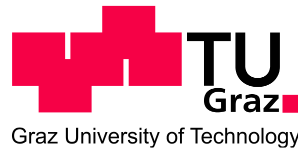

Alexander Schiffmann, BSc

MOSFET Aging Measurements and Hot-Electron Degradation Models

MASTER'S THESIS

to achive the university degree of

Diplom-Ingenieur

Master's degree programme: Advanced Materials Science

submitted to

Graz University of Technology

Supervisor:
Univ.-Prof. Peter Hadley
Institute of Solid State Physics

in cooperation with
ams AG (R&D EDA)

Graz, October 2016

EIDESSTATTLICHE ERKLÄRUNG

AFFIDAVIT

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit/Diplomarbeit/Dissertation identisch.

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis/diploma thesis/doctoral dissertation.

____________________________ ________________________________

Datum / Date Unterschrift / Signature

This thesis is dedicated to my family,

to my mother and my father, who gave me
the opportunity to study and always
supported me,

to my brother, who always spurred me to
become better,

to my grandparents, who were always there
for me.

Acknowledgement

From the Technical University of Graz the thesis was supervised by Univ.-Prof. Peter Hadley, head of the Institute of Solid State Physics. I want to thank him for the academic guidance and all his time he invested into this project.

The idea for this master thesis was developed by the Technology R&D EDA department at ams AG in Premstätten, managed by Ehrenfried Seebacher. He and Alexander Steinmair were responsible for the funding and the supervision of the project at ams AG. I want to thank them both for their kind support and for taking the time to read and correct my thesis. I worked directly with Alexander, who was a very competent and kind supervisor. The whole department made me feel very welcome during the six months I spent at the company. I like to mention Antonios Bazigos, Chong Jin, Birgit Sponer and Mingchun Tang, who I shared an office with. They supported me in many ways. Also Kund Molnar and Walter Pflanzl helped me a lot with their extended knowledge about simulation parameter extraction and the laboratory equipment. From the TCAD & Device R&D department Eleftherios Ioannidis was a big help in the laboratory and Martin Knaipp was so kind to provide me with a set of TCAD simulations for this thesis.

The inclusion of the HiSIM2 hot carrier degradation model into the thesis was only possible due to the ongoing support from the HiSIM Research Center at Hiroshima University. I want to thank Prof. Mitiko Miura-Mattausch, Ass.-Prof. Hideyuki Kikuchihara and Ass.-Prof. Hidenori Miyamoto for always providing me with latest updates of their model and their valuable advice.

Abstract

This work focuses on aging effects due to hot-carrier-injection (HCI) in MOSFETs. Hot carrier refers to fast electrons in the conducting channel. Electrons are accelerated by very high electric fields in the pinch-off region of the transistor. Their energy is high enough to generate trap states at the interface between the semiconductor and the gate oxide. Those traps capture charge which shifts the characteristics of the transistor.

To gain knowledge about the HCI effects, various measurements on n-MOSFET devices with a channel length of 0.18 µm have been performed. The devices were provided by ams AG. Aging experiments were conducted with different stress voltages and important device parameters were extracted to find how they change over time. It is observed that the stress effects scale with stress voltage and time. A short stress at high voltage causes the same threshold voltage shift as a long stress at low bias voltage. This scaling behavior is used to predict aging effects at typical operating conditions.

Since device aging due to stress is a major issue in modern semiconductor fabrication, it is of importance to understand the effects and find reliable ways of predicting how changes in the devices will affect the overall circuit performance. We used the HiSIM2 aging model developed at Hiroshima University. This model predicts the device degradation based on the bulk current which is affected by impact ionization events. A beta version of HiSIM2 was used to fit the behavior of the ams AG 0.18 µm MOSFET technology.

Furthermore we developed another aging simulation method based on the Takeda degradation model. A time depended power function, containing several model constants and parameters, was combined with the HiSIM2 MOSFET model. This heuristic approach made it possible to predict device degradation after long stress periods, but lacked the ability to scale for different applied stresses.

Contents

1 Fundamentals **1**

1.1 MOSFET Physics 1

1.1.1 Operating Regimes of the MOSFET 3

1.2 Hot-Carrier-Injection (HCI) Effect in MOSFETs 6

1.2.1 Hot Carriers 6

1.2.2 Hot Electron Transport Theory 6

1.2.3 Degradation Effects 7

1.2.4 Visualization of HCI Effects with Technology CAD Simulations . . 8

2 Experimental Realization **13**

2.1 Device Properties 14

2.2 On-Wafer Measurements 15

2.2.1 Measurement Equipment 16

2.2.2 Measurement Script 17

2.2.3 Data Extraction Routine 17

2.3 Measurements of Packaged Devices 18

2.3.1 Measurement Equipment 19

2.3.2 Measurement Script 20

2.3.3 Data Extraction Routine 20

2.4 Important Parameters for Analysis 21

2.5 Selection of Stress Conditions 22

3 HiSIM2 MOSFET Degradation Model **25**

3.1 HiSIM2 MOSFET Model 25

3.1.1 HiSIM2 Model Concept 25

3.1.2 Used Approximations within the HiSIM2 Model 26

3.2 Simulation Environment 27

3.3 Device Parameter Extraction 27

3.4 HiSIM2 Hot Carrier Degradation Model 30

3.4.1 Physical Background of the Model 32

3.4.2 Implementation into the Simulation Environment 33

3.4.3 Model Equations and Parameters 34

3.4.4 Sensitivity Analysis for Aging Parameters 38

3.5 Data Extraction Routine 44

3.6 Challenges with the HiSIM2 Degradation Model 44

4 Takeda Based Degradation Model **47**
4.1 Fundamentals . . . 47
4.1.1 Simulation Approach . . . 48
4.2 Implementation . . . 51
4.2.1 Model Part 1: Linear Regime . . . 52
4.2.2 Model Part 2: Saturation Regime . . . 53
4.3 Limitations of the Takeda Based Model . . . 54

5 Measurement Results and Interpretation **55**
5.1 Accuracy and Reproducibility of the Measurements . . . 55
5.1.1 Comparison to Existing Data . . . 55
5.1.2 Reproduction of Results . . . 58
5.2 Results for Various Stress Conditions . . . 60
5.2.1 Changes in Characteristic Curves due to Aging . . . 60
5.2.2 Parameter Changes due to Aging . . . 65
5.2.3 Extrapolation to Operating Conditions . . . 69
5.3 Width Variation . . . 72
5.4 Channel Length Variation . . . 72
5.5 HiSIM2 HC Degradation Model Verification . . . 73

6 Aging Simulation Results **75**
6.1 Simulated Changes in Characteristic Curves (HiSIM2) . . . 75
6.2 Simulated Parameter Changes (HiSIM2) . . . 80
6.3 Simulated Width Variation (HiSIM2) . . . 83
6.4 Simulated Channel Length Variation (HiSIM2) . . . 84
6.5 Takeda Model - Simulation Results . . . 85

7 Conclusions **93**

8 References **95**

9 Appendix **99**
9.1 Data Extraction Routine . . . 99
9.2 Data Difference Calculator . . . 109
9.3 Spectre Command Scripts for Simulation . . . 112
9.3.1 Initial nFET Simulation . . . 112
9.3.2 Extracted Device Parameters . . . 113
9.3.3 nFET Stress Simulation . . . 118

9.3.4 Exported Degradation Parameters for Aging 119
9.3.5 nFET Aging Simulation 120
9.4 Parameter Extraction Script for the Simulation 121
9.5 Takeda Model Part 1: Linear Regime 124
9.5.1 Main Execution File 124
9.5.2 Optimization Function 127
9.6 Takeda Model Part 2: Saturation Regime 133
9.7 Completion of Measurement Results Section 5 (Plots) 142
9.8 Completion of Simulation Results Section 6 (Plots) 145
9.8.1 HiSIM2 Aging Model Simulation Results 145
9.8.2 Takeda Model Simulation Results 148

List of Tables

1 List of conducted on-wafer measurement series 16
2 List of conducted measurements of packaged devices 19

List of Figures

1 Schematic of an unbiased n-MOSFET [1] 1
2 Schematic of a n-MOSFET in the subthreshold region [1] 3
3 Schematic of a n-MOSFET in the linear regime [1] 4
4 Schematic of a n-MOSFET at the point of pinch-off [1] 5
5 Schematic of a n-MOSFET in the saturation regime (with indication of the high electric fields) [1] 6
6 Detailed schematic of degradation effects due to hot electrons in a n-MOSFET [1] 8
7 TCAD simulation of the potential distribution in a n-MOSFET @ V_g = 0.85 V, V_d = 2.3 V [2] 9
8 TCAD simulation of the electric field density in x-direction in a n-MOSFET @ V_g = 0.85 V, V_d = 2.3 V [2] 10
9 TCAD simulation of the impact ionization rate in a n-MOSFET @ V_g = 0.85 V, V_d = 2.3 V [2] 11
10 TCAD simulation of the hole current density in a n-MOSFET @ V_g = 0.85 V, V_d = 2.3 V [2] 12
11 Wafer map (red dies have been cut for packaging) [3] 13
12 Net doping map of standard n-MOSFET device - TCAD simulation [2] . . 15
13 Photograph of measurement box with 24-pin socket (without lid) 20
14 Measured values of transconductance g_m for determination of best stress condition (M04) [4] 22
15 V_{th} as function of stress time for various drain voltages (M04) [4] 23
16 Comparison between measurement and simulation for initial transfer characteristic (linear plot) [4] 29
17 Comparison between measurement and simulation for initial transfer characteristic (semilogarithmic plot) [4] 29
18 Comparison between measurement and simulation for initial output characteristic (semilogarithmic plot) [4] 30
19 Block diagram: inclusion of the HiSIM2 aging model [1] 31

20 Trap density distribution as a function of energy [5] 35
21 Sensitivity analysis for $TRAGC0$ - transfer char. @ $t = 1e5$ s [4] 39
22 Sensitivity analysis for $TRAPGC1$ - transfer char. @ $t = 1e5$ s [4] 40
23 Sensitivity analysis for $TRAPE1$ - transfer char. @ $t = 1e5$ s [4] 41
24 Sensitivity analysis for $TRAPE0$ - transfer char. @ $t = 1e5$ s [4] 41
25 Sensitivity analysis for $TRAPLX$ - transfer char. @ $t = 1e5$ s [4] 42
26 Sensitivity analysis for $TRAPGC2$ - transfer char. @ $t = 1e2$ s [4] 43
27 Sensitivity analysis for $TRAPE2$ - transfer char. @ $t = 1e2$ s [4] 43
28 Block diagram of the heuristic Takeda based degradation model [1] 51
29 Comparison to existing data for initial transfer characteristics - linear plot (M13) [4] 56
30 Comparison to existing data for initial transfer characteristics - semilogarithmic plot (M13) [4] 56
31 Comparison to existing data for the relative changes of $g_{m,max}$ due to aging (M13) [4] 57
32 Comparison to existing data for the relative changes of $I_{d,lin}$ due to aging (M13) [4] 57
33 Comparison to existing data for the relative changes of $I_{d,sat}$ due to aging (M13) [4] 58
34 Comparison of relative change of V_{th} @ $V_d = 2.3$ V (M08,M09,MP04) [4] . . 59
35 Comparison of relative change of V_{th} @ $V_d = 3.0$ V (M10,M11,MP05) [4] . . 59
36 Transfer characteristics for various stress times, taken @ $V_d = 0.1$ V - linear plot (M10) [4] 60
37 Transfer characteristics for various stress times, taken @ $V_d = 0.1$ V - semilogarithmic plot (M10) [4] 61
38 Transfer characteristics for various stress times, taken @ $V_d = 1.8$ V - linear plot (M10) [4] 62
39 Transfer characteristics for various stress times, taken @ $V_d = 1.8$ V - semilogarithmic plot (M10) [4] 63
40 Output characteristics for various stress times, taken @ $V_g = 0.6$ V (M10) [4] 64
41 Output characteristics for various stress times, taken @ $V_g = 1.2$ V (M10) [4] 64
42 Output characteristics for various stress times, taken @ $V_g = 1.8$ V (M10) [4] 65
43 g_m for various stress times - extracted from transfer characteristics @ $V_d =$ 0.1 V (M10) [4] 66
44 Relative change of $g_{m,max}$ over time of stress for two stress conditions(M10,MP04) [4] 67

45 Relative change of V_{th} over time of stress for two stress conditions (M10,MP04) [4] . . . 68
46 Relative change of $I_{d,lin}$ and $I_{d,sat}$ over time of stress for two stress conditions (M10,MP04) [4] . . . 69
47 Extrapolation of degradation effect on V_{th} down to operating conditions (M10,M14,MP04,M18) [4] . . . 70
48 Parameter extraction for extrapolation to operating conditions [4] . . . 71
49 Relative change of V_{th} over time of stress for two different W (M11,M12) [4] 72
50 Relative change of V_{th} over time of stress for three different L (MP06,MP07,MP08) [4] . . . 73
51 Relative change of V_{th} over time of stress (top axis) and over accumulated bulk charge (bottom axis) (M10) [4] . . . 74
52 HiSIM2 v3.0.0 model transfer characteristics @ V_d = 0.1 V, V_b = 0 V compared to measurement results (M10) - linear plot [4] . . . 76
53 HiSIM2 v3.0.0 model transfer characteristics @ V_d = 0.1 V, V_b = 0 V compared to measurement results (M10) - semilogarithmic plot [4] . . . 76
54 HiSIM2 v3.0.0 model transfer characteristics @ V_d = 1.8 V, V_b = 0 V compared to measurement results (M10) - linear plot [4] . . . 77
55 HiSIM2 v3.0.0 model transfer characteristics @ V_d = 1.8 V, V_b = 0 V compared to measurement results (M10) - semilogarithmic plot [4] . . . 77
56 HiSIM2 v3.0.0 model output characteristics @ V_g = 0.6 V, V_b = 0 V compared to measurement results (M10) [4] . . . 78
57 HiSIM2 v3.0.0 model output characteristics @ V_g = 1.2 V, V_b = 0 V compared to measurement results (M10) [4] . . . 79
58 HiSIM2 v3.0.0 model output characteristics @ V_g = 1.8 V, V_b = 0 V compared to measurement results (M10) [4] . . . 80
59 HiSIM2 v3.0.0 model relative change of V_{th} over time of stress compared to measurement results (M10) [4] . . . 81
60 HiSIM2 v3.0.0 model relative change of $I_{d,sat}$ over time of stress compared to measurement results (M10) [4] . . . 82
61 HiSIM2 v3.0.0 model relative change of $I_{d,lin}$ over time of stress compared to measurement results (M10) [4] . . . 83
62 HiSIM2 v3.0.0 model relative change of V_{th} over time of stress for two different W (M11,M12) [4] . . . 84
63 HiSIM2 v3.0.0 model relative change of V_{th} over time of stress for three different L compared to measurement results (MP06,MP07,MP08) [4] . . . 85

64 Takeda model transfer characteristics @ $V_d = 0.1$ V, $V_b = 0$ V compared to measurement results (M10) - linear plot [4] 86
65 Takeda model transfer characteristics @ $V_d = 0.1$ V, $V_b = 0$ V compared to measurement results (M10) - semilogarithmic plot [4] 86
66 Takeda model transfer characteristics @ $V_d = 1.8$ V, $V_b = 0$ V compared to measurement results (M10) - linear plot [4] 87
67 Takeda model transfer characteristics @ $V_d = 1.8$ V, $V_b = 0$ V compared to measurement results (M10) - semilogarithmic plot [4] 88
68 Takeda model output characteristics @ $V_g = 0.6$ V, $V_b = 0$ V compared to measurement results (M10) [4] 89
69 Takeda model output characteristics @ $V_g = 1.2$ V, $V_b = 0$ V compared to measurement results (M10) [4] 90
70 Takeda model output characteristics @ $V_g = 1.8$ V, $V_b = 0$ V compared to measurement results (M10) [4] 90
71 Transfer characteristics for various stress times @ $V_d = 0.1$ V, $V_b = -1$ V (M10) [4] 142
72 Transfer characteristics for various stress times @ $V_d = 1.8$ V, $V_b = -1$ V (M10) [4] 142
73 Output characteristics for various stress times @ $V_g = 0.3$ V, $V_b = 0$ V (M10) [4] 143
74 Output characteristics for various stress times @ $V_g = 0.9$ V, $V_b = 0$ V (M10) [4] 143
75 Output characteristics for various stress times @ $V_g = 1.5$ V, $V_b = 0$ V (M10) [4] 144
76 HiSIM2 v3.0.0 model transfer characteristics @ $V_d = 0.1$ V, $V_b = 0$ V compared to measurement results (M10) [4] 145
77 HiSIM2 v3.0.0 model transfer characteristics @ $V_d = 1.8$ V, $V_b = 0$ V compared to measurement results (M10) [4] 146
78 HiSIM2 v3.0.0 model output characteristics @ $V_g = 0.3$ V, $V_b = 0$ V compared to measurement results (M10) [4] 146
79 HiSIM2 v3.0.0 model output characteristics @ $V_g = 0.9$ V, $V_b = 0$ V compared to measurement results (M10) [4] 147
80 HiSIM2 v3.0.0 model output characteristics @ $V_g = 1.5$ V, $V_b = 0$ V compared to measurement results (M10) [4] 147
81 Takeda model output characteristics @ $V_g = 0.3$ V, $V_b = 0$ V compared to measurement results (M10) [4] 148

82 Takeda model output characteristics @ $V_g = 0.9$ V, $V_b = 0$ V compared to measurement results (M10) [4] 148
83 Takeda model output characteristics @ $V_g = 1.5$ V, $V_b = 0$ V compared to measurement results (M10) [4] 149

1 Fundamentals

1.1 MOSFET Physics

The MOSFET (Metall-Oxide-Semiconductor Field-Effect Transistor) is the most commonly build of a transistor used in microelectronics. Field effect transistors are semiconductor devices, in which the current flow is controlled by means of an electric field. This is achieved by building a MOS capacitor, where the metal and the semiconductor are separated by an insulating oxide layer. In a MOSFET the MOS capacitor is used in combination with the control electrode, the so called gate. Below the metalized surface of the electrode usually a polysilicon layer is used as gate material. Compared to monocrystalline silicon, it is a good conductor. Its conductivity is further increased by doping. Usually it is degenerately doped. Polysilion gates replaced the metal ones mostly because it is easier to process them in a silicon technology. High temperature annealing used in the process would melt most metals. Between the polysilicon gate and the semiconductor a thin layer of siliconoxide (SiO_2) serves as insulator. In a n-channel MOSFET (n-MOSFET) the semiconductor substrate is p-doped silicon. Boron or aluminum are typical p-type dopant materials. Figure 1 shows a schematic of an unbiased four terminal n-MOSFET. The metalized electrodes are colored gray. Below the gate electrode there is the polysilicon in red and the SiO_2 in yellow. It sits atop the red colored p-type semiconductor substrate. The remaining three terminals are called bulk, source and drain. The bulk terminal serves as a connection to the semiconductor substrate. Below the contact there is a p^+ implant (dark red). [6]

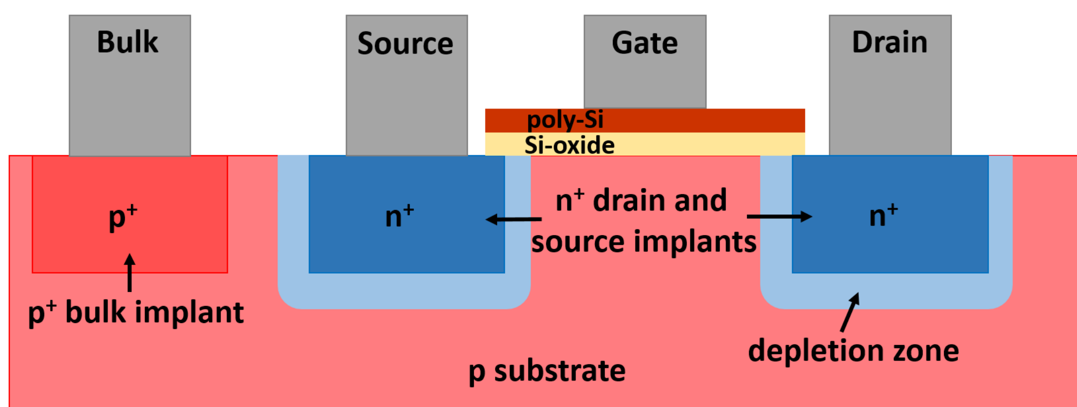

Figure 1: Schematic of an unbiased n-MOSFET [1]

When the MOSFET is switched on, a current from drain to source is established by applying a voltage between those terminals. The current can be controlled by another voltage applied to the gate. That is the basic functionality of the device. Below the source and drain contacts there are n^+ implants (dark blue in Figure 1). They are manufactured by letting a n-type dopant diffuse into the substrate. Typical n-type dopant materials are phosphorus or arsenic. Around the implants, at the junction between the n^+-doped area and the p-doped substrate, a depletion zone forms (light blue).[6]

A pn-junction is the interface between a n-doped and a p-doped semiconductor. In the n-doped part electrons are the majority carriers, while in the p-doped part holes serve as majority carriers. At the junction electrons tend to diffuse into the p-side. By doing so, they leave behind positively charged and immobile donor atoms at the n-side of the junction. The same happens to the holes on the p-side. They diffuse through the junction to the n-side, while leaving behind negatively charged immobile acceptor atoms. The area close to the interface loses its charge neutrality and a space charge region, also called depletion layer, is formed. An electric field is generated in the space charge region, which opposes the diffusion process. An equilibrium is obtained at a certain width of the depletion layer. If the doping concentration is the same, the depletion layer is symmetric around the junction. But in the case of the MOSFET, where the source and drain implants are higher doped, the depletion zone spreads wider into the p-doped side. [7]

There are MOSFET designs with only three terminals, where the bulk is connected to the source terminal. But in integrated circuits the four terminal design is mostly used. Since a vast amount of devices is grown on the same bulk, a connection with the bulk would directly connect all the source terminals. Shorting the source contacts like that is not desirable.

1.1.1 Operating Regimes of the MOSFET

The MOSFET device can be operated in different modes by applying voltages on the gate and drain contacts. An important device parameter is the threshold voltage V_{th}. As long as the applied gate voltage is smaller than the threshold voltage, the device is in the subthreshold regime. For a n-MOSFET V_{th} is positive. Equation 1 shows how V_{th} can be calculated. [8]

$$V_{th} = V_{fb} + 2 \cdot V_{ib} + \frac{\sqrt{2 \cdot \epsilon_S \cdot e \cdot N_a \cdot (2 \cdot V_{ib} + V_{SB})}}{C_{ox}} \quad (1)$$

V_{th}	Threshold voltage [V]	V_{ib}	Interface-bulk voltage [V]
V_{fb}	Flatband voltage [V]	e	Elementary charge [As]
N_a	Acceptor density $[\frac{1}{\mathrm{m}^3}]$	V_{SB}	Applied bulk voltage [V]
C_{ox}	Oxide capacitance [F]	ϵ_S	Permittivity of silicon $[\frac{[\mathrm{F}]}{\mathrm{m}}]$

Figure 2 shows a schematic of a n-MOSFET in the subthreshold regime. The applied gate voltage is positive and smaller than Vth. An electric field from gate to bulk is established. The holes in the p-doped bulk are pushed away from the oxide interface and a depletion zone is formed. The immobile, ionized acceptor atoms stay behind. They are negatively charged. While the MOSFET is in the subthreshold regime, there is only a very small subthreshold current flowing, when a drain voltage is applied. This is due to electrons, which are able to diffuse through the depletion zone towards the drain implant. By decreasing the gate voltage further, the depletion zone is diminished and the subthreshold current vanishes. The device is turned off. [6]

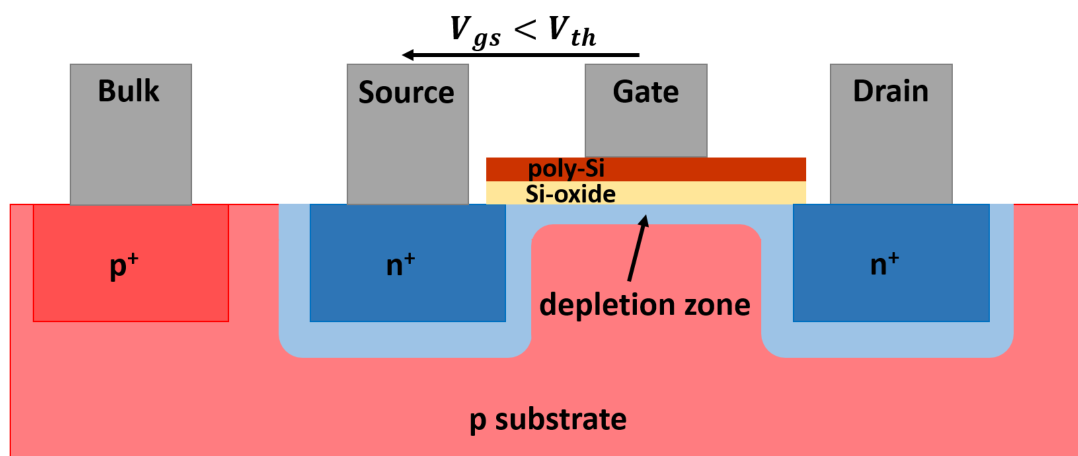

Figure 2: Schematic of a n-MOSFET in the subthreshold region [1]

To turn the MOSFET on, the gate voltage is increased. Once it equals V_{th}, the device is in inversion. The electric field from gate to bulk is increased and additionally to the repulsion of the holes, electrons are accumulated at the bulk-oxide interface. Those mobile carriers form a conductive channel below the interface. The channel reaches all the way from the source implant to the drain implant. If now a voltage between drain and source is applied, the mobile electrons can contribute to a current flowing from drain to source. [6] While $V_d < V_g - V_{th}$, the device is in the linear regime. The channel shows ohmic behavior and an increase in V_d leads to an increase in the drain current I_d. Equation 2 gives an approximated formula for the drain current. [9]

$$I_d = \frac{W}{L} \cdot \mu_n \cdot C_{ox} \cdot \left((V_g - V_{th}) \cdot V_d - \frac{V_d^2}{2} \right) \tag{2}$$

I_d	Drain current [A]	W	Device width [m]
L	Channel length [m]	μ_n	Electron mobility $[\frac{\mathrm{m}^2}{\mathrm{Vs}}]$

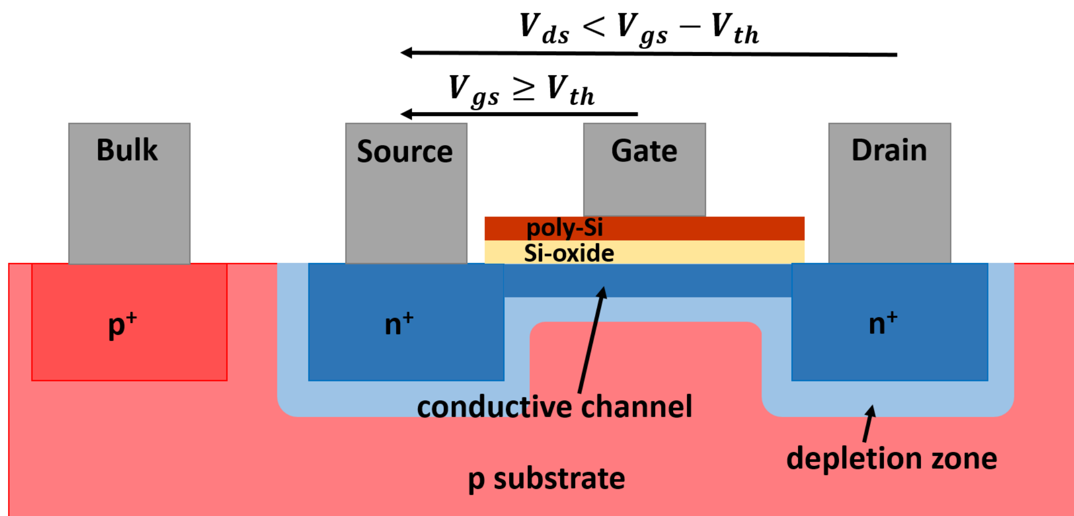

Figure 3: Schematic of a n-MOSFET in the linear regime [1]

When the drain voltage is now increased further until $V_d = V_g - V_{th}$ an additional effect occurs. The inverted channel is pinched off close to the drain implant. Due to the higher drain voltage, a small depletion layer between the channel and the drain implant is formed. Figure 4 shows a schematic of a n-MOSFET with a pinched off channel.

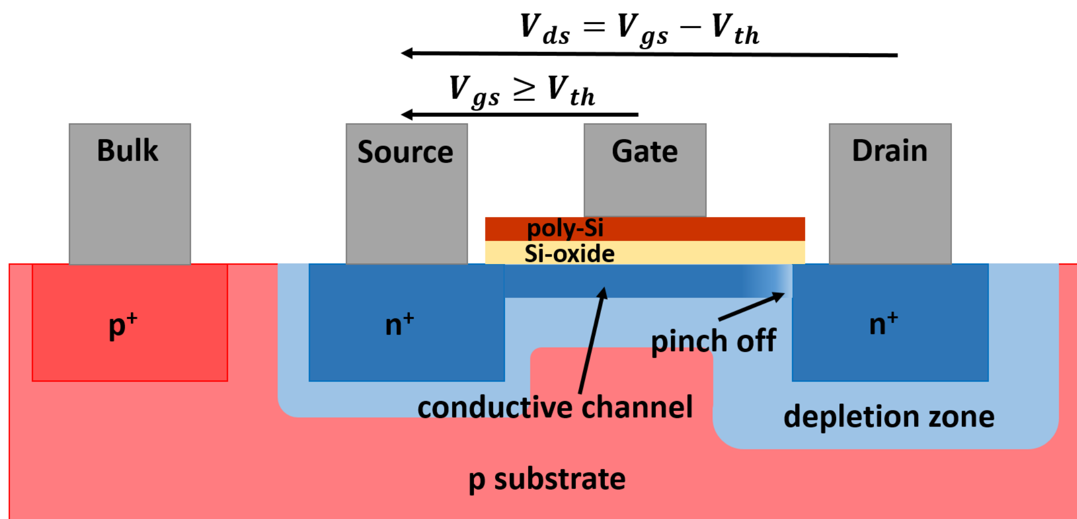

Figure 4: Schematic of a n-MOSFET at the point of pinch-off [1]

At the point of pinch-off the MOSFET has reached the saturation regime. In this operation mode the response of the device to the applied drain voltage changes. Despite the presence of the depleted pinch-off region, there is still a current flowing from drain to source. The electrons can pass through the depletion zone at the drain interface. But when the drain voltage is further increased to $V_d > V_g - V_{th}$, the drain current does not increase any more. It saturates. Though the drift current in the channel is increasing, also the depleted pinch-off region becomes wider and reduces the current. An equilibrium state is reached. By inserting $V_d = V_g - V_{th}$ into Equation 2, the saturation drain current can be calculated with the resulting Equation 3. [9]

$$I_{sat} = \frac{W}{2 \cdot L} \cdot \mu_n \cdot C_{ox} \cdot (V_g - V_{th})^2 \tag{3}$$

I_{sat} Drain current in saturation [A]

I_{sat} is not dependent on V_d any more. It is controlled by the gate voltage V_g. The transistor acts as a voltage controlled current source. Figure 5 shows the MOSFET at further increased drain voltage. Additionally the electric field in the pinch-off region is indicated. Due to the depletion zone, very high electric fields are present from the drain implant to the inverted channel.

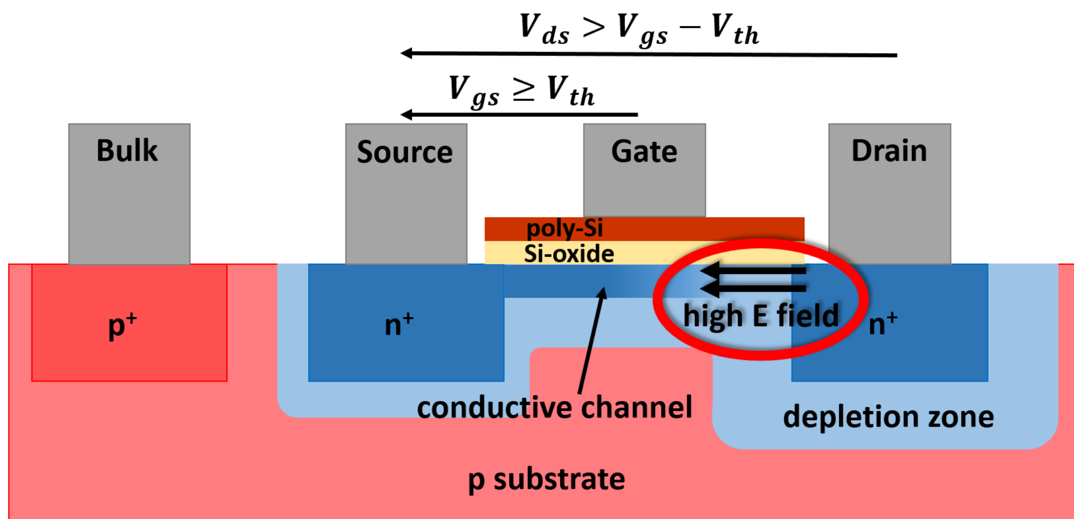

Figure 5: Schematic of a n-MOSFET in the saturation regime (with indication of the high electric fields) [1]

1.2 Hot-Carrier-Injection (HCI) Effect in MOSFETs

1.2.1 Hot Carriers

The high electric field in the pinch-off region of a MOSFET in saturation is the main reason for the appearance of hot carriers in the device. Carriers traveling through the high field are accelerated. By that, they reach energies, which are well beyond the equilibrium thermal energy in the semiconductor. That is why those fast carriers are called "hot". Hot carrier injection in n-MOSFET devices is due to electrons being accelerated in close proximity to the drain implant. [10]

1.2.2 Hot Electron Transport Theory

Theoretically ballistic transport of electrons over very long distances would be possible if certain conditions were fulfilled: The device would need to be at very low temperatures to make sure scattering is mainly due to material impurities. Also the amount of those impurities would need to be very low to increase the mean free path. The third condition would be that only a very small voltage is applied across the device. Even if the first two conditions could be fulfilled, it is necessary to apply a certain voltage to gain some output. That voltage is enough to heat up the electrons and lead to a decrease in the mean free path. Therefore in realistic devices it is necessary to consider hot ballistic transport. [11]

The two most important inelastic scattering effects for hot electrons are interactions with cold electrons and plasmons. When hitting a cold electron the hot electron transfers a part of its excess energy. By that the cold electron is excited into the conduction band. A new electron-hole pair is created. Thereby the scattering cross section increases with the incident electron energy. For the scattering on plasmons, there is a minimum energy below which no scattering occurs. [12]

1.2.3 Degradation Effects

The hot electrons themselves are not an issue for the device reliability. Problems arise from the effects, which are introduced by those electrons. When they are high enough in energy, the electrons contribute to two different mechanisms near the oxide interface.

The bandgap energy in silicon is $E_g = 1.12$ eV. When the accelerated electrons exceed the bandgap energy and reach energy levels in between 1.3 eV and 1.8 eV, they can be part of an effect called impact ionization. A hot electron creates an electron-hole pair by promoting an electron from the valence band into the conduction band. The hot electron loses the energy that is necessary for this process. If it has enough energy, the hot electron can even produce several electron hole pairs. The promoted electron also feels the electric field and is accelerated. If it becomes fast enough, it can take part in impact ionization as well. At some point it reaches the drain implant and contributes to the drain current. The hole on the other hand is transported out of the depletion region into the p-doped bulk. It leaves the device through the bulk contact and contributes to the bulk current. [10] [13]

The second effect is crucial regarding device degradation. If a hot electron has enough energy and its momentum is directed towards the bulk-oxide interface, it can be injected into the oxide. The energy barrier for the injection is 3.1 eV. This is much lower than the energy barrier for hole injection with 4.8 eV. Therefore hole injection is far less probable and is not considered a major problem. Some of the injected electrons may contribute to the gate current. Some, on the other hand get trapped at defect states in the gate oxide. Those trap states are energetically favorable for the electrons since they are low in energy. Detrapping is possible but not likely and they are called deep trap states. [13]

Additionally to the injection into the oxide, hot carriers can create interface trap states at the bulk-oxide interface. Interface trap states are usually dangling silicon bonds, which can be filled by an electron. To avoid trap states, hydrogen is used during processing to passivate those dangling bonds. The dissociation energy for a Si-H bond is ≈ 3 eV. When the kinetic energy of the hot electron exceeds the dissociation energy, it can break the

Si-H bond. The open silicon bond then acts as a trap for electrons. The density of the introduced interface trap states is highest in the pinch-off region, where the electrons have the highest energy. Interface traps are also called shallow traps. The probability for captured electrons to detrap is higher than for deep trap states. [14]

The schematic in Figure 6 provides a visualization of the described effects. It shows the pinch-off region of a n-MOSFET close to the drain implant with an indication of the high electric field. The accelerated "hot" electrons either take part in impact ionization or the generation of trap states at the bulk-oxide interface and inside the oxide.

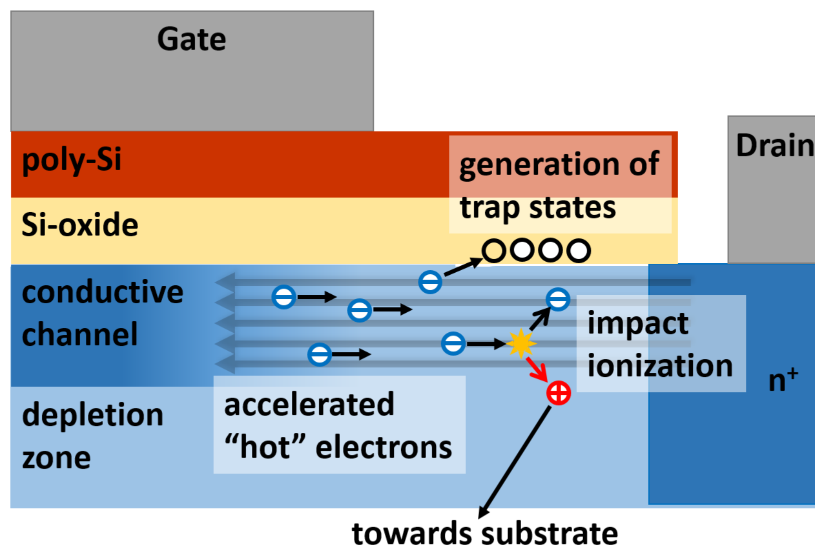

Figure 6: Detailed schematic of degradation effects due to hot electrons in a n-MOSFET [1]

Device degradation is an issue due to the trapped charges. Those are diminishing the impact of the vertical electric field from gate to bulk and therefore modulate the surface potential and the carrier mobility at the bulk-oxide interface. The current characteristics of the MOSFET are altered and the operating lifetime is reduced. Important device parameters change, e.g. the threshold voltage is shifted or the saturation current is decreased.

1.2.4 Visualization of HCI Effects with Technology CAD Simulations

A set of Technology Computer Aided Design (TCAD) simulations was produced to visualize the above mentioned degradation effects. The simulations below where kindly provided by the TCAD & Device R&D department at ams AG in Premstätten. In TCAD

simulations discretization approaches, like the finite element method, are used to calculate the charge transport in semiconductor devices.

The simulated device is a n-MOSFET with a channel length of $L = 0.18$ μm. The voltages applied are $V_g = 0.85$ V and $V_d = 2.3$ V. That is the same stress as was used for some of the conducted measurements (cf. Section 2). Due to ams AG company policies it is not allowed to show legends to the graphs. In all the figures a red coloring stands for high values of the simulated physical quantities, while blue parts indicate low values.

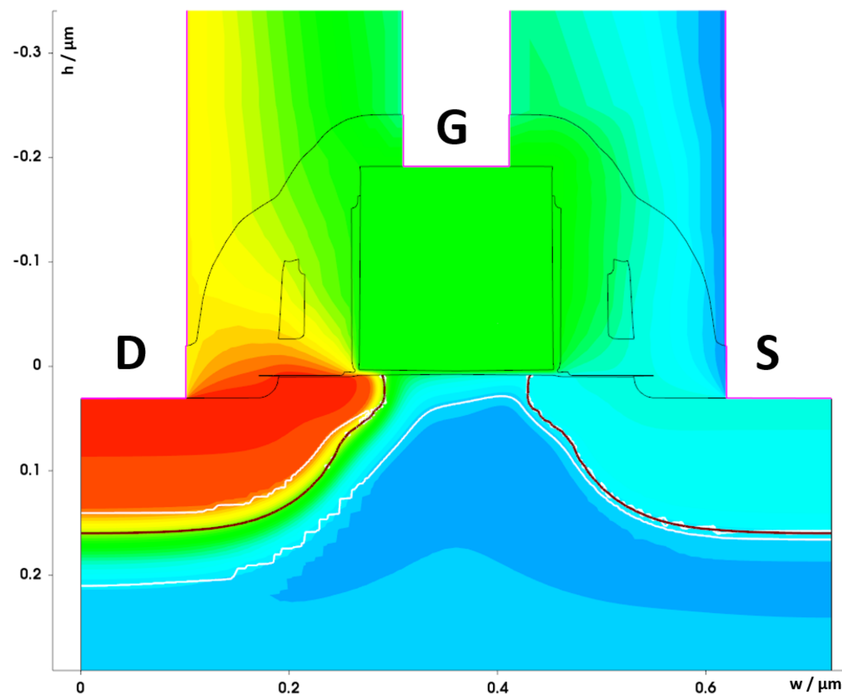

Figure 7: TCAD simulation of the potential distribution in a n-MOSFET @ $V_g = 0.85$ V, $V_d = 2.3$ V [2]

Figure 7 shows the potential distribution in the device. The red area at the drain symbolizes the applied drain voltage of $V_d = 2.3$ V and the green area at the gate the applied $V_g = 0.85$ V. The source contact is grounded. The thin white lines near the bulk-drain, the bulk-oxide and the bulk-source interfaces indicate the extend of the depletion zone. Since the terminal implants are higher doped than the bulk, the major part of the depletion zone is in the bulk. The p-n junctions at the gate and drain interfaces are reverse biased due to the applied stress voltage. The width of the depletion zone in this regions is determined by the amount of applied voltage.

Figure 8 gives a view of the simulated electric field density in x-direction (parallel to

the channel) at the drain-bulk interface. Due to the high applied voltages the device is in saturation and the channel is pinched off at the drain side. Because of the pinch-off, high electric fields build up in this region. The bigger red area marks the high field density, where electrons are accelerated towards the drain implant. Due to their kinetic energy, they are called hot electrons.

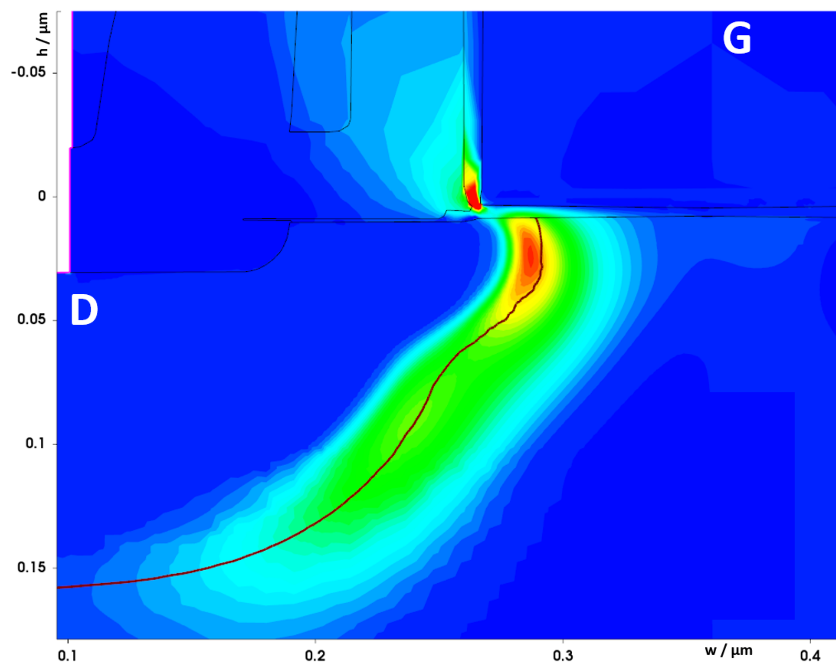

Figure 8: TCAD simulation of the electric field density in x-direction in a n-MOSFET @ $V_g = 0.85$ V, $V_d = 2.3$ V [2]

A consequence of the high field strength at the drain-bulk interface in the region of the pinched off channel is shown in Figure 9. It shows the impact ionization rate in the device. While in most parts of the n-MOSFET the rate is low (dark blue color), it suddenly rises at the depletion zone near the drain. The highest impact ionization rate is localized at the same spot as the peak in electric field density in Figure 8. As described above, the hot electrons either produce trap states at the oxide interface or lose their energy by producing electron hole pairs per impact ionization. Compared to the yellow areas, the rate in the red spot is about five to six orders of magnitude higher.

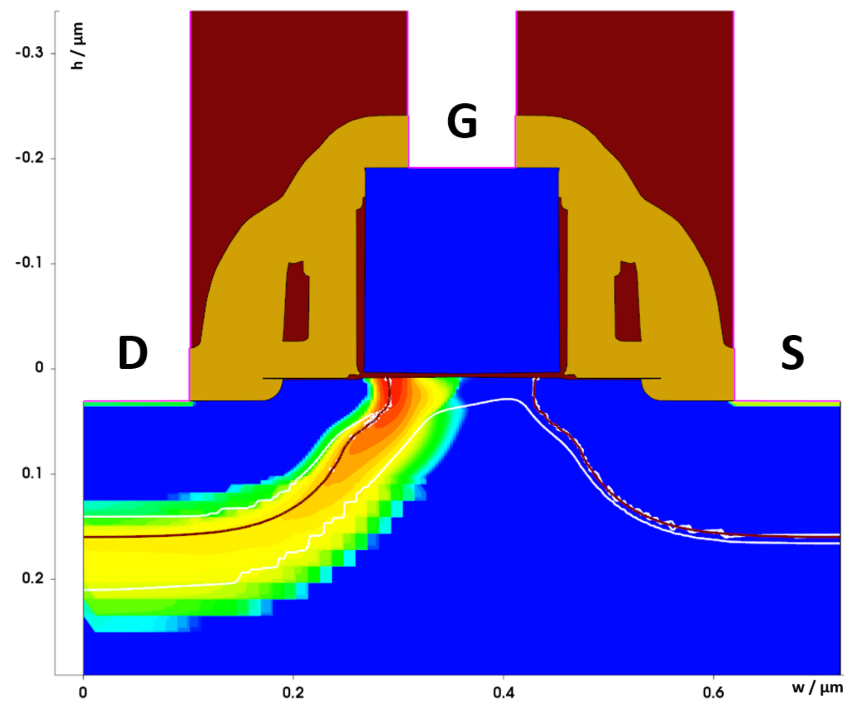

Figure 9: TCAD simulation of the impact ionization rate in a n-MOSFET @ $V_g = 0.85$ V, $V_d = 2.3$ V [2]

Due to impact ionization electron hole pairs are produced. The electrons mostly contribute to the drain current or fill trap states at the oxide interface. The holes on the other hand are sucked out into the bulk and leave the semiconductor as hole current via the bulk contact. To visualize that, the hole current density of the device is simulated in Figure 10. It can be seen that the hole current originates directly at the spot of maximum impact ionization due to the high fields. Almost all of the holes are produced in the pinched-off channel area close to the drain. After leaving the depletion zone they are equally distributed in the bulk semiconductor. This is the case because for the simulation the backside of the substrate is used as bulk contact. In other device geometries, where the bulk contact is behind the source contact, the hole current would flow in that direction.

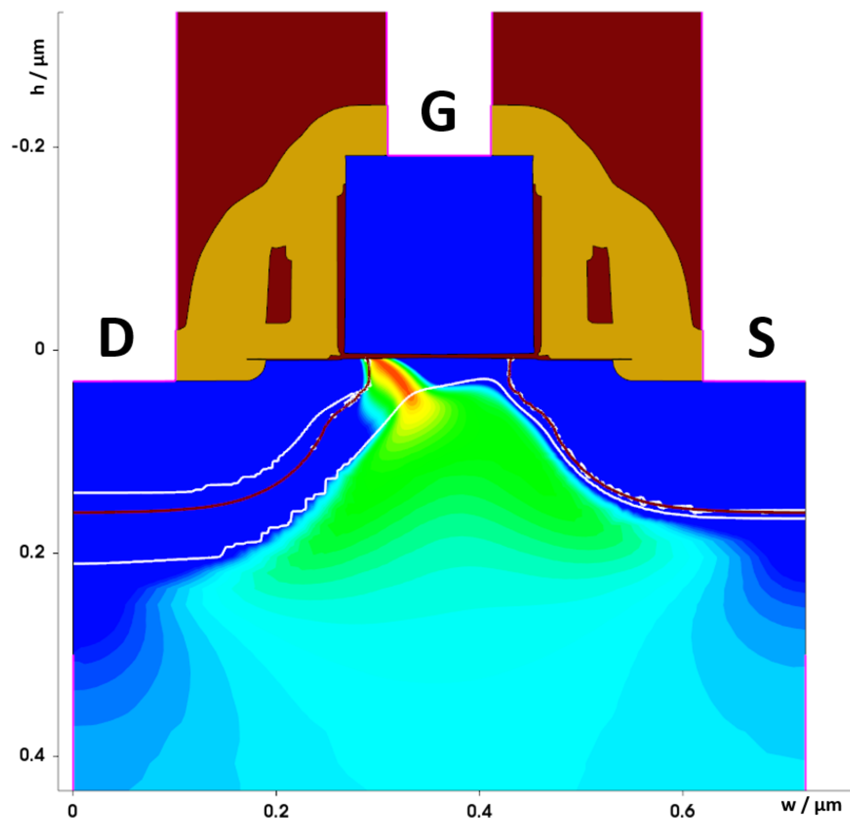

Figure 10: TCAD simulation of the hole current density in a n-MOSFET @ $V_g = 0.85$ V, $V_d = 2.3$ V [2]

The series of TCAD simulations above show how the electric field density in a n-MOSFET in the saturation regime is directly connected to the hole current density. This behavior is important for the degradation simulation with the HiSIM2 aging model. Details on the approach may be found below in Chapter 3.

2 Experimental Realization

For all the measurements, devices from an available MPW (Multi Project Wafer) from ams AG were used. This MPW contained multiple test structures of NFET devices with a short channel length. They are grouped in SLMs (Scribe Line Monitors) for width and length variation. Figure 11 shows a wafer map for the MPW.

Figure 11: Wafer map (red dies have been cut for packaging) [3]

The wafer map orientation in Figure 11 is notch to the east. Sections with identical content and shape on the wafer are called dies. In the map all the dies have continuous numbers from 1 to 58. The red dies (47-58) have been cut, so they could get packaged for long term stressing with a DC supply (cf. Section 2.3). The blue dies have been available for on-wafer measurements (cf. Section 2.2).

The primary investigated device for this thesis was a NFET with width $W = 10\ \mu$m and length $L = 0.18\ \mu$m positioned on SLM 001A. Since degradation effects become more severe with decreasing channel length, the focus was laid on those short channel devices. Also devices with variations in channel length and width were available for measurement. For all measured devices the operating voltage was $V_{dd} = 1.8$ V.

The idea behind the experiment was to build a simple alternating sequence of measurements and DC stress. First an initial measurement was done for the unstressed device. Afterwards the device was stressed for a certain amount of time under predefined

conditions. Then another measurement was conducted to see the changes due to stress. This sequence was repeated with increasing stress times in between. To gain sufficient data for the description of the device a total amount of 10 different characteristics was recorded at every measurement point. For those four different transfer characteristics and six different output characteristics all terminal currents were monitored. A transfer characteristic is obtained by plotting the drain current I_d over the gate voltage V_g for a constant drain voltage V_d. An output characteristic on the other hand is a plot of I_d over V_d for a constant V_g. Usually a whole field of output characteristics is measured by varying the applied gate voltage.

2.1 Device Properties

The devices were produced with an ams AG 0.18 μm HV-CMOS process technology called aH18. They are on the lower end of the scalable architecture with operating voltages from 1.8 V up to 50 V. The device is built with 3-6 metal layers and one additional power metal layer on top. The process is manufactured in a 7 inch fabrication line and shows a gate density of approximately 120 000 mm^{-2}. Some of the possible applications for this process are power management, motor control, printer drivers, DC/DC converters, switched power supplies and LCD drivers. For the aH18 process also a complete set of SPICE models based on numerous measurements is available. [15]

Figure 12 shows a TCAD (Technology Computer Aided Design) simulation of a nFET with a channel length of $L = 0.18$ μm. It gives information about the densities of doping species in the silicon. A quantitative legend or further information on the dopant species is not displayed due to ams AG company policies. The reddish and yellow colors of the drain and source implant indicate the n-doped areas. There the net doping decreases towards the bulk. This transition helps to smoothen jumps in the electric field. Without it the hot carrier effect would be even bigger due to higher fields from channel to drain. The blue colored regions are p-doped. For a nFET this is the bulk. Again there are different dopant densities. Darker areas are higher doped. The two dark blue spots near drain and source implant are called halo implants. Those are used to confine the depletion region near the terminal implants. For short channel devices this is necessary to prevent various undesirable effects. By reducing charge sharing effects it decreases threshold voltage degradation due to the reduced channel length. Also it reduces drain induced barrier lowering (DIBL) and the possibility of a punchthrough because the distance between the

source and drain depletion region is increased. [16]

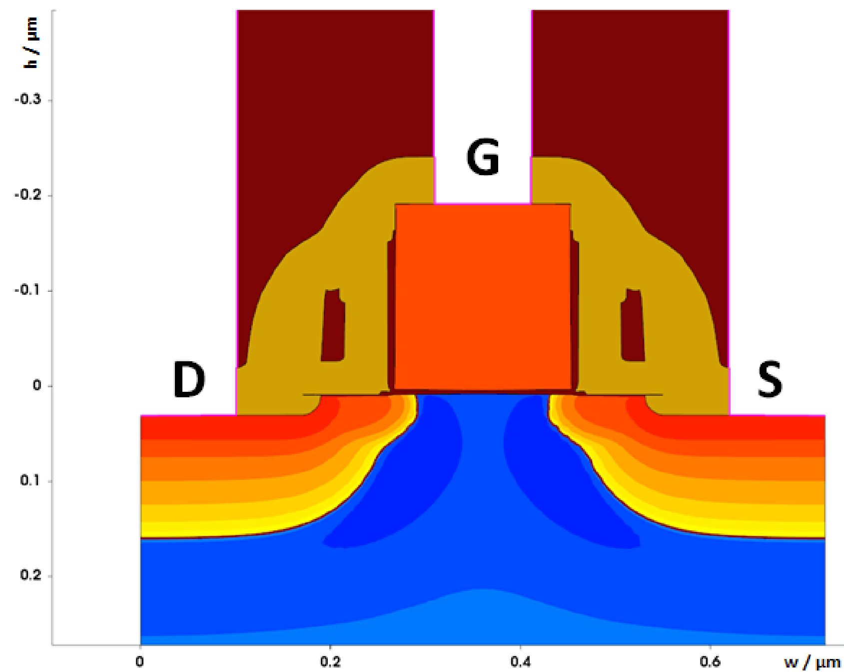

Figure 12: Net doping map of standard n-MOSFET device - TCAD simulation [2]

2.2 On-Wafer Measurements

Most of the measurements were done directly on the wafer by using a probing station to contact the devices. A further preparation of the wafer was not necessary. To allow long stressing times, the measurements were mostly conducted during nights and weekends with breaks in between. In the end all collected measurement data for one device was combined to a test series. This was only possible because the recovery effect of the devices was insignificant when stored at room temperature. Table 1 shows a complete list of the various on-wafer test series. For every series a new, unstressed device was used. Every SLM contains multiple functional devices. Therefore it is possible that the SLM name and the die number are the same for different measurement series. The names of the series will be used throughout the thesis to refer to Table 1.

Table 1: List of conducted on-wafer measurement series

Name .... Name and order of the measurement series
Type .... Device type
W .... Channel width
L .... Channel length
V_g .... Applied DC stress voltage on gate terminal
V_d .... Applied DC stress voltage on drain terminal
t_{stress} .... Overall stressing time of device
SLM .... Name of SLM containing the device
Die .... Number of die containing the device (cf. wafer map in Figure 11)

Name	Type	W / μm	L / μm	V_g / V	V_d / V	t_{stress} / s	SLM	Die
M01	NFET	10	0.18	1.3	1.98/2.3	815 800	001A	26
M02	NFET	10	0.18	0.85	2.3	100 000	001A	25
M03	NFET	10	0.18	0.85	2.3	40 000	001A	33
M04	NFET	10	0.18	0.85	var.	65 000	001A	25
M05	NFET	10	0.18	0.85	2.3	50 000	001A	34
M06	NFET	10	0.18	0.85	2.3	270 000	001A	14
M07	NFET	10	0.18	0.85	2.3	280 000	001A	11
M08	NFET	10	0.18	0.85	2.3	520 000	001A	10
M09	NFET	10	0.18	0.85	2.3	730 000	001A	9
M10	NFET	10	0.18	0.85	3.0	620 000	001A	24
M11	NFET	10	0.18	0.85	3.0	425 000	001A	35
M12	NFET	1	0.18	0.85	3.0	230 000	001A	35
M13	NFET	1	0.18	1.75	2.4	100 000	001A	35
M14	NFET	10	0.18	0.85	2.7	360 000	001A	8
M15	NFET	10	0.18	0.85	1.8	148 170	001A	16
M16	NFET	10	0.18	0.85	2.7	380 000	001A	23
M17	NFET	10	0.18	0.85	2.1	380 000	001A	19
M18	NFET	10	0.18	0.85	2.1	320 000	001A	7

2.2.1 Measurement Equipment

All the on-wafer measurements were conducted in an engineering laboratory at ams AG in Unterpremstätten. The wafer was loaded into a probing station (Süss Mictro Tech PM8) and fixed on the chuck by means of several vacuum outlets. Contacting the device was done with four manually adjustable needles. The needles were directly connected to a parameter analyzer with triax cables (Trompeter Electronics Triax TRC/50/2 410). Two calibrated parameter analyzers (HP 4155B Semiconductor Parameter Analyzer) were available for the measurements. The accuracy of the analyzer's SMUs (Source Measure Units) for the applied voltage was $\pm(0.02\ \% + 3\ \text{mV})$ in the used range of ± 20 V. The

measurement accuracy for the maximum measured currents in the range of ±10 mA was $\pm(0.04\ \% + 2\ \mu\text{A} + V_{out} \cdot 10\ \text{nA})$ with a resolution of 1 nA [17]. A connection via a GPIB cable was established between the analyzer and a computer to run the scripts for stress and measurement.

There are several delicate challenges to be looked after with this kind of measurement. The most important one is the handling of the needles. It is necessary to make full contact to the metalized pads on the wafer. This results in scratches on the surface, which should be plainly visible under the microscope. High contact resistances, due to bad contacting, can lead to undesired voltage drops which reduce the stress applied to the terminals. But applying to much pressure with the needles on the pads can also lead to a device failure when other layers and parts of the structure are damaged. Also the needles can break in this case. Besides the sensitive handling of the needles it is necessary to check them for metal flakes left behind on the tips. Those can increase the resistance and lead to a bad contact.

For long term measurements over several hours or days another issue is the consistency of the ambiance conditions. The used chuck had no possibility to cool or heat so the measurements were done at room temperature of 21°C. It was held constant by the air conditioning system of the laboratory. The chamber of the probing station was completely darkened and vibration isolation was used to assure stable conditions.

2.2.2 Measurement Script

The whole sequence of measurements and stress was executed from one script file. It is an EXCEL® file including a custom-built add-on for aging measurements. The file is not displayed in this thesis since it is intellectual property of ams AG. Inside the script a HP specific programing language was used to communicate with the analyzer via the GPIB. Inside the script all the measurement conditions were specified as well as the stress voltages and the stress time. Upon starting a measurement a result section inside the .xls file was created in which all the applied voltages and measured currents were stored.

2.2.3 Data Extraction Routine

The extraction routine for the measured data was developed in MATLAB® R2015a (8.5.0.197613). Since the EXCEL® control script stored the data as .xls files, it was necessary to change the file type. For easy access with different evaluation tools the format .csv was chosen. Besides changing the file format, the extraction routine made

important data accessible by combining it into a summary file. This step included the calculation of several important output quantities (cf. Section 2.4). Also summary files for all the characteristics were written and a folder substructure including all the single measurements was built. Therefore for any kind of evaluation the necessary data was already available.

Another issue was the stacking of adjacent measurements which were conducted with a pause in between. The second result file needed to begin right at the end of the first one. With some simple inputs the time stamps of the measurements were altered to display the overall stress time. In the end the data was appended to the summary file and all the partial measurements could be displayed as one single stressing measurement series. The whole extraction routine is included in the Appendix 9.1. For better understanding the code contains comments by the author.

For the evaluation of parameter degradation it was convenient to have the absolute and relative differences to the initial values of the unstressed device available. Another MATLAB® routine was written, which took the created summary file and calculated those differences. The results were stored in two extra summary files for further use. The commented script for this routine is included in the Appendix 9.2.

2.3 Measurements of Packaged Devices

Besides the on-wafer measurements, a second way for the gathering of data was implemented. Twelve of the devices were packaged for easier access to long term stressing data. The advantage of the packaged devices was, that no probing station was needed for stressing them. It was sufficient to connect the package to a DC supply. They therefore were easier to work with when measurement resources were scarce. But for the extraction of the characteristics again a parameter analyzer was used. Table 2 shows the measurement series done with the packaged devices. Unfortunately the devices could not be assigned to their former position on the wafer. They were given successive numbers to make distinction possible. Again due to the availability of multiple devices per SLM, the SLM names and die numbers can be the identical for various measurement series. The names of the series will be used throughout the thesis to refer to Table 1.

Table 2: List of conducted measurements of packaged devices

Name .... Name and order of the measurement series
Type .... Device type
W .... Channel width
L .... Channel length
V_g .... Applied DC stress voltage on gate terminal
V_d .... Applied DC stress voltage on drain terminal
t_{stress} .... Overall stressing time of device
SLM .... Name of SLM containing the device
Nr. .... Number given to the packaged device

Name	Type	W / μm	L / μm	V_g / V	V_d / V	t_{stress} / s	SLM	Nr.
MP01	NFET	10	0.18	0.85	2.3	7 200	001A	2
MP02	NFET	10	0.18	0.85	2.3	45 000	001A	4
MP03	NFET	10	0.18	0.85	2.3	710 100	001A	5
MP04	NFET	10	0.18	0.85	2.3	1 160 400	001A	6
MP05	NFET	10	0.18	0.85	3.0	381 200	001A	7
MP06	NFET	1	0.18	0.85	3.0	371 000	003A	1
MP07	NFET	1	0.40	0.85	3.0	393 500	003A	1
MP08	NFET	1	0.90	0.85	3.0	280 000	003A	1

2.3.1 Measurement Equipment

The packaged devices did not need a probing station for being measured. Since those were not available all the time, this was a big advantage. A standard 24-pin package was used. For connected the packages a metal box (Culus Listed Type 4/4x Enclosure) with a 24-pin socket (TEXTOOL), a small board, four BNC connectors and some copper wires was used. The material was provided by the Institute of Solid State Physics at Graz University of Technology. Figure 13 shows the inside of the assembled box. In this design the BNC connectors could be freely connected to all of the pins.

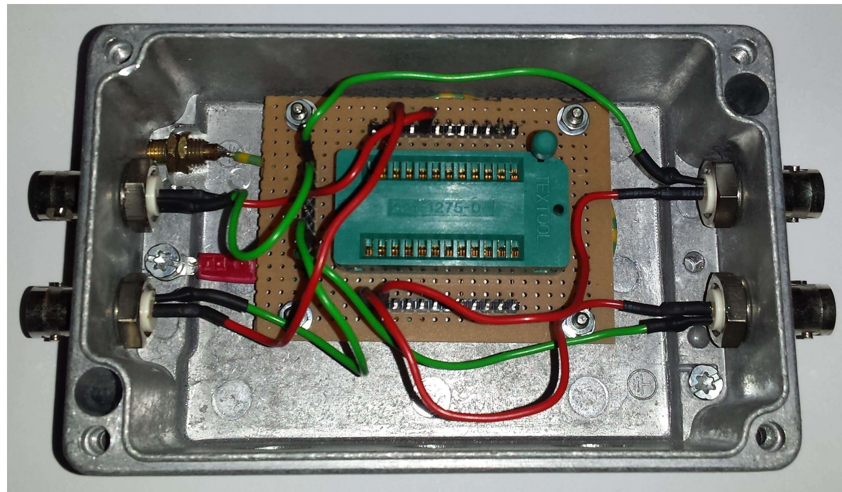

Figure 13: Photograph of measurement box with 24-pin socket (without lid)

For the measurement of the device characteristics only a parameter analyzer and a computer to run the script were needed. For the connection from the box to the SMUs BNC cables, BNC-triax connectors and triax cables were used. A further advantage was that for just stressing the packaged device not even an analyzer was needed. The box was directly connected to a DC-supply (Keithley 2231A-30-3 Triple Channel DC power supply) via BNC cables and BNC-banana connectors. Therefore parallel stressing of a packaged device and one on the wafer was possible with only one analyzer available.

Again the measurements were conducted in a laboratory at ams AG under constant ambiance conditions. Another challenge with this setup was to assure a proper connection to the pins inside the box. The applied stress voltage was checked with a voltmeter (Voltcraft VC140) directly at the package to assure that there was a good connection.

2.3.2 Measurement Script

To control the parameter analyzer the exact same EXCEL® script as for the on-wafer measurements was used (cf. Section 2.2.2). The advantage was that the collected data was stored in the same way, which made extraction easier.

2.3.3 Data Extraction Routine

Since the data was gathered with the same parameter analyzer control script, the same MATLAB® routine was used for both kinds of measurements (cf. 2.2.3). This was important to make direct comparison possible and convenient.

2.4 Important Parameters for Analysis

After the measurement and the data collection a few important parameters were extracted from the measured characteristics. Those are usually used to describe the properties of MOSFETs and are also important outputs from simulations. In the experiment the parameters were used to trace the changes in the devices due to the long term exposure to DC stress.

- **Saturation current $I_{d,sat}$:**
  It is the maximum current value in a transfer characteristic in saturation mode ($V_d = V_{dd} = 1.8$ V). In a functional device the current should increase with increasing V_g and the maximum is the last point of the characteristic (at $V_g = V_{dd} = 1.8$ V). The drain current exhibits a quadratic dependence on $(V_g - V_{th})^2$ (cf. Equation 3).

- **Linear current $I_{d,lin}$:**
  It is the maximum current value in a transfer characteristic in linear mode ($V_d = 0.1$ V). Again the current should increase with increasing V_g which makes the maximum the last point of the characteristic (at $V_g = V_{dd} = 1.8$ V).

- **Maximum transconductance $g_{m,max}$:**
  The transconductance is the derivative of the drain current I_d with respect to the gate voltage V_g. That is the slope of the transfer characteristic. For finding $g_{m,max}$ the linear transfer characteristic is differentiated and the maximum value is taken.

- **Threshold voltage V_{th}:**
  The threshold voltage is generally defined as the gate voltage at which the transistor starts to conduct. There are different methods of calculating V_{th}. This is an important thing to keep in mind when comparing data and results. For this experiment V_{th} was approximated with the constant current method. A certain value $I_{d,on}$ is used to define when the transistor is in on-state. The voltage according to $I_{d,on}$ in a transfer characteristic in saturation mode is taken as V_{th}. The value for $I_{d,on}$ is calculated using the channel width W and length L with the following equation:

  $$I_{d,on} = \frac{W}{L} \cdot 10^{-06} \tag{4}$$

  Though the constant current method has no direct physical connection to the channel conductivity it is widely used. It provides a simple and fast way for V_{th} extraction. [18]

2.5 Selection of Stress Conditions

It was of importance to use the right set of stress conditions (V_g and V_d) for the experiment. It would have been best to stress the device inside or at least close to standard operating conditions. On the other hand it was necessary to choose conditions which provide a detectable effect after a reasonable period of time. The assumption was that higher stress voltages would accelerate the degradation effects.

For choosing a suitable value for the gate voltage V_g the transconductance g_m of an unstressed device was evaluated. The idea was that at the point of maximum g_m the impact on the device due to stress would also be at a maximum. By considering measurements from the preparation state of the experiment, $g_{m,max}$ was found at $V_g \approx 0.85\ V$. Figure 14 shows an example transconductance curve with an indication of $g_{m,max}$. With increasing stress time also $g_{m,max}$ shifted. This shift was not taken into account for the experiment since it was not possible to do this in-situ. A constant value of $V_g = 0.85$ V was used.

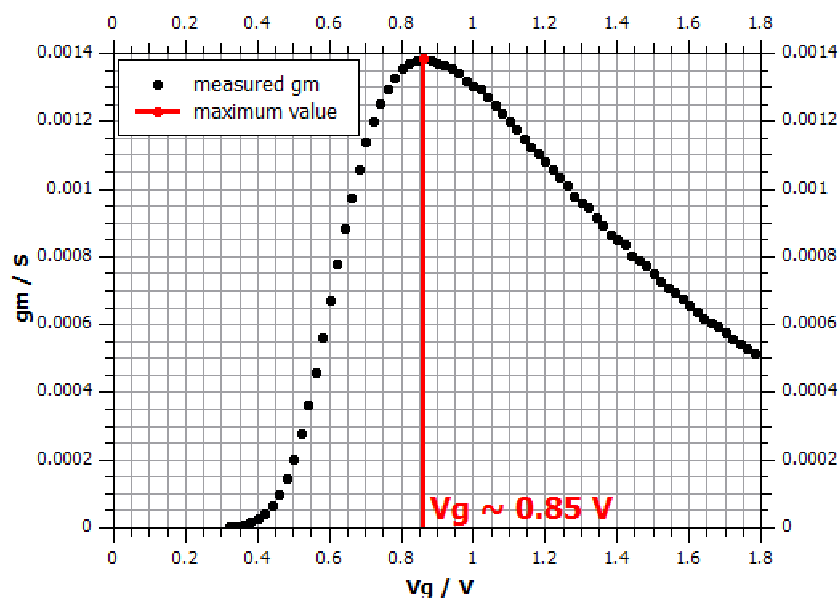

Figure 14: Measured values of transconductance g_m for determination of best stress condition (M04) [4]

The second important stress condition is the drain voltage V_d. Its value for the experiment was chosen empirically. First the operating voltage $V_{dd} = 1.8$ V was used, then V_d was increased to 2.0 V, 2.1 V, 2.3 V and finally to 2.5 V. The change in the parameter V_{th} was monitored. The results of those measurements are displayed in Figure 15.

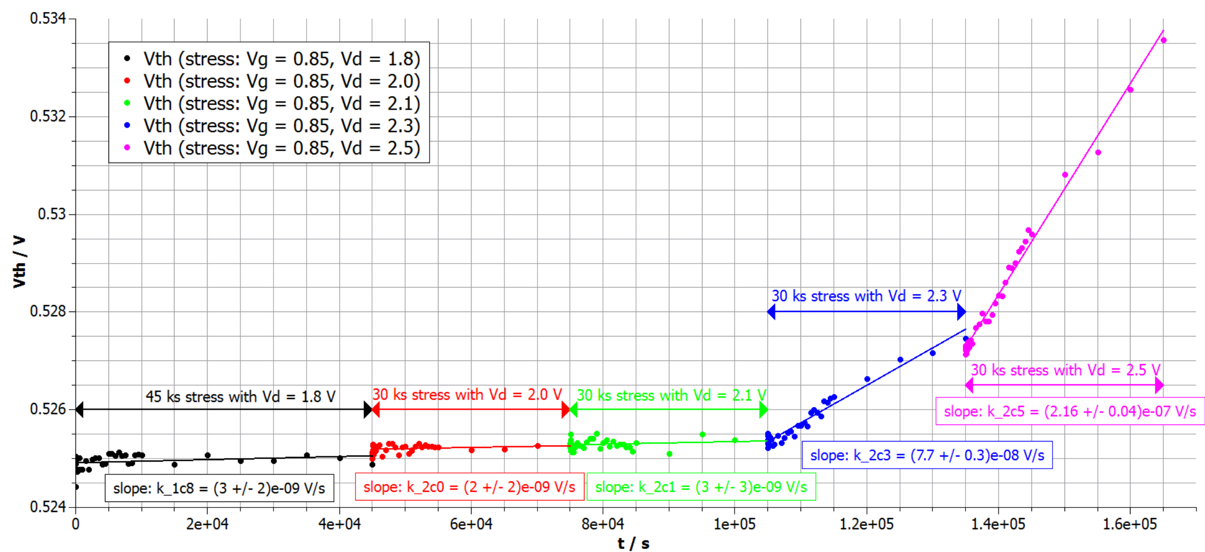

Figure 15: V_{th} as function of stress time for various drain voltages (M04) [4]

The results in Figure 15 suggested that a stress voltage up to $V_d = 2.0$ V would have been to little to see enough impact for a good evaluation. At least not in a reasonable amount of stress time. With a drain voltage of 2.3 V the slope suddenly increased by a factor of about 20. That was used for the standard measurements since it was a good compromise between effect visibility and still being as close as possible to operating voltage. Later on during the experiment also other, mostly higher drain voltages were used. In every measurement the devices were overstressed (V_d higher than V_{dd}).

3 HiSIM2 MOSFET Degradation Model

An important and central part of this thesis is the simulation of the measured devices with the new HiSIM2 3.0.0 Beta version including the dynamic hot-electron degradation model.

3.1 HiSIM2 MOSFET Model

Basically the HiSIM (Hiroshima-university STARC IGFET Model) MOSFET model, like most other models used in industry, is a so called SPICE (Simulation Program with Integrated Circuit Emphasis) model. SPICE is an open source circuit simulator, based on the Kirchhoff equations, which was developed in Berkeley at the University of California in the seventies.

3.1.1 HiSIM2 Model Concept

HiSIM2 is a bulk MOSFET model for circuit simulations written in Verilog A. It was developed by a team from Hiroshima University together with the Semiconductor Technology Academic Research Center (STARC).

The functionality is based on the calculation of the surface potential in the channel, which allows an unified description for all possible bias conditions. This is achieved by solving the Poisson equation for the drain and the source side. The model is physics driven and uses the drift-diffusion approximation. To obtain analytic solutions it is necessary to use the charge-sheet approximation as well as the gradual-channel approximation. The surface potential itself is a function of the voltages on the four MOSFET terminals. Simulation results are found by means of internal iterations inside the MOSFET model as well as global iterations of the whole simulated circuit. The challenge is to keep the device description physically accurate by putting high effort into the calculation of the surface potentials while at the same time keeping the CPU run time as low as possible. Even a small increase in run time for the simulation of a single device, can have a big impact on the simulation time of a complex circuit with many devices in it.

The long-channel basic MOSFET model is already extended with descriptions for short-channel and reverse-short-channel effects. The functionality and accuracy of the model has been tested for devices with a channel length of down to 45 nm. There are also versions for special MOSFET builds like high-voltage devices or double-gate devices available. The latest versions of the model also include extensions for device aging sim-

ulations. The effects in focus are HCI (Hot Carrier Injection) and NBTI (Negative Bias Temperature Instability). [19]

3.1.2 Used Approximations within the HiSIM2 Model

Like mentioned above, several approximations are used inside the HiSIM2 MOSFET model. This is necessary to achieve simulation results within a reasonable amount of computation time. Three important approximations are explained below.

Drift-Diffusion Approximation

A good way of describing carrier transport in a semiconductor is given by the Boltzmann transport equation (BTE). A big disadvantage of this semiclassical approach is that it poses a computational challenge since there are no analytical solutions for complex configurations. The Monte Carlo method would be a possibility to solve the BTE, but it needs a lot of computing power. Therefore the majority of semiconductor device models use the drift-diffusion equations. The big advantage is that those equations are local in respect to the driving forces of carrier transport like the electric fields and the gradient in carrier density. The drift-diffusion equations are derived from the first two moments of BTE by considering a steady state situation in a 1D geometry. Further necessary assumptions are that the carriers move under equilibrium conditions and that their velocity is an instantaneous function of the local electric field. Common drift-diffusion models are based on the current density equations, the continuity equations and the Poisson equation. For discretization of the equations a finite difference scheme may be used. [20] [21]

Charge-Sheet Approximation

The charge sheet approximation is often used in compact modeling to simplify and therefore speed up numerical computations. A few assumptions are made for that. First of all the charge layer built up due to channel inversion is assumed to be infinitesimally thin. This sheet of charge is located at the interface of the silicon bulk and the gate oxide. Also there is said to be no potential drop across the charge sheet. This approach uses the fact that the inversion layer is by orders of magnitude thinner than the depletion zone around it. To estimate the bulk charge below the very thin inversion layer the depletion approximation is used. [22]

Gradual-Channel Approximation

In the gradual channel approximation it is assumed that there is a gradual variation of the voltage along the channel from drain to source. Perpendicular to the channel from gate to bulk on the other hand there is a sudden variation in voltage. Those two parts can be looked at separately and are treated as one dimensional problems. The goal of this approximation is to calculate the drain current as a function of drain and gate voltages. [23]

3.2 Simulation Environment

The simulations were conducted on a Linux machine with the operating system Red Hat 5. The simulation environment at ams AG was Spectre® Simulator from Cadence®. Spectre is a circuit simulator for basic SPICE analysis which also supports scripts written in Verilog-A. There was a GUI available when using Spectre but it was also possible to run the simulation directly from the console by executing a file including the netlist. The used version of Spectre was 14.1.0.804.isr12. [24]

The netlist contained all the necessary information about the simulated circuit (nets, nodes and voltage supplies). It was included in a file containing the invoking commands for the used model code as well as the parameter definitions. The file format was .scs, a Spectre command script. The command syntax used in all those files was not Spectre language but SPICE simulation language. Spectre was used as an emulator. This way it was easier to produce result files which could be directly compared to the examples provided by Hiroshima University. An example file for the simulation of a single nFET device is included in the Appendix 9.3.1. This simulation is for an initial, unstressed device. It contains comments by the author. Though the file seems short and simple, one has to be aware that only a single device was simulated. The more complex the circuit becomes, the longer and more complicated a command file gets.

3.3 Device Parameter Extraction

Since the model should be usable for various different MOSFET devices it is necessary to have some variable parameters. Those device parameters determine how the model behaves. For fitting the simulation results to the measured curves, proper values needed to be assigned to all the device parameters. Examples for those are geometry related values like gate length L and width W, which are input instance parameters. Those are

changed according to the simulated device. The gate oxide thickness TOX is a fixed parameter, which is determined by the production process. Parameters like $SC1$ for scaling of the short channel effect, the flatband voltage $VFBC$, $MUEPH0$ to introduce effects of phonon scattering or $QME1$ for inclusion of quantum mechanical effects are used to further approach real device behavior with the simulation. Those are fit parameters and need to be extracted from measurement data. They vary with the process. Some of the input parameters just act as flags to decide which operations are included in the simulation. They can be set to 0 or 1. The flag $COISUB$ for example decides, if I_{sub} is calculated. Other flags do the same for the gate current, the gate induced drain leakage current, the flicker noise and the thermal noise. [19]

Overall there were 232 device parameters and flags to be set to adapt the model to the measured process. For various devices of one process, it should be sufficient to just change the geometry parameters. To achieve that, a few different devices, including the minimum and maximum device of the process, needed to be measured at various conditions. The results were processed by a parameter extraction script (intellectual property of ams AG). It delivered values for all the parameters by varying them and trying to achieve the best possible fit to all the input curves. This is a very tricky process which should be conducted by an experienced operator. The list of the extracted parameter for the used process is included in the Appendix 9.3.2.

To exhibit the quality of the extracted parameter set, the plots below show an overlay of the simulation and measurement data from an unstressed device. Since the simulation should be accurate for all applied bias conditions, a compromise was necessary. A perfect fit for all the curves was not possible. In Figure 16 the fit for the transfer characteristic in saturation region is very accurate (red line) while the one in linear region (blue line) is slightly off at high voltages. In the subthreshold region in Figure 17 on the other hand the blue line is closer to the measurement than the red one.

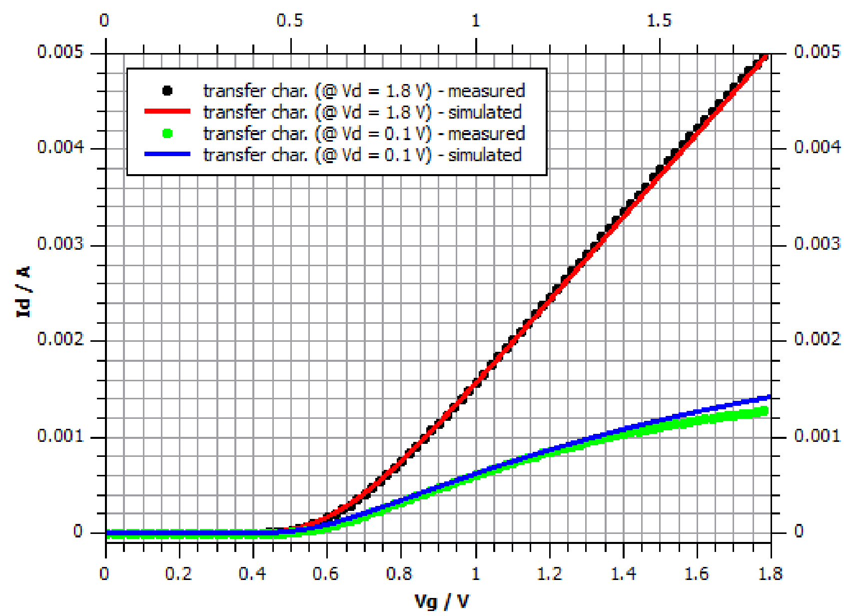

Figure 16: Comparison between measurement and simulation for initial transfer characteristic (linear plot) [4]

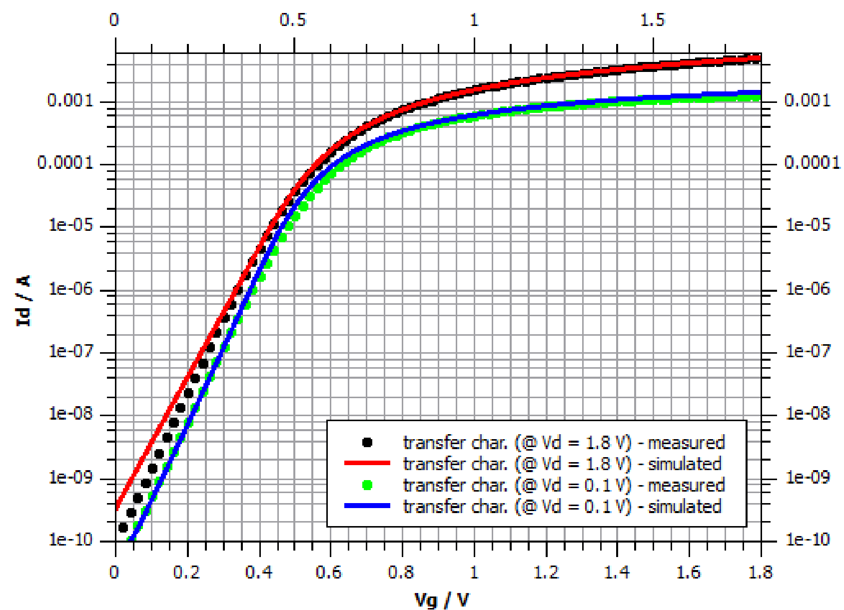

Figure 17: Comparison between measurement and simulation for initial transfer characteristic (semilogarithmic plot) [4]

For the output characteristics in Figure 18 both simulations fit the measurements properly with a slight deviation in the blue line for higher voltages.

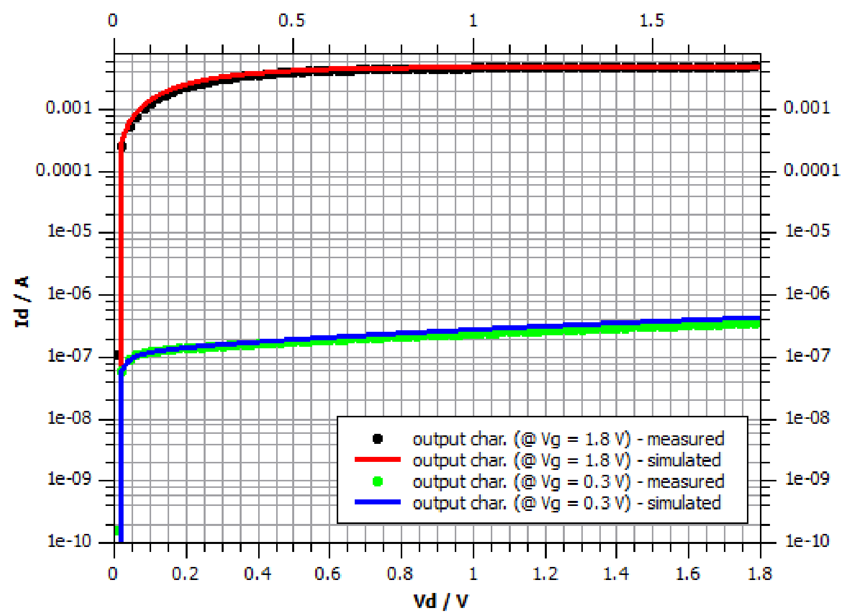

Figure 18: Comparison between measurement and simulation for initial output characteristic (semilogarithmic plot) [4]

In the plots above the trade-off in accuracy between different conditions is visible. But overall a quite good fit for all of those conditions was achieved.

3.4 HiSIM2 Hot Carrier Degradation Model

The HiSIM2 hot carrier (HC) degradation model is a newly developed extension to the existing HiSIM2 MOSFET model. It was first implemented in the HiSIM2 10.0.0 Alpha release. The latest available version for this work was the HiSIM2 3.0.0 Beta release from 09.05.2016. With the HiSIM2 approach calculations of the aging effects were for the first time directly accessible within a spice model. Up to now only hand-shake simulations with external tools, using semi-empirical equations, have been used. During the writing of this thesis, a close cooperation with Hiroshima University was established and the model was regularly updated and provided with patches and improvements. As in the basic MOSFET model, also in the aging model the focus was on the compromise between the highest possible accuracy of device behavior prediction and a reasonable runtime.

The basic principle of the HC model is to sum up all the charges flowing through the substrate of the device. This is done by integrating the substrate current I_{sub} over the total stress time t_{stress}. The major part of I_{sub} is caused by impact ionization due to hot carriers in the pinch-off region of the MOSFET. The generated holes leave the device over

the substrate and contribute to the substrate current. Furthermore hot carriers cause the formation of trap states at the bulk-oxide interface by breaking chemical bonds. It is assumed that the substrate current is proportional to the rate of trap production (cf. Section 1.2). Therefore I_{sub} is a good measure for the degradation due to hot electrons in the channel and at the bulk-oxide interface. The calculation of I_{sub} is done in the main part of the HiSIM2 MOSFET model. Like the other currents it is calculated by using a Poisson solver with certain approximations (cf. Section 3.1). The necessary initial inputs are the terminal voltages applied to the device.

The aging part of the model is calculated in a separated file, which is included into the Verilog A main file. As mentioned above the degradation model uses the hole current over the substrate to give an estimation for the overall number of induced trap states N_{trap} at the interface between bulk and gate oxide. After N_{trap} is calculated, it is given back to the main model. There it is used to update the Poisson equation. A recalculation of the currents leads to the final, aged result. Even though it is in an extra file, the big advantage of this aging model is that it is completely included into the HiSIM2 model structure. Therefore it is not necessary to change the modeling environment. This makes the application more convenient and practical for the user. The block diagram in Figure 19 gives a simple overview on how the concept with the inclusion of the aging model works.

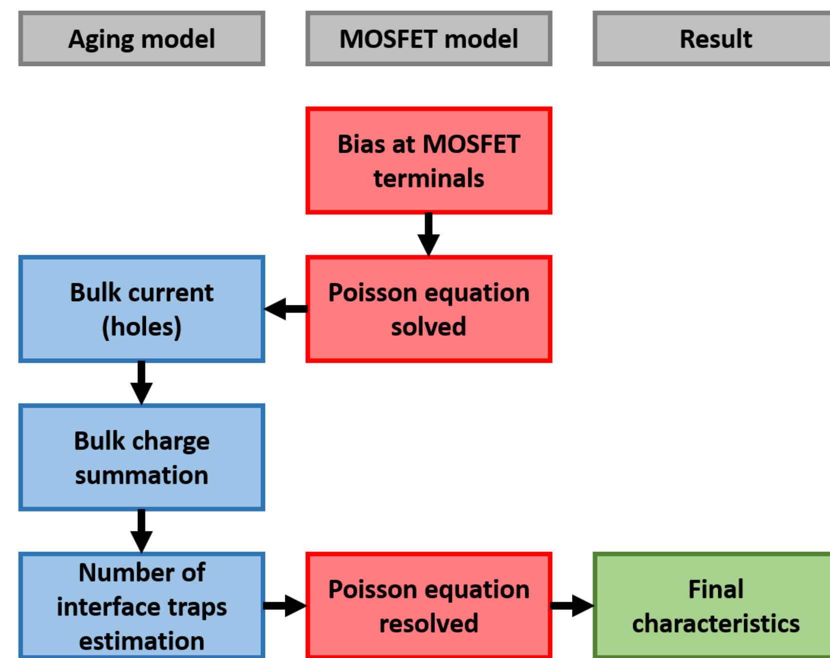

Figure 19: Block diagram: inclusion of the HiSIM2 aging model [1]

3.4.1 Physical Background of the Model

The hot electron degradation model for n-MOSFETs is based on the preceding work of Hiroshima University and STARC on the trap density increase due to device stressing. The change in trap density was experimentally found by measuring 1/f noise characteristics. A model was developed to reproduce those measurements with simulations. Most approaches for compact aging modeling focus only on the determination of the V_{th} shift which is not sufficient to explain all the changes in the device characteristics (cf. Section 5). For example the subthreshold slope and the mobility should be monitored as well. The basic idea was to develop a new model to predict all aging effects simultaneously.

As stated above in Section 1.2, the physical origin of the device degradation are the induced trap states in the gate dielectric. The intention was to model the trap density increase as a function of the stress time. The origin of the 1/f noise is the carrier trap and detrap process at the interface between substrate and oxide. To monitor this process, 1/f noise measurements were conducted. The noise intensity S_{id}, if normalized by I_d^2 was found to be proportional to the trap density N_{trap} at a certain energy E.

$$\frac{S_{id}}{I_d^2}(f) \propto N_{trap}(E) \tag{5}$$

S_{id} Measured noise density (function of the frequency f) $[\frac{1}{\mathrm{Hz}}]$

I_d Drain current [A]

N_{trap} Trap density (function of the energy E) $[\frac{1}{\mathrm{m}^3}]$

By investigating the substrate current of the device another proportionality was found between the normalized noise density and the product of substrate current and time $I_{sub} \cdot t$.

$$\frac{S_{id}}{I_d^2}(f) \propto I_{sub} \cdot t \tag{6}$$

I_{sub} Measured substrate current [A]

t Overall stress time [s]

Bringing Equation 5 and 6 together yielded in the discovery of an approximate universal relationship between N_{trap} and $I_{sub} \cdot t$.

$$N_{trap}(E,t) = A + B \cdot (I_{sub} \cdot t)^{0.8} \tag{7}$$

A	Additive constant (device and process specific) $[\frac{1}{\text{m}^3}]$
B	Multiplicative constant (device and process specific) $[\frac{\text{A s}}{\text{m}^3}]$

With Equation 7 it was possible to find values for the trap density by integrating the substrate current over the stress time. The exponent 0.8 in the equation was empirically found. N_{trap} was then included into the Poisson equation to achieve a consistent model for the device degradation.

$$\nabla^2 \phi = -\frac{q}{\epsilon} \cdot \left(p - n + N_D^+ - N_A^- - N_{trap}(t)\right) \tag{8}$$

ϕ	Electrostatic potential [V]	n	Electron concentration $[\frac{1}{\text{m}^3}]$
q	Electric charge [C]	N_D^+	Donor density $[\frac{1}{\text{m}^3}]$
ϵ	Permittivity $[\frac{[\text{F}]}{\text{m}}]$	N_A^-	Acceptor density $[\frac{1}{\text{m}^3}]$
p	Hole concentration $[\frac{1}{\text{m}^3}]$		

This simulation approach is very convenient since it does not just shift device characteristics due to a shift of V_{th}, but recalculates them for the physically changed device. If the number of trap states are taken into account for the simulation, the degraded properties of the device can be just simulated straight forward like for the initial device. [25]

3.4.2 Implementation into the Simulation Environment

In the Beta version of the degradation model it was still necessary to use two different command files for the implementation in the ams AG Spectre environment. The stressing part was needed to be executed before the aging part. In the first run the device was stressed and information about it was stored. This information was then used for simulating the degradation during the aging run. For later versions of the model it was planned to further simplify the usage.

First the stressing of the device was simulated by using an adapted command script. An example is displayed in the Appendix 9.3.3 with comments by the author. In the stress script it was very important to specify the stress voltages at gate and drain and the stress time for the simulation. Also the flag for the activation of the degradation model *CODEG* needed to be set to 1 and all the default aging parameters had to be put in the

script. The stress script created a special result file called "degradation.out". In there all the information about the aging of the device due to the applied stress was stored. An example for such a result file is included in the Appendix 9.3.4.

The second step for using the HiSIM2 HC degradation model was the execution of the aging script. Again an example is included in the Appendix 9.3.5. The aging file also had some special content for simulating the aged characteristics of the previously stressed device. In the first line of the subcircuit definition the data file "degradation.out" was included to get hold of the aged parameters. It was important to assure that the device geometries and the model definitions were the same for the stressing and the aging part. Again the $CODEG$ degradation flag was set to 1 and the last default aging parameter $TRAPLX$, which was not used during stressing, was defined. Also the imported degradation parameters needed to be defined properly. The results from the aging command script were the characteristics of the degraded device. Those could than be compared to the initial curves to see the changes due to aging.

To make the application of those two files easier, they could of course be combined into a common shell file. It would also be possible to include the starting sequence for the simulator and any other desired application. Simplified like this, it would be only necessary to execute a single shell script.

3.4.3 Model Equations and Parameters

For the calculation of the total number of trap states N_{trap}, a set of modeling equations is used inside the HC degradation script. Those equations are displayed below with some short explanations and annotations. The total trap amount is divided into two parts which are added in the end. There are deep trap states and shallow trap states. The shallow traps lie energetically close to the conduction band. There is only little energy necessary to fill and empty them. The deep trap states lie lower in energy, somewhere closer to the band gap center and therefore further from the conduction band. Because they are lower in energy, a carrier that is trapped here will not leave so easily again. Figure 20 shows a rough schematic of the trap density distribution for shallow and deep traps.

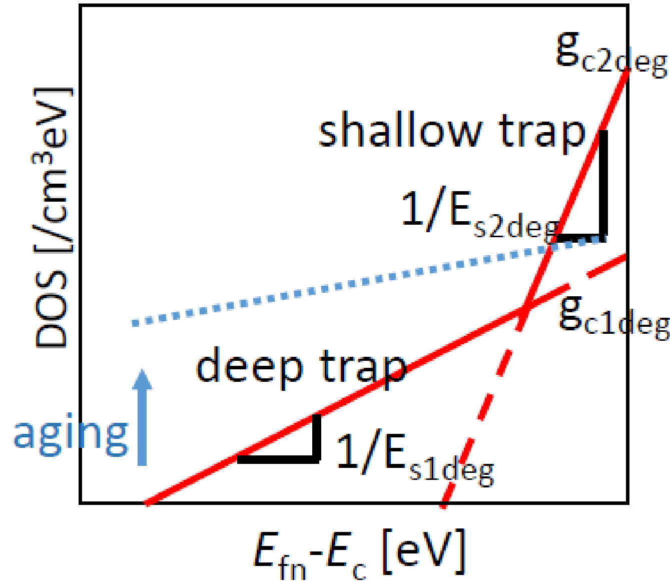

Figure 20: Trap density distribution as a function of energy [5]

The x-coordinate in Figure 20 shows the energy going from somewhere inside the band gap on the left side to the conduction band edge on the right side of the diagram. E_{fn} stands for the Fermi energy and E_c for the conduction band energy. The red lines mark the trap density distributions as functions of this energy. It is obvious that the shallow trap distribution has a higher maximum value than the one for deep traps ($g_{c2deg} > g_{c1deg}$). It is steeper as well ($\frac{1}{E_{s1deg}} > \frac{1}{E_{s2deg}}$). Therefore the shallow traps dominate near the conduction band edge. Which makes them less important for device degradation. Electrons will be trapped there and they will de-trap again, but there should be no change in the shallow trap distribution when stressing the device. So the number of shallow traps N_2 is assumed to stay constant. It is calculated in Equation 9 with the process specific input parameters $TRAPGC2$ and $TRAPE2$. The last part of the equation which looks like $\frac{x}{sin(x)}$ is a correction value for the temperature dependence.

$$N_2 = \boldsymbol{TRAPGC2} \cdot \boldsymbol{TRAPE2} \cdot \frac{\frac{k_B \cdot T}{\boldsymbol{TRAPE2}}}{sin\left(\frac{k_B \cdot T}{\boldsymbol{TRAPE2}}\right)} \tag{9}$$

N_2	Number of shallow trap states per volume element $[\frac{1}{\mathrm{m}^3}]$
$TRAPGC2$	Estimated maximum density of shallow traps (at conduction band edge) $[\frac{1}{\mathrm{m}^3\ \mathrm{eV}}]$
$TRAPE2$	Estimated inverse gradient of shallow trap density distribution [eV]
k_B	Boltzmann constant $[\frac{\mathrm{eV}}{\mathrm{K}}]$
T	Temperature [K]

The deep trap distribution in Figure 20 on the other hand is flatter. Some of those states are much lower in energy compared to the conduction band edge. De-trapping from those states is very unlikely. Due to HCI the number of those states increases with stress duration. This is indicated with the blue dotted line and the arrow. The more electrons are captured by deep traps, the bigger the degradation effects on the device (cf. Section 1.2). To know the overall number of trap states the number of deep traps N_1 needs to be approximated first. This is done in several steps. First in Equation 10 the auxiliary function Ag_{c1} is calculated with the input parameter $TRAGC0$, but without time dependency. Because of the minimum function Ag_{c1} has a maximum value of $9 \cdot 10^{-5}$. This maximum is an empirical value, which prohibits mathematical problems with the subsequent calculations.

$$Ag_{c1} = min\left(1.8 \cdot 10^{-6} \cdot exp\left(\frac{\frac{I_{sub}}{W}}{\boldsymbol{TRAGC0}}\right), 9 \cdot 10^{-5}\right) \tag{10}$$

Ag_{c1}	Auxiliary function for the calculation of g_{c1} $[\frac{\mathrm{m}}{\mathrm{A\ s}}]$
I_{sub}	Substrate current [A]
W	Channel width [m]
$TRAPGC0$	Introduced parameter for additional modulation of Ag_{c1} $[\frac{\mathrm{A}}{\mathrm{m}}]$

In a next step the maximum density of deep traps at the conduction band edge g_{c1} is evaluated in Equation 11 by using Ag_{c1} and the input parameter $TRAPGC1$. Here the dependence on the stress time t and the substrate current I_{sub} is added. An increase in t adds an offset to the whole deep trap distribution and shifts it to higher values.

$$g_{c1}(t) = \boldsymbol{TRAPGC1} + 4 \cdot 10^{18} \cdot \left(1 - 10^{-Ag_{c1} \cdot \frac{I_{sub} \cdot t}{W}}\right) \tag{11}$$

g_{c1}	Deep trap density distribution (function of stress time) $[\frac{1}{\mathrm{m^3\ eV}}]$
t	Stress time [s]
$TRAPGC1$	Estimated maximum density of deep traps for $t = 0$ s (at conduction band edge) $[\frac{1}{\mathrm{m^3\ eV}}]$

The gradient of the deep trap distribution function $\frac{1}{E_1}$ is determined by E_1 which is calculated in Equation 12. The input parameters $TRAPE1$ and $TRAPE0$ are used. The later is for additional geometry and process specific modulation of the equation. E_1 is a function of stress time t times I_{sub} as well. For longer times E_1 increases which leads to a flattening of the distribution in Figure 20. Trap states at lower energies become

more likely. Those are harder to de-trap and therefore have an increasing influence on the device degradation. A maximum value of $E_1 = 1$ is used for a flat distribution.

$$E_1(t) = min\left(\boldsymbol{TRAPE1} \cdot exp\left(\frac{\frac{I_{sub} \cdot t}{W}}{\boldsymbol{TRAPE0}}\right), 1\right) \tag{12}$$

E_1	Inverse gradient of deep trap density distribution (function of stress time) [eV]
$TRAPE1$	Estimated inverse gradient of deep trap density distribution for $t = 0$ s [eV]
$TRAPE0$	Introduced parameter for additional modulation of E_1 $[\frac{\text{A s}}{\text{m}}]$

The results for g_{c1} and E_1 are used in Equation 13 to calculate the number of deep trap states N_1. Again there is a correction value for the temperature dependence. There is also an additional exponential factor including the drain voltage, the surface potentials at drain and source and the input scaling factor $TRAPLX$. This correction value was introduced in the latest release and denotes to the fact that degradation is stronger for smaller V_{ds} in operation (cf. aged output characteristics in Section 5.2 below). At low V_{ds} the whole channel region is in inversion and there are lots of electrons near the oxide interface which fill the traps. If V_{ds} is high enough the channel is partially pinched off. The amount of electrons in this depleted region is lower and less trap states are filled. With $TRAPLX$ there is an additional way of modeling the length of the region in the channel where most of the trap states are located. It is another modification possibility for various geometries and processes.

$$N_1 = g_{c1}(t) \cdot E_1(t) \cdot \frac{\frac{k_B \cdot T}{E_1(t)}}{sin\left(\frac{k_B \cdot T}{E_1(t)}\right)} \cdot exp\left(-\frac{V_{ds} + P_{s0} - P_{sl}}{\boldsymbol{TRAPLX}}\right) \tag{13}$$

N_1	Number of deep trap states per volume element $[\frac{1}{\text{m}^3}]$
V_{ds}	Applied drain voltage [V]
P_{s0}	Calculated surface potential at source side [V]
P_{sl}	Calculated surface potential at drain side [V]
$TRAPLX$	Scaling factor for the affected channel length [V]

Finally in Equation 14 the constant amount of shallow traps and the time dependent number of deep traps are added to form the total number of trap states N_{trap}.

$$N_{trap} = (N_1 + N_2) \tag{14}$$

N_{trap} Total number of trap states per volume element $[\frac{1}{\mathrm{m}^3}]$

After that, the evaluation of N_{trap} inside the aging model is completed. The overall number of traps is then used in the basic model as an additional value in the Poisson equation. All calculations are conducted again and the result are updated and aged device characteristics.

3.4.4 Sensitivity Analysis for Aging Parameters

It is important to know about the effects of parameter changes on the simulation result. A sensitivity analysis was conducted for seven HiSIM2 aging parameters by varying their values around the default value. The series of plots below shows the influence of those parameters on the simulation outcome. The plotted curves are transfer characteristics in the saturation regime ($V_d = 1.8$ V) for a n-MOSFET device with $W = 10$ μm and $L = 0.18$ μm. For all the device parameters the HiSIM2 default values have been used. Except for the flag COISUB, which triggers the calculation of the substrate current. It needs to be set to 1 for doing aging simulations. An applied stress of $V_g = 1$ V and $V_d = 3$ V for $t = 1e5$ s was simulated. The data is displayed in linear or semilogarithmic plots depending on how the variations can be identified best. A black arrow indicates in which direction the curve is shifted, if the parameter is increased. To identify small changes, inset plots are added to magnify details. In all plots, the black line represents the HiSIM2 default value.

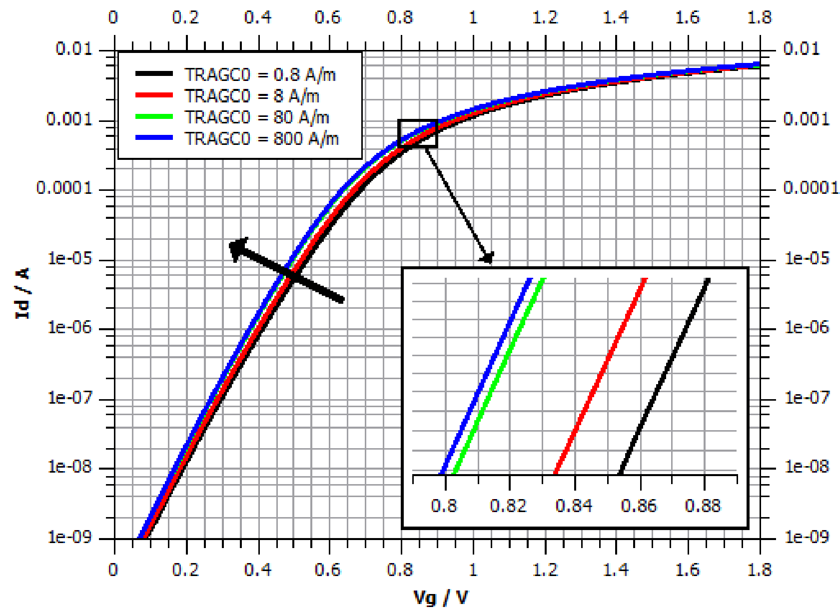

Figure 21: Sensitivity analysis for $TRAGC0$ - transfer char. @ $t = 1e5$ s [4]

The model parameter $TRAGC0$ is part of Equation 10 and influences the value of Ag_{c1} from inside the exponential function. Figure 21 shows that an increase of $TRAGC0$ shifts the characteristic to smaller voltages, because it decreases Ag_{c1}. With Eq. 11 this leads to a decrease of g_{c1} which decreases the total number of deep traps N_1 through Equation 13. Therefore there is less degradation. At a certain magnitude of $TRAGC0$ the shift is getting smaller (from green to blue curve) because Ag_{c1} is getting close to it's minimum value of $1.8 \cdot 10^{-6}$. Also simulations below the default value show no shift. Because of the chosen stress conditions the substrate current is about $3.9 \cdot 10^{-5}$ A. With the device geometry $W = 10$ μm and a smaller value for $TRAGC0$ the given maximum value of $9 \cdot 10^{-5}$ in Eq. 10 is reached. For choosing a good value for $TRAGC0$ it is necessary to keep those limitations in mind.

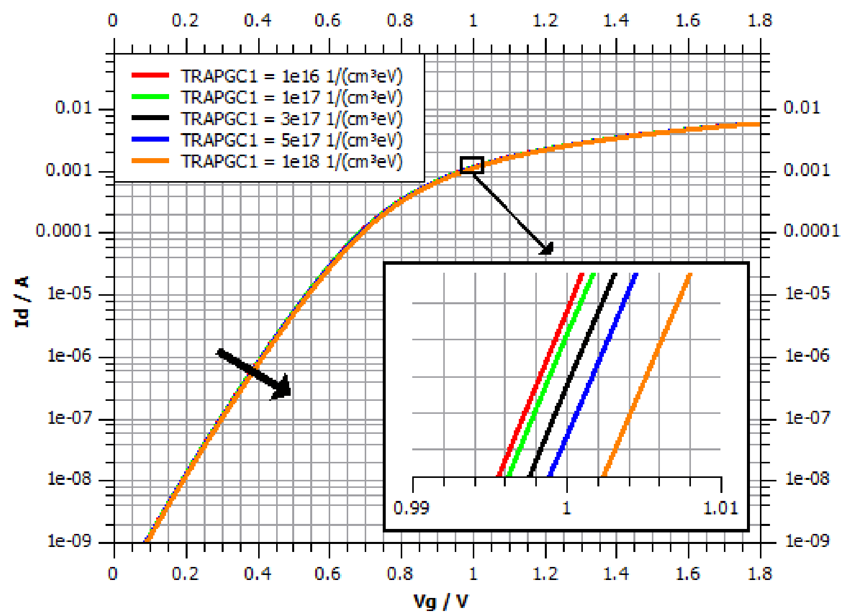

Figure 22: Sensitivity analysis for $TRAPGC1$ - transfer char. @ $t = 1e5$ s [4]

Figure 22 indicates only a minor change of the result for the variation of the parameter $TRAPGC1$. It is only an additive constant in Eq. 11 representing the initial number of deep traps at the conduction band edge. Only if the exponent is becoming small, the second part of the sum is in the range of $TRAGC1$. That happens at short times or low stresses. In this example due to the high stress it has little influence on the outcome. Still it is obvious that with an increasing of $TRAGC1$ the characteristic is shifted to higher voltages.

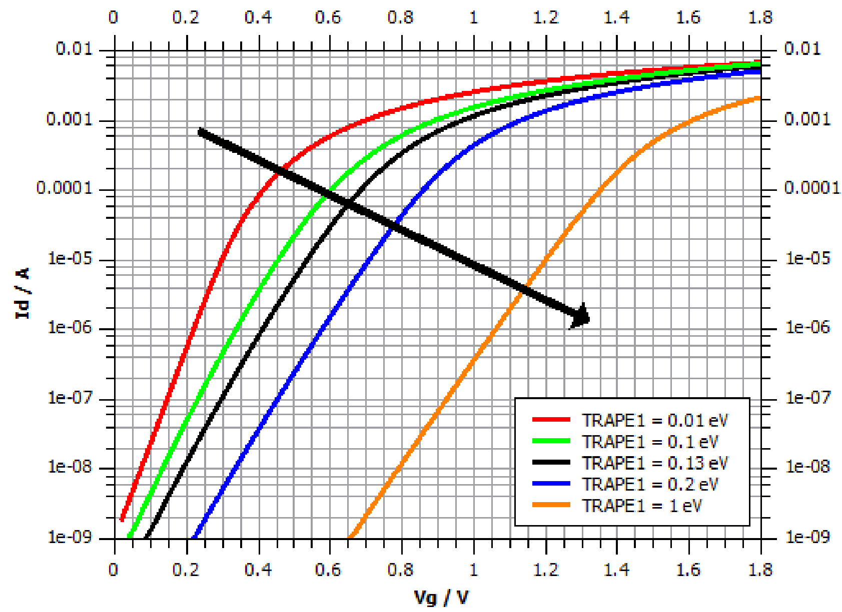

Figure 23: Sensitivity analysis for $TRAPE1$ - transfer char. @ $t = 1e5$ s [4]

Figure 23 indicates a high sensitivity of $TRAPE1$. By increasing $TRAPE1$ the aging effect is getting stronger, shifting the curves to higher voltages and also changing the slope. The parameter directly influences the slope of the trap density distribution $\frac{1}{E_1}$ via Equation 12. By increasing $TRAPE1$ the distribution is flattened. Due to the minimum function there is a limit with the maximum value of $E_1 = 1$.

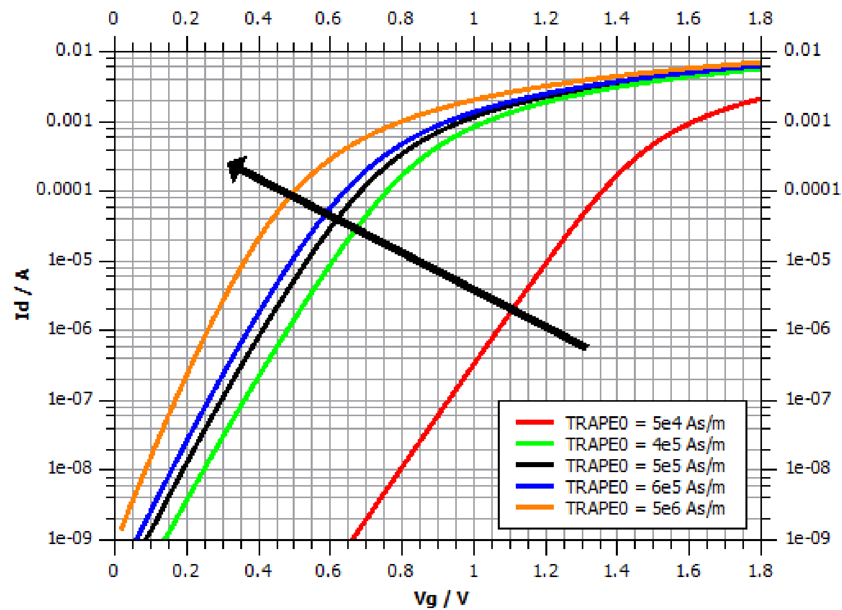

Figure 24: Sensitivity analysis for $TRAPE0$ - transfer char. @ $t = 1e5$ s [4]

For $TRAPE0$ the results are in Figure 24. They show the inverse behavior to $TRAPE1$, decreasing the aging effect for an increase in magnitude. Since $TRAPE0$ is part of the exponential function its impact on E_1 is even bigger. Like for $TRAGC0$ the right choice of the value is depending on the aging conditions and the device geometry.

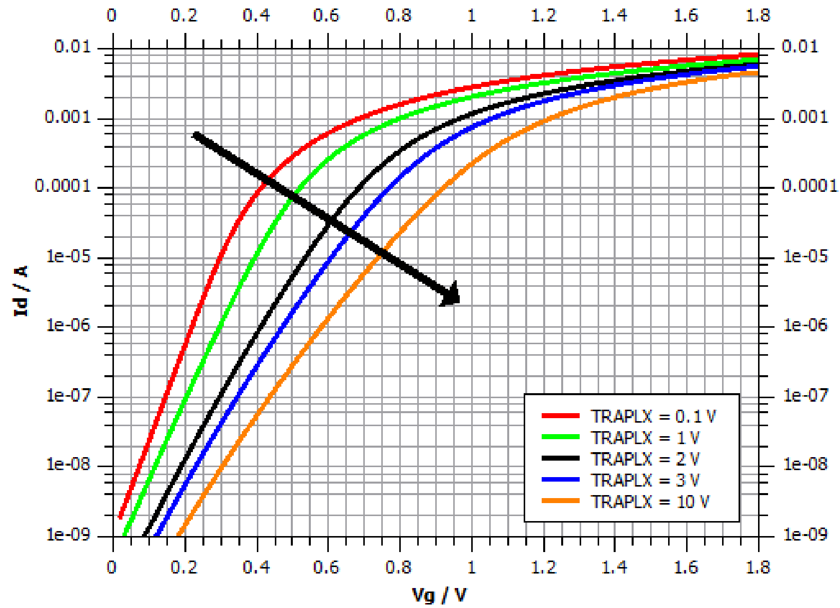

Figure 25: Sensitivity analysis for $TRAPLX$ - transfer char. @ $t = 1e5$ s [4]

As Figure 25 shows the parameter $TRAPLX$ has a remarkable impact on the simulation outcome as well. An increase of the parameter would strongly increase the aging effects, shifting the curves to higher voltages. The influence of $TRAPLX$ is comprehensible. It helps determine how big the area in the channel is, where deep trap states are present. In Equation 13 $TRAPLX$ serves as a scaling factor inside the exponential function.

For the last two parameters $TRAPGC2$ and $TRAPE2$ from Equation 9 the simulation needed to be changed to see how they affect the outcome. Those two parameters determine the number of shallow trap states, which is assumed to stay constant over time. For longer stress times the overall sum of traps N_{trap} in Equation 14 is dominated by the deep traps N_1. To make effects of the shallow trap parameters visible the stress time in the simulation is reduced $t = 100$ s.

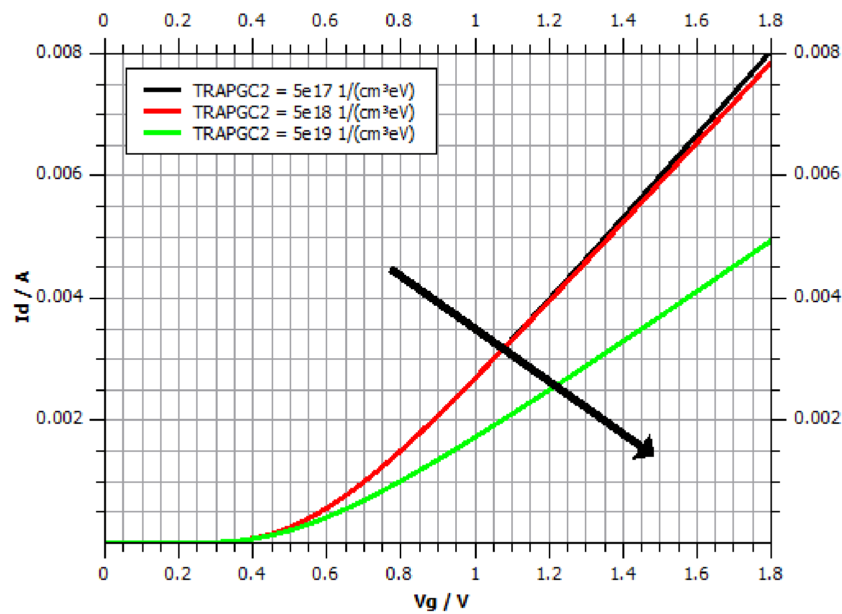

Figure 26: Sensitivity analysis for $TRAPGC2$ - transfer char. @ $t = 1e2$ s [4]

In Figure 26 only results for $TRAPGC2$ being higher as the default are displayed. Even a change of one order of magnitude has not much influence on the result. But by increasing $TRAPGC2$ by two orders of magnitude the influence of the shallow traps dominates the deep traps and the aging effect is strongly increased. Fine tuning of $TRAPGC2$ therefore is only an issue for short stress times or low stress conditions.

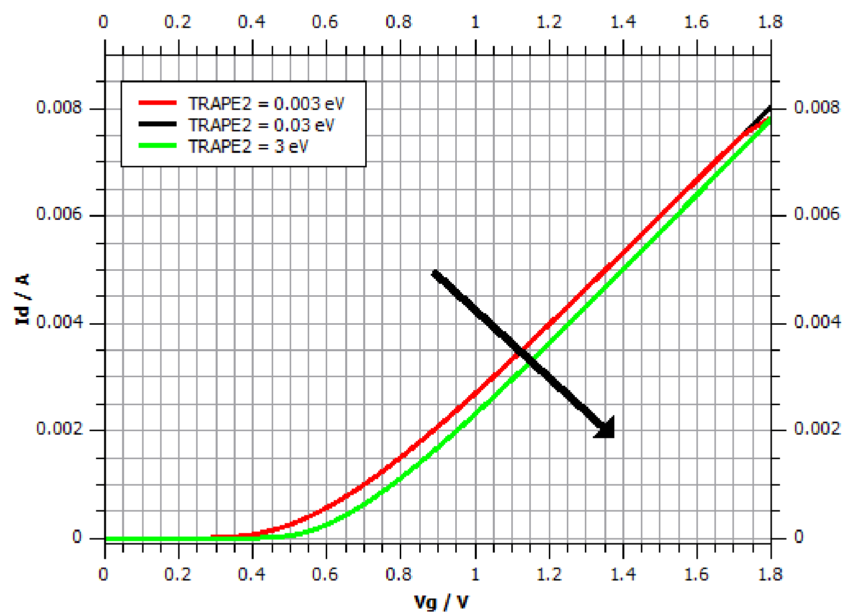

Figure 27: Sensitivity analysis for $TRAPE2$ - transfer char. @ $t = 1e2$ s [4]

The last analyzed parameter is $TRAPE2$ in Figure 27. A decrease in $TRAPE2$ of one order of magnitude has no influence on the result. Also the sensitivity for increasing $TRAPE2$ is not high. An increase of two orders of magnitude just gives a small shift to higher voltages. Therefore the degradation of the device gets stronger by increasing $TRAPE2$.

3.5 Data Extraction Routine

For the simulation results most of the data extraction was already done by the Spectre command scripts. At the end of the script a command was added to produce an output file in the format .print which included the data. In most of the cases it was enough to print the sweeped voltage (V_g or V_d) and the according drain current I_d for plotting the transfer and output characteristics.

For doing a more detailed examination of the results it was necessary to extract some parameters from the exported characteristic curves. For that purpose another set of Matlab files was created. From inside those files Spectre was called and simulations for various specified time points were conducted. After every single simulation the routine extracted the desired parameter from the result file and stored it. Also the relative change of the parameter was calculated in-situ. Then the next time point was simulated. The result was data for the parameter value and the relative change over time of stress, similar to the extracted parameter data from the measurements. This made a comparison between model and measurement possible. Such parameter extraction scripts were created for $I_{d,lin}$, $I_{d,sat}$ and V_{th}. The Matlab script for the extraction of $I_{d,sat}$ can be found in the Appendix 9.4 as an example for the functional principle.

3.6 Challenges with the HiSIM2 Degradation Model

The greatest challenge for working with the HiSIM2 HC degradation model was of course that the model was still under development. During the six month time frame of this work a new update with minor or even major changes in code and functionality was received every few weeks. It was on the one hand very positive since the support from Hiroshima University was absolutely necessary for improving the usability and accuracy of the model. On the other hand it was time consuming to get accustomed to the new version and set up a new simulation environment every time. This was necessary in most of the cases since there were not only changes in the modeling equations but also in the

way of implementing the model. Finally for this thesis the HiSIM2 3.0.0 Beta version was used and newer developments were not taken into account any more. Therefore all the simulation results below give just an impression of this specific point in the model development. Issues that may be present in those results, have already been or will surely be addressed by the HiSIM team of Hiroshima University.

4 Takeda Based Degradation Model

In 1983 E. Takeda and N. Suzuki introduced an empirical device degradation model for the simulation of hot carrier injection effects [26]. Though this was over 30 years ago, their model is still the basis for a major part of industry standard degradation evaluation techniques. Since the eighties MOSFET devices changed a lot. They are much smaller and more efficient by now. Furthermore the materials and processes have changed. Because of that also the models grew more complex. For this thesis a model based on the Takeda approach, used by the quality department of ams AG, was developed. The goal was to build a model which takes information from short stress times and directly extrapolates the changes due to degradation to much longer times.

4.1 Fundamentals

Back in 1983 the degradation of small n-MOSFET structures due to hot carriers was already an issue. Takeda and Suzuki found new relationships between drain voltage, substrate current and the degradation of the device. They elaborated a model to predict the lifetime of submicron devices. It was their goal to provide information about the shift of extracted parameters like the threshold voltage over time of stress. The model could be used for qualification of the devices by declaring a certain amount of change as a limit for an according time of stress. The parameter change was estimated with a simple power function:

$$\Delta V_{th} = A_0 \cdot t^n \tag{15}$$

ΔV_{th}	Threshold voltage shift [V]
A_0	Magnitude of degradation
t	Stress time [s]
n	Exponential parameter (slope in log-log plot)

The slope n is strongly dependent on the gate voltage but only little on the applied drain voltage. The magnitude of degradation A in Equation 15 on the other hand shows a strong dependency on V_d and can be expressed as follows:

$$A_0 \propto \exp\left(-\frac{\alpha}{V_d}\right) \tag{16}$$

α Scaling parameter for A

V_d Applied drain voltage [V]

Similar to the HiSIM degradation approach, the substrate current was used as an indication for the number of electron-hole pairs generated by impact ionization. This is important since the physical meaning of the parameter A_0 is proportional to the number of excess carriers generated by impact ionization. The Takeda model made it possible to predict device lifetime without doing long term stress tests. A few measurements for shorter times were enough to extract the necessary parameters n and A_0. The assumption that the parameter shift over time follows a power function provided satisfying results. Only the occurring saturation in parameter shift for very long stress times could not be reproduced with this method. [26]

4.1.1 Simulation Approach

The simulation program for the thesis was based on the Takeda-like approach used by the quality department at ams AG in Premstätten [27]. The calculation of the shift of an extracted design parameter due to aging in Equation 17 is very similar to the Takeda equation from 1983.

$$\Delta = \Delta_0 \cdot \left(\frac{t}{t_{ref}}\right)^n \tag{17}$$

Δ Relative parameter change

Δ_0 Model constant for amplitude of degradation

t Stress time [s]

t_{ref} Scaling factor including dependency on voltage and device geometry [s]

n Exponential model constant (slope in log-log plot)

The scaling factor for the stress time t_{ref} is a function of various model constants, ambient conditions, applied drain voltage and the channel length. The exact relation is given in Equation 18.

$$t_{ref} = A \cdot \exp\left(\frac{B}{V_d}\right) \cdot L^C \cdot \exp\left(-\frac{E_a}{k_B T}\right) \tag{18}$$

A	Model constant for the amplitude [s]
B	Model constant for scaling of the drain voltage dependency [V]
V_d	Applied drain voltage [V]
L	Channel length of the device [m]
C	Exponential model parameter for scaling of the length dependency
E_a	Model constant for scaling of the temperature dependency [eV]
k_B	Boltzmann constant $\frac{[\mathrm{eV}]}{[\mathrm{K}]}$
T	Temperature [K]

Originally the resulting relative parameter change Δ from Equation 17 was multiplied with the initial parameter value to gain an approximation of the overall change for longer times. The model constants were extracted from several short term stress measurements. For every considered design parameter like for example V_{th} or $I_{d,lin}$, a complete new set of model constants needed to be extracted. Therefore the calculation of one parameter was completely independent from another. The new correlated model approach tried to change this by using only one set of values for Δ_0, n, A, B, C and E_a. For the simulation of the device the HiSIM2 MOSFET model was used. For the degradation simulation the HiSIM2 aging model was replaced by the Takeda based equations.

The HiSIM2 MOSFET model uses a elaborated list of device parameters for modeling the device characteristics (cf. Section 3.3). All of those parameters are important for the result, but some of them have a very direct influence on the shape of the characteristic curves. By changing those device parameters accordingly the aging effect might be simulated very well. The approach results in degraded characteristic curves similar to those from the HiSIM2 degradation model. Not only knowledge about the design parameters but about the whole device behavior is gained that way.

The implementation into the HiSIM2 environment was straight forward. A simulation was done for the initial, unstressed device. Afterwards the Takeda model constants were determined and the Δ value was calculated. A selected set of HiSIM2 model parameters was "aged" by multiplication with Δ and the simulation was done again with those updated values. The following five parameters were used in the approach:

- ***NSUBP*:**
  Model parameter for the peak of the pocket impurity concentration. An increase in $NSUBP$ shifts the threshold voltage of the linear regime to higher V_g.

- ***MUEPH*1:**
  Model parameter for the carrier mobility due to phonon scattering. A decrease in $MUEPH1$ decreases the slope of the transfer characteristic in the linear regime.

- ***RD*:**
  Model parameter for the drain-contact resistance of the lightly doped drain region. An increase in RD decreases the maximum drain current $I_{d,lin}$ the linear regime due to an increased drain resistance.

- ***VMAX*:**
  Model parameter for the saturation carrier velocity. An increase in $VMAX$ increases the maximum drain current $I_{d,sat}$. This is necessary to weaken the effect of the increased drain resistance in the saturation regime.

- ***PTL*:**
  Model parameter for the strength of the punchthrough effect. An increase in PTL shifts the threshold voltage of the saturation regime to higher V_g.

For a fast review of accuracy some design parameters were extracted within the simulation and compared to the measurement data. The parameters in question were: $I_{d,lin}$, $I_{d,sat}$, $g_{m,max}$, $V_{th,lin}$ (extracted from the transfer characteristic in the linear regime at $I_d = 10\ \mu$A) and $V_{th,sat}$ (extracted from the transfer characteristic in the saturation regime at $I_d = 55.56\ \mu$A). After the first tests of the model it became clear that the change of the five device parameters used for aging could not be weighted equally. Since their influence on the HiSIM2 simulation is different, additional weights were introduced. The weights are used as scaling factors for the calculated change Δ. Though it makes the model approach even more heuristic, the introduction of the weights was necessary for the functionality.

4.2 Implementation

The implementation of the Takeda based model approach into the HiSIM2 MOSFET model environment was done with Matlab scripts. For the simulations, Spectre was called from inside the Matlab code. The simulation of the initial and unstressed device was exactly the same as explained in Section 3.2 above for the HiSIM2 degradation approach. The aging part on the other hand was completely different. The necessary inputs in the beginning were the time vector (times of stress for which the simulation was conducted) and the according measurement data for the relative change of the design parameters.

It was planned to do only one simulation step for aging all the five HiSIM2 parameters at once. But the results were not good, because the model constant optimization was not working properly. A solution was found by dividing the model in two parts: one for the linear operating regime of the device and one for the saturation regime. The determination of the model constants Δ_0, n, A, B, C and E_a was done in the first part. For the second part the same values were used and only the weights were optimized. A schematic of the model approach is displayed as a block diagram in Figure 28.

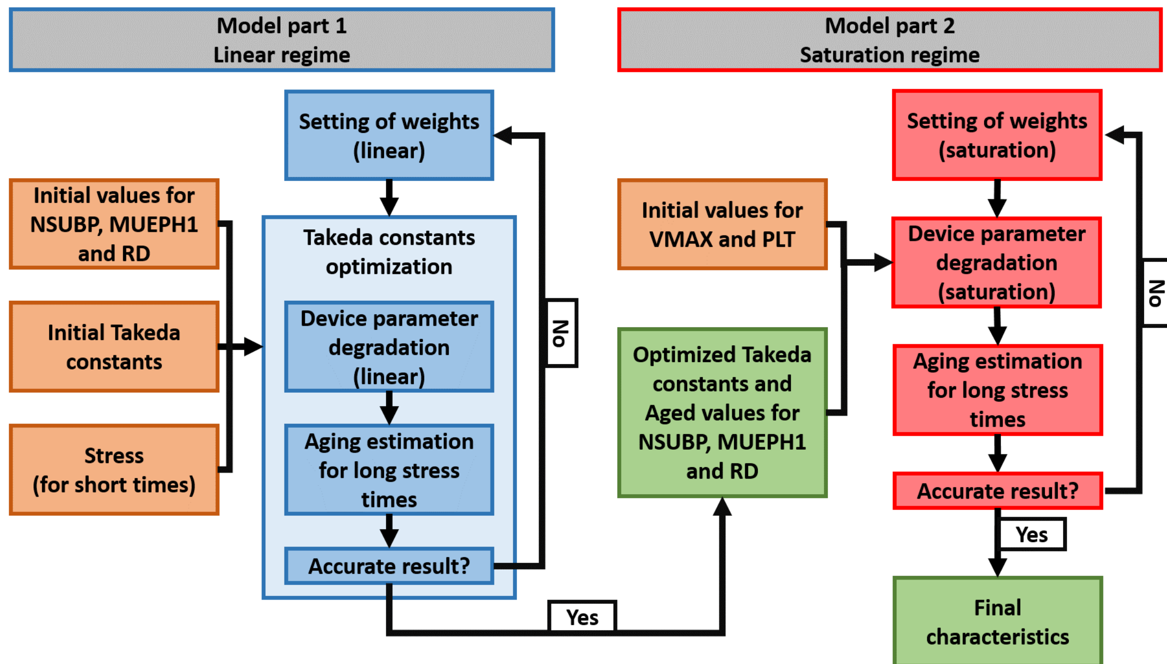

Figure 28: Block diagram of the heuristic Takeda based degradation model [1]

4.2.1 Model Part 1: Linear Regime

For the determination of the model constants in the first part of the model, a parameter optimization script written in Matlab was used. The script is intellectual property of ams AG. Its functionality is based on several extensive evaluation functions which are not displayed in the thesis. The original file optimized all parameters in question at once by varying them within preset boundaries. After every change the result was compared to available measurement data. The program tries to find proper parameter values by bringing the result as close to this data as possible. The necessary inputs for the script, besides the measurement data, are initial values for the parameters. Since the optimization runs needed a lot of computational power, it was of importance to find good initial values.

In order to provide the necessary functionalities for running the Takeda based simulation, the parameter optimization script was altered an extended. In Appendix 9.5.1 the main execution file is displayed. In there the time vector, the measurement data and the initial values for the model constants were specified. From there the data was given to the optimization functions.

During the optimization process of the model constants another Matlab function was invoked. It is displayed in Appendix 9.5.2. Within this function Spectre was called and the HiSIM2 simulations for the linear regime were conducted. This includes the degradation of the device parameters $NSUBP$, $MUEPH1$ and RD. The initial values for those three parameters as well as their weights were inputs for the function. The right values for the weights were found by trying different combinations and reacting on the results. The simulation time vector and optimized model constants were taken from inside the routine.

In a first step the initial linear characteristic was simulated and the design parameters were extracted for comparison to the measurement. Afterwards the stress was simulated. This was crucial because the model was expected to use only short simulated times and extrapolate to long stress times. In the displayed test case a stress of $V_g = 0.85$ V and $V_d = 3.0$ V was simulated for $t = 10$ ms and the extrapolation was done for 100 s, 1000 s and 10 ks (input time vector). The stress applied to the device was summed up by integrating over Equation 18. The accuracy of this integral was given by the number of supporting points in the stress simulation. The resulting overall stress value was normalized to give an estimation for the applied stress per second. A projection to any given stress time was possible by simply multiplying with this result. By using Equation 17 a value for Δ was calculated. Equation 19 shows how Δ was used to update

the device parameters:

$$para_{age} = para_0 \cdot (1 \pm \Delta \cdot w_{para}) \tag{19}$$

$para_{age}$	Aged device parameter
$para_0$	Initial device parameter
Δ	Relative parameter change
w_{para}	Determined weight for the parameter

Instead of calculating a Δ value for every parameter, the heuristic weights were used to scale the change for the different parameters. Depending on whether the parameter value should increase or decrease, a plus or minus sign was necessary.

For the evaluation of the aged characteristics, the altered device parameters were written to an external file. This file was included in a new HiSIM2 simulation to replace the initial device parameter values. Again the design parameters were extracted for comparison to the measurement. The whole procedure took place within the optimization script and was repeated until the best possible set of model constants was found. The tricky part was the inclusion of the weights. To find proper values the whole model part 1 was repeated several times as well.

4.2.2 Model Part 2: Saturation Regime

The most time expensive part of the model was already conducted in part 1 by finding proper values for the model constants while simultaneously simulating the device degradation in the linear operating regime. The changing of the linear device parameters in part 1 also had influence on the saturation regime. Tests showed that the degradation effects were overestimated. The device parameters for the saturation region, $VMAX$ and PTL, were used to diminish those effects. Varying them changed the results in the saturation region while having very little influence on the linear region. The simulation results from part 1 stayed about the same.

The Matlab file is displayed in Appendix 9.6. The functionality was very similar to the Matlab function in part 1. Additional inputs were the already aged values for the parameters $NSUBP$, $MUEPH1$ and RD and the model constants found by the optimization script. The only variables were the weights for the saturation parameters. They gave an additional degree of freedom and were found by means of trial and error. All the simulations in the script were done for both the saturation and the linear regime

to give a final degradation result for all the characteristics. For the aging of the device parameters the same approach as for part 1 was used (cf. Equation 19).

4.3 Limitations of the Takeda Based Model

The most obvious limitation is the heuristic nature of the model approach. For a proper simulation result a set of model constants and additional parameters, like the introduced weights, are necessary. In the beginning it was planned to develop a model which uses only one set of constants and is fully scalable for various stress voltages. With the introduction of the weight factors, there is now one Takeda equation for every aged device parameter. One can say there are several models, which look similar but are not identical. This was not intended but is the only way to gain reasonable results. The scalability is another issue. Tests showed that a certain set of constants is only valid for one geometry and one stress condition. Varying the applied stresses V_g or V_d produces unsatisfying results. Therefore the model, in its current state, is not ready to be used for industry standard degradation prediction. But improvements regarding the scalability should be possible, maybe by adding some more model parameters or extending the equations.

Altogether most of the requirements were fulfilled. The Takeda based model is able to predict long term aging effects of MOSFETs quite accurately. Once the model constants are known, a short stress simulation is sufficient as input. Both DC and AC stress may be simulated as long as there are enough supporting points in the stress file. For the determination of the model constants there is always the need for valid measurement data.

5 Measurement Results and Interpretation

In this section selected results, extracted from the measurement data, are presented. The results are interpreted and the quality of the measurements is shown. All the data was gathered according to Section 2. Some of the data is presented in common linear I-V graphs and in semilogarithmic plots to show details in the subthreshold region. For presenting the impact of device degradation, the relative change of extracted parameters is plotted over time in a double logarithmic plot.

5.1 Accuracy and Reproducibility of the Measurements

It is of importance to show conclusively that the measurement setup delivered valid and reproducible data. The reproducibility of the measured data is an important issue as well.

5.1.1 Comparison to Existing Data

To prove the validity of the measurement setup some already existent measurement data was used for comparison. That data was kindly provided by the Quality Department of ams AG. It included some device characteristics as well as the change of certain parameters over time and was acquired with a different, independent measurement setup. The measured device was a n-MOSFET with the dimensions $W = 1$ μm and $L = 0.18$ μm, which was stressed at $V_g = 1.75$ V and $V_d = 2.4$ V for $t = 1e4$ s. Those conditions were reproduced with measurement M13 (cf. Table 1).

Figures 29 and 30 show the transfer characteristics for the initial device in saturation region ($V_d = 1.98$ V) and in linear region ($V_d = 0.1$ V) in an overlay with the existent data points for comparison. The new data is in very good compliance with the preexisting data for both plots. This indicates the validity of the measurement setup and the extraction routine. Small deviations at high current values may arise from different contacting methods which can account for a difference of a few percents. Also process variability causes small differences for devices from different lots.

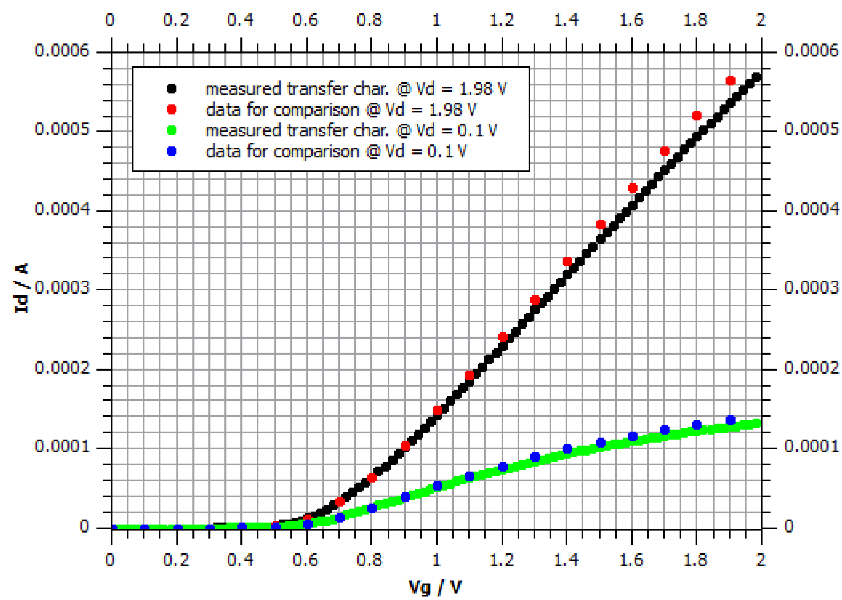

Figure 29: Comparison to existing data for initial transfer characteristics - linear plot (M13) [4]

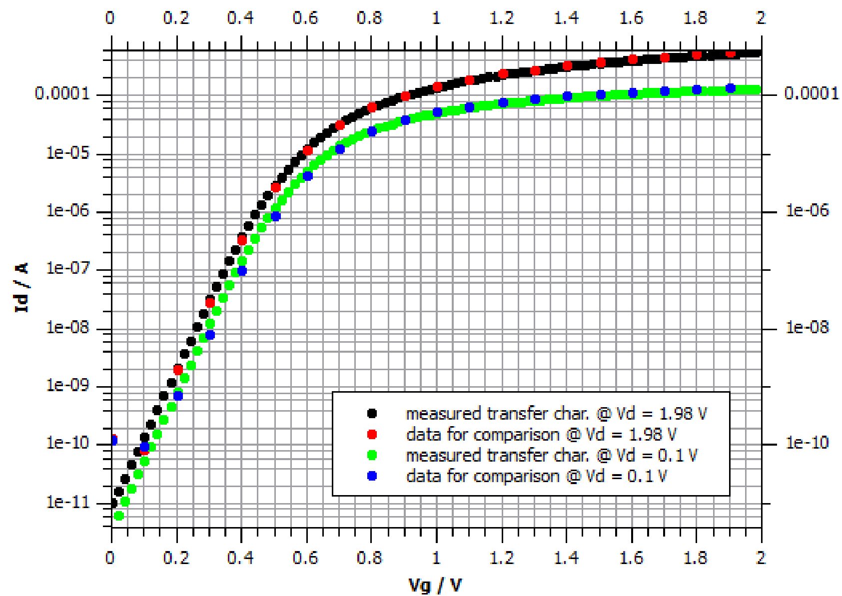

Figure 30: Comparison to existing data for initial transfer characteristics - semilogarithmic plot (M13) [4]

The second benchmark for the quality of the measurement setup is the validity of the device aging. To verify that, there is another set of graphs comparing the relative change of the parameters $g_{m,max}$, $I_{d,lin}$ and $I_{d,sat}$ to those of the existent measurement from Quality Department.

In Figures 31, 32 and 33 one can again observe good agreement between the two data series. The indication is now that not only the static measurements for the initial devices, but also the stressing sequence is working fine and delivers reliable data.

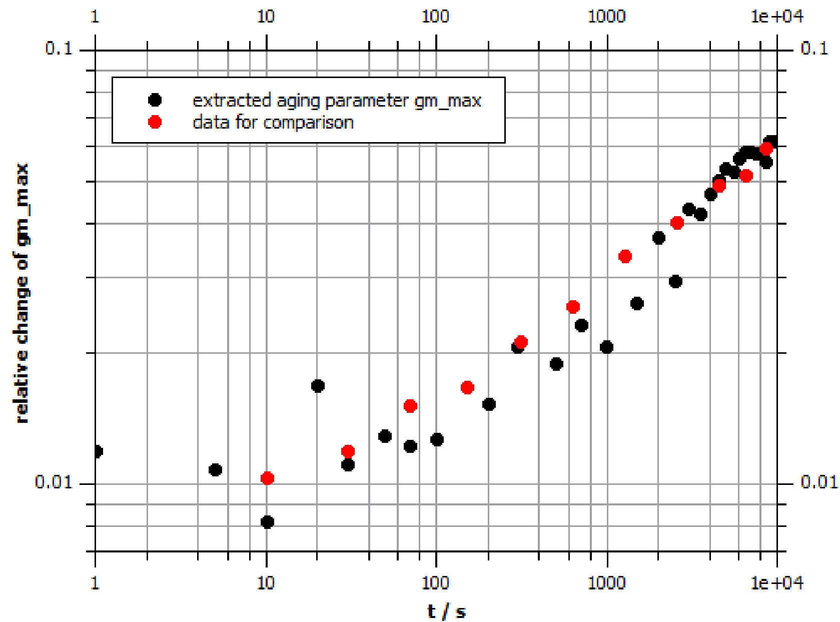

Figure 31: Comparison to existing data for the relative changes of $g_{m,max}$ due to aging (M13) [4]

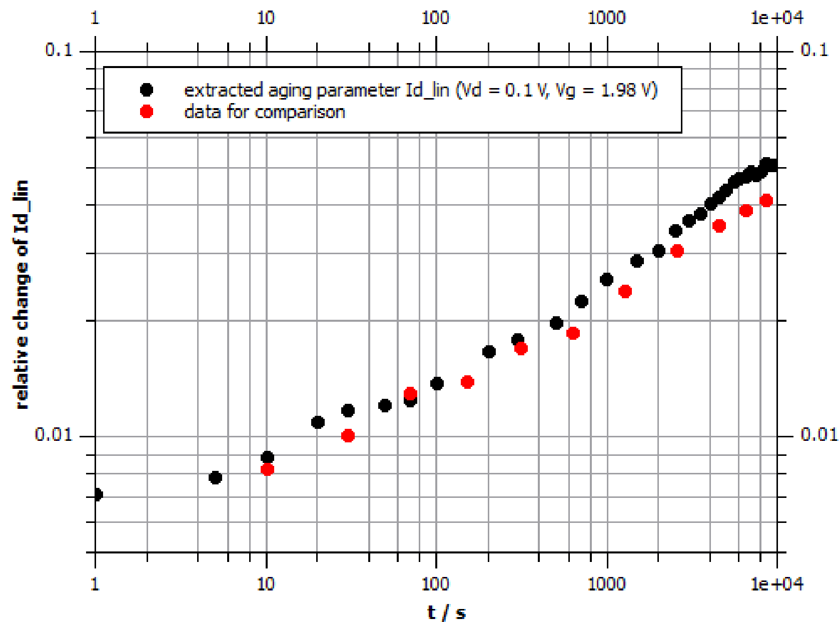

Figure 32: Comparison to existing data for the relative changes of $I_{d,lin}$ due to aging (M13) [4]

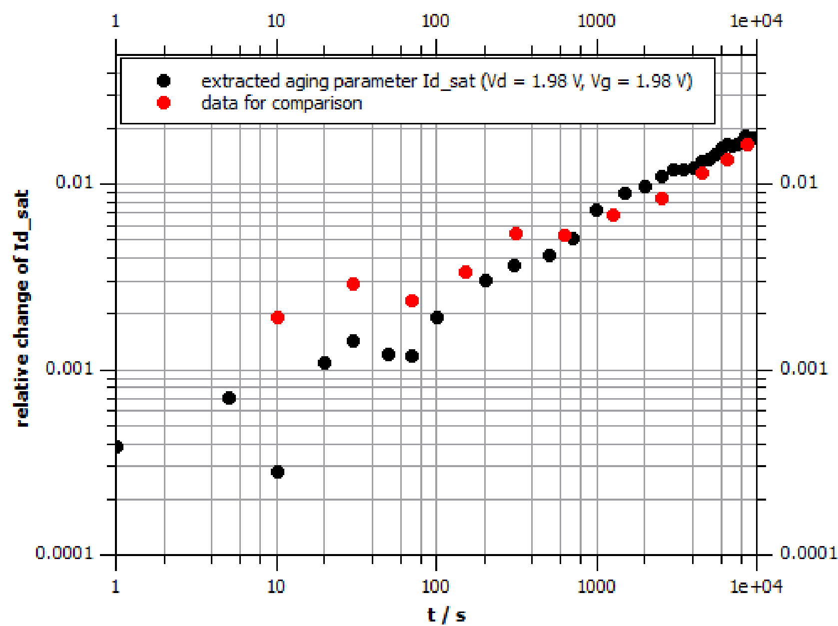

Figure 33: Comparison to existing data for the relative changes of $I_{d,sat}$ due to aging (M13) [4]

5.1.2 Reproduction of Results

Another important issue is the reproducibility of measurement results for identical devices and conditions. There is also the question about the comparability of data for the two measurement setups in use: the on-wafer approach and the packaged chip in the measurement box.

Figures 34 and 35 each compare two data series from on-wafer measurements (black and red) and one from a packaged device (green) for two different stress voltages V_d. Displayed is the relative change of the parameter V_{th} over time. In both graphs the results are very similar, only in Figure 35 the degradation of the packaged device is a bit higher. But for longer times the deviation is getting smaller. There is of course the possibility of slight differences in the results due to on-wafer variations of the process. For the on-wafer measurements dies from the outermost region of the wafer were not used. For the packaged devices this could not be done, since there was no numeration before cutting. Therefore it is possible that the device for MP05 originated from an edge or corner die with slightly different properties.

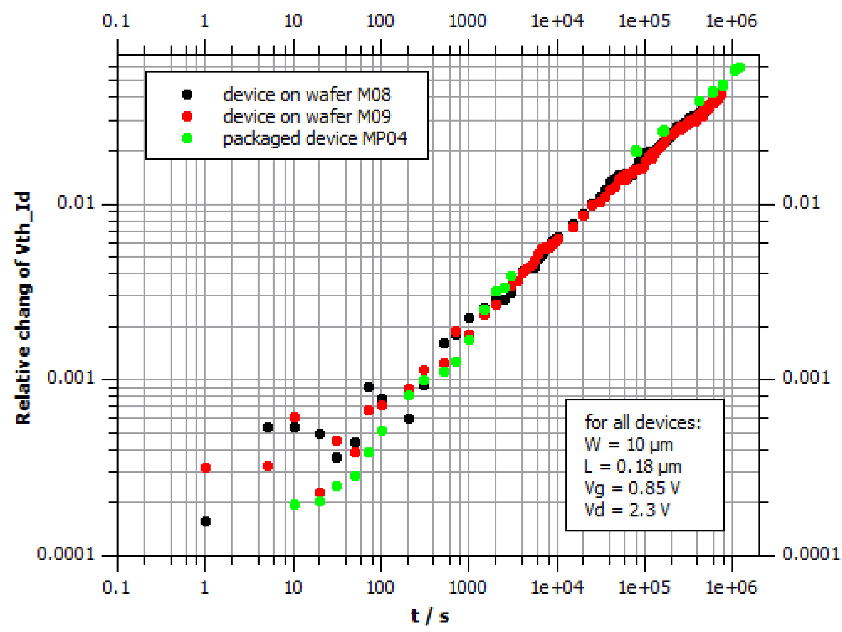

Figure 34: Comparison of relative change of V_{th} @ $V_d = 2.3$ V (M08,M09,MP04) [4]

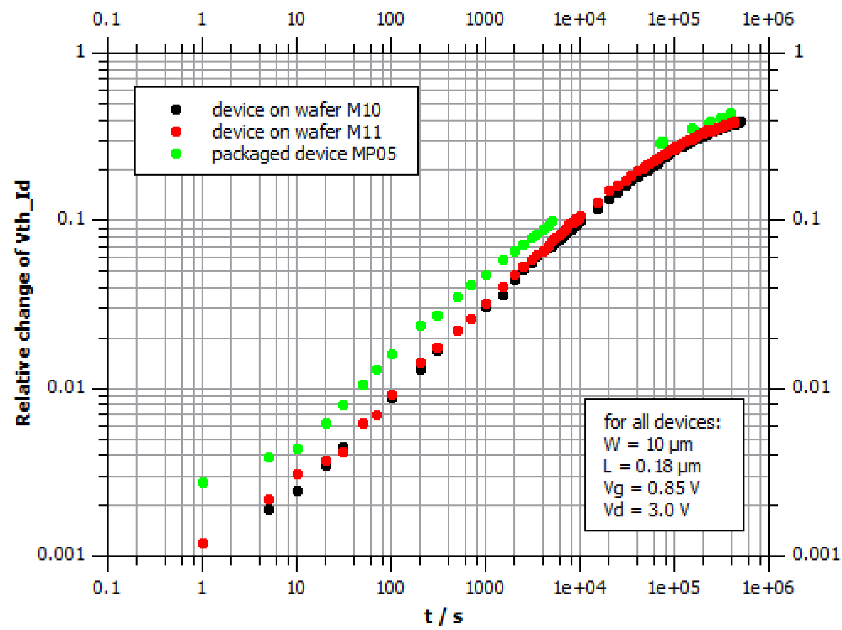

Figure 35: Comparison of relative change of V_{th} @ $V_d = 3.0$ V (M10,M11,MP05) [4]

5.2 Results for Various Stress Conditions

The results of various measurements with different aging conditions are presented and discussed here. An overview will be given on how the characteristics and the behavior of a nFET device are changed by continuous DC stress. As mentioned above, stressing at operation conditions (V_{dd} = 1.8 V) is not possible since the data acquisition would take far too long. The reasonable approach is to overstress the device with higher drain voltages and accelerate the aging process. From those results there is the possibility to extrapolate the magnitude of the effect for lower voltages.

5.2.1 Changes in Characteristic Curves due to Aging

Due to the applied DC stress the device characteristics undergo major changes. For reasonable parameter extraction processes it is absolutely necessary to know about those changes and their impact on the device. To give a good impression of what is happening there are graphs for the measured characteristics showing the initial data series as well as the curves after 100 s, 1000 s, 10 ks, 100 ks and 600 ks of stress. For better visibility of the degradation effects, the device was overstressed at V_g = 0.85 V and V_d = 3.0 V.

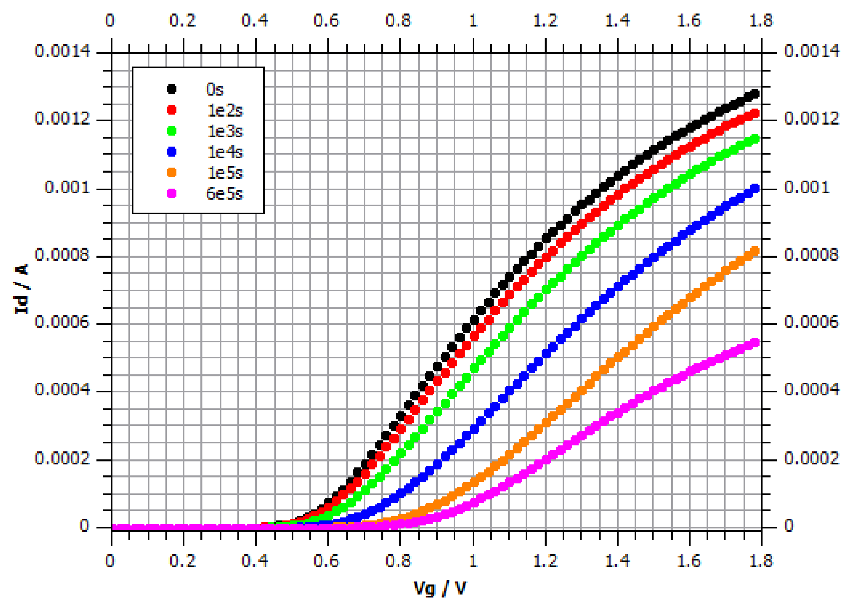

Figure 36: Transfer characteristics for various stress times, taken @ V_d = 0.1 V - linear plot (M10) [4]

Figure 36 shows how the transfer characteristic in the linear region changes with increasing stress time. It is clearly observable that the curve is not only shifted to higher

gate voltages. The slope changes as well. This change is more obvious for higher stress times, like for 6e5 s (purple dots).

Figure 37 shows the same data but in a semilogarithmic plot to give better view of the subthreshold region. Again the curve is not just parallel shifted but also changes its slope for increasing stress time.

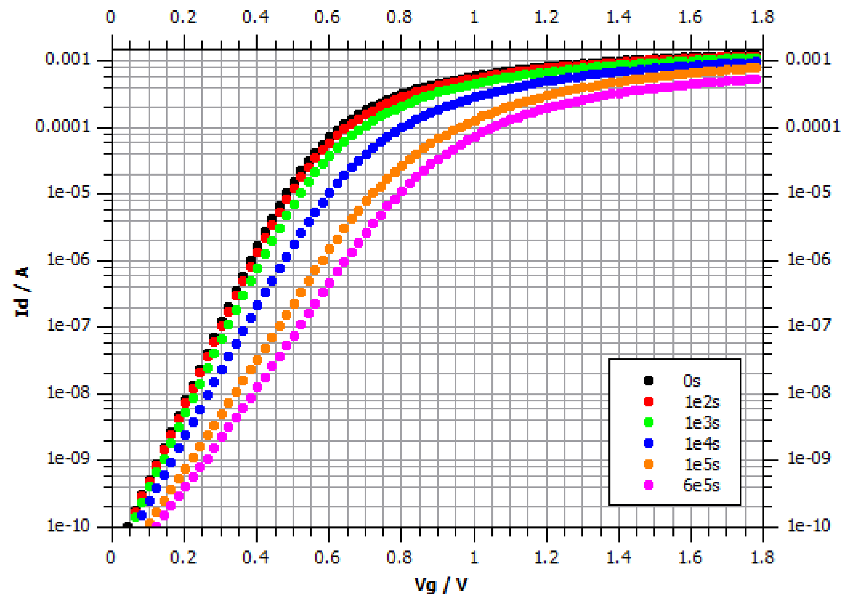

Figure 37: Transfer characteristics for various stress times, taken @ $V_d = 0.1$ V - semilogarithmic plot (M10) [4]

As explained above, the increasing amount of trap states near the bulk-oxide interface has some undesired effects on the device. When those traps are filled with electrons an electric field builds up which is weakening the field induced by the gate voltage. This means that for gaining the same amount of current while keeping V_d constant, the aged device needs to be supplied with a higher V_g. That is why the transfer characteristic shifts to the right in the displayed plots. The flattening in the slope of the aged curves directly corresponds to a decrease in conductivity. The channel resistance of the device is increased by aging.

Those changes have big impact on the device functionality. The parameter V_{th} for example is increased because a higher threshold voltage is necessary to switch the transistor on. A second, very important parameter is $I_{d,lin}$, which is the maximum current of the linear transfer characteristic (at $V_g = 1.8$ V). In Figure 36 for the initial device $I_{d,lin}$ is about 1.3 mA, after 6e5 s of stress it is below 0.6 mA. This is a reduction of over 50 %.

A circuit that relies on a certain current supply from the transistor may fails because of the degradation. This is where the device degradation is becoming a problem for real life applications.

In Figure 38 a transfer characteristic in the saturation regime is displayed after various times of applied DC stress. Again the change in form is more than just a parallel shift which is clearly visible for the curve at 6e5 s (purple dots). Again the important device parameters V_{th} and $I_{d,sat}$ are changed drastically.

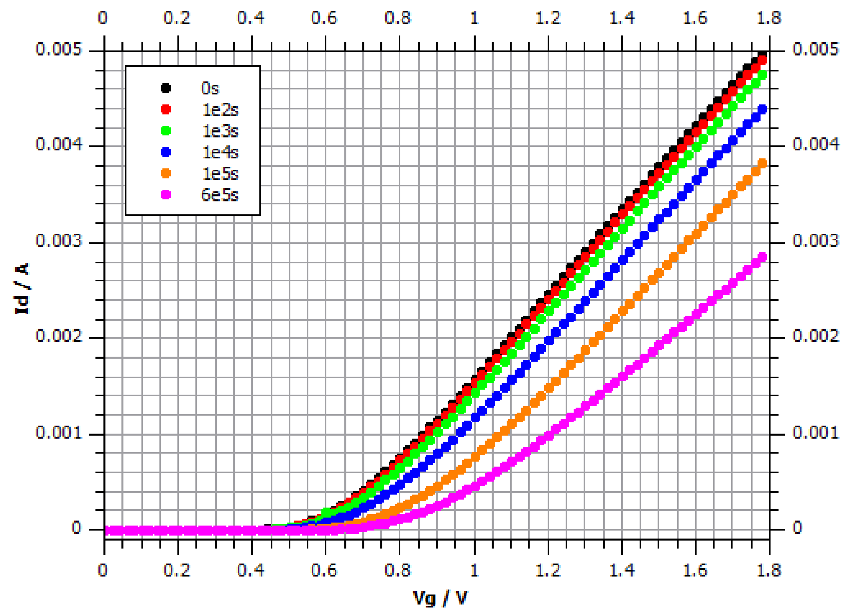

Figure 38: Transfer characteristics for various stress times, taken @ $V_d = 1.8$ V - linear plot (M10) [4]

The same is true for the semilogarithmic plot in Figure 39. Comparing this data to Figure 37, the characteristics in the linear regime, another issue is present. Though the drain voltage in the saturation regime is higher, the curves are closer together. Device degradation in the linear regime is worse than in the saturation regime. The reason for that was already explained during the introduction of the HiSIM2 model equations in Section 3.4.3. When the channel is partially pinched off in saturation mode, the electron density in this region is lower. A smaller amount of trap states is populated and the visible aging effect is diminished.

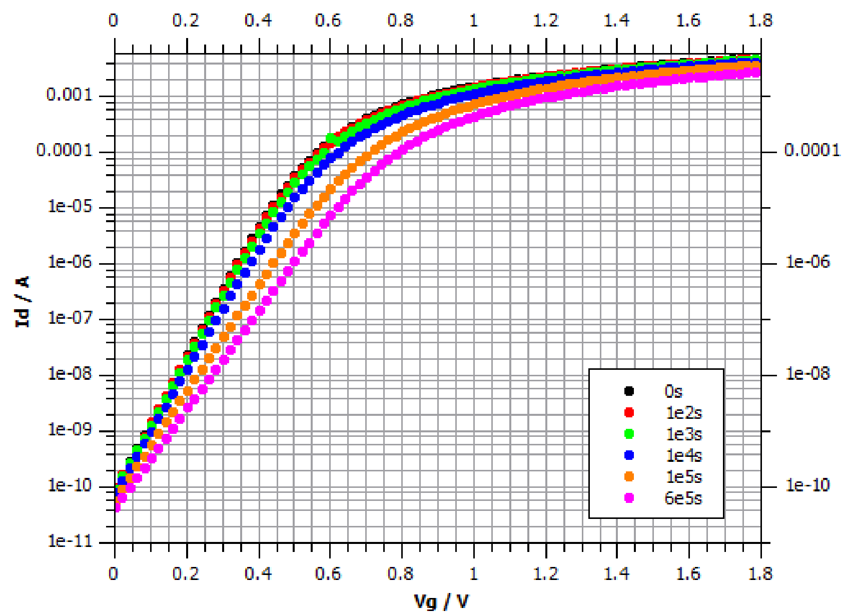

Figure 39: Transfer characteristics for various stress times, taken @ V_d = 1.8 V - semilogarithmic plot (M10) [4]

Figures 40, 41 and 42 show three different output characteristics and their evolution for increasing time of stress. The decrease in drain current over time of stress is plainly visible in all those graphs. Only the last two measurement series (orange and purple) in Figure 40 can not be taken into account, since due to the shift in V_{th} the applied gate voltage of V_g = 0.6 V is below the threshold. In this measurement series the point of V_{th} = 0.6 V was reached at around 3e5 s of stress.

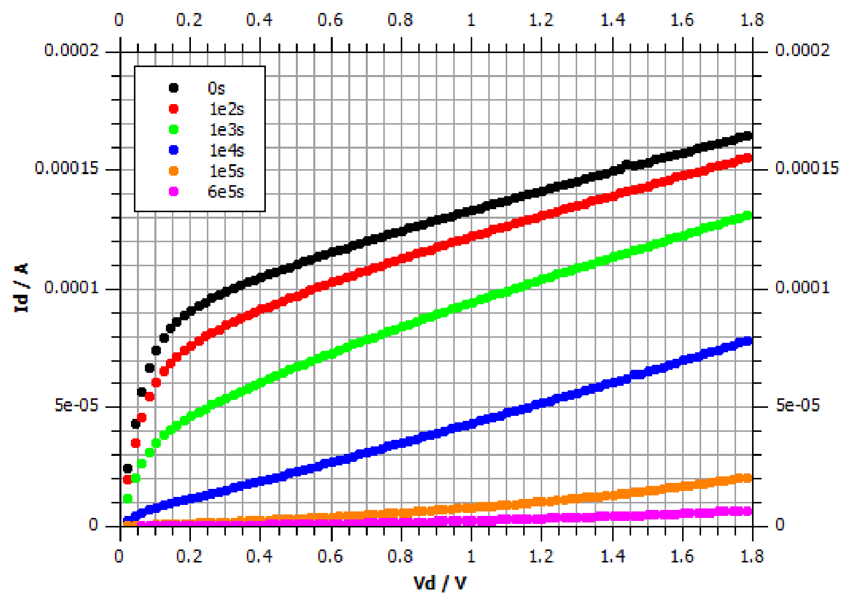

Figure 40: Output characteristics for various stress times, taken @ V_g = 0.6 V (M10) [4]

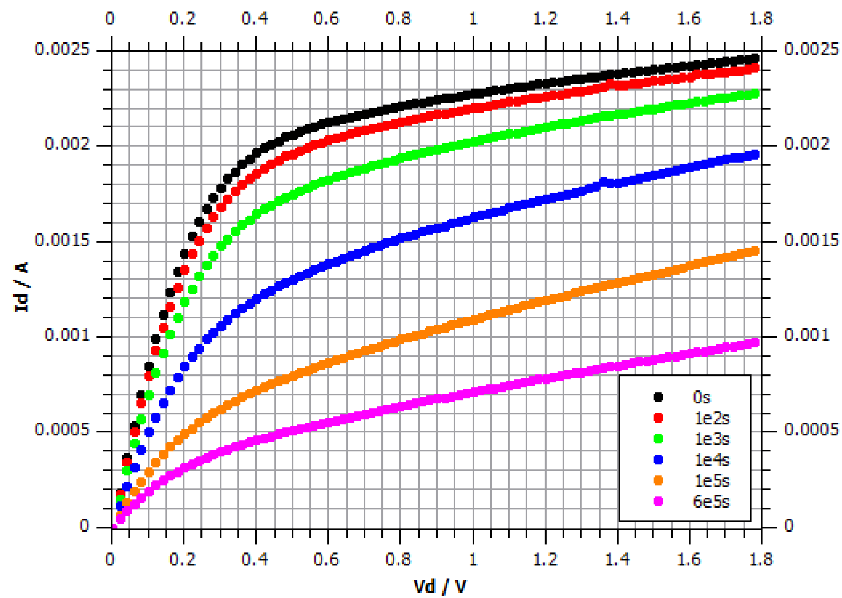

Figure 41: Output characteristics for various stress times, taken @ V_g = 1.2 V (M10) [4]

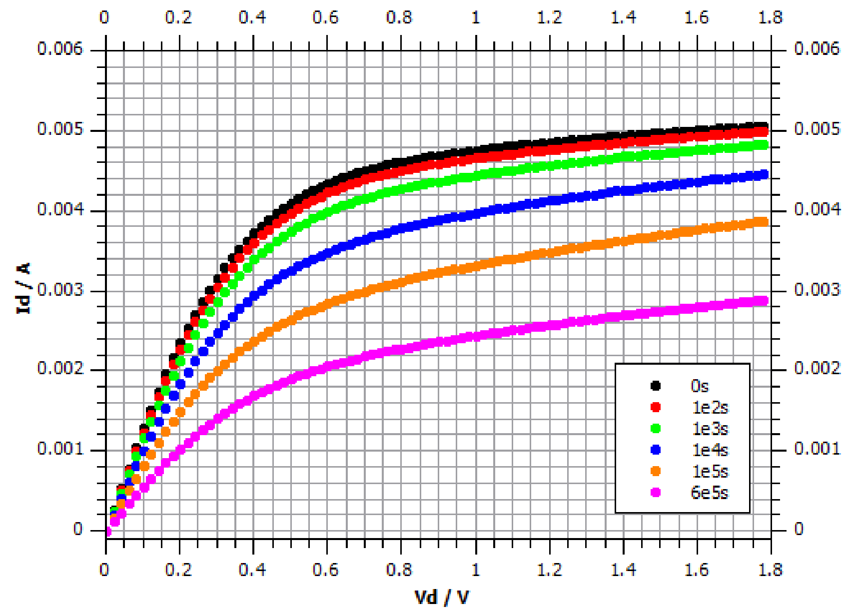

Figure 42: Output characteristics for various stress times, taken @ $V_g = 1.8$ V (M10) [4]

Above there is only parts of the ten different collected characteristics displayed. The others can be found in the Appendix 9.7.

5.2.2 Parameter Changes due to Aging

As shown above the device characteristics are strongly affected by the aging effects due to long term stressing. Subsequently all the extracted parameters change as well. The behavior of the transconductance g_m in Fig. 43 is connected directly to the transfer characteristics in Figure 36 and Figure 37 through the derivative:

$$g_m = \frac{dI_d}{dV_g} \tag{20}$$

The shift of the maximum value of g_m to higher voltages is a direct result from the shift of the transfer characteristic. Only at 6e5 s of stress (pink line) the maximum is not further shifted to the right. The shape of the corresponding transfer characteristic strongly differs from the others due to a vast decrease in slope. Also the value of $g_{m,max}$ decreases over time. If the maximum conductance through the channel is lowered, less current flows at the same bias conditions.

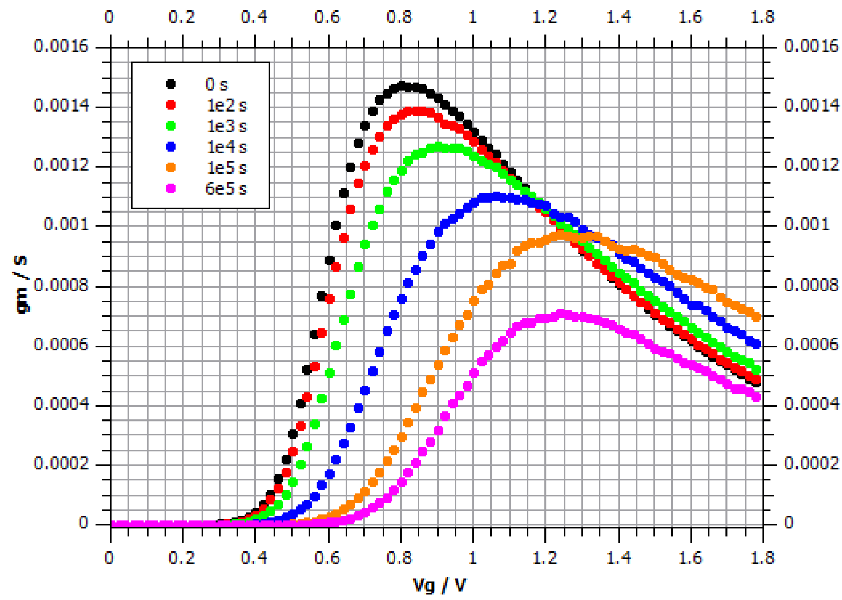

Figure 43: g_m for various stress times - extracted from transfer characteristics @ V_d = 0.1 V (M10) [4]

Figure 44 shows the relative change of $g_{m,max}$ over time for two different stress voltages. The black line for V_d = 3.0 V corresponds to Figure 43 above. To allow logarithmic scaling the relative change is plotted on the positive axis, although $g_{m,max}$ is actually decreasing. Even after only one second of stress there is a change of 1 %. For up to about 10 ks the change seems to increase exponentially (linear in the logarithmic plot). But then the slope declines and the relative change starts to saturate. This is clearly visible for the high stress (black dots) but also for lower stress (red dots) one can suspect the start of a saturation effect at long times of stress.

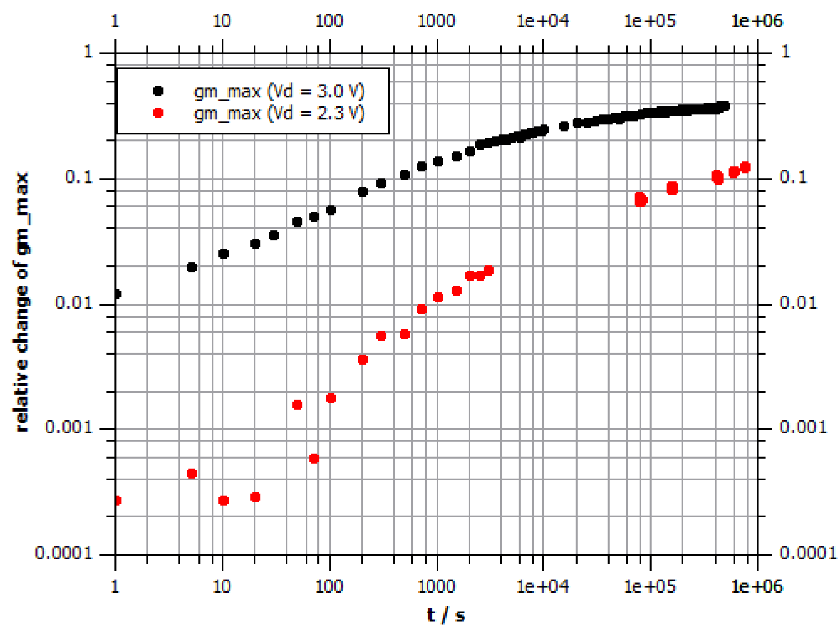

Figure 44: Relative change of $g_{m,max}$ over time of stress for two stress conditions(M10,MP04) [4]

An important parameter for device characterization is the threshold voltage V_{th}, since it defines when the MOSFET is switched on. As explained above for this work V_{th} was extracted using the constant current method. This is important to keep in mind, when comparing the plots to other results.

Figure 45 displays data series for the relative change of V_{th} over time of stress. A saturation effect for high stress (black dots) at long times is observable. For a lower stress voltage (red dots) this is not the case but it can be assumed that for stressing the device longer, a similar saturation would become visible. This decrease in the amount of change for long stress times was expected. The number of trap states at the interface is not increasing at a constant rate for all times. In the conducted measurements the stress time was chosen long enough to see a lowering of the this rate.

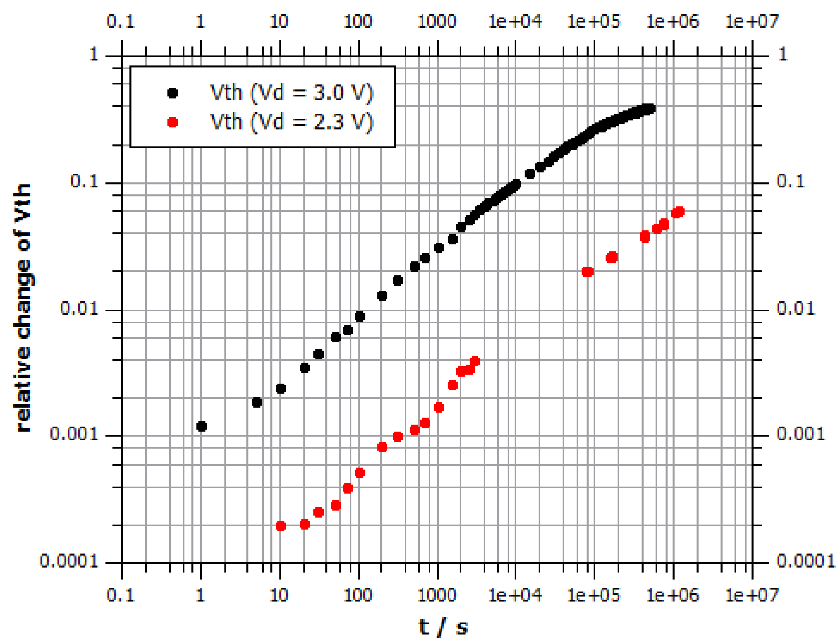

Figure 45: Relative change of V_{th} over time of stress for two stress conditions (M10,MP04) [4]

For the changes in the drain current parameters $I_{d,lin}$ and $I_{d,sat}$ in Figure 46 the explanations are again related to the difference in degradation for unlike drain voltages. The value extracted in the linear regime undergoes a bigger alteration than the one from saturation regime. $I_{d,lin}$ starts at higher values and exhibits a slightly different shape than $I_{d,sat}$. For the higher stress voltages (black dots) again a flattening at long times is observed. As for the same effect in the relative change of V_{th} this is yet another sign that the concentration of interface trap states saturates for long stress times.

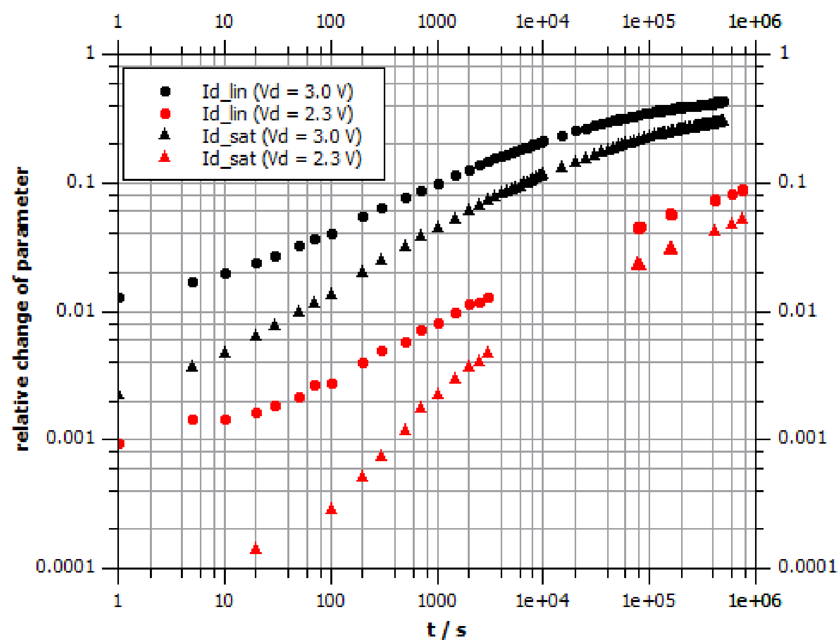

Figure 46: Relative change of $I_{d,lin}$ and $I_{d,sat}$ over time of stress for two stress conditions (M10,MP04) [4]

5.2.3 Extrapolation to Operating Conditions

There is one big problem with all the presented measurements up to this point. The device was always overstressed, meaning the stress voltage at the drain is far above the maximum operating voltage. For the ams AG 0.18 μm low volt process it is $V_{dd} = 1.8$ V. In the beginning of this experiment it was tried to age the devices at stresses close to V_{dd}, but it was soon clear that such measurements would take by far too much time. The amount of change was too small to see all the desired degradation effects. Because of that most of the measurements were conducted at $V_d = 2.3$ V. Data for $V_d = 3.0$ V, $V_d = 2.7$ V and $V_d = 2.1$ V was collected as well. To gain knowledge about the behavior at lower voltages, the gathered data was used to extrapolate the relative parameter change down to operating conditions for doing a reasonable lifetime estimation.

Figure 47 displays the measured relative changes in V_{th} for the four different stress conditions. Furthermore it contains fits through the data and an extrapolated fit for $V_d = 1.8$ V. For evaluation, first the measurement data was plotted. Since the data points were found to be on a straight line in the logarithmic plot, a power function was used to fit the threshold voltage shift over time:

$$\Delta V_{th} = A \cdot x^{\frac{1}{B}} \tag{21}$$

Apart from the saturation effect at long times for higher stresses, the patterns were aligned parallel in the logarithmic plot. Therefore a common value for the slope parameter B was found: $B = 1.99 \pm 0.03$. The fit parameter A for the intersection with the y-axis was derived for all of the four stress conditions by fitting the measurement data points:

$$V_d = 3.0 \text{ V}: \quad A = (9.8 \pm 0.5) \cdot 10^{-4} \text{ s}^{-1} \text{ (black line)}$$
$$V_d = 2.7 \text{ V}: \quad A = (3.6 \pm 0.3) \cdot 10^{-4} \text{ s}^{-1} \text{ (green line)}$$
$$V_d = 2.3 \text{ V}: \quad A = (6.4 \pm 0.2) \cdot 10^{-5} \text{ s}^{-1} \text{ (red line)}$$
$$V_d = 2.1 \text{ V}: \quad A = (2.97 \pm 0.03) \cdot 10^{-5} \text{ s}^{-1} \text{ (blue line)}$$

Those values for the fit parameter A have been plotted in Figure 48 as a function of the corresponding drain voltage. Again the relation of the points in the logarithmic plot was found to be linear and the same fit function from Equation 21 was used. The parameters for this fit are: $A' = (2.2 \pm 0.5) \cdot 10^{-8}$ s^{-1} and $B' = 0.102 \pm 0.002$. With this approach it was possible to extract a value for the parameter $A_{1.8}$ at a drain voltage of $V_d = 1.8$ V: $A_{1.8} = (7 \pm 2) \cdot 10^{-6}$ s^{-1}.

$A_{1.8}$ was then used to add a new function for the stress voltage $V_d = 1.8$ V to Figure 47 by keeping it parallel to the other fits (same B value). Information about the device at operating condition, that could not be gathered by means of measurements, was available.

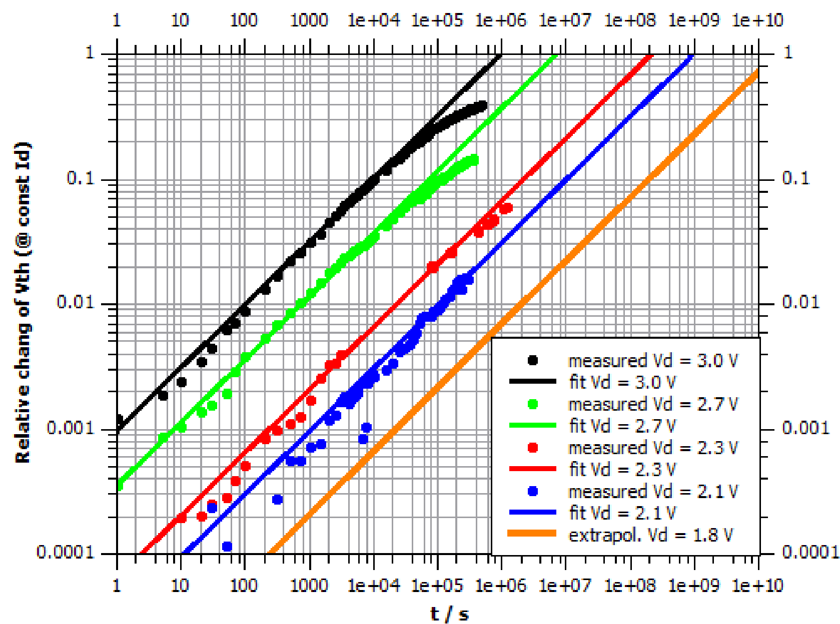

Figure 47: Extrapolation of degradation effect on V_{th} down to operating conditions (M10,M14,MP04,M18) [4]

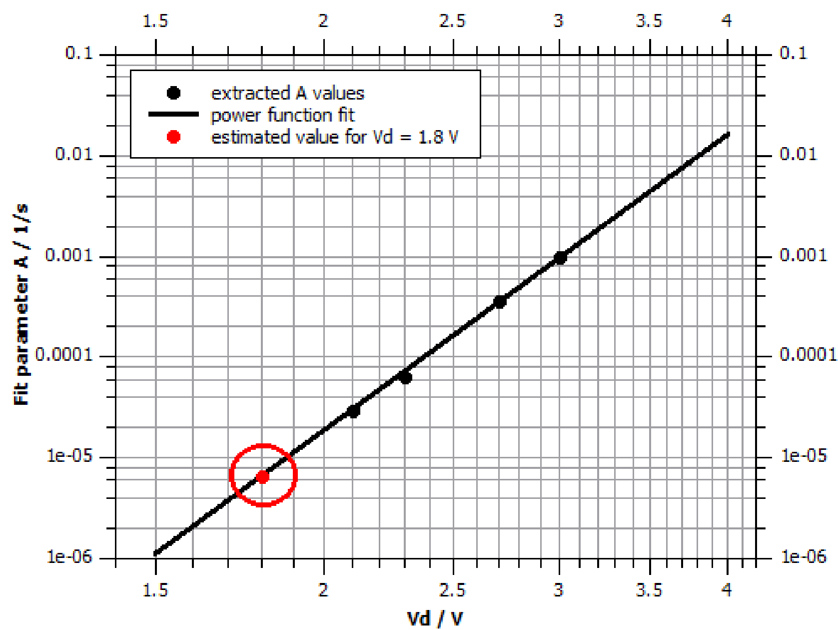

Figure 48: Parameter extraction for extrapolation to operating conditions [4]

With the additional information from Figure 47 it was possible to make estimations about the lifetime of a device used at maximum operating conditions (with the restriction of $V_g = 0.85$ V). After ten years of usage ($t \approx 3.2e8$ s) at a 100 % duty cycle the relative change of V_{th} is about 13 %. For a stress voltage of $V_d = 3.0$ V the same degradation is reached after about 30 ks (8 hours and 20 minutes). Therefore by over-stressing an estimation of long term degradation can be achieved with very little effort. There is the possibility of overestimating the degradation effects, since for higher stress times the curves start to saturate. For the green data points at $V_d = 2.7$ V a change of slope occurs even slightly before a relative change of 100 %. Mapped to a stress of $V_d = 1.8$ V the saturation of relative change would start at about $1e8$ s (3 years and 2 months) at a 100 % duty cycle.

However in a real life applications a duty cycle of 10 % is more realistic. In this case the estimated point of a relative change of $\Delta V_{th} = 10$ % would be at over 30 years of usage. This is far above the minimum demand of 10 years of life time which is industry standard for qualification of the process.

5.3 Width Variation

There were only two devices per die accessible with the same length but a different width, one with $W = 1$ μm and one with $W = 10$ μm. Aging measurements were conducted for both devices. Figure 49 shows the extracted relative change in threshold voltage. There is no essential difference between the two data series. Device degradation is obviously not dependent on the gate width, if the devices were manufactured with the same process. That is because the current density is the same. When the width is a factor 10 larger, the measured current is bigger by a factor 10 as well. For very narrow devices with a width of $W < 0.7$ μm, narrow-channel effects can occur.

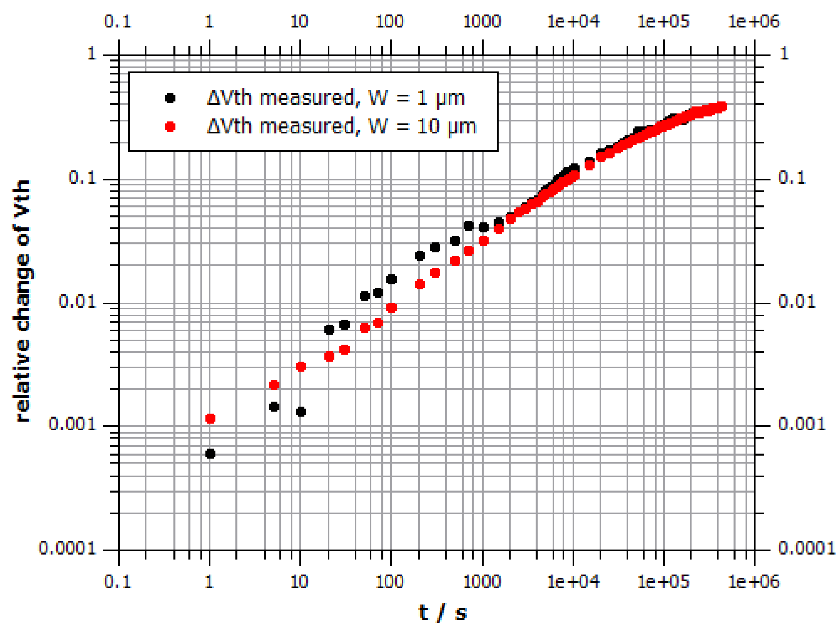

Figure 49: Relative change of V_{th} over time of stress for two different W (M11,M12) [4]

5.4 Channel Length Variation

For length variation there were three devices per die available with different channel lengths and the same width $W = 1$ μm. Those lengths were $L = 0.18$ μm, $L = 0.40$ μm and $L = 0.90$ μm. From the extracted relative changes in V_{th} in Figure 50 one can plainly see that the degradation effect is increasing with decreasing channel length. For a constant drain voltage, in a device with a shorter channel the electric field is higher. The electrons accelerate to higher velocities before they scatter. Furthermore in short channel devices the probability for electrons to scatter with phonons and impurities is less than for

longer channels. Electrons can reach the same or even a higher kinetic energy. Because of that the occurrence of hot electrons, which are able to create interface traps, is more likely. Furthermore those high energy electrons can produce more hot electrons by means of electron-electron scattering. [28]

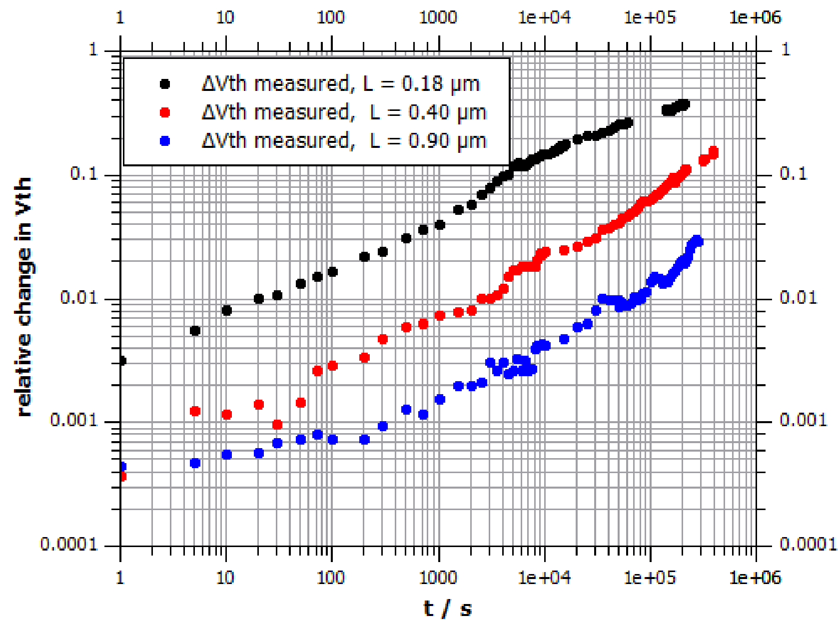

Figure 50: Relative change of V_{th} over time of stress for three different L (MP06,MP07,MP08) [4]

5.5 HiSIM2 HC Degradation Model Verification

Above in Section 3.4 the modeling approach of the HiSIM2 hot carrier degradation model was explained. The basis for the model is the assumption that the number of trap states at the bulk-oxide interface can be estimated by knowing about the substrate current I_{sub}. The amount of charge flowing out over the substrate contact is calculated by integrating I_{sub} over time of stress. The validity of this approach was checked by means of the aging measurements. During stressing of the device the parameter analyzer stored all currents on the four terminals every five seconds. Therefore the data for I_{sub} was accessible. With a Matlab script the substrate current during stress from measurement M10 was summed up over time to yield the overall charge Q_{sub}. The relative change of V_{th} was then plotted as a function of time and as a function of substrate charge. Figure 51 shows those plots. The graph shows that the two data series are in very good accordance

with each other. This is no proof for the validity of the model equations, but it makes the argument for the use of the model more compelling.

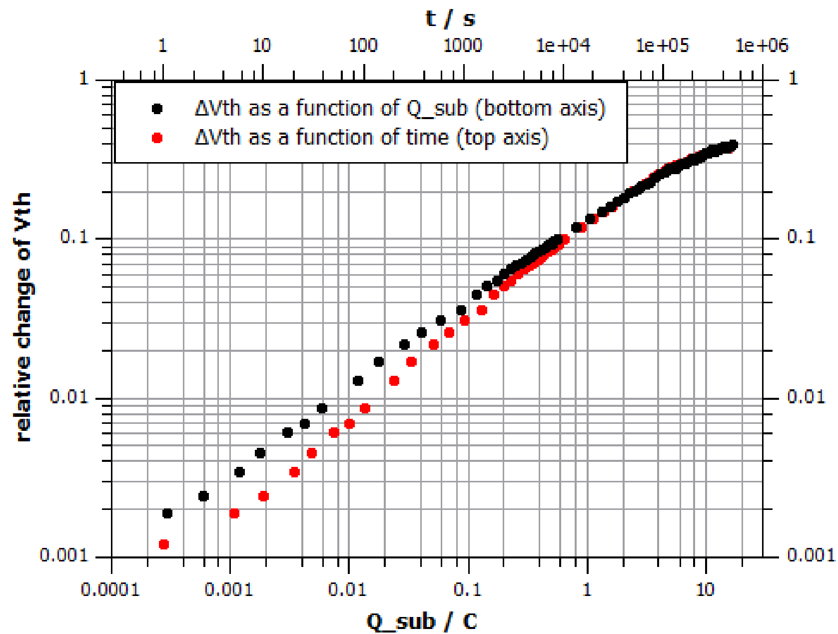

Figure 51: Relative change of V_{th} over time of stress (top axis) and over accumulated bulk charge (bottom axis) (M10) [4]

6 Aging Simulation Results

In this section the results from the various executed device degradation simulations due to hot carriers are presented. Most of them were obtained with the HiSIM2 aging model, since this was the main focus of this thesis. But also results from the modified Takeda aging approach are displayed. To have simulations comparable to the measurements presented in Section 5, over-stress with the conditions $V_g = 0.85$ V and $V_d = 3.0$ V was simulated. The evaluation of the measurement results and extrapolation to maximum operating conditions in Section 5.2.3 suggest that for the over-stressed device a stress time of 1e4 s is sufficient to simulate 10 years of usage. Therefore in the figures below there is always the measurement data (dots) and the according simulation result (solid line) plotted for $t = 0$ s and $t = 1e4$ s.

6.1 Simulated Changes in Characteristic Curves (HiSIM2)

The first presented simulations are characteristic curves of the initial and stressed device based on the HiSIM2 approach. Figures 52 and 53 show the linear transfer characteristic at $V_d = 0.1$ V in a linear and a semilogarithmic plot. For high V_g the simulation for the initial curve (green line) deviates from the measurement, for the subthreshold region on the other hand it is very accurate (cf. Section 3.3). The simulation of the aged device (blue line) is also good for the subthreshold region, but far off the measurement for the rest of the plot. According to the simulation the value for $I_{d,lin}$ would not change after 1e4 s of stress, which is proven wrong by measurement. Not just the end point is incorrect, but the trend for the whole curve. It is shifted too far to higher voltages and the slope is overestimated. For the thesis it was not possible to improve this result any further.

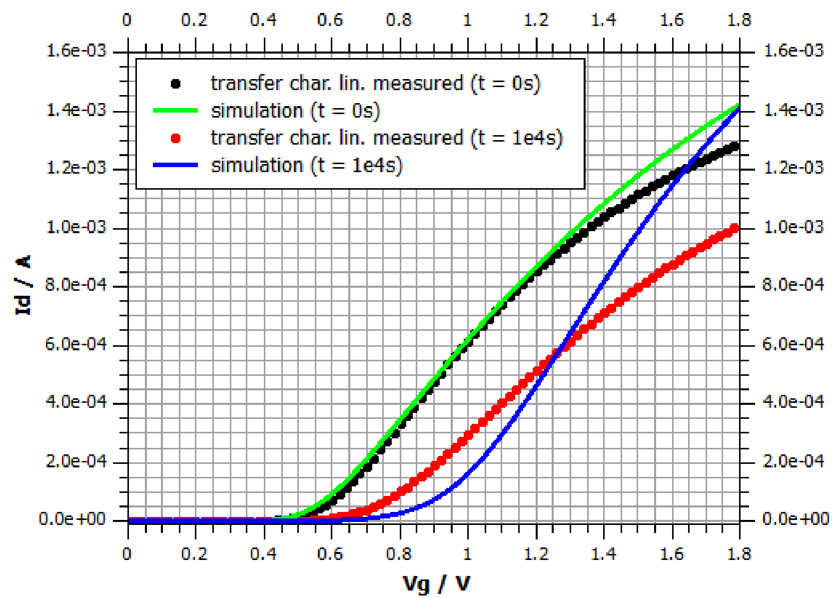

Figure 52: HiSIM2 v3.0.0 model transfer characteristics @ V_d = 0.1 V, V_b = 0 V compared to measurement results (M10) - linear plot [4]

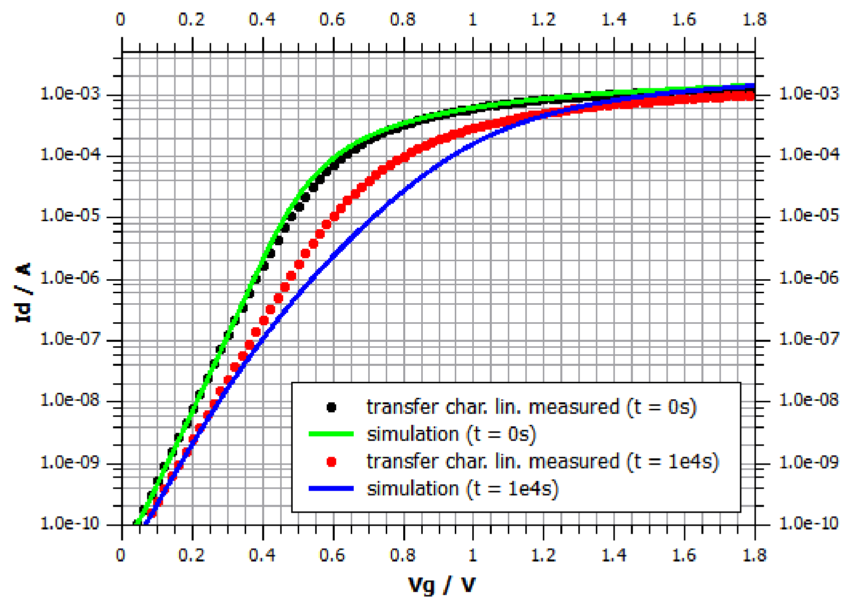

Figure 53: HiSIM2 v3.0.0 model transfer characteristics @ V_d = 0.1 V, V_b = 0 V compared to measurement results (M10) - semilogarithmic plot [4]

The simulations for the transfer characteristic in saturation at $V_d = 1.8$ V on the other hand display the opposite behavior. In Figures 54 and 55 both initial and aged curve are off the measurement in the subthreshold region. But for higher voltages they are in good accordance. The model seems to work well in the saturation regime while still having

inconsistencies for low V_d.

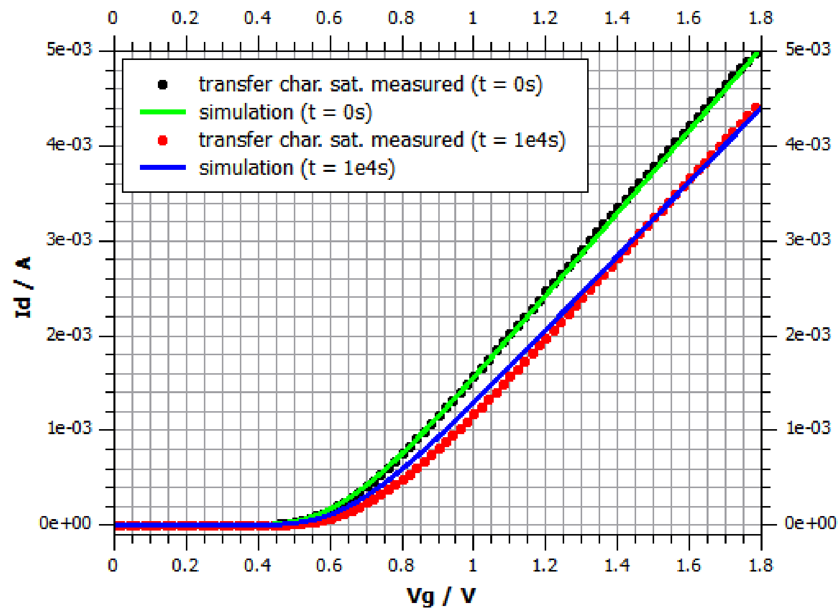

Figure 54: HiSIM2 v3.0.0 model transfer characteristics @ V_d = 1.8 V, V_b = 0 V compared to measurement results (M10) - linear plot [4]

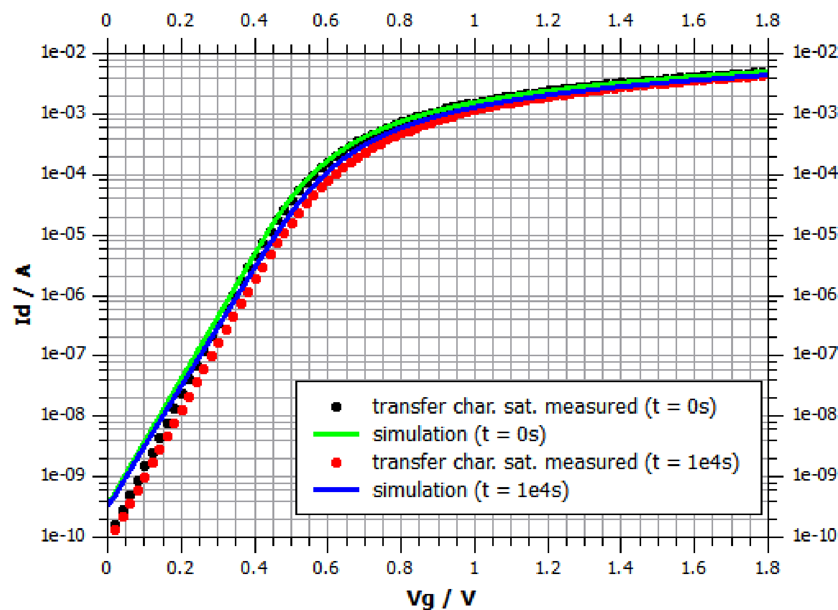

Figure 55: HiSIM2 v3.0.0 model transfer characteristics @ V_d = 1.8 V, V_b = 0 V compared to measurement results (M10) - semilogarithmic plot [4]

The problem at low drain voltages mentioned above, is visible in the output characteristic plots as well. Figure 56 shows an undesired ringing for the aged simulation at

low V_d. This is an indication of mathematical instability in the model. No converging solution was found. Trying to find the origin of the oscillations was not successful. The most likely assumption is an issue at the transition from the included aging model to the basic MOSFET part. Besides the ringing, the trend of the blue line is far from the measurement as well.

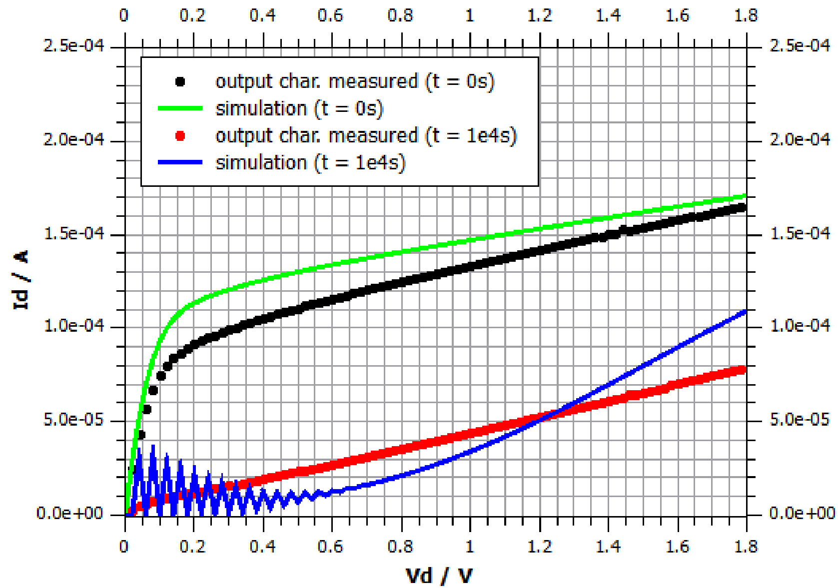

Figure 56: HiSIM2 v3.0.0 model output characteristics @ $V_g = 0.6$ V, $V_b = 0$ V compared to measurement results (M10) [4]

At higher gate voltage the issues of the model become less intense. In Figure 57 there is still a small ringing effect close to the origin but the result stabilizes. Between $V_d = 0.2$ V and $V_d = 1.4$ V the simulated aged curve exhibits an almost linear behavior. In this region the model is not able to represent the measured curve properly.

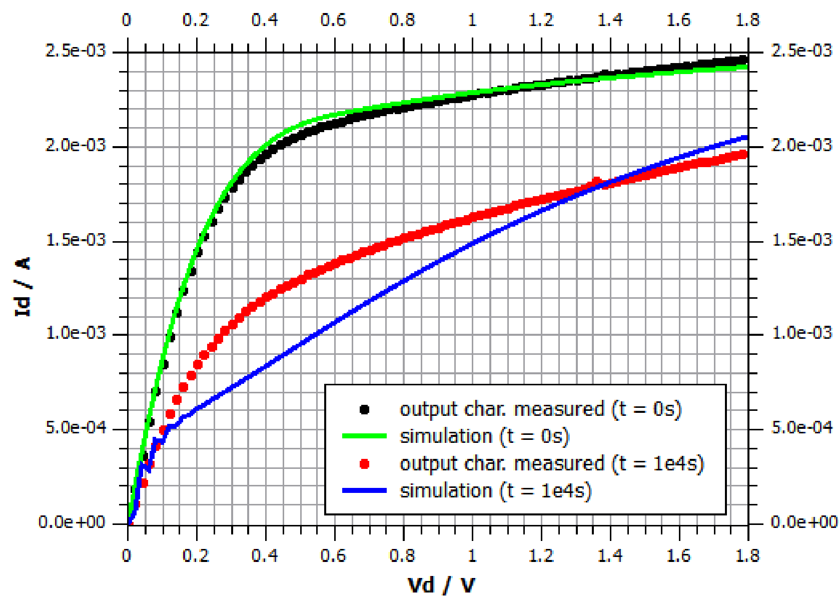

Figure 57: HiSIM2 v3.0.0 model output characteristics @ $V_g = 1.2$ V, $V_b = 0$ V compared to measurement results (M10) [4]

For the output curves at $V_g = 1.8$ V in Figure 58 the ringing is gone but other issues at small drain voltages occur. The simulation for $t = 1e4$ s is too close to the initial curve. This is the reason why $I_{d,lin}$ does not change in Figure 52 above. In the first part the gradient is too high while later on the curve becomes flat again. The simulation of slope with those two extremes is not close to measurement results.

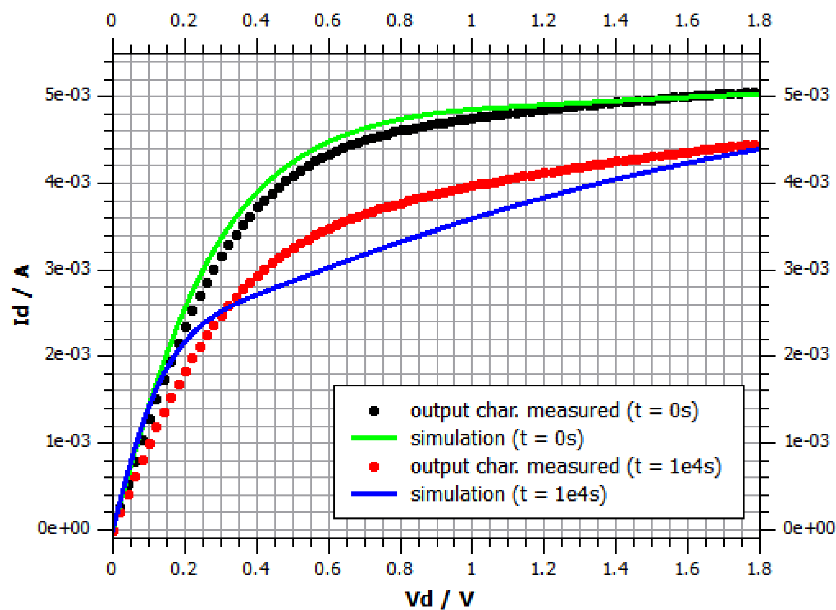

Figure 58: HiSIM2 v3.0.0 model output characteristics @ $V_g = 1.8$ V, $V_b = 0$ V compared to measurement results (M10) [4]

The conclusion of the simulations shown above is, that there are still issues in the HiSIM2 aging model approach present. The scalability with gate and drain voltages is not perfected yet. The ringing effects may point to instabilities in the mathematical background of the model. Since it is Beta version, it was not expected to deliver flawless results. Room for improvement is there and upcoming versions will surely deal with those issues. Simulations for the rest of the ten measured characteristics are available in the Appendix 9.8.1.

6.2 Simulated Parameter Changes (HiSIM2)

With a parameter extraction scripts written in Matlab it was possible to extract some parameters in question from the simulation results displayed above. To gather information about the trend, the extraction was done for 500 ks of stress time. The results are displayed together with the extractions from the measurements below in log-log plots.

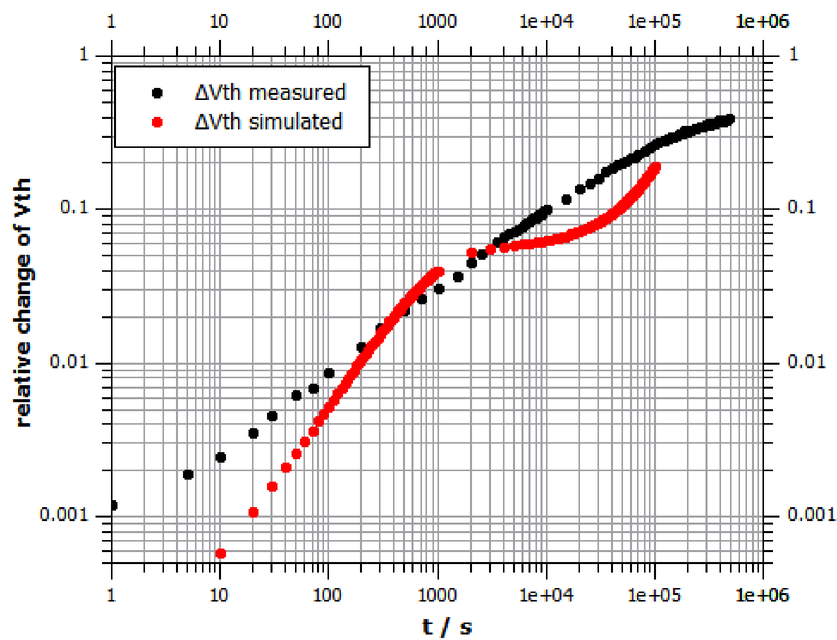

Figure 59: HiSIM2 v3.0.0 model relative change of V_{th} over time of stress compared to measurement results (M10) [4]

In Figure 59 the relative change of V_{th} is shown. The overall trend is close to the measurement. Except for a few points in the beginning, both data series share the same order of magnitude for all times. Looking at the details, there are some major differences. The simulated data displays are strong oscillations over time of stress. It changes between under- and overestimating ΔV_{th}. From the plot one can suggest that for longer stress times the oscillations would continue. The form of the data may again be an undesired side effect from the mathematical operations within the model. It was not possible to pin it to certain equation since analyzing the whole HiSIM2 MOSFET model was no goal of this thesis. The threshold voltage was extracted from the simulated transfer characteristics in saturation. That explains the good accordance to measurements, besides the oscillations, since the simulations for those characteristics were very accurate (cf. Figure 54).

For the relative change of the design parameter $I_{d,sat}$ in Figure 60 the same is true. It was extracted from the transfer characteristics in saturation as well and shows about the same qualities as ΔV_{th}. Again an oscillation is superimposing the trend of the data. The simulation is furthest off the measurement at around 1 ks of stress, but overall it gives a good representation of what was happening to the device.

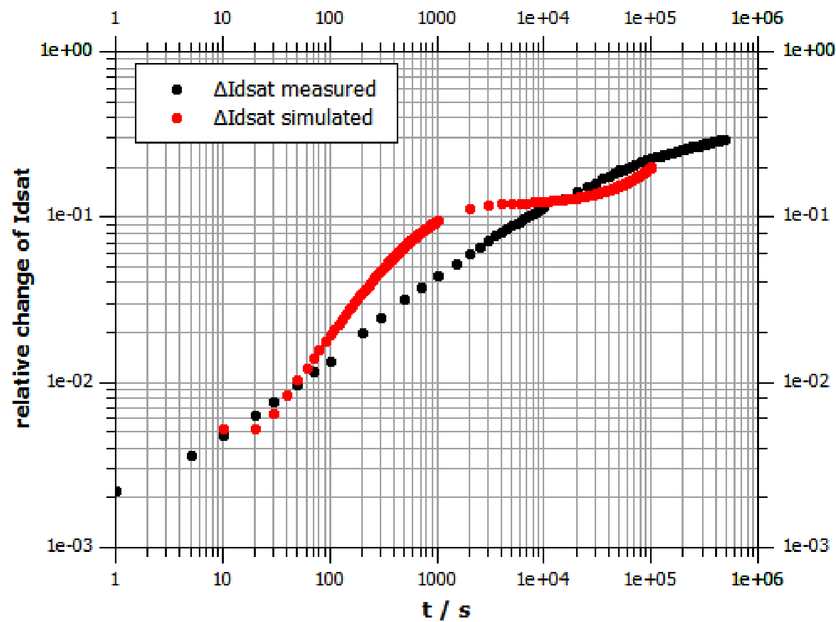

Figure 60: HiSIM2 v3.0.0 model relative change of $I_{d,sat}$ over time of stress compared to measurement results (M10) [4]

The change in $I_{d,lin}$ was extracted from the linear transfer characteristics. For those the simulation results could not provide the expected quality (cf. Figure 52). Therefore the data for $\Delta I_{d,lin}$ displayed in Figure 61 is not close to the real device. This time the absolute change was plotted due to the fact the sign changed. That could not be shown in a logarithmic plot. The axis is inverted to indicate that $I_{d,lin}$ is decreasing. Again the superimposed oscillation is found. Besides that, the trend of the curve is completely unexpected. First $I_{d,lin}$ is increasing up to about 1 ks of stress, which could not happen in a real device. It starts to decrease slowly then until it is back to the initial value at about 10 ks of stress. This is the point in Figure 52 where the initial and the aged characteristic are close together. Afterwards there is a major decrease, overestimating what is happening in the device.

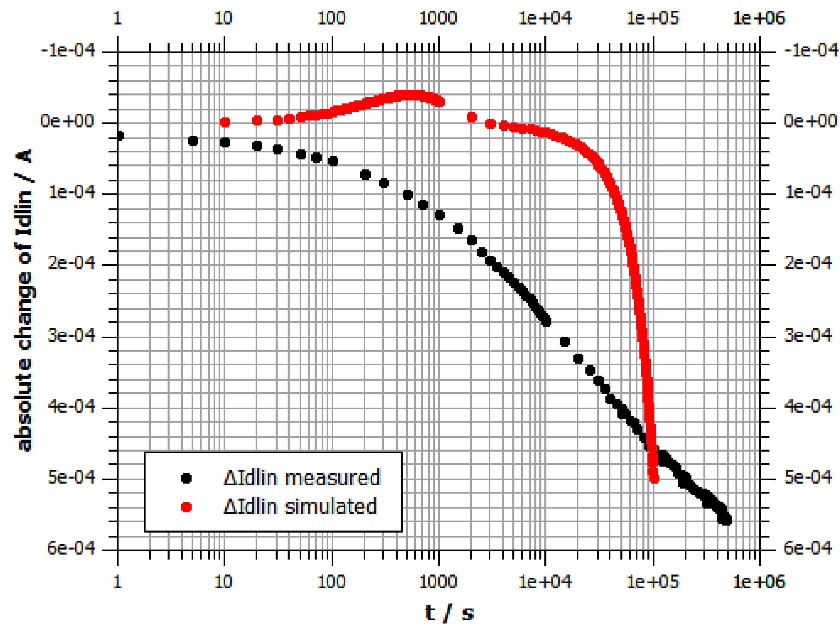

Figure 61: HiSIM2 v3.0.0 model relative change of $I_{d,lin}$ over time of stress compared to measurement results (M10) [4]

For the extracted parameter the model shows the same deficiencies in scaling for the drain voltage. The plots give a good indication on how the simulated characteristics may look like for all the time points. The time dependent evolution of the transfer characteristic in saturation, described by ΔV_{th} and $\Delta I_{d,sat}$, is close to reality. For the linear transfer characteristics, described by $\Delta I_{d,lin}$, there are issues present. To give an example $I_{d,lin}$ is increased instead of decreased at short stress times.

6.3 Simulated Width Variation (HiSIM2)

The stability of the model for variation of the width was tested for the standard device with the two available geometries $W = 1\ \mu m$ and $W = 10\ \mu m$. It was shown above that the aging effect is not dependent on W (cf. Section 5.3). The same is true for the simulation. Figure 62 shows the resulting data for relative change of V_{th}, which is about the same for both geometries.

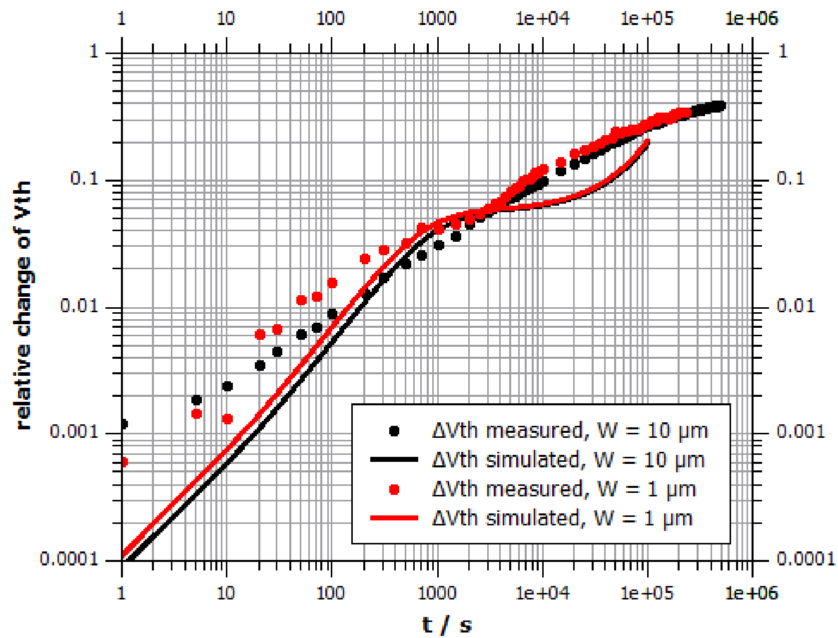

Figure 62: HiSIM2 v3.0.0 model relative change of V_{th} over time of stress for two different W (M11,M12) [4]

6.4 Simulated Channel Length Variation (HiSIM2)

To gain further information about the model, also the measurements for varying channel lengths (cf. Section 5.4) were simulated. Those devices had a width of $W = 1$ μm and lengths of $L = 0.18$ μm, $L = 0.40$ μm and $L = 0.90$ μm. The results in Figure 63, showing the relative change of V_{th}, give an indication on the length scalability of the model. The results do not give the expected trend, shown by measurement. The aging effect should decrease for longer channel lengths. But for $L = 0.18$ μm and $L = 0.40$ μm it stays about the same even with a slight increase for $L = 0.40$ μm. For $L = 0.90$ μm the trend is right again, the simulation shows a decrease in degradation. But only for time window between 10 s and 10 ks of stress. Above and below the result is again close to the other geometries. From the length variation simulations it can be concluded that the geometry scaling is another critical point of the aging model Beta version.

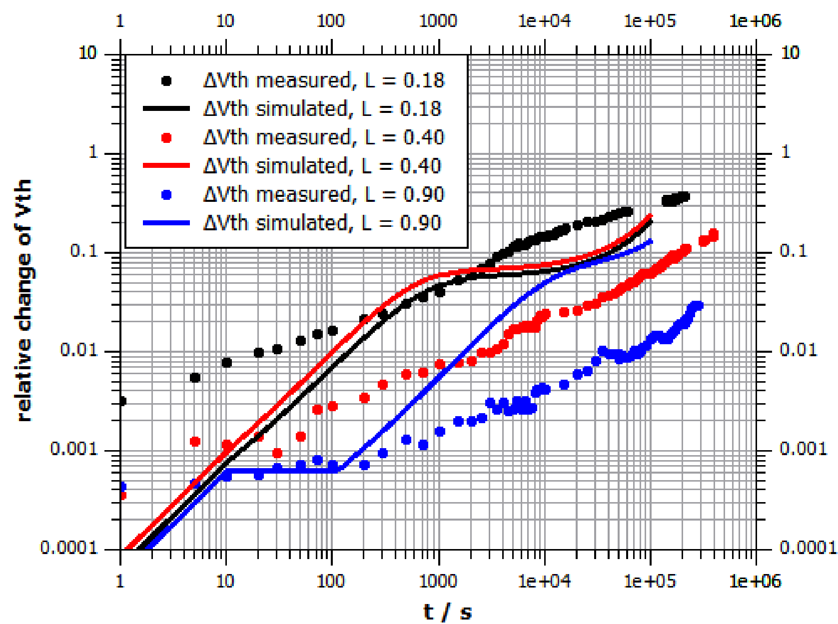

Figure 63: HiSIM2 v3.0.0 model relative change of V_{th} over time of stress for three different L compared to measurement results (MP06,MP07,MP08) [4]

6.5 Takeda Model - Simulation Results

Besides the HiSIM2 aging model, a second approach based on the Takeda model was analyzed. For simulating the device behavior the HiSIM2 MOSFET model was used here as well. Only the degradation calculation was done differently. The simulated stress duration was only 10 ms. The extrapolation up to 10 ks for the plots was done by the model. Some results for the characteristic curves are plotted below.

Figures 64 and 65 show the transfer characteristic in the linear regime. The simulation is very accurate in the subthreshold region and only slightly off for higher voltages. Despite the offset the slope and trend of the simulation comes close to the measurement.

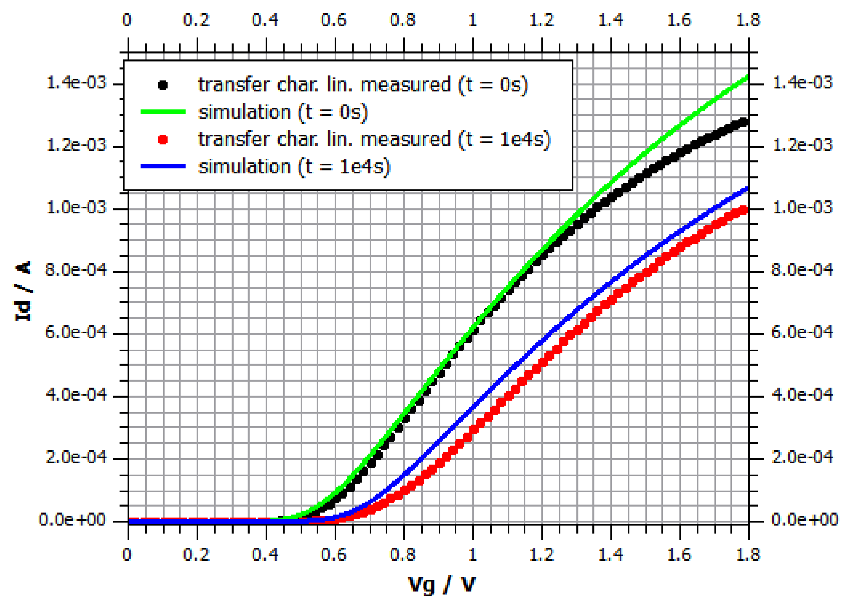

Figure 64: Takeda model transfer characteristics @ $V_d = 0.1$ V, $V_b = 0$ V compared to measurement results (M10) - linear plot [4]

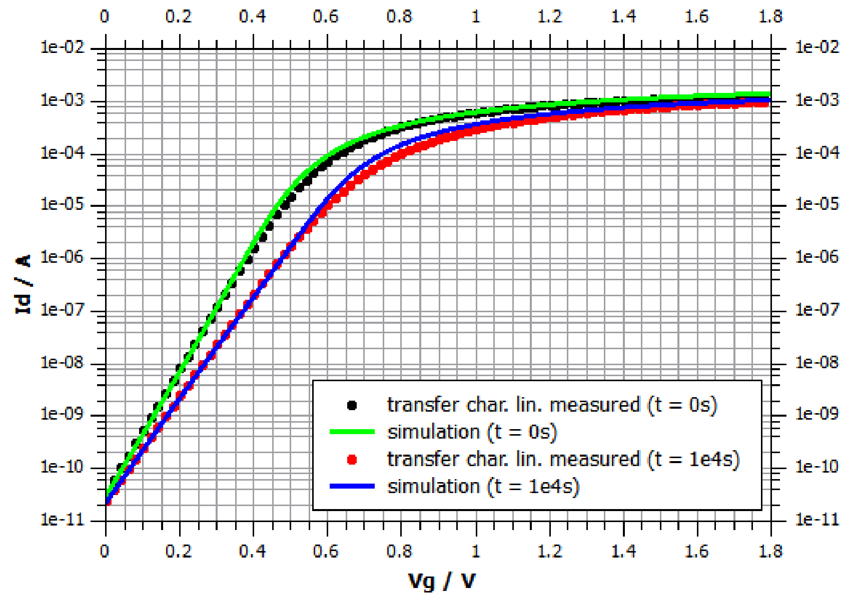

Figure 65: Takeda model transfer characteristics @ $V_d = 0.1$ V, $V_b = 0$ V compared to measurement results (M10) - semilogarithmic plot [4]

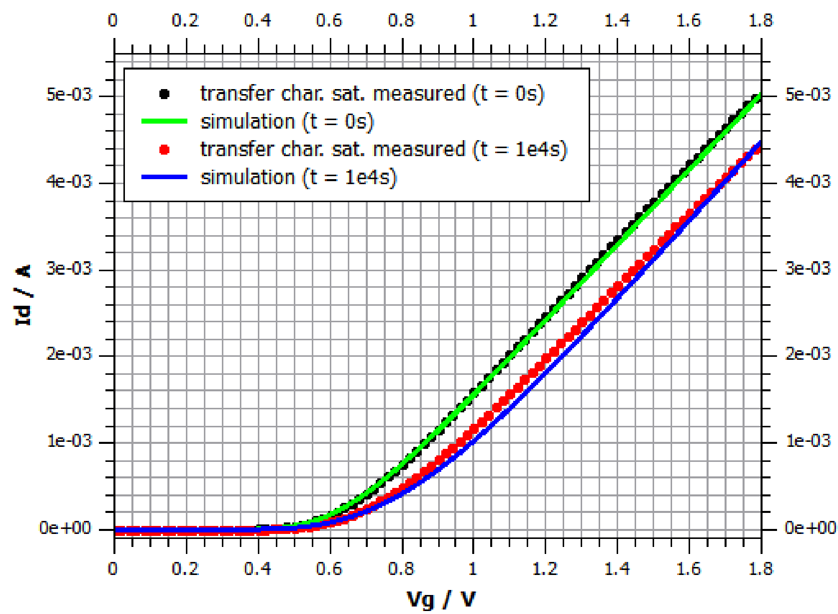

Figure 66: Takeda model transfer characteristics @ $V_d = 1.8$ V, $V_b = 0$ V compared to measurement results (M10) - linear plot [4]

For the transfer characteristic in saturation the simulation in Figure 66 is again in good accordance with the measurement above the threshold. In Figure 67 there is an obvious issue in the subthreshold region. The simulated lines for the initial and the aged characteristic cross at $V_g \approx 0.3$ V. This is not happening for the real device and is an undesired artifact from the Takeda simulation.

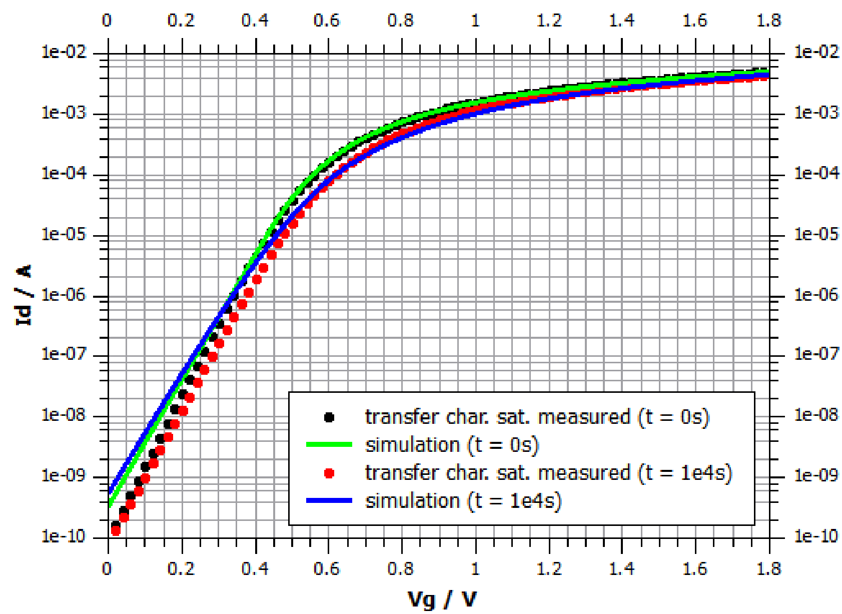

Figure 67: Takeda model transfer characteristics @ V_d = 1.8 V, V_b = 0 V compared to measurement results (M10) - semilogarithmic plot [4]

The degraded output characteristic simulations below give stable results in good accordance with the measurements. The issue with ringing in the subthreshold region is not present. This is another indication that this mathematical problem is connected with the interface between the HiSIM2 MOSFET model and HiSIM2 degradation model. Otherwise it should be visible here, since the MOSFET model is used with the Takeda approach as well.

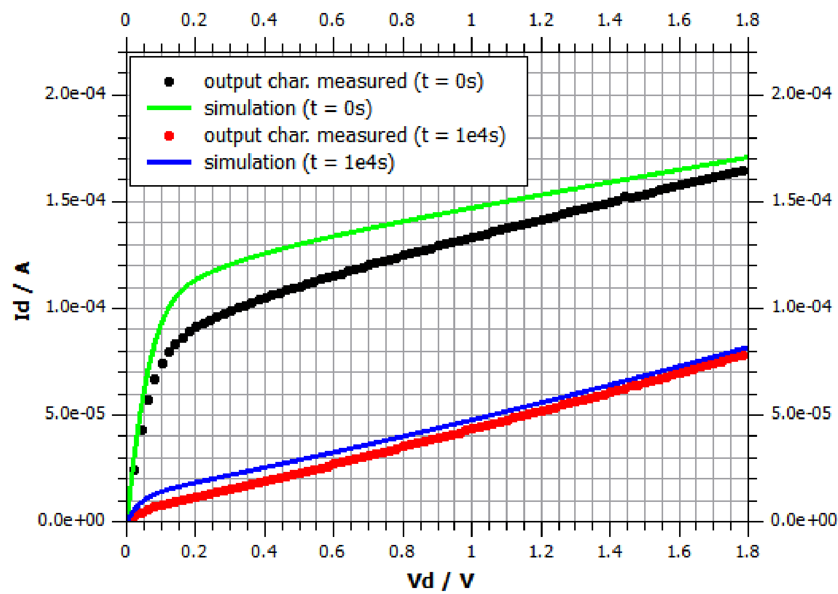

Figure 68: Takeda model output characteristics @ $V_g = 0.6$ V, $V_b = 0$ V compared to measurement results (M10) [4]

In Figure 68 the simulation for the aged curve is closer to the real device than the one for the initial curve. Though the trend looks good, this could also indicate a slight overestimation of the degradation effects.

The results in Figure 69 and Figure 70 are more accurate than the ones from done with the HiSIM2 model above. Still they display the same inaccuracy regarding the slope of the degraded curves. At low voltages the slope is overestimated and then for higher voltages it becomes too flat. Plots for the remaining three measured output characteristics are available in Appendix 9.8.2

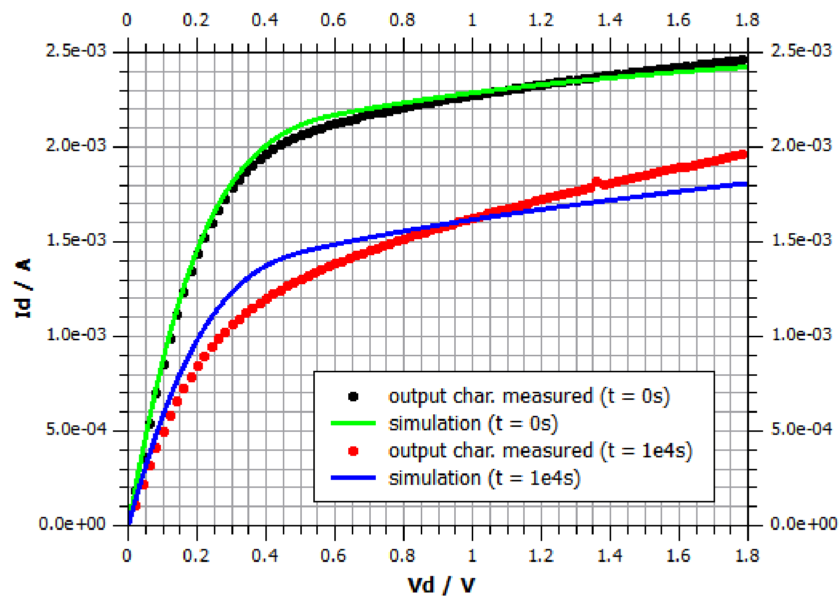

Figure 69: Takeda model output characteristics @ V_g = 1.2 V, V_b = 0 V compared to measurement results (M10) [4]

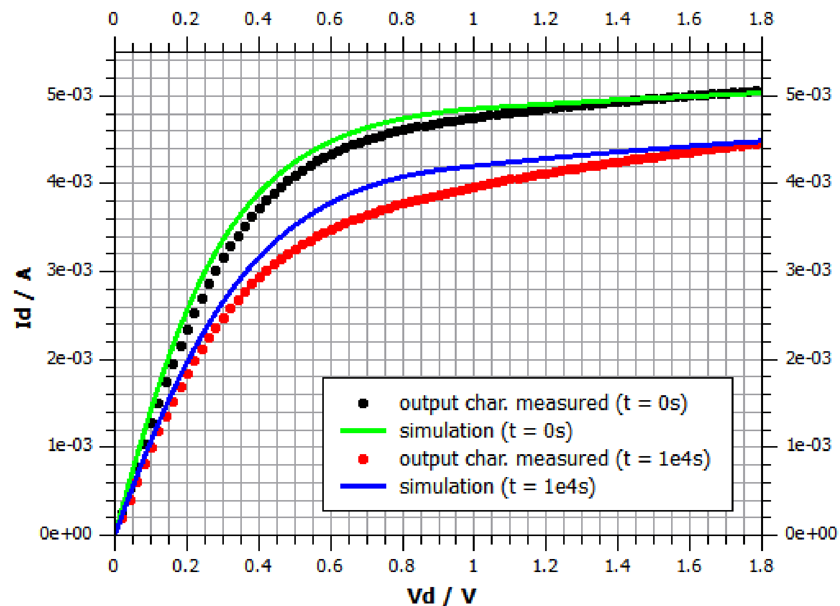

Figure 70: Takeda model output characteristics @ V_g = 1.8 V, V_b = 0 V compared to measurement results (M10) [4]

By comparing the plots above, the Takeda approach seems to deliver more stable and more accurate results. But there are some downsides as well. The elaborated formulation is only valid for one set of degradation parameters (gate voltage, drain voltage and stress time). The model is not scalable, which means that the setup needs to be updated

for every simulation. This includes all the model parameters and weights. Simulation for different conditions would be very time intensive because of that. Also the physical accuracy of this approach needs to be questioned, since results are produced mostly by fitting to the measurements. The building of a model for prediction of degradation without measurement data at hand is not possible. But that is the idea behind the HiSIM2 degradation approach: To predict changes by knowing only about the device and a set of physical parameters. Even if some of the results are not good enough by now, the overall approach is superior because the goal is to gain results by taking the device physics into account.

7 Conclusions

Aging effects due to hot-carrier-injection in MOSFETs were investigated and the experimental results were compared to state-of-the-art aging models. Within a short period of time we were able to build a functional and reliable measurement setup for conducting DC stress measurements on n-MOSFET devices. It consisted of two settings: the on-wafer measurement and the measurement of packaged devices. For both settings long term stress measurements up to 1 Ms of stress time were possible. The only limiting factor was the availability of the equipment. The reproducibility of results was shown as well as the accuracy of the setup by comparison to available, external data. We extracted various parameters from the measurement data and reconstructed the degradation of the devices due to hot carrier induced aging. An important result was the extrapolation of degradation effects in an overstressed device down to real-life operating conditions.

The Beta version of the HiSIM2 MOSFET hot carrier degradation model was used to simulate our measurements. The goal was to evaluate the functionality and accuracy of the model. The HiSIM Research Group from Hiroshima University supported us by immediately addressing issues, which emerged during testing. The HiSIM2 aging approach is meant to be physically more accurate than other SPICE models. Still the results of our tests revealed a few remaining challenges in the Beta version. While the accuracy of the simulation was good in the saturation regime of the MOSFET device, it contained undesired artifacts in the linear regime. To sum it up, the approach has great potential due to its physics based description of the device and the degradation effect, but for the time being it is not fully developed. Nonetheless, considering it was the first industry testing, the HiSIM2 degradation model performed well and exceeded our expectations.

The second modeling tool implemented was based on the Takeda model and was also used to simulate the measurements. We combined a pure heuristic approach with the HiSIM2 MOSFET model. At first sight the results obtained looked promising. The accuracy in all operating regimes of the MOSFET devices was good. The model was able to extrapolate the device behavior after a short simulated sequence of AC or DC stress to long times with good accordance to the measurement results. Limitations for this first version of the model were found regarding its scalability for various stress parameters. In the current state the model parameters would need to be determined for every device geometry and every operating condition by comparison to available measurement data. For

a reliable usage in industry further improvements and developments would be necessary.

The investigated models performed well, regarding the early stage in their development. There is certainly the possibility for them being used for industrial device degradation prediction in the future. Furthermore our work for this thesis provided our partners at ams AG with new insights into the effects of hot carrier degradation on n-MOSFET devices. The measurement setup we built may be used to gain information on device aging in future process development flows.

8 References

[1] Created with: Microsoft Corporation. Powerpoint 2016.

[2] Created by Dr. Martin Knaipp. Technology CAD Simulation.

[3] Created with: Microsoft Corporation. Excel 2013.

[4] Plotted with: Ion Vasilief. QtiPlot.

[5] Hiroshima University HiSIM Team. *HiSIM2 3.0.0 β*, May 2016.

[6] Frank Thuselt. *Physik der Halbleiterbauelemente - Einführendes Lehrbuch für Ingenieure und Physiker*. Springer-Verlag Berlin Heidelberg, 2 edition, 2011.

[7] Yannis P. Tsividis. *Operation and Modeling of the MOS Transistor*. McGraw-Hill Book Company, 1987.

[8] B. Van Zeghbroeck. Principles of Semiconductor Devices. `http://ecee.colorado.edu/~bart/book/book/chapter7/ch7_4.htm`, 2011.

[9] Peter Hadley. University lecture: Physics of Semiconductor Devices. `http://lampx.tugraz.at/~hadley/psd/lectures15/lectures.php`, 2015.

[10] Manish Prabhakar Pagey. Characterization and Modeling of Hot-Carrier Degradation in Sub-Micron nMOSFETs, August 2002.

[11] Jagdeep Shah. *Hot Carriers in Semiconductor Nanostructures: Physics and Applications*. Academic Press Inc., 1250 Sixth Avenue, Sand Diego, CA 92101, 1992.

[12] Jagdeep Shah. *Hot Carriers in Semiconductor Nanostructures: Physics and Applications*. Academic Press Inc., 1250 Sixth Avenue, Sand Diego, CA 92101, 1992.

[13] Zhi Cui. *Modeling and Simulation of Long Term Degradation and Lifetime of Deep-Submicron MOS Device and Circuit*. PhD thesis, Department of Electrical & and Computer Engineering at the University of Central Florida, Orlando, Florida, Spring Term 2005.

[14] Stanislav Tyaginov and Tibor Grasser. Modeling of Hot-Carrier Degradation: Physics and Controversial Issues. *2012 IIRW Final Report*, pages 206–215, 2012.

[15] ams AG Unterpremstaetten. *aH18* 0.18 *µm HV-CMOS Process Technology - ams Process Datasheet, Confidential*, September 2014.

[16] Kaushik Roy, Saibal Mukhopadhyay, and Hamid Mahmoodi-Meimand. Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits. *Proceedings of the IEEE*, 91:317–318, February 2003.

[17] Agilent Technologies. *Agilent 4155B/4156B User's Guide Vol.1, Edition 4*, 1997.

[18] Alvin L. S. Loke, Zhi-Yuan Wu, Reza Moallemi, C. Dru Cabler, Chad O. Lackey, Tin Tin Wee, and Bruce A. Doyle. Constant-Current Threshold Voltage Extraction in HSPICE for Nanoscale CMOS Analog Design. page 11, 2010.

[19] Hiroshima University and STARC. *HiSIM 2.9.0 User's Manual*. Hiroshima University, October 2015.

[20] Dragica Vasileska and Stephen M. Goodnick. *Computational Electronics*. Arizona State University, 2006.

[21] Vinod Kumar Khanna. Physics of carrier-transport mechanisms and ultra-small scale phenomena for theoretical modelling of nanometer MOS transistors from diffusive to ballistic regimes of operation. *Physics Reports 398*, pages 82–83, April 2004.

[22] A. B. Bhattacharyya. *Compact MOSFET Models for VLSI Design*. John Wiley & Sons (Asia) Pte Ltd, 2 Clementi Loop, 02-01. Singapore 129809, 2009.

[23] 6.012 Microelectronic Devices MIT OpenCourseWare and Circuits. *The Gradual Channel Approximation for the MOSFET*, Fall 2009.

[24] Cadence Design Systems. *Virtuoso Multi-Mode Simulation with Spectre Platform - Comprehensive analysis for robust design and verification*, 2013.

[25] H.Tanoue, A. Tanaka, Y. Oodate, T. Nakahagi, C. Ma, M. Miyake, H. J. Mattausch, M. Miura-Mattausch, K. Matsuzawa, S. Yamaguchi, T. Hoshida, M. Imade, R. Koh, and T. Arakawa. Universal Properties and Compact Modeling of Dynamic Hot-Electron Degradation in n-MOSFETs. In *Reliability Physics Symposium*. IEEE International, 2013.

[26] E. Takeda and N. Suzuki. An Empirical Model for Device Degradation Due to Hot-Carrier Injection. *IEEE Electron Device Letters*, EDL-4 No. 4:111–113, April 1983.

[27] Quality Department ams AG. *0.18um CMOS aC18 Reliability (Company Confidential)*, April 2016.

[28] Markus Bina. *Charge Transport Models for Reliability Engineering of Semiconductor Devices*. PhD thesis, Technischen Universität Wien, Fakultät für Elektrotechnik und Informationstechnik, March 2014.

9 Appendix

The Appendix includes several measurement, extraction and simulation scripts. Those may be used to reproduce results from this thesis or may be read to gain enhanced understanding on its content. Furthermore it contains several plots of measurement and simulation data. The figures were not considered essential inside the main chapters of the thesis, but still help to complete the conclusions given by the results.

9.1 Data Extraction Routine

Created with MATLAB® R2015a (8.5.0.197613).

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% MEASUREMENT DATA EXTRACTION SCRIPT %%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%% This file extracts data from .xls files created by the parameter analyzer control scripts.
%%% The output data format is .csv. The ouput files are sorted by time of stress.

clear all
close all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%% USER INPUT START %%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 01 Set the path to the INPUT file:
%     (with backslash at the end)
file_path = 'C:\Users\asch\Desktop\Master Thesis\Measurement Data\18_NFET_10_0c18_7_0c85_2c1\';

% 02 Set the INPUT file name:
%     (with .xls at the end)
file_name = '18_NFET_10_0c18_7_03_120ks_0c85_2c1.xls';

% 03 Define the folder name for the OUTPUT summary files:
folder_sum = 'data';

% 04 Define the folder name for the OUTPUT data files:
%     (folder included in folder_sum)
folder_all = 'all';

```

```
30 % 05 Define if an OUTPUT overall summary file already
     exists:
31 %     sum_exist = 0 (creation of a new overall summary
     file)
32 %     sum_exist = 1 (data is appended to existing overall
     summary file)
33 sum_exist = 1;
34
35 % 06 Define OUTPUT overall summary file name (for
     creation or expansion):
36 sum_name = '18_NFET_10_0c18_7_00_0c85_2c1_summary.csv';
37
38 % 07 Give the last time value in OUTPUT overall summary
     file:
39 %     (necessary if all_exist = 1)
40 sum_time = 200000;
41
42 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
43 %%%%%%%%%%%%%%%%% USER INPUT END %%%%%%%%%%%%%%%%%%
44 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
45
46 %definition of paths for output files and summary file
47 ex_path = [file_path,folder_sum,'\'];
48 path_all = [ex_path,folder_all,'\'];
49 %definition of output file name prefix
50 ex_name = [file_name(1:end-4),'_'];
51 %definiton of paths for file categories
52 path_trans_lin = [path_all,'trans_lin\'];
53 path_trans_sat = [path_all,'trans_sat\'];
54 path_trans_lin_lowVb = [path_all,'trans_lin_lowVb\'];
55 path_trans_sat_lowVb = [path_all,'trans_sat_lowVb\'];
56 path_out_Vg0c3 = [path_all,'out_Vg0c3\'];
57 path_out_Vg0c6 = [path_all,'out_Vg0c6\'];
58 path_out_Vg0c9 = [path_all,'out_Vg0c9\'];
59 path_out_Vg1c2 = [path_all,'out_Vg1c2\'];
60 path_out_Vg1c5 = [path_all,'out_Vg1c5\'];
61 path_out_Vg1c8 = [path_all,'out_Vg1c8\'];
62
63 %check existance of input file
64 if  exist([file_path,file_name])
65     disp('INFO: INPUT file was found!!!')
66     disp(['Path: ',file_path,file_name])
67 else
68     errorMessage = sprintf(['ERROR: The specified INPUT
         file does not ',...
```

```
69          'exist!!!\n%s'],['Please recheck the variables
              file path and '...
70          'file_name!!!']);
71      uiwait(warndlg(errorMessage));
72      return
73  end
74  %check existance of summary file
75  if sum_exist == 1 && exist([ex_path,sum_name])
76      disp('INFO: Data will be appended to existing summary
            file!!!')
77      disp(['Path: ',ex_path,sum_name])
78  elseif sum_exist == 1 && ~exist([ex_path,sum_name])
79      errorMessage = sprintf(['ERROR: The specified summary
            file was not '...
80          'found!!!\n%s'],'Please recheck the variable
              sum_nam!!!');
81      uiwait(warndlg(errorMessage));
82      return
83  end
84
85  %create directories for data
86  if ~isdir(ex_path)
87      mkdir(ex_path);
88  end
89  if ~isdir(path_all)
90      mkdir(path_all);
91  end
92  if ~isdir(path_trans_lin)
93      mkdir(path_trans_lin);
94  end
95  if ~isdir(path_trans_sat)
96      mkdir(path_trans_sat);
97  end
98  if ~isdir(path_trans_lin_lowVb)
99      mkdir(path_trans_lin_lowVb);
100 end
101 if ~isdir(path_trans_sat_lowVb)
102     mkdir(path_trans_sat_lowVb);
103 end
104 if ~isdir(path_out_Vg0c3)
105     mkdir(path_out_Vg0c3);
106 end
107 if ~isdir(path_out_Vg0c6)
108     mkdir(path_out_Vg0c6);
109 end
110 if ~isdir(path_out_Vg0c9)
```

```
111     mkdir(path_out_Vg0c9);
112 end
113 if ~isdir(path_out_Vg1c2)
114     mkdir(path_out_Vg1c2);
115 end
116 if ~isdir(path_out_Vg1c5)
117     mkdir(path_out_Vg1c5);
118 end
119 if ~isdir(path_out_Vg1c8)
120     mkdir(path_out_Vg1c8);
121 end
122 
123 %import data from sheet MPara (summary)
124 [num_sum,txt_sum,raw_sum] = xlsread([file_path file_name
        ],'MPara');
125 
126 %insert a time value for the first measurement
127 %(problems with logarithmic plotting for t = 0)
128 num_sum(1) = 0.1;
129 
130 %adjust time values if data is appended:
131 time = num_sum(:,1);
132 if sum_exist == 1
133     time = time + sum_time;
134 end
135 
136 %define loop for all the conditions
137 cur = [];
138 Vth_gm = [];
139 for cond = 1:10
140 
141     %load condition data from xls file
142     [num0,txt0,raw0] = xlsread([file_path file_name],['
            Cond-',num2str(cond)]);
143 
144     %transfer_lin
145     if cond == 1 || cond == 2
146         start = 2;
147         stop = 8;
148         step = 8;
149         head = ['Vg-V,Vd-V,gm-S,Ig-A,Id-A,Ib-A,Is-A'];
150         head_cond = {};
151         num_cond = [];
152         head_gm = {};
153         num_gm = [];
154 
```

```
155         %loop over all times and output of single files
156         for k1 = 1:numel(time)
157             num = num0(start+(k1-1)*step:stop+(k1-1)*
                   step,:)';
158 
159             %write current max values for export
160             cur(k1,1+(cond-1)*5) = num(end,4); %Ig
161             cur(k1,2+(cond-1)*5) = num(end,5); %Id
162             cur(k1,3+(cond-1)*5) = num(end,6); %Ig
163             cur(k1,4+(cond-1)*5) = num(end,7); %Ig
164             %current sum
165             cur(k1,5+(cond-1)*5) = sum(cur(k1,1+(cond-1)
                   *5:4+(cond-1)*5),2);
166 
167             %write summary file for condition
168             if k1 == 1
169                 num_cond(:,1) = num(:,1);
170                 head_cond(1,1) = {'Vg-V'};
171                 num_gm(:,1) = num(:,1);
172                 head_gm(1,1) = {'Vg-V'};
173             end
174             num_cond(:,k1+1) = num(:,5);
175             head_cond(1,k1+1) = {['Id-',num2str(time(k1)
                   ),'-A']};
176             num_gm(:,k1+1) = num(:,3);
177             head_gm(1,k1+1) = {['gm-',num2str(time(k1)),
                   '-S']};
178 
179             %export single data files
180             if cond == 1
181                 name = strcat(path_trans_lin,ex_name,'
                       trans_lin_',num2str(time(k1)),'.csv')
                       ;
182             elseif cond == 2
183                 name = strcat(path_trans_lin_lowVb,
                       ex_name,'trans_lin_lowVb_',num2str(
                       time(k1)),'.csv');
184             end
185             file = fopen(name,'wt');
186             fprintf(file,head);
187             fprintf(file,'\n');
188             fclose(file);
189             dlmwrite(name,num,'delimiter',',','precision
                   ',16,'-append');
190 
191             %calculate Vth with gmmmax
```

```
192             Vg = [];
193             gm = [];
194             Id = [];
195             Vg = num(:,1);
196             gm = num(:,3);
197             Id = num(:,5);
198             [gmmax,indmax] = max(gm);
199 
200             %calculation of tangent and zero point (Vth)
201             k_tan = gm(indmax);
202             d_tan = Id(indmax)-k_tan*Vg(indmax);
203             null_tan = -d_tan/k_tan;
204             Vth_gm(k1,cond) = null_tan;
205         end
206 
207         %export condition summary data
208         if cond == 1
209            name_cond = strcat(ex_path,ex_name,'
               trans_lin_Id_all.csv');
210            name_gm = strcat(ex_path,ex_name,'
               trans_lin_gm_all.csv');
211         elseif cond == 2
212            name_cond = strcat(ex_path,ex_name,'
               trans_lin_lowVb_Id_all.csv');
213            name_gm = strcat(ex_path,ex_name,'
               trans_lin_lowVb_gm_all.csv');
214         end
215         head_cond = strjoin(head_cond,',');
216         head_gm = strjoin(head_gm,',');
217         file_cond = fopen(name_cond,'wt');
218         fprintf(file_cond,head_cond);
219         fprintf(file_cond,'\n');
220         fclose(file_cond);
221         dlmwrite(name_cond,num_cond,'delimiter',',','
            precision',16,'-append');
222         file_gm = fopen(name_gm,'wt');
223         fprintf(file_gm,head_gm);
224         fprintf(file_gm,'\n');
225         fclose(file_gm);
226         dlmwrite(name_gm,num_gm,'delimiter',',','
            precision',16,'-append');
227     end
228 
229     %transfer_sat
230     if cond == 3 || cond == 4
231         start = 2;
```

```
232         stop = 8;
233         step = 8;
234         head = ['Vg-V,Vd-V,Vb-V,Ig-A,Id-A,Ib-A,Is-A'];
235         head_cond = {};
236         num_cond = [];
237 
238         %loop over all times and output of single files
239         for k1 = 1:numel(time)
240 
241             num = num0(start+(k1-1)*step:stop+(k1-1)*
                   step,:)';
242 
243             %write current max values for export
244             cur(k1,1+(cond-1)*5) = num(end,4); %Ig
245             cur(k1,2+(cond-1)*5) = num(end,5); %Id
246             cur(k1,3+(cond-1)*5) = num(end,6); %Ig
247             cur(k1,4+(cond-1)*5) = num(end,7); %Ig
248             %current sum
249             cur(k1,5+(cond-1)*5) = sum(cur(k1,1+(cond-1)
                   *5:4+(cond-1)*5),2);
250 
251             %write summary file for condition
252             if k1 == 1
253                 num_cond(:,1) = num(:,1);
254                 head_cond(1,1) = {'Vg-V'};
255             end
256             num_cond(:,k1+1) = num(:,5);
257             head_cond(1,k1+1) = {['Id-',num2str(time(k1)
                   ),'-A']};
258 
259             %export single data files
260             if cond == 3
261                 name = strcat(path_trans_sat,ex_name,'
                       trans_sat_',num2str(time(k1)),'.csv')
                       ;
262             elseif cond == 4
263                 name = strcat(path_trans_sat_lowVb,
                       ex_name,'trans_sat_lowVb_',num2str(
                       time(k1)),'.csv');
264             end
265             file = fopen(name,'wt');
266             fprintf(file,head);
267             fprintf(file,'\n');
268             fclose(file);
269             dlmwrite(name,num,'delimiter',',','precision
                   ',16,'-append');
```

```
270             end
271 
272             %export condition summary data
273             if cond == 3
274                 name_cond = strcat(ex_path,ex_name,'
                        trans_sat_all.csv');
275             elseif cond == 4
276                 name_cond = strcat(ex_path,ex_name,'
                        trans_sat_lowVb_all.csv');
277             end
278             head_cond = strjoin(head_cond,',');
279             file_cond = fopen(name_cond,'wt');
280             fprintf(file_cond,head_cond);
281             fprintf(file_cond,'\n');
282             fclose(file_cond);
283             dlmwrite(name_cond,num_cond,'delimiter',',','
                    precision',16,'-append');
284         end
285 
286 
287         %output
288         if cond > 4
289             start = 2;
290             stop = 8;
291             step = 8;
292             head = ['Vd-V,Vg-V,Vb-V,Ig-A,Id-A,Ib-A,Is-A'];
293             head_cond = {};
294             num_cond = [];
295 
296             %loop over all times and output of single files
297             for k1 = 1:numel(time)
298 
299                 num = num0(start+(k1-1)*step:stop+(k1-1)*
                        step,:)';
300 
301                 %write summary file for condition
302                 if k1 == 1
303                     num_cond(:,1) = num(:,1);
304                     head_cond(1,1) = {'Vd-V'};
305                 end
306                 num_cond(:,k1+1) = num(:,5);
307                 head_cond(1,k1+1) = {['Id-',num2str(time(k1)
                        ),'-A']};
308 
309                 %export single data files
310                 if cond == 5
```

```
311                 name = strcat(path_out_Vg0c3,ex_name,'
                       out_Vg0c3_',num2str(time(k1)),'.csv')
                       ;
312             elseif cond == 6
313                 name = strcat(path_out_Vg0c6,ex_name,'
                       out_Vg0c6_',num2str(time(k1)),'.csv')
                       ;
314             elseif cond == 7
315                 name = strcat(path_out_Vg0c9,ex_name,'
                       out_Vg0c9_',num2str(time(k1)),'.csv')
                       ;
316             elseif cond == 8
317                 name = strcat(path_out_Vg1c2,ex_name,'
                       out_Vg1c2_',num2str(time(k1)),'.csv')
                       ;
318             elseif cond == 9
319                 name = strcat(path_out_Vg1c5,ex_name,'
                       out_Vg1c5_',num2str(time(k1)),'.csv')
                       ;
320             elseif cond == 10
321                 name = strcat(path_out_Vg1c8,ex_name,'
                       out_Vg1c8_',num2str(time(k1)),'.csv')
                       ;
322             end
323             file = fopen(name,'wt');
324             fprintf(file,head);
325             fprintf(file,'\n');
326             fclose(file);
327             dlmwrite(name,num,'delimiter',',','precision
                   ',16,'-append');
328         end
329
330         %export condition summary data
331         if cond == 5
332            name_cond = strcat(ex_path,ex_name,'
                  out_Vg0c3_all.csv');
333         elseif cond == 6
334            name_cond = strcat(ex_path,ex_name,'
                  out_Vg0c6_all.csv');
335         elseif cond == 7
336            name_cond = strcat(ex_path,ex_name,'
                  out_Vg0c9_all.csv');
337         elseif cond == 8
338            name_cond = strcat(ex_path,ex_name,'
                  out_Vg1c2_all.csv');
339         elseif cond == 9
```

```
340                 name_cond = strcat(ex_path,ex_name,'
                        out_Vg1c5_all.csv');
341             elseif cond == 10
342                 name_cond = strcat(ex_path,ex_name,'
                        out_Vg1c8_all.csv');
343             end
344             head_cond = strjoin(head_cond,',');
345             file_cond = fopen(name_cond,'wt');
346             fprintf(file_cond,head_cond);
347             fprintf(file_cond,'\n');
348             fclose(file_cond);
349             dlmwrite(name_cond,num_cond,'delimiter',',','
                    precision',16,'-append');
350         end
351 end
352 
353 %define header for summary file:
354 head_sum = ['t-s,gmmax-S,gmmax-lowVb-S,Vthgm-V,Vthgm-
        lowVb-V,VthId-V,VthId-lowVb-V,' ...
355             'Igmax-lin-A,Idmax-lin-A,Ibmax-lin-A,Ismax-lin
                    -A,Isum-lin-A,' ...
356             'Igmax-linlowVb-A,Idmax-linlowVb-A,Ibmax-
                    linlowVb-A,Ismax-linlowVb-A,Isum-linlowVb-A
                    ,' ...
357             'Igmax-sat-A,Idmax-sat-A,Ibmax-sat-A,Ismax-sat
                    -A,Isum-sat-A,' ...
358             'Igmax-satlowVb-A,Idmax-satlowVb-A,Ibmax-
                    satlowVb-A,Ismax-satlowVb-A,Isum-satlowVb-A
                    '];
359 
360 %make new data matrix
361 num_sum = [time,num_sum(:,3:4),Vth_gm,num_sum(:,6),
        num_sum(:,8),cur];
362 
363 %write output summary file (write header if all_exist =
        0)
364 if sum_exist == 0
365     sum_path = strcat(ex_path,sum_name);
366     sum_file = fopen(sum_path,'wt');
367     fprintf(sum_file,head_sum);
368     fprintf(sum_file,'\n');
369     fclose(sum_file);
370     dlmwrite(sum_path,num_sum,'delimiter',',','precision'
            ,16,'-append');
371 elseif sum_exist == 1
372     sum_path = [ex_path sum_name];
```

```
373     dlmwrite(sum_path,num_sum,'delimiter',',','precision'
           ,16,'-append');
374 end
375 
376 % Eof
```

9.2 Data Difference Calculator

Created with MATLAB® R2015a (8.5.0.197613).

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %%%%%%%% DATA DIFFERENCE EXTRACTION SCRIPT %%%%%%%%%
3  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4  
5  %%% This file takes the extracted measurement summary.csv
      file created by the data extraction script and
     calculates the absolute and relative change of the
     device parameters over time.
6  %%% The output data format is .csv.
7  
8  clear all
9  close all
10 
11 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
12 %%%%%%%%%%%%%%%%% USER INPUT START %%%%%%%%%%%%%%%%%
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14 
15 % 01 Set the path to the INPUT file:
16 %    (with backslash at the end)
17 file_path = 'C:\Users\asch\Desktop\Master Thesis\
     Measurement Data\18_NFET_10_0c18_7_0c85_2c1\data\';
18 
19 % 02 Define INPUT overall summary file name:
20 %    (with .csv at the end)
21 in_name = '18_NFET_10_0c18_7_00_0c85_2c1_summary_clean.
     csv';
22 
23 % 03 Define OUTPUT absolute difference summary file name:
24 abs_name = '18
     _NFET_10_0c18_7_00_0c85_2c1_summary_clean_diff.csv';
25 
26 % 04 Define OUTPUT relative difference summary file name:
27 rel_name = '18
     _NFET_10_0c18_7_00_0c85_2c1_summary_clean_diff_rel.csv
     ';
```

```
28
29 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
30 %%%%%%%%%%%%%%%%%% USER INPUT END %%%%%%%%%%%%%%%%%%
31 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
32
33 %check existance of input file
34 if  exist([file_path,in_name])
35     disp('INFO: INPUT file was found!!!')
36     disp(['Path: ',file_path,in_name])
37 else
38     errorMessage = sprintf(['ERROR: The specified INPUT
           file does not ',...
39         'exist!!!\n%s'],['Please recheck the variables
             file path and '...
40         'in_name!!!']);
41     uiwait(warndlg(errorMessage));
42     return
43 end
44
45 %import data (with function csv_import.m)
46 data = csv_import([file_path,in_name],1,inf);
47 head = data(1,1:27);
48 head(1) = {'t-s'};
49 num = str2double(data(2:end,1:27));
50 time = num(:,1);
51 num = num(:,2:end);
52
53 %write new header for absolute difference output
54 head_abs = ['t-s,gmmax-diff-S,gmmax-lowVb-diff-S,Vthgm-
   diff-V,Vthgm-lowVb-diff-V,VthId-diff-V,VthId-lowVb-
   diff-V,' ...
55             'Igmax-lin-diff-A,Idmax-lin-diff-A,Ibmax-lin-
                diff-A,Ismax-lin-diff-A,Isum-lin-diff-A,'
                ...
56             'Igmax-linlowVb-diff-A,Idmax-linlowVb-diff-A,
                Ibmax-linlowVb-diff-A,Ismax-linlowVb-diff-A
                ,Isum-linlowVb-diff-A,' ...
57             'Igmax-sat-diff-A,Idmax-sat-diff-A,Ibmax-sat-
                diff-A,Ismax-sat-diff-A,Isum-sat-diff-A,'
                ...
58             'Igmax-satlowVb-diff-A,Idmax-satlowVb-diff-A,
                Ibmax-satlowVb-diff-A,Ismax-satlowVb-diff-A
                ,Isum-satlowVb-diff-A'];
59
60 %loop to calculate absolute differences and write them
   into diff
```

```
61  diff = [];
62  diff(1,1:size(num,2)) = 0;
63  for k2 = 2:size(num,1)
64      diff(k2,:) = abs(num(k2,:) - num(1,:));
65  end
66
67  %export absolute difference data to output file
68  diff_out = [time,diff];
69  out_path = strcat(file_path,abs_name);
70  out_file = fopen(out_path,'wt');
71  fprintf(out_file,head_abs);
72  fprintf(out_file,'\n');
73  fclose(out_file);
74  dlmwrite(out_path,diff_out,'delimiter',',','precision'
       ,16,'-append');
75
76  %write new header for relative difference output
77  head_rel = ['t-s,gmmax-diff,gmmax-lowVb-diff,Vthgm-diff,
       Vthgm-lowVb-diff,VthId-diff,VthId-lowVb-diff,' ...
78              'Igmax-lin-diff,Idmax-lin-diff,Ibmax-lin-diff,
                   Ismax-lin-diff,Isum-lin-diff,' ...
79              'Igmax-linlowVb-diff,Idmax-linlowVb-diff,Ibmax
                   -linlowVb-diff,Ismax-linlowVb-diff,Isum-
                   linlowVb-diff,' ...
80              'Igmax-sat-diff,Idmax-sat-diff,Ibmax-sat-diff,
                   Ismax-sat-diff,Isum-sat-diff,' ...
81              'Igmax-satlowVb-diff,Idmax-satlowVb-diff,Ibmax
                   -satlowVb-diff,Ismax-satlowVb-diff,Isum-
                   satlowVb-diff'];
82
83  %loop to calculate absolute differences and write them
       into diff
84  diff_rel = [];
85  diff_rel(1,1:size(num,2)) = 0;
86  for k2 = 2:size(num,1)
87      diff_rel(k2,:) = abs((num(k2,:) - num(1,:))./num(1,:)
           );
88  end
89
90  %export absolute difference data to output file
91  diff_rel_out = [time,diff_rel];
92  out_rel_path = strcat(file_path,rel_name);
93  out_rel_file = fopen(out_rel_path,'wt');
94  fprintf(out_rel_file,head_rel);
95  fprintf(out_rel_file,'\n');
96  fclose(out_rel_file);
```

```
97  dlmwrite(out_rel_path,diff_rel_out,'delimiter',',','
       precision',16,'-append');
98
99
100 %Eof
```

9.3 Spectre Command Scripts for Simulation

The files beyond are command scripts, which were used to run the HiSIM2 simulations inside the ams AG Spectre environment. In all the files, comments by the author are marked with **.

9.3.1 Initial nFET Simulation

```
1  ** HiSIM2_3.0.0-Beta_nfet_initial_transfer_sat
2
3  simulator lang = spice
4          ** Spectre is used as an emulator for Spice
              language
5
6  .hdl "../HiSIM2_3.0.0-Beta_VA-Code/hisim2.va"
7          ** The HiSIM2 Verilog A file is included
8
9  *Voltage sources:
10 Vb B 0 dc=0
11 Vd D 0 dc=1.8
12 Vs S 0 dc=0
13 Vg G 0 dc=0
14         ** The DC terminal voltages are set
15
16 *Subcircuit definition:
17 x1 D G S B nch W=10u L=0.18u NF=1
18         ** The subcircuit is defined with 4 terminals and
              the
19         ** geometries are set
20 .model nch hisim2_va
21         ** The used model is loaded
22 + CODEG  = 0
23         ** The flag for the degradation model is set to 0
24
25 *Device parameters
26 .include "parameters.scs"
27         ** The device parameter file is included
```

```
28
29 .options ingold=2
30         ** Simulation and output options are specified
31 .dc vg 0 1.8 0.02
32         ** The voltage sweep is specified (transfer char
               .)
33 .print dc par('-I(Vd)')
34         ** Results are exported to a .print file
35 .end
```

9.3.2 Extracted Device Parameters

```
 1 *Device parameters:
 2 + LEVEL        =                      62
 3 + COADOV       =                       1
 4 + CODFM        =                       0
 5 + COFLICK      =                       0
 6 + COGIDL       =                       1
 7 + COIGN        =                       0
 8 + COIIGS       =                       0
 9 + COIPRV       =                       1
10 + COISTI       =                       0
11 + COISUB       =                       1
12 + CONQS        =                       0
13 + COOVLP       =                       1
14 + COQOVSM      =                       1
15 + COQY         =                       0
16 + COPPRV       =                       1
17 + CORBNET      =                       0
18 + CORECIP      =                       1
19 + CORG         =                       0
20 + CORSRD       =                       1
21 + COTHRML      =                       0
22 + BB           =                       2
23 + BGTMP1       =               9.025e-05
24 + BGTMP2       =             -8.2289e-07
25 + BS1          =                       0
26 + BS2          =                     0.9
27 + CGBO         =                       0
28 + CGDO         =                   1e-12
29 + CGSO         =                   1e-12
30 + CISB         =                       0
31 + CISBK        =                       0
32 + CIT          =                       0
33 + CJ           =                  0.0005
```

```
34 + CJSW        =         5e-10
35 + CJSWG       =         5e-10
36 + CLM1        =           0.7
37 + CLM2        =             2
38 + CLM3        =           0.5
39 + CLM5        =           0.1
40 + CLM6        =        1.7944
41 + CTEMP       =             0
42 + CVB         =             0
43 + CVBK        =             0
44 + DDLTICT     =            10
45 + DDLTMAX     =            10
46 + DDLTSLP     =             0
47 + DIVX        =             0
48 + DLY1        =         1e-10
49 + DLY2        =           0.7
50 + DLY3        =         8e-07
51 + EG0         =         1.179
52 + EGIG        =             0
53 + FALPH       =             1
54 + GBMIN       =         1e-12
55 + GDL         =             0
56 + GDLD        =             0
57 + GDLP        =             1
58 + GIDL1       =         308.1
59 + GIDL2       =      42860000
60 + GIDL3       =        0.6343
61 + GIDL4       =             0
62 + GIDL5       =           0.2
63 + GLEAK1      =            50
64 + GLEAK2      =      10000000
65 + GLEAK3      =          0.06
66 + GLEAK4      =             4
67 + GLEAK5      =          7500
68 + GLEAK6      =          0.25
69 + GLEAK7      =         1e-06
70 + GLKB1       =         5e-16
71 + GLKB2       =             1
72 + GLKB3       =             0
73 + GLKSD1      =         1e-15
74 + GLKSD2      =       5000000
75 + GLKSD3      =      -5000000
76 + IBPC1       =             0
77 + IBPC2       =             0
78 + IGTEMP2     =             0
79 + IGTEMP3     =             0
```

```
 80 + JS0          =               5e-07
 81 + JS0SW        =                   0
 82 + KAPPA        =                 3.9
 83 + LL           =                   0
 84 + LLD          =                   0
 85 + LLN          =                   0
 86 + LOVER        =               1e-08
 87 + LP           =          1.2218e-07
 88 + LPEXT        =               1e-50
 89 + MJ           =                 0.5
 90 + MJSW         =                0.33
 91 + MJSWG        =                0.33
 92 + MPHDFM       =                -0.3
 93 + MUECB0       =              175.58
 94 + MUECB0LP     =                   0
 95 + MUECB1       =   92.16200000000001
 96 + MUECB1LP     =                   0
 97 + MUEPH0       =                 0.3
 98 + MUEPH1       =               27909
 99 + MUEPHL       =                   0
100 + MUEPHS       =                   0
101 + MUEPHW       =                   0
102 + MUEPLP       =                   1
103 + MUEPSP       =                   1
104 + MUEPWP       =             0.19387
105 + MUESLP       =                   1
106 + MUESR0       =                   2
107 + MUESR1       =     905830000000000
108 + MUESRL       =                   0
109 + MUESRW       =                   0
110 + MUESTI1      =                   0
111 + MUESTI2      =                   0
112 + MUESTI3      =                   1
113 + MUESWP       =                   1
114 + MUETMP       =              1.7166
115 + NDEP         =                   1
116 + NDEPL        =                   0
117 + NDEPLP       =                   1
118 + NDEPW        =             0.19227
119 + NDEPWP       =                   1
120 + NFALP        =               1e-19
121 + NFTRP        =         10000000000
122 + NINV         =                 0.5
123 + NINVD        =                   0
124 + NJ           =                   1
125 + NJSW         =                   1
```

```
126 + NOVER      =          2e+17
127 + NPEXT      =          2e+17
128 + NSTI       =          1e+17
129 + NSUBC      =     3.0945e+17
130 + NSUBCW     =              0
131 + NSUBCWP    =              1
132 + NSUBP      =     5.9594e+17
133 + NSUBPSTI1  =              0
134 + NSUBPSTI2  =              0
135 + NSUBPSTI3  =              1
136 + OVMAG      =            500
137 + OVSLP      =          2e-08
138 + PARL2      =          1e-08
139 + PB         =              1
140 + PBSW       =              1
141 + PBSWG      =              1
142 + PGD1       =              0
143 + PGD2       =              1
144 + PGD4       =              0
145 + PT2        =              0
146 + PT4        =          123.4
147 + PT4P       =              1
148 + PTL        =     0.00081638
149 + PTLP       =              1
150 + PTP        =            3.5
151 + PZADD0     =          0.005
152 + QME1       =              0
153 + QME2       =              0
154 + QME3       =              0
155 + RBDB       =             50
156 + RBPB       =             50
157 + RBPD       =             50
158 + RBPS       =             50
159 + RBSB       =             50
160 + RD         =     5.0761e-05
161 + RS         =     5.0761e-05
162 + RSH        =             72
163 + RSHG       =              0
164 + SAREF      =          1e-06
165 + SBREF      =          1e-06
166 + SC1        =              0
167 + SC2        =              0
168 + SC3        =              0
169 + SC3VBS     =              0
170 + SC4        =              0
171 + SCP1       =              0
```

```
172 + SCP2          =                    0
173 + SCP21         =                    0
174 + SCP22         =                    0
175 + SCP3          =                    0
176 + SCSTI1        =                    0
177 + SCSTI2        =                    0
178 + SLG           =           6.2247e-06
179 + SLGL          =                    0
180 + SLGLP         =                    1
181 + SUB1          =               106.26
182 + SUB1L         =           5.5211e-07
183 + SUB1LP        =                    1
184 + SUB2          =               25.116
185 + SUB2L         =           7.0445e-06
186 + SVBS          =               11.116
187 + SVBSL         =                    0
188 + SVBSLP        =                    1
189 + SVDS          =               1.2137
190 + SVGS          =               1.1442
191 + SVGSL         =          -4.5758e-06
192 + SVGSLP        =                    1
193 + SVGSW         =                    0
194 + SVGSWP        =                    1
195 + TNOM          =                   27
196 + TOX           =             4.45e-09
197 + TPOLY         =                2e-07
198 + VBI           =                  1.1
199 + VDIFFJ        = 0.000599999999999999
200 + VDSTI         =                    0
201 + VFBC          =               -1.044
202 + VFBOVER       =                    0
203 + VMAX          =             10000000
204 + VOVER         =              0.23791
205 + VOVERP        =                  0.3
206 + VOVERS        =                    0
207 + VOVERSP       =                  0.3
208 + VTHSTI        =                    0
209 + VTMP          =            -0.084214
210 + VZADD0        =                 0.01
211 + WFC           =                    0
212 + WL            =                    0
213 + WL1           =                    0
214 + WL1P          =                    1
215 + WL2           =                    0
216 + WL2P          =                    1
217 + WLD           =                    0
```

```
218 + WLN          =                    0
219 + WSTI         =                    0
220 + WSTIL        =                    0
221 + WSTILP       =                    1
222 + WSTIW        =                    0
223 + WSTIWP       =                    1
224 + WVTH0        =            0.0069006
225 + XL           =          -1.8737e-08
226 + XLD          =                    0
227 + XQY          =                1e-08
228 + XQY1         =                    0
229 + XQY2         =                    2
230 + XTI          =                    2
231 + XTI2         =                    0
232 + XW           =          -8.3527e-08
233 + XWD          =                    0
```

9.3.3 nFET Stress Simulation

```
1  ** HiSIM2_3.0.0-Beta_nfet_stress
2
3  simulator lang = spice
4          ** Spectre is used as an emulator for Spice
              language
5
6  *Stress time:
7  .param DEGTIME=1e4
8          ** The stress time for the simulation is
              specified
9
10 .hdl "../HiSIM2_3.0.0-Beta_VA-Code/hisim2.va"
11         ** The HiSIM2 Verilog A file is included
12
13 *Voltage sources:
14 Vb B 0 dc=0
15 Vd D 0 dc=3
16 Vs S 0 dc=0
17 Vg G 0 dc=0.85
18         ** The DC stress conditions are applied to the
              terminals
19
20 *Subcircuit definition:
21 x1 D G S B nch W=10u L=0.18u NF=1
22         ** The subcircuit is defined with 4 terminals and
               the
```

```
23          ** geometries are set
24  .model nch hisim2_va
25          ** The used model is loaded
26  + CODEG = 1
27          ** The flag for the degradation model is set to 1
28
29  *Aging parameters (default values):
30  + TRAGC0   = 0.8
31  + TRAPGC1  = 3.0E17
32  + TRAPGC2  = 5.0E17
33  + TRAPE0   = 5e5
34  + TRAPE1   = 0.13
35  + TRAPE2   = 0.03
36  + TAUCAP   = 1e-18
37  + DEGTIME  = DEGTIME
38  + VDSDEG   = 3
39  + VGSDEG   = 0.85
40          ** The values for the aging parameters are
              specified
41
42  *Device parameters
43  .include "parameters.scs"
44          ** The device parameter file is included
45
46  .options ingold=2
47          ** Simulation and output options are specified
48  .op
49  .print dc par('V(G)') par('V(D)') par('Isub')
50          ** Results are exported to a .print file
51  .end
52
53  ** The information about the aging of the device is
      stored in
54  ** the degradation.out file for further use.
```

9.3.4 Exported Degradation Parameters for Aging

```
1  ** Aged parameters:
2  .param x1_GC1DEG=4.03e+25
3  .param x1_GC2DEG=5e+23
4  .param x1_E1DEG=0.141089
5  .param x1_E2DEG=0.03
6  .param Vgse=0.844657
7  .param Vdse=2.98908
8  .param Vbse=-0.00534287
```

```
9  .param DEGTIME=10000 time:0
10         ** Exported aged parameters for further usage in
              the
11         ** degradation simulation.
```

9.3.5 nFET Aging Simulation

```
1  ** HiSIM2_3.0.0-Beta_nfet_aging_transfer_sat
2
3  simulator lang = spice
4          ** Spectre is used as an emulator for Spice
              language
5
6  .hdl "../HiSIM2_3.0.0-Beta_VA-Code/hisim2.va"
7          ** The HiSIM2 Verilog A file is included
8
9  *Voltage sources:
10 Vb B 0 dc=0
11 Vd D 0 dc=1.8
12 Vs S 0 dc=0
13 Vg G 0 dc=0
14         ** The DC terminal voltages are set
15
16 *Subcircuit definition:
17 .include "degradation.out"
18         ** The file with the degradation information is
              loaded
19         ** It includes the aged parameters
20 x1 D G S B nch W=10u L=0.18u NF=1
21         ** The subcircuit is defined with 4 terminals and
               the
22         ** geometries are set
23 .model nch hisim2_va
24         ** The used model is loaded
25 + CODEG  = 1
26         ** The flag for the degradation model is set to 1
27 + TRAPLX = 2
28         ** The aging parameter TRAPLX is set
29         ** (not inluded in the previous stressing script)
30
31 *Aged parameters (from degradation.out):
32 + GC1DEG    = x1_GC1DEG
33 + GC2DEG    = x1_GC2DEG
34 + E1DEG     = x1_E1DEG
35 + E2DEG     = x1_E2DEG
```

```
36          ** The aged parameters from the degradation file
               are
37          ** specifid for usage
38
39 *Device parameters
40 .include "parameters.scs"
41                  ** The device parameter file is included
42
43 .options ingold=2
44          ** Simulation and output options are specified
45 .dc vg 0 1.8 0.02
46          ** The voltage sweep is specified (transfer char
               .)
47 .print dc par('-I(Vd)')
48          ** Aged results are exported to a .print file
49 .end
```

9.4 Parameter Extraction Script for the Simulation

The script beyond was created with MATLAB® R2015a (8.5.0.197613). It is a routine for extracting the parameter $I_{d,sat}$ from HiSIM2 aging simulation result files.

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %%%%%%% EXTRACTION OF Id,sat FROM SIMULATION %%%%%%%
3  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4
5  %%% This file runs the HiSIM2 hot carrier degradation
      simulation and extracts the values for parameter Id,
      sat from the .print results file.
6  %%% The output data format is .csv.
7
8  clear all
9  close all
10
11 %define the name of the output file
12 name = '3c0_Idsat_extracted.csv';
13
14 %define the simulated time points
15 t = [[0:10:990]';[1000:1000:100000]'];
16
17 %call spectre for the ininital simulation of the transfer
       char.
18 !spectre 00_nfet_initial_trans_sat.scs +lqt 0 > spectre.
      log
19
```

```
20 %read initial data for Id from .print file
21 file_y = fopen('00_nfet_initial_trans_sat.print','r');
22 data = textscan(file_y,'%s','Delimiter','\n');
23 fclose(file_y);
24 data = data{1};
25 data = cell2mat(data(6:end-1));
26 x0 = data(1:end,1:11);
27 y0 = data(1:end,end-10:end);
28 x0 = str2num(x0);
29 y0 = str2num(y0);
30 %store Id,sat
31 result(1) = y0(end);
32
33 %loop for simulation of all time points
34 for k1 = 2:length(t);
35
36     %define current time point
37     time = t(k1);
38
39     %read content of .scs stress file into cell A
40     par_file = fopen('00_3c0_stress_template.scs','r');
41     i = 1;
42     tline = fgetl(par_file);
43     A{i} = tline;
44     while ischar(tline)
45         i = i+1;
46         tline = fgetl(par_file);
47         A{i} = tline;
48     end
49     fclose(par_file);
50
51     %change degradation time in A to current time point
52     txt = ['.param DEGTIME0=0 DEGTIME=',num2str(time)];
53     A{6} = sprintf('%s',txt);
54
55     % Write cell A into .scs stress file
56     delete('01_3c0_stress.scs')
57     out_file = fopen('01_3c0_stress.scs', 'w');
58     for i = 1:numel(A)
59         if A{i+1} == -1
60             fprintf(out_file,'%s', A{i});
61             break
62         else
63         fprintf(out_file,'%s\n', A{i});
64         end
65     end
```

```
66
67      %execute current scs. stress file with spectre
68      !spectre 01_3c0_stress.scs +lqt 0 > spectre.log
69      %execute .scs aging file with spectre
70      !spectre 02_nfet_age_trans_sat.scs +lqt 0 > spectre.
          log
71
72      %read aged data for Id from .print file
73      file_y = fopen('02_nfet_age_trans_sat.print','r');
74      data = textscan(file_y,'%s','Delimiter','\n');
75      fclose(file_y);
76      data = data{1};
77      data = cell2mat(data(6:end-1));
78      x0 = data(1:end,1:11);
79      y0 = data(1:end,end-10:end);
80      x0 = str2num(x0);
81      y0 = str2num(y0);
82      %store Id,sat
83      result(k1) = y0(end);
84
85  end
86
87  Idsat = result';
88  Idsat0 = Idsat(1);
89  dIdsat(1) = 0;
90  dIdsat_rel(1) = 0;
91
92  %loop for calculation of absolute and relative parameter
       change
93  for k2 = 2:length(Idsat);
94
95      dIdsat(k2) = Idsat0 - Idsat(k2);
96      dIdsat_rel(k2) = (Idsat0 - Idsat(k2))/Idsat0;
97
98  end
99
100 dIdsat = dIdsat';
101 dIdsat_rel = dIdsat_rel';
102
103 %define header and data for output file
104 head = ['t-s,Idsat-A,dIdsat-A,dIdsatrel-A'];
105 num = [t,Idsat,dIdsat,dIdsat_rel];
106
107 %delete existent file (to overwrite)
108 delete(name);
109
```

```
110 %write output .csv file
111 file = fopen(name,'wt');
112 fprintf(file,head);
113 fprintf(file,'\n');
114 fclose(file);
115 dlmwrite(name,num,'delimiter',',','precision',16,'-append
    ');
116
117
118 % Eof
```

9.5 Takeda Model Part 1: Linear Regime

Created with MATLAB® R2015a (8.5.0.197613).

9.5.1 Main Execution File

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %%% Parameter fitting script for takeda aging parameters.
3  %%% Only works in combination with takeda_age_sim.m
     function file.
4  %%%
5  %%% Valid for relative parameter changes of Vth_lin,
     Id_lin & gmmax
6  %%% @ t = 1e2, 1e3 & 1e4
7  %%%
8  %%% Alexander Schiffmann (asch), 27.6.2016
9  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10
11 close all;
12 clear all;
13
14 % setup for plots:
15 set(0,'DefaultLineMarkerSize',8);
16 set(0,'DefaultLineLineWidth',1.5);
17
18 % include data for x values (t):
19 x = [
20 % dVth_lin
21 1e2
22 1e3
23 1e4
24 % dId_lin
25 1e2
```

```
26 1e3
27 1e4
28 % dgmmax
29 1e2
30 1e3
31 1e4
32 ];
33
34 % Include measurement data for y values (relative
     parameter changes):
35 % order (always 3 values): dVth_lin,dId_lin,dgmmax
36 measured = [
37 % dVth_lin
38 0.02175842
39 0.081414022
40 0.246837784
41 % dId_lin
42 -0.04099549406318
43 -0.09919265400454
44 -0.2144965526869
45 % dgmmax
46 -0.05753702920017
47 -0.1379602200592
48 -0.2531358442658
49 ];
50
51 n = size(x,1);
52
53 % plotting measured values:
54 semilogx(x,measured,'b*');
55 grid on
56 xlabel('t / s')
57 ylabel('relative change of parameter')
58
59 %setting of options:
60 options=[
61 01      % OPTIONS(1)-debug level, 0 = silent mode
62 1      % OPTIONS(2)-fit, 0 no fit
63 1e-2   % OPTIONS(3)-termination tolerance
64 1e-12  % OPTIONS(4)-minimum absolute data value
65 1e-8   % OPTIONS(5)-minimum change of function for finite
     differences %1e-10
66 2e-5   % OPTIONS(6)-maximum change of function for finite
     differences  %2e-08
67 100    % OPTIONS(7)-maximum iteration count
68 1         % OPTIONS(8)-marquardt parameter
```

```
69  2          % OPTIONS(9)-marquardt multiplicator
70  1e-12  % OPTIONS(10)-relative parameter accuracy / root
       mean square error
71  0      % OPTIONS(11)-iteration count
72  0      % OPTIONS(12)-errmax;
73  0      % OPTIONS(13)-ierrmax;
74   ];
75
76  % include Takeda model constants for optimization
77  % fit (0/1)   start    minimum     maximum     eps     name
78  par=[
79  [ 1      10.000e+00       5.0000e+00     15.000e+00
       1.0000e-01 ] % A
80  [ 1      60.000e+00       45.000e+00     75.000e+00
       1.0000e-01 ] % B
81  [ 1      5.0000e+00       2.0000e+00     7.0000e+00
       1.0000e-01 ] % C
82  [ 1      0.5000e+00       0.1000e+00     1.0000e+00
       1.0000e-03 ] % n
83  [ 1      0.1000e+00       0.0100e+00     1.0000e+00
       1.0000e-03 ] % Ea
84  [ 1      20.000e+00       10.000e+00     30.000e+00
       1.0000e-01 ] % delta0
85  ];
86
87  pnames=[
88  'A         '
89  'B         '
90  'C         '
91  'n         '
92  'Ea        '
93  'delta0    '
94  ];
95
96  % define weighting preference:
97  w = [];
98
99  % give values to function pargnm.m for parameter fitting:
100 [par,f,err,OPTIONS,uncertainty,importance,C]= ...
101 pargnm('Takeda_lin_2_function',par,x,measured,pnames,
       options,w,pnames);
102
103 % plotting of results:
104 figure
105 semilogx(x(1:3),measured(1:3),'r*',x(1:3),f(1:3),'ro')
106 hold on
```

```
107 semilogx(x(4:6),measured(4:6),'b*',x(4:6),f(4:6),'bo')
108 semilogx(x(7:9),measured(7:9),'g*',x(7:9),f(7:9),'go')
109 hold off
110 grid on
111 xlabel('t / s')
112 ylabel('relative change of parameter')
113 legend('dVth-lin meas','dVth-lin sim','dIdlin meas','
       dIdlin sim','dgmmax meas','dgmmax sim','Location','
       NorthWest')
114 print('takeda_lin','-dpng')
115
116 toc
117
118 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
119 % Eof
```

9.5.2 Optimization Function

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %%% Parameter fitting script for takeda aging parameters.
3  %%% Only works in combination with takeda_lin_params.m
      script.
4  %%%
5  %%% Valid for relative parameter changes of Vth_lin,
      Id_lin & gmmax
6  %%% @ t = 1e2, 1e3 & 1e4
7  %%%
8  %%% Alexander Schiffmann (asch), 27.6.2016
9  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10
11 function y = nfetsim(par,x,pnames)
12
13 % define parameter name list:
14 par_names = cellstr(num2str(pnames));
15
16 % define used t values (update if x changes):
17 t_all = x(1:3);
18
19 % initial parameter values:
20 nsubp_0 = 5.9594e+17;
21 mueph1_0 = 27909;
22 rd_0 = 5.0761e-05;
23
24 % Id value for Vth extraction
25 Id_vth_sat = 55.56e-6;
```

```
26 Id_vth_lin = 1e-5;
27
28 % Takeda constants & variables
29 L = 0.18; %mum
30 kb = 8.6173303e-5; %eV/K
31 T = 25; %gC
32 T = T + 273.15; %K
33
34 % Takeda model constants:
35 par_data = par(:);
36 A = par_data(1);
37 B = par_data(2);
38 C = par_data(3);
39 n = par_data(4);
40 Ea = par_data(5);
41 delta0 = par_data(6);
42
43 % weights for parameter changes:
44 w_nsubp = 1;
45 w_mueph1 = 0.4;
46 w_rd = 1;
47
48 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
49 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
50
51 % simulate initial linear characteristic:
52 !spectre 02_nfet_tran_lin_initial.scs +lqt 0 > spectre.
      log
53
54 % open print files and import initial data:
55 file_init_lin = fopen('02_nfet_tran_lin_initial.print','r
      ');
56 data_lin = textscan(file_init_lin,'%s','Delimiter','\n');
57 fclose(file_init_lin);
58 data_lin = data_lin{1};
59 data_lin = cell2mat(data_lin(6:end-1));
60
61 % store initial characteristic for plotting:
62 vg_init_lin = data_lin(1:end,1:11);
63 vg_init_lin = str2num(vg_init_lin);
64 id_init_lin = data_lin(1:end,end-10:end);
65 id_init_lin = str2num(id_init_lin);
66
67 % Vth extraction initial (linear):
68 x_vth_lin_init = vg_init_lin(20:40);
69 y_vth_lin_init = id_init_lin(20:40);
```

```
70  p_lin_init = polyfit(x_vth_lin_init,y_vth_lin_init,8);
71  y_lin_fit_init = polyval(p_lin_init,x_vth_lin_init);
72  int = roots(p_lin_init - [0,0,0,0,0,0,0,0,Id_vth_lin]);
73  int_real = int(imag(int)==0);
74  int_res = int_real(int_real > 0.4);
75  int_res = int_res(int_res < 0.8);
76  if numel(int_res) == 1
77      vth_lin_init = {int_res};
78      vth_lin_init = cell2mat(vth_lin_init);
79  else
80      errorMessage = sprintf('ERROR!');
81      uiwait(warndlg(errorMessage));
82      disp('Calculation of Vth for initial linear curve not
            possible!')
83      return
84  end
85
86  % gmmax extraction initial (linear):
87  for k0 = 2:length(id_init_lin);
88      gm_init(k0-1) = (id_init_lin(k0)-id_init_lin(k0-1))/(
           vg_init_lin(k0)-vg_init_lin(k0-1));
89  end
90  gmmax_init = max(gm_init);
91
92  % Idlin extraction initial:
93  idlin_init = id_init_lin(end);
94
95  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
96
97  % delete old result files:
98  delete('00_takeda_result.csv')
99  delete('00_parameter_changes.csv')
100 delete('00_hisim_parameters.csv')
101 out_file = fopen('00_hisim_parameters.csv', 'w');
102 fclose(out_file);
103
104 % Loop for various times:
105 for kk = 1:length(t_all);
106     t = t_all(kk);
107
108     % simulate stress:
109     !spectre 03_nfet_stress.scs +lqt 0 > spectre.log
110
111     % open print file and import stress data:
112     file_stress = fopen('03_nfet_stress.print','r');
```

```
113     data_stress = textscan(file_stress,'%s','Delimiter','
          \n');
114     fclose(file_stress);
115     data_stress = data_stress{1};
116     data_stress = cell2mat(data_stress(6:end-1));
117 
118     % store stress data for evaluation:
119     t_stress = data_stress(1:end,1:11);
120     t_stress = str2num(t_stress);
121     vd_stress = data_stress(1:end,end-10:end);
122     vd_stress = str2num(vd_stress);
123 
124     % calculate sum of stress for Takeda model:
125     tsum = 0;
126     dt = t_stress(end)/(length(t_stress)-1);
127 
128     for k1 = 2:length(t_stress);
129         vd_m = (vd_stress(k1-1)+vd_stress(k1))/2;
130 
131         t0 = A*exp(B/vd_m)*L^C*exp(-Ea/(kb*T));
132         t1 = (t*dt)/(t0*t_stress(end));
133         tsum = tsum + t1;
134     end
135 
136     % calculate delta (factor for relative change of
          parameter):
137     delta = delta0*tsum^n;
138     delta = delta/100;
139 
140     % weight deltas for various parameters:
141     delta_nsubp = delta*w_nsubp;
142     delta_mueph1 = delta*w_mueph1;
143     delta_rd = delta*w_rd;
144 
145     % update parameters with deltas:
146     nsubp_new = nsubp_0*(1 + delta_nsubp);
147     mueph1_new = mueph1_0*(1 - delta_mueph1);
148     rd_new = rd_0*(1 + delta_rd);
149 
150     % write parameter value in vector for output:
151     hisim = [nsubp_new;mueph1_new;rd_new;vmax_new;ptl_new
          ];
152 
153     % write alter statement for parameters:
154     txt_nsubp = ['Set_nsubp_x1 alter sub=x1 param=NSUBP
          value=',num2str(nsubp_new)];
```

```
155     txt_mueph1 = ['Set_mueph1_x1 alter sub=x1 param=
           MUEPH1 value=',num2str(mueph1_new)];
156     txt_rd = ['Set_rd_x1 alter sub=x1 param=RD value=',
           num2str(rd_new)];
157
158     % print alter statements into aging.scs file:
159     delete('aging.scs')
160     out_file = fopen('aging.scs', 'w');
161     fprintf(out_file,'%s', txt_nsubp);
162     fprintf(out_file,'\n');
163     fprintf(out_file,'%s', txt_mueph1);
164     fprintf(out_file,'\n');
165     fprintf(out_file,'%s', txt_rd);
166     fclose(out_file);
167
168 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
169
170     % simulate aged linear characteristic:
171     !spectre 05_nfet_tran_lin_age.scs +lqt 0 > spectre.
           log
172
173     % open print files and import aged data:
174     file_age_lin = fopen('05_nfet_tran_lin_age.print','r'
           );
175     data_age_lin = textscan(file_age_lin,'%s','Delimiter'
           ,'\n');
176     fclose(file_age_lin);
177     data_age_lin = data_age_lin{1};
178     data_age_lin = cell2mat(data_age_lin(6:end-1));
179
180     % store aged characteristic for plotting:
181     vg_age_lin = data_age_lin(1:end,1:11);
182     vg_age_lin = str2num(vg_age_lin);
183     id_age_lin = data_age_lin(1:end,end-10:end);
184     id_age_lin = str2num(id_age_lin);
185
186     % Vth extraction aged (linear):
187     x_vth_lin_age = vg_age_lin(20:40);
188     y_vth_lin_age = id_age_lin(20:40);
189     p_lin_age = polyfit(x_vth_lin_age,y_vth_lin_age,8);
190     y_lin_fit_age = polyval(p_lin_age,x_vth_lin_age);
191     int = roots(p_lin_age - [0,0,0,0,0,0,0,0,Id_vth_lin])
           ;
192     int_real = int(imag(int)==0);
193     int_res = int_real(int_real > 0.4);
194     int_res = int_res(int_res < 0.9);
```

```
195     if numel(int_res) == 1
196         vth_lin_age = {int_res};
197         vth_lin_age = cell2mat(vth_lin_age);
198     else
199         errorMessage = sprintf('ERROR!');
200         uiwait(warndlg(errorMessage));
201         disp('Calculation of Vth for aged linear curve
                not possible!')
202         keyboard
203         return
204     end
205
206     % calculation of dVth (relative):
207     dvth_lin(kk) = (vth_lin_age - vth_lin_init)/
           vth_lin_init;
208
209     % calculation of dIdlin (relative):
210     idlin_age = id_age_lin(end);
211     didlin(kk) = (idlin_age - idlin_init)/idlin_init;
212
213     % gmmax extraction aged (linear):
214     for k2 = 2:length(id_init_lin);
215         gm_age(k2-1) = (id_age_lin(k2)-id_age_lin(k2-1))
                /(vg_age_lin(k2)-vg_age_lin(k2-1));
216     end
217     gmmax_age = max(gm_age);
218
219     % calculation of dgmmax (relative):
220     dgmmax(kk) = (gmmax_age - gmmax_init)/gmmax_init;
221
222     % write result file with aged parameters:
223     dlmwrite('00_hisim_parameters.csv',hisim,'delimiter',
           ',','precision',16,'-append');
224
225 end %end of time loop
226
227 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
228 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
229
230 % write results into y:
231 y = [dvth_lin(:);didlin(:);dgmmax(:)];
232
233 % write output file with HiSIM2 parameter changes:
234 out_file = fopen('00_parameter_changes.csv', 'w');
235 fclose(out_file);
```

```
236 dlmwrite('00_parameter_changes.csv',y,'delimiter',',','
       precision',16,'-append');
237
238 % write output file with new takeda parameter values:
239 out_file = fopen('00_takeda_result.csv', 'w');
240 fclose(out_file);
241 dlmwrite('00_takeda_result.csv',par_data,'delimiter',',',
       'precision',16,'-append');
242
243 % plot results (add to existing plot):
244 hold on
245 semilogx(x,y,'rx')
246 grid on
247
248 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
249 % Eof
```

9.6 Takeda Model Part 2: Saturation Regime

Created with MATLAB® R2015a (8.5.0.197613).

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  %%% Evaluation script for takeda aging model (saturation
      parameters).
3  %%% Only works after completion of Takeda_lin_params.m
      script.
4  %%%
5  %%% Valid for relative parameter changes of Vth_lin,
      Id_lin, gmmax,
6  %%% Vth_sat & Id_sat @ t = 1e2, 1e3 & 1e4
7  %%%
8  %%% Alexander Schiffmann (asch), 27.6.2016
9  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10
11 clear all
12 close all
13
14 % definition of time points:
15 t_all = [1e2;1e3;1e4];
16
17 % stacking of times to x vector (compare to number of
      used parameters):
18 x = [t_all;t_all;t_all;t_all;t_all];
19
20 % initial HiSIM parameter value input
```

```
21 nsubp_0 = 5.9594e+17;
22 nsubp_opt = [6.265412458216306e+17;6.93101489877659e
      +17;9.044357772723589e+17];
23 mueph1_0 = 27909;
24 mueph1_opt =
      [27335.75425738437;26088.89729100541;22130.02315810701];

25 rd_0 = 5.0761e-05;
26 rd_opt = [5.336755408120245e-05;5.903702508252483e
      -05;7.703806505708999e-05];
27
28 vmax_0 = 10000000;
29 ptl_0 = 0.00081638;
30
31 % Id value for Vth extraction
32 Id_vth_sat = 55.56e-6;
33 Id_vth_lin = 1e-5;
34
35 % various variables
36 L = 0.18; %mum
37 kb = 8.6173303e-5; %eV/K
38 T = 25; %gC
39 T = T + 273.15; %K
40
41 % Taked model constats:
42 A = 5;
43 B = 59.74911609762185;
44 C = 5.82200652177948;
45 n = 0.5017550580635527;
46 Ea = 0.1081926339024932;
47 delta0 = 20.16336937864933;
48
49 % weights for parameter changes:
50 w_vmax = 0.2;
51 w_ptl = 50;
52
53 % order (always 4 values): dVth_sat,dVth_lin,dId_sat,
      dId_lin,dgmmax
54 measured = [
55 0.0088290377926
56 0.03093400227493
57 0.1000747871889
58 0.02175842
59 0.081414022
60 0.246837784
61 -0.01337282797546
```

```
62 -0.04421256899894
63 -0.1154942187811
64 -0.04099549406318
65 -0.09919265400454
66 -0.2144965526869
67 -0.05753702920017
68 -0.1379602200592
69 -0.2531358442658
70 ];
71
72 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
73 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
74
75 % simulate initial characteristics
76 !spectre 01_nfet_tran_sat_initial.scs +lqt 0 > spectre.
     log
77 !spectre 02_nfet_tran_lin_initial.scs +lqt 0 > spectre.
     log
78
79 % open print files and import initial data
80 file_init_sat = fopen('01_nfet_tran_sat_initial.print','r
     ');
81 data_age_sat = textscan(file_init_sat,'%s','Delimiter','\
     n');
82 fclose(file_init_sat);
83 data_age_sat = data_age_sat{1};
84 data_age_sat = cell2mat(data_age_sat(6:end-1));
85
86 file_init_lin = fopen('02_nfet_tran_lin_initial.print','r
     ');
87 data_lin = textscan(file_init_lin,'%s','Delimiter','\n');
88 fclose(file_init_lin);
89 data_lin = data_lin{1};
90 data_lin = cell2mat(data_lin(6:end-1));
91
92 % store initial characteristics for plotting
93 vg_init_sat = data_age_sat(1:end,1:11);
94 vg_init_sat = str2num(vg_init_sat);
95 id_init_sat = data_age_sat(1:end,end-10:end);
96 id_init_sat = str2num(id_init_sat);
97
98 vg_init_lin = data_lin(1:end,1:11);
99 vg_init_lin = str2num(vg_init_lin);
100 id_init_lin = data_lin(1:end,end-10:end);
101 id_init_lin = str2num(id_init_lin);
102
```

```
103 % Vth extraction initial (saturation)
104 x_vth_sat = vg_init_sat(20:60);
105 y_vth_sat = id_init_sat(20:60);
106 p_sat = polyfit(x_vth_sat,y_vth_sat,10);
107 y_sat_fit = polyval(p_sat,x_vth_sat);
108 int = roots(p_sat - [0,0,0,0,0,0,0,0,0,0,Id_vth_sat]);
109 int_real = int(imag(int)==0);
110 int_res = int_real(int_real > 0.5);
111 int_res = int_res(int_res < 1);
112 if numel(int_res) == 1
113     vth_sat_init = {int_res};
114     vth_sat_init = cell2mat(vth_sat_init);
115 else
116     errorMessage = sprintf('ERROR!');
117     uiwait(warndlg(errorMessage));
118     disp('Calculation of Vth for initial saturation curve
            not possible!')
119     return
120 end
121 
122 % Vth extraction initial (linear)
123 x_vth_lin_init = vg_init_lin(20:40);
124 y_vth_lin_init = id_init_lin(20:40);
125 p_lin_init = polyfit(x_vth_lin_init,y_vth_lin_init,8);
126 y_lin_fit_init = polyval(p_lin_init,x_vth_lin_init);
127 int = roots(p_lin_init - [0,0,0,0,0,0,0,0,Id_vth_lin]);
128 int_real = int(imag(int)==0);
129 int_res = int_real(int_real > 0.4);
130 int_res = int_res(int_res < 0.8);
131 if numel(int_res) == 1
132     vth_lin_init = {int_res};
133     vth_lin_init = cell2mat(vth_lin_init);
134 else
135     errorMessage = sprintf('ERROR!');
136     uiwait(warndlg(errorMessage));
137     disp('Calculation of Vth for initial linear curve not
            possible!')
138     return
139 end
140 
141 % gmmax extraction initial (linear)
142 for k0 = 2:length(id_init_lin);
143     gm_init(k0-1) = (id_init_lin(k0)-id_init_lin(k0-1))/(
           vg_init_lin(k0)-vg_init_lin(k0-1));
144 end
145 gmmax_init = max(gm_init);
```

```
146
147 % Idsat and Idlin extraction initial
148 idsat_init = id_init_sat(end);
149 idlin_init = id_init_lin(end);
150
151 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
152
153 % delete old result files
154 %delete('00_takeda_result.csv')
155 delete('00_parameter_changes.csv')
156 delete('00_hisim_parameters.csv')
157 out_file = fopen('00_hisim_parameters.csv', 'w');
158 fclose(out_file);
159
160 % Loop for various times
161 for kk = 1:length(t_all);
162     t = t_all(kk);
163
164     % simulate stress
165     !spectre 03_nfet_stress.scs +lqt 0 > spectre.log
166
167     % open print file and import stress data
168     file_stress = fopen('03_nfet_stress.print','r');
169     data_stress = textscan(file_stress,'%s','Delimiter','
          \n');
170     fclose(file_stress);
171     data_stress = data_stress{1};
172     data_stress = cell2mat(data_stress(6:end-1));
173
174     % store stress data for evaluation
175     t_stress = data_stress(1:end,1:11);
176     t_stress = str2num(t_stress);
177     vd_stress = data_stress(1:end,end-10:end);
178     vd_stress = str2num(vd_stress);
179
180     % calculate sum of stress for Takeda model
181     tsum = 0;
182     dt = t_stress(end)/(length(t_stress)-1);
183
184     for k1 = 2:length(t_stress);
185         vd_m = (vd_stress(k1-1)+vd_stress(k1))/2;
186
187         t0 = A*exp(B/vd_m)*L^C*exp(-Ea/(kb*T));
188         t1 = (t*dt)/(t0*t_stress(end));
189         tsum = tsum + t1;
190     end
```

```
191
192     % calculate delta (factor for relative change of
           parameter)
193     delta = delta0*tsum^n;
194     delta = delta/100;
195
196     % weight deltas for various parameters
197     delta_vmax = delta*w_vmax;
198     delta_ptl = delta*w_ptl;
199
200     % update parameters with deltas
201     vmax_new = vmax_0*(1 + delta_vmax);
202     ptl_new = ptl_0*(1 + delta_ptl);
203
204     % write parameter value in vector for output
205     hisim = [nsubp_opt(kk);mueph1_opt(kk);rd_opt(kk);
           vmax_new;ptl_new];
206
207     % write alter statement for parameters
208     txt_nsubp = ['Set_nsubp_x1 alter sub=x1 param=NSUBP
           value=',num2str(nsubp_opt(kk))];
209     txt_mueph1 = ['Set_mueph1_x1 alter sub=x1 param=
           MUEPH1 value=',num2str(mueph1_opt(kk))];
210     txt_rd = ['Set_rd_x1 alter sub=x1 param=RD value=',
           num2str(rd_opt(kk))];
211     txt_vmax = ['Set_vmax_x1 alter sub=x1 param=VMAX
           value=',num2str(vmax_new)];
212     txt_ptl = ['Set_ptl_x1 alter sub=x1 param=PTL value='
           ,num2str(ptl_new)];
213
214     % print alter statements into aging.scs file
215     delete('aging.scs')
216     out_file = fopen('aging.scs', 'w');
217     fprintf(out_file,'%s', txt_nsubp);
218     fprintf(out_file,'\n');
219     fprintf(out_file,'%s', txt_mueph1);
220     fprintf(out_file,'\n');
221     fprintf(out_file,'%s', txt_rd);
222     fprintf(out_file,'\n');
223     fprintf(out_file,'%s', txt_vmax);
224     fprintf(out_file,'\n');
225     fprintf(out_file,'%s', txt_ptl);
226     fclose(out_file);
227
228 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
229
```

```
230     % simulate aged characteristics
231     !spectre 04_nfet_tran_sat_age.scs +lqt 0 > spectre.
          log
232     !spectre 05_nfet_tran_lin_age.scs +lqt 0 > spectre.
          log
233
234     % open print files and import aged data
235     file_age_sat = fopen('04_nfet_tran_sat_age.print','r'
          );
236     data_age_sat = textscan(file_age_sat,'%s','Delimiter'
          ,'\n');
237     fclose(file_age_sat);
238     data_age_sat = data_age_sat{1};
239     data_age_sat = cell2mat(data_age_sat(6:end-1));
240
241     file_age_lin = fopen('05_nfet_tran_lin_age.print','r'
          );
242     data_age_lin = textscan(file_age_lin,'%s','Delimiter'
          ,'\n');
243     fclose(file_age_lin);
244     data_age_lin = data_age_lin{1};
245     data_age_lin = cell2mat(data_age_lin(6:end-1));
246
247     % store aged characteristics for plotting
248     vg_age_sat = data_age_sat(1:end,1:11);
249     vg_age_sat = str2num(vg_age_sat);
250     id_age_sat = data_age_sat(1:end,end-10:end);
251     id_age_sat = str2num(id_age_sat);
252
253     vg_age_lin = data_age_lin(1:end,1:11);
254     vg_age_lin = str2num(vg_age_lin);
255     id_age_lin = data_age_lin(1:end,end-10:end);
256     id_age_lin = str2num(id_age_lin);
257
258     % Vth extraction aged (saturation)
259     x_vth_sat = vg_age_sat(20:60);
260     y_vth_sat = id_age_sat(20:60);
261     p_sat = polyfit(x_vth_sat,y_vth_sat,10);
262     y_sat_fit = polyval(p_sat,x_vth_sat);
263     int = roots(p_sat - [0,0,0,0,0,0,0,0,0,0,Id_vth_sat])
          ;
264     int_real = int(imag(int)==0);
265     int_res = int_real(int_real > 0.5);
266     int_res = int_res(int_res < 1);
267     if numel(int_res) == 1
268         vth_sat_age = {int_res};
```

```
269            vth_sat_age = cell2mat(vth_sat_age);
270        else
271            errorMessage = sprintf('ERROR!');
272            uiwait(warndlg(errorMessage));
273            disp('Calculation of Vth for aged saturation
                   curve not possible!')
274            return
275        end
276
277        % Vth extraction aged (linear)
278        x_vth_lin_age = vg_age_lin(20:40);
279        y_vth_lin_age = id_age_lin(20:40);
280        p_lin_age = polyfit(x_vth_lin_age,y_vth_lin_age,8);
281        y_lin_fit_age = polyval(p_lin_age,x_vth_lin_age);
282        int = roots(p_lin_age - [0,0,0,0,0,0,0,0,Id_vth_lin])
               ;
283        int_real = int(imag(int)==0);
284        int_res = int_real(int_real > 0.4);
285        int_res = int_res(int_res < 0.8);
286        if numel(int_res) == 1
287            vth_lin_age = {int_res};
288            vth_lin_age = cell2mat(vth_lin_age);
289        else
290            errorMessage = sprintf('ERROR!');
291            uiwait(warndlg(errorMessage));
292            disp('Calculation of Vth for aged linear curve
                   not possible!')
293            return
294        end
295
296        % calculation of dVth (relative)
297        dvth_sat(kk) = (vth_sat_age - vth_sat_init)/
               vth_sat_init;
298        dvth_lin(kk) = (vth_lin_age - vth_lin_init)/
               vth_lin_init;
299
300        % calculation of dIdsat (relative)
301        idsat_age = id_age_sat(end);
302        didsat(kk) = (idsat_age - idsat_init)/idsat_init;
303
304        % calculation of dIdlin (relative)
305        idlin_age = id_age_lin(end);
306        didlin(kk) = (idlin_age - idlin_init)/idlin_init;
307
308        % gmmax extraction aged (linear)
309        for k2 = 2:length(id_init_lin);
```

```
310            gm_age(k2-1) = (id_age_lin(k2)-id_age_lin(k2-1))
                   /(vg_age_lin(k2)-vg_age_lin(k2-1));
311        end
312        gmmax_age = max(gm_age);
313
314        % calculation of dgmmax (relative)
315        dgmmax(kk) = (gmmax_age - gmmax_init)/gmmax_init;
316
317        % write result file with aged parameters
318        dlmwrite('00_hisim_parameters.csv',hisim,'delimiter',
               ',','precision',16,'-append');
319
320    end %end of time loop
321
322    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
323    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
324
325    f = [dvth_sat(:);dvth_lin(:);didsat(:);didlin(:);dgmmax
           (:)];
326
327    out_file = fopen('00_parameter_changes.csv', 'w');
328    fclose(out_file);
329    dlmwrite('00_parameter_changes.csv',f,'delimiter',',','
           precision',16,'-append');
330
331    figure
332    semilogx(x(1:3),measured(1:3),'r*',x(1:3),f(1:3),'ro')
333    hold on
334    semilogx(x(4:6),measured(4:6),'b*',x(4:6),f(4:6),'bo')
335    semilogx(x(7:9),measured(7:9),'g*',x(7:9),f(7:9),'go')
336    semilogx(x(10:12),measured(10:12),'c*',x(10:12),f(10:12),
           'co')
337    semilogx(x(13:15),measured(13:15),'k*',x(13:15),f(13:15),
           'ko')
338    hold off
339    grid on
340    xlabel('t / s')
341    ylabel('relative change of parameter')
342    legend('dVth-sat meas','dVth-sat sim','dVth-lin meas','
           dVth-lin sim','dIdsat meas','dIdsat sim','dIdlin meas'
           ,'dIdlin sim','dgmmax meas','dgmmax sim','Location','
           NorthWest')
343    print('takeda_sat','-dpng')
344
345    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
346    %Eof
```

9.7 Completion of Measurement Results Section 5 (Plots)

The plots beyond are completing the measurement results for the stress conditions $V_g = 0.85$ V and $V_d = 3.0$ V.

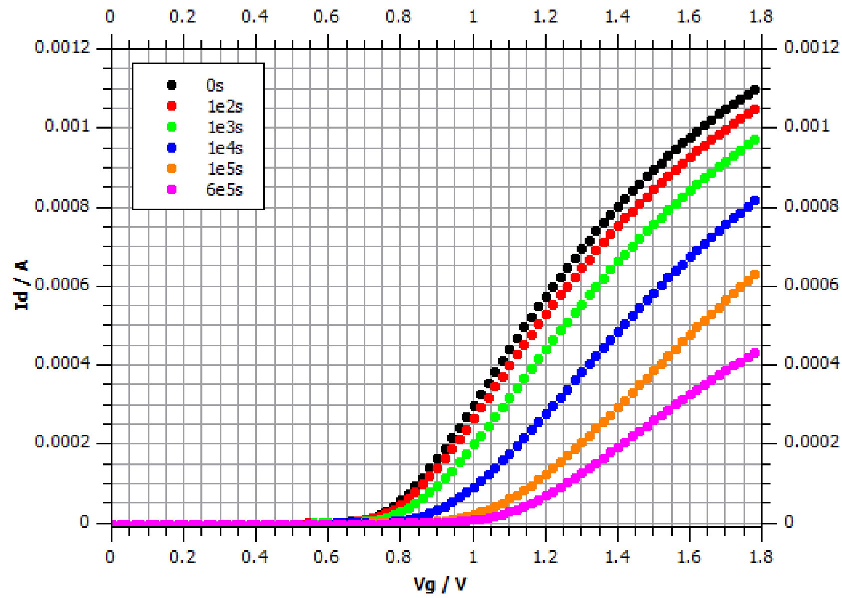

Figure 71: Transfer characteristics for various stress times @ $V_d = 0.1$ V, $V_b = -1$ V (M10) [4]

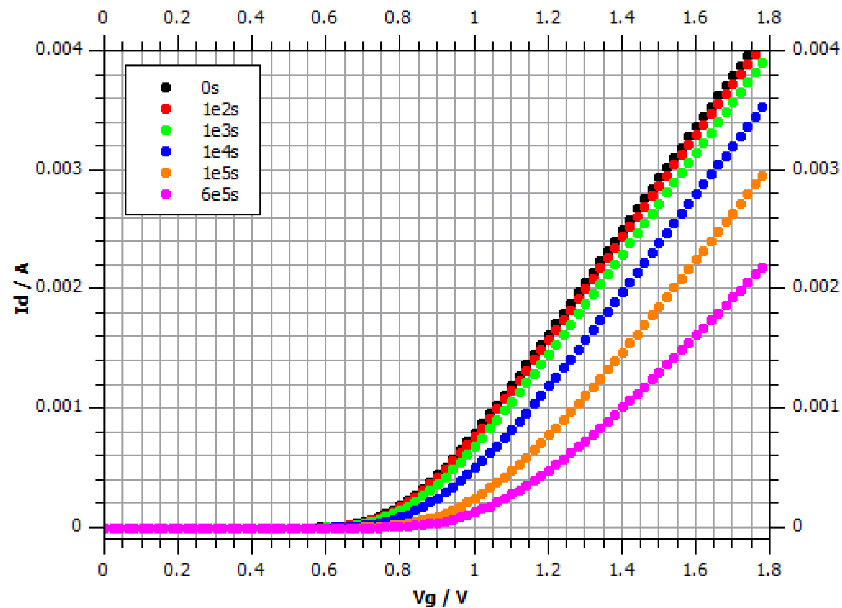

Figure 72: Transfer characteristics for various stress times @ $V_d = 1.8$ V, $V_b = -1$ V (M10) [4]

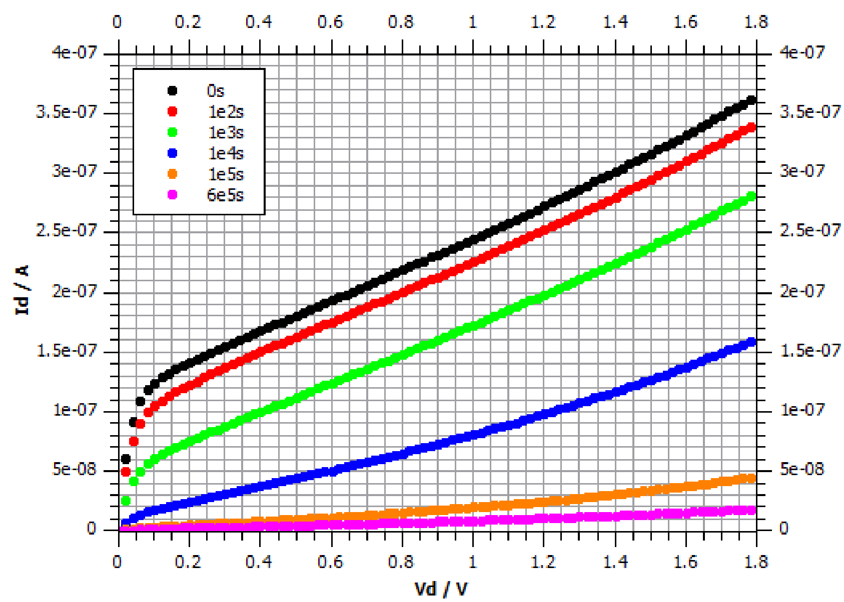

Figure 73: Output characteristics for various stress times @ $V_g = 0.3$ V, $V_b = 0$ V (M10) [4]

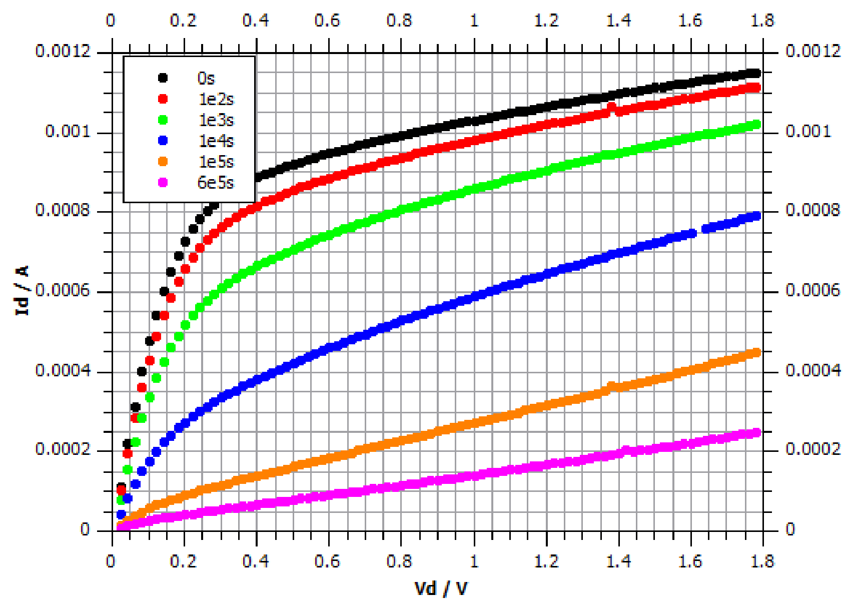

Figure 74: Output characteristics for various stress times @ $V_g = 0.9$ V, $V_b = 0$ V (M10) [4]

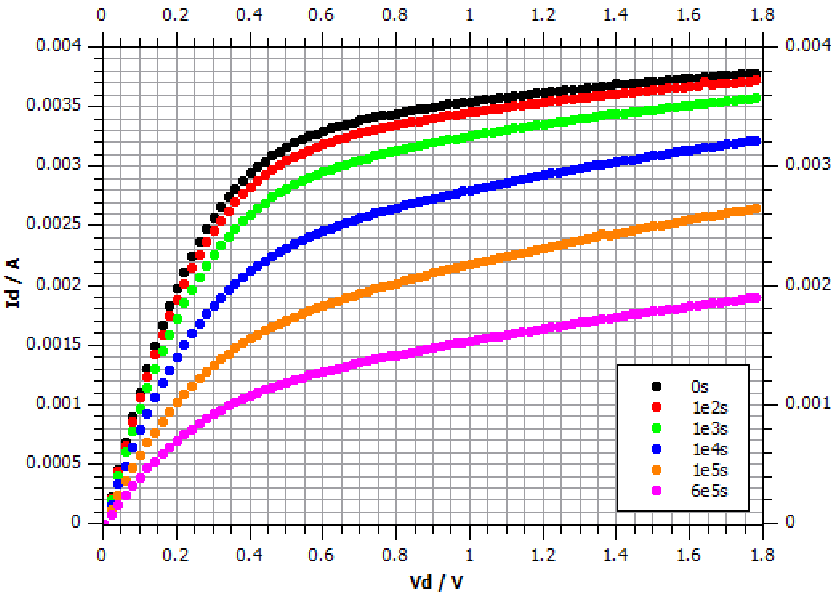

Figure 75: Output characteristics for various stress times @ $V_g = 1.5$ V, $V_b = 0$ V (M10) [4]

9.8 Completion of Simulation Results Section 6 (Plots)

The plots beyond are completing the aging simulation results for the stress conditions $V_g = 0.85$ V and $V_d = 3.0$ V. Both for the HiSIM2 aging approach and the Takeda model.

9.8.1 HiSIM2 Aging Model Simulation Results

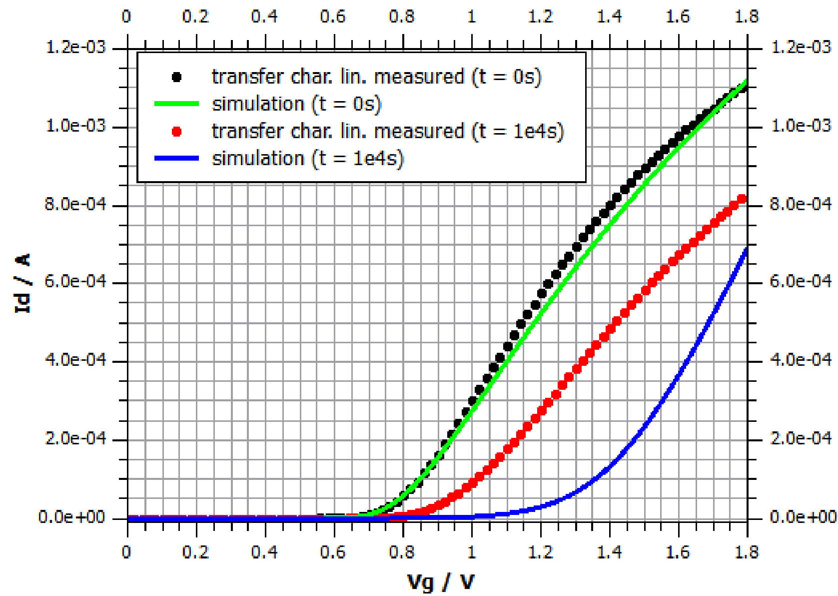

Figure 76: HiSIM2 v3.0.0 model transfer characteristics @ $V_d = 0.1$ V, $V_b = 0$ V compared to measurement results (M10) [4]

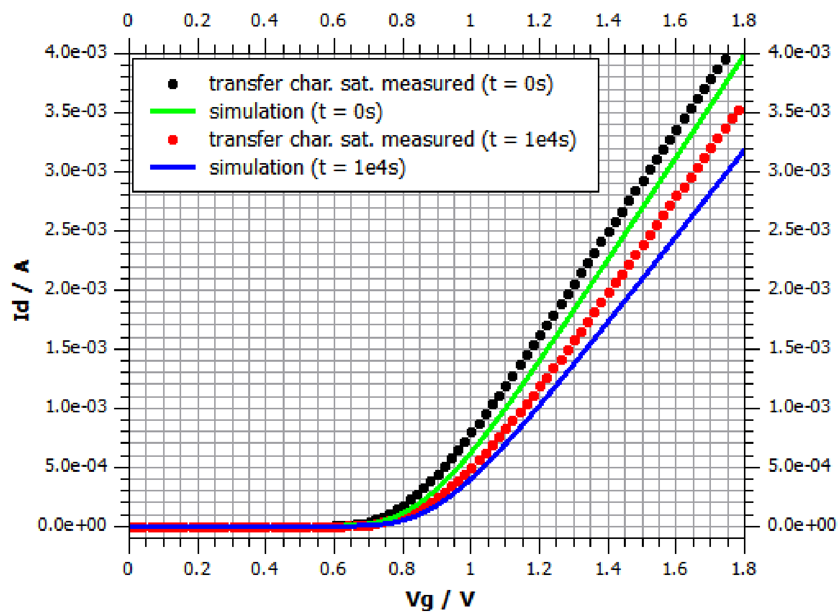

Figure 77: HiSIM2 v3.0.0 model transfer characteristics @ V_d = 1.8 V, V_b = 0 V compared to measurement results (M10) [4]

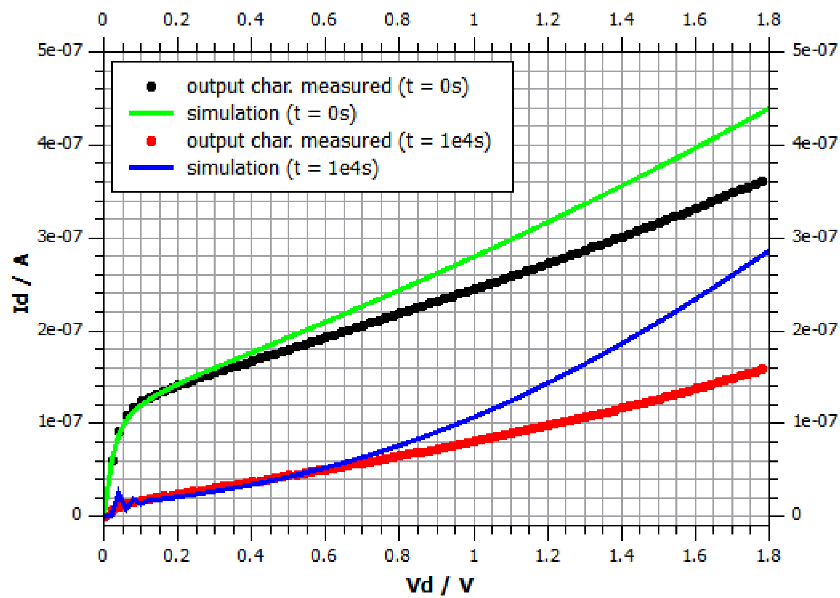

Figure 78: HiSIM2 v3.0.0 model output characteristics @ V_g = 0.3 V, V_b = 0 V compared to measurement results (M10) [4]

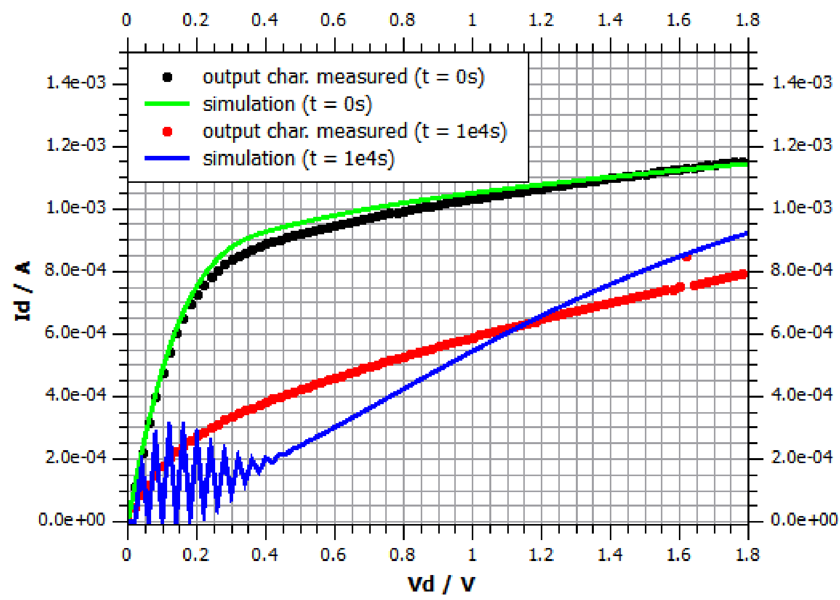

Figure 79: HiSIM2 v3.0.0 model output characteristics @ $V_g = 0.9$ V, $V_b = 0$ V compared to measurement results (M10) [4]

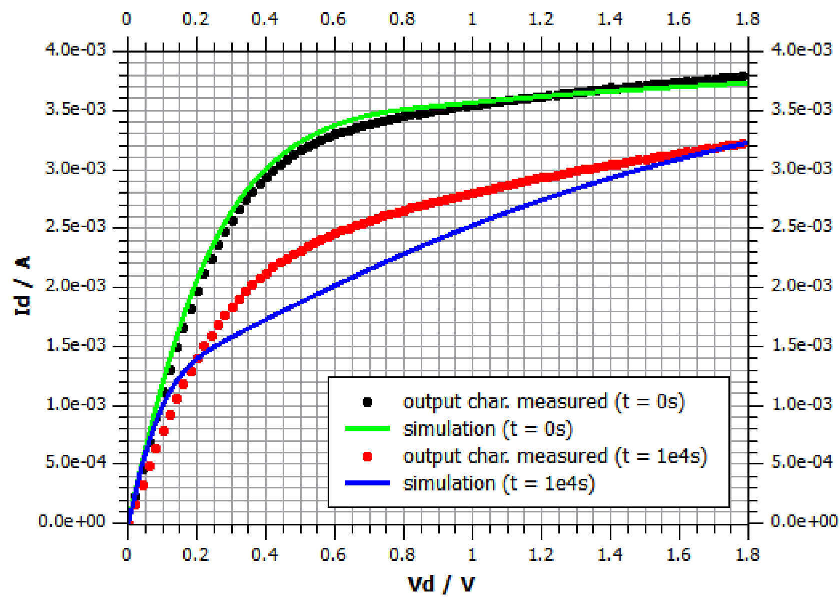

Figure 80: HiSIM2 v3.0.0 model output characteristics @ $V_g = 1.5$ V, $V_b = 0$ V compared to measurement results (M10) [4]

9.8.2 Takeda Model Simulation Results

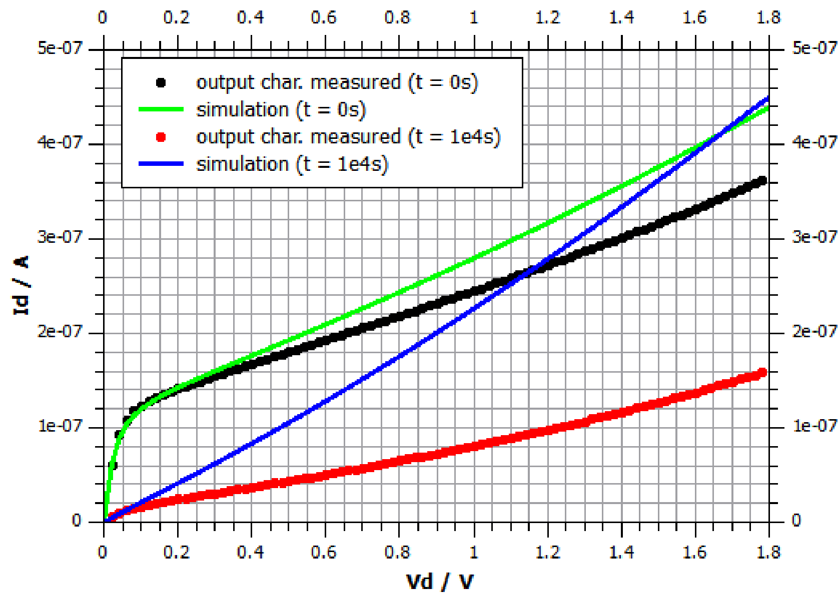

Figure 81: Takeda model output characteristics @ $V_g = 0.3$ V, $V_b = 0$ V compared to measurement results (M10) [4]

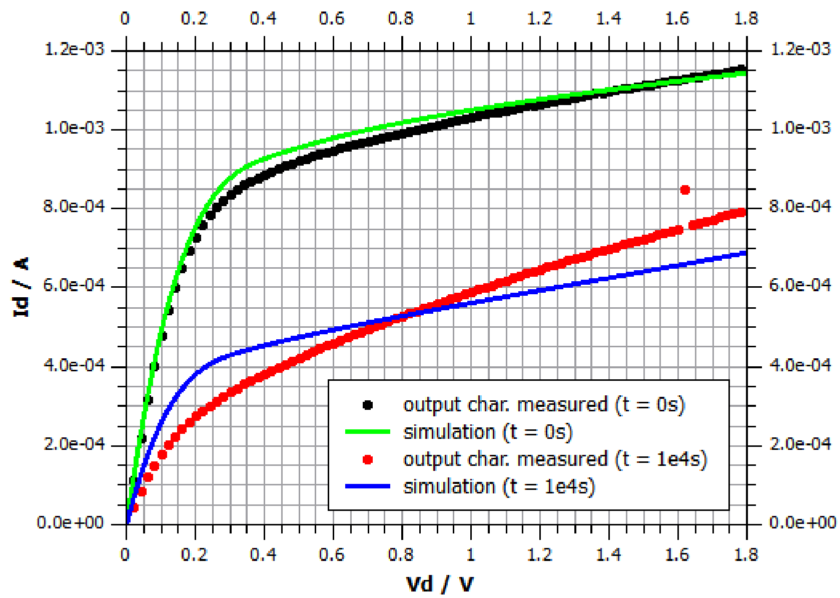

Figure 82: Takeda model output characteristics @ $V_g = 0.9$ V, $V_b = 0$ V compared to measurement results (M10) [4]

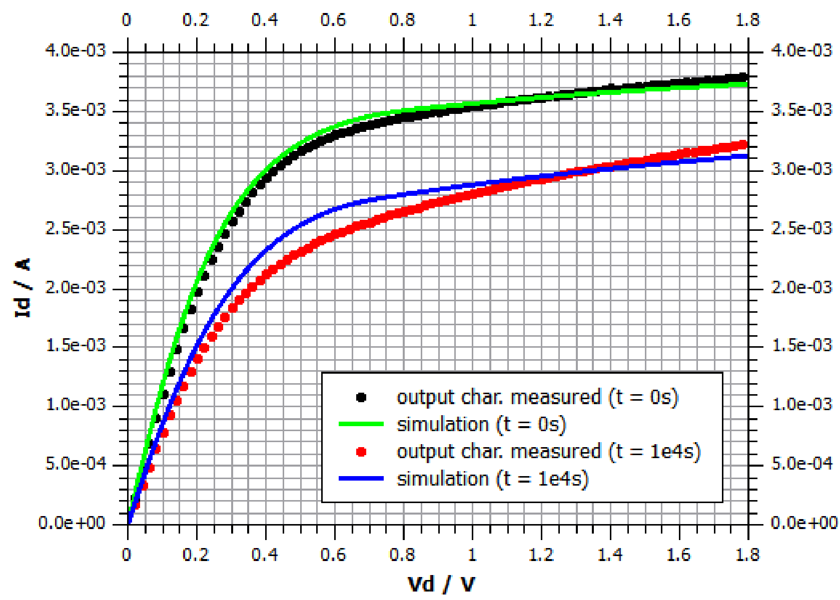

Figure 83: Takeda model output characteristics @ $V_g = 1.5$ V, $V_b = 0$ V compared to measurement results (M10) [4]