A Genetic Algorithm Based Remaining Lifetime Prediction for a VLIW Processor Employing Path Delay and I$_{DDX}$ Testing

Yong Zhao and Hans G. Kerkhoff

Testable Design and Test of Integrated Systems (TDT) Group
University of Twente, Centre for Telematics and Information Technology (CTIT)
Enschede, the Netherlands
yong.zhao@utwente.nl

Abstract—**In this paper, critical path-delay time, and quiescent and transient power-supply current testing have been applied to a 90nm VLIW processor, to predict the remaining lifetime of this processor. The test environment for validation, via implementing an accelerated test has been realized. The resulting delay and current measurement data is presented next, followed by applying a genetic algorithm (GA) to construct a lifetime prediction model for the processor. The calculated remaining lifetime predicted by power-supply testing is close to that of delay-time testing.**

Keywords—**remaining lifetime, failure prediction, delay testing, reliability testing, IDDQ and IDDT testing, DSP processor, aging**

I. INTRODUCTION

In recent years, with the scaling of the silicon technology for transistors into the sub-micron region, aging has been recognized as a significant phenomenon. Transistor aging under mechanisms e.g. Negative-Bias-Temperature-Instability (NBTI), Hot-Carrier Injection (HCI) and Time-Dependent Dielectric Breakdown (TDDB) will result in the deterioration of circuit performance over its lifetime.

In previous research we proposed techniques in terms of health monitoring of a target processor for parameters such as e.g. the critical path delay [1], quiescent power-supply current (I$_{DDQ}$) [2] and transient power-supply current (I$_{DDT}$) [3]. These testing techniques have demonstrated their effectiveness in assessing aging causing reliability degradation. This work aims to develop methods that use this health information to predict the remaining lifetime of the processor. The genetic algorithm (GA) [4] is employed for the prediction purpose. The GA is used for general optimization problems with a multimodal target; this paper will model the functional form of the aging degradation based on the GA method, then making a lifetime prediction of our target processor.

The remainder of this paper is organized as follows: Section II discusses our implemented reliability testing system. Section III presents the measurement results and basic analysis. Section IV illustrates the GA-based method to accomplish the lifetime prediction from the delay and I$_{DDX}$ test results. Finally, conclusions are provided.

II. IMPLEMENTED RELIABILITY TESTING SYSTEM

A. The Xentium processor

The reliability tests are carried out using the Xentium® processor as a vehicle, which is an UMC 90nm CMOS technology Very Large Instruction Word (VLIW) DSP processor [5]. It has been designed for high-performance computing in automotive as well as space applications, e.g. beam forming and a global navigation satellite system [6].

1) Architecture

The basic architecture of the Xentium core includes a datapath, a decoder/loop buffer part, an instruction cache, a control part, tightly coupled memories and interfaces. The datapath comprises of ten execution units and five register banks. Each execution unit is responsible for a certain class of instructions [5]. The Xentium can communicate to the peripherals via the Network Interface that directly links to the Xentium data/instruction ports.

2) Test program design

The critical path delay time (directly related to the clock frequency), I$_{DDQ}$ and I$_{DDT}$ testing are designed for monitoring the crucial part of the Xentium processor, the datapath, which is the most frequently used part during data processing. These testing methods can be used for the aging degradation detection of the Xentium processor [2].

a) Critical path delay testing program design

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 functional tests, at the lowest available clock frequency (4 MHz). Under this condition, it will be verified whether the Xentium still operates correctly at the typical V$_{DD}$ supply-voltage level.

If the test does not fail, the phase-locked loop (PLL) will feed a higher (max. 246 MHz) clock frequency, until the Xentium fails. If even at the highest clock frequency it still is operational, the V$_{DD}$ of the Xentium can be decreased to enhance the chance of failure. The maximum clock frequency will be measured if the Xentium fails at certain operation voltage, and the critical path delay will be the inverse value of it.

b) *I_{DDX} testing program design*

The general steps to initialize both I_{DDQ} and I_{DDT} testing are the same as in the case of delay testing. Differences are firstly the functional program design, and secondly in the I_{DDT} testing, the Xentium operating frequency is set to 31.25MHz, which is directly linked to the sampling specifications of the QT1411 I_{DDT} current monitor [7].

In the case of I_{DDQ} testing, when the mailbox is empty, the Xentium clock will be shut off to stop synchronization. The Xentium processor will then enter the wait state, which is required for the I_{DDQ} measurement [3].

In the I_{DDT} testing, the essential part is that based on the architecture of the Xentium processor, it has to show the current signature of all units in the datapath, i.e. the A, M, S, P, ST and LD units. Different units run in series instead of parallel in order to be able to monitor their separate power currents during the test [2].

B. Accelerated testing and the measurement setup

Periodic delay and I_{DDT} measurements of the Xentium processor with regard to aging degradation have been carried out. An *accelerated High Temperature Operating Life (HTOL) test* has been implemented of a duration of 1000 hours in order to comply to the JEDEC standards [8]. 46 processors have been stressed at a temperature of 125 ℃, a power supply of 1.1 V. The basic setup [2] of our accelerated aging system is shown in Figure 1. The driver board connects via USB to a PC, on which our dedicated software runs. On the HTOL board, there are 3 Xentium processors/DUTs (devices under test), with the connection wires from/to the driver board.

Figure 1. Actual implementation of the accelerated testing system for the Xentium, the cold-zone driver board (left), edge connector (middle) and hot-zone HTOL test board (right) with three DUTs can be recognized. (Courtesy of Maser Engineering).

At a 1-week (167 hours) interval, the stressed (temperature, power-supply, clock) HTOL boards have been removed from the oven. The delay and I_{DDX} measurements for the Xentium processor have been carried out after each interval.

III. MEASUREMENT RESULTS AND ANALYSIS

A. Path delay and I_{DDX} testing results

Based on our developed path delay and I_{DDX} testing program, the critical path delay and power current during the test run have been measured. The path delay measurement results are shown in Figure 2a, where the critical path delay of 46 processors in the case of fresh (non-aged) and 6 different aging times are presented; Figure 2b shows the I_{DDQ} values of all processors as compared to the initial fresh state. The I_{DDT} measurement results

is similar to the I_{DDQ}. Difference is that there are 6 units' current in the I_{DDT} testing [3]. The results are not shown here due to the space limitation in this paper. It can be found all test results show the degradation behaviour. The delay results increase while the current results decrease with respect to aging time.
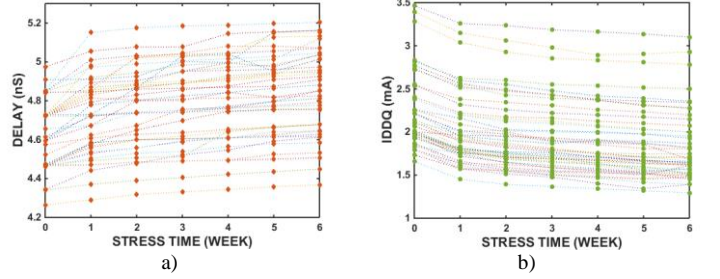

Figure 2. Measurement results of a) the critical path delay, b) I_{DDQ} of 46 Xentium processors. Stress times are on the horizontal axis.

B. Mean lifetime calculation via the HTOL model

In the HTOL model, the equivalent lifetime for the processor can be described as [9]:

$$ELT = AF * D * H \qquad (1)$$

where D denotes the number of processors tested, H is the number of testing hours per processor and AF is the acceleration factor. In our HTOL test, a high temperature and voltage stress have been applied. Therefore, the acceleration factor for the processor includes both the *temperature* and *voltage* acceleration factor.

The *temperature* acceleration factor can be constructed via the Arrhenius HTOL Model [9]:

$$AF_{T} = e^{\frac{E_a}{k}*\left(\frac{1}{T_O}-\frac{1}{T_S}\right)} \qquad (2)$$

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

The *voltage* acceleration factor is calculated based on JEDEC formulas [8]:

$$AF_{V} = e^{\beta \langle V_S - V_O \rangle} \qquad (3)$$

where β is a constant derived experimentally (normally 3.2), V_S is the stress voltage (1.1 volt in our case) and V_O is the operating voltage (1volt in our case).

The overall acceleration factor is now calculated to be:

$$AF = AF_T * AF_V \qquad (4)$$

In our HTOL test, the stress temperature is 393ºK and the stress voltage is 1.1 V. Since our device are target for a harsh environment application, we will consider the operating temperature as 120 ºC (393ºK) while the operating voltage is 1.1V. From Equation (2), (3) and (4) one can calculate the total acceleration factor to be 1.30.

In the HTOL test, the number (r) of failure processor(s) follows the probability function of the Chi-squared (χ^2) distribution [10].

$$r \sim \frac{\chi^2(\alpha, \upsilon)}{2} , \qquad (5)$$

where χ^2 /2 (Chi-squared/2) denotes the probability estimation for the number of failed processor(s). The parameter α, the confidence level (CL) or probability, is the applicable area under the χ^2 probability distribution curve; for calculating the reliability one normally uses α = 0.6 (or 60%). The parameter v, being the degree of freedom (DF), determines the shape of the χ^2 curve; normally reliability calculations often use v = 2r + 2. After the HTOL test, no failed processor was found, and hence the χ^2 therefore equals to 1.832 according to the χ^2 constant table. The mean lifetime equals to the total testing hours by χ^2/2 (replacement for the number of failed processors):

$$Mean\ lifetime = \frac{ELT}{r} \sim \frac{2 * ELT}{\chi^2(\alpha, v)}$$
$$= \frac{2 * AF * D * H}{\chi^2(\alpha, v)} \tag{6}$$

Since there are in total 46 chips tested with 1000 hours HTOL stress time by us, the mean lifetime under the normal operating condition calculated from Equation (6) is $6.52*10^4$ hours (7.44 years). The mean remaining lifetime after the HTOL test for the chips will be the subtraction from the whole lifetime to the aged time, which is $6.38*10^4$ hours

IV. THE REMAINING LIFETIME PREDICTION BASED ON GENETIC ALGORITHMS

A. Overview of data-driven based lifetime prediction

Basically, estimation of the remaining lifetime for a target device is based on collected history data and its current status. Target chips will eventually fail if health monitor parameters just reach a specified failure threshold. From the current status time t(i) up to the failure time T is referred to as the remaining lifetime. Therefore the crucial issue is to build one *degradation trend/model* based on known history data.

B. The GA procedure for degradation-trend optimization for path delay

A genetic algorithm (GA) [4] is an optimization technique modelled after natural evolution. A GA maintains a population of candidate solutions, called chromosomes, which are typically encoded as strings. The evolutionary process starts from a population of randomly generated individuals and uses successive iterations of selection, reproduction, and mutation to improve the quality of the candidate solutions. Selection is based on the 'fitness' of the candidate solution, i.e. how well it performs as a solution.

In our research, as seen from the degradation value (increment in delay or decrement in I_{DDX}) [3], the degradation follows a *non-linear* trend. Furthermore, every processor core (in a multi-processor SoC) has its own best fitted aging degradation model based on their health-monitor test results. However, there exists no model to be the *optimal* solution for all processors by the traditional optimization methods. Thus, it requires an efficient optimization algorithm to determine the combination of coefficients in the model which minimize the degradation trend error for *all* processors. The GA is more

likely to converge to a global optimum, since the algorithm searches from a population of points, and is based on probabilistic rules.

A GA was employed to optimize sets of coefficients for the proposed degradation model with regard to the path delay.

The flowchart of the used GA is shown in Figure 6. Fundamental operations of the GA are summarized as follows.

I. Initialization. Randomly generate a population of chromosomes, in our case it is 100. The chromosomes are actually the candidate solutions. Since our target is searching for the optimal degradation model for the delay, thus the chromosome for the optimization consists of information corresponding to the shape of the trend. From our previous delay testing results, delay value changes ($\Delta delay$) have shown to follow a *power* dependency with respect to aging time [1]; it is therefore assumed the path delay has a power-law relation with respect to the time t:

$$f(t) = delay(t) = a + b * t^c \tag{7}$$

Therefore, the evaluated population consist of chromosomes corresponding to three searching parameters: *a, b* and *c*.
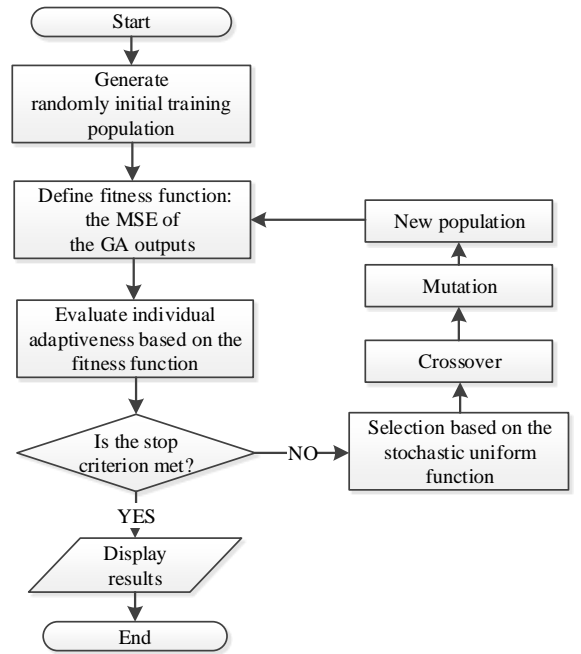

Figure 3. Flowchart of the GA procedure

II. The fitness function and adaptiveness evaluation. The fitness function is set to the mean-squared error (*MSE*) of the GA results and the measured results:

$$Fitness\ functional = F(x)$$
$$= \frac{\sum_N (delay_{GA} - delay_{measured})^2}{N} \tag{8}$$

After the generation of the new population, there will be an evaluation of the adaptiveness for each chromosome based on the fitness function $F(x)$. The larger the fitness value for

the chromosome, the less adaptiveness there will be, and the more rearward ranking for it will result.

III. Selection. Roulette-wheel selection was used, i.e. the probability that an individual was selected for placement in the next generation population was proportional to that individual's fitness calculated in step II.

IV. Crossover. With a pre-defined crossover probability (90% in our case), a crossover of the parents forms new offspring. The crossover is implemented via scattered crossover: individuals in the parent chromosome pool are paired off randomly. Then we create a random binary vector and select the bits (binary info in *p1* and *p2*) where the vector is a *1* from the first parent, and the bits where the vector is a *0* from the second parent, and combines the bits to form the child. This generates two new 'offspring' with a mixture of the two parents' characteristics.

V. Mutation. 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 a broader space. In our case the probability of mutation is set to 0.1%, which means 0.1% of the bits in the parent undergo mutation.

After mutation, a GA cycle is completed. In this way, the GA is set to run for a maximum of 100 generations. Meanwhile, delay data of our 30 processors has been used for the training purpose to get the degradation trend. The delay data of the other 16 processors is used for the purpose of validation, this is also referred to as the holdout method [11] for the accuracy evaluation of the predicted model. The optimization result with data is shown in Figure 4.
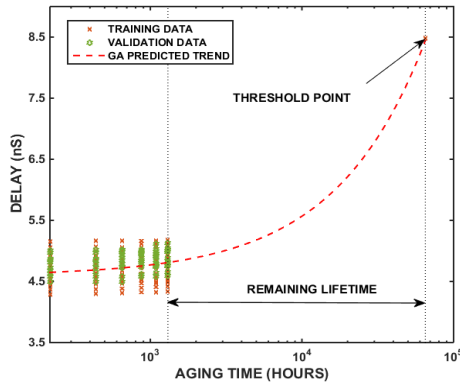

Figure 4. Predicted degradation trend of the path delay using real training data (red); validation data is also shown (green).

The remaining lifetime based on the GA model is calculated. 30 devices results in the training set are shown in Figure 5. The statistical results for both training set group and validation group are illustrated in Table I.

Groups	MAX.	MIN.	MEAN	STD.	RMSE
Training set	8.18	5.54	6.60	0.64	0.54
Validation set	6.84	5.81	6.22	0.43	0.32
Total	8.18	5.54	6.42	0.57	0.52

One can read that the MAX, MIN, MEAN, STD and RMSE in the training group are all larger than that in the validation group. That's reasonable since the number of devices in previous group is twice as the later one.

The accuracy [11] 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, in equation it is

$$\text{acc} = \frac{1}{v} * \sum_{i=1:v} \delta(\varepsilon(vi), MSEt) \tag{9}$$

v is the number of devices in the validation group, $\varepsilon(vi)$ is the error calculated from the validation group based on the training model, while $MSEt$ is the mean squared error of the training group, $\delta(\varepsilon(vi), MSEt) = 1$ if $\varepsilon(vi) \leq MSEt$. In our case, all errors from the validation group is less than the $MSEt$ from the training group, that means the prediction model has a full accuracy for all 46 devices.

C. The Remaining Lifetime Prediction (RLP) via I_{DDX}

Our previous research [1] proved that I_{DDQ} and I_{DDT} are highly correlated with the path delay according to our measurements. The absolute values of correlation coefficients are greater than 0.7 and very close to 1 for all chips.

Therefore a regression model was developed to map the delay with a given I_{DDQ} and I_{DDT}. Equation (9) shows the resulting mapping function between critical path delay and I_{DDQ} with $MSE = 1.142 * 10^{-5}$:

$$f(t) = g(I_{DDQ}) = -0.4356 * I_{DDQ}(t) + 5.5674 \tag{10}$$

Since I_{DDT} includes a number of attributes, denoted by $I_A(t)$, $I_{LD}(t)$, $I_M(t)$, $I_P(t)$, $I_S(t)$ and $I_{ST}(t)$, the mapping function (Equation (10)) is derived based on a multivariate regression:

$$f(t) = h(I_{DDT}) = 6.7598 + 0.036444 * I_A(t)$$
$$+ 0.1119 * I_{LD}(t) - 0.40886 * I_M(t)$$
$$+ 0.35631 * I_P(t) - 0.007251 * I_S(t)$$
$$- 0.28211 * I_{ST}(t), \tag{11}$$

in which $MSE = 1.916 * 10^{-3}$.

The strong correlation between path delay and I_{DDX} is proven again by the fittings above with tiny *MSEs*. It is reasonable to assume that the fitting curves/lines of chips would have a similar slope but different intercepts (offsets). Based on such an assumption and the previously GA optimized model (Equation (7)), we propose an algorithm to predict the remaining lifetime

of the Xentium by following. For reading convenience, the algorithm is described for each single chip with respect to I_{DDQ}. For I_{DDT}, the algorithm is analogous, replacing the I_{DDQ} measurement input $I_{DDQ}(t)$ and the mapping function (Equation (10)) by the I_{DDT} measurement input and the mapping function (Equation (11)), respectively.

Algorithm 1 <RLP via I_{DDQ}>

1: INPUT: GA based delay trend f(t), I_{DDQ} measurements $I_{DDQ}(t)$, delay threshold T_{th}, current operating time t_i

2: OUTPUT: Remaining lifetime L_{IDDQ}

3: Initialize $t = 0 : 6$ as time points for 6 weeks

4: Compute average offset $\bar{T}_{IDDQ} \leftarrow \overline{g(I_{DDQ})} - \overline{f(t)}$

5: Estimate remaining lifetime $L_{IDDQ} \leftarrow$
   $(f(t)|t = L_{IDDQ} + t_i) + \bar{T}_{IDDQ} = T_{th}$
   <end>

Based on our measurement data, the statistical RLP results of our 46 chips using Algorithm 1 are given in Table II. The results show that the average I_{DDT}-based RLP value is a little better than the I_{DDQ}-based one, but both methods have significantly close results. Compared to the mean remaining lifetime from the HOTL test, which is $6.38*10^4$ hours, the I_{DDQ}-based RLP performs a bit better in terms of standard deviation and RMSE. Nevertheless, both approaches have a very similar performance.

TABLE II.   STATISTICAL RLP RESULTS ($*10^4$ HOURS) VIA I_{DDX} BASED ON DEVELOPED GA MODEL AND ALGORITHM 1

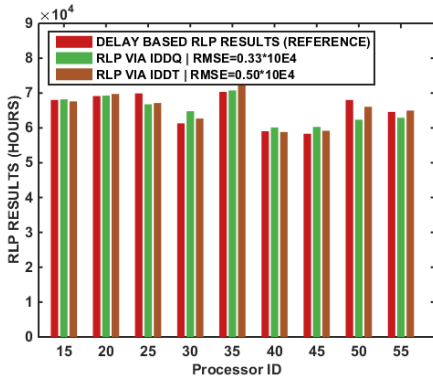| Methods	MAX.	MIN.	MEAN	STD.	RMSE
I_{DDQ}	7.26	5.87	6.47	0.33	0.33
I_{DDT}	7.92	5.47	6.48	0.49	0.5

Figure 5.   Remaining lifetime predicted via I_{DDX} for 9 random processors from the GA model and Algorithm 1.

Figure 5 shows the lifetime bar of 9 random Xentium processors with the RLP results from the I_{DDQ} and I_{DDT} test data. The mean remaining lifetime based on delay is marked by the dotted line in the figure. Again, the bars show that the RLP based on I_{DDQ} or I_{DDT} testing are very similar to the one based on the 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 path-delay testing.

V.   CONCLUSIONS

In this research, we have proposed the genetic algorithm based degradation model optimization and the remaining lifetime prediction based on the model of the 90nm VLIW Xentium processor. A reliability testing system with 1000 hours HTOL test for 46 chips has completed, with measurements carried out on the processor's path delay, I_{DDQ} and I_{DDT}. It shows that the remaining lifetimes predicted by the I_{DDQ} and I_{DDT} have a mean and standard deviation of (6.47, 0.33) $*10^4$ hours and (6.48, 0.5) $*10^4$ hours respectively, compared to (6.42, 0.57) $*10^4$ hours in the delay testing. This indicates that predicting the remaining lifetime with I_{DDX} can reach a good accuracy. Together with (embedded) processor software, the dependability of multi-processor SoCs can be increased.

REFERENCES

[1]   Y. Zhao and H. G. Kerkhoff, "Predicting aging caused delay degradation with alternative IDDT testing in a VLIW processor," presented at the Proceedings of the Workshop on Manufacturable and Dependable Multicore Architectures at Nanoscale, MEDIAN, Tallinn, Estonia, 2015.

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

[3]   Y. Zhao and H. G. Kerkhoff, "Unit-Based Functional IDDT Testing for Aging Degradation Monitoring in a VLIW Processor," presented at the Euromicro Conference on Digital System Design (DSD), 2015.

[4]   J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*: MIT press, 1992.

[5]   http://www.recoresystems.com.

[6]   K. H. G. Walters, S. H. Gerez, G. J. M. Smit, S. Baillou, G. K. Rauwerda, and R. Trautner, "Multicore soc for on-board payload signal processing," in *NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, 2011, pp. 17-21.

[7]   "http://www.ridgetopgroup.com/doc/QT-1411-HL-revB.pdf."

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

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

[10]   W. B. Nelson, *Accelerated testing: statistical models, test plans, and data analysis* vol. 344: John Wiley & Sons, 2009.

[11]   R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, 1995, pp. 1137-1145.