Maßnahmen zur Steigerung der Zuverlässigkeit integrierter Schaltungen auf Gatterebene hinsichtlich Gateoxiddefekten

Dissertation

zur

Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik

der Universität Rostock

vorgelegt von

Hagen Sämrow, geb. am 02.06.1979 in Schwerin

aus Hamburg

Hamburg, den 03.12.2013

urn:nbn:de:gbv:28-diss2015-0029-6

Gutachter:

Prof. Dr.-Ing. D. Timmermann, Universität Rostock

Prof. Dr.-Ing. M. Ortmanns, Universität Ulm

Prof. Dr.-Ing. habil. C. Haubelt, Universität Rostock

Tag der Einreichung: 03.12.2013

Tag der Verteidigung: 26.06.2014

Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit am Institut für Angewandte Mikroelektronik und Datentechnik der Universität Rostock. An erster Stelle möchte ich mich ausdrücklich bei Herrn Prof. Dr.-Ing. Dirk Timmermann dafür bedanken, dass er mir die Möglichkeit gab, diese Arbeit anzufertigen. Unsere konstruktiven und wegweisenden Konsultationen haben einen wesentlichen Anteil am Zustandekommen dieser Arbeit. Dass ich während meiner Tätigkeit als Chipdesigner im Anschluss meiner Universitätsanstellung die Dissertation beenden konnte, verdanke ich vor allem dem Entgegenkommen meiner Vorgesetzten und Kollegen der Firma Trinamic Motion Control. Prof. Dr. Maurits Ortmanns und Prof. Dr.-Ing. habil. Christian Haubelt danke ich für Ihr Interesse an meiner Arbeit und für die Übernahme der Zweitgutachten.

Ein besonderer Dank geht an Prof. Dr.-Ing. Frank Sill und Dr.-Ing. Claas Cornelius, welche sowohl mein Studium geleitet haben, als auch einen fundamentalen Anteil an meiner wissenschaftlichen Arbeit gehabt haben. In vielen Gesprächen haben sie mir den Weg zum wissenschaftlichen Arbeiten gezeigt und mich konstruktiv unterstützt. Ein herzlicher Dank gilt ebenfalls meinen ehemaligen Kollegen Andreas Tockhorn, Dr.-Ing. Jakob Salzmann und Philipp Gorski, die mir während unserer gemeinsamen Zeit als Wissenschaftliche Mitarbeiter viele Ideen und Anstöße gegeben haben. Ein Dank geht auch an Dr. rer.nat. Jörg Wilhelm für das Lektorat dieser Arbeit und an alle Mitarbeiter des Instituts für Angewandte Mikroelektronik und Datentechnik für das ausgesprochen angenehme Arbeitsklima.

Diese Arbeit wäre ohne familiäre Rückendeckung nicht zustande gekommen. So schulde ich meinen Eltern Marlis und Hagen Sämrow größten Dank für ihre immerwährende Unterstützung und Ermutigung. Mein ganz besonderer Dank gilt meiner lieben Freundin Britta Harder und unserer gemeinsamen Tochter Finnja, welche während der langwierigen Arbeiten viel Geduld und Entbehrungen aufbringen mussten und doch immer mit ganzem Herzen an meiner Seite standen.

Inhaltsverzeichnis

Verzeichnis der Bilder

Verzeichnis der Tabellen

Liste der Abkürzungen und Symbole

a	–	Wahrscheinlichkeit von 0→1-Übergängen am CMOS-Gatterausgang
a_{VEL}	–	Velocity saturation index
β	–	Weibullformparameter
γ	–	Body-effect-Koeffizient
γ_{ACC}	–	Feldabhängiger Beschleunigungsfaktor des E-Modells
ε_{OX}	–	Dielektrische Leitfähigkeit des Gateoxides
η	–	DIBL-Koeffizient
λ	–	Defektrate
λ_{NET}	–	Defektrate eines Knotens im Design
λ_{SYSTEM}	–	Systemausfall/-fehlerrate
μ	–	Mobilität
φ_S	–	Oberflächenpotential des Substrates
Φ_{ms}	–	Differenz der Austrittsarbeit des Gates und des Substrates
Φ_{OX}	–	Austrittsarbeit vom Substrat zum Oxid
A_{DIE}	–	Chipgröße
AF	–	Verschleißfaktor des Gateoxiddefektes
A_{GATE}	–	Gatefläche
A_{GATE_MAX}	–	Gatefläche des GOB-Simulator-Referenztransistors
AHI	–	Anode Hole Injection Modell
A_{IC}	–	Querschnittfläche eines Leiters
ALU	–	Arithmetische-Logische Einheit
AR	–	Aspect ratio
BERT	–	Berkeley Reliability Tools
B_{FN}	–	Konstante des Fowler-Nordheim tunneling
BISR	–	Built-in-Self-Repair
CAD	–	Computer-aided design
CAS	–	Circuit Aging Simulator
CCS	–	Composite Current-Source-Modell
CD	–	Common Drain
CETD	–	Kritische Elektronenfallendichte
CG	–	Common Gate
C_G	–	Kapazität zwischen Gate und Kanal

CHE	–	Channel Hot Electron
C_{IC}	–	Kapazität eines Leiters
C_{IN}	–	Eingangskapazität eines CMOS-Gatters
C_{IN_ADD}	–	Zusätzliche Eingangskapazität eines CMOS-Gatters
C_{LOAD}	–	Lastkapazität eines CMOS-Gatters
CMOS	–	Complementary metal-oxide-semiconductor
CMP	–	Chemical-mechanical polishing
CN	–	Common Node
CORS	–	Circuit Oxide Reliability Simulator
C_{OX}	–	Kapazität des Gateoxides je Fläche des Gateoxides
CPU	–	Central processing unit
CSM	–	Current-Source-Modell
Cu	–	Kupfer
D	–	Durchschnittliche Defektrate je Fläche
DAHC	–	Drain Avalanche Hot Carrier
DFM	–	Design for manufacturing
DFS	–	Dynamic frequency scaling
DFY	–	Design for yield
DGatter	–	Mittels verdoppelter Gatter verbesserte Schaltungen
DIBL	–	Drain induced barrier lowering
DR	–	Verschleißrate
DRC	–	Design-Rule-Check
DRM	–	Dynamic reliability management
DTM	–	Dynamic temperature management
d_{TMR}	–	Dauer des TMR-Modus
DTrans	–	Mittels verdoppelter Transistoren verbesserte Schaltungen
DVS	–	Dynamic voltage scaling
DwC	–	Duplication with comparison
E	–	Elektrisches Feld
$E_{0\rightarrow1}$	–	Energie zum Aufladen einer Lastkapazität eines CMOS-Gatters
E_{aEM}	–	Aktivierungsenergie der Elektromigration
E_{aGOB}	–	Aktivierungsenergie des Gateoxiddefektes
E_{aHC}	–	Aktivierungsenergie der Hot Carriers
E_B	–	Potentialbarriere
E_{DYN}	–	Energie zum Auf- bzw. Entladen einer Lastkapazität eines CMOS-Gatters
E_F	–	Fermienergie
E_L	–	Energie des unteren Leitungsbandes
Enh	–	Mittels verdoppelter Transistorstacks verbesserte CMOS-Designs
E_{OX}	–	Elektrisches Feld über dem Gateoxid
ERC	–	Electrical-Rule-Check
$f(t)$	–	Wahrscheinlichkeitsdichtefunktion

$F(t)$	–	Wahrscheinlichkeit vor oder zum Zeitpunkt t auszufallen
f_{CLK}	–	Taktfrequenz
f_{TMR}	–	Frequenz des TMR-Modus
G_H	–	Defektelektronentunnelrate
GOB	–	Gate oxide breakdown
H	–	Wasserstoff
HC	–	Hot carriers
I_{BTB}	–	Band-to-band-Leckstrom
I_{BULK}	–	Substratstrom
IC	–	Interconnect, Leiter
I_{D_DEFEKT}	–	Drain-Source-Strom eines defekten MOSFET (Renovell-Modell)
I_{DMIN_DEFEKT}	–	Minimaler Drain-Source-Strom eines defekten MOSFET (Renovell-Modell)
I_{DS}	–	Drain-Source-Strom
I_{DS_A}	–	Drain-Source-Strom des Transistors T_A (Renovell-Modell)
I_{DS_M}	–	Drain-Source-Strom des Transistors T_M (Renovell-Modell)
I_{DSAT}	–	Gesättigter Drain-Source-Strom
I_{DSAT_0}	–	Ursprünglicher gesättigter Drain-Source-Strom eines MOSFET
I_{DSAT_DEFEKT}	–	Gesättigter Drain-Source-Strom eines defekten MOSFET (Renovell-Modell)
I_{G_DEFEKT}	–	Gate-Source-Strom eines defekten MOSFET (Renovell-Modell)
I_{GATE}	–	Gateoxidleckstrom
I_{GMAX_DEFEKT}	–	Maximaler Gate-Source-Strom eines defekten MOSFET (Renovell-Modell)
I_{GOB}	–	Durchbruchstrom
I_{GOB_HIGH}	–	Durchbruchstrom am Gatterausgang bei $V_{OUT} = V_{HIGH}$
I_{GOB_LOW}	–	Durchbruchstrom am Gatterausgang bei $V_{OUT} = V_{LOW}$
I_{GOB_MAX}	–	Durchbruchstrom am Gatterausgang bei $V_{OUT} = V_{OUT\_MAX}$
I_{GOB_MIN}	–	Durchbruchstrom am Gatterausgang bei $V_{OUT} = V_{OUT\_MIN}$
I_{GSAT}	–	Gesättigter Gate-Source-Strom
$I_{MOD}(t)$	–	Modifizierte Stromkurve am CMOS-Gatterausgang eines defekten MOSFET oder eines MOSFET, das mit defekten MOSFET verbunden ist
I_{N_GOB}	–	Gesamter Strom durch n-MOSFET mit Gateoxiddefekten, die mit demselben Designknoten verbunden sind
$I_{OUT}(t)$	–	Stromkurve am CMOS-Gatterausgang
i_{OUT}	–	Strom am CMOS-Gatterausgang
I_{P_GOB}	–	Gesamter Strom durch p-MOSFET mit Gateoxiddefekten, die mit demselben Designknoten verbunden sind
I_{PN}	–	P-n-Leckstrom
I_{PT}	–	Punchthrough-Leckstrom
IQR	–	Interquartilsabstand

I_{SC}	–	Durchschnittlicher Kurzschlussstrom
I_{SUB}	–	Subthresholdleckstrom
$I_{TREIBER}$	–	Diskrete Stromkurve der Gatter am Eingangspin
ITRS	–	International Technology Roadmap for Semiconductors
J	–	Stromdichte
J_{CRIT}	–	Kritische Stromdichte für die Elektromigration
J_e	–	Stromdichte des Elektronentunnelstromes
J_{FN}	–	Stromdichte des Fowler-Nordheim tunneling
J_H	–	Defektelektronenstromdichte
J_{TAT}	–	Stromdichte des trap-assisted tunneling
k_B	–	Boltzmannkonstante
L	–	Transistorgatelänge
L_A	–	Transistorgatelänge des Transistors T_A (Renovell-Modell)
L_B	–	Transistorgatelänge des Transistors T_B (Renovell-Modell)
L_D	–	Länge der Verarmungsregion
L_{EFF}	–	Effektive Transistorgatelänge
L_M	–	Transistorgatelänge des Transistors T_M (Renovell-Modell)
L_{MIN}	–	Minimale Transistorgatelänge
L_{ORG}	–	Ursprüngliche Transistorgatelänge eines defektlosen MOSFET
LPR	–	Verlustleistungsreduzierte TMR-Designs
LVS	–	Layout-vs.-Schematic-Check
MOSFET	–	Metal-oxide-semiconductor field-effect transistor
$MTTF$	–	Mean time to failure
$MTTF_{GOB\_TRANSISTOR}$	–	Mean time to failure eines MOSFET mit einem GOB
NBTI	–	Negative Bias Temperature Instability
n_{EM}	–	Materialkonstante der Elektromigration
n_{GOB}	–	Temperaturabhängiger Exponent des Gateoxiddefektes
NLDM	–	Non-Linear-Delay-Modell
n_{SUB}	–	Substratstromabhängigkeitsexponent
N_{TRAP}	–	Elektronenfallendichte
OPC	–	Optical proximity correction
PBTI	–	Positive Bias Temperature Instability
PC	–	Technologieparameter des „alpha power law“-Modells
p_D	–	Parameter zur Bestimmung der maximale Verzögerungszeit eines zu berücksichtigenden Pfades in einem CMOS-Design
P_{DYN}	–	Dynamische Leistungsaufnahme
P_{GESAMT}	–	Gesamte Leistungsaufnahme
P_{IN}	–	Wahrscheinlichkeit des Eintritts in eine Elektronenfalle
P_{ON}	–	Wahrscheinlichkeit eines MOSFET, während der Betriebszeit durchgeschaltet zu sein
P_{OUT}	–	Wahrscheinlichkeit des Austritts aus einer Elektronenfalle
P_{SC}	–	Leistungsaufnahme durch Kurzschlussströme

P_{STAT}	–	Statische Leistungsaufnahme
PV	–	Technologieparameter des „alpha power law"-Modells
Q_{25}	–	0.25-Quantil
Q_{75}	–	0.75-Quantil
Q_{BD}	–	Durchbruchsladung
$Q_{CHANNEL}$	–	Ladung im Kanal
Q_f	–	Oxidladungsdichte
Q_{it}	–	Interfacedefektladungsdichte
Q_P	–	Kritische Elektronenlochflussdichte
$R(t)$	–	Zuverlässigkeit: Wahrscheinlichkeit der fehlerfreien Funktion bis zum Zeitpunkt t
rand	–	Zufallszahl zwischen 0 und 1
r_{COMP}	–	Anteil des Dual-Modus an der gesamten Betriebszeit
REF	–	Nicht modifizierte Referenzdesigns
R_{GOB}	–	Gateoxiddefektwiderstand
R_{INIT}	–	Widerstand zur Darstellung der initialen Treiberstärke in Spice-Simulationen
$R_{PARALLEL}(t)$	–	Zuverlässigkeit eines parallelen Systems
$R_{SERIAL}(t)$	–	Zuverlässigkeit eines seriellen Systems
RTL	–	Register transfer level
r_{TMR}	–	Anteil des TMR-Modus an der gesamten Betriebszeit
SAIF	–	Switching Activity Interchange format
SCHE	–	Substrate current induced Hot Electron
SFT	–	Streßfreie Temperatur
SGHE	–	Secondarly generated Hot Electron
SHE	–	Substrate Hot Electron
SHH	–	Substrate Hot Hole
Si	–	Silizium
SILC	–	Stress induced leakage current
SiO_2	–	Siliziumdioxid
SPA	–	Spannungspegelanalyse
SRD	–	Spin, rinse and dry
SSK	–	Spannungs-Strom-Kurve
SSK_{DB_HIGH}	–	Spannungs-Strom-Kurve der Datenbank für einen Ausgangspegel, der logisch 1 entspricht
SSK_{DB_LOW}	–	Spannungs-Strom-Kurve der Datenbank für einen Ausgangspegel, der logisch 0 entspricht
SSK_N	–	Spannungs-Strom-Kurve eines defekten n-MOSFET
SSK_P	–	Spannungs-Strom-Kurve eines defekten p-MOSFET
SSTA	–	Statistical static timing analysis
STA	–	Static timing analysis
T	–	Temperatur

t_{23}	–	Zuschaltzeitpunkt der redundanten Transistorstacks eines Enh-Designs
TAT	–	Trap-assisted tunneling
t_D	–	Verzögerungszeit eines CMOS-Gatters/einer CMOS-Schaltung
t_{D_23}	–	Um 2/3 vergrößerte Verzögerungszeit einer CMOS-Schaltung
t_{D_LIMIT}	–	Maximal zulässige Verzögerungszeit einer CMOS-Schaltung
$t_{D\_ORIGINAL}$	–	Ursprüngliche Verzögerungszeit einer CMOS-Schaltung
TDDB	–	Time-dependent dielectric breakdown
t_{ENDE}	–	Simulationsendzeitpunkt
t_{FALL}	–	Signalabfallzeit eines CMOS-Gatters
t_{FALL_IN}	–	Abfallzeit eines Eingangssignals eines CMOS-Gatters
t_{FALL_OUT}	–	Abfallzeit des Ausgangssignals eines CMOS-Gatters
t_{FINAL}	–	Zeitpunkt des Wechsels in den finalen Modus
t_{GOB}	–	Defekteintrittszeitpunkt
TMR	–	Triple modular redundancy
t_{OP}	–	Betriebszeit
t_{OX}	–	Gateoxidschichtdicke
t_{Pfad}	–	Maximale Verzögerungszeit zur Auswahl bestimmter Pfade im Design
t_{RISE}	–	Signalanstiegzeit eines CMOS-Gatters
t_{RISE_IN}	–	Anstiegzeit eines Eingangssignals eines CMOS-Gatters
t_{RISE_OUT}	–	Anstiegzeit des Ausgangssignals eines CMOS-Gatters
t_{SLOPE_IN}	–	Dauer der Eingangsflanke
ttf	–	Time to failure
$ttf_{1/E}$	–	Lebensdauer des 1/E-Modells
ttf_E	–	Lebensdauer des E-Modells
ttf_{EM}	–	Lebensdauer in Hinblick auf Elektromigration
ttf_{GOB}	–	Lebensdauer in Hinblick auf Gateoxiddefekte
ttf_{HC}	–	Lebensdauer in Hinblick auf Hot Carriers
ttf_{MIN}	–	Lebensdauer des GOB-Simulator-Referenztransistors
ttf_{TRANS}	–	Durch GOB-Simulator berechnete Lebensdauer eines MOSFET
t_{VIN_HALF}	–	CCS-Referenzzeitpunkt, an der der Wert der Eingangsspannung die Hälfte von V_{DD} erreicht
v	–	Durchschnittliche Geschwindigkeit eines Ladungsträgers
v_{SAT}	–	Gesättigte Geschwindigkeit eines Ladungsträgers
V_{DD}	–	Versorgungsspannung
V_{DS}	–	Drain-Source-Spannung
V_{DSAT}	–	Gesättigte Drain-Source-Spannung
V_{FB}	–	Flachbandspannung
V_{GS}	–	Gate-Source-Spannung
V_{GD}	–	Gate-Drain-Spannung
V_{HIGH}	–	Minimal zulässiger Spannungspegel eines CMOS-Gatters, der einer logischen 1 entspricht
VHDL	–	Very-high-speed integrated circuit Hardware Description Language

V_{IN_HIGH}	–	Kleinster simulierter Minimalspannungspegel eines Eingangspins eines CMOS-Gatters der Referenzbibliothek, der einer logischen 1 entspricht
V_{IN_LOW}	–	Größter simulierter Maximalspannungspegel eines Eingangspins eines CMOS-Gatters der Spice-Referenzbibliothek, der einer logischen 0 entspricht
V_{IN_MAX}	–	Berechnete maximale Spannungspegel der Eingangspins eines CMOS-Gatters
V_{IN_MIN}	–	Berechnete minimale Spannungspegel der Eingangspins eines CMOS-Gatters
V_{LOW}	–	Maximal zulässiger Spannungspegel eines CMOS-Gatters, der einer logischen 0 entspricht
VLSI	–	Very-large-scale integration
V_{OUT}	–	Gatterausgangsspannung
V_{OUT_MAX}	–	Berechnete maximale Spannungspegel der CMOS-Gatter-Ausgangspins
$V_{OUT\_MAX\_BASE}$	–	Maximale Spannungspegel der CMOS-Gatter-Ausgangspins der Spice-Bibliothek
V_{OUT_MIN}	–	Berechnete minimale Spannungspegel der CMOS-Gatter-Ausgangspins
$V_{OUT\_MIN\_BASE}$	–	Minimale Spannungspegel der CMOS-Gatter-Ausgangspins der Spice-Bibliothek
V_{SB}	–	Substratspannung
V_{SS}	–	Massespannung
V_{TH}	–	Schwellspannung
V_{TH0}	–	Schwellspannung bei $V_{SB} = 0$
V_{TH_DIBL}	–	Schwellspannung im DIBL-Fall
W	–	Transistorgatebreite
W_A	–	Transistorgatebreite des Transistors T_A (Renovell-Modell)
W_B	–	Transistorgatebreite des Transistors T_B (Renovell-Modell)
W_{EFF}	–	Effektive Transistorgatebreite
W_M	–	Transistorgatebreite des Transistors T_M (Renovell-Modell)
W_0	–	Ursprüngliche Transistorgatebreite eines defektlosen MOSFET

Erstes Kapitel

1 Einleitung und Motivation

Seit Anbeginn der Fertigung integrierter Schaltungen wurden deren Techniken immer weiter verbessert, um die Strukturgrößen der Transistoren, der Schaltungen und die Leitungen zwischen ihnen stetig zu verkleinern. Diese Größe definiert die kleinste im Halbleiterbereich erzeugbare Struktur, was in den meisten Fällen die Gateoxidlänge eines Metalloxid-Halbleiter-Feldeffekttransistors (MOSFET) ist. Bis in die 1960er Jahren noch mit über 10 µm angegeben, liegt sie seit langem im so genannten Submicron-Bereich, so dass Chips mittlerweile serienmäßig mit Abmessungen von 32 nm und kleiner gefertigt werden können. Diese Entwicklung wurde schon 1965 von Gordon Moore beschrieben, in dem er postulierte, dass sich die Integrationsdichte elektronischer Schaltkreise alle 18 Monate verdoppeln werde [Moo65]. Diese Entwicklung bietet viele Vorteile, weshalb sie auch vehement fortgesetzt wird. So erhöht sich die Anzahl elektronischer Bauteile je Chip, mit Moore's Gesetz als Faustformel, um das Zweifache alle 1.5 Jahre. Dadurch ist es möglich, mehr Funktionalität in die Chips zu integrieren. Des Weiteren verringern sich die Schaltzeiten innerhalb der Gatter und die Verbindungsleitungen zwischen ihnen werden ebenso verkürzt. Beides resultiert in einer geringeren Verzögerungszeit der Schaltung, was zu schnelleren Berechnungszeiten führt.

Diese Arbeit fokussiert sich auf MOSFETs, deren Miniaturisierung, auch Skalierung genannt, allerdings auch negative Aspekte mit sich bringt. So gewinnen quantenmechanische Effekte, beispielsweise das Tunneln von Elektronen, immer mehr an Einfluss auf das Gateoxid, als dünnste Schicht des Transistors. Gleichzeitig steigen die elektrischen Felder an, die auf das Gate wirken, da die Versorgungsspannung nicht in dem Maße skaliert werden kann wie die Strukturgrößen. Diese beiden Mechanismen wirken sich zweifach negativ aus. Zum einen steigen die statischen Ströme innerhalb einer Schaltung an, was in einem erhöhten Energieaufwand, größeren Kühlkörpern und höheren Temperaturen integrierter Schaltungen mündet. Zum anderen leidet die Zuverlässigkeit integrierter Schaltungen. Weil das Gateoxid durch die erwähnten Effekte über die Betriebsdauer einer Schaltung degeneriert, kann es zu

Funktionsstörungen und -ausfällen der Schaltung kommen. Durch die fortschreitende Skalierung wird diese Degeneration verstärkt und beginnt immer früher. Weiterhin führen der erhöhte Energieaufwand und die damit einhergehende Temperaturerhöhung und häufigere Temperaturwechsel auch zu einer Verstärkung zuverlässigkeitssenkender Effekte.

Aus diesem Grund hat das Gremium der International Technology Roadmap for Semiconductors (ITRS) die Zuverlässigkeit integrierten Schaltungen seit Mitte der 2000er als sehr dringendes Problem auf der Agenda. Die klassischerweise dem Produktionsingenieur zugedachte Zuverlässigkeitserhaltung und -steigerung integrierter Schaltungen bekommt immer mehr Bedeutung für den Designingenieur. Aufgrund der enormen Komplexität heutiger Chips, die mehrere Milliarden Transistoren enthalten können, sind vollständige Tests nicht mehr zu bewerkstelligen. Daher ist das rechnerunterstützte Chipdesign (engl. Computer Aided Design – CAD), vor allem im Bereich der kombinatorischen Schaltungen, auch in Hinblick auf die Zuverlässigkeit unterstützungsbedürftig [SIA07], damit die Ausbeute und die Langlebigkeit integrierter Schaltungen so hoch wie bisher bleiben. Weiterhin werden mehr Werkzeuge notwendig, um die Statistik der zuverlässigkeitssenkenden Effekte genauer nachzuempfinden bzw. die Auswirkungen auf das Gesamtdesign exakter vorherzusagen [SIA11].

Ziel dieser Arbeit ist die Entwicklung von Techniken zur Erhöhung der Zuverlässigkeit für kombinatorische integrierte Schaltungen und eines Simulators zur Analyse der Auswirkungen von Gateoxiddefekten. Die generierten Ansätze fokussieren sich auf die Gatterebene, da sie sich gut in CMOS-Designablauf integrieren lassen. Ferner wird auch auf die anderen Designparameter, wie Fläche, Verzögerungszeit und Leistungsaufnahme eingegangen. Besonders die Leistungsaufnahme steht dabei im Vordergrund, da ihre Reduzierung ebenfalls zur Erhöhung der Zuverlässigkeit beiträgt.

Für den Einstieg werden in Kapitel 2 die Grundlagen der CMOS-Technologie und der CMOS-Schaltungstechnik erklärt. Darauf aufbauend, werden in Kapitel 3 die Effekte, die in Nanometer-Technologien die Zuverlässigkeit vermindern, näher erläutert. Anschließend werden Techniken vorgestellt, die heutzutage herangezogen werden, um zuverlässigere Schaltungen zu erzeugen. Diese umfassen neben der Defektvermeidung vor allem auch Techniken zur Defektmaskierung bzw. -toleranz.

Insbesondere auf Gatterebene existieren wenig redundante Ansätze zur Zuverlässigkeitssteigerung kombinatorischer Schaltungen hinsichtlich Gateoxiddefekten. Kapitel 4 zeigt mehrere Lösungen zur Steigerung der Zuverlässigkeit., die auf redundanten Techniken der Gatter- und Architekturebene beruhen. In einem ersten Ansatz wird die Leistungsaufnahme üblicher redundanter Architekturen mit zusätzlichen Überwachungs- und geänderten Ausführungsstrukturen auf bis zu 40 % gesenkt, wobei die Zuverlässigkeit sogar um ca. 20 % gesteigert werden kann. Im Anschluss werden redundante Ansätze auf Gatterebene entwickelt, die dazu führen, dass der Verschleiß der Transistoren in Bezug auf Gateoxiddefekte und im Vergleich zu den Originaldesigns verlangsamt wird. Die Zuverlässigkeit kann schaltungsabhängig um bis 60 % erhöht werden, was einer Verdopplung verglichen mit bisherigen Ansätzen auf Gatterebene entspricht. Diese Verbesserungen werden mit

vergleichsweise geringen Steigerungen der Leistungsaufnahme und der Verzögerungszeit von ungefähr 10 % erreicht, wobei die Fläche um ca. 50 % erhöht wird. Ferner zeigt sich, dass dieser Ansatz sehr gut in den Standard-Designflow integriert werden kann. Um diesen Umstand auszunutzen, wird in Kapitel 5 die Entwicklung eines Zuverlässigkeitssimulators auf Gatterebene vorgestellt. Ziel hierbei war es, sowohl den Verschleiß einer integrierten Schaltung nachzuvollziehen, als auch darauf aufbauend Schwachstellen in einem Design zu erkennen, um die in Kapitel 4 vorgestellten redundanten Techniken effizient einzusetzen. Die gesamte Struktur der Arbeit ist in Abbildung 1-1 grob zusammengefasst, wobei blau gekennzeichnete Bereiche den neuen Beitrag durch diese Arbeit markieren.

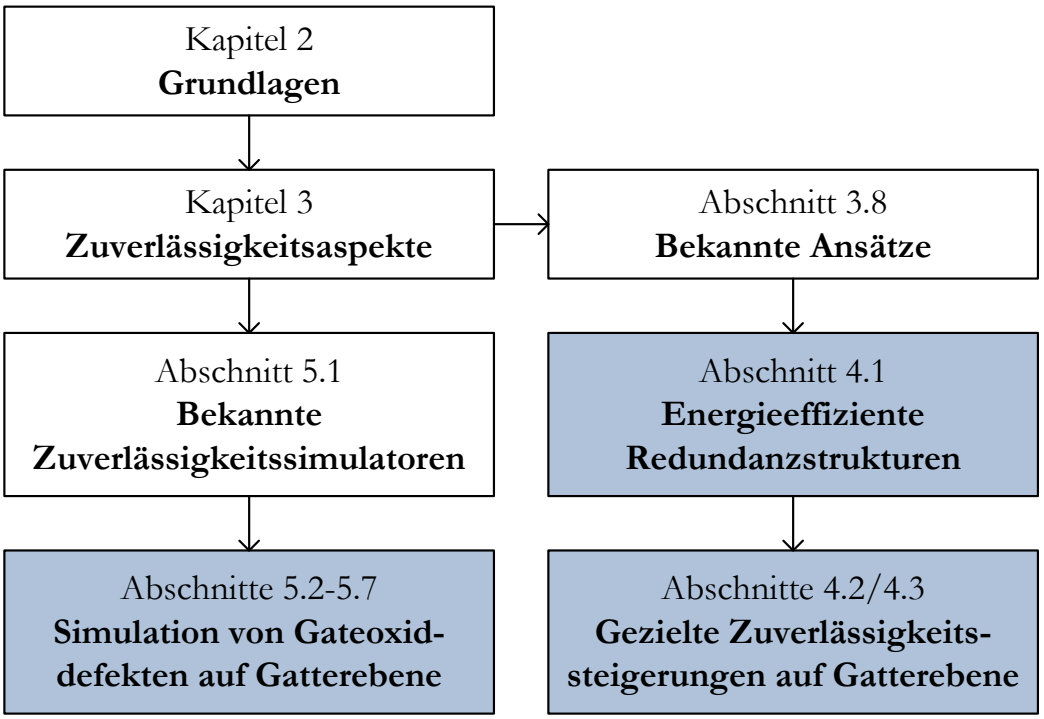


Abbildung 1-1 : Struktur der vorliegenden Arbeit. Blau unterlegte Bereiche kennzeichnen den neuen Beitrag dieser Arbeit

Zweites Kapitel

2 CMOS-Logik und CMOS-Technologie

Dieses Kapitel wird einen Überblick über grundlegende Funktionsprinzipien und den Aufbau der CMOS-Technologie und -Logik geben. Dabei steht CMOS für „komplementärer Metalloxid Halbleiter“ (engl. Complementary metal-oxide-semiconductor). Prinzipiell wird der Begriff CMOS für zwei zu unterscheidende Bereiche verwandt. Zum einen definiert die CMOS-Logik eine Logikfamilie, was das Verknüpfen von integrierten Bauelementen (Transistoren) zu logischen Bausteinen (Logikgatter wie z. B. „UND“, „ODER“) beinhaltet. Zum anderen umfasst die CMOS-Technologie den industriellen Fertigungsprozess, mit dem man nach wie vor den größten Teil aller weltweit produzierten integrierten Schaltungen herstellt. Im Folgenden wird der Begriff CMOS für beide Bereiche verwandt, da sie aufeinander aufbauen.

Da eine integrierte Schaltung verschiedenste Aufgaben und Untereinheiten umfassen kann, gibt es verschiedene Abstraktionsebenen, die aufeinander aufbauen, um Forschung und Entwicklung zu systematisieren. Ausgehend von einer bestimmten Aufgabe, die ein Chip ausüben soll, wird diese in der Systemebene in verschiedene Module aufgeteilt. Dazu zählen beispielsweise zentrale Steuereinheiten (engl. central processing unit – CPU), Speicher- und Rechenblöcke, Sensoren und andere Strukturen. Diese Module werden dann in nebenläufige Prozesse aufgespalten (Algorithmusebene), die aus den eigentlichen Funktionen und deren Kontrollstrukturen bestehen. Nun werden die gewünschten Algorithmen auf Speicherelemente (Register) und die dazwischen liegenden logischen Blöcken aufgeteilt, weshalb diese Architekturebene auch Registertransferebene (engl. register transfer level – RTL) genannt wird. Um das in dieser Ebene gewünschte Verhalten der Schaltung zu beschreiben, werden Hardwarebeschreibungssprachen genutzt. Die für diese Arbeit verwendete Sprache ist VHDL (engl. Very-high-speed integrated circuit Hardware Description Language). Mittels VHDL ist es möglich, Auswirkungen auf Zustände und Ausgangsvektoren (engl. outputs) der Schaltung als Reaktion auf bestimmte Eingangsvektoren (engl. inputs) und interne Zustände zu definieren. Sollte diese Strukturbeschreibung der Schaltung auf elementare und komplexere logische

Funktionen automatisch abbildbar sein („synthesefähig“), kann das erzeugte RTL-Design mit Hilfe eines Synthesetools in Logikgatter und sequentielle Gatterelemente wie Latches und Flipflops überführt werden, was als Gatterebene bezeichnet wird. Diese Gatter liegen für bestimmte CMOS-Technologien als Bibliothek von Standardgattern vor, in der ihr Verhalten für bestimmte Randbedingungen (Temperatur, Produktionsbedingungen,…) beschrieben ist. Um auch nichtlineare Abhängigkeiten der realen Welt nachzubilden und zu simulieren, werden die Gatter in der Transistorebene auf grundlegende elektronische Bauelemente reduziert, was sowohl die grundlegenden Bauelemente der CMOS-Gatter (Transistoren) als auch parasitäre Elemente (Widerstände, Kapazitäten und Induktivitäten) umfasst. Da deren Verhalten auf Differentialgleichungen basiert, kann das Gesamtverhalten der Schaltung viel realistischer vorhergesagt werden. Die Beschreibung, auf der der Produktionsprozess aufbaut, ist das Layout. Hierbei werden die elektronischen Bauelemente in vertikal unterscheidbare Schichten dargestellt. Dadurch ist es möglich, Masken für den sequentiellen Produktionsprozess herzustellen, damit die Produktion der Schaltung Schicht für Schicht durchgeführt werden kann. Durch Extraktion des Layouts und Analyse der aus dem Aufbau resultierender physikalischen Effekte kann eine sehr genaue Vorhersage des zu erwartenden Verhaltens erzielt werden. All diese Ebenen sind in Anlehnung an das Gajski-Kuhn Y-Diagramm in Abbildung 2-1 von oben nach unten mit abnehmender Abstraktionsstufe dargestellt. Der Vorteil einer größeren Abstraktionsstufe ist die schnellere Simulation des Schaltungsverhaltens und ein größerer Effekt bei eventuellen Schaltungsverbesserungen, allerdings sinkt auch die Genauigkeit der Vorhersagen. Ungenauigkeiten aufgrund einer einfacheren Modellierung führen zu ungenaueren Simulationen. Um keinesfalls falsche Vorhersagen zu erzeugen, werden deshalb pessimistischeren Annahmen getroffen, so dass sich diese Ungenauigkeiten nicht als Fehler im fertigen Chip manifestieren.

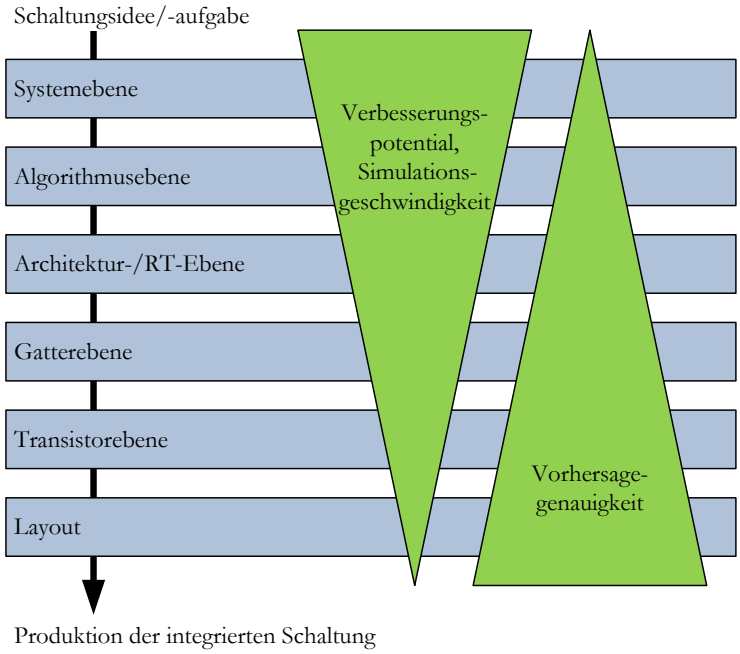


Abbildung 2-1 : CMOS-Designprozess aufgeteilt in die einzelnen Abstraktionsebenen

In den anschließenden Unterkapiteln werden zuerst MOS-Feldeffekttransistoren und ihre Verhaltensparameter erläutert. Darauf aufbauende CMOS-Gatter und deren digitale Synthese werden danach näher betrachtet, sowie Verhaltensparameter erläutert, die eine Schaltung auf Gatter- und RT-Ebene beschreiben bzw. charakterisieren. Da diese Arbeit dazu dient, die Zuverlässigkeit von integrierten Schaltungen im Designprozess zu analysieren und zu verbessern, dienen die im folgenden Kapitel vorgestellten Grundlagen [Wes93] [Rab96] zum Verständnis der weiteren Kapitel.

2.1 Der MOS-Feldeffekttransistor

Das Material der CMOS-Technologie ist das Silizium. Silizium ist ein Halbleiter der Gruppe IV und ist in purer Form ein schlechter Leiter, da alle seine vier Valenzelektronen chemisch stabile Bindungen mit benachbarten Siliziumatomen bilden, so dass eine Gitterstruktur fast ohne freie Ladungsträger entsteht – Abbildung 2-2 a). Werden nun Fremdatome (Störstellen) in das Gitter eingeführt, was als Dotierung bezeichnet wird, erhöht sich die Leitfähigkeit des Siliziums, da die Beweglichkeit der Ladungsträger (Elektronen, Elektronenlöcher) durch neu hinzugefügten Zwischenniveaus zwischen Valenz- und Leitungsband erhöht wird. Atome mit fünf Valenzelektronen, so genannte Donatoren, die in das Silizium-Gitter eingebracht werden, setzen ein Elektron frei, was sich nun leicht ablösen und frei im Gitter bewegen kann und damit die Leitfähigkeit erhöht – Abbildung 2-2 b). Dies nennt man eine n-Dotierung und das Gitter ist ein Elektronenleiter, da Elektronenüberschuss besteht. Analog dazu fehlt bei Einbringung von Fremdatomen mit nur drei Valenzelektronen, die Elektronenakzeptoren, ein Elektron zur Nachbarbildung – Abbildung 2-2 c). Bei dieser p-Dotierung kann die Fehlstelle (Elektronenloch, Defektelektron) nun leichter durch andere Valenzelektronen besetzt werden, weshalb die Fehlstelle durch das Gitter „transportiert“ wird. Es entsteht eine Löcherleitung oder p-Leitung.

Diese unterschiedlichen Leitungsarten werden für die CMOS-Schaltungstechnik genutzt und führen zu zwei verschiedenen Typen von MOS-Feldeffekttransistoren (MOSFET), welcher das grundlegende Bauelement der CMOS-Technologie bildet. Er besteht aus einer gut leitenden Steuerelektrode (engl. Gate) über einer isolierenden Gateoxidschicht aus SiO_2 der Dicke t_{OX}, die

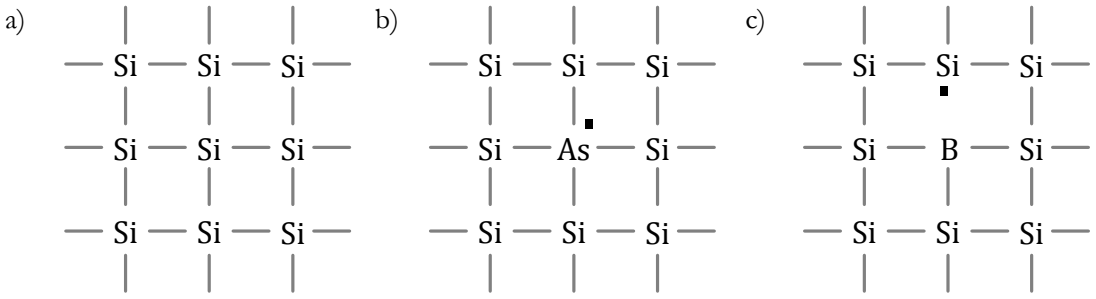


Abbildung 2-2 : Gitterstrukturen des Siliziums (zweidimensional, real: dreidimensional):
a) Reines Silizium, ohne freie Ladungsträger
b) Silizium mit Arsen (Donator), ein freies Elektron je Arsenatom
c) Silizium mit Bor (Akzeptor), ein Defektelektron je Boratom

eine Fläche mit der Länge L und der Breite W aufspannen. Darunter befindet sich das Grundmaterial aus Silizium, was als Substrat (engl. body, bulk) bezeichnet wird. Der Name Metalloxid-Halbleiter leitet sich aus der früher üblichen Verwendung von Metall als Gatematerial ab. Heutzutage wird jedoch zumeist Polysilizium verwendet, während der Name blieb. Wie erwähnt, gibt es nun zwei Typen eines MOSFET. Der n-MOSFET besteht p-dotiertem Substrat und zwei Inseln aus n-dotiertem Material, die an das Gate angrenzen – Abbildung 2-3 a). Diese beiden Gebiete werden Drain und Source genannt. Beim p-MOSFET ist die Dotierung umgekehrt – Abbildung 2-3 b). Da der Drainanschluss in CMOS mit der Versorgungsspannung verbunden ist, wird diese auch als V_{DD} dargestellt. Analog verhält es sich mit dem Sourcegebiet, welches in CMOS mit der Masse V_{SS} verbunden wird.

Im Folgenden wird näher auf die Herstellung und die die Funktionsweise der MOSFET-Typen eingegangen. Dabei werden die Erläuterungen auf den selbstsperrenden Anreicherungstypen beschränkt, der erst mit Anlegen einer Spannung Strom leitet. Eine andere Form ist der selbstleitende Verarmungstyp, der in der Praxis aber nicht so häufig verwendet wird.

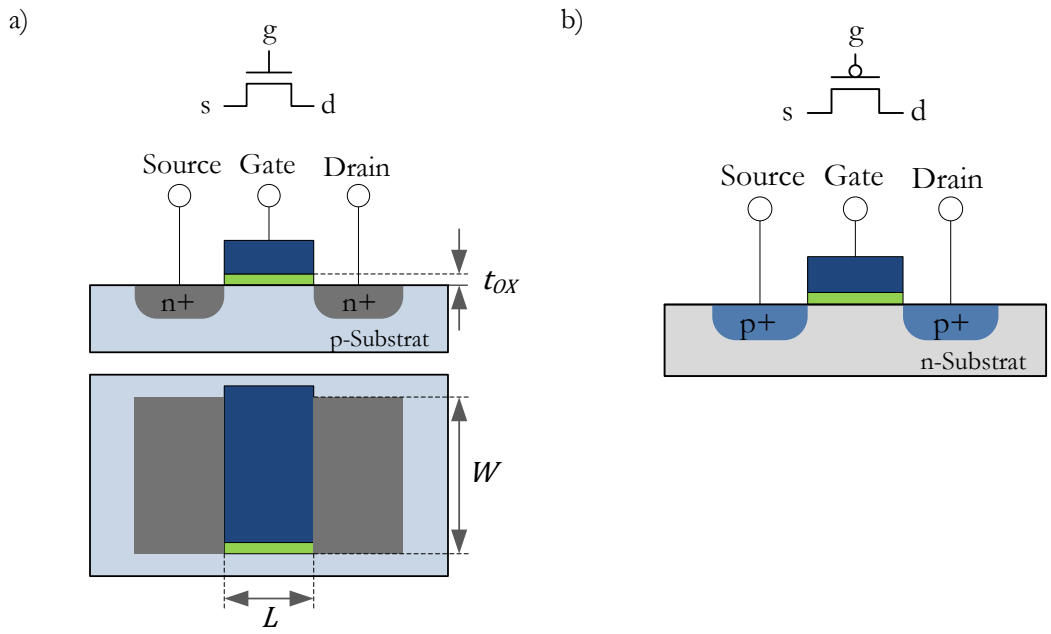


Abbildung 2-3 : CMOS-Transistortypen
a) n-MOSFET mit Symbol, Querschnitt und Draufsicht
b) p-MOSFET mit Symbol und Querschnitt

2.1.1 Herstellung

Die Produktion integrierter Schaltungen erfordert, dass beide MOSFET-Typen auf dem gleichem Siliziummaterial aufgebaut werden müssen. Dafür wird eine Wanne (engl. well) mit dem Substratmaterial (n-Substrat für p-MOSFET, p-Substrat für n-MOSFET) für mindestens einen beider Typen angelegt. In modernen Prozessen werden für beide Typen Wannen angelegt, die auf ein epitaxisches Material aufgetragen werden. Dieses Material ist ein hochwertiges und mehrfach gereinigtes Silizium, das in Scheiben mit einem Durchmesser von bis zu 300 mm (engl. Wafer)

zur weiteren Produktion der integrierten Schaltungen hergestellt wird. Dieser nachfolgende Produktionsprozess basiert auf mehrfach wiederholten Prozessschritten, mit denen Strukturen in das Ausgangsmaterial eingebracht werden. Dies ist notwendig, da Transistoren und die dazugehörigen Verbindungen in mehreren Schichten aufgebaut werden müssen. Der Basisprozess ist dabei die Fotolithografie. Hierbei wird ein lichtempfindlicher Fotolack aus Polymer auf den Wafer gebracht. Danach wird dieser Lack mit Hilfe einer Maske belichtet. Diese Maske ist für jede Ebene mit den gewünschten Strukturen angefertigt worden. Anschließend wird der Fotolack in einem chemischen Bad entwickelt, so dass die belichteten Stellen aus dem Lack entfernt werden, sollte es ein Positivlack sein. Beim Negativlack werden die unbelichteten Stellen entfernt. Mit Hilfe der nachfolgenden Wärmebehandlung (engl. soft baking) wird der Lack auf dem Wafer stabilisiert. Dann wird frei liegendes Material, dass nicht durch den Fotolack geschützt ist, durch eine Säurebehandlung weggeätzt. Der Wafer wird im Anschluss mit demineralisiertem Wasser gereinigt und mit Stickstoff getrocknet, damit so wenig Fremdpartikel in der Schaltung sind wie möglich (engl. spin, rinse and dry – SRD). In der Folge können in die ausgeätzten Strukturen verschiedene Materialien mit unterschiedlichen Techniken, wie Diffusion und Ionenimplantation, Abscheidung, eingebracht werden. Wenn dies geschehen ist, wird im abschließenden Schritt der Fotolack entfernt. Sollten noch eventuell vorhandene Unebenheiten ausgeglichen werden, wird der Wafer noch chemisch-mechanisch poliert (CMP). Die Abfolge der Fotolithografie ist in Abbildung 2-4 dargestellt.

Die vorgestellten Prozessschritte werden mehrfach ausgeführt, um die einzelnen Schichten aufzutragen, die die Transistoren und die Leiterbahnen bilden sollen. Dafür werden zuerst die aktiven Regionen definiert, die die Transistoren bilden werden. Andere Regionen werden mit SiO_2 gefüllt. Es dient sowohl als Gateisolator als auch als Isolator zwischen den Strukturen.

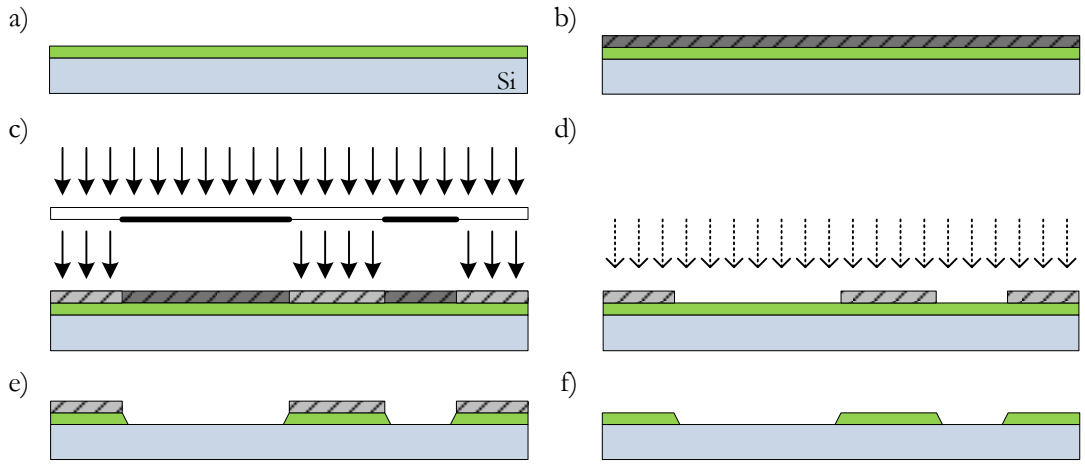


Abbildung 2-4 : Ablauf der Fotolithografie anhand des Aufbringens einer SiO_2-Schicht:
a) Ausgangslage: Abgelagerte SiO_2-Schicht auf dem Si-Substrat
b) Ablagerung des negativen Fotolacks
c) UV-Belichtung der SiO_2-Schicht durch eine optische Maske
d) Chemisches oder physikalische Trockenätzprozess der SiO_2-Schicht nach Abätzen des unbelichteten und nicht entwickelten Fotolackes
e) Nach dem Ätzprozess
f) Nach dem Entfernen des Fotolackes

Danach werden die leichtdotierten Wannen durch Ionenimplantation eingebracht. Für die Transistoren werden die stark dotierten Source- und Draingebiete in die Wannen implantiert oder durch Diffusion eingebracht. Anschließend wird das Gateoxidmaterial SiO_2 durch thermische Oxidation aufgebracht und darauf das Polysilizium aufgetragen. Es folgt die Erstellung der Metallebenen der Leiterbahnen, deren Kontakte zu den Transistoranschlüssen (engl. contact) und die Verbindungen untereinander (engl. via) nacheinander je nach Anzahl der Metallebenen als Backendprozess der Fotolithografie eingefügt werden. Die nun fertige Schaltung wird mit Pads auf der obersten Metallebene versehen, um die äußeren Bondanschlüsse bereit zu stellen und der gesamte Chip wird in Plastikmaterial verpackt (engl. packaging). Dies dient sowohl dem Schutz vor der äußeren Umgebung als auch der Wärmeableitung und mechanischer Stabilität.

Da die Komplexität integrierter Schaltungen und auch deren Produktion zunimmt, sind Designer gezwungen, Designregeln, die durch die Produktion vorgegeben werden, einzuhalten. Dazu gehören z. B. bestimmte Breiten und Abstände zwischen den einzelnen Strukturen und Materialen. Davon sind vor allem Layoutingenieure betroffen. Ferner verursacht die fortschreitende Skalierung zahlreiche Herausforderungen für Prozessingenieure, die sich auch auf das Design auswirken können. Beispielsweise wird die Stepperbelichtung, die die gesamte Maske ablichtet, zunehmend von der Scannerbelichtung abgelöst, die Teile der Maske auf den Wafer belichtet. Außerdem stößt man zunehmend an die Grenzen der Optik, die mittels neuer Verfahren wie der Immersionslithografie oder der Optical Proximity Correction (OPC) weiter gedehnt werden. Dazu resultiert aus der Verkleinerung der Strukturen eine Annäherung der Leiterbahnen, was zu steigenden Kapazitäten und damit langsameren und störanfälligeren Leitungen führt. Aus diesem Grund werden so genannte low-k-Materialien als SiO_2-Ersatz untersucht und eingesetzt, da diese eine geringere Dielektrizitätskonstante haben und somit die Kapazitäten zwischen den Leitern senken. Das Gegenteil dazu sind high-k-Materialien, die als Ersatz für SiO_2 als Gateoxid gedacht sind, da die Skalierung zu einer immer geringeren Gateoxiddicke führt, welche anfällig für Tunneleffekte ist, die im nächsten Kapitel näher erläutert werden. Durch Einsatz der high-k-Materialien, die eine größere Permittivität haben als SiO_2, muss die Gateoxiddicke t_{OX} nicht so stark verringert werden. Allerdings entstehen dadurch neue Herausforderungen, wie z. B. eine erhöhte Schwellspannung oder verringerte Ladungsträgerbeweglichkeit.

2.1.2 Funktionsweise

Vereinfacht erscheint der MOSFET als Schalter, der entweder Strom leitet oder nicht. Allerdings wird der Stromfluss I_{DS} zwischen Drain und Source nicht nur „an“ und „aus“ geschaltet, sondern über die Steuerspannung V_{GS}, die zwischen Gate und Source anliegt, geregelt, da seine Funktionsweise auf dem Feldeffektprinzip beruht. Hierbei wird mittels eines elektrischen Feldes die Leitfähigkeit des Kanals unter dem Gate gesteuert, so dass der MOSFET wie ein spannungsgesteuerter Widerstand wirkt. Die Stromleitung erfolgt entweder durch Elektronen (n-MOSFET) oder durch Elektronenlöcher (p-MOSFET), je nachdem welche dieser Teilchen als Majoritätsladungsträger durch die Dotierung unipolarer Transistoren vorliegen. Die

Funktionsweise beider MOSFET-Typen ähnelt sich, weshalb bei der Erläuterung im Folgenden nur auf den n-MOSFET eingegangen wird. Eventuelle Unterschiede werden allerdings angemerkt.

Üblicherweise ist das p-dotierte Substrat beim n-MOSFET mit Masse verbunden. Ist nun die Spannung über dem Gate $V_{GS} < 0$, werden aufgrund der negativen Spannung zwischen Substrat und Gate die Elektronenlöcher zum Gate geleitet und zwischen Drain und Source ($I_{DS} = 0$) fließt kein Strom. Dies nennt man Akkumulation, da sich die Elektronenlöcher im Kanal zwischen Drain und Source ansammeln – Abbildung 2-5 a). Ist $V_{GS} > 0$, allerdings noch kleiner als die Schwellspannung V_{TH} (engl. threshold voltage) – $V_{GS} < V_{TH}$, werden die Löcher vom Gate abgestoßen und ins „Substratinnere“ zurückgedrängt, während die negativ geladenen Ionenrümpfe der Akzeptoren im Kanal „zurückbleiben“ – Abbildung 2-5 b). Dies wird Verarmung genannt (engl. depletion).

Steigt die Gate-Source-Spannung über die Schwellspannung des Transistors an, zieht die steigende Anzahl der positiven Ladungsträger eine steigende Anzahl an Elektronen zum Gate. Dieses Stadium wird Inversion genannt, da nun ein leitender Kanal von freien Elektronen zwischen Drain und Source unter dem Gate im Substrat (p-dotiert) entsteht, deren Majoritätsladungsträger die Elektronenlöcher sind. Die Schwellspannung V_{TH} ist vom Ausmaß der Substratdotierung und der Gateoxiddicke t_{OX} abhängig und ergibt sich bei einer Substratspannung $V_{SB} = 0$ aus der Differenz der Flachbandspannung V_{FB} und der Differenz Φ_{ms} der Austrittsarbeit des Gates (m) und des Substrates (s):

$$V_{TH0} = V_{FB} - \Phi_{ms} \propto \frac{t_{OX}}{W \cdot L \cdot \varepsilon_{OX}} \tag{1}$$

mit ε_{OX} als Permittivität von SiO_2. Die Flachbandspannung ist eine Spannung $V_{GS} > 0$, die die verbogenen Energiebänder des Siliziums in der Grenzschicht (Raumladungszone) zwischen Substrat und Gateoxid wieder begradigt, da durch das Anlegen eines äußeren Energiefeldes die Potentialverhältnisse am Rand des Substrates verändert werden, was auch als Bandkrümmung bezeichnet wird. Ist die Potentialdifferenz zwischen Source und Substrat $V_{SB} > 0$, vergrößert sich die Schwellspannung, da die Verarmungsschicht wächst (engl. body effect):.

$$V_{TH} = V_{TH0} + \gamma \cdot \left(\sqrt{V_{SB} + \varphi_S} - \sqrt{\varphi_S}\right) \; mit \; \gamma \propto \frac{t_{OX}}{\varepsilon_{OX}} = \frac{1}{C_{OX}} \tag{2}$$

mit φ_S als Oberflächenpotential des Substrates und γ als body-effect-Koeffizient und C_{OX} als Kapazität des Gateoxides je Fläche des Gateoxides.

Ist ein leitender Kanal entstanden, hängt der Stromfluss I_{DS} von der Spannung V_{DS} zwischen Drain und Source ab, die die notwendige Potentialdifferenz für I_{DS} bildet:

$$V_{DS} = V_{GS} - V_{GD} \tag{3}$$

Ist $V_{DS} = 0$ ($V_{GS} = V_{GD}$), bildet sich kein elektrisches Feld, das einen Stromfluss erzeugt – Abbildung 2-5 c). Sobald eine positive Potentialdifferenz zwischen Drain und Source

vorhanden ist, wird ein Strom I_{DS} durch den Kanal erzeugt. Dieser Modus wird auch als linear bzw. resistiv bezeichnet, da der Stromanstieg von I_{DS} linear zum Spannungsanstieg von V_{DS} verläuft und der Transistor sich somit wie ein Widerstand verhält – Abbildung 2-5 d). Die Ladung $Q_{CHANNEL}$ im Kanal ist abhängig von der Kapazität C_G zwischen Gate und Kanal und der Spannung ($V_{GS} - V_{TH}$), die ursächlich für die Ladung im Kanal ist:

$$Q_{CHANNEL} = C_G(V_{GS} - V_{TH}) \tag{4}$$

wobei C_G wiederum abhängig von der Transistorfläche und der Gateoxiddicke t_{OX} ist:

$$C_G = W \cdot L \cdot \frac{\varepsilon_{OX}}{t_{OX}} = W \cdot L \cdot C_{OX} \tag{5}$$

Der Stromfluss zwischen Drain und Source I_{DS} ist die Ladungsmenge $Q_{CHANNEL}$ je Zeit, die benötigt wird, um den Kanal zu durchqueren. Diese Zeit ist der Quotient aus der Länge L des Transistors und der durchschnittliche Geschwindigkeit v, die ein Ladungsträger braucht, um den Kanal, auf den ein durch V_{DS} verursachtes elektrisches Feld E wirkt, zu durchqueren:

$$v = \mu \cdot E = \mu \cdot \frac{V_{DS}}{L} \tag{6}$$

mit μ als Mobilität. Daraus kann I_{DS} berechnet werden:

$$I_{DS} = \frac{Q_{CHANNEL}}{\frac{L}{v}} = \beta \cdot (V_{GS} - V_{TH} - \frac{V_{DS}}{2}) \cdot V_{DS} \tag{7}$$

mit der Zusammenfassung der geometrischen und technologieabhängigen Faktoren:

$$\beta = \mu \cdot C_{OX} \cdot \frac{W}{L} \tag{8}$$

Ist nun V_{DS} groß genug, sinkt V_{GD} unter die Schwellspannung. Dies führt zu einer Aufhebung der Inversion nahe dem Draingebiet. Allerdings werden die Elektronen durch ihre Geschwindigkeit ins Draingebiet gestoßen und der Stromfluss I_{DS} bleibt erhalten. Der Transistor ist nun im gesättigten Zustand und wirkt wie eine Stromquelle, deren Stromfluss I_{DS} nun nur noch durch V_{GS} gesteuert wird – Abbildung 2-5 e):

$$I_{DS} = \frac{\beta}{2} \cdot (V_{GS} - V_{TH})^2 \tag{9}$$

Oft wird auch der Begriff des gesättigten Stromflusses I_{DSAT} verwendet mit $V_{GS} = V_{DS} = V_{DD}$:

$$I_{DSAT} = \frac{\beta}{2} \cdot (V_{DD} - V_{TH})^2 \tag{10}$$

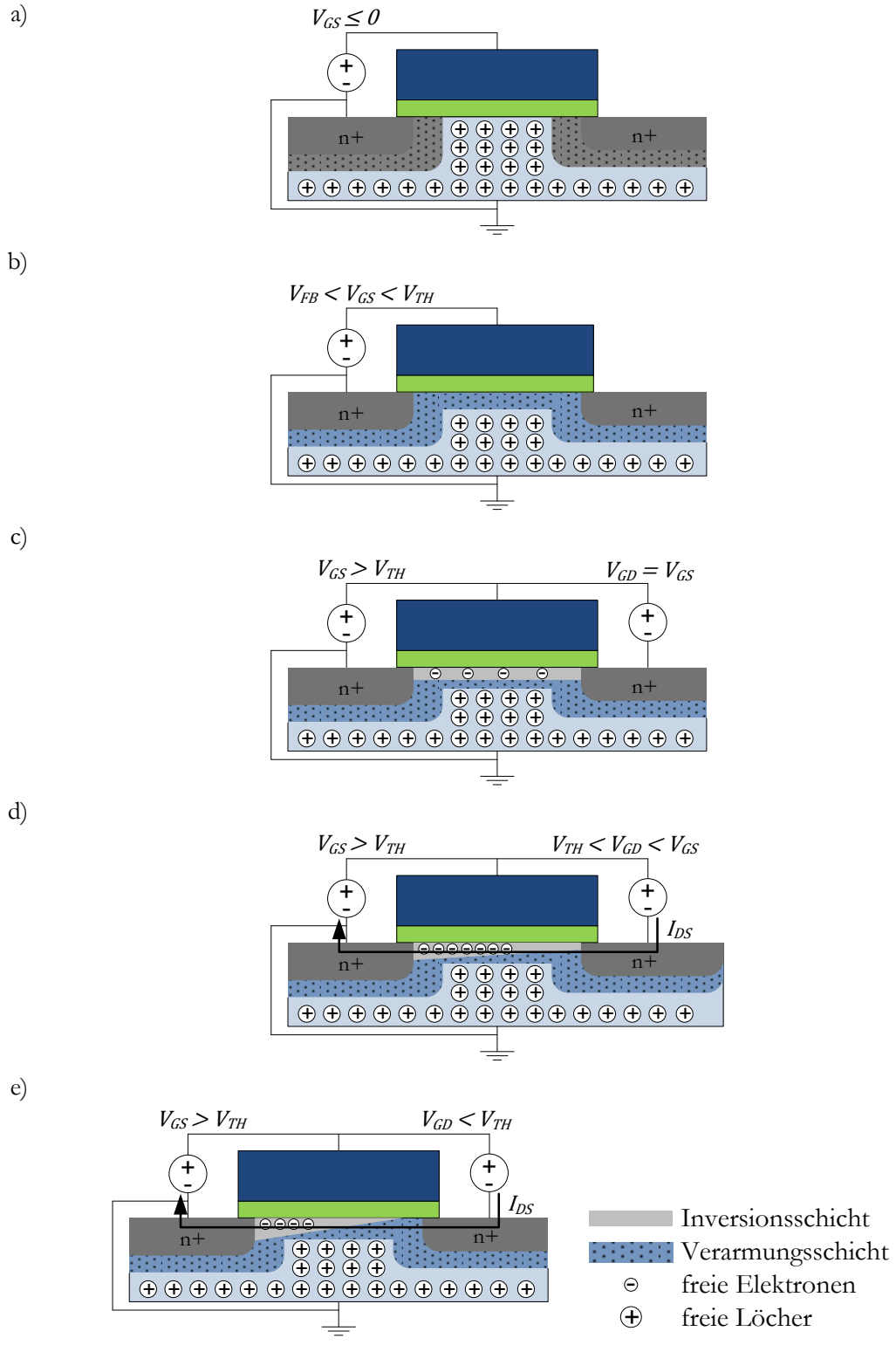


Abbildung 2-5 : Operationsmodi eines n-MOSFET:

a) Akkumulation: Kein Kanal, $I_{DS} = 0$
b) Bildung einer Verarmungsregion (Verdrängung der Fehlstellen) unter dem Gateoxid
c) Linearmodus: Bildung eines leitenden Kanals (Inversion) unter dem Gateoxid, $I_{DS} = 0$
d) Linearmodus: Erhöhung des Stromes I_{DS} durch Senkung der Spannung V_{GD}
e) Sättigung: Strom I_{DS} ist nur noch abhängig von V_{GS}

Abbildung 2-6 a) zeigt die Strom-Spannungskurve für den Drainstrom eines idealen n-MOSFET, wie er bisher hergeleitet wurde. Mit einer Gatespannung $V_{GS} < V_{TH}$ fließt kein Strom. Für höhere Spannungen steigt der Strom linear mit V_{DS} bis die Drain-Source-Spannung den Sättigungspunkt erreicht und I_{DS} unabhängig von V_{DS} wird:

$$V_{DS} = V_{GS} - V_{TH} \tag{11}$$

Die Funktionsweise des p-MOSFET ist analog zu dem des n-MOSFET. Allerdings wird das Substrat mit der Versorgungsspannung V_{DD} verbunden und der Stromfluss I_{DS} wird durch den Drift von Elektronenlöchern verursacht. Weil die Mobilität dieser Ladungsträger geringer ist als die der Elektronen, ist der Stromfluss I_{DS} kleiner als bei einem gleichgroßen n-MOSFET. Da das Substrat eines p-MOSFET üblicherweise mit der Versorgungsspannung verbunden ist, ist die Strom-Spannungskurve im dritten Quadranten dargestellt – Abbildung 2-6 b).

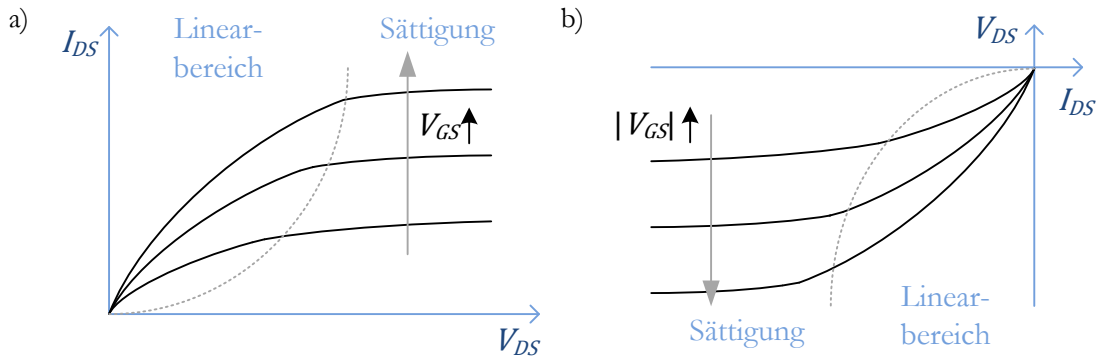


Abbildung 2-6 : Charakteristische I-V-Kurve
a) eines idealen n-MOSFET
b) eines idealen p-MOSFET

2.1.3 Kurzkanaleffekte

Aufgrund der immer kleineren Gatelängen treten ungewollte Kurzkanaleffekte auf, die in aktuellen Nanometer-Technologien nicht mehr vernachlässigt werden können. So kommt es bei hohen elektrischen Feldern, wie in Nanometer-Technologien üblich, zur Verringerung der Mobilität μ der Ladungsträger aufgrund deren großen Streuung (engl. mobility degradation). Dies führt zu einer Verringerung der Geschwindigkeit v der Ladungsträger (engl. velocity saturation), und somit auch zu einer Senkung des Stromflusses I_{DS}, vor allem im gesättigten Zustand des Transistors. Ab einer bestimmten Feldstärke bleibt die Geschwindigkeit konstant ($v = v_{SAT}$), was zu einer linearen Abhängigkeit des Drainstromes von der Drainspannung führt, im Gegensatz zur quadratischen Abhängigkeit aus Gleichung (9):

$$I_{DS} = C_{OX} \cdot W \cdot (V_{GS} - V_{TH}) \cdot v_{SAT} \tag{12}$$

Sollte die Versorgungsspannung nicht die kritische Feldstärke hervorrufen, werden Werte für I_{DS} erreicht, die zwischen beiden Extremen liegen. Das „alpha power law"-Modell von [Sak90] beschreibt den Strom I_{DS} in Abhängigkeit des velocity saturation index a_{VEL} mit folgenden Gleichungen:

$$I_{DS} = \begin{cases} 0 & \text{für } V_{GS} \leq V_{TH} \\ \frac{W}{L_{EFF}} \cdot \frac{P_C}{P_V} \cdot V_{DS} \cdot (V_{GS} - V_{TH})^{\frac{\alpha_{VEL}}{2}} & \text{für } V_{DS} \leq V_{DSAT} \\ \frac{W}{L_{EFF}} \cdot P_C \cdot (V_{GS} - V_{TH})^{\alpha_{VEL}} & \text{für } V_{DS} \leq V_{DSAT} \end{cases} \tag{13}$$

mit PC und PV als Technologieparameter und mit:

$$V_{DSAT} = P_V \cdot (V_{GS} - V_{TH})^{\frac{\alpha_{VEL}}{2}} \tag{14}$$

Ferner ist nicht mehr die geometrische Länge L für den Stromfluss verantwortlich, sondern die effektive Länge L_{EFF}, weil der Kanal durch die Verarmungsregion mit der Länge L_D zwischen Drain und Substrat verkleinert wird:

$$L_{EFF} = L - L_D \tag{15}$$

Aus diesen Überlegungen ergibt sich auch, dass es ab einer bestimmten Versorgungsspannung keinen Sinn mehr macht, sie weiter zu erhöhen, um I_{DS} zu erhöhen, da die Geschwindigkeit v_{SAT} nicht überschritten werden kann.

Mit sinkender Gatelänge steigt auch der Einfluss der Verarmungsregionen unter Source und Drain auf die Ladungsträger, während der Einfluss des Gates abnimmt. Effektiv sinkt die Schwellspannung mit kleinerem L. Dazu steigt auch der Einfluss der Spannung V_{DS} auf die Schwellspannung, da bei sinkendem L die Potentialbarriere, die für die Ladungsträger zu überwinden ist, um in den Kanal zu gelangen, geringer wird je größer V_{DS} ist – Abbildung 2-7 a). Dieser Effekt (engl. drain induced barrier lowering – DIBL) sorgt auch für eine sinkende Schwellspannung, da der DIBL-Koeffizient η im Bereich von 0.02 bis 0.1 liegt:

$$V_{TH\_DIBL} = V_{TH} - \eta \cdot V_{DS} \tag{16}$$

Überschreitet V_{DS} einen bestimmten Wert, kann es passieren, dass sich die Verarmungsregionen von Source und Drain berühren und so einen Stromfluss I_{PT} verursachen. Dieser wird punchthrough genannt und ist ein unerwünschten Leckstrom, der vor allem von V_{DS} und der effektiven Gatelänge L_{EFF} abhängig ist – Abbildung 2-7 b).

Weitere Leckströme resultieren aus Diffusionseffekten, bei denen Ladungsträger aufgrund von Konzentrationsunterschieden bewegt werden. Der Subthreshold Leakage ist ein Stromfluss zwischen Drain und Source, der auch bei einer Gatespannung $V_{GS} < V_{TH}$ zu verzeichnen ist. Eine Spannung $V_{DS} > 0$ verursacht einen Diffusionsstrom von Ladungsträgern, da sich bei

Transistoren mit geringen Gatelängen $L < 2$ nm eine schwache Inversionsschicht unter dem Gateoxid bildet, welcher exponentiell von der Schwellspannung abhängig ist:

$$I_{SUB} \propto \frac{W}{L_{EFF}} \cdot e^{(V_{GS} - V_{TH})} \tag{17}$$

Je geringer die Schwellspannung und die effektive Gatelänge, desto größer wird der Subthreshold Leakage. Des Weiteren führt auch ein Anstieg der Temperatur zu einem steigenden Subthreshold Leakage. Ein weiterer Diffusionsstrom entsteht durch die in Sperrrichtung geschaltete Diode, die sich zwischen Substrat und den Drain- und Sourcegebieten bildet. Dieser p-n leakage I_{PN} ist abhängig von Temperatur, Dotierung und Fläche des Transistors.

Das band-to-band leakage I_{BTB} und der Gateoxidleckstrom I_{GATE} sind weitere Leckstromarten, die auf den Tunneleffekt zurückzuführen sind [Tau98]. Als Tunneleffekt bezeichnet man das Überwinden einer Potentialbarriere von quantenmechanischen Teilchen, wie z. B. Elektronen, obwohl die Barriere ein höheres Energieniveau aufweist als das Teilchen. Dies kann mit dem Welle-Teilchen-Dualismus der Elektronen erklärt werden. Dabei wird beim Auftreffen auf eine Potentialbarriere nur ein Teil der Wellenfunktion des Elektrons reflektiert. Der verbliebene Anteil durchdringt die Barriere, dessen Energie aber exponentiell abnimmt. Ist die Barriere allerdings schmal genug, kann die kinetische Energie des Elektrons ausreichen, um die Wellenfunktion des Elektrons beim Austritt zu erhalten, so dass das Elektron die Barriere durchdringt.

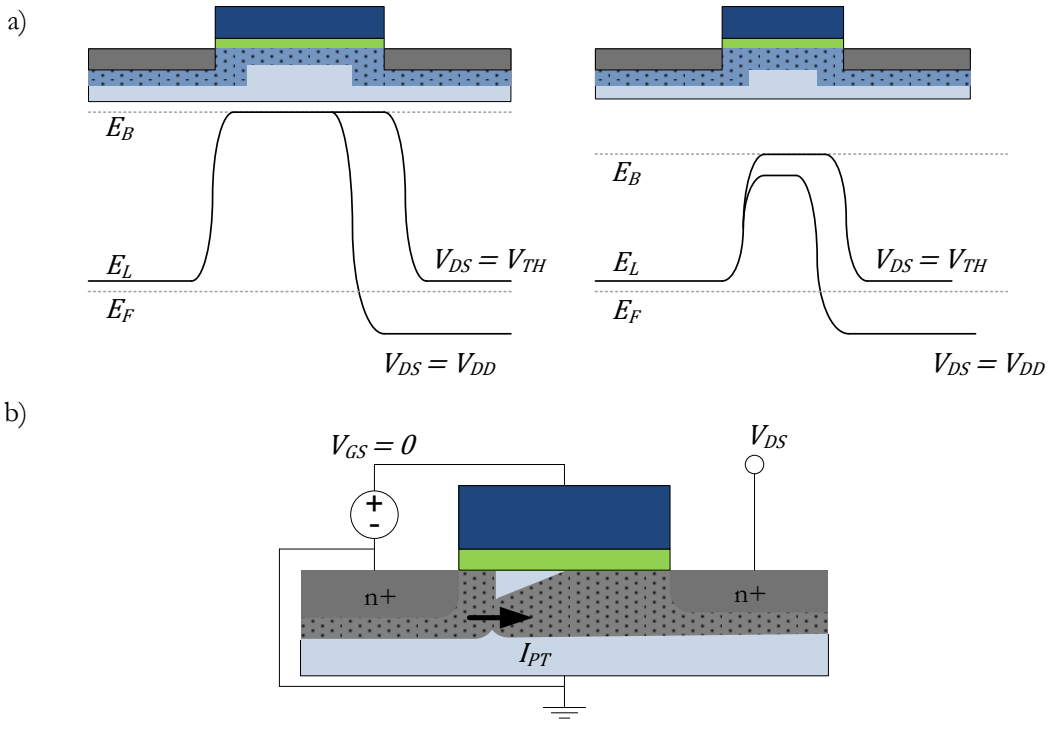


Abbildung 2-7 : Kurzkanaleffekte

a) DIBL: Verringerung der Potentialbarriere E_B, wodurch Elektronen leichter in den Kanal eintreten können (E_F – Fermienergie; E_L – Energie des unteren Leitungsbandes)

b) Punchthrough: Stromfluss I_{PT} durch eine Verbindung der Verarmungsregionen von Drain und Source

Beim band-to-band tunneling werden Elektronen durch hochdotierte in Sperrrichtung geschaltete p-n-Übergänge in beide Richtungen zwischen den n- und p-dotierten Gebieten, wie beispielsweise zwischen Substrat und Drain/Source-Gebieten, getunnelt. Dazu bedarf es einer ausreichend hoher Feldstärke über dem p-n-Übergang und dass das Valenzband des p-dotierten Gebietes höher ist als das n-dotierte Leitungsband.

Das gate oxide tunneling umfasst den Transport von Elektronen und Elektronenlöcher zwischen Gate und Substrat durch das Gateoxid. Es lässt sich in zwei Arten unterteilen. Beim Fowler-Nordheim tunneling gelangen Elektronen in das Leitungsband des Gateoxides, sobald die Gatespannung V_{GS} größer ist als die Austrittsarbeit Φ_{OX} vom Substrat zum Oxid [DiM97]. Die Stromdichte J_{FN} der Elektronen kann folgendermaßen approximiert werden [Len69], wobei $B_{FN} \approx 265\ \mathrm{MV/cm}$ eine Konstante darstellt:

$$J_{FN} \propto {E_{OX}}^2 \cdot e^{-\frac{B_{FN}}{E_{OX}}} \tag{18}$$

Es ist ersichtlich, dass der Tunnelstrom nur vom elektrischen Feld E_{OX} über dem Gateoxid abhängig ist. Bei Transistoren mit höheren Gateoxiddicken kann so ein konstanter Tunnelstrom erzeugt werden, um das Gateoxid für Zuverlässigkeitsanalysen und für das Burn-In zu stressen. Bei neueren CMOS-Technologien hingegen dominiert das direct tunneling, da V_{GS} und t_{OX} kleiner sind, und $V_{GS} < \Phi_{OX}$ dafür eine Vorrausetzung ist. Hierbei tunneln die Elektronen direkt durch das Gateoxid, ohne das Leitungsband des Gateoxides als Zwischenstation zu nutzen [Tau98]:

$$J_{FN} \propto {E_{OX}}^2 \cdot e^{-\frac{B_{FN}}{E_{OX}}\cdot\left[1-\left(1-\frac{V_{OX}}{\Phi_{OX}}\right)^{3/2}\right]} \tag{19}$$

2.2 CMOS-Gatter

Logikgatter stellen Operatoren der booleschen Algebra dar, die auf zwei Elementen beruhen. Diese zwei Elemente sind die 0 und die 1 des Binärsystems. Diese beiden Werte werden in CMOS mit Hilfe von Spannungen dargestellt. Die 0 entspricht der Masse V_{SS}, während 1 dem Wert der Versorgungsspannung V_{DD} entsprechen sollte. Innerhalb der Gatter sind Transistoren folgendermaßen aufgebaut. P-MOSFETs verbinden den Ausgang des Gatters mit der Versorgungsspannung (Pull-Up-Netzwerk), n-MOSFETs mit Masse (Pull-Down-Netzwerk). Die Transistoren sind dabei untereinander so verschaltet und mit den Eingängen verbunden, dass der gewünschten Operator gebildet wird. Aus diesem Grund wird diese Schaltungsfamilie als Complementary-MOS bezeichnet, da das Pull-Up-Netzwerk komplementär zum Pull-Down-Netzwerk aufgebaut ist – Abbildung 2-8 a). Dadurch ist immer nur eine der beiden Versorgungsspannungen mit dem Ausgang verbunden und ermöglicht somit einen definierten Spannungspegel, der entweder V_{SS} oder V_{DD} entspricht. Die folgenden Erklärungen beziehen sich auf das einfachste Gatter – das Nicht-Gatter (engl. Inverter), deren Pull-Up- und Pull-Down-

Netzwerk aus jeweils einem p- bzw. n-MOSFET besteht – Abbildung 2-8 b). Weitere Gatterarten sind das Bufferelement und folgende Gatter mit zwei Eingängen: das Und (AND), das Oder (OR), das Nicht-Und (NAND), das Nicht-Oder (NOR), das Entweder-Oder (XOR) und das Nicht-Entweder-Oder (XNOR). Diese Gatter sind auch mit mehr als zwei Eingängen implementierbar. Sequentielle Elemente sind Latches und Flipflops. In dieser Arbeit wird nur das D-Flipflop verwendet, das das Eingangssignal an D mit jedem Taktwechsel des CLK-Signals von V_{SS} auf V_{DD} an den Ausgang Q weiterreicht.

Der Inverter ist deshalb das Nicht-Gatter, da es eine 1 am Eingang zu einer 0 transformiert und umgekehrt – Abbildung 2-8 c). Sollte der Eingang mit der positiven Versorgungsspannung verbunden sein, leitet der n-MOSFET des Inverters den Strom vom Ausgangsknoten zur Masse, um so den Ausgangsknoten zu entladen, während der p-MOSFET nicht durchschaltet. Dadurch wird der Ausgang des Inverters mit Masse verbunden, also ein Spannungswert von $V_{OUT} = V_{SS}$ erzeugt. Soll nun ein hoher Spannungspegel am Ausgang erzeugt werden (eine 1), muss der Eingang mit Masse verbunden sein, da nun der n-MOSFET sperrt, während der p-MOSFET den positiven Versorgungsspannungsknoten mit dem Ausgang verbindet ($V_{OUT} = V_{DD}$). Folgt einem hohen Spannungspegel am Eingang nach einer bestimmten Zeitspanne ein niedriger oder umgekehrt, so tritt die Veränderung des Ausgangspegels nicht unmittelbar nach der Eingangsveränderung auf, da die Transistoren Kapazitäten ausbilden, die sich erst ent-/aufladen müssen. Diese Kapazitäten bilden sich zwischen Gate und den anderen Bestandteilen des Transistors (Substrat, Drain, Source) aufgrund des Aufbaus und durch die verschiedenen Arbeitsbereiche, bei dem sich die Ladungszustände im Transistor verändern und damit auch die Kapazitätswerte. Im Allgemeinen werden die Kapazitätswerte zu einer Lastkapazität C_{LOAD} zusammengefasst, wenn sie nicht zu dem jeweiligen Gatter gehören. Ausgehend vom Gate

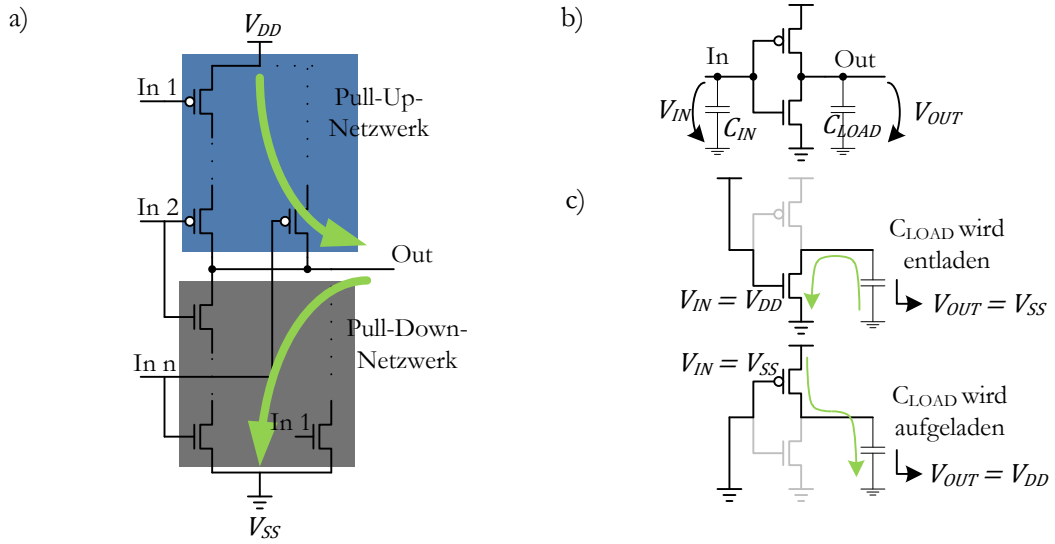


Abbildung 2-8 : Aufbau der CMOS-Gatter

a) CMOS-Funktionsprinzip durch Pull-Up-Netzwerk, bestehend aus p-MOSFET, und Pull-Down-Netzwerk, bestehend aus n-MOSFET
b) Wichtige Parameter eines CMOS-Gatters anhand des Inverters
c) Funktionsweise des CMOS-Inverters

werden die zusammengefassten Werte der eigenen Transistoren, die ein Teil der Last der vorherigen Gatter ausmacht, als Eingangskapazität C_{IN} benannt. Das Fanout eines Gatters ist das Verhältnis von C_{LOAD}/C_{IN}. Das Entladen (fallende Ausgangsflanke) bzw. Aufladen (steigende Ausgangsflanke) der Lastkapazität, die sich am Ausgangsknoten befindet, erfordert eine endliche Zeitspanne. Diese Verzögerungszeit t_D (engl. delay) ist abhängig von inneren und fremden Faktoren. Die inneren Faktoren umfassen die Parameter der Transistoren, die den Stromfluss, der C_{LOAD} am Ausgangsknoten auf- oder entlädt, direkt beeinflussen. Wichtig hierbei sind die Gatebreiten, die Gateoxiddicken, die Dotierung des Substrates, die Temperatur und die Versorgungsspannungen. Andere Faktoren werden deshalb als fremd bezeichnet, da sie nur mittelbar auf den Stromfluss der Transistoren der Gatter Einfluss nehmen. Dazu gehören C_{LOAD} und die Zeitspanne, die notwendig ist, um C_{IN} aufzuladen (Anstiegzeit t_{RISE}) bzw. zu entladen (Abfallzeit t_{FALL}). Diese charakteristischen Zeitkriterien werden folgendermaßen definiert:

Tabelle 2-1 : Zeitkriterien der CMOS-Gatter (Abbildung 2-9)

Zeitkriterium	Abk.	Definition
Verzögerungszeit	t_D	Zeit, zwischen der das Eingangs- und das Ausgangssignal eines Gatters die Spannungslevel $V_{DD}/2$ über- bzw. unterschreiten
Eingangsanstiegzeit	t_{RISE_IN}	Zeitspanne, vom Zeitpunkt an, an dem das Eingangssignal eine untere Grenze (V_{LOW}) überschreitet bis zu dem Zeitpunkt, an dem der Pegel eine obere Grenze (V_{HIGH}) erreicht
Eingangsabfallzeit	t_{FALL_IN}	Zeitspanne, vom Zeitpunkt an, an dem das Eingangssignal eine obere Grenze (V_{HIGH}) unterschreitet bis zu dem Zeitpunkt, an dem der Pegel eine untere Grenze (V_{LOW}) erreicht
Ausgangsanstiegzeit	t_{RISE_OUT}	Zeitspanne, vom Zeitpunkt an, an dem das Ausgangssignal eine untere Grenze (V_{LOW}) überschreitet bis zu dem Zeitpunkt, an dem der Pegel eine obere Grenze (V_{HIGH}) erreicht
Ausgangsabfallzeit	t_{FALL_OUT}	Zeitspanne, vom Zeitpunkt an, an dem das Ausgangssignal eine obere Grenze (V_{HIGH}) unterschreitet bis zu dem Zeitpunkt, an dem der Pegel eine untere Grenze (V_{LOW}) erreicht

Abbildung 2-9 zeigt diese fünf Kriterien anhand der Signalübergänge eines Inverters, wobei die Werte von V_{LOW} üblicherweise bei 10 % bzw. 20 % von V_{DD} liegen, während 80 % bzw. 90 % des V_{DD}-Wertes als V_{HIGH} definiert wird.

Die Verzögerungszeit eines Gatters kann durch folgende Gleichung approximiert werden:

$$t_D = \frac{t_{OX} \cdot L_{EFF} \cdot C_{LOAD}}{\square_0 \cdot \varepsilon_{OX} \cdot W \cdot (V_{DD} - V_{TH})^{\alpha}} \tag{20}$$

Eine geringe Verzögerungszeit resultiert aus einer großen Transistorbreite, einer geringen Schwellspannung und einer hohen Versorgungsspannung, während ein dickes Gateoxid und hohe Ausgangskapazitäten die Verzögerungszeit ansteigen lassen. Bei Gattern, mit mehreren Eingängen, werden diese Charakteristika für jeden Eingang gesondert aufgenommen, während die anderen so beschaltet sind, dass der jeweilige Eingang durch seine Pegeländerung eine Ausgangspegeländerung hervorruft (engl. Single-Input-Switching).

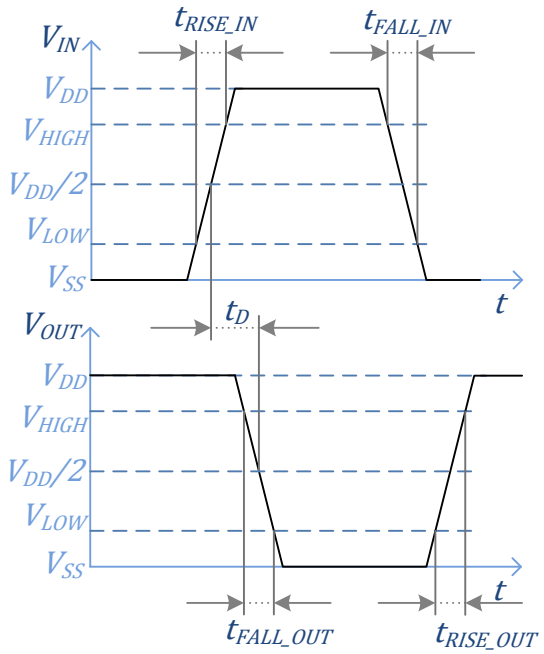


Abbildung 2-9 : Zeitkriterien eines CMOS-Gatters

Besondere Beachtung muss auch der Dimensionierung der Transistoren eines Gatters zukommen, und damit auch der gesamten Schaltung. Dies umfasst die Abstimmung der Gatebreiten aller beteiligten Transistoren. Da p-MOSFET und n-MOSFET unterschiedliche Mobilitätswerte besitzen, müssen beispielsweise die p-MOSFET größer sein als die n-MOSFET, damit die resultierenden Widerstände ungefähr gleich sind und damit auch die Verzögerungszeit bei fallender und steigender Flanke. Allerdings steigt damit auch die Eingangskapazität für vorige Gatter. Ein Verfahren, um eine möglichst kleine Verzögerungszeit für eine Schaltung zu erreichen, ist der „logical effort"-Ansatz [Sut99], wobei die Gatter mit einem Inverter gleicher Treiberstärke verglichen werden und daraus der logical effort berechnet werden kann. Durch Veränderung des Fanouts jedes Gatters mittels Anpassung der Gatebreiten der jeweiligen Transistoren wird der Verstärkungsfaktor für jedes Gatter solange angepasst, dass dieser für jedes Gatter möglichst gleich ist. Das führt dann zur minimalen Verzögerungszeit der Schaltung. Bei dem Aufbau größerer Schaltungen aus einer RTL-Beschreibung wird ähnlich verfahren, wenn die Verzögerungszeit der Schaltung minimal ausfallen soll, was im nächsten Unterkapitel deutlich wird.

2.3 Die Gattersynthese

Die Überführung der RTL-Beschreibung in eine Beschreibung, die einzelne Gatter zu einer Netzliste zusammenfügt, die dem RTL-Code entspricht, wird Synthese genannt. Hierzu ist eine Gatterbibliothek notwendig, die alle verfügbaren Gatter als so genannte Standardzellen beinhaltet. Diese Standardzellen sind für jede Technologie schon als Layout vorhanden. Daraus ergeben sich

mehrere Vorteile. Zum einen sind die Zellhöhe, die Masse- und Versorgungsspannungsanschlüsse für alle Zellen gleich, so dass das Platzierungstool sie neben- und übereinander anordnen kann ohne Verbindungsprobleme für die Masse- und Versorgungsspannungsleitungen. Weiterhin sind durch das fertige Layout verschiedene Dateiformate vorhanden, um diverse Verifikationstools des gesamten Layouts bedienen zu können. Beispiele hierfür sind der Design-Rule-Check (DRC), der Layout-vs.-Schematic-Check (LVS) und der Electrical-Rule-Check (ERC) und andere Tests, die korrekte Produktionsmasken und das intendierte Verhalten der Schaltung garantieren sollen. Zum anderen sind durch das Layout relevante Daten, die für die genaue Analyse des Zeitverhaltens und der Stromaufnahme notwendig sind, verfügbar. Aus dem Layout können beispielsweise die Stromflüsse und die Kapazitäten extrahiert werden, sowohl die gewollten als auch die parasitären. Daher sind genaue Spice-Modelle für die Transistorebene Bestandteile heutiger Standardzellbibliotheken. Des Weiteren sind Bibliotheken für die Gatterebene vorhanden, die sowohl funktionale Parameter als auch Parameter für das Zeitverhalten, die Stromaufnahme, die Störanfälligkeit (engl. noise) und den Flächenbedarf enthalten. Diese Parameter sind für bestimmte Prozessbedingungen und Umgebungsparameter verfügbar und ermöglichen so die Synthese einer RTL-Beschreibung zu einer Gatternetzliste als auch die Analyse dieser Netzliste für veränderte Bedingungen und manuelle Änderungen seitens des Designers.

Der standardmäßige Ablauf dieses Prozesses ist in Abbildung 2-10 dargestellt. In der Abbildung, die eine zusammenfassende Übersicht über den Syntheseflow gibt, sind die einzelnen Stationen als Rechtecke dargestellt, Dateien als Ellipsen. Als erstes wird die RTL-Beschreibung analysiert und in generische Zellen aufgeteilt (engl. elaboration). Dieser Schritt ist technologie- und damit bibliotheksunabhängig und dient der Verifikation des vorhandenen RTL-Designs, ob alle beschriebenen Elemente als Gatterkonstrukte darstellbar sind („synthesefähig“). Sollte dies

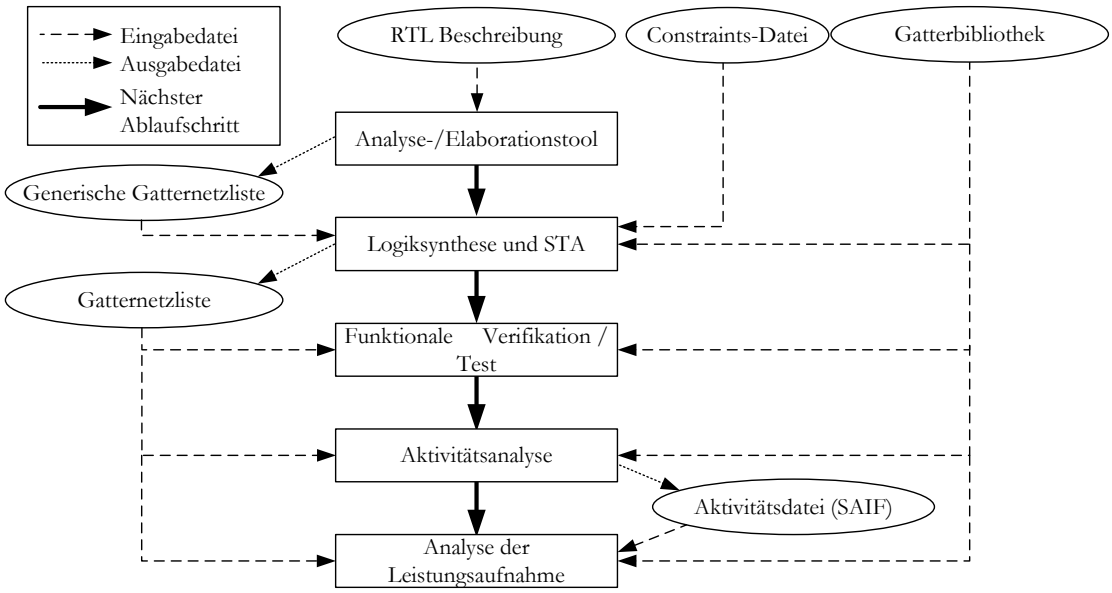


Abbildung 2-10 : Gattersyntheseablauf

der Fall sein, werden die Gatter der generischen Netzliste in vorhandene Standardzellen überführt. Dabei werden wichtige globale Parameter (engl. constraints), wie z. B. Grenzen hinsichtlich der Stromaufnahme, der Verzögerungszeit, der Fläche oder auch der Einsatz von Scan-Test-Gattern, und auch spezielle Limitierungen für bestimmte Bereiche der Schaltung berücksichtigt. Während der Erstellung der Netzliste wird eine Analyse des statischen Zeitverhaltens durchgeführt, damit eventuelle Zeitvorgaben unmittelbar die Netzlistenerstellung beeinflussen. Danach wird die Netzliste funktionell auf Basis der Funktionsbeschreibungen der Standardzellen verifiziert. Da hierbei die einzelnen Knoten der Gatternetzliste auf ihre logischen Pegel analysiert werden, um das korrekte Verhalten der gesamten Schaltung auf diverse Eingangssignale zu testen, ist anschließend eine Aktivitätsdatei vorhanden, die den Anteil der beiden Pegelzustände jedes Knotens an der simulierten Betriebsdauer enthält. Dadurch ist eine genaue Analyse der dynamischen und statischen Stromaufnahme im Anschluss möglich.

2.3.1 Analyse des statischen Zeitverhaltens

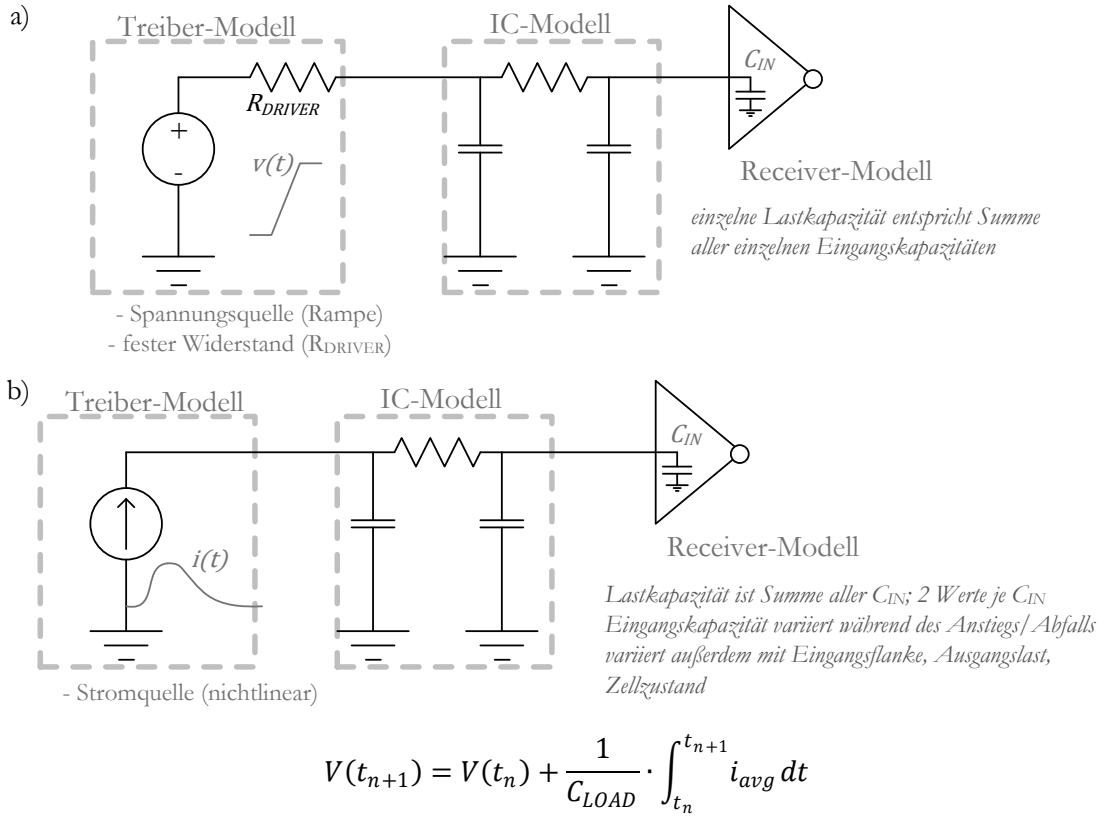
Bei der statischen Zeitanalyse (engl. static timing analysis – STA) werden die einzelnen Verzögerungszeiten der Gatter aller möglichen Pfade durch das Design summiert. Der Pfad mit der größten Zeitspanne ist der zeitkritischste (engl. critical path) und gibt die maximal mögliche Arbeitsfrequenz einer taktsynchronen Schaltung vor. Da Spice-Simulationen zu aufwändig für den Designprozess sind, werden für die STA einfachere und schneller zu berechnende Delaymodelle verwendet. Ein weit verbreiteter Ansatz, das statische Zeitverhalten auf Gatterebene zu analysieren, ist das Non-Linear-Delay-Model (NLDM), das die Verzögerungs-, Eingangsanstieg- und -abfallzeiten der einzelnen Gattertypen in Lookuptabellen abspeichert, deren Eingangswerte Lastkapazitäten und Eingangsanstiegs- bzw. -abfallzeiten darstellen. Dies bedeutet, dass für eine bestimmte Eingangsflanke (t_{RISE_IN} oder t_{FALL_IN}) und Lastkapazität C_{LOAD} ein bestimmter Wert für die Verzögerungszeit t_D und die Ausgangsflanke (t_{RISE_OUT} oder t_{FALL_OUT} je nach Funktion des Gatters) ausgegeben wird. Eventuelle Zwischenwerte werden interpoliert. Der Weg, den das Signal für eine Ausgangspegeländerung benötigt, wird in drei Abschnitte aufgeteilt – Abbildung 2-11 a). Am Anfang steht das Treibermodell, welches eine Spannungsquelle darstellt, deren Verhalten durch die Eingangsflanke bestimmt wird. Dann folgt das Leitermodell (engl. interconnect (IC) model), das den Übertragungsweg vom Gatter zum nächsten repräsentiert. Hierbei sind zumeist RC-Glieder in unterschiedlichen Ausführungen für längere Leitungen gebräuchlich. Sehr gute Approximationen werden mit dem T- bzw. dem π-Modell erreicht. Den Übertragungsweg schließt dann das Receivermodell ab, das im Fall des NLDM eine Lastkapazität als Summe aller Eingangskapazitäten des betreffenden Netzes darstellt. So kann für jedes Gatter eines Pfades nach und nach aufgrund der Eingangsflanke und der Lastkapazität die Verzögerungszeit und die Eingangsflanke für das folgende Gatter berechnet werden. Durch Summation der Verzögerungszeiten der Zellen aller Pfade des gesamten Designs kann für beide möglichen Flanken der kritische Pfad ermittelt werden.

Ein zweites Modell speichert die vorhandenen Daten ebenfalls in Lookuptabellen. Allerdings sind die Tabellenausgabewerte nicht Verzögerungs- bzw. Transitionszeiten, sondern stellen

Stromkurven dar. Dieses Modell wird deshalb Current-Source-Modell (CSM) genannt, da das Treibermodell nicht mehr einer Spannungsquelle, sondern einer nichtlinearen Stromquelle i_{OUT} entspricht. Paare aus Strom- und Zeitwerten stellen diese Stromquelle dar, woraus dann der Spannungsverlauf des Knotens berechnet werden kann. Da in dieser Arbeit das CSM von Synopsys™ (Composite Current Source – CCS) verwendet wurde, wird als Receivermodell eine Kombination von zwei Kapazitätswerten verwendet, die ab Erreichen eines Signalpegels von $V_{DD}/2$ wechseln, um sich verändernde Eingangskapazitätswerte während des Schaltvorganges zu berücksichtigen – Abbildung 2-11 b). Durch Integration der Stromkurve i_{OUT} kann dann der Spannungspegelverlauf des Ausgangssignals berechnet werden:

$$dv = \frac{dt \cdot i_{OUT}}{C_{LOAD}} \tag{21}$$

Hieraus können dann leicht die Verzögerungszeit und die Flankenzeiten berechnet werden. Der Vorteil dieser Variante, die jedoch mehr Speicherplatz benötigt, ist eine flexiblere Handhabung von nichtlinearen Versorgungsspannung- und Temperaturskalierungen, was auch komplexere Analysen ermöglicht. Hierbei wird die Schaltung für mehrere (extreme) Prozess- und Umgebungsbedingungen analysiert.



$$V(t_{n+1}) = V(t_n) + \frac{1}{C_{LOAD}} \cdot \int_{t_n}^{t_{n+1}} i_{avg}\, dt$$

Abbildung 2-11 : Verzögerungszeitmodelle auf Gatterebene
a) Non-Linear-Delay-Model
b) Composite Current Source Modell

2.3.2 Analyse der Leistungsaufnahme

Die Leistungsaufnahme einer Schaltung kann in drei Komponenten unterteilt werden. Zwei davon entstehen durch Umschaltvorgänge in CMOS-Gattern und damit nur, wenn das Gatter „aktiv" ist. Dies sind die dynamische Leistungsaufnahme P_{DYN} und die Leistungsaufnahme durch Kurzschlussströme P_{SC}. Die dritte Komponente umfasst die Zusammenfassung aller Leckströme (engl. leakage), die in einem Transistor und damit in einem CMOS-Gatter auftreten können. Sie verursachen ständig eine statische Leistungsaufnahme P_{STAT}. Die wichtigsten Leckstromarten sind Subthreshold Leakage I_{SUB}, p-n-Leckstrom I_{PN}, Punchthrough Leakage I_{PT}, band-to-band leakage I_{BTB} und Gateoxidleckstrom I_{GATE}, deren Ursachen im Unterkapitel 2.1.3 erläutert wurden:

$$P_{STAT} = V_{DD} \cdot (I_{SUB} + I_{PN} + I_{BTB} + I_{GATE} + I_{PT}) \tag{22}$$

Im „aktiven" Zustand eines CMOS-Gatters wird für das Aufladen der Lastkapazität Energie benötigt. Bei einem 0-zu-1-Übergang am Ausgang wird die Lastkapazität durch einen Strom i_{OUT} von der Versorgungsspannung zum Ausgangsknoten über das Pull-Up-Netzwerk aufgeladen. Bei einem anschließenden 1-zu-0-Übergang führt ein Strom vom Ausgang durch das Pull-Down-Netzwerk zu einer Entladung der Lastkapazität zur Masse. Die Energie, die für diesen Vorgang notwendig ist, ergibt sich aus der Integration des Stromes über die Zeit für beide Vorgänge, die auf dieselbe Art berechnet werden:

$$\begin{aligned} E_{DYN} &= E_{0\to1} = \int_0^t V_{OUT} \cdot i_{OUT}\, dt \\ E_{DYN} &= \int_0^t V_{OUT} \cdot C_{LOAD} \cdot \frac{dV_{OUT}}{dt} dt = \frac{1}{2} \cdot C_{LOAD} \cdot {V_{DD}}^2 \end{aligned} \tag{23}$$

Daraus lässt sich die Leistungsaufnahme berechnen, da die Energie bei Zustandsänderungen des Ausgangsknotens aufgewandt wird, deren Häufigkeit mit Hilfe der Taktfrequenz f_{CLK} und der Wahrscheinlichkeit a der 0-zu-1-Übergänge am Gatterausgang angegeben werden kann:

$$P_{DYN} = f_{CLK} \cdot \alpha \cdot C_{LOAD} \cdot {V_{DD}}^2 \tag{24}$$

Kurzschlussströme wiederum entstehen während des Umschaltens des Gatterausganges, wenn eine leitende Verbindung zwischen Versorgungsspannungsleitung und Masse entsteht. Dies resultiert aus der kurzen Zeitspanne, in der sowohl die Transistoren des Pull-Up- als auch des Pull-Down-Netzwerkes leitend sind, und führt zu folgendem Leistungsverbrauch P_{SC} bei einem durchschnittlichen Stromfluss I_{SC}, wobei die Aktivität auch hierbei involviert ist, da der Kurzschlussstrom nur bei Signalwechseln fließt:

$$P_{SC} = V_{DD} \cdot I_{SC} \tag{25}$$

Eine genaue Analyse der Leistungsaufnahme auf Gatterebene erfolgt anhand von Daten aus der Gatterbibliothek, die die statische und dynamische Leistungsaufnahme der einzelnen Gatter je Inputsignalkombination, Lastkapazität und Umgebungsbedingungen gespeichert hat. Weil der Aktivität eine besondere Rolle bei der Berechnung der dynamischen Leistungsaufnahme

zukommt, wird diese aus einer Datei zur Berechnung herangezogen. Diese durch die funktionale Simulation erzeugte Aktivitätsdatei weist den Knoten einer Gatternetzliste und seinen möglichen Spannungspegeln Wahrscheinlichkeitswerte zu, die auf die gesamte Simulationszeit bezogen sind. Damit ist es möglich, für jedes Gatter die Leistungsaufnahme zu berechnen, woraus durch Summation dann die gesamte Leistungsaufnahme errechnet werden kann.

Drittes Kapitel

3 Zuverlässigkeit von CMOS-Schaltungen

Neben den üblichen Schaltungscharakteristika, wie Verzögerungszeit, Leistungsaufnahme und Fläche rücken zunehmend die Lebensdauer und die Zuverlässigkeit der zu entwickelnden Schaltung in den Fokus des Designers. Was seit jeher ein wichtiger Aspekt der Forschung in der Produktion von Halbleiterschaltungen war, dringt verstärkt in das Betätigungsfeld des Chipentwicklers, da der Aufbau integrierter Schaltungen an sich schon Auswirkungen auf deren Zuverlässigkeit hat. Die Skalierung verstärkt zuverlässigkeitssenkende Effekte, so dass diese vom Designer in der Chipentwicklung beachtet werden müssen, weil defekte Schaltungen nicht in Gänze auszuschließen sind und der Anteil der defektlosen Chips nach der Produktion mit fortschreitender Verkleinerung abnimmt. Vollständige Tests, die alle potentiellen Fehler der Schaltungen nach der Produktion abdecken, sind allerdings aufgrund der immer weiter steigenden Komplexität nicht mehr (wirtschaftlich) durchführbar. Das folgende Kapitel befasst sich mit den grundsätzlichen Fehlermechanismen, die integrierte Schaltungen betreffen. Im Fokus stehen dabei Verschleißerscheinungen, insbesondere die Transistoren betreffend, da die verkleinerten Strukturgrößen anfälliger für Verschleiß sind und so zu einer verringerten Lebensdauer führen. Die Darstellung reicht von den fundamentalen physikalischen Grundlagen bis zu vorhandenen Methoden, um potentiell logischen Fehler entgegen zu treten. Grundlegende Begrifflichkeiten werden im folgenden ersten Abschnitt erläutert, wobei auch auf die Wahrscheinlichkeitstheorie eingegangen wird.

3.1 Fehler- und wahrscheinlichkeitstheoretische Grundlagen

Zur Abgrenzung von Ursache und Wirkung werden die im Alltagsgebrauch synonym verwandten Wörter Störung und Fehler in dieser Arbeit als auch in der Fachliteratur unterschieden. Eine Störung (engl. fault) ist ein Defekt der Hardware, während der Fehler (engl.

failure, error) das Unvermögen des Systems darstellt, die vorgesehene Funktion korrekt auszuführen. Einem Fehler als Wirkung geht also immer eine Störung als Ursache voraus. Umgekehrt wiederum führt eine Störung/Defekt nicht automatisch zu einem Fehler, da diese kompensiert werden können [Kor07].

Die Defektrate $\lambda(t)$ einer Komponente an einem definierten Zeitpunkt t stellt die Wahrscheinlichkeit dar, mit der die Komponente in einem bestimmten Zeitintervall einen Defekt erleidet. Dieser Defekt kann dann zum Ausfall der Komponente und damit zum Ausfall eines Systems aus Komponenten führen. Die aus den einzelnen Komponenten resultierende Fehlerrate bzw. Ausfallrate des Systems $\lambda_{SYSTEM}(t)$ ist quasi die erwartete Anzahl an Ausfällen (ungewollte Funktionsweise) je Zeitintervall eines funktionierenden Systems für zukünftige Zeitintervalle. Die Fehlerrate integrierter Schaltungen ist abhängig vom aktuellen Lebensalter, von der Temperatur, von der verwendeten Technologie und von den physikalischen Gegebenheiten (elektrische/physikalische Spannung, elektrischer Stromfluss, Dimensionierung, …). Die bekannte Badewannenkurve (Abbildung 3-1) gibt die Abhängigkeit der Fehlerrate vom Lebensalter wieder. Diese Darstellung fasst drei verschiedene Arten von Fehlerquellen in einer Kurve zusammen. Am Anfang werden eine unbestimmte Anzahl an Komponenten aufgrund von Produktionsfehlern einen oder mehrere Defekte aufweisen, und somit fehlerhafte Ergebnisse erzeugen. Mit der Zeit allerdings nimmt diese „Säuglingssterblichkeit“ ab und somit auch die Ausfallrate, da die fehlerhaften Schaltungen aussortiert wurden. Ursache für diese Frühfehler sind unvollständige Systemtests und stark geschädigte Komponenten, die schon nach einer geringen Lebensdauer, vollständige Defekte ausbilden. Aus diesem Grund werden auch so genannte „Burn-in-Tests“ durchgeführt, um anhand dieser Stresstests mit erhöhten Spannungen und extremen Temperaturen das Lebensalter der Schaltung künstlich zu erhöhen. In der zweiten Phase dominieren die zufälligen Ausfälle, deren Ausfallrate eine Konstante darstellt, weil sie über die Lebensdauer stabil bleibt. Am Ende der Lebensdauer verursachen dann Verschleißerscheinungen einen erneuten Anstieg der Fehlerrate, was die dritte Phase darstellt. Dieser Altersverschleiß wird durch den Betrieb der Schaltung verursacht und auch durch den

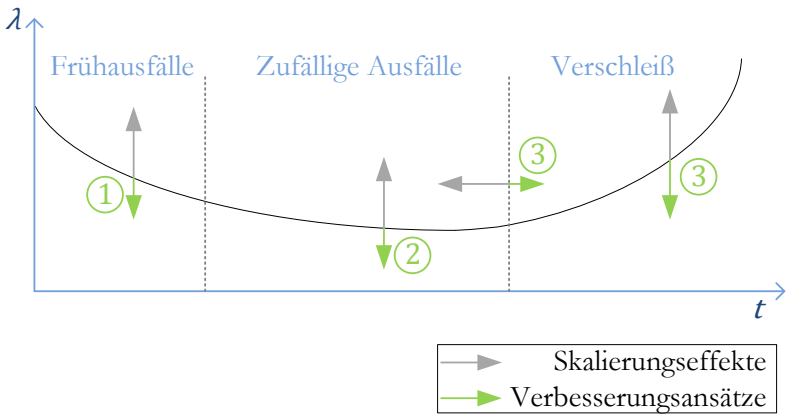


Abbildung 3-1 : Badewannenkurve und Verbesserungsansätze
1) Yielderhöhung
2) Vermeidung bzw. Maskierung von zufälligen und transienten Fehlern
3) Verzögerung und Minimierung von Verschleißeffekten

Produktionsprozess beeinflusst, da kleinere Defekte schon nach der Herstellung Verschleißerscheinungen fördern können. In Abbildung 3-1 sind auch die Effekte der Skalierung und die möglichen Verbesserungsansätze eingezeichnet, die in den nachfolgenden Abschnitten näher betrachtet werden. Durch die verkleinerten Strukturgrößen steigt die Ausfallrate generell an und der Verschleiß tritt früher ein. Verbesserungsansätze, die diesen Skalierungseffekten entgegenwirken, können nach der Phase in drei Gruppen eingeteilt werden. In der ersten Phase wird versucht, die Ausbeute (engl. yield) zu erhöhen bzw. konstant zu halten, da dies vor allem Produktionsfehler betrifft. Defekte, die zu Fehlern und Ausfällen in Phase 2 führen, sind sowohl flüchtiger als auch permanenter Natur und es wird versucht, sie zu vermeiden bzw. sie zu maskieren. Die letze Phase 3 soll dann solange wie möglich hinausgezögert werden und eventuelle Verschleißdefekte sollen abgemildert werden, so dass z. B. Schaltungscharakteristika, wie die Verzögerungszeit oder die Leistungsaufnahme, sich langsam(er) verschlechtern und somit nicht sofort zu Fehlern und Ausfällen führen (engl. graceful degradation).

Die Zuverlässigkeit einer einzelnen Komponente kann man als Wahrscheinlichkeit *R(t)* beschreiben. Sie ergibt sich aus der Lebenszeit *ttf* (engl. time to failure) der Komponente. Da im Folgenden auch Defekte nicht sofort zu Funktionsfehlern führen werden, wird der Begriff Lebensdauer *ttf* synonym für einen defektfreien Betrieb der jeweiligen Komponente verwendet. Wenn wir von permanenten und irreparablen Fehlern bzw. Defekten ausgehen, die eine zum Zeitpunkt $t = 0$ funktionierende Komponente nach *ttf* inkorrekte Ergebnisse erzielen lassen, stellen die Funktionen *f(t)* die Wahrscheinlichkeitsdichte und *F(t)* die kumulative Verteilungsfunktion von *ttf* dar [Kor07]:

$$f(t) = \frac{dF(t)}{dt} \quad , \qquad F(ttf) = \int_0^t f(\tau) d\tau \tag{26}$$

F(t) ist die Wahrscheinlichkeit der Komponente, vor und zum Zeitpunkt *t* auszufallen. Die Zuverlässigkeit *R(t)* ist in dem Sinne das Gegenteil zu *F(t)*, da sie die Wahrscheinlichkeit angibt, in welchem Maße die Komponente kontinuierlich im Intervall [0;*t*] fehlerfrei funktioniert:

$$R(t) = 1 - F(t) \tag{27}$$

Die Fehlerrate einer Komponente, welche gleich der Defektrate ist, kann dann mit folgender Gleichung ausgedrückt werden [Kor07]:

$$\lambda(t) = \frac{f(t)}{1 - F(t)} = -\frac{1}{R(t)} \cdot \frac{dR(t)}{dt} \; mit \; \frac{dR(t)}{dt} = -f(t) \tag{28}$$

Bei konstanter Fehlerrate $\lambda(t) = \lambda$, wie in Phase 2 der Badewannenkurve, ergibt sich aus:

$$\frac{dR(t)}{dt} = -\lambda \cdot R(t) \tag{29}$$

folgende Lösung für die Zuverlässigkeit bei $R(t = 0) = 1$:

$$R(t) = e^{-\lambda t} \tag{30}$$

was eine exponentielle Verteilung für die Lebensdauer *ttf* ergibt.

Für nicht konstante Defektraten (Phase 1 und 3) wird zumeist eine Weibullverteilung mit den Konstanten λ (Skalierungsparameter $\lambda = 1/ttf$) und dem Weibullparameter β (Formparameter) für die Wahrscheinlichkeitsverteilung angenommen [Kor07]:

$$f(t) = \lambda \cdot \beta \cdot t^{\beta-1} \cdot e^{-\lambda t^{\beta}} \tag{31}$$

woraus sich folgende Fehlerrate ergibt:

$$\lambda(t) = \lambda \cdot \beta \cdot t^{\beta-1} \cdot e^{-\lambda t^{\beta}} \tag{32}$$

Durch die Anpassung des Formfaktors β kann man die Fehlerrate ansteigen (Verschleiß: $\beta > 1$), konstant ($\beta = 1$) und absinken (Frühausfälle: $\beta < 1$) lassen. Die Gleichung für Zuverlässigkeit wird dabei folgendermaßen berechnet:

$$R(t) = e^{-\lambda t^{\beta}} \tag{33}$$

Die kumulative Verteilungsfunktion der Lebensdauer $F(t)$ ist dann [Wu02a]:

$$F(t) = 1 - e^{-\left(\frac{t}{MTTF}\right)^{\beta}} \tag{34}$$

Der Parameter *MTTF* wird eingeführt, weil Wahrscheinlichkeiten wie die Zuverlässigkeit $R(t)$ verschiedener Schaltungen schwer miteinander zu vergleichen sind. Eine auf die Zuverlässigkeit Bezug nehmende Metrik ist die mittlere Zeit bis zum ersten Ausfall *MTTF* (engl. Mean Time to Failure). Sie gibt die durchschnittliche Lebenszeit eines Systems/einer Komponente an und definiert sich folgendermaßen [Kor07]:

$$MTTF = \int_0^{\infty} t \cdot f(t)dt = -\int_0^{\infty} t\frac{dR(t)}{dt}dt = -t \cdot R(t)|_0^{\infty} + \int_0^{\infty} R(t)dt \tag{35}$$

Da $R(t = \infty) = 0$, ergibt sich:

$$MTTF = \int_0^{\infty} R(t)dt \tag{36}$$

In der Literatur wird auch häufig der Begriff *ttf63* oder *T63* verwendet, der im Grunde das gleiche bedeutet. Hierbei wird die *MTTF* als empirischer Zeitparameter für Ausfall- oder Defekteintrittszeitpunkte betrachtet, welcher den Zeitpunkt angibt, zu dem 63.2 % aller Ausfälle bzw. Defekte zu verzeichnen sind. Bei der Auswertung von Simulationen wird aufgrund der diskreten Messzeitpunkte t_i im Intervall $[0;t_{ENDE}]$, an denen der Ausfall der Komponenten geprüft wird, folgende Formel zur Berechnung der *MTTF* herangezogen:

$$MTTF = \sum_1^{i_{Ende}} \frac{R(t_i) + R(t_{i-1})}{2} \cdot (t_i - t_{i-1}) \tag{37}$$

wobei t_{ENDE} den Zeitpunkt darstellt, bei dem alle Komponenten ausgefallen sind.

Wird nun ein System von Komponenten betrachtet, ergibt sich die Zuverlässigkeit aus den Fehlerraten der einzelnen Komponenten. Dafür wird nachfolgend die Zuverlässigkeit von seriellen und parallelen Systemen betrachtet. Auch hier gilt die Annahme, dass die Defektrate einer Komponente gleich der Ausfall-/Fehlerrate ist. Wenn ein Teil eines seriellen Systems ausfällt, fällt unmittelbar das gesamte System aus. Dies kann vereinfacht für alle CMOS-Schaltungen gelten, wenn davon ausgegangen wird, dass ein Defekt im Transistor oder in einer Leitung zwangsläufig zu logischen Fehlern auf den Netzen führt, und somit am Ausgang des Systems falsche Ergebnisse generiert werden können, wobei das System dann als fehlerhaft deklariert wird. Die Zuverlässigkeit $R_{SERIAL}(t)$ eines seriellen System ergibt sich dann aus der Multiplikation der Zuverlässigkeit $R_i(t)$ aller n Komponenten [Rom05]:

$$R_{SERIAL}(t) = \prod_{i=1}^{n} R_i(t) \tag{38}$$

Parallele Systeme dagegen operieren gemäß ihrer Aufgabe, solange mindestens eine Komponente korrekt arbeitet:

$$R_{PARALLEL}(t) = 1 - \prod_{i=1}^{n} (1 - R_i(t)) \tag{39}$$

Dies gilt zum Beispiel für redundante Systeme, allerdings mit der Voraussetzung, dass fehlerhafte Komponenten nicht das Ergebnis verfälschen können, beispielsweise indem sie identifiziert und ausgeschaltet werden. Beide Gleichungen und die Annahme, dass ein Defekt unmittelbar zu einem Fehler führt, stellen nur vereinfachte Sichtweisen auf integrierte Schaltungen dar, da vor allem Verschleißeffekte erst mit stärkerem Ausmaß des Defekts zu logischen Fehlern führen bzw. führen können.

3.2 Übersicht Zuverlässigkeit

Fehler, die auf integrierte Schaltungen einwirken, können nach der Dauer ihres Effekts unterschieden werden. Im Fokus der Forschung standen in den letzten Jahren flüchtige bzw. transiente Defekte, die das System nur kurzfristig funktional oder leistungsmäßig beeinträchtigen. Hauptaugenmerk wird dabei auf die Soft Errors gelegt. Hierbei verursachen energiereiche Partikel logische Fehler. Diese Partikel sind vor allem Neutronen der kosmischen Strahlung oder α-Teilchen des Package-Materials. Wenn diese Teilchen den Chip passieren, können Elektronen-Loch-Paare generiert werden, die von Source- und Diffusionsgebieten aufgefangen werden können. Wenn genügend Ladung „angesammelt" wird, kann sie den Zustand eines Netzes stören und den Zustand „kippen" [Wes93]. Vor allem bei sequentiellen Elementen, wie Latches, Flipflops und SRAM-Zellen kann dieses Phänomenen zu Fehlern führen [Mit05]. Studien haben gezeigt, dass die Fehlerrate für einzelne Komponenten über die Technologien verschiedener Strukturgrößen nahezu konstant bleibt, allerdings steigt durch die erhöhte Anzahl der

Komponenten auch die Anfälligkeit integrierter Schaltungen gegenüber Soft Errors [Haz03]. Der Anstieg der Systemfehlerrate kann durch eine geringere Skalierung der Versorgungsspannung und der Kapazitätsgrößen verringert werden. Des Weiteren zeigen Schaltungen des Silicon-on-Insulator-CMOS, bei dem das Substrat durch isolierende Schichten unterbrochen wird, eine bessere Defektresistenz als das bulk-CMOS infolge der kleineren Si-Ebenen, weil hier weniger Ladung gesammelt werden kann [Muk05].

Neben den Soft Errors verursachen auch externe Umgebungsvariationen transiente Defekte. So führen Störungen der Spannungsversorgung zu Performanceverlusten bzw. -variationen und Rauschabstandsverringerungen. Außerdem entstehen durch unterschiedliche Last- und Aufgabenverteilungen zu bestimmten Zeiten Temperaturschwankungen auf dem Chip. Dies führt zu unerwünschten, da kaum vorhersagbaren Performancevariationen [Wes93]. Zurückzuführen ist das zum einen auf den Drainstrom, der über die Schwellspannung auch von der Temperatur beeinflusst wird. Zum anderen hängen die Widerstandswerte der Leitungen von der Temperatur ab. Somit variiert dadurch auch die Verzögerungszeit der Schaltungen.

Des Weiteren bewirken auch operative Effekte auf dem Chip flüchtige Defekte. Sie werden zumeist durch ungewollte, parasitäre kapazitive oder/und induktive Kopplungen hervorgerufen. Dazu zählt das Übersprechen (engl. cross talk), das zu Störungen von Signalpegeln und dadurch zur Beeinflussung der Verzögerungszeit und der Signalintegrität führt [Cat67], und andere nachfolgend erläuterte Effekte, die beim Schalten der Logikgatter auftreten. Einerseits kann das Schalten zum Anheben des Massepegels auf den Versorgungsleitungen (engl. ground bounce) und infolgedessen zu falsch schaltenden Gattern, Taktungsfehlern und Jitter führen [Hey03]. Andererseits werden vor allem analoge Bauelemente einer Mixed-Signal-Schaltung durch das aus dem Schalten generierte Substratrauschen (engl. substrate noise) gestört [Van01]. Ferner steigt mit zunehmender Skalierung der Einfluss der Leckströme auf die Zuverlässigkeit. Sie entladen ungewollt dynamische Knoten, erhöhen aber auch stark den Stromverbrauch und damit auch mittelbar die Temperatur [Sil07].

Die flüchtigen Defekte können über den gesamten Zeitraum der Badewannenkurve auftreten, allerdings sind sie vor allem für Phase 2 während der „normalen“ Betriebszeit relevant, wobei hierbei maximal nur kurzfristige Systemausfälle zu verzeichnen sind, da nach einer definierten Zeitspanne die Schaltung wieder korrekte Ergebnisse erzielt, spätestens mit Neustart des gestörten Systems. Die Defekte, die zu irreparablen Ausfällen in allen drei Phasen führen, sind dauerhaft und werden deshalb nachfolgend als permanente Defekte bezeichnet. Im günstigsten Fall manifestieren sich diese Defekte nicht als Fehler im System und führen daher nicht zu einem Systemausfall. Aufgrund des Auftretens der Störung können hierbei zwei grundlegende Gruppen unterschieden werden. Zum einen führen Initialdefekte zu Fehlern oder Variationen der Schaltung, die in der Phase 1 der Badewannenkurve beobachtet werden können („Säuglingssterblichkeit“) und werden zumeist vor der ersten Inbetriebnahme beim Kunden als Fehler erkannt. Verschleißeffekte hingegen sind die andere Gruppe und werden im Anschluss an dieses Kapitel eingehend erläutert, da der Fokus dieser Arbeit auf der Vermeidung bzw. Verzögerung von Verschleißeffekten liegt. Diese Verschleißmechanismen werden aufgrund der Skalierung immer früher auftreten und deren unerwünschten Effekte stärker ausfallen.

Initialdefekte sind vor allem auf Produktionsfehler zurückzuführen und vermindern die Ausbeute, was das Verhältnis der korrekt funktionierenden Chips zu allen produzierten darstellt:

$$Yield = \frac{\#(funktionsfähig\ eingestufte\ Chips)}{\#(produzierte\ Chips)} \tag{40}$$

Produktionsfehler, die zur Verringerung der Ausbeute beitragen, können in zwei Gruppen zusammengefasst werden. Einerseits vermindern zufällige Mängel, wie Schmutzpartikel in der Atmosphäre der Fabrik die Ausbeute, andererseits verursachen produktionstechnische Probleme, wie beispielsweise Fehlkalibrierungen, lithografische Effekte, unvollständige Wafer-Planarisierung oder verschmutzte Materialen, systematische Mängel, die zu defekten Chips führen. Die Schmutzpartikel können sowohl Leitungen trennen als auch zwischen Leitern Brücken und damit Kurzschlüsse bilden. Die Wahrscheinlichkeit, dass ein Partikel einen Defekt verursacht ist abhängig von der Chipgröße A_{DIE} und der durchschnittlichen Defektrate D (Defekte je Flächenabschnitt). Ein üblicher Ansatz ist die Annahme, dass die Defekte Poisson-verteilt sind, was zu folgender Ausbeute je Wafer führt [Wes93]:

$$Yield_{Wafer} = e^{-A_{DIE} \cdot D} \tag{41}$$

Da die Ausbildung eines kritischen Defekts von der Partikelgröße, der Lage, dem Layout und dessen Strukturdichte abhängig ist, wurde der Begriff der kritischen Fläche eingeführt [Kor98]. Diese Metrik gibt Aufschluss über die Anfälligkeit des Designs für zufällige Produktionsfehler und damit für eine Verringerung der Ausbeute. Zufällige Defekte sind weit schwieriger zu vermeiden als systematische, die während der Anlaufzeit (engl. ramp-up) einer neuen Prozesstechnologie immer weiter reduziert werden können.

Beide Defektgruppen verursachen wiederum entweder funktionale Defekte, bei der die Chips nicht die gewünschte Funktion ausführen können, oder aber parametrische Defekte, bei denen der korrekt funktionierende Chip bestimmte andere geforderte Spezifikationen, wie eine Mindestfrequenz, nicht einhalten kann [Shi03] [Whi07]. Verursacht durch Prozessvariationen in der Produktion, steigt der Einfluss parametrischer Defekte aufgrund der Skalierung weiter an. Die physikalischen Dimensionen auf dem Chip sind mittlerweile so klein, dass winzige Schwankungen bei der Herstellung, wie sie statistisch vorkommen, immer größere Variationen elektrischer Eigenschaften gleicher Chips hervorrufen.

Auch Entwurfsfehler können Fehler der Hardware provozieren. Hierzu zählen z. B. Wettlaufsituationen, Latch-Up oder die Verletzung von Designregeln. Nur durch eine entsprechende Qualitätssicherung der Entwurfsschritte, der verwendeten Bibliotheken und der CAD-Werkzeuge sind diese Fehler vermeidbar.

In Abbildung 3-2 sind die erwähnten Defektarten, inklusive der Verschleißeffekte, die in den folgenden Kapiteln näher erläutert werden, systematisiert.

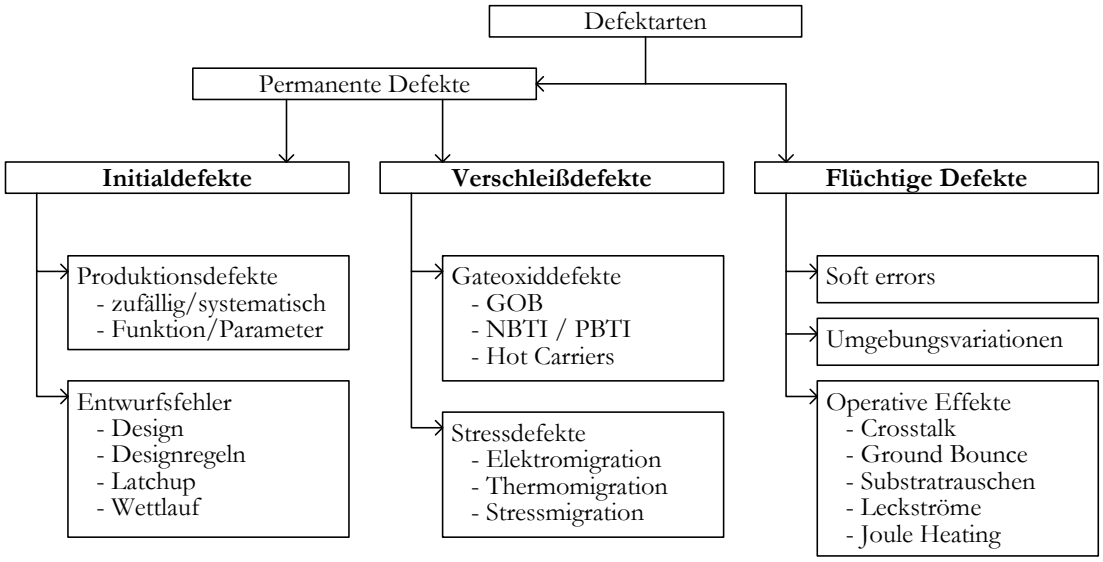


Abbildung 3-2 : Defektarten in CMOS-Schaltungen

3.3 Leitungsverschleiß: Elektromigration, Stresseffekte und thermische Einflüsse

Verschleißmechanismen, die sich auf Leitungen und deren Verbindungen, die Vias, auswirken, sind Migrationseffekte, bei denen das Material der Leitungen aufgrund des Stromflusses von der ursprünglichen Lage wegtransportiert werden. Durch Kollisionen der Elektronen mit den Metallionen („Elektronenwind“) und zu einem kleinen Teil auch aufgrund des anliegenden elektrischen Feldes werden die Ionen in Richtung der Elektronen bewegt. So entstehen an einem Ende der Migration höhere Leitungswiderstände und damit Performanceverluste oder sogar physische Leitungsunterbrechungen (engl. void), am anderen Ende eine Materialanhäufung (engl. hillock), was zu Kurzschlüssen führen kann [Bla69]. Das derzeitig akzeptierte Modell für Lebensdauer in Hinblick auf die Elektromigration ist Black’s Gleichung [Sri04]:

$$ttf_{EM} \propto (J - J_{CRIT})^{-n_{EM}} \cdot e^{\frac{E_{a_{EM}}}{k_B \cdot T}} \tag{42}$$

wobei J die Stromdichte der Leitung, J_{CRIT} die kritische Stromdichte für die Elektromigration, E_{aEM} die Aktivierungsenergie der Elektromigration, k_B die Boltzmannkonstante, n_{EM} eine Materialkonstante und T die absolute Temperatur in Kelvin ist. Der zweite Teil der Gleichung ist der Arrhenius-Gleichung ähnlich und demonstriert die Abhängigkeit der Elektromigration von der Temperatur. Deshalb wird auch zwischen Elektro- und Thermomigration unterschieden [Ru00]. Während die Elektromigration durch bereits vorhandene Defekte stark unterstützt wird, basiert die Thermomigration auf einem Materialtransport ohne Defektstellen, nur hervorgerufen durch eine genügend hohe Temperatur und Stromdichte durch den Leiter/Via, die zur vereinfachten Analyse auf die Schaltwahrscheinlichkeit zurückzuführen ist [Gup11]:

$$j = \frac{i}{A_{IC}} = \frac{C_{IC} \cdot V_{DD} \cdot f_{CLK}}{A_{IC}} \tag{43}$$

wobei A_{IC} die Querschnittfläche und C_{IC} die Kapazität des Leiters bzw. des Vias ist.

Ein weiterer Leitungsdefekt, der den vorhergehenden ähnlich ist, ist die Stress Migration. Hierbei entstehen mechanische Verschiebungen durch unterschiedliche thermische Ausdehnungen der verschiedenen Materialien, was vor allem den Übergang einzelner Vias zu größeren Leitern betrifft. Dieser Stress ist proportional zum Unterschied zwischen Betriebstemperatur und Produktionstemperatur, wo kein mechanischer Stress vorhanden ist. Ein großes Problem der Stressmigration ist die stressfreie Temperatur (SFT), die aufgrund der glockenförmigen Temperaturabhängigkeit der Stressmigration von der Temperatur ungefähr bei den Stresskonditionen (200°C) für Elektromigrationstests liegt, während die Temperatur, bei der Stressmigration auftritt, ungefähr der Betriebstemperatur (100 °C – 125 °C) entspricht [Her10].

Mit Einführung kleinerer Technologieknoten werden Migrationseffekte weiter verstärkt. Allein aus Gleichung (43) ist zu erkennen, dass mit der nicht proportionalen Skalierung der Versorgungsspannung gegenüber der Querschnittfläche die Stromdichte ansteigen wird. Dies wird teilweise kompensiert, indem das Aspect Ratio AR, d.h. das Verhältnis der Höhe der Leitung zu ihrer Breite, erhöht wird. Ferner ist die Lebensdauer bei Aluminiumleitern größer als bei Kupferleitungen, die in den neueren Technologien eingesetzt werden [Ala05]. Bei konstanter Stromdichte wird davon ausgegangen, dass die Elektromigration schwächer wird, da besonders an den Übergängen von der Leitungsebene zur nächsten Via-Ebene der Hauptteil der Migration stattfindet, deren Fläche sich mit der Skalierung verringert. Weiterhin wird versucht, der Elektromigration mittels low-k Dielektrikum-Materialien und veränderten Prozessschritten wie dem Ummanteln der Kupferoberfläche (engl. Cu-Cladding) und mit anderen Materialien entgegen zu wirken [McP07]. Dieses Cu-Cladding wirkt sich auch positiv auf die Stressmigration aus, da es ein verzögerndes Element darstellt, sobald die Fehlstellenbildung beginnt. Bei kleineren Technologien werden allerdings auch Stressmigrationseffekte bei kleineren Metallleitern beobachtet, was zu einem erhöhten Auftreten dieses Migrationseffektes aufgrund der Skalierung führt.

Ein weiterer Effekt in den Leitungen ist das Self-Heating bzw. Joule-Heating. Hierbei wird aufgrund des Stromflusses Wärme im Leiter produziert, was zu Performanceverlusten aufgrund von erhöhten Widerständen und verstärkter Elektromigration führt [Wes93]. Auch dieser Mechanismus ist abhängig von der Stromdichte und deren Einfluss wird daher mit der Skalierung erhöht. Ferner verstärkt auch die Verwendung von low-k Materialien das Joule-Heating, da sie schlechtere Temperaturleiteigenschaften besitzen.

3.4 Transistorverschleiß

Einen signifikanten Effekt haben erhöhte Temperaturen auch auf den Verschleiß der Transistoren [McP07], insbesondere auf den Verschleiß des Gateoxides, was die „verwundbarste“

Stelle im Transistor darstellt. Dabei wirken sich die unterschiedlichen Mechanismen von verschiedenen Seiten aus. Das Time-Dependent Dielectric Breakdown (TDDB), oftmals auch synonym als Gate Oxide Breakdown (GOB) bezeichnet, wirkt sich vom Gateanschluss durch Tunnelströme auf das Gate aus, während Hot-Carriers (HC) Ladungsträger sind, die vom Source-Drain-Kanal in das Gateoxid diffundieren und dort Schäden anrichten. Eine dritte wichtige Ursache für Defekte im Gateoxid sind die beiden BTI-Effekte (NBTI – engl. Negative Bias Temperature Instability, PBTI – Positive Bias Temperature Instability), bei denen der Transistor durch Defekte an der Grenzschicht zwischen Silizium und Gateoxid geschwächt wird. Dabei sind die Auswirkungen dieser Ursachen im Groben ähnlich, weil sich das Verhalten des Transistors mit der Zeit verändert, vor allem die Strom-Spannungskurven. Weil sie sich im Detail unterscheiden, werden die Defektursachen, ihr Mechanismus und ihre Auswirkungen nachfolgend näher erläutert.

3.4.1 Bias Temperature Instability

Die Bias Temperature Instability (BTI) verändert Transistoreigenschaften, wie die Schwellspannung V_{TH}, die Konduktanz g_m, den Drainstrom I_{DSAT} und die statischen Ströme. Da dieser Effekt bei erhöhter Temperatur (100 °C bis 250 °C) und bei einer negativen Gate-Source-Spannung V_{GS} auftritt, sind vor allem p-MOSFET von diesem Phänomen betroffen (NBTI). Bedingungen, die das NBTI fördern, werden während des Burn-Ins aber auch während des normalen Betriebs in hochperformanten Schaltungen erreicht. Durch die Veränderung der Transistoreigenschaften sind ungewollte zeitliche Abweichungen der Pfade einer Schaltung möglich, so dass es zu logischen Fehlern kommen kann [Kun06]. Zwei verschiedene Modelle werden herangezogen, um die Änderung der Transistoreigenschaften zu erklären. Zum einen werden Ladungen über das gesamte Oxid angesammelt (engl. charge trapping), was zu einer Veränderung der Ladungsverteilung und -dichte des Gateoxids führt [Leb94], wobei dann Elektronen aufgenommen und wieder abgeben werden können. Zum anderen können Ladungen direkt an der Barriere zwischen Si und SiO_2 die Bindungen von Elektronenpaaren aufbrechen. Dieser Interfacedefekt (engl. interface trap, fast interface state) kann umgehend ein Elektron aufnehmen, wenn der Transistor in der Sättigung ist, was dann ebenfalls zu einer Veränderung der Ladungsverteilung des Gateoxides führt, da dieser Defekt negativ geladen wird [Her88]. Beide Abläufe werden als Falle (engl. trap) bezeichnet, da sie Ladungen „fangen“, obwohl sie meistens neutral sind. In der Nähe positiver Ladungsträger (Anode) bzw. negativer Ladungsträger (Kathode) können sie dementsprechend schnell positiv bzw. negativ aufgeladen werden [Dum02]. Der Einfluss dieser beiden Defektmodelle kann über die Schwellspannungsgleichung (1) eines p-MOSFET erklärt werden, deren Flachbandspannung V_{FB} mit der folgenden Gleichung gegeben ist [Sch03]:

$$V_{FB} = \varphi_{ms} - \frac{Q_f}{C_{OX}} - \frac{Q_{it}(\varphi_S)}{C_{OX}} \tag{44}$$

mit Q_f als gesamte Oxidladungsdichte und Q_{it} als Interfacedefektladungsdichte, die vom Oberflächenpotential φ_S an der Si/SiO_2-Barriere abhängt. Dies sind die beiden einzigen

Parameter, die eine Verschiebung der Schwellspannung unter NBTI-Stress verursachen können, wobei ein positiver Anstieg einer oder beider Dichten zu einem Absinken der Schwellspannung führt. Durch diese Verschiebung der Schwellspannung wird nach (10) auch der Drainstrom I_{DSAT} beeinflusst. Neben der Schwellspannungsänderung führt auch eine Veränderung der Mobilität zur Verschlechterung dieser Parameter, was auch aus interface traps resultiert. Ein bemerkenswertes Phänomen bei dieser Defektart ist die teilweise Regeneration, wenn die Stressparameter zurückgenommen werden [She06]. NBTI-Effekte werden wieder wichtiger, da sowohl die elektrischen Felder als auch die Temperaturen während des Betriebs steigen. Des Weiteren führt eine Annäherung von Gatespannung V_{GS} und Schwellspannung V_{TH} bei gleichbleibenden bzw. ansteigenden Schwellspannungsvariationen ΔV_{TH} zu größeren Abweichungen des Drainstroms ΔI_{DSAT} [McP07]:

$$\frac{\Delta I_{DSAT}}{I_{DSAT}} = \theta \left(\frac{\Delta V_{TH}}{V_{GS} - V_{TH}} \right) \tag{45}$$

3.4.2 Hot Carrier Effekt

Übersteigt das anliegende elektrische Feld den für NBTI kritischen Wert, treten Hot Carriers auf, die das Gateoxid schädigen, und damit die Eigenschaften des Transistors dauerhaft verändern. Hot Carriers sind Ladungsträger (Elektronen und Defektelektronen) des Gate-Source-Kanals, die eine hohe kinetische Energie durch die Beschleunigung elektrischen Felder erfahren. Diese Ladungsträger werden dann zu Hot Carriers, wenn sie durch die Beschleunigung in das Gateoxid eintreten, da deren Auftreten dort nicht vorgesehen ist. Dabei sind verschiedene Szenarien für das Eindringen der Ladungsträger postuliert worden (Tabelle 3-1):

Tabelle 3-1 : Hot-Carrier Eindringmechanismen

Injektionsart	Abk.	Wirkungsweise
Channel Hot Electron	*CHE*	Abbildung 3-3 a): Diese als „Lucky Electrons" bekannten Elektronen werden durch die Drain-Source Spannung entlang dem Kanal so stark beschleunigt, dass sie die Si/SiO_2-Barriere durchdringen können und dort einen Strom erzeugen [Leb94]. Dieser Effekt ist maximal bei $V_{DS} \approx V_{GS}$ [Cot79].
Drain Avalanche Hot Carrier	*DAHC*	Abbildung 3-3 b): Bei der Beschleunigung im Kanal können Ladungsträger mit den Siliziumatomen kollidieren und Elektronen-Loch-Paare bilden (Stoß-ionisation). Bei einem starken elektrischen Feld (in der Nähe des Drains) kann sich ein Schnellballeffekt (engl. avalanche) hervorrufen, da die erzeugten Ladungsträger teilweise ebenfalls genug Energie erreichen, um ins Gateoxid zu gelangen oder weitere Elektronen-Loch-Paare zu generieren [Tak83]. Daher ist dieser Effekt bei $V_{DS} >> V_{GS}$ am stärksten ausgeprägt.
Substrate Hot Electron / Hot Hole	*SHE SHH*	Abbildung 3-3 c): Hierbei werden aufgrund einer hohen Substratspannung Ladungsträger des Substrats Richtung Gateoxid beschleunigt. Die kinetische Energie wird in der Raumladungszone weiter erhöht und die Ladungsträger können dann in das Gateoxid eindringen [Tak83]
Substrate current induced Hot Electron	*SCHE*	Abbildung 3-3 d): Bei $V_{DS} < V_{GS}$ können durch Elektronen, die durch Stoßionisation mit dem Substratstrom erzeugt wurden, in das Gateoxid eindringen [Tak83].
Secondarly Generated Hot Electrons	*SGHE*	Photonen, die im hochenergetischen Feld beim Drain erzeugt werden, generieren Elektronen-Loch-Paare, die ins Gateoxid gelangen können [Koh80].

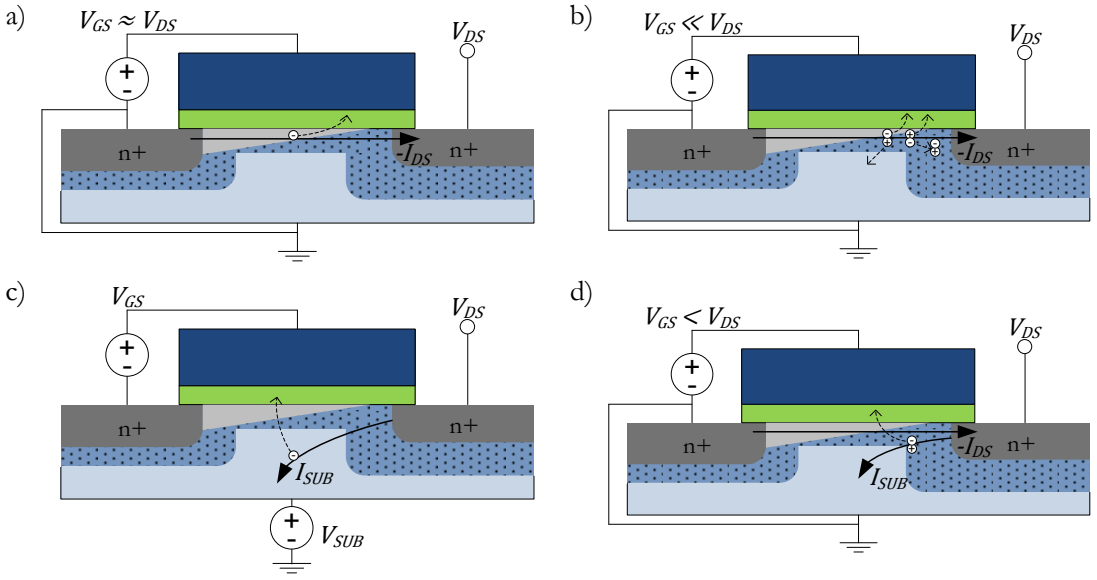


Abbildung 3-3 : Hot-Electron-Injektionsarten:
a) Channel Hot Electron
b) Drain Avalanche Hot Carrier
c) Substrate Hot Electron
d) Substrate Current Induced Hot Electron

All diese Effekte sorgen dafür, dass Elektronen und Löcher das Gateoxid und/oder die Barriere zwischen diesem und dem Kanal beschädigen. Diese Schädigung führt vor allem dazu, dass die Strom-Spannungskurve des MOSFET verändert wird, was zu unausgeglichenen Schaltungen führen kann. Wieder dienen die Modelle des charge trapping und der Interfacedefekte zur Erklärung der Verschiebung der Ladungsdichte und -verteilung des Gateoxides [DiM89]. Durch die Verminderung des Gate-Source-Stroms sind vor allem n-MOSFETs von diesem Effekt betroffen, die mit der Zeit langsamer schalten, da der Drainstrom verringert wird. Ferner wird auch die Lebensdauer eingeschränkt, die mit folgender Gleichung dargestellt werden kann [McP07]:

$$ttf_{HC} \propto {I_{BULK}}^{-n_{SUB}} \cdot e^{\frac{E_{a_{HC}}}{k_B \cdot T}} \tag{46}$$

wobei I_{BULK} der Substratstrom ist, der durch Spannungsstress hervorgerufen wird, n_{SUB} der Exponent für die Substratstromabhängigkeit und E_{aHC} die Aktivierungsenergie sind. Da E_{aHC} für n-MOSFETs bei -0.2 eV bis 0.4 eV liegt, während sie für p-MOSFET immer positiv ist (0.1 eV bis 0.4 eV), ist die Lebensdauer der n-MOSFETs geringer.

Die Herabsetzung der Versorgungsspannung in den letzten Jahren hat erheblich dazu beigetragen, Hot-Carrier-Effekte zu entschärfen. Allerdings werden aufgrund der nicht zu erwartenden weiteren Verringerung von V_{DD} und auch wegen des Einsatzes neuer Materialien, Hot-Carrier-Defekte wieder verstärkt untersucht werden müssen [McP07].

3.5 Grundlagen des Gate Oxide Breakdown

Wird das Gateoxid derart geschwächt, dass es seine Funktionsfähigkeit verliert, spricht man vom Gateoxiddefekt (engl. Gate Oxide Breakdown – GOB, auch Dielectric Breakdown). Wirkt ein kleines elektrisches Feld auf das Gate, wie im „normalen" Betrieb üblich, ist der Prozess zeitabhängig, und wird deshalb auch Time-Dependent Dielectric Breakdown (TDDB) genannt. Im Folgenden werden die Begriffe synonym verwendet, da ein Gateoxiddefekt zwar auch durch Spannungsspitzen hervorgerufen werden kann, dies allerdings nicht Inhalt dieser Arbeit ist. Weil das Ausmaß dieses Effekts mit dem Anstieg des elektrischen Feldes über dem Gate und kleineren Gateoxiddicken zunimmt, fokussiert sich diese Arbeit auf das Gate Oxide Breakdown als Defektursache für logische und parametrische Fehler, die vermieden bzw. maskiert werden sollen. Zudem folgen Gateoxiddefekte einem progressiven Verlauf, so dass diese sich erst langsam mit veränderten Transistoreigenschaften wie bei HC- und NBTI-Defekten, die auch Gateoxiddefekte verschlimmern können, vor dem logischen Defekt anbahnen. Dementsprechend können später vorgestellte Verbesserungen in integrierten Schaltungen auch diese Effekte vermindern.

Zwei grundlegende Erklärungsmuster wurden aufgestellt, um die Prozesse des Gateoxiddefektes zu erklären. Zum einen werden externe Faktoren, wie Prozessunreinheiten und der Fabrikationsprozess an sich (Ionenimplantation, Plasmaschäden oder mechanischen Stress) im Oxid als Ursachen benannt [Hor97] [Shi98], andererseits sind Modelle postuliert worden, die reine innere Prozesse als Ursachen für die Generierung von Ladungsfallen identifizieren, die dann ihrerseits Gateoxiddefekte erzeugen. Sie werden im folgenden Abschnitt näher betrachtet. Danach wird das Perkolationsmodell von Degraeve vorgestellt, dem es gelingt, von den physikalischen Ursachen zu abstrahieren und dadurch die Modelle zu vereinen, in dem die Entstehung eines leitenden Pfades durchs Gateoxid in den Vordergrund gestellt wird. Abschließend wird die zeitliche Abfolge eines Gateoxiddefektes zusammenfassend als progressiver Verlauf beschrieben, wie er Stand der Forschung ist und so auch in dieser Arbeit Anwendung findet.

3.5.1 Physikalische Modelle

Besonders Wasserstoffatome wurden als mögliche Verunreinigungen identifiziert, die zumindest einen signifikanten Einfluss auf das Entstehen von Gateoxiddefekten haben. Diese Wasserstoffatome, die bei der Produktion, z. B. bei der nassen thermischen Oxidation des Siliziumdioxides, an der Si/SiO_2-Barriere eingelagert werden, können durch hot electrons freigesetzt werden, da diese die Si-H-Bindungen aufbrechen können [DiM95]. Dies führt nicht nur zu einem Interface Trap, sondern auch zu einer Elektronenfalle im Oxid durch die Verbindung des Ions mit Silizium [Sta96]. Mit Studien [Sta02], in denen gezeigt wurde, dass Leckströme unter Stressbedingungen durch Tunneling Gateoxiddefekte verursachen, konnte auch der für Interface Traps wichtige Isotopeneffekt nachgewiesen werden.

Ähnlich dem AHR-Modell, zeigt das Anode Hole Injection Modell (AHI) große Übereinstimmungen mit experimentellen Daten bei starken elektrischen Feldern. Ursächlich für

die Defektelektronenfehlstellen (engl. hot holes) sind hot electrons, die die Anode erreichen. Dort werden die Löcher generiert, die dann vom Oxid aufgefangen werden (charge trapping), was in Abbildung 3-4 zu sehen ist. Hier lösen Elektronenlöcher Verbindungen im SiO_2. Diese Fallen führen zu einer erhöhten Stromdichte, was zu mehr hot electrons führt, die wiederum mehr Defektelektronen generieren können [Che86]. Ältere Messungen haben eine Gatespannungsschwelle von 7 V bis 8 V errechnet, ab der Elektronenlöcher gefangen werden [Lom05], so dass dieses Modell im Grunde nicht geeignet war, das Auftreten von Gateoxiddefekten bei kleineren Spannungen zu erklären. Untersuchungen um die Jahrtausendwende haben jedoch gezeigt, dass ein Ionisationsprozess von Minoritätsladungsträgern Elektronenlöcher auch bei geringeren Spannungen herauslösen kann [Bud98].

Die Projektion dieses Modells zur Prognose von Defekten wird auch als 1/E-Modell bezeichnet, da die Lebensdauer proportional zum Faktor $1/E_{OX}$ sein sollte, wobei E_{OX} das elektrische Feld über dem Gate darstellt [Ala00]. Dies folgt aus dem Verhältnis der kritischen Elektronenlochflussdichte Q_P zur Defektelektronenstromdichte J_H, die wiederum ein Produkt der

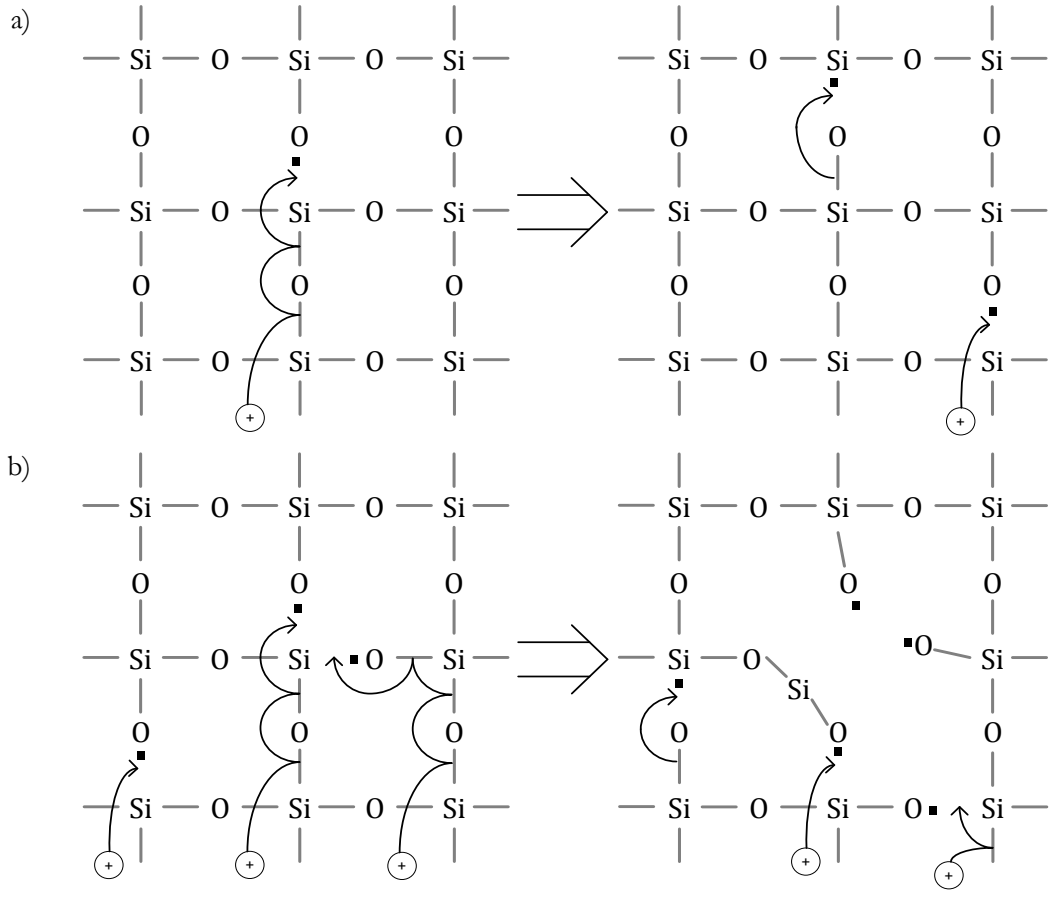


Abbildung 3-4 : Anode Hole Injektionsmodell
a) Geringe Defektelektronenstromdichte
b) Hohe Defektelektronenstromdichte mit der Möglichkeit permanenter Defektstellen, wenn zwei Hot Holes die Verbindungen eines Si-Atoms schädigen

Stromdichte J_e des Elektronentunnelstromes und der Tunnelrate G_H der generierten Defektelektronen ist:

$$ttf_{\frac{1}{E}} \approx \frac{Q_P}{J_H} = \frac{Q_P}{J_e \cdot G_H} mit\, J_e \propto e^{-\frac{1}{E_{OX}}}\, und\, G_H \propto e^{-\frac{1}{E_{OX}}} \tag{47}$$

Daraus ergibt sich folgende Abhängigkeit für die Lebensdauer:

$$ttf_{\frac{1}{E}} \propto e^{1/E_{OX}} \tag{48}$$

Da die kritischen Elektronenlochflussdichte Q_P nicht proportional zum elektrischen Feld E_{OX} über dem Gate ist, steht dieses Modell im Widerspruch zum ebenfalls populären Thermochemischen Modell (oder auch „E-Modell"), das das elektrische Feld E_{OX} als Hauptursache für die Defektgenerierung ausmacht, während der Strom durch das Gate im Gegensatz zum 1/E-Modell eine untergeordnete Rolle spielt. Modelle, die diese Richtung verfolgen, wurden schon Ende der 1970er aufgestellt [Cro79]. So wird in diesem Modell die Temperatur, die jedes Material mit der Zeit beeinflusst, im Gegensatz zum AHI-Modell auch berücksichtigt. Der Schwachpunkt des SiO_2-Gitters liegt demnach in bestimmten Si-O-Bindungen, deren Winkel zwischen 150° und 180° liegt. Diese Bindungen werden zu reinen Si-Si-Bindungen, wenn dort Sauerstoff-Atome fehlen. Durch deren relative „Schwäche" und der Polarität der Si-O-Bindungen kann es zum Aufbrechen dieser Bindungen kommen, wenn ein elektrisches Feld auf das Oxid wirkt. Dieses Feld wirkt auf die Polarität der Si-O-Bindungen, so dass sich lokale elektrische Felder bilden, die stärker auf die Moleküle wirken als das äußere Feld. Die „schwachen" Si-Si-Bindungen können durch diese lokalen Felder genügend thermischer Energie ausgesetzt werden, so dass sie brechen und eine Elektronenlochfalle bilden können [McP98]. Eine Projektion der Lebensdauer aufgrund dieses Modells weist demnach folgende Abhängigkeit auf:

$$ttf_E \propto e^{\left(\frac{E_{a_{GOB}}}{k_B \cdot T} - \gamma_{ACC} \cdot E_{OX}\right)} \tag{49}$$

mit E_{aGOB} als Aktivierungsenergie und γ_{ACC} als feldabhängigen Beschleunigungsfaktor.

Experimentelle Daten wurden mit diesem Modell sehr gut beschrieben, allerdings zeigten SHE-Injektionsexperimente, dass die Durchbruchsladung Q_{BD} von der Energie der tunnelnden Elektronen abhängig ist und nicht vom elektrischen Feld [Vog00]. In [Wu02b] wurde gezeigt, dass diese Durchbruchsladung abhängig von der Gatespannung V_{GS} und deren Polarität ist, so dass diese beiden Parameter als essentiell für die Projektion von Gateoxiddefekte aufgrund dieses Modells gelten [McP07]:

$$ttf_{GOB} \propto V_{GS}^{\,n_{GOB}(T)} \tag{50}$$

mit n_{GOB} als temperaturabhängiger Exponent, der typischerweise zwischen 40 und 48 liegt.

Dies gilt vor allem für geringere Oxiddicken, da hier das direkte Tunneln der Elektronen für die Fallengenerierung zuständig ist und deren Energie durch die Spannung an der Anode bestimmt wird. Neben den vorgestellten und populärsten Modellen zur Erklärung der Bildung von Elektronenfallen gibt es weitere, auf die hier nicht näher eingegangen wird. So soll z. B. auch Strahlung einen Einfluss bei der Entstehung von Traps haben. Darüber hinaus wird auch ersichtlich, dass hot electrons eine Rolle spielen, zumindest teilweise als Initiatoren, so dass auch die Spannung zwischen Drain und Source beachtet werden muss.

Vor dem eigentlichen Durchbruch, also dem Breakdown, verändert sich das Verhalten des Transistors durch die Stressbedingungen, die während des Einsatzes der Schaltung auf ihn wirken, wie angelegte elektrische Felder, Temperatur und Leckströme. So erhöht sich der ursprüngliche Leckstrom um eine stressbedingte Leckstromkomponente (engl. Stress Induced Leakage Current – SILC). Dieser Leckstrom ist auf tunnelnde Elektronen zurückzuführen, die immer in eine Ladungsfalle gelangen und dann wieder am anderen Ende des Gateoxides austreten. Dies wird deshalb auch als trap-assisted tunneling (TAT) bezeichnet [Wu99]. Hierbei wurde auch gezeigt, dass die Elektronen einen Teil ihrer Energie verlieren und somit nicht so stark zum Verschleiß beitragen wie das direkte Tunneln der Elektronen. Allerdings eignet sich dieser Strom sehr gut dazu, den Status des Verschleißes zu analysieren, da dessen Stromdichte J_{TAT} proportional zur Elektronenfallendichte N_{TRAP} ist:

$$J_{TAT} \propto N_{TRAP} \cdot \frac{P_{IN} \cdot P_{OUT}}{P_{IN} + P_{OUT}} \tag{51}$$

mit P_{IN} und P_{OUT} als Wahrscheinlichkeit, dass ein Elektron in eine Falle tritt bzw. austritt, wobei sich diese beiden Komponenten im Gleichgewicht befinden, und somit den Strom bestimmen.

3.5.2 Perkolationsmodell zur Abstraktion der physikalischen Ursachen

Die vorgestellten Modelle erklären die Entstehung von Ladungsfallen, die eigentliche Bildung eines leitenden Pfades durch das Oxid kann dagegen unabhängig von physikalischen Ursachen gesehen werden, da alle Modelle von kritischen Defektdichten ausgehen [Sta02], seien es nun zerstörte Verbindungen (E- und AHR-Modell) oder eine kritische Defektelektronenflussdichte wie in neueren AHI-Abhandlungen.

Ein simples Modell für die Ausbildung des Pfades ist die Knotenperkolation von Degraeve [Deg95], wobei Ladungsfallen mit definierter Wahrscheinlichkeit zwischen Gate und Substrat im Oxid unabhängig von den anderen eingefügt werden. Um die traps existieren Sphären mit konstantem Radius, die den leitenden Bereich um die Falle darstellen. Sobald ein leitender Pfad (über die Sphären) von einer Seite zur anderen erreicht wurde, stellt dies einen Durchbruch durch Ladungsträger dar, die entlang der Sphären „springen“ und somit einen Strom verursachen. Bei vielen Perkulationssimulationen lässt sich anhand des Verhältnisses der Anzahl der generierten Durchbrüche zu allen simulierten Gateoxiden ein kritische Elektronenfallendichte errechnen (engl. critical electron trap density – CETD).

Mehrere experimentell bestätigte Statistiken können so modellhaft beschrieben werden:

- Mit sinkender Gateoxiddicke t_{OX} sinkt die CETD und somit steigt die Wahrscheinlichkeit eines Breakdowns, da weniger traps gebraucht werden, um einen leitenden Pfad zu bilden
- Für dünnere t_{OX} wird die statistische Streuung der Breakdowns größer, da weniger traps einen Pfad bilden können
- Mit steigender Gatefläche A_{GATE} steigt die Wahrscheinlichkeit, dass sich doch noch ein leitender Pfad findet, und somit die Wahrscheinlichkeit für einen Breakdown

3.5.3 Jeder Durchbruch ist anders – die Entwicklung des progressiven Breakdowns

Wenn nun ein Durchbruch entstanden ist, ist seine Ausprägung unterschiedlicher Natur, wie Abbildung 3-5 zeigt. Hier sind die verschieden Stadien des Gateoxides anhand der logarithmischen Abhängigkeit des Gatestromes I_{GATE} von der Gatespannung V_{GS} unter Stressbedingungen abgebildet. Der auf dem ursprünglichen Leckstrom folgende SILC hat noch den gleichen exponentiellen Verlauf wie der defektfreie Leckstrom.

Darauffolgende Durchbrüche, die zum finalen Stadium bis hin zum Kurzschluss führen, können derart unterschiedliche Ausprägungen haben, dass sie in zwei Kategorien unterteilt werden. Frühere Untersuchungen gingen immer von so genannten harten Durchbrüchen (engl. hard breakdown) aus. Hierbei sind die Durchbrüche so ausgeprägt, dass der Stromverlauf durchs Gateoxid einen ohmschen Charakter aufweist, der Defekt im Grunde wie ein Widerstand wirkt [Ghe01]. Die grundsätzliche Idee hinter diesem harten Durchbruch geht davon aus, dass durch den stetig ansteigenden Strom eine kritische Verlustleistung erreicht wird, die lokal die Temperatur derart ansteigen lässt, dass das Silizium um den Durchbruch schmilzt, entlang des Defektes fließt und dadurch eine Verbindung durchs Gateoxid entstehen lässt (engl. thermal runaway), deren Kurzschlusscharakter dann von der Größe des Durchbruches abhängt [Ala02]. Im Gegensatz dazu wird beim Mitte der 1990er erkannten weichen Durchbruch (engl. soft breakdown), oder auch quasi breakdown genannt, die kritische Verlustleistungsgrenze nicht erreicht. Die Entwicklung des Stromes I_{GATE} zur finalen Zerstörung des Gateoxides ist sehr „verrauscht“ und der Strom vollzieht einige Sprünge. Wie in der Abbildung 3-5 a) zu sehen, ist der Verlauf der Stromkurve I_{GATE} eine Potenzfunktion, deren Ausmaß sich zwischen SILC und dem hartem Durchbruch befindet, weshalb auch der Begriff SILC-B-Modus für den weichen Durchbruch verwendet wird. Im Gegensatz zu ersten Vermutungen, die den weichen Durchbruch nur bei dünneren Gateoxiddicken entdeckten [Wei97], zeigten umfangreiche Charakterisierungen, dass dieser nicht in erster Linie von Transistorparametern (Gateoxiddicke, -fläche, Substrattyp) abhängig ist, die Strom-Spannungskurve allerdings schon [Mir00]. Die Beziehung, in der harter und weicher Durchbruch zueinander stehen, war Gegenstand kontroverser Studien, in wie weit der weiche Durchbruch ein Vorläufer des harten oder eine andere Art des Durchbruches bis zum finalen Kurzschluss ist, und ob beide denselben physikalischen Ursachen unterliegen.

Diese Unsicherheiten, so wie die nicht exakte Differenzierung beider Modi, was bei unterschiedlichen Stressbedingungen und Tests zu verschiedenen Interpretationen der entstandenen Modi führen kann, führen zu einem weiteren Modell, das des progressiven Durchbruches [Lom05]. Hierbei wird der GOB-Prozess in zwei grundlegenden Phasen unterteilt, der initialen Phase der Defektgenerierung, also der stetige Verschleiß des Gateoxides, und der darauffolgende progressive Breakdown. Dieser Breakdown kann dann wieder in drei aufeinander folgende Phasen aufgeteilt werden, was in Abbildung 3-5 b) gut nachvollziehbar wird, wo der Strom I_{GATE}, der durch das Gateoxid fließt, linear als Funktion der Zeit dargestellt ist. Als erstes gibt es aufgrund des stressbedingten Leckstromes (SILC) eine Art rauschenden Breakdown (engl. noise oder digital breakdown), in dem der Strom I_{GATE} kaum bis linear ansteigt, allerdings immer wieder Sprünge aufweist. Danach folgt die Hauptetappe (engl. main stage, auch analog breakdown oder SILC-B-Modus), in der I_{GATE} ansteigt bis dann in der letzten Phase eine Sättigung des Stromes stattfindet und somit eventuell ein finaler Kurzschluss erreicht ist. Erst zu diesem Zeitpunkt ist eine funktionelle Störung des Transistors gegeben, während in den vorigen Stadien die Charakteristika (Schwellspannung, Leckstrom, Konduktanz, Strom-Spannungskurve, usw.) verändert werden.

Ausgehend von den älteren Begrifflichkeiten stellt sich der progressive Durchbruch nach der Entstehung von stressbedingten Defekten als die Entwicklung vom weichen zum harten Breakdown mit eventuell anschließendem Kurzschluss dar. Die zeitliche Abfolge und Entwicklung eines progressiven Durchbruches ist abhängig von den Transistor- und Stressparametern, wobei der Parameter der Verschleißrate DR (engl. degradation rate) eingeführt wird [Lom05]. Sie ist die Veränderung des Durchbruchstromes $I_{GOB} = I_{GATE}(t > t_{SILC})$ über die Zeit zwischen zwei definierten Punkten, üblicherweise mit einer Dekade Abstand:

$$DR = \frac{dI_{GOB}}{dt} \tag{52}$$

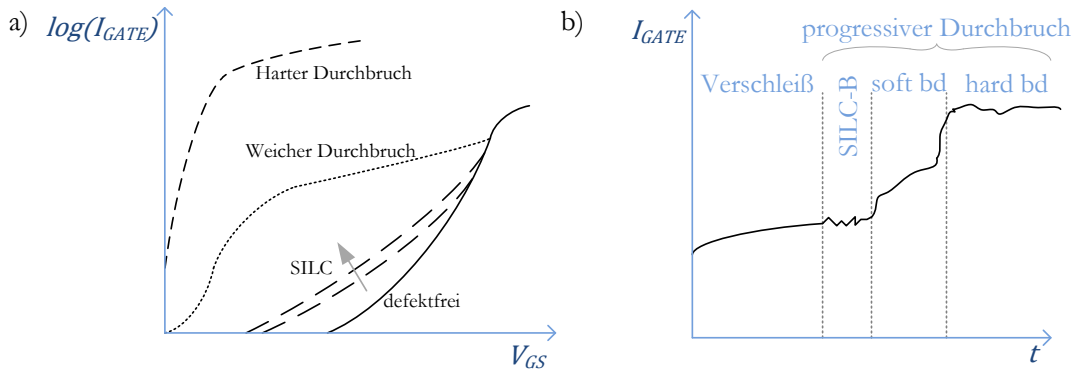


Abbildung 3-5 : Charakteristischer Durchbruchsstrom I_{GATE}

a) Typischer Verlauf des Stromflusses durch das Gate während verschiedener Durchbruchstadien unter Stressbedingungen
b) Zeitlicher Verlauf des Durchbruchstromes während der verschiedenen Phasen des progressiven Breakdowns unter Stress mit geringen Spannungen

Diese Defektrate für einen Durchbruch ist exponentiell abhängig von der Gateoxiddicke t_{OX} und unabhängig von Gateoxidfläche und dem Transistortyp [Ker07]. Die Transistorgatelänge beeinflusst signifikant den Mittelwert und die Streuung von DR. So steigt der Mittelwert von DR mit sinkender Gatelänge, während die Streuung abnimmt [Lin03].

3.6 Transistormodelle und Modellierungsparameter

Aus den erläuterten grundlegenden Modellen ist ersichtlich, dass die Entstehung eines Defektes stochastischen Gesetzen unterliegt und etwaige Simulationen, wie sie im Rahmen dieser Arbeit erarbeitet und durchgeführt wurden, dies zu berücksichtigen haben. Aus diesem Grund sind zwei Defektparameter von entscheidender Bedeutung. Zum einen wurde für jeden Transistor einer Schaltung der charakteristische Zeitpunkt $ttf_{GOB} = MTTF_{GOB\_TRANSISTOR}$ der Entstehung eines Defektes seines Gateoxides mittels einer kumulierten Verteilungsfunktion dargestellt. Diese wurde für jeden Transistor „individuell“ aus gegebenen Transistor- und Stressparametern abgeleitet. Deren Einfluss auf ttf_{GOB} wird im folgenden Abschnitt erläutert. Ferner ist die Entwicklung des Durchbruches, wie im vorigen Kapitel erläutert, in einem Parameter von entscheidender Bedeutung – die Entwicklung des Stromes I_{GOB} durch das Oxid nachdem ein Breakdown entstanden ist ($t > ttf_{GOB}$). Dessen Verlauf wird auch in diesem Kapitel thematisiert. Dieser Parameter ist deshalb von Bedeutung, da Modelle zur Simulation von Gateoxiddefekten diesen Stromfluss modellieren. Durch korrekte Steuerung von I_{GOB} über den zeitlichen Simulationsverlauf ist eine exakte Modellierung von integrierten Schaltungen mit mehreren Defekten möglich. Verwendete und weitere Transistormodelle folgen im Anschluss des Kapitels über die Modellierungsparameter.

3.6.1 Modellierungsparameter – ttf_{GOB} und I_{GOB}

Es herrschte Uneinigkeit darüber, ob der Stromfluss I_{GOB} nun einem linearen oder exponentiellen Anstieg über die Zeit unterliegt. Die exakte Bestimmung gestaltet sich schwierig, da das Rauschen die unterliegende Funktion maskiert. Betrachtet über den gesamten Verlauf wird ein exponentieller Anstieg postuliert, der sich innerhalb einzelner Dekaden von I_{GOB} als lineare Funktion darstellt [Lom05]:

$$I_{GOB}(t) = I_{GOB}(0) \cdot e^{\frac{t}{\tau_{GOB}}} \quad mit\ \tau_{GOB} \propto \frac{1}{DR} \tag{53}$$

wobei $I_{GOB}(0)$ der Stromfluss zu Beginn des Durchbruches zum Zeitpunkt $t = 0$ und τ_{GOB} die exponentielle Anstiegzeit sind. So können alle Phasen des progressiven Durchbruches mit einer Funktion abgedeckt werden.

Die Zeit bis zum Erscheinen des Durchbruches ttf_{GOB} ist von mehreren Stressfaktoren abhängig. Aus den vorigen Kapiteln ist ersichtlich, dass die Temperatur T und die Spannung V_{GS} großen Einfluss auf den Verschleiß des Gateoxides ausüben. Studien, die die

Lebensdauerprojektion zu einer Abhängigkeit zusammenfassen, die mit Hilfe einer Potenzfunktion beschrieben werden kann (siehe Gleichung (50)), stimmen mit Beobachtungen überein, wobei der spannungsabhängige Verschleißfaktor *AF* (engl. acceleration factor – AF) exponentiell von V_{GS} abhängt [McP07]:

$$AF = -\frac{\partial ln(ttf_{GOB})}{\partial V_{GS}} = \frac{n_{GOB}(T)}{V_{GS}} \tag{54}$$

wobei $n_{GOB}(T)$ den temperaturabhängigen Verschleißfaktor darstellt, da *AF* temperaturabhängig für eine konstante Spannung bzw. spannungsabhängig für eine konstante Temperatur ist. Wird nun Gleichung (54) nach V_{GS} differenziert, ergeben sich folgende empirischen Gleichungen [Wu02c]:

$$\frac{V_{GS}}{ttf_{GOB}} \cdot \frac{\partial ttf_{GOB}}{\partial V_{GS}} = n_{GOB}(T) \tag{55}$$

$$\frac{d}{dT}\left(\frac{1}{ttf_{GOB}} \cdot \frac{\partial ttf_{GOB}}{\partial V_{GS}}\right)_{MTTF_{GOB}} = 0 \tag{56}$$

wobei Gleichung (55) die differenzierte Form von Gleichung (54) darstellt, die den spannungsabhängigen Spannungsverschleiß beschreibt, während Gleichung (56) den temperaturunabhängigen Spannungsverschleißfaktor darstellt. Wenn nun für den Faktor $n_{GOB}(T)$ eine lineare Funktion in Abhängigkeit der Temperatur angenommen wird:

$$n_{GOB}(T) = a + b \cdot T < 0 \tag{57}$$

Es ergibt sich schließlich folgende Abhängigkeit der Lebensdauer von der Gatespannung:

$$ttf_{GOB} \propto {V_{GS}}^{a+bT} \tag{58}$$

Für die Temperaturabhängigkeit der Lebensdauer ergibt sich folgende Gleichung für eine konstante Spannung V_{GS}:

$$ttf_{GOB} \propto e^{\frac{c(V_{GS})}{T} + \frac{d(V_{GS})}{T^2}} \tag{59}$$

mit den spannungsabhängigen Konstanten *c* und *d*.

Obwohl noch über den Einfluss der Gateoxiddicke t_{OX} auf die Lebensdauer spekuliert wird und auch eine Beeinflussung des Weibullparameters β durch t_{OX} experimentell ermittelt wurde, wird in der Folge von der einfachen linearen Abhängigkeit ausgegangen, die sich auch mit Daten für die kritische Defektdichte vieler Studien deckt [Sta02] [Wu02a] und mit dem Perkolationsmodell korrespondiert [Deg95]:

$$ttf_{GOB} \propto t_{OX} \tag{60}$$

Aus dem Perkolationsmodell ergibt sich auch, dass die Wahrscheinlichkeit für einen Durchbruch mit steigender Gateoxidfläche $A_{GATE} = W_{EFF} \cdot L_{EFF}$ größer wird [Wu02a]:

$$\frac{ttf_{GOB}(A_1)}{ttf_{GOB}(A_2)} = \left(\frac{A_2}{A_1}\right)^{1/\beta} \tag{61}$$

wobei A_1 und A_2 die Gateoxidflächen von zwei Transistoren darstellen.

3.6.2 Transistormodelle des Gateoxiddefektes

Aus den vorigen Abschnitten ist ersichtlich, dass das Erstellen von Transistormodellen zur Simulation von Gateoxiddefekten nicht trivial ist. Dennoch gibt es eine Vielzahl an Studien zu diesem Thema, um damit das Verhalten von Schaltungen mehrerer auch defekter Transistoren quantitativ zu erfassen. In den 1990er Jahren entwickelte Segura ein Modell [Seg95], mit dem der harte Durchbruch gut nachgebildet werden konnte. Dabei wurde die Veränderung des Isolationsvermögens des Gateoxides mittels Einfügung von ohmschen Elementen nachempfunden, um zu testen, ob automatische Testmustergeneratoren diese Art von Defekten entdecken können. Defekte an verschiedenen Orten (Gate-Substrat, Gate-Drain, Gate-Source) wurden mit der Lage des parasitären Widerstandes modelliert. In Abbildung 3-6 ist das Modell für einen Gateoxiddefekt eines n-MOSFET zwischen Gate und Sourcegebiet (a) bzw. Substrat (b) respektive Draingebiet (c) dargestellt. Der ohmsche Kurzschluss R_{GOB} dient dazu, das Ausmaß des Defektes zu modellieren und teilt dabei den Transistor in zwei Hälften, wenn der Defekt zwischen Gate und Substrat auftaucht. Die Summe der Längen L_1 und L_2 beider serieller Transistoren entspricht der Länge des defektlosen Transistors [Seg95]:

$$L_{ORG} = L_1 + L_2 \tag{62}$$

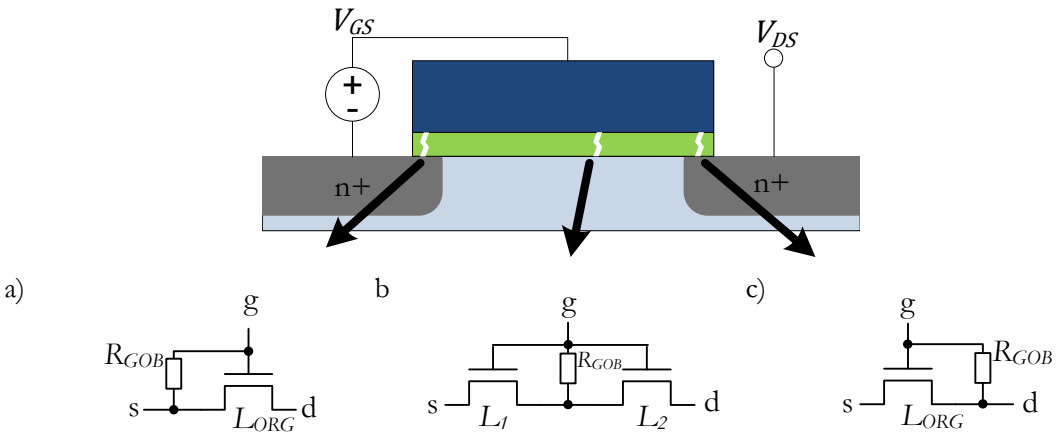


Abbildung 3-6 : Gateoxiddefektmodell von Segura [Seg95]:
a) GOB zwischen Gate und Source
b) GOB zwischen Gate und Substrat
c) GOB zwischen Gate und Drain

Um das Verhalten der Potenzfunktion aus Gleichung (50) gut nachzubilden, wurden bei [Rod03] spannungsabhängige Stromquellen zwischen Gate und Source und zwischen Gate und Drain eingefügt – Abbildung 3-7 a). Mit diesem einfachen Modell ist es möglich, den defektbedingten Leckstrom in die Schaltung einzufügen.

Zur Simulation von Gateoxiddefekten innerhalb eines Ringoszillators wurde ein Transistormodell von Kaczer entwickelt [Kac02]. Hierbei wurde ein Defekt mittels zweier zusätzlicher Transistoren T_S und T_D modelliert, die zwischen Gate und Drain bzw. Source eingefügt wurden, da die Kontaktregion des Defektes zwischen Gate und Substrat als Elektronensenke bei n-MOSFETs fungiert und somit ein zusätzliches Draingebiet in der Mitte des Kanals darstellt. Dieses Draingebiet und der Defekt an sich werden mit Hilfe der verbundenen Drains von T_S und T_D und einem Widerstand R_{GOB} nachgebildet. In Abbildung 3-7 b) ist die Umwandlung vom Querschnitt eines defekten Transistors zum elektrischen Modell ersichtlich. Außerdem wurden noch zwei ohmsche Widerstände R_S und R_D hinzugefügt, die den Widerstand in den Gate- und Sourceerweiterungen repräsentieren sollen. Durch Variierung der Transistorlängen L_S und L_D von T_S und T_D kann der Defekt im Gate positioniert werden.

Ein anderer Ansatz wurde in [Syr87] gewählt, bei dem ein Transistor in eine 2-D-Matrix von Transistoren gleichen Typs überführt wurde, um den Defekt zwischen den Transistoren zu platzieren – Abbildung 3-8. Die Länge der Transistoren der Matrix ist dabei minimal und die Breite wird so angepasst, dass die Strom-Spannungs-Kurven des Originaltransistors möglichst genau nachgebildet werden. Ein Defekt kann zwischen den Transistoren eingefügt werden, in dem ein Knoten des Gitters direkt über einen Widerstand R_{GOB} mit dem gemeinsamen Gate aller Transistoren verbunden wird. Je nach Lage und Größe des Widerstandes kann ein Defekt an verschiedenen Punkten mit unterschiedlichem Ausmaß im Gate simuliert werden. Hiermit werden die Reduzierung der maximalen Ströme I_{DSAT} und I_{GSAT}, sowie die negativen Werte des Drainstromes I_{DS} bei geringen Gatespannungen V_{GS} sehr gut modelliert – Abbildung 3-9 [Ren01].

a) b)

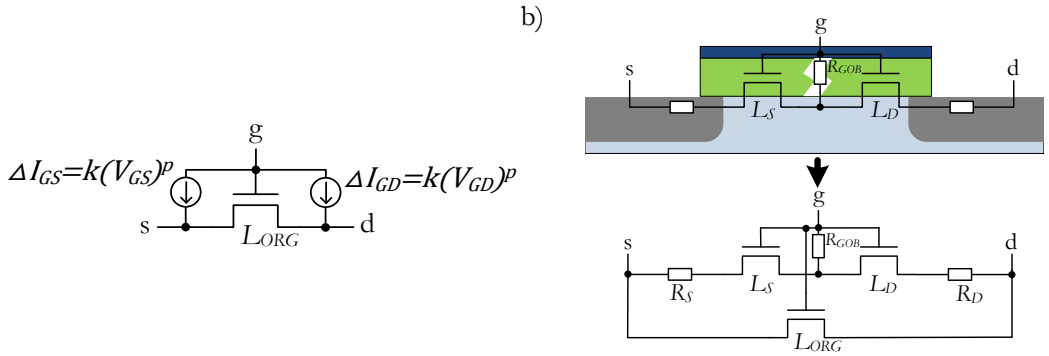


Abbildung 3-7 : Defektoxidmodelle
a) Mit eingefügten Stromquellen zur Modellierung defektbedingtem Leckstroms (p – Leckstrompotenz; k – Faktor zur Modellierung des Defektausmaßes)
b) Herleitung des Kazcer-Modells

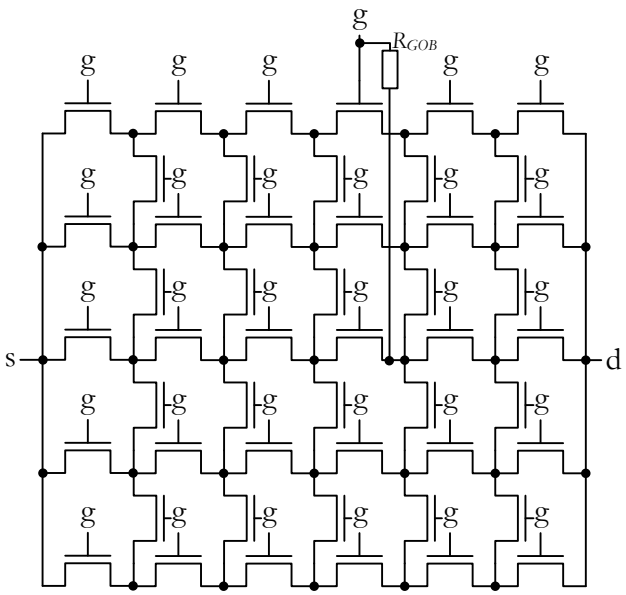


Abbildung 3-8 : Nichtlinear Split –Defektoxidmodell

Trotz der Genauigkeit des nichtlinearen Splitmodells [Ren01] sind zwei entscheidende Nachteile zu konstatieren. Zum einen sind Transistoren mit minimaler Länge, wie allgemein üblich in integrierten Schaltungen, nicht modellierbar, da sie nicht aufgespaltet werden können. Zum anderen erfordert eine Vielzahl von Transistoren eine vielfach größere Spice-Simulationszeit.

Um diesen Nachteilen zu entgehen, hat Renovell auf Basis des Splitmodells das „Non-Linear Non-Split MOS model“ entwickelt, was dem Kaczer-Modell ähnelt [Ren03a]. Die Nachbildung der Stromkurven defekter Transistoren wird dabei mit zwei zusätzlichen Transistoren, der Anpassung der Breite W_0 des Originaltransistors sowie einem ohmschen Widerstand R_{GOB}

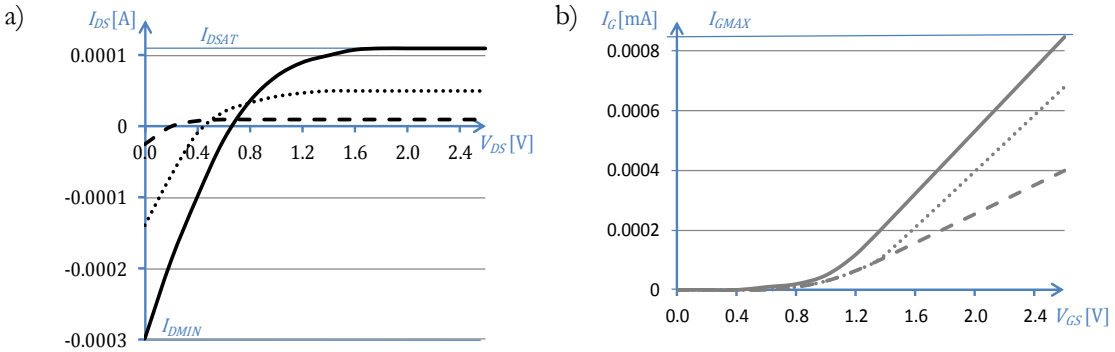


Abbildung 3-9 : Strom-Spannungskurven defekter n-MOSFET

a) Drainstrom I_{DS} in Abhängigkeit von V_{DS} für unterschiedliche Gatespannungen V_{GS} (schwarze Line – V_{GS} am größten, gestrichelte Linie – V_{GS} am kleinsten); Der Defekt verringert I_{DSAT} und senkt den Stromfluss I_{DS} (0) auf $I_{DMIN} < 0$

b) Gatestrom I_G in Abhängigkeit von V_{GS} für unterschiedliche V_{DS} (graue Linie – $V_{DS} = 0$, ansteigend zur gestrichelten Linie)

erreicht. Der Modellierungsprozess gliedert sich in vier Schritte. Als erstes wird die Breite W_0 auf W_M reduziert, damit $I_{DSAT\_DEFEKT} < I_{DSAT\_0}$ des defekten Transistors T_0 erreicht wird. Um nun einen negativen Drainstrom $I_{D\_DEFEKT} < 0$ bei geringer Drainspannung V_{DS} fließen zu lassen, wird ein Transistor T_A zwischen Gate und Drain des Originals eingefügt, dessen Gate und Drain mit dem Gate des gesamten Modells verbunden wird. Dadurch entsteht ein Stromfluss I_{D_DEFEKT} der sich aus den Drainströmen I_{DS_A} des Transistors T_A und I_{DS_M} des Transistors T_M ergibt, der den ursprünglichen Transistor T_0 mit veränderter Breite W_M darstellt:

$$I_{D\_DEFEKT} = I_{DS\_M} - I_{DS\_A} \tag{63}$$

Da bei einer Spannung von $V_{DS} = 0$ für einen n-MOSFET $I_{DS\_M} = 0$ ist, reicht es, das Verhältnis W_A/L_A des Transistors T_A für diese Spannung so anzupassen, dass es den Wert von I_{DMIN_DEFEKT} wiedergibt, da der Drainstrom bei diesen Spannungskonditionen nur von diesem Verhältnis abhängt. Um den Strom I_{G_DEFEKT}, der einen Strom vom Gate zur Source des Modells fließen lässt, korrekt widerzuspiegeln, ist ein weiterer Transistor T_B notwendig, der zwischen Gate und Source eingefügt wird, und dessen Gate und Drain mit dem Gate des Modells verbunden werden. Das Verhältnis W_B/L_B wird derart eingestellt, dass der maximale Strom I_{GMAX_DEFEKT} bei $V_{DS} = 0$ ebenfalls korrekt modelliert wird:

$$I_{GMAX\_DEFEKT} = I_{DS\_A} + I_{DS\_B} \; mit \; V_{DS} = 0 \tag{64}$$

Diese drei Schritte sind in Abbildung 3-10 a) bis c) noch einmal mit dazu gehörigen Strom-Spannungskurven dargestellt. Durch die Anpassung der Breiten W_A, W_B und W_M ist es möglich, Defekte an jeder beliebigen Stelle nachzubilden, wenn die dazugehörigen charakteristischen Werte bekannt sind (I_{DMIN_DEFEKT}, I_{GMAX_DEFEKT}). Der letzte Schritt umfasst nun die Modellierung des Ausmaßes des Defektes, d.h. des Widerstandes des Defektpfades vom Gate zum Substrat – Abbildung 3-10 d). Dazu wird der gemeinsame Knoten (engl. common node – CN in Abbildung 3-10 c)), in dem die Gates aller drei Transistoren und die beiden Drainanschlüsse der zusätzlichen Transistoren verbunden sind, in einen gemeinsamen Gateanschluss (engl. common gate – CG) und einen gemeinsamen Drain (engl. common drain – CD) aufgeteilt. CG umfasst nun alle Gateanschlüsse der einzelnen Transistoren, während der gemeinsame Drain des Modells (dem Punkt des Defektes im Substrat) durch die verbundenen Draingebiete der Transistoren T_A und T_B dargestellt wird. Der Widerstand des leitenden Pfades zwischen Gate und Substrat entspricht nun einem Widerstand R_{GOB} beliebiger Größe zwischen CG und CD (Abbildung 3-10 d)). Durch dieses nun vollständige Modell ist es möglich, die sich bildenden Stromflüsse innerhalb eines defekten Transistors mit beliebigem Defekt zwischen Gate und Substrat je nach den anliegenden Spannungen korrekt nachzubilden. Dieses Modell wurde auch in [Ren03b] benutzt, um eine Inverterkette mit Defekten nachzubilden. Dabei wurde analysiert, ob Delaytesttechniken die Gateoxiddefekte aufspüren können, was aufgrund des veränderten Zeitverhaltens möglich ist.

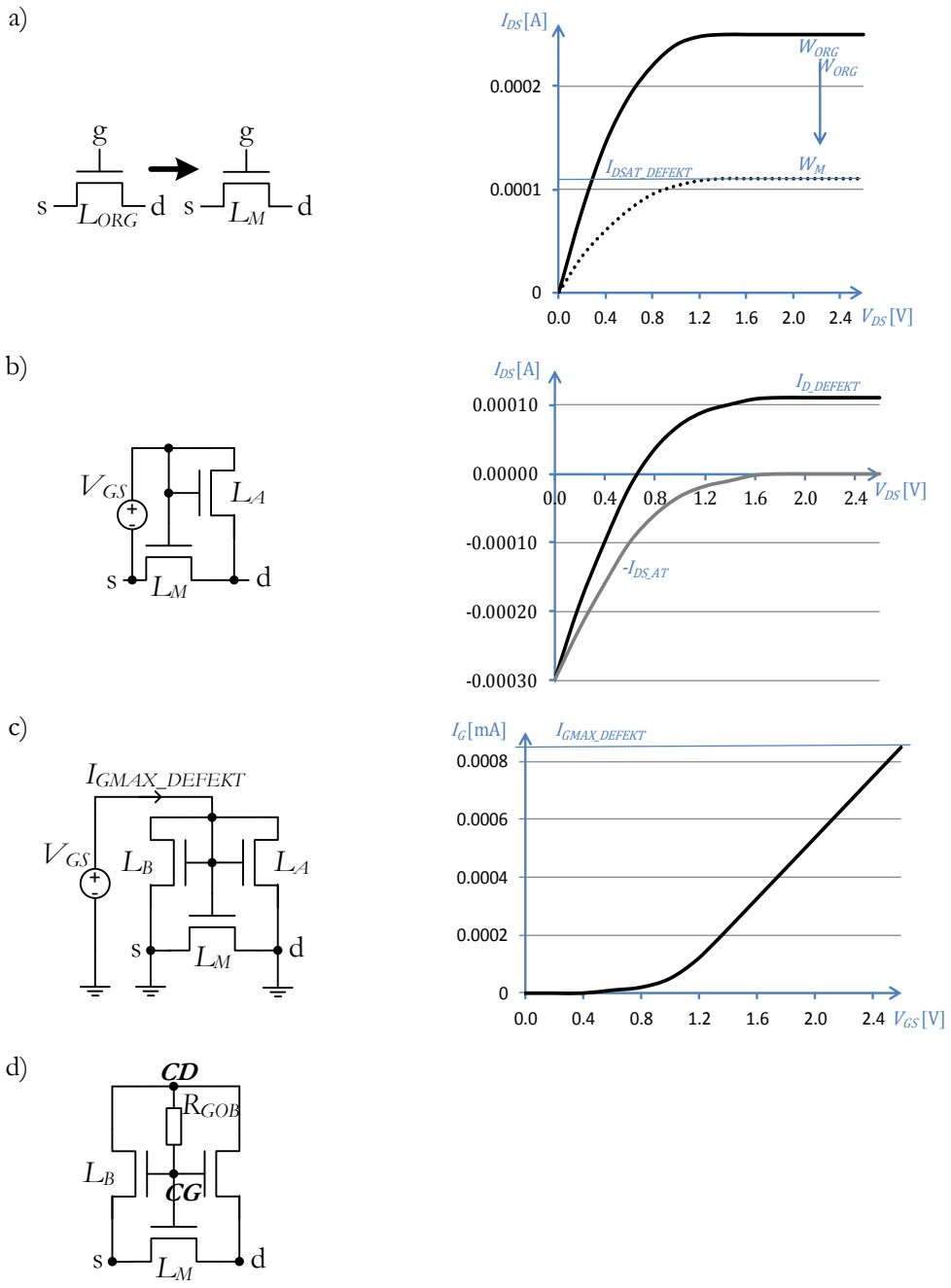


Abbildung 3-10 : Non-Linear Non-Split-Modell von Renovell

a) Reduzierung der Breite W_M des Originaltransistors zur Anpassung von I_{DSAT_DEFEKT}
b) Einfügen des Transistors T_A zur Anpassung von I_{DMIN_DEFEKT}
c) Einfügen des Transistors T_B zur Anpassung von I_{GMAX_DEFEKT}
d) Vollständiges Modell mit Widerstand R_{GOB} zwischen CG und CD zur Regelung des Defektstromes I_{GOB}, welcher das Ausmaß des GOB darstellt

3.7 Verwendete Simulationsmodelle und -einstellungen

Die Tabelle 3-2 gibt einen Überblick über die vorgestellten Modelle und eine Abschätzung, in wie weit die Modelle geeignet sind, um den progressiven Prozess des Gateoxiddefektes zu modellieren:

Tabelle 3-2 : Vergleich der vorgestellten Transistormodelle zur Modellierung von Gateoxiddefekten mit negativer (-), positiver (+) oder neutraler (o) Bewertung im Vergleich der Modelle zueinander

Modell nach:	Segura	Kaczer	Syrzycki	Renovell
Anwendung für das progressive Modell geeignet:	(-) Nur harter BD	(+)	(+)	(+)
Für alle Transistoren geeignet:	(+)	(+)	(-) nicht für L_{MIN}	(+)
Übertragbarkeit auf beliebige CMOS-Technologie:	(+)	(+)	(+)	(+)
Simulationsaufwand Spice (maximale # zusätzlicher Elemente je Transistor Tr):	(+) (1 R, 1 Tr)	(-) (3 R, 2 Tr)	(--) (1 R, x Tr)	(o) (1R, 2 Tr)
Übertragung der I-V-Kurve in die Gatterebene:	(-) Nur harter BD	(+)	(+)	(+)

Obwohl das Segura-Modell sehr einfach nachzustellen ist, fehlt aufgrund des nicht vorhandenen Soft Breakdowns ein entscheidendes Element, damit eine Schaltung mit sich verändernden Gateoxiddefekten über die Zeit simuliert werden kann. Allerdings sind die Simulationszeiten aufgrund des geringeren Mehraufwandes bei Spice-Simulationen geringer als bei den anderen Modellen. Daher wurde dieses Modell für erste Untersuchungen verwendet, bei denen die Auswirkungen von harten Durchbrüchen analysiert wurden [Sae09a]. Ein weiterer Nachteil gegenüber den anderen Modellen ergab sich aus der gewollten Übertragung defekter Transistoren in die Standardgatter der Gatterebene. Aus Simulationen mit den anderen Modellen hätte man im Gegensatz zum Segura-Modell die für das CCS-Modell benötigten zeitlich veränderbaren Strom-Spannungswertepaare nutzen können. Die Wahl fiel allerdings auch nicht auf das 2D-Modell, weil eine umfangreiche Charakterisierung der vorhandenen CMOS-Technologie nicht möglich gewesen wäre, da die meisten Standardgatter Transistoren mit minimaler Länge L_{MIN} verwenden. Durch die sehr gute Anwendbarkeit für beliebige CMOS-Bibliotheken diente es als Vorlage für die Verwendung des Renovell-Modells. Dies und der etwas geringere Mehraufwand durch zusätzliche Transistoren und Widerstände führte zum dem Entschluss, das Renovell-Modell anstatt das Kazcer-Modells für diese Arbeit zu verwenden.

So wurde eine Bibliothek angefertigt, die Gateoxiddefekte über den Umweg der 2-D-Gitter-Modelle charakterisiert. Auf diesem Weg war es möglich, eine gute und genaue Anpassung der vorhandenen 65-nm-CMOS-Bibliothek von STMicroelectronics durchzuführen. Dazu wurden zuerst nicht defekte Transistoren der Bibliothek, mit nicht minimalen Längen L in ein 5x5 2-D-Gitter aus Transistoren gleichen Typs überführt. Um nun Defekte an verschiedenen Stellen im

Gate zu modellieren, wurden an den einzelnen Knotenpunkten des Gitters Widerstände R = 0 Ω zwischen dem Gate und dem Knotenpunkt eingefügt. Die entstandenen Strom-Spannungskurven wurden nach dem Beispiel Renovells in ein Defektmodell, wie im vorigen Abschnitt beschrieben, überführt. Nach [Ren03a] konnten dann die Verhältnisse W_M/W_0, W_A/W_0 und W_B/W_0 für die Transistoren mit minimalen Transistorlängen übernommen werden, um so die Transistoren der Standardgatter der Bibliothek mit eventuellen Defekten gezielt zu manipulieren. Die Entwicklung des Stromes I_{GOB} wurde mit Hilfe des Widerstandes R_{GOB} modelliert, wobei am Anfang des Durchbruches ein zufällig großer Widerstand $R_{GOB} > 1$ MΩ zugewiesen wurde, um einen Startpunkt für den progressiven Breakdown zu setzen, der sich dann mit Gleichung (53) entwickelt.

Je nach Phase der Badewannenkurve wurde $\beta = 1$ (Phase 2, Defektrate ist konstant) und $\beta = 1.4$ (Phase 3, Defektrate steigt an) gesetzt. Letzterer Wert ergibt sich aus der konstanten Gateoxiddicke $t_{OX} = 1.3$ nm der vorhandenen Bibliothek [Wu02a] [Sta02] [Deg95]. Für Simulationen auf RTL-Ebene (Kapitel 4.1) wurden Stuck-at-Fehler eingefügt, um den ungünstigsten Fall hinsichtlich des Defektverhaltens anzunehmen. Hierbei werden einzelne Netze ab dem Zeitpunkt des Einfügens permanent auf Masse V_{SS} (logischer Wert von 0: Stuck-at-0) oder auf den Wert der Versorgungsspannung V_{DD} (logischer Wert von 1: Stuck-at-1) gesetzt. Das Einfügen vollzieht sich dabei zufällig mit vorgegebener Defektrate, die für alle Transistoren gleich war. Für Spice-Simulationen wurde der Eintritt des Defektes mit Hilfe der Gleichungen (58), (59), (60) und (61) relativ berechnet. Ausgangspunkt war jeweils ein Transistor, der die maximale Fläche aller genutzten Transistoren einnimmt, so wie permanent der maximalen Spannung V_{GS} ($V_{GS} = V_{DD}$ für n-MOSFET, $V_{GS} = V_{SS}$ für p-MOSFET) ausgesetzt wurde. Ihm wird die minimale Lebensdauer $ttf_{GOB} = 1$ zugewiesen. Dieser Wert stellt den Minimalwert für ttf_{GOB} dar. Davon ausgehend wurden die aktuellen Werte des Transistors in Relation zu den erreichbaren Minimalwerten gesetzt und ttf_{GOB} für jeden Transistor individuell berechnet. Aus diesem Grund sind alle später folgenden Zeitskalen relativ und dimensionslos, wobei Defekte ab dem Zeitpunkt $t > 0$ auftreten können. Mit Hilfe von ttf_{GOB} und einem Zufallswert war es dann möglich, für jeden Durchlauf jedem Transistor einen Wert $t = t_{GOB}$ zuzuordnen, bei dem der Verschleiß beginnt und dann mit Hilfe von Gleichung (53) seinen Verlauf nimmt. Die Anpassung der Werte für V_{GS} wurden mit Hilfe der Aktivitätsdateien erstellt, da sich aus ihnen, die Auswirkung der Wahrscheinlichkeit eines high- oder low-Pegels auf ttf_{GOB} aus Gleichung (58) ergibt. So ergibt sich eine lineare Abhängigkeit der Lebensdauer von der Wahrscheinlichkeit P_{ON}, mit der ein Transistor während der Betriebszeit durchgeschaltet ist:

$$ttf_{GOB} \propto P_{ON} \tag{65}$$

3.8 Fehlervermeidung und Verschleißverzögerung

In den vorigen Kapiteln wurden die Grundlagen von zuverlässigkeitssenkenden Effekten transienter und permanenter Defekte behandelt. Dieses Unterkapitel gibt einen Überblick, welche Maßnahmen publiziert wurden, um deren Einfluss auf integrierte Schaltungen so gering wie möglich zu halten und logische Fehler möglichst zu vermeiden. Im zweiten Abschnitt werden konkrete veröffentliche Ansätze zur Korrektur und Vermeidung von permanenten und flüchtigen Defekten vorgestellt. Dabei wird ersichtlich, dass Redundanz in vielerlei Form angewandt wird, um Fehler zu vermeiden, da somit Defekte entweder detektiert und korrigiert, als auch maskiert werden. Da auch in dieser Arbeit diverse Formen der Redundanz präsentiert werden, um Verschleißeffekte zu minimieren bzw. Fehler hinauszuzögern, liegt der Fokus der vorgestellten Ansätze auf Verbesserungen, die mit redundanten Modulen erreichbar sind.

3.8.1 Übersicht – Redundanz

Die sogenannte Hardware-Redundanz wird auf Architektur-, RTL-, Gatter- bzw. Transistorebene auf zweierlei Arten verwendet – als Mittel zur Fehlererkennung und zur Fehlermaskierung [Sie91]. Bei der Fehlererkennung wird eine Schaltung einmal verdoppelt und die Ausgänge beider Teileinheiten werden mit einer Vergleichseinheit verknüpft (engl. duplication with comparison – DwC). Sollten Abweichungen zwischen den Ergebnissen beider Schaltungen auftreten, werden diese durch die Vergleichsschaltung detektiert und über ein Fehlersignal gemeldet – Abbildung 3-11 a). Wird die Schaltung mehr als nur verdoppelt, können Fehler in einer der Einheiten durch die anderen maskiert werden. Diese Methode wird als n-modulare Redundanz bezeichnet [Kor07]. Hierbei werden die Ausgänge der redundanten Einheiten in einer Entscheidereinheit (engl. voter) verknüpft. Als Gesamtergebnis wird der Mehrheitsentscheid der Ergebnisse der Teileinheiten zur Weiterverarbeitung geleitet. Um den Flächenaufwand so gering wie möglich zu halten, wird oft die dreifach modulare Redundanz (engl. triple modular redundancy – TMR) verwendet [Von56]. Hierbei ist die Originalschaltung verdreifacht und das Ergebnis wird anhand einer 2-aus-3-Mehrheit entschieden – Abbildung 3-11 b). Deshalb eignet sich dieser Ansatz sehr gut zur Maskierung von Soft Errors. Aber auch Produktionsfehler [Via08]

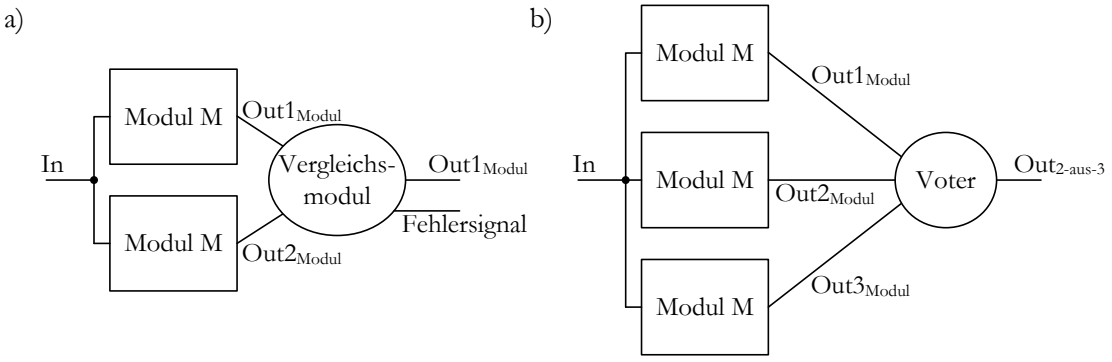


Abbildung 3-11 : Grundformen der Hardware-Redundanz
a) DwC zur Fehlererkennung
b) TMR zur Fehlermaskierung als populärste Form der n-modularen Redundanz

und permanente Defekte werden bis zu einem gewissen Grad nicht weitergeleitet, immer in Abhängigkeit von der Fehleranzahl, der Anzahl der betroffenen Einheiten und davon wie viele und welche Ausgänge der Teileinheiten unkorrekt berechnet werden.

Aus Sicht der Fehlertoleranz ist die Entscheidereinheit der kritischste Teil des Konstrukts. Zahlreiche Ansätze zur Verbesserung des TMR fokussieren sich darauf, z. B. indem das Entscheidungsschema verändert [Mit00] oder die Einheit als sich selbst prüfender Block implementiert wird [Gai88]. Hauptnachteil bei allen redundanten Techniken ist der erhöhte Flächenaufwand und die daraus resultierende gesteigerte Stromaufnahme, was sich durch die gesteigerte Wärmeentwicklung wiederum negativ auf die Zuverlässigkeit auswirkt. Um den Flächenaufwand zu reduzieren, wurden diverse Techniken der zeitlichen Redundanz entwickelt. Das Grundprinzip hierbei ist, dass die Berechnungen, die in den zusätzlichen Einheiten der vorgestellten Flächenredundanz stattfinden, durch mehrmaliges Wiederholen in einem Modul durchgeführt werden [Kai00]. Somit sind flüchtige Fehler und Fehler in Registerstufen, die die Zwischenergebnisse speichern, detektier- und korrigierbar. Permanente Fehler und damit auch Verschleißeffekte in den Berechnungseinheiten sind dagegen nicht erkennbar mit dieser Form der Redundanz. Eine weitere Form ist die Code-Redundanz. Hierbei sind zumeist die Eingangs- bzw. Ausgangsvektoren mit zusätzlichen Bits versehen. Zur Fehlererkennung reicht ein Bit für die Paritätsberechnung, wobei mehrere Bits die Erkennungsleistung erhöhen. Zur Fehlerkorrektur sind dann mehrere Bits notwendig. Nachteil dieser Form der Redundanz sind der zusätzliche Aufwand in Fläche, Leistungsaufnahme und Verzögerungszeit durch die zusätzlichen Signalleitungen und die Module, die zur De-/Kodierung notwendig sind.

3.8.2 Konkrete Ansätze zur Defektvermeidung und -maskierung

Klassischerweise ist die Steigerung der Ausbeute eine Domäne der Materialforscher und der Produktionsingenieure. Allerdings gibt es auch im Schaltungsentwurf Techniken, um die Ausbeute zu erhöhen. So gibt es für das Layout von integrierten Schaltungen das so genannte Design for Manufacturability/Yield (DFM/DFY) im Anschluss an das Layout. Hierbei wird die Ausbeute des erstellten Layouts mittels Anpassungen, die nicht oder sehr geringfügig die Parameter der Originalschaltung ändern, erhöht, weil hier auf die Produktionsschritte und deren potentielle Schwachstellen bzw. Schwierigkeiten Rücksicht genommen wird. Maßnahmen, die zum DFM zählen, umfassen u.a. die Verkleinerung der critical area, die Anpassung der Leitungsbreiten und -abstände und das Einfügen von Dummy-Feldern für eine ebenere Fläche nach dem CMP, das Vervielfachen von einzelnen Vias und die Korrektur von Abbildungsfehlern beim fotolithografischen Prozess durch Veränderung der Strukturumrisse (engl. optical proximity correction – OPC).

Ferner enthielten integrierte Schaltungen mit regulären Strukturen, wie Speicher, schon in den 1980ern redundante Zellreihen von Speicherelementen, um defekte Zellen auszutauschen. Die An- und Abschaltung erfolgt dann mittels Sicherungen (engl. fuses) [Moo86]. Diese Methode wird heute mehr denn je eingesetzt. Bezogen auf den Logikentwurf mit wenigen regelmäßigen Strukturen, sind die meisten Verfahren zur Ausbeutensteigerung auf die funktionale

Defektvermeidung ausgelegt. In der Studie von Chen [Che95] werden die beiden Standardtechniken Rekonfiguration und Layoutmodifikation anhand eines Addiererentwurfes eingesetzt. Rekonfiguration meint dabei den Einsatz von zusätzlichen Bit-Slices, wobei mittels Laser fehlerhafte Slices abgetrennt bzw. die funktionierenden verbunden werden. Layoutmodifikationen sind Veränderungen des fertigen Entwurfs zur Erhöhung der Ausbeute bzw. Zuverlässigkeit. Dazu zählen beispielsweise das Duplizieren der Vias und die Verbreiterung von Signalleitungen, ähnlich dem DFM, sowie das Verdichten von Flächen oder das erneute Zuordnen von Metall-Ebenen [Ci95]. Des Weiteren können Floorplanmodifikationen besonders bei Verwendung von redundanten Modulen die Ausbeute zusätzlich vergrößern [Haz03].

Dem Prinzip der Redundanz folgt auch der Ansatz von Sirisantana [Sir04], allerdings auf Transistorebene. Hierbei werden Transistoren parallel zum Originaltransistor eingefügt, sofern deren Einsatz bestimmten Vorgaben an die Verzögerungszeit, Leistungsaufnahme und Fläche der Schaltung nicht entgegensteht. Da ein eingefügter Transistor, wie in Abbildung 3-12 b) zu sehen, die Lastkapazität für vorige Gatter erhöht, wurden dort Transistoren eingefügt, wo noch zeitlicher Spielraum (engl. slack) vorhanden war. Derart eingesetzt wurde die Verzögerungszeit der Gesamtschaltung nicht erhöht. Die experimentellen Ergebnisse ließen eine Yieldverbesserung von bis zu 400 % bei den größten Designs (c7552) erkennen. Bei kleineren Schaltungen lag sie bei 50 %, allerdings waren die verbesserten Ergebnisse deutlich näher an einer Ausbeute von 100 % als die ursprünglichen. Das Defektmodell ging dabei von zufälligen Produktionsdefekten aus, die den Transistor Stuck-open Fehler produzieren lassen und somit keinen Einfluss auf den logischen Pegel nehmen, wenn der Transistor defekt ist. Dadurch bleibt die logische Funktion bei vorhandenen Defekten gewährleistet, das Zeitverhalten wird aber durch den Ausfall von treibenden Transistoren negativ verändert. Hierzu bedarf es weiterer Ansätze auf der Systemebene, um zum einen die Timing-Fehler zu detektieren, und zum anderen diese zu kompensieren. Für die Anpassung der Verzögerungszeiten eignen sich hierfür unter anderem Dynamic Frequency/Voltage Scaling (DV/FS) oder Body Biasing [Tsc02].

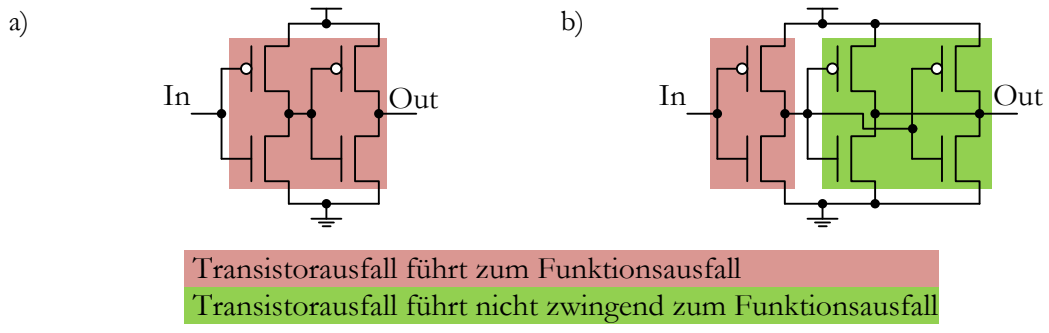


Abbildung 3-12 : Ansatz Sirisantanas zur Erhöhung der Ausbeute [Sir04]
a) Originalschaltung, bei der ein Ausfall eines der Transistoren zum Systemausfall führt
b) Verbesserte Schaltung, bei dem der zweite Inverter verdoppelt wurde, so dass dieser einen Ausfall des Transistors kompensiert; Allerdings erhöht sich t_D für den vorigen Inverter aufgrund der erhöhten C_{LOAD}

Zur Reduzierung parametrischer Defekte sind mittlerweile auch einige Techniken entwickelt worden, um Chips, die sonst nach der Produktion verloren gehen würden, noch nutzen zu können. So können spezielle Messmodule, wie die Modified Vernier Delay Line, genutzt werden, um Daten über das exakte Systemverhalten zu erhalten [Dat06]. Aufbauend auf solchen Kenndaten kann dann beispielsweise mittels Body-biasing die Frequenz und der Leckstrom des Systems so angepasst werden, dass es den Anforderungen genügt. Dies gelingt dann durch Variation des Schwellwertes der Transistoren [Sil07].

Operative, also transiente Effekte werden vor allem mittels Ansätzen in unteren Designebenen, wie entkoppelnden Kondensatoren [Hey03] und entkoppelnden Leitungsanordnungen [Ho03], angegangen, um die parasitären Einflüsse zu mindern. Leckströme werden durch die Verwendung von Transistoren mit unterschiedlichen Oxidschichtdicken oder Schwellspannungen, sowie durch Umordnen der Transistorgruppierung, z. B. durch Stacking, und durch den Einsatz von so genannten Sleep-Transistoren gesenkt [Sil07]. Die Auswirkungen von Umgebungsvariationen werden derweil auf System- und Architekturebene gemindert, wie z. B. das Kühlen des Chips oder eine dynamische Anpassung der Last zur Kontrolle der Temperatur und des Stromverbrauches [Hua00].

Da ein Forschungsschwerpunkt in den letzten Jahren auf der Vermeidung bzw. Verhinderung von Soft Errors lag, gibt es hier eine Reihe von Lösungsansätzen in allen Entwurfsebenen. So werden Gatter mittels Tiefpassfilter [Sas06] oder Erhöhung der kritischen Ladung anfälliger interner Knoten robuster gemacht [Oma07]. Soft Errors können auch durch die Verwendung fehlerkorrigierender Codes maskiert werden [Spi04]. Zur Erkennung von transienten Fehlern gibt es auch verschiedenste Ansätze. Sie reichen von der Anwendung fehlererkennender Codes, wie z. B. der Bildung von Paritäten, bis zum Einsatz von Überwachungsmodulen, wie beispielsweise Watch-Dog-Prozessoren [Mah88]. Software stellt dann wieder den korrekten Zustand vor dem Auftreten des Fehlers her, z. B. durch die wiederholte Ausführung von Instruktionen (Rollback) [Pat02].

Ein anderer sehr viel versprechender Ansatz mittels Redundanz wurde in [Mit05] veröffentlicht. Hierbei werden die Flipflops einer Schaltung verdoppelt und der Ausgang beider Speicherelemente mit einem Muller-C-Element stabil gehalten. Da die Kopie jedes Flipflops durch das schon vorhandene Flipflop im Scanpfad bereitgestellt wird, bedarf es nur einer geringfügigen Erweiterung der Logik und dem Einfügen des Muller-C-Elementes (Abbildung 3-13). Damit stellt dieser Ansatz durch die geringen zusätzlichen Kosten und einer Erhöhung der Zuverlässigkeit auf Chipebene um mehr als eine Größenordnung eine effiziente Möglichkeit zur Verringerung von Soft Errors dar.

Ein Ansatz, permanenten Defekten zu begegnen, ist sie erst gar nicht entstehen zu lassen. Es existieren verschiedene adaptive Systeme, die den Betrieb derart steuern sollen, dass ein permanenter Defekt gar nicht erst während der Betriebslaufzeit auftritt. Dazu zählen unter anderem das Dynamic Temperature- und insbesondere das Dynamic Reliability Management (DTM [Bro01] und DRM [Sri03]). Solche Ansätze verwenden physikalische Parameter, wie z. B. die Temperatur, um Rückschlüsse auf die Betriebslaufzeit zu ziehen bzw. die Leistung zu

beschränken, um die Betriebslaufzeit garantieren zu können. Beispielsweise kann in einem Multi-Prozessor-System die arbeitende zentrale CPU auf einen anderen Rechenkern verlagert werden oder Prozesseinheiten werden mittels DFS verlangsamt [Weg11], um dadurch sowohl die Maximaltemperatur als auch die Temperaturverteilung zu verbessern.

Neben der Defektvermeidung sind Ansätze zur Defektbeseitigung bzw. Defekttoleranz/-maskierung mittels redundanten Elementen bekannt. Aus dem Bereich des Yield Engineering existieren Ansätze auf den unteren Designebenen, die in gleichem Maße auch Defekte vermeiden. Dazu gehört zum Beispiel das Einfügen redundanter Vias [Wu02c] oder Verbindungsleitungen. Dadurch kann ein Signal im Falle des Ausfalls eines Vias (bzw. einer Leitung) immer noch übertragen werden. Weitere Ansätze der Redundanz sind in komplexen integrierten Systemen bereits implizit vorhanden, werden aber bislang noch nicht effektiv zur Verlängerung der Laufzeit genutzt. So stehen sowohl Rechen- als auch Kommunikationsressourcen in Multi-Prozessor-Systemen bzw. in Networks-on-Chip zur Verfügung. Durch diese Art der Redundanz können Fehler allerdings nicht vermieden, wohl aber toleriert werden. Dazu bedarf es wiederum einer geeigneten Systemkontrolle um Anwendungen gezielt auf die verbleibenden funktionalen Komponenten zu verteilen [Hun04] bzw. Daten auf die intakten Kommunikationsressourcen zu übertragen [You07].

Auf RTL- bzw. Gatterebene ist Rekonfiguration bzw. Built-in-Self-Repair (BISR) eine Art der Maskierung von Defekten. In [Koa09] wurde ein Ansatz vorgestellt, in dem eine unbestimmte Ansammlung an Logikzellen zu konfigurierbaren Logikblöcken zusammengefasst werden, die vervielfacht und über eine Schaltmatrix miteinander verbunden bzw. an- und abgeschaltet

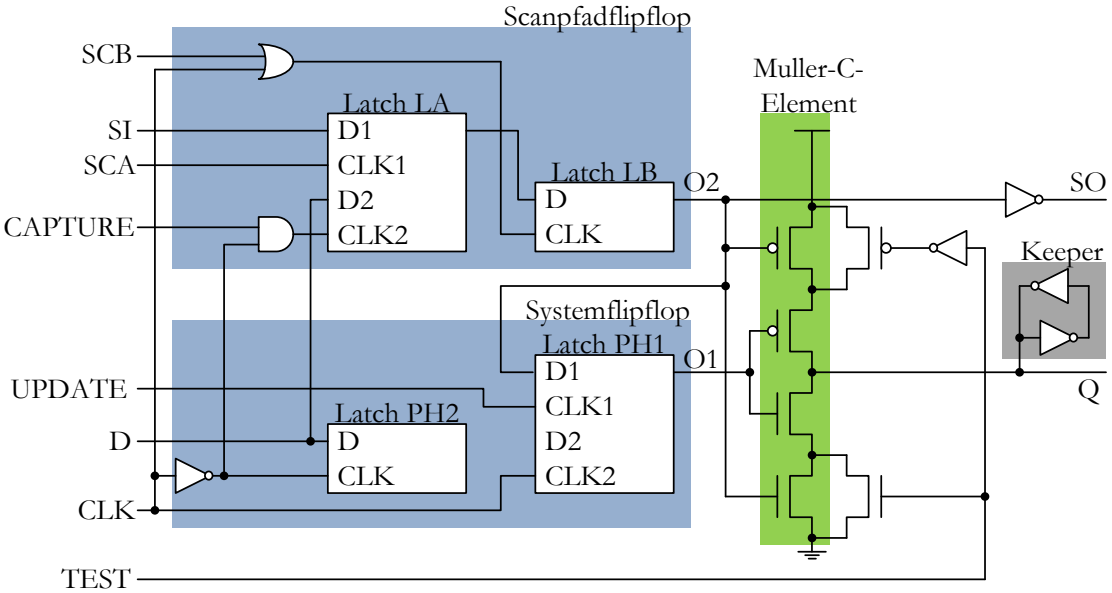


Abbildung 3-13 : Soft-Error-Blockierung bei Verwendung des Scanpfadflipflops [Mit05]
Normaler Betrieb: SCA = SCB = UPDATE = TEST = 0; CAPTURE = 1;
Scanpfad-FF wird zum Schatten-Flipflop
solange O1 ≠ O2 wird kein Wert weiter gereicht und der Keeper hält Q stabil

werden. Kritisch hierbei ist der Leitungs- und Schaltaufwand, der mit Verwendung weniger unterschiedlicher Standardgatter verringert werden kann. Zudem wurden Gattersynthesestrategien vorgestellt, mit denen eine ungeordnete Netzliste in reguläre Strukturen umgewandelt wird und die Anzahl an Schalttransistoren durch Verwendung größerer regulärer Blöcke verringert werden kann.

Ein Ansatz, der die einfache Verdopplung von Transistoren zur Ausbeutenerhöhung von Sirisantana auf Verschleißdefekte erweitert, ist die Verwendung von Shadow-Transistoren [Cor08]. Diese werden nach Wahrscheinlichkeit eines Defektes aufgrund von TDDB ausgesucht. Als Defektmodell wurde Segura's Modell verwendet, so dass im Gegensatz zu Sirisantana defekte Transistoren noch einen Einfluss auf den Logiklevel der angeschlossenen Netze ausüben. Bei einem zusätzlichen Bedarf an 20 % Fläche, 11 % Verzögerungszeit und 14 % Leistungsaufnahme wurde die *MTTF* um durchschnittlich 42 % erhöht, mit 59 % als Maximum.

Ziel der vorgestellten Ansätze ist es, einen frühzeitigen und vollständigen Ausfall des Systems zu vermeiden. Stattdessen erreicht man eine schleichende Verminderung der Systemperformance, weil immer mehr Ressourcen nicht mehr zur Verfügung stehen. Mit Blick auf die ganzheitliche Zuverlässigkeit ergibt sich ein weiterer wichtiger Vorteil dieser Schaltungen. Dieser besteht darin, auch produktionsbedingte Ausfälle automatisch kompensieren zu können. Sowohl die parametrische als auch die funktionale Ausbeute können aufgrund der redundant ausgelegten Elemente erhöht werden, wenn auch mit einer geringeren Performance im Defektfall.

Viertes Kapitel

4 Redundanz – Verbesserung der Zuverlässigkeit auf Gatterebene als Designziel

Das vorherige Kapitel hat sowohl die Gründe für eine Berücksichtigung von Zuverlässigkeitsaspekten im Designablauf hinsichtlich Gateoxiddefekten erläutert, als auch eine Reihe von Möglichkeiten aufgezeigt, integrierte Schaltungen schon im Designprozess robuster gegen diverse Defektmechanismen zu gestalten. Dabei hat sich gezeigt, dass auf unteren Schaltungsebenen die Hardware-Redundanz einen großen Stellenwert einnimmt, um die Zuverlässigkeit in integrierten Schaltungen zu erhöhen, wodurch der Designer Einfluss auf die Lebensdauer der produzierten Schaltung nehmen kann. In den folgenden Abschnitten werden diverse Ansätze entwickelt, um durch Redundanz die Zuverlässigkeit gegenüber Verschleißeffekten zu steigern.

Als erstes wurden komplett redundante Schaltungen untersucht, d.h. die Schaltung wurde in Gänze vervielfacht, und in der Hinsicht verbessert, dass die einzelnen Teile unabhängig voneinander operieren können, wodurch verglichen mit den redundanten Originaldesign sowohl die Zuverlässigkeit erhöht, als auch der Stromverbrauch gesenkt werden konnte. Anschließend wird eine Analyse vorgestellt, mit der untersucht wird, mit welcher Granularität der Redundanz die beste Zuverlässigkeitssteigerung erreicht werden kann. Ausgehend von dem Ergebnis, dass mit redundanten Gattern sehr gute Betriebszeiten erreicht werden können, wurde dieser Ansatz mit Einbeziehung von lokalen Gatterinformationen wie Gatefläche und Aktivität verfeinert, so dass trotz gesunkenem Flächenaufwand durch Redundanz die Zuverlässigkeitssteigerungen nahezu beibehalten werden können. Dabei wurden auch Elemente eingefügt, mit denen redundante Teile erst nach einer unbestimmten Betriebszeit zugeschaltet werden, so dass sich Verschleißeffekte bei den redundanten Transistoren erst nach der Zuschaltung entwickeln. Zum Abschluss wird eine Untersuchung auf Transistorebene diskutiert, die mehrere Ansätze vergleicht. Der Fokus dieser Verbesserungen liegt klar bei der Verminderung bzw. Verzögerung

von Verschleißeffekten. Allerdings können durch die gezeigten Ansätze natürlich auch Fehler permanenter Produktionsdefekte verhindert werden, so dass auch eine Erhöhung der Ausbeute stattfindet.

4.1 Energieeffiziente Schaltungsredundanz – Funktionale Verbesserungen der Triple Modular Redundancy

Das Funktionsprinzip des TMR wurde schon im Abschnitt 3.8.1 vorgestellt. Da die Leistungsaufnahme eine immer wichtigere Rolle spielt, auch in Hinblick der Verringerung der Zuverlässigkeit durch höhere Temperaturen, werden im Folgenden zwei aufeinander aufbauende Varianten des TMR vorgestellt, um diese Verlustleistung zu senken mit Berücksichtigung der Zuverlässigkeit gegenüber permanenten Fehlern. Anders als die vorgestellten Ansätze in [Zhu04] und [Eln02], in denen TMR- und Duplex-Systeme früh auf höheren Ebenen optimiert werden, fokussieren sich die nachfolgenden Verbesserungen auf die Gatterebene, so dass sie einfach in den vorgestellten Syntheseprozess zu integrieren sind.

Eine Methode, um die dynamische Verlustleistung zu verringern, ist der Gebrauch von parallelen Datenpfaden [Cha92]. Dabei werden Datenpfade ebenfalls redundant implementiert. Beide Pfade werden allerdings so genutzt, dass jeweils das Ergebnis eines Pfades an den Ausgang geleitet wird, während der andere Pfad noch einen Takt Zeit hat die Berechnung zu beenden. Dadurch lässt sich die Operationsfrequenz der Gesamtschaltung halbieren, ohne dass sich der Durchsatz verringert. Im Vergleich zur einfachen Originalschaltung wird allerdings in dieser Variante noch keine Verlustleistung eingespart, da zwar f_{CLK} aus Gleichung (24) halbiert wird, derweil die Lastkapazität C_{LOAD} aufgrund der verdoppelten Fläche auch verdoppelt wird. Einsparungen sind so nur möglich, wenn auch die Betriebsspannung V_{DD} gesenkt werden kann, da die Operationsfrequenz verringert werden kann und somit auch V_{DD} nach (20). Wichtig bei dieser Methode ist die korrekte Implementierung des Verteilungsmechanismus.

Ausgehend von einem redundanten Design, z. B. einer TMR-Schaltung, kann mittels der schon vorhandenen dreifach vorhandenen Datenpfade und einer geeigneten Verknüpfung und Aufteilung der Berechnungsschritte die Verlustleistung verringert werden, ohne dass die Betriebsspannung gesenkt werden muss. Im Folgenden wird ein Ansatz vorgestellt, der diese Tatsache ausnutzt, um die Verlustleistung redundanter kombinatorischer Schaltungen zu reduzieren und gleichzeitig die Zuverlässigkeit gegenüber Verschleißdefekten zu erhöhen.

Grundlage der Schaltung ist eine TMR-Implementierung. Zwei verschiedene Ansätze sind denkbar. Zum einen wird ständig zwischen einem TMR-Modus zur Fehlererkennung und einem Einzelmodus mit reduzierter Frequenz gewechselt (Abschnitt 4.1.1). Sollten Fehler auftreten, wird die Schaltung im normalen TMR-Modus betrieben. Verschiedene Varianten für den besten Kompromiss zwischen Fehlererkennung und Verlustleistungsreduzierung wurden getestet. Der zweite Ansatz (Abschnitte 4.1.2 bis 4.1.5) basiert auf dem ersten, ist durch Einführung weiterer Modi und Phasen allerdings flexibler und stellt mehrere finale Strategien vor, mit der die

Gesamtschaltung bei Erkennen fehlerhafter Module bis zum Ende betrieben werden kann. Dadurch werden sogar die durch Verschleiß hervorgerufenen Effekte verlangsamt, was im Endeffekt eine erhöhte Betriebszeit der Gesamtschaltung zur Folge hat. Dieser Ansatz arbeitet abwechselnd mit zwei der drei Module in zwei Phasen: einer Parallelphase mit reduzierter Frequenz und einer Vergleichsphase zur Erkennung von Fehlern. Der TMR-Modus mit Teilnahme aller Module wird genutzt, um fehlerhafte Einheiten zu lokalisieren, sowie als finaler Modus, wobei auch der Betrieb nur mit einem Modul Berücksichtigung findet.

Aufgrund eines festgelegten Wechsels von Phasen, in der die Module mit gleichen Eingangsvektoren zur Fehlererkennung und -lokalisation arbeiten, und Phasen, in denen die einzelnen Module unterschiedliche Vektoren bearbeiten, verlieren diese Implementierungen die Fähigkeit, über den gesamten Zeitraum des „normalen" Betriebs flüchtige Fehler zu maskieren. Allerdings werden permanente Fehler entdeckt und lokalisiert. Durch den festen Wechsel der Phasen wird es für höhere Betriebsebenen bei Auftreten eines Fehlers leicht ermöglicht, zum vorher fehlerlosen Status zurückzukehren und potentiell fehlerhafte Ergebnisse bis zum Zeitpunkt der letzten fehlerlosen Phase zurückzusetzen.

Als Benchmarks wurden die kombinatorischen ISCAS-Schaltungen ausgesucht. Die Anzahl der ausgewählten Designs sind in der nachfolgenden Tabelle 4-1 aufgeführt:

Tabelle 4-1 : Schaltungscharakteristika der ISCAS-Designs

ISCAS-Design	***# Ports***	***# Transistoren***
c1908	58	3092
c2670	373	4950
c3540	72	6876
c432	43	784
c499	73	2092
c5315	301	9930
c6288	64	10112
c7552	315	14336

4.1.1 Optimierte Wechsel zwischen TMR- und Einzelmodus

Der erste Ansatz zur Steigerung der Energieeffizienz ist ein stetiger Wechsel zwischen einem TMR-Modus und einem Einzelmodus [Sae10]. Im Einzelmodus werden die Eingänge nacheinander durch die Demultiplexer-Einheit zu den jeweiligen Eingängen der Module geleitet. Am Ausgang der Gesamtschaltung ist die Multiplexer-Einheit. Sie leitet den Ausgang der Module nacheinander an den Ausgang der Gesamtschaltung weiter. Dadurch entsteht ein dreifach paralleles System während dieses Modus und die Frequenz der einzelnen Module wird auf ein Drittel reduziert, wobei die Frequenz der gesamten Systems und somit der Durchsatz konstant bleibt. Dieser Modus stellt den Hauptmodus dar, der regelmäßig durch den TMR-Modus unterbrochen wird. Hierbei werden alle drei Module gleichzeitig mit denselben Eingängen beschaltet. Der Ausgang der Gesamtschaltung wird dann durch den Mehrheitsentscheid der

Ausgänge aller drei Module bestimmt. In diesem Modus kann außerdem mittels einer Fehlerdetektionseinheit erkannt werden, ob in einem der Module Fehler aufgetreten sind, in dem die Modulausgänge zusätzlich gegeneinander verglichen werden. Wenn ein Fehler erkannt wird, indem eine Einheit in der höheren Designebene das Fehlersignal der Detektionseinheit auswertet, bleibt die Gesamtschaltung im TMR-Modus, wobei dann dieser und weitere Fehler so lange wie möglich maskiert werden.

Die Gesamtschaltung ist in Abbildung 4-1 a) dargestellt. Die drei zusätzlichen Einheiten (funktionale De-/Multiplexer und Fehlererkennung) sorgen für ein kontrolliertes Umschalten zwischen den beiden Modi, ein korrekte Zuordnung der Ein- und Ausgänge und die Fehlererkennung. Die einzige Aufgabe der Demultiplexer-Einheit besteht in der Generierung der Enable-Signale für die Eingangsregister der Einzelmodule. Durch das „Mode"-Signal wird angezeigt, welche Register an- bzw. ausgeschaltet werden. Während des TMR-Modus sind alle Register ständig aktiviert und reichen die Eingänge gleichzeitig an alle Module weiter. Im Gegensatz dazu werden im Einzelmodus jeweils nur die Register eines Moduls eingeschaltet, während die Register der anderen beiden Module deaktiviert werden. Durch einen simplen round-robin-Mechanismus werden so jedem Modul alle drei Takte neue Eingangsdaten zugeordnet. Dadurch wird eine um zwei Drittel reduzierte Betriebsfrequenz für die Module erreicht, was im Gegensatz zum konventionellen TMR zu einer um zwei Drittel reduzierten Verlustleistung führt. Diese Verminderung ist allerdings der Idealfall, deren Effekt durch den Einsatz der zusätzlichen Einheiten gemindert wird. Die Weiterleitung der Ausgänge zum Gesamtausgang wird in der Multiplexer-Einheit ähnlich organisiert. Während des TMR-Modus werden die Ausgänge per Mehrheitsentscheid weiter transferiert. Dies wird mittels Bit-Voter für jedes Ausgangsbit realisiert. Im Einzelmodus werden die Ergebnisse des Moduls, das mit dem Taktsignal neue Daten am Eingang übernimmt, an den Ausgang geleitet. Die korrekte Zuordnung in beiden Modi erfolgt bitweise über eine Multiplexstruktur.

Die Fehlererkennung ist während des TMR-Modus aktiv. Sie erzeugt ein Fehlersignal, sobald ein Ausgangsbit von den anderen beiden Pendants abweichen sollte. Wenn der Fehler durch eine höhere Ebene als permanenter Fehler deklariert wird, beispielsweise durch wiederholtes oder längeres Auftreten, wird der TMR-Modus beibehalten, damit (wieder) korrekte Ausgänge generiert werden. Die Fehlererkennung profitiert dabei vom dreifachen Aufbau, da ein eventuell fehlerhaftes Ausgangssignal zweimal gegen ein korrektes Signal in einem XOR-Abschnitt getestet wird. Alle Signale werden dann durch einen Baum von 3-Eingangs-Oder-Gattern zu einem Fehler-Signal zusammengefasst – Abbildung 4.1 b), wobei eine logische 1 einen Fehler anzeigt. Der kritischste Fall wäre eine Überdeckung dieses Fehlersignals. Aufgrund der Erzeugung der ursprünglichen Fehlersignale im XOR-Abschnitt wird ein Fehler zweifach in den Oder-Baum eingespeist. Dadurch wird die Fehlersignalverdichtung des Oder-Baumes zuverlässiger, weil mindestens zweimal ein Stuck-at-Fehler auftreten müsste, um das Fehlersignal zu überdecken. Zwischen den Oder-Gattern maskiert ein Stuck-at-0-Fehler ein Fehlersignal, in den Oder-Gattern vor dem treibenden Inverter ein Stuck-at-1, da dann am Ausgang des Oders eine 0 erscheinen würde.

Aus diesem Grund werden die Fehlersignale der drei XOR-Gatter eines Ausgangsbits solange wie möglich nicht in einem Oder zusammengeführt, so dass im Grunde drei kleinere Oder-Bäume durch die Zusammenschaltung mit einem abschließenden Oder zu einem großem Oder-Baum zusammengeführt werden. Deshalb würde ein einzelner Stuck-at-Fehler im Baum erst in der letzten Stufe ein Fehlersignal überdecken können. Dieser Ansatz erhöht zwar die Verdrahtung ein wenig, verbessert aber die Robustheit des Oder-Baumes.

Zur Analyse der effizientesten Schaltstrategie bedarf es zweier Parameter. Zum einen die Schaltfrequenz f_{TMR}, mit der in den TMR-Modus geschaltet wird. Zum anderen gibt die Dauer d_{TMR} an, wie viele Takte der TMR-Modus beibehalten wird bis in den Einzelmodus zurückgeschaltet wird. Je größer beide Parameter sind, desto wahrscheinlicher wird ein Fehler in den Modulen erkannt, weil der Anteil des TMR-Modus an der Betriebszeit t_{OP} steigt. Dieses Verhältnis, im Folgenden als TMR-Rate r_{TMR} bezeichnet, wird folgendermaßen berechnet:

$$r_{TMR} = f_{TMR} \cdot d_{TMR} \cdot t_{OP} \tag{66}$$

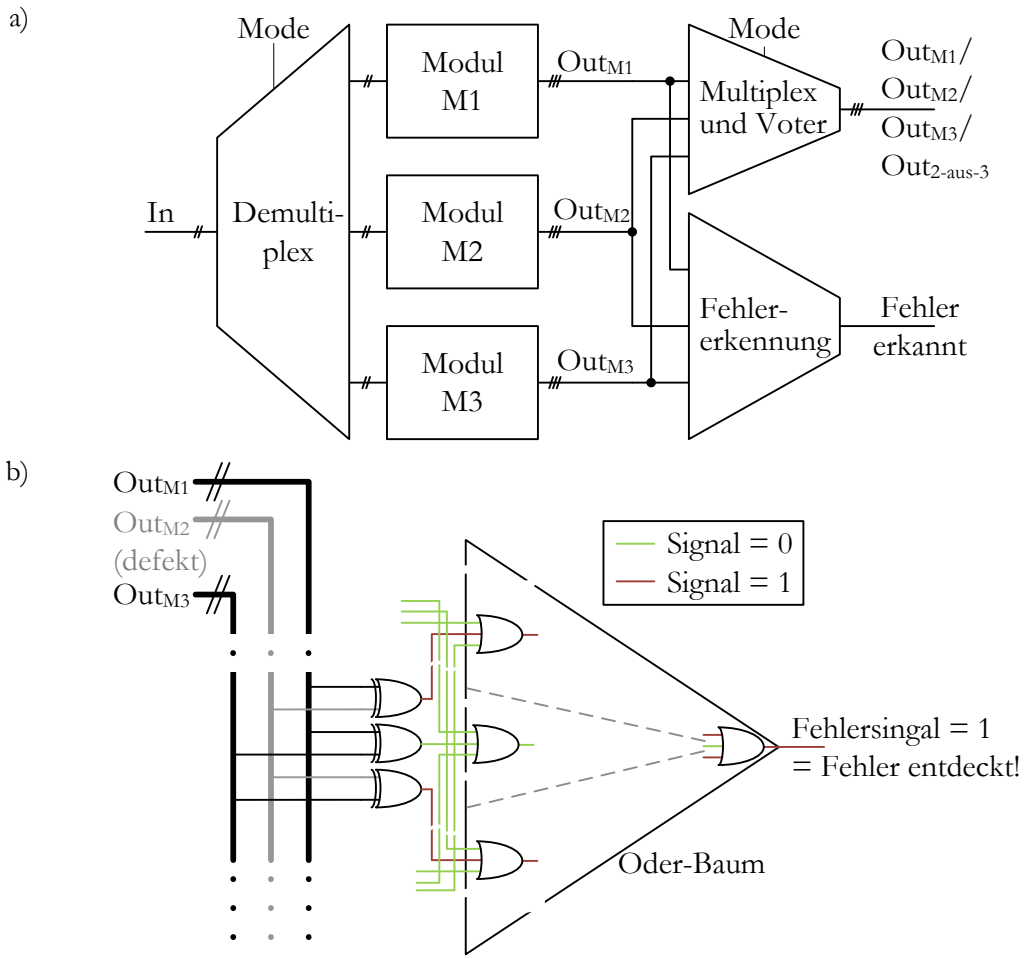
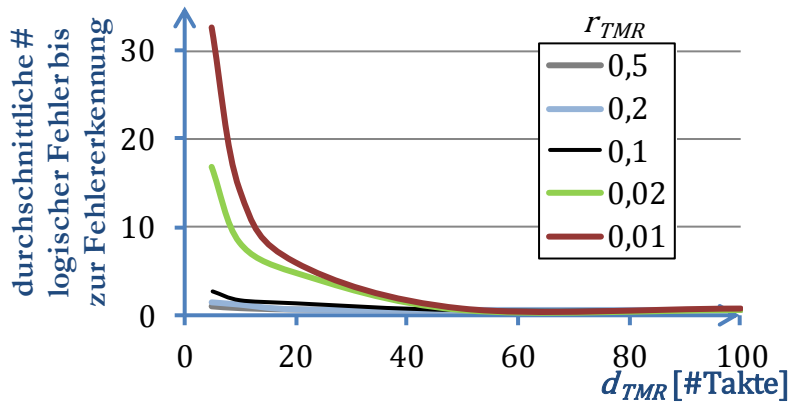


Abbildung 4-1 : Erweiterung eines konventionellen TMR-Designs zur Reduzierung der Leistungsaufnahme mittels paralleler Datenpfade
a) Aufbau
b) Prinzip der Fehlererkennung

Eine TMR-Rate von $r_{TMR} = 1$ bedeutet, dass der TMR-Modus die gesamte Betriebszeit ausgeführt wird, während bei $r_{TMR} = 0$ das Gegenteil mit einem ununterbrochenen ausgeführten Einzelmodus der Fall ist. Während mit steigendem r_{TMR} die Fehlererkennungsfähigkeit und damit die Zuverlässigkeit augenscheinlich zunimmt, steigt allerdings auch die Verlustleistung, da der sparende Einzelmodus seltener zum Einsatz kommt.

Deshalb wurde der Einfluss von d_{TMR} auf die Erkennung von permanenten Fehlern untersucht. Dazu wurden ISCAS-Designs dahingehend analysiert, wie lange ein vom Simulationsstart vorhandener Stuck-at-Fault in einem der Module von der Fehlererkennungsschaltung unbemerkt bleibt, wobei die Parameter r_{TMR} und d_{TMR} variiert wurden. Dafür wurden die aufgetretenen logischen Fehler der Schaltung bei zufällig generierten Eingangssignalen summiert, bis die Schaltung den Fehler erkannte und somit im TMR-Modus verbleiben würde. Am Anfang des Simulationslaufs wurde immer der TMR-Modus ausgeführt, um gleiche Startbedingungen für die Simulationen zu garantieren. Man könnte dies auch in dem Sinne interpretieren, dass immer ein Art Selbsttest nach dem Starten der Schaltung ansteht. Des Weiteren ist dies auch für den Vergleich vorteilhaft, da früh erkannte Fehler im ersten TMR-Modus erkannt werden und somit zu keinen falschen Ergebnissen führen, da diese ja nur in den Einzelmodusphasen auftreten können. So erhalten diese erfolgreichen Simulationsläufe eine

a)



b)

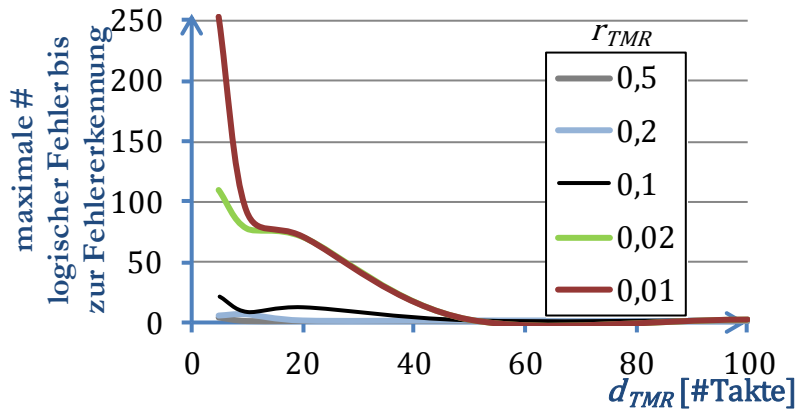


Abbildung 4-2 : Durchschnittliche (a) und maximale (b) Anzahl an logischen Fehler der Gesamtschaltung bis die Fehlerdetektionseinheit einen Fehler registriert hat

Fehlerzahl von 0. Abbildung 4-2 zeigt exemplarisch die Ergebnisse für das Design c6288. Hier wurde die durchschnittliche und maximale Fehleranzahl über der TMR-Dauer d_{TMR} aufgetragen. Ferner sind auch unterschiedliche Kurven für unterschiedliche Werte von r_{TMR} abgebildet. Zwei wichtige Schlüsse können gezogen werden. Wenn die TMR-Rate groß genug ist (hier: $r_{TMR} > 0.1$), können Fehler schnell erkannt werden, da die Schaltungen sogar mit geringen Werten von $d_{TMR} < 10$ Takten die Fehler in den ersten Vergleichsphasen, wenn das Design im TMR-Modus ist, identifiziert. Noch wichtiger ist die Tatsache, dass bei kleinen TMR-Raten die logischen Fehler auch auf sehr geringe Werte nahe oder gleich Null gesenkt werden können, wenn d_{TMR} nur groß genug ist. So betrug die Fehleranzahl bei allen Versuchen maximal 2 bei einer Dauer $d_{TMR} > 50$ Takte. Dies stimmt sehr gut mit der Strategie überein, möglichst viel Leistung mittels kleiner TMR-Raten einzusparen. Wenn nun d_{TMR} groß genug ist, kann ein Fehler trotz geringer r_{TMR} schnell erkannt werden. Der einzige Nachteil dieser Strategie ist, dass bei erfolgter Fehlererkennung mehr Ergebnisse als unsicher eingestuft werden müssen, da mehr Zeit seit dem letzten Wechsel aus dem TMR-Modus vergangen ist, was im Endeffekt den Durchsatz schmälert.

Die nächste Untersuchung betrifft die Verminderung der Zuverlässigkeit, die durch die zusätzlichen Gatter im Gegensatz zum konventionellen TMR zu erwarten ist. Dazu wurden verschiedene Simulationen auf RTL-Ebene ausgeführt. Die Fehler wurden dabei als Stuck-at-Faults an allen Netzen der Schaltung über die Zeit eingefügt. Die Wahl des jeweiligen Netzes erfolgte zufällig mit gleicher Fehlerwahrscheinlichkeit λ_{NET} für alle Netze, da auf RTL-Ebene simuliert wurde. Durch die konstante Fehlerrate ergibt sich für das Gesamtdesign eine flächenabhängige Zuverlässigkeit, was bedeutet, dass mit steigender Fläche auch die Ausfallrate λ_{SYSTEM} steigt. Da beim Eintritt einer Fehlersituation der TMR-Modus bis zum Ende durchgeführt wird, wurde dieser durchgehend während dieser Tests ausgeführt, um die Ausfallzeit der Schaltung zu analysieren. Als Referenzschaltungen dienten die einfache Schaltung ohne jegliche Erweiterungen und die TMR-Variante mit Votern an jedem Designausgangsbit. In Abbildung 4-3 a) sind die *MTTF*-Werte nach Gleichung (37) der beiden TMR-Varianten (konventionell und verlustleistungsreduziert) im Vergleich zur „ungeschützten“ Schaltung

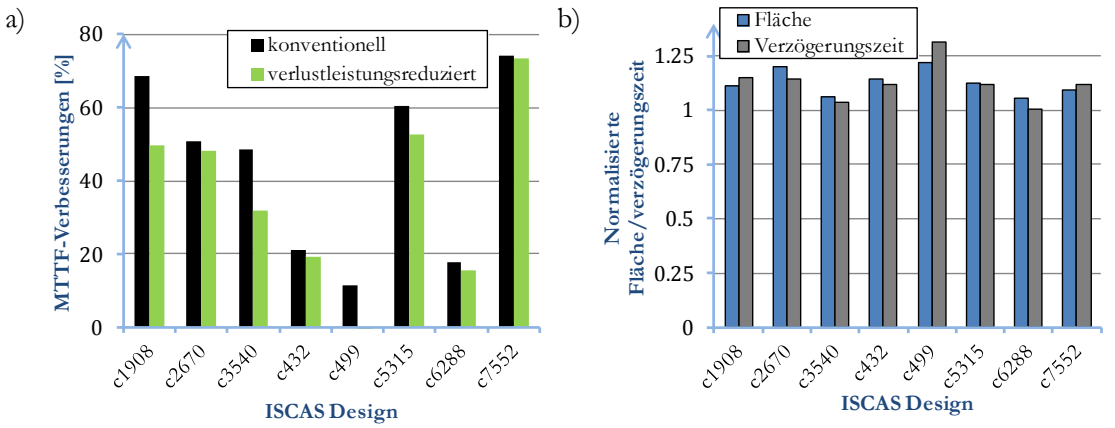


Abbildung 4-3 : Ergebnisse der verlustleistungsreduzierenden TMR-Implementierungen
a) Verbesserungen gegenüber der MTTF des Originaldesigns
b) Mehraufwand gegenüber der Fläche und der Verzögerungszeit des konventionellen TMR

dargestellt. Die *MTTF* aller leistungssparenden Varianten ist immer geringer als die der konventionellen TMR-Schaltung, was durch die gleiche Maskierungsmethode der leistungssparenden Variante bei leicht erhöhtem Flächenbedarf zu erwarten war. Die Abweichungen liegen in einem Bereich zwischen 1 % und 11 %. Ferner kann die Zuverlässigkeit allerdings soweit sinken, dass sogar das Referenzdesign bessere Ergebnisse erzeugt, wie im Fall des Design c499. Das liegt daran, dass der Flächenaufwand von der Anzahl der Ein-und Ausgangspins und der Gesamtfläche abhängig ist, was aus Abbildung 4-3 b) ersichtlich ist. Hier haben die kleinsten Designs – c432, c499 – und die Schaltungen mit vielen Ausgangspins – c1908, c2670 und c499 – den größten zusätzlichen Flächenbedarf. Das liegt am größerem Anteil der Zusatzfläche zur Gesamtfläche, sowie an der linearen Abhängigkeit der Fläche der Multiplex-Einheit und der Fehlererkennung von der Anzahl der Ausgangspins, da jedes Bit sowohl zur Fehlererkennung als auch zur Mehrheitsentscheidung herangezogen wird. Sollten nun beiden Faktoren (geringe Fläche, viele Designpins) zusammenkommen, wie beim c499, wird die Zuverlässigkeit des Systems sogar derart verringert, dass die ungeschützten Designs besser abschneiden, wenn der verbesserte TMR-Ansatz gewählt wird, da der Flächenbedarf der Kontrolleinheiten zu groß wird. Andererseits wird bei einer sinkender relativen Zusatzfläche aufgrund einer großen Gesamtfläche, wie bei den größten Designs c6288 und c7552, die *MTTF*-Abweichung der leistungssparenden TMR-Variante sehr gering. Die Verzögerungszeit der Schaltungen ist besonders von der Anzahl der Ausgangspins abhängig, da sie die Anzahl der Stufen des Oder-Baums (Fehlererkennung) bestimmen und somit die zusätzliche Verzögerungszeit. Zusätzlich wird der Einfluss auch dadurch bestimmt, wie schnell das Ursprungsdesign ist. Dabei sinkt der relative Einfluss der zusätzlichen Einheiten mit steigender ursprünglicher Verzögerungszeit. Sollte die modifizierte Verzögerungszeit in den kritischen Bereich kommen, ist ein Oder-Baum mit Registerstufen eine geeignete Gegenmaßnahme, da eine Verzögerung des Fehlersignals um ein paar Takte unerheblich ist, wenn eine große TMR-Frequenz f_{TMR} ausgewählt wurde. Wichtig zu erwähnen ist, dass die Gesamtschaltung dauerhaft eine Latenz von drei Takten aufweist, da mit jedem Umschalten von Einzel- auf TMR-Modus zwei Takte „gewartet“ werden muss, damit die im Einzelmodus zugewiesenen Eingangsvektoren korrekt berechnet und nacheinander wieder ausgegeben werden. Die funktionalen De-/Multiplexer-Einheiten kompensieren dies intern, damit die Latenz konstant bleibt, allerdings ist sie dadurch auch dauerhaft größer als die minimal mögliche ein Takt lange Latenz im TMR-Modus.

Hinsichtlich der Leistungsaufnahme ist das Einsparpotential enorm gegenüber dem konventionellen TMR. Dies wird in Abbildung 4-4 deutlich. Hier sind die Werte der Leistungsaufnahme der verschiedenen Designs in Relation zu deren konventionellem TMR-Design dargestellt. Die Schaltungen wurden im fehlerlosen Zustand mit unterschiedlichen TMR-Raten simuliert. Resultat ist eine lineare Abhängigkeit der Leistungsaufnahme r_{TMR}, wobei Schaltungen mit einer größeren Gesamtfläche mehr Nutzen aus der Wechselstrategie ziehen, was sich mit den Ergebnissen zur Zuverlässigkeit und zum Flächenbedarf deckt. Durch die relativ gesehen geringeren Flächenkosten, die sich nach Gleichung (24) linear auf die Leistungsaufnahme auswirken, kann eine Leistungseinsparung von bis zu 50 % bei einer kleinen TMR-Rate erzielt

werden (c6288). Im Gegensatz dazu erreichen kleinere Designs und die Schaltungen mit vielen Ausgangspins nur geringe Einsparungen von maximal 5 % (c499, c432, c2607). Die anderen allerdings erreichen signifikante Einsparungen solange $r_{tmr} < 0.75$ ist. Die maximale Einsparung von 50 % entspricht einer Leistungssteigerung von 45 % gegenüber dem ungeschützten Design, welches als Konstante um die 33 % der Leistungsaufnahme des konventionellen TMR in Abbildung 4-4 dargestellt ist.

Zusammenfassend lässt sich statuieren, dass der Einsatz unterschiedlicher Betriebsmodi zur Senkung der dreifachen Leistungsaufnahme des konventionellen TMR gegenüber dem ungeschützten Referenzdesign eine gute Ausgangsbasis für weitere Verbesserungen bietet. Vor allem größere Designs können von den Modifizierungen profitieren, da der zusätzliche Flächenbedarf der Kontrolleinheiten bei diesen Schaltungen relativ gering ausfällt. Im Folgenden werden sowohl zusätzliche Modi eingeführt, als auch verschiedene Strategien zur Wahl des Finalmodus, in dem die Schaltung schließlich bis zum Betriebsende operieren wird. Ziel dabei war es nicht nur die Verlustleistung weiter zu reduzieren, sondern auch die Zuverlässigkeit im Gegensatz zum konventionellen TMR zu verbessern.

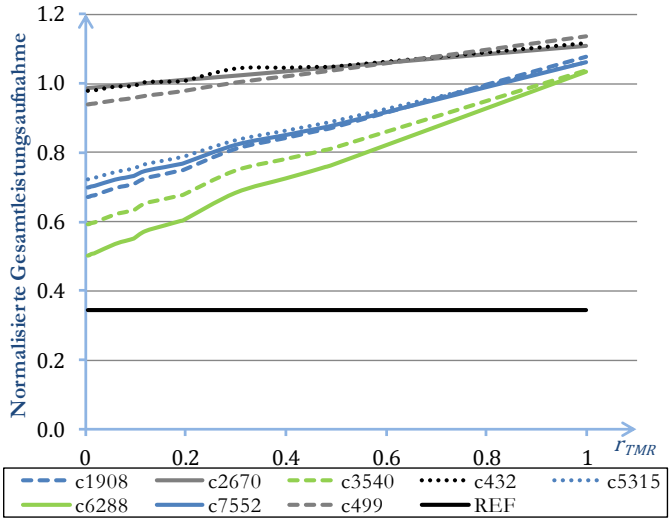


Abbildung 4-4 : Gesamte Verlustleistungsaufnahme der optimierten Schaltungen, normiert auf das konventionelle TMR-Design, in Abhängigkeit von der TMR-Rate; nicht redundante Originalschaltungen als REF abgebildet

4.1.2 Weiterentwicklung zu einem flexiblen Schaltungsbetrieb

Die Betrachtungen des vorigen Abschnittes haben gezeigt, dass vor allem bei großen kombinatorischen Designs die Reduzierung der Leistungsaufnahme größer und die der Zuverlässigkeit geringer ist. Aus diesem Grund und durch die fehlende Verfügbarkeit größerer anerkannter kombinatorischer Vergleichsschaltungen [Har00], wie die ISCAS-Designs, wurden Arithmetische-Logische Einheiten (engl. arithmetic logic units – ALU) in unterschiedlichen Ausführungen für die folgenden Analysen verwendet. Dies erlaubt eine Untersuchung

verschiedener Designs hinsichtlich variierender Schaltungsgrößen und Verzögerungszeiten bei Beibehaltung der grundsätzlichen logischen Funktion. Die Schaltungsparameter der verwendeten ALUs sind in Tabelle 4-2 dargestellt:

Tabelle 4-2 : Syntheseergebnisse der Referenzschaltungen (ALU1 bis ALU5)

Design Parameter	***ALU1***	***ALU2***	***ALU3***	***ALU4***	***ALU5***
# der Ausgangsbits	32	64	64	96	96
# der Logikgatter	2335	7078	11839	15542	23093
Verzögerungszeit [ns]	2.63	8.36	4.66	12.51	6.72

Die ursprünglichen ungeschützten Referenzschaltungen (ALU1 bis ALU5) wurden verwendet, um daraus sowohl konventionelle TMR-Schaltungen (TMR1 bis TMR5) als auch die verlustleistungsreduzierten Designs (LPR1 bis LPR5, LPR – engl. Low Power Redundant designs) abzuleiten. Ähnlich wie die leistungssparenden TMR-Schaltungen aus Abschnitt 4.1.1 wurden die LPR-Schaltungen mit zusätzlichen Kontrollstrukturen versehen, damit einerseits die Schaltungen in den verschiedenen Modi arbeiten können, und andererseits Fehler sowohl erkannt als auch lokalisiert werden können [Sae11].

Die in Abschnitt 4.1.1 verwendeten Modi (TMR, Einzel) wurden um einen dritten Modus erweitert – den Dual-Modus. Hierbei werden jeweils zwei der drei Module betrieben, während das dritte abgeschaltet ist. Diese Abschaltung ist im einfachsten Fall, wie nachfolgend angewandt, ein Clock-Gating, in dem Eingangsregister ausgeschaltet sind, so dass die Eingänge des Moduls stabil bleiben. Sie könnten aber auch mittels Sleep-Transistoren komplett von der Betriebsspannung getrennt werden, um auch den Leckstrom zu senken. Dies hätte neben der Verlustleistungsreduzierung den Vorteil eines geringeren Verschleißes der Transistoren. Dieser Dual-Modus unterteilt sich wiederum in zwei Phasen: Vergleichsphase und Parallelphase. Eine komplette Aufstellung der Modi und die Übergänge zwischen ihnen gibt Abbildung 4-5 und wird im Folgenden näher erläutert, wobei der jeweils aktive Modus abhängig vom Fehlerzustand der Gesamtschaltung ist.

Ein Fehler wurde noch nicht erkannt: Die Schaltung nutzt den Dual-Modus. Ähnlich dem Wechsel zwischen TMR- und Einzelmodus in Abschnitt 4.1.1, wird nun das Design abwechselnd mit einer festen Vorgabe in der Vergleichs- oder in der Parallelphase betrieben. Während der Parallelphase wird jeweils ein Eingangsvektor von einem Modul bearbeitet, während der nächste Vektor im nächsten Takt vom anderen Modul berechnet wird. Dies bewirkt ein um die Hälfte gesenkte Betriebsfrequenz für die Module und damit eine Reduzierung der Stromaufnahme bei gleichbleibendem Durchsatz. In der Vergleichsphase wiederum wird beiden Modulen der gleiche Inputvektor zugewiesen, so dass Ergebnisse am Ausgang mit Hilfe der Fehlererkennungseinheit miteinander verglichen werden können.

Ein Fehler wurde während der Vergleichsphase (Dual-Modus) erkannt: Wenn erkannt wurde, dass ein Defekt in einem der Module vorhanden ist, wird in den TMR-Modus geschaltet. Die Fehlererkennung wurde dahingehend erweitert, dass anhand der Ausgänge der Defekt einem

Modul zugewiesen werden kann. Wenn nun ein Modul als defekt markiert und ausgeschaltet wurde, werden die anderen noch funktionierenden weiter im Dual-Modus betrieben. Auf diese Weise kann der leistungssparende Modus beibehalten werden.

Weitere Fehler wurden erkannt: Sollte nun ein Defekt in den beiden funktionierenden Modulen erkannt werden, wurden folgende fünf finale Modi implementiert:

F1: Das Gesamtdesign wird sofort abgeschaltet, da beispielsweise eventuelle Zuverlässigkeitsanforderungen sehr hoch sind.

F2: Die Gesamtschaltung wird nur noch im TMR-Modus betrieben.

F3: Genauso wie mit F2 wird das Gesamtdesign im TMR-Modus betrieben. Sollte allerdings die Fehlererkennung Defekte in allen drei Modulen anzeigen, wird das Design abgeschaltet.

F4: Die beiden defekten Module werden abgeschaltet und das funktionierende Modul wird im Einzelmodus betrieben.

F5: Genauso wie in F4 wird das funktionierende Modul im Einzelmodus betrieben, allerdings wird es regelmäßig vom TMR-Modus unterbrochen. Sollte nun das letzte Modul als defekt deklariert werden, wird das gesamte Design ausgeschaltet.

Um diese Änderungen schaltungstechnisch zu implementieren, wurden die Kontrolleinheiten der ursprünglichen Verbesserung aus Abbildung 4-1 a) angepasst, wobei der grundsätzliche Aufbau allerdings gleich bleibt. Die Aufgaben der funktionalen De-/Multiplexer-Einheit haben sich nur marginal verändert. Die korrekte Weiterleitung und Ausgabe der Ein- und Ausgänge mit Hilfe der Enable-Signale muss nun auch für die zusätzlichen Modi implementiert werden. Ferner reduziert sich die Latenz der Gesamtschaltung auf zwei Takte durch den Dual-Modus. Die Fehlererkennung wurde umfangreicher modifiziert, um unterschiedliche Zustände zu

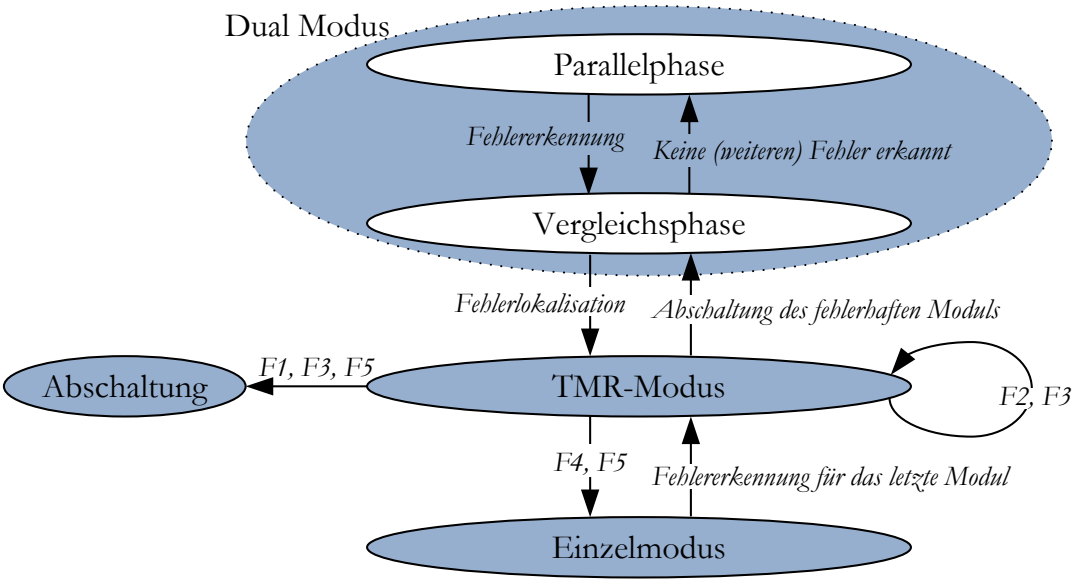


Abbildung 4-5 : Modi der LPR-Schaltungen

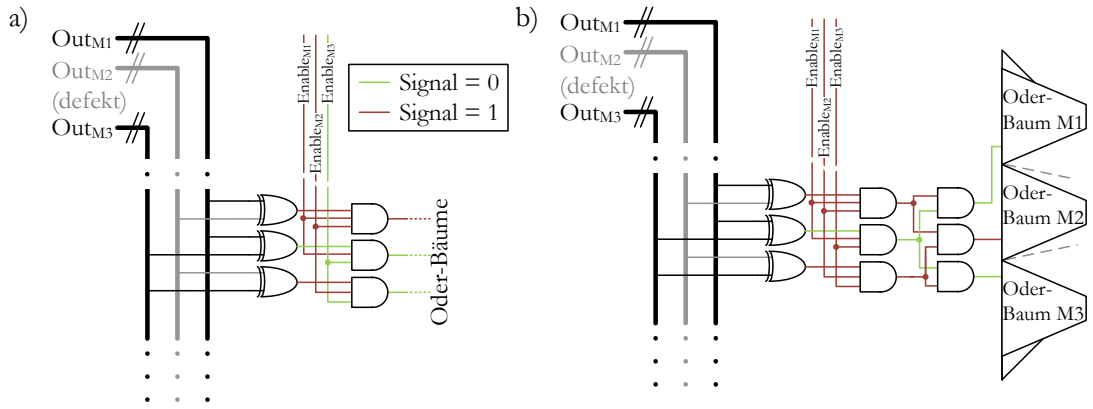


Abbildung 4-6 : Beispielhafte Darstellung einer Gatterschaltung zur Fehlererkennung eines Ausgangsbits der Module M1, M2. M3

a) Während der Vergleichsphase (nur M1 und M2 werden verglichen durch jeweilige Beschaltung der Enable-Signale)
b) Während des TMR-Modus (alle Module zugeschaltet)

berücksichtigen, denn neben einer Fehlererkennung (Dual-Modus in Vergleichsphase) wird auch eine Fehlerlokalisation (TMR-Modus) durchgeführt.

Abbildung 4-6 zeigt exemplarisch eine Fehlererkennung (a) und -lokalistation (b) für ein Bit. Wenn zwei Module in der Vergleichsphase des Dual-Modus operieren (M3 ist in Ruhe), registriert eine XOR-Reihe Bit für Bit, ob Unterschiede in den Ausgängen der Module vorhanden sind. Sollte dies der Fall, generieren die beteiligten XOR-Gatter ein Fehlersignal, das in einen Oder-Baum geführt wird. Die eingefügte erste Und-Gatter-Reihe stellt sicher, dass nur die XOR-Signale der aktiven Module weitergereicht werden. Signale der inaktiven Module werden auf 0 gesetzt. Während des TMR-Modus wird die zweite Und-Gatter-Reihe aktiv, um die defekten Module zu erkennen, während die erste Und-Gatter-Reihe transparent wird, da alle Enable-Signale gleich 1 sind. Zur Fehlerlokalisation werden die drei kleineren Oder-Bäume der vorigen Schaltung genutzt, einer für jedes Modul. Die zweite Und-Gatter-Reihe kombiniert die drei möglichen Kombinationen (M1/M2, M1/M3, M2/M3) der XOR-Gatter-Reihe, um die Signale dann an alle drei Oder-Bäume auszugeben. Bei einem Unterschied zwischen den Modulen in diesem Bit wird ein Oder-Baum eine 1 bis zum finalen Fehlersignal des Oder-Baums durchreichen. Die Kombination der Fehlersignale am Bitausgang kann mittels Tabelle 4-3 zur Lokalisation des fehlerhaften Moduls herangezogen werden.

Neben der Möglichkeit zur Fehlerlokalisation bietet die Aufteilung in drei Oder-Bäume einen weiteren Vorteil: Redundanz zur Fehlerprüfung im Oder-Baum. Wenn während der Vergleichsphase die XOR-Signale an zwei oder alle drei Bäume weitergereicht werden, kann mittels Vergleich der finalen Signale erkannt werden, ob Defekte in den Bäumen vorhanden sind, so dass das Design abgeschaltet werden kann. Auch die kritischen Stuck-at-0-Fehler auf den Schaltungsnetzen im Oder-Baum können so erkannt werden, da sie eventuell erkannte Defekte in den Modulen maskieren könnten. Mit ein wenig mehr Aufwand in Form eines Zählers ist auch möglich gezielt Einsen in inaktive Oder-Bäume einzuleiten und so für jedes Eingangsbit des

Oder-Baumes zu prüfen, ob diese Einsen bis zum Ende des Baumes durchgereicht werden. Dieser Aufwand steigt dann allerdings mit der Anzahl der Ausgangsbits eines Designs. Weitere Mechanismen wurden implementiert, um Defekte in den Kontrollstrukturen zu erkennen, z. B. in dem die Enable-Signale zweifach generiert werden – einmal in der Multiplexer- und einmal in der Demultiplexer-Einheit. Unterschiede können so auf Defekte in den beiden Einheiten hinweisen. Sollten Defekte in den Kontrolleinheiten von der Schaltung bemerkt werden, schaltet diese sich selbständig aus.

Tabelle 4-3 : Zuweisung der defekten Module aufgrund der Kombination der aufgetretenen Fehlersignale

Fehlersignal Kombination 1		***Fehlersignal Kombination 2***	***Defektes Modul***
M1 XOR M2	Und	M1 XOR M3	M1
M1 XOR M2	Und	M2 XOR M3	M2
M1 XOR M3	Und	M2 XOR M3	M3

4.1.3 Reduzierte Verlustleistung

Auch für die LPR-Designs gilt, dass mit ansteigender Fläche der relative Zusatzaufwand gesenkt werden kann, da er hauptsächlich von der Anzahl der Ausgangspins abhängig ist. Dennoch reichen die zusätzlichen Flächenkosten relativ zum konventionellen TMR von 17 % für das kleinste Design (LPR1) bis zu 4 % für die größte Schaltung (LPR5). Die Verzögerungszeit wird vor allem für die schnelleren Designs stärker erhöht (LPR1: +30 %, LPR3: +18 %), während der Zuwachs der langsamen Schaltungen (LPR2, LPR4) unter 10 % bleibt. Diese Nachteile, die in Abbildung 4-7 a) ersichtlich sind, werden durch die Reduzierung der Verlustleistung aufgehoben, sofern diese ein wichtiger Parameter für das Design ist. Abbildung 4-7 b) illustriert dies eindringlich, da die Stromaufnahme im fehlerfreien Fall auf bis zu 40 % (LPR5) reduziert werden kann. Hier wird die Leistungsaufnahme des Gesamtdesigns relativ zum konventionellen TMR über die Vergleichsrate r_{COMP} darstellt. Die Vergleichsrate repräsentiert ähnlich r_{TMR} den Anteil der Zeit, die die Schaltung in der Vergleichsphase operiert, wenn es komplett im fehlerfreien Dual-Modus aktiv ist. Aufgrund der Fehleranalyse zur Abhängigkeit von d_{TMR} wurde die Vergleichsphase ausreichend lang ausgeführt, um den Einfluss dieses Parameters auf die Ergebnisse marginal zu halten. In Abbildung 4-7 b) ist zu sehen, das selbst mit der höchsten Vergleichsrate ($r_{COMP} = 1$) nur 75 % der Verlustleistung generiert werden, da ein Modul ständig inaktiv ist. Durch die lineare Abhängigkeit der Verlustleistung von r_{COMP} wird die Stromaufnahme mit sinkender Vergleichsrate weiter reduziert. Im günstigsten der getesteten Fälle ($r_{COMP} = 0.02$) liegen die Einsparungen zwischen unter 50 % (LPR1) und 60 % (LPR5), was dem ungeschützten Referenzdesign (schwarze Linie) schon nah kommt, welche im Durchschnitt bei 33 % liegt. Dies liegt daran, dass die Schaltung im Dual-Modus zumeist in der Parallelphase operiert, wodurch die durchschnittliche Betriebsfrequenz der Module auf knapp über 50 % reduziert werden kann und nur zwei der drei Module aktiv neue Eingangsvektoren

erhalten. Insgesamt ist ersichtlich, dass die Leistungseinsparungen sehr groß sein können, wobei die größeren Schaltungen am meisten von den Verbesserungen profitieren.

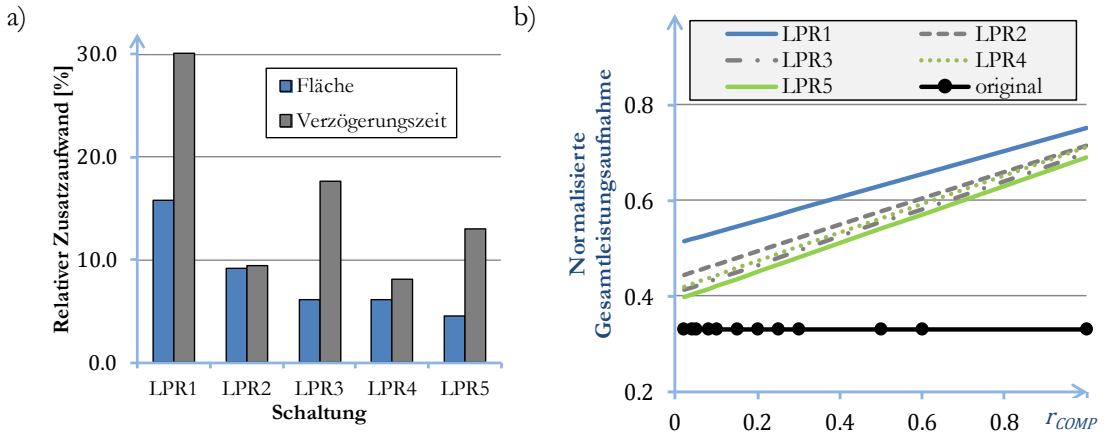


Abbildung 4-7 : Ergebnisse für die leistungssparenden LPR-Implementierungen
a) Mehraufwand gegenüber der Fläche und der Verzögerungszeit des konventionellen TMR
b) Reduzierung der Stromaufnahme gegenüber dem konventionellen TMR in Abhängigkeit von der Verweildauer in der Vergleichsphase (r_{COMP})

4.1.4 Gesteigerte Zuverlässigkeit

Die Einfügung der Stuck-at-Fehler in die Netzliste erfolgte wie bei den Vorläufer-Designs mit konstanter Fehlerrate λ_{NET} über die Zeit für jedes Netz. Um auch den Einfluss der Fehlerrate einzubeziehen, wurden Simulationen (Anzahl > 100) mit unterschiedlichen konstanten Fehlerraten λ_{NET} durchgeführt. Da die Zuverlässigkeit $R(t)$ die Wahrscheinlichkeit ist, mit der die Schaltung bis zum Zeitpunkt t korrekt arbeitet, wurden folgenden Regeln für die Festlegung eines Ausfallzeitpunktes aufgestellt, damit auch die Regenerationsmöglichkeit der Schaltung Berücksichtigung findet:

- Der Ausfallzeitpunkt ist erreicht, wenn sich die Schaltung eigenständig ausschaltet
- Der erste Zeitpunkt, zu dem fehlerhafte Ausgänge während der finalen Modi registriert werden, stellt den Ausfallzeitpunkt dar
- Sollte auf einen fehlerhaften Ausgang während der Parallelphase nicht durch die folgende Vergleichsphase geeignet reagiert werden (mittels Wechsel zum TMR-Modus), stellt der Zeitpunkt, zu dem der fehlerhafte Ausgang auftrat, den Ausfallzeitpunkt dar
- Sollten noch keine der finalen Modi aktiv sein und die Kontrolleinheiten der LPR-Designs einen Defekt in den Modulen bemerken, der z. B. in der vorigen Parallelphase zu fehlerhaften Ausgängen führte, wird der Betrieb der Schaltung weitergeführt und noch kein Ausfallzeitpunkt vermerkt

Abbildung 4-8 stellt die Zuverlässigkeitsergebnisse von drei Referenzschaltungen und deren zuverlässigkeitsverbessernden Derivaten sowohl absolut für ALU3 und relativ zum

konventionellen TMR für ALU1 und ALU2 dar. Abgebildet sind die durchschnittlichen *MTTF*-Werte aller Simulationen – ermittelt nach Gleichung (37) – über die variierende Fehlerrate λ_{NET}, wobei die Vergleichsrate bei $r_{COMP} = 0.1$ bzw. 0.05 und 0.02 lag. Des Weiteren wurden die Resultate in die unterschiedlichen finalen Modi unterteilt und als gesonderte Kurven dargestellt, um auch dort die Unterschiede herauszuarbeiten. Generell ist zu erkennen, dass die sofortige Abschaltung (F1) erwartungsgemäß zu einer geringeren oder maximal gleichen Zuverlässigkeit führt wie das des ungeschützte Referenzdesigns. Dies liegt am dreifachen Flächenbedarf und an der Tatsache, dass schon nach der Defektdeklaration von zwei Modulen das Gesamtdesign abgeschaltet wird. So ist diese Strategie nur sinnvoll, falls besondere Zuverlässigkeitsanforderungen gestellt werden, so dass bei Defekten diese zu erkennen sind, und das System dann leistungssparend abgeschaltet werden soll. Die anderen Strategien waren immer besser als das Referenzdesign. Weiterhin ist augenscheinlich, dass die Modi mit Einzelmodus zum Ende (F4, F5) teilweise weitaus bessere Ergebnisse erzielten als die TMR-Varianten (F2, F3). Hauptursache wird die verkleinerte effektive „operierende Fläche" nach Erreichen des finalen Modus sein, weil die Wahrscheinlichkeit, dass weitere Defekte in Module eingefügt werden, die zur Berechnung des Ergebnisses des Gesamtsystems beitragen, geringer ist als bei Benutzung aller drei Module. Die TMR-Strategien wurden deshalb mit untersucht, da Defekte nicht zwangsläufig an mehreren/allen Ausgängen der Module zu Fehlern führen müssen, obwohl in zwei von drei Modulen schon Fehler gefunden wurden. Somit ist es möglich, dass das Gesamtdesign bei Benutzung des Mehrheitsentscheides noch weiterhin korrekt arbeitet. Dies ist auch daran zu erkennen, da z. B. F2 und F3 trotz größerer Fläche zumeist ähnlich zuverlässig – zwischen -10 % und +5 % zum konventionellen TMR – sind wie die TMR-Schaltungen.

F3 schneidet zusammengefasst besser ab als F2, da trotz Defekten in allen Modulen das Design noch korrekte Ergebnisse produzieren kann, während F2 die Schaltung schon abschalten würde. Beste Resultate zeigten die Einzelmodus-Strategien (F4, F5), die sogar manchmal (F5) bzw. meistens (F4) zuverlässiger waren als das konventionelle TMR-Design. Sowohl die automatische Abschaltung als auch Defekte in den Votern und/oder den Fehlerbäumen senken die *MTTF* des finalen Modus F5 gegenüber F4, da F5 noch den TMR-Modus zu Kontrollzwecken nutzt. Aus Abbildung 4-8 b) zeigt sich, dass die relative *MTTF* bei kleineren Schaltungen mit steigender Fehlerrate steigt, da einerseits bei F4 und F5 der Vorteil der kleineren operativen Fläche verstärkt wird und andererseits das konventionelle TMR schneller fehlerhaft wird, während der anfängliche Betrieb mit nur zwei Modulen der LPR-Schaltungen zu längeren fehlerfreien Betriebszeiten führt. In Abbildung 4-8 c) zeigt sich allerdings, dass ab einer bestimmten Fehlerrate dieser Effekt nicht mehr vorhanden ist. Da nun auch durch das größere Design eine kleinere Fehlerrate zu schnelleren Ausfällen bei den LPR-Schaltungen führt, bleibt die Relation zum konventionellen TMR konstant bzw. sinkt für F1 bis F3 leicht ab einer bestimmten Fehlerrate, da wahrscheinlich der zusätzliche Flächenbedarf der Kontrolleinheiten zu mehr Fehlern in diesen Bereichen führt. Aus dem Vergleich der drei Abbildungen lässt sich generell schließen, dass eine höhere Vergleichsrate zu besseren *MTTF*s führt.

Abbildung 4-9 veranschaulicht die Entwicklung der Verlustleistung des Gesamtdesigns LPR2 über der Zeit. Der erste abrupte Wechsel bei t_{FINAL} stellt den Wechsel in den finalen Modus dar,

wobei der Zeitpunkt als Durchschnitt über alle Simulationen errechnet wurde. Der eventuelle zweite Wechsel auf $P_{GESAMT} = 0$ ist die automatische Abschaltung der Modi F1, F3, F5. Vor t_{FINAL} ist die Stromaufnahme gleich für alle finalen Modi. Bis zu diesem Zeitpunkt ist maximal ein Defekt vorhanden ist, und alle Schaltungen operieren im Dual-Modus. Danach gibt es gravierende Unterschiede, da nun zwei Module als defekt deklariert wurden. Zum einen schaltet F1 sofort ab und verbraucht keine Leistung mehr. Die anderen schalten alle drei Module für den dauerhaften TMR-Modus an (F2, F3) oder das übrig gebliebene nicht defekte Modul für den dauerhaften Einzelmodus (F4, F5). F2 und F3 nehmen ein wenig mehr Strom auf als das konventionelle TMR-Design durch den leicht größeren Flächenbedarf der Kontrollstrukturen. Das gleiche gilt für F4 und F5 im Gegensatz zur Referenzschaltung.

Hierbei werden auch die Vorteile des automatischen Abschaltens offenbar. Zum einen kann

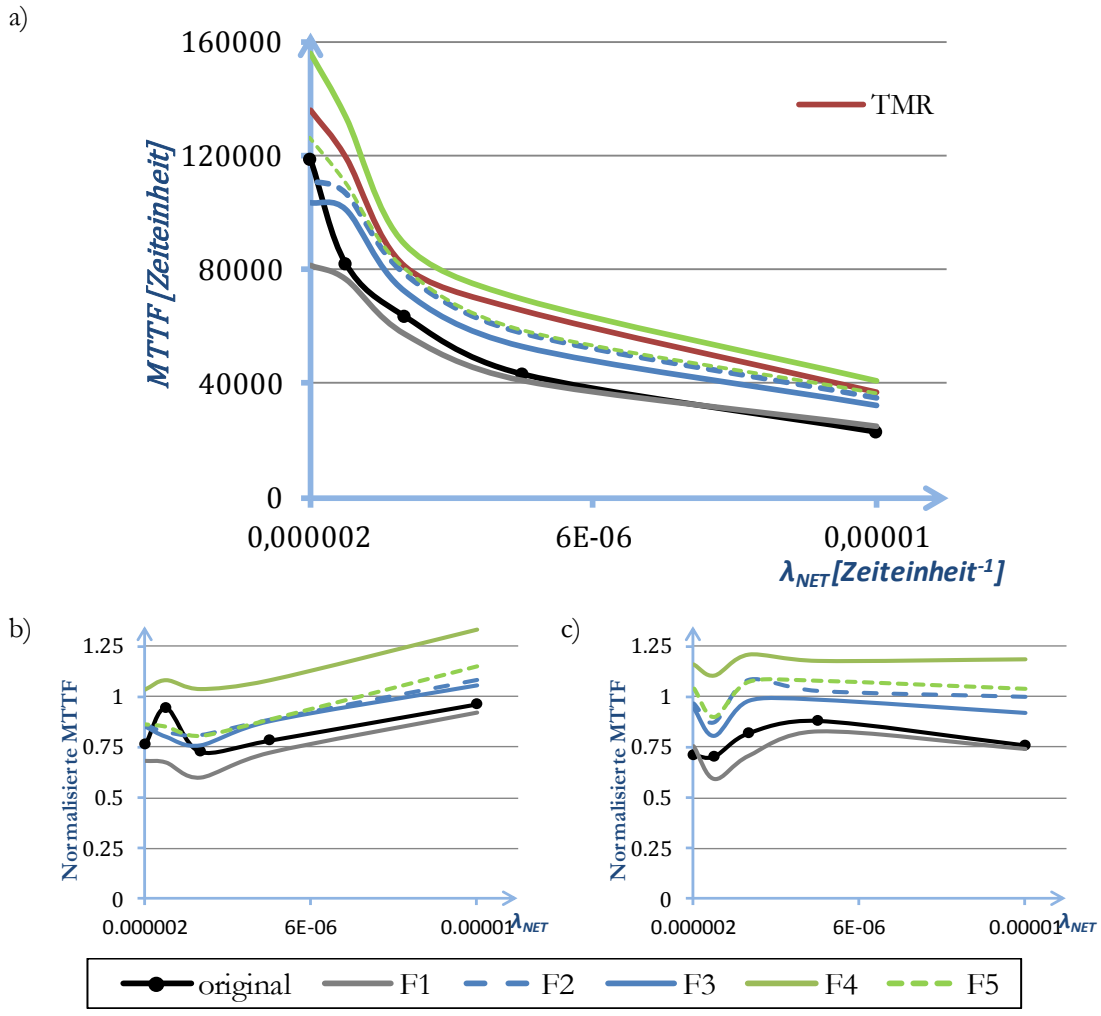


Abbildung 4-8 : *MTTF*-Ergebnisse über verschiedene Fehlerraten für die leistungssparenden LPR-Implementierungen

a) Absolute *MTTF* für ALU3, TMR3 und LPR3 mit $r_{COMP} = 0.1$
b) *MTTF* relativ zu TMR1 für ALU1 und LPR1 mit $r_{COMP} = 0.05$
c) *MTTF* relativ zu TMR2 für ALU2 und LPR2 mit $r_{COMP} = 0.02$

das Abschalten als einfacher Indikator genutzt werden, der fehlerhafte Ergebnisse einer defekten Schaltung anzeigt. Dadurch ist das Gesamtsystem, deren Teilsystem die redundante Schaltung darstellt, robuster auslegbar. Zum anderen ist die Reduzierung der Verlustleistung das Kriterium für die durchgeführten Verbesserungen, so dass auch im Fall des Versagens der Schaltung als Teil des Gesamtsystems die Verlustleistung des kompletten Systems durch das Abschalten einfach reduziert werden kann, da es nicht mehr gebraucht wird. Der spätere Abschaltzeitpunkt von F3 gegenüber F5 ergibt sich aus nicht erkannten Fehlern und damit einhergehend ein nicht erfolgtes Abschalten, wodurch sich der durchschnittliche Abschaltzeitpunkt verschiebt.

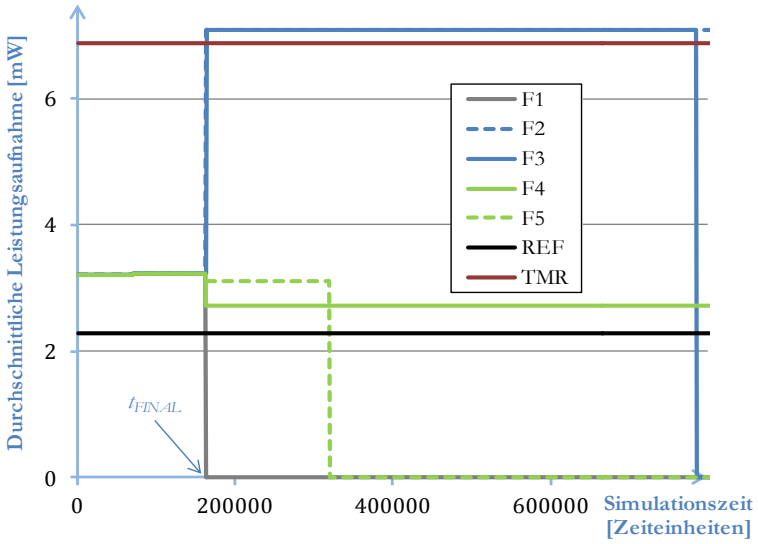


Abbildung 4-9 : Entwicklung der Leistungsaufnahme des LPR2-Designs und der unterschiedlichen finalen Modi über die simulierte Betriebsdauer

4.1.5 Fazit: LPR-Schaltungen

Abschließend kann konstatiert werden, dass die LPR-Designs, die die schon vorhandenen parallelen Datenpfade des TMR nutzen, die Forderung nach gesenkter Verlustleistung bei möglichst gleichbleibender Zuverlässigkeit im Vergleich zum konventionellen TMR erfüllen [Sae11]. Bei einem geringfügig erhöhten Flächenbedarf (maximal 15 %) sind Verlustleistungsreduzierungen von bis zu 60 % im fehlerfreien Betrieb möglich, was allerdings von der Größe der Originalschaltung abhängig ist. Es wurden auch Strategien aufgezeigt, mit denen die Zuverlässigkeit sogar erhöht werden konnte, indem nur noch korrekt funktionierende Module mit den finalen Modi F4 und F5 weiter betrieben wurden. Dies beließ auch weiterhin die Stromaufnahme unter der des konventionellen TMR-Designs. Ferner wurden Kontrollstrukturen dahingehend erweitert, dass Defekte in diesen erkannt werden können und dass die Gesamtschaltung bei zu vielen Modulfehlern oder bei Kontrollstrukturfehlern automatisch abgeschaltet wird. Dadurch kann die Zuverlässigkeit weiter erhöht werden, da eine geringere Stromaufnahme generell die Betriebstemperatur absenkt, wodurch weniger Defekte erzeugt

werden, wie es in Kapitel 3.3 dargestellt wurde. Dies führt eindeutig zu der Schlussfolgerung, dass wenn die LPR-Schaltungen eingesetzt werden sollen, die finalen Modi F4 oder F5 die richtige Wahl sind, je nachdem ob eine automatische Abschaltung bei zu vielen Defekten notwendig ist (F5) oder ob eine höchstmögliche Zuverlässigkeit (F4) wichtig ist.

Neben der erhöhten Fläche sollte auch eine erhöhte Verzögerungszeit, eine um einen Takt erhöhte Latenz und der Verlust der Maskierungsfähigkeit von flüchtigen Fehlern berücksichtigt werden, wenn die möglichen Verbesserungen gegen die Nachteile aufgewogen werden. Dies wird wie immer vom Designziel und den Vorgaben hinsichtlich der Verzögerung, der Fläche, der Leistung und der Zuverlässigkeit abhängig sein.

4.2 Redundanz auf Gatter-Ebene

Weil auch Ansätze zur Erhöhung der Zuverlässigkeit durch Einsatz von Redundanz auf Transistorebene existieren [Sir04] [Cor08], werden nachfolgend Simulationen vorgestellt [Sae09c], die untersuchen, auf welcher Designebene – Transistor, Gatter, RTL – Redundanz am effizientesten einsetzbar ist. Aufgrund des vorläufigen Charakters dieser Tests, die zu weiteren Modifikationen führen, und der Durchführung von Spice-Simulationen, wurde ein 4x4-Wallace-Multiplizierer als Grundschaltung gewählt [Wal64], da die Anzahl der Transistoren zwar gering genug für akzeptable Simulationszeiten ist, aber dennoch aussagekräftige Ergebnisse erwartbar waren. Um die Redundanz auf den unterschiedlichen Ebenen vergleichbar zu gestalten, basieren alle redundanten Implementierungen auf einer kompletten Verdopplung aller Transistoren und keine weiteren Kontrollstrukturen, wie Voter, oder ähnliche Module. Auf RTL-Ebene wurde der Multiplizierer im Ganzen als Block verdoppelt (Blockredundanz). Die Ein- und Ausgänge für die Blöcke sind ohne Trennelemente miteinander verknüpft. Zwar sind auf dieser Ebene auf den ersten Blick zusätzliche Kontrollelemente zwingend erforderlich, dennoch wurde aus oben genannten Vergleichsgründen dieser Ansatz verfolgt. Überdies ist zu beachten, dass ein Defektmodell verwendet wird, dass nicht unmittelbar zum Fehler führt, sondern das Design über die Zeit schädigt, so wie es aufgrund von Gateoxiddefekten zu erwarten ist.

Die Verdopplung der Gatter wird als Gatterredundanz bezeichnet. Hierbei sind die Logikgatter des Designs verdoppelt, d.h. bei gleichen Eingängen ist das Ausgangsnetz für beiden Gatter dasselbe und das Ausgangssignal wird dann an die folgenden Gatter weitergeleitet. Auf der untersten Ebene (Transistorredundanz) wird dann jeder Transistor verdoppelt und seine Anschlüsse sind mit den originalen Netzen verbunden. In den Abbildungen 4-10 a) bis c) sind die Unterschiede der drei Ansätze veranschaulicht. Diese Herangehensweise führt zu einem doppelten Flächenaufwand aller Implementierungen und einer verdoppelten Leistungsaufnahme der Ursprungsschaltung durch die verdoppelte Gesamtlastkapazität der redundanten Transistoren, wie es in Abbildung 4-10 d) dargestellt ist. Dort ist außerdem die Verzögerungszeit abgebildet, die im fehlerlosen Fall allerdings gleich bleibt, da die Erhöhung der Lastkapazität durch die verdoppelte Treiberstärke kompensiert wird.

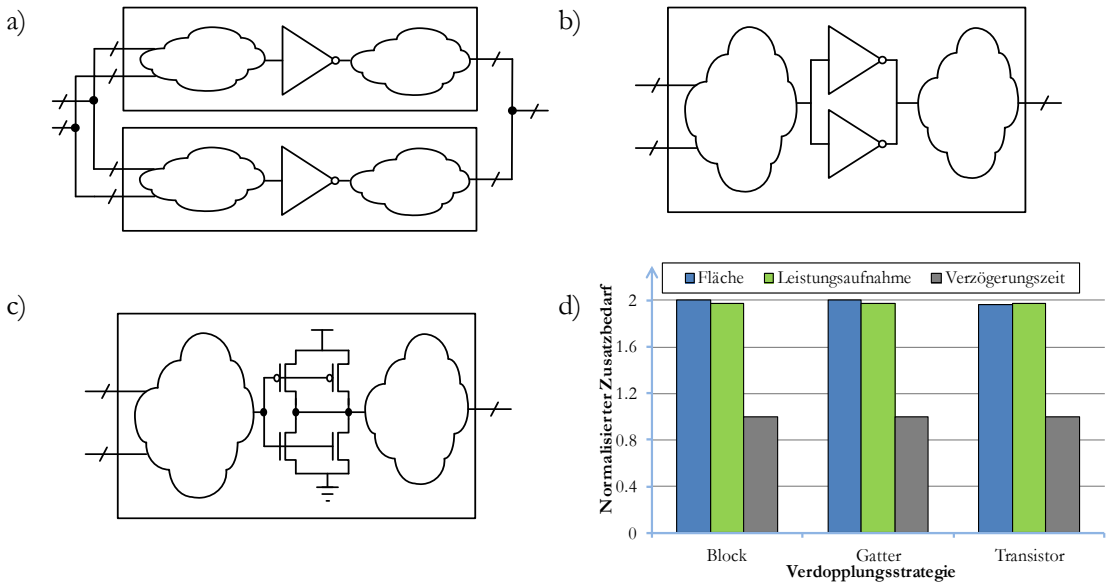


Abbildung 4-10 : Implementierungsmöglichkeiten für redundante 4x4-Wallace Multiplizierer zur Verbesserung der Zuverlässigkeit

a) Blockredundanz: das System wird im Ganzen verdoppelt
b) Gatterredundanz: die einzelnen Gatter des Systems werden verdoppelt
c) Transistorredundanz: die Transistoren werden separat verdoppelt
d) Notwendiger Zusatzaufwand für die Fläche, die Leistungsaufnahme und die Verzögerungszeit des Multiplizierers

Die Defektsimulationen wurden für den normalen Betrieb ausgelegt. Dabei wurden nach und nach harte Breakdowns nach dem Renovell-Modell (R_{GOB} = 1 Ω) mit konstanter Fehlerrate für die Transistoren in das System eingefügt, um den größtmöglichen Fehler je Transistor einzufügen. Die Vermutung liegt nahe, dass die feinere Granularität der Transistorredundanz eine bessere Zuverlässigkeit erzielt, da der Defekt möglichst nah am Defektort korrigiert wird. Außerdem ließen Ergebnisse mit einem älterem Segura-Defektmodell auch darauf schließen [Sae09a]. Allerdings wurden die besten Resultate mit der Gatterredundanz erreicht. Dies ist in Abbildung 4-11 a) zu sehen, in der die Zuverlässigkeitswerte der redundanten Multiplizierer dargestellt sind. Dabei wurde zu definierten Zeitschritten die Korrektheit der erzeugten Ausgangssignale einer simulierten Schaltung geprüft. Durch über 100 Simulationsläufe für jedes Design konnte die Zuverlässigkeit *R(t)* ermittelt werden, indem die Summe aller korrekt funktionierenden Designs zu einem Zeitpunkt in Relation zur Gesamtzahl der Simulationen gesetzt wurde. Beispielsweise sind nach 50 Zeitschritten mehr als drei Viertel aller Gatterredundanz-Systeme funktionstüchtig, während nur rund ein Drittel der Transistorredundanz-Systeme korrekt arbeiten. Die ursprünglichen Multiplizierer und die mit Blockredundanz sind zu diesem Zeitpunkt alle ausgefallen.

Die Zuverlässigkeitskurve zusammenfassend sind die *MTTF*-Verbesserungen in Abbildung 4-11 b) dargestellt. Die *MTTF* wurde dabei nach (37) berechnet. Interessanterweise führt die Blockredundanz sogar zu einer Verschlechterung der Zuverlässigkeit. Dies ist damit zu erklären, dass bei den redundanten Designs im Durchschnitt doppelt so viele Fehler auftreten wie

bei der Originalschaltung. Führen diese Defekte nun dazu, dass unkorrekte Pegel im Design erreicht werden, werden diese zu den Ausgängen weitergeleitet, wo diese dann gegen vielleicht korrekte Pegel konkurrieren und die Ausgangspegel dadurch undefiniert und damit falsch werden. So folgt daraus, dass Defekte entweder direkt am Gatter ausgeglichen oder aber Schaltungserweiterungen wie ein Vergleicher an den Ausgängen implementiert werden müssen, damit die unterschiedlichen Ausgangspegel kompensiert werden können. Die Defekte direkt am Entstehungsort zu kompensieren wird mit der Gatter- und Transistorredundanz erreicht, so dass eventuelle unkorrekte Pegel im System sofort maskiert werden können.

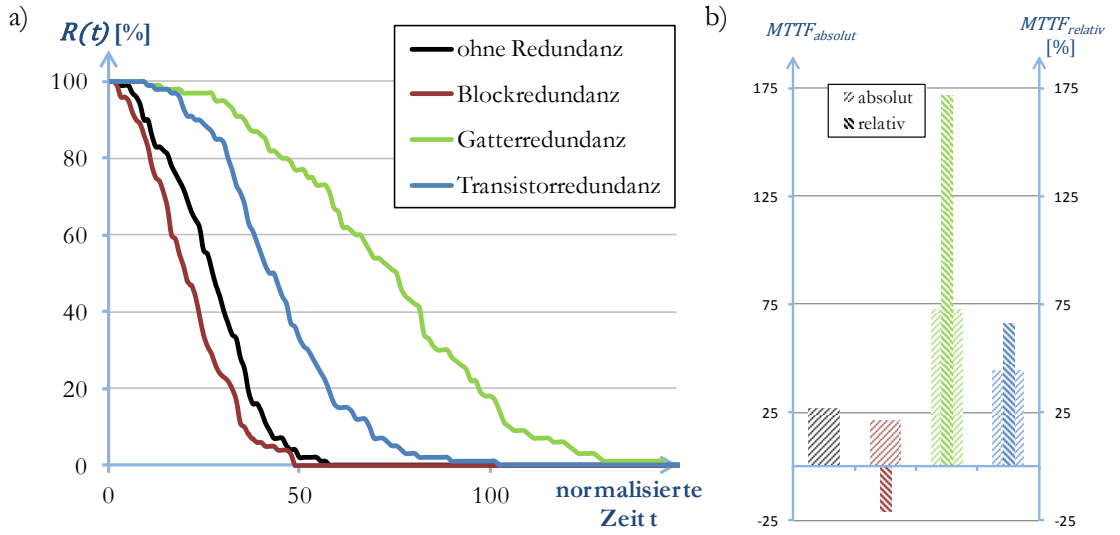


Abbildung 4-11 : Simulationsergebnisse der redundanten Multiplizierer
a) Zuverlässigkeit *R(t)*
b) *MTTF*: absolut und relativ zum Basismultiplizierer

Die Unterschiede zwischen beiden Implementierungen liegt in der Verdopplung der Transistorstacks, wie es in Abbildung 4-12 a) und b) erkennbar ist. Bei der Transistorverdopplung sind die beiden Stacks miteinander verbunden, während bei der Gatterredundanz durch die separate Verdopplung der Stacks diese keine elektrische Verbindung aufweisen außer am Gatterausgang (Output) und an den Masse- bzw. Versorgungsspannungsleitungen. Parallele Transistorstacks werden in beiden Redundanzstrategien auf die gleiche Weise verdoppelt. In Abbildung 4-12 c) wird der Unterschied zwischen einem korrekt funktionierendem Stack und einem identisch aufgebautem Stack, dessen unterer Transistor allerdings von einem Gateoxiddefekt betroffen ist, deutlich. Vor dem Schaltzeitpunkt t_0 ist das Ausgangsnetz korrekt auf einen hohen Spannungspegel, der V_{DD} entspricht, aufgeladen, während die anderen beiden Netze beider Stacks auf V_{SS} entladen sind. Nach t_0 steigt der Eingang auf V_{DD}, was zu einem Entladungsvorgang des Ausgangsnetzes führt. Als erstes steigen die Spannungspegel der Netze Netz1 und $\text{Netz1}_{\text{DEFEKT}}$ durch den Entladestrom an. Dadurch steigt die Spannung zwischen Drain und Source des mittleren Transistors und damit auch der Spannungspegel auf Netz2 und $\text{Netz2}_{\text{DEFEKT}}$. Dies wiederum führt zu einem Stromfluss durch den dritten Transistor und damit zu einer Verbindung mit V_{SS}. Dadurch werden die Kapazitäten der Netze Netz2 und

$Netz2_{DEFEKT}$ und somit auch Netz1 und $Netz1_{DEFEKT}$ wieder entladen, was den Entladungsvorgang des Ausgangsnetzes komplettiert. Durch den Gateoxiddefekt des unteren Transistors wird dessen Gate auf eine Spannung aufgeladen, die dessen Defekt entspricht und die kleiner ist als V_{DD}. Das führt zu einem nicht voll durchgeschalteten Transistor und damit einerseits zu einer verlangsamten Flanke und andererseits zu einem Spannungspegel der Netze $Netz1_{DEFEKT}$ und $Netz2_{DEFEKT}$, die größer sind als die kleinstmögliche Massespannung V_{SS}. Wenn nun ein Defekt in einem Transistorstack auftaucht, wird der verdoppelte und intakte Stack eines Gatters, das durch die Gatterredundanz verdoppelt wurde, nur über das Ausgangsnetz beeinflusst, während der defekte Stack einen Einfluss auf die verdoppelten Transistoren der Transistorredundanz durch die Verbindungen zwischen ihnen ausübt. Dadurch werden auch diese Transistoren des redundanten Stacks nicht vollständig geöffnet.

Aus diesem Grund erzielt die Gatterredundanz zuverlässigere und schnellere Systeme, was durch die *MTTF*-Resultate aus Abbildung 4-11 c) erkennbar ist, da die Gatterredundanz eine *MTTF* um über 170 % gegenüber dem Originaldesign steigern kann, während die Transistorredundanz nur gut 65 % erreicht. Ähnliche Verhältnisse sind auch in Abbildung 4-13 a)

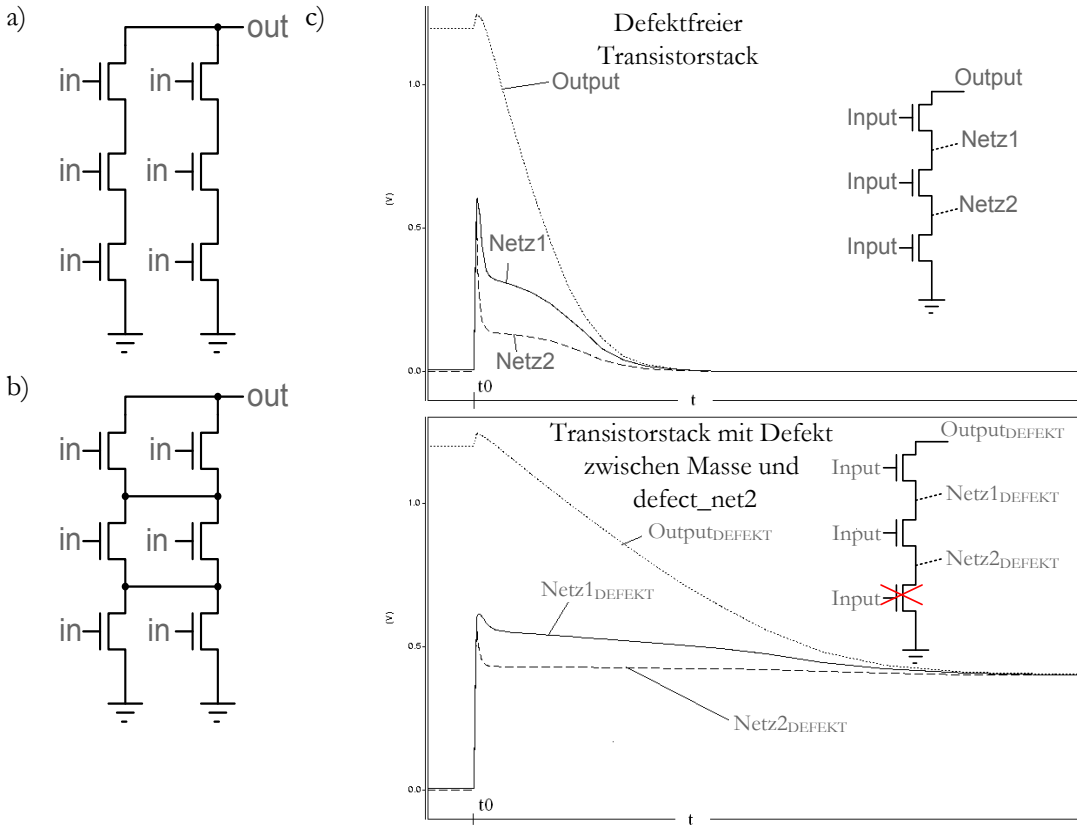


Abbildung 4-12 : Unterschiede der Transistor- und Gatterredundanz
a) Verdoppelter Transistorstack der Gatterredundanz
b) Verdoppelter Transistorstack der Transistorredundanz
c) Spannungsverläufe der Netze eines defektfreien und eines Transistorstacks mit Defekt im untersten Transistor (markiert mit einem roten Kreuz)

zu sehen. Hier ist die Veränderung der Verzögerungszeit des Multiplizierers über der Zeit dargestellt, wobei die unterschiedlichen Endpunkte der Kurven durch die divergierende Zuverlässigkeit gegeben sind. Aufgrund der konstanten Fehlerrate steigt die Anzahl der Fehler mit der Zeit an, und somit auch die Anzahl der Gatter mit einer verlangsamten Verzögerungszeit im Gegensatz zum ursprünglichen Zustand. Dadurch entsteht eine lineare Abhängigkeit der Gesamtverzögerungszeit von der Anzahl der Fehler bzw. der Zeit, was eine fortschreitende Verschlechterung – „graceful degradation" – bei allen Multiplizierern darstellt. Die Originaldesigns werden schnell langsamer, da ihnen die Möglichkeit fehlt, die vorhandenen Defekte zu kompensieren. Das Delay der Multiplizierer mit Blockredundanz steigt noch schneller an. Dies resultiert aus den konkurrierenden Ausgangssignalen, weil mit steigender Fehleranzahl die Verzögerungszeit der beiden Blöcke immer stärker voneinander abweicht und so für das Ausbilden eines korrekten Pegels der gemeinsamen Ausgangssignale mehr Zeit in Anspruch nimmt, sofern die Pegel nicht unbestimmt bleiben. Durch die oben ausgeführte langsamere (Ent-)Ladungszeit der Gatterausgangsnetze werden auch die Multiplizierer mit Transistorredundanz langsamer als die mit Gatterredundanz, wobei der Unterschied in der Verzögerungszeit gering ausfällt, was aufgrund des ähnlichen Aufbaus erwartbar war.

In Abbildung 4-13 b) wird ein ähnliches Verhalten deutlich. Hier ist die Gesamtleistung über die Zeit abgebildet. Durch die erhöhte statische Leistungsaufnahme der defekten Transistoren steigt auch die Gesamtleistungsaufnahme, bis die statischen Ströme durch die Defekte die Gesamtleistung dominieren, womit der fast gleiche lineare Anstieg beider Redundanzstrategien zu erklären ist. Dies ist in Abbildung 4-13 c) dargestellt, wo der relative Anteil der statischen

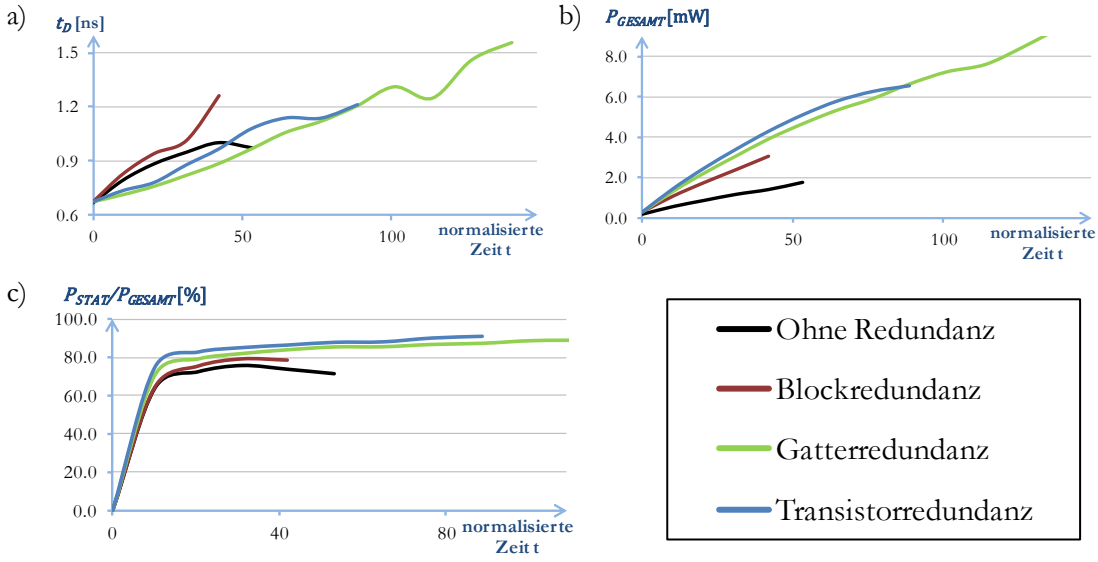


Abbildung 4-13 : „Graceful Degradation"-Verhalten der verschiedenen redundanten Implementierungen des 4x4-Wallace-Multiplizierers mit ansteigender Defektzahl

a) Verlauf der Verzögerungszeit t_D über die Zeit
b) Verlauf der gesamten Leistungsaufnahme P_{GESAMT} über die Zeit
c) Steigende Anteil der statischen Leistungsaufnahme an P_{GESAMT} über die Zeit

Leistungsaufnahme P_{STAT} an der Gesamtleistung P_{GESAMT} über die Zeit abgebildet ist. Schon nach $t = 10$ steigt der Anteil der statischen Ströme auf über 70 % für alle Designs, für Gatter- und Transistorredundanz sogar auf über 80 %. Weil diese beiden Designs fehlerhafte Transistoren besser kompensieren können, ist die Leistungsaufnahme und der Anteil der statischen Leistungsaufnahme generell größer als bei den anderen beiden Designs, da nur funktionierende Schaltungen für die Abbildungen 4-13 berücksichtigt wurden.

Die Strategie mit redundanten Gattern stellt sich somit am vorteilhaftesten heraus. Einerseits ist die Zuverlässigkeit deutlich höher als bei der Blockredundanz und beim Originaldesign. Andererseits steigen auch die Verzögerungszeit und die Leistungsaufnahme langsamer mit der Anzahl der Fehler als bei der Transistorredundanz. Ein weiterer Grund, um diese Strategie weiter zu verfolgen, um zuverlässigere Schaltungen zu erreichen, ist die gute Integrationsmöglichkeit von redundanten Gattern in den Designflow [Sae09b]. Daher dienen die vorgestellten Untersuchungen als Basis für Erweiterungen der Gatterredundanz in den nächsten Abschnitten.

4.3 Gezielte Zuverlässigkeitssteigerung zur Betriebszeit

Ein wesentlicher Nachteil der Redundanz hinsichtlich permanenter Fehler ist der gleichzeitige Verschleiß der Transistoren des Originaldesigns als auch der zusätzlichen Transistoren beginnend mit der Betriebszeit, da sowohl Original und Kopie der gleichen Belastung durch die Ladungen und elektrischen Felder ausgesetzt sind. Dementsprechend ist die Wahrscheinlichkeit eines Defektes in beiden Fällen gleich, wenn man gleiche Transistoren nach der Produktion voraussetzt. Der folgende Ansatz berücksichtigt dies, in dem redundante Teile während des Betriebes von der Schaltung elektrisch getrennt bleiben – nachfolgend als Standby-Phase betitelt. Bei Bedarf, z. B. in dem logische Fehler oder zu stark verzögerte Ergebnisse von höheren Designebenen detektiert werden, werden die redundanten Elemente in der Redundanz-Phase zugeschaltet und können dann zur Stabilisierung der Spannungspegel und zur Reduzierung der Verzögerungszeitverzögerung beitragen [Sae12b]. Aus den Erfahrungen der verbesserten Zuverlässigkeit von Transistorstacks gegenüber redundanten Transistoren, die im vorigen Kapitel erläutert wurden, werden komplette Transistorstacks verdoppelt, wenn redundante Elemente eingefügt werden sollen.

Um den relativen Flächenzuwachs möglichst gering zu halten, werden einfache Transistoren als Schaltelemente den komplexeren Transmissiongates bevorzugt. Der zu erwartende Spannungsabfall durch die Schwellspannung fällt nicht so sehr ins Gewicht, da sie nur als Unterstützung dienen. Diese Schalt-Transistoren werden zwischen den Eingangsnetzen und den Gates des redundanten Transistorstacks implementiert. Im geschlossenen Zustand verhindern diese Schalt-Transistoren, dass das elektrisches Feld über dem redundanten Stack diesen nicht so beansprucht wie die Originaltransistoren. Um den Stack gänzlich von der Schaltung während der Standby-Phase zu trennen, ist ein zusätzlicher Transistor zwischen Ausgangsnetz und Stack notwendig – der Stack-Transistor. Diese Differenzierung ist notwendig, da beide zusätzlichen „Transistorarten“ unterschiedlichen Anforderungen unterliegen. In Abbildung 4-14 wird

beispielhaft der n-MOSFET-Stack eines NAND2-Gatters nach der Maßgabe dieses Ansatzes verdoppelt. Während der Standby-Phase werden die Schalt- und der Stack-Transistor ausgeschaltet, was zu undefinierten Zuständen an den Gates der redundanten Transistoren führt. Es bildet sich dort kein elektrisches Feld aus, das den Verschleiß dieser Transistoren auslösen könnte. In der Redundanzphase wiederum werden die Spannungspegel der Gattereingänge über die geöffneten Schalt-Transistoren zu den Gates der redundanten Transistoren weitergeleitet. Die Ausgangslast kann dann über den geöffneten Stack-Transistor und den verdoppelten Transistorstack entladen werden. So wird die fallende Ausgangsflanke t_{FALL_OUT} beschleunigt und der minimale Spannungspegel V_{OUT_MIN} am Ausgangsnetz verringert. Beide Effekte vermindern eventuelle Verschleißeffekte an potentiell beschädigten Originaltransistoren. Im Falle von verdoppelten p-MOSFET-Stacks werden andererseits die steigende Ausgangsflanke t_{RISE_OUT} und der maximale Spannungspegel V_{OUT_MAX} des Ausgangsnetzes verbessert.

Natürlich werden mit den Schalt- und Stacktransistoren potentielle Fehlerquellen hinzugefügt, allerdings sind sie während der Betriebsphase, in der die Stacks noch nicht zugeschaltet sind, auch nicht offen, so dass ihr Verschleiß nach Gleichung (65) erst mit dem Betrieb der redundanten Stacks beginnt.

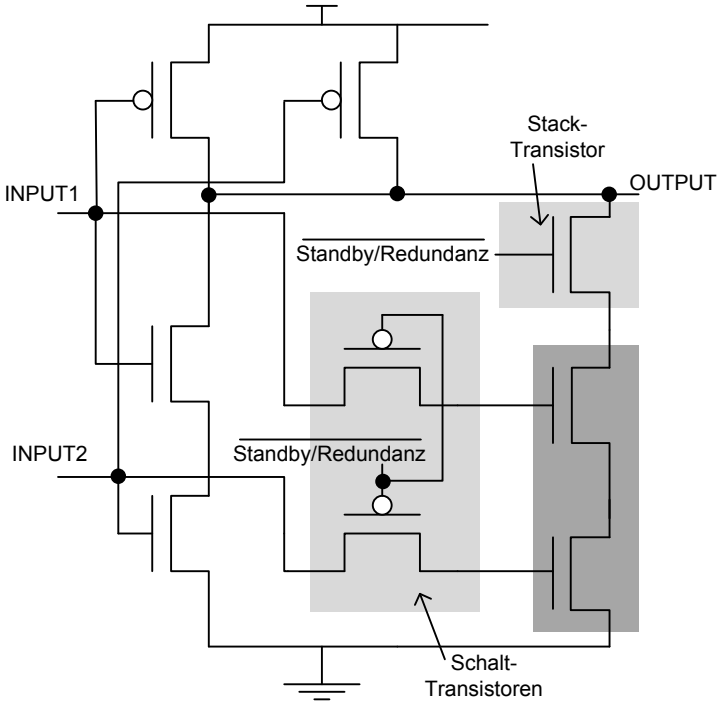


Abbildung 4-14 : NAND2-Gatter mit redundantem n-MOSFET-Transistorstack

4.3.1 Parametrisierung der Kontrolltransistoren

Als erstes musste nun die möglichst beste Anpassung der Transistorparameter (Breite, Typ) in Hinblick auf alle Schaltungsparameter, wie Fläche, Leistungsaufnahme und Verzögerungszeit, vorgenommen werden. Obwohl alle drei Schaltungsparameter beachtet wurden, wurde nur der Einfluss auf die Verzögerungszeit untersucht. Diese Vorgehensweise resultiert aus dem Schluss, dass aus Sicht der Fläche und der Leistungsaufnahme minimal dimensionierte Schalt- und Stack-Transistoren die bessere Wahl sind. Neben der direkten Auswirkung beider Transistorarten auf die Fläche, wird die Leistungsaufnahme mittelbar durch die Dimensionierung der Schalt-Transistoren beeinflusst. Da diese zusätzlich ins originale Design eingefügt werden, wird automatisch die Lastkapazität für das treibende Gatter erhöht und damit nach Gleichung (24) die Leistungsaufnahme. In der Standby-Phase wird die ursprüngliche Lastkapazität um den Wert der Drainkapazitäten gesteigert, in der Redundanz-Phase zusätzlich um den Wert der verdoppelten Transistorstacks, so dass der verdoppelte Stack diesen Zusatzaufwand dominiert, wenn der Schalt-Transistor so klein wie möglich ist. Durch diese Rückschlüsse auf Flächenbedarf und Leistungsaufnahme tendierte die Auswahl der Transistorbreite eindeutig in Richtung klein dimensionierter Schalt- und Stack-Transistoren. Ähnlich verhielt es sich in Betrachtung der Verzögerungszeit. Eine kleinere Lastkapazität ist gleichbedeutend mit einer schnelleren Umladung, was auch durch Gleichung (20) deutlich wird. Da durch die Schalt-Transistoren kein Auf- bzw. Entladepfad verläuft, kongruiert eine minimale Dimensionierung für eine geringere Verlangsamung der Verzögerungszeit in der Standby-Phase mit dem verringerten Flächenzuwachs und der geringeren Leitungsaufnahme. Da aber während der Redundanz-Phase die Auf- bzw. Entladung durch den Stack-Transistor geleitet wird, ist eine Minimaldimensionierung aus Sicht der Verzögerungszeit wohl nur in den wenigsten Fällen die beste Wahl und damit auch zum Teil in Hinblick auf die Erhöhung von Querströmen. Ferner wird vor allem der Gateoxiddefekt der entscheidende Faktor für die Verzögerungszeit in dieser Phase sein. Denn der Einfluss der verlangsamten Treiberfähigkeit des defekten Transistors auf die Verzögerungszeit der Gatterreihe ist signifikant, wenn sie nicht durch den redundanten Gegenpart aufgefangen und somit die gestörte Flanke unterstützt werden kann.

Für die Transistortypen der Schalt-Transistoren (p-MOSFET oder n-MOSFET) war aufgrund der unterschiedlichen Weiterleitung von high- oder low-Pegeln auch nicht vorab ersichtlich, welche Wahl die bessere ist, während die Stack-Transistoren definitiv den gleichen Typ wie die redundanten Stacks haben mussten.

Aus den genannten Unsicherheiten wurden zahlreiche Simulationen mit unterschiedlichen Parametereinstellungen (Transistorbreite, Transistortyp) für Stack- und Schalt-Transistoren durchgeführt, damit deren negativen und positiven Auswirkungen auf die Schaltungsparameter während beider Phasen ermittelt werden konnten. Dazu wurde ein Inverter in mehreren Treiberstärken mit dem zu schaltenden Eingang des Gatters mit den redundanten Elementen verbunden. Am Ausgang wurden wiederum Inverter verschiedener Treiberstärken getrieben. Die Schalt- und Stack-Transistoren wurden mit unterschiedlichen Transistorbreiten und -typen getestet und die Verzögerungszeit, sowie die Spannungspegel am Ausgang dokumentiert. Aus den

umfassenden Ergebnissen unter besonderer Beachtung der Verzögerungszeit in der Redundanz-Phase wurden folgende Regeln für die Dimensionierung der Transistorparameter aufgestellt:

Tabelle 4-4 : Parametereinstellungen für Schalt- und Stack-Transitoren

	Breite	***Typ***
Schalt-Transistor	minimal	Invers dem redundanten Stack
Stack-Transistor	Gleich dem redundanten Stack	

4.3.2 Auswahl der zu verdoppelnden Transistorstacks

Zwei Kriterien wurden angelegt, um ein Auswahlverfahren der zu verdoppelnden Transistorstacks zu ermöglichen. Das erste ist eine Grenze für die Wahrscheinlichkeit P_{ON}, mit der ein Transistor während der Betriebszeit durchgeschaltet ist, weil die Lebensdauer eines Transistors nach (65) von P_{ON} abhängig ist. Zwei Grenzen (0.5; 0.66) wurden festgelegt, wobei $P_{ON} > 0.5$ bedeutet, dass es nur darauf ankommt, ob der p-MOSFET oder der n-MOSFET eines Netzes eine höhere Wahrscheinlichkeit hat, dass er offen ist. Sollte bei einem Gatter mit zwei Eingängen jeweils ein Transistortyp das Kriterium erfüllen, wurde der Transistorstack ausgewählt, dessen Transistoren in Reihe geschaltet sind. Bei $P_{ON} > 0.66$ werden nur die Stacks gewählt, deren Transistoren öfter als zwei Drittel der Betriebszeit durchschalten. Das zweite Kriterium beachtet die Verzögerungszeit t_D der gesamten Schaltung. Bei der Auswahl wurden nur die Pfade der Schaltung berücksichtigt, deren Verzögerungszeit $t_{Pfad} \geq p_D \cdot t_D$ ist. Stacks von Gattern, die nicht Teil dieser Pfade sind, wurden nicht verdoppelt. Drei Werte waren möglich: 0; 0.75 und 0.9. Ein Wert von $p_D = 0$ bedeutet dabei, dass alle Gatter ausgesucht werden können, während bei einem Wert von $p_D = 0.9$ nur Stacks von Gattern verdoppelt werden können, die Teil eines Pfades sind, dessen Verzögerungszeit $t_{Pfad} \geq 0.9 \cdot t_D$ ist. Folgende sechs verschiedene zuverlässigkeitssteigernde Szenarien Enh1 bis Enh6 (Enh – Enhancement, engl. Verbesserung) wurden dadurch definiert:

Tabelle 4-5 : Parametereinstellungen für die zuverlässigkeitssteigernden Szenarien

Szenario	P_{ON}	p_D
Enh1	0.5	0
Enh2	0.5	0.75
Enh3	0.5	0.9
Enh4	0.66	0
Enh5	0.66	0.75
Enh6	0.66	0.9

4.3.3 Negative Auswirkungen in der Standby-Phase

Es wurden vier verschiedene Schaltungen auf Transistorebene untersucht: ein 16-Bit Ripple-Carry-Addierer (rca16), ein 16-Bit Carry-Look-Ahead-Addierer (cla16) und zwei ISCAS-Designs (c432, c1908). Die Schaltungsparameter der Originaldesigns sind nachfolgend aufgelistet:

Tabelle 4-6 : Parametereinstellungen für die zuverlässigkeitssteigernden Szenarien mit Gesamtfläche der Transistorgates A_{GATE}, Gesamtleistungsaufnahme P_{GESAMT} und Schaltungsverzögerungszeit t_D

Design	*Schaltungsparameter*		
	Gesamt-A_{GATE} [nm²]	P_{GESAMT} [nW] bei 10 MHz	t_D [ns]
c432	25.8	5.4	0.616
rca16	43.5	11.6	0.445
cla16	55.9	12.3	0.306
c1908	60.1	14.1	0.917

Nicht überraschend werden alle drei Schaltungsparameter negativ in der Standby-Phase beeinflusst. Den größten Zuwachs verlangen die Szenarien Enh1 und Enh4, bei denen keine Restriktionen in Hinblick auf die Verzögerungszeit bestehen, da bei diesen Szenarien die meisten Stacks verdoppelt wurden. In den Abbildung 4-15 a) ist die relative Fläche zum Originaldesign für alle Szenarien dargestellt. Enh1, bei dem alle Gatter einen verdoppelten Stack erhalten, hat den größten Flächenzuwachs, welcher zwischen 60 % und 76 % liegt. Die Fläche des Enh4-Szenarios liegt ungefähr bei 140 % zum Originaldesign. Nur beim Ripple-Carry-Addierer rca16 werden nur 25 % hinzugefügt, da in dieser Schaltung die Verzögerungszeiten der einzelnen Designpfade nicht so ausgeglichen sind wie in den anderen Designs. Die anderen Szenarien liegen zwischen 5 % (rca16) und 20 % (cla16) Flächenzuwachs.

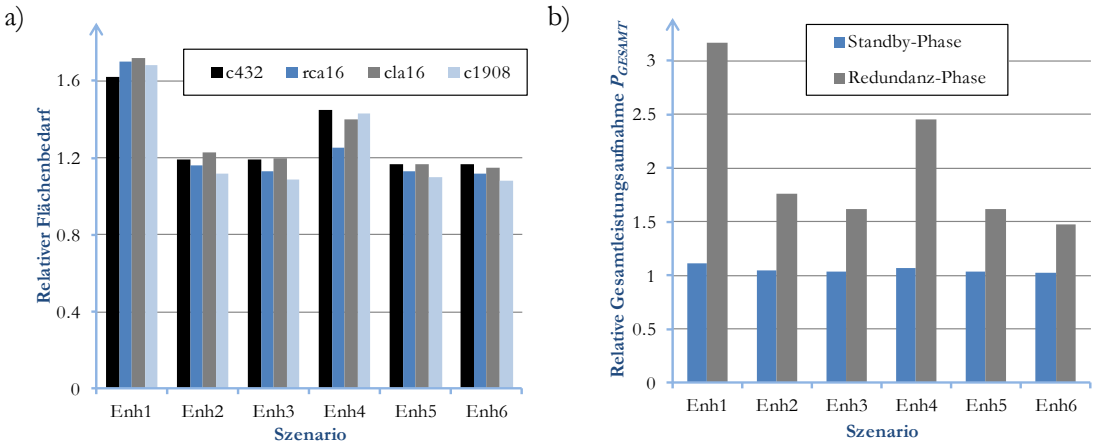


Abbildung 4-15 : Zusatzaufwand relativ zum jeweiligen Originaldesign
a) Fläche aller Designs
b) Gesamte Leistungsaufnahme des Designs cla16 in beiden Phasen

Aufgrund der erhöhten Lastkapazitäten steigert sich auch die Leistungsaufnahme. Der größte negative Einfluss wurde beim Carry-Look-Ahead-Addierer cla16 festgestellt, da hier aufgrund der ausgeglichenen Designpfade auch der Flächenzuwachs zumeist am größten ist. Aus diesem Grund ist die relative Leistungsaufnahme zum Originaldesign für cla16 in Abbildung 4-15 b) dargestellt. Die größte Leistungserhöhung liegt bei 12 % (Enh1). Zusätzlich zur Leistungsaufnahme in der Standby-Phase wurde auch die der Redundanz-Phase abgebildet. Wie zu sehen, steigert sich die Leistungsaufnahme in der Redundanzphase um mindestens 50 % (Enh6), im schlechtesten Fall (Enh1) erhöht sie sich auf mehr als das Dreifache. Allein hieran ist zu erkennen, dass die Redundanzphase nur im Fehlerfall und nicht während der „normalen" Anwendung eingesetzt werden sollte, weil sonst die Leistungsaufnahme und damit mittelbar auch die Zuverlässigkeit über die Temperatur negativ beeinflusst wird. In der Anwesenheit von Defekten allerdings schwindet der zusätzliche Anteil der eingefügten Transistoren an der Gesamtstromaufnahme durch die Dominanz der statischen Ströme durch die Defekte, was gut in Abbildung 4-16 zu beobachten ist. Hier ist die Entwicklung der durchschnittlichen gesamten Leistungsaufnahme der Originale und der Enh1-Designs, welche den größten redundanten Teil aller Szenarien hat, für alle vier Schaltungen dargestellt. Im Laufe der Zeit werden weitere Defekte eingefügt. Die Abweichungen zwischen Original und Enh1-Design sinken von anfänglich über 200 % auf am Schluss 20 % (c1908), 12.5 % (c432) und unter 8 % für die beiden Addierer.

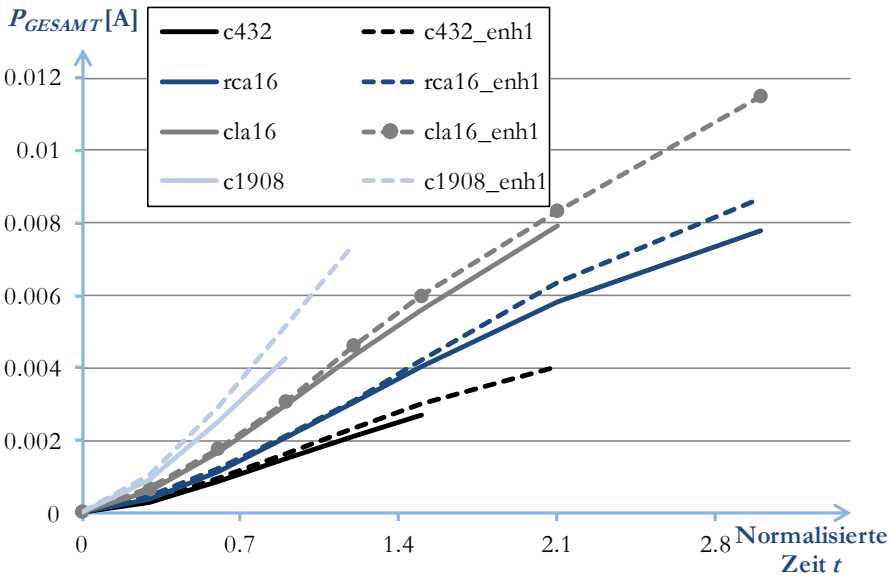


Abbildung 4-16 : Durchschnittliche Gesamtleistungsaufnahme der größten Szenarien und der Originaldesigns über die Zeit

Schließlich steigt, wie oben erwähnt, während der Standby-Phase auch die Verzögerungszeit an, da die Lastkapazitäten für einige Gatter vergrößert werden. Im schlechtesten Fall (Enh1)

liegen die Abweichungen bei +13.7 % und lassen sich je nach Design und Szenario auf bis zu 2 % senken (Enh6 bei c1908). Nachfolgend sind alle Werte aufgeführt:

Tabelle 4-7 : Relative Verzögerungszeit der Designs in der Standby-Phase

Design	*Szenario*					
	Enh1	**Enh2**	**Enh3**	**Enh4**	**Enh5**	**Enh6**
c432	1.137	1.069	1.069	1.125	1.069	1.069
rca16	1.132	1.083	1.083	1.094	1.083	1.083
cla16	1.123	1.104	1.103	1.111	1.095	1.095
c1908	1.137	1.024	1.019	1.100	1.024	1.020

4.3.4 Zuverlässigkeitssteigerung mit dem Wechsel in die Redundanzphase

Um die Auswirkungen auf die Zuverlässigkeit zu untersuchen, wurden alle Designs und Szenarien mit Defekten über der Zeit simuliert. Dabei wurde für jeden Transistor ein Defekteintrittszeitpunkt t_{GOB} nach den Gleichungen (60), (61) und (65) berechnet. Das Ausmaß R_{GOB} des Defektes veränderte sich mit der Zeit. Dieser Wert wurde gemäß Gleichung (53) entsprechend des jeweiligen untersuchten Zeitpunktes angepasst. Ferner wurden diese Werte nur einmal für die Ursprungsschaltung berechnet und galten auch für alle Derivate Enh1 bis Enh6 für den jeweiligen Simulationslauf. So wurden vergleichbare Bedingungen in jedem Simulationslauf hergestellt. Redundante Transistoren wurden generell defektfrei gehalten. Somit stellt jeder Zeitpunkt der simulierten Enh-Designs den Wechsel in die Redundanzphase dar, wenn die Ursprungsschaltung schon vom Verschleiß betroffen war, die redundanten Stacks allerdings nicht, weil sie noch nicht zugeschaltet worden sind.

Bei jedem Szenario konnten Zuverlässigkeitssteigerungen beobachtet werden. Beispielhaft

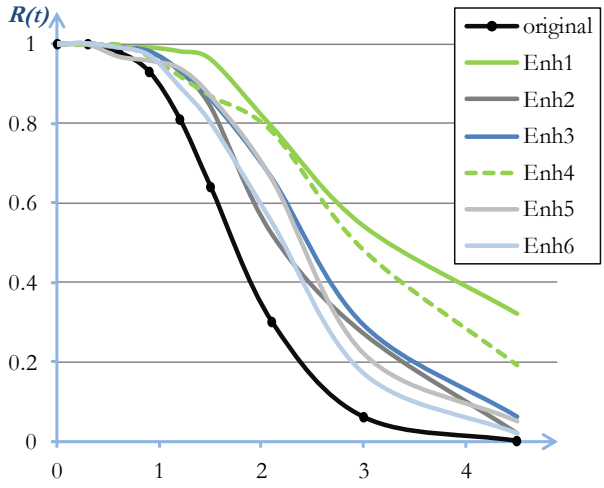


Abbildung 4-17 : Zuverlässigkeitskurve der aller Szenarien des Designs rca16

sind die Zuverlässigkeitskurven von rca16 in Abbildung 4-17 aufgetragen. Die Zuverlässigkeit der Originalschaltung ist als schwarze Linie dargestellt. Die besten Ergebnisse werden von den Szenarien mit den meisten redundanten Stacks erreicht (Enh1, Enh4). Zum Beispiel war zum Zeitpunkt $t = 4.5$ keines der originalen Designs funktionstüchtig, während ein Drittel aller Enh1-Schaltungen noch korrekte Ausgaben produzierten. Auch Enh4-Designs zeigten eine gute Robustheit gegenüber dem Gateoxiddefekten, so liegt die Wahrscheinlichkeit, dass die Schaltungen zum letzten simulierten Zeitpunkt $t_{ENDE} = 4.5$ noch korrekt arbeiten bei ca. 20 %. Simulationen nach diesem Zeitpunkt wurden nicht mehr durchgeführt, damit die gesamte Simulationszeit nicht zu groß wurde. Allerdings war der Ripple-Carry-Addierer rca16 die einzige Schaltung, die zu diesem Zeitpunkt noch zuverlässige Ausgaben errechnete, so dass diese Grenze nur bei rca16 überschritten wurde.

Ein Blick auf die relative *MTTF* in Abbildung 4-18 im Vergleich zum Original zeigt Verbesserungen bei jedem Szenario und jeder Schaltung. Auch hier ist klar der Vorteil von den Enh4- und vor allem von den Enh1-Szenarien durch die größere Anzahl an redundanten Stacks in Hinblick auf die Zuverlässigkeit zu erkennen. Steigerungen von 63 % (c1908) bis zu 76 % (rca16) sind bei Enh1-Szenarien möglich. Nur die Verbesserung bei c432 beläuft sich auf vergleichsweise niedrige 33 %. Mit den Szenarien, die weniger zusätzliche Fläche verbrauchen (Enh2, Enh3, Enh5, Enh6), werden Zuverlässigkeitssteigerungen zwischen 3 % und 35 % erreicht. Sie sind aber nur eine Alternative, wenn starke Flächenrestriktionen zu beachten sind. Ein sehr guter und kostengünstiger Kompromiss in Bezug auf Fläche, Leistungsaufnahme und Verzögerungszeit stellt das Enh4-Szenario dar, da hier mit geringerer Fläche als bei Enh1 die Zuverlässigkeit um bis zu 65 % (c1908) gesteigert werden kann. Bei größeren Schaltungen können die Einschränkungen hinsichtlich des kritischen Pfades sehr nützlich sein. Das deuten die besseren Ergebnisse der Enh5- und Enh6-Szenarien im Vergleich zu den Enh2- und Enh3-Designs an. Hier werden mit weniger Flächenzuwachs gleiche bis bessere Zuverlässigkeitssteigerungen erreicht.

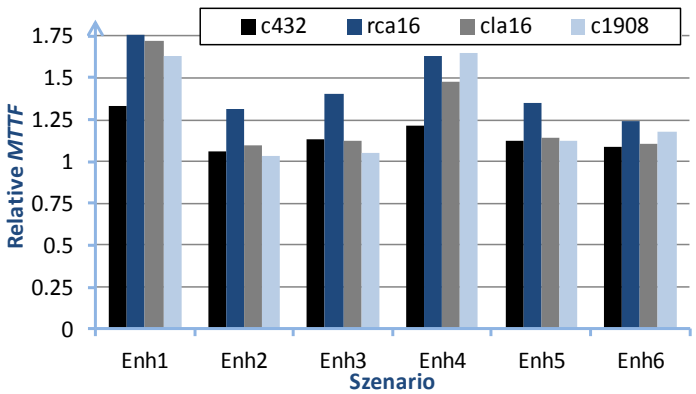


Abbildung 4-18 : MTTF aller Designs und Szenarien relativ zu den Originalschaltungen

4.3.5 Verlangsamung des Verschleißes der Verzögerungszeit in der Redundanzphase

Die im Vergleich geringen Verbesserungen der c432-Schaltungen in Bezug auf die *MTTF* sind dadurch zu erklären, dass die Defekte hier vor allem eher das Timing verändern, als dass falsche Spannungspegel an den Netzen erreicht werden. Dieser Effekt wird in Abbildung 4-19 a) ersichtlich, wo die durchschnittliche Verzögerungszeit aller Szenarien abgebildet ist. Deutlich wird hier, dass zum Zeitpunkt $t = 1.5$ die Verzögerung zweimal so groß ist wie zum Zeitpunkt $t = 0$. Bei einem vorgegebenen Taktsignal mit einer Taktperiode von 1 ns wären 50 % aller Originalschaltungen zu langsam und würden fehlerhafte Ausgaben produzieren. Im Gegensatz dazu, sind alle Szenarien bis auf Enh2 schneller als das Original. Verbleibend bei dem Beispiel mit $f_{CLK} = 1$ GHz würden ein durchschnittliches Design der Szenarien Enh1 und Enh3 noch bis $t = 1.2$ korrekt operieren, während die durchschnittliche Verzögerungszeit der Enh4-Schaltung sogar noch bis $t = 1.5$ genügend klein sein würde. Daraus ist zu erkennen, dass die *MTTF* weiter gesteigert werden könnte, wenn noch zeitliche Restriktionen in Form eines Taktsignals beachtet werden müssten. Um dies deutlicher aufzuzeigen, ist die Entwicklung der zum Original relativen Verzögerungszeit der Enh6-Szenarios (rca16) in einem Boxplot-Diagramm in Abbildung 4-19 b) über die Zeit dargestellt. Verglichen mit anderen Ergebnissen stellen diese nicht den besten Fall dar, verdeutlichen aber sehr gut die Verlangsamung des Verschleißes der Verzögerungszeit. Bei diesem Diagramm ist die Verzögerungszeit der Schaltung für die Simulationszeitpunkte in vier Bereiche zusammengefasst dargestellt, falls sie noch korrekt funktioniert. Die Verzögerungszeit wurde für jeden Simulationslauf jeweils relativ zur Verzögerungszeit des Originaldesigns gespeichert. Danach wurden die Werte ansteigend angeordnet und zu einem der vier Blöcke zugewiesen, wobei die Anzahl der Werte je Block gleich ist. So beinhaltet jeder Bereich ein Viertel aller zugrunde gelegten Schaltungen. Der weiße Bereich wird als unteres Quartil, der graue als oberes Quartil bezeichnet. Dazwischen befindet sich der Median, der den Datenbereich zur

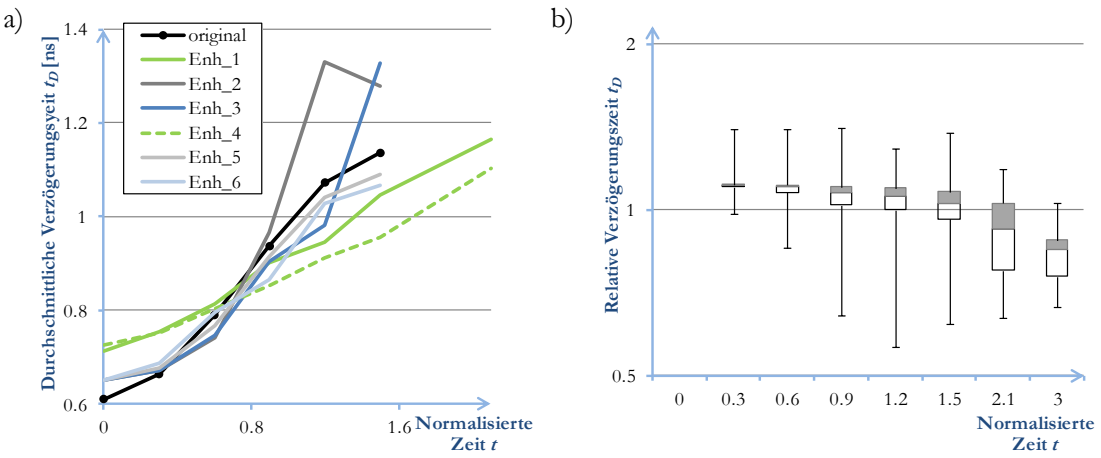


Abbildung 4-19 : Entwicklung der Verzögerungszeit t_D über die Zeit t
a) Durchschnittliche Verzögerungszeit aller Szenarien des Designs c432
b) Relative Verzögerungszeit des Szenarios Enh6 zu t_D des Originals rca16

Hälfte aufteilt. Das obere und untere Ende der Antennen geben jeweils die Extremwerte wieder, also den größten bzw. den kleinsten Wert zu dem jeweiligen Simulationszeitpunkt. In dieser Abbildung 4-19 b) ist gut die Verlangsamung des „graceful degradation"-Verhaltens durch die redundanten Stacks zu erkennen. So verkleinert sich der Wert des Medians mit jedem Zeitschritt. Ab Zeitpunkt $t = 2.1$ sind annähernd 75 % der redundanten Schaltungen schneller als das Original, ab $t = 3.0$ so gut wie alle noch funktionierenden Designs. So ist auch hier die Verbesserung der Verzögerungszeit in Anwesenheit von Defekten deutlich erkennbar.

4.3.6 Vergleichende Untersuchungen für taktsynchrone Logik

Um nun sowohl den Aspekt der vorgegebenen Zeitverzögerung aufgrund taktsynchroner Schaltungen einzubeziehen, als auch die vorgestellten Verbesserungen in den Kontext mit anderen aber ähnlichen Ansätzen zu vergleichen, wurden weitere Untersuchungen durchgeführt. Hierbei wurden fünf Basisschaltungen mit fünf verschiedenen zuverlässigkeitssteigernden Implementierungsstrategien über die Zeit simuliert. Dabei wurden Verschleißdefekte in die Netzlisten eingefügt, wie sie im vorigen Abschnitt vorgestellt wurden. Die Designs umfassten drei ISCAS-Schaltungen (c432, c1908, c3540) und zwei 16-Bit-Addierer mit unterschiedlichen Berechnungsformen: ein Ripple-Carry-Addierer rca16 und ein Carry-Look-Ahead-Addierer cla16. Diese Designs wurden mit der schon vorgestellten Methode Enh4 verstärkt. Dabei rangierte der Zusatzaufwand im Vergleich zu den Originalschaltungen zwischen 25 % (rca16) bzw. 41 % (cla16) bei den Addierern und 50 % (c432, c1908) bzw. 59 % (c3540) bei den ISCAS-Schaltungen. Die ermittelte zusätzliche Fläche wurde als Grenze für zwei weitere Implementierungen genutzt. Zum einen wurden solange redundante Transistoren an den Stellen eingefügt, die am kritischsten hinsichtlich TDDB waren, bis das Flächenlimit erreicht wurde. Dieser Ansatz, im Folgenden DTrans genannt, orientiert sich an den Shadow Transistoren [Cor08]. Zum anderen wurden die gefährdetsten Gatter solange verdoppelt, bis ebenfalls die jeweilige Grenze erreicht wurde. Dieser Ansatz DGatter orientiert sich an der kompletten Gatterverdopplung aus Kapitel 4.3 mit einer vorgegebenen Flächenbegrenzung. Zusätzlich zu Enh4 wurden noch zwei weitere Implementierungen mit den vorgestellten redundanten Transistorstacks simuliert: um den Aspekt der Schaltungsverzögerungszeit zu berücksichtigen, die schon vorgestellte Strategie Enh5 und um den Einfluss einer restriktiveren Auswahl der gefährdetsten Gatter zu untersuchen, eine Strategie Enh7 mit $P_{ON} = 0.8$ und $p_D = 0$. Beide Implementierungsarten verursachten weniger Flächenaufwand als Enh4, da weniger Transistorstacks verdoppelt wurden. Bei Enh7 lag die zusätzliche Fläche von der Originalschaltungsfläche aufsteigend zwischen 90 % bis 96 % relativ zu Enh4, während bei Enh5 die größeren Designs weniger Zusatzfläche in Anspruch nahmen als die kleineren, da hier (rca16, c432) der Anteil der Gatter an Pfaden, die am zeitkritischsten sind, größer ausfällt als bei c1908 und c3540. So wurden für Enh5 nur zwischen 67 % und 90 % des Mehraufwandes von Enh4 gebraucht. Abbildung 4-20 zeigt nun die Verzögerungszeit und die gesamte Leistungsaufnahme der Schaltungen im Vergleich, wenn noch kein Defekt eingefügt wurde. Es zeigen sich dabei zwei konträre Schlussfolgerungen der verschiedenen Ansätze. Zum einen wird die Verzögerungszeit vor allem von den redundanten Transistorstacks negativ beeinflusst, da die Schalttransistoren

zwar die Lastkapazität erhöhen, allerdings keine zusätzlichen Treiber vorhanden sind, da die redundanten Stacks noch nicht zugeschaltet sind. Dies ist allerdings bei DTrans der Fall, da die redundanten Transistoren von Anfang an aktiv sind und somit die zusätzliche Lastkapazität kompensieren. Die Erhöhung der Delays bei DGatter im Gegensatz zu DTrans resultiert wahrscheinlich aus der veränderten Balance der Pfade, die bei DTrans nur punktuell erfolgt und somit auch die Schaltungen sogar beschleunigen kann. Bei den größeren Schaltungen c1908 und c3540 zeigt sich die geringere Zusatzfläche der Enh5-Strategie, die zu einer besseren Verzögerungszeit führt. Beim cla16-Addierer ist wiederum die DTrans-Implementierung am langsamsten, während alle anderen ungefähr die gleiche Verzögerungszeit haben. Dies ist die Folge aus einem sehr schnellen und sehr balancierten Design, was durch punktuelle Veränderungen wie beim DTrans stärker beeinflusst wird. Im Gegensatz zur Verzögerungszeit wird aufgrund der noch inaktiven redundanten Stacks und der kleinen Schalttransistoren die Leistungsaufnahme der EnhX-Strategien nur geringfügig um maximal 8 % vergrößert, während die Leistungsaufnahme bei DTrans und DGatter auf bis zu 143 % bzw. 153 % des originalen c3540-Designs ansteigen kann.

In der Folge wurden nun Defekte in die Designs eingeführt. Neben dem üblichen Einführungsalgorithmus, der die jeweilige Schaltung immer mehr Defekten mit ansteigender Defektrate aussetzt, um die Verschleißphase zu simulieren, wurden weitere Aspekte berücksichtigt. Neben der individuellen Verteilung der Defekteintrittszeitpunkte t_{GOB} für die Transistoren der Originalschaltung wurde für eventuelle redundante Transistoren genauso verfahren. Individuell bedeutet, dass der Defektzeitpunkt und -verlauf ist für jeden Transistor gesondert berechnet wurde. Die Übernahme der Defektinformationen für jeden gleichen Transistor sichert einen gerechten Vergleich, da nun sowohl die originalen als auch die

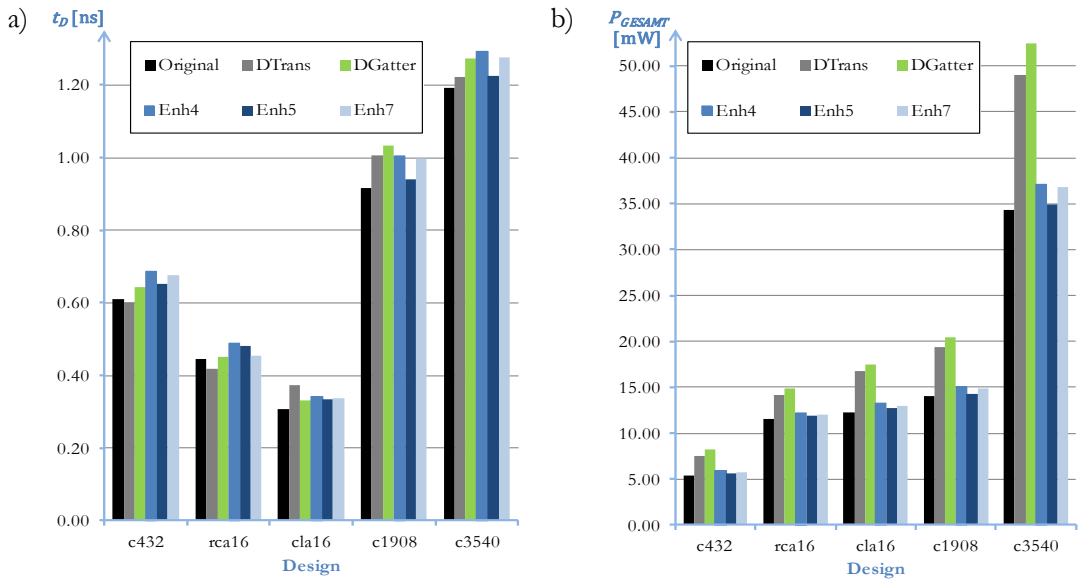


Abbildung 4-20 : Absolute Werte der Verzögerungszeit a) und der Leistungsaufnahme b) der verschiedenen Implementierungen im defektfreien Zustand der Schaltungen

redundanten Transistoren unterschiedlicher Implementierungen für einen Durchlauf gleiche Ausfallzeitpunkte aufweisen, sofern in einem Design der jeweilige Transistor in mehreren Implementierungen auftaucht. Bei den Originaltransistoren und den redundanten Transistoren der Strategien DTrans und DGatter beginnt die Zeit des Verschleißes mit $t = 0$, während die Transistoren der redundanten Stacks erst mit Einschalten der redundanten Anteile des Designs zu verschleißen beginnen. Dies ist zwar in dem Sinne eine Benachteiligung gegenüber den Implementierungen mit redundanten Transistorstacks, dennoch wurde dieser Ansatz gewählt, um auch für die redundanten Stack einen Alterungseffekt hinzuzufügen. So sollte diese pessimistische Herangehensweise für die nachfolgend präsentierten Ergebnisse zur Zuverlässigkeit bedacht werden. Die redundanten Stacks wurden erst zugeschaltet, wenn zu einer bestimmten Zeit t_{23} eine Grenze der Schaltungsverzögerungszeit $t_{D\_23} = 5/3 \cdot t_{D\_ORIGINAL}(t = 0)$ übertroffen wurde, wobei $t_{D\_ORIGINAL}$ die Verzögerungszeit des Originaldesigns darstellt. Dies bedeutet also in diesen Simulationen, das die redundanten Stacks zugeschaltet wurden, wenn die Verzögerungszeit das Originaldelay um zwei Drittel aufgrund der verlangsamenden Defekte übertroffen hat. Ab diesem Zeitpunkt t_{23} begann auch der Verschleiß der redundanten Stacks, so dass ein Defekt in den Transistoren nach der Zeit $t = t_{23} + t_{GOB}$ eingefügt wurde und sich dann mehr und mehr Richtung hartem Durchbruch entwickelte. Als Fehler wurden dann folgende Kriterien festgelegt. Ein einziger logischer Fehler bei Eingabe aller Inputvektoren hat zur Folge, dass das jeweilige Design als fehlerhaft deklariert wird. Im Falle des Originaldesigns und der beiden Implementierungen DTrans und DGatter führt das zum Abbruch des Durchlaufes. Sollten bei den anderen Implementierungen die redundanten Stacks noch nicht angeschaltet sein, weil t_{D_23} noch nicht erreicht wurde, so wurden die redundanten Anteile angeschaltet und der Durchlauf zum Fehlerzeitpunkt wiederholt. Da auch ein zeitkritischer Aspekt berücksichtigt werden sollte, wurden die Designs auch als fehlerhaft klassifiziert, wenn sie die Verzögerungszeit $t_{D\_LIMIT} = 2 \cdot t_{D\_ORIGINAL}(t = 0)$ übertroffen haben. Dies bedeutet eine Reaktion auf einen parametrischen Fehler, falls die kombinatorische Schaltung doppelt so langsam geworden ist.

Wie erwartet, stellt sich wieder ein „graceful degradation"-Verhalten aller Schaltungen ein, wie es in den Abbildungen 4-21 a) und b) dargestellt ist. In Abbildung 4-21 a) ist exemplarisch die Entwicklung der durchschnittlichen Verzögerungszeit für die verschiedenen noch funktionierenden Implementierungen des Designs c3540 zu sehen. Wie auch bei den anderen Schaltungen steigt die Kurve der Originalschaltungen am schnellsten an, da keine redundanten Anteile in der Schaltung vorhanden sind, um den Verlust der Treiberfähigkeit der defekten Transistoren etwas entgegenzusetzen. Die Schaltungen DTrans verzeichnen den geringsten Anstieg. Dies liegt daran, dass dies der präziseste Redundanzalgorithmus ist, so dass die gefährdetsten Stellen im Design, die das Delay belasten, am ehesten getroffen werden. Die Strategie DGatter senkt im Mittelteil der Kurve die Verzögerungszeit unter die des Originals.

Das schlechtere Abschneiden dieses Algorithmus gegenüber DTrans liegt in der Redundanzstrategie begründet. Die leichtere Handhabung aufgrund von redundanten Gattern anstatt Transistoren bringt Nachteile bei der Auswahl der Transistoren mit sich. Wenn nur gezielt die gefährdetsten Transistoren ausgesucht werden, wird auch die andere Hälfte des Gatters verdoppelt, die allerdings nicht so anfällig für Gateoxiddefekte ist, da deren

Schaltwahrscheinlichkeit geringer, weil entgegengesetzt zur gefährdeten Hälfte ist. Bei der DTrans-Strategie wird zumeist nur eine Hälfte eines Gatters redundant ausgelegt, also der Transistor(stack), der häufiger schaltet. Bei den Untersuchungen zur vollumfänglichen Redundanz aus Kapitel 4.2 fiel dies nicht ins Gewicht, da dort das komplette Design verdoppelt wurde. Hier zeigen sich nun die Vorteile der gezielten Auswahl bei DTrans. Bei den Designs mit redundanten Stacks steigt zuerst die Kurve ähnlich stark an wie beim Originaldesign. Allerdings werden mit fortschreitender Zeit immer mehr Designs ihre redundanten Transistoren anschalten, so dass die Verzögerungszeit nicht mehr so stark ansteigt und die Designs sich sich so dem vorzeitigen Ende wie bei der Originalschaltung entziehen.

In der Abbildung 4-21 b) ist die durchschnittliche Leistungsaufnahme dargestellt. Aufgrund ihrer redundanten Anteile, die nicht an- bzw. abgeschaltet werden können, bleibt die Leistungsaufnahme der Strategien DTrans und DGatter bei den größeren Designs immer um mindestens 50 % bis über 100 % höher als bei Enh4. Dies widerspricht auf den ersten Blick der Tatsache, dass die statischen Ströme mit der Zeit die Stromaufnahme dominieren. Allerdings steigt bei DTrans und DGatter auch der Anteil der defekten redundanten Transistoren, da diese auch von Anfang an altern, so dass die höhere Leistungsaufnahme gegenüber den EnhX-Implementierungen erhalten bleibt. Die geringere Stromaufnahme von DGatter gegenüber DTrans ist damit zu erklären, dass DTrans auch noch mit mehr Defekten als DGatter funktioniert und somit auch eine erhöhte Stromaufnahme in die Durchschnittsberechnung einfließen lässt.

Aus der unterschiedlichen Länge der Kurven ist schon ersichtlich, dass die EnhX-Strategien entscheidende Verbesserungen hinsichtlich der Zuverlässigkeit zulassen. Dies ist exemplarisch für das c432-Design aus der Zuverlässigkeitskurve *R(t)* in Abbildung 4-22 zu sehen. Bei diesem kleinsten Design erreichen zum Zeitpunkt $t = 1.2$ noch rund 58 % aller Originaldesigns korrekte Ergebnisse und eine ausreichende Verzögerungszeit, während bei den anderen

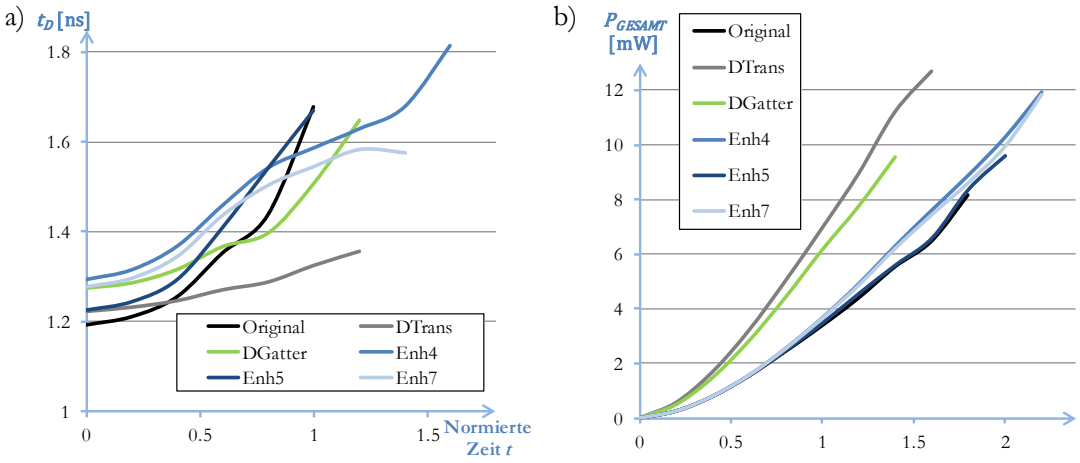


Abbildung 4-21 : Entwicklung der Schaltungsparameter mit ansteigender Defektrate
a) Durchschnittliche Verzögerungszeit aller Szenarien des Designs c3540
b) Durchschnittliche Leistungsaufnahme aller Szenarien des Designs c1908

Implementierungen noch über vier Fünftel funktionieren (Enh5 – 77 %). Zum Zeitpunkt $t = 2$ sind dann keine der nicht redundanten Schaltungen aktiv, während bei den Enh4-Designs noch ein Drittel keine Fehler erzeugen und ein Viertel aller Enh7-Designs, also die Strategien mit redundanten Stacks, deren Redundanzauswahl sich nur auf die Schaltwahrscheinlichkeit bezieht. Die anderen Implementierungen befinden sich zwischen diesen beiden und dem Enh5-Design, dessen Redundanzanteil am geringsten bei allen Implementierungen ausfällt. Diese Reihenfolge spiegelt gut die der anderen Testdesigns wieder, bis auf eine Ausnahme. Dieses Design ist das einzige, bei dem DGatter bessere Ergebnisse erzielt als DTrans. Dies liegt daran, dass die bessere Auswahl durch DTrans bei der kleinsten Schaltung noch nicht zum Tragen kommt.

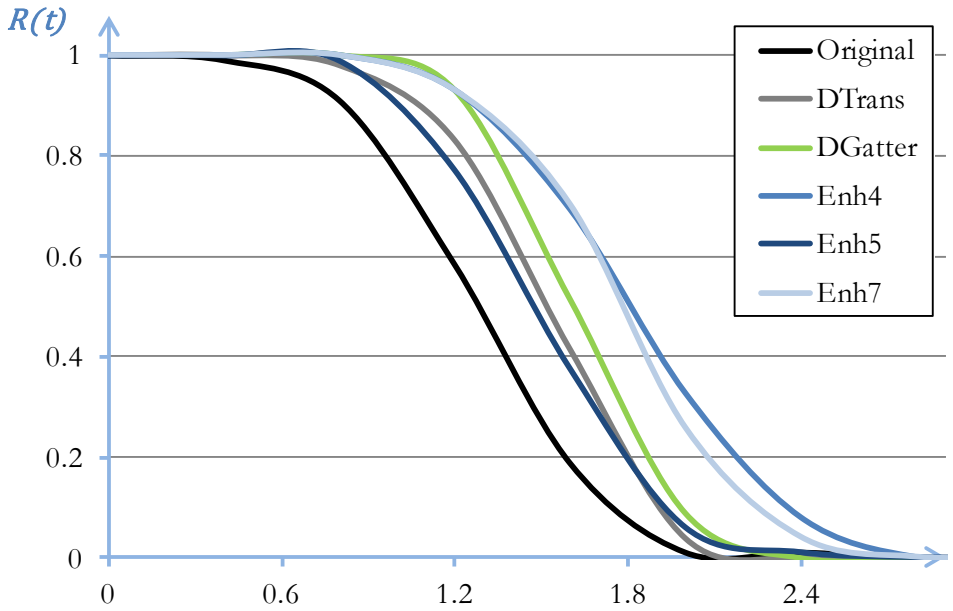


Abbildung 4-22 : Zuverlässigkeit aller Implementierungen des c432-Designs

Um einen Vergleich zwischen allen Designs zu ermöglichen wurden die *MTTF*-Werte für alle Implementierungen aller Designs in den Abbildungen 4-23 dargestellt. Generell ist zu sehen, dass die Enh5-Implementierung nur bei den langsameren Designs (rca16, c1908) brauchbare Verbesserungen erzielt, da die redundanten Stacks in den schnelleren Pfaden dazu beitragen, das System unter t_{D_LIMIT} zu halten, während bei den schnelleren Designs auch langsamere Pfade schnell genug altern, um die Grenze zu erreichen, was die redundanten Stacks der Enh5-Strategie nicht aufzuhalten vermögen. Die Verbesserungen durch die nicht schaltbaren redundanten Transistoren von DTrans und DGatter bleiben relativ konstant, wobei zu konstatieren ist, dass bei größeren Schaltungen DTrans immer besser abschneidet, und die größeren Designs noch mehr von der Redundanz profitieren. Hier könnte auch der Einsatzort für DGatter liegen, da in den größeren Designs das aus der Balance gebrachte Timing durch ausreichend eingeführte zusätzliche Transistoren über die Verschleißphase hinweg konstant gehalten werden kann. Bei den EnhX-Implementierungen ist zu erkennen, dass vor allem bei größeren Designs große

Verbesserungen zu erwarten sind, mit Ausnahme der Enh5-Strategie. Um auch einen Anhaltspunkt über den Umschaltzeitpunkt, der für diese Simulationen bei t_{D_23} gewählt wurde, zu bekommen, ist eine weitere *MTTF*-Darstellung in Abbildung 4-23 b) eingefügt. Hier ist die *MTTF* aller Implementierungen aller Designs zu sehen, wenn das Limit bei $t_{D\_LIMIT} = t_{D\_23}$ gesetzt wird. Dadurch würden die Verbesserungen der EnhX-Strategie an Kraft verlieren, denn der Vorteil der späteren Alterung zum Vorteil der Schaltung würde so spät kommen, dass nun nicht mehr viel Spielraum für weitere Verschlechterungen der Verzögerungszeit vorhanden wäre. Der Schluss daraus ist, dass das Einschalten der redundanten Transistorstacks zu einem früheren Zeitpunkt erfolgen sollte, als beim Auftreten erster gravierender Timingfehler. So bietet sich neben der Detektion von Timingfehlern oder der Überwachung des Timings ein vorher festgelegter Zeitpunkt an, bei dem ein Zuschalten erfolgen soll, um den eventuell schon eingetretenen Verschleiß zu mindern.

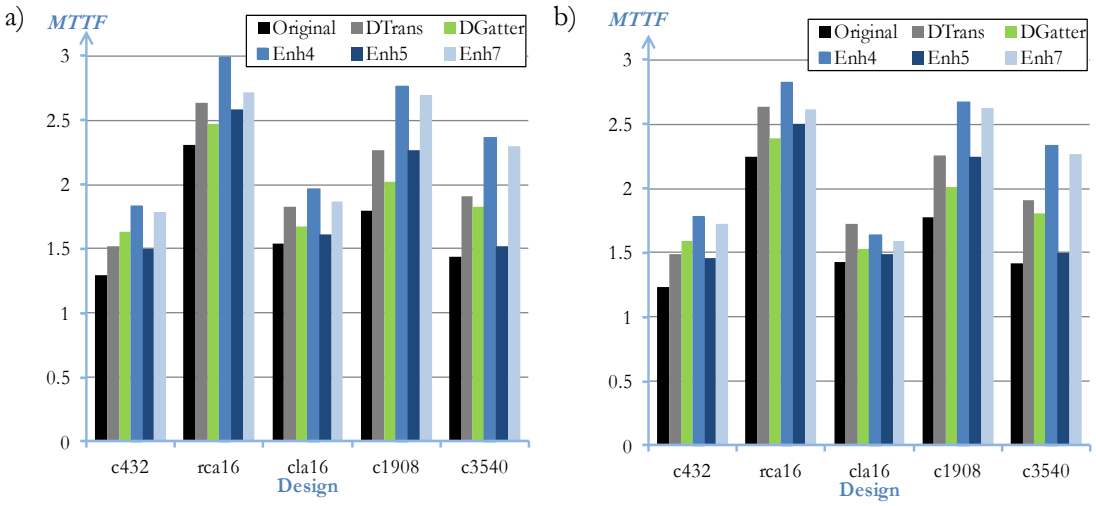


Abbildung 4-23 : MTTF-Werte aller Implementierungen aller Designs
a) Bei einem Verzögerungszeitlimit $t_{D\_LIMIT} = 2 \cdot t_{D\_ORIGINAL}(t=0)$
b) Bei einem Verzögerungszeitlimit $t_{D\_LIMIT} = 5/3 \cdot t_{D\_ORIGINAL}(t=0) = t_{D\_23}$

4.3.7 Fazit: Redundante Transistorstacks

Als abschließende vergleichende Darstellung wurde die auf das Originaldesign normierte *MTTF* aller Implementierungen aller Designs und für beide Fälle der Delaylimitierung in Abbildung 4-24 ausgesucht. Für das Einfügen von Redundanz auf Transistorebene (DTrans) werden zwischen 14 % und 19 % für die kleineren Designs und 26 % bzw. 33 % *MTTF*-Steigerung bei c1908 und c3540 erreicht. Die Gatterverdopplung DGatter erreicht nur beim größten und kleinsten Design passable 26 %, sonst zwischen 7 % und 12 %. Interessanterweise erhöht ein geringeres Verzögerungszeitlimit t_{D_LIMIT} zumeist die normierte *MTTF*, während bei den EnhX-Implementierungen diese in den meisten Fällen gesenkt wird, da der Umschaltzeitpunkt zu spät für das Verlangsamen der Verschleißerscheinungen kommt, während bei DTrans und DGatter die Verschleißeffekte von Anfang an gemindert werden. Ferner ist zu

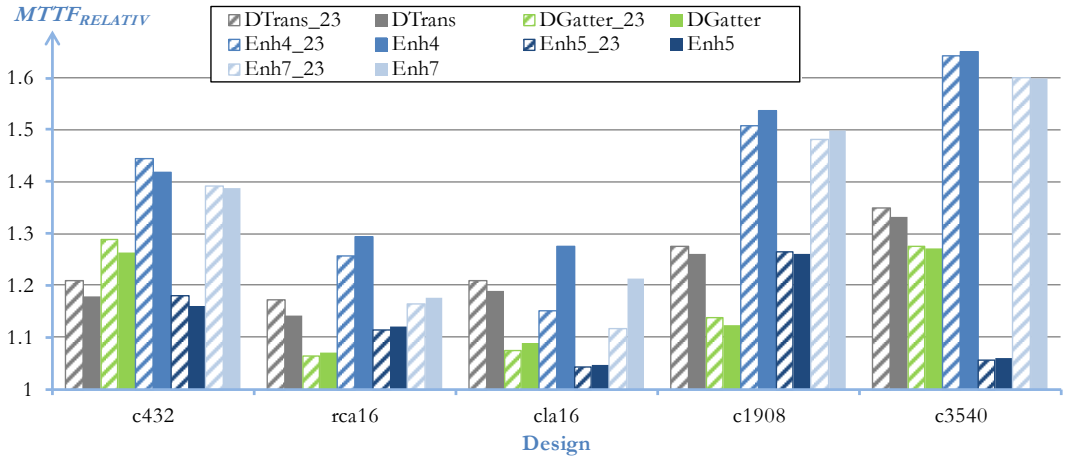


Abbildung 4-24 : Relative MTTF-Steigerungen aller zuverlässigkeitssteigernden Implementierungen normiert auf das jeweilige Originaldesign

sehen, dass die redundanten Transistorstacks ohne Delayberücksichtigung Enh4 und Enh7 vor allem in den großen Designs (c1908, c3540) zur Geltung kommen. Hier sind *MTTF*-Steigerungen von ca. 50 % bis zu 65 % möglich. In den kleineren Designs sinkt dann die MTTF-Steigerung von Enh7 auf 17% bis 38 %, während die Steigerung bei Enh4 immer über 27 % bleibt. Enh5 kann nur bei langsamen Designs (rca16, c1908, c432) sinnvoll eingesetzt werden, bei denen Steigerungen von über 10 % erreicht werden.

Der Vorteil der verspäteten Alterung der redundanten Transistoren der EnhX-Implementierungen erzwingt Steigerungen von über 50 % bei größeren Designs, wobei nochmal zu bemerken ist, dass eine stärkere Alterung simuliert wurde als es wahrscheinlich der Fall ist, da die langsame Verschlechterung der Transistorgates in Phase 2 übersprungen wird und die verstärkte Alterung sofort mit Inbetriebnahme des redundanten Stacks beginnt. So ist sogar ein größeres Verbesserungspotential vorhanden. Ferner kann über den Umschaltpunkt in Hinblick auf Verzögerungszeitrestriktionen aufgrund von taktsynchronen Schaltungen die Zuverlässigkeit positiv beeinflusst werden. Wichtig ist hierbei, dass die Umschaltung nicht zu spät erfolgt.

Ein weiterer Vorteil ist die geringere Steigerung der Leistungsaufnahme, die zwischen +3 % und +8 % während des normalen Betriebs liegt, wenn die redundanten Transistorstacks noch nicht dazugeschaltet worden sind. Dies rückt den Verschleiß während dieser Phase auf das Niveau des Originaldesigns, wohingegen eine erhöhte Stromaufnahme der DTrans- und DGatter-Strategien, die bei durchschnittlich +40 % liegt, zu einer erhöhten lokalen Temperatur führt und somit zu einer schnelleren Alterung.

Für eine generelle Betrachtung der Temperaturabhängigkeit der Zuverlässigkeit bemüht man oft die Arrheniusgleichung [JED11], die auf der van't Hoffschen Regel basiert. Diese besagt, dass eine Temperaturerhöhung um 10 K die Geschwindigkeit chemischer Reaktionen verdoppelt. Damit ergibt grob gerechnet eine Halbierung der Lebensdauer, wenn die Temperatur dauerhaft um 10 K erhöht wird. Ergebnisse experimenteller Untersuchungen für Gateoxiddefekte [Sue04]

lassen sich allerdings nicht mit der Arrheniusgleichung erklären. Allerdings gilt für die beobachtete exponentielle Temperaturabhängigkeit, dass eine Erhöhung der Temperatur um 100 K die Defektrate DR verzehnfacht. Im Endeffekt führt jede Verringerung der Leistungsaufnahme zu einer Zuverlässigkeitsverbesserung, sollte sie zu einer Temperaturverringerung führen, z. B. aufgrund unveränderter Kühlmechanismen.

So bietet die gezielte Einschleusung von redundanten Transistorstacks an gefährdeten Stellen im Design eine kostengünstige Möglichkeit, die Zuverlässigkeit zu steigern. Kostengünstig in dem Sinn, dass die Zusatzfläche über den Einführungsalgorithmus gesteuert wird. Weiterhin wird die Leistungsaufnahme nur geringfügig beeinflusst und damit auch die übliche Alterung am Anfang der Betriebszeit. Zudem kann auf Produktionsdefekte reagiert werden. Sowohl die Auswirkungen funktionaler als auch parametrischer Defekte können dann direkt nach der Herstellung beseitigt bzw. gemindert werden, wenn man sofort die redundanten Teile dazuschaltet.

Kostengünstig aber auch in dem Sinn, dass das Einfügen auf Gatterebene erfolgen kann, und damit problemlos in den üblichen Designprozess integriert wird, was in Abbildung 4-25 dargestellt ist. Hierbei folgt auf die Aktivitätsanalyse der Algorithmus zum Einfügen redundanter Transistorstacks, dass die Gatternetzliste modifiziert und somit neue Tests und Analysen notwendig macht. Allerdings ist dabei nur ein neuer Designdurchlauf notwendig.

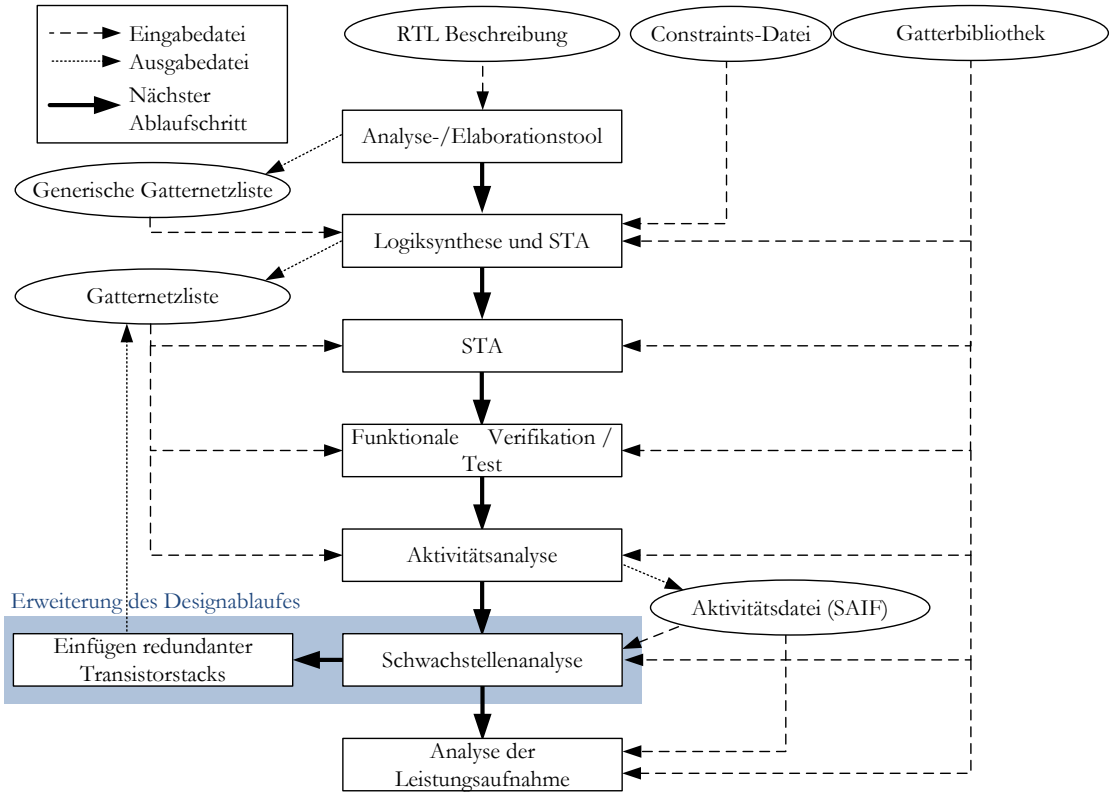


Abbildung 4-25 : Erweiterter CMOS-Designflow zur Einführung redundanter Transistorstacks

Fünftes Kapitel

5 Analyse von Gateoxiddefekten auf Gatterebene

Aus dem vorigen Kapitel ist ersichtlich geworden, dass partielle Redundanz eine einfache und kostengünstige Methode darstellt, zuverlässigere Schaltungen auf Gatterebene zu generieren. Aus den Untersuchungen ist allerdings auch der Schluss gezogen worden, dass das gezielte Einfügen der redundanten Elemente zumindest hinsichtlich der Verzögerung von Gateoxiddefekteffekten dem zufälligen Einfügen vorzuziehen ist. Im einfachsten Fall kann das über die Aktivitätsdatei geschehen, allerdings wird hier nur der Faktor V_{GS} mit Hilfe von P_{ON} der in Abschnitt 3.6.1 dargestellten Parametern, die sich auf ttf_{GOB} auswirken können, berücksichtigt, während t_{OX}, T und A_{GATE} unberücksichtigt bleiben. Aus diesem Grund wird in dem folgenden Kapitel der neu entwickelte GOB-Simulator vorgestellt, welcher der Defektanalyse kombinatorischer integrierter Schaltungen dient [Sae12a]. Damit ist es möglich, die Auswirkungen von Gateoxiddefekten auf kombinatorische Schaltungen aufzuzeigen. Hierbei werden Gateoxiddefekte über einen definierten Zeitraum in die Schaltung eingefügt und das dadurch modifizierte Verhalten der Schaltung analysiert. So ist es möglich, die Parameter und deren Veränderungen der sich „verschleißenden“ Schaltung über diesen Zeitraum nachzuvollziehen. Wenn die Analysen in einer Art Monte-Carlo-Verfahren in mehreren Durchläufen durchgeführt werden, ist es möglich, die Schwachstellen in einem Design hinsichtlich TDDB zu erkennen, um dann redundante Elemente gezielt einzufügen. Ein Vorteil ist dabei die Einordnung des GOB-Simulators auf Gatterebene, um ihn problemlos in den Designablauf zu integrieren.

Bei der Analyse von Schaltungsparametern werden auf Gatterebene Standardgatter-bibliotheken verwendet, um für jedes einzelne Gatter korrekte Daten für deren aktuelle Position im gewählten Design auszugeben, wie es in Kapitel 2.3 erläutert wurde. Auf Basis dieser Daten ist es möglich, korrekte Untersuchungen bzw. Simulationen durchzuführen. Der GOB-Simulator nutzt eine weitere Datenbasis. Diese Datenbank enthält Werte für die üblichen Gatterparameter,

allerdings auch andere wichtige Parameter, um auch die Effekte von Gateoxiddefekten nachzubilden. Daraus wird dann für ein Gatter, deren Transistoren ein Gateoxiddefekt erleiden, ein modifizierter Bibliothekseintrag erstellt, um damit die Standarduntersuchungen hinsichtlich Verzögerungszeit und Leistungsaufnahme durchführen zu können. Da die Gateoxiddefekte die Stromkurven der Transistoren verändern, war es naheliegend das CCS-Modell, vorgestellt in Kapitel 2, für die Bibliothek zu verwenden, da die Stromquellen des Treibermodells direkt manipuliert werden können, um einen Defekt nachzubilden. Damit sind weitere Untersuchungen möglich, um die Spannungspegel der Netze in Anwesenheit von Gateoxiddefekten zu analysieren.

So umfasst ein Durchlauf des GOB-Simulator zwei wesentliche Teilschritte: die Platzierung der Defekte in die Schaltungsnetzliste und die Analyse ihrer Auswirkungen auf das Design. Unterkapitel 5.2 erörtert diese Defektplatzierung. Dabei sind je nach gewünschtem Simulationszeitpunkt des Schaltungszustandes die Gatter zu lokalisieren, deren Eintrag modifiziert werden muss, damit deren Gateoxiddefekt Berücksichtigung bei weiteren Analysen findet. Darauf folgt die Erläuterung der Designanalyse, welche sich in zwei grundlegende Schritte unterteilt. Zum einem werden die Spannungspegel der einzelnen Netze des Designs auf das Über- bzw. Unterschreiten bestimmter Grenzen untersucht. Zum anderen werden die Gatterbibliotheken derart den Gegebenheiten angepasst, dass STA- und Leistungsanalysetools den Verschleiß des Designs modellhaft nachbilden können. Für die realistische Modifizierung dieser Parameter und die Spannungspegelvoraussage wurden zahlreiche Spice-Simulationen durchgeführt, die ebenfalls im Verlauf der Erläuterung des Modifizierungsalgorithmus näher betrachtet werden. Die Einordnung in den Standard Design Flow, sowie Ergebnisvergleiche des GOB-Simulators mit reinen Spice-Simulationen schließen das Kapitel ab.

5.1 Einordnung und Vergleich mit anderen Zuverlässigkeitssimulatoren

Erste Simulatoren, die Verschleißmechanismen adressieren, wurden erstmals Mitte der 1990er vorgestellt im Rahmen der Berkeley Reliability Tools (BERT) [Tu93]. Diese Zusammenfassung verschiedener Simulatoren nutzt zwei Module, die sich mit dem Verschleiß des Gateoxides befassen. Zum einen simuliert der Circuit Aging Simulator (CAS) den Alterungsprozess aufgrund von Hot Carriers Effekten. Hierbei werden die Drain-und Substratströme für jeden Transistor berechnet und in Relation zueinander gesetzt. Zusammen mit verschiedenen Datensätzen aus unterschiedlichen Stresstests von realen Schaltungen wird dann ein Alterungsparameter für die gesamte Schaltung berechnet und somit die Verschlechterung der Verzögerungszeit des Designs. Das andere Modul ist der Circuit Oxide Reliability Simulator (CORS). Dieser nutzt die Spannungspegel aller Transistoren aus Spice-Simulationen und vom Nutzer bereitgestellte Statistiken, um die Wahrscheinlichkeit von Fehlern aufgrund von Gateoxiddefekten vorherzusagen. Hierbei können die beiden vorgestellten und bis dahin gültigen Modelle für Gateoxidbreakdown, das E- bzw. das 1/E-Modell, für die Fehlerverteilungsberechnung

ausgesucht werden. Des Weiteren nutzt dieser Simulator aufgrund der noch nicht vorhandenen Studien über weiche Durchbrüche nur das Modell der harten Gateoxiddefekte.

Der in [Sal08] vorgestellte Ansatz generiert gatterspezifische Parameter, wie die Verzögerungszeit, mittels Spice-Simulationen. Hierbei werden drei verschiedene Zustände für die Transistoren während der Simulationen aufgrund der z. B. aus Aktivitätsdateien erlangten Eingangsvektoren für die Transistoren ausgewertet: kompletter Stress, moderater Stress und Erholungsphase. Daraus wird dann auf Gatterebene die Vergrößerung der Schwellspannung berechnet, was zusammen mit probabilistischen Fehlerverteilungen Analysen zur Verzögerungszeit des Designs ermöglicht. Der Hot-Carrier-Simulator GLACIER [Wu00] generiert auch Daten über die Veränderung der Verzögerungszeit der gesamten Schaltung aufgrund von HC Effekten. Dafür wird im Voraus die Standardzellenbibliothek für jedes Gatter modifiziert, indem diese um eine Dimension für das Schaltverhalten des NLDM erweitert, so dass die verschleißbedingten Effekte durch einen Parameter Schalthäufigkeit in der Bibliothek verfügbar sind. Grundlage dieser Modifizierungen sind Spice-Simulationen. Ein Zeitanalysetool generiert dann Dateien für das Zeitverhalten der Gatter, die dann für eine STA auf Gatterebene genutzt werden. Dieser Ansatz ist vergleichbar mit einem probabilistischen Simulator, der die Performance aufgrund von Parameterschwankungen nach der Produktion modelliert [Li92].

In modernen Analysetools existieren mehrere Ansätze. Zum einen werden auf Transistorebene Verschleißeffekte mittels Parameter modifiziert, womit der Stromfluss im Bauteil verändert wird [Cad03]. Zum anderen wird auf der Gatterebene agiert, indem man die Gatterverzögerungszeit pauschal um definierte Prozentpunkte vergrößert [Syn11]. Grundlage bei beiden sind technologiebedingte Parameter. Eine analytische Methode zur Vorhersage von fehlerhaften, größeren Designs wurde in [Jia10] vorgestellt. Hierbei wurden harte Durchbrüche als ursächlich für mögliche Störungen der Schaltung herausgestellt. Umstände, die zu kritischen Spannungsdifferenzen (V_{HIGH} - V_{LOW} < Grenzwert) einzelner Netze führen und somit zu Funktionsfehlern, wurden mittels modifizierter Weibull-Wahrscheinlichkeitsverteilungen der Transistoren dargestellt. Daraus entwickelten die Autoren eine Vorhersage für Fehler des gesamten Designs, sowie Dimensionierungsregeln einzelner Gatter, um funktionalen Fehlern entgegen zu wirken.

Intention des hier vorgestellten Simulators ist es, Verschleißerscheinungen aufgrund von Gateoxiddefekten auf Gatterebene zu simulieren. Der GOB-Simulator unterscheidet sich von vorhergehenden Arbeiten in der Art, dass Gattern nicht pauschal, sondern individuell nach definierten Kriterien eine Fehlerwahrscheinlichkeit bzw. eine individuelle ttf_{GOB} zugeordnet wird. Damit ist es möglich, den probabilistischen Charakter der Defektgenerierung zu bewahren und bei gleichen Startbedingungen nicht gleiche Ergebnisse für mehrere Durchläufe zu erlangen. Reproduzierbar werden die Ergebnisse dann durch Zusammenfassung mehrerer Durchläufe, wodurch Schwachpunkte innerhalb eines Designs aufgezeigt werden. Diese Art der Analyse ähnelt dem Monte Carlo Verfahren. Des Weiteren wird neben der funktionellen Analyse auch die Veränderung der Verzögerungszeit gatterweise ermöglicht, um den Einfluss des progressiven Breakdowns über die Zeit untersuchen zu können, wie es bei anderen Defektarten schon aufgezeigt wurde. Allerdings wurde hier das CCS-Modell verwendet, da die Manipulation der

Stromkurven schon durch die Defektart vorgegeben wird und so leicht in das Modell übertragen werden kann. Um dies zu erreichen, wurde der schon vorgestellte Dualismus aus Transistormodellen und Gatteranalyse einiger Simulatoren übernommen, so dass Transistormodelle von Defekten mit einem Spice-Simulator analysiert wurden, die als Datenbasis für den GOB-Simulator dienen. Dieser Simulator operiert dann auf Gatterebene und erstellt Spannungspegelreports über die Netze eines vorgegebenen Designs. Außerdem wird eine Gatterbibliothek generiert, damit das Zeit- und Stromaufnahmeverhalten auch mit Standard-Designtools analysiert werden kann.

5.2 Defektplatzierung

Um ein realistisches Fehlermodell zu erhalten, werden Defekte im GOB-Simulator auf Transistorebene platziert. Dies bedeutet, dass die Defektwahrscheinlichkeit der Transistoren der einzelnen Gatter ausschlaggebend für das Einfügen modifizierter Gatterbeschreibungen ist. Die Ausbildung eines Transistordefektes erfolgt anhand einer vorgegebenen Wahrscheinlichkeits-verteilung. Implementiert wurden vier grundlegende Verteilungen: Gleich-, Normal-, Exponential- und Weibullverteilung. Da die Normal- und die Exponentialverteilung auch mit Hilfe der Weibullverteilung dargestellt werden können und weil diese, wie in Kapitel 3.1 erläutert, für die Darstellung von Verschleißerscheinungen sehr verbreitet und zutreffend ist, wurde die Weibullverteilung als Referenzverteilung eingestellt und für alle hier vorgestellten Simulationen mit zufälliger Defekteinfügung genutzt. Dabei ist auch der Weibullfaktor β einstellbar, wobei $\beta = 1$ einer Exponentialverteilung entspricht, welche üblich für Phase 2 der Badewannenkurve ist.

Gleichung (34) gibt die Weibullgrundgleichung für die Verteilung der Lebensdauer ttf_{GOB} jedes Transistors vor. Dieser Wert wird mit Hilfe der Gleichungen (58) bis (61) und (65) individuell für jeden Transistor berechnet und ist relativ zu einem Referenztransistor zu betrachten. Ausgangspunkt ist jeweils ein Transistor, der die größte Fläche aller genutzten Transistoren einnimmt, sowie permanent der Spannung V_{GS} ausgesetzt wurde, die am schnellsten zu Gateoxiddefekten führt, $V_{GS} = V_{DD}$ für n-MOSFET, $V_{GS} = V_{SS}$ für p-MOSFET. Diesem Transistor wird die minimale Lebensdauer $ttf_{MIN} = 1$ zugewiesen. Dieser Wert stellt den Minimalwert für ttf_{GOB} jedes Transistors dar. Davon ausgehend werden die aktuellen Werte des Transistors, wie Gatespannung, Transistorfläche, -dicke und Temperatur, in Relation zu den erreichbaren Maximalwerten gesetzt und $ttf_{TRANS} = ttf_{GOB}$ für jeden Transistor individuell berechnet. Aus diesem Grund ist die Zeitskala relativ und dimensionslos, wobei Defekte ab dem Zeitpunkt $t > 0$ auftreten können und in Abhängigkeit der modifizierten Lebensdauer ttf_{TRANS} mit jedem Zeitschritt wahrscheinlicher werden. In Verbindung mit Gleichung (34) ergibt sich daher individuell für jeden Transistor folgende kumulative Weibullverteilung:

$$F_{TRANS}(t) = 1 - e^{-\left(\frac{t}{ttf_{TRANS}}\right)^{\beta}} \tag{67}$$

Um einen Defekt mittels des GOB-Simulators korrekt einzufügen, wird der konkrete Zeitpunkt für den jeweiligen Simulationslauf berechnet. Dafür wird $F_{TRANS}(t)$ ein gleichverteilter Zufallswert *rand* zwischen 0 und 1 zugeordnet. Aus Gleichung (67) ist es dann möglich einen Zeitpunkt $t = t_{GOB}$ für jeden Transistor zuzuordnen, bei dem dieser einen Defekt für den einzelnen Analyselauf ausbildet. Abbildung 5-1 zeigt dies an einem Beispiel. Danach verändert sich der Widerstand R_{GOB} über die restliche Zeit nach Gleichung (53). Als Ausgangspunkt dient hierbei ein zufälliger Wert für den Widerstand, der einen soft-Breakdown darstellt ($R_{GOB}(t_{GOB}) > 1$ MΩ), um das progressive Modell des Gateoxiddefektes abzubilden.

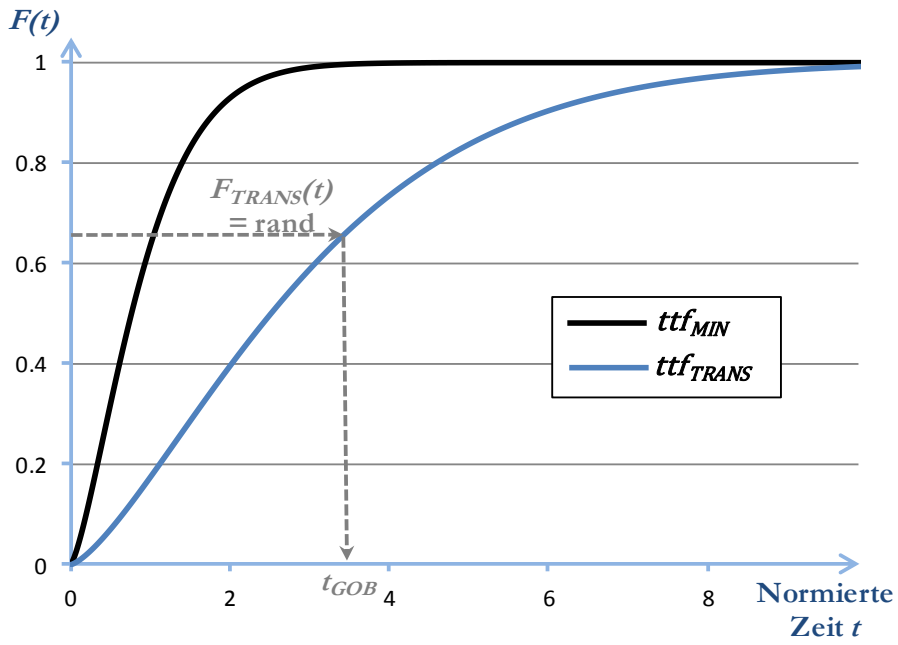


Abbildung 5-1 : Beispiel für die Zuweisung eines Defekteintrittszeitpunktes t_{GOB}: Ausgehend von der Verteilungskurve F(t) für den Referenztransistor mit $ttf_{MIN} = 1$, wird die Kurve für den jeweiligen Transistor berechnet (in diesem Beispiel: $P_{ON} = 0.5$ und $A_{GATE} = ½ \cdot A_{GATE\_MAX}$ mit Gleichungen (61) und (65) → $ttf_{TRANS} \approx 3.28$). Dann wird ein Zufallswert für $F_{TRANS}(t) = rand$ zugewiesen, woraus ein Defekteintrittszeitpunkt t_{GOB} ermittelt werden kann.

5.3 Spice-Datenbasis: Grundlagen der Defektanalyse

Wenn ein Transistor einen Defekt ausgebildet hat, wird das Verhalten von mit ihm verbundenen Gattern, inklusive des eigenen, variabel beeinflusst. Dabei kann es auch zu Interaktionen mehrerer Breakdowns kommen, die ebenfalls modelliert werden. Daten aus grundlegenden Spice-Simulationen auf Basis des Defektmodells von Renovell werden dazu verwendet, Parameter der Gatter so anzupassen, dass ein möglichst realistisches Verhalten mit Defekt innerhalb der Schaltung dargestellt wird. Danach ist es möglich, die Spannungspegel der

Netze zu den untersuchten Zeitpunkten zu errechnen, als auch Stromkurven und Stromaufnahmeparameter so anzupassen, dass mit der modifizierten Gatterbibliothek, Veränderungen des statischen Zeitverhaltens und der Leistungsaufnahme mit proprietären Analysetools darstellbar sind. Diese Datenbank wird auch für die spätere Spannungspegelanalyse verwendet, die sich zwar auf andere Zielparameter fokussiert, deren Berechnungsschritte allerdings von den gleichen Basisdaten abhängig sind.

Da Spice-Simulationen als Grundlage dienen, war ein Defektmodell auf Transistorebene notwendig. Die Wahl fiel auf das Modell von Renovell, dessen Ergebnisse das Schaltverhalten von defekten Transistoren realistisch nachbilden und mit den Ergebnissen und Defektverläufen aus Veröffentlichungen anderer Autoren übereinstimmen, wie es in Abschnitt 3.6.3 dargelegt wurde. Die Umwandlung eines defektfreien Transistors in einen Transistor mit Gateoxiddefekt wurde schon im Abschnitt 3.6.2 erläutert.

5.3.1 Datenbasis des GOB-Simulators: Spice-Simulationen

Um die auftretenden Fälle innerhalb einer kombinatorischen Schaltung möglichst realistisch zu berechnen, wurde eine Vielzahl an Spice-Simulationen durchgeführt. Diese Simulationen sind notwendig, damit beim Erstellen der Gatterbibliothek in der nächsten Phase die notwendigen Daten für das Receiver- und Treibermodell des CCS-Modells für die Gatterebene korrekt vorhanden sind.

Es wurde jeweils ein Gatter mit unterschiedlichen Treiberstärken und Umgebungsvariablen simuliert und dabei ein Transistor des Gatters mit unterschiedlichen Widerstandswerten R_{GOB} als defekt mit dem oben beschriebenen Modell dargestellt. Ausgehend vom Single-Input-Switching Verhalten für die Gatterbibliothek wurde dann der Eingang geschaltet, der mit dem defekten Transistor verbunden ist. Spannungspegel etwaiger anderer Eingänge wurden stabil auf dem Wert gehalten, der eine Spannungspegeländerung am Ausgang des Gatters gewährleistet.

Um eine umfangreiche Datenbasis zum Erstellen einer CCS-Bibliothek zu erlangen, ist es notwendig, sowohl für das Receiver- als auch für das Treibermodell Daten mit variablen Simulationsparametern zu generieren. Das Leitungsmodell ist dabei irrelevant. Das Treibermodell ist eine zeit- und spannungsabhängige Stromquelle, während das Receivermodell zwei Kapazitätswerte umfasst, um die Eingangskapazität je nach Eingangsspannungspegel dynamisch anzupassen. Sowohl Treiber- als auch Receivermodell sind abhängig von der Eingangsflanke, der Ausgangskapazität und dem Zustand des Gatters. Da die Ausgangsflanke gleich der Eingangsflanke für das folgende Gatter ist, kann sie leicht über die Integration der Stromkurve des Gatterausganges berechnet werden, welche über verschiedene Werte der Eingangsflanke und der Ausgangskapazität interpoliert wird. Die Eingangsflanke wird über eine Spannungsquelle mit einstellbarem Widerstand R_{INIT} dargestellt, wobei R_{INIT} aufgrund der durch Defekte gesenkte Treiberstärken sehr große Werte annehmen kann. Neben diesen „Standard"-Simulationsparametern, wie Eingangsflanke t_{SLOPE_IN}, Lastkapazität C_{LOAD} und Gatterzustand, sind weitere Parameter wichtig, um die Besonderheiten der Gateoxiddefekte mit in Betracht zu ziehen. Zum einem können Netze, die üblicherweise je nach Schaltzustand des Gatters mit einer

Versorgungsleitung verbunden sind, durch ein defektes Gate mit der anderen Versorgungsspannungsleitung verbunden werden, so dass deren Spannungspegel Zwischenwerte annehmen, die nicht V_{DD} und V_{SS} entsprechen. Daher wurden der Spannungsquelle zur Eingangsflankengeneration auch von V_{DD} und V_{SS} variierende Werte zugewiesen. Zum anderen bestimmt auch die Gesamteingangskapazität über die Flanke am schaltenden Eingangspin, so dass auch eine Änderung von C_{IN_ADD} die Flankensteilheit über das RC-Glied modifiziert. Dieser Parameter umfasst auch die Anpassung der Eingangskapazität, da diese sich aufgrund des Defektmodells auch verändert, da einerseits der Originaltransistor an Breite verliert, während die beiden zusätzlichen Transistoren ihre Drainkapazitäten zu C_{IN} hinzufügen.

Schließlich sind auch noch die Variablen des Defekts, wie Defektposition und Defektwiderstand R_{GOB} zu variieren, um eine möglichst umfangreiche Datenbasis zu erhalten. In Tabelle 5-1 sind die Wertebereiche für variierende Parameter angegeben, während Abbildung 5-2 eine Übersicht über die Simulationsumgebung schematisch darstellt.

Tabelle 5-1 : Eingabeparameter des Spice-Simulationen

Simulationsparameter	Minimalwert	Maximalwert
V_{IN_HIGH}	$V_{DD} - 0.1 * (V_{DD} - V_{SS})$	V_{DD}
V_{IN_LOW}	V_{SS}	$V_{SS} + 0.1 * (V_{DD} - V_{SS})$
R_{INIT}	1 kΩ	80 MΩ
C_{LOAD}	0.5 fF	32 fF
C_{IN_ADD}	1/2/3 fF – $C_{IN\_ORIGINAL}$	32 fF – $C_{IN\_ORIGINAL}$
R_{GOB}	10 kΩ	1 MΩ

Da dies eine Vielzahl von Simulationen ergibt, wurde die Zahl der verfügbaren Gattertypen beschränkt. Die Wahl fiel auf Inverter in vier verschiedenen Treiberstufen, zweifach NAND und NOR mit drei Treiberstufen und jeweils zwei Treiberstufen für dreifach NAND und NOR. Mit dieser Auswahl ist es möglich, auch komplexere Gatter effizient durch Kombination dieser Grundgatter darzustellen.

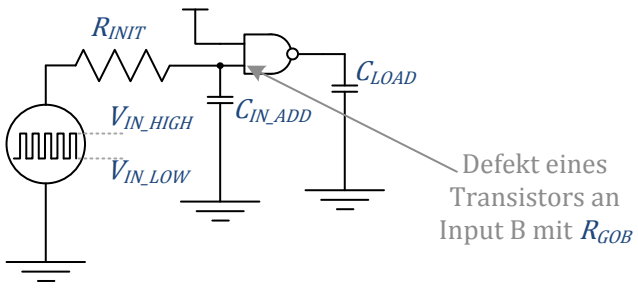


Abbildung 5-2 : Simulationssetup mit den zu variierenden Eingangsparametern für die grundlegenden Spice-Simulationen anhand des Beispiels eines NAND2

Folgende Simulationsergebnisse und Parameter sind für die Fehleranalyse und die Modifizierung der Gatterbibliothek im nächsten Schritt notwendig und wurden deshalb in die Datenbasis aufgenommen – Abbildung 5-3:

- Stromkurve $I_{OUT}(t)$ am Gatterausgang als CCS-Hauptbestandteil zur Berechnung der Ausgangsflanke
- Flankenanstiegs- bzw. -abstiegszeiten t_{RISE_IN} / t_{FALL_IN} am Gattereingang zur Bestimmung der CCS-Eingangskapazitäten und als Berechnungshilfe bei der Fehleranalyse
- Flankenanstiegs- bzw. -abstiegszeiten t_{RISE_OUT} / t_{FALL_OUT} am Gatterausgang als Berechnungshilfe bei der Fehleranalyse
- Minimale und maximale Spannungspegel an Gatterein- und -ausgängen V_{IN_MIN}, V_{IN_MAX}, V_{OUT_MIN}, V_{OUT_MAX} für die Spannungspegelanalyse
- Stromflüsse I_{GOB} durch den Widerstand R_{GOB} abhängig von Eingangsspannungspegeln (V_{IN}) zur Modifizierung der Stromkurven

Da diese Simulationen nur Defekte berücksichtigen, die Transistoren des jeweiligen Gatters ausbilden, sind weitere Analysen notwendig. Um Spannungspegel am Gatterausgang, der mit einem defektem Transistor verbunden ist, korrekt vorherzusagen, sind Daten über Spannungs-Strom-Kennlinien am Ausgang des Gatters für die Pegelanalyse notwendig, welche im nächsten Abschnitt erläutert wird. Dabei werden die Spannungspegel für jedes Netz über Ströme, die zum und vom Ausgangsknoten fließen, errechnet. Daher wurden die Gatter mit unterschiedlichen eigenen defekten Transistoren und einem Defekt am Ausgang simuliert. Es wurden nur Defekte untersucht, die der Ausgangsflanke entgegenwirken. Das bedeutet beispielsweise, dass bei einer

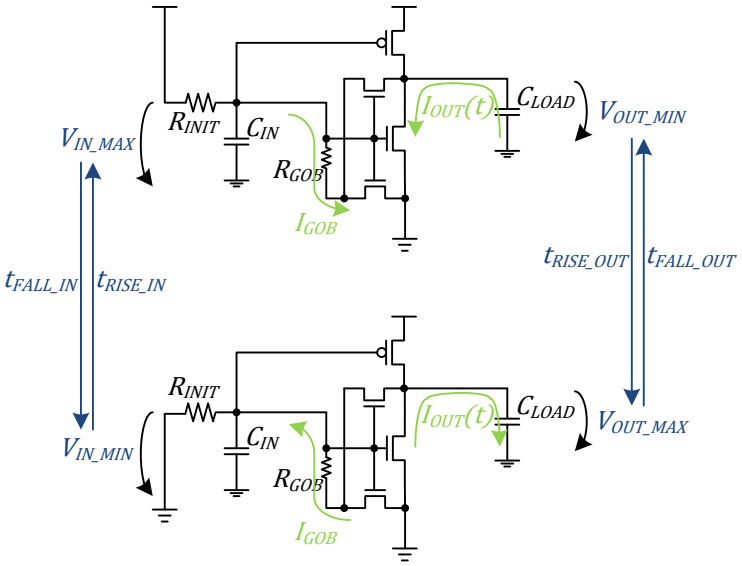


Abbildung 5-3 : Darstellung der gespeicherten Parameter (alle im Bild benannten) für die Verwendung in der Designanalyse anhand der Simulation eines n-MOSFET-Defektes eines Inverters

steigenden Flanke ein defekter n-MOSFET mit dem Gatterausgang verbunden wurde, da dies dazu führt, dass die zu erreichende Endspannung V_{OUT} auf einen Wert unter V_{DD} gesenkt wird. Dadurch fließt ein Strom I_{GOB} vom sich öffnenden Pull-Up-Netzwerk über den Defekt zur Masse. Bei der fallenden Flanke wurde dementsprechend ein defekter p-MOSFET am Ausgang angelegt. Diese Spannungs-Strom-Kennlinie aus V_{OUT} und I_{GOB} (nachfolgend SSK_{DB_HIGH} für den High-Pegel und SSK_{DB_LOW} für den Low-Pegel genannt) wurde für verschiedene Gatter, Gatterdefekte und Defekte am Ausgang ausgewertet und in einer weiteren Datenbasis gespeichert. Mit diesen beiden Datenbasen ist es möglich, die Spannungspegelanalyse und die Stromkurvenanpassung durchzuführen. Die bereitgestellten Daten dienen dabei als Stützpunkte für Interpolationen, falls Werte der Anfrageparameter an die Datenbasis nicht exakt den gespeicherten Werten entsprechen sollten.

Da durch die Defekte auch die Spannungspegel der einzelnen Knoten, die V_{DD} oder V_{SS} entsprechen sollten, diese Grenzen nicht mehr erreichen können, wurden folgende Grenzen für die Defektdeklaration eingeführt. Die untere Grenze V_{LOW} setzt das Limit an einen Knoten, das nicht überschritten werden darf, damit ein Design aufgrund des Defektes an diesem Knoten noch nicht als fehlerhaft klassifiziert wird, während V_{HIGH} nicht unterschritten werden darf. Das sind auch die beiden Parameter aus Abbildung 2-9, die die Grenzwerte für die Flankenanstiegs- und -abfallzeiten definieren und somit auch ein Parameterset für die Standardbibliothek darstellen, das für die Spannungspegelanalyse genutzt werden kann. Folgende Flankengrenzen wurden festgelegt:

$$V_{LOW} = V_{SS} + \frac{1}{3} \cdot (V_{DD} - V_{SS}) \tag{68}$$

$$V_{HIGH} = V_{DD} - \frac{1}{3} \cdot (V_{DD} - V_{SS}) \tag{69}$$

Diese Grenzen erlauben eine flexible Bibliotheksgenerierung bei gleichzeitiger Einhaltung von störungsfreien Spannungspegeln, die am Ausgang eines defektfreien Gatters wieder zu den ursprünglichen Spannungspegeln V_{DD} und V_{SS} führen. Flexibel in dem Sinne, dass die Spannungspegel an Netzen mit defekten Transistoren bis zu diesen Werten ansteigen bzw. absinken können, da erst bei Über- bzw. Unterschreitung von V_{LOW} und V_{HIGH} ein fehlerhafter Spannungspegel durch die Spannungspegelanalyse angezeigt wird. Dadurch war eine konsistente Behandlung der Flanken an Gatterein- und -ausgängen möglich, da die Grenzen für die Flankengenerierung immer gleich bleiben.

5.4 Gatterlevel-Defektanalyse

Die Defektanalyse auf Gatterebene erfolgt in zwei Schritten. Als erstes werden die minimal und maximal erreichbaren Spannungspegel V_{OUT_MIN} und V_{OUT_MAX} der Gatterausgänge ermittelt, damit die Pegel aller Netze zwischen den Gatters des Designs überprüft werden. Darauf folgt die Generierung einer modifizierten Gatterbibliothek, damit sowohl das statische Zeitverhalten, als auch die Leistungsaufnahme des Designs mit proprietären Analysetools ermittelt werden können.

Die Defektanalyse erfolgt sowohl zeitlich als auch schaltungstechnisch sequentiell. Das bedeutet einerseits, dass sie erst für alle Netze des Designs für einen Zeitschritt durchgeführt wird, bevor zum nächstfolgenden Zeitschritt gewechselt wird. Andererseits wird die Fehleranalyse für jedes Gatter von den Eingangspins bis zu den Ausgangspins des Designs berechnet, quasi von vorne nach hinten, da Spannungspegel an den Gattereingängen Auswirkungen auf die Spannungspegel V_{OUT_MIN} bzw. V_{OUT_MAX} und die Stromkurve $I_{OUT}(t)$ an den Ausgängen haben. Da mit der Gatterbibliothek das statische Zeitverhalten und die Leistung pinweise berechnet werden, werden folgende Punkte des Ablaufs der Defektanalyse je Zeitschritt t_X und Gatter auch von Pin zu Pin neu durchgeführt, da je nach Pin unterschiedliche Transistoren am Eingang und im Gatter selbst berücksichtigt werden müssen:

1. Prüfung für alle Inputpins, ob mit Eingang verbundene Gatter analysiert wurden
    a. Wenn nicht, dann weiter mit nächstem Gatter des Analyseablaufes
2. Prüfung für alle Inputpins, ob Defekte des Gatters oder Defekte am Ausgang des Gatters neu entstanden sind oder deren Ausmaß sich verändert hat
    a. Wenn nicht, dann weiter mit nächstem Gatter des Analyseablaufes
3. Spannungspegelanalyse (SPA) des Gatters je Inputpin zur Ermittlung der minimalen und maximalen Spannungspegel am Ausgang V_{OUT_MIN} und V_{OUT_MAX}
    a. Wenn $V_{OUT\_MIN} > V_{LOW}$ oder $V_{OUT\_MAX} < V_{HIGH}$, dann Gatter und Design als defekt klassifizieren und Abbruch der Defektanalyse mit Defektzeitpunkt $t_{DESIGNFEHLER} = t_X$
4. Berechnung der Stromkurve $I_{OUT}(t)$ je Inputpin
5. Modifizierung der Gatterbibliothek für Zeitpunkt $t = t_X$ je Inputpin
    a. Anpassen der Netzliste
    b. Anpassen der Gatterbibliothek

5.4.1 Spannungspegelanalyse

Die Spannungspegelanalyse (SPA) verfolgt zwei Ziele. Einerseits wird ermittelt, ob Transistordefekte in einem Design eine so verringerte Spannungsdifferenz zwischen High- und Low-Pegel auf einzelnen Designnetzen verursachen, dass eine korrekte Verarbeitung der Pegel durch die nachfolgenden Standardgatter nicht immer möglich ist. Andererseits werden die berechneten Strom- und Spannungswerte in der folgenden Stromkurvenmodifizierung

weiterverarbeitet. Nachfolgend wird der Algorithmus für Berechnung der Spannungspegel eines Netzes vorgestellt, das am Gatterausgang mit einem oder mehreren defekten Transistoren eines oder mehrerer anderer Gatter verbunden ist. Der Ablauf gliedert sich dabei in folgende Schritte:

1. Ermittlung der minimalen und maximalen Spannungspegel $V_{OUT\_MIN\_BASE}$ und $V_{OUT\_MAX\_BASE}$ des aktuellen Gatters ohne Einbeziehung von gatterfremden Defekten am Gatterausgang
2. Berechnung des minimal erreichbaren Spannungspegel V_{OUT_MIN} und des dazugehörigen statischen Stromflusses I_{OUT_MIN}
    a. Prüfung, ob ein oder mehrere defekte p-MOSFET mit dem Ausgang verbunden sind
    b. Wenn nicht, dann ist $V_{OUT\_MIN} = V_{OUT\_MIN\_BASE}$ und $I_{OUT\_MIN} =0$; weiter mit 3. sonst
    c. Datenbankabfrage der Spannungs-Stromkurve (SSK_P) bestehend aus I_{GOB} und V_{IN} der defekten p-MOSFETs
    d. Datenbankabfrage der Spannungs-Stromkurve (SSK_N) bestehend aus I_{GOB} und V_{IN} der defekten n-MOSFETs, falls welche vorhanden sind
    e. Vergleich der Spannungs-Stromkurven SSK_P und SSK_N mit der SSK_{DB_LOW} des jeweiligen Gatters zur Ermittlung von V_{OUT_MIN} und I_{OUT_MIN}
3. Berechnung des maximal erreichbaren Spannungspegel V_{OUT_MAX} und des dazugehörigen Stromflusses I_{OUT_MAX}
    a. Prüfung, ob ein oder mehrere defekte n-MOSFET mit dem Ausgang verbunden sind
    b. Wenn nicht, dann ist $V_{OUT\_MAX} = V_{OUT\_MAX\_BASE}$ und $I_{OUT\_MAX} =0$; weiter mit 4. sonst
    c. Datenbankabfrage der Spannungs-Stromkurve (SSK_N) bestehend aus I_{GOB} und V_{IN} der defekten n-MOSFETs
    d. Datenbankabfrage der Spannungs-Stromkurve (SSK_P) bestehend aus I_{GOB} und V_{IN} der defekten p-MOSFETs, falls welche vorhanden sind
    e. Vergleich der Spannungs-Stromkurven SSK_P und SSK_N mit der SSK_{DB_HIGH} des jeweiligen Gatters zur Ermittlung von V_{OUT_MAX} und I_{OUT_MAX}
4. Übernahme der errechneten Werte für die Spannungspegelauswertung, zur Modifizierung der Stromkurve $I_{OUT}(t)$ und zur Anpassung der Werte für die Leckströme in der modifizierten Gatterbibliothek

Anschließend werden die einzelnen Schritte des Algorithmus näher erläutert, wobei sich Punkt 2 und 3 derart gleichen, dass nur Punkt 2 erörtert wird und Unterschiede zu Punkt 3 erklärt werden.

Die Berechnung der Spannungspegel ohne Defekt am Ausgang (Punkt 1) wird vorgenommen, um Abfrageparameter für die Datenbankanfrage zu erhalten. Eingangsparameter für die in Abschnitt 5.3.1 vorgestellten Spice-Simulationen waren u.a. die Eingangspegel. Da ein Netz am

Ausgang eines Standardgatters allein durch einen eigenen Defekt nicht mehr die maximal und minimal möglichen Pegel V_{DD} und V_{SS} erreichen könnte, wurde dies in den Simulationen berücksichtigt. Die ermittelten Werte $V_{OUT\_MIN\_BASE}$ und $V_{OUT\_MAX\_BASE}$ dienen im Verlauf der Spannungspegelanalyse als Abfrageparameter für die Berechnungen der Spannungs-Stromkurven SSK_N und SSK_P, da unterschiedliche Pegel am Eingang des Defekts diese Kurven beeinflussen. Des Weiteren werden die Werte für die Berechnung von $V_{OUT\_MIN\_BASE}$ und $V_{OUT\_MAX\_BASE}$ der nachfolgenden Gatter benötigt, da sie die Eingabeparameter V_{IN_HIGH} und V_{IN_LOW} darstellen und somit Abfrageparameter für die Datenbasis darstellen. Sollten für verschiedene Inputpins des Gatters unterschiedliche Werte erreicht werden, wird im jeden Fall der ungünstigere Wert übernommen.

Neben dem eigenen Defekt und den Eingangspegeln am zu untersuchenden Gatter sind die Ausgangspegel abhängig von Schäden an Transistoren der Gatter, die über das Ausgangsnetz mit dem aktuellen Gatter verknüpft sind. Diese Einflüsse werden in der Spannungspegelanalyse in Punkt 2 und 3 berücksichtigt. Der Minimalpegel V_{OUT_MIN}, der idealerweise V_{SS} entsprechen sollte, wird anhand der fallenden Flanke des aktuellen Gatters berechnet. Negativ beeinflusst wird er durch folgende Parameter:

- Ein Maximalpegel am Eingang, mit $V_{IN\_MAX} < V_{DD}$, d.h. der n-MOSFET zur Entladung der Lastkapazität ist nicht komplett durchgeschaltet
- Ein Defekt am n-MOSFET oder p-MOSFET des aktuellen Gatters und Pins
- P-MOSFET-Defekte nachfolgender Gatter, da sie das Ausgangsnetz mit V_{DD} verbinden können

Nachfolgende n-MOSFET-Defekte wiederum ziehen den Pegel Richtung V_{SS}, was den Minimalpegel positiv beeinflussen würde. Allerdings sind die Auswirkungen nicht so groß wie beim Defekt eines vergleichbaren p-MOSFET, da der Widerstand des Defektes aufgrund des fast geschlossenen Transistors sehr groß ist. Diese Effekte kehren sich bei der Berechnung von V_{OUT_MAX} um, da hier die steigende Ausgangsflanke entscheidend ist. Hierbei ist dann am Eingang auch der Minimalpegel $V_{OUT\_MIN\_BASE}$ des Vorgängers entscheidend. Tabelle 5-2 zeigt die positiven und negativen Effekte auf, die bei mehreren Defekten an einem Netz auftreten können, wie es in Abbildung 5-4 dargestellt ist:

Tabelle 5-2 : Auswirkungen bestimmter Umgebungsparameter auf die minimalen und maximalen Spannungspegel: mit negativem (-), positivem (+) oder keinem (o) Einfluss

Auswirkungen auf das Ausgangsnetz	V_{OUT_MIN}	V_{OUT_MAX}
$V_{IN\_MIN} > V_{SS}$	(o)	sinkt (-)
$V_{IN\_MAX} < V_{DD}$	steigt (-)	(o)
Eigener Defekt	steigt (-)	sinkt (-)
Defekt eines p-MOSFET am Ausgang	steigt (-)	steigt (+)
Defekt eines n-MOSFET am Ausgang	sinkt (+)	sinkt (-)

Abbildung 5-4 zeigt vereinfacht, wie diese Gateoxiddefekte die Pegel im statischen Fall beeinflussen. In diesem Beispiel werden fünf Pins fünf verschiedene Gatter über das Netz n von einem Inverter getrieben, wobei drei Defekte mit n verbunden sind – Abbildung 5-4 a). Wenn nun dieses Netz n in der Spannungspegelanalyse betrachtet werden soll, werden das jeweilige Netz und seine mit ihm verbundenen Transistoren in ein einfaches elektrisches Ersatzschaltbild überführt. Da zur Berechnung der statische Fall genügt, werden der Treiber und die Defekte an Transistoren der nachfolgenden Gatter als Widerstände dargestellt. Für den Fall, dass der Minimalpegel V_{OUT_MIN} von n untersucht werden soll, ergibt sich Abbildung 5-4 b), im Fall des Maximalpegels V_{OUT_MAX} Abbildung 5-4 c).

Da in der Datenbank Strom-Spannungskurven enthalten sind und der Berechnungsalgorithmus sich schrittweise an den korrekten Wert für V_{OUT_MIN} und V_{OUT_MAX} annähert, stellen der Treiber und die Defekte streng genommen keine Widerstände sondern spannungsabhängige Stromquellen dar, die erst im statischen Fall der Extrempegel als Widerstände angesehen werden können. So wird auch zuerst die spannungsabhängige Stromkurve des Treibers ermittelt. Diese ist schon als SSK_{DB_LOW} für den niedrigen und als SSK_{DB_HIGH} für den hohen Ausgangspegel in der Datenbank vorhanden und kann anhand der aktuellen Parameter des Treibers, die als Eingangsparameter für die Datenbank genommen werden, ausgegeben werden. Im Fall des niedrigen Ausgangspegels V_{OUT_MIN} ist dieser abhängig

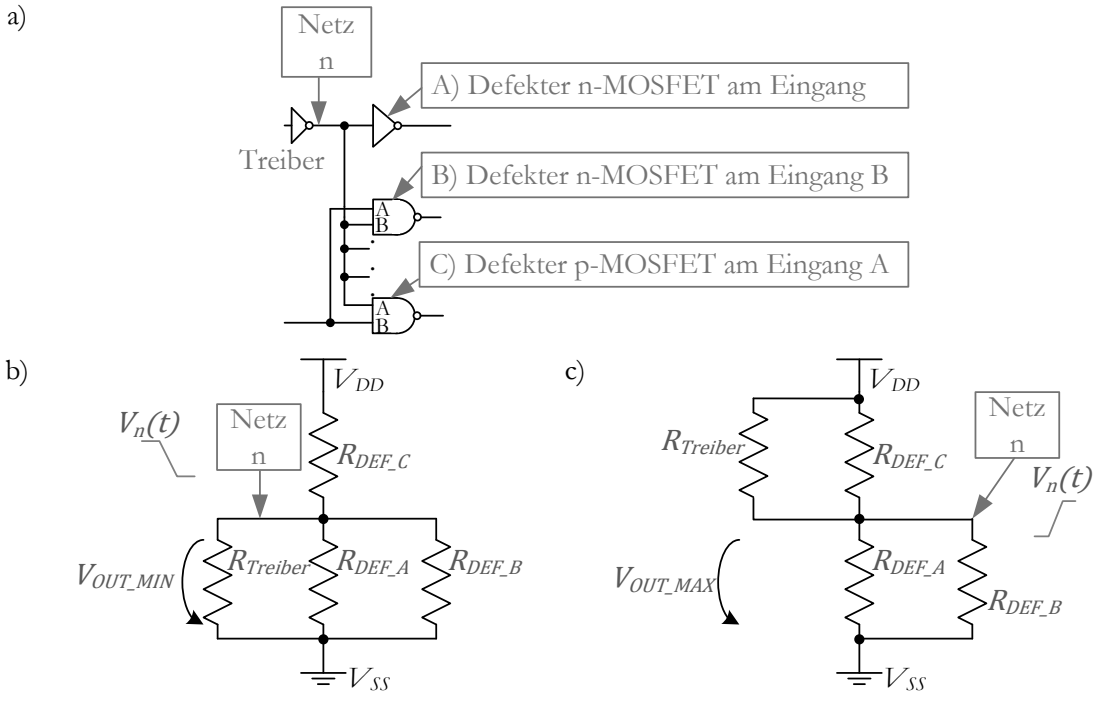


Abbildung 5-4 : Spannungspegelanalyse: Berechnung der extremen Spannungspegel durch elektrische Ersatzschaltbilder im statischen Fall, wenn das Umladen von C_{LOAD} beendet ist:

a) Analyse der Spannungspegel von Netz n, das mit einem treibenden Inverter und drei defekten Transistoren als Eingangsnetz verbunden ist
b) Ersatzschaltbild zur Berechnung des Minimalpegels V_{OUT_MIN}
c) Ersatzschaltbild zur Berechnung des Maximalpegels V_{OUT_MAX}

vom Parameter V_{IN_MAX} des Eingangsnetzes, welches $V_{OUT\_MAX\_BASE}$ des mit dem Pin verbundenen treibenden Gatters entspricht, bei V_{OUT_MAX} ist $V_{IN\_MIN} = V_{OUT\_MAX\_BASE}$ entscheidend für den Ausgangspegel. Weitere Eingangsparameter für die Datenbank sind aktuelle Gatterparameter des Treibers, wie der Defektwiderstand R_{GOB} eines vorhandenen defekten Transistors, der Gattertyp und Lastkapazitäten an Ein- und Ausgangspin. Sie bilden die diskrete Funktion $I_{TREIBER}(V)$. Die beiden für die aktuelle Situation des Treibers ermittelten Strom-Spannungskurven SSK_{DB_LOW} und SSK_{DB_HIGH} werden nun als diskrete Stromfunktion $I_{TREIBER}(V)$ für die Berechnung der Spannungspegel V_{OUT_MIN} bzw. V_{OUT_MAX} verwendet:

V_{OUT_MIN}-Berechnung: $$I_{TREIBER}(V) = SSK_{DB\_LOW} \tag{70}$$

V_{OUT_MAX}-Berechnung: $$I_{TREIBER}(V) = SSK_{DB\_HIGH} \tag{71}$$

Weiterhin sind die Stromwerte der Defekte anderer Gatter am Ausgangspin notwendig für die Berechnung der Maximalpegel. Aus Abbildung 5-4 b) ist ersichtlich, dass bei fehlendem p-MOSFET-Defekt keine Notwendigkeit besteht, den Minimalpegel zu berechnen, da dann $V_{OUT\_MIN} = V_{SS}$ ist. Sollte allerdings ein Defekt das Netz n mit V_{DD} verbinden, müssen die Strom-Spannungskurven durch alle Defekte berücksichtigt werden. Analog dazu ist eine Berechnung von V_{OUT_MAX} nur notwendig, wenn ein n-MOSFET das Netz mit V_{SS} verbindet, sonst ist $V_{OUT\_MAX} = V_{DD}$. Die spannungsabhängigen Ströme der Defekte werden zur aktuellen Situation passend aus der Datenbasis gesucht und in zwei diskreten Strom-Spannungskurven SSK_N und SSK_P zusammengefasst. Sie umfassen die Summe aller n-MOSFET-Defekte:

$$I_{N\_GOB}(V) = SSK_N = \sum_{\forall\, n-MOSFET-DEFEKTE} I_{GOB}(V) \tag{72}$$

bzw. aller p-MOSFET-Defekte:

$$I_{P\_GOB}(V) = SSK_P = \sum_{\forall\, p-MOSFET-DEFEKTE} I_{GOB}(V) \tag{73}$$

Um die minimalen und maximalen Spannungspegel zu erhalten, werden der Knotenpunktsatz von Kirchhoff angewendet:

$$I_{TREIBER}(V_{OUT\_MIN}) = I_{P\_GOB}(V_{OUT\_MIN}) - I_{N\_GOB}(V_{OUT\_MIN}) = I_{GOB\_MIN} \tag{74}$$

$$I_{TREIBER}(V_{OUT\_MAX}) = I_{N\_GOB}(V_{OUT\_MAX}) - I_{P\_GOB}(V_{OUT\_MAX}) = -I_{GOB\_MAX} \tag{75}$$

Die Werte von $I_{TREIBER}(V_{OUT\_MIN})$ und der Differenz aus Gleichung (74) werden separat aus den vorhandenen SSK für Spannungswerte von $V_i = V_{SS}$ bis $V_i = V_{LOW} + vdiff$ berechnet, wobei V_i jeweils um *vdiff* erhöht wird. Bei dem Wert V_i, bei dem sich die geringste Unterschied zwischen beiden Seiten der Gleichung ergibt, wird als Minimalpegel festgelegt. Sollte dies der Wert $V_i = V_{LOW} + vdiff$ sein, erreicht V_{OUT_MIN} nicht den akzeptierten Mindestwert V_{LOW} und das Ausgangsetz und somit das Gatter und das Design werden als defekt deklariert. Bei Gleichung (75) und V_{OUT_MAX} wird genauso so verfahren, allerdings reichen die Spannungswerte von $V_i = V_{DD}$ bis $V_i = V_{HIGH} - vdiff$.

Neben der Kontrolle der Spannungspegel werden die errechneten Spannungspegel für die Modifizierung der Stromkurve I_{OUT} verwendet, welche im nächsten Kapitel erläutert wird. Die dazugehörigen Ströme $I_{GOB}(V_{OUT\_MIN})$ und $I_{GOB}(V_{OUT\_MAX})$ für jedes der defekten Gatter am Ausgangspin sind statische Stromflüsse, die zur Berechnung der Leckströme mit herangezogen werden. Sie finden daher auch Eingang im Schritt zur Anpassung der Gatterbibliothek, welcher im Abschnitt 5.4.3 behandelt wird.

5.4.2 Berechnung der CCS-Stromkurve

Neben der Veränderung von Spannungspegeln auf Designnetzen bewirken Gateoxiddefekte eine Modifizierung des Zeitverhaltens des Designs. Das wird bei der (Neu-)Berechnung der CCS-Stromkurve berücksichtigt. Nachdem einmalig die CCS-Stromkurven nichtdefekter Gatter aus den Spice-Simulationen übernommen wurden, werden diese auch als Ausgangsbasis für die Stromkurven $I_{OUT}(t)$ der Gatter genutzt, dessen Zeitverhalten sich verändert. Dazu gehören nicht nur die Gatter, deren Transistoren einen Defekt aufweisen, sondern auch die defektfreien Gatter, deren Ausgänge mit einem defekten Transistor verbunden sind. Deren Stromkurve wird auch angepasst, da deren Ausgang durch das defekte Gate des Transistors mit V_{SS} oder V_{DD} über den Widerstand R_{GOB} verbunden wird. Dadurch ist das Auf-und Entladen der Kapazität C_{LOAD} am Ausgangsnetz gestört, und die für das statische Zeitverhalten relevante Umladezeiten t_{RISE} bzw. t_{FALL} werden verändert und können dadurch alle nachfolgenden Gatter, auch die ohne Defekt, beeinflussen. Bei defekten Gattern wird das Umladen der Ausgangskapazität C_{LOAD} allein durch die verminderte Treiberstärke verlangsamt. Diese Veränderungen werden mittels der Modifizierung der CCS-Stromkurve berücksichtigt, da die zur Berechnung der Gatterlaufzeit t_D und der Umladezeiten t_{RISE}, t_{FALL} verwendete Stromkurve $I_{OUT}(t)$ den (Ent-)Ladestrom der Lastkapazität C_{LOAD} des Gatters darstellt. Diese kann wie im statischen Fall über die Gleichungen (74) und (75) berechnet werden. Allerdings müsste die SSK für alle Spannungspegel zwischen V_{OUT_MIN} und V_{OUT_MAX} berechnet werden, was einen großen Berechnungsaufwand zur Folge hätte. Des Weiteren muss der Parameter Zeit und dadurch auch die Kapazität C_{LOAD} am jeweiligen Netz berücksichtigt werden. Aus diesen Gründen wurde folgende Vorgehensweise für die Modifikation der CCS-Stromkurve festgelegt:

1. Berechnung der Stromflüsse $I_{N\_GOB}(V_{LOW})$, $I_{N\_GOB}(V_{HIGH})$, $I_{N\_GOB}(V_{DD}/2)$, $I_{P\_GOB}(V_{LOW})$, $I_{P\_GOB}(V_{HIGH})$ und $I_{P\_GOB}(V_{DD}/2)$ durch alle vorhanden defekten Transistorgates am Ausgangsnetz zu bestimmten Spannungspegeln
2. Festlegung der CCS-Eingangsparameter t_{SLOPE_IN} und C_{LOAD}
3. Ermittlung der CCS-Referenzzeit t_{VIN_HALF}, die Zeit zur der die Eingangsspannung V_{IN} die Hälfte von V_{DD} erreicht
4. Berechnung der modifizierten Stromkurve $I_{MOD}(t)$ für jede Eingangskombination aus Punkt 2, die den Strom $I_{OUT}(t)$ ersetzt.
5. Anpassung der Stromkurve $I_{MOD}(t)$ zur Einhaltung der CCS-Bibliotheksvorgaben und Unterteilung der Stromkurve in maximal 10 Funktionspaare

Die Umgebungsparameter zur Auswahl der Werte aus der Spice-Datenbank werden aus dem Design übernommen. Im Folgenden werden die einzelnen Punkte des Modifikationsalgorithmus näher erläutert.

Im Gegensatz zur Spannungspegelanalyse ist das Umladen der Lastkapazität ein dynamischer Vorgang, da sich die Spannungspegel am Eingang und am Ausgang des betrachteten Gatters mit der Zeit verändern und damit auch die Stromflüsse am Ausgangsnetz. Kernstück der CCS-Stromkurven-Modifikation ist die Veränderung der ursprünglichen Stromkurve mittels Strömen, die durch alle vorhandenen Gateoxiddefekte zu bestimmten Spannungspegel des Ausgangsnetzes fließen. Dazu werden neben den schon bestimmten Strom-Spannungspaaren (I_{GOB_MIN}; V_{OUT_MIN}) und (I_{GOB_MAX}; V_{OUT_MAX}) aus Gleichung (74) und (75) der erfolgten Spannungspegelanalyse weitere Strom-Spannungspaare berechnet. Diese Strom-Spannungspaare ergeben sich folgendermaßen:

$$I_{GOB\_LOW} = I_{GOB}(V_{LOW}) = I_{P\_GOB}(V_{LOW}) - I_{N\_GOB}(V_{LOW}) \quad (76)$$

$$I_{GOB\_HIGH} = I_{GOB}(V_{HIGH}) = I_{P\_GOB}(V_{HIGH}) - I_{N\_GOB}(V_{HIGH}) \quad (77)$$

$$I_{GOB\_HALF} = I_{GOB}\left(\frac{V_{DD} - V_{SS}}{2}\right) = I_{P\_GOB}\left(\frac{V_{DD} - V_{SS}}{2}\right) - I_{N\_GOB}\left(\frac{V_{DD} - V_{SS}}{2}\right) \quad (78)$$

Diese Stromflüsse durch die angeschlossenen Defekte zu bestimmten Spannungspegeln auf dem Ausgangsnetz führen zu einer Funktion $I_{GOB}(V_{OUT})$ für das jeweilige Netz, wenn man die Werte zwischen den Stützstellen interpoliert – Abbildung 5-5. Die Stromwerte sind in der Spice-Datenbank durch die vorherigen Simulationen vorhanden und werden dann je nach Design- und Defektsituation summiert.

Die Modifikation der CCS-Stromkurve erfolgt dann für jedes Parameterpaar (Eingangsflanke; Lastkapazität) separat. Die Ermittlung des Referenzwertes t_{VIN_HALF} – Zeitpunkt an dem die Eingangsspannung den Wert $V_{DD}/2$ passiert – dient dabei einerseits der Synchronisation der Stromkurven der Parameterpaare und ist andererseits zwingend für das CCS-Modell vorzugeben.

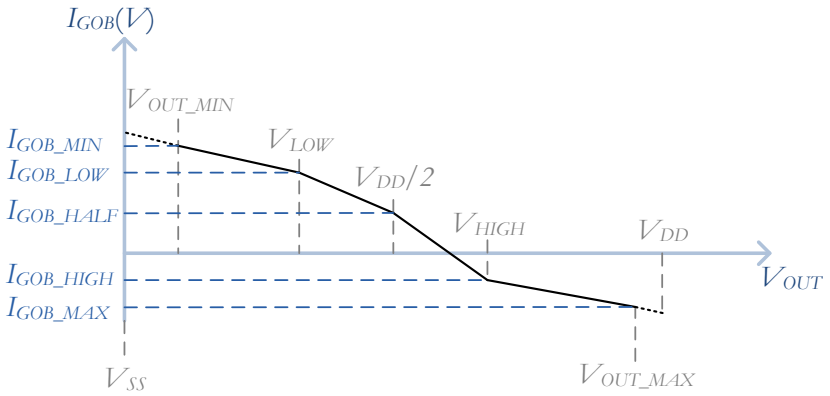


Abbildung 5-5 : Ermittelter Stromfluss durch die Defekte an Netz n in Abhängigkeit vom Ausgangssignalpegel V_{OUT}; bezogen auf das Beispiel aus Abbildung 5-3 ist das Ausmaß des p-MOSFET-Defektes größer als das der beiden n-MOSFET-Defekte zusammen, da sowohl $|I_{GOB\_MIN}| > |I_{GOB\_MAX}|$ und $V_{OUT\_MIN} - V_{SS} > V_{DD} - V_{OUT\_MAX}$

Dieser Wert ist für alle Parameterpaare für dieses Gatter und die vorgegebene Flankenrichtung in der CCS-Bibliothek gleich.

Ausgehend von der ursprünglichen Stromkurve $I_{OUT}(t)$ wird diese mittels der Stromkurve $I_{GOB}(V_{OUT})$ an die jeweilige Defektsituation angepasst. Dabei wird die Strom-Spannungskurve $I_{OUT}(t)$ des Gatters aus der Spice-Datenbank entnommen, die der jeweiligen Situation des treibenden Gatters mit Berücksichtigung defekter Transistoren und deren Ausmaß R_{GOB} am genauesten entspricht. Diese Stromwerte der Kurve werden für die fallende Flanke mit -1 multipliziert. Für die Berechnung wird die Kurve $I_{OUT}(t)$ in eine Vielzahl an Teilstücken, deren Abstand *dt* entspricht, aufgeteilt. Nun wird für jedes Teilstück vom Funktionsbeginn $I(t_0) = 0$ und $I(t_0 + dt) > 0$ an, die Kurve sowohl vom Stromwert als auch vom Zeitpunkt angepasst. Das Prinzip beruht darauf, dass die Stromkurve einem Transport der Ladung $Q_{LOAD} = C_{LOAD} / (V_{DD} - V_{SS})$ mittels Strom $I_{OUT}(t)$ darstellt. Dieser Ladungstransport stellt das Umladen der Lastkapazität C_{LOAD} dar. Dieses Umladen wiederum wird durch die Defekte beeinflusst (Tabelle 5-3):

Tabelle 5-3 : Auswirkungen von Gateoxiddefekten am Ausgangsnetz eines Gatters auf die Umladeflanken t_{RISE}/t_{FALL} relativ zu den Originalflanken ohne Defekte: mit negativem (-), positivem (+) oder keinem (o) Einfluss

Auswirkungen auf das Zeitverhalten	**t_{RISE}**	**t_{FALL}**
$I_{GOB} > 0$	sinkt (+)	steigt (-)
$I_{GOB} < 0$	steigt (-)	sinkt (+)
Eigener Defekt	steigt (-)	steigt (-)

Ein Wert $I_{GOB} > 0$ bedeutet dabei, dass p-MOSFET-Defekte überwiegen und stets ein Stromfluss von V_{DD} zum Gatterausgang vorherrscht. Ist I_{GOB} negativ, sind n-MOSFET-Defekte vorhanden und neben dem Strom des treibenden Gatters ist ein Strom vom Ausgangsnetz zu V_{SS} vorhanden. Allerdings kann es sein, dass bei mehreren Defekten am Ausgang mit unterschiedlich betroffenen Transistortypen $I_{GOB}(V_{OUT\_MIN}) > 0$ ist, da R_{GOB} an den n-MOSFET-Defekten durch den fast geschlossenen Transistor sehr groß ist und $I_{GOB}(V_{OUT\_MAX}) < 0$, da nun die p-MOSFET fast geschlossen sind.

Diese Einflüsse werden, wie in Abbildung 5-6 dargestellt, berücksichtigt. Für jedes Teilstück von t_{i-1} zu $t_i = t_{i-1} + dt$ wird die transportierte Ladung q_i folgendermaßen berechnet:

$$q_i = \frac{I_{OUT}(t_i) + I_{OUT}(t_{i-1})}{2} \cdot dt \tag{79}$$

Daraus ergibt sich folgende Spannungsdifferenz dV_i zum vorigen Zeitpunkt auf dem Ausgangsnetz:

$$dV_i = \frac{q_i}{C_{LOAD}} \tag{80}$$

Die Spannungsdifferenz wird zum Wert der Spannung V_{i-1} des vorigen Zeitpunktes addiert, um den aktuellen Spannungspegel $V_i = V_{OUT}(t_i)$ auf dem Ausgangsnetz zu erhalten. Der Strom

$I_{GOB}(V_i)$ wiederum stört den ursprünglichen Ladungstransport, in dem er durch den eigenen Stromfluss Ladung auf C_{LOAD} lädt oder Ladung von C_{LOAD} abzieht. Eine dazugehörige Kurve kann mit q_i und dem dazugehörigen Spannungspegel $V_{OUT}(t_i)$ abgebildet werden, wobei Werte zwischen den einzelnen Zeitpunkten t_i interpoliert werden können. Mit Hilfe der Abhängigkeiten $I_{GOB}(V_{OUT})$ und $I_{OUT}(t)$ wird die modifizierte Stromkurve $I_{MOD}(t)$ berechnet, welche den realen Ladungstransport zum Umladen der Lastkapazität C_{LOAD} darstellt. Folgender Wert zum Zeitpunkt t_{MOD_i} ergibt sich:

$$I_{MOD}\left(t_{MOD\_i}\right) = I_{OUT}\left(t_i\right) + I_{GOB}\left(V_i\right) \tag{81}$$

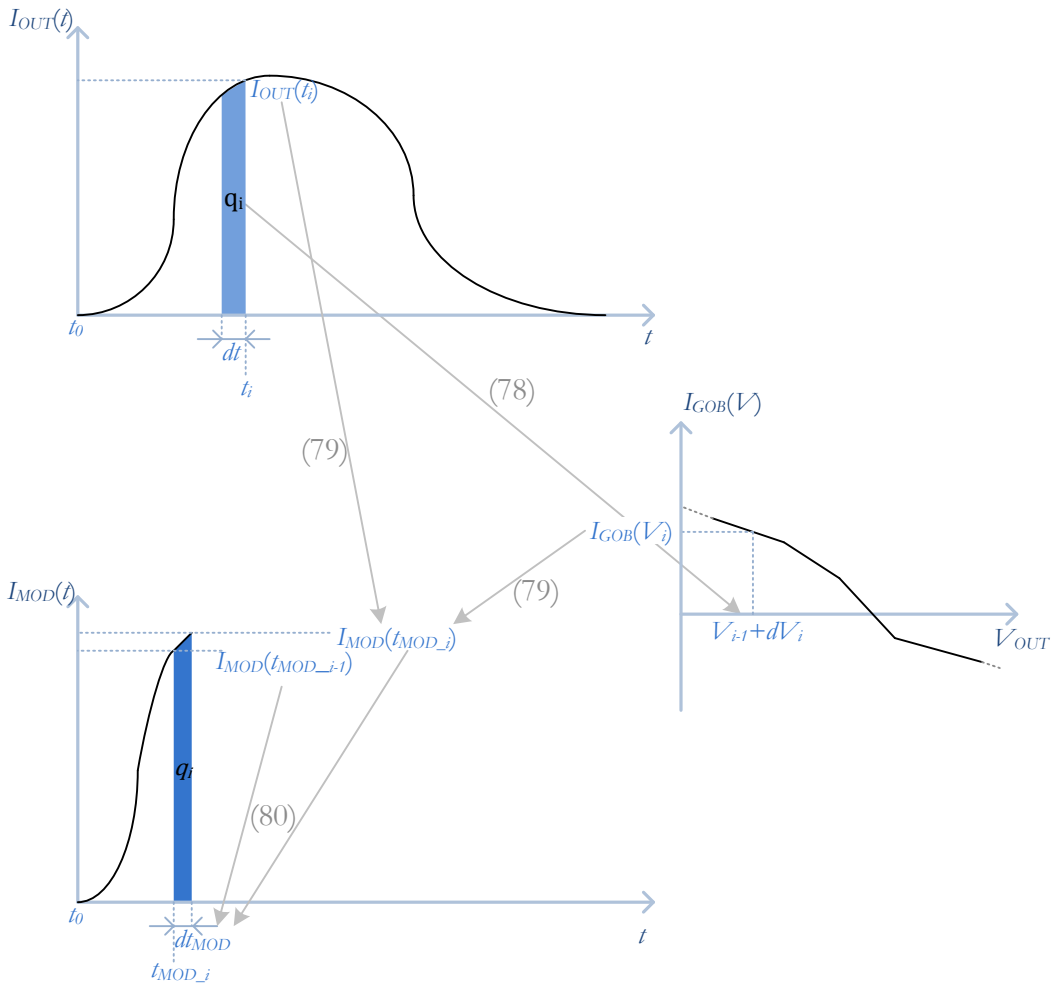


Abbildung 5-6 : Berechnung eines neuen Wertepaares der modifizierten Stromkurve $I_{MOD}(t)$ für einen ansteigenden Ausgangsspannungspegel: Der Betrag I_{MOD} ergibt sich in diesem Beispiel aus der Ladungsänderung q_i für C_{LOAD} durch den Stromfluss am Treiber (I_{OUT}) und den spannungsabhängigen Stromfluss I_{GOB} von V_{DD} zu C_{LOAD}; der neue Zeitpunkt $t_{MOD\_i} + dt_{MOD}$ ist notwendig, damit zu diesem Zeitpunkt der gleiche Ausgangspegel mit $I_{MOD}(t)$ vorhanden ist wie zum Zeitpunkt $t_i + dt$ mit der Originalstromkurve $I_{OUT}(t)$

Damit der nächste Wert der Stromkurve zum Zeitpunkt $t_i + dt$ angepasst werden kann, muss erst wieder der Spannungspegel der Originalkurve zu diesem Zeitpunkt erreicht sein, da $I_{OUT}(t)$ auch von V_{OUT} abhängt. Dies wird durch folgende Berechnung der Zeitdifferenz dt_{MOD} zum Zeitpunkt des letzten Berechnungsschrittes $t_{MOD\_i-1} = t_{MOD\_i} - dt_{MOD}$ erreicht, damit auch die Ladungsänderung von $I_{MOD}(t_{MOD\_i-1})$ zu $I_{MOD}(t_{MOD\_i})$ der Ladung q_i entspricht:

$$dt_{MOD} = \frac{2 \cdot q_i}{I_{OUT}(t_{i-1}) + I_{GOB}(V_{i-1}) + I_{OUT}(t_i) + I_{OUT}(V_i)} \tag{82}$$

Somit wurde das Wertepaar $(t_i; I_{OUT}(t_i))$ der Originalkurve zum Wertepaar $(t_{MOD\_i}; I_{MOD}(t_i))$, welches neben der Originalkurve die vorhandenen Defekte je nach Simulationszeit und Gattersituation berücksichtigt. Dann kann zum nächsten Zeitpunkt $t_{i+1} = t_i + dt$ gewechselt werden. Am Ende der Berechnungsschritte ergeben alle Wertepaare die neue Funktion $I_{MOD}(t)$.

Sollte der Anfangspegel ungleich V_{DD} (fallende Flanke) oder V_{SS} (steigende Flanke) sein, so wird erst mit der Modifikation der Stromkurve begonnen, wenn V_i den Anfangswert erreicht hat. Trotzdem wird t_0 als Startwert für die modifizierte Kurve übernommen, da der Umladevorgang schon beim Anfangspegel beginnt.

Dies ist auch ein Grund, warum die modifizierte Kurve noch einmal angepasst werden muss. Die anderen Gründe sind der Spannungspegel zum Ende des Umladevorganges und dass Analysetools generell von maximalen Spannungspegeldifferenzen von $V_{DD} - V_{SS}$ ausgehen und mit nur wenigen Wertepaaren die Spannungskurve berechnen. Daher kommt es zu der in Punkt 5 aufgezählten Anpassung der modifizierten Stromkurve $I_{MOD}(t)$. Für die steigende Flanke wird die Stromkurve folgendermaßen aufgeteilt (Abbildung 5-7): Das Wertepaar, an dem der Spannungspegel $V_{OUT} = (V_{DD} - V_{SS})/10$ erreicht, wird als Referenz ausgesucht. Sollte V_{MIN} größer sein als dieser Wert, wird das erste Wertepaar das Referenzpaar. Die Differenz dieses Startpegels zu V_{SS} wird in ein weiteres Wertepaar umgerechnet, welches das erste Wertepaar der CCS-Stromkurve darstellt. Dann folgt das Referenzpaar. Weitere Paare folgen im Abstand von $dV = (V_{DD} - V_{SS})/10$. Zusätzlich wird das Wertepaar in die CCS-Stromkurve übernommen, das den maximalen Absolutwert von $I_{MOD}(t)$ enthält, damit dieser durch die Interpolation der CCS-Berechnung nicht verloren geht. Sollte aufgrund eines verringerten Endpegels die resultierende Spannungskurve nicht V_{DD} erreichen, wird ein weiteres Wertepaar hinzugefügt, damit keine Warnungen oder Fehlermeldungen durch das Analysetool generiert werden. Dementsprechend wird mit der fallende Flanke verfahren, wobei bei V_{DD} angefangen wird. Schließlich sollten am Ende der Berechnung der CCS-Stromkurven für jedes Gatter, Pin, Flankenwechsel und entsprechender Parameterpaare aus Eingangsflanke und Lastkapazität jeweils eine CCS-Stromkurve entstanden sein, die die Defektsituation des Gatters berücksichtigt und den geforderten Gegebenheiten des Analysetools entspricht.

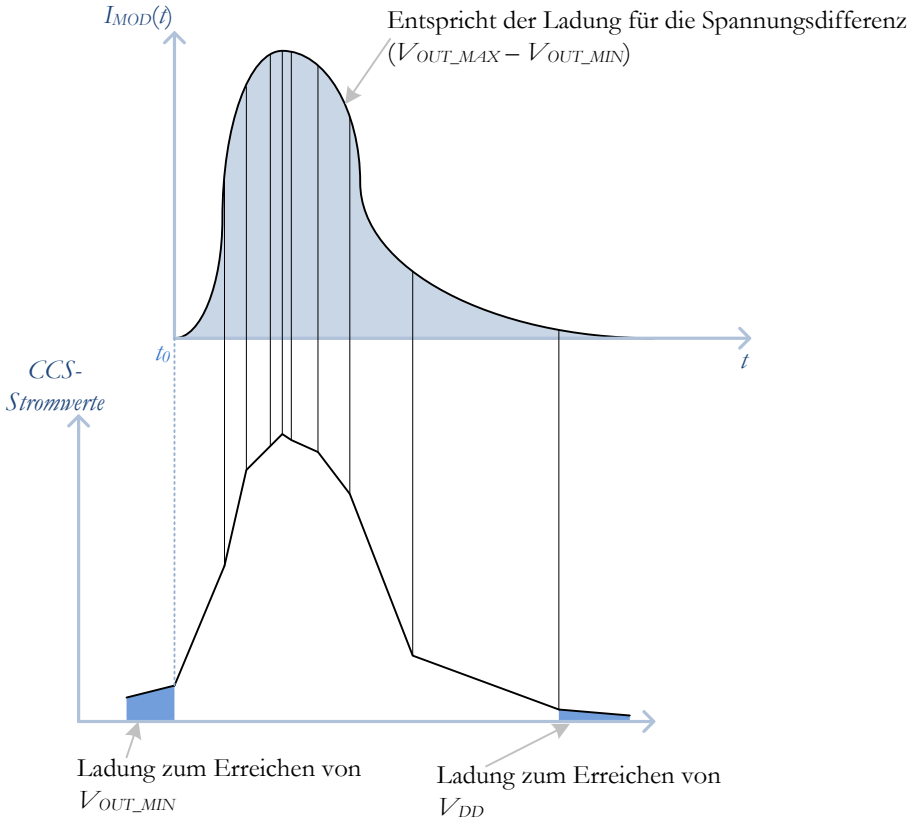


Abbildung 5-7 : Umwandlung der Stromkurve $I_{MOD}(t)$ für einen ansteigenden Ausgangspegel in CCS-Stromwert-Zeitpaare mit Berücksichtigung von V_{OUT_MIN} und V_{OUT_MAX} durch Erhöhung des Startwertes bei t_0 und Einfügen eines Wertepaares vor t_0, damit das CCS-Analysetool bei t_0 den korrekten Spannungspegel V_{OUT_MIN} errechnet, sowie ein zusätzliches Wertepaar am Ende damit trotz eines final niedrigeren Ausgangspegel V_{DD} erreicht wird

5.4.3 Erstellung der Gatterbibliothek

Um nun für jeden Zeitschritt eine Analyse mit den proprietären Tools vorzunehmen, wird eine Standardzellbibliothek erstellt, die sowohl die nicht defekten Gatter enthält, als auch Gatter, die selbst defekt sind bzw. deren Ausgang mit einem defekten Gatter verbunden ist, damit sich eventuelle Änderungen der Flankenwechsel auch auf nachfolgende Gatter auswirken. Zudem wird dadurch natürlich auch die Verzögerungszeit des Gatters an sich modifiziert. Um diese Modifizierungen im Design zur Geltung zu bringen, wird die Netzliste angepasst, um die modifizierten Gatter von den Standardgattern zu unterscheiden. Die Bibliothek wird dann mit den „neuen“ Gattertypen erweitert. Hierbei werden die Werte der Standardgatter übernommen, bis auf die CCS-Stromkurven, die im vorigen Kapitel neu berechnet wurden und bis auf die Leckstromparameter, die mittels berechneter Werte der Spannungspegelanalyse, $I_{GOB}(V_{OUT\_MIN})$ und $I_{GOB}(V_{OUT\_MAX})$ für jeden defekten Eingangspin, auch angepasst werden.

5.5 Vergleich des GOB-Simulators mit Spice-Simulationen

Um die Berechnungen der Defektanalyse zu testen, wurden Spice-Simulationen ausgewählter Testdesigns gegen die Berechnungen der Spannungspegelanalyse und der statischen Zeitanalyse verglichen. Der Aufbau der Tests, der Designs und die Ergebnisse werden in den nächsten Abschnitten vorgestellt.

5.5.1 Aufbau der Testdesigns

Es wurden elf verschiedene Designs entworfen, damit alle Gattertypen getestet werden konnten. Sie erreichten nicht die Komplexität der ISCAS-Design, da die Testschaltungen so entworfen wurden, dass bei Gattern mit mehreren Pins nur ein Pin schaltete, damit die Single-Input-Switching-Berechnungen der STA Berücksichtigung fanden. Das erste Design ist eine zehnstufige Inverterkette (Test_1), während die Tests 2 bis 5 jeweils Inverter gleicher Treiberstärke enthalten, die in einem achtstufigen Baum angeordnet wurden, wobei Test 5 die größte Treiberstufe beinhaltet und Test 2 die kleinste. Des Weiteren wurden NAND2- und

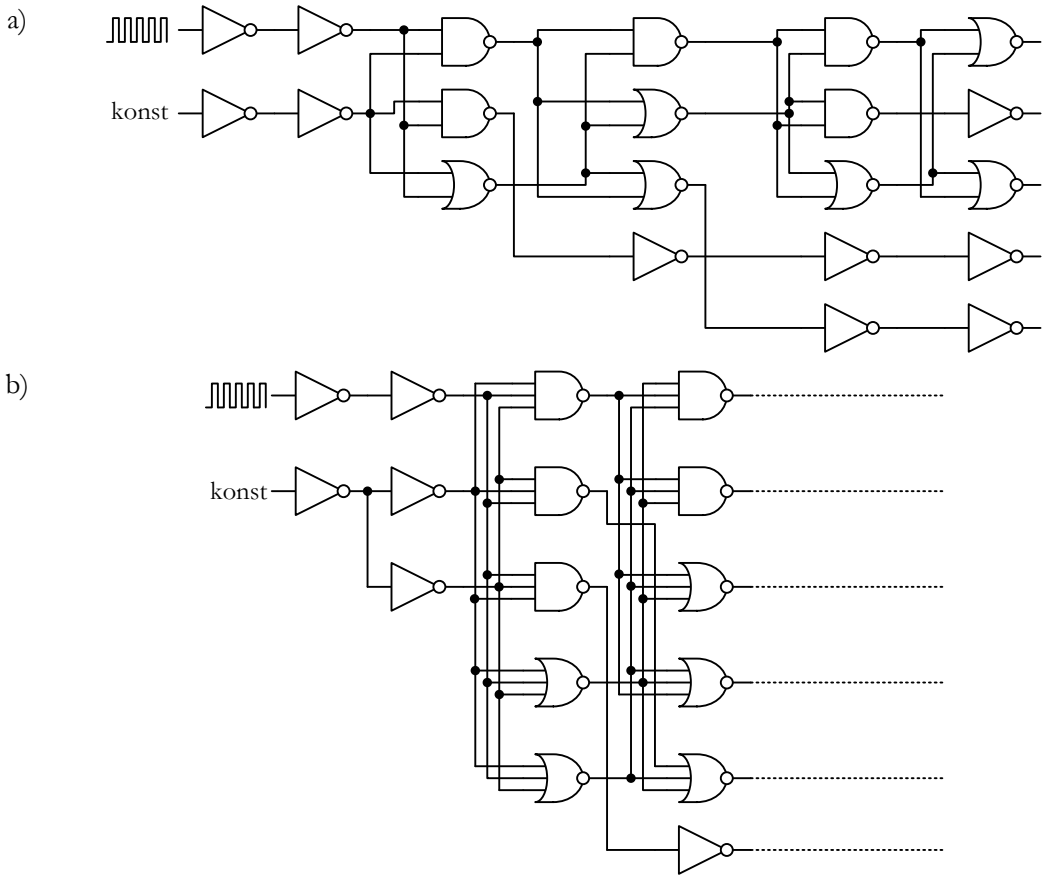


Abbildung 5-8 : Test-Designs:
a) NAND2-, NOR2-Gatter und Inverter in einem sechsstufigen Baum
b) Anfang des achtstufigen Baumes mit NAND3-, NOR3-Gattern und Invertern

NOR2-Gatter in den Tests 6 bis 8 mit Invertern in einem sechsstufigen Baum zusammengefasst – Abbildung 5-8 a), wobei die Treiberstufen mit der Designnummer ansteigen. Den Abschluss bildeten NAND3- und NOR3-Gatter, die zusammen mit Invertern jeweils in einem achtstufigen Baum simuliert wurden (Tests 9 bis 11) – Abbildung 5-8 b). Auch sind die Gatter des Tests 11 die treiberstärksten. Bei den Tests 6 bis 11 wurden teilweise Schaltungseingänge konstant gehalten, während andere ständig schalteten, damit die Ausgänge aller Gatter immer schalteten, während nur ein Eingang schaltete.

5.5.2 Ergebnisse des Vergleich der Spice-Simulationen mit der SPA

Abbildung 5-9 a) und b) zeigen die Ergebnisse für den Vergleich der mit dem GOB-Simulator errechneten Minimal-/Maximalpegel in Bezug zu den mit Spice simulierten Ergebnissen. Dargestellt sind die Ergebnisse, die exemplarisch für alle Testdesigns 1 bis 5 anzusehen sind. Um die Vielzahl der Simulationsläufe gut gegeneinander zu vergleichen, wurden alle Abweichungen in Streuungsquartilen zusammengefasst, einmal für „alle“ Simulationen über die Gesamtheit der Zeitpunkte und dann jeweils für bestimmte Zeitschritte, wobei ein weibullverteilter Verschleiß angenommen wurde, so dass mit größer werdenden Zeitschritten die Anzahl und das Ausmaß der Gateoxiddefekte zunimmt. Der Median befindet sich durchgängig bei 0 ±0.01. Zudem ist ersichtlich, dass ein Großteil der errechneten Pegel nur minimal von den simulierten abweicht. Zum Beispiel ist in Abbildung 5-9 a) zu sehen, dass 50 % der Werte (Interquartilsabstand IQR) zwischen $Q_{.25}$ (unteres Ende des grauen Balkens) und $Q_{.75}$ (oberes Ende des schwarzen Balkens) weniger als 1 % von den Spice-simulierten Werten abweichen, genauso wie bei den

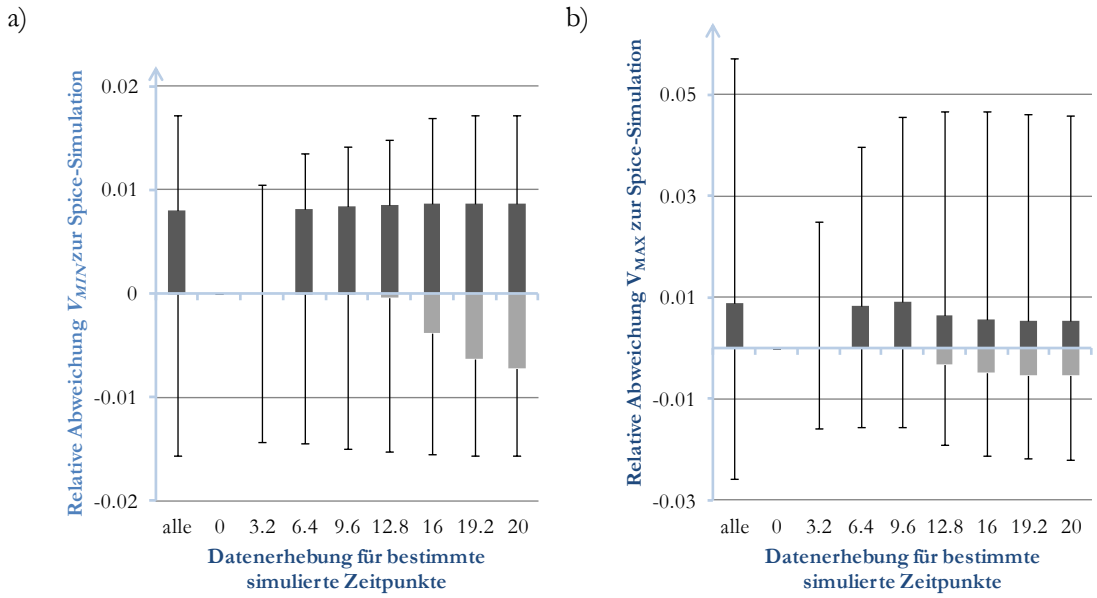


Abbildung 5-9 : Ergebnisse der Spannungspegelanalyse im Vergleich zu Spice-Simulationen:

a) Minimalpegelanalyse des Testdesigns 5
b) Maximalpegelanalyse des Testdesigns 3

Maximalwerten. Die Extremwerte liegen bei Abbildung 5-9 a) im Bereich von ±0.02. Beim Testdesign 3 liegen die Extremwerte zwischen -3 % und +6 %. Diese vergrößerte Extremwertbereich ist vor allem bei den kleineren Treiberstufen zu erkennen. Dies ist darauf zurückzuführen, dass die Inverter mit geringen Treiberstufen im Fall eines eigenen Defektes sehr langsame Flanken erzeugen. Diese wurden von den grundlegenden Spice-Simulationen nicht behandelt, so dass diese Extremwerte durch Extrapolationen über die Datenbankgrenzen größer ausfallen.

Bei den Testdesigns 6 bis 11 sind ähnliche Ergebnisse erzielt worden, wie es in den Abbildungen 5-11 a) bis d) dargestellt ist. Auch hier erzielen die Designs mit größeren Treiberstufen bessere Ergebnisse. Bei den Maximalpegeln befindet sich der IQR zwischen -1 % und +1.5 % bei den Tests 6 bis 8. Bei den Tests 9 bis 11 liegt die untere Grenze des IQR bei -1.5 %. Die Minimalpegelgrenzen für das IQR der Tests 6 bis 11 liegen bei -2 % und +2 %. Mit der Ausnahme bei Test 6, wo die untere Grenze bei $t = 20$ -3.7 % ist, was ebenso durch die geringen Treiberstufen in diesem Design erklärbar ist. Die sehr starken Maximalabweichungen von +30 % bis -30 % dieser Designs, was exemplarisch für Test 8 in Abbildung 5-11 e) zu sehen ist, entstehen zum größten Teil aufgrund der Tatsache, dass der GOB-Simulator bei Überschreiten eines bestimmten Limits den Wert von $(V_{DD} - V_{SS})/2$ annimmt, obwohl der Spice-Simulator den Pegel noch gerade so innerhalb der Grenze errechnet. Dies ist allerdings in sehr seltenen Fällen geschehen, was aus Abbildung 5-10 ersichtlich wird, wo die durchschnittlichen Abweichungen aller Designs für alle Simulationen zum Zeitpunkt $t = 9.6$ und $t = 16$ als Beispiel dargestellt sind. Die Abweichungen liegen im Bereich zwischen -2 % und +1 %. Nur bei zwei Designs beim Zeitschritt $t = 16$ werden größeren Abweichungen erzielt, so dass man festhalten kann, dass die Spannungspegel sehr exakt vom GOB-Simulator errechnet werden.

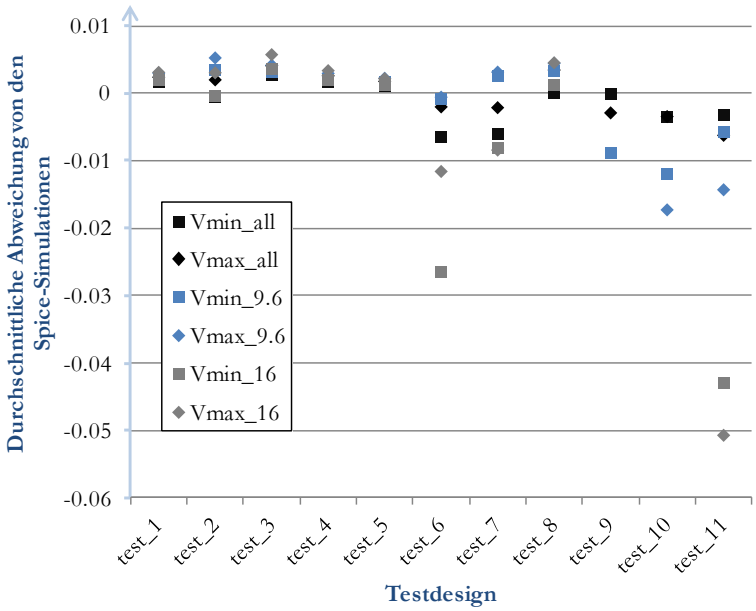


Abbildung 5-10 : Durchschnittliche Abweichungen der Spannungspegelanalyse von den Spice-Simulationen der Testdesigns (normiert auf 1)

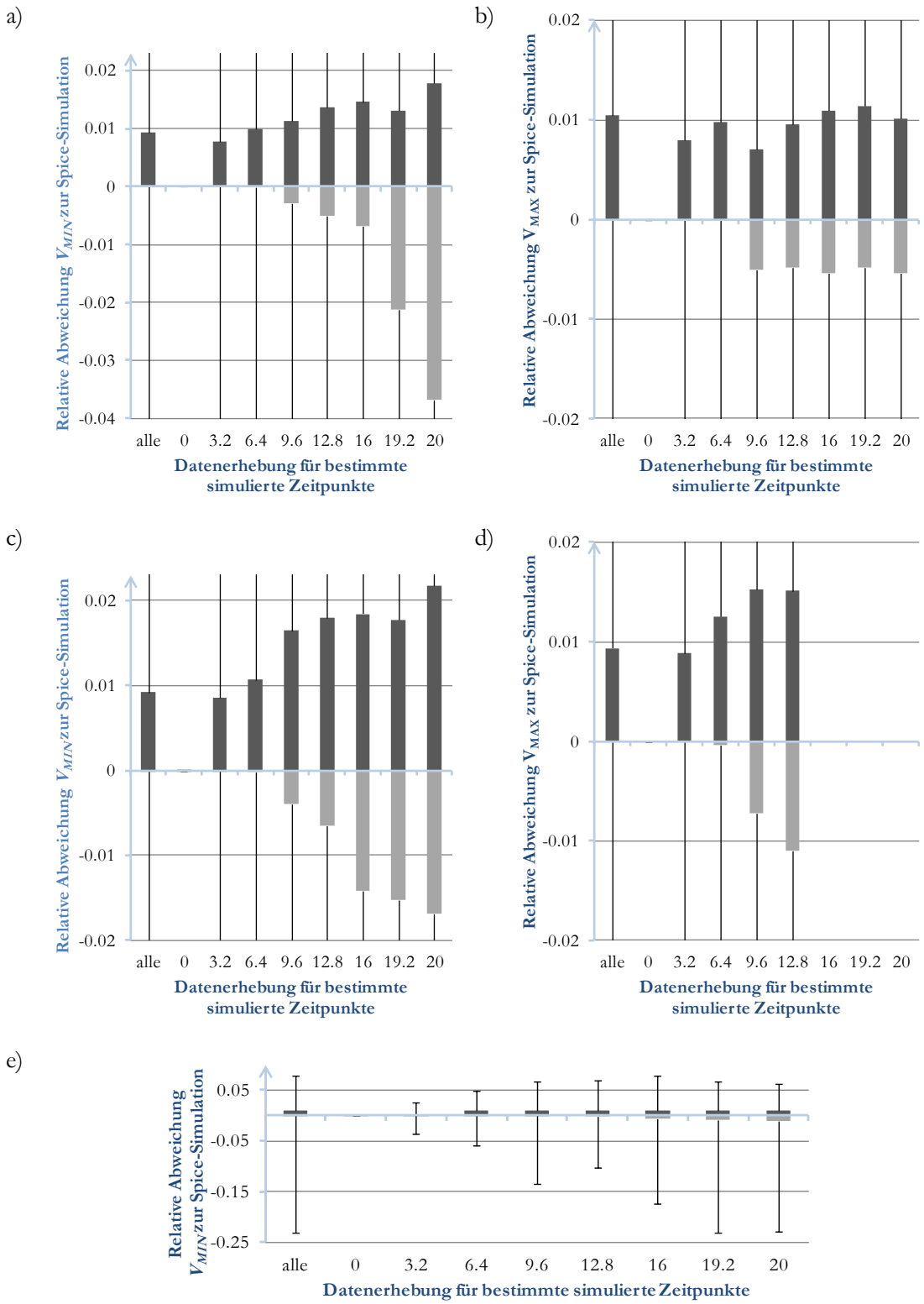


Abbildung 5-11 : Ergebnisse der SPA im Vergleich zu Spice-Simulationen:

a) Interquartilsabstand der Minimalpegelanalyse des Testdesigns 6
b) Interquartilsabstand der Maximalpegelanalyse des Testdesigns 7
c) Interquartilsabstand der Minimalpegelanalyse des Testdesigns 11
d) Interquartilsabstand der Maximalpegelanalyse des Testdesigns 10
e) Minimalpegelanalyse des Testdesigns 8

5.5.3 Vergleich der Spice-Simulationen mit der STA der modifizierten Gatterbibliothek

Ergebnisse zum Zeitverhalten sind in Abbildung 5-12 dargestellt. Hier werden die Durchschnittswerte für spezifische Zeitpunkte über alle Simulationsläufe des Designs test_2 und test_8 aufgezeigt. Zu sehen ist die immer vorhandene Abweichung der STA (mit Synopsys™ DesignCompiler™ errechnet) von den Spice-Simulationen, da die Tools vom schlechtesten Fall ausgehen und das Schalten der nächsten Stufe später als mit Spice erfolgt, wo schon mit Überschreiten der Schwellspannung am Eingang der nächsten Stufe das Schalten dieser Stufe beginnt. Allerdings ist ebenso gut zu erkennen, dass die Verzögerungszeit der Gattersimulationen dem Verlauf der sich verschlechternden Verzögerungszeitkurve der Spice-Simulationen folgt, was gut den Verschleiß der Schaltung widerspiegelt.

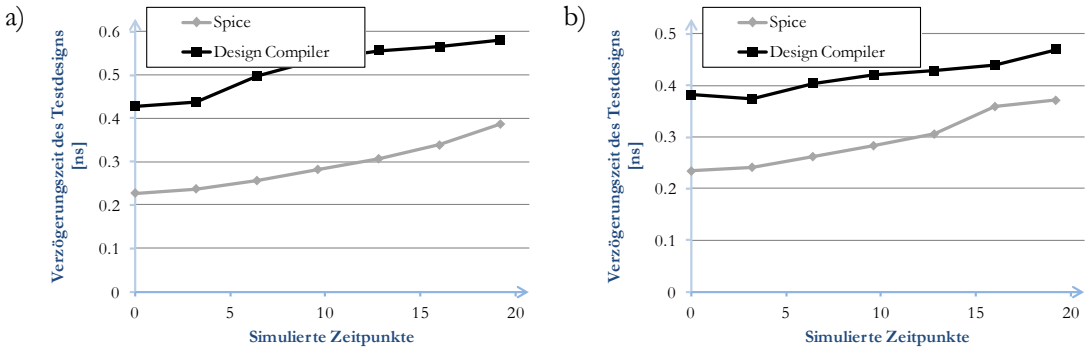


Abbildung 5-12 : Ergebnisse der STA mit Synopsys Design Compiler im Vergleich zu Spice-Simulationen:

a) Testdesigns 2
b) Testdesigns 8

5.6 Statistische Untersuchungen und Schwachstellenanalyse mit Hilfe mehrerer Simulationsläufe

Da sich der probabilistische Einfluss durch die Berechnung der Defekteintrittszeit t_{GOB} auf die Simulation auswirkt, wird ein Durchlauf zur Analyse potentieller Schwachpunkte in einer Schaltung nicht ausreichen. Aus diesem Grund empfiehlt es sich, eine gewisse Anzahl von Durchläufen durchzuführen, damit Ergebnisse ähnlich dem Monte-Carlo-Verfahren für die Parameteranalyse zu erreichen sind.

Wenn mehrere Durchläufe der STA zu eine Art statistische Zeitanalyse (engl. statistical static timing analysis – SSTA) zusammenfasst werden, kann für die untersuchten Zeitpunkte eine kumulative Verteilung des kritischen Pfades generiert werden, mit der abgeschätzt werden kann, wie viele der Schaltungen zu unterschiedlichen Zeitpunkten noch für eine bestimmte Taktfrequenz geeignet sind. Ferner eignet sich diese Form zur Darstellung einer Entwicklung der

Schaltung während der Verschleißphase, wie es in Abbildung 5-13 zu sehen ist. Hier sind vier der Testdesigns mit ihren kritischen Pfaden als kumulierte Verteilung abgebildet.

Zum einen kann man erkennen, wie die Streuung der Verzögerungszeitwerte des kritischen Pfades mit zunehmender simulierter Betriebsdauer ansteigt, so dass ein größer werdender Anteil der Schaltungen bestimmte Grenzen der Verzögerungszeit nicht mehr einhalten kann, während der Verschleiß voranschreitet. Zum anderen sieht man beim Vergleich der Testdesigns 7 und 8, die sich nur durch schwächere Treiber für 7 unterscheiden, dass aufgrund der schwächeren Treiber die Streuung ebenfalls zunimmt, da beispielsweise Gateoxiddefekte mit kleinerem Ausmaß nicht mehr so gut kompensiert werden können wie beim Design 8. Ein weiteres Detail ist beispielsweise in Abbildung 5-13 b) zu sehen, wo wenige Designs, die obwohl der Verschleiß schon weiter fortgeschritten war, schneller waren als Schaltungen, mit geringerer Betriebsdauer, da Gateoxiddefekte in wenigen Fällen auch den kritischen Pfad beschleunigen können, wenn z. B. nur eine Schaltrichtung eines defekten Gatters für den kritischen Pfad von Bedeutung ist.

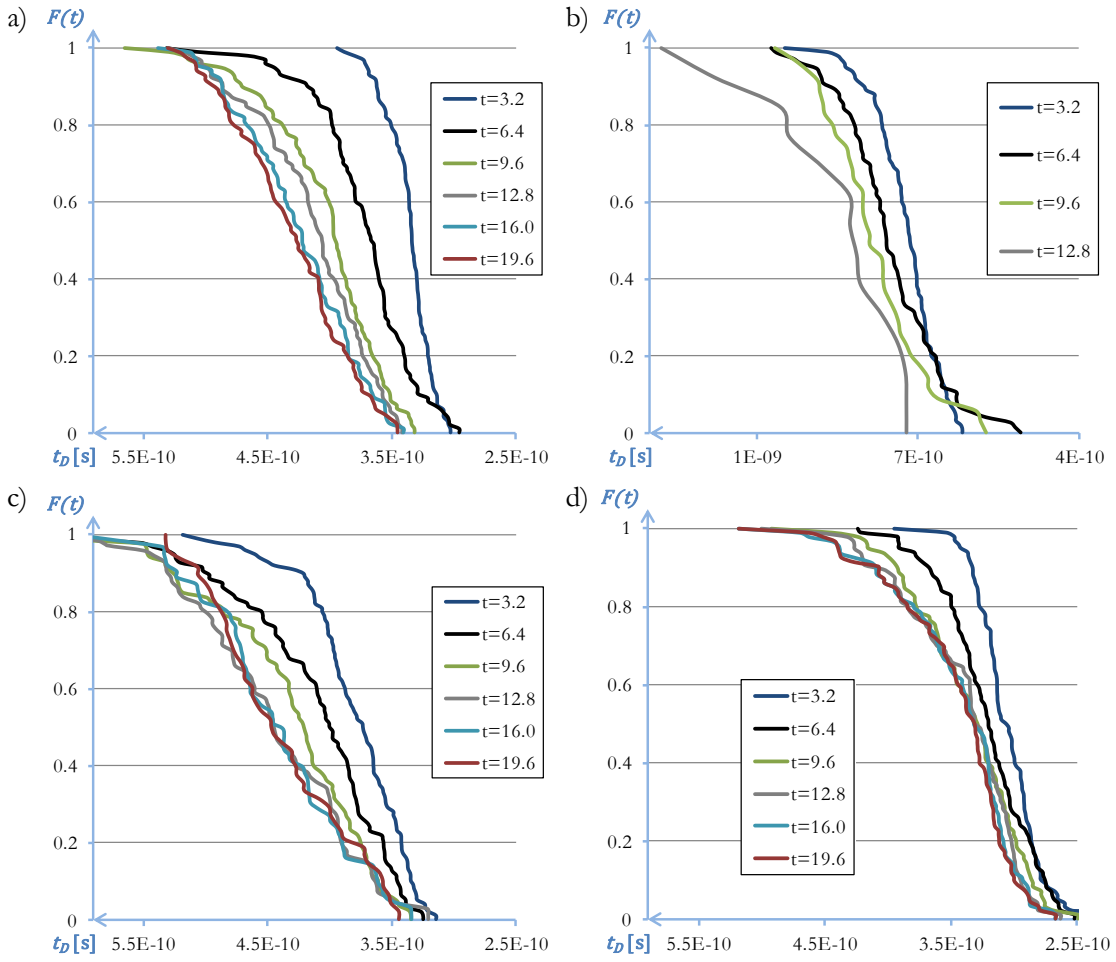


Abbildung 5-13 : Verlauf der Verzögerungszeit des kritischen Pfades als kumulative Verteilungsfunktion über die sinkende Verzögerungszeit:
a) Testdesign 3 b) Testdesign 11 c) Testdesign 7 d) Testdesign 8

5.7 Fazit: GOB-Simulator

Hauptziel des GOB-Simulators ist die Einbeziehung von Verschleißerscheinungen in den Designflow zur Gattersynthese, wie er in Abbildung 4-25 erweitert vorgestellt wurde. Wie gezeigt, werden Simulationen durchgeführt, um zu testen, ob die Schaltungen noch den Ansprüchen genügend großer Signalpegelabstände erfüllen. Hinzu kommt die Modifizierung der Gatternetzliste und -bibliothek. Damit ist es dann möglich, die Verzögerungszeit des Designs für verschiedene Zeitpunkte des Verschleißes nachzuvollziehen. In Abbildung 5-14 ist ersichtlich, dass der GOB-Simulator (dunkelblau unterlegte Komponenten) sich nach der erstmaligen Synthese des Designs in den Designablauf einfügt, also wenn Netzliste und Aktivitätsdatei erstellt worden sind. Die mehrmalige Durchführung eines GOB-Simulator-Durchlaufes kann genutzt werden, um eine statistische Analyse des Design zu generieren, und um eine Schwachstellenanalyse durchzuführen, die potentielle Probleme einer Schaltung aufzuzeigen kann, da sie die Gatter markiert, die am ehesten an Verschleißerscheinungen leiden werden. Dies kann dann der Verbesserung der Schaltung dienen, wie beispielsweise dem Einfügen von redundanten Transistorstacks aus Kapitel 4. Dieser Ansatz eignet sich gut zur Kombination mit dem GOB-Simulator, da steigenden bzw. fallenden Flanken differenziert betrachtet werden können. Beispielsweise könnte eine Schwachstellenanalyse ermitteln, dass nur eine der beiden Flanken eines gefährdeten Gatters den kritischen Pfad des Designs nachhaltig vergrößern könnte, so dass nur der betreffende Stack verdoppelt wird.

Für die Anwendung des GOB-Simulators bedarf es einer vorhandenen Standardgatterbibliothek, einer Netzliste, probabilistischer Parameter, wie Weibullfaktor und/oder Verteilungsart, und wenn möglich einer SAIF-Aktivitätsdatei, da diese Dateien Informationen über den logischen Status (0, 1, X) einzelner Netze über den gesamten Simulationslauf erfassen und damit für eine genauere Vorhersage der Ausfallzeitpunkte t_{GOB} benutzt werden. Auf Basis der originalen Bibliothek und Netzliste und der Aktivitätsdatei werden für einen definierten Zeitraum der Verschleißphase Spannungspegelberichte und CCS-Bibliotheken generiert. Aufgrund dieser Ergebnisse kann die Einhaltung logischer Spannungspegel kontrolliert und das statische Zeitverhalten analysiert werden.

Vergleiche mit akkuraten Spice-Simulationen zeigten eine genaue Nachbildung der Spannungspegel innerhalb eines Designs, sowie eine gute Vorhersage der Schaltungsverzögerungszeit für die STA auf Gatterebene. Daraus ist zu folgern, dass Verschleißerscheinungen, die auf die zu untersuchende Schaltung wirken, gut auf Gatterebene simuliert werden können.

Verbesserungspotential liegt noch vor allem bei der Modifizierung der Stromkurve, da dieser Teil aufgrund der Behandlung aller Designpins verhältnismäßig lange dauert. So können interne Strukturen des Simulators schneller programmiert werden, beispielsweise um Datenbankzugriffe zu optimieren. Andererseits liegt eine Art Vorauswahl derjenigen Gatter nahe, die am ehesten für Gateoxiddefekte in Frage kommen, um die anderen für die Analyse nicht zu modifizieren, so dass zwar ein Teil der Genauigkeit verloren gehen kann, die Berechnungszeit allerdings gesteigert werden kann. Eine andere Methode wäre es, die Analyse in großen Zeitschritten durchzuführen

und bei Eintreten eines Schaltungsfehlers kleinere Zeitabstände vor dem Fehlereintritt zu simulieren. Trotz der vorhandenen Verbesserungsmöglichkeiten ergibt sich ein Geschwindigkeitsvorteil gegenüber Spice-Simulationen, da die Spannungspegelanalyse und die Modifizierung der Gatternetzliste und -bibliothek mit C++ berechnet werden und die nachfolgende STA auf Gatterebene durchgeführt wird. Ferner kann auf die Daten voriger Defektzeitpunkte zurückgegriffen werden. Dem gegenüber stehen Spice-Simulationen, die für jeden Defektzeitpunkt wieder in Gänze neu gestartet werden. Zudem steht dann noch die Auswertung der Analysen an.

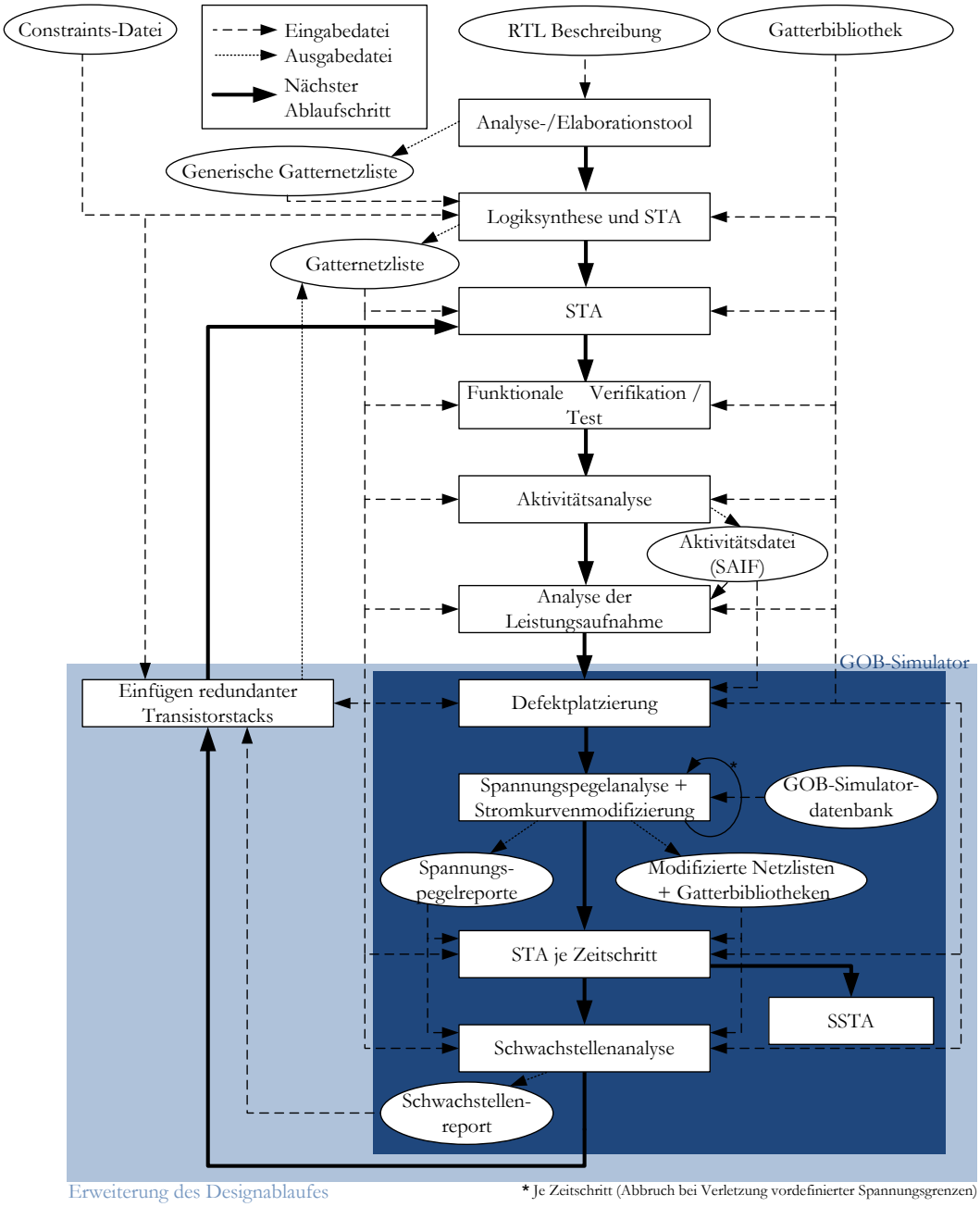


Abbildung 5-14 : Erweiterter CMOS-Designflow mit GOB-Simulator und dem Einfügen redundanter Transistorstacks

Sechstes Kapitel

6 Zusammenfassung und Ausblick

Die fortschreitende Skalierung integrierter CMOS-Schaltungen führt zu verkleinerten Strukturgrößen und schnelleren Schaltungen, allerdings auch zu einer erhöhten Leistungsaufnahme und einer verminderten Zuverlässigkeit. Vor allem Verschleißerscheinungen, wie das Gateoxidbreakdown, vermindern immer früher und in einem gesteigerten Maß die Lebensdauer und die Funktionalität integrierter Schaltungen, da sie die Eigenschaften der grundlegenden Bauelemente, der Transistoren, mit der Zeit immer weiter verschlechtern. Aus diesem Grund sind Techniken und Werkzeuge im heutigen Schaltungsdesign unumgänglich, die die Zuverlässigkeit der Schaltungen in höheren Designebenen zumindest berücksichtigen und gegebenenfalls verbessern. In der vorliegenden Arbeit werden bekannte Ansätze der Zuverlässigkeitssteigerung in Form redundanter Systeme auf die Verringerung des Verschleißes fokussiert und neue Ansätze entwickelt, um Gateoxiddefekten auf Gatterebene entgegen zu treten.

Dabei wird auch besonders die Leistungsaufnahme der Schaltungen berücksichtigt, da sie sich mittelbar durch ihre Auswirkung auf die Temperatur negativ auf die Zuverlässigkeit auswirkt. So wird im ersten Teil der bekannte Triple-Modular-Redundancy-Ansatz derart verändert, dass Leistungseinsparungen von bis zu 60 % für größere Designs möglich sind, bei gleichzeitiger Steigerung der Zuverlässigkeit um ein Viertel gegenüber dem konventionellem TMR bei einem Flächenmehrbedarf von 4 % und einer um 12 % gesteigerten Verzögerungszeit.

Weiterhin wurden redundante Gatter entwickelt, die die Zuverlässigkeit verglichen mit der Basisschaltung steigern können. Weil die redundanten Elemente erst im späteren Verlauf der Betriebszeit bei Bedarf zuschalten werden können, ermöglichen sie trotz bis zu 60 % Flächenzuwachs eine problemlosere Integration in die Basisschaltungen, da die Stromaufnahme der simulierten Designs nur um maximal 12 % steigt und die Verzögerungszeit zwischen 2 und 14 % zunimmt. Unterschiedliche Strategien des Redundanzumfangs wurden simuliert und gegen Ansätze mit redundanten Transistoren (DTrans) bzw. Gattern (DGatter) verglichen, die von

Betriebsbeginn an mit den Originaltransistoren schalten. Der Vergleich zeigte trotz pessimistischer Annahmen für die in dieser Arbeit entwickelten Zuverlässigkeitsstrategien bessere Ergebnisse hinsichtlich der Zuverlässigkeitssteigerung und auch der erhöhten Leistungsaufnahme. Da die redundanten Transistorstacks des vorgestellten Ansatzes während der normalen Betriebszeit nicht an den Umschaltvorgängen im Design teilnehmen und die Gateoxide der redundanten Transistoren nicht dem Stress der anliegenden elektrischen Felder ausgesetzt werden, unterliegen sie auch nicht dem Verschleiß der Originaltransistoren, wie es auch beim DTrans- bzw. DGatter-Ansatz der Fall ist. Wenn sie nun zu einem Zeitpunkt zugeschaltet werden, an dem Verschleiß schon eingesetzt hat, erhöhen sie die Zuverlässigkeit der Basisschaltungen um bis zu 65 %, während der DTrans-Ansatz die Zuverlässigkeit höchstens um 34 % steigern kann, da auch schon redundante Transistoren verschlissen sein können. Nachteilig wirkt sich neben dem Ansatz auf Transistorebene die erhöhten Leistungsaufnahme von 50 % im defektfreiem Zustand der Schaltung für DTrans aus, da dies zu erhöhten Temperaturen und somit zu mehr Verschleiß führt.

Das Reizvolle an dem vorgestellten Ansatz ist neben der flexiblen Wahl der Einfügestrategie die Auswahl und das Einfügen auf Gatterebene, welches eine Designebene ist, in der noch wenig Ansätze zur Zuverlässigkeitssteigerung übernommen wurden. Die redundanten Stacks können auf Layoutebene erzeugt und die Parameter ihres Schaltverhalten leicht in die Gatterebene überführt werden, so dass vorhandene Software diese Elemente mit wenigen Umstellungen als normale Gatter in den Syntheseprozess einbinden könnten.

Da in diesem Zug gezeigt werden konnte, dass die gezielte Einschleusung von redundanten Elementen ein gewinnbringender Kompromiss zwischen Fläche und Zuverlässigkeit darstellt, wurde im zweiten Teil dieser Arbeit ein Simulator entwickelt, der die Zuverlässigkeit integrierter Schaltungen hinsichtlich Gateoxiddefekten auf Gatterebene untersucht. Es wurde ein Softwaretool ausgearbeitet, um die Auswirkungen von Gateoxiddefekten auf eine vorhandene Schaltung zu verschiedenen Zeitpunkten im Betrieb zu simulieren. Dazu werden zufällig nach Vorgabe von realen Defektverläufen Gateoxiddefekte eingefügt. Anschließend werden Spannungspegel im Design geprüft und Gatterbibliotheken für den jeweiligen Betriebszeitpunkt erstellt. Dadurch ist es möglich, einerseits zu simulieren, ob das Design noch überall korrekte Pegel an den Gatterausgängen erzeugt, und andererseits Auswirkungen auf die Verzögerungszeit und die Leistungsaufnahme auf Gatterebene aufzuzeigen, so dass während des Syntheseschrittes Zuverlässigkeitsaspekte berücksichtigt werden können.

In Verbindung mit den redundanten Transistorstacks kann ein Design bei Berücksichtigung des probabilistischen Verhaltens gezielt auf Schwachstellen hinsichtlich Verschleißeffekte analysiert werden, um so im Anschluss die Performance gezielt eingefügter Redundanz steigern zu können.

Um die CMOS-Gattersynthese zu unterstützen wurde bei allen entwickelten Ansätzen darauf geachtet, dass immer eine Integration in den Designablauf gegeben ist.

Für weiterführende Arbeiten bietet sich eine Erweiterung des Simulators für andere Verschleißeffekte, wie Hot Carrier und NBTI an, da die Mechanismen ähnlich auf die

Performance der Transistoren einwirken. Zudem kann die Berechnungszeit des Simulators durch geeignete Algorithmen verbessert werden. Bei den entwickelten zuverlässigkeitssteigernden Gattern auf Designebene bietet sich eine Untersuchung hinsichtlich der Verwendung unterschiedlicher Gateoxiddicken an, da diese mittlerweile schon standardmäßig im Design verwendet werden können. Weiterhin eignet sich der vorgestellte Ansatz, um die Ausbeute integrierter Schaltungen zu erhöhen. Analysen in dieser Richtung und der Vergleich mit realen Schaltungen versprechen gute Ergebnisse.

Literaturverzeichnis

[Ala00] Alam, M.A., Bude, J., Ghetti, A., Field acceleration for oxide breakdown-can an accurate anode hole injection model resolve the E vs. 1/E controversy?, In *Proceedings of the 38th Annual IEEE International Reliability Physics Symposium*, S. 21-26, San Jose, USA, 2000

[Ala02] Alam, M.A., Weir, B.E., Silverman, P.J., A study of soft and hard breakdown - Part I: Analysis of statistical percolation conductance, In *IEEE Transactions on Electron Devices*, Vol. 49, Nr. 2, Nr. 2, S. 232-238, 2002

[Ala05] Alam, S.M., Wei, F.L., Gan, C.L., Thompson, C.V., Troxel, D.E., Electromigration reliability comparison of Cu and Al interconnects, In *Proceedings of the 6th International Symposium on Quality of Electronic Design* (ISQED), S. 303-308, San Jose, USA, 2005

[Bla69] Black, J.R., Electromigration failure modes in aluminum metallization for semiconductor devices, In *Proceedings of the IEEE*, Vol. 57, Nr. 9, S. 1587-1594, 1969

[Bro01] Brooks, D., Martonosi, M., Dynamic thermal management for high-performance microprocessors, In *Proceedings of the 7th International Symposium on High-Performance Computer Architecture* (HPCA), S. 171-182, Los Alamitos, USA, 2001

[Bud98] Bude, D.J., Weir, B.E., Silverman, P.J., Explanation of stress-induced damage in thin oxides, In *Technical Digest of International Electron Devices Meeting* (IEDM), S. 179-182,San Francisco, USA, 1998

[Cad03] Cadence™, *Reliability Simulation in Integrated Circuit Design*, Techn. white paper, 2003

[Cat67] Catt, I., Crosstalk (Noise) in Digital Systems, In *IEEE Transactions on Electronic Computers*, Vol. EC-16, Nr. 6, S. 743-763, 1967

[Cha92] Chandrakasan, A.P., Sheng, S., Brodersen, R.W., Low-power CMOS digital design, In *IEEE Journal of Solid-State Circuits*, Vol. 27, Nr. 4, S. 473-484, 1992

[Che86] Chen, I.-C., Holland, S., Young, K.K., Chang, C., Hu, C., Substrate hole current and oxide breakdown, In *Applied Physics Letters* , Vol. 49, Nr. 11, S. 669-671, 1986

[Che95] Chen Z., Koren I., Techniques for Yield Enhancement of VLSI Adders, In *Proceedings of the IEEE International Conference on Application Specific Array Processors* (ASAP), Los Alamitos, USA, 1995

[Chi95] Chiluvuri, V.K.R.; Koren, I.; , "Layout-synthesis techniques for yield enhancement," In *IEEE Transactions on Semiconductor Manufacturing*, Vol. 8, Nr. 2, S. 178-187, 1995

[Cor08] Cornelius, C., Sill, F., Sämrow, H., Salzmann, J., Timmermann, D., da Silva, D., Encountering gate oxide breakdown with shadow transistors to increase reliability, In *Proceedings of the 21st Annual symposium on Integrated circuits and system design* (SBCCI), Gramado, Brasilien, 2008

[Cor08] Cornelius, C., Sämrow, H., Timmermann, D.,. Router layout for reduced area costs in networks-on-chip. In *Proceedings 13. Symposium Maritime Elektrotechnik*, S. 125-130, Rostock, Deutschland, 2010

[Cot79] Cottrell, P.E., Troutman, R.R., Ning, T.H., Hot-electron emission in n-channel IGFETs, In *IEEE Journal of Solid-State Circuits*, Vol. 14, Nr. 2, S. 442- 455, 1979

[Cro79] Crook, D. L., Method of Determining Reliability Screens for Time Dependent Dielectric Breakdown, In *Proceedings of 17th Annual Symposium on Reliability of Physics*, S. 1-7, San Francisco, USA, 1979

[Dat06] Datta, R., Abraham, J.A., Diril, A.U., Chatterjee, A., Nowka, K., Adaptive Design for Performance-Optimized Robustness, In *Proceedings of 21st IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems* (DFT), S. 3-11, Arlington, USA, 2006

[Deg95] Degraeve, R., Groeseneken, G., Bellens, R., Depas, M., Maes, H.E., A consistent model for the thickness dependence of intrinsic breakdown in ultra-thin oxides, In *Technical Digest of International Electron Devices Meeting* (IEDM), S. 863-866, 1995

[DiM89] DiMaria, D. J., Stasiak, J. W., Trap creation in silicon dioxide produced by hot electrons, In *Journal of Applied Physics*, Vol. 65, Nr. 6, S. 2342-2356, 1989

[DiM95] DiMaria, D. J., Gartier, E., Mechanism for stress-induced leakage currents in thin silicon dioxide films, In *Journal of Applied Physics*, Vol. 78, Nr. 6, S. 2342-2356, 1995

[DiM97] DiMaria, D. J., Stathis, J. H.., Ultimate limit for defect generation in ultra-thin silicon dioxide, In *Applied Physics Letters*, Vol. 71, 1997

[Dum02] Dumin, D.J., Oxide wearout, breakdown, and reliability, In *Oxide Reliability*, Vol. 23, World Scientific, 2002

[Eln02] Elnozahy, E., Melhem, R., Mosse, D., Energy-efficient duplex and TMR real-time systems, In *Proceedings of 23rd IEEE Real-Time Systems Symposium* (RTSS), S. 256-266, Austin, USA, 2002

[Gai88] Gaitanis, N., The design of totally self-checking TMR fault-tolerant systems, In *IEEE Transactions on Computers*, Vol. 37, Nr. 11, S. 1450-1454, 1988

[Ghe01] Ghetti, A., Characterization and modeling of the tunneling current in Si-SiO2-Si structures with ultra-thin oxide layer, In *Microelectronic Engineering*, Vol. 59, Nr. 1-4, S. 127-136, 2001

[Gup11] Gupta, T., Heron, O., Ventroux, N., Zimmer T., Marc F., Bertolini C., System Level Analysis and Accurate Prediction of Electromigration, In *Proceedings of European Workshop on CMOS Variability* (VARI), Grenoble, Frankreich, 2011

[Har00] Harlow, J.E., Overview of popular benchmark sets, In *IEEE Design & Test of Computers*, Vol. 17, Nr. 3, S. 15-17, 2000

[Haz03] Hazucha, P., Karnik, T., Maiz, J., Walstra, S., Bloechel, B., Tschanz, J., Dermer, G., Hareland, S., Armstrong, P., Borkar, S., Neutron soft error rate measurements in a 90-nm CMOS process and scaling trends in SRAM from 0.25-µm to 90-nm generation, In Technical digest of IEEE International Electron Devices Meeting (IEDM), S. 21.5.1- 21.5.4, Washington, USA, 2003

[Her88] Heremans, P., Bellens, R., Groeseneken, G., Maes, H.E., Consistent model for the hot-carrier degradation in n-channel and p-channel MOSFETs, In *IEEE Transactions on Electron Devices*, Vol. 35, Nr. 12, S. 2194-2209, 1988

[Her10] Heryanto, A., Pey, K.L., Lim, Y.K., Liu, W., Wei, J., Raghavan, N., Tan, J.B., Sohn, D.K., Study of stress migration and electromigration interaction in copper/low-$\varkappa$ interconnects, In *Proceedings of IEEE International Reliability Physics Symposium* (IRPS), S. 586-590, Anaheim, USA, 2010

[Hey03] Heydari, P., Pedram, M., Ground bounce in digital VLSI circuits, In *IEEE Transactions on Very Large Scale Integration Systems*, Vol. 11, Nr. 2, S. 180-193, 2003

[Ho03] Ho, R., Mai, K., Horowitz, M., Managing wire scaling: a circuit perspective, In Proceedings of the IEEE International Interconnect Technology Conference, S. 177-179, Burlingame, USA, 2003

[Hor97] Hori T., *Gate Dielectrics and MOS ULSIs: Principles, Technologies, and Applications*, Springer, Berlin, 1997

[Hua00] Huang, M., Renau, J., Seung-Moon, Y., Torrellas, J., A framework for dynamic energy efficiency and temperature management, In *Proceedings of 33rd Annual IEEE/ACM International Symposium on Microarchitecture* (MICRO), S. 202-213, Monterey, USA, 2000

[Hun04] Hung, W., Addo-Quaye, C., Theocharides, T., Xie, Y., Vijakrishnan, N., Irwin, M.J., Thermal-aware IP virtualization and placement for networks-on-chip architecture, In *Proceedings of IEEE International Conference on Computer Design: VLSI in Computers and Processors* (ICCD), S. 430- 437, San Jose, USA, 2004

[Jed11] JEDEC Solid State Technology Association (JEDEC), *Failure Mechanisms and Models for Semiconductor Devices*, JEP122G, 2011

[Jia10] Jianxin F., Sapatnekar, S.S., Scalable methods for the analysis and optimization of gate oxide breakdown, In *Proceedings of 11th International Symposium on Quality Electronic Design* (ISQED), S. 638-645, San Jose, USA, 2010

[Kac02] Kaczer, B., Degraeve, R., Rasras, M., Van de Mieroop, K., Roussel, P.J., Groeseneken, G., Impact of MOSFET gate oxide breakdown on digital circuit operation and reliability, In *IEEE Transactions on Electron Devices*, Vol. 49, Nr. 3, S. 500-506, 2002

[Kai00] Kaijie W., Karri, R., Algorithm level re-computing with shifted operands-a register transfer level concurrent error detection technique, In *Proceedings of International Test Conference*, S. 971-978, Atlantic City, USA, 2000

[Ker07] Kerber, A., Rohner, M., Pompl, T., Duschl, R., Kerber, M., Lifetime Prediction for CMOS Devices with Ultra Thin Gate Oxides Based on Progressive Breakdown, In *Proceedings of 45th Annual IEEE International Reliability physics symposium*, S. 217-220, Phoenix, USA, 2007

[Koa09] Koal, T., Vierhaus, H. T., Logic Self Repair Based on Regular Building Blocks, In *Proceedings of 22nd International Conference on Architecture of Computing Systems* (ARCS), S. 1-6, Delft, Niederlande, 2009

[Koh80] Kohyama, S., Furuyama, T., Mimura, S., Iizuka, H., Non-Thermal carrier generation in MOS structures, In *Proceedings of the Conference on Solid State Devices*, S. 85-92, 1980

[Kor98] Koren, I., Koren, Z., Defect tolerance in VLSI circuits: Techniques and yield analysis, In *Proceedings of the IEEE*, Vol. 86, Nr. 9, S. 1819-1836, 1998

[Kor07] Koren, I. , Krisha, C., *Fault-tolerant Systems*, M Kaufmann, 2007

[Kun06] Kunhyuk K., Kufluoglu, H., Alain, M.A., Roy, K., Efficient Transistor-Level Sizing Technique under Temporal Performance Degradation due to NBTI, In *Proceedings of International Conference on Computer Design* (ICCD), S. 216-221, San Jose, USA, 2006

[Leb94] Leblebici, Y., Kang, S.M., Simulation of hot-carrier induced MOS circuit degradation for VLSI reliability analysis, In *IEEE Transactions on Reliability*, Vol. 43, Nr. 2, S. 197-206, 1994

[Len69] Lenzlinger M., Snow E., Fowler-Nordheim tunneling into thermally grown SiO_2, In *Journal of Applied Physics*, Vol. 40, S. 278-283, 1969

[Li92] Li, P.-C., Stamoulis, G.I., Hajj, I.N., A probabilistic timing approach to hot-carrier effect estimation, In *Technical Digest of IEEE/ACM International Conference on Computer-Aided Design* (ICCAD), S. 210-213, 1992

[Lin03] Linder, B.P., Stathis, J.H., Frank, D.J., Lombardo, S., Vayshenker, A., Growth and scaling of oxide conduction after breakdown, In *Proceedings of 41st Annual IEEE International Reliability Physics Symposium*, S. 402- 405, Dallas, USA, 2003

[Lom05] Lombardo, S., Stathis, J.H., Linder, B., Pey, K.L., Palumbo, F., Tung, C.H., Dielectric breakdown mechanisms in gate oxides, In *Journal of Applied Physics*, Vol. 98, Nr. 12, S. 121301-121301-36, 2005

[Mah88] Mahmood A., McCluskey E.J., Concurrent Error Detection Using Watchdog Processors-A Survey, In *IEEE Transactions on Computers*, Vol. 37, Nr. 2, 1988

[McP98] McPherson, J.W., Mogul, H.C., Underlying physics of the thermochemical E model in describing low-field time-dependent dielectric breakdown in SiO2 thin films, In *Journal of Applied Physics*, Vol. 84, Nr. 3, S. 1513-1523, 1998

[McP07] McPherson, J.W., Reliability Trends with Advanced CMOS Scaling and The Implications for Design, In *Proceedings of IEEE Custom Integrated Circuits Conference* (CICC), S. 405-412, San Jose, USA, 2007

[Mir00] Miranda, E., Sune, J., Rodriguez, R., Nafria, M., Aymerich, X., Fonseca, L., Campabadal, F., Soft breakdown conduction in ultrathin (3-5 nm) gate dielectrics , In *IEEE Transactions on Electron Devices*, Vol. 47, Nr. 1, S. 82-89, 2000

[Mit00] Mitra, S., McCluskey, E.J., Word-voter: a new voter design for triple modular redundant systems, In Proceedings of the 18th IEEE VLSI Test Symposium, S. 465-470, Montreal, Kanada, 2000

[Mit05] Mitra, S., Seifert, N., Zhang, M., Shi, Q., Kim, K.S., Robust system design with built-in soft-error resilience, In *Computer*, Vol. 38, Nr. 2, S. 43-52, 2005

[Moo65] Moore, G., Cramming More Components Onto Integrated Circuits, In *Electronics*, S. 114-117, 1965

[Moo86] Moore, W.R., A review of fault-tolerant techniques for the enhancement of integrated circuit yield, In *Proceedings of the IEEE*, Vol. 74, Nr. 5, S. 684-698, 1986

[Muk05] Mukherjee, S.S., Emer, J., Reinhardt, S.K., The soft error problem: an architectural perspective, In *Proceedings of the 11th International Symposium on High-Performance Computer Architecture* (HPCA), S. 243- 247, San Francisco, USA, 2005

[Oma07] Omana, M., Rossi, D., Metra, C., Latch Susceptibility to Transient Faults and New Hardening Approach, In *IEEE Transactions on Computers*, Vol. 56, Nr. 9, S. 1255-1268, 2007

[Pat02] Patterson, D. et al.: Recovery Oriented Computing (ROC): Motivation, Definition, Techniques, and Case Studies. *Berkeley Computer Science Technical Report*, 2002

[Rab96] J. Rabaey, *Digital Integrated Circuits: A Design Perspective*, Prentice Hall, 1996

[Ren01] Renovell M., Gallière J.M., Azais F., Bertrand Y., Analysing the Characteristics of MOS Transistors in the Presence of Gate Oxide Short, In *Proceedings of IEEE Design and Diagnostics of Electronic Circuits and Systems Workshop* (DDECS), S. 155-161, Gyor, Ungarn, 2001.

[Ren03a] M. Renovell, Galliere J.M., Azais F., Bertrand Y., Modelling the Random Parameters Effects in a Non-Split Model of Gate Oxide Short, In *Journal of Electronic Testing* (JETTA), Vol. 19, Nr. 4, S. 377-386, 2003

[Ren03b] Renovell, M., Galliere, J.M., Azais, F., Bertrand, Y., Delay testing of MOS transistor with gate oxide short, In *Proceedings of 12th Asian Test Symposium* (ATS), S. 168-173, Xian, China, 2003

[Rod03] Rodriguez, R., Stathis, J.H., Linder, B.P., A model for gate-oxide breakdown in CMOS inverters, In *IEEE Electron Device Letters*, Vol. 24, Nr. 2, S. 114-116, 2003

[Rom05] Romeu, J., Understanding Series and Parallel Systems Reliability, In *START*, Vol. 11, Nr. 5, 2005

[Ru00] Ru, C.Q., Thermomigration as a driving force for instability of electromigration induced mass transport in interconnect lines, In *Journal of Materials Science*, Vol. 35, Nr. 24, S. 5575-5579, 2000

[Sae09a] Sämrow, H., Cornelius, C., Sill, F., Tockhorn, A., Timmermann, D., Comparison of Strategies for Redundancy to improve Reliability concerning Gate Oxide Breakdown, *Workshop für Testmethoden und Zuverlässigkeit von Schaltungen und Systemen* (TUZ), S. 97-102, Bremen, Deutschland, 2009

[Sae09b] Sämrow, H., Cornelius, C., Sill, F., Tockhorn, A., Timmermann, D., Automated Insertion of Twin Gates to improve Reliability concerning Gate Oxide Breakdown, *SPIE Europe - Microtechnologies for the New Millennium*, Nr. 736310, Dresden, Deutschland, 2009

[Sae09c] Sämrow, H., Cornelius, C., Sill, F., Tockhorn, A., Timmermann, D., Twin Logic Gates - Improved Logic Reliability by Redundancy concerning Gate Oxide Breakdown, In *Proceedings of the 22nd Annual symposium on Integrated circuits and system design* (SBCCI), S. 315-320, Natal, Brasilien, 2009

[Sae10] Sämrow, H., Cornelius, C., Salzmann, J., Tockhorn, A., Timmermann, D., Utilizing Parallelism of TMR to Enhance Power Efficiency of Reliable ASIC Designs, In *Proceedings of 6th International Conference on Computer Engineering and Systems* (ICCES), S. 251-256, Kairo, Ägypten, 2010

[Sae11] Sämrow, H., Cornelius, C., Gorski, P., Salzmann, J., Tockhorn, A., Timmermann, D., Functional Enhancements of TMR for Power Efficient and Error Resilient ASIC Designs, In *Proceedings of 14th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems*, S. 183-188, Cottbus, Deutschland, 2011

[Sae12a] Sämrow, H., Cornelius, C., Gorski, P., Salzmann, J., Timmermann, D., Effiziente Simulation von Gateoxiddefekten auf Gatterebene mit Transistorlevel-Genauigkeit, *15th Workshop Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen* (MBMV), S. 157-168, Kaiserslautern, Deutschland, 2012

[Sae12b] Sämrow, H., Cornelius, C., Gorski, P., Salzmann, J., Timmermann, D., Selective redundancy to improve reliability and to slow down delay degradation due to gate oxide breakdown, *15th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems* (DDECS), S. 12-15, Tallinn, Estonia, 2012

[Sak90] Sakurai, T. Newton, A. R., Alpha-Power Law MOSFET Model and Its Application to CMOS Inverter Delay and Other Formulas, In *IEEE Journal of Solid-State Circuits,* Vol. 25, Nr. 2, S. 584-594, 1990

[Sal08] Saluja, K.K., Vijayakumar, S., Sootkaneung, W., Xaingning Yang, NBTI Degradation: A Problem or a Scare?, In *Proceedings of 21st International Conference on VLSI Design* (VLSID), S. 137-142, Hyderabad, Indien, 2008

[Sas06] Sasaki, Y., Namba, K., Ito, H., Soft Error Masking Circuit and Latch Using Schmitt Trigger Circuit, In *Proceedings of 21st IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems* (DFT), S. 327-335, Arlington, USA, 2006

[Sch03] Schroder, D.K., Babcock, J.A., Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing, In *Journal of Applied Physics,* Vol. 94, Nr. 1, S. 1-18, 2003

[Seg95] Segura, J., De Benito, C., Rubio, A., Hawkins, C.F., A detailed analysis of GOS defects in MOS transistors: testing implications at circuit level, In *Proceedings of International Test Conference*, S. 544-551, Washington, USA, 1995

[She06] Shen, C., Li, M.-F., Foo, C. E., Yang, T., Huang, D. M., Yap, A., Samudra, G. S., Yeo, Y.-C., Characterization and Physical Origin of Fast Vth Transient in NBTI of pMOSFETs with SiON Dielectric, In Technical Digest of IEEE International Electron Devices Meeting (IEDM), S. 1-4, San Francisco, USA, 2006

[Shi98] Shiga, K., Komori, J., Katsumata, M., Teramoto, A., Sekine, M., A new test structure for evaluation of extrinsic oxide breakdown, In *Proceedings of the 1998 International Conference on Microelectronic Test Structures* (ICMTS), S. 197-200, Kanazawa, Japan, 1998

[Shi03] Shivakumar P., Keckler, S.W., Moore, C.R., Burger, D., Exploiting microarchitectural redundancy for defect tolerance, In *Proceedings of 21st International Conference on Computer Design*, S. 481-488, San Jose, USA, 2003

[SIA07] Semiconductor Industry Association (SIA), *International Technology Roadmap for Semiconductors*, 2007

[SIA11] Semiconductor Industry Association (SIA), *International Technology Roadmap for Semiconductors*, 2011

[Sie91] Siemwiorek, D.P., Architecture of fault-tolerant computers: an historical perspective, In *Proceedings of the IEEE*, Vol. 79, Nr. 12, S. 1710-1734, 1991

[Sil07] Sill, F., *Untersuchung und Reduzierung des Leckstroms integrierter Schaltungen in Nanometer-Technologien bei konstanten Performanceanforderungen*, Dissertation, Fakultät für Informatik und Elektrotechnik, Universität Rostock, 2007

[Sir04] Sirisantana N., Paul, B.C., Roy K., Enhancing yield at the end of the technology roadmap, In *IEEE Design & Test of Computers*, Vol. 21, Nr. 6, S. 563-571, 2004

[Spi04] Spica, M., Mak, T.M., Do we need anything more than single bit error correction (ECC)?, In *Records of the 2004 International Workshop on Memory Technology, Design and Testing*, S. 111-116, San Jose, USA, 2004

[Sri03] Srinivasan, J. et al.: RAMP: A Model for Reliability Aware Microprocessor Design. In *IBM Research Report*, RC23048, 2003

[Sri04] Srinivasan, J., Adve, S.V., Bose, P., Rivers, J.A., The case for lifetime reliability-aware microprocessors, In Proceedings of 31st Annual International Symposium on Computer Architecture, S. 276-287, München, Deutschland, 2004

[Sta96] Stathis, J.H., Cartier E., The Role of Atomic Hydrogen in Degradation and Breakdown of SiO2 Films, In *Proceedings of International Conference on Solid State Devices and Materials*, S. 791-793, Yokohama, Japan, 1996

[Sta02] Stathis, J. H., Reliability limits for the gate insulator in CMOS technology, *IBM Journal of Research and Development* , Vol. 46, Nr. 2.3, S. 265-286, 2002

[Sue04] Suehle, J.S., Zhu, B., Che, Y., Bernstein, J.B., Acceleration factors and mechanistic study of progressive breakdown in small area ultra-thin gate oxides, In *Proceedings of IEEE International Reliability Physics Symposium* (IRPS), S. 95-101, Phoenix, USA, 2004

[Sut99] Sutherland, I., Sproull, B., Harris, D., *Logical Effort*, Morgan Kaufmann, 1999

[Syn11] *Synopsys™ DesignCompiler™ Manual*, 2011.

[Syr87] M. Syrzycki, Modeling of Spot Defects in MOS Transistors, In *Proceedings of International Test Conference*, S. 148-157, Washington, USA, 1987

[Tak83] Takeda, E., Nakagome, Y., Kume, H., Asai, S., New hot-carrier injection and device degradation in submicron MOSFETs, In *IEE Proceedings I of Solid-State and Electron Devices*, Vol. 130, Nr. 3, S. 144-150, June 1983

[Tau98] Taur, Y., Ning, T.H., *Fundamentals of Modern VLSI Devices*, New York: Cambridge, Univ. Press, 1998

[Toc10] Tockhorn, A., Cornelius, C., Sämrow, H., Timmermann, D., Modeling Temperature Distribution in Networks-on-Chip using RC-Circuits, In Proceedings of 13th IEEE International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS), S. 229-232, Wien, Österreich, 2010

[Tsc02] Tschanz, J.W., Kao, J.T., Narendra, S.G., Nair R., Antoniadis, D.A., Anantha P. De V., Adaptive Body Bias for Reducing Impacts of Die-to-Die and Within-Die Parameter Variations on Microprocessor Frequency and Leakage„ In *IEEE Journal Of Solid-State Circuits*, Vol. 37, S. 1396-1402, 2002

[Tu93] Tu, R.H., Rosenbaum, E., Chan, W.Y., Li, C.C., Minami, E., Quader, K., Ko, P.K., Hu, C., Berkeley reliability tools-BERT, In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 12, Nr. 10, S. 1524-1534, 1993

[Van01] Van Heijningen, M., Badaroglu, M., Donnay, S., De Man, H., Gielen, G., Engels, M., Bolsens, I., Substrate noise generation in complex digital systems: efficient modeling and simulation methodology and experimental verification, In *Digest of Technical Papers IEEE International Solid-State Circuits Conference*, S. 342-343, 2001

[Via08] Vial, J., Bosio, A., Girard, P., Landrault, C., Pravossoudovitch, S., Virazel, A., Using TMR Architectures for Yield Improvement, In *Proceedings of IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems*, S. 7-15, Boston, USA, 2008

[Vog00] Vogel, E.M., Suehle, J.S., Edelstein, M.D., Wang, B., Chen, Y., Bernstein, J.B., Reliability of ultrathin silicon dioxide under combined substrate hot-electron and constant voltage tunneling stress, In *IEEE Transactions on Electron Devices*, Vol. 47, Nr. 6, S. 1183-1191, 2000

[Von56] Von Neumann, J.: Probabilistic logics and the synthesis of reliable organisms from unreliable components", In *Automata Studies*, Princeton University Press, 1956

[Wal64] Wallace, C. S., A Suggestion for a Fast Multiplier, In *IEEE Transactions on Electronic Computers,* , Vol. EC-13, Nr. 1, S.14,17, 1964

[Weg11] Wegner, T., Gag, M., Timmermann, D., Impact of proactive temperature management on performance of Networks-on-Chip, In *Proceedings of International Symposium on System on Chip* (SoC), S. 116-121, Tampere, Finnland, 2011

[Wei97] Weir, B.E., Silverman, P.J., Monroe, D., Krisch, K.S., Alam, M.A., Alers, G.B., Sorsch, T.W., Timp, G.L., Baumann, F., Liu, C.T., Ma, Y., Hwang, D., Ultra-thin gate dielectrics: they break down, but do they fail?, In *Technical Digest of International Electron Devices Meeting* (IEDM), S. 73-76, 1997

[Wes93] Weste N., Eschraghian K., *Principles of CMOS VLSI Design, A System Perspective*„ Addison Wesley, 1993

[Whi07] White D., A new approach to higher yielding silicon, White Paper, Synopsys

[Wu99] Wu, J., Register, L.F., Rosenbaum, E., Trap-assisted tunneling current through ultra-thin oxide, In *Proceedings of IEEE 37th Annual International Reliability Physics Symposium*, S. 389-395, San Diego, USA, 1999

[Wu00] Wu, L. et al.: GLACIER: A Hot Carrier Gate Level Circuit Characterization and Simulation System for VLSI Design, In *Proceedings of First International Symposium on Quality of Electronic Design*, San Jose, USA, 2000

[Wu02a] Wu, E. Y., Nowak, E. J., Vayshenker, A., Lai, W. L., Harmon, D. L., CMOS scaling beyond the 100-nm node with silicon-dioxide-based gate dielectrics, In *IBM Journal of Research and Development*, Vol. 46, Nr. 2.3, S. 287-298, 2002

[Wu02b] Wu, E., et al., Polarity-dependent oxide breakdown of NFET devices for ultra-thin gate oxide, In Proceedings of 40th Annual Reliability Physics Symposium, S. 60-72, Dallas, USA, 2002

[Wu02c] Wu, E., Suñé, J., Lai, W., Nowak, E., McKenna J., Vayshenker, A., Harmon D., Interplay of voltage and temperature acceleration of oxide breakdown for ultra-thin gate oxides, In *Solid-State Electronics*, Vol. 46, Nr. 11, 2002, S. 1787-1798

[You07] Kim, Y.B., Kim, Y.-B., Fault Tolerant Source Routing for Network-on-chip, In *Proceedings of 22nd IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems* (DFT), S. 12-20, Rom, Italien, 2007

[Zhu04] Zhu, D., Melhem, R., Mosse, D., Elnozahy, E., Analysis of an energy efficient optimistic TMR scheme, In *Proceedings of 10th International Conference on Parallel and Distributed Systems* (ICPADS), S. 559-568, Newport Beach, USA, 2004

Kurzreferat

Die fortschreitende Skalierung integrierter Schaltungen zur stetigen Verringerung der Strukturgrößen führt zu einer Verbesserung der dynamischen Parameter. Allerdings erreichen dadurch die Verschleißerscheinungen der Transistoren, wie Gateoxiddefekte, aufgrund von Kurzkanaleffekten ein Ausmaß, das die Lebensdauer dieser Schaltungen allein durch den Betrieb signifikant begrenzt. Weiterhin wird aufgrund ansteigender statischer Ströme durch die Defekte im Laufe der Betriebsdauer auch die Leistungsaufnahme erhöht, was wiederum zu einer Verringerung der Zuverlässigkeit führt. Die vorliegende Arbeit identifiziert die Notwendigkeit Verschleißeffekte bereits im Design zu berücksichtigen, um dieser Entwicklung schon als Schaltungsdesigner Rechnung zu tragen. Viele CAD-Tools, die oberhalb der Transistorebene angesiedelt sind, verwenden nur pauschale Erhöhungen der Verzögerungszeit, um den Verschleiß zu modellieren. Ferner existieren wenig redundante Ansätze zur Zuverlässigkeitssteigerung auf Gatterebene hinsichtlich Gateoxiddefekten. Um dieser Problematik zu begegnen, wurden neue Ansätze zur Erhöhung der Zuverlässigkeit für kombinatorische integrierte Schaltungen vorgestellt. Des Weiteren befasst sich diese Arbeit mit der Entwicklung eines Simulators zur Analyse der Auswirkungen von Gateoxiddefekten. Die generierten Ansätze fokussieren dabei die Gatterebene, womit sie sich in einen standardisierten CMOS-Designablauf integrieren lassen. Weiterhin wird auch auf die anderen Designparameter, wie Fläche, Verzögerungszeit und besonders die Leistungsaufnahme, deren Reduzierung zu weniger Verschleiß führt, eingegangen.

Es werden mehrere Lösungen zur Steigerung der Zuverlässigkeit, die auf redundanten Techniken der Gatter- und Architekturebene beruhen, für eine 65 nm Technologie aufgezeigt. In einem ersten Ansatz wird die Leistungsaufnahme üblicher redundanter Architekturen mit geänderten Ausführungsstrukturen auf bis zu 40 % gesenkt, wobei die Zuverlässigkeit sogar um ca. 20 % gesteigert werden kann. Im Anschluss werden Ansätze auf Gatterebene entwickelt, die dazu führen, dass der Verschleiß der Transistoren in Bezug auf Gateoxiddefekte und im Vergleich zu den Referenzschaltungen verlangsamt wird. Die Zuverlässigkeit kann schaltungsabhängig um bis 60 % erhöht werden, was einer Verdopplung verglichen mit bisherigen Ansätzen auf der Gatterebene entspricht. Diese Verbesserungen werden mit vergleichsweise geringen Steigerungen der Leistungsaufnahme und der Verzögerungszeit von ungefähr 10 % erreicht, wobei die Fläche um circa 50 % erhöht wird, indem selektiv redundante Transistorstacks in der Entwurfsphase implementiert, aber erst im Laufe der Betriebsdauer zugeschaltet werden. Dieser Ansatz integriert sich sehr gut in den Standard-Designflow. Um nun die „Schwachstellen“ einer Schaltung zu ermitteln, wird die Entwicklung eines Zuverlässigkeitssimulators auf Gatterebene vorgestellt. Ziel war es, den probabilistischen Verschleißverlauf einer integrierten Schaltung realistisch nachzuvollziehen und die Auswirkungen auf Designparameter mit vorhandenen CAD-Tools simulieren zu können, weshalb Gateoxiddefekte ausgehend von Transistormodellen auf die Gatterebene in das Composite Current Source Modell übertragen wurden. Damit wird bereits nach der Synthese der Gatternetzliste eine genaue Zuverlässigkeitsabschätzung möglich. Testergebnisse zeigen, dass die Genauigkeit der berechneten Werte die Qualität von komplexeren Transistormodellen erreicht.

Abstract

The performance of integrated circuits have been increased permanently over the past decades due to the proceeding scaling of the MOSFET dimensions and at the cost of an increasing structural vulnerability of the transistor devices. Unfortunately, this process results in negative short channel effects which deteriorates the transistor gate oxide during the operation of the integrated circuit. The extent of these dielectric degradation leads to a significant reduction of circuit reliability which could not be neglected by chip designers. Furthermore, static currents through the defects additionally aggravates those reliability issues.

This work identifies the imperative to consider gate oxide breakdown already at design stage. Many CAD tools at gate level respond to these issues by increasing the circuit delay globally to operate under less critical conditions. Further on, only few approaches at gate level exist to increase reliability under consideration of transistor gate defects. A major result of this work is the increase of reliability for combinatorial circuits using a new flow of exiting approaches and creating new redundant structures. Additionally, a reliability simulator to analyze and demonstrate the degradation behavior of integrated circuits over the operating time is presented. All solutions focus at gate level to be fully and seamlessly capable of being integrated into the standard design flow. Thereby, critical design parameters, as area consumption, delay and especially current consumption will be considered.

This work presents multiple solutions which are based on redundant approaches at gate and architecture level to increase circuit reliability for a 65 nm technology. First, the current consumption of integrated designs will be decreased by 60 % with an additional reliability gain of 20 % due to altered processing structures of conventional TMR architectures. Furthermore, gate level approaches have been developed which decelerates the transistor degradation as compared to the original designs. The reliability is increased by about 60 % which represents a duplication in contrast to state-of-the-art approaches. This comes in hand with a relative small increase of the current consumption and delay of the circuit of 10 % due to the fact that the redundant transistor stacks will be activated not before circuit parameter degradation is detected during the operation time. The area is increased to 1.5 times of the original one. This approach could be integrated easily into the standard design flow. To know the weaknesses of a generated design, a reliability simulator is introduced which is based on transistor level models. These models are transferred to the Composite Current Source model to analyze gate level netlists with the objective to demonstrate realistically the behavior of the circuit in dependence of several transistor gate defects that are inserted randomly. These defects degrades design parameters over time. By using libraries and netlists which are generated by the simulator, proprietary CAD tools could simulate the impact of these defects. Thereby, a precise estimation of reliability issues is possible right after the synthesis which reaches the degree of transistor level simulations.

Maßnahmen zur Steigerung der Zuverlässigkeit integrierter Schaltungen auf Gatterebene hinsichtlich Gateoxiddefekten

Thesen der Dissertation

zur

Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik

der Universität Rostock

vorgelegt von

Hagen Sämrow, geb. am 02.06.1979 in Schwerin

aus Hamburg

Hamburg, den 02.12.2013

Thesen

1. Die fortschreitende Skalierung von CMOS-Nanometer-Technologien führt zu verstärkten Abnutzung des Gateoxides der MOSFET aufgrund von Kurzkanaleffekten. Daraus resultiert eine verkürzte Betriebsdauer und eine früher eintretende Verschleißphase der gesamten Schaltung, da sich die Schaltungsparameter ständig verschlechtern.
2. Statische Ströme durch das defekte Gateoxid der betroffenen Transitoren verringern zusätzlich die Zuverlässigkeit, da sie zu erhöhten thermalen Stress und damit zu einem stärkeren Verschleiß führen.
3. Nach der Ausbildung des ersten Defektes nimmt der Gateoxidverschleiß einen progressiven Verlauf. Zuerst steigt der Leckstrom durch das Gateoxid kaum bis linear an. Dann folgt der soft breakdown, wobei der Stromfluss exponentiell ansteigt. In der letzten Phase ist der Strom gesättigt und kann zu einem Kurzschluss durch das Gate führen (hard breakdown).
4. Neben der funktionellen Störung des Transistors während des hard breakdowns, werden Parameter, wie Schwellspannung, Leckstrom, Konduktanz, Strom-Spannungskurve, usw., bereits vorher beeinträchtigt. Deshalb verschlechtert sich die Performance integrierter Schaltungen im Verlauf des Betriebes erheblich.
5. Die Wahrscheinlichkeit zur Ausbildung eines Gateoxiddefektes ist im Wesentlichen abhängig von der Betriebsdauer, der Schaltwahrscheinlichkeit, der Temperatur, der Gateoxiddicke und der Gateoxidfläche des Transistors.
6. Ansätze zur Defektmaskierung existieren vor allem im Layout, wobei mittels redundanter Strukturen die Zuverlässigkeit erhöht wird. Auf Architekturebene sind neben der Defektmaskierung vor allem adaptive Strategien zur Defektvermeidung publiziert worden.
7. Auf Gatterebene, in der die Synthese der RTL-Architektur in Standardzellennetzlisten stattfindet, sind wenig Ansätze vorhanden, in der Gateoxiddefekte Berücksichtigung finden.
8. Auf Transistorebene kann mit Hilfe zufällig eingefügter Transistoren die Ausbeute integrierter Schaltungen signifikant verbessert werden. Eine Erweiterung dieses Ansatzes mit Berücksichtigung der Wahrscheinlichkeit von Gateoxidefekten erhöht die Zuverlässigkeit der Schaltungen.
9. Triple Modular Redundancy (TMR) Ansätze erhöhen die Zuverlässigkeit integrierter Schaltungen gegenüber flüchtigen Defekten. Wird das ursprüngliche TMR um einen weiteren Modus erweitert, in der die dreifach ausgelegten Module unabhängig voneinander Ergebnisse generieren, wird die Verlustleistung eines TMR Designs auf die Hälfte gesenkt. Die Zuverlässigkeit gegenüber dem originalen TMR wird nur geringfügig verringert, während die Fläche nur minimal erhöht wird. Die Fähigkeit zur Vermeidung temporärer Defekte wird verloren.
10. Die höchsten Einsparung erzielt man mit größeren Schaltungen mit wenig Ausgangspins.

11. Ein hoher Anteil des Einzelmodus an der gesamten Betriebsdauer führt zu einer größeren Verringerung des Leistungsaufnahme, was wiederum mittelbar die Zuverlässigkeit erhöht. Ein geringer Anteil des TMR-Modus an der gesamten Betriebsdauer wird durch eine ausreichend lange Verweildauer im TMR Modus sehr gut kompensiert.
12. Wird ein weiterer Modus hinzugefügt, in der nur zwei Module gleichzeitig zwischen paralleler und gleichzeitiger Abarbeitung der Eingangsvektoren schalten, wird die Leistungsaufnahme weiter gesenkt, da das dritte Modul ruht. Der Flächenaufwand wird nur geringfügig erhöht.
13. Durch Auswahl eines geeigneten finalen Modus, wenn mehrere Defekte von der Schaltung entdeckt worden sind, wird die Zuverlässigkeit des ursprünglichen TMR Designs sogar erhöht. Hierbei wird das verbliebene und korrekt funktionierende Modul zur weiteren Abarbeitung der Eingangsvektoren ausgesucht.
14. Die Verdopplung von Gattern in einer integrierten Schaltung verbessert die Zuverlässigkeit gegenüber Gateoxiddefekten signifikant.
15. Verschleiß redundant eingefügter Transistorstacks wird durch späteres Zuschalten vermieden. Dadurch erhöhen sich die Zuverlässigkeit integrierter Schaltungen kurz vor und in der Verschleißphase im Gegensatz zu anderen redundanten Ansätzen erheblich.
16. Die Beachtung der Schaltwahrscheinlichkeit bei der Auswahl redundanter Transistorstacks führt zu einem guten Kompromiss zwischen zusätzlichem Flächenaufwand und einer wesentlichen Steigerung der Zuverlässigkeit. Die Relevanz hinsichtlich der Schaltungsverzögerungszeit sollte nur bei sehr starken Flächenrestriktionen berücksichtigt werden.
17. Der fortschreitende Verschleiß kann durch das spätere Zuschalten der redundanten Strukturen verlangsamt werden, wobei der Zuschaltzeitpunkt die Zuverlässigkeitserhöhung durch die redundanten Transistorstacks beeinflusst.
18. Das Einfügen der vorgestellten redundanten Transistorstacks in den standardisierten Syntheseablauf erfolgt problemlos, da sie als Standardgatter implementiert werden können.
19. Die Auswertung von Simulationen auf Transistorebene und eine Überführung in das CCS-Modell dient als Grundlage für einen Simulator auf Gatterebene, der den Einfluss von Gateoxiddefekten auf synthetisierte Netzlisten über die Betriebsdauer aufzeigt.
20. Die Auswahl des CCS-Modells als Grundlage für die statische Zeitanalyse ist zwingend erforderlich, da Gateoxiddefekte die Stromkurve am Ausgang des Gatters verändern. Das CCS-Modell verwendet Stromkurven zur Ermittlung der Gatterlaufzeit.
21. Die Defektanalyse einer Netzliste wird sowohl zeitlich als auch schaltungstechnisch sequentiell abgearbeitet, um sowohl dem progressiven Verlauf der Defekte, als auch der gegenseitigen Beeinflussung der Gatter untereinander gerecht zu werden.

22. Spannungspegel eines Gatters und Gatterlaufzeit werden neben einem eventuellen eigenen Defekt auch durch verbundene defekte Gatter an den Eingangsnetzen und am Ausgangsnetz beeinflusst., so dass die CCS-Stromkurven auch für Gatter, die nicht von einem Defekt betroffen sind, neu generiert werden müssen.

23. Gateoxiddefekte können sich auch positiv auf die Performance und die Ausgangspegel einzelner angeschlossener Gatter auswirken. Negative Auswirkungen fallen allerdings deutlicher ins Gewicht. Gateoxiddefekte eines Transistors in einem Gatter wirken sich immer negativ auf die Laufzeit und die Spannungspegel am Gatterausgang aus.

24. Spannungspegel innerhalb eines Designs werden mittels des vorgestellten Simulators sehr gut prognostiziert.

25. Die CCS-Stromkurve der einzelnen Gatter wird mit Hilfe der Stromflüsse durch Defekte des Gatters und durch anliegende Defekte erstellt. Die Analyse der Netzlisten, die mit den modifizierten CCS-Stromkurven simuliert werden, stellt die langsame Verschlechterung der Performance der integrierten Schaltung gut dar.

26. Die Zusammenfassung der ermittelten Verzögerungszeiten der statischen Zeitanalyse mit Hilfe modifizierter CCS-Bibliotheken und -Netzlisten gestattet eine Schwachstellenanalyse ähnlich der Statistischen Statischen Zeitanalyse. Die Auswertung dieser Analyse kann in Kombination mit den redundanten Transistorstacks gut in den Syntheseablauf integriert werden, so dass in der Synthese auch Auswirkungen eventueller Gateoxiddefekte Berücksichtigung finden.