Health Monitoring and Life-time Prognostics to Enable Dependable Many-processor SoCs

Yong Zhao

HEALTH MONITORING AND LIFE-TIME PROGNOSTICS TO ENABLE DEPENDABLE MANY-PROCESSOR SOCS

Yong Zhao

HEALTH MONITORING AND LIFE-TIME PROGNOSTICS TO ENABLE DEPENDABLE MANY-PROCESSOR SOCS

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof.dr. T.T.M. Palstra,
on account of the decision of the Doctorate Board,
to be publicly defended
on Thursday, the 12th of December 2019 at 12.45 hours

by

Yong Zhao

born on the 20th of March 1988
in Tai'an, China

This dissertation has been approved by:

Dr. ir. H.G. Kerkhoff University of Twente (EWI)

Cover design: Michel Wolf
Printed by: Ipskamp Printing
Lay-out: Yong Zhao
ISBN: 978-90-365-4916-5
DOI: 10.3990/1.9789036549165

Graduation Committee:

Chairman/secretary:

Prof.dr.ir.	J. N. Kok	University of Twente (EWI)

Supervisor:

Dr. ir.	H.G. Kerkhoff	University of Twente (EWI)

Committee Members:

Prof.dr.ir.	G.J.M Smit	University of Twente (EWI)
Prof.dr.ir.	A. Pras	University of Twente (EWI)
Prof. dr.	Z. Peng	University of Linköping (CS)
Prof.dr.ir.	S. Hamdioui	Delft University of Technology (EWI)
Prof.dr.	M. I. A. Stoelinga	University of Twente (EWI)
Dr.ir.	G. Rauwerda	Technolution B.V.

Acknowledgements

This thesis is not only a result of the continuous hard work, enthusiasm and consistent efforts during the past years, but also the encouragement and support from a number of people. Therefore, I would like to take this opportunity to pay my sincere thanks to all of them.

I would like to show my sincere gratitude to my supervisor Hans Kerkhoff for his mentorship and support throughout my PhD work, also for his great patience and effort spent on reviewing my thesis. His scientific attitude and open-minded discussions have helped me substantially in completing the PhD research. His advice on both research as well as shaping my personality have been priceless. Also, special thanks to Prof. Gerard Smit for his time of reviewing my thesis. I also would like to thank my former colleagues at the University of Twente: Andreina Zambrano, Jinbo Wan, Aamir Khan, Ahmed Ibrahim, Hassan Ebrahimi, Xiao Zhang, Ghazanfar Ali, Bert Helthuis, Marlous Weghorst, Thelma Nordholt and so on. I keep great memories to know them and work with them at the University of Twente. Also acknowledgements to Eelke Strooisma and Tijs Lammertink for the cooperation and support during my PhD project. The intense cooperation with Recore Systems, and Gerard Rauwerda is especially appreciated which made my research possible.

I would also like to thank all my friends, especial Zheming Zhu, Huan Wang, Jiabiao Zhang, René Groothedde, Florian Oosterberg, Jinfeng Mu, Ying Du, Lantian Chang, Meiru Mu, Xiaoyan Zhang, Xin Zhang, Lei Zhang, Junwei Xue, Qiang Wang and all my friends for their friendship and support.

Finally, I would like to pay my deepest thanks to my family, my father who always stands behind and encourages me in every aspect of life, my mother who I will never be able to describe my appreciation for her love, and my sister who supports my entire life in every way.

Abstract

Nowadays, with the requirement of more powerful data-processing capabilities and the availability of advanced IC technologies, an increased number of complex designs of Many-Processor System-on-Chips (MP-SoCs) have been proposed. They are increasingly applied in life- or mission-critical applications such as automotive, military and aerospace. Hence these SoCs endure much more severe external stress conditions in terms of temperature, shock and radiation as compared to conventional consumer applications.

Furthermore, the effort to shrink dimensions of transistors for enabling more complexity has not only resulted in an extremely high level of device density, but has also accelerated the wear-out of devices, circuits and associated electronic systems. Hence this has contributed to serious dependability challenges.

In this thesis, first the dependability challenges in our target MP-SoC design have been elaborated. Next, possible techniques have been explored to enable dependable design. The functionality of all the designed and implemented hardware as well as the developed software programs have been validated in silicon. Its effectiveness was evaluated using actual measurement results and the analyses of the results were based on developed mathematical models and algorithms.

In the scope of this thesis, based on the aging mechanisms like NBTI and dependability analysis of our target MP-SoCs, as well as based on our actual application of these systems, the mean downtime was required to be close to zero. As such, reliability, availability and maintainability are major issues in the approach to enable the implementation of dependable MP-SoCs.

Meanwhile, a prognostic health-monitoring approach needs to be taken to attain a mean downtime close to zero. It typically includes the usage of health monitors (HMs) as well as the application of prognostics life-time prediction software. Based on these, a repair action for a degrading and potentially faulty processor core via remapping can be executed by using spare or not fully-employed processor cores; as a result, the system can act before the occurrence of a failure, thus resulting in a full-time available system.

The health monitoring approach proposed in this thesis includes one embedded hardware HM and another software-based HM. The first one is capable of carrying out voltage and temperature measurements as well as delay-time monitoring, while the latter

includes the critical-path delay monitoring, I_{DDQ} monitoring as well as unit-based I_{DDT} monitoring.

In order to validate the feasibility, the developed software-based HM including implemented hardware as well as the designed software program was implemented within our Xentium-based MP-SoCs. The size of the software-based HM programs is sufficiently small for nowadays processors and their power consumption is negligible.

The setup of our accelerated testing experiment was presented with the measurement results of our MP-SoCs with regard to the critical-path delay, I_{DDQ} and I_{DDT}. The correlation coefficients between their results were modelled and provided. The approach is generic and can be applied similarly in other SoCs by extracting functional features using delay monitor and particular states for the $I_{DDQ/T}$ monitor.

Based on the crucial health-monitoring information regarding the dependability of the system, the remaining lifetime prediction could be estimated. It was calculated from the present moment until the time the health-monitoring data reaches the pre-set repair threshold. A genetic-algorithm based degradation optimization model for the critical-path delay result was proposed, and in addition, an alternative remaining lifetime-prediction method based on the I_{DDX} monitoring results for the Xentium processor was developed; it can reach a good accuracy and also reduce the measurement time as compared to the critical-path delay approach.

In conclusion, our proposed health-monitoring based dependability approach and lifetime-prediction technique have proven to be feasible and efficient to enable the design of a dependable SoC. The successful integration of the software programs like e.g. for the critical-path delay and the I_{DDX} indicates that these techniques can be incorporated into any generic MP-SoC with no or little changes.

Samenvatting

Tegenwoordig neemt de behoefte aan meer krachtige dataverwerking toe. De beschikbaarheid van geavanceerde IC-technologieën maakt complexe ontwerpen van Many-Processor System-on-Chips (MP-SoC's) mogelijk en noodzakelijk. Ze worden in toenemende mate gebruikt in toepassingen die cruciaal zijn voor de veiligheid, zoals in de auto-, militaire- en lucht- en ruimtevaartindustrie. In vergelijking met conventionele consumententoepassingen worden deze SoC's blootgesteld aan veel zwaardere externe stresssituaties met betrekking tot o.a. temperatuur, schokken en straling.

Bovendien heeft de inspanning om de afmetingen van transistoren te verkleinen om zo meer complexiteit mogelijk te maken niet alleen geleid tot een extreem hoge transistor dichtheid, maar ook tot een versnelde slijtage van circuits en elektronische systemen. Dit resulteerde dus ook in grote uitdagingen op het gebied van hun betrouwbaarheid.

In het proefschrift zijn de betrouwbaarheidsuitdagingen eerst in het door ons beoogde MP-SoC ontwerp uitgewerkt. Vervolgens zijn mogelijke technieken onderzocht om een betrouwbaar ontwerp te kunnen maken. De functionaliteit van alle ontworpen en geïmplementeerde hardware en software is geverifieerd via een silicium implementatie. De effectiviteit werd geëvalueerd aan de hand van actuele meetresultaten, en de analyses van de resultaten werden gebaseerd op ontwikkelde wiskundige modellen en algoritmes.

Het proefschrift is gebaseerd op verouderingsmechanismen zoals NBTI en een betrouwbaarheidsanalyse van onze doel MP-SoC's in een werkelijke toepassing van het systeem. Als eis moest de gemiddelde uitvaltijd bijna nul zijn. Als gevolg hiervan zijn bedrijfszekerheid, beschikbaarheid en onderhoudbaarheid belangrijke aandachtspunten in de aanpak om de implementatie van betrouwbare MP-SoC's mogelijk te maken.

Er is in dit proefschrift gekozen voor een prognostische benadering van health-monitoring om een gemiddelde uitvaltijd van bijna nul te bereiken. Dit omvat het gebruik van health monitors (HM's) welke specifieke metingen verrichten, en de toepassing van prognostische software voor het voorspellen van de levensduur van bijvoorbeeld de processor rekenkernen van een MP-SoC. Op basis hiervan kan een reparatie actie voor een gedegradeerde en mogelijk binnenkort defecte processorkern via hergroepering worden uitgevoerd door gebruik te maken van reserve of onvolledig gebruikte processorkernen. Het systeem kan dus actie ondernemen voordat er een defect optreed, wat resulteert in een altijd beschikbaar systeem.

Dit proefschrift omvat een voorgestelde aanpak voor health monitoring via een geïntegreerde hardware HM en een software deel van de HM. Allereerst is het nodig om spannings- en temperatuurmetingen en vertragings-tijd metingen uit te voeren. Dit behelst kritisch-pad vertraging monitoring, I_{DDQ}-monitoring en I_{DDT}-monitoring per proceskern.

Om de haalbaarheid te valideren werd een ontwikkelde software-gebaseerde HM met inbegrip van de gerealiseerde hardware en het ontworpen softwareprogramma geïmplementeerd in onze op de Xentium-gebaseerde MP-SoC's. De omvang van de software-gebaseerde HM-programma's was klein genoeg voor hedendaagse processoren en hun extra stroomverbruik is hierdoor te verwaarlozen.

Vervolgens is de opzet van onze test experimenten op basis van veroudering gepresenteerd. Dit resulteerde in meetresultaten van onze MP-SoC's met betrekking tot de kritisch-pad vertraging, I_{DDQ} en I_{DDT} stroom metingen. De correlatiecoëfficiënten tussen deze resultaten zijn gemodelleerd en gepresenteerd. De aanpak is generiek en kan op dezelfde manier worden toegepast in andere SoC's door het extraheren van functionele functies met behulp van een vertragings-tijd monitor en metingen van de I_{DDQ} en I_{DDT} -monitoren.

Op basis van de data van de cruciale health-monitoring m.b.t de degradatie van het systeem kon een voorspelling van de resterende levensduur worden gedaan. Dit werd berekend vanaf het huidige meet moment tot het moment dat de health-monitoring data de vooraf ingestelde reparatie drempel heeft bereikt. Er is gebruik gemaakt van een genetisch -algoritme. Het is gebaseerd op een model voor de optimalisatie van de degradatie van het resultaat van de kritisch-pad vertraging. Daarnaast werd een alternatieve methode voor het voorspellen van de resterende levensduur ontwikkeld op basis van de I_{DDQ} en I_{DDT} monitoring resultaten voor de Xentium processor. Er is aangetoond dat in vergelijking met de kritisch-pad vertraging methode, een goede nauwkeurigheid bereikt kan worden en dat ook de meettijd te verkorten is.

Onze voorgestelde health-monitoring aanpak is gebaseerd op zowel betrouwbaarheidsbenadering als levenstijdvoorspelling; deze zijn beide haalbaar en efficiënt gebleken om zodoende het ontwerp van betrouwbare SoC's mogelijk te maken. De succesvolle integratie van de softwareprogramma's zoals de kritisch- pad vertraging, de I_{DDQ} en I_{DDT} geeft aan dat deze technieken in elke generieke MP-SoC met geen of weinig veranderingen opgenomen kunnen worden.

Contents

LIST OF ACRONYMS

ADC	Analog to Digital Converter
AF	Acceleration Factor
AFE	Analogue/mixed-signal Front-Ends
API	Application Program Interface
AT	Accelerated Testing
ATPG	Automatic Test Pattern Generation
BF	Beam Former
BISTR	Built-in-Self-Test-and-Repair
CBM	Condition-Based Maintenance
CFR	Constant Failure Rate
CMOS	Complementary Metal-Oxide Semiconductor
CRISP	Cutting edge Reconfigurable ICs for Stream Processing
DfT	Design for Testability
DM	Dependability Manager
DMA	Direct Memory Access
DRM	Dynamic Reliability Management
DSP	Digital Signal Processing
DUT	Device Under Test
DVFS	Dynamic Voltage/Frequency Scaling
EI	Embedded Instruments
EM	Electromigration
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FIT	Failure in Time

FSM	Finite State Machine
GA	Genetic Algorithm
GNSS	Global Navigation Satellite System
GPD	General Purpose Device
HASS	Highly Accelerated Stress Screening
HCI	Hot Carrier Injection
HM	Health Monitor
HMP	Health Monitoring and Prognostics
HT	Hilbert Transform
HTOL	High Temperature Operating (Bias) Life
IC	Integrated Circuit
IIP	Infrastructural IP
IJTAG	Internal Joint Test Action Group
IM	Infant Mortality
IP	Intellectual Property
MDT	Mean Down Time
MOSFET	Metal-Oxide-Semiconductor Field-Effect Transistor
MP-SoC	Many-Processor System-on-Chip
MSE	Mean Squared Error
MTBF	Mean Time Between Failure
MTTF	Mean Time To Failure
NBTI	Negative Bias Temperature Instability
NI	Network Interface
NoC	Network-on-Chip
NOP	No OPeration
PTC	Power Temperature Cycling
QoS	Quality-of-Service

RFD	Reconfigurable Fabric Device
RLP	Remaining Life-time Prediction
RMSE	Rooted Mean Squared Error
ROSC	Ring Oscillator
RUL	Remaining Useful Lifetime
SBHM	Software-based Health Monitoring
SBST	Software-based Self-Test
SIBs	Segment Insertion Bits
SoC	System-on-Chips
SRAM	Static Random Access Memory
STARS	Sensor Technology Applied in Reconfigurable Systems
TAP	Test Access Port
TDDB	Time Dependent Dielectric Breakdown
TMR	Triple Modular Redundancy
TRE	Test Response Evaluator
UAV	Unmanned Aerial Vehicle
VLIW	Very Large Instruction Word

Chapter 1

INTRODUCTION

ABSTRACT *– This chapter presents an introduction to the research scope of this thesis. The dependability challenges with regard to CMOS technology scaling are discussed first. Traditional methods to cope with these challenges are briefly indicated, but these are not sufficient anymore, especially in safety-critical applications. This requires new techniques for enhancing the dependability of such integrated systems. With regard to this thesis, a motivation for the proposed research is provided, as well as a formulation of the research problem statements to be answered. Finally, an outline of the thesis is presented.*

1.1 INTRODUCTION

The budget for homeland security in the US only, exceeds 40 billion dollars annually in 2017 [Home 17]. Worldwide this number is estimated to be a multiple of this amount. Part of these budgets are allocated to reconfigurable multi-sensory systems [Kerk 09], monitoring the environment in many aspects, especially around security-sensitive compounds like major harbours and airfields and defence systems. A not sufficiently dependable design of such systems can not only lead to environmental and financial disasters but also loss of human life.

With the integrated circuit technological advances in microelectronics, such as computers and networking systems, one of the key challenges in those systems is a decreasing reliability [Tamb 14]. This is among others caused by the increase in electric field across the transistor gate-oxide, channel, and interconnects which aggravates transistor-degradation mechanisms [Inte 09]. In the chip process nodes above 100 nm, this rate of degradation processes was sufficiently low that it did not raise concerns about end-of-lifetime failures. But in advanced process nodes below 90 nm, the degradation mechanisms severely threaten the chip reliability.

Previous studies [Whit 08] indicated that the wear-out failures (failure in time, FIT) appear much earlier in products employing recent CMOS technology nodes as compared to older technologies, as can be seen in Figure 1.1. Failures during the infant mortality (IM) will be eliminated by burn-in highly accelerated stress screening (HASS). The period of constant failure rate (CFR) decreases and wear-out failures are occurring earlier in time. Meanwhile, the chance of IM and the CFR also increase with the down scaling of technology. Therefore, the continuous operation and the capability to deal with possible faults of such systems like System-on-Chips (SoC) become worse. The consequence is that the chance of a fault results in an increased mean down time (MDT) [Kuma 80], and hence the mean service time will be significantly affected.

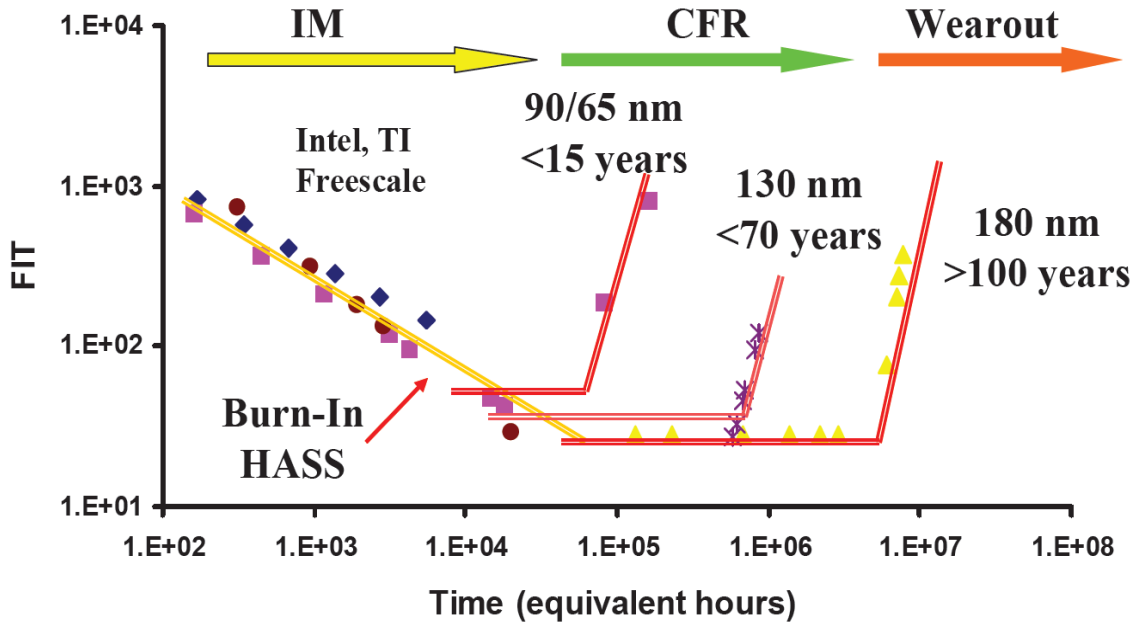


Figure.1: Normalized reliability data of manufacturers at the product level in terms of failures in time (FIT) as technology scales down. [Whit 08].

Dependability represents the degree of confidence that the system will operate as expected and that the system will not fail in normal use [Aviz 01]. Dependability has become essential in our modern society. Dependability as a concept encompasses several attributes. In [Aviz 04] the following attributes were defined for dependability: 1) reliability, 2) availability, 3) maintainability, 4) safety, 5) integrity and 6) security.

The different attributes of dependability have been defined as [Aviz 01]:

Reliability: capability of a system to provide the continuity of its correct service, or the probability a system functions correctly under a given set of operating conditions at any given time. Usually, the reliability of a system is expressed through mean time between failures (MTBF) or mean time to failure (MTTF).

Availability: readiness of a system for its correct service, or the probability a system correctly delivers its service at any given time. The availability of a system is expressed through the uptime.

Maintainability: ability of a system to undergo modifications and repairs, or the probability a system can be repaired at any given time if it fails to deliver correct functionality.

Safety: capability of the system to avoid catastrophic consequences with regard to the users or the environment.

Integrity: capability of a system to avoid any alterations.

Security: capability of a system to prevent the unauthorized disclosure of information.

To enhance the dependability of a system, and address potential threats quickly, the adaptation/extension of a dependable system [Star 11] should be very rapid. This involves hardware as well as software. This thesis follows the current trend in Many-Processor System-on-Chip (MP-SoC) design, where after the massive introduction of embedded software to increase the flexibility of the system, now also the hardware should be reconfigurable to anticipate better on performing different tasks. MP-SoCs with more and more processing cores are being widely used nowadays [Jong 17]. Therefore, methods to enhance the dependability of an MP-SoC with billions of transistors are attracting increased research interests.

1.2 DEPENDABILITY CHALLENGES OF MODERN ICS

Aggressive scaling of transistor continues to provide higher performance, in addition to lower power and cost. Moore's law shows that the transistor density of ICs (integrated circuits) will roughly double every two years. This is due to the innovations in process technology and devices, as shown in Figure 1.2 [Holt 16]. Process technology transitions have changed from bipolar to MOSFETs, to CMOS, to voltage scaling, and power-efficient scaling. This in addition to using tungsten plugs, trench isolation, strained silicon, high-k/metal gates, FinFETs and multi-gate FinFETs. The introduction of strained silicon has improved the drive current, while high-k/metal gates reduced current leakage and heat. Finally, the FinFET addressed limitations with regard to electrostatics and short-channel effects [Holt 16].

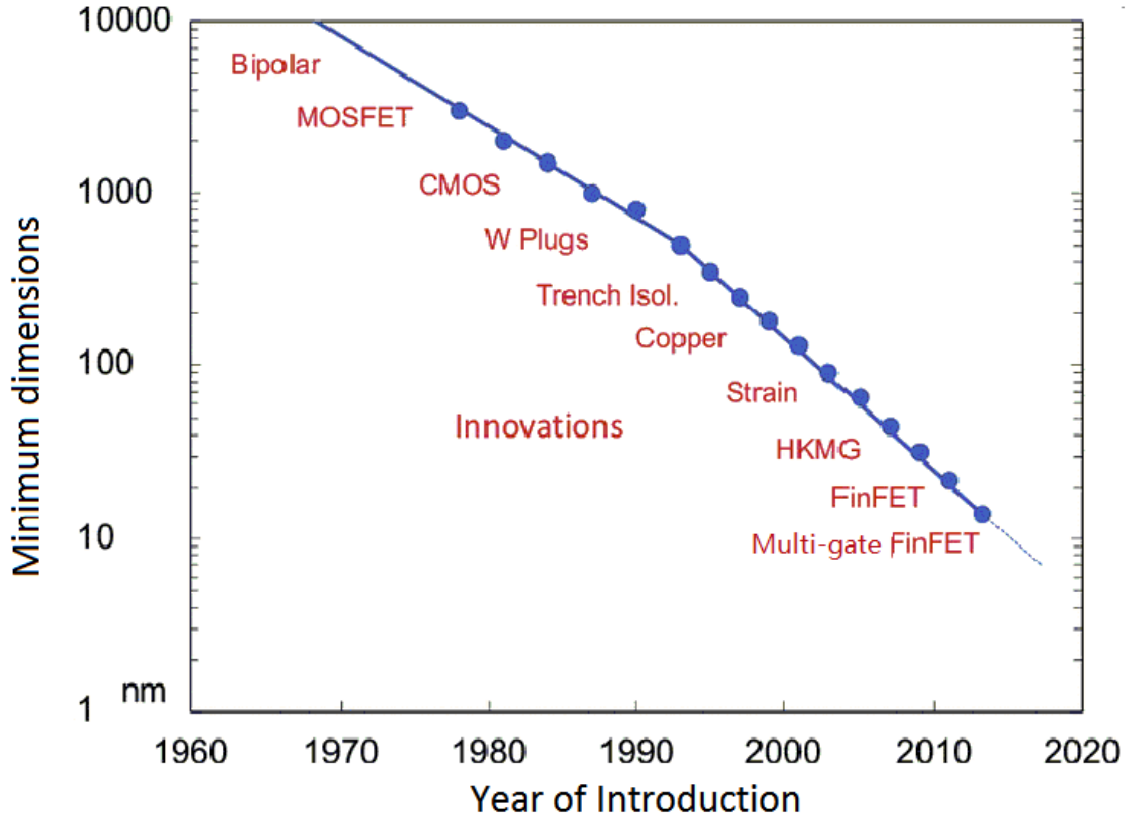


Figure 1.2: Moore's law with year of process innovations and technology nodes [Holt 16].

1.3 Dependable System Design

Unfortunately, the down-scaling of transistor size is negatively impacting degradation and aging of devices, circuits and associated electronic systems. The major aging mechanisms of microelectronic MOS devices are negative bias temperature instability (NBTI), the time-dependent dielectric breakdown (TDDB), hot carrier injection (HCI) and electro-migration (EM).

NBTI is mainly observed in p-channel MOS transistors operating with a negative gate-to-source voltage. It can result to an increase in threshold voltage (V_{TH}) and a decrease of the drain current [Paul 05].

HCI occurs when charge carriers (electrons or holes) are trapped in the gate dielectric; this leads to a permanent change of the transistor characteristics, such as a shift of the threshold voltage [Mari 13].

TDDB occurs in the thin dielectric layer between the control gate and the conducting channel of the transistor [Bern 06]. Consequently, the electrical property of the layer will gradually change until a hard breakdown takes place.

EM is found in interconnects due to the aggressive interconnect scaling, which will lead to time-related faults or permanent open wire faults [Scor 91].

In practice, NBTI is the dominant degradation mechanism. In this thesis we did not include HCI, TDDB and EM in our research because of their small effect in the CMOS technologies used. These mechanisms will be briefly discussed in Chapter 2.

The above aging mechanisms will affect integrated-system dependability, which can result in faults and failures. As illustrated in Figure 1.3, based on the dependability attributes [Buja 06], a dependable system can have either a non-redundant or redundant design [Aviz 04]. The non-redundant systems only rely on fault avoidance to prevent defects and faults from occurring by e.g. the usage of more mature and reliable semiconductor processing technologies for IC fabrication. Redundant systems employ in addition to fault avoidance, techniques to improve the reliability and availability resulting in extra costs and complexity [Grad 16].

Redundant systems often are fault-tolerant designs, which strive for maintaining system functions and avoid system failures. In essence, they must be able to continue working to a level of satisfaction even in the presence of faults.

The fault-tolerant feature is basically achieved through redundancy, particularly dual or triple modular redundancy (TMR) [Aran 17], which belong to the so-called fault-masking redundancy [Müll 11]. It is a technique to ignore faults by a sort of voting protocol where in the case the main and backups do not provide the same results, the flawed output is ignored. There are special software and instrumentation packages designed into fault-tolerant systems. Typically, components have multiple backups and are separated into smaller "segments" that act in case of a fault, and extra redundancy is built into all interconnections [Vyto 92].

Different from masking redundancy, the most recent fault management strategy is to use dynamic redundancy. In such a system, the availability of the computational resources and the varying requirements with regard to reliability and performance is being considered. Therefore, it is more flexible in redundancy allocation, e.g. using on-the-fly reconfiguration of resources [Wang 07]. In addition, fault forecasting, failure prognostics and fault prediction models can be employed, meaning to estimate and determine whether faults are likely to take place in the future. This can be accomplished by monitoring critical parameters such as temperature, current or voltage of the parts of interest [Ozce 17], [Carv 15]; this is also referred to as the health monitoring and prognostics (HMP) technique.

Besides fault-tolerant systems design for the dependability, adaptive protection such as adaptive dynamic voltage/frequency scaling (DVFS) techniques can be implemented to reduce the likelihood of hidden failures [Bern 12], [Pfei 14], for enhancing the dependability of target systems.

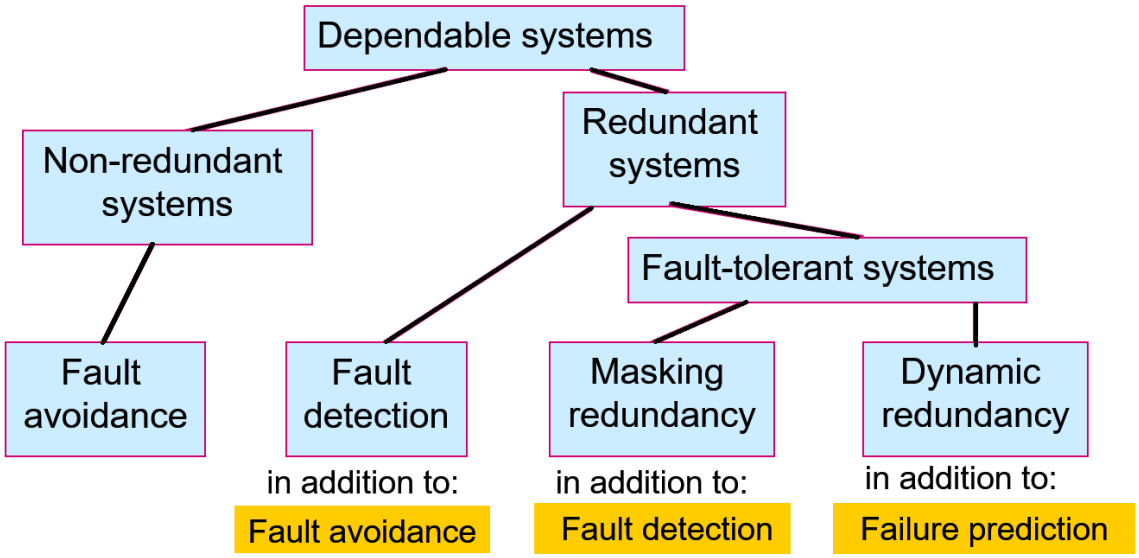


Figure 1.3: Dependable system designs based on different fault/failure management strategies [Aviz 04].

1.4 Health Monitoring and Life-time Prognostics

Health monitoring and life-time prognostics is a proactive method to identify degradations and to determine the moment of likely failure occurrence [Kelk 97]. Health-

monitoring can provide information for maintenance and potential replacement such that a failure can be prevented in advance. Life-time prognostics provides, combined with relevant information from health monitors, a prediction on the remaining life-time (RLP) of the system. The health monitoring and prognostics offer the possibility of maintenance and replacement based on actual demands in a dependable system, and hence, the possibility of significant cost savings [Bagu 08].

Non-invasive health monitoring, e.g. temperature, highlights not affecting the integrity and function of the potential faulty circuit. The major challenge of a health-monitoring design is identification and location of key monitoring parameters as these will provide crucial health information of the target system. For instance, canary circuits [Shah 08] incorporated into integrated circuits can detect aging-induced performance degradation in a predictive manner, because they always fail earlier than normal operating circuits. In contrast, our voltage monitor [Wan 14] provides analogue measurement information on the health of the core (e.g. power dissipation), and a quiescent current (I_{DDQ}) monitor [Kunh 07] can potentially indicate the *functional* degradation of the processor due to an aging effect. This could be combined later-on with data from a processor-workload monitor [Bara 13]. Delay monitors [Vald 13], in combination with voltage monitors, can show the system (frequency) operating degradation that can be caused by aging behaviour. These monitors are typically for observing the stress experienced in field operations.

The prognostics procedure is based upon the employment of the (remaining) life-time prediction, which can provide a warning of failures sufficiently early to be useful for available pre-maintenance actions. This is also referred to as Condition-Based Maintenance (CBM) [Rao 96]. Many models for RLP can be found, e.g. Lu and Meeker [Lu 93] proposed a model to predict the remaining lifetime of the device. Later-on, Wang [Wang 00] proposed an optimal critical level and a monitoring-intervals determination method. The health-monitoring based life-time prediction can not only increase system availability, but also reduces the cost for normal scheduled maintenance activities [Kim 17].

The development of the life-time prediction based maintenance philosophy is of major interest across almost all industrial environments in which the availability,

reliability and performance of machinery is critical [Lee 15]. However, developing such capabilities is a significant technical challenge. One reason is that the natural variation between different health monitors can be so significant that it becomes difficult to have error-free monitoring results. In addition, the target dependability of a monitoring system for degradation is a statistical process which can vary between different systems. Furthermore, since health monitoring usually employs more than one parameter for the system, how to process the multi-dimensional monitoring parameters to get an accurate prediction model will be a considerable challenge. This process is known as sensor data fusion [Velá 13] and is rapidly gaining interest.

1.5 RESEARCH PROBLEM STATEMENTS

The research in this thesis has been performed in the frame work of the Sensor Technology Applied in Reconfigurable Systems (STARS) project [Star 10]. The STARS project aimed to develop sensors and sensor networks based on a scalable, dependable and reconfigurable multi-processor system applied in the context of the security domain [Star 10], [Dech 13]. One example of the applications used in the STARS project is the latest antenna system for wireless telecommunications [Dech13]. The radar in the communication system offers some degree of reconfigurability where the beam can be adjusted very quickly with regard to the amount of information needed. Other application areas can be applied as well, for instance as described in [Kerk 12].

Traditionally, static reliability management for the processors in MP-SoCs are often seen to meet reliability specifications, e.g. (stuck-at, open) faults detection during design-time [Braa11]. In our reconfigurable MP-SoCs, however, features change after production and even during run-time, in fractions of seconds [Kerk 12]. This indicates our system should be self-aware of its reliability, and should possess the capability of dynamically adjusting the operating conditions of the MP-SoC. This approach has been referred to as Dynamic Reliability Management (DRM) [Karl 08], [Srin 04].

Our special concern in this thesis is to develop approaches for dependability enhancement for such DRM systems to be used in safety-critical parts/functions in STARS applications. The most important attributes for dependability in our case are reliability, MDT and maintainability. For instance, in the Unmanned Aerial Vehicle (UAV) application [Star 11], [Dixo 06], the (control) communication is the most

demanding dependability requirement, as basically no mean down time should be allowed. This would result in temporarily loss of control of the UAVs; for example, a few micro-seconds unavailability of the communication from the on-board processor may have catastrophic consequences. The importance of this item was later stressed unexpectedly by the "Sentinel, RQ-170" accident in Afghanistan, in which case the UAV landed in Iran because its guidance system was quickly altered [Star 11]. It illustrated why a full-time operational service (no down time) is required for our systems in STARS. The health monitoring and prognostics based on the monitoring results, as a proactive method, has been chosen for enhancing the dependability of our MP-SoCs.

This leads to the main research question this thesis is going to tackle: how to integrate different health monitoring and prognostics techniques for enhancing the dependability of our MP-SoCs? This can be formulated more specifically by the following research questions:

1) Traditional dependability solutions employ typically worst-case designs and the fault-detection based testing method, which helps the system to reach a certain dependability level. Which types of dependability measures are necessary for our reconfigurable MP-SoC applications? (Chapter 2)

2) Since our MP-SoCs are used for the safety-critical applications with zero MDT, which types of circuits can be used for monitoring the dependability of our MP-SoC to secure a zero mean down time and long life-time? Which kind of parameters should be monitored and at which locations to improve the life-time prognostics model? (Chapter 3 and Chapter 4)

3) Since dependability is a life-long topic, how to implement and evaluate the developed monitoring circuits? Is it possible to observe the aging degradation of our MP-SoCs via these developed techniques? (Chapter 5)

4) After obtaining the health monitoring information, how to employ life-time prognostics for our MP-SoCs? What kind of models can be utilized for life-time prediction of the system, for maintenance purposes via isolation and spare cores repair? How can we validate the remaining life-time? (Chapter 6)

1.6 OUTLINE OF THE THESIS

The remainder of this thesis has been organized as follows:

In Chapter 2, the dependability challenges in our target MP-SoC design will be elaborated. The required basic background of aging mechanisms and dependable system design in many-processor SoCs will be provided. Our target MP-SoCs will be introduced, and the related works on dependability enhancement techniques will be discussed.

In Chapter 3, an embedded health-monitoring infrastructure for our target dependable MP-SoC will be proposed. An all-in-one health monitor will be designed and evaluated, which is capable of carrying out voltage and temperature measurements as well as delay-time monitoring. It satisfies the dependability requirements of a DRM system. Moreover, simulation results of its behaviour and power dissipation will be discussed.

Chapter 4 will propose an in-situ health-monitoring technique of the performance degradation detection for a VLIW processor, the Xentium®. The functional software-based aging detection program, including the delay monitoring, I_{DDQ} monitoring as well as unit-based I_{DDT} monitoring will be presented and explained.

The functionality of all the designed and implemented hardware as well as the developed (monitoring) software program will be validated in Chapter 5. The setup of the accelerated testing experiment will be presented which includes the measurement results for 48 Xentium processors with regard to changes in the delay, I_{DDQ} and I_{DDT}. The correlation coefficients between their results are modelled and provided.

In chapter 6, a genetic-algorithm based degradation optimization model will be introduced, and the reason why an alternative remaining lifetime prediction method based on the I_{DDX} monitoring results for the Xentium processor is used. The developed algorithm for the remaining lifetime prediction will be explained and the statistical values will be compared after applying it to the I_{DDQ} and I_{DDT} monitoring results.

Finally, Chapter 7 answers all the research questions as stated in Chapter 1, and the overall conclusions of our research are provided; also several suggestions for future work are given.

REFERENCES

[Aran 17] L. A. Aranda, P. Reviriego and J. A. Maestro, "A Comparison of Dual Modular Redundancy and Concurrent Error Detection in Finite Impulse Response (FIR) Filters Implemented in SRAM-based FPGAs through Fault Injection," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 99, pp. 1-5, 2017.

[Aviz 01] A. Avizienis, J-C. Laprie, and B. Randell, "Fundamental concepts of dependability," in Laboratory for Analysis and Architecture of Systems (LAAS-CNRS) Technical Report no. 01-145, Apr. 2001.

[Aviz 04] A. Avizienis, J. C. Laprie, B. Randell and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," in IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 1, pp. 11-33, Jan. 2004.

[Bagu 08] Y. G. Bagul, I. Zeid and S. V. Kamarthi, "A Framework for Prognostics and Health Management of Electronic Systems," in IEEE Aerospace Conference, Big Sky, MT, pp. 1-9, 2008.

[Bara 13] R. Baranowskia*, et al.*, "Synthesis of Workload Monitors for On-Line Stress Prediction," in IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), New York City, NY, USA, pp. 137-142, 2013.

[Bern 06] J.B. Bernstein, M. Gurfinkel, X. Li, J. Walters, et al., "Electronic circuit reliability modeling," Microelectronics Reliability, Vol. 46, No. 12, pp. 1957–1979, Dec. 2006.

[Bern 12] E. E. Bernabeu, J. S. Thorp, and V. Centeno, "Methodology for a security/dependability adaptive protection scheme based on data mining," in IEEE Transactions on Power Delivery, vol. 27, pp. 104-111, 2012.

[Braa 11] T. D. ter Braak, H. A. Toersche, A. B. J. Kokkeler and G. J. M. Smit, "Adaptive resource allocation for streaming applications," in Proceedings of the International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation, Samos, Greece, pp. 388-395, 2011.

[Buja 06] G. Buja and R. Menis, “Conceptual frameworks for dependability and safety of a system,” in Proc. IEEE Int. Symp. Power Electronics, Electrical Drives, Automation and Motion, pp. 44-49, May 2006.

[Carv 15] M. De Carvalho, “Innovative Techniques for Testing and Diagnosing SoCs,” Doctoral dissertation, Politecnico di Torino, Italy, 2015.

[Dech 13] F. Dechesne, M. Warnier, and J. van den Hoven, “Ethical requirements for reconfigurable sensor technology: a challenge for value sensitive design,” in Ethics and Information Technology, vol. 15, pp. 173-181, September 2013.

[Dixo 06] S. R. Dixon and C. D. Wickens, “Automation Reliability in Unmanned Aerial Vehicle Control: A Reliance-Compliance Model of Automation Dependence in High Workload,” in Human Factors, vol. 48, pp. 474-486, 2006.

[Grad 16] E. Grade, A. Hayek and J. Börcsök, “Implementation of a fault-tolerant system using safety-related Xilinx tools conforming to the standard IEC 61508,” in International Conference on System Reliability and Science (ICSRS), Paris, pp. 78-83, 2016.

[Holt 16] W. M. Holt, “Moore's law: A path going forward,” in IEEE International Solid-State Circuits Conference (ISSCC), pp. 8-13, 2016.

[Home 17] Homeland Security, “Budget In Brief: Fiscal Year 2017,” pp. 1-2. Retrieved 23 March 2017: https://www.dhs.gov/sites/default/files/publications/FY2017_ BIB-MASTER.pdf

[Inte 09] “International Technology Roadmap from Semiconductors,” 2009 Edition (Design). http://www.itrs2.net/

[Jong 17] R. Jongerius, A. Anghel, G. Dittmann, et al., “Analytic multi-core processor model for fast design-space exploration,” in IEEE Transactions on Computers, vol. 99, pp. 1-16, 2017.

[Karl 08] E. Karl, D. Blaauw, D. Sylvester and T. Mudge, “Multi-Mechanism Reliability Modeling and Management in Dynamic Systems,” in IEEE Transactions on VLSI Systems, pp. 476-487, April 2008.

[Kelk 97] N. Kelkar, D. Dasgupta, M. Pecht, et al., “Smart Electronic Systems for Condition-Based Health Management,” in Quality and Reliability Engineering International, Vol. 13, pp. 3-7, 1997.

[Kerk 09] H. G. Kerkhoff, “Dependable reconfigurable multi-sensor poles for security,” in 15th IEEE International Mixed-Signals, Sensors, and Systems Test Workshop (IMS3TW), ISBN 978-1-4244-4618-6, Scottsdale (AZ), USA, pp. 1-6, June 2009.

[Kerk 12] H. G. Kerkhoff and Y. Zhao, “The design of dependable flexible multi-sensory System-on-Chips for security applications,” in IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), Tallinn, Estonia, pp. 133-138, 2012.

[Kim 17] N.-H. Kim, D. An, and J.-H. Choi, “Introduction in Prognostics and Health Management of Engineering Systems,” Springer Publishing Press, pp. 1-24, 2017.

[Kuma 80] A. Kumar, M. Agarwal, "A Review of Standby Redundant Systems", in IEEE Transactions on Reliability, vol. R-29, no. 4, pp. 290-294, 1980.

[Kunh 07] K. Kunhyuk, K. Keejong, et al., “Characterization and Estimation of Circuit Reliability Degradation under NBTI using On-Line I_{DDQ} Measurement,” in 44th ACM/IEEE Design Automation Conference (DAC), pp. 358-363，2007.

[Lee 15] J. Lee, B. Bagheri, and H.-A. Kao, “A cyber-physical systems architecture for industry 4.0-based manufacturing systems,” in Manufacturing Letters, vol. 3, pp. 18-23, 2015.

[Lu 93] C. J. Lu and W. Q. Meeker, “Using Degradation Measures to Estimate a Time-to-Failure Distribution,” in Technometrics, vol. 35, pp. 161-174, 1993.

[Mari 13] E. Maricau and G. Gielen, “Analog IC Reliability in Nanometer CMOS,” Springer Publishing Press, ISBN 978-1-4614-6162-3, 2013.

[Müll 11] N. Müllner and O. Theel, “The Degree of Masking Fault Tolerance vs. Temporal Redundancy,” in IEEE Workshops of International Conference on Advanced Information Networking and Applications, Biopolis, pp. 21-28, 2011.

[Paul 05] B. C. Paul, K. Kunhyuk, et al., “Impact of NBTI on the temporal performance degradation of digital circuits,” in IEEE Electron Device Letters, vol. 26, pp. 560-562, 2005.

[Pfei 14] P. Pfeifer, Z. Pliva, P. Weckx and B. Kaczer, “On reliability enhancement using adaptive core voltage scaling and variations on nanoscale FPGAs,” in 15th Latin American Test Workshop (LATW), Fortaleza, Brazil, pp. 1-4, 2014.

[Rao 96] B.K.N. Rao, “Handbook of Condition Monitoring,” Elsevier Science Publishers Ltd., Oxford, 1996.

[Scor 91] A. Scorzoni, B. Neri, C. Caprile, and F. Fantini, “Electromigration in thin-film interconnection lines: Models, methods and results,” in Materials Science Reports, Vol. 7, pp. 143-220, 1991.

[Shah 08] N. Shah, R. Samanta, M. Zhang, J. Hu, and D. Walker, “Built-In Proactive Tuning System for Circuit Aging Resilience,” in IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems (DFT), pp. 96-104, 2008.

[Star 10] “STARS: Sensor Technology Applied in Reconfigurable systems”, 2010. http://cas.et.tudelft.nl/Research/project.php?id=33

[Star 11] B. Starr, “Drone that crashed in Iran was on CIA recon mission, officials say,” in CNN news, 2011, https://edition.cnn.com/2011/12/06/world/meast/us-iran-drone/index.html

[Srin 04] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, “The case for lifetime reliability-aware microprocessors,” in Proceedings of 31st Annual International Symposium on Computer Architecture, pp. 276–287, 2004.

[Tamb 14] L. A. Tambara, F. L. Kastensmidt, P. Rech and C. Frost, “Decreasing FIT with diverse triple modular redundancy in SRAM-based FPGAs,” in IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), Amsterdam, pp. 153-158, 2014.

[Ozce 17] B. Ozcelik and C. Yilmaz, “Seer: A Lightweight Online Failure Prediction Approach,” in IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), Turin, Italy, pp. 624-625, 2017.

[Vald 13] M. D. Valdes-Pena, J. Fernandez Freijedo, M. J. Moure Rodriguez, et al., "Design and Validation of Configurable Online Aging Sensors in Nanometer-Scale FPGAs," in IEEE Transactions on Nanotechnology, vol. 12, pp. 508-517, 2013.

[Velá 13] J. M. R. Velázquez, F. Mailly and P. Nouet, "System-level simulations of multi-sensor systems and data fusion algorithms," in Microsystem Technologies, pp. 1-10, 2013.

[Vyto 92] J. Vytopil, ed., "Formal Techniques in Real-Time and Fault-Tolerant Systems," in Lecture Notes in Computer Science, Nijmegen, the Netherlands, vol. 571, January 8-10, 1992.

[Wan 14] J. Wan and H. G. Kerkhoff, "An embedded offset and gain instrument for OpAmp IPs," in Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1-4, 2014.

[Wang 00] W. Wang, "A model to determine the optimal critical level and the monitoring intervals in condition-based maintenance," in International Journal of Production Research, vol. 38, pp. 1425-1436, 2000.

[Wang 07] S. Wang, L. Wang and Faquir Jain, "Dynamic redundancy allocation for reliable and high-performance nanocomputing," in IEEE International Symposium on Nanoscale Architectures, San Jose (CA), USA, pp. 1-6, 2007.

[Whit 08] M. White and Y. Chen. "Scaled CMOS Technology Reliability Users Guide," in NASA Electronic Parts and Packaging (NEPP) Program JPL, 2008.

Chapter 2

BACKGROUND AND RELATED WORK

ABSTRACT *– This chapter will cover the terminology and basics of aging mechanisms and their impact on the dependability of advanced IC systems, especially on many-processor system-on-chips (MP-SoCs). Two dependability architectures of MP-SoCs with enhanced dependability features will be presented, which are the scan-based logic BISTR for dependability and prognostic health monitoring for dependability. Based on the target application areas of the MP-SoCs, dependability requirements will be explained. Dependability is jeopardized by aging, and several aging mechanisms are briefly discussed. Physical aging can be emulated by means of accelerated stress testing of which the basic principles are introduced. Next, the basics of existing dependability-enhancement techniques of employing health monitoring and lifetime prediction are briefly discussed.*

Parts of this chapter have been published as paper titled "Design of an Embedded Health Monitoring Infrastructure for Accessing Multi-Processor SoC Degradation," in the Euromicro Conference on Digital System Design, 2014 [Zhao 14], and "Highly Dependable Multi-processor SoCs Employing Lifetime Prediction Based on Health Monitors," in IEEE 25th Asian Test Symposium (ATS), 2016 [Zhao 16].

2.1 INTRODUCTION

Our past developments in dependable chip design have dealt with dependable digital scan-based Built-in-Self-Test-and-Repair (BISTR) of homogeneous many-processor SoCs [Kerk 10a] and dependable analogue/mixed-signal front-ends (AFE) and mixed-signal SoCs [Kerk 10b], [Khan 11]. This thesis will deal with the design of dependable, complex System-on-Chips based on Prognostics Health Monitoring (PHM).

Because of target applications in space and military [Dixo 06], these systems must feature a high degree of scalability, reconfigurability and dependability. The last item includes attributes like high reliability (long lifetime expectancy under harsh conditions), full (100%) availability (no MDT), and maintainability (able to repair) in the case of safety-critical systems.

This chapter is organized as follows. In section 2.2, several aging mechanisms are explained. Accelerated testing (AT) for aging-related reliability assessment is introduced in section 2.3. In section 2.4, the scan-based BISTR dependability architecture of our homogeneous many-processor SoC is discussed. Subsequently the PHM architecture for dependability is introduced in section 2.5. Here, the existing techniques for achieving high dependability by usage of embedded health monitors (also referred to as embedded instruments (EIs)) around processor cores with new IJTAG (IEEE 1687) compatibility [Ieee 16] are discussed. Then some basics on health monitors and the remaining lifetime prediction are reviewed in sections 2.6 and 2.7 respectively. Finally, the conclusions are presented in section 2.8.

2.2 AGING MECHANISMS

The effort to construct smaller transistors has resulted in an extremely high level of device density and computational performance improvement. However, the down-scaling of transistor parameters is negatively impacting degradation and wear-out of devices,

circuits and associated electronic systems during their operational life time (aging). This has resulted in serious dependability challenges. [Axer 11]

The major aging mechanisms of microelectronic MOS devices are negative/positive bias temperature instability (NBTI/PBTI), gate oxide breakdown, or the time-dependent dielectric breakdown (TDDB), hot carrier injection (HCI) and electro-migration (EM). These mechanisms can be seen in Figure 2.1; it is recognized that NBTI is the dominant factor [Kean 10] for aging in Intel Pentium® P4 to Dual core Itanium® 2 processors, or its technology nodes from 140 nm to 45 nm. But also in current 7 nm FinFETs, aging remains an issue because of NBTI [Pari 18]. These aging mechanisms are briefly reviewed below.

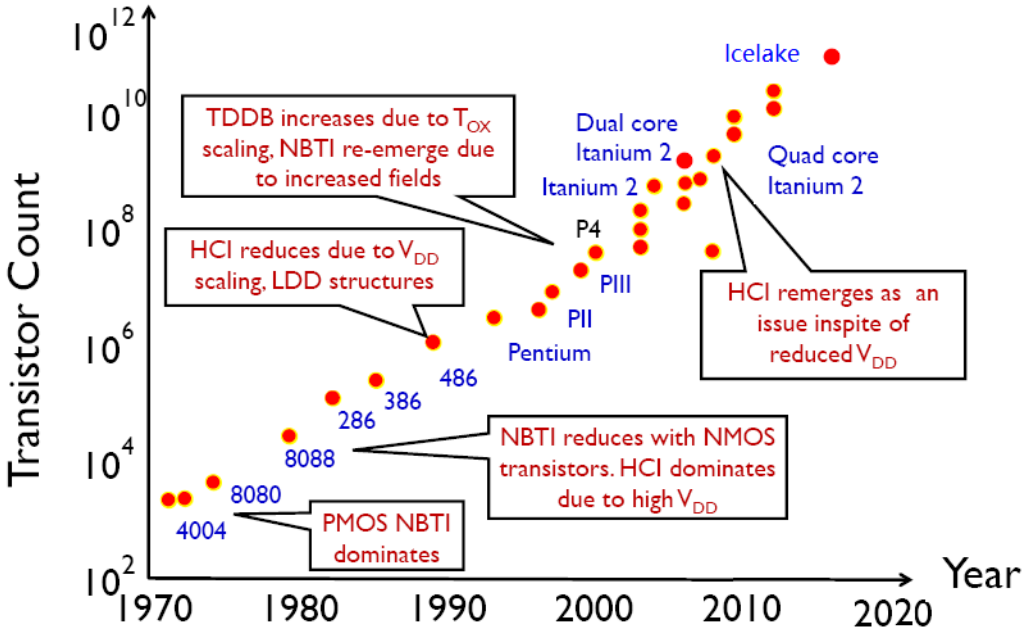


Figure 2.1: Different aging mechanisms affecting different technology nodes over time [Kean 10].

Bias Temperature Instability

Negative Bias Temperature Instability (NBTI) is mainly observed in p-channel MOS transistors (or FinFETs) operating with a negative gate-to-source voltage (V_{GS}). Normally it is caused by two tightly coupled mechanisms: interface-state generation and holes trapping in the oxide traps [Gras 09]. They can be accelerated by elevated temperature and voltage levels, resulting in an increase in threshold voltage (V_{TH}) and a

decrease of the drain current and transconductance [Paul 05], [Pari 18]. NBTI has become a serious CMOS (including FinFETs) reliability concern, because of its impact on the critical parameters of the PMOS transistor. Positive Bias Temperature Instability (PBTI), on the other hand, is observed in n-channel MOS transistors if V_{GS} is positively biased. It has a similar mechanism as NBTI that can negatively impact the reliability. In practice, NBTI is the dominant degradation mechanism in this thesis. Due to the BTI effect, delay faults can be introduced since it will increase the threshold voltage (V_{TH}).

Hot Carrier Injection

Hot carrier injection (HCI) occurs when charge carriers (electrons or holes) are trapped in the gate dielectric; this leads to a permanent change of the transistor characteristics, such as a shift of the threshold voltage because of interface-state generation [Mari 13]. The HCI is strongly related to the internal electric field of a transistor. As the down-scaling of the supply voltage is far slower than the shrinking of the channel length and oxide thickness, the internal electric field continuously increases, thereby worsening the reliability issues.

Time-Dependent Dielectric Breakdown

Time-Dependent Dielectric Breakdown (TDDB) occurs in the thin dielectric layer between the control gate and the conducting channel of the transistor [Bern 06]. Consequently, the electrical property of the layer will gradually change until a hard breakdown takes place. The occurrence of the TDDB is proportional to the current density flowing through the oxide layer, which is accelerated by the increase of supply voltage and temperature. It has not been taken into consideration in our research.

Electro-Migration

Electro-Migration (EM) is found in interconnects due to the aggressive interconnect scaling. High resistances or broken wires can result from the EM effect, which will lead to time-related faults or permanent open wire faults [Scor 91]. New materials which are

being used for interconnection can enhance this effect in the future. We did not include EM in our research because of its small effect in the CMOS technologies used.

There are other mechanisms that can cause failures in integrated systems, e.g. intermittent (transient) faults, which are usually caused by internal parameter degradation or material instability; a gate-dielectric soft breakdown is an example of an intermittent fault [Mahe 03]. Intermittent faults often precede the occurrence of permanent faults as the degradation progresses. Transient faults are also known as random faults. They usually occur as a result of temporary environmental conditions, such as temperature variations, the effect of high-energy particles or electromagnetic interference [Mahe 03].

2.3 ACCELERATED TESTING FOR AGING ASSESSMENT

In order to observe the aging effect of a system in a relatively short time, Accelerated Testing (AT) will be conducted for predicting the reliability under normal operating conditions [Saha 11].

AT is applied by manufacturing industries to assess or demonstrate component and subsystem reliability, to certify components, and to detect failure modes in order to be corrected [Rahi 07]. With the requirement for rapid product development, AT has become increasingly important because of fast changing technologies, more complicated products with more components, and higher customer expectations for a better reliability.

There are complex practical and statistical models involved in accelerating the deterioration of a target system over time that can fail in different ways [Saha 11]. Generally, accelerating stressor variables (e.g., workload, temperature, voltage) are extrapolated via a physically reasonable statistical model, to obtain estimates of the life-time or long-term performance at lower, normal levels of the accelerating stressor variable(s).

Two typical types of AT are often carried out for the reliability assessment in the semiconductor area. The first one is the High Temperature Operating (Bias) Life (HTOL) [Esco 06] test, sometimes referred to as the burn-in test. This is a well-known method to weed out infant mortality failures. The second one is the Power Temperature Cycling

(PTC) test [Jede 11]. The accelerating variables are often according to the JEDEC standards [Jede 10], [Jede 11].

The AT results are used to estimate the expected (remaining) lifetime of the system under normal operating conditions [Saha 11]. In our research, the HTOL and temperature cycling test will be executed for our health monitors. The setup and related test results will be described in Chapter 5.

2.4 THE SCAN-BASED BISTR FOR DEPENDABLE MP-SOCS

Nowadays, the technological advances have enabled the integration of a significant number of processors into a single silicon die, which is known as the many-processor or multi-processor system-on-chip (MP-SoCs) [Fu 14]. As a result of its capability of parallel computing and multi-tasking, the application of MP-SoCs can be found in space exploration systems [Pisc 12], military systems [Dixo 06], communication systems [Shan 14], [Sepu 12] and industry [Bork 07]. The dependability challenge has become imminent because of the technological and complexity advances and strict timing schedule of multi/many-core interaction [Axer 11]. However, the reconfigurable architecture of MP-SoCs makes it possible to use fault detection, failure prediction and resource remapping techniques to enhance the system dependability.

Within the CRISP project [Zhan 09], a homogeneous MP-SoC containing nine-processor cores (Xentium®) has been implemented and tested in 90 nm TSMC CMOS technology [Kuik 08], [Zhan 11]. It is shown in the insert of Figure 2.2. This so-called reconfigurable fabric device (RFD) design has been enhanced with an on-chip dependability manager (DM), which under command of an ARM926-based General Purpose Device (GPD) can generate and multicast deterministic scan-based test vectors for the Xentium processor core [Reco 11]. For communication, a packet-switched Network-on-Chip (NoC) with routers (R) was employed [Wolk 09], including network interfaces (NI).

The Xentium processor is a Very Large Instruction Word (VLIW) processor made in UMC 90 nm CMOS technology. A photomicrograph of the Xentium is shown in Figure 2.3. It has a silicon area of 1.2 mm^2 and runs at a clock frequency of 200 MHz. This processor core has been developed as part of the RFD depicted in Figure 2.2. The Xentiums are interconnected by a NoC. Each single Xentium is able to connect via the NIs to the adjacent routers of the NoC; they can also be connected to more conventional bus architectures (e.g. Amba) to communicate with other required peripherals.

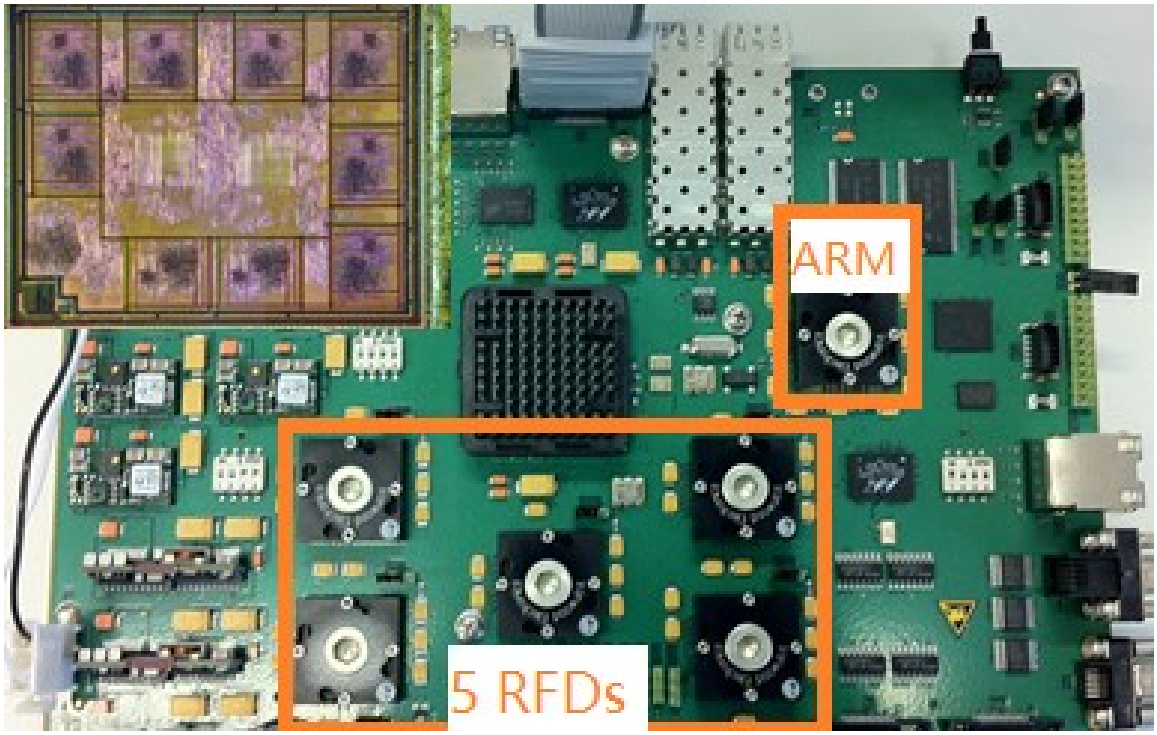


Figure 2.2: Set-up of the CRISP MP-SoC system (5 RFDs) at board level. The inset shows the photomicrograph of the RFD consisting of 9 Xentium processors [Reco 11]. The ARM-based general-purpose processor can be seen at the right middle.

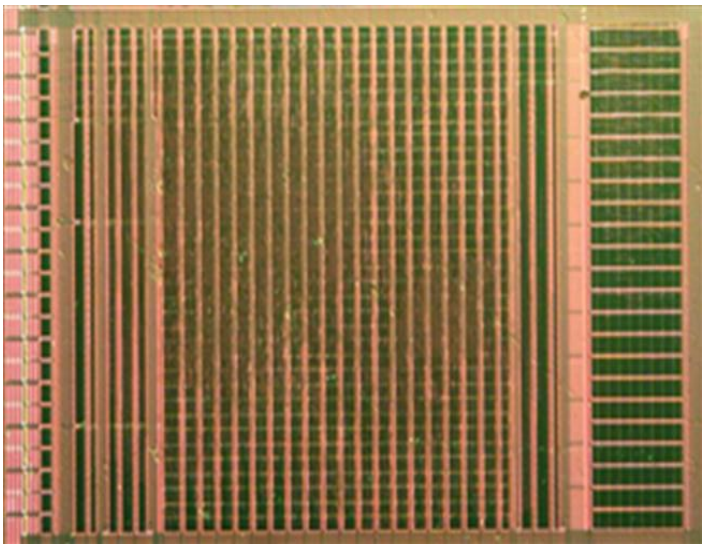


Figure 2.3: Photomicrograph of the standard cell Xentium® processor core [Reco 11]. (Courtesy of Recore Systems)

This MP-SoC (RFD) is an ultra-low power digital signal processing (DSP) system designed for high-performance computing in automotive as well as space and military applications, e.g. a global navigation satellite system (GNSS) and a beam former (BF) [Zhan 09]. The illustration of main dependability attributes of this chip are based on the so-called "mailbox" application and is shown in Table 2.1. The key feature of this approach regarding dependability is that if a Xentium core is found faulty by the on-chip controlling dependability manager (DM) in the RFD, it is electronically quarantined and a spare (or not fully used) processor takes over its tasks via run-time mapping [Braa 16]. Its main disadvantage is that it reacts *after* a fault has occurred.

Table 2.1: Main dependability attributes of the MP-SoC as specified for performing the "mailbox" application [Zhan 11].

Attribute	Value / Range
Reliability (MTTF)	8760 hours
Non-availability (MDT)	Less than 96 ms (100 MHz clock)
Maintainability (MTTR)	Less than 10 ms

This dependability has been achieved by a BISTR approach via the on-chip DM [Zhan 11]. The motivation to introduce a DM in an MP-SoC is to build an on-chip dependability test environment in which the correctness of the internal processor cores/tiles of an MP-SoC can be verified. The DM has been designed and implemented as a stand-alone Infrastructural IP (IIP) for the dependability test of an MP-SoC. The DM consists of an automatic test-pattern generator (ATPG) for test-vector generation, a test-response evaluator for test-response evaluation (TRE). Furthermore, a finite-state machine (FSM) has been used for internal control and communication with special dependability software running on the GPD. The reseeding technique has been adopted in the design of the DM-TPG to achieve test-vector compression [Zhan 11].

The design has been optimized to cause as little interference as possible to normal system operations. This is achieved by testing processor cores while they are in idle state using the NoC segments unoccupied by user applications. However, it is not always possible to find unused NoC routes from the DM to the cores under test [Zhan 11].

For an MP-SoC, it is usually very difficult to physically repair a faulty core in the chip package in field. In that sense, there is no maintainability at the core level. At system level, an MP-SoC can be considered as a repairable system if the faulty cores can be detected and electronically isolated. The computing tasks can be remapped to fault-free processor cores [Zhan 11]. The MP-SoC is considered as functionally correct until the number of working cores drops below a threshold value K as described in a K-out-of-N: G system [Shao 91]. Parameter K is a fixed number determined by both application requirements and individual core performance and N denotes the total number of available processors in the SoC. In the CRISP case, a 6-out-of-9 system is shown as example in Figure 2.4. Based on the dependability test results, the MP-SoC can implement a remapping of the spare cores, i.e. S1, S2 and S3 in Figure 2.4. In this figure, a standby normally working redundant system is shown in Figure 2.4a, the dependability test is depicted in Figure 2.3b and application remapping is indicated in Figure 2.4c. Greyed areas denote the application load. W represents a working (operational) core, S standby cores and T are labelled cores being tested. The core with a cross is the one as being tested as faulty. In the remapped situation in Figure 2.4c, S3 has become the replaced core for W'3.

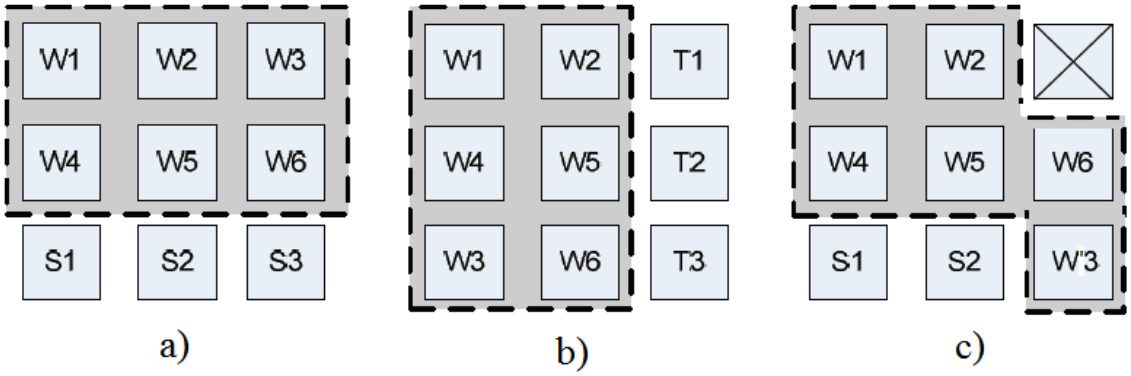


Figure 2.4: Dependability management and run-time re-mapping in scan-based BISTR MP-SoCs, a) a standby normally working redundant system, b) while executing the dependability test, and c) application of remapping [Zhan 11].

2.5 Prognostic Health-Monitoring for Dependable MP-SoCs

With the requirement of more powerful data-processing capabilities, and the availability of advanced IC design technology, more complex designs of MP-SoCs in terms of the number of processor cores have been proposed [Das 16], [Mcke 17], [Zhao 14].

For example, these MP-SoCs can support a wide range of internal voltages and frequencies, and are able to support dynamic voltage and frequency scaling (DVFS) to minimize the task-computation energy [Sing 13], [Dama 13]. This has motivated researchers in recent years to jointly optimize lifetime dependability by using intelligent task/core mappings [Huan 10]. The increasing number of cores (>64) also adds to more flexibility for system dependability management.

These MP-SoCs can be applied in a harsh environment for life- or mission-critical applications, such as automotive, military and aerospace [Star 10], [Das 16], [Mcke 17]. Different from desktop or normal applications, these devices have much more severe external stress conditions such as temperature, shock and radiation. For example, the transmission controller and wheel sensors in cars are required to work in an ambient

temperature of around 200 °C [Wats 15]. In aerospace applications, the control electronics must function correctly within a temperature range from -55 °C to 200 °C. Besides that, the requirement of the MDT of these systems is always close to 0 (a very high availability), since a few micro-seconds unavailability of the control from the processor could have catastrophic consequences. As such, reliability and availability are becoming major requirements in the approach of designing dependable MP-SoCs.

As an example, a heterogeneous MP-SoC, the so-called Moon IC is shown in Figure 2.5 [Reco 11]. The Moon IC contains a control-processor core ARM 926, an upgraded version of the Xentium® (more digital signal processing capability), three Montium® cores, a LEON core, and one ADC and associated peripherals.

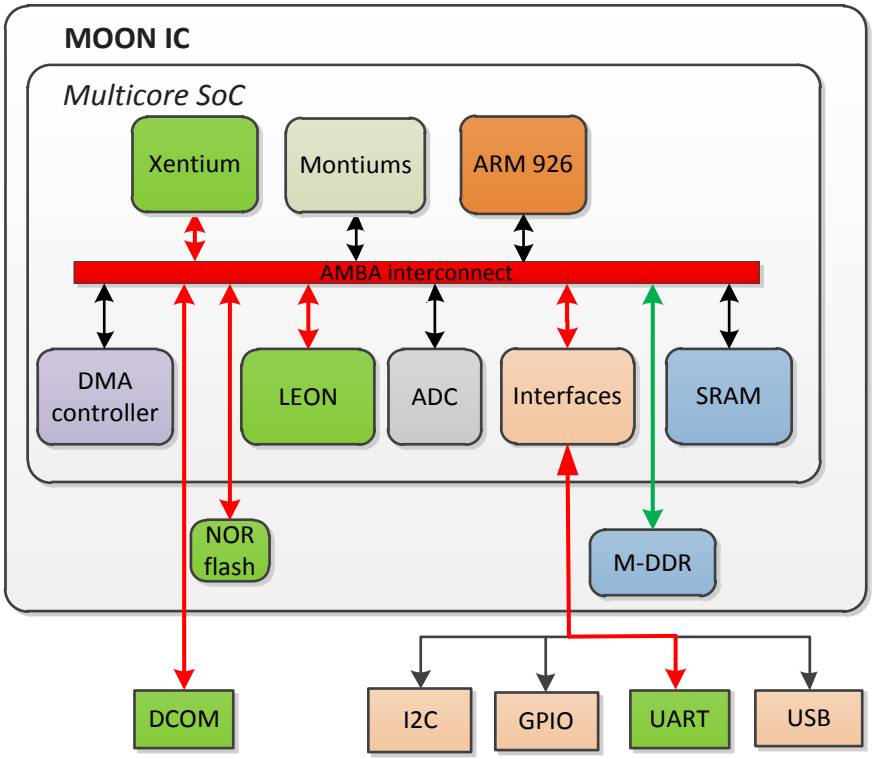


Figure 2.5: Block diagram of a heterogeneous MP-SoC, the Moon IC [Reco 11].

Compared to the previously discussed scan-based BISTR MP-SoC, the focus of dependability enhancement for the Xentium® processor core is similar. However, the approach of dependability for the Xentium® in the Moon IC will be different, since there should be no MDT for it while executing life-critical applications [Reco 11], [Star 10]. For instance, based on the application of UAV communication for this Moon IC in the STARS project [Kerk 12], the dependability attributes and associated metrics of the Xentium® while executing control and beamsteering are listed in Table 2.2.

It is required that potential failures of the Xentium will be predicted, and maintenance should be performed in advance before system failure. Therefore, one key new feature compared to the dependability approach of the scan-based BISTR MP-SoC is a 100% availability in the target applications, for the above-mentioned reasons.

Table 2.2: The dependability attributes and associated metrics of the control and beamsteering in the STARS project for a dependable MP-SoC [Kerk 12]

Dependability attributes	**Metrics**
Reliability	0.996 (1 yr.), 0.932 (20 yrs.).
Life time	20 years
Availability	100 %, MDT is 0 for the Control and Command part
Maintainability	MTTR, limited best case: 10 ms
Safety	100%

To achieve this, a *proactive* approach needs to be taken, and nowadays a prognostic health-monitoring (PHM) approach [Pech 09], [Zhao 14] becomes more and more promising. It typically includes making usage of (non-) invasive health monitors (HMs),

a HM communication network and application of embedded prognostics (remaining life-time prediction) software.

Based on the monitored health information of the processor cores in MP-SoCs in the field operation, with the development of degradations near a core, at some point based on the remaining lifetime prediction (RLP) result, an embedded control processor (e.g. ARM, Figure 2.5) will determine to take a repair action for the processor core via remapping [Ahon 11]. As alternative, cores can also be freed of low-priority tasks, and subsequently added to the pool of partly spare core resources (not required to be the same type). Lowering the local core clock frequencies or power-supplies (PVT) also provide many possibilities for reducing degradation [Kerk 12].

After the system receives the crucial health-monitoring info regarding the dependability of the system, the RLP can be estimated. It will be calculated from the present moment until the time the health-monitoring data reaches the pre-set repair threshold [Zhao 16].

Figure 2.6 shows a similar 6-out-of-9 system in a PHM dependable MP-SoC. Difference is that the MP-SoC can implement a remapping of the spare cores, i.e. S1, S2 and S3, based on the PHM results for each operational core. In this figure, a standby normally working redundant system is shown in Figure 2.6a, the PHM is depicted in Figure 2.6b (with W3 estimated to be a potential failure) and a possible application remapping is indicated in Figure 2.6c. It can be observed that another advantage compared to scan-based BISTR approach is that the PHM based cores will not use the NoC segments occupied by user applications while executing the HM operations.

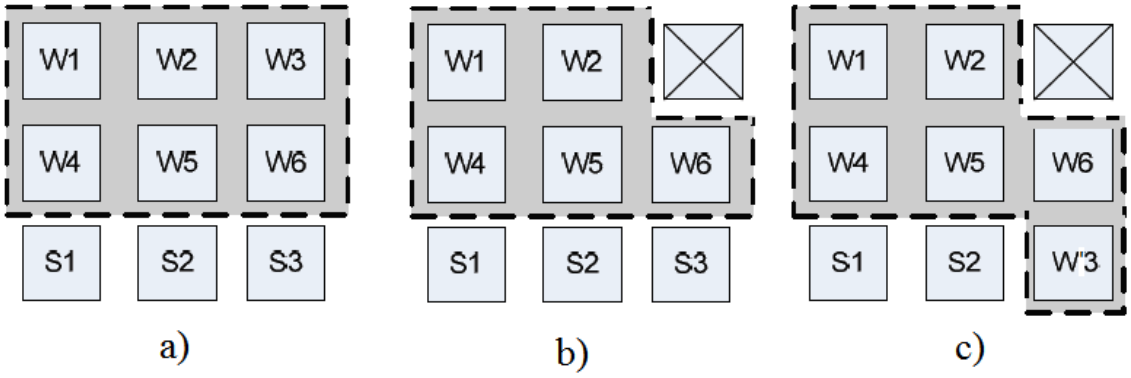


Figure 2.6: Dependability management and run-time re-mapping in PHM MP-SoCs, a) a standby normally working redundant system, b) while PHM estimates a potential failure of W3, and c) possible application remapping (S3 is used for W'3).

Regarding the availability and maintainability, the PHM MP-SoC will become much more robust, because the PHM will give a clue on the degradation of the system so that it can act before the occurrence of a failure, thus resulting in a full-time available system. The goals of this thesis are to develop new techniques for health-monitoring using the chosen monitors and to develop remaining lifetime prediction models, and thus be able to act before the impending failure occurs and hence improve the Quality-of-Service (QoS) of the application.

2.6 Health Monitoring for Dependable SoCs

Health monitoring is based on sensors or health monitors observing e.g. the environmental conditions (e.g. temperature, humidity, air pressure etc.) and the performance degradation of target embedded SoCs. These monitors will be located in an optimal way, and will also be integrated on-chip. The information from these monitors will be used to assess the probabilities of malfunctioning and predict the remaining lifetime of a processor core. Furthermore, the information will be used to reconfigure hardware (or recalibrate) to reduce the reliability hazard [Star 10].

Therefore, selecting the monitoring parameters that capture the present key health status, and relate these parameters to the aging behaviour of the SoC is crucial. In this section, some existing health monitors will be classified into several groups. These monitors can obtain the degraded signal of e.g. the aging caused by additional path delay,

the I_{DDQ} current or parameters like temperature and voltage that can provide for instance data on the health of a processor core (e.g. power dissipation).

2.6.1 CANARY CIRCUITS

The first class of health monitors consist of so-called canary circuits [Shah 08], also referred to as prognostics cells; they are incorporated into integrated circuits to detect aging-induced performance degradation in a predictive manner.

They can provide an advance warning of failure because they always fail earlier than the normal operating circuits. Since it is a conservative design for detecting aging, it is hard to get the accurate health monitoring information.

In [Wang 08], a feedback scheme using canary replicas provides for aggressive scaling for SRAM standby leakage power reduction without losing data.

An adaptive technique for compensating manufacturing and environmental variability in subthreshold circuits using canary flip-flops (FF) is presented in [Fuke 12], which can warn for future timing errors.

2.6.2 FUNCTIONAL MONITORS

Another class of health monitor is referred to as *functional* monitors. In this case some performance parameters with regard to the target processor are monitored.

Interesting functional monitors are for instance delay monitors [Hyun 10], [Vazq 10]; they can determine the system (frequency) operating degradation that can be caused by the aging behaviour. A simple logic BIST structure could monitor functional (permanent and transient) logic faults [Zhan 11].

Other health monitors can measure quiescent current (I_{DDQ}) [Kunh 07], [Path 18], which can potentially show the *functional* degradation of the processor due to aging effects. This monitor can be combined with a processor workload monitor [Bara 13], indicating the present stress condition of the target processor.

2.6.3 TECHNOLOGICAL MONITORS

The class of *technological* monitors are the most popular ones. Examples are monitors for Electro Migration (EM), NBTI, PBTI, TDDB and HCI. An important concern is the effect of different aging mechanisms with regard to different technology nodes.

The Ridgetop Group® has designed and developed those type of monitors which are available in 90 nm and 65 nm [Ridg 13]. These monitors can provide information at the transistor level during aging, and hence indirectly in terms of the target processor performance. Other work such as published in [Kim 12], [Oman 10] also proposed several NBTI monitors, which are designs that can directly observe the aging of targeted embedded processors.

2.6.4 ENVIRONMENTAL MONITORS

The last class is *environmental* monitors. An example is one that monitors the local chip temperature [Tzu 12], [Path 18]. It will affect the standby current, the speed of the target processor and drastically accelerate the aging process. Another example is a power-supply voltage monitor [Wan 14]. An increase in local power supply will also accelerate the aging process, and in addition affect the delay measurement.

2.7 REMAINING LIFETIME PREDICTION

The monitoring data from the health monitors are collected and processed to determine the health condition of our processor cores in MP-SoCs. Future health condition and thus (remaining) life-time of the MP-SoCs can be predicted based on the available monitoring data. Optimal maintenance actions are scheduled based on the predicted life-time, so that preventive repairs can be performed to minimize maintenance costs and attain a high dependability of the target MP-SoCs.

There are several papers in the field of prognostics and lifetime prediction. Generally, these papers focus on either methods that are driven by historical data or those

that are statistical-model oriented. There are four main approaches, namely Brownian motion with drift (Wiener processes), Gamma processes, the covariate-based hazard model and the degradation-based model. This section reviews relevant research in the degradation process modelling and remaining lifetime predictions.

2.7.1 WIENER PROCESSES

In physics, a Wiener process [Cox 65] aims to model the tiny fluctuation in the movement of small particles in fluids and air. Thus, it can be used to model an object in terms of increasing or decreasing performance, similar to the random walk of small particles.

Mathematically, a Wiener process can be represented as $Y(t) = \lambda * t + \gamma B(t), t \geqslant 0$, where Y(t) is the degradation path of the process, λ denotes a defined drift parameter, γ is a diffusion coefficient, and B(t) is the standard Brownian motion [Cox 65]. The definition of the remaining lifetime at time t can be represented by the first passage time of Y(t) crossing a threshold, and resulting from this the remaining lifetime follows an inverse Gaussian distribution.

Applications are described in [Whit 97] with a degradation model of self-regulating heating cables and [Liao 06] with a model for the light intensity of LED lights in contact-image scanners.

Wiener processes imply that the future state of the degradation process is independent of its current state. The variance of Y(t) is proportional to the length of time while it is being measured. Finally, the mean degradation path ($\lambda * t$) should be linear or can be linearized.

Considering the above factors, the Wiener process is a poor generic model for the reliability model of a process, and has not been further considered in our research.

2.7.2 GAMMA PROCESSES

Gamma processes are suitable for modelling monotonic and gradual degradation processes such as wear-out processes or fatigue-crack propagation, in which the

deterioration takes place gradually in a sequence of tiny but positive increments. Therefore, in Gamma processes, degradation processes are monotonic and evolve only in one direction.

A Gamma process $Y(t), t \geq 0$ has several properties [Abde 75]:

(i) the increment of $Y(t_i) - Y(t_{i-1})$ for a given time interval $\Delta = t_i - t_{i-1}$ has a Gamma distribution $Ga(v(t_i) - v(t_{i-1}), \sigma)$ with shape function $v(t) > 0$ and a scale parameter $\sigma > 0$;

(ii) the increments for any set of disjoint time intervals are independent random variables that have the distributions described in property (i);

(iii) $Y(0) = 0$; using a Gamma process, the remaining lifetime at time t can be calculated if the failure threshold is set [Noor 09], which is quite straightforward.

Nevertheless, when applying a Gamma process, it is only appropriate in modelling monotonic degradation. Furthermore, the Gamma process implies memoryless behaviour: the future degradation is independent of its previous or current degradation state.

For these reasons, Gamma processes will not be included into our lifetime-prediction model.

2.7.3 COVARIATE-BASED HAZARD MODELS

In applications such as the aging of an electronic device, the degradation process is caused by one or more factors that are termed covariates [Lawl 04]. For example, the aging can be affected by the normal operation and the environmental factors such as temperature and voltage. The covariate-based hazard model can incorporate such covariates in life-time modelling.

The proportional hazards model, which is one of the basic covariate-based models, has been widely researched since Cox's pioneering work [Cox 72].

Different from the other three models is that the remaining lifetime prediction is based on the estimation at which moment the health-monitoring data will reach the failure threshold. Covariate-based hazard models do not require precise failure threshold information for the prediction. The main assumption is that the ratio of the hazards for

any two monitoring parameters remains constant over time, meaning they are proportional, and this constant ratio is known as the hazard ratio [Cox 72]. Therefore, this model has not been taken into consideration in our research, since the degradation speed and hence hazard ratio for our health monitors can be different.

2.7.4 DEGRADATION-BASED MODEL

The degradation-based model is the most popular model for failure life-time prediction. Degradation means the reduction in performance and reliability that is caused by the normal aging of a component. The degradation phenomena are stochastic processes with many kinds of failure mechanisms involved; therefore, it should be modelled employing several approaches. This model can be generally classified into experienced-based approaches [Weib 51] and/or a data-driven prognostics approach [Pech 10].

Experienced-based approaches

Based on knowledge of the degradation processes which lead to a failure of the system, different physics-of-failure modelling approaches [Fan 11] can be applied, such as Log-Normal, Exponential and Weibull distributions. The most popular one amongst them is the Weibull distribution due to its ability to conduct different types of behaviour including infant mortality and wear-out with the failure rate following a bathtub curve [Weib 51].

Data-driven approaches

A significant property of the data-driven approach is that real-time health-monitoring data has to be collected by in-situ monitors. Such data includes parameters concerned with environmental variations and the operational loads in the monitored system, which reflects the fatigue or aging. Based on various applications of the approach, the data can be generally categorized into two kinds: event data and condition-monitoring data. Event data is recorded data that implies a failure or error, whereas condition-monitoring data is the normal in-field health-monitoring data. To achieve an accurate

model or reliable remaining lifetime prediction, event data as well as health-monitoring data acquisition of the many-core chip is essential [Jard 06].

Nowadays, two of the most popular approaches behind these models are machine learning and random-coefficient regression methods. The machine-learning approach [Heng 09] uses observed data and some statistical techniques such as least squares. The random-coefficient regression methods use the health-monitoring data to build a degradation path from which it infers the remaining-lifetime distribution. The latter is based on the following assumptions: (1) the condition of the device deteriorates with operating time and the level of deterioration can be observed at any time; (2) the device being monitored comes from a population of devices, each of which exhibits the same degradation form; (3) the distribution of the random term across the population of devices is known; and (4) the error in the degradation signal is independent and identically distributed across the population of devices. Lu and Meeker [Lu 93] proposed a model to predict the remaining lifetime of the device and later Wang [Wang 00] proposed an optimal critical level and a monitoring intervals determination method. The most popular approach nowadays has been provided by Gebraeel [Gebr 05], which is focused on computing a remaining lifetime distribution for a single operating device using sensor-based monitoring signals; it is based on the historical data and real-time health-monitoring information. This model will continuously update the parameters of the random coefficient model within a Bayesian framework [Mack 92]. Furthermore, a Brownian motion error process was assumed instead of the independent normal random error used by Lu and Meeker [Lu 93]. This results in a closed-form probability remaining lifetime distribution. For the above-mentioned reasons and assumptions which fit for the health-monitoring results, the machine-learning method in the data-driven approach will be employed by us for the life-time prediction.

2.8 CONCLUSIONS

This chapter has introduced the dependability challenges in modern SoC design, and provides the basic background of dependable system design issues in many-processor SoCs. The dependability attributes reliability and availability are the most crucial in this thesis. The PHM dependable MP-SoC has been introduced and compared with the scan-based BISTR dependable MP-SoC. Applications are for instance in the security or safety-critical areas and hence a very high degree of dependability should be reached. For this purpose, extensive use is made of available core resources, and associated health monitoring and lifetime prognostics techniques.

Different health monitors have been introduced, and a combination of them has to be employed for a PHM dependable MP-SoC, since aging of a SoC always behaves differently depending on the monitored parameters.

Existing remaining lifetime prediction models have been reviewed, and the data-driven model in the degradation-based model has been chosen to be the focus of this thesis.

The PHM enables the preventive (automatic) maintenance/repair by spare parts or priority ranking of tasks among processors. It guarantees a high dependability, among its attributes a high reliability and 100% availability for a certain lifetime of very complex SoCs.

REFERENCES

[Abde 75] M. Abdel-Hameed, "A Gamma Wear Process," in IEEE Transactions on Reliability, vol. R-24, pp. 152-153, 1975.

[Ahon 11] T. Ahonen, T.D. ter Braak, S.T. Burgess, R. Geißler, et al., "CRISP: Cutting Edge Reconfigurable ICs for Stream Processing," in Reconfigurable Computing: From FPGAs to Hardware/Software Codesign, Springer Verlag, ISBN 978-1-4614-0061-5, London, 2011.

[Axer 11] P. Axer, M. Sebastian, and R. Ernst. "Reliability analysis for MPSoCs with mixed-critical, hard real-time constraints," in 9th International Conference on Hardware/Software Codesign and System Synthesis, pp. 149-158, 2011.

[Bara 13] R. Baranowskia, A. Cookb, M.E.Imhof, C. Liud, and H.-J. Wunderlich, "Synthesis of Workload Monitors for On-Line Stress Prediction," in IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), New York City, USA, pp. 137-142, 2013.

[Bern 06] J.B. Bernstein, M. Gurfinkel, X. Li, J. Walters, et al., "Electronic circuit reliability modeling," in Microelectronics Reliability, Vol. 46, No. 12, pp. 1957-1979, Dec. 2006.

[Bork 07] S. Borkar. "Thousand core chips: a technology perspective," in Proceedings of the 44th Annual Design Automation Conference (DAC), pp. 746-749, ACM, 2007.

[Braa 16] T.D. Braak, "Run-time mapping: dynamic resource allocation in embedded systems," PhD thesis, ISBN: 978-90-365-4213-5, University of Twente, Enschede, 2016.

[Cox 65] D.R. Cox and H.D. Miller, "The theory of stochastic processes." CRC press, ISBN 0-412-151707-7, London, 1965.

[Cox 72] D.R. Cox, "Regression models and life-tables," in Journal of the Royal Statistical Society. Series B (Methodological), pp. 187-220, 1972.

[Dama 13] M. Damavandpeyma, S. Stuijk, et al., "Throughput-constrained DVFS for scenario-aware dataflow graphs," in Proceedings IEEE Symp. Real-Time Embedded Technol. Appl., pp. 175-184, 2013.

[Das 16] A. Das, A. Kumar and B. Veeravalli, "Reliability and Energy-Aware Mapping and Scheduling of Multimedia Applications on Multiprocessor Systems," in IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 3, pp. 869-884, 1 March 2016.

[Dixo 06] S. R. Dixon and C. D. Wickens, "Automation Reliability in Unmanned Aerial Vehicle Control: A Reliance-Compliance Model of Automation Dependence in High Workload," in Human Factors, vol. 48, pp. 474-486, 2006.

[Esco 06] L. A. Escobar and W. Q. Meeker, "A review of accelerated test models," in Statistical Science, pp. 552-577, 2006.

[Fan 11] J. Fan, K. Yung, and M. Pecht, "Physics-of-Failure-Based Prognostics and Health Management for High-Power White Light-Emitting Diode Lighting," in IEEE Transactions on Device and Materials Reliability, vol. 11, pp. 407-416, 2011.

[Fu 14] Y. Fu and D. Wentzlaff, "PriME: A parallel and distributed simulator for thousand-core chips," in IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Monterey, CA, pp. 116-125, 2014.

[Fuke 12] H. Fuketa, M. Hashimoto, Y. Mitsuyama and T. Onoye, "Adaptive Performance Compensation With In-Situ Timing Error Predictive Sensors for Subthreshold Circuits," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no. 2, pp. 333-343, Feb. 2012.

[Gebr 05] N. Z. Gebraeel, M. A. Lawley, R. Li, and J. K. Ryan, "Residual-life distributions from component degradation signals: A Bayesian approach," in *IIE Transactions,* vol. 37, pp. 543-557, 2005.

[Gras 09] T. Grasser and B. Kaczer. "Evidence That Two Tightly Coupled Mechanisms Are Responsible for Negative Bias Temperature Instability in Oxynitride MOSFETs," in IEEE Transactions on Electron Devices, Vol. 56, No. 5, pp. 1056-1062, May 2009.

[Heng 09] A. Heng, S. Zhang, A.C.C. Tan, and J. Mathew, “Rotating machinery prognostics: State of the art, challenges and opportunities,” in Mechanical Systems and Signal Processing, vol. 23, pp. 724-739, 2009.

[Huan 10] L. Huang and Q. Xu, “Energy-efficient task allocation and scheduling for multi-mode MPSoCs under lifetime reliability constraint,” in Proceedings Conf. Design, Autom. Test Eur. (DATE), pp. 1584-1589, 2010.

[Hyun 10] Y. Hyunbean, T. Yoneda, M. Inoue, et al., “Aging test strategy and adaptive test scheduling for SoC failure prediction,” in IEEE 16th International On-Line Testing Symposium (IOLTS), pp. 21-26, 2010.

[Ieee 16] “IEEE P1687.1, Standard for the Application of Interfaces and Controllers to Access 1687 IJTAG Networks Embedded Within Semiconductor Devices”, 2016.

[Jard 06] A. K. S. Jardine, D. Lin, and D. Banjevic, “A review on machinery diagnostics and prognostics implementing condition-based maintenance,” in Mechanical Systems and Signal Processing, vol. 20, pp. 1483-1510, 2006.

[Jede 10] “JEDEC standard JESD22-A108D,” http://www.jedec.org/ standardsdocuments/, November 2010.

[Jede 11] “JEDEC standard JESD22-A105C,” http://www.jedec.org/ standardsdocuments/, January 2011.

[Kean 10] J. Keane, T.-H. Kim, and C. H. Kim, “An on-chip NBTI sensor for measuring PMOS threshold voltage degradation,” in IEEE Transactions on Very Large-Scale Integration (VLSI) Systems, vol. 18, pp. 947-956, 2010.

[Kerk 10a] H.G. Kerkhoff, and X. Zhang, “Design of an infrastructural IP dependability manager for a dependable reconfigurable many-core processor,” in 5th IEEE International Symposium on Electronic Design, Test & Applications (DELTA), ISBN 978-0-7695-3978-2, Ho Chi Minh City, Vietnam, pp. 270-275, January 2010.

[Kerk 10b] H.G. Kerkhoff, and J. Wan, “Dependable digitally-assisted mixed-signal IPs based on integrated self-test & self-calibration,” in IEEE 16th

International Mixed-Signals, Sensors and Systems Test Workshop (IMS3TW), La Grande Motte, France, pp. 1-6, June 2010.

[Kerk 12] H. G. Kerkhoff and Y. Zhao, “The design of dependable flexible multi-sensory System-on-Chips for security applications,” in IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), Tallinn, Estonia, pp. 133-138, 2012.

[Khan 11] M.A. Khan and H.G. Kerkhoff, “A system-level platform for dependability enhancement and its analysis for mixed-signal SoCs,” in 14th IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), ISBN 978-1-4244-9753-913-15, Cottbus, Germany, pp. 17-22, April 2011.

[Kim 12] T. T. Kim, P.-F. Lu, and C.H. Kim, “Design of ring oscillator structures for measuring isolated NBTI and PBTI,” in IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1580-1583, 2012.

[Kuik 08] O.J. Kuiken, X. Zhang and H.G. Kerkhoff, “Built-in self-diagnostics for a NoC-based reconfigurable IC for dependable beamforming applications,” in the 23rd IEEE International Symposium on Defect and Fault-Tolerance in VLSI systems (DFT), ISBN 978-0-7695-3365-0, Cambridge, MA, USA, pp. 45-53, 2008.

[Kunh 07] K. Kunhyuk, K. Keejong, A.E. Islam, et al., “Characterization and Estimation of Circuit Reliability Degradation under NBTI using On-Line I_{DDQ} Measurement,” in 44th ACM/IEEE Design Automation Conference (DAC), pp. 358-363，2007.

[Lawl 04] J. Lawless and M. Crowder, “Covariates and random effects in a gamma process model with application to degradation and failure,” in Lifetime Data Analysis, vol. 10, pp. 213-227, 2004.

[Liao 06] H. Liao and E. A. Elsayed, “Reliability inference for field conditions from accelerated degradation testing,” in Naval Research Logistics, vol. 53, pp. 576-587, 2006.

[Lu 93] C. J. Lu and W. Q. Meeker, “Using Degradation Measures to Estimate a Time-to-Failure Distribution,” in Technometrics, vol. 35, pp. 161-174, 1993.

[Mack 92] D.J.C. MacKay, “A practical Bayesian framework for backprop networks,” in Neural Computation, vol. 4, pp. 448-472, 1992.

[Mahe 03] A. Maheshwari, I. Koren and N. Burleson, “Techniques for transient fault sensitivity analysis and reduction in VLSI circuits,” in Proceedings 18th IEEE Symposium on Defect and Fault Tolerance in VLSI Systems (DFT), Boston (MA), USA, pp. 597-604, 2003.

[Mari 13] E. Maricau and G. Gielen, “Analog IC Reliability in Nanometer CMOS,” Springer Publishing Press, ISBN 978-1-4614-6162-3, 2013.

[Mcke 17] M. McKeown et al., “Piton: A Manycore Processor for Multitenant Clouds,” in IEEE Micro, vol. 37, no. 2, pp. 70-80, Mar.-Apr. 2017.

[Noor 09] J. M. van Noortwijk, “A survey of the application of gamma processes in maintenance,” in Reliability Engineering and System Safety, vol. 94, pp. 2-21, 2009.

[Oman 10] M. Omana, D. Rossi, N. Bosio, and C. Metra, “Self-checking monitor for NBTI due degradation,” in IEEE 16th International Mixed-Signals, Sensors and Systems Test Workshop (IMS3TW), pp. 1-6, 2010.

[Pari 18] N. Parihar, R. G. Southwick, et al., “Modeling of NBTI Kinetics in RMG Si and SiGe FinFETs, Part-I: DC Stress and Recovery,” in IEEE Transactions on Electron Devices, vol. 65, no. 5, pp. 1699-1706, May 2018.

[Path 18] J. Pathrose, G. Ali and H. G. Kerkhoff, “IJTAG compatible analogue embedded instruments for MPSoC life-time prediction,” in IEEE 19th Latin-American Test Symposium (LATS), Sao Paulo, pp. 1-4, 2018.

[Paul 05] B. C. Paul, K. Kunhyuk, et al., “Impact of NBTI on the temporal performance degradation of digital circuits,” in IEEE Electron Device Letters, vol. 26, pp. 560-562, 2005.

[Pech 09] M. Pecht, “Prognostics and health management of electronics,” in Encyclopedia of Structural Health Monitoring, 2009.

[Pech 10] M. Pecht and R. Jaai, “A prognostics and health management roadmap for information and electronics-rich systems,” in Microelectronics Reliability, vol. 50, pp. 317-323, 2010.

[Pisc 12] R. Piscitelli and A.D. Pimentel, “Design space pruning through hybrid analysis in system-level design space exploration,” in Design Automation & Test in Europe Conference & Exhibition (DATE), pp. 781-786, 2012.

[Rahi 07] M. K. Rahim, J. A. Roberts, et al., “Continuous In-Situ Die Stress Measurements During Thermal Cycling Accelerated Life Testing,” in Proceedings of the 57th IEEE Electronic Components and Technology Conference, Reno (NV), USA, pp. 1478-1489, 2007.

[Reco 11] RecoreSystems. http://www.recoresystems.com, 2011.

[Ridg 13] Ridgetop, “ProCheck Reference Manual,” https://www.ridgetopgroup.com, 2013.

[Saha 11] S. Saha, J. R. Celaya, V. Vashchenko, et al., “Accelerated aging with electrical overstress and prognostics for power MOSFETs,” in IEEE Energytech, pp. 1-6, 2011.

[Scor 91] A. Scorzoni, B. Neri, C. Caprile, and F. Fantini, “Electromigration in thin-film interconnection lines: Models, methods and results,” in Materials Science Reports, Vol. 7, pp. 143-220, 1991.

[Sepu 12] J. Sepulveda, G. Gogniat, R. Pires, W. J. Chau and M. Strum, “Hybrid-on-chip communication architecture for dynamic MP-SoC protection,” in 25th Symposium on Integrated Circuits and Systems Design (SBCCI), Brasilia, pp. 1-6, 2012.

[Shah 08] N. Shah, R. Samanta, M. Zhang, J. Hu, and D. Walker, “Built-In Proactive Tuning System for Circuit Aging Resilience,” in IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems (DFT), pp. 96-104, 2008.

[Shan 14] T. Shan, Z. Ziyuan and S. Yongtao, “System-level design methodology enabling fast development of baseband MP-SoC for 4G small cell base station,” in Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, pp. 1-6, 2014.

[Shao 91] J. Shao and L. R. Lamberson, “Modeling a shared-load k-out-of-n: G system”, in IEEE Transactions on Reliability, 40(2), pp. 205-209, June 1991.

[Sing 13] A. K. Singh, A. Das, and A. Kumar, “Energy optimization by exploiting execution slacks in streaming applications on multiprocessor systems,” in Proceedings 50th Annual Design Autom. Conf. (DAC), pp. 1-7, 2013.

[Star 10] “STARS: Sensor Technology Applied in Reconfigurable systems”, 2010. http://cas.et.tudelft.nl/Research/project.php?id=33

[Tzu 12] C. Tzu-Ting, H. Po-Tsang and C. Ching-Te, et al., “On-chip self-calibrated process-temperature sensor for TSV 3D integration,” in IEEE International SOC Conference (SOCC), Niagara Falls, NY, pp. 370-375, 2012.

[Vazq 10] J.C. Vazquez, V. Champac, I. C. Teixeira, et al., “Programmable aging sensor for automotive safety-critical applications,” in Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 618-621, 2010.

[Wan 14] J. Wan and H.G. Kerkhoff, “An embedded offset and gain instrument for OpAmp IPs,” in Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1-4, 2014.

[Wang 00] W. Wang, “A model to determine the optimal critical level and the monitoring intervals in condition-based maintenance,” in International Journal of Production Research, vol. 38, pp. 1425-1436, 2000.

[Wang 08] J. Wang and B. H. Calhoun, "Techniques to Extend Canary-Based Standby V_{DD} Scaling for SRAMs to 45nm and Beyond," in IEEE Journal of Solid-State Circuits, vol. 43, no. 11, pp. 2514-2523, Nov. 2008.

[Wats 15] J. Watson and G. Castro, “A review of high-temperature electronics technology and applications,” in Journal of Materials Science: Materials in Electronics, vol. 26, pp. 9226-9235, 2015.

[Weib 51] W. Weibull, “A statistical distribution function of wide applicability,” in Journal of applied mechanics, pp. 293-297, 1951.

[Whit 97] G.A. Whitmore and F. Schenkelberg, “Modelling Accelerated Degradation Data Using Wiener Diffusion with a Time Scale Transformation,” in Lifetime Data Analysis, vol. 3, pp. 27-45, 1997.

[Wolk 09] P.T. Wolkotte, "Exploration within the Network-on-Chip Paradigm," PhD Thesis, University of Twente, ISBN: 978-90-365-2757-6, 2009.

[Zhan 09] X. Zhang and H. Kerkhoff. "Design of a Highly Dependable Beamforming Chip," in 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools, pp. 729-735, Aug. 2009.

[Zhan 11] X. Zhang and H.G. Kerkhoff, "A dependability solution for homogeneous MPSoCs," in 17th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), ISBN 978-0-7695-4590-5, Pasadena, USA, pp. 53-62, 2011.

[Zhao 14] Y. Zhao and H. G. Kerkhoff, "Design of an Embedded Health Monitoring Infrastructure for Accessing Multi-processor SoC Degradation," in 17th Euromicro Conference on Digital System Design (DSD), Verona, Italy, pp. 154-160, 2014.

[Zhao 16] Y. Zhao and H. G. Kerkhoff, "Highly Dependable Multi-processor SoCs Employing Lifetime Prediction Based on Health Monitors," in IEEE 25th Asian Test Symposium (ATS), Hiroshima, Japan, pp. 228-233, 2016.

Chapter 3

EMBEDDED HEALTH MONITORS FOR DEPENDABLE MP-SOCS

ABSTRACT *– From the previous chapter, one can conclude that Many-Processor System-on-Chips (MP-SoCs) have been intensively researched and developed for their powerful data-processing capability. However, an emerging problem facing them is reduced dependability as transistors shrink and complexity increases. An embedded health-monitoring infrastructure for a highly dependable MP-SoC is presented in this chapter. Different from the traditional approach of a dependable design, our approach is based on life-time prognostics based on data from health monitors that are embedded near the target processor core in the SoC. This enables the preventive repair using spare parts or priority ranking of tasks among processors. This chapter will deal with the state-of-the-art of available health monitors and our health-monitoring architecture in the MP-SoC. Furthermore, our all-in-one health monitor including simulation results is presented and a discussion on the power dissipation of different dependable systems is held.*

Parts of this chapter have been published as paper titled "Power-Dissipation Comparison of Two Dependability Approaches for Multi-Processor System" in the International Conference on Design & Technology of Integrated Systems, 2013 [Zhao 13], and "Design of an Embedded Health Monitoring Infrastructure for Accessing Multi-Processor SoC Degradation" in the Euromicro Conference on Digital System Design, 2014 [Zhao 14].

3.1 INTRODUCTION

The developments of the semiconductor technologies and requirements for high data-streaming applications, have resulted in Many-Processor System-on-Chips (MP-SoCs) commonly found in embedded platforms nowadays. They contain many usually heterogeneous, processing cores with specific functionalities reflecting the requirements of the expected application domain [Paul 11].

With the continuing developments in electronic-system integration, process dimensions have shrunk, the power-supply voltage has become lower, and the operational frequency and chip current have increased continuously. It has led to a severe dependability reduction of systems during lifetime [Kane 10].

A BISTR method has been designed earlier by us for accomplishing a dependable MP-SoC [Kerk 10]. Rather different from this approach is the usage of embedded health monitors for dependability [Kerk 12] being the basis of this chapter. It applies health monitoring (HM) for life-time prognostics and self-repair to enhance the dependability.

Generally, health-monitoring is the process of collecting (real-time) measurement information from the target device in order to reason about its health status. This has a twofold advantage. First, by catching failures proactively, maintenance actions can be scheduled at a time before the failure becomes active. This leads to less unscheduled downtime and ideally to a totally dependable system. Second, it uses the approach of prognostics, thereby reducing the life-cycle cost and testing costs as compared to a scheduled inspection and/or maintenance.

Existing methods have been focussed on the degradation of individual parts/components and usually one health parameter is being monitored. For example, Karl et al. [Karl 08a] proposed an online failure-prediction monitor that digitally quantifies the negative bias temperature instability (NBTI) effect by using the frequency degradation of a ring oscillator. Vazquez et al. designed and implemented an aging monitor to predict a delay failure [Vazq 09].

In this context, our research aims to develop a method that uses health information to predict the remaining useful lifetime (RUL) of processor cores (i.e. the Xentium® processor) being monitored in an MP-SoC. Such a technique enables our system to be self-aware of its dependability. It allows to dynamically adjust the operating conditions of the MP-SoC, e.g. operating frequency, supply voltage and the temperature limit, in

order to achieve peak performance benefits while meeting the lifetime specification. This approach has been termed as dynamic reliability management (DRM) [Karl 08b].

In order to obtain the most accurate prediction result of the remaining lifetime in the DRM system, it is crucial to identify the monitoring parameters that capture the key present health status, and relate it to the aging behaviour of the SoC. In this chapter, based on the underlying aging mechanisms (NBTI effect is assumed as the dominant aging mechanism), an all-in-one health monitor is presented which is capable of carrying out voltage and temperature measurements as well as dependability tests (via delay-time monitoring).

Such a compact monitor is related to an analogue-to-frequency converter method based on the ring oscillator (ROSC) design concept [Chan 14]. Due to its simple structure and digital outputs it can be used extensively in the standard-cell based circuits in our MP-SoC. This type of monitor enables high-volume data collection and the monitoring of chip dependability throughout the system lifetime. The health-monitoring data supplied by these monitors will be used for the maintenance action afterwards, via the IJTAG (IEEE 1687) standard [Ieee 16] node with some adaptations. Compared to previous ROSC-based monitors that specifically focus on one parameter, our target is to obtain more monitoring parameters and health information from this monitor [Zhao 14]. The monitor communicates with the embedded general-purpose processor, and thus the health information will be transferred to this processor during field operation.

In our research, the additional power dissipation for our proposed health-monitoring architecture for a dependable MP-SoC will be discussed and compared to the previous BISTR for dependability technique [Zhao 13], [Zhan 11]. This is of special importance since nowadays power-awareness has become a major concern for embedded systems [Liu 11].

The power dissipation components of the previous implemented BISTR approach [Zhan 11] have been evaluated in [Zhao 13]. It is shown that the application of scan-test vectors and transporting results to the cores is the major power contributor. This is avoided in our new embedded health monitors and the lifetime prediction technique, meanwhile accomplishing the same high level of dependability. It will be shown that our health-monitoring approach consumes less power under the same dependability specifications.

3.2 THE EMBEDDED HEALTH-MONITORING INFRASTRUCTURE IN OUR MP-SOC

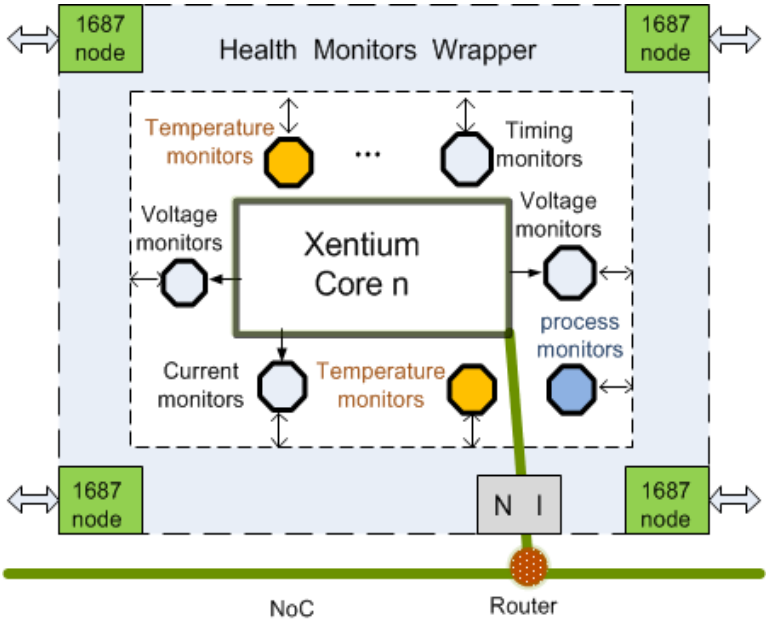
To improve the quality of maintenance of our dependable MP-SoC, a number of embedded health monitors have been used. The voltage and current monitors provide information on the health of the core (e.g. power dissipation). Temperature monitors can help to detect a local increase of temperatures caused by developing faults in the core. This could be combined with a processor workload monitor [Bara 13]. The (slack) delay monitors, in combination with voltage monitors, can show the system (frequency) operating degradation that can be caused by aging mechanisms.

3.2.1 GENERAL ARCHITECTURE

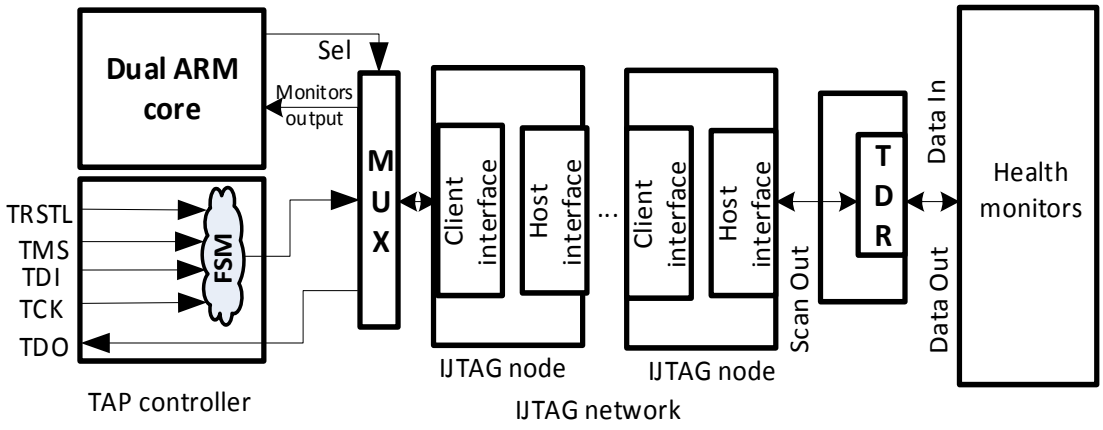
Figure 3.1a shows the health-monitoring infrastructure for one Xentium® processor core (within an MP-SoC) from the company Recore Systems; it is an IJTAG-based structure normally used during final test via the test access port (TAP) controller [Zhao 16]. The embedded health monitors are located close to the core, in a wrapper style. The signals from the monitors are, if required, converted to digital data in the health monitors wrapper (Figure 3.1a). Worst case this requires (shared) amplification and data conversion (ADC) in the wrapper of the monitor. The monitor wrapper is basically a smart register that can provide commands to the monitors and collects digital measurement data that can be shifted out. It follows the protocol of the IJTAG standard [Ieee 16]. The digital health-monitoring data is subsequently transported via the network interface (NI) and NoC (Figure 3.1a) to one of the two ARM cores (Figure 3.1b) in the MP-SoC. All Xentium cores communicate with each other via the NI and NoC.

As shown in Figure 3.1b, to get the health-monitoring information for the subsequent task to the general processor(s), an 8-bit 2-terminal multiplexer, a select line and an 8-bit bus are introduced between the IJTAG 1687 node and the TAP controller. The resident embedded dual ARM cores control the mux. The key point of focus in our research is the design of health monitors; for the IJTAG network in the proposed infrastructure we refer to research published in for instance [Shib 16], [Ali 16].

Generally, scan registers are used for network configuration and for data delivery to/from the health monitors; the latter is often referred to as a Test Data Register (TDR). The client interface and the host interface in the IJTAG network is used to enable a plug and play module integration and a hierarchical construction [Zade 12]. A client interface is attached to the host interface. Client interfaces receive scan-control signals attached to host interfaces. Ultimately, the top-level client interfaces of modules are interfaced with the TAP controller, and the lowest-level host interfaces of modules are interfaced with our health monitors.



a)



b)

Figure 3.1: Proposed health-monitoring infrastructure. a) the set-up of embedded health monitors in the Xentium core (within a MP-SoC) [Kerk 12], b) the communication between health monitors and ARM cores via the IJTAG network interface [Zhao 14].

3.2.2 AN EXAMPLE: EMBEDDED I_{DDT} MONITOR/INSTRUMENT WITH IJTAG-COMPATIBLE INTERFACE

This section gives an example of an I_{DDT} embedded monitor/instrument which is compatible with the emerging standard IEEE 1687. By using this standard, one can configure the monitor in the proper modes (e.g. set the window) and subsequently measuring the currents; it can be flawlessly fused with other IJTAG embedded instruments and a proper network structure [Ibra 16a]. The measurements can be carried out via the TAP during final testing, or internally during lifetime via an embedded IJTAG controller [Kerk 12]. It opens the road for zero mean downtime, and significantly increased dependability in the case of homogeneous multi-processor SoCs.

In Figure 3.2, the embedded I_{DDT} monitor is globally shown, as well as the IJTAG wrapping around the monitor. Off-chip as well as on-chip I_{DDT} monitors have been developed since the nineties, where in the design, speed and resolution are main issues [Yang 11]. The monitor consists (Figure 3.2) of a current-to-voltage conversion, taking care to remain as close to V_{DD} for the core under test as is possible. As the resulting voltages are small, several amplification stages are required after this. The last step is the translation to a digital (in our case 14-bits) word, via a voltage-to-frequency conversion.

Furthermore, several supporting circuits are required, like a controller and a samples memory. Registers for the BIST of the monitor, a BIST enable and BIST pass/fail are available (Figure 3.2). Another register is for calibration purposes. For starting the I_{DDT} measurement and get (measurement) samples and finally indicate the completion of samples, three registers are required. The above combinations are referred to as Control and Status TDRs. Finally, the 14-bit I_{DDT} output data, is combined in the Measurements TDR. In Figure 3.2, also four ScanMuxs and Segment Insertion Bits (SIBs) can be seen for the four TDRs which have been added to optimize the access for the monitor [Ibra 16b].

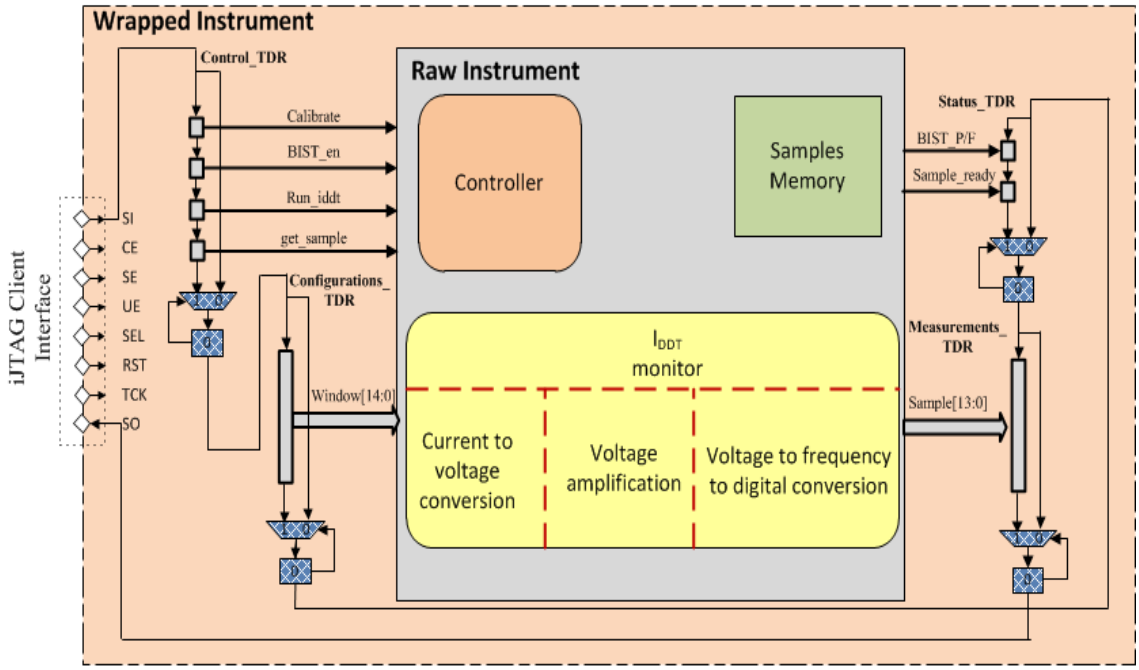


Figure 3.2: A general scheme of an IJTAG-compatible I_{DDT} monitor for the remaining lifetime prediction [Ibra 16b].

3.3 A Programmable All-in-One ROSC-based Health Monitor

This section presents the architecture and simulation results of our designed all-in-one health monitor, employing the embedded health-monitoring infrastructure for our dependable MP-SoC.

3.3.1 State-of-the-Art of ROSC-based Health Monitors

From the previous discussion, for the CMOS transistor technology from 90 nm to 45nm, with aggressive oxide thickness scaling, the NBTI effect dominates the degradation of the threshold voltage (V_{TH}).

Since NBTI is sensitive to operating conditions, it is extremely difficult to predict the degradation during design time. Therefore, the question of using on-chip structures for real-time health monitoring of devices and circuits has been raised. Meanwhile, as the

performance degradation is a statistical process, many monitors may be required for a DRM system for the statistical analysis.

As an example, the on-chip ring oscillator (ROSC) has been effectively applied in monitoring the reliability of a system (one of the most important attributes for dependability in our case) in terms of delay performance degradation [Kean 10a], [Wang 15]. In order to explore the on-line workload characterization of mobile platforms, an on-chip ROSC based thermal monitor has been employed [Sayed 11], where the ROSC has been used for workload-variation quantification. In technology-based health monitors, ring oscillators have been widely used for NBTI measurements [Hong 15]. In order to measure the aging effect under stress conditions, these monitors consist of a pair of ring oscillators, one of which experiences accelerated stress. The monitor proposed in [Kean 10b], [Wang 15] measures the beat frequency (i.e. the frequency difference of two oscillators), which is attributed to the V_{TH} shift due to aging. The monitor proposed in [Tae 08] employs a controllable external analogue bias to map beat-frequency changes to the V_{TH} shift. All these monitors require some delay stages as well as a reference ring oscillator. In a highly dependable system, however, the health status not only requires technological monitors, but also functional/performance monitors should be used (e.g. delay [Chan 14]).

These monitors are specifically targeted to one parameter, and some designs require large delay stages to get a high sensitivity of measuring the degradation and are hence costly in terms of silicon area. In order to get an accurate life-time prediction afterwards, one health-monitor or one parameter is not sufficient; a combination of monitors and monitoring parameters has to be employed, which can be deducted from for instance, structural health monitoring [Farr 07] and battery-lifetime prediction [Ecke 12].

Therefore, in the next section, a design of an all-in-one health monitor is presented, which can be used in large numbers in our MP-SoC with much health monitoring information on temperature, voltage and delay.

3.3.2 The All-in-One Monitor Design Principle

The (NBTI) aging degradation can be affected by temperature and the supply voltage; at the processor level it is reflected by the increase of the delay while transferring

data and hence a decrease in operating frequency. In our designed health monitor, in 65 nm TSMC CMOS technology, the delay-time based degradation simulation will only consider the NBTI mechanism.

Figure 3.3 shows the full schematic for the proposed all-in-one health monitor. There are three operational modes for this monitor, being the calibration mode, the normal measurement mode and the delay-time monitoring mode. Two pairs of multiplexers (T_sel, T_sel_l and V_sel, V_sel_l) are used to switch between each mode.

Compared to the previously mentioned ROSC-based monitors, the major difference is that our monitor can be configured for different health-monitoring purposes, being a temperature monitor, supply-voltage monitor and NBTI-based aging monitor. This is accomplished by two ROSCs and some programmable switches in the monitor.

For one ring oscillator, besides the process variation to be discussed later, the measured output is a frequency f which will be the combined effect of the temperature, supply voltage and (NBTI) aging in the oscillator in the normal use case. This can be written as: $f = f(Tx, Vx, Ax)$, where Tx, Vx, Ax denote the temperature variation, the supply-voltage variation and the NBTI aging effect during the lifetime respectively. The oscillation frequency change during the lifetime of the monitor quantifies the measured parameter change. As the other two monitored parameters have to be fixed if one parameter is to be measured, two ring oscillators with 101 delay-stage logic gates and some switches are used for the separate operations of different health-monitoring tasks. The number of delay stages is determined by the maximum counter operating frequency and the required accuracy.

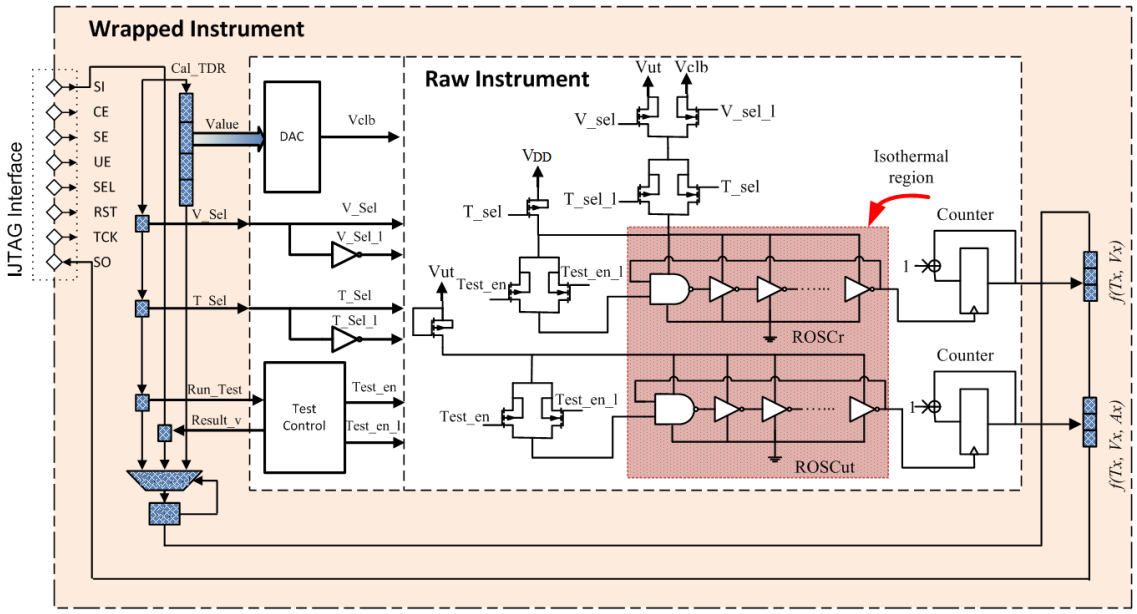


Figure 3.3: Proposed structure of the all-in-one health monitor for measurement of temperature, voltage and delay for dependability evaluation, including IJTAG wrapper.

The first oscillator (ROSCr) in Figure 3.3 is used for the temperature (T) and voltage (Vut) monitoring with an adjustable supply voltage (Vclb) for calibration purposes. The frequency can be mapped with the temperature and supply-voltage model f (T, V) after the calibration, which will be explained in the next section. The ROSCr will be only turned on in the case the health monitor is in the calibration mode and the normal measurement mode, in order to reduce the aging effects by stress.

The second oscillator (ROSCut) operates in the delay-time monitor mode while ROSCr is turned off. It should be kept under the same operating conditions (i.e. supply voltage, temperature) as the monitored processor core to ensure a similar degradation rate as the target processor.

The oscillator outputs in the monitor will be fed into counters, and their digital data is registered in the storage units. This information can be shifted out by the IJTAG interface as shown in Figure 3.3 and subsequently transferred to embedded control processors.

3.3.3 THE CALIBRATION AND MONITORING STRATEGY

The monitor outputs versus different monitored parameters have been simulated in the 65 nm TSMC CMOS process technology. The calibration of the monitor is elaborated first, and then the monitoring strategy regarding the different parameters is treated.

A. The calibration steps

The oscillation frequency output for the monitor can be formulated approximately as [Tzu 12]:

$$fosc = \frac{I_{DD}}{K \cdot C \cdot V_{DD}} \qquad \text{Eq. (3.1)}$$

where K is a fitting constant, C the output capacitance of the oscillator at each inverter stage, V_{DD} the supply voltage of the oscillator, and I_{DD} the average current drawn by the oscillator. As a result $fosc \propto I_{DD}$.

The drawn current for the oscillator can be described by:

$$I_{DD} = A \cdot \mu(T) \cdot \left(V_{GS} - V_{TH}(T)\right)^2 \qquad \text{Eq. (3.2)}$$

where A is a constant that is dependent on the MOSFET parameters width, length, and trans-conductance coefficient and the oxide gate capacitance C_{ox}. The carrier mobility is denoted by $\mu(T)$ and $V_{TH}(T)$ is the threshold voltage of the MOSFET in the ring oscillator, both at temperature T in degrees Celsius. They are specified in Eq. (3.3) and Eq. (3.4):

$$\mu(T) = \mu_0 \cdot \left(\frac{T}{T_0}\right)^k \qquad \text{Eq. (3.3)}$$

$$V_{TH}(T) = V_{TH}(T_0) + \alpha \cdot (T - T_0)^2 \qquad \text{Eq. (3.4)}$$

T_0 and μ_0 are the initial temperature and carrier mobility. Furthermore k and α are fitting parameters. By substituting Eq. (3.3) and Eq. (3.4) into Eq. (3.2), the current I_{DD} for the oscillator can be derived as:

$$I_{DD} = A \cdot \mu_0 \cdot \left(\frac{T}{T_0}\right)^k \cdot (V_{GS} - V_{TH}(T_0) - \alpha \cdot (T - T_0)^2)^2 \qquad \text{Eq. (3.5)}$$

As a result, the oscillation frequency $fosc$ of the monitor is:

$$\text{fosc} = \frac{A \cdot \mu_0 \cdot \left(\frac{T}{T_0}\right)^k \cdot (V_{GS} - V_{TH}(T_0) - \alpha \cdot (T - T_0)^2)^2}{K \cdot C \cdot V_{DD}} \quad \text{Eq. (3.6)}$$

$$= B \cdot \left(\frac{T}{T_0}\right)^k \cdot (V_{GS} - V_{TH}(T_0) - \alpha \cdot (T - T_0)^2)^2$$

$$= g(V_{GS}, T; [B, k, \alpha])$$

where

$$B = \frac{A \cdot \mu_0}{K \cdot C \cdot V_{DD}} \quad \text{Eq. (3.7)}$$

and g is the function of V_{GS} and T.

The fitting parameters in Eq. (3.6) being B, k and α must be obtained via calibration. In the calibration mode, the temperature and supply voltage Vclb in the ROSCr (Figure 3.3) are being swept, while the output frequency is measured. The steps are as follows:

I. Set the initial temperature (T=27 °C), since this will be a time-consuming process if lots of temperature points are swept, three points (T= 27 °C, 57 °C and 87 °C) can be chosen for fitting parameters B, k and α;
II. Sweep the $Vclb$ (from $V_{DD} - 0.03$ V to $V_{DD} + 0.03$ V with a step of 0.01 V) and measure the output frequency $fosc$ of ROSCr;
III. Set the next temperature in the 1st step and repeat 2nd step to get the required output values fosc for ROSCr;
IV. Apply curve fittings for Eq. (3.6) and derive the coefficients of B, k and α.

The effect of sweeping Vclb is equivalent to varying V_{GS}. Results of curve fittings at three different temperatures of the output of the monitor with the swept ΔV_{GS} values are shown in Figure 3.4. The fitted results for B, k and α are 767.2, 0.14 and 0.06 respectively.

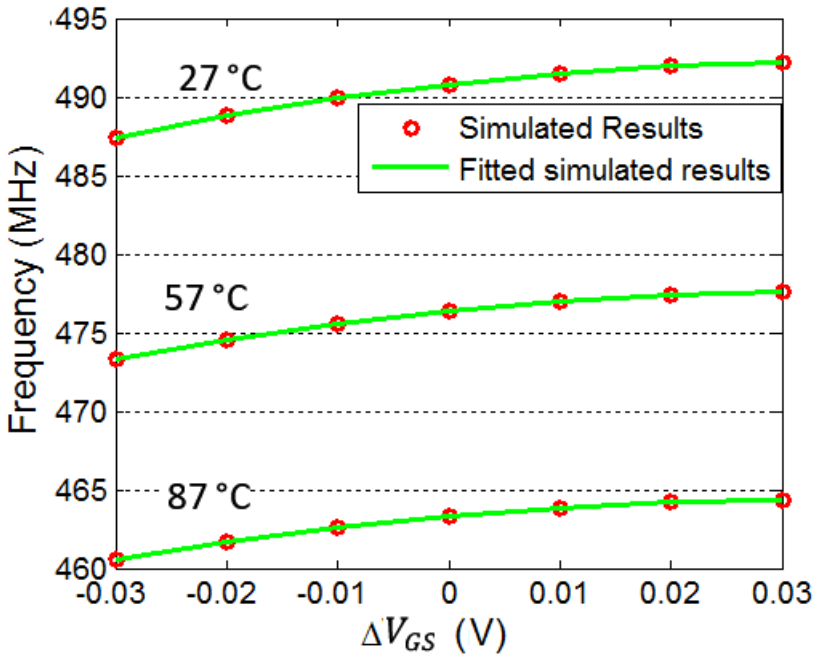


Figure 3.4: Calibration of the ROSCr with ΔV_{GS} at three different temperatures. The dots represent simulated discrete sweeping outputs with respect to three different temperatures under different operating voltage settings, and the solid lines are the fitted results.

B. Simulation of our monitor as temperature monitor

Figure 3.5 shows the result of the curve fitting of the monitor output at different temperatures in three supply-voltage cases. As one can observe, the output frequency range changes from 490 MHz up to 540 MHz over a 27 °C to 127 °C temperature span in the typical case of 1.2 V. However, the supply voltage has a high impact on the temperature monitor (75 MHz difference for delta of 0.1 V V_{DD} at 27 °C), and therefore in this mode, the operating voltage for ROSCr will be kept stable, meaning the operational supply will be connected to V_{DD} (1.2 V) in Figure 3.3. As will be discussed later on, the ring oscillator shows a good sensitivity with respect to measuring temperatures.

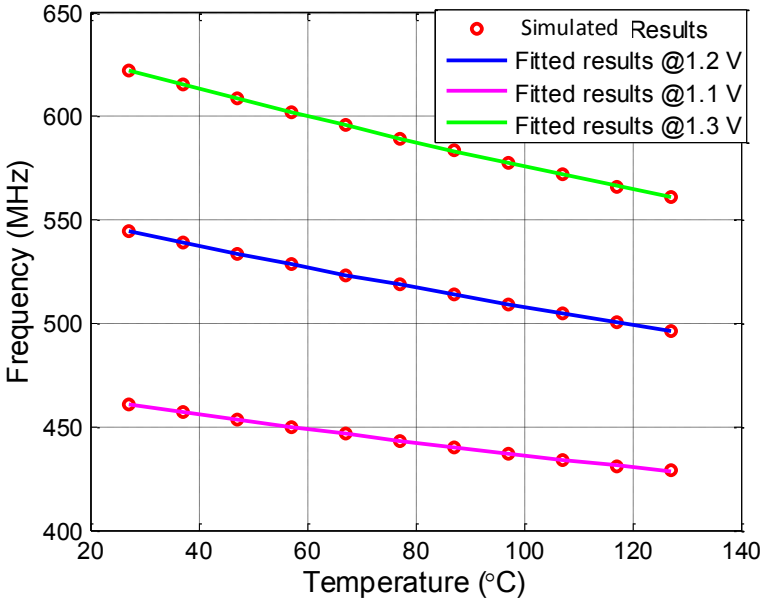


Figure 3.5: Discrete simulated oscillator frequencies resulting from temperature sweeps (from 27 °C to 127 °C with a step of 10 °C) under the condition of three different supply voltages. The solid line is the interpolation between the dots of the simulation results.

C. Simulation of our monitor as voltage monitor

Figure 3.6 shows the result of the curve fitting at different operating voltages in the case the temperature is 27 °C. As one can observe, the output frequency range changes from 400 MHz up to 700 MHz over a 0.35 V change in voltage. For operating into this mode, the temperature has to be monitored first, and subsequently the voltage supply of ROSCr will be connected to Vut in Figure 3.3 to enter this monitoring mode. As will be discussed later on in more detail, the ring oscillator shows a good sensitivity with respect to measuring voltages.

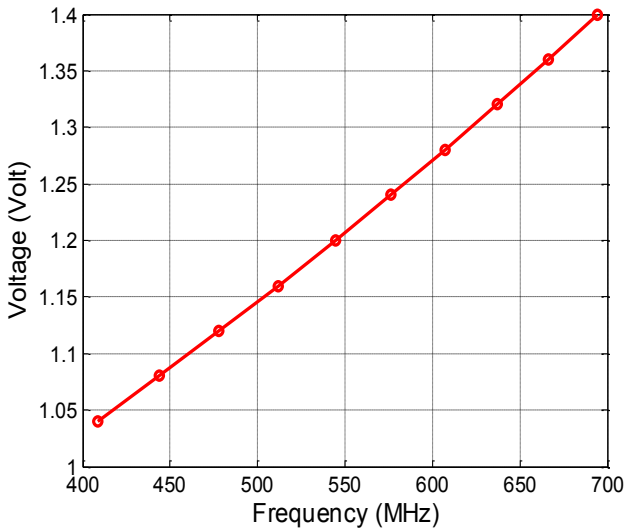


Figure 3.6: Discrete simulated oscillator frequency resulting from voltage sweeps from 1.04 V to 1.4 V; the temperature is 27 °C. The solid line is the interpolation between the dots of the simulation results.

D. Simulation of our monitor as aging monitor

For the simulations with respect to the delay-time increase, the NBTI power model (PM) degradation [Kole 07] has been applied. The PMOS transistor threshold degradation can be written as:

$$\Delta V_{TH} = A \cdot \exp(\gamma \cdot V_{Stress}) \cdot \exp\left(-\frac{E_a}{kT}\right) t_{stress}^{n} \qquad \text{Eq. (3.8)}$$

The parameter ΔV_{TH} denotes the voltage-threshold degradation, V_{Stress} the power-supply voltage, T the temperature, k the Boltzmann constant, and t_{stress} the stress time. The fitting parameters γ, A, n and E_a have to be extracted from actual stress measurements.

In this aging monitoring mode, the frequency output of ROSCut in Figure 3.3 will be mapped to the delay of itself. If the clock skew factor is small, the reciprocal of the frequency can be assumed to be the delay [Bowm 02]. Figure 3.7 shows the frequency/delay simulation result of the aging effect on our designed monitor under three operating conditions. The life time for aging is set to 10^8 seconds (150 days). One can observe that the higher the temperature, the lower the ring-oscillator frequency will become; another remark is that an increased supply voltage results in a higher ring-oscillator frequency. This occurs because the temperature is a strong function of the

number of broken Si-H bonds [Kunh 07]. The contribution of hole trapping to the V_{TH} shift is weakly dependent on the temperature but strongly dependent on the stress voltage. However, one can observe in Figure 3.7 that the voltage variation is much larger than aging effect, and the temperature variation is dramatically larger. Therefore, the monitor requires a stabilized voltage and temperature environment.

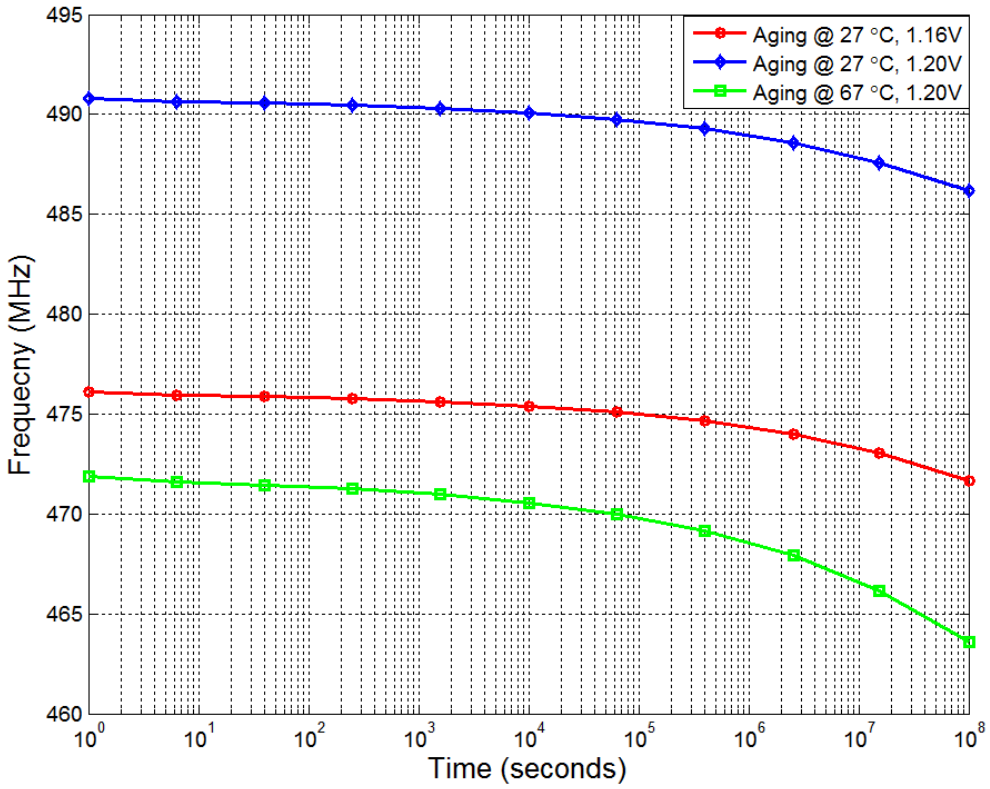


Figure 3.7: The aging simulation during 10^8 seconds (150 days) of lifetime under three different operating conditions of temperature and supply voltage.

In general, as a function of the time of operation, the monitor frequency shows a decreasing trend and hence an increasing delay. However, compared with the voltage monitoring result, the aging effect has less influence (frequency decrease) on the monitor.

3.3.4 Consideration of Process Variations in the Monitor

Besides fluctuations in operating conditions and the aging degradation, the process variations also add to the randomness in the monitor output as e.g. V_{TH} differs among devices. As shown in Figure 3.8 under the operating temperature of 27 °C and voltage supply of 1.2 V, in case of a maximum V_{TH} process variation of 1% (490.8 MHz to 491.2 MHz), a maximum error of $\frac{491.18-490.81}{490.81} = 0.075\%$ for the monitor frequency output is observed. This shows a weak relationship between innate V_{TH} process variations and the output frequency.

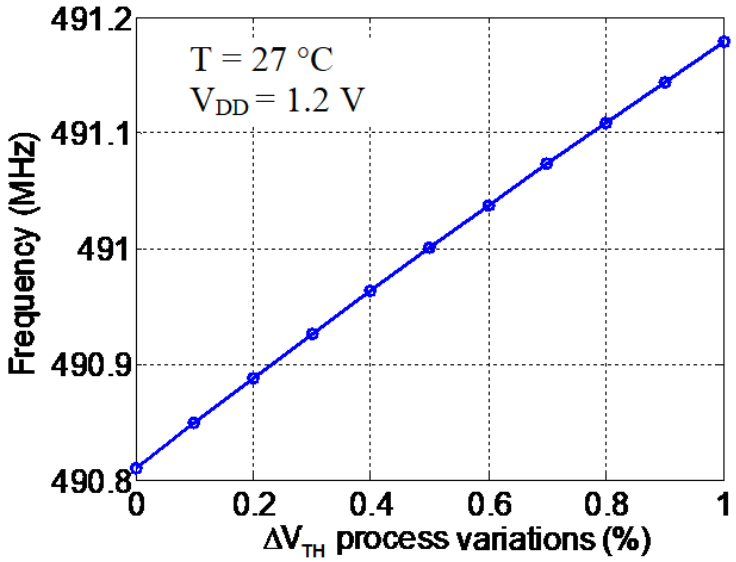


Figure 3.8: The monitor output frequency if the process variation (V_{TH}) ranges from 0 to around 1% of the normal initial MOSFET transistor threshold voltage V_{TH0} (around 490 mV).

The countermeasures taken for compensating the process variations are that normally a number of our monitors are distributed in and around each core in the MP-SoC, so that at the data-process level of health monitors for the life-time prognostics, a statistical analysis can compensate the process variation [Lee 04].

The benefits of the embedded health-monitoring approach are that since the monitors use the same process technology and work under nearly similar conditions as the target processor core, the process variations can be approximately estimated as the same during aging degradation.

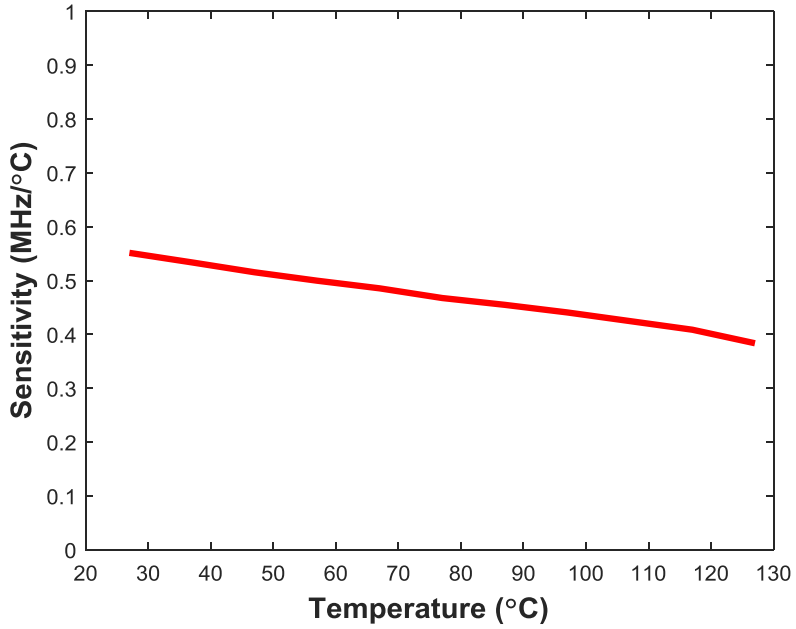
3.3.5 SENSITIVITY ANALYSIS OF THE MONITOR

A sensor's sensitivity indicates how much the sensor's output changes related to the input quantity changes. For the designed ring-oscillator based monitor, the dynamic sensitivity of key monitoring parameters (T, V and NBTI aging, as input) have been calculated from simulation data. As can be observed from the Figures 3.9a, b and c, all sensitivities decrease with the increased monitored parameter. In particular, the temperature sensitivity almost linearly decreases from 0.55 MHz/°C to 0.4 MHz/°C (measured outputs from 500 MHz to 540 MHz at 1.2 V in Figure 3.5) with an average of around 0.5 MHz/°C.

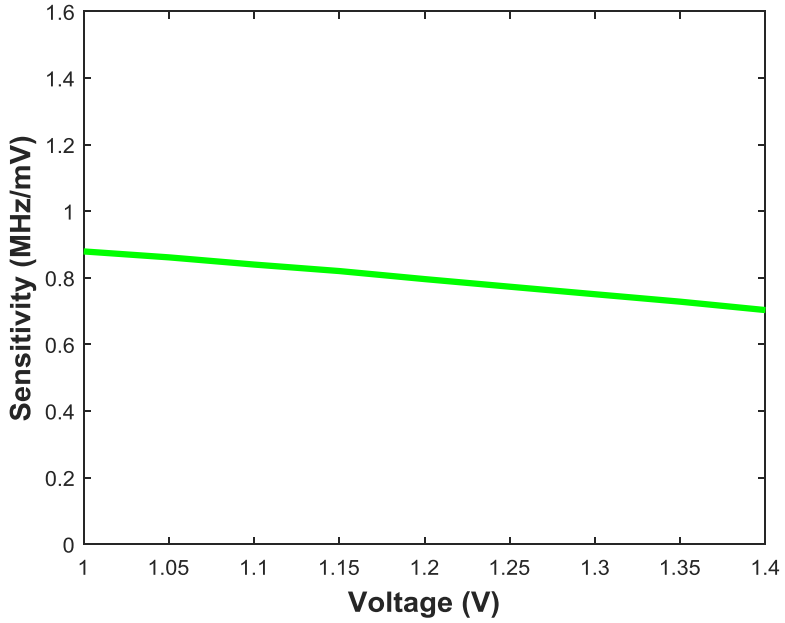
The voltage sensitivity also shows a near linear correlation relationship with its value, the average sensitivity (from 0.9 MHz/mV to 0.7 MHz/mV with outputs from 410 MHz to 690 MHz in Figure 3.6) being around 0.8 MHz/mV.

The aging sensitivity is much higher in the first two years (1~7 MHz/year); however, it will decrease significantly afterwards to less than 1 MHz/year level.

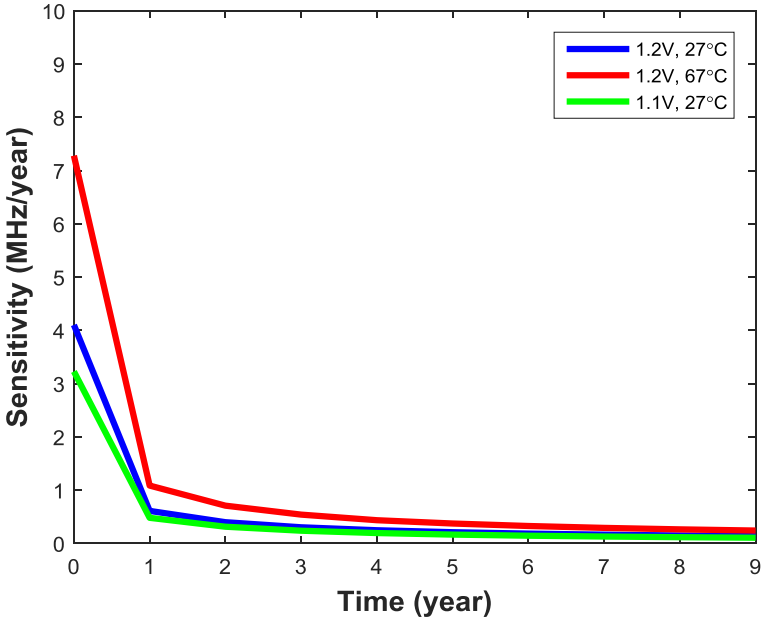
However, by using several of these monitors and activate them later in time (fresh device), a high sensitivity can be guaranteed over a longer time span at the cost of increased silicon area. Especially the aging sensitivity exhibits an increased sensitivity if temperature or voltage increases. Therefore, the aging monitor has to be employed in the case when the temperature and voltage readings are stabilized. In general, it is shown above that the monitor exhibits a good and relatively stabilized sensitivity to the changing parameters.


Sensitivity (MHz/°C)
Temperature (°C)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
20
30
40
50
60
70
80
90
100
110
120
130

a)


Sensitivity (MHz/mV)
Voltage (V)
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1
1.05
1.1
1.15
1.2
1.25
1.3
1.35
1.4

b)



c)

Figure 3.9: Sensitivities (frequency displacement per monitored parameters (T, V, aging)) in the case the EI function is monitoring (a) The temperature if operational supply voltage is 1.2 V, (b) The voltage if temperature is 27 °C, (c) The NBTI aging in three scenarios (1.2 V, 27 °C; 1.2 V, 67 °C; 1.1 V, 27 °C).

3.3.6 Overhead Evaluation of the Monitor Area

Table 3.1 shows the different parameters such as area overhead of the two discussed methods; one is the BISTR method while the second one concerns the health-monitoring based method.

Note that the number of logic gates is calculated in NAND-2 gate equivalents (4 transistors). One difference concerning the setup of these two methods is that there will be one dependability manager and nine Xentium cores in one RFD chip (Figure 2.2); in the PHM method, the number of the health monitors per Xentium core is set to 10.

One can observe that in the latter case the area overhead is reduced around 11 times as compared to the first approach, because of less logic in the designed health monitors.

Table 3.1: Area overhead comparison between the BISTR based method and the PHM based method; note that a similar processor core is assumed in both methods.

	The BISTR based method	**The PHM based method**
Number of logic gates per Xentium core (in NAND-2 equivalent)	346521	346521
Total number of Xentium cores in one chip	9	9
Number of logic gates for dependability	382166	~300
Area overhead for dependability	10.9%	0.9%

3.4 POWER-DISSIPATION OF THE HEALTH-MONITORING INFRASTRUCTURE

The previous BISTR approach for dependability detects a faulty Xentium processor, which is a reaction *after* the occurrence of a fault. The health monitoring and prognostics approach is predicting (action *before* fault occurrence) at which moment a processor core becomes a potential risk for correct operation with a certain confidence (e.g. 90%). In this case, relevant parameters are monitored per Xentium processor core (like temperature, delay, voltage and current), which relate to its life-time (i.e. health). To be able to compare this approach to the previous one in terms of power dissipation, the same dependability attribute specifications and conditions have been used as listed in Table 2.1 for our target system. It is noted that for both dependability approaches the same mission profile has been assumed.

In the health-monitoring approach, some features regarding the power dissipation are:

- during the dependability test for all Xentium cores, only current and voltage monitors are connected to the local IP pins

- full on-line operation of the NoC for normal functional applications and health monitors measurement data transfer in parallel is possible

During the HM procedure, the first step in the dependability test of the HM prognostics approach (Figure 3.1b) is to send a message from an ARM core via the NoC to (a) Xentium core(s) which health monitor(s) have to start a measurement. Next, a monitor has to start measuring the parameter concerned, e.g. the temperature, and in case of an analogue output, it has to be converted into a digital word in the analogue front end (AFE) in the wrapper of the monitor (Figure 3.1a). This data is subsequently sent (in parallel with the normal application data) over the NoC (typically an IJTAG network nowadays) to the ARM core (Figure 3.1b). Based on this data, the ARM updates the lifetime prediction of this Xentium core and decides to label it correct or potentially faulty. In the same way as in the BISTR approach, the repair actions can be carried out being to take out a potentially faulty core, remap its task to a spare processor and connect it. This is the reason why this last part has not been included in the power-dissipation comparison calculations.

From the previous remarks it is clear that the health monitors and AFE, NoC and one ARM core are involved in the dependability-test power dissipation. Their contributions to power dissipation will be explained in detail below and reveal its major contributors. Notice that the Xentium core itself is not used at all in terms of its logic in the decision process.

3.4.1 Power Dissipation of the Health Monitors

Several options for on-chip monitors for health monitoring in terms of power dissipation have been investigated: temperature T, I_{DD}, voltage and delay monitors. Their specifications and power dissipation are summarized in Table 3.2. Our current approach will use four temperature monitors at different locations around the core; two voltage monitors are employed, and a single delay and a current monitor.

The total power dissipation of the health monitors for a single Xentium is 11.3 mW (V_{DD} @ 3.3 V) using Table 3.2 [Zhao 13].

Table 3.2: Specifications and static power dissipation of four types of health monitors.

Monitor Type	Range	Resolution	Power (mW)
Temperature [Wang 13]	-50 ºC ~ 150 ºC	± 0.8 ºC	0.0037
Current I_{DDQ} [Maed 13]	0 – 100 µA	50 nA	0.160
Voltage [Qazi 11]	0 - 1.2 V	1 mV	5.5
Delay [Fick 10]	0 - 4 ns	0.8 ps	0.11

3.4.2 ANALOG FRONT END POWER DISSIPATION

In the health monitoring system approach, the AFE (e.g. for temperature) is optional and it will only be active in case the analogue monitors are active. The current AFE consists of an OpAmp and an ADC. The continuous power consumption of our OpAmp with a high gain (>50 dB) is 1.4 mW; in the case of our pipelined ADC with an accuracy of 12-bits, the continuous power consumption is 4.1 mW [Wan 14]. Hence in the case of a shared AFE this results in 5.5 mW. As a continuous operation is not necessary even for multiplexed health monitors, a practical operation below 1 mW is possible. Purely digital health monitors are obviously always preferred and the future way to go.

3.4.3 NOC POWER DISSIPATION

In the health-monitoring approach, only very few simple monitor commands like e.g. “start measurement” and few data such as periodic measurement results of around eight times 12-bit words are transported over the NoC to an embedded processor (e.g. ARM926) every time. Hence its contribution to power dissipation is negligible. Note that

nowadays, the communication for health monitors is more often accomplished via the IJTAG network.

3.4.4 POWER DISSIPATION OF THE ARM CORE

The tasks of the embedded ARM core are significant in the health-monitoring approach for MP-SoCs. First, some basic control has to be provided such as generation / transmission of Xentium core headers and monitor(s) assignments to start measurements. This requires a maximum of 100 instructions. The same holds for closing the HM activity by the ARM. The next step requires reading of 12-bits data of e.g. 10 monitors at a time from the NoC and to store this data with a time stamp. This step requires around 1300 instructions [Tiwa 94]. However, the most complex part for the ARM is the life-time prediction calculation. An example of a combined linear/exponential stochastic degradation path model shows this will require a program size of around 15 kB [Tiwa 94]. The amount of instruction that is executed is roughly 50k [Tiwa 94]. The last step involves storage of crucial (abstracted) data from the calculations in the memory requiring a maximum of 100 instructions. The total number of executed instructions in the ARM core for determining the lifetime for a Xentium core, and storing crucial data is therefore 51.6k instructions. Following the power-estimation model of an instruction set simulator (ISS) as suggested in [Arm 00], based on actual values of the physical current during execution of an instruction, the total software-program power consumption W_p can be written as:

$$W_p = V_{DD} \cdot \sum_{i=0}^{M-1} (I_i \cdot N_i) \qquad \text{Eq. (3.9)}$$

where V_{DD} denotes the ARM supply voltage (1.2 V), M the number of total tasks, N_i the total instructions during executing task i, and I_i is the current during the execution of each instruction for the task i. The power consumption for software is the difference that the ARM core processor is turned from idle mode (W_{idle}) to the active mode (W_{act}). The average data throughput N_u for our ARM processor is 200 MIPS (million instructions per second) [Arm 00]. The average current per instruction can be calculated from:

$$I_u = \frac{W_{act} - W_{idle}}{V_{DD} \cdot N_u} \qquad \text{Eq. (3.10)}$$

This results in our case in 0.15 nA per instruction. The total software power dissipation for the ARM is hence 51.6k multiplied by the above current per executed instruction, being 9.39 mW. Using the idle power value of the ARM (34 mW) and the previous program dissipation, a total of around 44 mW results for the contribution of the ARM core.

3.4.5 Additional Power Dissipation for the Dependability Test

One may conclude that the health monitors (11.3 mW) and AFE power dissipation (5.5 mW) are responsible for 16.8 mW in the continuous mode for a single Xentium core. The ARM power dissipation (44 mW) is the other power contributor, amounting to a total of 60.8 mW for a single Xentium during 258 μs (51.6k instructions). For three Xentiums the time required will be 774 μs. This power dissipation is around 27% of the BISTR approach (225 mW during 600 ms [Zhao 13]).

Hence it is preferred to have the health-monitoring based approach for dependability in our MP-SoC in terms of power dissipation.

3.5 CONCLUSIONS

In this chapter, an embedded health-monitoring approach for a new highly dependable MP-SoC has been proposed. A programmable all-in-one health monitor has been designed which is capable of carrying out voltage and temperature measurements as well as non-invasive delay-time monitoring. Features include an average temperature sensitivity of 0.5 MHz/°C and range of 27 °C to 127 °C, a voltage sensitivity of 0.8 MHz/mV and range of 1 V to 1.4 V and finally aging of 4 MHz/year (in the first year at 1.2 V, 67 °C) respectively. Simulation results show that the monitor exhibits a good and relatively stabilized sensitivity to the changing parameters considering the (innate V_{TH}) process variations.

This chapter provides a foundation for our life-time prognostics and guarantees a high dependability with 100% availability for a certain lifetime of a very complex dependability management system. Besides that, the silicon area for the PHM approach has been compared to the BISTR method, resulting in a decrease of around 10 times. Furthermore, the power-dissipation for the PHM approach was less than one third (27%) of the BISTR approach, showing it to be a real energy-saving solution.

REFERENCES

[Ali 16] G. Ali, A. Badawy, and H. G. Kerkhoff, "Accessing on-chip temperature health monitors using the IEEE 1687 standard," in IEEE International Conference on Electronics, Circuits and Systems (ICECS), pp. 776-779, 2016.

[Arm 00] ARM Limited. "ARM9TDMI (Rev 3) Technical Reference Manual." Available: http://infocenter.arm.com/help/topic/com.arm.doc.ddi0180a/DDI0180.pdf， 2000.

[Bara 13] R. Baranowskia, A. Cook, M.E. Imhof, C. Liud, and H.-J. Wunderlich, "Synthesis of Workload Monitors for On-Line Stress Prediction," in IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), New York City, USA, pp. 137-142, 2013.

[Bowm 02] K. A. Bowman, S. G. Duvall and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," in IEEE Journal of Solid-State Circuits, vol. 37, no. 2, pp. 183-190, Feb. 2002.

[Chan 14] T.-B. Chan, P. Gupta, A. B. Kahng, and L. Lai, "Synthesis and analysis of design-dependent ring oscillator (ddro) performance monitors," in IEEE Transactions on Very Large Scale Integration (VLSI) systems, vol. 22, pp. 2117-2130, 2014.

[Ecke 12] M. Ecker, J. B. Gerschler, J. Vogel, et al., "Development of a lifetime prediction model for lithium-ion batteries based on extended accelerated aging test data," in Journal of Power Sources, pp. 248-257, 2012.

[Farr 07] C. R. Farrar and K. Worden, "An introduction to structural health monitoring," in Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 365, pp. 303-315, 2007.

[Fick 10] D. Fick, N. Liu, Z. Foo, et al., "In situ delay-slack monitor for high-performance processors using an all-digital self-calibrating 5ps resolution time-to-digital converter," in IEEE International Solid-State

Circuits Conference Digest of Technical Papers (ISSCC), pp. 188-189, 2010.

[Hong 15] H. Hong, J. Lim, H. Lim, and S. Kang, "Lifetime reliability enhancement of microprocessors: mitigating the impact of negative bias temperature instability," in ACM Computing Surveys (CSUR), vol. 48, pp. 9-34, 2015.

[Ibra 16a] A. Ibrahim and H. G. Kerkhoff, "Analysis and design of an on-chip retargeting engine for IEEE 1687 networks," in 21th IEEE European Test Symposium (ETS), pp. 1-6, 2016.

[Ibra 16b] A. Ibrahim and H. G. Kerkhoff, "An IJTAG-compatible I_{DDT} Embedded Instrument for Health Monitoring and Prognostics," in IEEE International Test Conference (ITC), poster session, 2016.

[Ieee 16] "IEEE P1687.1, Standard for the Application of Interfaces and Controllers to Access 1687 IJTAG Networks Embedded Within Semiconductor Devices", 2016.

[Kane 10] N. Kanekawa, E. H. Ibe, T. Suga, et al., "Dependability in electronic systems: mitigation of hardware failures, soft errors, and electro-magnetic disturbances," in Springer Science & Business Media, ISBN: 978-14-419-6715-2, 2010.

[Karl 08a] E. Karl, P. Singh, D. Blaauw, and D. Sylvester, "Compact In-Situ Sensors for Monitoring Negative-Bias-Temperature-Instability Effect and Oxide Degradation," in IEEE International Solid-State Circuits Conference (ISSCC), pp. 410-623, 2008.

[Karl 08b] E. Karl, D. Blaauw, D. Sylvester, and T. Mudge, "Multi-Mechanism Reliability Modeling and Management in Dynamic Systems," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 16, pp. 476-487, 2008.

[Kean 10a] J. Keane, T.-h. Kim, X. Wang, and C. H. Kim, "On-chip reliability monitors for measuring circuit degradation," in Microelectronics Reliability, vol. 50, pp. 1039-1053, 2010.

[Kean 10b] J. Keane, K. Tae-Hyoung, and C. H. Kim, "An On-Chip NBTI Sensor for Measuring pMOS Threshold Voltage Degradation," in IEEE

Transactions on Very Large Scale Integration (VLSI) Systems, vol. 18, pp. 947-956, 2010.

[Kerk 10] H. G. Kerkhoff and X. Zhang, "Design of an Infrastructural IP Dependability Manager for a Dependable Reconfigurable Many-Core Processor," in Proceedings of the IEEE International Symposium on Electronic Design, Test & Applications (DELTA), Ho Chi Minh City, Vietnam, pp. 270-275, 2010.

[Kerk 12] H. G. Kerkhoff and Y. Zhao, "The design of dependable flexible multi-sensory System-on-Chips for security applications," in IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), Tallinn, Estonia, pp. 133-138, 2012.

[Kole 07] M. Kole, "Circuit realiability simulation based on Verilog-A," in IEEE International Behavioral Modeling and Simulation Workshop (BMAS), San Jose, CA, pp. 58-63, 2007.

[Kunh 07] K. Kunhyuk, K. Keejong, A. E. Islam, et al., "Characterization and Estimation of Circuit Reliability Degradation under NBTI using On-Line I_{DDQ} Measurement," in 44th ACM/IEEE Design Automation Conference (DAC), pp. 358-363, 2007.

[Lee 04] J.-M. Lee, C. Yoo, I. Lee, "Statistical process monitoring with independent component analysis," in Journal of Process Control, vol. 14, pp. 467-485, 2004.

[Liu 11] T. Liu, Y. Zhao, C. J. Xue, and M. Li, "Power-aware variable partitioning for DSPs with hybrid PRAM and DRAM main memory," in the 48th Design Automation Conference (DAC), San Diego, California, pp. 405-410, 2011.

[Maed 13] N. Maeda, S. Komatsu, M. Morimoto, et al., "A 0.41 μA Standby Leakage 32 kb Embedded SRAM with Low-Voltage Resume-Standby Utilizing All Digital Current Comparator in 28 nm HKMG CMOS," in IEEE Journal of Solid-State Circuits, vol. 48, pp. 917-923, 2013.

[Paul 11] P. Paulin, "Programming challenges & solutions for multi-processor SoCs: an industrial perspective," in the 48th Design Automation Conference (DAC), San Diego, California, pp. 262-267, 2011.

[Qazi 11] M. Qazi, K. Stawiasz, L. Chang, and A. P. Chandrakasan, “A 512kb 8T SRAM macro operating down to 0.57 V with an AC-coupled sense amplifier and embedded data-retention-voltage sensor in 45 nm SOI CMOS, ” in IEEE Journal of Solid-State Circuits, vol. 46, pp. 85-96, 2011.

[Saye 11] M. A. Sayed and P. H. Jones, “Characterizing non-ideal impacts of reconfigurable hardware workloads on ring oscillator-based thermometers,” in Proc. International Conference on Reconfigurable Computing and FPGAs, pp. 92-98, 2011.

[Shib 16] K. Shibin, S. Devadze, and A. Jutman, “On-line fault classification and handling in IEEE1687 based fault management system for complex SoCs,” in 17th Latin-American Test Symposium (LATS), pp. 69-74, 2016.

[Tae 08] K. Tae-Hyoung, R. Persaud, and C. H. Kim, “Silicon odometer: an on-chip reliability monitor for measuring frequency degradation of digital circuits,” in IEEE Journal of Solid-State Circuits (JSSC), vol. 43, pp. 874-880, 2008.

[Tiwa 94] V. Tiwari, S. Malik, and A. Wolfe, “Power analysis of embedded software: a first step towards software power minimization,” in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 2, pp. 437-445, 1994.

[Tzu 12] C. Tzu-Ting, H. Po-Tsang, C. Ching-Te, et al., “On-chip self-calibrated process-temperature sensor for TSV 3D integration,” in IEEE International SOC Conference (SOCC), Niagara Falls, NY, pp. 370-375, 2012.

[Vazq 09] J. C. Vazquez, V. Champac, A. M. Ziesemer, et al., “Built-in aging monitoring for safety-critical applications,” in IEEE International On-Line Testing Symposium (IOLTS), pp. 9-14, 2009.

[Wan 14] J. Wan and H. G. Kerkhoff, “An embedded offset and gain instrument for OpAmp IPs,” in Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1-4, 2014.

[Wang 13] C.-C. Wang, C.-L. Chen, R.-C. Kuo, et al., "On-Chip Process and Temperature Monitor for Self-Adjusting Slew Rate Control of 2 times V_{DD} Output Buffers," in IEEE Transactions on Circuits and Systems, vol. 60, pp. 1432-1440, 2013.

[Wang 15] X. Wang, Q. Tang, P. Jain, et al., "The dependence of BTI and HCI-induced frequency degradation on interconnect length and its circuit level implications," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 23, pp. 280-291, 2015.

[Yang 11] S.-Y. Yang, C. A. Papachristou, and M. Tabib-Azar, "Improving bus test via I_{DDT} and boundary scan," in 38th annual Design Automation Conference (DAC), Las Vegas, Nevada, USA, pp. 307-312, 2001.

[Zade 12] F. G., Zadegan, U. Ingelsson, et al., "Reusing and retargeting on-chip instrument access procedures in IEEE P1687," in IEEE Design & Test of Computers, vol. 29, pp. 79-88, 2012.

[Zhan 11] X. Zhang and H. G. Kerkhoff, "A Dependability Solution for Homogeneous MPSoCs," in IEEE 17th Pacific Rim International Symposium on Dependable Computing, Pasadena, USA, pp. 53-62, 2011.

[Zhao 13] Y. Zhao, X. Zhang, and H. G. Kerkhoff, "Power Dissipation Comparison of two dependability Approaches for MP-SoC," in 8th International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), Abu Dhabi, UAE, pp. 56-61, 2013.

[Zhao 14] Y. Zhao and H. G. Kerkhoff, "Design of an Embedded Health Monitoring Infrastructure for Accessing Multi-processor SoC Degradation," in 17th Euromicro Conference on Digital System Design (DSD), Verona, Italy, pp. 154-160, 2014.

[Zhao 16] Y. Zhao and H. G. Kerkhoff, "Highly Dependable Multi-processor SoCs Employing Lifetime Prediction Based on Health Monitors," in Proc. IEEE 25th Asian Test Symposium (ATS), Hiroshima, Japan, pp. 228-233, 2016.

Chapter 4

SOFTWARE-BASED HEALTH MONITORING FOR DEPENDABLE MP-SOCS

ABSTRACT *– In contrast to the previous technique of health monitors, this chapter proposes a software-based health monitoring (SBHM) method for dependable SoC design, by monitoring the actual processor cores directly via developed software programs. Parameters concerned with the reliability monitoring are discussed, which show that delay, quiescent and transient currents are proper candidates in theory. The major part of this chapter deals with the health monitoring program designed for the target monitored DSP processor - the Xentium®. The proposed monitoring technique is actually a software-based self-test (SBST). This is the first time the SBST technique is being applied in the health monitoring field for a processor. Its implementation, validation and accuracy will be described and verified by the aging test described in the next chapter.*

Parts of this chapter have been published as paper titled "Application of Functional I_{DDQ} Testing in a VLIW Processor towards Detection of Aging Degradation," in the International Conference on Design & Technology of Integrated Systems, 2015 [Zhao 15a], and "Unit-Based Functional I_{DDT} Testing for Aging Degradation Monitoring in a VLIW Processor," in the Euromicro Conference on Digital System Design, 2015 [Zhao 15b].

4.1 INTRODUCTION AND MOTIVATION

As introduced previously, due to the continuous scaling in the semiconductor manufacturing technology and high system complexity, processor chips face growing reliability problems. Aging has been recognized as a significant phenomenon in transistors and circuits. Aging mechanisms [Niga 09], e.g. NBTI, HCI and TDDB will result in the deterioration of circuit performance over its lifetime. The previous chapter proposed health monitors embedded in the target MP-SoC (around the monitored processor cores) for reliability enhancement. Degradation data of the monitors is subsequently used to estimate the degradation of the actual devices in the chip. However, there are chances that the monitors do not exactly match the degradation process of the monitored processor. Moreover, a sufficient number of such monitors needs to be considered to reach a high monitoring accuracy, leading to more area overhead in the circuit design.

In this chapter, we propose a software-based health-monitoring (SBHM) technique for a target processor, and comparing it with the previous embedded monitors design. With this technique, the target processor is measured directly, and thereafter its degradation is determined. The measured parameters are direct outputs from the target processor, instead of resulting from the embedded monitors. This approach can therefore address all sources (such as aging and process variations) of lifetime changes of the monitored processors, providing the most accurate monitoring accuracy. Furthermore, with the increasingly operational frequencies of integrated circuits, testable design becomes more difficult for the embedded health monitors. For instance, in the monitoring of performance parameters such as critical-path delay, it normally requires at-speed testing. The SBHM solutions are being investigated in this chapter to provide a reliability-related degradation observation with an analysis of the required monitoring infrastructures, programs and test time.

Nowadays, a popular testing technique for processors and processor-based systems (e.g. Systems-on-Chips) is called Software-Based Self-Test (SBST) [Psar 10]. The basic idea is generating test programs to be executed by a processor, and sequentially testing the processor itself or other components in the system, in order to detect possible faults in the produced results. The main advantage of this technique is that it does not require any extra hardware, thus avoiding an area penalty. Moreover, SBST methods can be

carried out at-speed. For these reasons, SBST is increasingly applied for processor control and observation, and it is often used in combination with other testing approaches. However, in order to execute the SBST, the processor has to be off-line. Therefore, a spare processor is required while executing the SBST on a target processor core.

Our proposed SBHM technique is based on the SBST design. However, the purpose is to observe degradation or obtaining aging information of the monitored processor, instead of testing for fault detection. It is known that the aging behaviour will alter the transistor threshold voltage (V_{TH}) with time [Wang 07], thereby leading to changes in parameters such as critical delay, quiescent and transient (I_{DDQ} and I_{DDT}) power-supply currents. For instance, it was reported that V_{TH} can change by 15% while the leakage current increases by a factor of 5 for each technology node as a result of aging [De 99]. The path delay, I_{DDQ} and I_{DDT} monitoring techniques have been developed and realized based on our SBHM program.

Our proposed SBHM method has been experimentally evaluated on a Very Large Instruction Word (VLIW) processor, the Xentium [Reco 11], which is part of a reconfigurable Many-Processor SoC (MP-SoC). In modern IC applications, reconfigurable MP-SoCs are increasingly used in different domains. This is because they can be easily configured to match the specific requirements (e.g. performance, area and power consumption) of the target application. VLIW processors represent a popular choice among reconfigurable processors.

The basic idea for the dependability management is that the health monitoring will take place in the Xentium-based MP-SoC during a *non-operational* phase for the health monitoring, since on-chip functional units are required. Therefore, a spare Xentium is required for executing the health-monitoring. Next, the health-monitoring data will be evaluated. If a failure is predicted in one Xentium processor core by the prognostic model to be discussed later on, a spare core will be activated for substitution. This is referred to as the remapping process [Braa 16]. Since a periodic health-monitoring will take place in all processors, a rotation of remapping is required.

The major challenge for the proposed technique is to develop a suitable SBHM program for Xentium processors in the SoC. In general, with respect to delay monitoring, the approach is to explore the maximum speed it can reach by several functional verification programs at the given supply voltage. In I_{DDQ} monitoring, however, the

Xentium needs to be set to a static state, i.e. the "WAIT" state. In contrast, in I_{DDT} monitoring, different functional units in the Xentium datapath are controlled in such way that they will function in a serial sequence, thus avoiding parallelism of the VLIW processor characterization.

Our delay monitoring is validated by means of an evaluation board, while the I_{DDQ} and I_{DDT} monitoring is measured employing an HTOL board. The difference is that delay monitoring needs no extra hardware, but the $I_{DDQ/T}$ monitoring is based on our SBHM and measured by the Ridgetop current sensor QT1411 [Ridg 12]. This chapter will deal with the SBHM programs, which is the software point of view. The effectiveness of the designed monitoring program, and detailed hardware and software setup for the health monitoring will be demonstrated in the next chapter.

4.2 The Concept of Software-based Health Monitoring for SoCs

SBHM is a completely different approach to enhance the dependability of multi-processor Systems-on-Chips (SoCs), as compared to the previous health-monitoring technique with embedded monitors in the SoC. This section first introduces the concept of the Software-Based Self-Test (SBST) technique, and then the SBHM and our new SBHM applied to a Very Large Instruction Word (VLIW) processor is discussed.

4.2.1 Software-Based Self-Test (SBST)

Nowadays, the most popular technique regarding the testing of many-processor SoCs is the Software-Based Self-Test (SBST), also referred to as a form of functional testing [Psar 10]. In the first place, the latest technologies show that timing-related faults can be detected more accurately using SBST, since it allows executing a test at the full chip speed. Second, it is cost as well as time effective, since fewer tester channels and also less memory is required on the tester. Last but not least, these tests can be carried out on the target processor SoCs within the life-time span.

Many efforts have been made in the past decades with regard to the development of SBST. In general, the principle of SBST is to run functional test patterns based on the

processor architecture by for instance exploiting processor resources to test the processor itself and the components around it [Psar 10]. The purpose is to apply a special test program to the target processor. This can be some algorithms with a sequence of instructions capable of detecting possible faults in the specified location of the processor by observing the produced results. SBST does not require circuit modifications, making it particularly suitable for testing third-party processors that can hardly be modified [Port 12].

The application area of SBST mainly focuses on testing stuck-at faults [Rief 16] as well as delay faults [Guru 07] in the processor. With regard to the applicability, SBST can already be found in the manufacturing flow of microelectronics, e.g. in [Bern 11] where the author describes an industrial case study. Moreover, SBST can be used to identify faults in the normal lifetime by performing on-line testing [Port 12]. Besides that, the SBST technique can also be used for diagnostic purposes [Lago 07].

Current techniques include various strategies to generate test programs either manually or automatically. For example, grading techniques to characterize processors with test programs have been proposed in [Psar 10]. However, reducing costs related to both test-program fault grading and test application is still an open question. The efficient generation of SBST programs is the main critical issue; an alternative approach aimed at reducing the efforts related to the test-program development is using particular tools to generate pseudo-random patterns [Batc 99].

4.2.2 SOFTWARE-BASED HEALTH MONITORING (SBHM)

The reliability enhancement in the previous chapter is a degradation health monitoring-based approach. Voltage and temperature monitors are used as environmental monitors. Delay monitors are employed to monitor the aging degradation of transistors of the monitor itself instead of transistors of the target processor. These health monitors are distributed around the target processor, so they can experience the same stress conditions as the embedded processors. However, health monitors do not account for lifetime variations due to the innate randomness of degradation mechanisms. This also holds for process mismatch between the health monitors and the transistors in the processor.

In this chapter, Software-based health monitoring (SBHM) based reliability enhancement is proposed. With this technique the target processor is measured directly to determine its degradation, providing the most accurate monitoring accuracy. Examples of software-based monitoring are [Karl 08], [Woot 12], which focus on monitoring several transistors in the target processor. However, these monitors can hardly be employed in the processor e.g. in a critical path, which may contain hundreds of transistors. Meanwhile other possible parameters such as the critical (path) delay and current monitoring regarding the degradation of the processors need to be investigated. Therefore, we proposed the SBHM approach at the processor level to observe the degradation.

The new SBHM technique proposed in this research aims at monitoring a Very Large Instruction Word (VLIW) processor being used in high-level dependability (e.g. automotive and space) applications, to enhance its dependability during life-time usage. The existing SBST methods for fault detection with an output indicate either faulty or fault-free tests. The primary outputs in our method are health monitoring related measurement results, such as critical delay, the quiescent current (I_{DDQ}) or transient current (I_{DDT}). They can be used in addition for a prognostic model for maintenance purposes. The detailed monitoring program design will be explained in section 4.5. The theoretical analysis for choosing these health-monitoring parameters is explained in the next section.

4.3 Theoretical Analysis of Health Monitoring in a Processor-based SoC

This section discusses the theory behind the usage of health monitoring for estimating the lifetime prediction of processor-based SoCs. First, the propagation delay and analysis of currents with aging at the logic-gate level will be explained based on existing models, and subsequently the resulting behaviour at the circuit level will be deducted. It can be stated that the degradation/health of the cores in SoCs can be obtained by monitoring the degradation of critical delays and currents, which turn out to be highly correlated as shown in our next chapter.

4.3.1 THE MOS TRANSISTOR DEGRADATION MODEL

Because of the scaled technology with the usage of ultra-thin oxide, NBTI has become a significant performance degradation mechanism [Kean 10]. One of the most critical problems regarding NBTI is to characterize its temporal behaviour with aging. At the transistor level, a number of works [Kufl 06], [Cha 14] have reported threshold voltage degradation (ΔV_{TH}) in analytical forms, and they have verified their models with measurement data. Degradation of the PMOS transistor due to NBTI leads to an increase in device V_{TH}. Under DC stress condition, ΔV_{TH} closely follows a power law with respect to time (t):

$$\Delta V_{TH}(t) = K \cdot t^n \qquad \text{Eq. (4.1)}$$

where K is a technology dependent constant, which is determined by temperature, V_{DD} and the transistor geometry [Kufl 06]. The parameter *n* is the coefficient of the power law which particularly depends on the experimental setup. The original RD model assumes a time exponent of $n = 1/4$ [Jepp 77], while in an accelerated testing environment n will be larger than this value.

The drain current of a PMOS transistor in the saturation region can be approximately calculated as [Saku 90]:

$$Id = \mu \cdot B \cdot (V_{GS} - V_{TH})^m \qquad \text{Eq. (4.2)}$$

in which μ is the hole mobility, B is a technology dependent constant determined by the length and width of the P-type transistor and V_{GS} denotes the gate voltage. The parameter m is the carrier-velocity saturation index, which is a natural extension of the historical Shockley model where m is set to 2. For a short-channel PMOS transistor, m is typically around 1.3 [Saku 90]. It has been shown that at a constant load capacitance (C_L) and supply voltage (V_{DD}), the propagation delay (τ) of a logic-gate depends on the saturation drain current (Id) of the PMOS transistor [Paul 05]:

$$\tau = \frac{C_L \cdot V_{DD}}{Id} = \frac{A}{(V_{GS} - V_{TH})^m}, \quad \text{in which } A = \frac{C_L \cdot V_{DD}}{\mu \cdot B} \qquad \text{Eq. (4.3)}$$

The propagation delay due to NBTI-induced ΔV_{TH} can be obtained by taking the differential of the above equation with respect to V_{TH}. Thereafter, applying the Taylor series expansion at V_{TH0}, which is the original V_{TH} without aging, and neglecting higher-order terms, this results in:

$$\tau(V_{TH0} + \Delta V_{TH}) = \tau(V_{TH0}) + \Delta V_{TH} \cdot \tau'(V_{TH0}) \qquad \text{Eq. (4.4)}$$

here V_{TH} is a function of time, and $\tau'(V_{TH0})$ is the first derivative of τ as a function of V_{TH} at the point of V_{TH0}; therefore the propagation-delay degradation $\Delta\tau$ is:

$$\Delta\tau = \tau(V_{TH0} + \Delta V_{TH}) - \tau(V_{TH0}) = \Delta V_{TH} \cdot \tau'(V_{TH0}) \quad \text{Eq. (4.5)}$$

Calculating the first derivative of τ as function of V_{TH} at the point of V_{TH0} from Eq. (4.3), one will get:

$$\tau'(V_{TH0}) = \frac{-m \cdot A}{(V_{GS} - V_{TH0})^{m+1}} \quad \text{Eq. (4.6)}$$

In the case $\tau 0$ represents the original propagation delay without aging, being:

$$\tau 0 = \frac{A}{(V_{GS} - V_{TH0})^{m}} \quad \text{Eq. (4.7)}$$

Then by deducting from Eq. (4.6) and Eq. (4.7), one obtains:

$$\tau'(V_{TH0}) = \frac{-m \cdot A}{(V_{GS} - V_{TH0})} \cdot \tau 0 \quad \text{Eq. (4.8)}$$

Replacing the $\tau'(V_{TH0})$ in Eq. (4.5) with Eq. (4.8), and based on Eq. (4.1) one can derive:

$$\Delta\tau = \frac{-m \cdot A \cdot \tau 0}{(V_{GS} - V_{TH0})} \cdot \Delta V_{TH} = \frac{-m \cdot A \cdot \tau 0 \cdot K}{(V_{GS} - V_{TH0})} \cdot t^{n} \quad \text{Eq. (4.9)}$$

This shows $\Delta\tau$ will change linearly with the V_{TH} degradation (ΔV_{TH}). Therefore, the propagation-delay degradation has a power dependency with respect to the aging time at the same rate n as in Eq. (4.1).

In the same way of deducting Eq. (4.9) from Eq. (4.2), one can get:

$$\Delta Id = \frac{-m \cdot K \cdot \Delta Id0}{(V_{GS} - V_{TH0})} \cdot t^{n} \quad \text{Eq. (4.10)}$$

in which $Id0$ is the original drain current without aging, showing the drain current change ΔId has a power dependency to aging time at the same rate as Eq. (4.1).

4.3.2 The Circuit Degradation Model

At the circuit level, however, the transistors in a circuit can experience different voltage and temperature stresses, and hence the overall performance degradation caused by aging is more complex than a logic-gate degradation. However, the performance

degradation of the circuit can be estimated as follows. The degradation of propagation delay ($\Delta\tau c$ in Eq. (4.11)) is the accumulation of delays changes ($\Delta\tau, i$) of all individual logic-gate i in the critical path. Similarly, the degradation of current (ΔIdc in Eq. (4.12)) can be approximated as the accumulation of all individual logic-gate drain-current changes ($\Delta Id, i$ in Eq. (4.12)). Assuming that the major stress conditions of the different gates are mostly close to each other, i.e. n in Eq. (4.1) is the same for all gates in the monitored critical path, and using Eq. (4.9), then the degradation of the critical-path delay can be approximated as:

$$\Delta\tau c \approx \sum_{i=1}^{N} \Delta\tau, i = \sum_{i=1}^{N} \frac{-m_i \cdot A_i \cdot \tau_{0,i} \cdot K_i}{(V_{GS,i} - V_{TH0,i})} \cdot t^n \qquad \text{Eq. (4.11)}$$

$$\Delta Idc \approx \sum_{i=1}^{N} \Delta Id, i = \sum_{i=1}^{N} \frac{-m_i \cdot K_i \cdot Id_{0,i}}{(V_{GS,i} - V_{TH0,i})} \cdot t^n \qquad \text{Eq. (4.12)}$$

where N is the number of logic gates in the critical path, m_i, A_i, $\tau_{0,i}$ and K_i represent the different values in Eq. (4.11) and Eq. (4.12) for different logic gates. Therefore, the degradation regarding the circuit critical delay and drain current closely follows the trend of the power law with respect to aging time, which is similar to the V_{TH} degradation in a logic gate as shown in Eq. (4.1). This shows the correctness of our health-monitoring technique for aging-related performance degradation detection, via critical-path delay monitoring and quiescent current monitoring.

4.4 The Baseline Architecture of Our Target VLIW Processors

This section provides the detailed architecture of our target VLIW processor, the Xentium® [Walt 11] from Recore Systems. As mentioned previously, the goal of this thesis is to investigate health-monitoring techniques for digital processors in SoCs. This includes the development of health monitors and monitoring methods, the assessment of these techniques in a digital processor, and based on the monitoring results, predicting the remaining life-time of our target processor. In order to validate these different techniques we have selected a Digital Signal Processor (DSP), the Xentium processor, whose architecture and evaluation SoC, the so-called Moon IC, has been shown in the second chapter. It includes a datapath, a decoder / loop buffer unit, an instruction cache,

a control unit, tightly coupled memories and interfaces such as the cache port and (Amba-based) AHB-Bridge.

The Xentium datapath has been designed based on a VLIW architecture to increase the computing parallelism. It comprises of ten execution units and five register banks. Each execution unit is responsible for a certain class of instructions. For example, the A and S units (A0, A1, S0 and S1) perform arithmetic and logic operations. The C and P units (C0 and P0) perform instructions under the control of the program counter, while the M units (M0 and M1) are for multiplier arithmetic. The E units (E0 and E1) include the load (LD) and store (ST) units; they perform the load/store (LD/ST) instructions respectively. More detailed information can be found in [Reco 11].

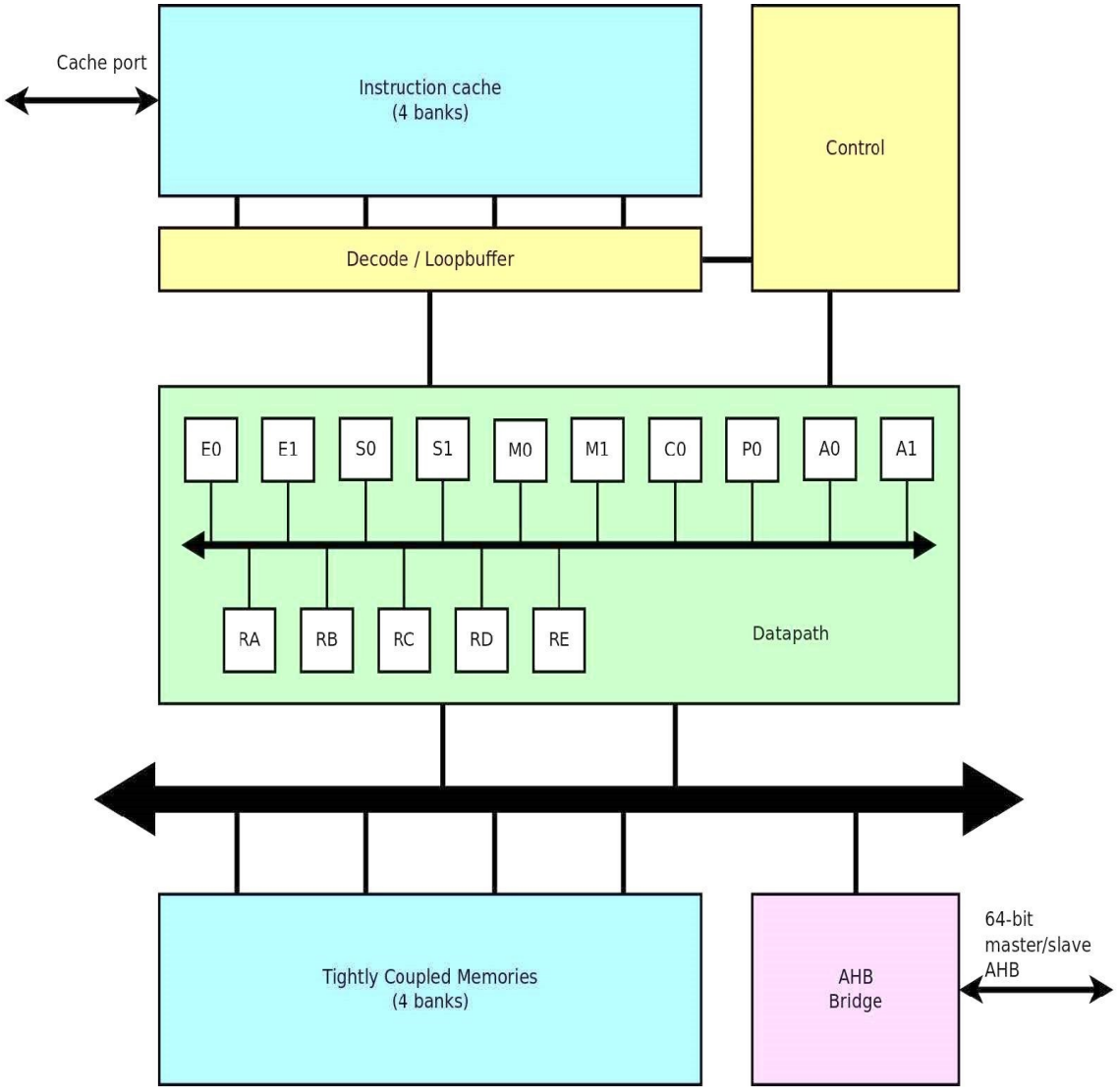


Figure 4.1: The Xentium® processor [Reco 11] architecture overview. All interconnections are bidirectional.

All execution units can access five register banks (RA, RB, RC, RD and RE) in parallel (Figure 4.1). Each unit is connected to the control and the decoder/loopbuffer part. The control part is responsible for functions like e.g. a pipelined state-machine and program-counter computation. The decoder/loopbuffer part is in charge of performing the decoded/loop instruction. The tightly-coupled memories are low-latency SRAMs communicating with the datapath in parallel.

The requirement is that all operations of the Xentium processor have to be covered by our monitoring program. The delay monitoring is for observing the critical-path delay of the datapath when executing some functional DSP programs e.g. Fast Fourier Transform (FFT), Finite Impulse Response (FIR) and Hilbert Transform (HT) etc. The I_{DDQ} monitoring is meant to observe the quiescent power-supply current in the case the processor enters a static state. The I_{DDT} monitoring is to measure the dynamic power-supply current of each execution unit inside the datapath and explore the current signature for the purpose of performance monitoring/evaluation. The developed test programs in our SBHM will be explained in detail in the next section.

4.5 HEALTH-MONITORING TEST PROGRAM DESIGN

As mentioned previously, the Xentium processors are based on a combination of common functional units (A and S units, C and P units, M unit, LD and ST units). Exploiting the characteristics of the functional units, our solution allows running a test program autonomously without any manual effort during the monitoring process. Since our method employs the SBHM technique, it does not require the usage of traditional ATPG tools, nor the adoption of any Design for Testability (DfT) technique.

A VLIW processor is characterized by the fact that all operations are executed by parallel instruction packets, each characterized by its own functional units. In our developed health-monitoring test program, parameters such as the number of instruction packets, the number and type of functional units embedded into each instruction packet, as well as the access mode of the register file must be taken into account.

Considering the particular architecture of the VLIW processor, several solutions have been proposed in order to produce an assembly code suitable for our Xentium processor. The proposed monitoring program, e.g. the delay monitoring, is based on some functional programs typically used by the Xentium for verification purposes. The I_{DDX} monitoring program is based on the test of a specific unit in the Xentium processor, requiring to perform a well-defined sequence of instructions in well-defined instruction packets.

As indicated in Chapter 2 (Figure 2.5), the Xentium processor is evaluated by us in a SoC, the so-called Moon IC. The monitoring program has been developed in combination with another embedded processor, the Leon processor (Figure 2.5). This processor provides control signals to the Xentium via a NI and the NoC. The communication infrastructure is illustrated in Figure 4.2. The NI can directly link to the Xentium data/instruction ports. It contains the configuration registers, mailboxes, timers and direct memory access (DMA). The configuration registers are used e.g. for the start/reset function. The timer can be used for monitoring the execution time. The DMA can be used for large data volume read/write operations for the Xentium. The mailbox is the communication medium for the Xentium and Leon processors during the handshake and program upload etc.

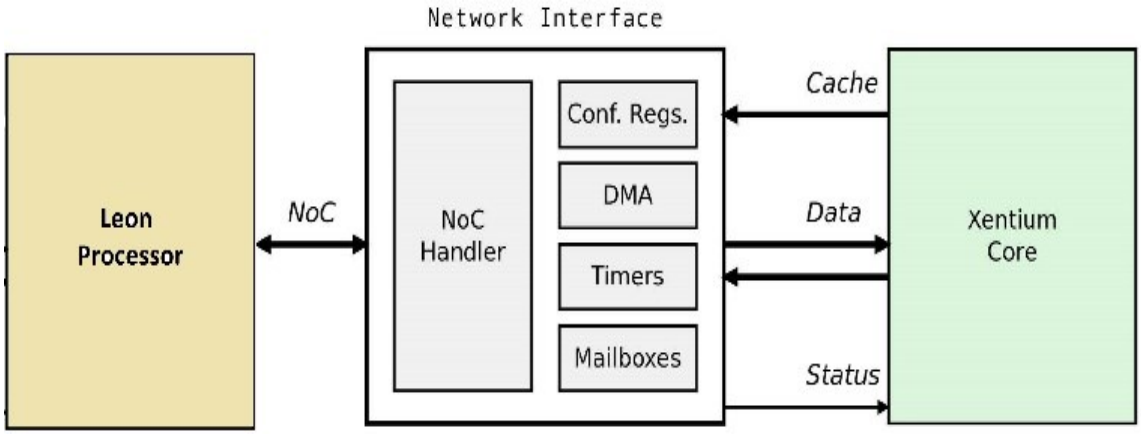


Figure 4.2: Access to the Xentium processor via the Network Interface. A Leon processor [Aero 10] is used for the control of the Xentium [Reco 11].

Our approach always has two parts in all test programs: one from the Xentium side that runs the test programs, and the other from the Leon side that runs the control program. Therefore, the whole flow executing the health monitoring has been developed in two branches. The program which runs on the Xentium was developed using assembly code while the program which runs on the Leon has been written in the C-language. In the following parts, the details of these monitoring programs are presented.

4.5.1 DELAY-MONITORING TEST-PROGRAM DESIGN

In the past years, delay-testing was mostly performed by resorting to DfT solutions, including scan and BIST. However, the following problems are associated with the path-delay testing with regard to the above solutions:

- It becomes harder to test circuits at increased operational speed nowadays in practice.
- The number of delay paths which are targeted by the test can be very large.
- They are invasive testing methods, basically disturbing (loading) the circuits.

Therefore, in today's delay testing, these solutions need to be substituted by a SBST/functional approach, which only acts on functional inputs and monitors functional outputs, without the usage of conventional DfT structures. In particular, the SBST approach may be used in those cases where at-speed testing is crucial, or DfT cannot be used.

In our research on the functional program design of delay monitoring [Zhao 15c], the basic idea of measuring the critical path delay is to measure the maximum clock frequency where the Xentium still operates correctly during the execution of all designed verification tests. However, the SoC design specifications typically need to meet worst-case conditions in practice. Therefore, considering process variations, supply noise and temperature, which are the main sources of path-delay variation, the final delay tolerance band can have a large margin of process corners, resulting in a quite conservative value. This is especially the case in the critical path(s), which determine(s) the maximum speed of the processor. For example, in the Xentium processor under typical operation conditions, the defined maximum clock frequency is 200 MHz. However, our

experiments show that even in the over-clocked speed at 240 MHz, the Xentium can still operate without any issue.

Therefore, if in our design at the lowest available clock frequency (4 MHz) the Xentium does not fail, the PLL (for health-monitoring usage) will iteratively provide a higher (max. 246 MHz) clock frequency, until it fails. If even at the highest clock frequency it is still operational, the supply voltage (V_{DD}) of the Xentium can be decreased to enhance the chance of failure. This can be completely fulfilled by our SBHM program. The evaluation of the delay-monitoring program will be described in the next chapter.

The flow of the delay-monitoring technique is based on five main steps: the Initialization, the Handshake, the Xentium Sweeping, the Functional Program Verification and Test Results Evaluation. The detailed steps are illustrated in Figure 4.3. The flow is based on the Xentium manifest, which contains all the features/parameters of the processor under test, such as the address of the peripherals, timeout information, cache size, tightly-coupled memory map, etc. These parameters are defined as global since they are configured with respect to the characteristics of the Xentium processor. The details of each of these steps are described hereafter.

The Initialization phase

Before running the delay-monitoring program, the communication between the Leon and Xentium should be initialized. On the Leon side, on-board switches are set to put the Xentium into the debugging mode, in which case the Xentium processor can be controlled by the Leon. Next, the Leon processor is powered up. An Application Program Interface (API) in the Leon subsequently powers up the Xentium processor.

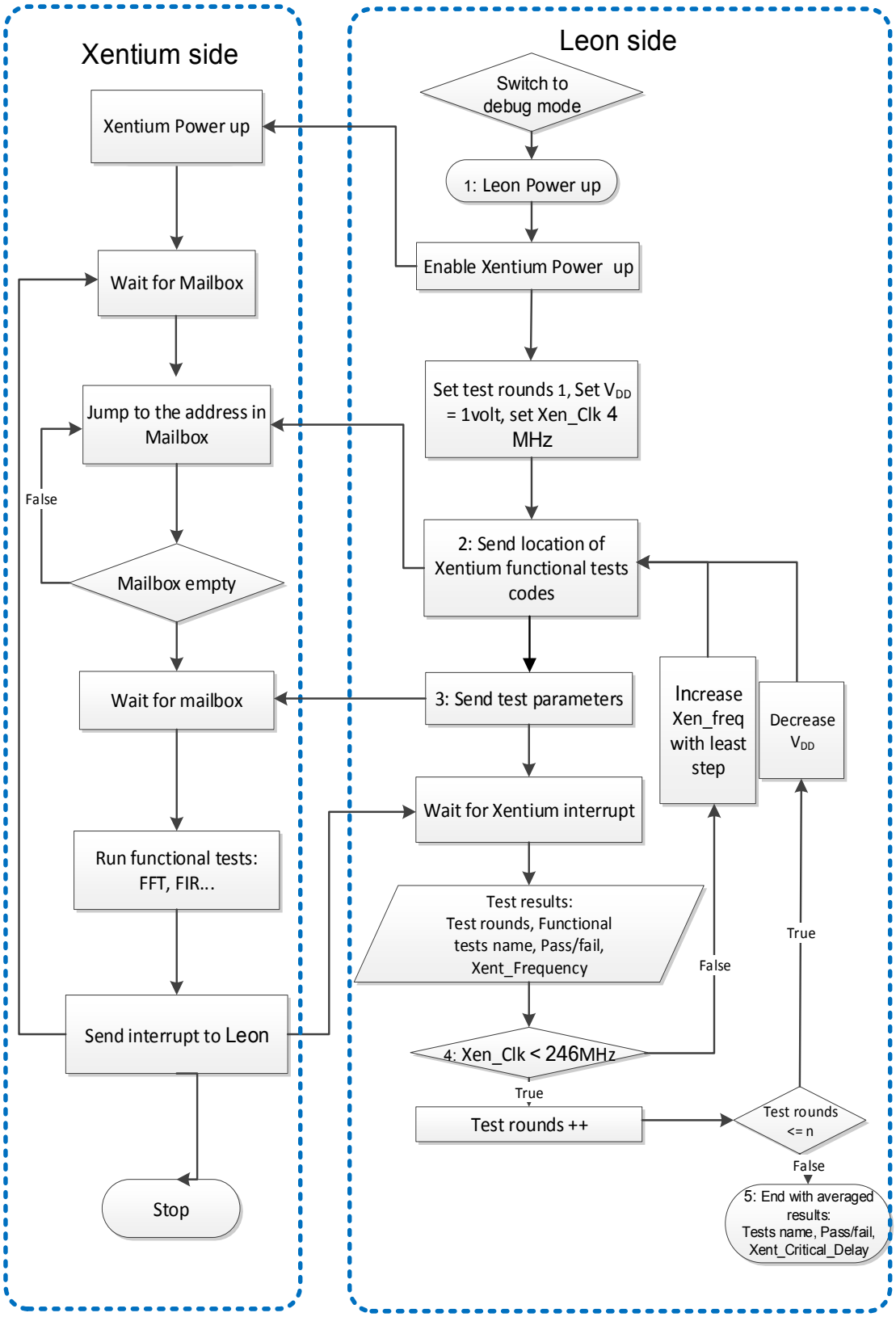


Figure 4.3: Test flow for the functional delay monitoring of the Xentium processor under control of the Leon processor.

During the initialization phase, some parameters are set: at the Leon side, e.g. the communication baud rate, setting the Xentium to the minimum clock frequency (4 MHz), wait-timeout value, and test rounds for the verification program run. At the Xentium side, the mailbox-address allocations, GPIO (connected to the peripherals) registers settings, and local memory-address allocations etc. are being set.

After the initialization, the Leon and Xentium processor are ready for handshaking to start the communication, which consists often of program uploading, setting up interrupts and downloading of test results.

The Handshake phase

The completion of the handshake sequence is a flag indicating the communication between the Xentium and Leon is ready to start. During the handshake, the mailbox (numbered with 2 in Figure 4.3 at the Leon side) is the medium for the two processors. It is a communication infrastructure in the Network Interface that can be accessed by the Leon as well as the Xentium processor. During the test, the Xentium processor will first wait and verifies if the Leon processor sends an interrupt to claim its READY status by setting the mailbox. The Xentium side will check the mailbox register. Subsequently, the Xentium will send an interrupt back to indicate its READY status as well if the Leon is ready.

The Functional Program Verification phase

After the handshaking, the Leon processor will send the addresses of the executable functional verification programs from the memory, e.g. a DDR, to the mailbox. In the meantime, the Xentium will continue to fetch the test code until the mailbox is empty (numbered with 3 in Figure 4.3).

In our delay-monitoring program, a set of verification functional test programs are employed to check if critical parts in the Xentium function well. Table 4.1 shows these functional programs and their descriptions. As one can see, these programs are typically composed of a set of test operations of an arithmetic nature.

Table 4.1: Applied functional programs for the Xentium verification.

Functional programs	Function description
test_fft_2048	Computing a 2048-points FFT. High efficiency code, using multiple units at the same time.
test_cfir	Computing a complex filter
test_dsp_hilbert	Computing a Hilbert filter as a function call. High efficiency code, using multiple units at the same time.
test_dsp_cfir	Computing a complex filter as a function call. High efficiency code, using multiple units at the same time.

During the verification phase, these operations will be fed into the Xentium and after calculation by the Xentium, the results will be compared to the expected ones to check the correct functionality of the Xentium.

The Xentium Sweeping phase

As analysed previously, the operational frequency needs to be swept in order to find at which moment the Xentium will fail at a certain speed (numbered with 4 in Figure 4.3); the supply voltage V_{DD} needs to be swept as well in order to increase the chance of failure occurrence. Besides, in order to increase the measurement accuracy, in our design, for each specified voltage and frequency of the Xentium, n rounds (in Figure 4.3) of verification testing are performed; n can be determined based on the test time, in our case, n is set to 10. This is because our SBHM can be completed within 50k clock cycles with an average of 125 MHz (sweeping from 4 MHz to 246 MHz), which leads to less than 1 millisecond test time. The test time is calculated in Eq. (4.13),

$$\text{Test time} = \sum_{i=1}^{N} \frac{1}{f_i} \cdot N_{CK,i} \qquad \text{Eq. (4.13)}$$

where f_i is the sweeping frequency and $N_{CK,i}$ is the number clock cycles of a test at this frequency.

Test Results Evaluation phase

During the sweeping phase, the Xentium is expected to be found faulty at one certain frequency and supply voltage. However, at a certain voltage, the Xentium can pass the functional verification program at one Xentium clock frequency, but can fail it next time. This is caused by that the fact the Xentium clock is set by the *PLL* of the Leon processor at some discrete frequencies instead of continuous values, hence the measured maximum clock frequency of the Xentium is discrete. The solution is that by taking 10 times a measurement at each frequency point, the average of these 10 measurement results will be taken as the final value (numbered with 5 in Figure 4.3).

4.5.2 I_{DDQ}-MONITORING TEST-PROGRAM DESIGN

I_{DDQ} testing refers to the circuit testing method based upon the measurement of steady-state/quiescent power-supply current. The principle is that in steady state, when all switching transients are settled-down, a CMOS circuit draws little current, and hence exhibits almost no power dissipation. The leakage current in a defect-free CMOS circuit is negligible, in the order of few nano-amperes [Mata 15]. However, in case of a defect such as a gate-oxide short or a bridging fault, a conduction path from power-supply to ground is formed, resulting in an exponential increment of the circuit current. Therefore, I_{DDQ} testing has been typically developed to detect process-oriented physical defects, such as open and bridging faults. Test-vector generation for target DUTs (Device Under Test) and a scan test are required [Rajs 00]. In the past, a potential relationship has been made between I_{DDQ} tests and NBTI-related effects [Wang 12].

New possibilities for I_{DDQ} testing are now being investigated because in the case of safety-critical systems, the reliability level has to be increased for final tests. In addition, also during the life-time usage, a constant awareness in terms of dependability is required by the market. In our design [Zhao 15a], the target is to monitor the performance-related aging degradation in the Xentium processor, instead of using fault-based testing [Zhan 15a]; the value of I_{DDQ} with respect to the aging time will be evaluated later on to prove the effectiveness of our technique.

Our I_{DDQ} functional monitoring program design will be outlined in this section. The key idea behind the method for the I_{DDQ} monitoring lies in letting the Xentium enter a stable and repeatable state, or a quiescent state. For this purpose, a 'WAIT' instruction has been especially employed; it can be set in the configuration registers of the NI (Figure 4.2).

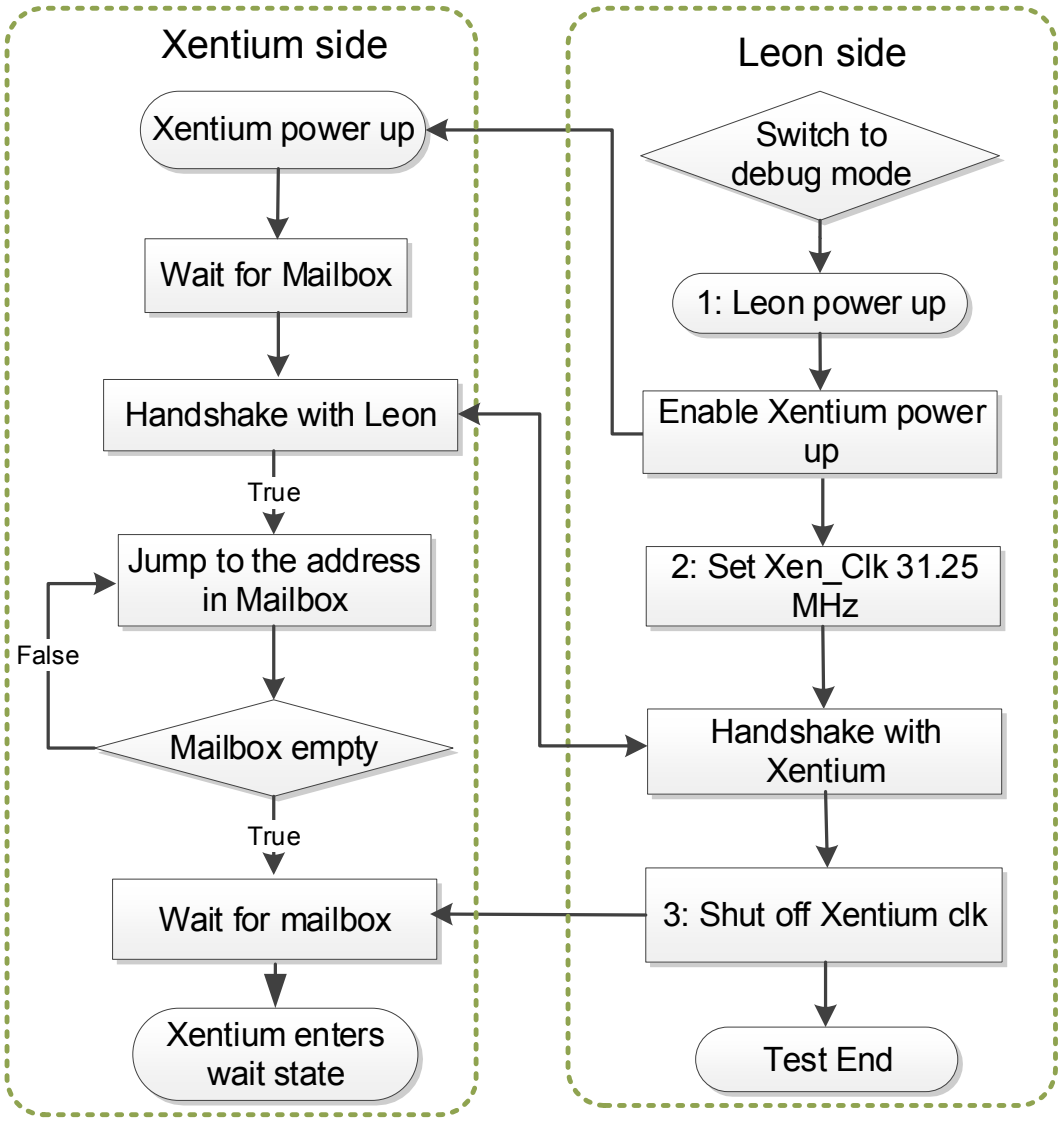


Figure 4.4: Test flow for the I_{DDQ} monitoring of the Xentium processor under control of the Leon processor.

The flow of the I_{DDQ} monitoring is based on four main steps: the Initialization, the Handshaking, the Xentium Clock Control and the Xentium Wait, which is shown in

Figure 4.4. Basically, the first two steps are similar as in the delay-monitoring program design process. After the Xenium is in the Wait state, the I_{DDQ} value of the Xentium can be measured, of which the hardware will be described in the next chapter. For the last two steps, the detailed description is provided below.

The Xentium Clock Control

During the test, the clock for the Xentium is set two times by the Leon (numbered with 2 and 3 in Figure 4.4). The first time, the Xentium clock frequency is set to 31.25 MHz. Unlike the delay, which is based on clock-frequency sweeping, the maximum frequency or the critical delay can be calculated using our evaluation board (will be described in Chapter 5). However, both the I_{DDQ} monitoring as well as the I_{DDT} monitoring will not provide a direct output of the current values. Instead, the program will control the Xentium to enter in a pre-defined state, after which the supply currents of the Xentium will be measured by other instruments (the QT 1411 [Ridg 12] in our application, to be illustrated in the next Chapter).

In order to better measure the transition of the current, it is suggested that the operational frequency of the Xentium is equal to or multiple-times the sampling frequency of the current measurement instrument, which is 31.25 MHz in our case [Ridg 12]; therefore, the Xentium is set to the same frequency.

The second time is at the moment the Leon turns off the Xentium clock; this is because even after the Xentium enters the wait state, it can still perform other operations (see the next section). Therefore, from the Leon side, the Xentium clock needs to be shut off in order to stop synchronization in the processor, which is required for the I_{DDQ} measurement.

The Xentium Wait

After execution of the command “WAIT” in the Xentium, the Xentium might wake up from the idle state to let it enter an intermediate state, or preparation state. This can be found in various system specific use-cases, such as Mailboxes, Timers and Interrupts. The Xentium Wait state is a combinational operation of executing this “WAIT” command and shutting off the Xentium clock by the Leon. (Numbered with 3 in Figure 4.4)

4.5.3 UNIT-BASED I_{DDT} MONITORING TEST-PROGRAM DESIGN

I_{DDT} testing measures transient currents instead of static leakage current. The dynamic supply current waveform of the IC shows a spike whenever the circuit makes a transition from one logic state to another logic state [Makk 95]. If there is a process defect such as open or short, the current transition shows significant difference, which can be used to differentiate fault-free from faulty circuits.

I_{DDT} testing is potentially used for linking resulting behaviour to possible defects in the circuit. Strategies using the I_{DDT} test can vary. Extracting current signatures which can be used for circuit integrity analysis, e.g. current-shape analysis [Makk 95], look at currents in the time-domain and frequency-domain [Germ 99]. The average energy consumption ratios measurement [Bhun 02] is another strategy which can be used.

These I_{DDT} testing techniques have demonstrated their effectiveness in specifically designed small-sized circuits, while applications in real environments using field-testing are limited. Meanwhile, these methods normally aim at detecting potential manufacturing defects. Furthermore, for traditional I_{DDT} testing, a time-consuming scan-based current test is required, which is not suitable for large processors. In our I_{DDT} monitoring technique, the performances of the processors have to be investigated during life-time usage. Therefore, our monitoring results will provide real-time data with regard to the aging status of the target processor, which can be used for prognostic purposes later on.

The flow of the I_{DDT} monitoring in our design [Zhao 15b] is based on four main steps: the Initialization, the Handshake, the Xentium Clock Control and the Unit-based I_{DDT} monitoring; detailed steps are shown in Figure 4.5. Basically, the communication of Initialization between the Leon and the Xentium are using the same settings as in the I_{DDQ} monitoring approach. The same approach holds for the Handshake phase. The Xentium clock is set to 31.25 MHz as well for the same reason as previously stated. Therefore, the main effort is to design the functional program running at the Xentium side.

As mentioned before, on the basis that the whole functionality of the Xentium processor is covered by our test program, enhancing the dependability and hence reliability of it lies in monitoring its crucial part, the datapath. The delay monitoring can

detect the functionality of the datapath of the Xentium well while preforming the previously described DSP-based verification tests. Our goal of the designed I_{DDT} functional program is similar: to monitor the dynamic power current of each execution unit inside the datapath.

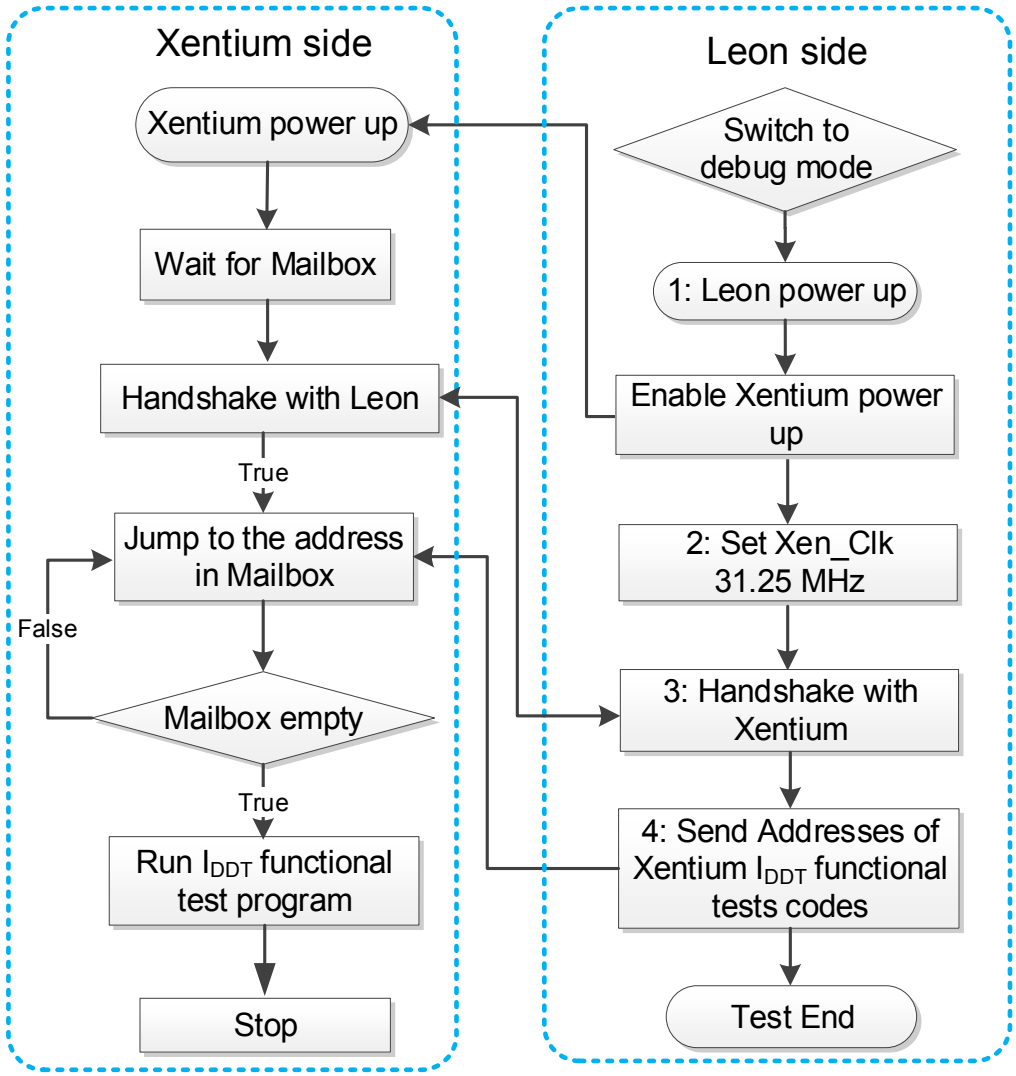


Figure 4.5: Test flow for the functional I_{DDT} monitoring of the Xentium processor under control of the Leon processor.

In the datapath (Figure 4.1) of the Xentium, the 10 functional units (A and S units, C and P units, M unit, LD and ST unit) and 5 register files are organized into 13 instruction slots. Each functional unit has its own instruction slot, with the remaining 3 instruction slots used for the register files. One of these instruction slots is used for the register

reading. Another instruction slot is used for the register writing to register files RA and RB and the register writing to register file RC. The last instruction slot is reserved for the register writing to register file RC and this register writes to register files RD and RE [Reco 11].

The instructions contained in an instruction packet should by rule be issued in parallel (i.e. at the same time) to the instruction slots. An instruction packet consists of a maximum of 12 instruction slots. The header encodes for which instruction slots the instruction packet contains instructions.

Generally, the allocation of instructions to instruction packets determines the schedule of the Xentium program. The duration of an execution of the instruction packet is also referred to as a cycle. In each cycle, one can assign different instructions into one instruction packet. Therefore, different combinations of functional units will be active in that cycle. In the case only one specified functional unit is filled into the instruction packet (i.e. one instruction slot is occupied), one can actually monitor/test the behaviour of the specified functional unit.

Based on the architecture of the Xentium processor, the designed I_{DDT} functional program has to show the current signature for all functional units, i.e. the A, M, S, P, ST and LD units. As explained above, one can activate one functional unit in one cycle/instruction packet; this is analogous to that different functional units run in series instead of parallel, in order to be able to monitor their separate power currents during the test. This approach we refer to as the unit-based I_{DDT} monitoring program [Zhao 15b].

During the execution of the program, different phases in the case of different active units are defined in each cycle. Each unit will run for a constant (e.g. 100) clock cycles (considering the memory size of the current measurement instrument QT1411 [Ridg 12] and the sampling speed). Thereafter a separation by a NOP (No OPeration instruction) is inserted, and the monitoring program will enter the next phase and so on.

Similar as in the I_{DDQ} program, after the Xenium completes the execution of the monitoring program after each phase, the Xentium can be measured for its I_{DDT} value, where similar hardware as the I_{DDQ} monitor will be used, as described in the next chapter.

4.6 CONCLUSIONS

In this chapter, a technique has been proposed for aging-related performance degradation detection in the 90 nm Xentium VLIW processor, via SBHM. It includes a delay-monitoring program, I_{DDQ} monitoring program and unit-based functional I_{DDT} monitoring program. These programs set the states of the Xentiums via a Leon processor to enable the health-monitoring with extra hardware described in the next Chapter.

The number of instructions of the total delay-monitoring program is less than 50k; for the I_{DDQ}-monitoring program it is less than 10k and for the I_{DDT}-monitoring program it is less than 20k. This is small enough for nowadays processors. The total test time is in the millisecond range. The power consumption for all SBHM programs is neglectable, based on previous power-consumption calculations in Chapter 3.

Our SBHM technique is generic in enhancing the dependability of SoCs consisting of many target processors for in-field health monitoring and evaluation, advancing with features of direct degradation measurements as compared to conventional embedded health monitors. Therefore, it can provide the most accurate monitoring accuracy as compared to other monitoring techniques. The design of our SBHM in this chapter lays the foundation for a real implementation of a health-monitoring system.

For the application of this technique to other SoCs, it is important to extract functional features using other DSP-based functional tests (for the delay monitor) and particular (quiescent/transient) states that can be delivered by the processor (for the $I_{DDQ/T}$ monitor).

REFERENCES

[Aero 10] http://ams.aeroflex.com/pagesproduct/prods-hirel-leon.cfm, 2010.

[Batc 99] K. Batcher and C. Papachristou, "Instruction randomization self test for processor cores," in 17th IEEE VLSI Test Symposium (VTS), pp. 34-40, 1999.

[Braa 16] T.D. Braak, "Run-time mapping: dynamic resource allocation in embedded systems," PhD thesis, ISBN: 978-90-365-4213-5, University of Twente, Enschede, 2016.

[Bern 11] P. Bernardi, M. Grosso, E. Sanchez, and O. Ballan, "Fault grading of software-based self-test procedures for dependable automotive applications," in Design, Automation & Test in Europe (DATE), pp. 1-2, 2011.

[Bhun 02] S. Bhunia, K. Roy, and J. Segura, "A novel wavelet transform based transient current analysis for fault detection and localization," in Proceedings of Design Automation Conference (DAC), pp. 361-366, 2002.

[Cha 14] S. Cha, C.C. Chen, T. Liu, et al., "Extraction of threshold voltage degradation modeling due to negative bias temperature instability in circuits with I/O measurements," in IEEE 32nd VLSI Test Symposium (VTS), pp. 1-6, 2014.

[De 99] V. De and S. Borkar, "Technology and design challenges for low power and high performance," in Proceedings of the International Symposium on Low Power Electronics and Design, San Diego, California, USA, pp. 163-168, 1999.

[Germ 99] A. Germida, Y. Zheng, J. F. Plusquellic, and F. Muradali, "Defect detection using power supply transient signal analysis," in Proceedings of the International Test Conference (ITC), pp. 67-76, 1999.

[Guru 07] S. Gurumurthy, R. Vemu, J. A. Abraham, and D. G. Saab, "Automatic Generation of Instructions to Robustly Test Delay Defects in

Processors," in IEEE European Test Symposium (ETS), pp. 173-178, 2007.

[Jepp 77] K.O. Jeppson, et al., "Negative bias stress of MOS devices at high electric fields and degradation of MOS devices," in Journal of Applied Physics, Vol:48, Issue:5, pp. 2004-2014, 1977.

[Karl 08] E. Karl, P. Singh, D. Blaauw, and D. Sylvester, "Compact in-situ sensors for monitoring Negative-Bias-Temperature-Instability effect and oxide degradation," in IEEE International Solid-State Circuits Conference (ISSCC), pp. 410-623, 2008.

[Kean 10] J. Keane, T.-H. Kim, and C. H. Kim, "An on-chip NBTI sensor for measuring PMOS threshold voltage degradation," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 18, pp. 947-956, 2010.

[Rief 16] A. Riefert, R. Cantoro, M. Sauer, et al., "A flexible framework for the automatic generation of sbst programs," in IEEE Transactions on Very Large Scale Integration Systems (VLSI), vol 24, pp. 3055-3066, 2016.

[Kufl 06] H. Kufluoglu and M. A. Alam, "Theory of interface-trap-induced NBTI degradation for reduced cross section MOSFETs," in IEEE Transactions on Electron Devices, vol. 53, pp. 1120-1130, 2006.

[Lago 07] J. Lagos-Benites, D. Appello, P. Bernardi, et al., "An effective approach for the diagnosis of transition-delay faults in SoCs based on SBST and scan chains," in 22nd IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT), pp. 291-302, 2007.

[Makk 95] R. Z. Makki, S. Shyang-Tai, and T. Nagle, "Transient power supply current testing of digital CMOS circuits," in Proceedings of the International Test Conference (ITC), pp. 892-901, 1995.

[Mata 15] S. Matakias, Y. Tsiatouhas et al. "A current monitoring technique for I_{DDQ} testing in digital integrated circuits," in Journal of VLSI Integration, vol. 50, pp. 48-60, 2015.

[Niga 09] T. Nigam, "Impact of transistor level degradation on product reliability," in IEEE Custom Integrated Circuits Conference (CICC), pp. 431-438, 2009.

[Paul 05] B. C. Paul, K. Kunhyuk, H. Kufluoglu, M. A. Alam, and K. Roy, “Impact of NBTI on the temporal performance degradation of digital circuits,” in IEEE Electron Device Letters, vol. 26, pp. 560-562, 2005.

[Port 12] M. Portela-Garcia, M. Grosso, et al., “On the use of embedded debug features for permanent and transient fault resilience in microprocessors,” in Microprocessors and Microsystems, vol. 36, pp. 334-343, 2012.

[Psar 10] M. Psarakis, D. Gizopoulos, E. Sanchez, and M. S. Reorda, “Microprocessor Software-Based Self-Testing,” in IEEE Design & Test of Computers, vol. 27, pp. 4-19, 2010.

[Rajs 00] R. Rajsuman, “Iddq testing for CMOS VLSI,” in Proceedings of the IEEE, vol. 88, pp. 544-568, 2000.

[Reco 11] Recore Systems. http://www.recoresystems.com, 2011.

[Ridg 12] Ridgetop group. http://www.ridgetopgroup.com/doc/QT-1411-HL-revB.pdf, 2012.

[Saku 90] T. Sakurai and A. R. Newton, “Alpha-Power law MOSFET Model and its Applications to CMOS Delay and other Formulas,” in IEEE J. Solid-State Circuits, pp: 584-594, Feb 1990.

[Walt 11] K. H. G. Walters, S. H. Gerez, G. J. M. Smit, et al., “Multicore soc for on-board payload signal processing,” in NASA/ESA Conference on Adaptive Hardware and Systems (AHS), pp. 17-21, 2011.

[Wang 07] W. Wang, S. Yang, S. Bhardwaj, et al., “The impact of NBTI on the performance of combinational and sequential circuits,” in Proceedings of the Design Automation Conference (DAC), San Diego, California, pp. 364-369, 2007.

[Wang 12] Y. Wang, S.D. Cotofana and L. Fang, “Statistical reliability analysis of NBTI impact on FinFET SRAMs and mitigation technique using independent-gate devices,” in IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH), pp. 109-115, 2012.

[Woot 12] S. N. Wooters, A. C. Cabe, Q. Zhenyu, et al., “Tracking On-Chip Age Using Distributed, Embedded Sensors,” in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, pp. 1974-1985, 2012.

[Zhao 15a] Y. Zhao and H. G. Kerkhoff, "Application of functional I_{DDQ} testing in a VLIW processor towards detection of aging degradation," in International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), pp. 1-5, 2015.

[Zhao 15b] Y. Zhao and H. G. Kerkhoff, "Unit-Based Functional I_{DDT} Testing for Aging Degradation Monitoring in a VLIW Processor," in Proceedings of the Euromicro Conference on Digital System Design (DSD), pp. 353-358, 2015.

[Zhao 15c] Y. Zhao and H. G. Kerkhoff, "Predicting aging caused delay degradation with alternative I_{DDT} testing in a VLIW processor," in the Proceedings of the Workshop on Manufacturable and Dependable Multicore Architectures at Nanoscale, MEDIAN, Tallinn, Estonia, pp. 27-32, 2015.

Chapter 5

ACCELERATED TESTING IMPLEMENTATION FOR THE MP-SOC AND RESULTS ANALYSIS

ABSTRACT *- In the previous chapter, the software-based health monitoring technique with respect to a 90 nm many-processor SoC was explained, and in addition the associated monitoring programs were designed. This chapter presents the monitoring results based on the designed software programs and this data will be used for reliability evaluation in e.g. prognostic software for lifetime prediction. The latter will be explained in Chapter 6. The implementation of our accelerated testing (AT) approach has been investigated in this chapter. The measurement data of the critical-path delay,* I_{DDQ} *and* I_{DDT} *resulting from AT as well as analysis of the results are being presented. It is found that the degradation of the critical-path delays for the processor are characterized by the power of the aging trend, while the same holds for the quiescent current and transient current monitoring results. This is in coherence with the behaviour of the NBTI aging mechanism, but contradicting other aging mechanisms such as TDDB. It confirms that NBTI aging is the dominant factor for the 90 nm processor technology.*

Parts of this chapter have been published as paper titled "Application of Functional I_{DDQ} Testing in a VLIW Processor towards Detection of Aging Degradation" in the International Conference on Design & Technology of Integrated Systems, 2015, and "Highly Dependable Multi-Processor SoCs Employing Lifetime Prediction Based on Health Monitors" in the Asian Test Symposium, 2016.

5.1 INTRODUCTION

The technique for enhancing the dependability of our Many-Processor SoCs via software-based health monitoring (SBHM) has been previously proposed [Zhao 15a], [Zhao 15b]. Parameters concerned with the reliability monitoring will be discussed, which show that critical-path delay, quiescent and transient currents are potential candidates. These different health-monitoring techniques will provide (e.g. periodic) data for predicting the remaining lifetime of processor cores in MP-SoCs during their lifetime.

In this chapter, the designed measurement set-up for the Xentium-based MP-SoC with all the functional programs is described; it is capable to find correlations by means of health monitoring and many measurements in the real world for the Xentium processor. Our designed schemes as well as the implemented PCBs will be presented.

In order to shorten the lifetime tests, accelerated testing (AT) is conducted for predicting the remaining life-time under normal operating conditions. Based on standard principles of experimental design, these tests are usually conducted by applying several accelerated stress levels with certain profiles, e.g. constant-stress, step-stress, and cyclic-stress [Liao 06].

During the AT and software-based health-monitoring (SBHM), the following tasks must be executed:

1. Carrying out the accelerated-testing procedure for a number of the Xentium-based MP-SoCs.
2. Performing the critical-path delay monitoring of the Xentium using the Recore validation board. Start with fresh chips, then measure aged chips subsequently. A one-week interval has been chosen for aging according to the JEDEC standard [Jede 10].
3. Measuring the $I_{DDQ/T}$ currents of the Xentium using the hot board implemented by Maser Engineering (Figure 5.2a) at regular time intervals after the aging tests. Start with fresh chips, then use aged chips subsequently.
4. Derive the potential correlation between different health-monitoring techniques of the Xentium as function of aging using all previous results. This requires extensive calculations.

These targets and the measurement system will be shown and discussed in sections 5.2 and 5.3. Afterwards, the measurement results of the critical-path delay, I_{DDQ} and I_{DDT} resulting from AT as well as analysis of the results are presented in sections 5.4 and 5.5 respectively.

5.2 ACCELERATED AGING TESTING

In order to provide measurement data with respect to the aging of integrated circuits (silicon, excluding packaging), a proper aging test plan had to be devised [Zhao 14]. Ideally, the conditions should be close to what can be expected in the mission profile of the application, which is a radar system in our case [Star 10].

However, in order to reduce the normal-life aging time, stress can be applied in terms of the operational temperature, processor core power-supply as well as the processor workload and processor clock frequency. Standards are followed according to the JEDEC standardization organization [Jede 10], [Jede 11].

Our primary goal is to investigate the aging of Xentium cores (and hence remaining lifetime) by using health monitors. Furthermore, potential correlations between the different monitoring techniques i.e. the critical-path delay, I_{DDQ} and I_{DDT} of the implemented Xentium processor are to be investigated. Our prediction of the remaining life-time will be based on the measured health-monitoring data and will be dealt with in the next chapter.

Two types of standard reliability tests have been carried out: first the High Temperature Operating (Bias) Life (HTOL) test [Esco 06], sometimes referred to as the burn-in test. This is a well-known method to weed out (silicon) infant-mortality failures. The parameters we used in HTOL are listed in Table 5.1 [Kerk 14].

Table 5.1. The HTOL conditions used for our life-time prediction.

Number of devices	48
Stress temperature	125 °C
Stress power supply	1.2 V
Stress clock frequency	240 MHz
Duration	1000 hrs. (6 weeks)

The second test to be carried out is the Temperature Cycling (TC) test [Jede 11], of which the parameters are provided in Table 5.2. This test is usually referred to as shock-temperature cycling, and it affects both the silicon as well as the package. This rather aggressive test could eventually lead to failure of the core or MP-SoC.

Table 5.2. The TC conditions used for our lifetime prediction.

Number of devices	48
Stress temperatures	0 °C -125 °C
Stress power supply	1.2 V
Stress clock frequency	240 MHz
Duration	1000 cycles (one hour per cycle)
Dwell time	30 minutes
Ramp up/down time	15 minutes

During the HTOL and TC test, the following accelerating factors can be controlled:

- The typical operating voltage for the Xentium can be increased within 20%
- The ambient temperature

- The typical Xentium operating frequency can be increased with 20%.
- The stress workload program

In order to compare the degradation rate of the functional test results with respect to the process variation, in total 48 devices have been stressed at a temperature of 125 °C, using a power supply of 1.2 V (1 V typical) and an over-clocking frequency of 240 MHz (200 MHz typical).

The accelerated testing experiments have been divided into two batches; each batch of 24 chips has been stressed and subsequently measured. This has to do with the oven size which is not capable to house all 48 chips at once.

5.3 Accelerated Testing System Design and Measurement Implementation

The basic setup [Zhao 15a] for our accelerated testing system is shown in Figure 5.1, where the hot and cold sections are located at the right and left side respectively, separated by the backplane edge connector. The hot section (HTOL test) PCB will be put in the oven for the AT, while the cold section PCB resides outside the oven. On the cold section PCB, there are two crystal oscillators for the Xentium based MP-SoC, as well as a microcontroller Arduino Mega used for getting access as well as generating the stress workload for the Xentium via the wires UART, DCOM and GPIO. The cold section PCB is connected via an USB port to a PC on which dedicated software runs.

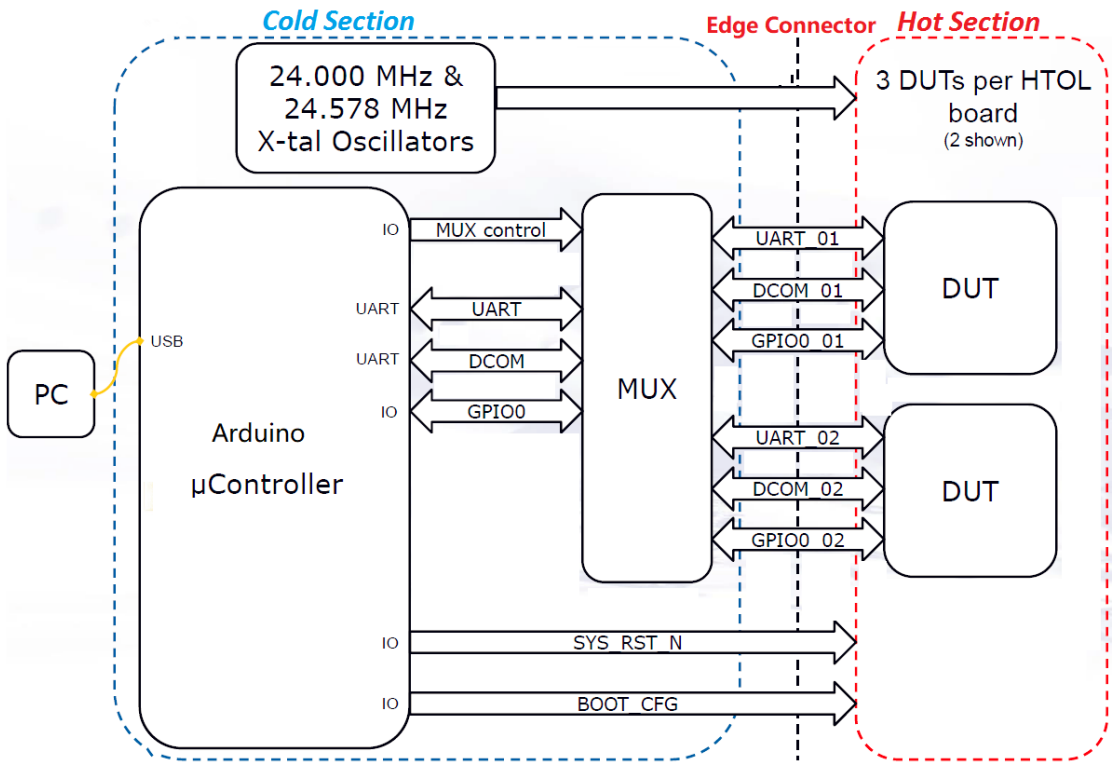
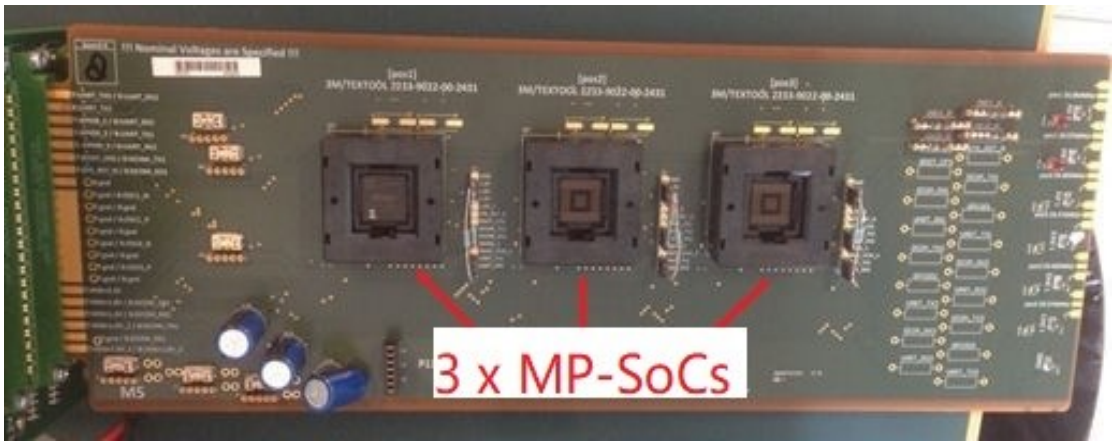
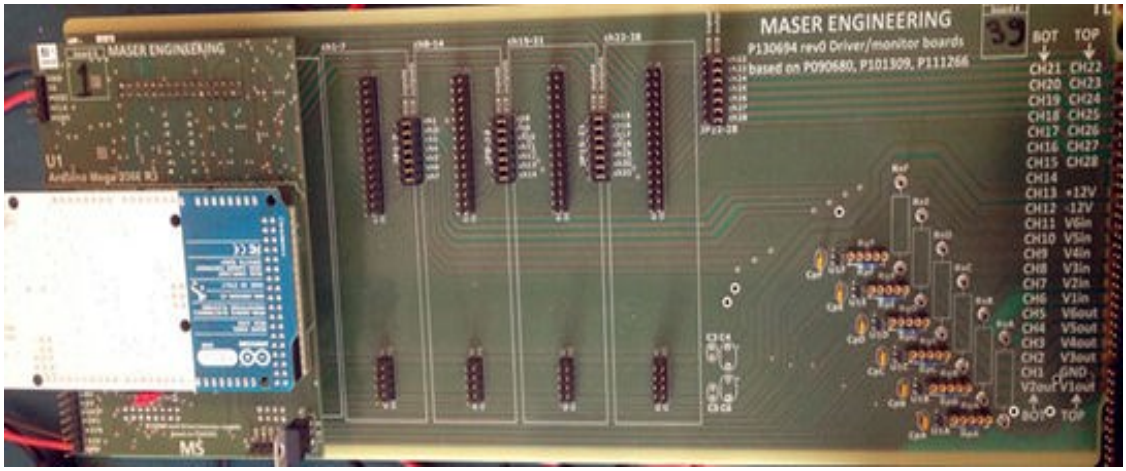


Figure 5.1: Basic setup of accelerated testing for the Xentium (DUT) scheme, including the cold section board (left), edge connector (middle) and hot HTOL test board (right) with three DUTs.

The outlook of the cold section board, edge connector and HTOL test board in practice is shown in Figures 5.2a), b) and c) respectively. The designs of the schematic of the HTOL boards were from us, while the actual layout and implementation was carried out by Maser Engineering.



a)



b)

c)

Figure 5.2. a) The implementation of the hot-section board of the MP-SoC (one of eight boards). b) The implementation of the cold-section control board including the Arduino Mega 2560 at the left. c) The implementation of the edge connector board (front and back view). (Courtesy of Maser Engineering).

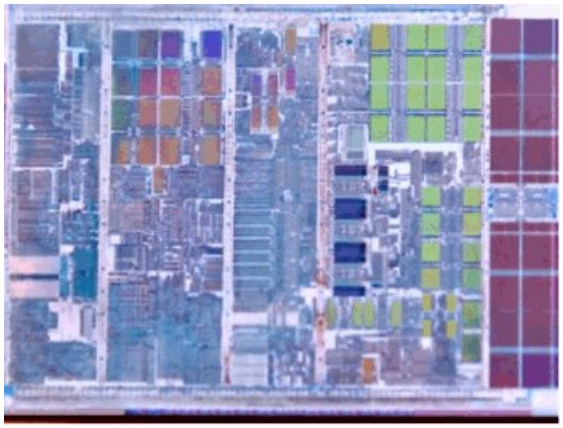
The hot-section board including the MP-SoCs is surrounded by standard pull-down resistors and power-supply decoupling capacitors. Main signals for the tests are DCOM TX/RX, UART TX/RX and the oscillator signals, which are used for control and observation of the accelerating test. These signals are connected via the connectors in the backplane boards. By this connection, the SoCs can communicate with the control system.

Based on the design information of the SoC, software in the Arduino processor was developed to stress the Xentium in terms of the workload. The cold-section board has the same pin connections as the hot-section board, which are used to send the test codes to the three SoCs in the hot-section board. Most of the lines are direct connections from the edge connector to two 22-pins connectors.

The edge connector is the interface between the hot and cold-section boards. They consist of straight interconnections. They are mounted at the oven interface (outside – inside).

Figure 5.3 shows the HTOL test oven, the MP-SoC, and the Temperature Cycling (TC) oven.

a)



b)

c)

Figure 5.3: a) The HTOL test chamber with the hot boards inside. b) Photograph of the MP-SoC used during tests. (Courtesy of Recore Systems). c) Test setup for the TC tests, incorporating three MP-SoC chips and an external work-program control (Arduino) and Maser Engineering board.

The HTOL oven from Maser Engineering was used first, while the TC oven is our Vötsch heating/cooling system, cycling between 0 ^{o}C and 125 ^{o}C. The ramping time was 30 minutes and the dwelling time 15 minutes.

At a one-week interval, the hot-section boards were removed from the oven. The critical-path delay has been subsequently measured employing the Xentium evaluation board, as shown in Figure 5.4a. The measurement flow has been explained in the previous chapter in Figure 4.3. Besides manually decreasing the power-supply voltage (Figure 4.3 step 4) during frequency stepping for the Xentium, the critical-path delay can be measured automatically, and the measurement results were obtained after running the developed software program (Figure 4.3).

The measurements of I_{DDQ} and I_{DDT} currents are known from literature [Saba 02] to predict potential reliability failures. Our current measurements were carried out in three different ways.

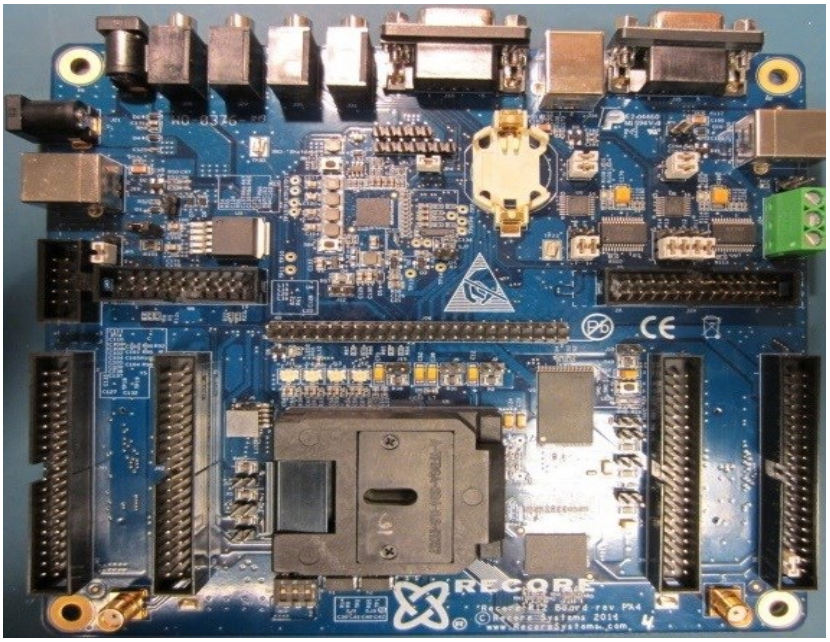
The first one uses a current-monitor device developed by the company Maser Engineering; however, the resolution, as well as speed (1 measurement / minute) basically only provides a very rough average current, and has therefore little value for our reliability evaluations.

A more accurate approach makes use of the Agilent N6781 (low) current monitor power supply. The sample speed is 200 kS/s in this case, it has a 100 mA range and a resolution of 35 μA. As the clock speed of the application is 32 MHz (based on the previous SBHM program settings), the result is still too much averaged over time. For one sample this equals to 160 instructions of the application on average. The next section shows a measurement result of the Agilent current-monitor and it is good to compare this instrument with the results obtained via the Ridgetop QT1411 monitor approach to be discussed later on.

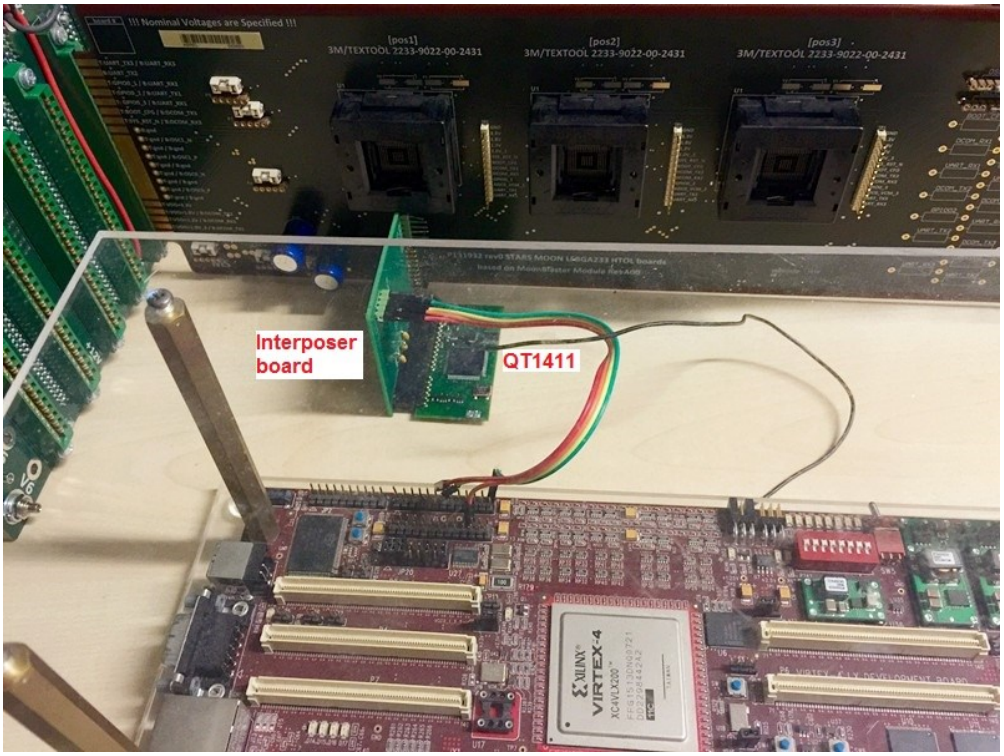
In the end we have chosen for using the Ridgetop QT1411 current monitor, which is shown in Figure 5.4b. In the photograph of the printed-circuit boards of the QT1411 and the interposer board, the control unit (Virtex-4 FPGA) for the QT1411 and the HTOL test board can be recognized.

For the I_{DDT} measurement, the sampling speed should be sufficiently fast. The QT1411 has a maximum sample rate of 50 MS/s, a 100 mA range, and a resolution of 12 μA. In Figure 5.4b, the QT1411 monitor can be seen in the middle at the bottom, our designed interposer board is in the middle on top of the monitor which is subsequently connected to the HTOL board. As specific pulses and test flows are required for operating the QT1411, these have been programmed via a Virtex-4 FPGA board shown at the bottom of the photograph. This control program can:

- Set operational commands dealing with I_{DDT} measurements, the QT1411 sampling speed, its operational mode selection, start and end trigger signals.
- Apply the correct access sequence and data to store the test results in the flash memory of the QT1411 board.
- Bypass test control signals in the QT1411 to a logic analyzer for the test-result evaluation in real time.



a)



b)

Figure 5.4: a) Delay measurement system: the evaluation board with the MP-SoC (bottom). b) $I_{DDQ/T}$ measurement system: the Virtex-4 control unit (bottom), the Ridgetop QT 1411 current sensor (middle, with a connection interposer board shown vertically) and the developed HTOL board (top). (Courtesy of Recore Systems, Ridgetop Europe and Maser Engineering).

5.4 Measurement Results of the Accelerating Life Testing

This section will present the measurement results based on our developed functional monitoring programs and accelerated testing.

The 1000 hours stress tests have been split into 6 times of 167 hours (1 week = 168 hour), with interim read-point measurements. After each week, the oven is turned off and the other stressors removed. Subsequently the boards and devices are taken out of the chamber for interim measurements, which include the critical-path delay, I_{DDQ} and I_{DDT} measurements.

5.4.1 Critical-Path Delay / Maximum Operating Frequency Measurement Results

Figure 5.5 shows the results of delay measurements of 8 fresh (out of 48) embedded Xentium chips using the Xentium evaluation board, and employing the previous delay-monitoring test program (section 4.5.1). It confirms the general theory that increasing the core supply voltage (V_{DD}) lowers the critical-path delay. Chips c30 and c36 overlap as well as chips c29 and c33.

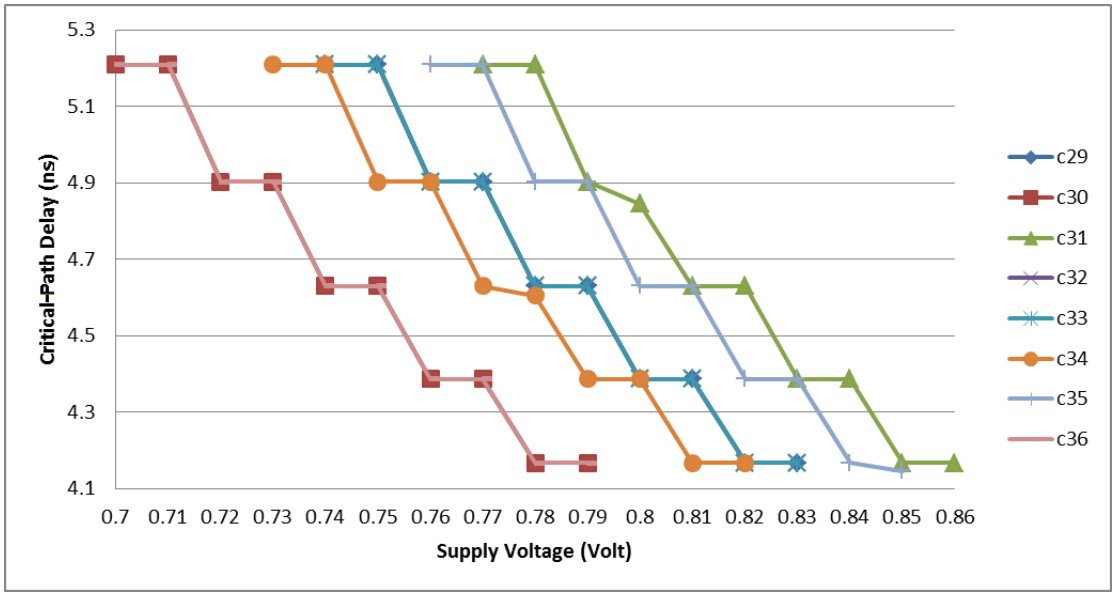


Figure 5.5: Critical-path delay test results for 8 (out of 48) fresh (non-aged) Xentiums operating at 30 °C, using the Xentium evaluation board; the X-axis shows the power-supply V_{DD} of the Xentium core.

While observing Figure 5.5, one can notice that e.g. the chip c29 delay (~ 4.4 ns, V_{DD} = 0.8 V) has 0.4 ns difference, with the chip c31 delay. This is caused by the process variation for our Xentium-based MP-SoC. At the right side of each curve the chip will fail, while at the left side it will be fault-free, because the curve represents the maximum critical-path delay the Xentium can tolerate.

After the next (stress) phase, the measurements will be repeated and a shift in curves is expected because of aging.

All Xentium-related aging tests have been carried out for six aging times, which is one whole HTOL stress round (1000 hours). During each stress test, all Xentium processors are stressed at 125 °C with 1.2 V supply voltage. The stress duration for each time slot is 167 hours. After each week of stress, the oven is stopped, the chips are powered off and taken out from the oven. Then the interim reliability tests are carried out at 30 °C. Figure 5.6 shows the critical-path delay test results after (stressed) aging. Only the results of 24 chips in the first batch of HTOL stress are shown, since the other 24 in the second batch behave similarly. Note that the operational voltage of the Xentium has been decreased to 0.8 V to capture the critical-path delay, as there are failing devices from 0.7 V to 0.9 V during the flow of operations of the critical-path delay monitor (Figure 4.3).

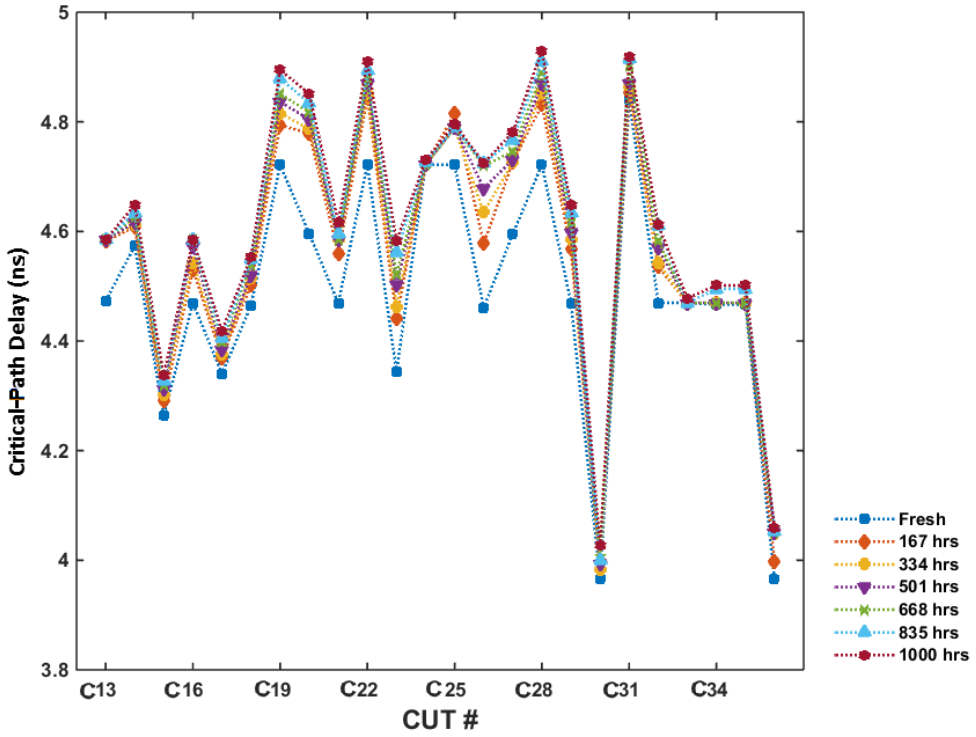


Figure 5.6: Critical-path delay test results for 24 (out of 48) aged Xentiums. The X-axis is the CUT (circuit under test) number from c13 to c36 (not all chip numbers are shown). The Y-axis shows the measured delay results in the case the operational voltage is 0.8 V.

One can observe that most of the chips show an increased delay (decreased maximum clock frequency) over stress time, except for chips number 24 and 33. The latter two did not show any degradation of the path delay from the start; chip 34 only degrades a little after 167 hours. The delays of most of the chips increase largely after the 1st week stress, compared to the following stress weeks. The largest delay-increase occurs after a stress time of 334 hours, in chip number 26. The average delay degradation rate after 1000 hours for all 48 Xentiums is around 2%. This indicates that the 1000-hours stress caused aging degradation for the Xentiums can be less than the process variations in different (wafer) corners.

5.4.2 I_{DDQ} Measurement Results

First, the measurements of the quiescent power current are presented using the Ridgetop current monitor QT1411, interposer board and HTOL board; subsequently the transient I_{DDT} measurements are provided.

After starting the test flow for I_{DDQ}, as discussed before in section 4.5.2, the quiescent current was measured of the first batch of 24 fresh Xentium-embedded SoCs. As can be seen in Figure 5.7, this current varies between 1.7 mA and 3.5 mA, depending on the chip. The average is around 2.3 mA, excluding the high-value outliers in the chips 15, 30 and 36. The rest of the I_{DDQ} values range within 0.35 mA of the average value, which is 15%.

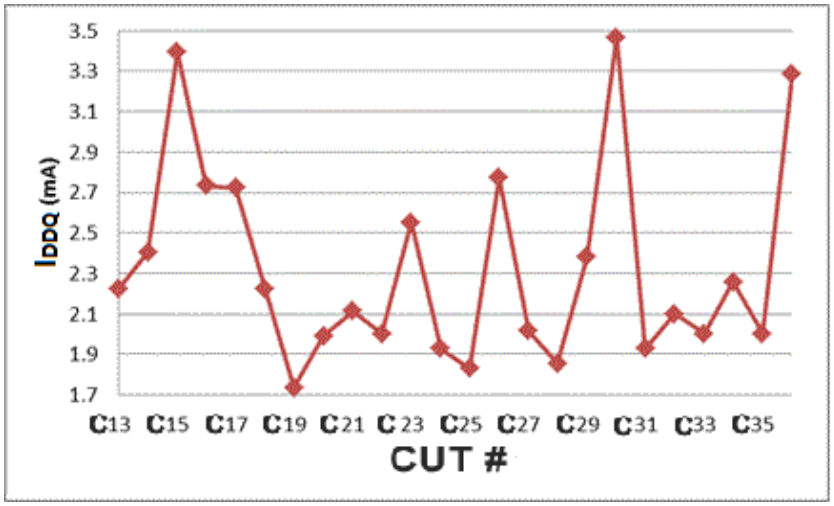


Figure 5.7: Measurement results of I_{DDQ} of 24 (out of 48) fresh Xentiums numbered from c13 to c36 (not all chip numbers are shown) at 30 °C using the QT1411 monitor and HTOL boards.

Figure 5.8 shows the I_{DDQ} measurement results for 24 Xentiums after aging; as one can see, most of the chips show a decreased I_{DDQ} over the stress time which was to be expected from physics. The I_{DDQ} values of most of the chips increase significantly after the 1st week of stress, as compared to the following stress weeks. One can also observe that the chip variation of the I_{DDQ} value such as c15 and c19 can be around 1.7 mA, which is much larger than the 1000-hours stressed caused by (HTOL) aging, being around 0.4 mA. This indicates that in the case of future embedded I_{DDQ} health monitors, calibration is necessary before measurements take place.

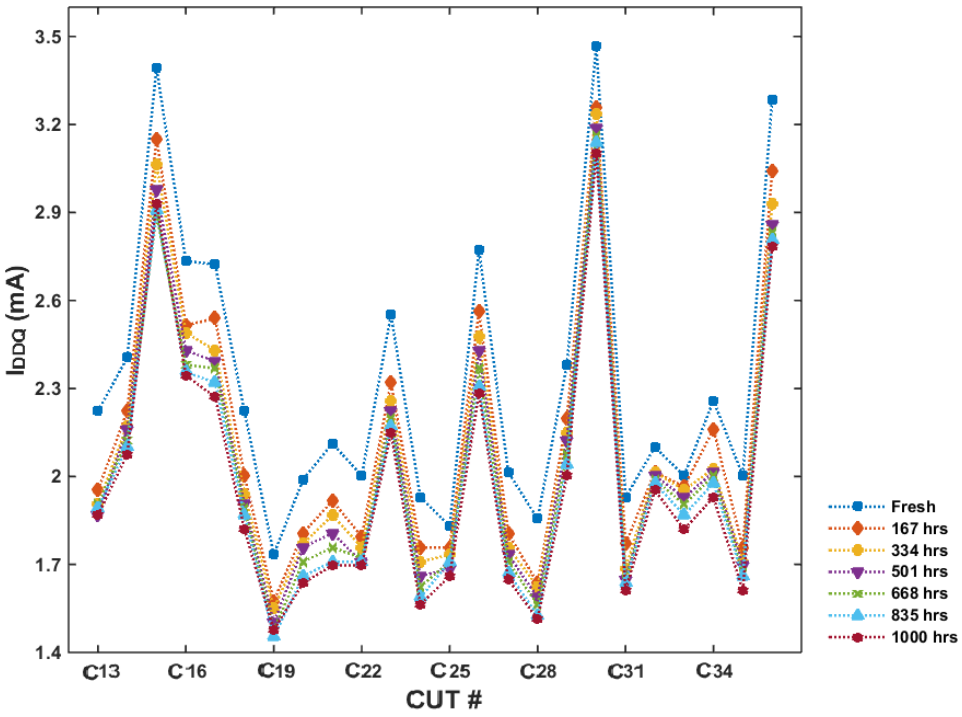


Figure 5.8: Measurement results for the I_{DDQ} of 24 (out of 48) Xentiums at 30 °C. The X-axis represents the CUT number from c13 to c36, while the Y-axis indicates the I_{DDQ} in the case where the operational voltage is 1 V.

5.4.3 I_{DDT} Measurements Results

This section introduces first the I_{DDT} measurement system and results via an Agilent N6781, and then another measurement system and results via our HTOL board and QT1411 monitor.

A. Using the Xentium evaluation board and the Agilent N6781 measurement system

The measurements were carried out using the Xentium evaluation board, in which case only the power current from the Xentium is involved, as can be seen in Figure 5.9. The measurements were carried out on two chips (c8 and c9) and the difference between them is indicated by the shading; it is around 0.2 mA, and in the worst case 2% of the maximum value (~15.5 mA) of the I_{DDT}. This current varies between 8 mA and 15 mA, depending on the chip.

From Figure 5.9 one can observe the different parts of the program (section 4.5.2) running on the Xentium being sequentially: the maximum workload program, A phase, M phase, S phase, P phase, ST phase and LD phase.

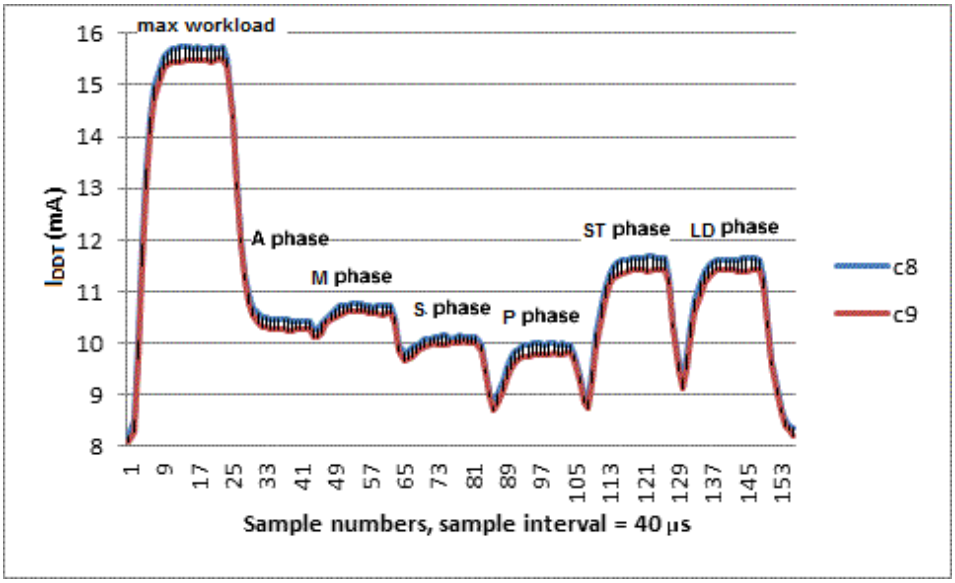


Figure 5.9: Result of I_{DDT} measurements using the Xentium evaluation board, employing the Agilent N6781 current-measuring power supply. Vertical axis shows I_{DDT} in milli-amperes, while the horizontal axis depicts the different sample points (1 sample point ~ 160 instructions) for the Xentium.

B. Using the HTOL board and the QT1411 current monitor

After starting the test flow for I_{DDT}, as discussed before, the transient current was measured of the first batch of 24 fresh Xentium-embedded SoCs. In Figure 5.10, the measurement result of one fresh chip (c13) is shown.

In order to better illustrate the different shapes in the different samples (as compared to Figure 5.9 measured with the Agilent N6781), the following should be noticed:

- because of the limited memory size for measured data in the QT1411, as compared to the Agilent current-measuring power supply, the number of loops in the test program has been reduced by a factor of four. If the horizontal scales would be identical, the length during each unit-phase (A, M etc.) will be less.

- because the QT1411 is much more accurate (sampling period is 40 µs) as compared to the Agilent instrument (sampling period is 32 ns), and also includes accurate

decoupling compensation, the transients of the current reaching the peak will be less smooth. This is in contrast with Figure 5.9.

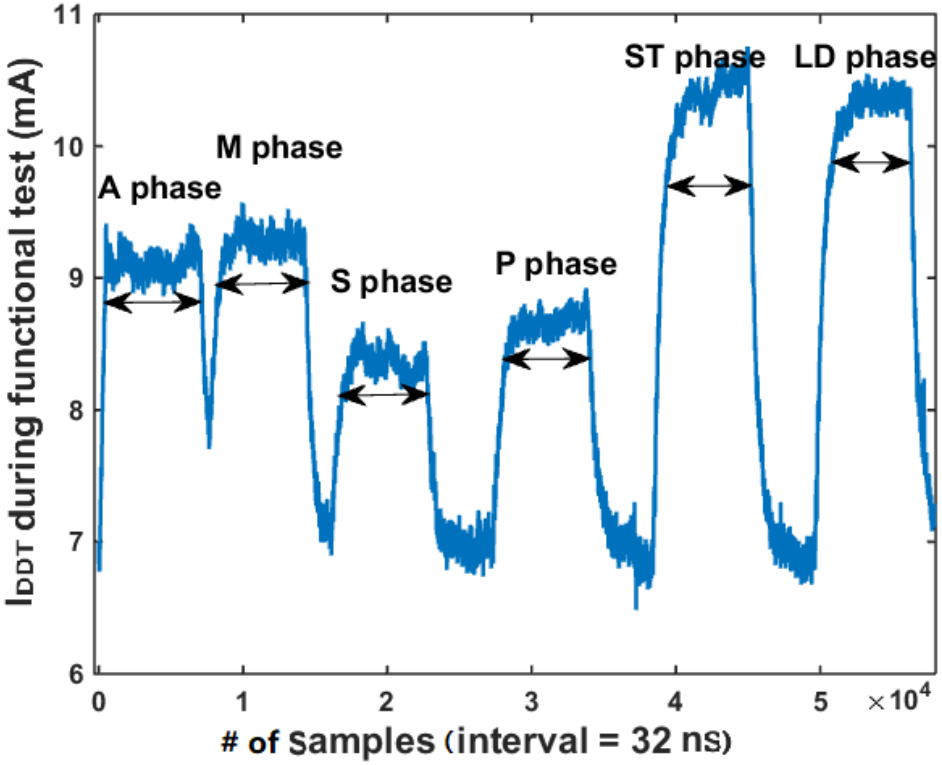


Figure 5.10: Measurements of I_{DDT} of one Xentium processor (c13) using the QT1411 current monitor and HTOL boards; one round of I_{DDT} measurements includes 6 phases in the case different execution units are running sequentially in the Xentium.

Based on our developed I_{DDT} testing program in section 4.5.3, the current during the test run (around 60000 samples per run) has been measured. The result in Figure 5.10 shows the transient currents of 6 execution units in the Xentium, in which one can distinguish 6 phases.

The peaks of the I_{DDT} (except the maximum workload values) in these 6 units/phases have been extracted. They can be sorted in a descending order of I_{DDT} values such that the currents in the ST unit > LD unit > M unit > A unit > P unit > S unit.

Figure 5.11 shows the fresh measurement results for 8 (out of 48) Xentiums in the first batch of the accelerated testing; this current varies between 6 mA and 15 mA, depending on the chip. Similar I_{DDT} currents can be observed as in Figure 5.10.

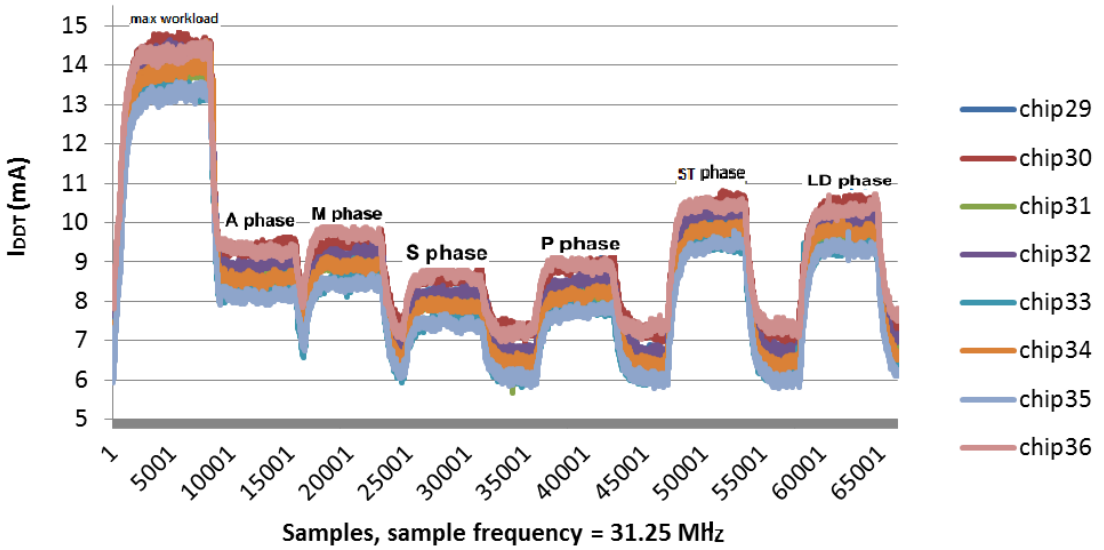


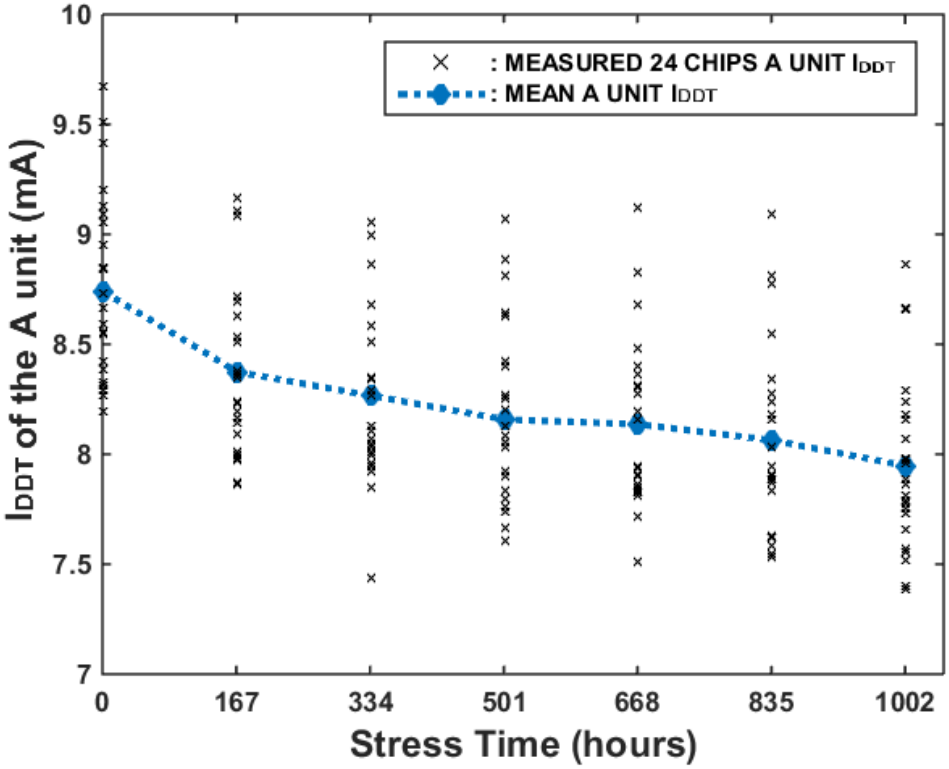
Figure 5.11: I_{DDT} test results employing the QT1411 current monitor and the HTOL board for 8 (out of 48) fresh chips at a temperature of 30 °C and a supply voltage of 1V. The X-axis indicates the sample points, while the sampling interval for the measurement is 32 ns; the Y-axis unit for I_{DDT} is in mA.

The maximum amplitudes of the current in the 6 phases/units during the I_{DDT} monitoring have been measured. Figure 5.12 shows the I_{DDT} measurement results based on the different units, i.e. the A unit, LD unit, M unit, P unit, S unit and ST unit (see section 4.5.3) versus aging. Meanwhile, the mean I_{DDT} values of 24 Xentiums of each individual unit are represented by the blue line.

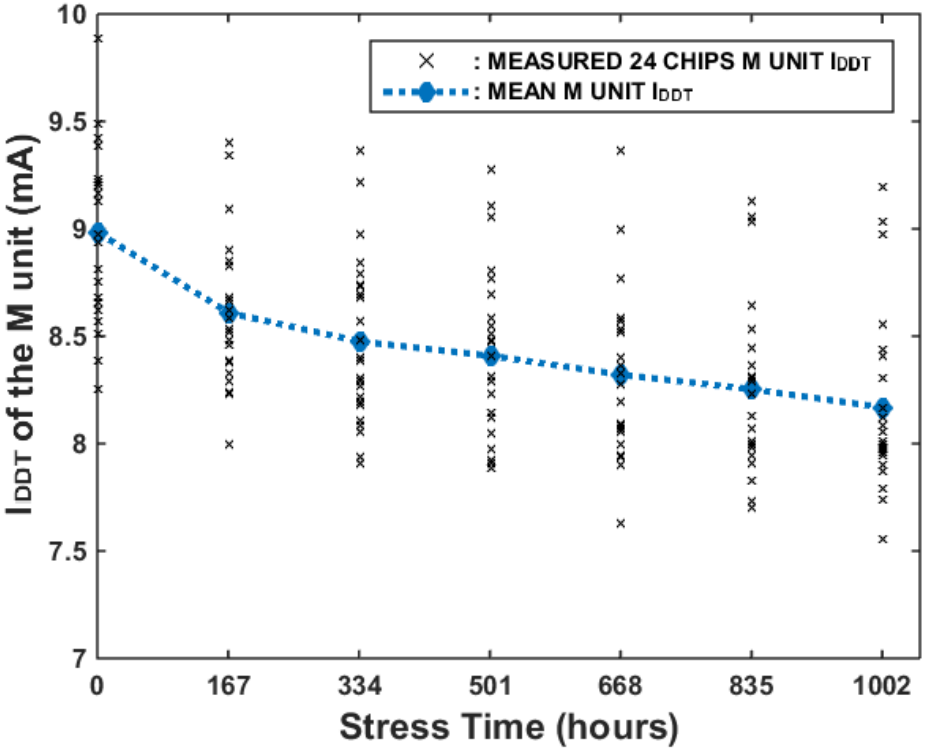
One can observe that all chips show a decreased value of I_{DDT} of all different units over the stress time which was to be expected from a physical point of view. The maximum degradation for most of the chips is occurring in the first stress period, as compared to the following stress weeks.

The range of measured currents in each stress period of each unit is normally larger than the degradation over the whole period; e.g. in Figure 5.12a, the measurement results at 668 hours, show a range of 1.5 mA, while the whole mean-degradation from 0 to 1002 hours is around 0.8 mA for most units. Similar as in the case of I_{DDQ} techniques, this indicates that future embedded I_{DDT} health monitors should undergo a calibration before use.

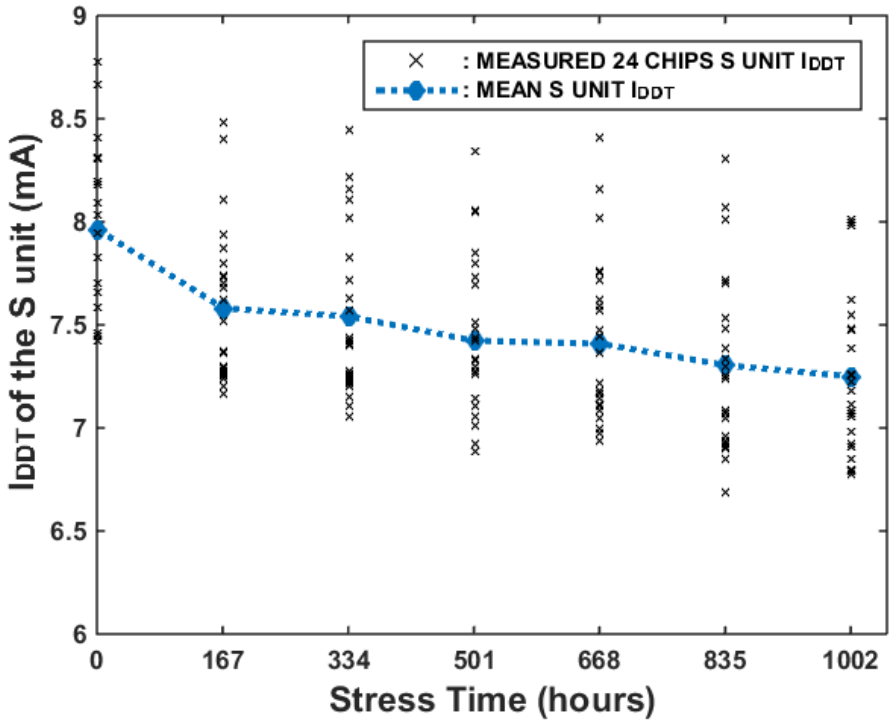
The largest average-I_{DDT} decrease after 1002-hours HTOL stress occurs in the A unit, which is seen from around 8.8 mA to 7.9 mA. The largest range variation is measured after a stress time of 1002 hours in the LD unit.



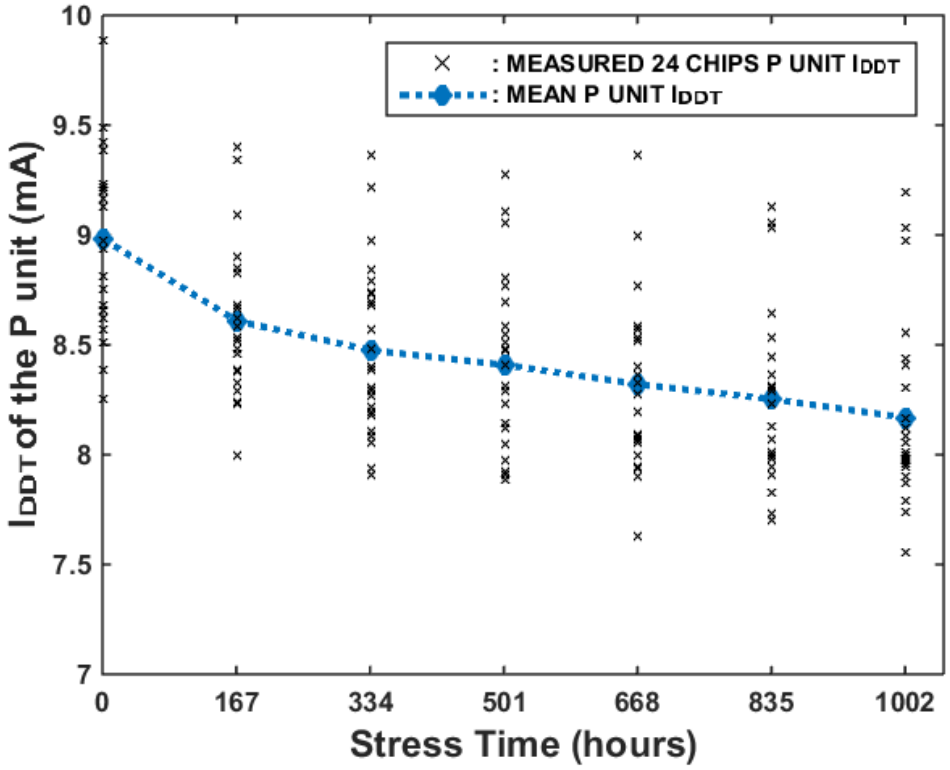
a) 24 Xentium cores, I_{DDT} measurements during the A phase versus aging.



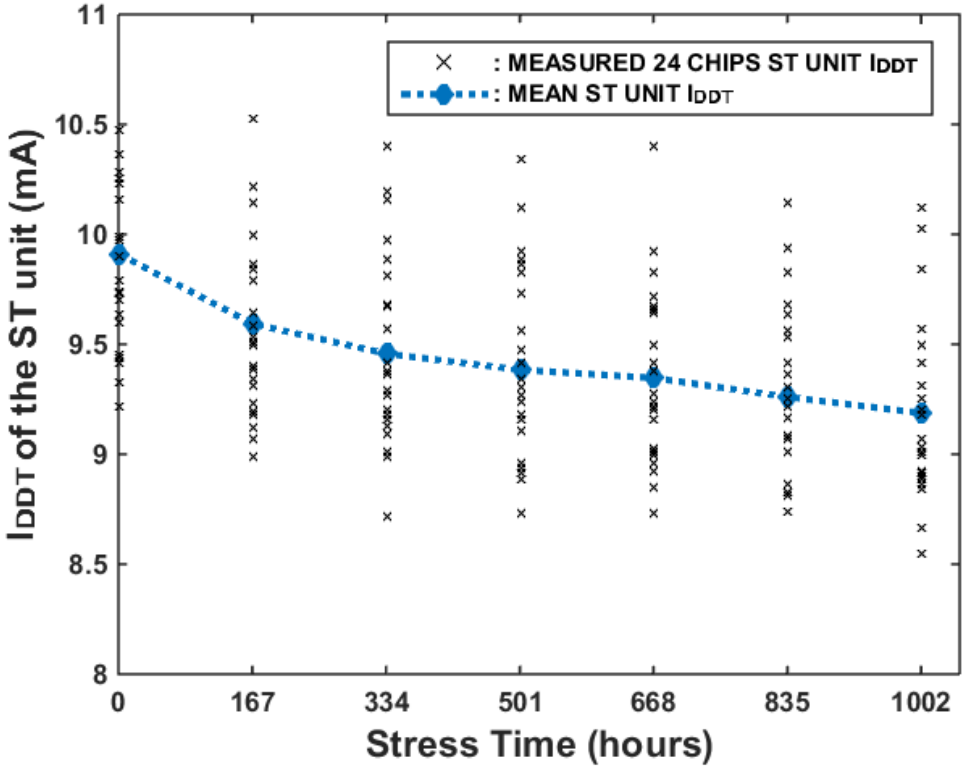
b) 24 Xentium cores, I_{DDT} measurements during the M phase versus aging.



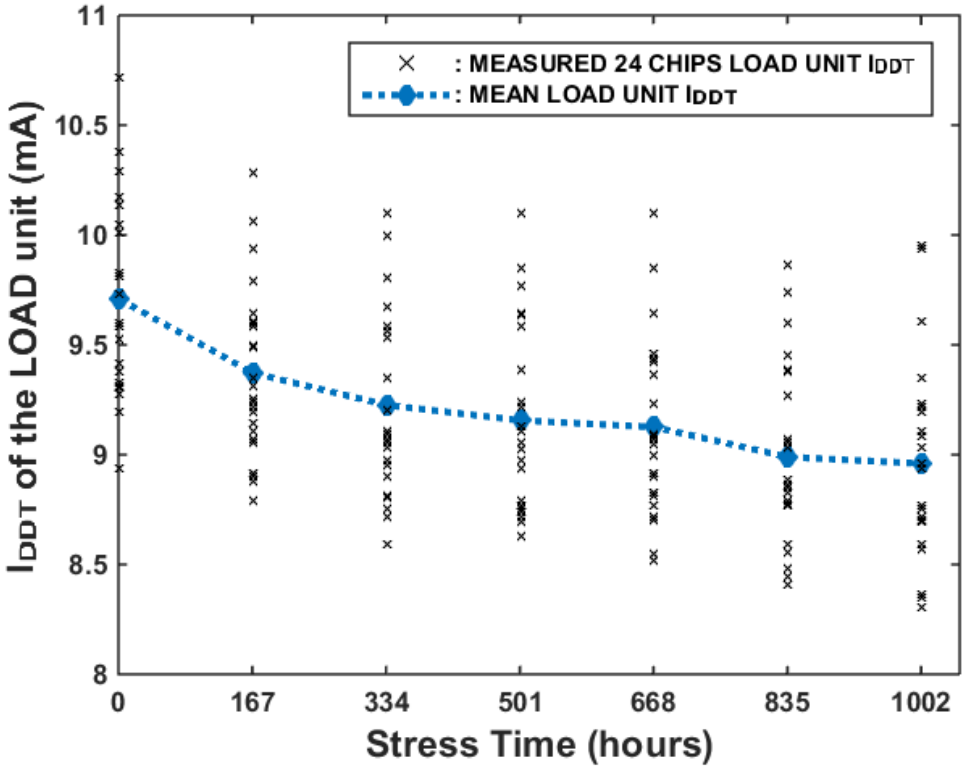
c) 24 Xentium cores, I_{DDT} measurements during the S phase versus aging.



d) 24 Xentium cores, I_{DDT} measurements during the P phase versus aging.



e) 24 Xentium cores, I_{DDT} measurements during the ST phase versus aging.



f) 24 Xentium cores, I_{DDT} measurements during the LD phase versus aging.

Figure 5.12 (a - f): Measurement results for I_{DDT} of 24 (out of 48) Xentiums at 30 °C with regard to different Xentium functional units. The X-axis indicates the stress time. The Y-axis represents the I_{DDT} values in the case the operational voltage is 1 V, and the arithmetic mean value is indicated with blue dashed lines.

5.5 Accelerated Testing Results Analysis for All the Measured Parameters

This section shows the HTOL-based AT results analysis from the previous section, being the critical-path delay, I_{DDQ} and I_{DDT}.

5.5.1 Critical-Path Delay Results Analysis

Figure 5.13 visualizes the delay degradation versus the operational time based on the measurement results shown in Figure 5.6. The mean delay values of 24 Xentiums is

drawn in the orange colour, while a blue power-law line is fitted with this mean delay value.

As one can see, the variation of the individual chip measurements is typically larger than our limited degradation period. All Δdelay values are positive indicating an increase of delay in the critical-path of the Xentium processors over time, which is expected from theory. From the orange line in Figure 5.13, one can derive that the mean delay value changes (Δdelay) have a *power* dependency with respect to the aging time t:

$$\Delta delay(t) = l \cdot t^m \qquad \text{Eq. (5.1)}$$

where *l* is a specific constant for each specific Xentium, and the power index m is 0.39 and 0.4 for two batches of measured Xentiums respectively.

Considering the aging mechanism of NBTI, it has been verified based on experiments that the threshold voltage changes ΔV_{TH} have a power-law increase under DC stress conditions [Chak 04]:

$$\Delta V_{TH}(t) = Kd \cdot t^n \qquad \text{Eq. (5.2)}$$

where Kd is a parameter which is dependent of temperature, power supply V_{DD}, as well as device geometry and technology, and n is the power-law coefficient. The equation has a similar form as was found in Eq. (5.1).

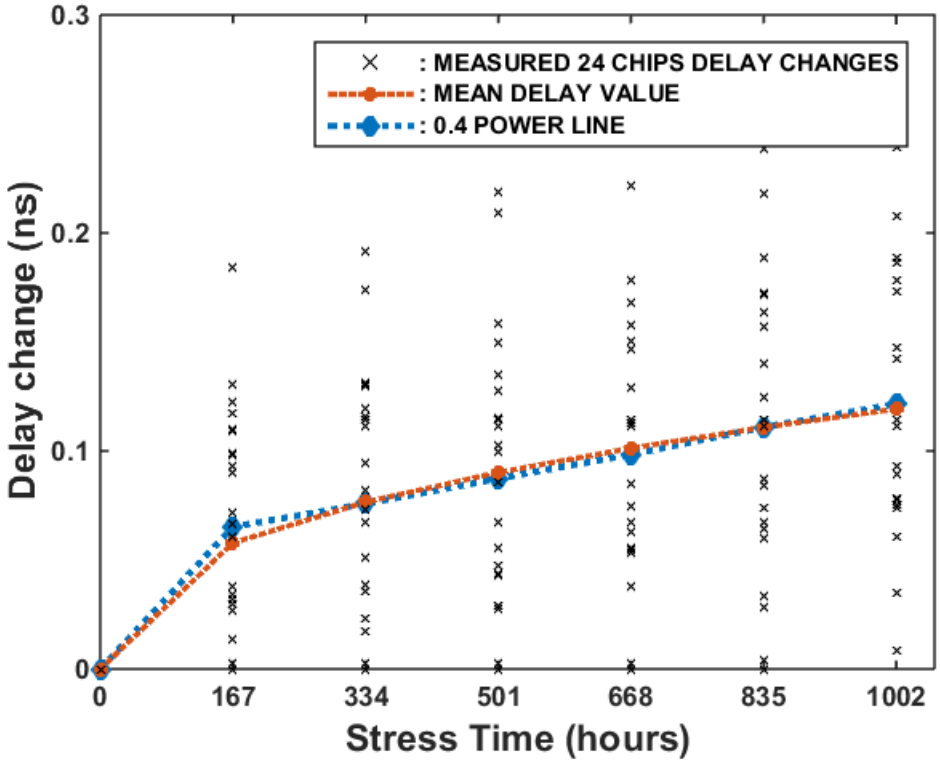


Figure 5.13: Measured critical-path delay changes using the Xentium evaluation board for 24 (out of 48) Xentiums. Arithmetic mean is used (the orange line), and the least-squares fitting (the blue line) is employed for deriving the fitting coefficient.

5.5.2 I_{DDQ} Results Analysis

Figure 5.14 shows that the I_{DDQ} changes have a similar power degradation behaviour as the previous delay changes:

$$\Delta I_{DDQ}(t) = p \cdot t^{0.39} \qquad \text{Eq. (5.3)}$$

where p is a specific constant for each specific Xentium.

In the first place, a difference is the mean degradation rate of I_{DDQ} which is somewhat smaller (the power-law coefficient is 0.39) as compared to the delay monitor. The I_{DDQ} changes are negative, showing the I_{DDQ} will decrease with stress which is confirmed by theory.

The measurement variations at each stress-time are quite different, the range of measurement variation at 1002 hours (around 0.3 mA) is somewhat smaller than the whole mean degradation difference (around 0.4 mA).

The orange line shows the plot of the mean I_{DDQ} values for these 24 Xentiums, while the blue line indicates the derived fitting result employing the least-squares fitting.

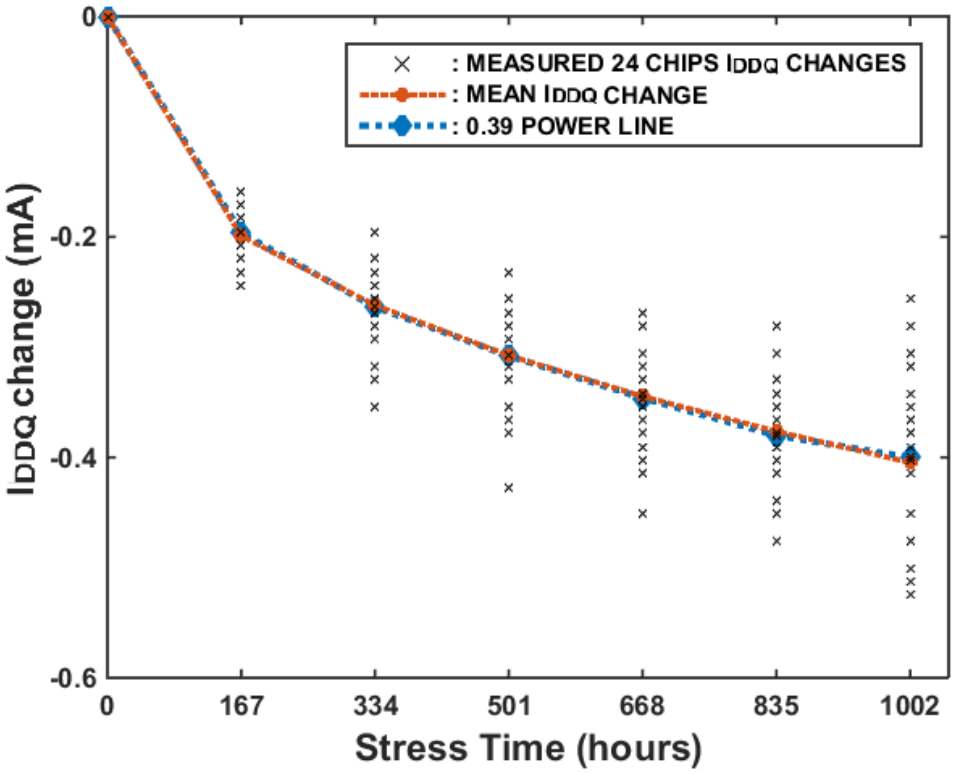


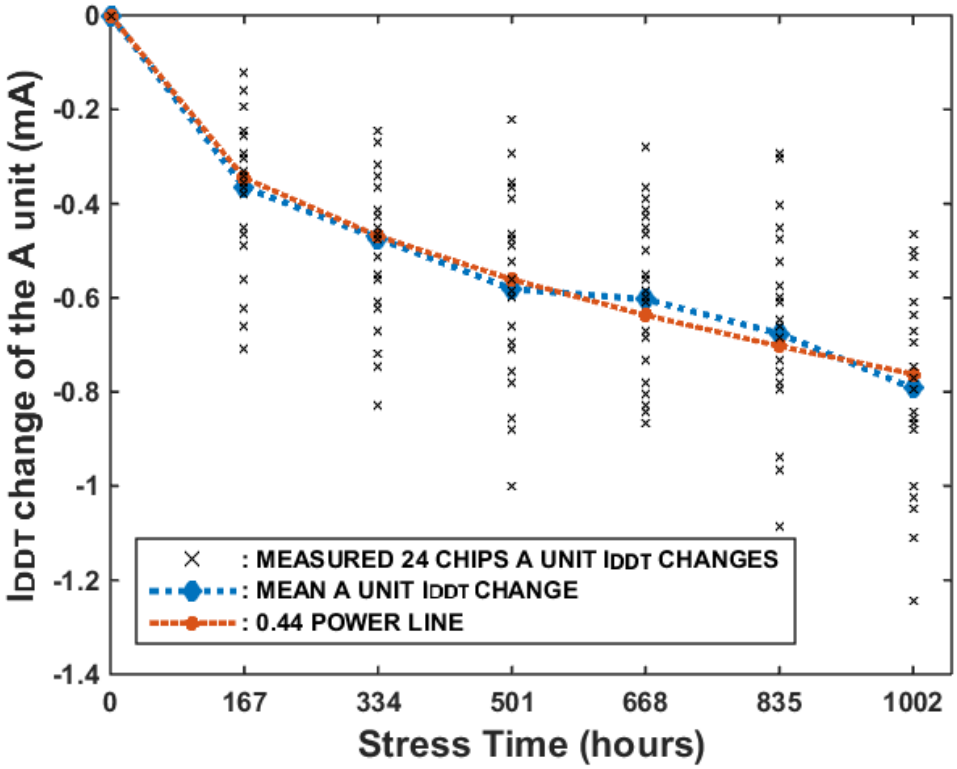
Figure 5.14: Measurement result of I_{DDQ} changes versus stress time of 24 (out of 48) Xentiums, based on the I_{DDQ} measurement system shown in Figure 5.4.

It should be noted that under other aging mechanisms such as TDDB, the I_{DDQ} can increase with time [Maso 04]; the fact that this did not show up in our test results indicates that NBTI is the dominating aging mechanism in 90 nm technology, which has been confirmed by other research [Chak 04].

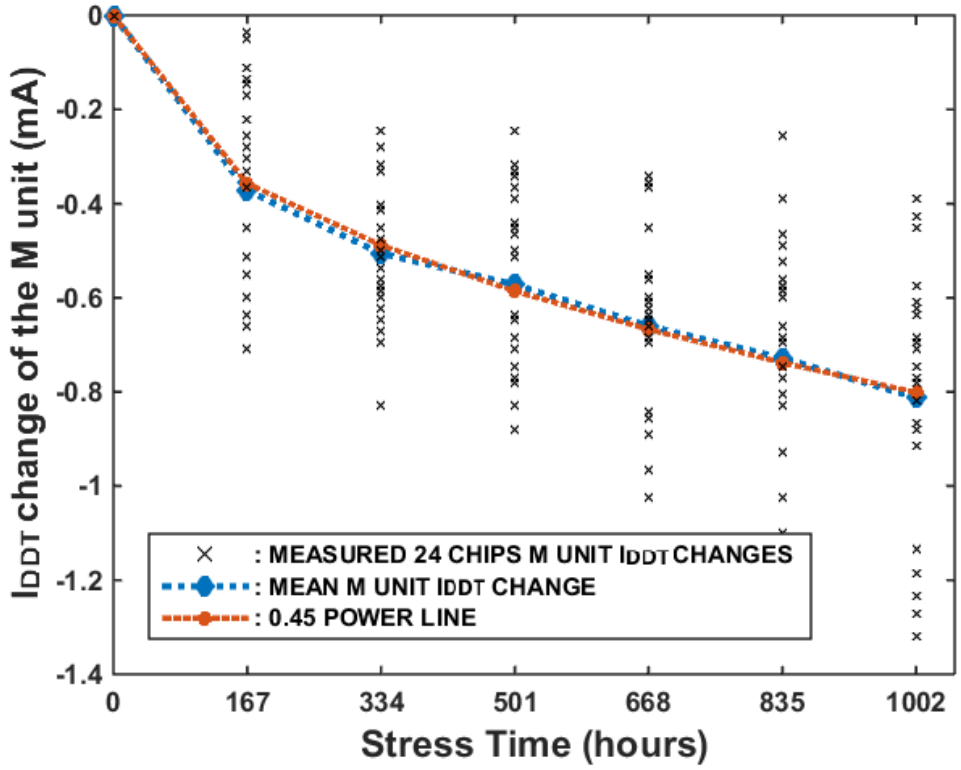
Based on the measurement results, the correlation coefficient between the I_{DDQ} testing units and delay has been calculated to be -0.94. This verifies that these two (delay and I_{DDQ}) monitoring techniques have a strong correlation, and therefore one monitoring technique can be the other technique's alternative in aging-monitor applications.

5.5.3 I_{DDT} Results Analysis

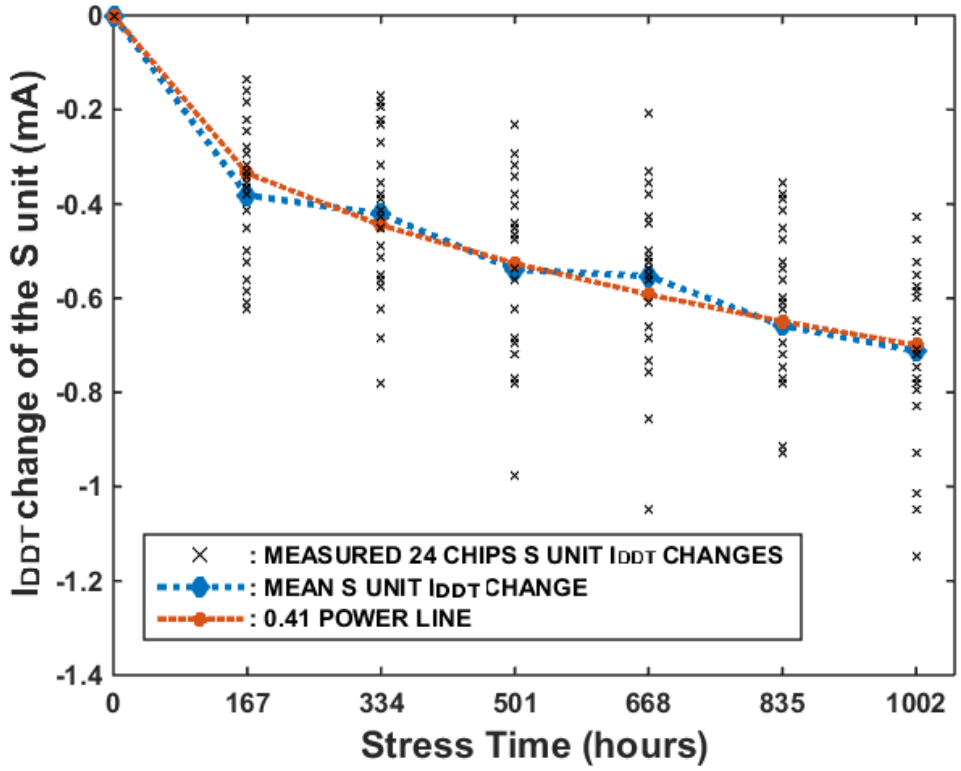
The maximum amplitudes of the current in 6 phases/units during the I_{DDT} monitoring have been measured. Figure 5.15 shows the I_{DDT} change over time which is similar in terms of dependency with respect to the aging time as in the critical-path delay and I_{DDQ} aging.



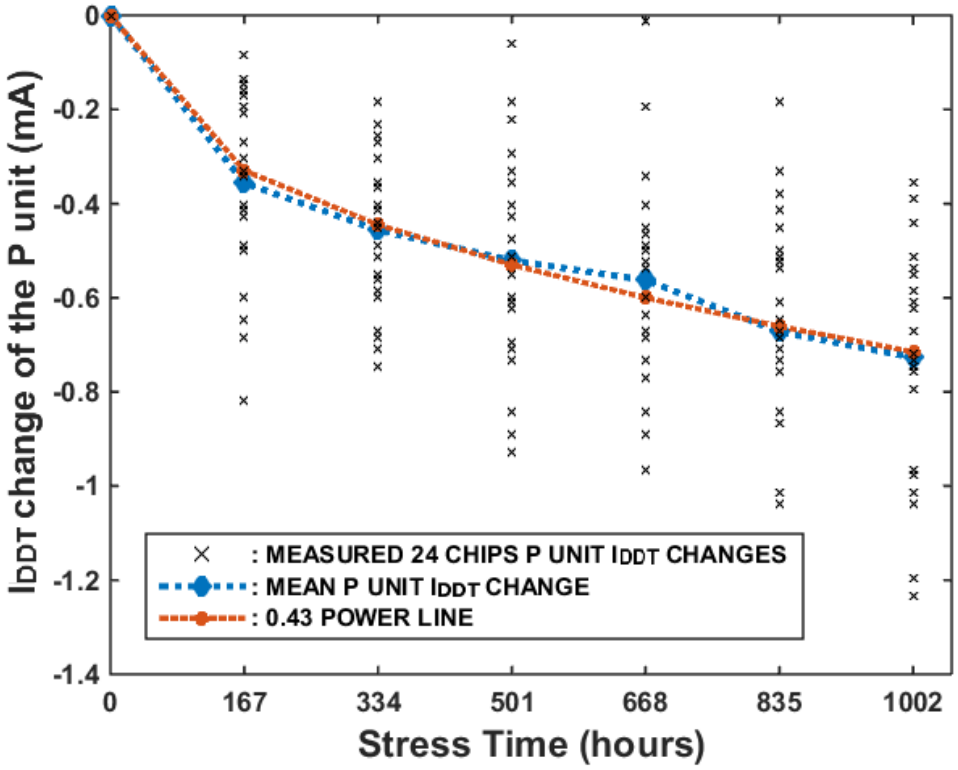
a) 24 Xentium cores, I_{DDT} change measurements during the A phase versus aging.



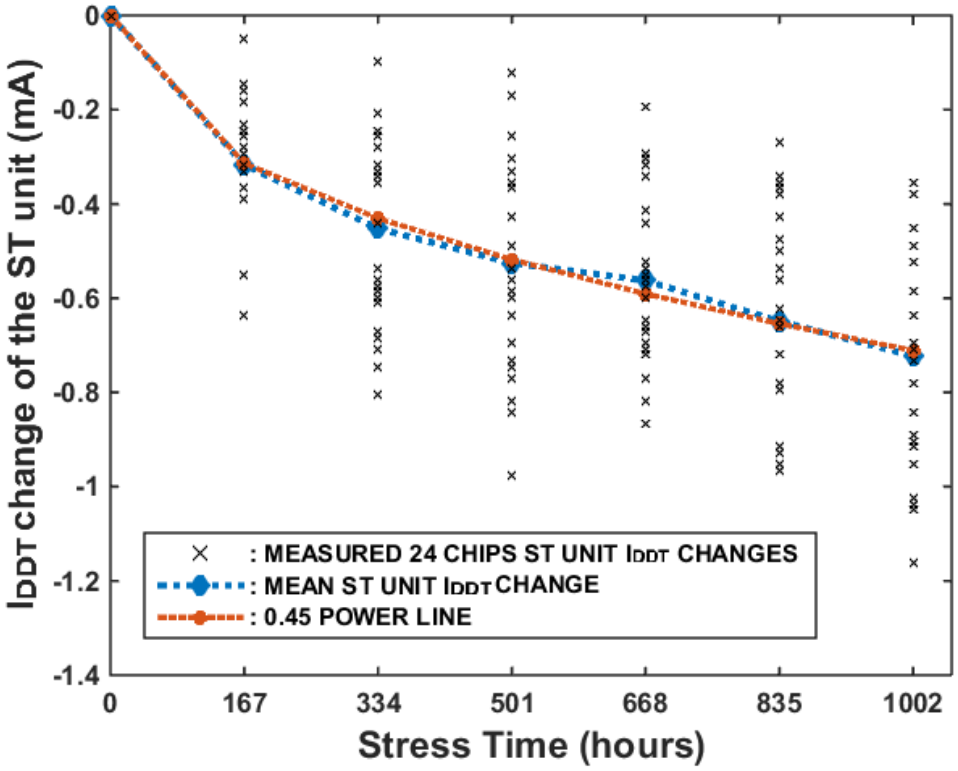
b) 24 Xentium cores, I_{DDT} change measurements during the M phase versus aging.



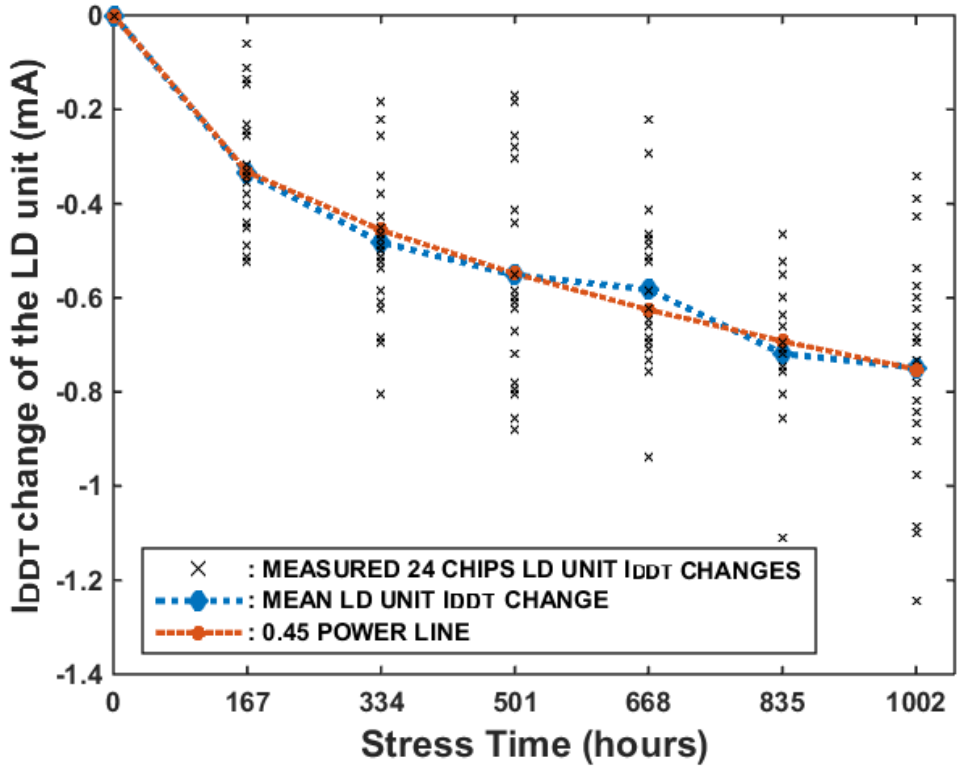
c) 24 Xentium cores, I_{DDT} change measurements during the S phase versus aging.



d) 24 Xentium cores, I_{DDT} change measurements during the P phase versus aging.



e) 24 Xentium cores, I_{DDT} change measurements during the ST phase versus aging.



f) 24 Xentium cores, I_{DDT} change measurements during the LD phase versus aging.

Figure 5.15 (a - f): Aging dependency of I_{DDT} change measurement results drawn from the I_{DDT} changes in 6 Xentium execution units. Measurements are taken at a temperature of 30 °C, while the V_{DD} is 1 V.

The blue line in Figure 5.15 plots the mean I_{DDT} values of 24 Xentium processors. The I_{DDT} changes in all execution units show a similar degradation trend.

All ΔI_{DDT} values are negative, indicating a decrease of I_{DDT} in all units of the Xentium processors over time. The most significant change for all chips is occurring in the first stress period, as compared to the following stress weeks. The aging-caused I_{DDT} decrease for the whole degradation period is e.g. in Figure 5.15c around 0.7 mA; however, the process variation in each stress period of each specific Xentium unit is normally larger than the whole degradation period. For instance in Figure 5.15c, the measurement results at 1002 hours, show a range of 0.8 mA. There is a somewhat higher value than expected at 668 hours, in most cases for all I_{DDT} units; this could be due to a measurement parameter (e.g. T) issue.

The mean I_{DDT} value changes (plotted as a blue line) have a power-law dependency with respect to the aging time t:

$$\Delta I_{DDT}(t) = m \cdot t^{u} \qquad \text{Eq. (5.4)}$$

where m is a specified constant for each unit (A, M, etc.) of the Xentium. The power-scale coefficient (u) differs in the 6 units ranging from 0.41 (S unit) to 0.45 (ST unit).

The power-law relation is in line with our previous delay and I_{DDQ} testing results shown in Eq. (5.1) and Eq. (5.3). This verifies the effectiveness of all our testing techniques for aging (trend) detection of (Xentium) processors. Hence similarly, one can conclude that NBTI is the dominating aging mechanism in our 90 nm technology.

Compared to the I_{DDQ} testing proposed previously in [Maso 04], the measurement results show that I_{DDT} is more sensitive to aging as compared to the I_{DDQ} test, where the power-scale coefficient is around 0.39. Moreover, monitoring the I_{DDT} results of different units provides more comprehensive understanding of the aging in the processor.

Figure 5.16 illustrates the correlation analysis between the measured Δdelay and ΔI_{DDT}. One can observe that the correlation coefficients between different I_{DDT} testing units and delay are ranging from -0.82 to -0.93. This verifies that these two (delay and I_{DDT}) monitoring techniques have a strong correlation, and therefore one monitoring technique can be the other technique's alternative in aging-monitor applications.

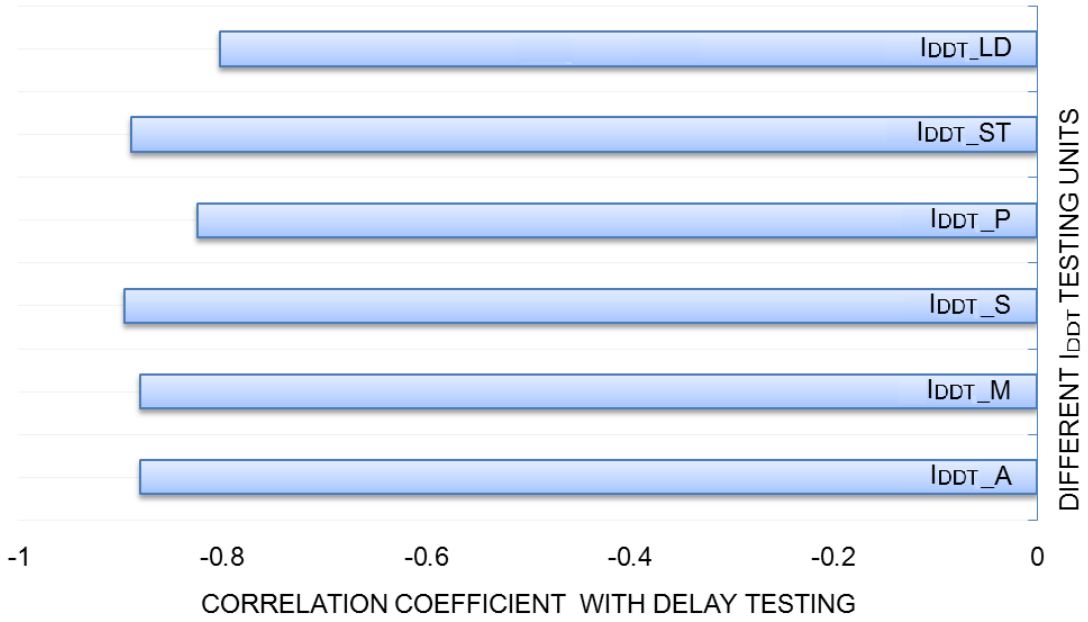


Figure 5.16: Correlation coefficients between 6 different units I_{DDT} measurement results with respect to the delay testing results using the evaluation board. A value of -1 indicates a full correlation, while -1 to -0.7 indicates a strong correlation.

5.6 CONCLUSIONS

In this chapter, the functionality of all designed and implemented boards as well as the developed (monitoring) software program have been validated. 1000 hours accelerated testing experiments with extensive (> 4 Gbit data) measurements have been completed on 48 (2 batches of Xentiums with 24 chips) stressed in each batch. To limit the number of graphs in this thesis, only the measurement results for the first batch have been shown and analysed, while the second batch is similar to the ones presented in this thesis. The measurement data shows that individual process-variations of chips are larger than the aging effect during the 1000-hours stress period; hence for the lifetime prediction, first a calibration step for the HMs should be employed to compensate for variation of individual chips. The data shows that both delay changes, I_{DDQ} changes and I_{DDT} changes in each functional unit of the processor have a power-law trend, which has the same degradation trend as NBTI aging degradation at the (V_{TH}) transistor level. This proves that NBTI is the dominating aging mechanism in our 90 nm technology. Meanwhile, strong correlation coefficients have been observed between the results from these different monitoring techniques, testifying that the $I_{DDQ/T}$ monitoring approach is a serious candidate in aging-caused delay degradation detection. Furthermore, as the lifetime-prediction model (Chapter 6) requires, the lifetime threshold can be extracted from the path-delay monitoring results. Our degradation analyses based on our monitors and the strong correlation between these monitors form the basis for our lifetime prediction to be discussed next.

REFERENCES

[Chak 04] S. Chakravarthi, A. T. Krishnan, V. Reddy, et al., "A comprehensive framework for predictive modeling of negative bias temperature instability," in IEEE International Reliability Physics Symposium Proceedings (IRPS), pp. 273-282, 2004.

[Esco 06] L. A. Escobar and W. Q. Meeker, "A review of accelerated test models," in Statistical Science, pp. 552-577, 2006.

[Kerk 14] H. G. Kerkhoff, J. Wan, and Y. Zhao, "Linking aging measurements of health-monitors and specifications for multi-processor SoCs," in IEEE International Conference On Design & Technology of Integrated Systems In Nanoscale Era (DTIS), pp. 1-6, 2014.

[Jede 10] "JEDEC standard JESD22-A108D," http://www.jedec.org/ standardsdocuments/, November 2010.

[Jede 11] "JEDEC standard JESD22-A105C," http://www.jedec.org/ standardsdocuments/, January 2011.

[Liao 06] H. Liao and E. A. Elsayed, "Reliability inference for field conditions from accelerated degradation testing," in Naval Research Logistics, vol. 53, pp. 576-587, 2006.

[Maso 04] P. W. Mason, A. J. L. Duca, C. H. Holder, et al., "A methodology for accurate assessment of soft-broken gate oxide leakage and the reliability of VLSI circuits," in IEEE International Reliability Physics Symposium (IRPS), pp. 430-434, 2004.

[Saba 02] S. S. Sabade and D. M. H. Walker, "I_{DDQ} test: will it survive the DSM challenge?," in IEEE Design & Test of Computers, vol. 19, pp. 8-16, 2002.

[Star 10] "STARS: Sensor Technology Applied in Reconfigurable systems", 2010. http://cas.et.tudelft.nl/Research/project.php?id=33

[Zhao 14] Y. Zhao, E. Strooisma, and H. G. Kerkhoff, "Xentium Health Test Plan," University of Twente, Technical Report, Nr. ewi14 / CAES:2015, the Netherlands, 2015.

[Zhao 15a] Y. Zhao and H. G. Kerkhoff, "Application of functional I_{DDQ} testing in a VLIW processor towards detection of aging degradation," in International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), pp. 1-5, 2015.

[Zhao 15b] Y. Zhao and H. G. Kerkhoff, "Unit-based functional I_{DDT} testing for aging degradation monitoring in a VLIW processor," in Proceedings of the Euromicro Conference on Digital System Design (DSD), pp. 353-358, 2015.

Chapter 6

REMAINING LIFETIME PREDICTION FOR THE MP-SOC VIA I_{DDX} MONITORING

ABSTRACT *– Previous chapters have shown the usage of the health-monitoring method for obtaining health information of the target SoCs in combination with the implementation of the accelerated testing of the Xentium processor. Based on the experimental results, this chapter illustrates the development of the degradation trend models based on a Genetic Algorithm (GA). The construction of the remaining lifetime prediction model for the Xentium employs the alternate I_{DDX} monitoring which is presented later. It is found that the calculated remaining lifetime predicted by I_{DDX} monitoring is close to that of delay-time monitoring.*

Parts of this chapter have been published as paper titled "A Genetic Algorithm Based Remaining Lifetime Prediction for a VLIW Processor Employing Path Delay and I_{DDX} Testing" in the International Conference on Design & Technology of Integrated Systems, 2016, and "Highly Dependable Multi-Processor SoCs Employing Lifetime Prediction Based on Health Monitors" in the Asian Test Symposium, 2016.

6.1 INTRODUCTION

Aging causes performance degradation and finally a failure in MP-SoCs. It can be life-threatening [Yanj 09], especially in applications such as for automobiles, aircrafts or medical equipment, as introduced in Chapter 2. The previous chapters have discussed the effectiveness of using the health-monitoring (HM) method in a target Xentium processor for parameters such as e.g. the critical-path delays [Zhao 15a], quiescent power-supply current (I_{DDQ}) [Zhao 15b] and transient power-supply current (I_{DDT}) [Zhao 15c]. These monitoring techniques have demonstrated their effectiveness in assessing aging causing reliability degradation. This chapter aims to develop methods that use this health-data information to predict the remaining lifetime of our monitored Xentium processors. By making failures more predictable, maintenance actions can be taken at some level before the failure becomes critical. This leads to less unscheduled downtime [Brot 00].

Based on the underlying aging mechanisms, this chapter will model the degradation in a statistical way, which is one of the data-driven approaches of the degradation-based model described in section 2.7.4. The failure rate for electronic devices is normally characterized by their own bathtub curve during their whole lifetime [Yudo 11], but it is observed that the degradation can follow different paths at different moments of their life.

Figure 6.1 illustrates the notion that in a degradation model an asset will eventually fail if its degradation signal just reaches a specified failure threshold δ, which depends on the operational voltage V_{DD} and environmental conditions (e.g. temperature). For the scenario shown in Figure 6.1, the measured I_{DDX} (in chapter 5) is applicable as a degradation signal, since it will decrease with the operational time because of the aging. The remaining lifetime will be calculated from the current status to the time when the degradation signal Y(t) reaches the (dynamic) failure threshold δ, which is assumed to be known in advance from the design.

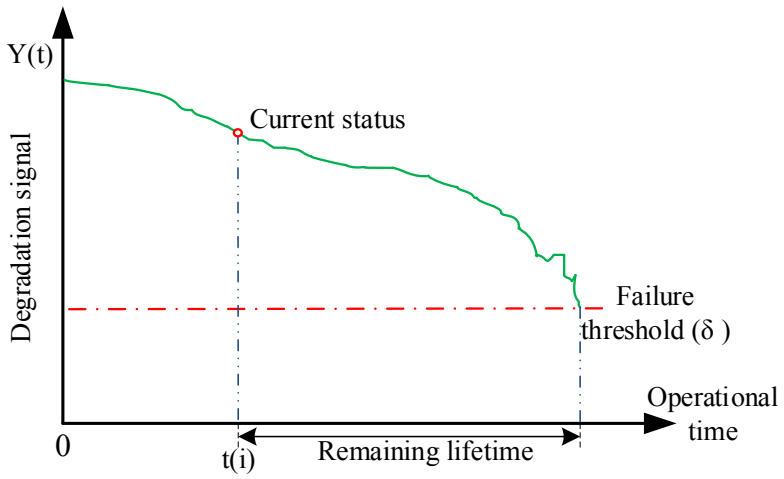


Figure 6.1: A single degradation signal Y(t) with the remaining lifetime defined.

It is clear that determining the degradation trend and building the degradation model are crucial for an accurate remaining lifetime prediction. In our case, a genetic algorithm (GA) [Holl 92] is employed for the purpose of prediction. Normally genetic algorithms are used for general optimization problems with a multimodal target. Our research will model the aging degradation based on the GA method, and subsequently determining a lifetime prediction of our (embedded) Xentium processor in the target MP-SoC, in such a way that the potential lifetime of the target MP-SoC can be enhanced based on the previously mentioned remapping techniques (Figure 2.4).

Another key concept in our current approach is alternative monitoring, which will use the I_{DDX} monitoring instead of the critical-path delay monitoring. The latter is in the current set up more time consuming and less economic to use for degradation monitoring.

Previous chapters show that the monitoring results between the critical-path delay and the I_{DDX} are highly correlated. This chapter will build a mapping function between them based on the available health-monitoring data and usage of regression techniques. On top of that, we have developed an algorithm to employ the I_{DDX} monitoring measurement results to carry out the lifetime prediction.

The remainder of this chapter is organized as follows: section 6.2 gives a brief introduction to degradation-trend optimization. In section 6.3 the GA application and usage in our model based on the critical-path delay monitoring measurement results for building our degradation path is explained, which includes expressions of selected paths in a statistical way. Section 6.4 illustrates the GA-based method to accomplish the lifetime prediction from the I_{DDX} monitoring results. Finally, conclusions are provided.

6.2 DEGRADATION-TREND OPTIMIZATION BASED ON GENETIC ALGORITHMS

This section illustrates the construction of the degradation trend via data mining techniques using genetic algorithms. This trend will be employed for the remaining lifetime prediction.

With the development of artificial intelligence, genetic algorithms have been widely used to solve path planning and optimization problems [Sonm 15], [Tunc 12].

Genetic algorithms are a subset of evolutionary algorithms (EA) [Gold 88], [Mitc 98]. Since they are based upon robust natural selection processes, genetic algorithms have demonstrated the capability of dealing with optimization problems that conventional techniques find difficult to solve.

They have shown to be capable of solving problems like:

- optimizing continuous & discontinuous parameters
- optimizing complex cost functions
- processing measured & simulated data
- can deal with missing samples in a data stream

The usage of genetic algorithms offers several advantages, since they do not require an initial approximation of the solution, can handle discontinuous and nonlinear objective functions, and allow seeking in data with multiple local optima [Kwon 06]. Furthermore, they can use and operate a set of candidate solutions rather than refining a single-candidate solution as provided by other optimizers.

In the following part, the genetic algorithm (GA) procedure of Holl [Holl 92] has been applied for the optimization of the degradation trend.

6.3 GA PROCEDURE FOR OUR CRITICAL-PATH DELAY DEGRADATION OPTIMIZATION AND LIFETIME PREDICTION

In our research, as seen from the measured degradation values (increase in delay or decrease in I_{DDX}) [Zhao 15c], the degradation follows a *non-linear* trend. Furthermore, every Xentium processor (in a multi-processor SoC) has its own best fitted aging degradation model based on their health-monitor test results.

Based on the static-timing analysis, the generic critical-path delay is available for a generic Xentium processor [Reco 11]. Meanwhile, the measured critical-path delay for each Xentium is available. However, there exists no model that is the *optimal* solution for all monitored Xentium processors by the traditional optimization methods. Thus, it requires an efficient optimization algorithm to determine the combination of coefficients in the model which minimizes the degradation trend error for *all* Xentium processors. A GA is more likely to converge to a global optimum, since the algorithm starts searching from a given (measured) population of points, and is based on probabilistic rules.

Since the outcome of the GA optimization of the critical-path delay degradation is the trend for a group of Xentium processors, the optimized remaining lifetime prediction will be available for this group. The remaining lifetime of each Xentium can be calculated by including the latest critical-path delay results in the learned degradation trend respectively with the generic critical-path delay threshold.

6.3.1 GA PROCEDURE FOR OPTIMIZING THE CRITICAL-PATH DELAY DEGRADATION

Our GA has been employed to optimize sets of coefficients for the proposed degradation model with regard to the critical-path delay.

Generally, in the GA procedure, a population of candidate solutions (called individuals) to an optimization problem is evolved towards a better solution. The individual has unique chromosomes, which are used to be evolved during the so-called crossover and mutation operations. This is realized by encoding the chromosomes into

binary bits (strings of 0s and 1s), and change these bits during the GA procedure [Holl 92].

The evolution procedure starts from a population of randomly generated individuals, and these individuals will follow an iterative process, with the population in each iteration called a generation [Holl 92]. In each generation, the fitness of every individual in the population is evaluated. The more fit individuals are selected from the current population, and each individual's *chromosomes* are modified (recombined and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration. Genetic algorithms normally terminate when a maximum number of generations has been produced [Gold 88].

The flowchart of our GA in the critical-path delay optimization procedure is shown in Figure 6.2. Fundamental operations of the GA are summarized as follows:

1. *Initialization*: our target is searching for the optimal degradation model for the critical-path delay, based on historical data. As from the previous introduction, the individuals in the GA optimization are referred to as candidate solutions. These solutions consist of measured data corresponding to the shape of the critical-path delay degradation trend. From our historical delay data, the delay value changes (Δ delay) have shown to follow a *power* dependency with respect to the aging time [Zhao 15a]; one can therefore safely assume a critical-path delay in the Xentium has a power-law relation with respect to the time t:

$$f(t) = \text{delay}(t) = a + b * t^{c} \qquad \text{Eq. (6.1)}$$

The solution of parameters *a*, *b* and *c* in this case are the fittest individuals among the population of one GA generation, and a population consists of 100 individuals which are randomly generated in our case. This number has been chosen because if the initial population is too large, the algorithm will result in high computation costs; if it is too small, the algorithm will probably miss the optimal solution [Gold 88].

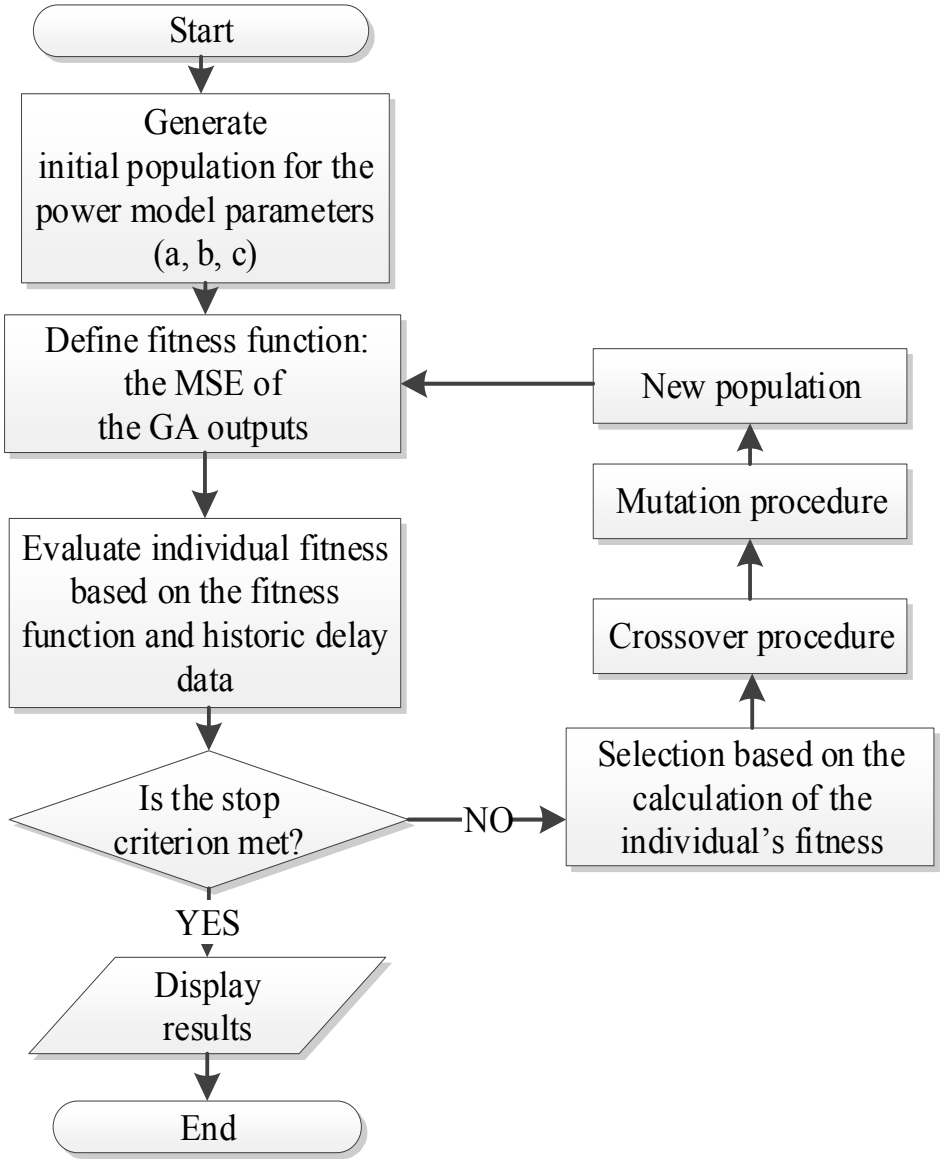


Figure 6.2: Flowchart of the used GA procedure for our delay-degradation trend optimization.

2. *The fitness function and fitness evaluation*: the fitness function is determined by the mean-squared error (MSE) of the GA results and the measured results and defined as:

$$\text{Fitness function} = \frac{\sum \text{N} \left(\text{delay}_{\text{GA}} - \text{delay}_{\text{measured}}\right)^2}{\text{N}} \qquad \text{Eq. (6.2)}$$

where N is number of our candidate solutions (a, b and c), i.e. 3, delay_{GA} are the GA selected individuals and $\text{delay}_{\text{measured}}$ are the measured critical-path delay results.

After the generation of the new population, there will be an evaluation of the adaptiveness for each individual based on the fitness function. The smaller the fitness value for the individual, the more front ranking this individual will become.

3. *Selection:* for this procedure the roulette-wheel selection [Blic 96] was used, i.e. the probability that an individual selected for placement in the next generation population is proportional to that individual's fitness calculated in step 2.

4. *Crossover:* together with the Mutation procedure in the next step, the second-generation population of individuals are now being generated. In these two steps, a pair of the "parent" solution is selected for producing a "child" solution, in such a way that, a new population is created which typically shares as many of the characteristics of their "parents". A crossover of the parents forms new offspring. As can be seen from Figure 6.3, the crossover is implemented via the so-called scattered crossover technique [Spal 03]. Individuals in the parent are paired off randomly. Then a random binary vector is created which selects the chromosome-bits (0's and 1's in Figure 6.3). If the vector bit is a *1,* the corresponding chromosome-bits of child 1 will originate from the Parent 1, and if the bit is a *0,* they will come from Parent 2. Finally the bits are combined to form two children (Child 1 and Child 2). This results in two 'children' with a mixture of the characteristics of both parents. A crossover probability of 1.0 indicates that all the selected chromosomes are used in reproduction. However, empirical studies have shown that better results are achieved by a crossover probability between 0.65 and 0.95, which implies that the probability of a selected chromosome surviving to the next generation unchanged (apart from any changes arising from mutation) ranges from 0.35 to 0.05. In our case a probability of 0.9 has been used.

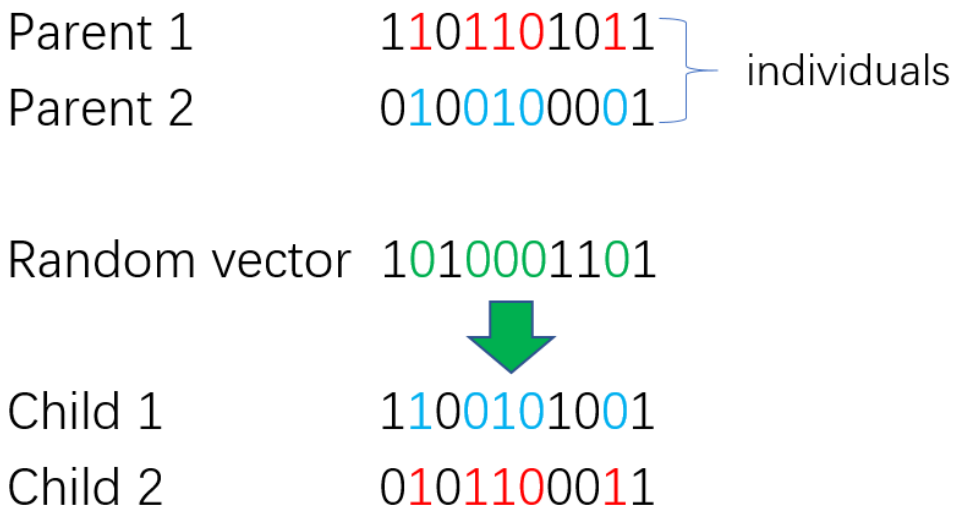


Figure 6.3: The (scattered) two individuals i.e. Parent 1 and Parent 2 crossover operation. Colours show the chromosomes operation based on the bit-value of the random vector.

5. *Mutation:* the mutation options specify how the GA makes small random changes in the individuals within the population to create mutation children. The purpose of the mutation operation is to provide genetic diversity and enable the genetic algorithm to search for a broader space. Mutation attempts to introduce some random alteration of the chromosomes, e.g. 0 becomes 1 (Figure 6.3) and vice versa; this occurs infrequently and therefore mutation is in the order of about one bit changed in a thousand tested, which means 0.1% of the chromosome-bit in the parent undergo mutation.

6.3.2 The Remaining Lifetime Calculation based on the GA trained Critical-path Delay Degradation Trend

After mutation, the GA cycle is completed. In our case, the GA is set to run for a maximum of 100 generations. Meanwhile, the critical-path delay results of our 30 Xentium processors have been used for training purposes to obtain the delay-degradation trend model. Critical-path delay results of the remaining 18 processors are used for the purpose of validation; this is also referred to as the holdout method [Koha 95] for the accuracy evaluation of the predicted model. The optimization result with data is shown

in Figure 6.4, where the generic remaining lifetime for all data (48 Xentium processors) is illustrated.

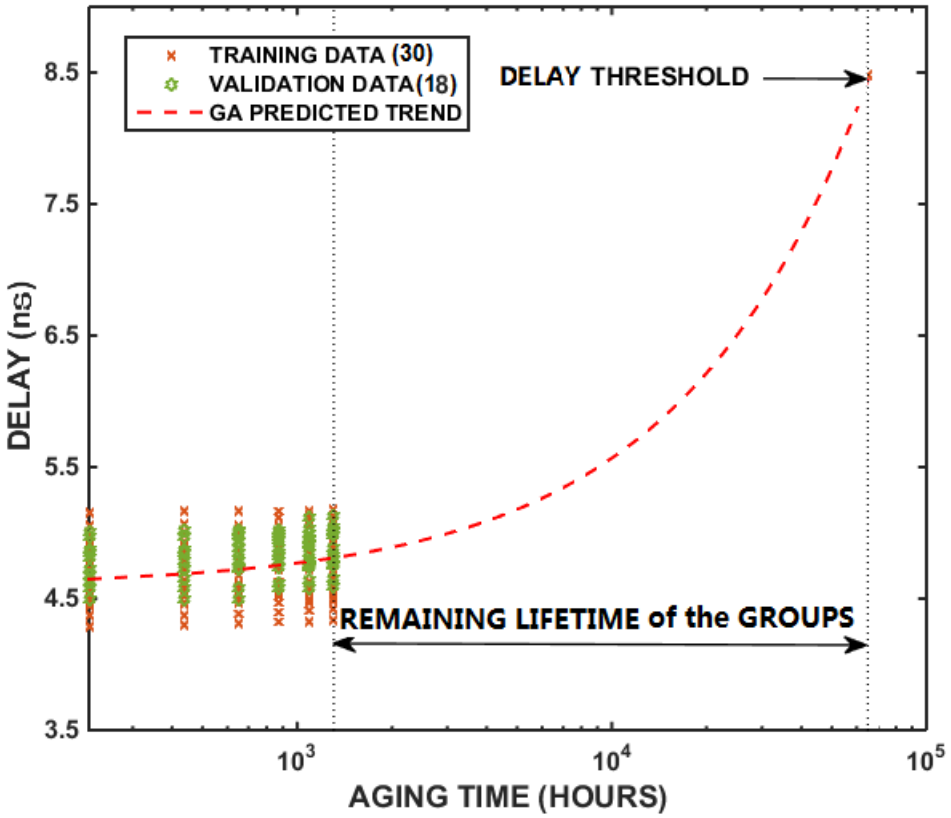


Figure 6.4: Predicted degradation trend of the critical-path delay using real training data (red); validation data is also shown (green). The delay threshold point of 8.5 ns is determined via the static-timing analysis of the netlist after synthesis, the x-axis has a log scale.

The remaining lifetime prediction (RLP) of each Xentium is calculated based on the GA learned trend, the latest critical-path delay measurement result and the critical-path delay-threshold point of 8.5 ns, which is determined by the critical-path delay analysis for the Xentium processor [Reco 11]. The generic learned degradation trend of 30 devices in the training set are shown in the red line in Figure 6.4. The statistical RLP results for both the training set and validation set are provided in Table 6.1.

Table 6.1: Statistical RLP results (*10^4 hours) for the training set, validation set and all devices via delay-testing based on the developed GA model.

Groups	RLP MAX	RLP MIN	RLP MEAN	Standard deviation	Root mean squared error
Training set (30)	8.18	5.54	6.60	0.64	0.54
Validation set (18)	6.84	5.81	6.22	0.43	0.32
Total (48)	8.18	5.54	6.42	0.57	0.52

The sets above exhibit a normal distribution [Pate 96]. The root mean squared error (RMSE) indicates the absolute fit of the model to the data. One can observe that the MAX, MEAN, STD and RMSE in the training set are all larger than that in the validation set, while the MIN is smaller in this case. This is reasonable since the number of devices in the training set is almost twice the size of the training set.

The accuracy (acc) [Koha 95] of the predicted model is defined as the probability of correctly using the validation group and the trained model for predicting the remaining lifetime within the same range as the training group, or written mathematically:

$$\text{acc} = \frac{1}{v} * \sum_{i=1:v} \delta(\varepsilon(vi), MSEt) \qquad \text{Eq. (6.3)}$$

where v denotes the number of devices in the validation group, $\varepsilon(vi)$ is the error of each device calculated from the validation group based on the training model, while MSEt is the mean squared error of the training group. The term $\delta(\varepsilon(vi), MSEt) = 1$ if $\varepsilon(vi) \leq MSEt$ [Koha 95]. In our case, all errors from the validation group were shown to be less than the $MSEt$ from the training group, indicating a full accuracy (100%) of the trained degradation-model verified by each device inside the validation set.

6.3.3 Verification of the Critical-path Delay-based RLP

In order to verify whether the updated life-time prediction result (of one group of Xentiums) can reach a good accuracy, the previous GA-based degradation path has been

employed and applied to our Xentiums, which has shown how the RLP works during actual life-time (application).

Figure 6.5 shows our RLP method. Six times (6 weeks), measurements have been carried out during the HTOL test of 30 Xentium processors, while the RLP results were updated in an interval of three weeks. The value of RLP1 is generated from the first 3 weeks results while RLP2 is the update based on the following 3 weeks results (Figure 6.5).

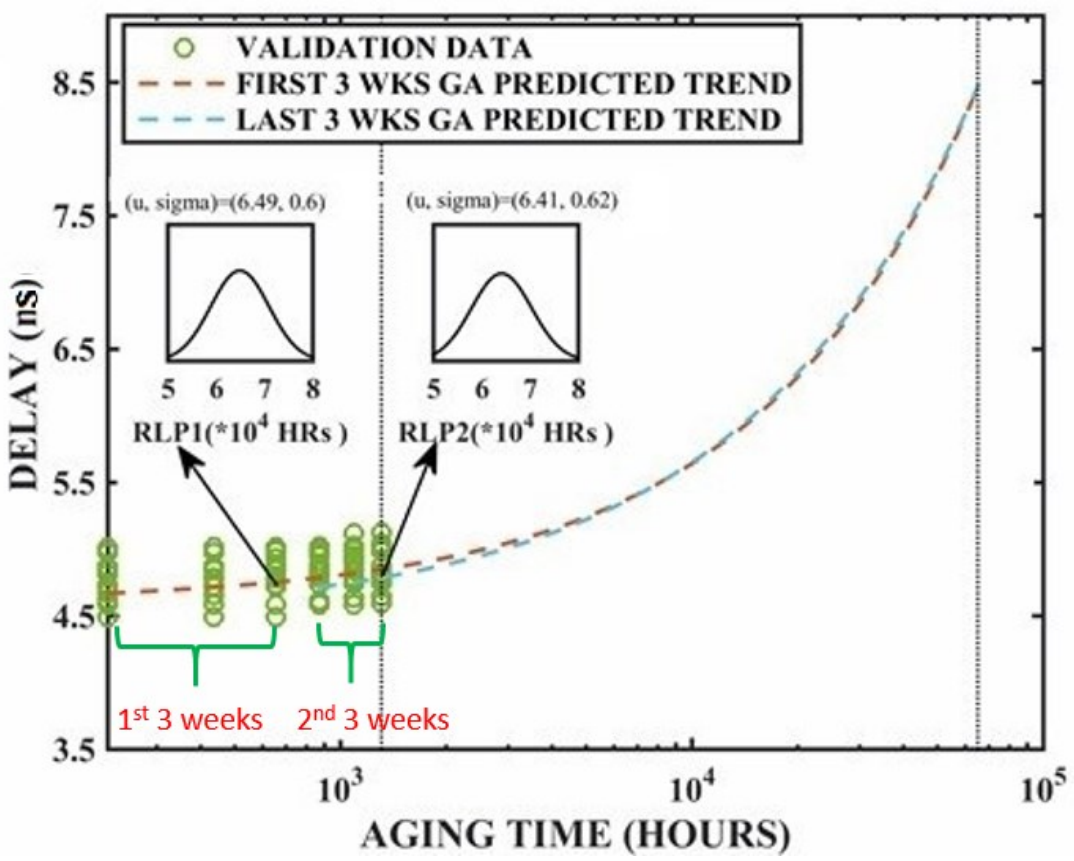


Figure 6.5: Example of RLP results (30 Xentium processors) update with available new measurements of I_{DDX} data; normal distribution of RLP1 (weeks 1-3) and RLP2 (weeks 4-6), with the update interval of 3 weeks.

The GA predicted trends do not change significantly for both cases (blue and orange lines in Figure 6.5). As can be seen from Figure 6.5, the fault-free device after the temperature cycling tests does not violate our previous RLP results (RLP > 0). The most right grey vertical line shows the lifetime on the aging axis after it passes the 8.5 ns clock boundary (faulty processor). Furthermore, the RLP results for these two cases have a mean and standard deviation of (6.49, 0.6)*10^4 and (6.41, 0.62)*10^4 hours separately

(Figure 6.5), indicating that after 3 weeks HTOL-stress time, a decrement of remaining lifetime results of:

$$(6.49 - 6.41) * 10^4 \text{ hours} = 800 \text{ hours} \quad \text{Eq. (6.4)}$$

This result will be compared to our HTOL based lifetime of 3 weeks in the following.

In our HTOL experimental tests (section 5.2), the *temperature* acceleration factor AF_{T} can be constructed via the Arrhenius HTOL model [Esco 06]:

$$AF_{\mathrm{T}} = e^{\frac{E_a}{k}*\left(\frac{1}{T_o} - \frac{1}{T_s}\right)} \quad \text{Eq. (6.5)}$$

where k denotes the Boltzmann constant (8.62×10^{-5} eV/K), T_O is the operating temperature (in degrees Kelvin), T_S is the stress temperature (in degrees Kelvins) and E_a is the activation energy (normally around 0.7 eV) for the respective failure mechanism (e.g. BTI).

The *voltage* acceleration factor AF_{V} can be calculated based on the JEDEC formula [Jede 11]:

$$AF_{\mathrm{V}} = e^{\beta\,(V_S - V_O)} \quad \text{Eq. (6.6)}$$

where β is a constant derived experimentally (normally around 3.2), V_S is the stress supply voltage (1.2 V in our case) and V_O is the operating voltage (1 V in our case).

The overall acceleration factor can now be calculated as:

$$AF = AF_T * AF_V \quad \text{Eq. (6.7)}$$

Our stress temperature has been set to 125 °C (398 °K) and the stress voltage at 1.2 V. From Equations (6.5), (6.6) and (6.7) one can now calculate the total acceleration factor to be 1.44.

Considering the acceleration factor (AF) [Jede 11] in the HTOL test, 3 weeks HTOL stress equals to $3\ weeks * AF = 3 * 7 * 24 \text{ hours} * 1.44 = 725.8 \text{ hours}$. Therefore the error for the GA updated RLP results is $800 - 725.8 = 74.2$ (hours), and hence the error of the updated RLP is $74.2/725.8 = 10.2\%$. This shows the updated RLP results based on the updated GA prediction can reach a good accuracy.

6.4 REMAINING LIFETIME PREDICTION FOR THE XENTIUM VIA ALTERNATIVE I_{DDX} MONITORING

Our previous research [Zhao 15a] proved that I_{DDQ} and I_{DDT} are highly correlated with the critical-path delay according to our measurements. Therefore, we propose a similar GA-based degradation trend optimization approach, using the alternate signature, i.e. the I_{DDX} monitoring results to predict our critical-path delay degradation monitoring. This is because I_{DDQ} or I_{DDT} (on-chip) based health-monitoring costs much less time and efforts to measure as compared to our present critical-path delay degradation monitoring approach (Chapter 4). The remaining lifetime prediction has been calculated based on these I_{DDX} measurement results and their high correlation with critical-path delay data.

6.4.1 GENERAL PROCEDURE

The chosen genetic algorithm is again employed to determine the combination of coefficients (a0, b0, c0 in Figure 6.6) in our delay-degradation model, in order to minimize the degradation trend error for *all* measured Xentium processors via HTOL and temperature-cycling (TC) test results (historic data) [Zhao 16a]. In this procedure, the sample Xentium processors are used for critical-path delay monitor purpose, while the field-usage Xentium processors will only monitor the I_{DDX}. Therefore, the key is to map the measured I_{DDX} of each Xentium processor to the group of field-usage Xentium processors. The steps are briefly described as follows and depicted in Figure 6.6.

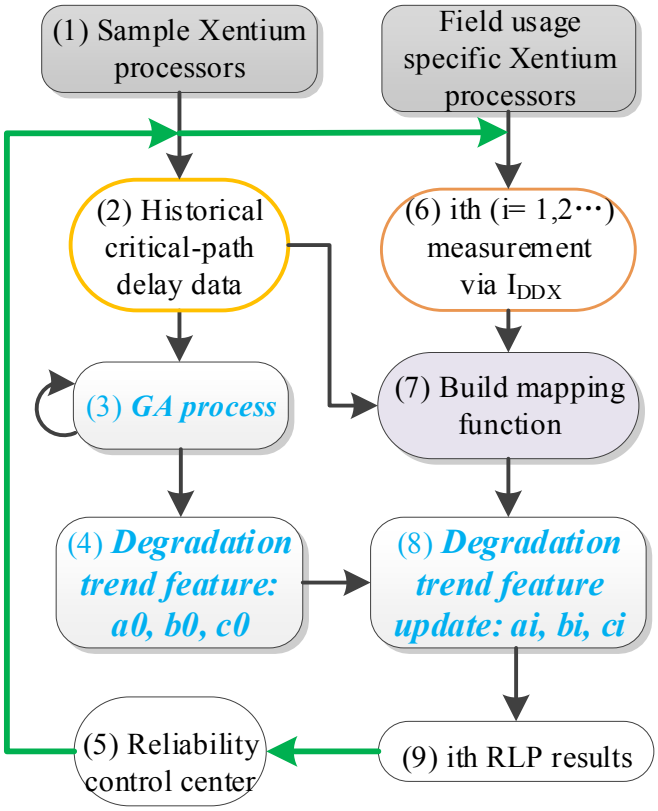


Figure 6.6: Overview of the used lifetime-prediction method via life-testing I_{DDX} testing for a specific Xentium processor in the field.

Step 1): Employ a set of sample Xentium processors (1) to perform the critical-path delay testing, which will provide the general degradation trend. The health-monitoring via (delay-correlated) I_{DDX} testing (6) is employed during life time in each of the field-usage Xentium processors, which will determine whether the processor will be replaced.

Step 2): Based on the historical data (2), apply the GA procedure (3) in the case of the sample Xentium processors, meaning to prepare inputs for the GA-based delay-degradation model f(t). The delay-measurement driven power-law relation is used with respect to the time t of the learning model:

$$f(t) = a + b * t^c \qquad \text{Eq. (6.8)}$$

now subsequently define the individuals (a, b and c) representing the degradation feature, and apply the fitness function, the selection function, cross-over method, etc. From these procedures extract the original parameters a0, b0 and c0 (4) after the training process.

Step 3): For a field-usage specific Xentium processor, new I_{DDX} monitoring data are added as input to our flow. In the cause of time, one can construct the mapping function (7) via historical critical-path delay from the set of sample Xentium processors as well as historic I_{DDX} testing data from field usage ones respectively.

Step 4): Based on (4) and (7) the parameters ai, bi and ci are updated (8), and the remaining lifetime prediction result is evaluated via the ith I_{DDX} tests, based on the mapping function (7) and the GA-learned degradation model (4).

Step 5): After this, the RLP results (9) are transferred to the reliability control center (5) for decisions on potential core replacement.

Step 6): The RLP results based on the new (field-usage) core-specific I_{DDX} measurements are updated by the reliability center. (Repeat from step 3).

The building of the mapping function in Figure 6.6 will be executed via regression techniques, which will be explained in the next section.

6.4.2 Building the Mapping Function between Sample and Field-usage Xentium Cores via Regression Techniques

Our previous research [Zhao 15a] proved that I_{DDQ} and I_{DDT} are highly correlated with the critical-path delay according to our measurements. The absolute values of the correlation coefficients are greater than 0.7 and very close to 1 for all chips.

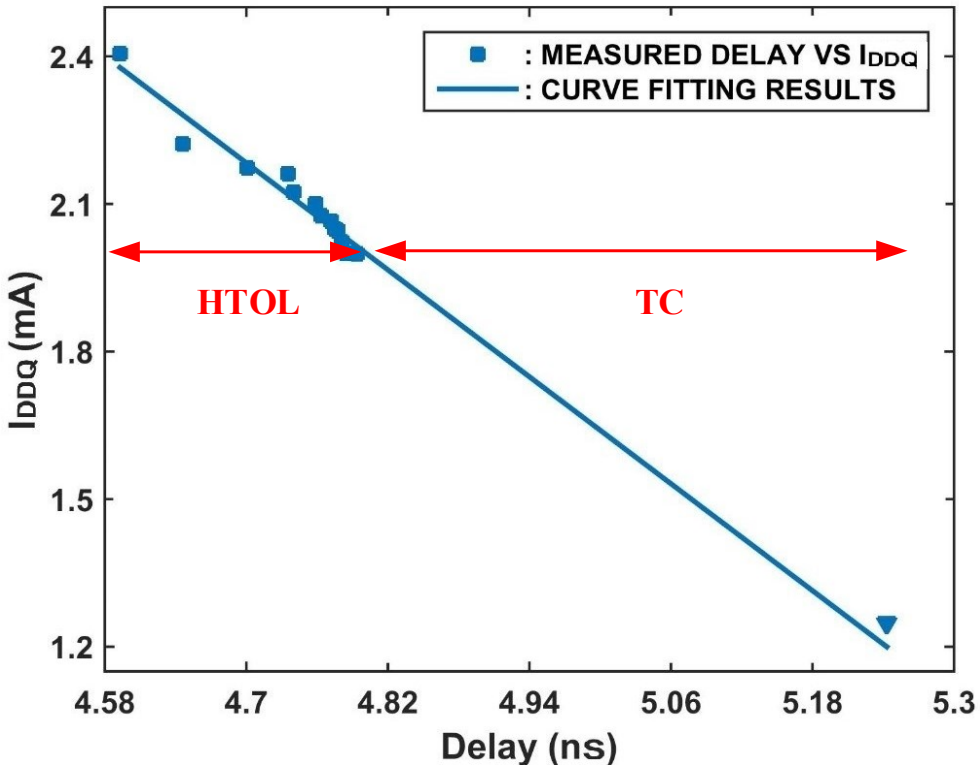


Figure 6.7: Representative measurement results of the I_{DDQ} of a Xentium processor core versus measured critical-path delay. The symbol▼ represents the measured I_{DDQ} versus critical-path delay of the same Xentium after temperature cycling (TC).

The temperature-cycling stress test for the Xentium used 1000 cycles (1000 hours) in addition to the previous 1000-hours HTOL tests (Table 5.1) [Nels 09]. Even for this extended duration, the previous power laws remained valid. This is indicated in Figures 6.7 and 6.8 for I_{DDQ} and I_{DDT} versus critical-path delay respectively. In Figure 6.8 three I_{DDT} results are shown, i.e. the I_{DDT} of three execution units (A, LD and M) of the Xentium processor as explained in Chapter 4.

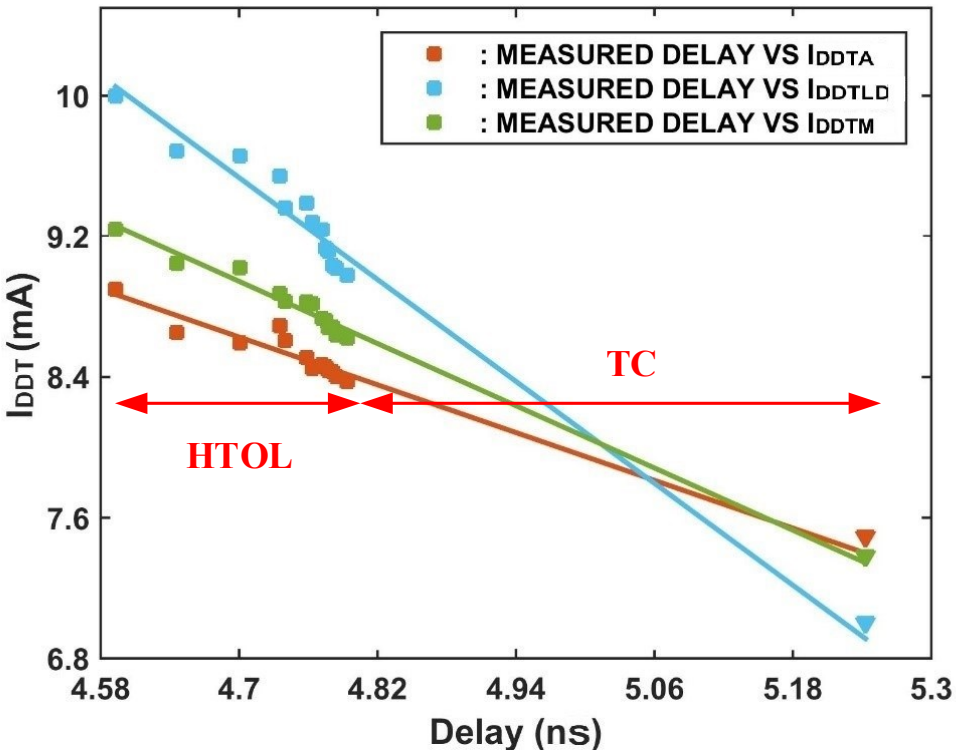


Figure 6.8: Representative measurement results of the I_{DDT} of a Xentium processor core for several internal functional units (being A, LD and M) versus measured critical-path delay. The symbols ▼ represent the measured I_{DDT} versus critical-path delay of the same Xentium after temperature cycling (TC).

Even in the most aged case we had available for the Xentium processors, having a 1000-hours HTOL test with an additional 1000-hours TC test, the devices continued to operate fault-free. This proves the feasibility of our lifetime-prediction calculations.

Because the I_{DDX} measurement results are highly correlated with the critical-path delay as mentioned previously and observed from both Figures 6.7 and 6.8, a linear regression model [Brei 97] can be developed to map the critical-path delay results with a given I_{DDQ} and I_{DDT} value.

This mapping function will determine the relationship between the critical-path delay and the I_{DDX} in our Xentium processor. Therefore, the averaged value of the critical-

path delay and the I_{DDX} of all measured 48 chips are used for building the regression model. Because of the 6-weeks HTOL measurement, 6 points for the critical-path delay and the I_{DDX} are available for constructing the linear regression; this means there are 6 points in the variable of $f(t)$, $I_{DDQ}(t)$, $I_A(t)$, $I_{LD}(t)$, $I_M(t)$, $I_P(t)$, $I_S(t)$ and $I_{ST}(t)$ in the Equations (6.1), (6.9) and (6.10). Therefore, the task is to construct a linear regression between $f(t)$ with respect to $I_{DDQ}(t)$, and a multivariate linear regression [Brei 97] between $f(t)$ with regard to $I_A(t)$, $I_{LD}(t)$, $I_M(t)$, $I_P(t)$, $I_S(t)$ and $I_{ST}(t)$. The resulting mappings via regression are referred to as $g(I_{DDQ})$ and $h(I_{DDT})$ respectively.

With the least mean squared error target, Eq. (6.9) shows the resulting mapping function $g(I_{DDQ})$ between the critical-path delay f(t) and $I_{DDQ}(t)$:

$$f(t) = g(I_{DDQ}) = -0.4356 * I_{DDQ}(t) + 5.5674 \qquad \text{Eq. (6.9)}$$

in which the mean squared error (MSE) is $1.142 * 10^{-5}$, indicating the rooted mean squared error (RMSE) is around 0.21 % of the I_{DDQ} range (1.7 – 3.3 mA, Figure 5.7) or 0.17 % of the critical-path delay range (3.9 – 4.9 ns, Figure 5.6). A strong correlation between the critical-path delay and the I_{DDQ} monitoring results is proven again by the fittings above with the calculated small mean squared errors.

Since I_{DDT} includes a number of I_{DDT} measurements for each execution unit in the Xentium processor (Figure 5.11), denoted by $I_A(t)$, $I_{LD}(t)$, $I_M(t)$, $I_P(t)$, $I_S(t)$ and $I_{ST}(t)$, the mapping function ($h(I_{DDT})$ in Eq. (6.10)) is derived based on a multivariate linear regression via MATLAB [Brow 09]:

$$\begin{aligned} f(t) = h(I_{DDT}) = {} & 6.7598 + 0.036444 * I_A(t) \\ & + 0.1119 * I_{LD}(t) - 0.40886 * I_M(t) \\ & + 0.35631 * I_P(t) - 0.007251 * I_S(t) \\ & - 0.28211 * I_{ST}(t)\,, \end{aligned} \qquad \text{Eq. (6.10)}$$

in which the mean squared error is $1.916 * 10^{-3}$, indicating the rooted mean squared error (RMSE) is around 0.43 % of the whole I_{DDT} range (8 – 10.5 mA, Figure 5.11) or 0.71 % of the critical-path delay range (4.5 - 5.3 ns, Figure 5.6). Similarly, a strong

correlation between the critical-path delay and the I_{DDT} monitoring results is proven again by the fittings above with the calculated small mean squared errors.

Based on the previous mapping functions and the earlier derived GA optimized model (Eq. (6.1)) for each single chip with respect to I_{DDQ}, the RLP is calculated when the mapping function $g(I_{DDQ})$ in Eq. (6.9) reaches the delay-threshold point 8.5 ns [Reco 11]. For I_{DDT}, the calculation is analogous in this case; the RLP is calculated when the mapping function $h(I_{DDT})$ in Eq. (6.10) reaches the delay-threshold point.

6.4.3 The Performance Evaluation of the RLP Results

Based on our measurement data, the statistical RLP results of our 48 chips are given in Table 6.2. The results show that the average I_{DDT}-based RLP value is slightly better than the I_{DDQ}-based one, but both methods have significantly close results. Compared to the remaining lifetime from the critical-path delay test (6.42, 0.57) $*10^4$ hours (Table 6.1), the I_{DDX}-based RLPs perform somewhat better in terms of standard deviation and root mean squared error. Nevertheless, both approaches have a very similar performance.

Table 6.2: Statistical RLP results (*104 hours) via I_{DDX} based on the developed GA model.

Methods	RLP MAX.	RLP MIN.	RLP MEAN	Standard deviation	Root mean squared error
I_{DDQ}	7.26	5.87	6.47	0.33	0.33
I_{DDT}	7.92	5.47	6.48	0.49	0.5

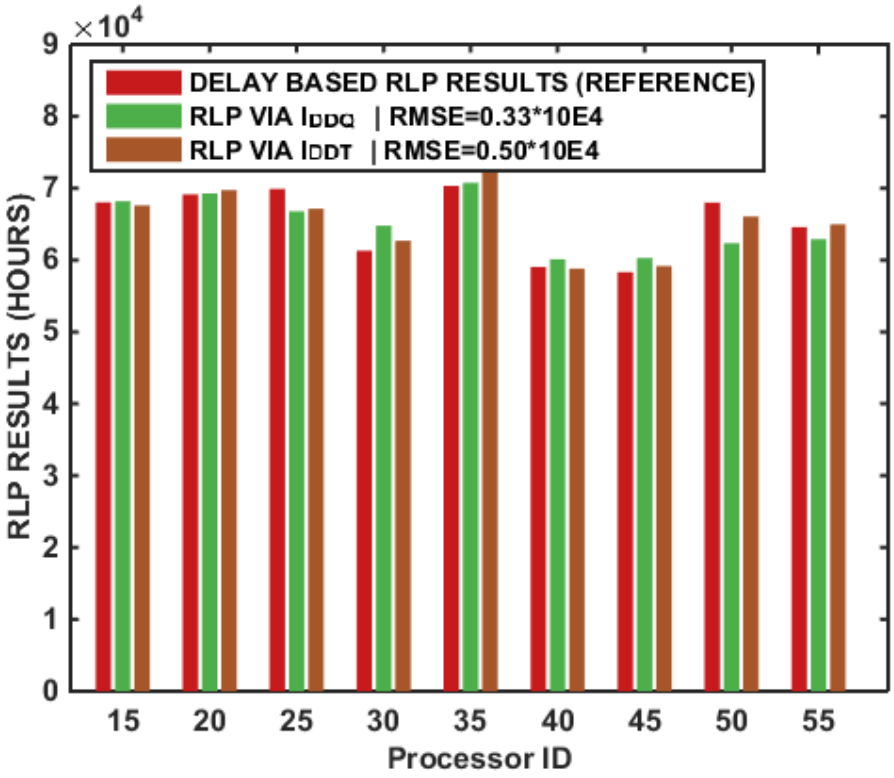


Figure 6.9: Remaining lifetime predicted via I_{DDX} for 9 randomly chosen processors from the GA model.

Figure 6.9 shows the remaining lifetime bar of 9 randomly chosen Xentium processors with the RLP results from the I_{DDQ} and I_{DDT} test data. Again, the bars show that the RLP based on I_{DDQ} or I_{DDT} testing are very similar to the one based on the critical-path delay testing. These results enable the calculation of the RLP of chips based on I_{DDQ} or I_{DDT} (on-chip) health-monitoring testing that costs much less time and efforts to measure as compared to the in this thesis used critical-path delay testing (Chapter 4).

6.5 CONCLUSIONS

In this chapter, reliability-testing experiments with 1000 hours of HTOL and 1000 hours of TC stress tests for 48 chips have been completed, with measurements carried out on the critical-path delay, I_{DDQ} and I_{DDT} currents of the Xentium processor, part of our MP-SoC.

The genetic-algorithm based degradation model optimization has been proposed. In order to validate the accuracy of our prediction, our chips have been divided into a training set and validation set. It has been found that the validation set for predicting the remaining lifetime is within the same range as the training set, demonstrating a full accuracy for our trained degradation-model.

In addition, an alternative remaining lifetime prediction based on the I_{DDX} monitoring result for the 90 nm VLIW Xentium processor has been developed. This is based on the strong correlation found between the critical-path delay and the I_{DDX} monitoring results, which is proven by building a mapping function with small mean squared errors. This is accomplished by the least squared error-based regression approach for deriving the required parameters of the mapping functions.

Based on the developed mapping function, it was shown that the remaining lifetimes predicted by the I_{DDQ} and I_{DDT} have a mean value and standard deviation of (6.47, 0.33) $*10^4$ hours and (6.48, 0.49) $*10^4$ hours respectively, compared to (6.42, 0.57) $*10^4$ hours in the critical-path delay testing. This proves that predicting the remaining lifetime with I_{DDX} can reach a good accuracy, being an alternative of using the critical-path delay for measurements, thereby reducing the measurement time and efforts compared to the used critical-path delay testing. Based on our RLP results, the dependability of multi-processor SoCs can be significantly increased via timely counteractions. For the future, also data-fusion of HM measurements could be considered, thereby potentially improving the life-time prediction.

REFERENCES

[Blic 96] T. Blickle, L. Thiele, "A Comparison of Selection Schemes Used in Evolutionary Algorithms," In Evolutionary Computation, pp. 361–394, ISSN 1063-6560, 1996.

[Brow 09] S. H. Brown, "Multiple linear regression analysis: a matrix approach with MATLAB," in Alabama Journal of Mathematics, vol 34, pp 1-3, 2009.

[Brei 97] L. Breiman, and J. H. Friedman, "Predicting multivariate responses in multiple linear regression," in Journal of the Royal Statistical Society: Series B (Statistical Methodology), vol 59, pp. 3-54, 1997.

[Brot 00] T. Brotherton, G. Jahns, J. Jacobs, and D. Wroblewski, "Prognosis of faults in gas turbine engines," in IEEE Aerospace Conference, Big Sky, MT, pp. 163-171, vol.6, 2000.

[Esco 06] L. A. Escobar and W. Q. Meeker, "A review of accelerated test models," in Statistical Science, pp. 552-577, 2006.

[Gold 88] D. E. Goldberg and J. H. Holland, "Genetic Algorithms and Machine Learning," in Machine Learning, vol. 3, pp. 95-99, October 1988.

[Holl 92] J. H. Holland, "Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence," MIT press, ISBN: 0262581116, 1992.

[Jede 11] "JEDEC standard JESD22-A105C," http://www.jedec.org/ standardsdocuments /January 2011.

[Koha 95] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in Proceedings of the 14th international joint conference on Artificial intelligence, pp. 1137-1145, 1995.

[Kwon 06] C. Kwon and S. D. Sudhoff, "Genetic algorithm-based induction machine characterization procedure with application to maximum torque per amp control," in IEEE Transactions on Energy Conversion, vol. 21, pp. 405-415, 2006.

[Mitc 98] M. Mitchell, "An Introduction to Genetic Algorithms," MIT Press, ISBN:0262631857, 1998.

[Nels 09] W. B. Nelson, "Accelerated testing: statistical models, test plans, and data analysis," John Wiley & Sons, ISBN: 0471697362, 2009.

[Pate 96] J. K. Patel and C. B. Read, "Handbook of the normal distribution, second edition," CRC Press, ISBN 9780824793425, 1996.

[Reco 11] RecoreSystems. http://www.recoresystems.com, 2011.

[Sonm 15] A. Sonmez, E. Kocyigit, and E. Kugu, "Optimal path planning for UAVs using Genetic Algorithm," in International Conference on Unmanned Aircraft Systems (ICUAS), pp. 50-55, 2015.

[Spal 03] J. C. Spall, "Introduction to stochastic search and optimization," John Wiley & Sons, Inc., Hoboken, New Jersy, ISBN:9780471330523, 2003.

[Tunc 12] A. Tuncer and M. Yildirim, "Dynamic path planning of mobile robots with improved genetic algorithm," in Computers & Electrical Engineering, vol. 38, pp. 1564-1572, 2012.

[Yanj 09] L. Yanjing, K. Young Moon, E. Mintarno, et al., "Overcoming Early-Life Failure and Aging for Robust Systems," in IEEE Design & Test of Computers, vol. 26, pp. 28-39, 2009.

[YuDo 11] L. Yu-Dong, L. Hong-Wei, E. Yun-Fei and W. Ming, "Pre-designed prognostic cells for host circuits reliability monitoring," in International Conference on Electric Information and Control Engineering (ICEICE), pp. 1550-1552, 2011.

[Zhao 15a] Y. Zhao and H. G. Kerkhoff, "Predicting aging caused delay degradation with alternative I_{DDT} testing in a VLIW processor," in Proceedings of the Workshop on Manufacturable and Dependable Multicore Architectures at Nanoscale, MEDIAN, Tallinn, Estonia, pp. 27-32, 2015. https://pdfs.semanticscholar.org/a016/0e9ae327fb90b17f4d3f79dbc2ad14475db6.pdf

[Zhao 15b] Y. Zhao and H. G. Kerkhoff, "Application of functional I_{DDQ} testing in a VLIW processor towards detection of aging degradation," in International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), Naples, Italy, pp. 1-5, 2015.

[Zhao 15c] Y. Zhao and H. G. Kerkhoff, "Unit-Based Functional I_{DDT} Testing for Aging Degradation Monitoring in a VLIW Processor," in the Euromicro Conference on Digital System Design (DSD), Funchal, Portugal, pp. 353-358, 2015.

[Zhao 16a] Y. Zhao and H. G. Kerkhoff, "Highly Dependable Multi-processor SoCs Employing Lifetime Prediction Based on Health Monitors," in Proceedings of IEEE 25th Asian Test Symposium (ATS), Taipei, Taiwan, pp. 228-233, 2016.

[Zhao 16b] Y. Zhao, H. G. Kerkhoff, "A genetic algorithm based remaining lifetime prediction for a VLIW processor employing path delay and I_{DDX} testing," in International Conference on Design and Technology of Integrated Systems in Nanoscale Era (DTIS), Istanbul, Turkey, pp. 10-14, 2016.

Chapter 7

CONCLUSIONS, CONTRIBUTIONS AND RECOMMENDATIONS

7.1 INTRODUCTION

In this chapter, the final conclusions and recommended future work of our research is presented based on the results of the previous chapters. First, the general conclusions are summarized in section 7.2. The main contributions of all research provided in this thesis are described and subsequently summarized in section 7.3. Limitations of the current research and some possible directions for a continuation of our research, are discussed in section 7.4. Our research-related publications are presented in section 7.5.

7.2 GENERAL CONCLUSIONS

This thesis addresses dependability enhancement techniques developed for a new generation MP-SoCs, which are implemented for application in safety-critical areas. The key idea is to be able to carry out pro-active system maintenance / repair by employing health-monitoring techniques. Based on the health-monitoring results, a lifetime prediction approach has been developed to determine the preventive maintenance moment of the target MP-SoCs, in order to keep full functionality at all time. This is the *proactive* approach for repair, which is different from traditional dependability solutions, where typically the fault-detection based testing method is used, leaving the system with unavailable service time during the *reactive* repair.

The objective of this study is the development of dependability-enhancement techniques for widely used modern MP-SoCs for safety-critical applications. The proposed health-monitoring technique reaches the stated goal, and is validated through accelerated testing of our test chip, a Xentium-based MP-SoC, and the statistical analysis of the measurement results.

The life-time prediction (LTP) technique is built on the statistical analysis of a vast amount of multi-dimensional health-monitoring data, subsequent statistical analysis and an advanced machine-learning algorithm (genetic algorithms). A GA-based degradation path has been optimized and was validated. Based on this path, the updated life-time prediction results can reach a good accuracy (10.2% error), as has been found in the field at a 3-weeks measurement interval. Afterwards, an alternative remaining lifetime prediction method based on the I_{DDX} monitoring result for the Xentium has been validated

and developed. This indicated that more health-monitoring information can be fed into our model, resulting in a potentially more accurate prediction result.

In conclusion, all research questions raised at the beginning of this thesis have been answered. The design and implementation of the health-monitoring and life-time prediction technique has proven to be successful. Our proposed dependability enhancement approach has proven to be feasible.

Note that these techniques can be incorporated into any generic processors with no or little change in the architecture of the software programs like e.g. for the critical-path delay and the I_{DDX}. Meanwhile, it is compatible for usage in the previous, current and the next generation process-technology SoCs because of their independence from process technologies. This makes it easier to import these monitoring and life-time prediction techniques from a previous technology to the current or next generations.

7.3 MAJOR CONTRIBUTIONS

The major contributions of the research in this thesis are summarized in the following sections. The techniques developed in the research are solutions to the nowadays widely applied safety-critical systems with a high dependability, with the special requirement that the system downtime should be equal or close to zero. The applications of the proposed techniques are for mixed-signal System-on-Chips, employing a number of generic digital processor cores. This means that our all-in-one embedded health-monitoring infrastructural IP, as well as the software-based health monitoring techniques and the I_{DDX} monitoring-based remaining life-time prediction method are not only linked to our Xentium-based MP-SoC, but can be applied to other MP-SoCs as well. The contributions of this thesis are explained in the following subsections.

7.3.1 Design for Dependability of Our Many-Processor SoC

Based on the analysis of application areas of our new generation MP-SoCs, as well as the requirement of a more powerful data-processing capability, new techniques for enhanced dependability of MP-SoCs have been introduced.

In the field operation for MP-SoC, our approach is based on monitored health information, with regard to the development of degradation near or in a core. After one obtains the crucial health-monitoring info regarding the degradation of the system, the remaining lifetime prediction can be determined. It will be calculated from the current moment until the moment the health-monitoring signal reaches a preset failure threshold. At some point based on the remaining lifetime-prediction result, a general-purpose control processor (e.g. an ARM) in the MP-SoC will determine to take a repair action for the target monitored core via remapping. As an alternative, cores can also be freed from low-priority tasks, and subsequently added to the pool of spare core resources. Lowering local core clock frequencies or power-supplies also provide possibilities. This enables preventive maintenance. It guarantees a high dependability and 100% availability of our MP-SoC.

7.3.2 An Embedded Health-Monitoring Infrastructural IP

In order to obtain the most accurate prediction of the remaining lifetime for a core in our MP-SoC, it is crucial to identify the monitoring parameters that capture the key present health status, and relates it to the aging behaviour of the SoC.

In our research, based on the underlying aging mechanisms, in Chapter 3 an all-in-one health monitoring infrastructure circuit has been designed which is capable of carrying out voltage and temperature measurements as well as non-invasive reliability monitoring (via delay-time monitoring). The data from the temperature monitor can help to detect temperature increases due to developing faults in the core. The delay monitors, in combination with voltage monitors, can show the system (frequency) reliability degradation caused by aging. Simulation results show that the monitor exhibits a relatively stabilized sensitivity to the changing parameters (temperature, voltage and NBTI aging). Due to their relatively small size, simple digital outputs and low-power characteristics, they can be used extensively in the standard cell-based circuits in our MP-SoC.

The health-monitoring data supplied by our monitor will be used for maintenance actions afterwards, via the IJTAG (IEEE 1687) standard node with some adaptations. The monitor communicates with the embedded control processor (ARM), and thus the health information can be processed during field operation.

7.3.3 A SOFTWARE-BASED HEALTH MONITORING TECHNIQUE FOR THE AGING DEGRADATION DETECTION

Compared to hardware embedded health monitors introducing extra area cost, and the aging monitoring of itself instead of target processors, our proposed software-based health monitoring (software-based self-test, SBST) technique in Chapter 4 not only avoids the hardware penalty, but also increases the monitoring accuracy by monitoring the actual devices in the processor directly via developed software programs.

Our technique includes suitable monitoring programs executed by our target processor. It can be employed to provide real-time health data with regard to the aging status of the target processor; this data can be used for prognostic purposes. Parameters such as critical-path delay, quiescent and transient power-supply currents are monitored.

One advantage of our designed critical-path delay-monitoring technique is the at-speed measurement, overcoming the difficulty of hard to test high-speed circuits with traditional delay-testing techniques. Another advantage is that our approach only acts on functional inputs and monitoring functional outputs, without usage of (internal) DfT structures. In particular, our software-based delay monitoring may be used in those cases where at-speed testing is crucial, or DfT cannot be used.

Our designed software based I_{DDQ} and I_{DDT} monitoring can be used during life-time usage. The difference between these two is that I_{DDQ} monitoring is meant to observe the quiescent current in the case the processor enters a static state, while the I_{DDT} monitoring observes the dynamic current of each execution unit inside the datapath.

In I_{DDT} monitoring, the designed I_{DDT} functional program is focused on the key elements of the Xentium, i.e. monitoring the dynamic current of each (10) execution units inside the datapath of the Xentium, which is referred to as the unit-based I_{DDT} monitoring.

Generally, these parameters and the software-based programs can be used for the health-monitoring of other types of processors as well.

7.3.4 ACCELERATED TESTING SYSTEM DESIGN AND MEASUREMENT IMPLEMENTATION

In order to prove the feasibility of our designed software-based health monitoring techniques, a proper reliability test plan, being accelerated testing (AT), has been conducted for Xentium-based MP-SoCs in Chapter 5. This will reduce the normal-life aging time of the Xentium processor. The stress concerns the operational temperature, processor core power-supply as well as the processor workload, according to the JEDEC standard.

One Arduino-Xentium-FPGA based reliability test system has been designed, including a cold Arduino-based board for the control of the test, and a hot Xentium-based DUT board, as well as an off-the-shell FPGA board for the collection of the (currents) test results.

The AT has been carried out for 48 Xentium-based MP-SoCs for 1000-hours HTOL and 1000-hours TC. The health monitoring of the critical-path delay, quiescent current and transient current of the Xentium have been measured at regular time intervals during the reliability tests. We found that critical-path delay changes, I_{DDQ} changes and I_{DDT} changes in each functional unit of the processor have a power-law trend. Meanwhile, strong correlation coefficients have been observed between the results from these three monitoring techniques. This proves the $I_{DDQ/T}$ monitoring can be an alternative candidate for aging-caused delay-degradation detection.

7.3.5 THE I_{DDX} MONITORING-BASED REMAINING LIFE-TIME PREDICTION

The health-monitoring data of the Xentium is multi-dimensional (multi-chips and multi-monitoring parameters for each chip). However, in the real implementation of our technique, current monitoring is much preferred for its time efficiency and easy-measurement features as compared to the used critical-path delay monitoring. An alternative technique of the remaining lifetime prediction for the Xentium via I_{DDX} monitoring has been proposed in Chapter 6.

In the first place, generic algorithms (GA)-based degradation trend optimization algorithms have been proposed based on the critical-path delay monitoring results. The remaining lifetime was calculated based on the GA-predicted degradation trend and a

committed failure-threshold point. The chips have been divided into a training set and validation set. It has been found that the validation set for predicting the remaining lifetime is within the same range as the training set, demonstrating the full accuracy of our trained degradation-model.

Afterwards, based on the degradation trend learned by the GA process from the critical-path delay data, and the mapping function via regression analysis, an algorithm for estimating the remaining lifetime using the I_{DDX} monitoring results was proposed. A quite similar mean with a small standard deviation have been found between the critical-path delay monitoring and I_{DDX} monitoring results, indicating that predicting the remaining lifetime with I_{DDX} health-monitoring can reach a good accuracy.

7.4 FUTURE WORK AND RECOMMENDATIONS

The research work described in this thesis can be extended in several ways. First, the all-in-one embedded monitor can be used after the manufacturing phase to detect faults in the infant period of the target ICs. In addition, besides the proposed monitoring parameters it is recommended to monitor more interesting ones, such as a workload monitor. This will require new infrastructural designs.

Second, applying the developed software-based monitoring approach in other dependable processors would be interesting. Moreover, the I_{DDT} monitoring of more crucial parts inside the target multicore processors, e.g. inter-core communication part, shared memory parts and I/Os can be considered as additional topics for future work.

Finally, more parameters can be introduced in our remaining life-time prediction algorithm, like e.g. slack-delay monitors. Also a new prediction algorithm can be interesting, for instance, a data-fusion based machine-learning approach for all multi-dimensional health-monitoring data, and subsequent feature extraction based on this for building the degradation path. The life-time prediction based on these extensions could be a promising direction.

LIST OF OWN PUBLICATIONS

[Zhao 13a] Y. Zhao, X. Zhang and H. G. Kerkhoff, "Power-dissipation comparison of two dependability approaches for multi-processor systems," in 8th International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), pp. 56-61, 2013.

[Zhao 13b] Y. Zhao and H. G. Kerkhoff, "An embedded health-monitoring infrastructure for a reliable multi-core processor," in Proceedings of the Second Workshop on Manufacturable and Dependable Multicore Architectures at Nanoscale (MEDIAN), pp. 31-34, 2013.

[Zhao 14] Y. Zhao and H. G. Kerkhoff, "Design of an embedded health monitoring infrastructure for accessing multi-processor SoC degradation," in 17th Euromicro Conference on Digital System Design (DSD), pp. 154-160, 2014.

[Zhao 15a] Y. Zhao and H. G. Kerkhoff, "Unit-based functional I_{DDT} testing for aging degradation monitoring in a VLIW processor," in 18th Euromicro Conference on Digital Systems Design (DSD), pp. 353-358, 2015.

[Zhao 15b] Y. Zhao and H. G. Kerkhoff, "Predicting aging caused delay degradation with alternative I_{DDT} testing in a VLIW processor," in Proceedings of the final Workshop on Manufacturable and Dependable Multicore Architectures at Nanoscale (MEDIAN), pp. 27-32, 2015.

[Zhao 15c] Y. Zhao and H. G. Kerkhoff, "Application of functional I_{DDQ} testing in a VLIW processor towards detection of aging degradation," in 10th International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), pp. 1-5, 2015.

[Zhao 16a] Y. Zhao and H. G. Kerkhoff, "Highly dependable multi-processor SoCs employing lifetime prediction based on health monitors," in IEEE 25th Asian Test Symposium (ATS), pp. 228-233, 2016.

[Zhao 16b] Y. Zhao and H. G. Kerkhoff, "A genetic algorithm based remaining lifetime prediction for a VLIW processor employing path delay and I_{DDX}

testing,” in International Conference on Design and Technology of Integrated Systems in Nanoscale Era (DTIS), pp. 10-14, 2016.

[Kerk 12a] H. G. Kerkhoff and Y. Zhao, “The design of dependable flexible multi-sensory System-on-Chips for security applications,” in 15th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS), pp. 133-138, 2012.

[Kerk 12b] H. G. Kerkhoff, J. Wan and Y. Zhao, “Hierarchical modeling of automotive sensor front-ends for structural diagnosis of aging faults,” in 18th International Mixed-Signals, Sensors and Systems Test Workshop (IMS3TW), pp. 91-96, 2012.

[Kerk 14] H. G. Kerkhoff, J. Wan and Y. Zhao, "Linking aging measurements of health-monitors and specifications for multi-processor SoCs," in 9th IEEE International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), Santorini, pp. 1-6, 2014.