Circuit Reliability Simulation based on Verilog-A

Marq Kole
NXP Semiconductors
marq.kole@nxp.com

ABSTRACT

An iterative simulation environment for reliability problems due to long-term circuit degradation has been developed for use in SPICE-like simulators. A model developed for the NBTI (Negative Bias Temperature Instability) degradation effect in the standard Verilog-A language can be used with this reliability simulator. Integration in a design environment allows easy access to reliability simulation for the circuit designer.

1. INTRODUCTION

For the latest nano-scale CMOS process generations reliability issues are on the rise again. At the very small dimensions of these devices the high electric fields imply hot carrier issues, while also temperature instabilities may occur. However, models for these degradation effects are not standardized and need to be developed, similar to regular compact device models. Also, the support for reliability simulation in standard circuit-level simulators varies per product and mostly relies in proprietary model code that gives no insight in what is actually modeled. The essential peer-review process that has been so successful in making the BSIM3v3, BSIM4 and PSP models for MOSFET devices applicable to many technologies and application areas has not been working yet in the area of degradation models. Unfortunately, the need for such models and indeed a high quality in these models is rising fast as process technology is no longer able to mitigate the influence of hot carrier injections fully. In addition, previously irrelevant degradation mechanisms such as NBTI are threatening to become significant sources of long-term circuit failure. Proprietary developments are not able to keep up with the research in the semiconductor physics underlying these effects.

To enable circuit designers to determine the influence of long-term circuit degradation on the performance of their circuit a reliability simulation system is needed. In this system they will be able to correctly determine the amount of degradation each individual device in the circuit is experiencing, and to adjust the devices' parameters such that that degradation is reflected in the device behavior and as a result in the circuit performance.

1.1. Outline of the Paper

In the next section we will first discuss the nature of long-term circuit degradation and the most prominent degradation mechanisms that affect circuit reliability. Special attention will be given to the NBTI mechanism as that has been the reason for recent heightened activity in the field of circuit-level reliability. After that we will dive into the possibilities for reliability simulation at the circuit level and discuss both existing solutions in proprietary tools and the open Presto environment developed in the current context. Before going into discussing the detailed implementation of Presto the NBTI model used to measure device stress due to NBTI is presented. In the section after that we are able to present the Presto environment that should allow an IC designer to perform circuit-level reliability simulation. Finally, some experimental results with the Presto tool are shown to illustrate its efficiency and ability to turn out a model of a circuit at a certain age of typical use in an application.

2. CIRCUIT-LEVEL RELIABILITY

There has already been quite some investigation in reliability simulation in the late 80's and early 90's of the last century [10], and it is commercially available from several vendors [2,8]. However, reliability simulation is not in widespread use and knowledge on both the simulation as well as the use of the degradation models is limited. It is quite visible in the marketplace that proprietary models prevail – the models are often closed source and little or no information is available on what is actually modeled.

This is an undesirable situation as the impact of the NBTI and HCI (Hot-Carrier Injection) failure mechanisms is increasing. Where in older process technologies the reliability effects were mitigated through either technology optimization or reduction of the electric field strength by bringing down the supply voltage levels, neither of these solutions are an option for nano-scale devices developed in the advanced CMOS processes of today. The result is that circuit-level reliability is becoming the responsibility of the designer. This can only be solved by providing a similar design environment for reliability simulation as for traditional circuit simulation: by using a reliability simulation tool based on foundry provided reliability parameters for industry standard models.

For the reliability simulation method a good model is the BERT approach [5] shown in Figure 1, which supports two basic requirements:

- Strict separation of the time scales of the degradation effects and the circuit behavior;
- Ability to perform serial degradation simulations to take stress feedback into account.

The first item is important to prevent undesirable interference between the device behavior and the degradation effects as would be observed by accelerating the degradation simulation to the level of circuit simulation. The second effect allows a designer to observe second order degradation – a device experiences more or less stress due to another device that has degraded. This may lead

978-1-4244-1567-0/07/$25.00 © 2007 IEEE

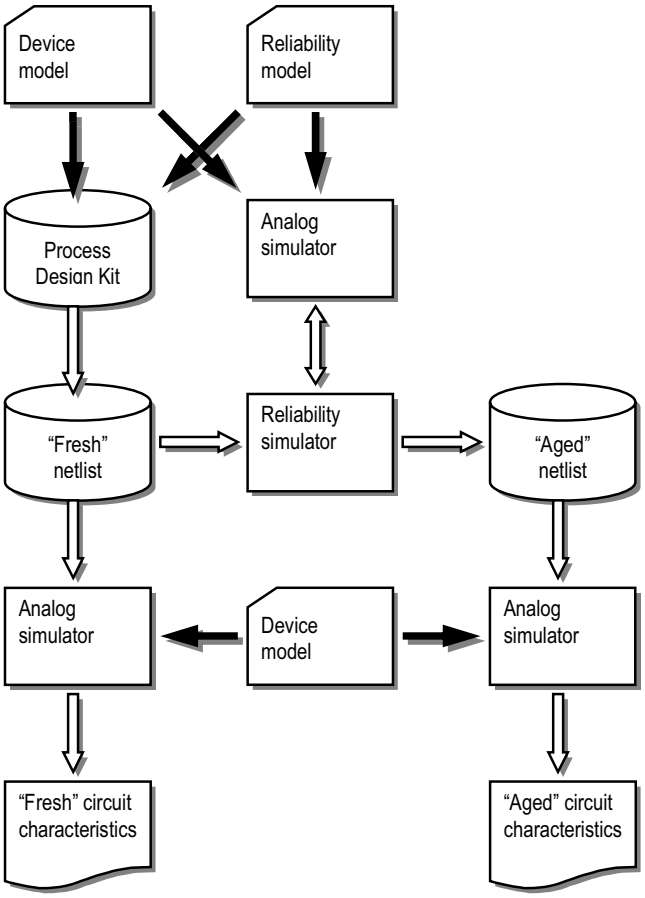

Figure 1: Reliability simulation environment required for the BERT model.

to overly pessimistic design if the feedback is negative, or lifetime problems in case of positive feedback.

To be able to set an industry standard for reliability models for the NBTI and HCI effects a model and simulation environment are used which are generic and not dependent on proprietary models and reliability simulation solutions.

2.1. Negative Bias Temperature Instability (NBTI)

Negative Bias Temperature Instability – discovered already in 1967 [4] – is a degradation mechanism that occurs when a negative bias is applied to a MOS device – a $V_{gs} > 0$ for a PMOS, a $V_{gs} < 0$ for an NMOS. This negative bias results in a hole population in the bulk underneath the gate oxide. This degradation is strongly temperature dependent with a higher temperature inflicting a stronger degradation, hence the name. For PMOS devices this situation occurs each time the device is in inversion. There is no lateral component in the degradation in the sense that current flow has no influence on NBTI degradation. The observed influence of NBTI on the device behavior is adequately described by a threshold voltage shift [6].

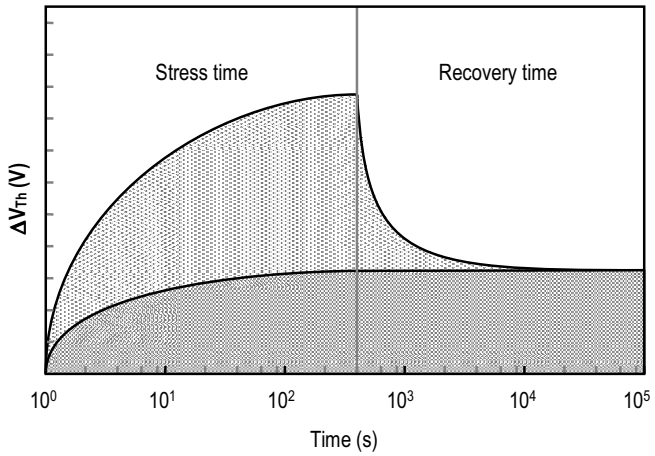

Figure 2: Threshold voltage shift due to NBTI degradation with permanent damage (dark shading) and recovery (light shading).

The NBTI effects also appears in NMOS devices but negative biasing ($V_{gs} < 0$) of these devices is not a typical operating condition in CMOS circuits and therefore of less consequence.

In the situation of a negative bias possibly with a rise in temperature the gate-oxide interface to the channel is degraded uniformly resulting in interface states as well as oxide charges. Next to the above degradation there is also a mechanism of hole trapping. This mechanism has the same effect as the interface states and oxide charges on the behavior of the device but is completely reversible. As a result, the trapping part of the degradation will disappear once the negative bias is removed and only the effects of the interface states and oxide charges remains, as can be seen in Figure 2.

For NMOS devices there is also a PBTI (Positive Bias Temperature Instability) effect which is very similar to the NBTI effect but becomes active on positive biases for NMOS devices and although it has a lower onset, the degradation develops faster and can potentially be just as destructive as NBTI in PMOS devices.

2.2. Hot-Carrier Injection (HCI)

Hot-Carrier Injection is a phenomenon related to the damage high velocity carriers (electrons or holes) can do on the crystal lattice in the conducting parts of semiconductor devices [10]. The carriers can attain these high velocities due to a strong electric field that originates from relatively large voltage differences in small devices. As a result, short channel MOSFET devices will have a higher danger of HCI degradation than long channel devices. The resultant effects are bulk and interface states along the pinch-off part of the MOS channel. The damage starts at the drain side of the channel and progresses towards the source side over time.

A question remains how for NMOS devices the HCI degradation mechanism interacts with the PBTI mechanism. The two mechanisms can be active at the same time in a suitably biased

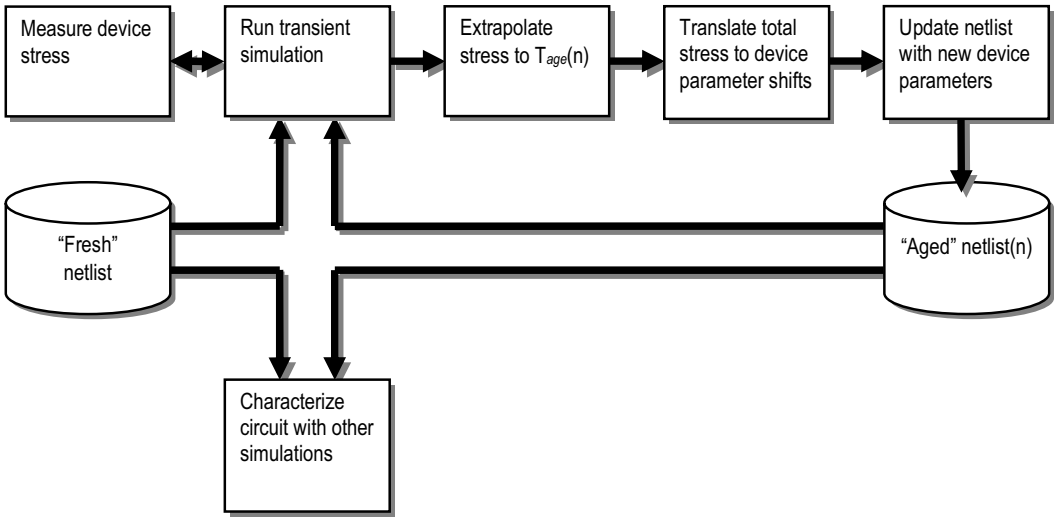

Figure 3: BERT methodology for iterative reliability simulation .

device but their combined influence has not been rigorously researched.

As a degradation mechanism HCI is not considered in the current reliability model and simulator implementation, but needs to be considered for future extensions in which multiple degradation mechanisms need to be simulated simultaneously.

3. RELIABILITY SIMULATION

For simulating circuit reliability the BERT methodology [5] as shown in Figure 3 is used. At this point we introduce the use of Verilog-A to perform most of the steps in this method:

1. The fresh netlist contains a Verilog-A model for each semiconductor device for which the degradation is to be determined. This Verilog-A model is connected to the semiconductor device in such a way that all relevant stress-inducing variables can be measured – the terminal voltages and terminal currents.
2. For the stress measurement a transient simulation needs to be run to capture the stress during *typical* operation.
3. During the simulation the measurements of the stress-inducing variables are integrated in the model to provide a *mean stress* over the simulation time. The use of Verilog-A also allows the stress measurement to skip an initial period that can be used for the circuit mode of operation to stabilize.
4. At the end of the simulation run, the mean stress is extrapolated to the required age. It is important that this extrapolation is essentially linear as the assumption is that the typical behavior executes for the full time period of the age.
5. With the *extrapolated mean stress* the shift of affected device parameters can be calculated. Verilog-A enables the model to write out these parameter shifts with a reference to the originating device. This is needed to perform the update of the circuit netlist to finally get the *aged netlist*.
6. With the aged netlist one can now perform the same loop once more, returning *intermediate aged netlists* with each pass. This enables a designer to take the effect of the degradation-induced changes in device behavior into account already during the reliability simulation. However, total time needed for a reliability simulation scales linearly with the number of iterations.

This approach also indicates that if a circuit has multiple modes of operation a reliability simulation should be performed for all modes of operation to be able to determine the likelihood of a reliability problem in an application.

4. NBTI STRESS MODEL

As shown in the previous section, a model for degradation effects should perform two main functions:

- It should measure the stress a device experiences;
- It should convert the measured stress to a change in behavior of the affected device;

As stress is different for each device and each mechanism there is no standard unit in which to express stress. Therefore, the measurement of stress and its impact on device behavior are intricately linked. This way the stress measurement can remain internal to the model.

On the other hand, the measurement of stress is only dependent on the physics of the device, while the change in behavior is linked to the way behavior is modeled and hence to the model of the device. As certain behavior is modeled differently in, for example, the BSIM4 model and the PSP model for a MOSFET device, the actual changing of behavior may also have to be performed differently.

In case of NBTI the Vth0 (threshold voltage) parameter is changed for a MOSFET instance based on the BSIM4 model. If the instance is based on the PSP model the Vfb (flatband voltage) parameter is adjusted.

4.1. NBTI Stress Measurement

According to Parthasarathy et al [11] the NBTI stress can be modeled in the following ways:

The permanent damage ΔD_P is described using the following equation:

$$\Delta D_P(t_{stress}) = A_p \exp(\gamma_p V_{GS}) \exp\left(-\frac{E_{ap}}{kT}\right) t_{stress}^{\ n} \quad (1)$$

Where V_{GS}, T and t_{stress} are the gate-source voltage, the temperature, and the time the stress is applied. These are the only stress-inducing variables used to describe the NBTI effect.

The recoverable damage ΔD_R is described using the equation:

$$\Delta D_R(t_{stress}) = A_r \exp(\gamma_r V_{GS}) \exp\left(-\frac{E_{ar}}{kT}\right) \ln\left(1+\frac{t_{stress}}{\tau_r}\right) \quad (2)$$

The damage recovery ΔR of the recoverable damage ΔD_R is described by:

$$\Delta R(t_{stress}, t_{re\,cov}) = A_{rr} \exp(\gamma_{rr} V_{GSre\,cov}) \exp\left(-\frac{E_{arr}}{kT}\right) \ln\left(1+\frac{\frac{t_{re\,cov}}{t_{stress}}}{\tau_{rr}}\right) \Delta D_R \quad (3)$$

In Listing 1 a simple model of the NBTI degradation effect is shown that only performs the permanent damage calculation as described by equation (1). In practice this will give mostly too optimistic results as neither the recoverable damage nor the recovery are applied, so the remaining recoverable damage is ignored.

Listing 1: simple NBTI model

```
`include "disciplines.vams"

`define XML_FILE_NAME "nbti.xml"

module mnbti (d, g, s, b);
inout d, g, s, b;
electrical d, g, s, b;
parameter integer active = 1 from [0:1];
parameter real dta = 0.0;
parameter real tage = 0.0 from [0:inf);
parameter real tstart = 0.0 from [0:inf);
parameter real tstop = 1.0e+34 from (tstart:inf);
parameter real alpha_dp = 0.0;
parameter real VG_dp = 0.0;
parameter real Ea_dp = 0.0;
parameter real n_dp = 1.0;
parameter real tsat_dp = 1.0e+06 from [0:inf);
parameter real Vfb = 0.0;
localparam real eq_stress_time = tstop - tstart;

branch (g, d) gd;
branch (g, s) gs;
branch (d, b) db;
branch (s, b) sb;

integer stressed, calculate, update_file;
real vgs, exp_temp_dp, c_dp, exp_vg_dp, eq_vgs, dp, delta_vfb;
real eq_nbti_dp, eq_time, eq_stress_time_int;
analog begin

  // Initialization
  @(initial_step)
    if (active && analysis("tran") && (tage > 0)) begin
      update_file  = $fopen(`XML_FILE_NAME);
      calculate = ((alpha_dp == 0) || (Ea_dp == 0)) ? 0 : 1;
    end

  // Stress measurement
  if (calculate) begin
    // activation flag to sync with user specified tstart and tstop
    stressed = ($abstime > tstart) ? (($abstime > tstop) ? 0 : 1) : 0;

    vgs = (abs(V(db)) >= abs(V(sb))) ? V(gs) : V(gd);
    if (vgs > 0) vgs = 0; // positive bias not taken into account
    exp_temp_dp = Ea_dp / $vt($temperature + dta);
    exp_vg_dp = VG_dp * abs(vgs);
    c_dp = alpha_dp * exp(-exp_temp_dp) * exp(exp_vg_dp);

    // Permanent damage calculation
    eq_nbti_dp = idt(((exp_vg_dp == 0) || !stressed)
      ? 0.0 : pow(c_dp, 1.0/n_dp), 0.0);
    eq_time = idt((!stressed) ? 0.0 : 1.0, 0.0);
  end // if calculate

  // Output of  stress data
  @(final_step)
    if (calculate) begin
      // saturation after a certain time
      eq_stress_time_int =
        (eq_stress_time >= tsat_dp) ? tsat_dp : eq_stress_time;
      dp = (eq_time > 0.0)
        ? pow(abs(eq_nbti_dp) * eq_stress_time_int/eq_time, n_dp)
        : 0.0;

      // the computation of parameter changes is done right here
      delta_vfb = dp * tage/eq_stress_time;
      if (-delta_vfb) begin
        $fwrite(update_file, " <instance name=\"%M\">\n");
        $fwrite(update_file, "  <parameter name=\"%s\"", "vfb");
        $fwrite(update_file, "  value=\"%1.8g\" delta=\"%1.8g\"/>\n",
          Vfb, -delta_vfb);
        $fwrite(update_file, " </instance>\n");
      end
      $fclose(update_file);

    end // if calculate
end // analog

endmodule // mnbti
```

The code is a little more complex than necessary for permanent damage only, but essentially all measurements of the damage are integrated in the body of the model to keep track of the signal variations over the time of typical operation. The full model that includes recoverable damage as well as damage recovery, and the dependencies of frequency and duty cycle as described in [11] contains no less than 13 integrators.

This also points us at one of the problems of reliability simulation: the overhead of evaluating the stress measurement model. Due to the amount of calculations this currently amounts to a factor between 1.5 and 2 for a single iteration using a single transient run. This can only be overcome by creating more efficient models.

4.2. Behavioral Change due to NBTI Stress

At this moment all published models for the NBTI effect implement the change in device behavior simply through a threshold voltage (V_{Th0}) shift. All other phenomena are suitable accounted for in this way. As the PSP and MM11 model do not use the threshold voltage as a parameter, the flatband voltage V_{FB} is used instead. According to the model descriptions threshold voltage and flatband voltage are directly proportional to each other.

For the BSIM4 model the threshold voltage shift is even a separate parameter (delvth0) that can be used directly as it is meant to be used for mismatch, such as caused by circuit degradation.

5. THE PRESTO ENVIRONMENT

Presto is a circuit-level reliability simulator originally developed for degradation simulation based on a Verilog-A model of the NBTI effect. Although the model had originally been developed using the UDF interface of the circuit simulator ELDO, this turned out to be non-portable to other simulation tools. By using Verilog-A as a standard language for compact device models it has become possible to make this model available in other circuit simulators and perform circuit-level reliability simulations with it.

5.1. Reliability Simulation Loop

Concerning the BERT reliability simulation methodology shown in Figure 3, the Verilog-A code of the model takes care of the stress measurement, the extrapolation, and the conversion from extrapolated stress to a change in device parameter values. What is missing is the update of the netlist to reflect the new values. This is essentially what Presto does. The output from the Verilog-A model is in an XML format – Presto converts this XML code into netlist statements specific for the circuit simulator in use. At this moment it can support this functionality for Spectre™, Eldo™ and 2 proprietary circuit simulators.

5.1.1. Parameter updates

Unfortunately, parameter updates are not supported in all circuit simulators. In the simulators that do support them, the aged circuit can simply be obtained by combining the original circuit with the parameter updates due to the stress measurement extrapolation. For instance in Spectre™ the **alter** statement can be used to update the semiconductor device parameters anywhere in the circuit hierarchy.

For models in Verilog-A the **defparam** statement has essentially the same function but unfortunately to our knowledge no analog simulators with Verilog-A support actually support this statement.

If none of the above solutions work the only remaining option is full canonical netlist expansion.

Presto does not provide parameter updates if there has been no observable stress and therefore no parameter shifts. It also allows one to filter out small changes relative to the parameter values – this gives the user the ability to reduce the amount of data for larger designs and concentrate on the possible

5.1.2. Device identification

At this moment, the hierarchical reference for the device for which a particular set of parameter updates is meant is determined from the hierarchical name of the NBTI measuring model. As this degradation model is – in current implementations – located inside a subcircuit that takes these parameters and passes them to the semiconductor device, the solution is to shorten the hierarchical path by just one name – the name of the degradation model instance – to get the name of the subcircuit that needs to take the parameter updates.

Of course, when the degradation model is integrated directly in the Verilog-A code of the semiconductor model, there is no need to restructure the hierarchical reference as it will already be correct – degradation model and device model are one and the same instance.

5.1.3. Iteration control

The most important service that Presto provides is to perform the iterations in the BERT methodology so for each new iteration the value of *age* is updated. However, as the iteration starts from an already aged circuit, the *age* value of the current netlist should be deducted from the previous value of *age*:

$$T_{age}(n+1) = T_{user}(n+1) - T_{age}(n) \quad (4)$$

Where T_{age} is the age value used for extrapolation, and T_{user} is the *age* value provided by the user.

5.1.4. Pure Verilog-A implementation

As an alternative to Presto a trick can be used in Verilog-A to mark every combined device-with-degradation model instance with a random number and use a table model to have each device retrieve the correct parameter update. This trick relies on undocumented features of the simulator – Verilog-A does not guarantee that the evaluation of instances always proceeds in the same order, but in practice it does. Still, this may not work in all circumstances or in all simulators.

Now the only thing that has to be done outside of the circuit netlist is for each iteration to provide a value for the *age* to extrapolate to, and to restart the transient simulation until the *final age* is reached. Post-processing of the results – finding the devices

that have had the highest stress in the circuit can be achieved by correlating the table model containing the updates with a list of devices and corresponding random number values.

The upcoming version 2.3 of the standard will probably make this trick obsolete as the same functionality is expected to be achieved with standard Verilog-A modeling at better performance.

5.2. User Control of the Reliability Simulation

Through command-line options or through a Cadence DFII™ integration module written in SKILL™, the user can control the reliability simulation with Presto. Normally, the value of *age* – or rather a list of values for *age* in case of an iterative approach – is the most important parameter. It is also possible to provide different time units (seconds up to years) and to ask for a summary showing the development of degradation over time (i.e. iterations) for each device.

6. EXPERIMENTAL RESULTS

Running Presto on a real-life circuit – a latched comparator from a SAR ADC circuit designed in a 65nm CMOS process technology with an NBTI model conjoined with each PMOS device – will readily give results. For a temperature of 65 ºC the chart in Figure 4 shows the shift in threshold voltage due to NBTI stress in four different operating modes. From the vertical scale in µV one can infer that NBTI is not an issue for this particular circuit when operated under normal conditions.

Without a common, agreed model extraction of NBTI degradation parameters is no common practice. Some impact analysis points at GO2 (thick gate oxide) MOSFET devices as the most likely to suffer from NBTI degradation [3]; other concerns in literature point at memory devices – in particular for the static noise margin characteristic – to be likely victims of NBTI due to the long static stress in operation [9]. Presto provides the opportunity to analyse the impact of NBTI and other degradation effects on circuits and the long-term effects of their actual operation.

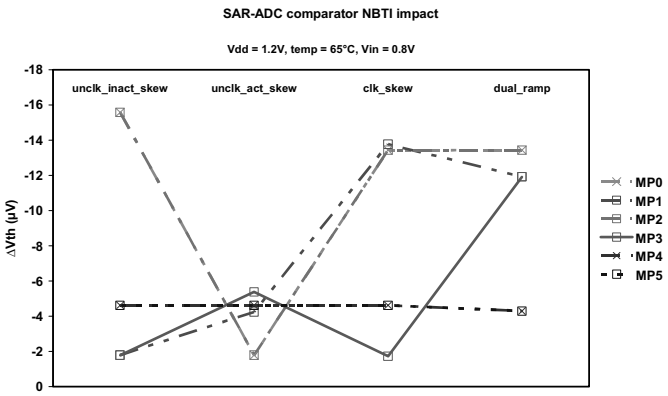

Figure 4: Presto simulation results of the threshold voltage shift (ΔVth) due to NBTI for 6 PMOS devices in a latched comparator after 10 years of normal stress (V_{dd} = 1.2 V, T = 65 ºC).

7. FUTURE RESEARCH

An important aspect in future research is the case of multiple degradation effects acting upon the same device. An example could be HCI and PBTI for NMOS devices in nano-scale CMOS processes. Also statistical simulations in combination with degradation models is an area of investigation.

ACKNOWLEDGEMENTS

The author would like to thank the Alliance Reliability Team in the Crolles2 facility for valuable discussions and cooperation. Moreover, the feedback from various users in the NXP process development and analog design communities is highly appreciated.

REFERENCES

[1] M.A. Alam, "A Critical Examination of the Mechanics of Dynamic NBTI for PMOSFETs", *IEEE Intl. Electron Devices Mtg.,* December 2003, pp. 345.

[2] Cadence Design Systems, "Reliability Simulation In Integrated Circuit Design", white paper, *http://www.cadence.com/whitepapers/5082_ReliabilitySim_FNL_WP.pdf*, 2003.

[3] P. Chaparala, J. Shibley, P. Lim, "Threshold voltage drift in PMOSFETS due to NBTI and HCI," *IEEE International Integrated Reliability Workshop Final Report*, pp. 95–97, October 2000.

[4] B. Deal, M. Sklar, A. Grove, E. Snow, "Characteristics of the Surface-State Charge (Qss) of Thermally Oxidized Silicon", *Journal of the Electrochemical Society*, Volume 114, pp. 266, 1967.

[5] C. Hu, "The Berkeley Reliability Simulation BERT: an IC Reliability Simulator", *Microelectronics Journal*, Volume 23, Issue 2, pp. 97–102, 1992.

[6] V. Huard, M. Denais, C.R. Parthasarathy, "NBTI degradation: From physical mechanisms to modelling", *Microelectronics Reliability*, Volume 45, pp. 1–23, 2005.

[7] Int. Sematech, "Semiconductor Device Reliability Failure Models", *http://ismi.sematech.org/docubase/document/3955axfr.pdf*, 1996.

[8] M. Karam, W. Fikry, H. Haddara, H. Ragai, "Implementation of Hot-Carrier Reliability Simulation in Eldo", *IEEE International Symposium on Circuits and Systems*, Volume 5, pp. 515–518, May 2001.

[9] S. Kumar, C. Kim, S. Sapatnekar, "Impact of NBTI on SRAM Read Stability and Design for Reliability", *IEEE International Symposium on Quality Electronics Design*, pp. 73–78, March 2006.

[10] M. Lunenborg, "MOSFET Hot-Carrier Degradation, Failure Mechanisms and Models for reliability circuit simulation", *Ph.D. Thesis, University of Twente*, The Netherlands, 1996.

[11] C.R. Parthasarathy, M. Denais, V. Huard, G. Ribes, D. Roy, C. Guerin, F. Perrier, E. Vincent, A. Bravaix, "Designing in reliability in advanced CMOS technologies", *Microelectronics Reliability*, Volume 46, pp. 1464–1471, 2006.