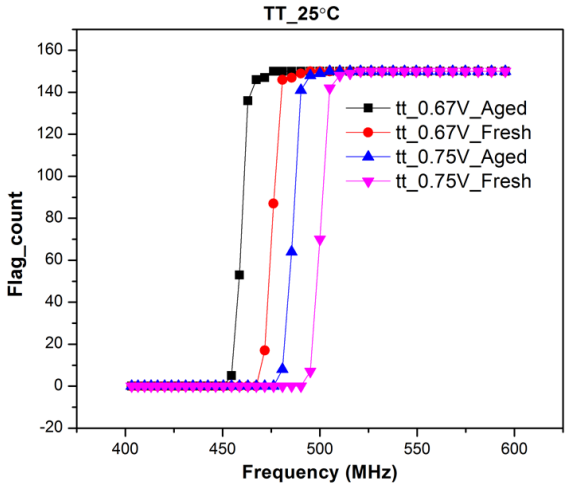


Figure 53: Shift in the frequency of different voltages for typical process

in a 12–years AES circuit, the flag count depends on the voltage. For instance, in reference to Figure 52,to assure the circuit to be operational at 440 MHz, the designer will be forced to choose the least voltage of 0.75 V to guarantee zero flag count until 12 years aged circuit. However, the Fresh circuit might have worked at the lower 0,67V voltage, with a significant power consumption gain over 12 years.

The role of slack is important in the real setup of a circuit working flow. OPP selection is a critical criterion, as shown in figures 52 and 53. Simultaneously, the presence of negative slack (or in our case, a positive flag count) indicates that the design cannot operate at the specified clock frequency. In converse, a zero flag count indicates that the design can operate at the predetermined frequency and further [89].

In this work, we propose to apply our prediction framework to find the OPP based on the concept of Time Window (TW). The idea is to predict the effect of a given OPP periodically and react immediately to know the circuit's failure range before redesigning. At any give time t, we predict the flags count at time t+TW: if Flag Count (t+TW)=0, the corresponding OPP is viable will be used by shifting the voltage range. If the Flag Count is positive, we will explore a more relaxed OPP by either lowering the frequency or raising the voltage. Once we selected a new OPP, we move the analysis to the next time window (t+TW) and repeat this process until a the desired age limit. For each step, a new prediction step is processed by our model. It is a lightweight model to be executed on an embedded processor to allow periodic online prediction and further adaptation.

D.Power Minimization for fixed frequency OPP

In this section, we are targeting the power optimization of a circuit. Several digital electronics applications demand minimum energy/power consumption for a given performance (i.e. for a fixed Frequency). The demand for reduced energy/power

consumption with minimum voltage and temperature is not feasible for a more extended period while aging. Thus, special attention is given to the voltage of a circuit to achieve energy optimization. Dynamic voltage and frequency scaling with our model are proposed to accomplish the energy optimization for a 28nm FDSOI Technology. The entire region of the circuit operation depends on the transistor's threshold voltage and other factors. Performance estimation by designing energy-harvesting of a digital circuit-level must account for PVTA variation.

Traditionally, predicting a large-scale circuit behavior under PVTA and workload conditions requires much theoretical analysis, gate-level modeling, and logical verification. Especially when designing a digital circuit for a minimum operating point, the ultimate problem is predicting the circuit behavior at the critical path level. This proposed algorithm solves this issue. A complete workflow is presented in this thesis, along with methods used to improve operational reliability. We implemented the algorithm as shown in Figure 54. This algorithm handles many OOP of the circuit, such as voltage, temperature, and process, at a time. . The input is the fixed frequency (f), timing interval (TI), voltage (V), and observation time (OT). Then it enters into the loop and checks if the critical path delay that is flag count is zero or not. If the flag count is one, the current OPP is not viable because there will be a timing violation during the observation Time Window: the voltage is added 0.01 times and then enters the loop to simulate to check again for flag rise in the critical path. If the flag count is zero the OPP is validated for time t and the observation point will be moved forward by adding three months' age to every 150 near-critical paths. So, at each operating time we look at the circuit 3 months in the future and decide if we can keep the same voltage or increase it further.

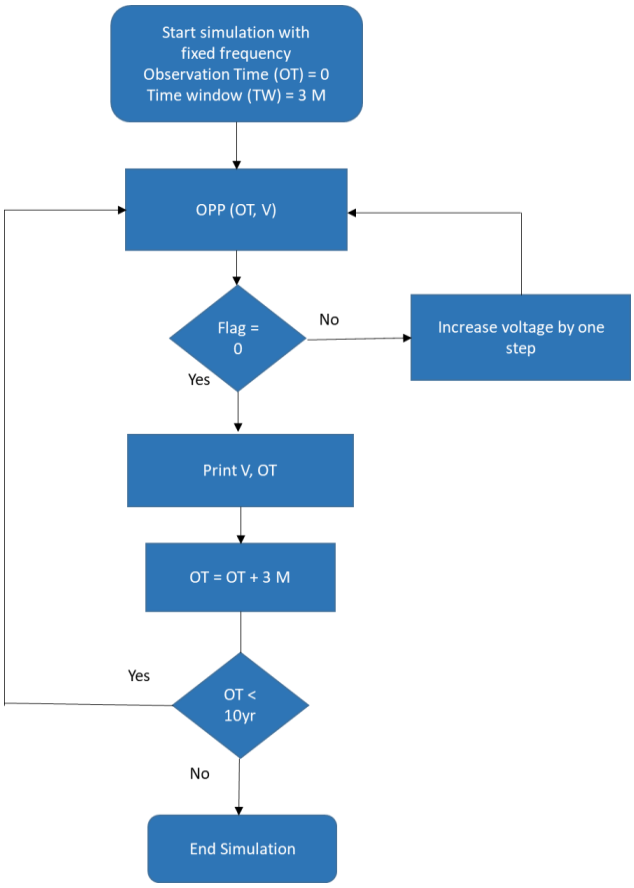


Figure 54: Flow chart for the fixed frequency algorithm

Figure 55 depicts the algorithm results for a 500 and 600 MHz fixed frequency for an SS (slow-slow) process. The black color squared dots represent 500 MHz frequency and its corresponding voltage we can operate at the corresponding time. At 0.73V with 2 years, 1 month, the flag rises, the voltage is a step up to 0.74V, and operates further. Again, there will be a flag rise at 7years 8months period, and the algorithm identifies the flag and the voltage stepped to 0.75V to perform the critical path without any flag rise.

In the same way, 600 MHz frequency was also simulated to get the results. Thus from the two frequencies, we can note that the circuit we can operate without a flag is only at 0.75 for 500 MHz and 0.77V for 600 MHz. We can see that for lower frequencies the circuit ages slower, so we can maintain a more efficient OPP longer. Finally, we compared our results with STMicroelectronics [77] with the same circuit. On the other hand, our aim is to operate the circuit at lower voltages, with significant gains in terms of total power dissipation.

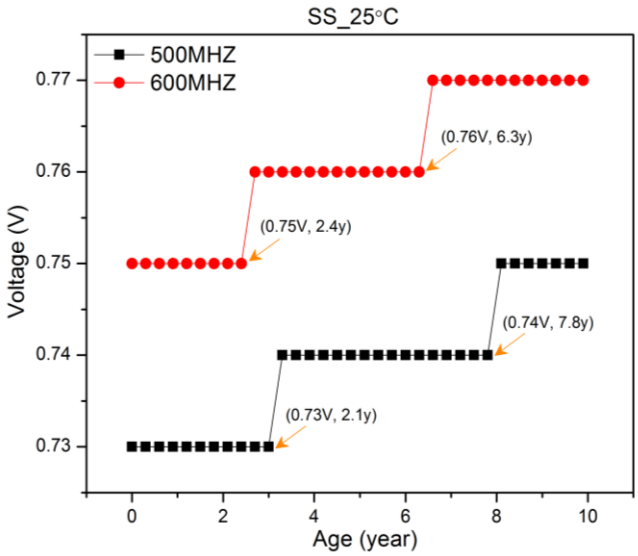


Figure 55: Fixed frequency graph for SS_25C

E. Fine-grain Timing Prediction

In the previous section, we chose a fixed frequency and time window. But while the circuit aging fast at the beginning while time passes, this process slows down. For the proposed fixed frequency algorithm, we used a specified time window of 3 months: this is too long for a fresh circuit aging really fast, but unreasonably small for an old circuit for which aging will be much slower As a result, we are losing potential optimization on the Fresh circuit by running our adaption too seldom, and wasting computational resources on an aged circuit by running it too often. We therefore developed a fine-grain prediction refinement algorithm which uses a variable window. At the beginning, the time window is weeks instead of months. For every 3 prediction periods, the time increases of the prediction window of 1, 2, 3, 4…..,15 weeks, etc., to reach until 10 years of period. Here we start by predicting a brief period of 1 week and then gradually increasing our prediction window to 15 weeks. That is the TW is extremely small at the beginning and gradually increases further up to 5/6 months. Figure 56 shows the graph for a fine-grain prediction graph for a fixed frequency of 500 and 600 MHz. Figure 57 shows the fine-grain prediction graph with a time window in months. We can see the time window precisely and accurately with our proposed fixed frequency algorithm from the above two charts.

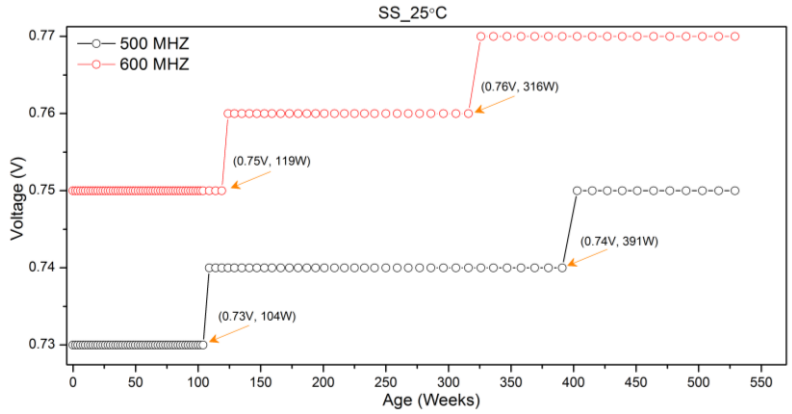


Figure 56: Fine-grain prediction graph with a time window in weeks

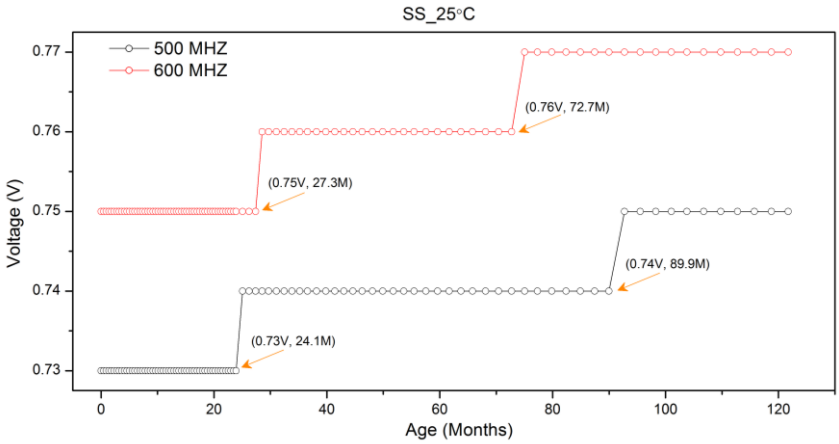


Figure 57: Fine-grain prediction graph with a time window in months

F.Aging Aware OPP Overclocking

The fine grain prediction of our work allows for innovative optimization strategies thanks to its fast reaction time. With the help of our proposed model, it is possible to dynamically select OPP and change them before any fault arises. A possible application is overclocking.. Manually accelerating the high performance of the digital circuit with specified limits to achieve faster execution time is called overclocking.. Thus, the interesting factor with overclocking to know the maximum frequency range of a circuit. The clock frequency is limited by the longest path in the circuit with a given OPP (PVT) conditions. Here, we consider the worst-case critical path for performance improvement that over clockers activity. During the runtime of the circuit, there will be a high probability of circuit failure is possible. The impact of on-chip to find the overclocking reliability and considering the timing errors to address the faster circuit [90][91].

The design of worst-case, along with processor performance with the help of overclocking, is possible. To force the circuit within a frequency limit helps to detect and recover from timing errors. The overclocked on-chip circuit exposes a large amount of heat and increases power consumption. It reduces the system's lifetime. The highest performance of the digital circuit can be achieved through the optimum voltage and frequency range limit of the design. Solving this optimization problem becomes one of the challenges in the semiconductor IC industry. It is targeted to every digital design to have the minimum energy delay in achieving maximum optimization. The fine-grain prediction approach which lead us for innovative optimization strategies. Dynamically it is possible to change voltage and frequency before any delay fault arises. Aging aware overclocking of any OPP is possible by setting high frequency and high voltage, with significant performance boosts. Regrettably, aging affects the working condition and controlling probability and these solutions it is applied rarely.

Based on our prediction framework, we can define an overclocking OPP selection: at any time window, choosing the OPP with the highest possible working frequency. Figure 58 shows the plotted results. This optimization strategy was run on AES synthetized for 2GHZ/1V for almost 4 years (200 weeks). We can observe from the figure that there is a stress on the system and the curve on aging is quite steep, thus to avoid delay errors, we forced to lower both the frequency and voltage in the second half. A model was tuned by adjusting voltage and frequency. The temperature and process are fixed. This algorithm is tuned in a typical high-performance energy-delay optimization. The propagation delay of 150 worst-case critical paths is tested with our algorithm to get the results. Here the influence of voltage and degradation of critical paths are highly noted. It is purposefully extreme and it is useful for energy performance with a short and long lifetime to perform the tasks and workloads.

The traditional way of detecting voltage and frequency for a digital circuit is quite hard and time-consuming. Thus, our approach will help predict it in a simple way.

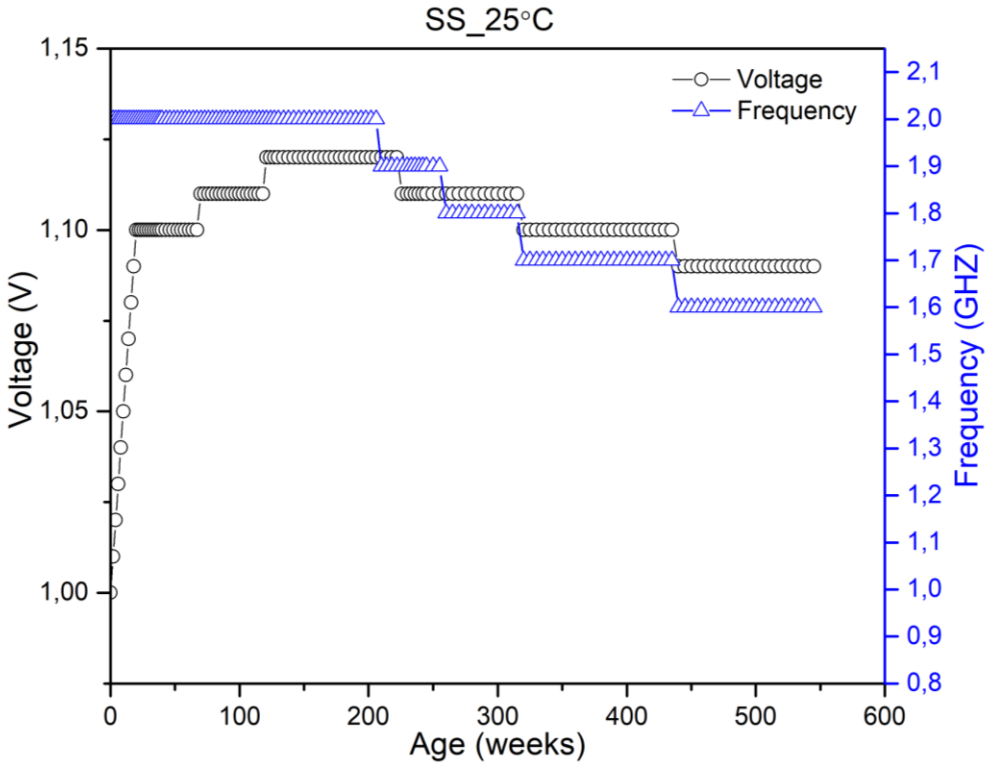


Figure 58: Maximum performance optimization strategy algorithm

G.Maximum Performance with OPP Cap

In the previous Section, we showed how it is possible to overlock our AES circuit with a significant performance gain, but with the side effect of a great stress and an accelerated aging in the second part of its lifetime. However, in most application cases there is no need for such an aggressive approach : instead of aiming for the absolute maximal performances, the optimization can be “capped” to stop at a pre-defined performance level which satisfy the system’s specifications Figure 59 shows the OPP optimization with a cap at 1V/1.5 GHz, where the voltage is stepping from 0.1V. The regulation is performed to maintain the same voltage without any flags. Thus it reaches 1.6V, where a flag arises is highlighted in the red bar, and the voltage is stepped down to 1.5V for some weeks. In order to maintain the minimum voltage, the frequency is adjusted to the next frequency level. Here both the voltage and frequency are adjusted to find the maximum performance of the circuit within the pre-defined cap. The performance of different OPP in terms of PVTA results in saving energy-optimization of the design.

.

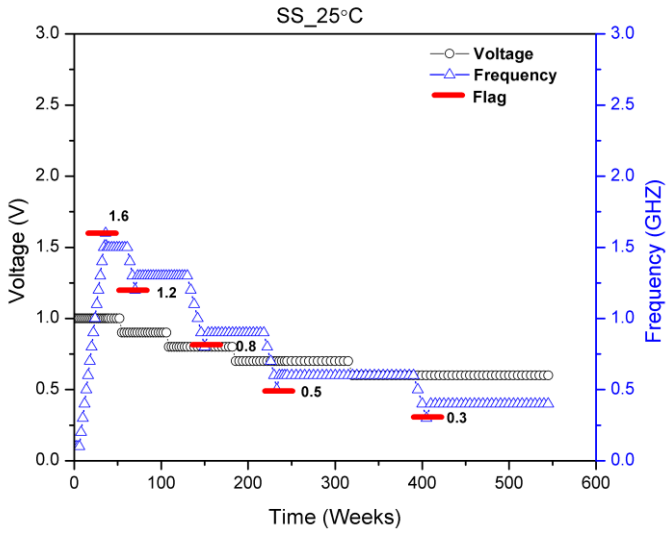


Figure 59: OPP optimization capped at 1V/1.5 GHz

Here, we proposed a novel optimization methodology, which predicts the voltage and frequency effectively. These proposed models can also be used for on-line estimation of the circuit degradation. Besides, it is also used to perform reliability simulations of complex systems. Through our simulation results, we could conclude that it could be compared with traditional DVFS strategies with an error rate of 1 percentage difference. It is one of the strategies, which perform performance, energy, and reliability test. The traditional system on the chip requires a monitor or time sensor to estimate the performance accurately. At the same time, our proposed model needs the only simulation to monitor the circuit effectively with a minimum error rate.

H. Conclusion

This chapter proposes and demonstrates some cases of system-Level application of the proposed methodology presented in chapter 3. First, the proposed models are used for off-line estimating the circuit degradation. They can be used in a dynamic system to track the maximum operating frequency and minimum voltage. The estimation was taken into account the actual conditions of operation. To avoid the failure of the circuit, the maximum operating conditions are predicted. It also is used for the on-line estimation of any digital circuit on 28nm FDSOI technology. We considered 150 NCP to predict the lifetime of a circuit under different OPP conditions. Here, two different strategies were implemented one is maximum performance optimization and the other one is optimization with capped performances.

Chapter V: Conclusion and Perspectives

The variation due to PVTA affects the advanced nanometric technology in terms of the reliability issue. The traditional safety margin approach is not a permanent solution to this problem due to higher design costs. Hence, the novel mathematical model along with the ML algorithm is used to solve the issues.

A thorough explanation of CMOS aging mechanisms of FDSOI technology is given in chapter 1The PVTA variation and its impact while implementing it on the digital circuit are explained. In addition to that, a recent approach of embedded and In-situ monitors with their performance violations are discussed.

The state-of-the-art review of the traditional ML approach and its types is explained in chapter 2. The ML algorithm is broadly categorized into supervised, unsupervised, semi-supervised, and reinforcement learning methods are described in detail in the first part of chapter 2. ML algorithm in embedded applications and its approach on system-on-chip are discussed. At last, evaluation of ML algorithm such as root mean square error, mean absolute error, and mean absolute percentage error are discussed.

The proposed ML algorithm is used to train the model to predict the delay of each CPs in the digital circuits. This makes this algorithm the best candidate to be used in the dynamic compensation schemes. Hence, this approach of prediction of aging using two different digital circuits in chapter 3. The analysis of ranking variations of critical paths is demonstrated and understand the interpretation of PVTA is discussed. Moreover, the performance of our proposed model is compared with the SPICE simulation and validates the improved accuracy of our prediction method.

Various dynamic compensation schemes using our proposed model are demonstrated in Chapter 4. Two novel approaches to our prediction methods are executed. One is the Aging aware OPP overclocking, and the other is the maximum performance with OPP Cap methods. This approach of our model helps to predict the circuit failure by analyzing the flag count variations in the circuit. It also illustrates the flexibility to adjust the supply voltage and frequency based on design requirements.

Perspectives

- This thesis work has demonstrated the proposed ML algorithm for two different digital circuits offline. However, a thorough analysis needs to be done online to know the usage of our model in an On-chip process.

- The dynamic voltage and frequency scheme proposed in this thesis using the ML algorithm has been simulated offline. In order to confirm the results, the online simulation with the chip needs to be analyzed on a silicon measurement. It is also essential to measure the power consumption and reliability of the circuit. But, accurate power consumption can be evaluated by using our ML algorithm is possible.

- Future works will include exploring new OPP Optimization strategies and integrating with classical Monitor Insertion Flow, most notably identifying an optimal Path selection strategy. We also plan to exploit its computational simplicity for embedded applications to allow online aging-ware OPP adaptation

- The examples used in chapter 3 for analyzing the degradation of the circuit does not contain memories. In general, most of the complex circuits on-chip contain multiple memories. The path between a flip-flop and memory timing analysis is possible by using our proposed model that needs to be tested.

- Investigation of layout of the circuit and Critical path selections to be done along with the net model, parasitic, and routing of the systems.

List of Publicaitons

- **S.Kalpana**, M. Portolan, L. Anghel, "Activity Aware Prediction of Critical path Aging on FDSOI Technology", in Microelectronics Reliability Journal, **2021**.
- M. Portolan, R. S. Feitoza, G. T. Tchendjou, V. Reynaud, S. **kalpana**, M.Barragan, E. Simeu, P.Maistre, L. Anghel, R. Leveugle, Salvador Mir, "A Comprehensive End-to-end Solution for a Secure and Dynamic Mixed-signal 1687 System", *IEEE 26th International Symposium on On-Line Testing and Robust System Design (IOLTS)*,Napoli, Italy, *July* **2020**
- G. Di Natale, E. Ioana Vatajelu, **S. Kalpana**, and L. Anghel, "Hidden-Delay-Fault Sensor for Test, Reliability and Security," *Proceedings of Design Automatic Test European. Conf. Exhib. DATE 2019,* pp. 316–319, **2019**.
- **S. Kalpana**, Michele Portolan, Lorena Anghel, "Run-Time Aging Prediction Through Machine Learning", *International Test Conference*, US, Oct28-Nov2, **2018**

References

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Proc. IEEE*, vol. 86, no. 1, pp. 82–85, 1998.

[2] D. Bennett, "Science Focus Magazine," https://www.sciencefocus.com/future-technology/cool-gadgets, 2021.

[3] T. Tiwari, "How Artificial Intelligence, Machine Learning, and Deep Learning are Radically Different?," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 8, no. 2, p. 1, 2018.

[4] Y. Wang, S. Cotofana, and L. Fang, "A unified aging model of NBTI and HCI degradation towards lifetime reliability management for nanoscale MOSFET circuits," in *Proceedings of the 2011 IEEE/ACM International Symposium on Nanoscale Architectures, NANOARCH 2011*, pp. 175–180, 2011.

[5] D. Lorenz, M. Barke, and U. Schlichtmann, "Aging analysis at gate and macro cell level," *IEEE/ACM Int. Conf. Comput. Des. Dig. Tech. Pap. ICCAD*, pp. 77–84, 2010.

[6] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge, "Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits," *Proc. IEEE*, vol. 98, no. 2, pp. 253–266, 2010.

[7] J. Tschanz *et al.*, "Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging," *Dig. Tech. Pap. - IEEE Int. Solid-State Circuits Conf.*, pp. 292–294, 2007.

[8] A. Sivadasan, R. J. Shah, V. Huard, F. Cacho, and L. Anghel, "NBTI aged cell rejuvenation with back biasing and resulting critical path reordering for digital circuits in 28nm FDSOI," *Proc. 2018 Des. Autom. Test Eur. Conf. Exhib. DATE 2018*, vol. 2018-Janua, pp. 997–998, 2018.

[9] L. Anghel, A. Benhassain, A. Sivadasan, F. Cacho, and V. Huard, "Early system failure prediction by using aging in situ monitors: Methodology of implementation and application results," *2016 IEEE 34th VLSI Test Symp.*, pp. 1–1, 2016.

[10] A. Sivadasan, F. Cacho, S. Ahmed, Benhassain, V. Huard, and L. Anghel, "Workload impact on BTI HCI induced aging of digital circuits: A system level analysis," *CEUR Workshop Proc.*, vol. 1566, pp. 38–40, 2016.

[11] S. S. Sapatnekar, "What Happens when Circuits Grow Old : Aging Issues in CMOS Design," 2013.

[12] B. Halak, V. Tenentes, and D. Rossi, "The impact of BTI aging on the reliability of level shifters in nano-scale CMOS technology," *Microelectron. Reliab.*, vol. 67, pp. 74–81, 2016.

[13] Y. Zhao and H. G. Kerkhoff, "Highly Dependable Multi-processor SoCs

Employing Lifetime Prediction Based on Health Monitors," in *Proceedings of the Asian Test Symposium*, pp. 228–233, 2016.

[14] T. Sakurai and A. R. Newton, "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," *IEEE J. Solid-State Circuits*, vol. 25, no. 2, pp. 584–594, Apr. 1990.

[15] B. C. Paul, K. Kang, H. Kufluoglu, M. A. Alam, and K. Roy, "Impact of NBTI on the temporal performance degradation of digital circuits," *IEEE Electron Device Lett.*, vol. 26, no. 8, pp. 560–562, 2005.

[16] D. K. Schroder and J. A. Babcock, "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *J. Appl. Phys.*, vol. 94, no. 1, pp. 1–18, 2003.

[17] T. Grasser, "Stochastic charge trapping in oxides: From random telegraph noise to bias temperature instabilities," vol. 52, no. 1. Elsevier Ltd, 2012.

[18] A. W. Strong *et al.*, "Reliability Wearout Mechanisms in Advanced CMOS Technologies," *Reliab. Wearout Mech. Adv. C. Technol.*, pp. 1–624, 2009.

[19] F. Ahmed and L. Milor, "Ring oscillator based embedded structure for decoupling PMOS/NMOS degradation with switching activity replication," in *International Conference on Microelectronic Test Structures (ICMTS)*, no. c, pp. 118–121, Mar. 2010.

[20] D. P. Ioannou, S. Mittl, and G. La Rosa, "Positive Bias Temperature Instability Effects in nMOSFETsWith HfO2/TiN Gate Stacks," *Trans. Device Mater. Reliab. IEEE*, vol. 9, no. 2, pp. 128–134, 2009.

[21] E. Maricau and G. Gielen, *Analog IC reliability in nanometer CMOS*. 2013.

[22] T. H. Ning, "Hot-electron emission from silicon into silicon dioxide," *Solid State Electron.*, vol. 21, no. 1, pp. 273–282, 1978.

[23] H. J. Lee and K. K. Kim, "Analysis of time dependent dielectric breakdown in nanoscale CMOS circuits," *Int. SoC Des. Conf.*, pp. 440–443, 2011.

[24] J. W. McPherson, "Time dependent dielectric breakdown physics - Models revisited," *Microelectron. Reliab.*, vol. 52, no. 9–10, pp. 1753–1760, 2012.

[25] M. M. Mahmoud, N. Soin, and H. A. H. Fahmy, "Design framework to overcome aging degradation of the 16 nm VLSI technology circuits," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 33, no. 5, pp. 691–703, 2014.

[26] T. D. Burd, T. A. Pering, A. J. Stratakos, and R. W. Brodersen, "Dynamic voltage scaled microprocessor system," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1571–1580, 2000.

[27] T. Kuroda *et al.*, "Variable supply-voltage scheme for low-power high-speed CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 33, no. 3, pp. 454–461, 1998.

[28] S. Das *et al.*, “RazorII: In situ error detection and correction for PVT and ser tolerance,” *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 32–48, 2009.

[29] M. Nicolaidis, “Time redundancy based soft-error tolerance to rescue nanometer technologies,” *Proc. 17th IEEE VLSI Test Symp. (Cat. No.PR00146)*, pp. 86–94, 1999.

[30] L. Lai, V. Chandra, R. C. Aitken, and P. Gupta, “SlackProbe: A flexible and efficient in situ timing slack monitoring methodology,” *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 33, no. 8, pp. 1168–1179, 2014.

[31] M. Wirnshofer, L. Heiß, A. N. Kakade, N. P. Aryan, G. Georgakos, and D. Schmitt-Landsiedel, “Adaptive voltage scaling by in-situ delay monitoring for an image processing circuit,” *Proc. 2012 IEEE 15th Int. Symp. Des. Diagnostics Electron. Circuits Syst. DDECS 2012*, pp. 205–208, 2012.

[32] J. Keane, X. Wang, D. Persaud, and C. H. Kim, “An all-in-one silicon odometer for separately monitoring HCI, BTI, and TDDB,” *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 817–829, 2010.

[33] T. H. Kim, P. F. Lu, K. A. Jenkins, and C. H. Kim, “A Ring-Oscillator-Based Reliability Monitor for Isolated Measurement of NBTI and PBTI in High-k/Metal Gate Technology,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 23, no. 7, pp. 1360–1364, 2015.

[34] R. Moazzami, J. Lee, and C. Hu, “Temperature acceleration of Time Dependent Dielectric Breakdown,” *Electron Devices, IEEE*, vol. 36, no. 11, pp. 2462–2465, 1989, [Online].

[35] N. Suzumura, M. Ogasawara, T. Furuhashi, and T. Koyama, “Study on vertical TDDB degradation mechanism and its relation to lateral TDDB in Cu/low-k damascene structures,” in *2014 IEEE International Reliability Physics Symposium*, , pp. 3A.4.1-3A.4.6, Jun. 2014.

[36] D. Wang, W. L. Wang, M. Y. Huang, A. Lek, J. Lam, and Z. H. Mai, “Failure mechanism analysis and process improvement on time-dependent dielectric breakdown of Cu/ultra-low-k dielectric based on complementary Raman and FTIR spectroscopy study,” *AIP Adv.*, vol. 4, no. 7, pp. 0–9, 2014.

[37] B. P. Das, “Random Local Delay Variability: On Chip measurement and modeling,” 2009.

[38] S. K. Saha, “Modeling process variability in scaled CMOS technology,” *IEEE Des. Test Comput.*, vol. 27, no. 2, pp. 8–16, 2010.

[39] V. S. Daniel Arbet, Lukas Nagy, *Ultra-Low-Voltage IC Design Methods*, no. Cell Interaction-Regulation of Immune Responses, Disease Development and Management Strategies. IntechOpen, 2020.

[40] C. Park *et al.*, “Reversal of temperature dependence of integrated circuits operating at very low voltages,” *Tech. Dig. - Int. Electron Devices Meet.*, pp. 71–

74, 1995.

[41] S. M. Sze, *Physics of Semiconductor Devices*, Wiley ed 1.

[42] H. Su, F. Liu, A. Devgan, E. Acar, and S. Nassif, "Full chip leakage estimation considering power supply and temperature variations," in *Proceedings of the 2003 international symposium on Low power electronics and design - ISLPED '03*, p. 78, 2003.

[43] M. D. Miljana Sokolović, "Digital circuits delay analysis," January, 2015.

[44] B. Halak, *Ageing of integrated circuits : causes, effects and mitigation techniques*. Springer, Cham, 2020.

[45] N. Devtaprasanna, A. Gunda, P. Krishnamurthy, S. M. Reddy, and I. Pomeranz, "Test generation for open defects in CMOS circuits," in *Proceedings - IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 41–49, 2006.

[46] R. H. Ramlee and M. Zwolinski, "Using Iddt current degradation to monitor ageing in CMOS circuits," *Proc. - 2016 26th Int. Work. Power Timing Model. Optim. Simulation, PATMOS 2016*, pp. 200–204, 2017.

[47] J. B. Kim, "Current monitoring circuit for fault detection in CMOS integrated circuit," *Int. J. Electron.*, vol. 95, no. 10, pp. 999–1007, 2008.

[48] N. Pour Aryan, L. Heiß, D. Schmitt-Landsiedel, G. Georgakos, and M. Wirnshofer, "Comparison of in-situ delay monitors for use in Adaptive Voltage Scaling," *Adv. Radio Sci.*, vol. 10, pp. 215–220, 2012.

[49] A. Benhassain, S. Mhira, F. Cacho, V. Huard, and L. Anghel, "In-situ slack monitors: taking up the challenge of on-die monitoring of variability and reliability," in *2016 1st IEEE International Verification and Security Workshop (IVSW)*, vol. 6, pp. 1–5, Jul. 2016.

[50] F. Cacho, A. Benhassain, R. Shah, S. Mhira, V. Huard, and L. Anghel, "Investigation of critical path selection for in-situ monitors insertion," *2017 IEEE 23rd Int. Symp. On-Line Test. Robust Syst. Des. IOLTS 2017*, pp. 247–252, 2017.

[51] L Bemimi, G DeMicheli, *Dynamic power management: design techniques and CAD tools*, 1st ed. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

[52] G. Gammie *et al.*, "Smart reflex power and performance management technologies for 90 nm, 65 nm, and 45 nm mobile application processors," *Proc. IEEE*, vol. 98, no. 2, pp. 144–159, 2010.

[53] T. B. Chan, W. T. J. Chan, and A. B. Kahng, "Impact of adaptive voltage scaling on aging-aware signoff," *Proc. -Design, Autom. Test Eur. DATE*, no. 1, pp. 1683–1688, 2013.

[54] D. Gutierrez, "How to Use Analytics-Driven Embedded Systems to Drive Smart Technology Development," https://insidebigdata.com/2016/06/03/how-to-use-analytics-driven-embedded-systems-to-drive-smart-technology-development/,

2016.

[55] D. M. Tartakovsky and S. Broyda, “Programming Collective Intelligence,” in *Journal of Contaminant Hydrology*, vol. 120–121, pp. 129–140, 2011.

[56] C. M. Bishop, *Pattern Recognition and Machine Learning*, vol. 53, no. 9. 2013.

[57] K. Ramasubramanian and A. Singh, “Introduction to Machine Learning and R,” in *Machine Learning Using R*, Berkeley, CA: Apress, pp. 1–33, 2019.

[58] M. Awad and R. Khanna, *Machine Learning in Action: Examples*. 2015.

[59] J. J. D. W. L. Chao, “Integrated Machine Learning Algorithm for Human Age Estimation.” 2011.

[60] M. A. Wiering, H. Van Hasselt, A. D. Pietersma, and L. Schomaker, “Reinforcement learning algorithms for solving classification problems,” *IEEE SSCI 2011 Symp. Ser. Comput. Intell. - ADPRL 2011 2011 IEEE Symp. Adapt. Dyn. Program. Reinf. Learn.*, pp. 91–96, 2011.

[61] C. W. Anderson, D. C. Hittle, A. D. Katz, and R. M. Kretchmar, “Synthesis of reinforcement learning, neural networks and PI control applied to a simulated heating coil,” *Artif. Intell. Eng.*, vol. 11, no. 4, pp. 421–429, 1997.

[62] Z. A. Almaliki, “Do you know how to choose the right machine learning algorithm among 7 different types?,” https://towardsdatascience.com/do-you-know-how-to-choose-the-right-machine-learning-algorithm-among-7-different-types-295d0b0c7f60, 2019.

[63] J. Lee, M. Stanley, A. Spanias, and C. Tepedelenlioglu, “Integrating machine learning in embedded sensor systems for Internet-of-Things applications,” *2016 IEEE Int. Symp. Signal Process. Inf. Technol. ISSPIT 2016*, pp. 290–294, 2017.

[64] P. Sattigeri, J. J. Thiagarajan, M. Shah, K. N. Ramamurthy, and A. Spanias, “A scalable feature learning and tag prediction framework for natural environment sounds,” *Conf. Rec. - Asilomar Conf. Signals, Syst. Comput.*, vol. 2015-April, pp. 1779–1783, 2015.

[65] L. Deng, “Deep learning: From speech recognition to language and multimodal processing,” *APSIPA Trans. Signal Inf. Process.*, vol. 5, no. 2016, pp. 1–15, 2016.

[66] S. Peshin *et al.*, “A Photovoltaic (PV) Array Monitoring Simulator,” 2015.

[67] S. Rao *et al.*, “An 18 kW solar array research facility for fault detection experiments,” in *2016 18th Mediterranean Electrotechnical Conference (MELECON)*, no. April, pp. 1–5, Apr. 2016.

[68] M. G. S. Murshed, C. Murphy, D. Hou, N. Khan, G. Ananthanarayanan, and F. Hussain, “Machine learning at the network edge: A survey,” *arXiv*, pp. 1–34, 2019.

[69] S. Branco, A. G. Ferreira, and J. Cabral, “Machine learning in resource-scarce embedded systems, FPGAs, and end-devices: A survey,” *Electron.*, vol. 8, no. 11,

2019.

[70] J. T. Mentzer and K. B. Kahn, “Forecasting technique familiarity, satisfaction, usage, and application,” *J. Forecast.*, vol. 14, no. 5, pp. 465–476, 1995.

[71] T. M. Mccarthy, D. F. Davis, S. L. Golicic, and J. T. Mentzer, “The evolution of sales forecasting management: A 20-year longitudinal study of forecasting practices,” *J. Forecast.*, vol. 25, no. 5, pp. 303–324, 2006.

[72] D. M. Norris, I. Markovic, and M. Antibodies, “Evaluation of Extrapolative Forecasting Methods: Results of a survey of Academicians and Practitioners,” vol. I, no. October 1981, pp. 215–217, 2003.

[73] M. Mohammed, M. B. Khan, and E. B. M. Bashie, *Machine learning: Algorithms and applications*, no. July. 2016.

[74] M. Altieri, S. Lesecq, E. Beigne, and O. Heron, “Towards on-line estimation of BTI/HCI-induced frequency degradation,” *IEEE Int. Reliab. Phys. Symp. Proc.*, p. CR6.1-CR6.6, 2017.

[75] M. A. SCARPATO, “Estimation de la performance des circuits numériques sous variations PVT et vieillissement,” University of Grenoble Alpes, 2017.

[76] A. Sivadasan *et al.*, “Architecture-and workload-dependent digital failure rate,” *IEEE Int. Reliab. Phys. Symp. Proc.*, p. CR8.1-CR8.4, 2017.

[77] R. J. Shah, “Reliability Improvement by Dynamic Wearout Management using In-Situ Monitors Riddhi Jitendrakumar Shah To cite this version : HAL Id : tel-03103505,” 2021.

[78] N. E. H. Weste and D. M. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, vol. 53, no. 9. 2013.

[79] S. Geisser, S. Geisser, S. Geisser, and S. Geisser, “Predictive inference : an introduction,” New York, NY <etc.>: Chapman and Hall, 1993.

[80] M. Kuhn and K. Johnson, *Applied Predictive Modeling [Hardcover]*. 2013.

[81] J. Watt, R. Borhani, and A. Katsaggelos, *Machine Learning Refined: Foundations, Algorithms, and Applications (pp. I-Iv)*. Cambridge: Cambridge University Press, 2016.

[82] R. Chadha and J. Bhasker, "An ASIC low power primer: Analysis, techniques and specification," 2013.

[83] Synopsys, “Power Compiler User Guide,” *Synopsys*, no. September, pp. 113–194, 2009.

[84] A. Sivadasan *et al.*, “Workload dependent reliability timing analysis flow,” *Proc. 2017 Des. Autom. Test Eur. DATE 2017*, pp. 736–737, 2017.

[85] T. K. Ho, “Random Decision Forest,” 1995.

[86] J. Feng and T. ShangGuan, *Information Computing and Applications: Third*

International Conference, ICICA 2012 Chengde, China, September 14-16, 2012 Proceedings, vol. 9. 2012.

[87] H. C. Chen, D. H. C. Du, and L. R. Liu, "Critical Path Selection for Performance Optimization," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 12, no. 2, pp. 185–195, 1993.

[88] X. Fu, H. Li, and X. Li, "Testable critical path selection considering process variation," *IEICE Trans. Inf. Syst.*, vol. E93-D, no. 1, pp. 59–67, 2010.

[89] J. J. L. N. S. S. James J. CurtinWilliam E. Dougherty, "Method for eliminating negative slack in a netlist via transofrmation and slack categorization," US 78,810,062 B2, 2010.

[90] D. Ernst *et al.*, "Razor: A low-power pipeline based on circuit-level timing speculation," *Proc. Annu. Int. Symp. Microarchitecture, MICRO*, vol. 2003-Janua, pp. 7–18, 2003.

[91] S. Heo, K. Barr, and K. Asanović, "Reducing Power Density through Activity Migration," *Proc. Int. Symp. Low Power Electron. Des.*, pp. 217–222, 2003.