Aging Simulation with variation of several model parameters

“Bipolar Arbeitskreis”, Erfurt
2017/11/10

Klaus-Willi Pieper
Infineon Technologies AG
Agenda

• Introduction
• The concept of one aging parameter
• Aging simulation flow
• Aging model example
• Lookup table approach
• Simulation example
• Conclusions
Reliability measurements

- **Static stress** is applied to the device; it is kept at a certain operating point at a certain temperature for a certain time (e.g. $V_{DS}=4.8\,V$, $V_{GS}=3.1\,V$, $T=-40\,^\circ C$ for $t=1000\,s$)
- **Observables** like $v_{tlin}$, $v_{tsat}$, $i_{dlin}$, $i_{danalog}$, $i_{dsat}$ are measured after static stress
- Stress effects that observables are changing according to a **power law**

\[
\frac{\Delta P}{P} = c \cdot t_{stress}^{n} \quad (1)
\]

where $n$ is constant (e.g. 0.5) and $c$ depends on the operating point, temperature and geometry.
**Introduction**

**lifetime** $\tau$ is reached when $\Delta P/P$ of an observable reaches a given failure criteria $F_C$ (as the first one) after stress at a certain operating point (Op).

$$F_C = c \left[\tau(Op)\right]^n \quad (2)$$

Equation (1) and (2) give

$$\frac{\Delta P}{P} = F_C \left(\frac{t_{\text{stress}}}{\tau(Op)}\right)^n = F_C age^n \quad (3)$$

which leads to the definition of ‘age’

$$age = \frac{t_{\text{stress}}}{\tau(Op)} \quad \text{(static)} \quad \text{or} \quad age = \int_0^{t_{\text{stress}}} \frac{dt}{\tau(Op(t))} \quad (4)$$

(static) \quad \text{(dynamic)}$$
The concept of one age parameter

One parameter ‘age’ describes completely the aging status of a device. With the value of age all shifts of observables can be calculated. ➔ to be discussed
Aging Simulation Flow

Goal: Predict if a circuit will fulfill specs also after a long circuit operation time, e.g. after 10 years

1) Fresh Simulation (any simulation, DC, AC, transient, ...)

2) Simulation of transient test pattern (representative for the whole operating time of the product, simulated time: $t_{\text{sim}}$)
   a. Calculation of age for each instance by integration over simulated time
   b. Extrapolation of age to circuit operation time
   c. Determination of model parameter shifts for each instance
   d. Storing of model parameter shifts

3) Aged simulation, same as 1) with model parameter shifts
Aging Simulation Flow

\[ age(t_{sim}) = \int_{0}^{t_{sim}} f(I_d, I_b, W, L, T) \, dt \]  

(2a)

age for simulation time

\[ age(t_{op}) = \left( \frac{t_{op}}{t_{sim}} \right) age(t_{sim}) \]  

(2b)

age for circuit operation time

(2c)

Look up table with age, model parameters

read out and store model parameters

(2d)

\[ t_{sim} \]: simulated time  
\[ t_{op} \]: circuit operation time

alter statements file
Aging model example

Lucky Electron Model for hot carrier injection (HCI)

• With length dependence and thermal activation

\[
\frac{\Delta P}{P} = A \cdot \left( \frac{I_D}{W} \right)^n \left( \frac{I_B}{I_D} \right)^{m\cdot n} \cdot t^n \cdot L^{-p} \cdot \exp \left[ -\frac{E_a}{k} \cdot \left( \frac{1}{T} - \frac{1}{T_{\text{ref}}} \right) \right]
\]

(4)

First part based on Eq. (9) of [C. Hu et al., T-ED, Vol.32, pp. 375-385, 1985]

• typical numbers:

\[n = 0.5 \quad m = 3 \quad E_a = -0.1 \text{ eV} \quad p = 1\]

pre-factor A must be determined to get a reasonable number for the technology you are using

⇒ Implementation into verilogA model
Aging model example

Stress operating point | Model | Measured
---|---|---
VGS=1.4V, VDS=2.8V | 0.79 | 0.79
VGS=2.1V, VDS=2.8V | 0.39 | 0.29
### Lookup table approach (model parameters)

<table>
<thead>
<tr>
<th>age</th>
<th>vth0</th>
<th>u0</th>
<th>ags</th>
<th>dsub</th>
<th>lpe0</th>
<th>ndep</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000</td>
<td>0.223</td>
<td>481.6</td>
<td>0.950</td>
<td>0.670</td>
<td>9.10E-08</td>
<td>2.34E+17</td>
</tr>
<tr>
<td>0.010</td>
<td>0.225</td>
<td>477.8</td>
<td>0.995</td>
<td>0.469</td>
<td>8.77E-08</td>
<td>2.64E+17</td>
</tr>
<tr>
<td>0.016</td>
<td>0.226</td>
<td>476.8</td>
<td>1.005</td>
<td>0.438</td>
<td>8.70E-08</td>
<td>2.72E+17</td>
</tr>
<tr>
<td>0.025</td>
<td>0.226</td>
<td>475.5</td>
<td>1.018</td>
<td>0.406</td>
<td>8.62E-08</td>
<td>2.82E+17</td>
</tr>
<tr>
<td>0.040</td>
<td>0.227</td>
<td>474.0</td>
<td>1.034</td>
<td>0.373</td>
<td>8.52E-08</td>
<td>2.95E+17</td>
</tr>
<tr>
<td>0.063</td>
<td>0.229</td>
<td>472.0</td>
<td>1.052</td>
<td>0.340</td>
<td>8.41E-08</td>
<td>3.12E+17</td>
</tr>
<tr>
<td>0.100</td>
<td>0.230</td>
<td>469.6</td>
<td>1.073</td>
<td>0.306</td>
<td>8.27E-08</td>
<td>3.36E+17</td>
</tr>
<tr>
<td>0.159</td>
<td>0.232</td>
<td>466.6</td>
<td>1.098</td>
<td>0.274</td>
<td>8.11E-08</td>
<td>3.70E+17</td>
</tr>
<tr>
<td>0.251</td>
<td>0.234</td>
<td>462.8</td>
<td>1.125</td>
<td>0.242</td>
<td>7.91E-08</td>
<td>4.20E+17</td>
</tr>
<tr>
<td>0.398</td>
<td>0.237</td>
<td>458.1</td>
<td>1.153</td>
<td>0.212</td>
<td>7.67E-08</td>
<td>5.00E+17</td>
</tr>
<tr>
<td>0.631</td>
<td>0.241</td>
<td>452.2</td>
<td>1.183</td>
<td>0.184</td>
<td>7.37E-08</td>
<td>6.35E+17</td>
</tr>
<tr>
<td>1.000</td>
<td>0.245</td>
<td>445.0</td>
<td>1.209</td>
<td>0.151</td>
<td>7.20E-08</td>
<td>8.29E+17</td>
</tr>
</tbody>
</table>

- Several model parameters can be used to describe aging effects
Lookup table approach (observables)

- NMOS $L = 3 \times L_{\text{min}}$
- NMOS $L = L_{\text{min}}$
- Simulated with one model parameter set belonging to $age = 0.398$
Simulation example

1. Fresh DC-Op simulation $I_2/I_1 = 1.249$
2. Transient simulation with stress integration and automatic determination of model parameter shifts
3. Aged DC-Op simulation $I_2/I_1 = 0.977$

<table>
<thead>
<tr>
<th>Alter-statements file for aged simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Set_vth0_M11_n_modelcard</strong>   alter mod=M11.n_modelcard param=vth0 value=0.22 // age = 3.69e-14</td>
</tr>
<tr>
<td><strong>Set_u0_M11_n_modelcard</strong> alter mod=M11.n_modelcard param=u0 value=481.6 // age = 3.69e-14</td>
</tr>
<tr>
<td><strong>Set_ags_M11_n_modelcard</strong> alter mod=M11.n_modelcard param=ags value=0.95 // age = 3.69e-14</td>
</tr>
<tr>
<td><strong>Set_dsub_M11_n_modelcard</strong> alter mod=M11.n_modelcard param=dsub value=0.67 // age = 3.69e-14</td>
</tr>
<tr>
<td><strong>Set_lpe0_M11_n_modelcard</strong> alter mod=M11.n_modelcard param=lpe0 value=9.09e-08 // age = 3.69e-14</td>
</tr>
<tr>
<td><strong>Set_ndep_M11_n_modelcard</strong> alter mod=M11.n_modelcard param=ndep value=2.34e+17 // age = 3.69e-14</td>
</tr>
<tr>
<td><strong>Set_vth0_M12_n_modelcard</strong>   alter mod=M12.n_modelcard param=vth0 value=0.24 // age = 0.61</td>
</tr>
<tr>
<td><strong>Set_u0_M12_n_modelcard</strong> alter mod=M12.n_modelcard param=u0 value=452.19 // age = 0.61</td>
</tr>
<tr>
<td><strong>Set_ags_M12_n_modelcard</strong> alter mod=M12.n_modelcard param=ags value=1.17 // age = 0.61</td>
</tr>
<tr>
<td><strong>Set_dsub_M12_n_modelcard</strong> alter mod=M12.n_modelcard param=dsub value=0.194 // age = 0.61</td>
</tr>
<tr>
<td><strong>Set_lpe0_M12_n_modelcard</strong> alter mod=M12.n_modelcard param=lpe0 value=7.39e-08 // age = 0.61</td>
</tr>
<tr>
<td><strong>Set_ndep_M12_n_modelcard</strong> alter mod=M12.n_modelcard param=ndep value=6.22e+17 // age = 0.61</td>
</tr>
</tbody>
</table>
Conclusions

- Aging Simulation can be done without tools like RelExpert, MOSRA, etc. just with a VerilogA model.
- An aging VerilogA model has been implemented:
  - “Lucky Electron Model” (HCI)
  - Lookup table approach
  - Model writes alter statements
- The approach allows to describe aging effects by shifting several model parameters.
Part of your life. Part of tomorrow.
Aging model example

same procedure as on slide 4

\[
age = \left( \frac{A}{F_c} \right)^{\frac{1}{n}} \frac{I_B^m}{I_D^{m-1}} \frac{1}{WL^n} \exp \left[ - \frac{E_a}{nk_B} \left( \frac{1}{T} - \frac{1}{T_{ref}} \right) \right] t_{stress}
\]

and in the quasi static approximation for the dynamic case

\[
age = \left( \frac{A}{F_c} \right)^{\frac{1}{n}} \frac{1}{WL^n} \exp \left[ - \frac{E_a}{nk_B} \left( \frac{1}{T} - \frac{1}{T_{ref}} \right) \right] \int_0^{t_{stress}} \frac{I_B^m}{I_D^{m-1}} dt
\]

\(\Rightarrow\) Implementation into verilogA model