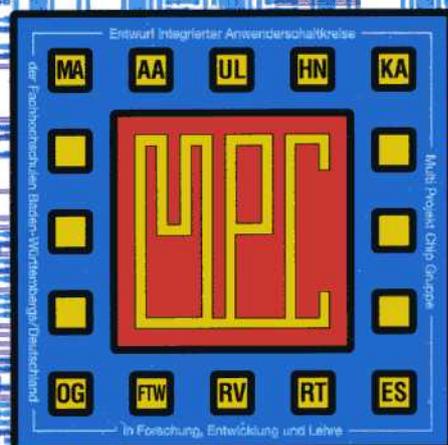


MULTIPROJEKT CHIP-GRUPPE

BADEN - WÜRTTEMBERG

Workshop Juni 1995

Offenburg



**MULTIPROJEKT
CHIP-GRUPPE
BADEN - WÜRTTEMBERG**

Workshop Juni 1995

Offenburg

Herausgeber: Fachhochschule Ulm

© 1995 Fachhochschule Ulm

Das Werk und seine Teile sind urheberrechtlich geschützt. Jede Verwertung in anderen als den gesetzlich zugelassenen Fällen bedarf deshalb der vorherigen schriftlichen Einwilligung des Herausgebers Prof. A. Führer, Fachhochschule Ulm, Prittwitzstraße 10, 89075 Ulm.

Inhaltsverzeichnis

- | | |
|--|-----------|
| 1. Meß- und Analyseverfahren in der Halbleitertechnologie R. Riekeles, FH Reutlingen | Seite 5 |
| 2. Einsatz des BSIM3-Modells zur Simulation von analogen und digitalen CMOS-Schaltungen im Submikron-Bereich T. Gneiting, H. Khakzar, FH Esslingen | Seite 19 |
| 3. Strukturierter hierarchischer Entwurf am Beispiel des μ P-Kernels FHOP O. Feißt, D. Jansen, FH Offenburg | Seite 38 |
| 4. Diamond-Transistoren für Transimpedanzverstärker J. Storch, H. Clauss, FH Heilbronn | Seite 46 |
| 5. Rechnergesteuertes Testen eines Analog-ICs mit „VEE-Test“ A. Diemer, M. Basler, G. Forster, FH Ulm | Seite 59 |
| 6. Ein Demonstrationsprogramm zur Problematik der Temperaturverteilung auf integrierten Schaltungen G. Albert, FH Mannheim | Seite 75 |
| 7. FPGA als konfigurierbarer DMA-Controller mit Multiplexer- Demulti- plexerfunktion zur Echtzeitspeicherung von Videosequenzen F. Imminger, H. Sauerburger, FH Furtwangen | Seite 82 |
| 8. Netzlistengenerator als Interface zwischen Design Architect und Spice P. Xipteras, G. Kampe, FH Esslingen | Seite 97 |
| 9. LOGTUTOR Interaktives Lernprogramm für die Einführung in LOG/iC mit PLD E. Prochaska, FH Heilbronn | Seite 105 |

Meß- und Analyseverfahren in der Halbleitertechnologie

R. Riekeles (FH-RT)

Inhalt:

In-Prozess Messungen

Test- und Hilfsstrukturen

Fehleranalyse

MPC-XIII.Workshop am 30.06.1995 in Offenburg

Einführung

Die Herstellung von komplexen integrierten Halbleiterbauelementen (z.B. Abb.1) erfordert eine aufwendige und gut angepaßte Meßtechnik, um eine wirtschaftliche Fertigung mit guten Ausbeuten und einem möglichst hohen Qualitätsstandard zu erreichen.

Die Aufgaben der Meßtechnik in den verschiedenen Stadien der Bauelemente-Herstellung sind vielfältig:

Im Ablauf der Waferbearbeitung: - In-Prozess Messungen

Am fertig prozessierten Wafer: - Funktionsprüfung der Chips
- Kontrolle der Technologie-Parameter an Teststrukturen (PCM-Process-Control-Monitor)

Am Bauelement im Gehäuse: - Endmessung (Funktionsprüfung)
- Lebensdauertests
- Fehleranalysen

In diesem Beitrag werden einige ausgewählte Verfahren aus dem Bereich der In-Prozess Messungen und der Fehleranalyse dargestellt.

In-Prozess Messungen

| Messgröße | Verfahren | Gerät |
|--|---|--|
| Dotierung | Widerstandsmessung Kristallstörung | Vier-Spitzen-Messplatz Therma-Probe |
| Schichtdicke | Opt. Interferrometrie Ellipsometrie Oberflächenprofil- Abtastung | Nanospec Dektak |
| Struktur -Abstand -Passgenauigkeit | Opt. Inspektion | Mikroskop |

Die Kontrolle der verschiedenen dotierten Schichten nach der Ionenimplantation und der Eindiffusion der Dotierstoffe läßt sich mit unstrukturierten Testwafern, die unter gleichen Bedingungen prozessiert werden, am einfachsten durchführen.

Mit der **Vier-Spitzen-Methode** (siehe Abb. 2 u. 3) können im Wertebereich von 10^{-3} bis 10^{+4} Ohm zuverlässige Werte für den Schichtwiderstand (Quadratwiderstand) nach der folgenden Formel bestimmt werden:

$$R_s = \rho \times 1/d = \pi / \ln 2 \times U/I = 4,532 \times U/I$$

Die Schichtdicke (d) kann mit speziellen Präparationstechniken (Schrägschliffe und Anätzung) optisch gemessen und in den meisten Fällen als gut reproduzierbar und relativ konstant betrachtet werden. Mit dem aus Abb.4 ersichtlichen Zusammenhang zwischen dem spezifischen Widerstand (ρ) und der Ladungsträgerkonzentration läßt sich mit der Widerstandsmessung die Dotierung charakterisieren.

Die Dotierung soll auf dem Wafer eine möglichst gleichmäßige Verteilung aufweisen, wie dies am Beispiel in Abb.5 ersichtlich ist.

Die Kontrolle der zahlreichen Dotierprozesse erfordert eine entsprechende Anzahl von Testwafern, die bei den zunehmend größeren Durchmessern der Scheiben einen erheblichen Kostenfaktor darstellen. Vorteilhaft sind daher Verfahren, die Messungen direkt auf den Produkt-Wafern erlauben. Mit dem in Abb. 6 u.7 dargestellten Meßprinzip der **"Thermaprobe"** ist dies möglich. Hier wird mit einem LASER eine Wärmewelle (Frequenz im MHz-Bereich) angeregt und die Änderung der optischen Reflektivität der Kristalloberfläche mit einem zweiten LASER detektiert. Die Ausbreitung und die Reflektion dieser Wellen wird durch Kristallstörungen beeinflusst.

Wie aus Abb.8 ersichtlich ist, lassen sich diese Referenzwerte (TW-units) der implantierten Dotierstoffkonzentration zuordnen. Die Meßwerte hängen allerdings auch von der Tiefe und der Art der Kristallstörungen ab. Mit Arsen und Phosphor wird bei den in Abb.8 mit den Pfeilen gekennzeichneten Dosen der Si-Einkristall bereits amorphisiert, so daß sich in diesem Bereich eine Zuordnung nur durch eine Vielzahl von fein abgestuften Referenzmessungen durchführen läßt.

Test- und Hilfsstrukturen

Auf den Wafern werden in der Regel einige Chips ausgespart und an diesen ausgewählten Positionen Teststrukturen eingeblendet. An diesen können nach dem vollständigen Prozessdurchlauf die relevanten Parameter elektrisch gemessen werden. Ein Beispiel für ein Teststruktur-Layout zur Kontrolle der Schichtwiderstände der verschiedenen Diffusionsebenen in einem Bipolar-Prozess ist in Abb.10 dargestellt. Der vollständige Prozess-Kontroll-Monitor umfaßt 6 Blocks mit je 28 Anschlußpad's, die alle in dieser Technologie realisierten Bauelemente als einzeln meßbare Muster enthalten.

In Abb.11 sind einige der Meßparameter, die Zielwerte und die Meßgrenzen aufgelistet. Diese Monitor-Strukturen sind für die Halbleiter-Prozestechnik ein wichtiges Hilfsmittel, um Neuanläufe und kontinuierlich Verbesserungen gezielt durchführen zu können.

Fehleranalyse

Einige Aufgaben und Verfahren, die an den teilweise auch während des Herstellungsprozesses zur Qualitätssicherung und Fehleranalyse in der nebenstehenden Übersicht zusammengestellt.

| <u>Fehleranalyse</u> (Qualitäts-Prüfungen) | <u>Verfahren</u> |
|--|--------------------------------------|
| Struktur | Optische Mikroskopie (Video-Technik) |
| Aufbau / Querschnitt / Profil | Elektronenmikroskopie (REM, TEM) |
| Statistische Fehler | Rastertunnelmikroskop |
| Prozessfehler | Röntgenaufnahmen, Spektroskopie |
| Layoutfehler | Sekundär-Ionen Massenspektrometrie |
| | Fokussierte Ionenstrahlen (FIB) |

Im Folgenden wird das als letztes aufgeführte Verfahren, bei dem als "Werkzeug" ein **Fokussierter Ionenstrahl (FIB)** eingesetzt wird, beschrieben. Im Prinzip ist dieses bildgebende Verfahren der Raster-Elektronenmikroskopie sehr ähnlich. Mit dem energiereicheren Ionenstrahl lassen sich zusätzlich die Oberflächen abtragen (Sputtern) und durch Gaszusätze Ätzprozesse und Abscheidungen von Schichten (CVD) gezielt ausführen.

Der Aufbau einer Ionen-Strahl-Säule ist aus Abb. 12 u. 13 ersichtlich. Um die störende Aufladung der Oberfläche zu vermeiden wird diese zusätzlich mit Elektronen bestrahlt (Abb.14).

Die Proben lassen sich in der Vakuumkammer auf einer entsprechenden Halterung in alle Richtungen bewegen und verkippen. Bei den derzeit angebotenen Geräten liegt die Auflösung bei ca. 7-10 nm. Die Ionenströme können durch Blenden im Durchmesser und der Stromstärke variiert werden und liegen im Bereich von 10 pA bis 1nA. Damit sind die erreichbaren Sputterraten relativ gering. Durch eine gezielte Zuführung von Ätzgasen (Br, J oder Cl) lassen sich Querschnitte durch vielschichtige Oberflächenstrukturen auch mit einer gewissen Selektivität ätzen (siehe Abb.15, 16 u. 17).

"Cut and wire", also das Auftrennen und wieder Verbinden von Leiterbahnen in verschiedenen Ebenen ist eine besonders interessante Anwendung dieses Verfahrens. Es können hiermit an fertigen Bauelementen nachträglich Korrekturen der internen Verdrahtung vorgenommen werden. Die Verifikation von Layout-Fehlern, die Überprüfung der Abhilfemaßnahmen oder der Einbau von zusätzlichen Funktionen können bei der schnellen Realisierung von Prototypen sehr hilfreich sein (Ablauf und Beispiel siehe Abb.18 u. 19).

CMOS Inverter

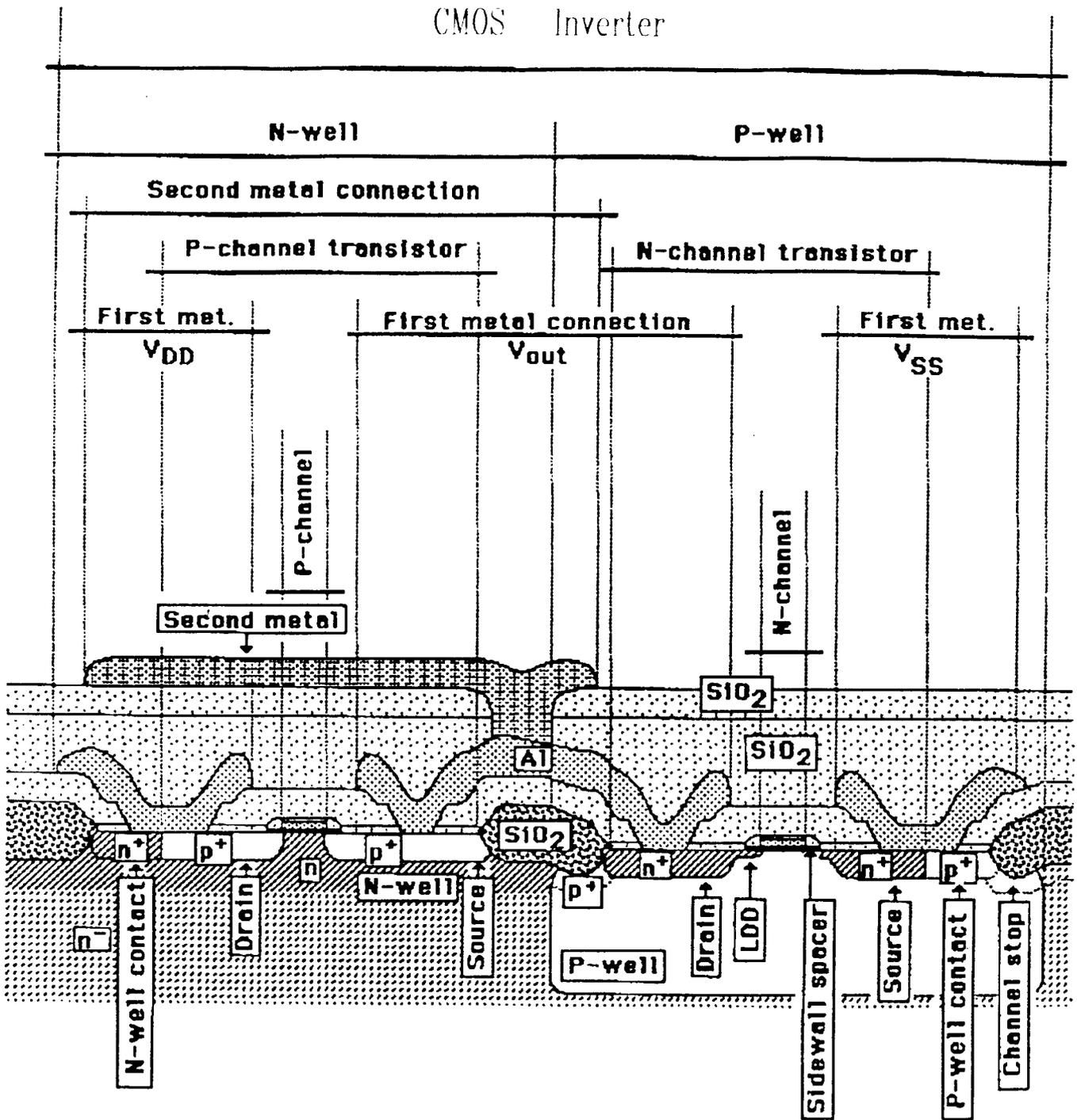


Abb. 1

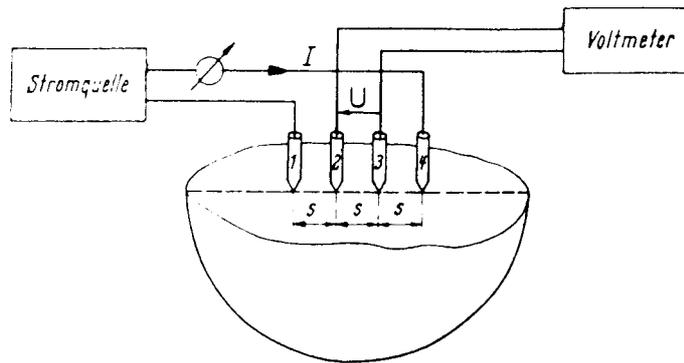


Abb. 2 Vierspitzen-Meßanordnung (nach H. Salow u. a. 1963)

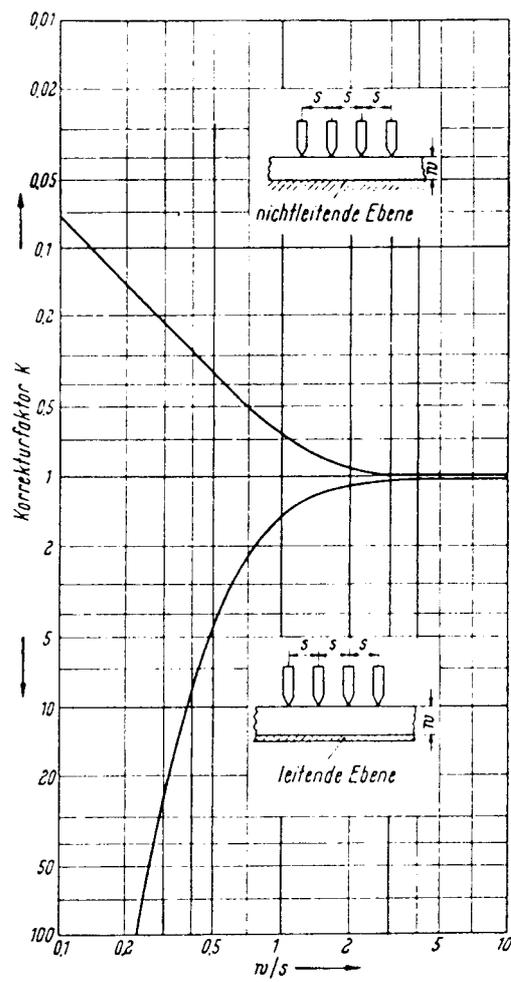


Abb. 3 Korrekturfaktoren für Schichtwiderstandsmessungen nach der Vierspitzen-Methode (nach H. Salow u. a. 1963)

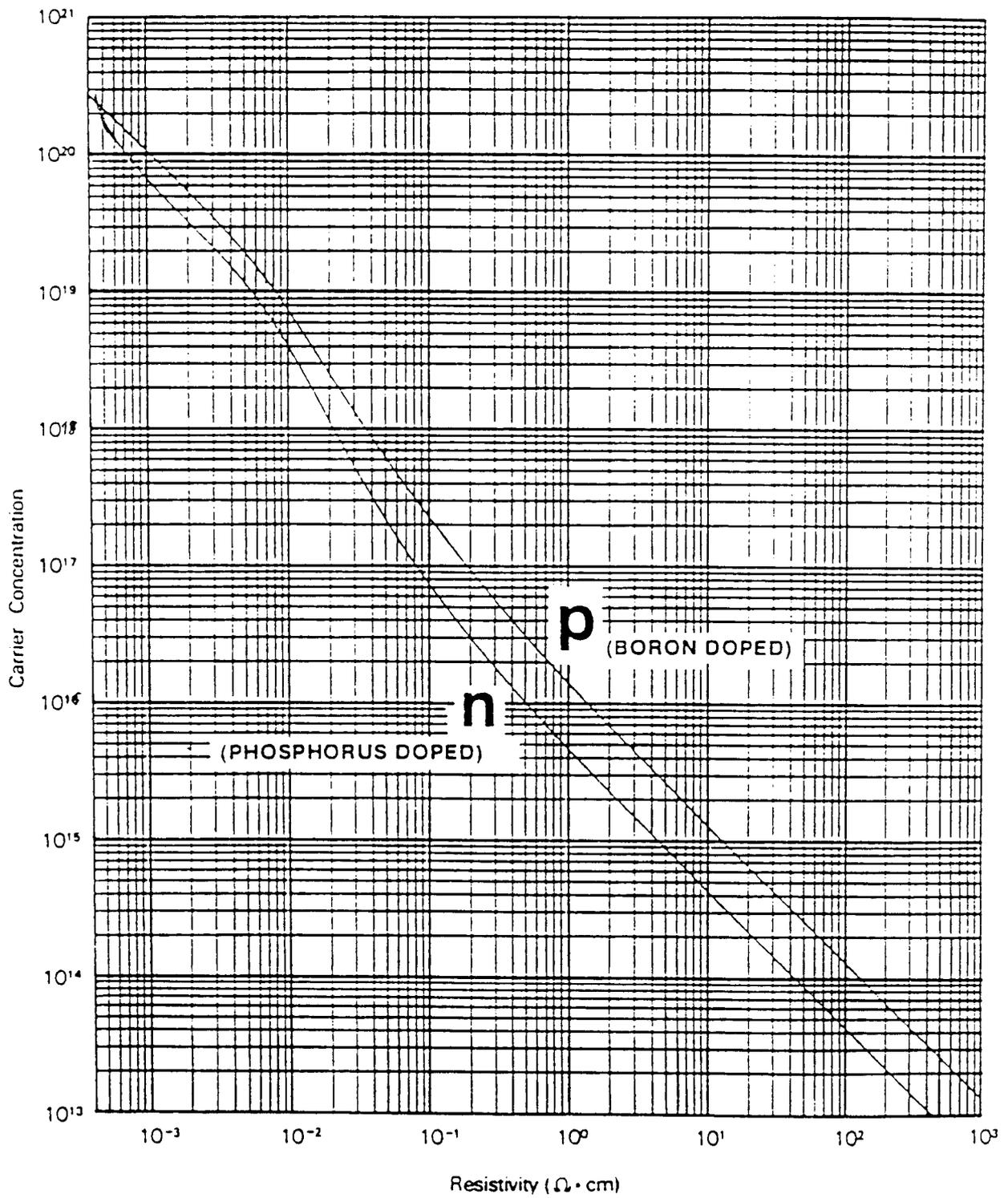


Abb. 4 Silicon carrier concentration versus resistivity. (After Thurber et al., *Natl. Bur. Standards Spec. Publ. 400-64*, May 1981, tables 10 and 14.)

Wafer ID: 8E14 Nr.7
 Lot ID: Ann 1100C, 20sec
 Process Step: RTA-Ann
 Equipment ID: NV10
 Date/Time: 10-MAY 08:34
 Operator ID: Riekeles

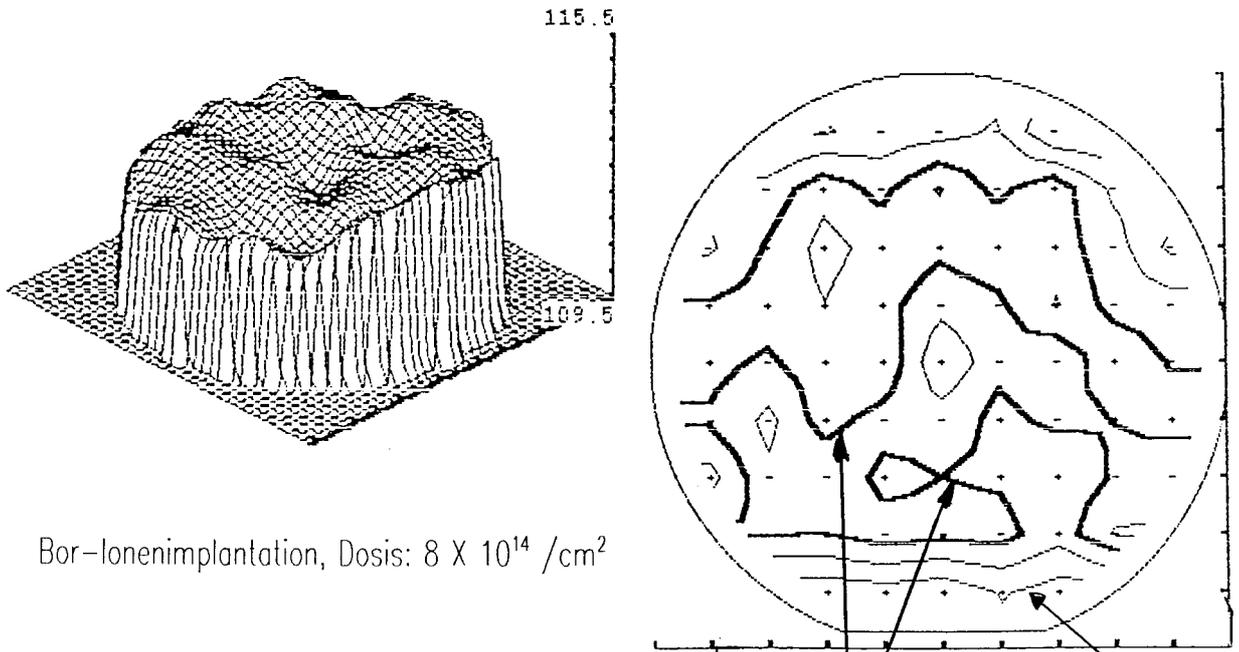
Wafer Size: 4in - 100 mm
 Probe Pattern: Full wafer -
 Edge Distance: 5
 Wafer Type: P - Type
 Temperature: 23

Total Points: 69
 Good Points: 69
 Minimum: 111.3676
 Maximum: 113.4524
Average: 112.4596
 Median: 112.4394
Standard Dev.: 0.3759

Units: Sheet (Ohms/sq)
 Edge Corrected: Yes
 Temp. Corrected: No
 Probe Head ID: TC 150um 100g
 Probe Cal Factor: 1
 Probe Corrected: No

Process Limits - Target: None
 Hi Valid: None Hi Spec: None Hi Control: None
 Lo Control: None Lo Spec: None Lo Valid: None
 Data Filtered between Hi and Lo Valid: No

Surface Plot - Rotation: 45 Tilt: 30
 Vertical Scale - Min: 109.5 Max: 115.5 Base: 109.5



Bor-Ionenimplantation, Dosis: $8 \times 10^{14} / \text{cm}^2$

Contour Values 111.7448 112.1022 112.4596 112.817 113.17
 Target Value: 112.4596 (Average) Interval: 5 Lines

Abb.5 Messung des Schicht-
 widerstands (Wafer-Map)

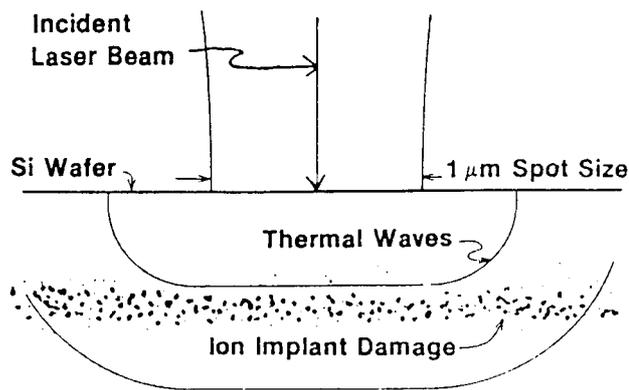
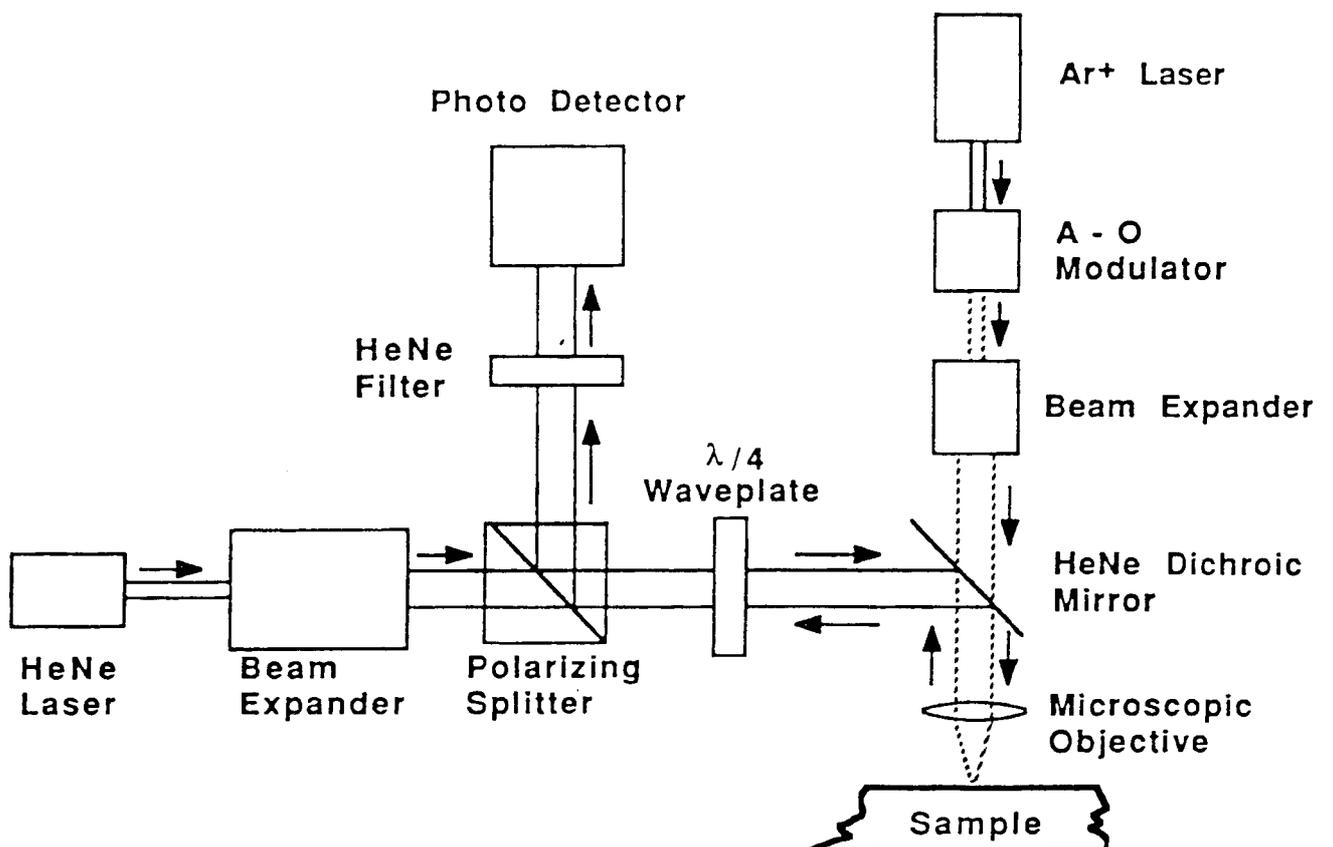


Abb. 6 Prinzip der Anregung von Wärmewellen

Abb. 7 Aufbau des Meßgeräts: "Thermaprobe"



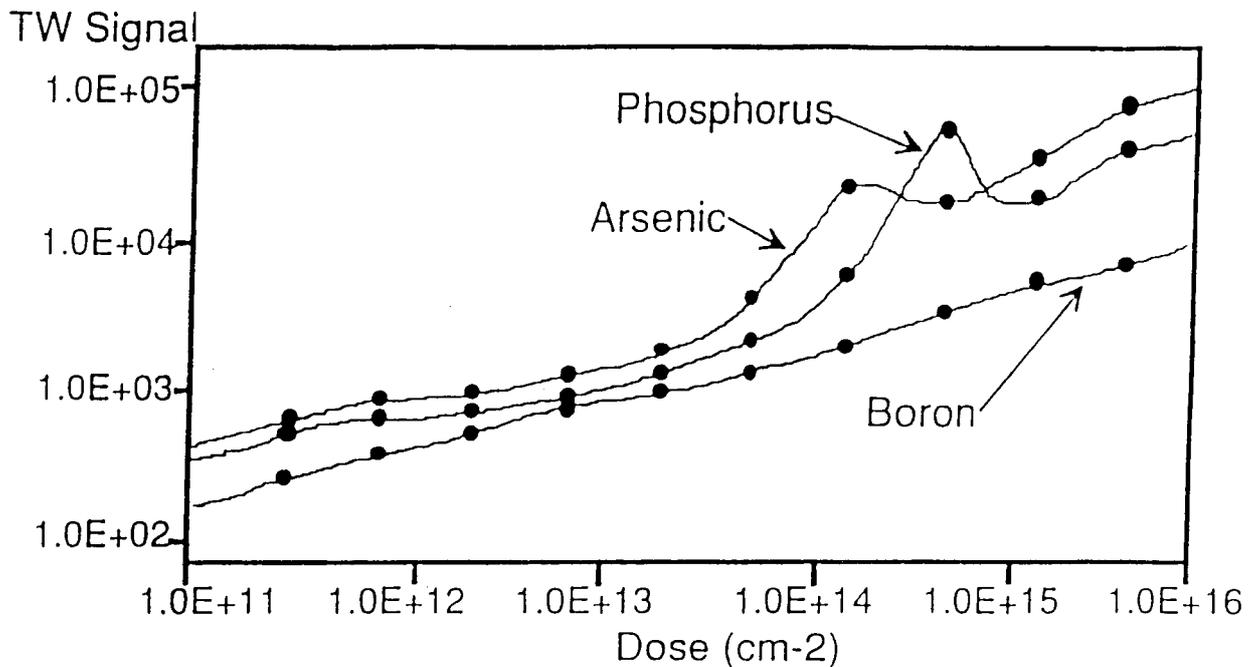
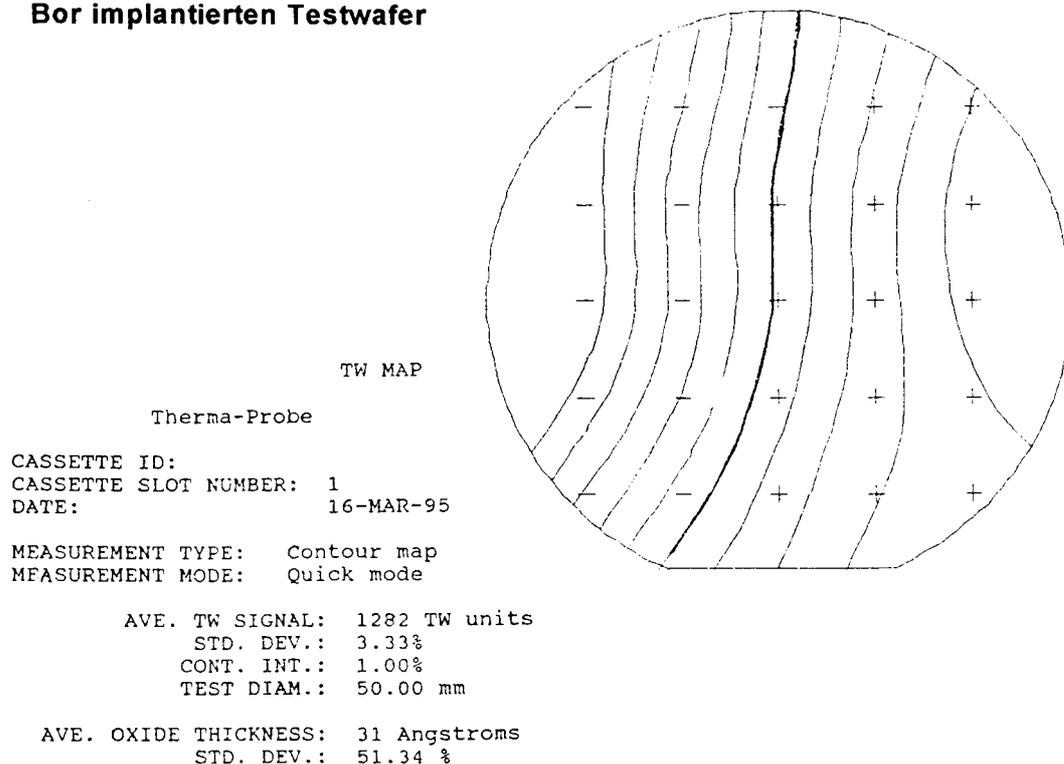


Abb. 8 TW-Meßwerte in Abhängigkeit von der implantierten Dosis

Abb. 9 Verteilung der TW-Meßwerte auf einem Bor implantierten Testwafer



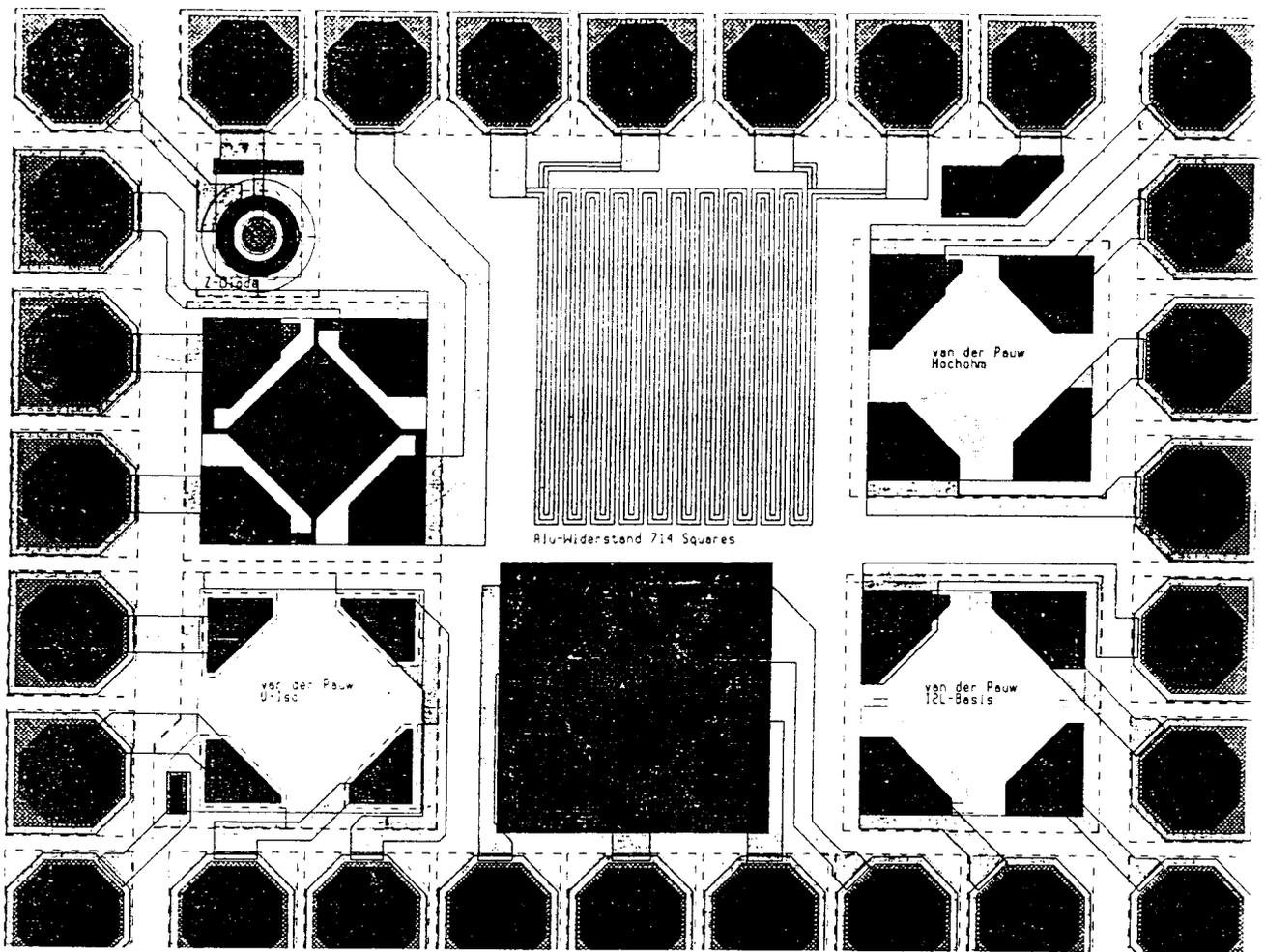


Abb. 10 Teststruktur zur Bestimmung der Schichtwiderstände bei einem Bipolar - Prozeß (SIEMENS - NEL)

Keithley Meßprogr.: G400NELxx

Testreihenfolge: L - S - A - N

Meßkarte: G40x

| Pos | Parameter | Struktur- name | Block | Parameter- name | Meßbedingungen | Bewer- | Ver- | Para- | Para- | Ver- | Bewer- | Dim | Padbelegung | |
|-----|-------------------------------|-------------------|-------|--------------------|-----------------------------|--------|--------|--------|--------|--------|--------|-------|--|--|
| | | | | | | tungs- | wurfs- | meter- | meter- | wurfs- | tungs- | | | |
| | | | | | | grenze | grenze | grenze | grenze | grenze | grenze | | | |
| | | | | | | Min | Min | Min | Ziel | Max | Max | Max | | |
| 1 | NPN Stromverstärkung | N1 | L | B | $U_{CE} = 5V, I = 500\mu A$ | 20 | 70 | 70 | 110 | 200 | | 300 | C: 22 / B: 21 / E: 23 / S: 20 | |
| 2 | NPN $B_{Linearität}$ | N1 | L | LIN | $I = 20 / 200\mu A$ | 0.80 | 0.90 | 0.90 | 0.98 | 1.02 | | 1.10 | C: 22 / B: 21 / E: 23 / S: 20 | |
| 3 | Basis/Emitter - Pinchwid. | RDE | N | RNP | $I = 33\mu A, SZ = 2.96$ | 1.0 | | 2.0 | 3.7 | 6.0 | | 9.0 | $k\Omega / \square$ I+: 10, 26 / L: 1, 11 | |
| 4 | Basisschichtwiderstand | KD | A | RSB | $I = 4.53mA$ | 150 | 165 | 170 | 175 | 195 | 195 | 220 | Ω / \square I+: 23 / L: 22, 19 / U+: 24 | |
| 5 | Basisschichtwiderstand | RD | L | SQZ | $I = 1mA, RB / RSB$ | 10.0 | | 13.0 | 15.0 | 17.0 | | 20.0 | I+: 3, 4 / L: 28, 20 | |
| 6 | Emitterwiderstand | RE | N | RNE | $I = 10mA, SZ = 20.7$ | 5.0 | 7.5 | 7.5 | 8.5 | 9.5 | | 10.5 | Ω / \square I+: 1 / L: 6 | |
| 7 | NPN Flußspannung U_{BE} | N1 | L | UBE | $I_{BE} = 50\mu A$ | 0.640 | | 0.655 | 0.675 | 0.675 | | 0.690 | V I+: 21, 22 / L: 23 | |
| 8 | Sperrspannung U_{CE0} | N1 | L | UCE | $I_{CE} = 500\mu A$ | 15.0 | 20.0 | 20.0 | 23.0 | 30.0 | | 40.0 | V I+: 22 / L: 23, 20 | |
| 9 | Sperrspannung U_{CB0} | N1 | L | UCB | $I_{CB} = 10\mu A$ | 20.0 | | 35.0 | 45.0 | 55.0 | | 70.0 | V I+: 22 / L: 21, 20 | |
| 10 | Kollektorwid. ROCT | VER | N | ROC | $I = 10mA$ | 25.0 | | 30.0 | 36.0 | 42.0 | 42.0 | 50.0 | Ω I+: 20 / L: 19, 11 / U+: 18 | |
| 11 | Hochohmwiderst. NEL 1 | RR | N | RNH | $U = 2V, SZ = 8.2$ | 1.50 | 1.70 | 1.80 | 2.00 | 2.20 | 2.30 | 2.70 | $k\Omega / \square$ U+: 1, 26 / L: 27, 11 | |
| | Hochohmwiderst. NEL 2 | RR | N | RNH | $U = 2V, SZ = 8.2$ | 1.50 | 1.70 | 1.80 | 2.00 | 2.20 | 2.30 | 2.70 | $k\Omega / \square$ U+: 1, 26 / L: 27, 11 | |
| 12 | lat. PNP Stromverstärkg. | PL | L | B | $U_{CE} = 5V, I = 50\mu A$ | 30 | 50 | 50 | 70 | 90 | | 130 | C: 16 / B: 15 / E: 14 / S: 20 | |
| 13 | Sperrspannung U_{CE0} | PL | L | UCE | $I_{CE} = 50\mu A$ | 20.0 | 30.0 | 30.0 | 40.0 | 50.0 | | 70.0 | V I+: 14 / L: 16, 20 | |
| 14 | PNP ohne Alu Stromverst. | PLO | L | B | $U_{CE} = 5V, I = 50\mu A$ | 10 | | 40 | 70 | 100 | | 150 | C: 11 / B: 13 / E: 12 / S: 20 | |
| 15 | Iso- PNP Stromverstärkg. | PLI | L | B | $U_{CE} = 5V, I = 50\mu A$ | 20 | | 50 | 110 | 200 | | 250 | C: 8 / B: 9 / E: 10 / S: 20 | |
| 16 | Iso- PNP Sperrspg. U_{CE0} | PLI | L | UCE | $I_{CE} = 50\mu A$ | 22.0 | | 30.0 | 38.0 | 45.0 | | 60.0 | V I+: 10 / L: 8, 20 | |
| 17 | vert. PNP Stromverstärkg. | PV | L | B | $U_{CE} = 5V, I = 50\mu A$ | 50 | | 200 | 350 | 600 | | 800 | C: 5 / B: 6 / E: 7 / S: 20 | |
| 18 | vert. PNP Stromverstärkg. | PV | L | B | $U_{CE} = 5V, I = 500\mu A$ | 50 | | 100 | 200 | 300 | | 400 | C: 5 / B: 6 / E: 7 / S: 20 | |
| 19 | Sperrspannung U_{CE0} | PV | L | UCE | $I_{CE} = 500\mu A$ | 20.0 | 30.0 | 30.0 | 40.0 | 50.0 | | 70.0 | V I+: 7 / L: 5, 20 | |
| 20 | Subst- PNP Stromverst. | PS | N | B | $U_{CE} = 5V, I = 50\mu A$ | 60 | | 100 | 140 | 200 | | 300 | C: 11 / B: 12 / E: 13 | |
| 21 | Subst- PNP Sperrsp. U_{CE0} | PS | N | UCE | $I_{CE} = 50\mu A$ | 20.0 | | 30.0 | 40.0 | 50.0 | | 70.0 | V I+: 13 / L: 11 | |
| 22 | Basiskontaktwiderstand | KWD | N | RKB | $I = 10mA$ | 1.00 | | 1.50 | 2.00 | 2.50 | 2.50 | 3.50 | Ω I+: 2 / L: 4 / U+: 5 / U-: 5 | |
| 23 | Übergangswid. A11/A12 | VIA | A | RGJ | $I = 100mA$ | 0.035 | | 0.045 | 0.055 | 0.065 | | 0.080 | Ω I+: 11 / L: 9 / U+: 10 / U-: 10 | |

Abb. 11 Testliste zur Messung der Bauelemente-Parameter

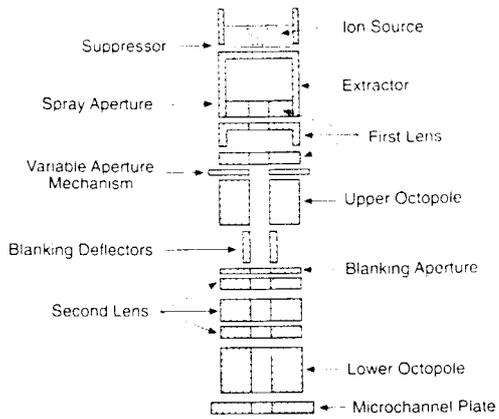


Abb. 12 Aufbau der Ionen-Strahl-Säule

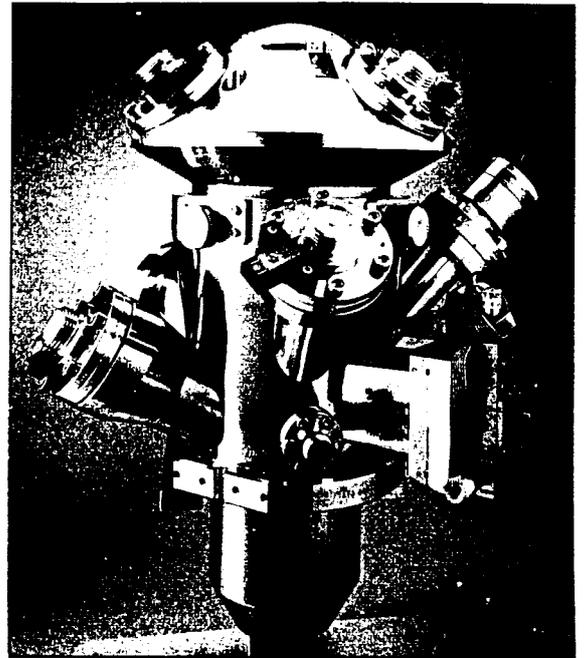


Abb. 13 Ionen-Strahl-Säule der Fa. Micrion

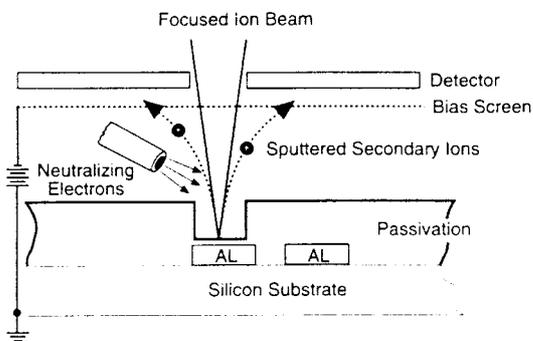
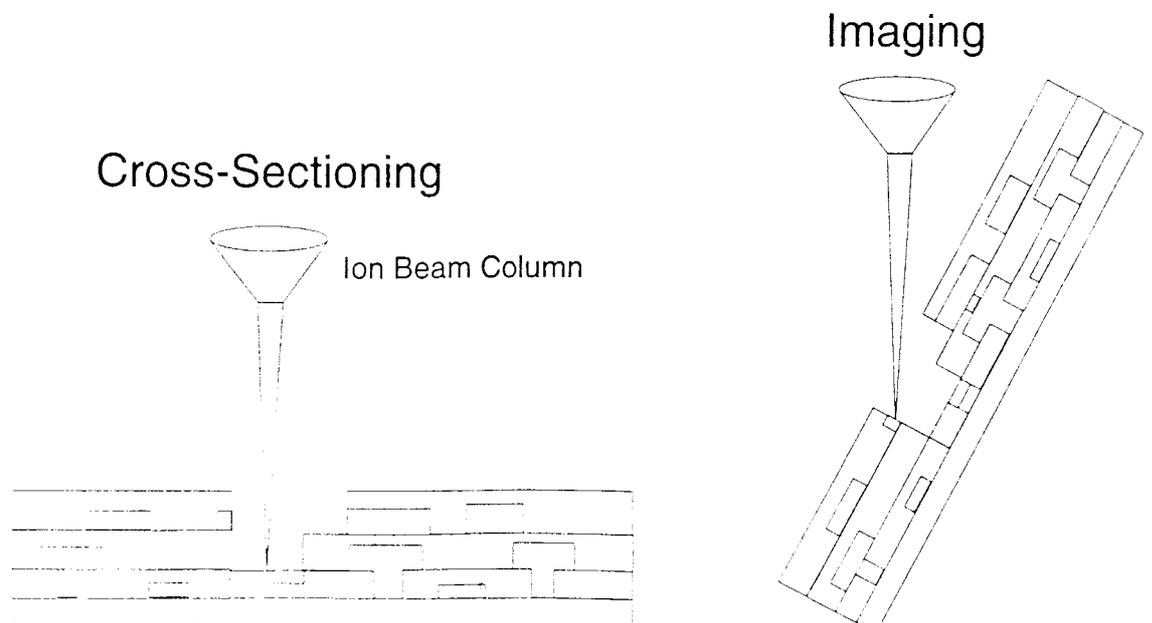


Abb. 14 Kompensation der Aufladung durch Elektronenbestrahlung

Abb. 15 Mit dem Ionenstrahl Querschnitte ätzen und darstellen



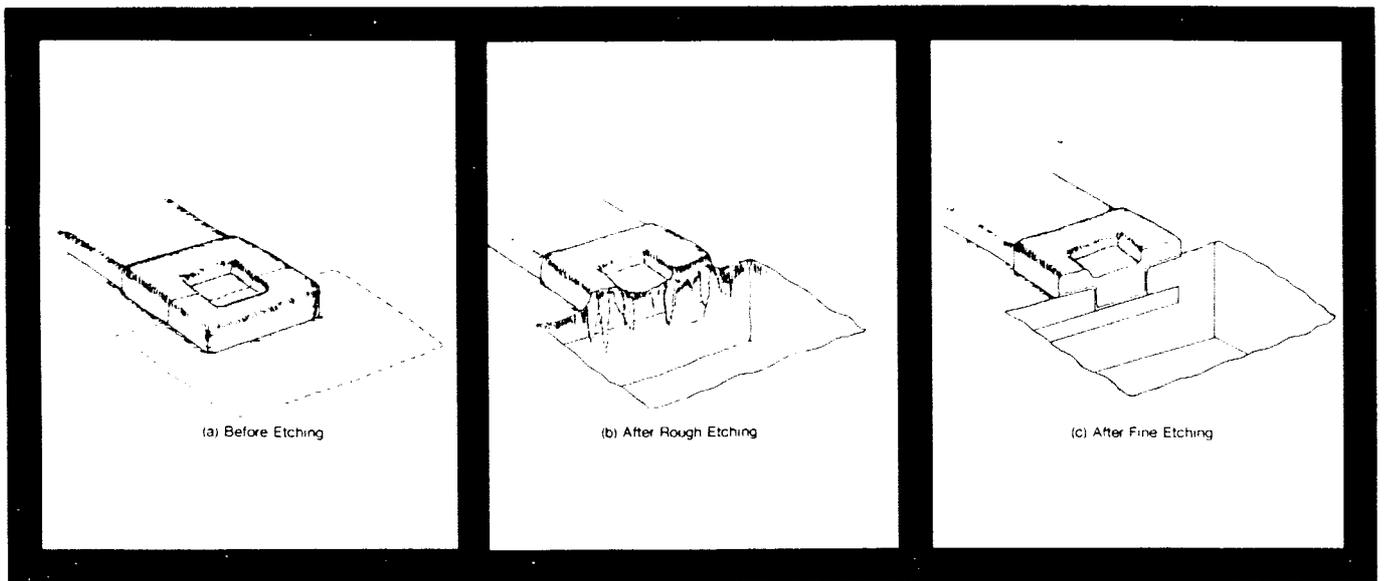


Abb. 16 Ablauf des Ätzprozesses

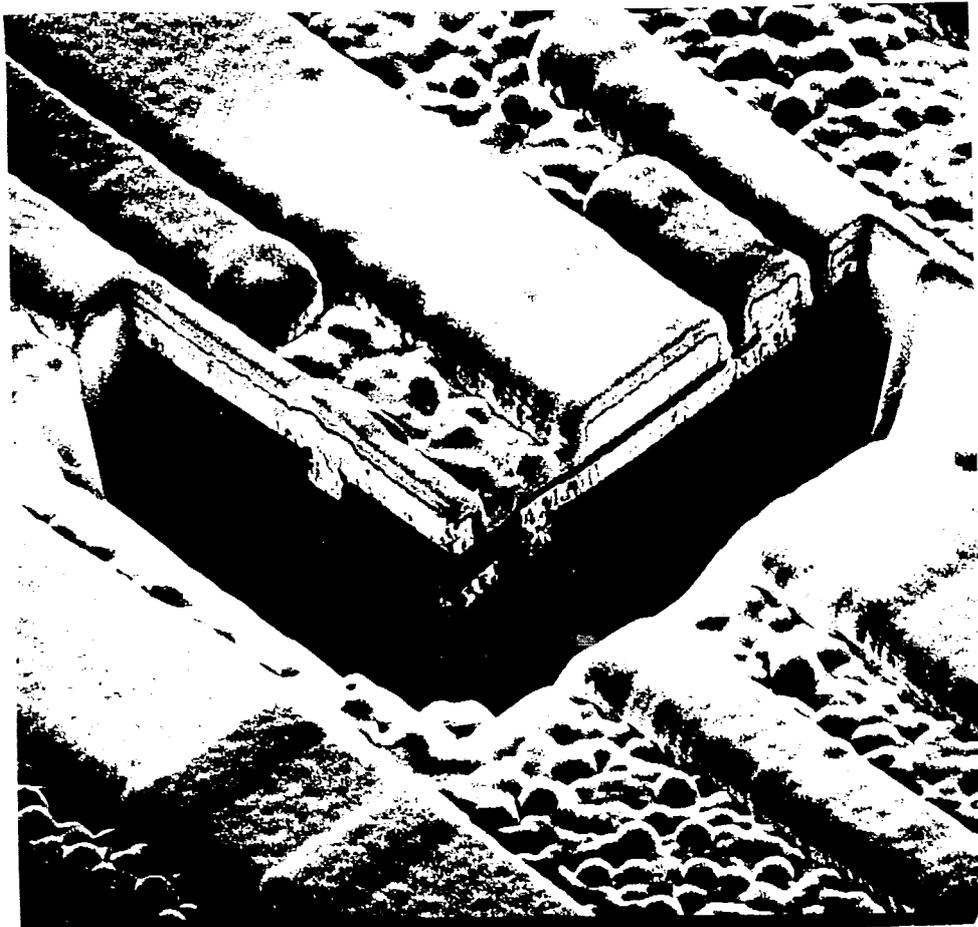


Abb. 17 Querschnitt durch eine mehrlagige Leiterbahnstruktur

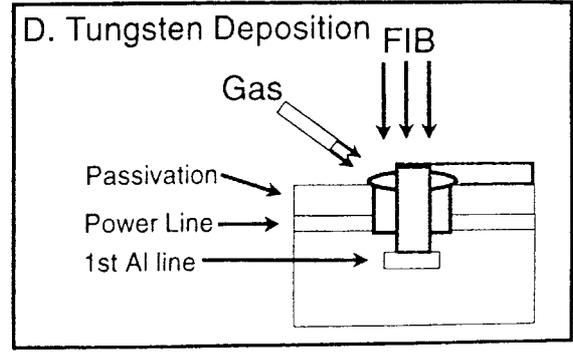
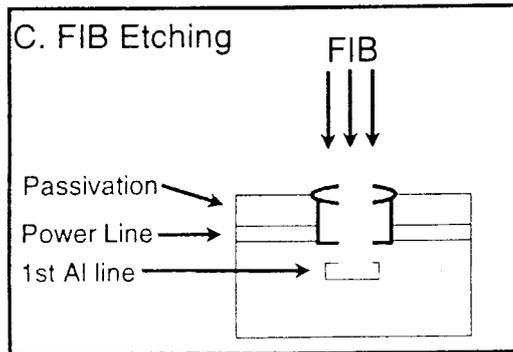
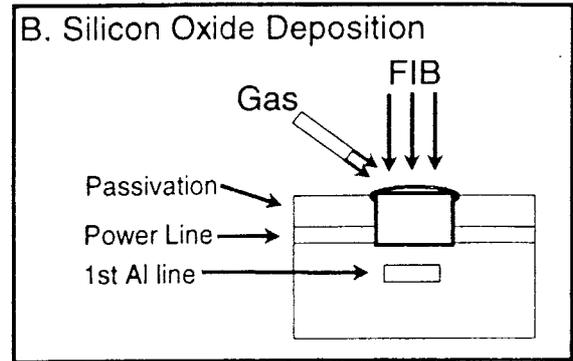
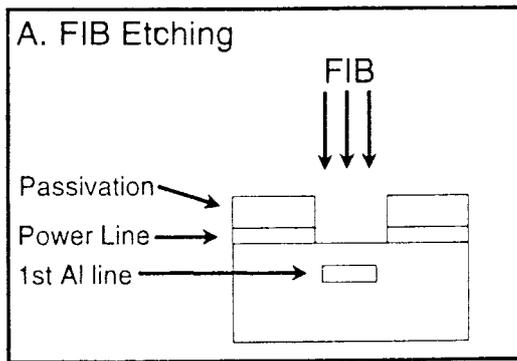


Abb. 18 Aufschneiden und verbinden von mehrlagigen Leiterbahnen
(cut and wire)



Abb. 19 Wolfram Abscheidung zur Kontaktierung zwischen isolierten
Metallebenen

Einsatz des BSIM3-Modelles zur Simulation von analogen und digitalen CMOS-Schaltungen im Submicron-Bereich

Prof. Dr.-Ing. H. Khakzar ¹⁾, Dipl.-Ing. (FH) Thomas Gneiting ²⁾

Fachhochschule Esslingen - Hochschule für Technik

¹⁾ FB Elektronik/Mikroelektronik, ²⁾ FB Nachrichtentechnik

1. Grundlagen des Modells

Das BSIM3-Modell (BSIM = Berkeley Short-Channel IGFET Model) der University of California at Berkeley wurde entwickelt, um der immer weiter fortschreitenden Miniaturisierung im Schaltungsdesign Rechnung zu tragen. Es läßt sich bis hin zu einer minimalen Kanallänge von $0.25\mu\text{m}$ einsetzen. Bei BSIM3 handelt es sich um ein physikalisch orientiertes Modell, das sehr stark von Geometrie und Herstellungsprozeß der MOS Transistoren abhängt. Das bedeutet, daß sehr starke Abhängigkeiten zwischen Dimensionierungsparametern wie Kanallänge L oder -weite W sowie von Prozeßgrößen wie der Oxidschichtdicke T_{OX} bestehen.

Die Modellparameter des BSIM3-Modelles haben alle eine physikalische Bedeutung und werden nicht nur wie bei vielen anderen Modellen zur reinen Kurvenanpassung verwendet. Die zweidimensionale Formulierung der Modellgleichungen ermöglicht die Berücksichtigung von Effekten, die durch die kurze Kanallänge und hohe Feldstärke bei modernen MOS Transistoren hervorgerufen werden. Die wichtigsten hiervon sind [HUAN94]:

- Schwellenspannungsreduzierung
- Ungleichmäßige laterale und vertikale Dotierung
- Reduzierung der Ladungsträgerbeweglichkeit
- Bulkeffekt
- Sättigung der Ladungsträgergeschwindigkeit
- Drain-induzierte Barriererniedrigung (DIBL)
- Kanallängenmodulation (CLM)
- Verringerung des Ausgangswiderstandes durch "Heiße Elektronen"
- Leitung im Bereich unterhalb der Schwellenspannung
- Parasitäre Widerstände an Source/Drain
- LDD-Regionen

2. Entwicklung von Extraktionsroutinen zur Bestimmung der Modellparameter

Das Labor Elektronik/Integrierte Elektronik der Fachhochschule Esslingen beschäftigt sich seit etwa einem Jahr mit der Modellierung von MOS-Transistoren mit dem BSIM3-Modell. Schon zu Beginn dieser Arbeiten stellte sich heraus, daß die auf dem Markt erhältlichen Extraktionsprogramme zur Modellparameterbestimmung oft auf Optimierungsstrategien beruhen und den physikalischen Charakter des BSIM3-Modelles nicht berücksichtigen. Daher wurden im Rahmen von Diplomarbeiten eigene Extraktionsroutinen unter dem HP Tool IC-CAP entwickelt. Diese Routinen basieren auf den Modellgleichungen und ergeben physikalisch sinnvolle Modellparameter, was bei dem Einsatz von Optimierungsstrategien nicht immer der Fall ist. Die Extraktionsroutinen für BSIM3 wurden mit den Daten von mehreren $0,5\mu\text{m}$ - Prozessen von verschiedenen Herstellern getestet.

[HUAN94] J. H. Huan, Z.H. Liu, M. C. Jeng, K. Hui, M. Chan, P. K. Ko and C. Hu:
"BSIM3 Manual (Version 2.0)", University of California, Berkeley, CA, 1994



FHTE

Fachhochschule für Technik Esslingen

Using the BSIM3 - Model for Extraction and Simulation of Submicron MOS Devices

Prof. Dr.-Ing. Haybatolah Khakzar
Dipl.-Ing.(FH) Thomas Gneiting

Fachhochschule für Technik Esslingen



Physical Effects of Submicron MOS Transistors, modeled in BSIM3

Threshold Voltage

- vertical and lateral non-uniform doping
- short channel effects
- narrow channel effects

Mobility

- mobility reduction due to vertical fields

Carrier Velocity Saturation

Drain Current

- bulk charge effect
- subthreshold conduction
- source/drain parasitic resistance

Output Resistance

- drain induced barrier lowering (DIBL)
- channel length modulation (CLM)
- substrate current induced body effect (SCBE)



Boundaries of Geometry

In [1] Measurements and Simulations within the following borders are reported:

| | |
|----------------------------------|-------------|
| Gate Oxid Thickness (T_{OX}) | 70 Å |
| Channel Length (L) | 0.25 - 50µm |
| Channel Width (W) | 1.5 - 50µm |

Scalability

- ⇒ Prediction of device behaviour of advanced devices (decrease of gate length, gate oxid) based on the parameters of the current process.
- ⇒ Due to the physical nature of the model, the dependencies of important geometry and process parameters are built in [1].

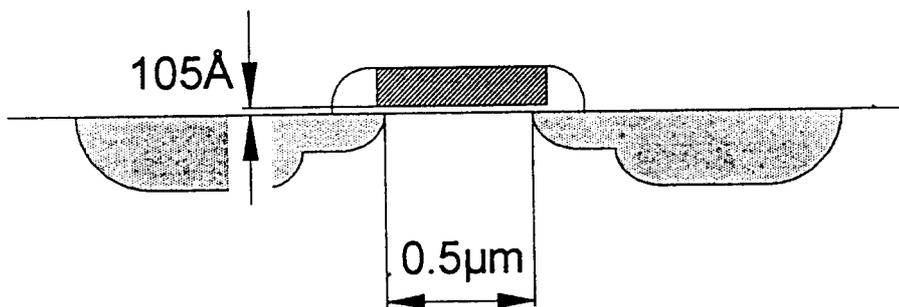
Statistical Purposes

- ⇒ Due to the physical meaning of many parameters, the BSIM3 model is ideal for statistical analysis of process fluctuations



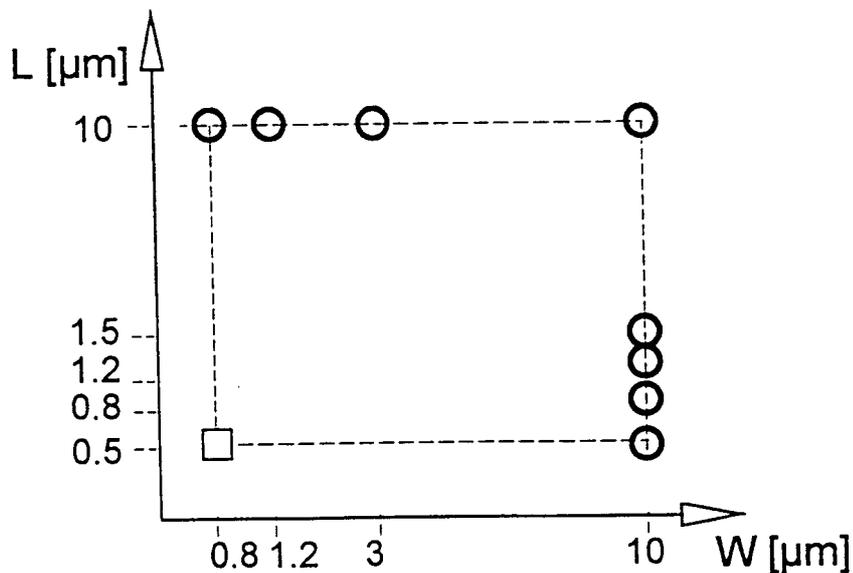
Parameter Extraction

Devices used for Measurements



0.5 μm CMOS Process with p-substrate and n-wells, n+ poly-Si gates and LDD structures

Geometry of measured Devices



**FHTE**

Fachhochschule für Technik Esslingen

Extraction Routines

⇒ No standardized IC-CAP routines have been available at the time, we started the work

For the parameter extraction, we used routines and calculations, following the extraction procedure recommended by University of Berkeley.

We do not use optimizations !

The following BSIM3 model parameters can be extracted with our routines:

Threshold voltage

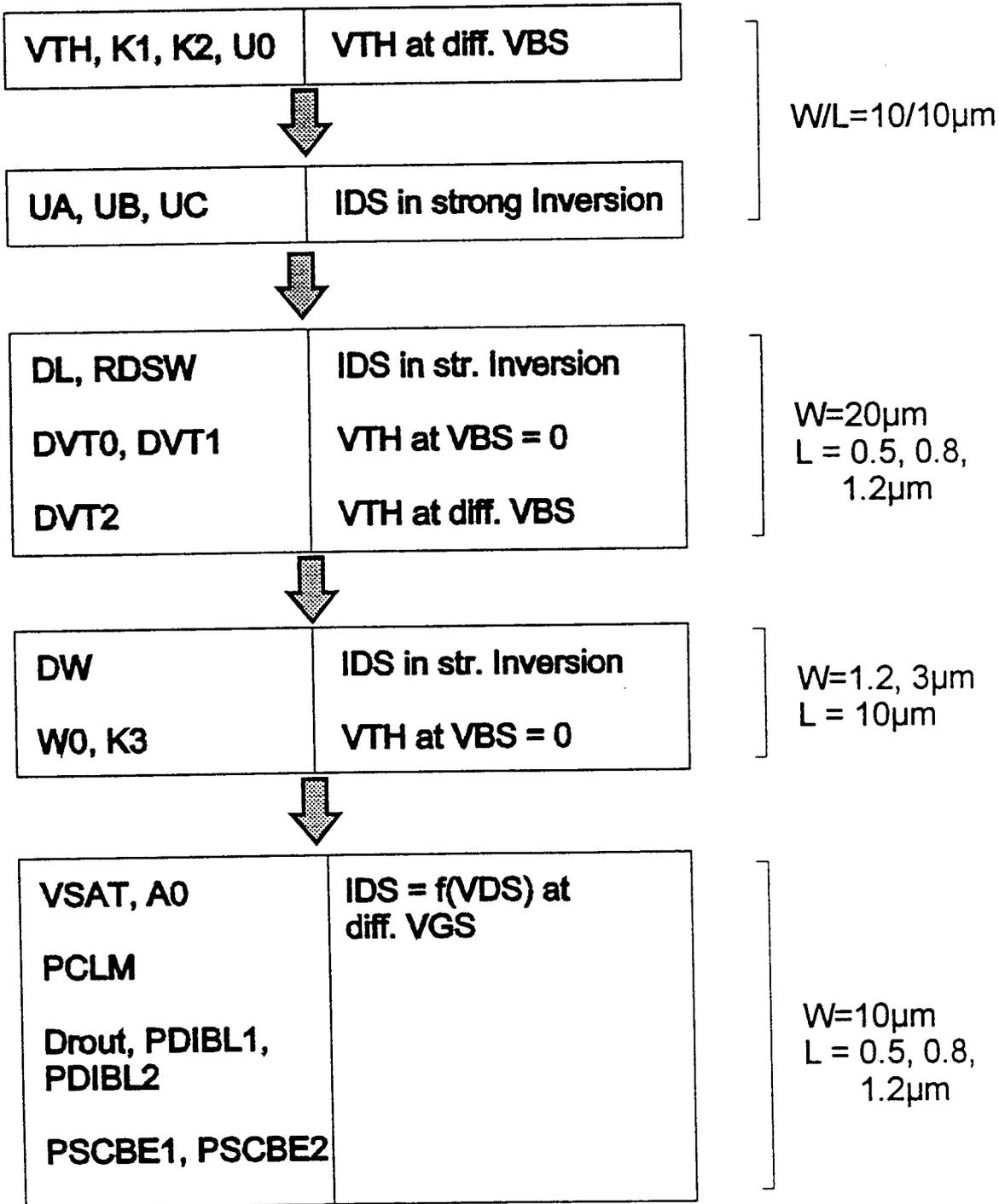
- | | |
|--------------------|-------------------------|
| • VTO, K1, K2 | Basic threshold voltage |
| • DVT0, DVT1, DVT2 | Short channel effects |
| • W0, K3 | Narrow channel effects |

Mobility

- | | |
|--------------|-------------------------|
| • U0 | Mobility |
| • UA, UB, UC | Degradation of mobility |



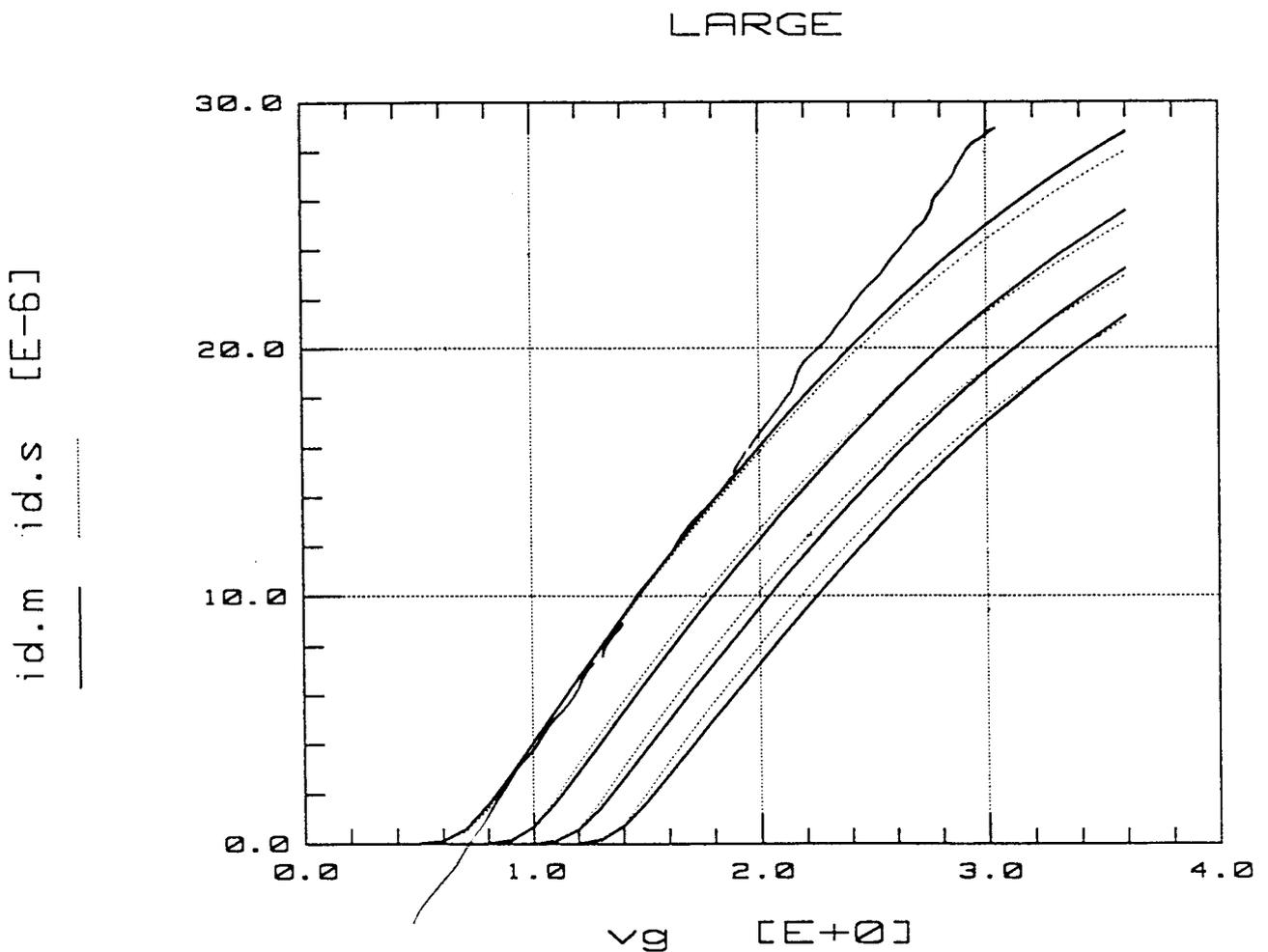
Flowchart of Extraction





Results

NMOS, W/L=10/10 μ m, $I_{DS} = f(V_{GS})$ at diff. V_{BS}



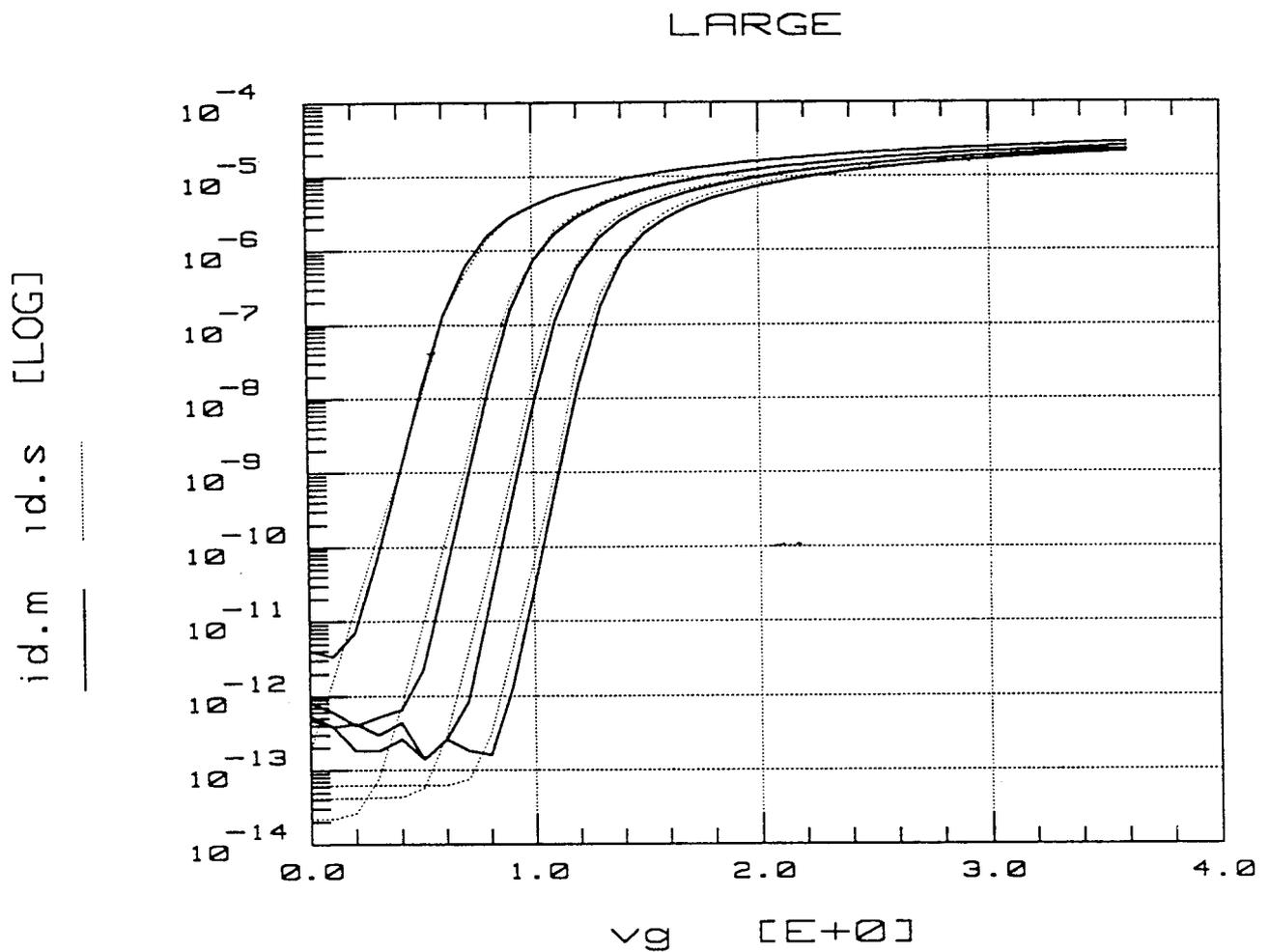
Mobility:

$$\mu^{\text{eff}} = \frac{\mu_0}{1 + UA \cdot ((V_{gs} + V_{th}) / T_{ox}) + UB \cdot ((V_{gs} + V_{th}) / T_{ox})^2 + UC \cdot V_{bs}}$$



Results

NMOS, W/L=10/10 μ m, $\log(I_{DS}) = f(V_{GS})$ at diff. V_{BS}



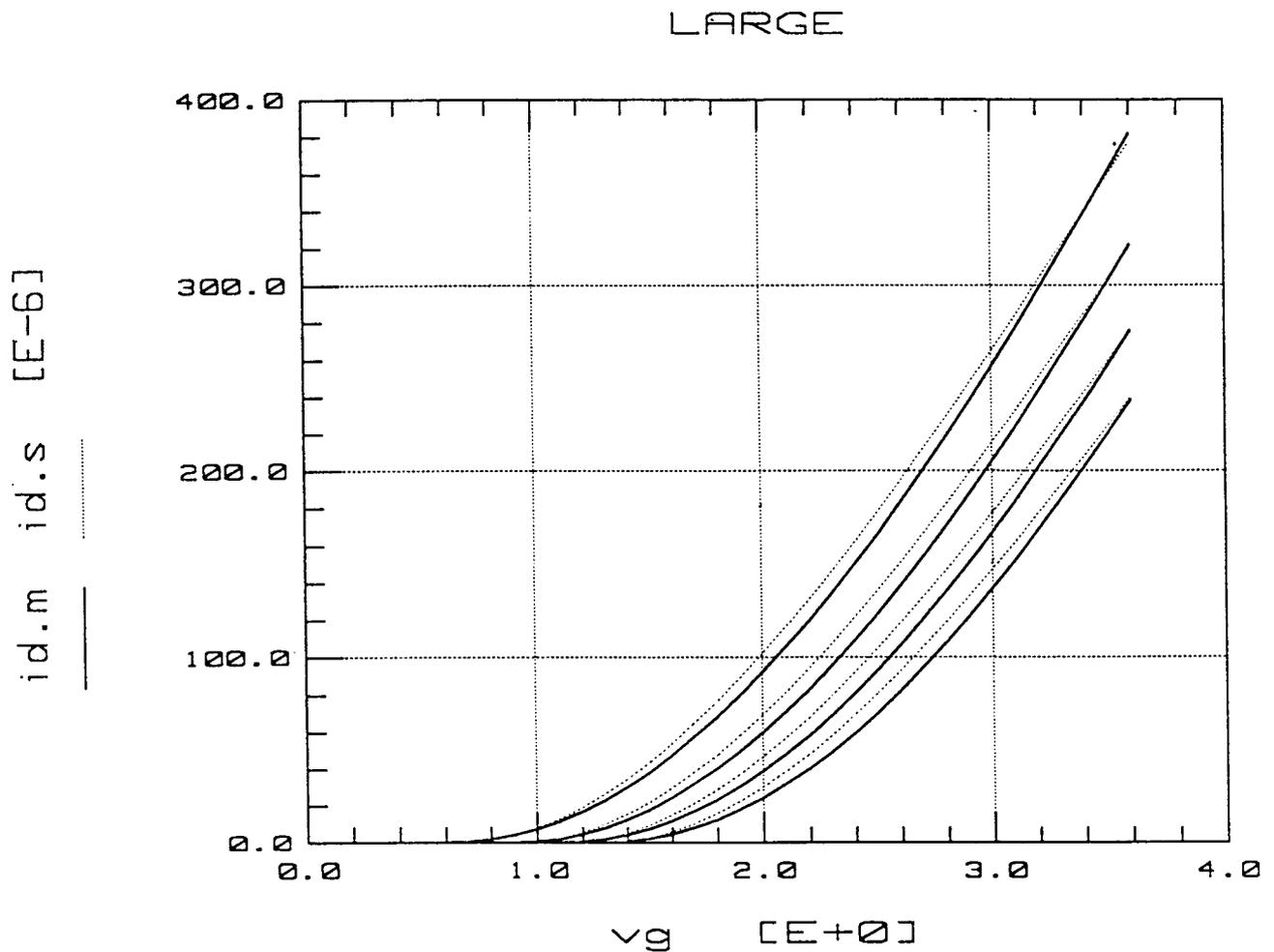
Subthreshold Current:

$$I_{ds0} = I_{s0} \cdot \left(1 - \exp\left(\frac{-V_{ds}}{V_{tm}}\right) \right) \cdot \exp\left(\frac{V_{gst} - V_{off} + \dots}{n \cdot V_{tm}}\right)$$



Results

NMOS, $W/L=10/10\mu\text{m}$, $I_{DS} = f(V_{GS})$ at diff. V_{BS}
and high Drain-Voltage V_{DS}



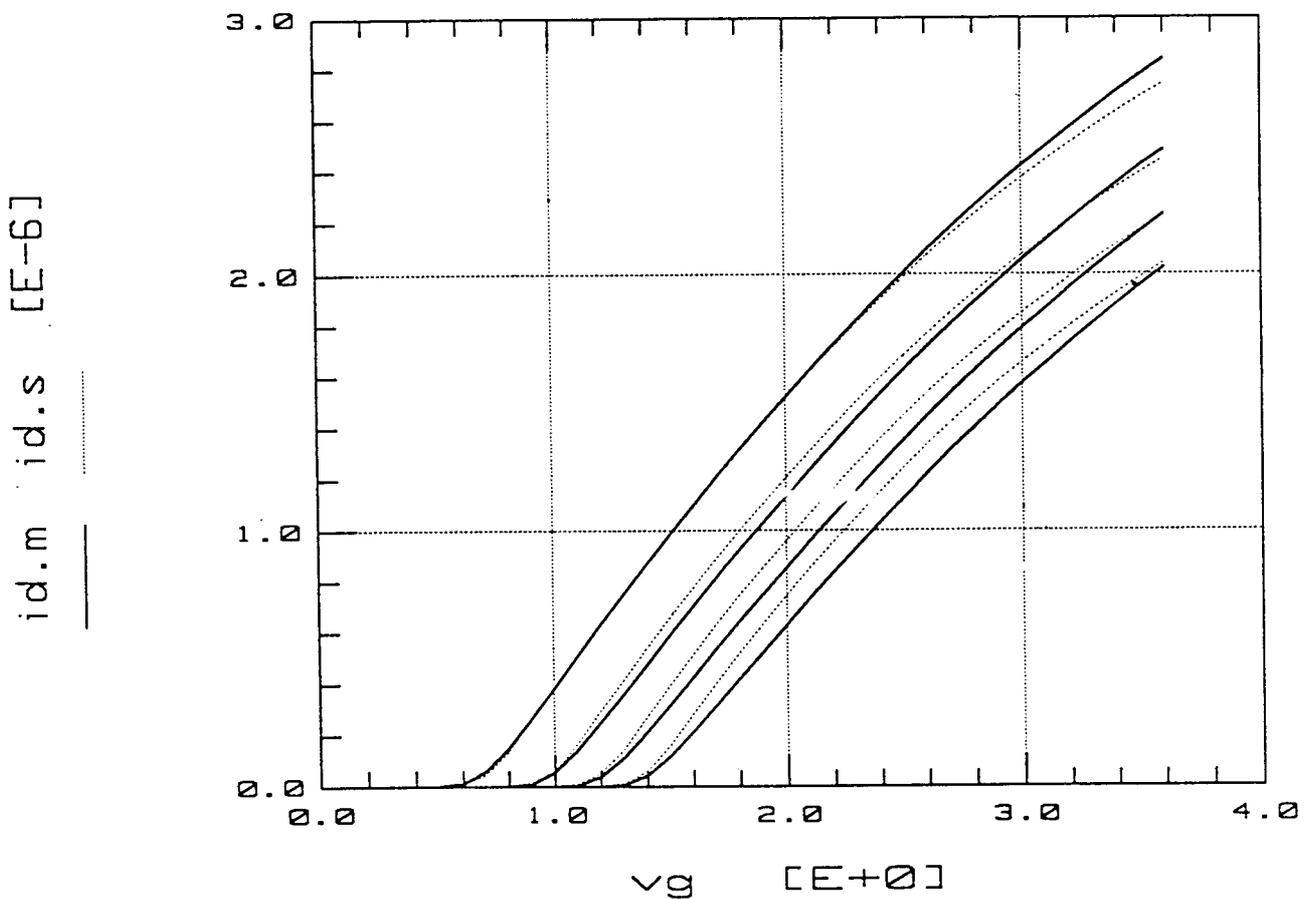
Bulk-Charge-Effect:

$$A_{\text{bulk}} = \left(1 + \frac{K_1}{2\sqrt{\phi_s - V_{bs}}} \cdot \frac{A_0 \cdot L}{L + 2 \cdot \sqrt{X_j \cdot X_{\text{dep}}}} \right) \cdot \frac{1}{1 + k_{\text{eta}} \cdot V_{bs}}$$



Results

NMOS, $W/L=0.8/10\mu\text{m}$, $I_{DS} = f(V_{GS})$ at diff. V_{BS}



Channelwidth reduction ΔW

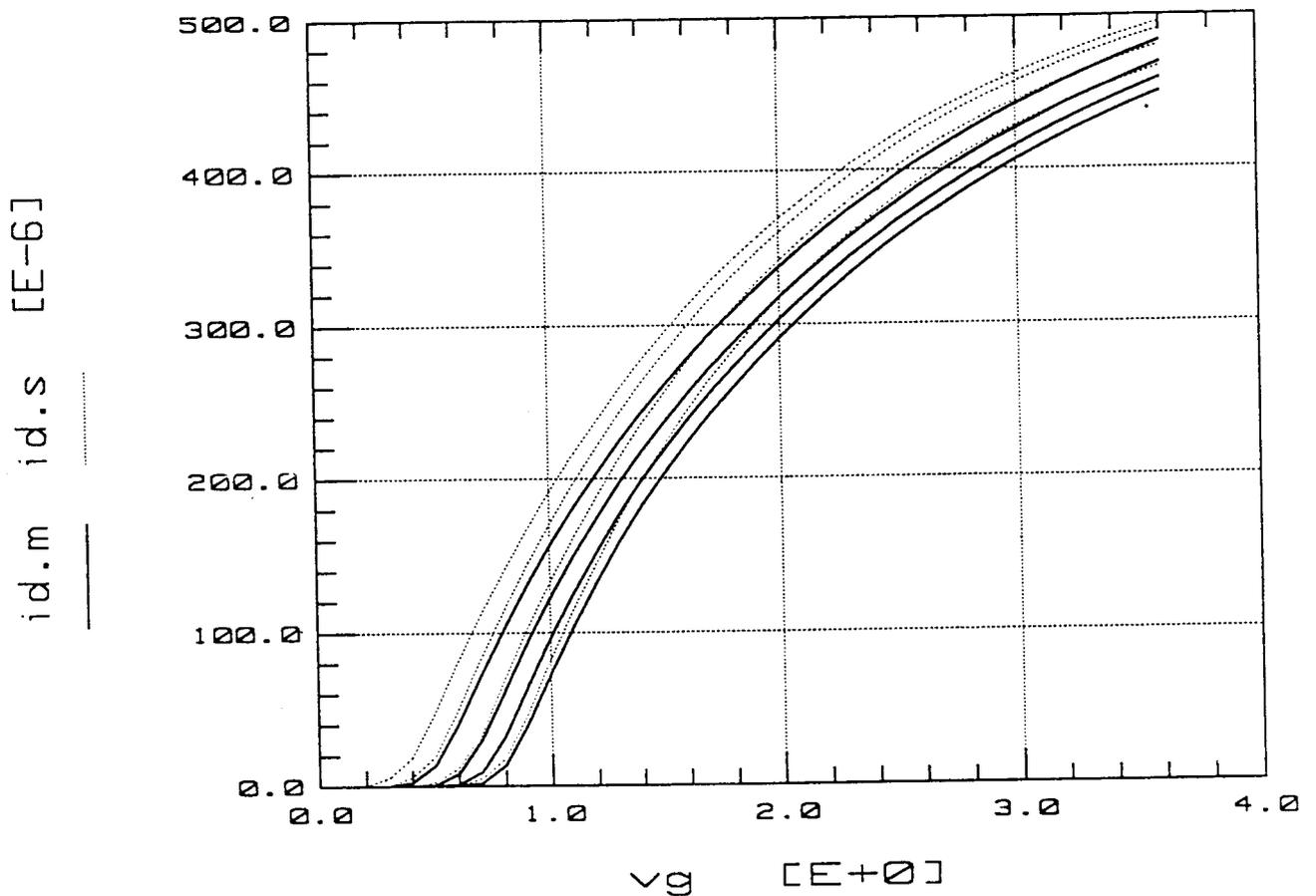
Narrow channel effects on threshold voltage

$$V_{th} = V_{Tideal} + \dots + K3 \cdot \frac{T_{ox}}{(W + W0)} \cdot \phi_s$$



Results

NMOS, $W/L=10\mu\text{m}/0.5\mu\text{m}$, $I_{DS} = f(V_{GS})$ at diff. V_{BS}



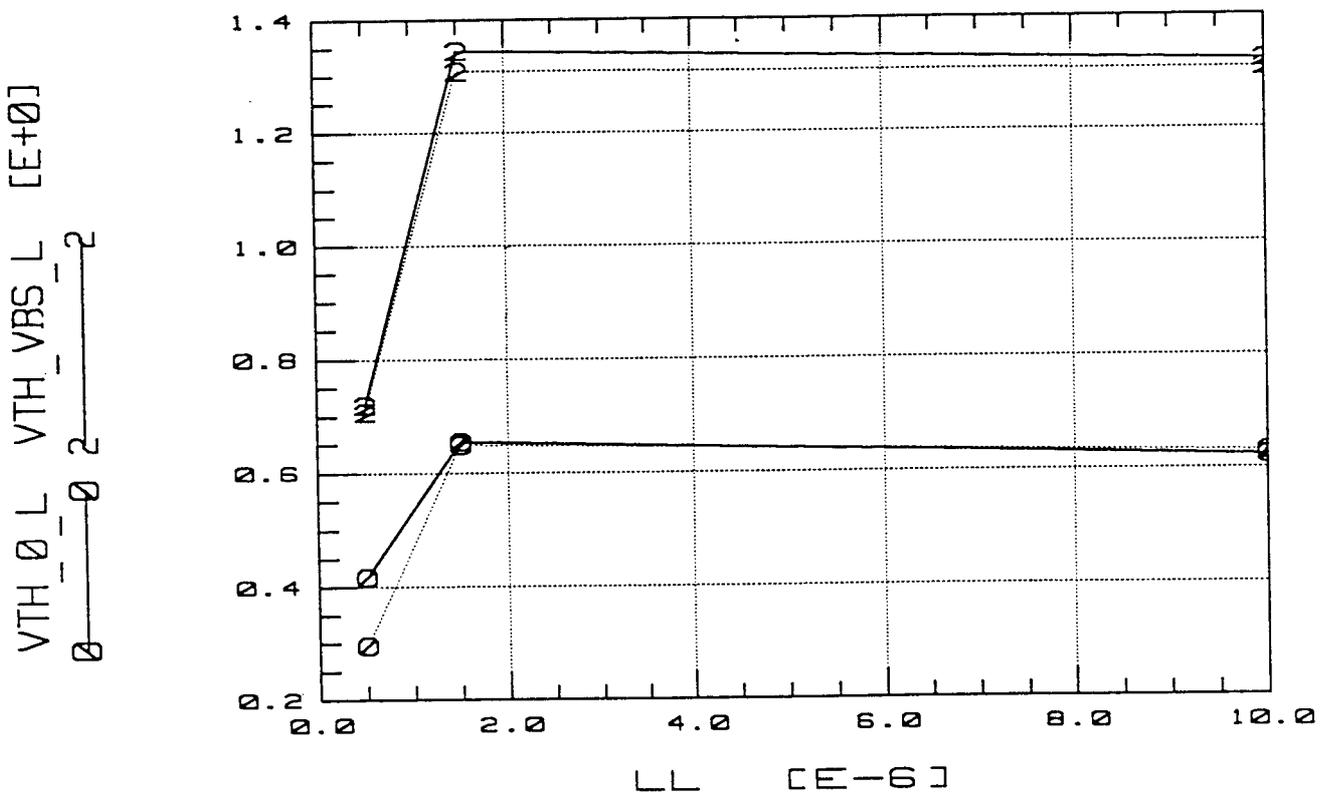
RDS and Channellength Reduction ΔL :

$$R_{DS} = R_{DS0} + \frac{R_{DSW}}{W}$$



Results

NMOS, $V_{TH} = f(L)$ at diff. V_{BS}



Threshold voltage:

$$V_{th} = V_{Tideal}$$

$$+ K1 \cdot (\sqrt{\phi_s - V_{bs}} - \sqrt{\phi_s}) - K2 \cdot V_{bs}$$

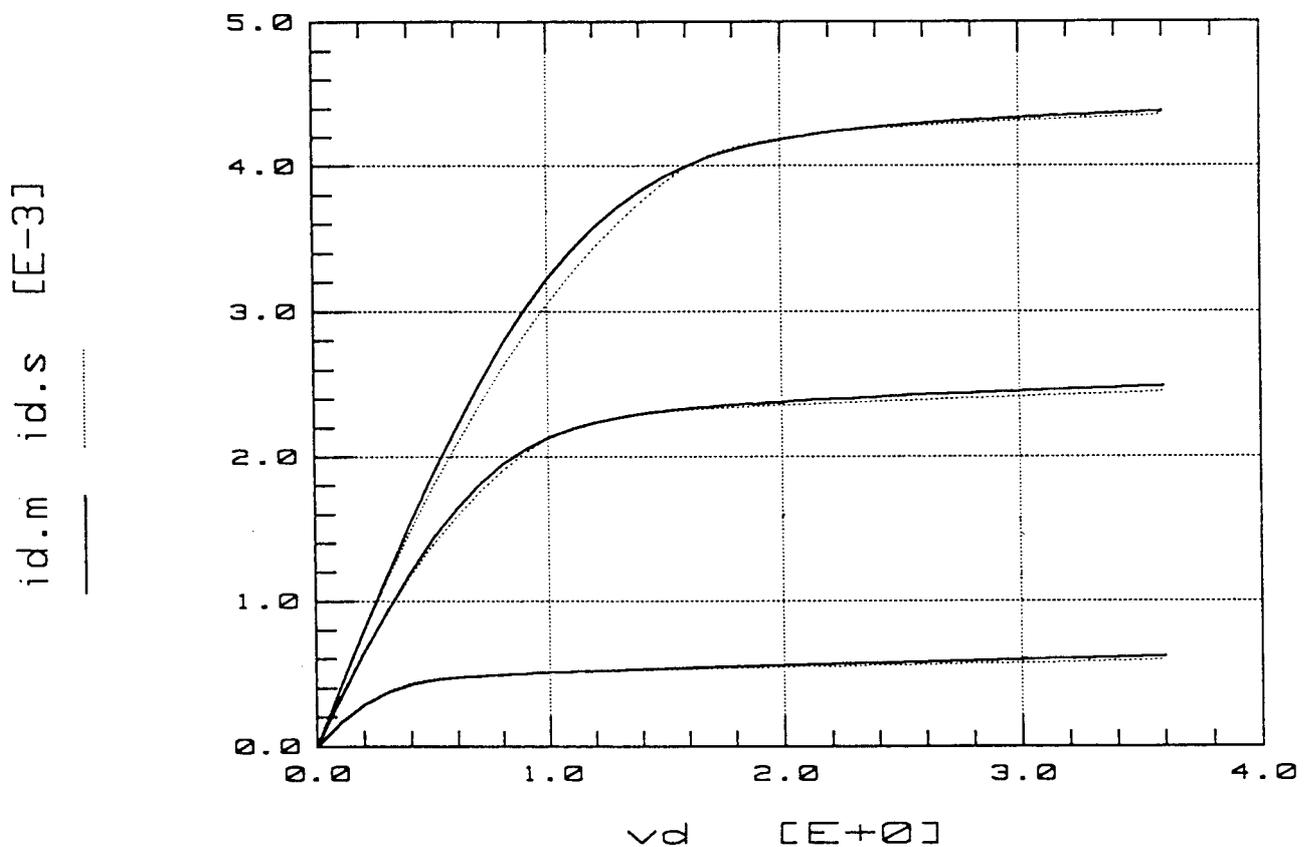
$$+ K1 \cdot \left(\sqrt{1 + \frac{N_A x}{L} \cdot \sqrt{\frac{\phi_s}{\phi_s - V_{bs}}}} - 1 \right) \cdot \sqrt{\phi_s} - \Delta U_{th}$$



Results

NMOS, W/L=10/0.5 μ m, $I_{DS} = f(V_{DS})$ at diff. V_{GS}

SHORT



Linear Region:

$$I_{ds} = \mu_{eff} \cdot C_{ox} \cdot \frac{W}{L} \cdot \frac{1}{1 + \frac{V_{ds}}{E_{sat} \cdot L}} \cdot \left(V_{gs} - V_{th} - \frac{A_{Bulk} \cdot V_{ds}}{2} \right) \cdot V_{ds}$$

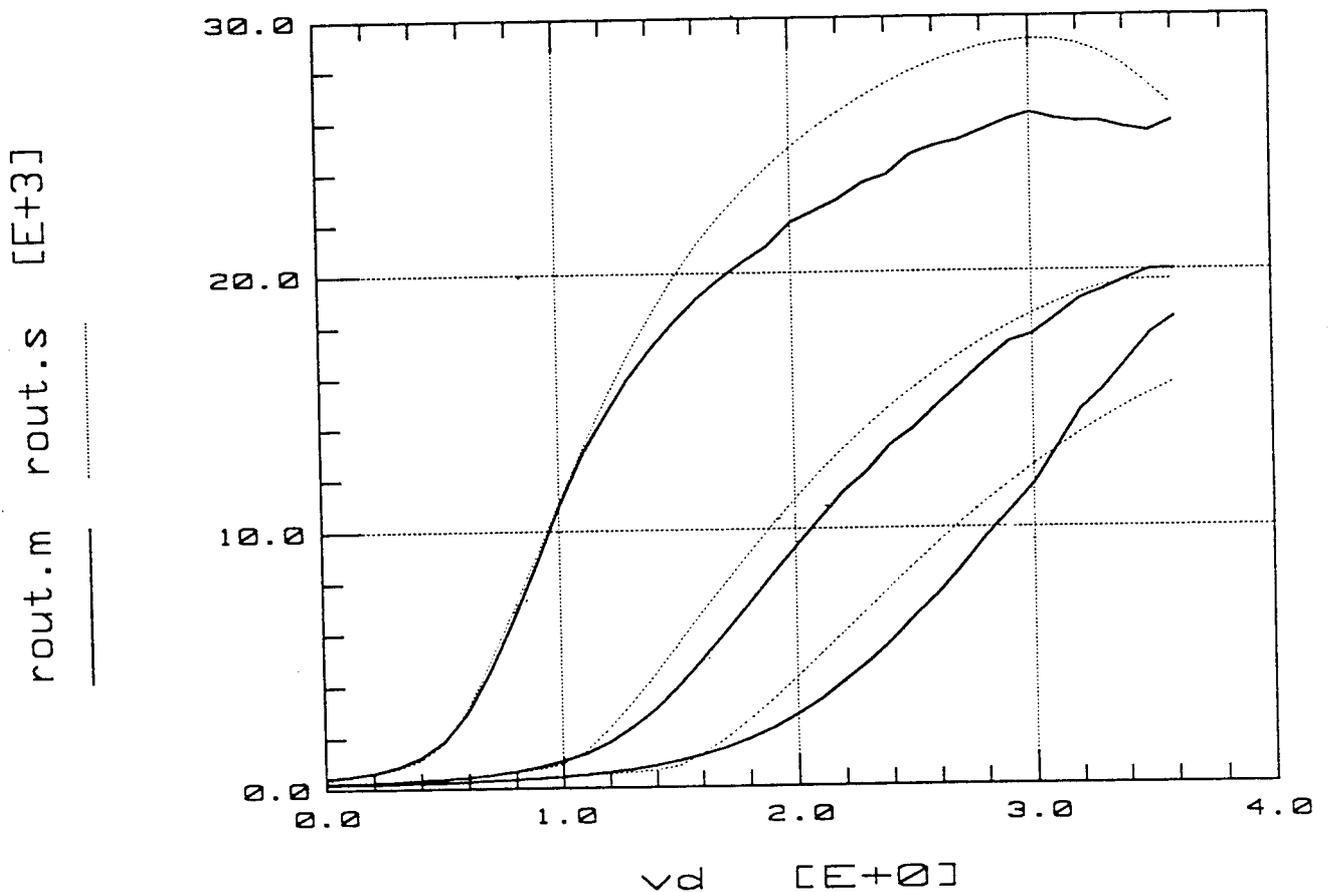
Saturation Region:

$$I_{dsat} = W \cdot v_{sat} \cdot C_{ox} \cdot (V_{gs} - V_{th} - A_{Bulk} \cdot V_{dsat})$$



Results

NMOS, W/L=10/0.5 μ m, $R_{out} = f(V_{DS})$ at diff. V_{GS}



Output Resistance:

$$I_{ds} = I_{dsat} \cdot \left(1 + \frac{V_{ds} - V_{dsat}}{V_A} \right) \cdot \left(1 + \frac{V_{ds} - V_{dsat}}{V_{ASCBE}} \right)$$

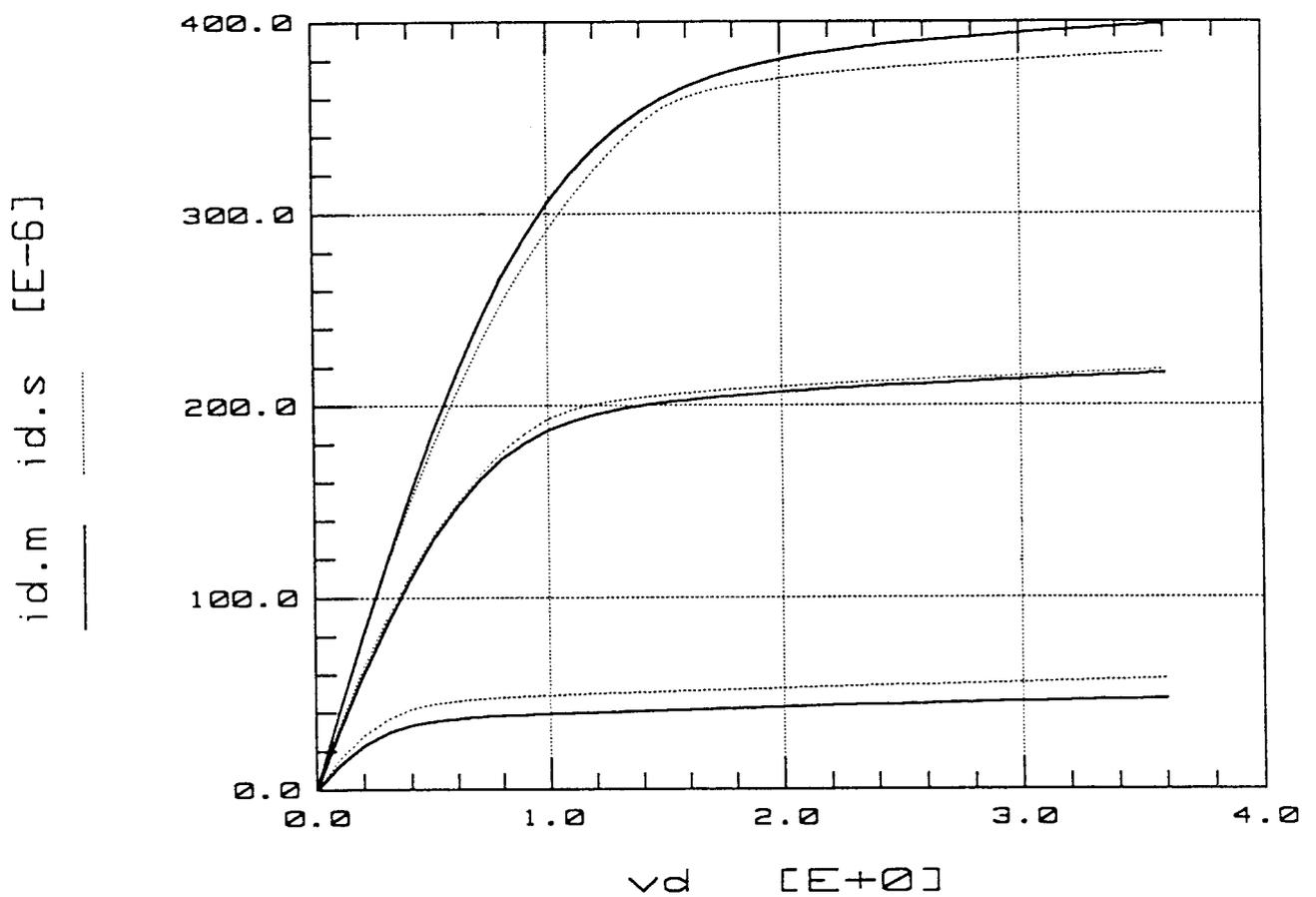
$$V_A = V_{Asat} + \left(1 + \text{eta} \cdot \frac{I_{Ldd}}{1} \right) \cdot \left(\frac{1}{V_{ACLM}} + \frac{1}{V_{ADIBL}} \right)^{-1}$$



Results

NMOS, $W/L=0,8/0.5\mu\text{m}$, $I_{DS} = f(V_{DS})$ at diff. V_{GS}

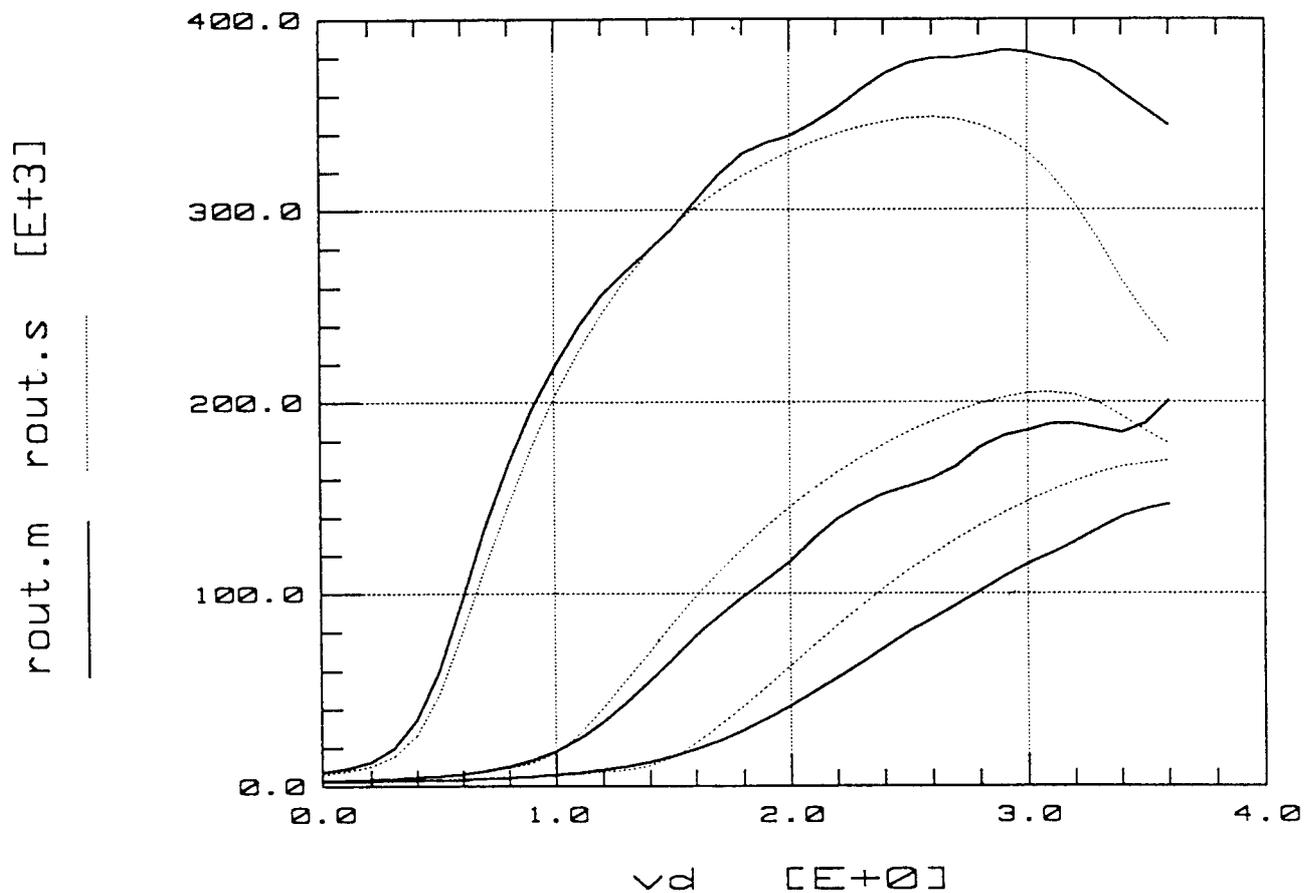
SMALL





Results

NMOS, $W/L=0,8/0,5\mu\text{m}$, $R_{\text{out}} = f(V_{\text{DS}})$ at diff. V_{GS}





Positive Aspects of BSIM3

- physical model
- valid in the deep submicron region
- number of parameters = 45 (less than BSIM1,2)
- BSIM3 includes geometrical dependencies of parameters
- usable for statistical purposes

BSIM3 shows a good behavior in the following diagrams:

- $I_D = f(V_{GS})$
- $\log(I_D) = f(V_{GS})$
- $I_D = f(V_{DS})$
- g_{DS}, g_M
- R_{out}



Summary

- Introduction to BSIM3
- A procedure of parameter extraction has been implemented into IC-CAP.
- The extraction of the parameter is done without any optimization and is only based on the model-equations. This allows a good use of the model parameters for further statistical analysis.
- The main equations have been shown with measured and simulated data.
- The simulations of drain current and output resistance show a good agreement between measured and simulated curves in every region.

Literature

- [1] Huang, Liu, Jeng, Hui, Chan, Ko and Hu: 'BSIM3 Manual', Version 2.0, University of Berkeley, California
- [2] Khakzar, Mayer, Oettinger: 'Entwurf und Simulation elektronischer Schaltungen mit SPICE', Expert Verlag, Ehningen

Strukturierter, hierarchischer Entwurf am Beispiel des μ P-Kernels FHOP

O. Feißt, D. Jansen

ASIC - Design - Center, Fachhochschule Offenburg

email: o.feisst@fh-offenburg.de

Juli 95

An der Fachhochschule Offenburg wurde im Sept. 93 das Projekt eines implantierbaren 16 Bit Mikroprozessor - Kernels FHOP ins Leben gerufen. Ausgehend von dem in einem Testchip erfolgreich erprobten unstrukturierten Entwurf wurde durch gezielten Einsatz von strukturiertem Routen unter Nutzung der Fähigkeiten zum hierarchischen Arbeiten in der MENTOR - IC - Station eine erheblich verkleinerte und flächenmäßig optimierte Struktur abgeleitet, die sich mit 4 Quadratmillimetern Fläche durchaus mit kommerziellen Mikroprozessor - Kernen vergleichen läßt.

Das FHOP-Projekt

Mit der Verfügbarkeit von High-Level Synthesetools und der Hardwarebeschreibungssprache VHDL besteht heute die Möglichkeit, auch im Bereich der Lehre und Forschung Projekte anzugehen, die früher große Teams erfordert hätten. So wird an der FH - Offenburg das Projekt FHOP (steht für "First Homemade Operational Processor") ins Leben gerufen mit dem Ziel,

- den Studenten Einblicke in den Aufbau und die Funktion von Mikroprozessoren zugeben,
- die neuen Werkzeuge wie VHDL und Synthese an einem anspruchsvollen Beispiel zu erproben,
- für den Einsatz in komplexen ASICs über einen eigenen lizenzfreien Mikroprozessorkern zu verfügen.

Das Ergebnis dieser Arbeiten ist ein Mikroprozessor - Kernel mit 16 bit Architektur, über dessen Funktion in [9] ausführlich berichtet wird und der in einem Testchip erfolgreich erprobt wird.

Bei diesem ersten Testchip steht nur die Funktion im Vordergrund. Der komplett in VHDL beschriebene FHOP wird deshalb ohne hierarchische

Strukturierung und ohne Rücksicht auf Flächenverbrauch plaziert und geroutet. Der Testchip umfaßt außer dem Kernel noch ein 256-Byte RAM und ein Peripherie-Modul (parallele Ein-/Ausgabe-Einheit). Der Kernel ist als flaches Standardzellen-Design (Microcode-ROM als Block) realisiert (Abb. 1) und weist eine Chipfläche von 13.2 mm² (davon Microcode-ROM 1.4 mm²) auf. Er ist aus 20810 Transistoren aufgebaut. Die verwendete Technologie ist der ES2 - 1.0 μ m - ECPD10 CMOS - Prozeß.

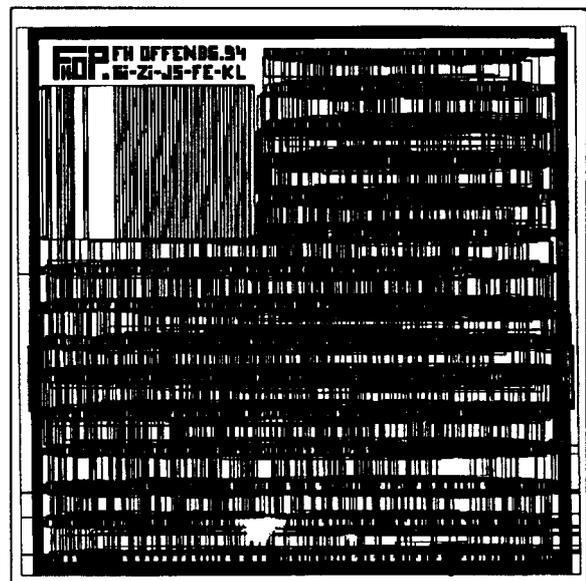


Abb. 1: Der Kernel FHOP V1.0 als Block

Der Chip wurde mit EuroChip Run #114 (Sept. 94) gefertigt und Ende November 94 geliefert. Er ist, abgesehen von einem kleinen Problem beim peripheren PIO-Modul, voll funktionfähig, was sich auch in der ersten Demonstrations-Applikation (Taschenrechner) bestätigt.

Entwurf einer flächenmäßig optimierten FHOP - Struktur

Die Untersuchung des Flächenverbrauchs der ersten FHOP - Version zeigt zwei Bereiche auf, die Ansatzpunkte für eine Optimierung bieten

- die sehr aufwendige ALU
- der Registersatz.

Weiterhin steht mit der neuen 0.7 um Technologie ein flächenmäßig verkleinerter Prozeß zur Verfügung. Die beiden Maßnahmen werden im folgenden ausführlich erläutert.

Optimierung der ALU

Da das VHDL-Modell der Original-ALU eine reine Verhaltensbeschreibung darstellt, kann man nicht mit einem optimalen Syntheseresultat rechnen. Die auf ES2 optimierte ALU umfaßt folgerichtig auch 37 Sheets mit 890 Instances (bzw. 6400 Transistoren).

Um dieses Ergebnis zu verbessern, wird von Hand eine eigene ALU mit den gleichen Eigenschaften auf Schematic Capture Ebene entworfen und simuliert. Das Resultat ist mit 227 Instances (3132 Transistoren) auf 5 Sheets deutlich kleiner als die erste Version. Bei einer durchschnittlichen Flächenbelegung von 1600 Transistoren/mm² bei der ES2 1.0µ-Bibliothek ergibt sich somit ein Flächengewinn von ca. 2 mm².

Migration von 1.0µ auf 0.7µ

Die Auslieferung der EuroChip CDROM Libraries V2.02 im März 95 ermöglicht uns den Zugriff auf die 0.7µ-Bibliothek von ES2.

Die Umstellung erweist sich als problemlos, da die ES2-Bibliotheken fast identisch aufgebaut sind. Auch die Datenorganisation sowie die Systemumgebungs- und Location Map-Variablen (Mentor-spezifisch) sind einheitlich. Es ist somit möglich, die auf ES2 1.0µ gemappten Schematics zugrunde zu legen, d.h. man kann sich einen zweiten kompletten Logiksynthese-Durchlauf sparen. Folgende Schritte sind beim Umstieg durchzuführen:

- Neugeneration aller verwendeten Megazellen
- Update der Schematics;

Der Update ist problemlos bis auf folgende Ausnahmen:

- LIBIECL2 hat keinen Ersatz in 0.7µ
- LIBADD2 wird ersetzt durch LIBFADD2

- geänderter Pfad zum Symbol hzpull_del bei Verwendung der Autologic-Bibliothek (.../hzpull_del > .../LIBHZPULL/hzpull_del)

- Neugeneration aller Viewpoints

Reguläres Routing des Registersatzes

Als weiterer Verbesserungsansatz bietet sich an, die reguläre Struktur des aus sechs identischen 16-Bit-Registern bestehenden Registersatzes zu nutzen. Dazu ist es sinnvoll, zuerst ein 16-Bit-Register zu optimieren und als fertige Hardmakro-Zelle in die ES2-Library aufzunehmen. Anschließend baut man den Registersatz aus 6 einzelnen Registerzellen auf und nimmt schließlich den gesamten Registersatz als Block wiederum in die Library auf.

Um diesen Weg zu gehen, kann man schon auf Schematic Capture Ebene im Design Architect erste Vorbereitungen treffen:

- Vergabe der "phy_comp"-Property am Blocksymbol des 16-Bit-Registers und vorbereitend auch gleich am Blocksymbol des Registersatzes.

Die Symbole werden dadurch in der IC-Station als fertige Blöcke betrachtet und nicht weiter nach unten aufgelöst (Voraussetzung: man stellt beim "logic loading" die Option "user defined" ein).

- Vergabe von Preplace-Properties an den Instances des 16-Bit-Registers

Die relative Lage eines Instance im Floorplan kann schon hier definiert werden.

Es gibt vier Preplace-Properties (seed, group_seed, place, group_place), von denen die seed-Properties durch die automatischen Place&Route-Tools übergangen werden können, während die place-Properties unbedingt bindend sind. Eine seed-Property könnte z.B. den Wert "T1:L3" (entspricht: Top 1, Left 3) haben, was soviel bedeutet, daß dieses Instance in der ersten Floorplan-Reihe von oben an dritter Stelle von links plaziert werden soll. Diese Informationen werden beim Autofloorplan-Aufruf verarbeitet, so daß man ohne manuellen Eingriff sofort den Floorplan samt der plazierten Zellen erhält.

Die nächsten Möglichkeiten für entscheidende Eingriffe in den Design-Ablauf ergeben sich in der IC-Station:

Als wichtiger "Manipulationsknopf" erweist sich hier der *ecpd07-Prozess*- Beschreibungsfile der Firma ES2, der die Technologie, wichtige Design

Rules etc. beschreibt. Durch einige Änderungen kann dieser Prozeß für das Block-Routing optimiert werden und wird im Folgenden als "ecpd07_block" bezeichnet. Die wichtigsten Änderungen betreffen die Einstellungen der "port_styles"- und der "power_styles"-Variablen.

Durch die Definition der Power-Ports auf nur einem statt auf zwei Routing Levels (metal1 und metal2) mithilfe der *port_styles*-Variablen kann erreicht werden, daß evtl. nötige Vias (Durchkontaktierungen) auf der vollen Port-Länge platziert werden. Führt man diese Änderung nicht durch, wird bei Bedarf nur ein einziges Via in der Portmitte gesetzt.

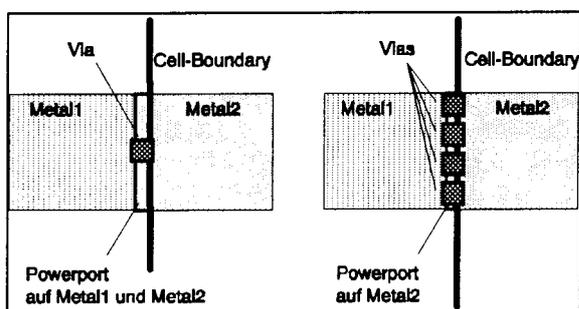


Abb. 2: Die Wirkung der *port_styles*-Variablen

Die zweite wichtige Änderung betrifft die *power_styles*-Variable und damit das Aussehen des Power-Routings. Mit der Zielrichtung minimales Power-Routing kann man hier bis auf die Stile *row_to_bus*, *internal_row* und *tie_down* alle Einstellungen entfernen.

Wenn man nun die neue Zelle, z.B. *16bitreg*, kreiert, sollte man beim Ausfüllen des entsprechenden Formulars auf folgende Einstellungen achten:

- Einstellung von *ecpd07_block* als Zellenprozess,
- Eintrag der verwendeten Preplace-Properties (z.B. *seed*) unter "logic properties to copy",
- Einstellung der "user_defined"-Option bei "logic loading" (s.o.).

Nachdem man den Floorplan mit den vorplatzierten Zellen automatisch erzeugt hat, stößt man beim automatischen Platzieren der Ports auf ein weiteres Problem. Alle Anschlüsse, auch die breiteren Power-Ports, werden durch "autoplace ports" mit der Breite der Signal-Ports platziert. Diese Breite kann aber mithilfe der "*net_comp*"-Property beeinflusst werden. Da es im Schaltplan jedoch kein physikalisches Äquivalent für die Block - Ports in der IC-Station gibt, kann man diese Property nicht schon auf Schaltplan-Ebene

vergeben. Man muß die Ports erst in der IC-Station platzieren, die Power-Ports selektieren, die Property hinzufügen (wichtig: Value "*power_port*", Owner "*port*") und die Ports erneut platzieren lassen.

Durch diese Prozedur erscheinen die Power-Ports beim zweiten Platzierungsdurchlauf mit der korrekten Breite.

Bei der Bearbeitung der Strukturen ist weiter von Vorteil, daß komplette Ports dupliziert werden können, welche dann auch Mitglieder des gleichen Netzes sind.

Hierzu muß zuerst ein Bestandteil des Ports selektiert und anschließend der dazugehörige Port mit *activate port* für Editieroperationen (z.B. copy) vorbereitet werden. Ab diesem Zeitpunkt werden alle hinzugefügten Shapes/Paths automatisch Mitglieder des aktivierten Ports bzw. seines Netzes (normale Editieraktionen sind nur im *deactivate-status* sinnvoll). Zu beachten ist auch, daß bei einem erneuten *autoplace ports/pins* die Duplikationsinformation verloren geht, d.h. es wird wieder nur ein Port platziert.

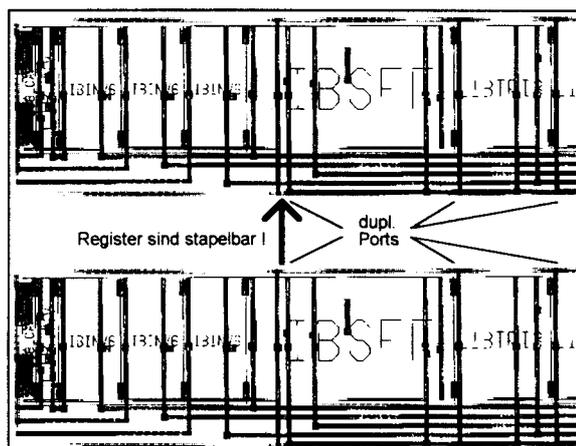


Abb. 3: Port-Duplikation ermöglicht ein Stapeln der 16-Bit-Register (abutting)

Wenn man diese Möglichkeit beim 16-Bit-Register nutzt, so daß in der Top- und der Bottom-Reihe des Blocks die Ports exakt an der gleichen Stelle sitzen, dann lassen sich die einzelnen Registerzellen ohne weiteren Routingaufwand zu einem Registerblock aufeinanderstapeln (s. Abb. 3). Es müssen dann im Registerblock nur noch die Enable-Leitungen an die einzelnen Register geführt und das Power-Routing vervollständigt werden.

Prinzipiell zeigt sich, daß das Arbeiten mit dem *autoroute overflow*-Kommando oft die besten Ergebnisse liefert (kürzeste Verbindungen), v.a. wenn man als Option "*non preferred routing length = 0*" einstellt.

Bei diesem Vorgehen müssen die Cap-Cells (Abschlußzellen einer Floorplan-Reihe) manuell plaziert werden.

In Abb. 4 sind zum Vergleich die verschiedenen Versionen des Registersatzes einander gegenübergestellt. Man erkennt, daß durch den strukturierten Entwurf ein beträchtlicher Flächengewinn erzielt wird, was besonders deutlich beim Vergleich der 1.0 μ -Entwürfe (Version 1 und 2) auffällt. Dies liegt daran, daß der vollautomatische Entwurf (Abb 4 - 1) mit dem Original-Prozeß *ecpd10* erzeugt wurde und das Power-Routing somit in keinsten Weise optimiert ist.

restliche Logik (ALU, Steuerung etc.) in der IC-Station interaktiv zu einem eigenen Block (*alu&rest*) aufbauen, um mehr Einflußmöglichkeiten auf das Aussehen des Gesamtchips zu haben.

Man wechselt dazu kurzzeitig in den *CE-Modus*, bricht die Hierarchie-Strukturen auf und baut sie neu auf. Dies ist mithilfe des *Floorplan Hierarchy Windows* und der *Flatten/Partition*-Kommandos sehr einfach möglich.

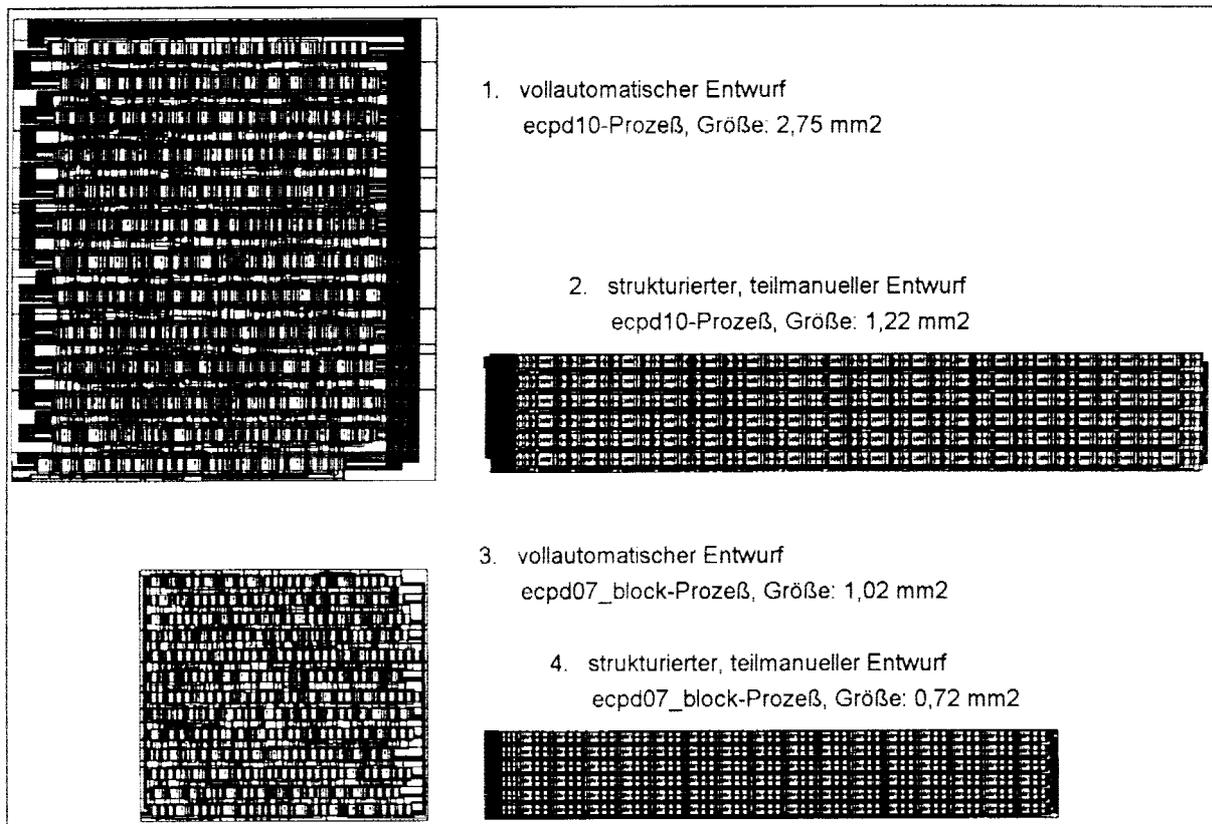


Abb. 4 Vergleich der verschiedenen Registersatz-Entwürfe

Der vollautomatische Entwurf (Abb 4-3) mit dem angepaßten Prozeß *ecpd07_block* wird demgegenüber schon recht klein, so daß der Gewinn durch strukturierten Entwurf (Abb 4-4) geringer ausfällt, aber immer noch ca. 30% beträgt.

Hierarchisches Routing des gesamten Kernels

Abschließend gilt es nun, den kompletten Kernel unter Verwendung der vorhandenen Blöcke hierarchisch neu aufzubauen.

Nachdem man die Zelle kreiert, den Floorplan automatisch erzeugt und die vorhandenen Blöcke (Microcode-Rom, Registersatz) nach den eigenen Wünschen plaziert hat (Abb. 5), kann man die

Nun führt man im Context des neuen Blocks *alu&rest* folgende Aktionen durch:

- *ecpd07_block* als Prozeß einstellen,
- *ES2-Library ecpd07* hinzubinden, (*attach library*)
- *autoplace std cells* mit Option "initial" (Zellen werden "vorläufig" plaziert),
- *autoplace ports*; "*net_comp*"-Property an Power-Ports,
- erneutes *autoplace ports*, (s. "Reguläres Routing des Registersatzes"),
- evtl. Ports duplizieren (z.B. Power, Clock).

Nachdem man so den Block *alu&rest* in seinem Aussehen grob definiert hat, kehrt man wieder in den Context der Topzelle zurück, um dort die Platzierung zu vervollständigen, d.h. die noch fehlenden Ports der Topzelle zu plazieren. Dabei kommt hier wieder das schon beschriebene Vorgehen mit der Vergabe der *"net_comp"*-Property (gewünschte Port-Breite über *"port_styles"*-Variable im Prozeß einstellen) und dem evtl. Duplizieren von Ports zur Anwendung.

Die Auswirkung der bis hierhin durchgeführten Aktionen (ohne Port-Duplikation) ist in Abb. 6 dargestellt. Man erkennt anhand der Overflows deutlich, daß die Port-/Pin-Plazierung bei Nutzung der Default-Einstellung ein ungünstiges Ergebnis liefert, was für den Autorouter einen erhöhten Verdrahtungsaufwand bedeutet.

Man hat nun aber die Möglichkeit, das Port-/Pin-Placement vom Topzellen-Context aus hierarchisch zu ändern, d.h. man kann sowohl die Lage der Topzellen-Ports, als auch die der Block-Pins beeinflussen. Dazu muß man sich in Erinnerung rufen, daß im Block *alu&rest* zum jetzigen Zeitpunkt nur die Standardzellen plaziert sind, aber noch keinerlei Routing existiert und somit auch die Pins uneingeschränkt bewegt werden können.

Der Designer hat also die Möglichkeit, von außen das Aussehen des Gesamtchips festzulegen. Zu beachten ist lediglich, daß Ports sowohl mit *autoplace* als auch mit *select & move_in_row* manipuliert werden können, während bei Pins nur das *autoplace*-Kommando greift. Hier gibt es nun das Problem, daß die Power-Ports der Blöcke normalerweise eine andere Breite haben als die Power-Ports der Topzelle (verschiedene *"power_port"*-Einstellungen in den Prozessen !). Ein *autoplace pins* vom Topzellen-Context aus würde somit die Power-Pins am Block mit der Breite der im Topzellen-Prozeß definierten breiteren Power-Ports erzeugen. Das bedeutet, daß man Power-Ports immer nur im zugehörigen Context automatisch plazieren sollte.

Das Ergebnis des so hierarchisch optimierten Port-/Pin-Placements mitsamt der duplizierten Ports (VCC und VSS 4-fach, Clock 3-fach) ist in Abb. 7 zu sehen.

In Abb. 8 ist zusätzlich auch noch die Verteilung der Overflows des Clock-Netzes hervorgehoben.

Man erkennt, daß die Overflows durch die Duplikation der Clock-Ports sowohl im Registersatz als auch im Block *alu&rest* günstig verteilt sind.

Nachdem so das äußere Aussehen definiert ist, kann man das Innere des Blocks *alu&rest* endgültig fertigstellen und begibt sich dazu wieder in den Block-Context.

Zuerst werden die Standardzellen unter Berücksichtigung des optimierten Port-Placements erneut plaziert (*autoplace std cells* mit Option *"resume"*) und anschließend wird der ganze Block verdrahtet.

Dabei sind folgende Einstellungen von Vorteil:

- **Definition kritischer Netze**, z.B. Clock, mit dem Kommandos *"change net priority"* (diese Netze werden vom Autorouter zuerst verarbeitet).
- **Port-Rows "unslideable" machen**; die relative Lage der Ports bleibt dadurch erhalten, die Rows werden aber als Ganzes verschoben.

Um die Verschiebung der Rows (vergl. Abb. 9) wieder auszugleichen und die äußere Verdrahtung fertigzustellen, kehrt man wieder in den Topzellen-Context zurück. Nachdem man dort das Block-Placement nochmals angepasst hat (Abb. 10), beginnt man mit dem abschließenden Routing.

Der in Abb. 11 dargestellte erste Versuch wird mit den Default-Einstellungen des Autorouters und mit *"internal ring"* ("vertical bus" etc. wurde entfernt) als *power_style* im Prozeß durchgeführt. Man sieht, daß der Autorouter für jeden äußeren Power-Port, bzw. für jedes Overflow, einen eigenen *internal ring* anlegt und außerdem die Blöcke etwas verschiebt.

Für den zweiten Versuch werden deshalb die duplizierten Power-Ports wieder entfernt, so daß nur ein Power-Paar übrig bleibt (Abb. 12). Außerdem werden die äußeren Port-Pows "unslideable" gemacht und beim Autorouter die Option *"fixed channel size"* eingestellt. Dadurch ist der Autorouter gezwungen, alle Plazierungen und Abstände beizubehalten (Abb. 13). Man sieht auch, daß sich die Anzahl der Power-Ringe deutlich verringert hat, nämlich auf ein Ringpaar für die äußeren Ports und ein Ringpaar für die Power an den Blöcken. Die Verbindung der beiden Ringsysteme wird über das am Block *alu&rest* duplizierte Power-Port-Paar hergestellt.

Um dieses Ergebnis noch zu verbessern, muß man manuell eingreifen. So zeigt Abb. 14, wie die Zelle aussieht, nachdem man eines der Ringpaare entfernt und den Rest optimiert hat.

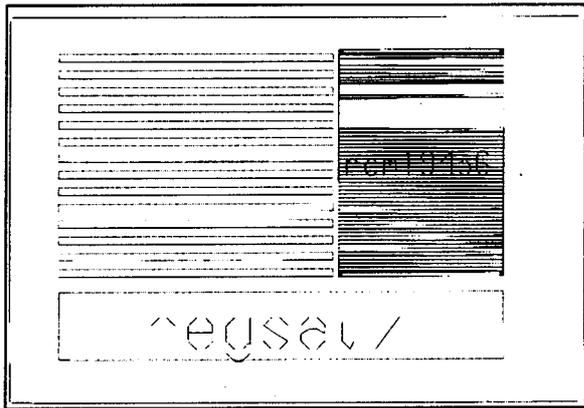


Abb. 5: Nach Platzierung der Blöcke

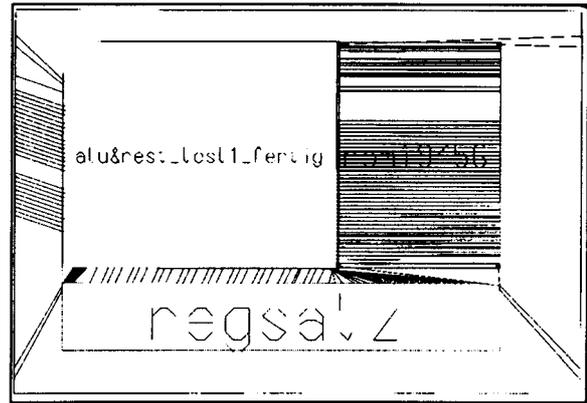


Abb. 9: Nach dem Routen von ALU&REST

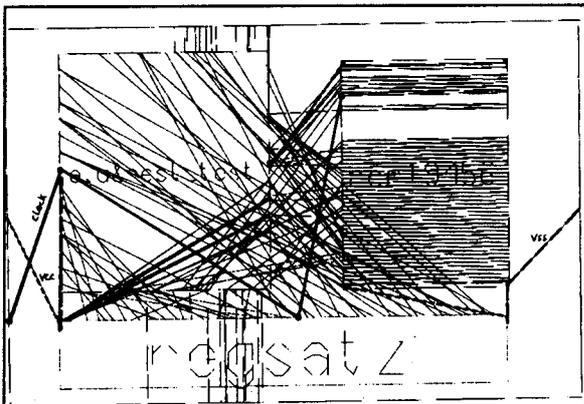


Abb. 6: "autoplace ports/pins" nach "partitioning"

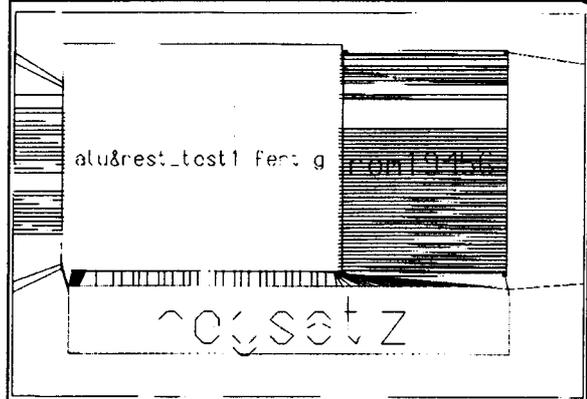


Abb. 10: Mit optimiertem Block-Placement

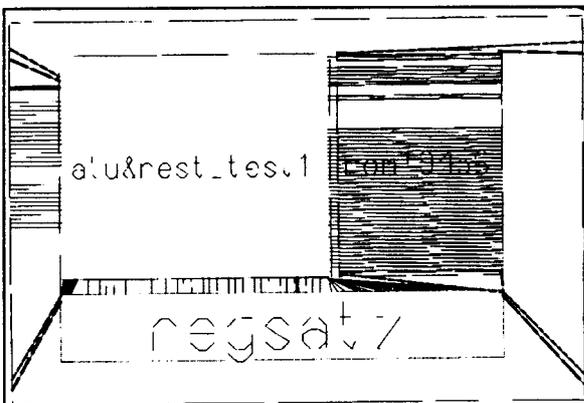


Abb. 7: Hierarch. optimiertes Port-/Pin-Placement

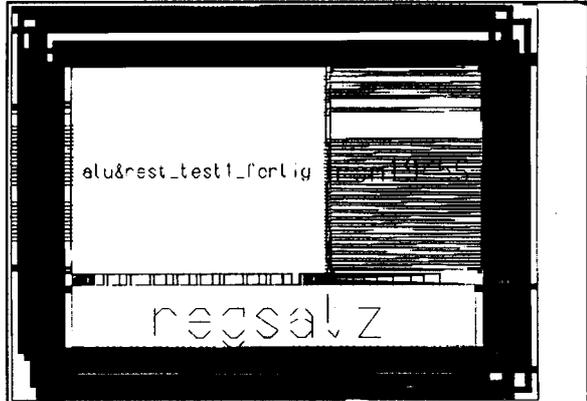


Abb. 11: Mit einem "internal ring" je Power-Port

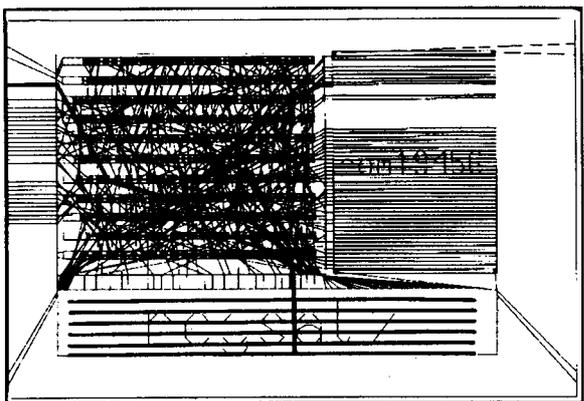


Abb. 8: Die Overflows des Clock-Netzes

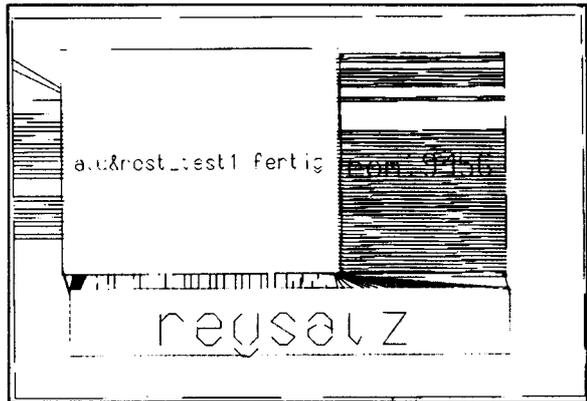


Abb. 12: Nach der Reduktion auf ein Power-Paar

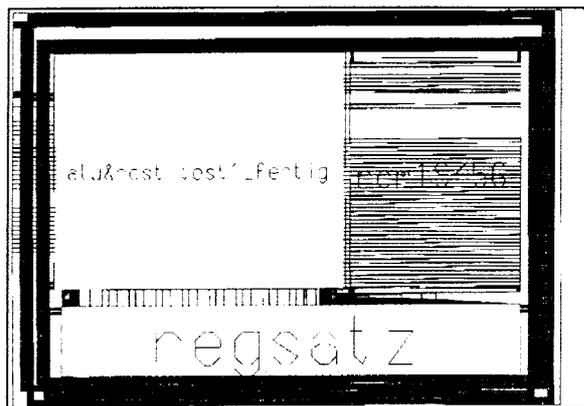


Abb. 13: Mit reduzierten Power-Ringen

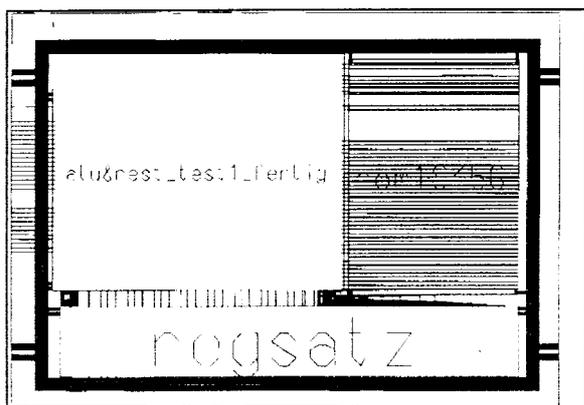


Abb. 14: Power-Routing manuell optimiert

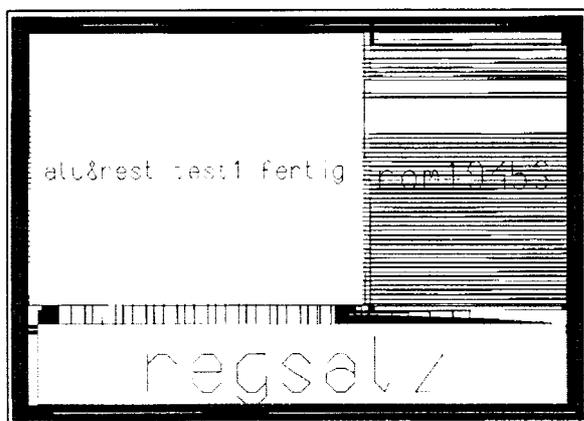


Abb. 15: Weiter optimiert und kompaktiert

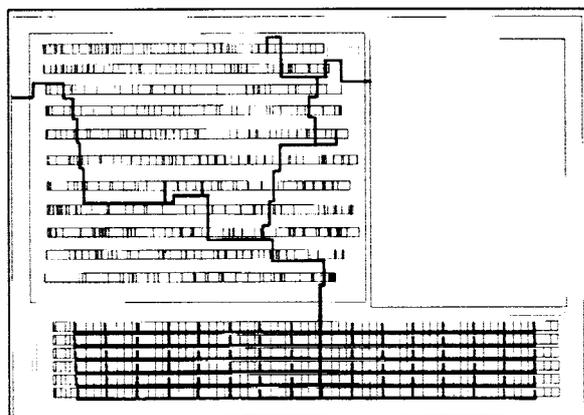


Abb. 16: Das geroutete Clock-Netz

Die endgültige Version ist in Abb. 15 zu sehen. Hier sind die äußeren Ports so nahe wie möglich mithilfe des "compact"-Kommandos an die Power-Ringe herangeschoben worden.

Als Letztes ist es noch interessant, das Routing der als kritisches Netz deklarierten Clock-Leitung zu betrachten. In Abb. 16 läßt sich erkennen, wie sehr sich der Autorouter an den Overflows des Netzes orientiert hat und dadurch ein recht zufriedenstellendes Ergebnis liefert.

Zusammenfassung

Der FHOP-Kernel wurde in mehreren Schritten soweit verbessert, daß die neue Version 1.1 nur noch 30 % des Flächenbedarfs der Version 1.0 aufweist. Dabei wurde der Flächengewinn in etwa gleich stark durch den Umstieg auf die kleinere 0.7µ-Technologie und den strukturierten Entwurf des Registersatzes erzielt.

Die Ersparnis durch Verwendung des 0.7µ-Prozesses beträgt nach Tabelle 1 etwa 40 %. Die etwas größere Zahl beim FHOP-Kernel ergibt sich durch die größere, synthetisierte Original-ALU und durch das relativ ineffektive Routing der 1.0µ-Version.

Das strukturierte Routing liefert eine Ersparnis von ca. 30 % (Tabelle 2). Auch hier liegt die Erklärung für den größeren Wert beim 1.0µ-Registersatz im ineffektiven Routing der unstrukturierten Version.

Einen Überblick über die Größenverhältnisse gibt Abb. 17. Bei einer Fläche unter 4 Quadratmillimetern ist der FHOP-Mikroprozessor-Kernel V1.1 damit durchaus mit kommerziellen Prozessorkernen vergleichbar und kann in Zukunft effektiv in komplexen anwendungsspezifischen integrierten Schaltungen eingesetzt werden.

| | 1.0 µ | 0.7 µ | Gewinn |
|---------------|-------|-------|--------|
| Register satz | 1,22 | 0,72 | 41 % |
| 19-kBit ROM | 1,40 | 0,9 | 36 % |
| FHOP-Kernel | 13,21 | 5,34 | 60 % |

Tabelle 1: Flächengewinn (in mm²) durch Umstieg von ES2 1.0µ auf ES2 0.7µ

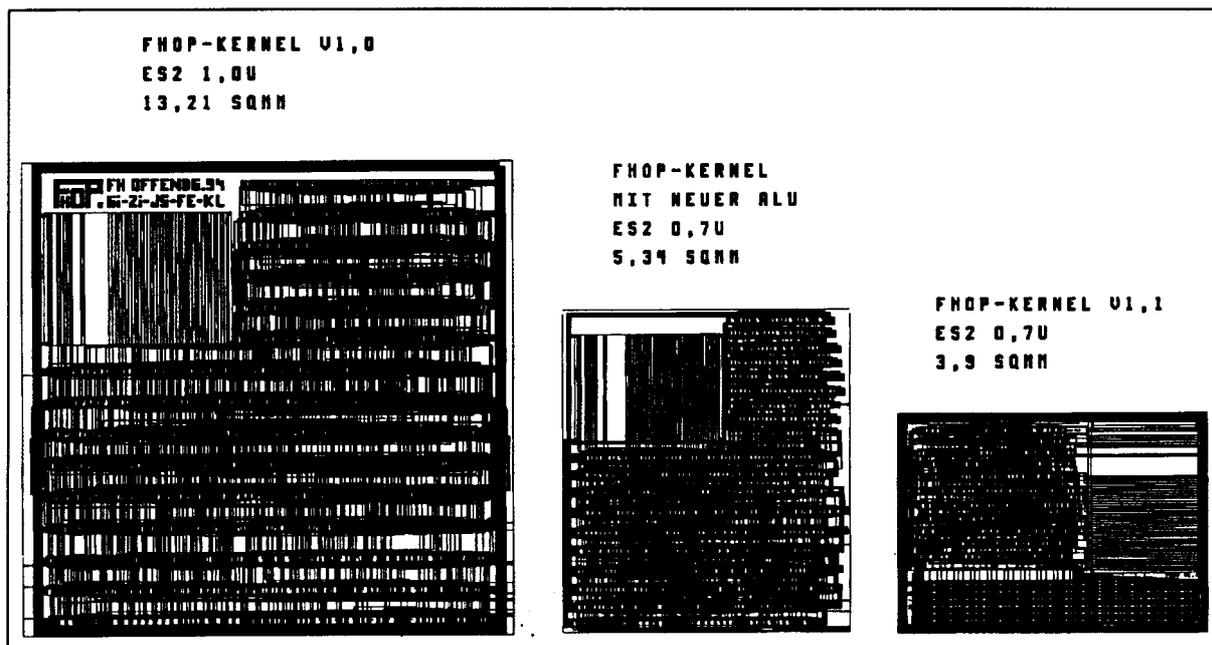


Abb. 17: Vergleich der verschiedenen FHOP-Entwürfe

| | autoplace & -route | strukturiert | Gewinn |
|-------------------|--------------------|--------------|--------|
| Registersatz 1.0µ | 2,75 | 1,22 | 56 % |
| Registersatz 0.7µ | 1,02 | 0,72 | 29 % |
| FHOP-Kernel 0.7µ | 5,34 | 3,9 | 27 % |

Tabelle 2: Flächengewinn (in mm²) durch strukturierten Entwurf

Literaturverzeichnis:

[1] Mentor Graphics, Software Version 8.4_1: "IC Station Automated Layout Procedures Manual" Juli 94

[2] Mentor Graphics, Software Version 8.4_1: "IC Station Technologie Definition Manual" Juli 94

[3] Mentor Graphics, Software Version 8.4_1: "IC Station Design Flow Manual" Juli 94

[4] Mentor Graphics, Software Version 8.4_1: "Properties Reference Manual" Juli 94

[5] Dantzer-Sorensen, M. , Andersen, A.C.: "Mentor Graphics V8.2 ES2 ecpd10 and ecpd0, IC Station User's Guide V1.1" EUROCHIP DTH, 14. Sept. 94

[6] European Silicon Structures: "ES2 Mentor V8 Design Kit Release Notes V3.0.0" 28. Okt. 94

[7] European Silicon Structures: "ES2 Mentor V8 Design Kit User Notes V3.0.0" 28. Okt. 94

[8] Klöser, F.: " Entwurf eines Ein-/Ausgabemoduls für den µP-Kernel FHOP mit VHDL und Integration aller FHOP-Komponenten auf einem Test-ASIC" Diplomarbeit an der FH Offenburg, Aug. 94

[9] Zimpfer, F. , Gieringer, Th. , Jansen, D.: "Entwicklung eines 16-Bit Mikroprozessor-Kernels mit Hilfe von VHDL" Bericht vom MPC-Workshop in Ulm, Jan. 94

Diamond-Transistoren für Transimpedanzverstärker

Inhalt:

1. Problemstellung
2. Weiterentwicklung vom Einzeltransistorverstärker zum Diamond-Transistor
3. Die Entstehung des Diamond-Buffers
4. Verwendungsbeispiele für Diamond-Transistor und Diamond-Buffer
 - 4.1 spannungsgesteuerte Stromquelle
 - 4.2 Transimpedanzverstärker
5. Probleme bei der Realisierung in integrierten Schaltungen
6. Stand der Technik heute

1. Problemstellung

Das Ziel dieser Arbeit war zu Beginn der Entwurf von Verstärkern mit möglichst hoher Bandbreite, d.h. vom DC-Bereich bis zu mehreren hundert Megahertz. Dazu sollten die neuaufgekommenen sogenannten "Diamondschaltungen" verwendet werden, die in der Zwischenzeit von mehreren Herstellern vermehrt auf den Markt geworfen werden. Dabei ging es nicht unbedingt darum, die kommerziellen Bausteine zu übertreffen, sondern das "Innenleben" zu verstehen und zu versuchen, derartige Schaltungen unter Berücksichtigung der Möglichkeiten der FH zu integrieren. Im folgenden Kapitel wird nun die Entwicklung einer einfachen Verstärkerschaltung bis zum kompletten Diamondtransistor dargestellt. Genauere Simulationsergebnisse können der dazugehörigen Diplomarbeit entnommen werden.

2. Weiterentwicklung vom Einzeltransistorverstärker zum Diamond-Transistor

Die Abbildung 1 zeigt einen Einzeltransistor in Emitterschaltung, wobei willkürlich zum besseren Vergleich mit den folgenden Schaltungsvarianten eine Verstärkung von $V_U = 2$ und ein Ruhestrom von 2mA eingestellt worden ist.



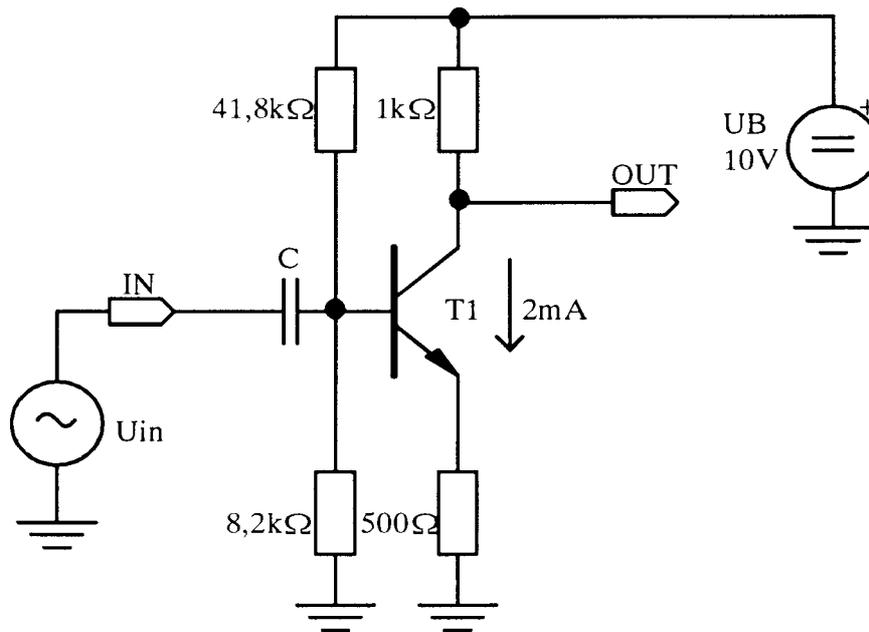


Abbildung 1: einfacher AC-gekoppelter Verstärker $V_U \approx \frac{1k\Omega}{500\Omega} = 2$

Dadurch, daß die Schaltung zum Einstellen des Arbeitspunktes (AP) einen Basisspannungsteiler benötigt, wird zum Ankoppeln der DC-Offset-freien Wechselspannungsquelle ein Kondensator benötigt. Dadurch ist die Anordnung natürlich nicht in der Lage Gleichspannungen zu verstärken. Weitere Nachteile sind der Ausgangsspannungsoffset und die schlechte Integrierbarkeit der Schaltung (platzintensive Widerstände, Kondensator).

Die Abbildung 2 zeigt die erste Verbesserung der Schaltung: Durch Einführen einer symmetrischen Stromversorgung und Aufteilen des Emittewiderstandes des Transistors wird erreicht, daß die Wechselspannungsquelle direkt an die Basis angekoppelt werden kann. Leider funktioniert diese einfache Kompensation der Basis-Emitterspannung des Transistors natürlich nur bei einer bestimmten Temperatur, weil die Steilheit eines Transistors temperaturabhängig ist. Außerdem besteht weiterhin der Ausgangsspannungsoffset von etwa 3V, wodurch, wie auch schon bei der vorangegangenen Schaltung, nur ein recht kleiner Ausgangsspannungshub erreicht werden kann, da die Kollektor-Emitterspannung stets größer als ca. 0,5V sein muß, damit der Transistor im aktiven Bereich arbeitet.

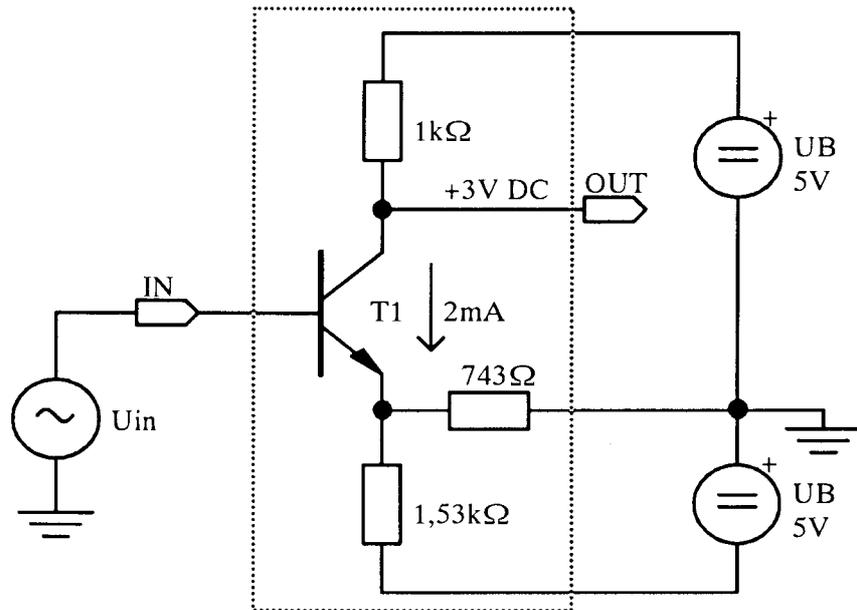


Abbildung 2: Kompensation von U_{BE} $V_U \approx \frac{1k\Omega}{(1,53k\Omega \parallel 743\Omega)} \approx 2$

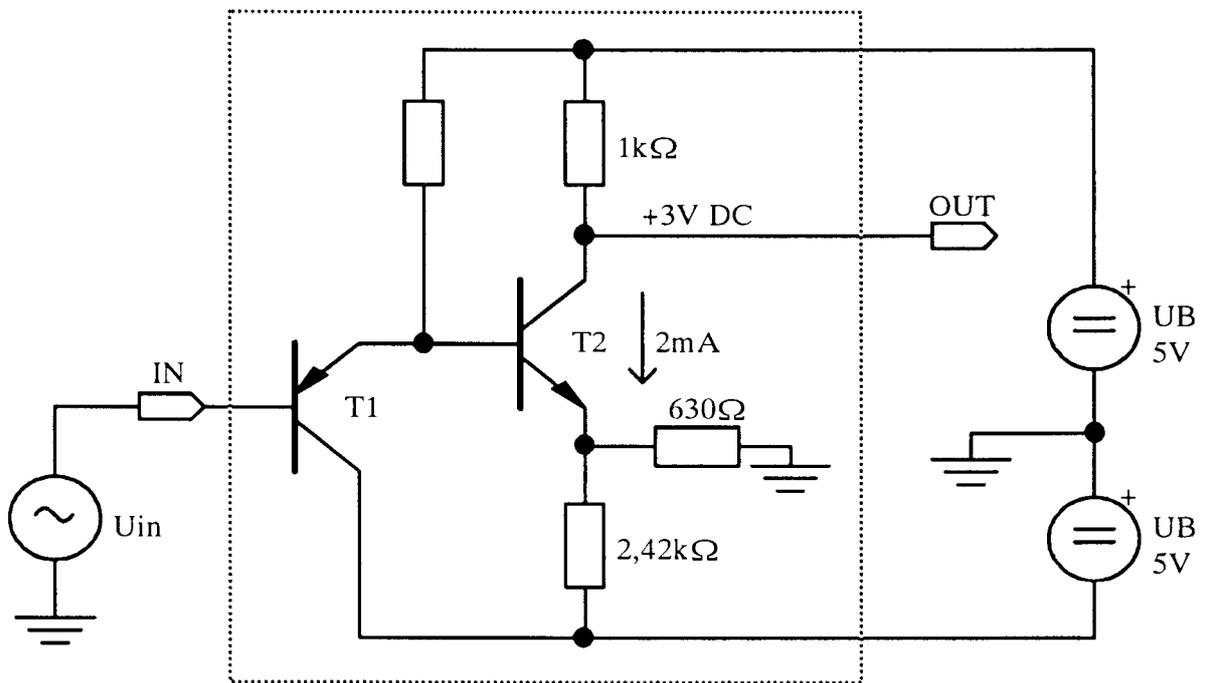


Abbildung 3: verbesserte Kompensation von U_{BE} $V_U \approx \frac{1k\Omega}{(2,42k\Omega \parallel 630\Omega)} \approx 2$

In der Abbildung 3 wurde nun ein zusätzlicher PNP-Transistor eingefügt, um den Eingangsspannungsoffset zu kompensieren, was auf Grund ähnlicher Temperaturkoeffizienten besser funktioniert, als bei der vorangegangenen Schaltung. Die Aufteilung des Emittorwiderstandes ist jedoch trotzdem notwendig, um den Ruhestrom der Schaltung einstellen zu können. Weiterhin sind die verstärkungsbestimmenden Widerstände noch immer nicht ruhestromfrei, wodurch bei der Dimensionierung der Einfluß auf den AP des Verstärkers beachtet werden muß, d.h. die Berechnung der Widerstandswerte ist relativ umständlich. Außerdem entsteht auf diese Weise ein unnötiger Stromverbrauch, der wie folgt vermieden werden kann:

Das Bild 4 zeigt die Weiterentwicklung der Schaltung. Hier wurden, um die Schaltung besser integrieren zu können, sämtliche Widerstände durch platzsparende Stromquellen ersetzt, wodurch zusätzlich die Aufteilung des Emittorwiderstandes von T_2 entfallen kann und die verstärkungsbestimmenden Widerstände ruhestromfrei werden. Um die Ausgangsoffsetspannung zu kompensieren, wurde eine Z-Diode eingefügt, was zwar keine ideale Lösung darstellt, aber vom Prinzip her durchaus, zumindest in einem gewissen Temperturbereich, funktionieren würde.

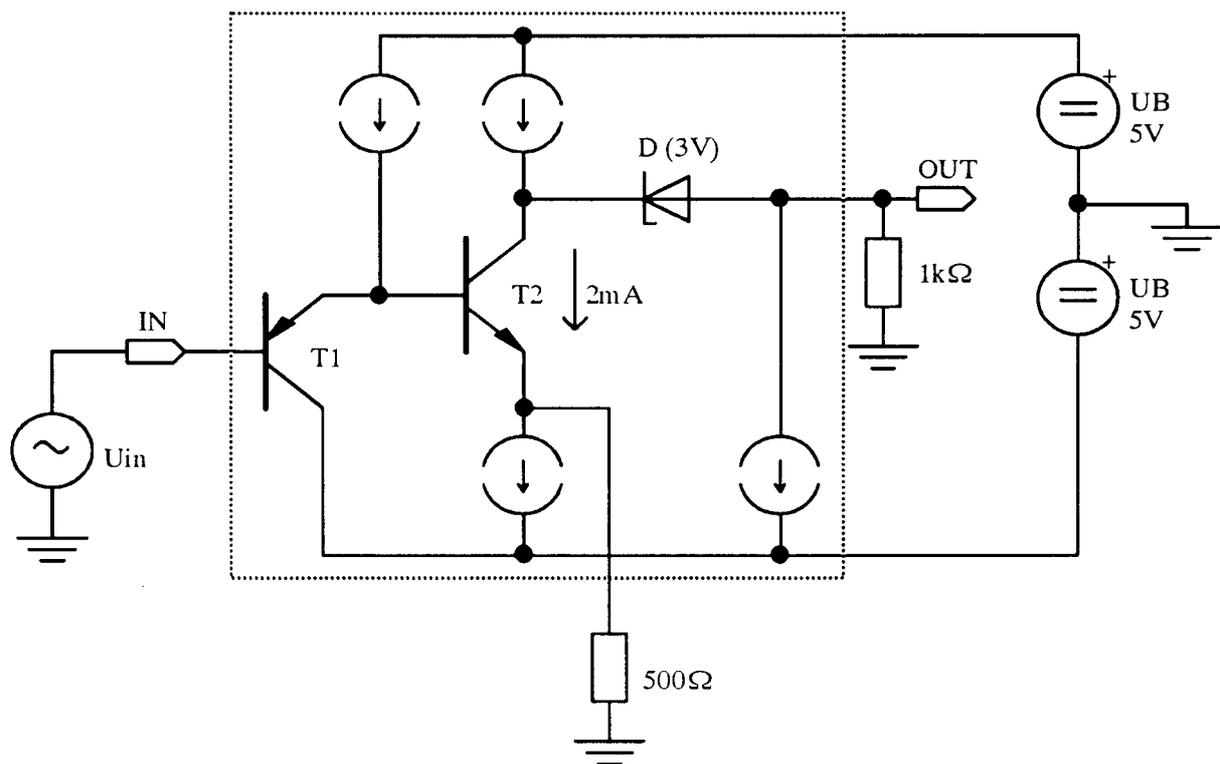


Abbildung 4: Verbesserung der Integrierbarkeit

Besser ist jedoch, die störende Z-Diode durch einen PNP-Stromspiegel zu ersetzen, wie in Abbildung 5 gezeigt wird. Da die Dimensionierung der Stromquellen jedoch relativ heikel wäre und die Schaltung durch ihren unsymmetrischen Aufbau unterschiedliche Slew-Rates für positive bzw. negative Flanken besitzt, ist sie immer noch nicht optimal, daher wird die Schaltung komplementär-symmetrisch ergänzt, wodurch man zum eigentlichen Diamondtransistor gemäß Abbildung 6 kommt.

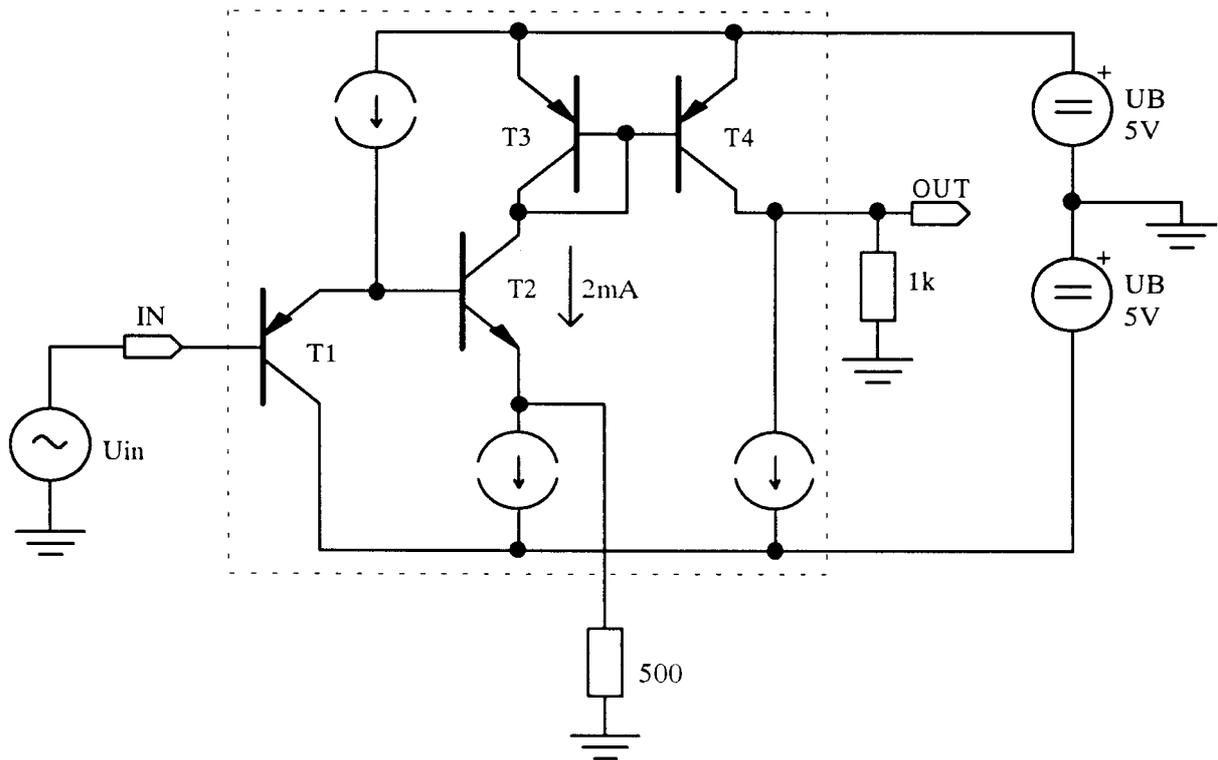


Abbildung 5: Ersatz der Widerstände durch Stromquellen

$$V_U \approx \frac{1k\Omega}{500\Omega} = 2$$

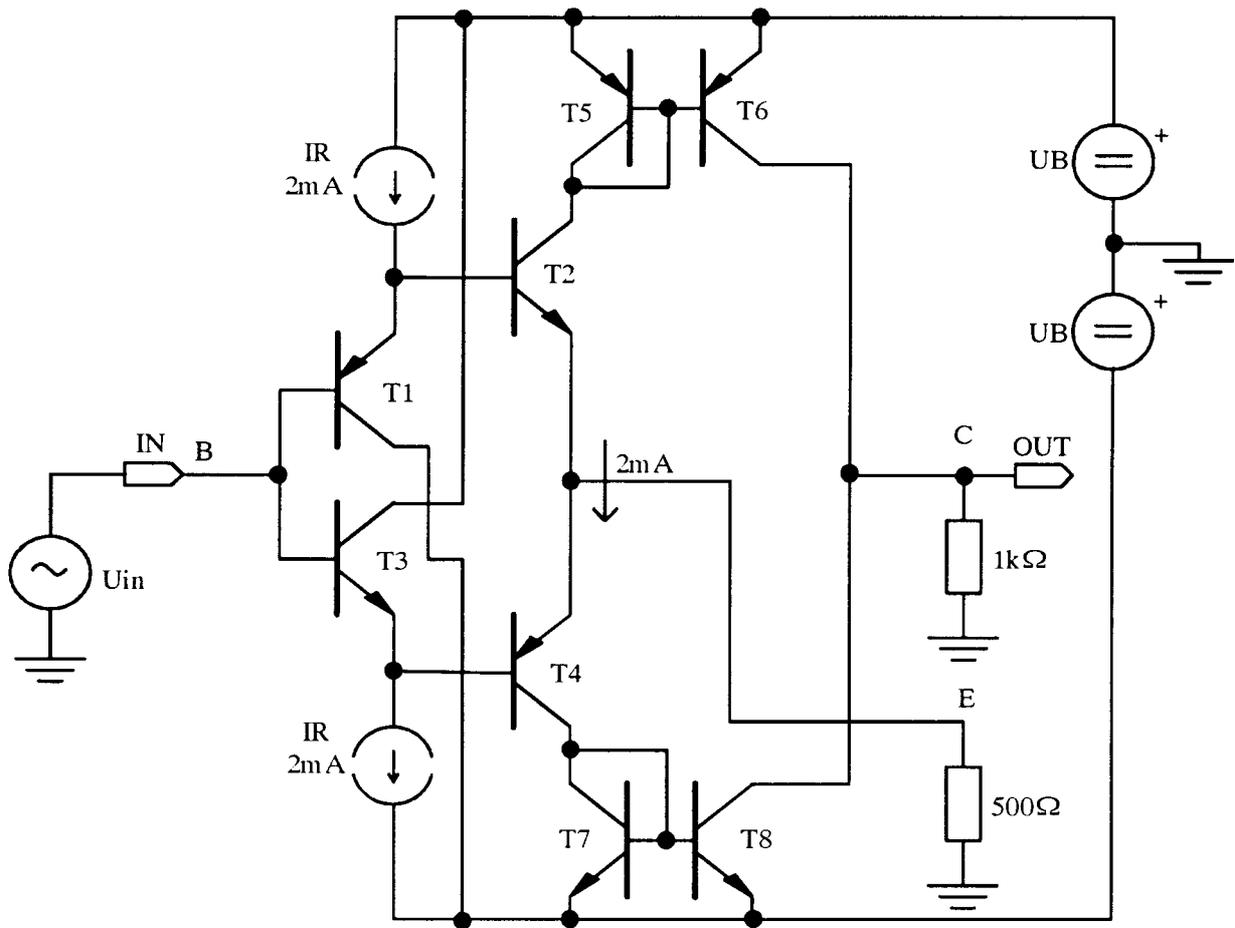
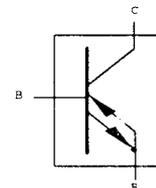


Abbildung 6: kompletter Diamondtransistor (DT) $V_U \approx \frac{1k\Omega}{500\Omega} = 2$

Die Abbildung 6 zeigt die eleganteste Lösung des Problems, den kompletten Diamondtransistor, der alle Vorteile der vorangegangenen Schaltungen in sich vereint: die verstärkungsbestimmenden Widerstände sind ruhestromfrei - die Aussteuerbarkeit ist unabhängig von der Signalrichtung (uneingeschränkt vom Ruhestrom) - hohe symmetrische Slew-Rate-Werte verglichen mit herkömmlichen Strukturen - einfache Einstellung des Ruhestroms usw...



vorgeschlagenes Schaltzeichen (nach Burr-Brown):

Die Schaltung stellt nach außen hin einen idealen Transistor dar, der folgende Eigenschaften hat:

- "idealer" Transistor:
- hochohmiger Eingang B (kein Basisstrom)
 - niederohmiger Ein- bzw. Ausgang E
 - hochohmiger Ausgang C
 - kein Basis-Emitter-Spannungsoffset (idealer Spannungsfolger)
 - Offset-Spannung und Ruhestrom an Eingang und Ausgang verschwinden
 - positive und negative Eingangsspannungen bzw. Ausgangsströme möglich
 - programmierbare Steilheit:

$$S \approx 1,6 \cdot \frac{I_R}{U_T}; U_T = 26mV$$
 - relativ unabhängig vom äußeren AP
 - keine minimale Kollektor-Emitterspannung

Die Abbildung 7 zeigt zum Vergleich den Verlauf der Steilheit beim normalen bipolaren Transistor und beim Diamondtransistor. Man erkennt, daß beim DT der Betrag der Steilheit im Gegensatz zu der des normalen Transistors in einem bestimmten Bereich nahezu konstant ist.

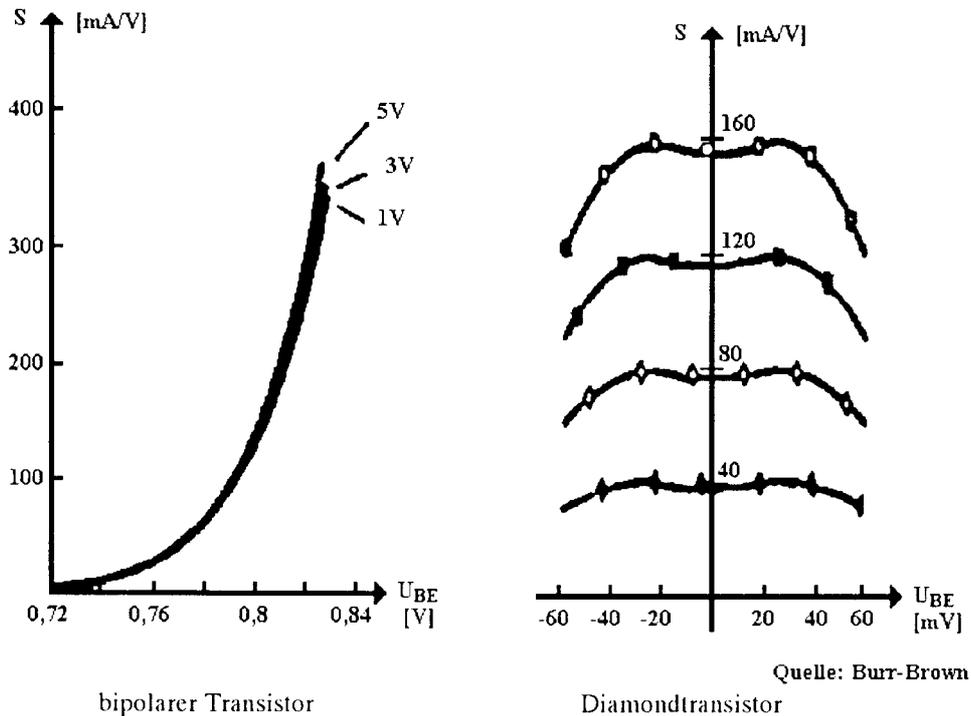


Abbildung 7: Vergleich bipolarer Transistor - Diamondtransistor

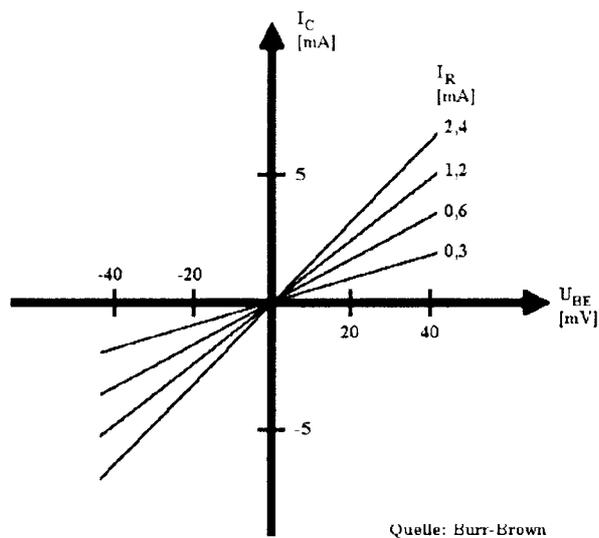


Abbildung 8:
 I_C als Funktion der Eingangsspannung U_{BE} bei unterschiedlichen Ruhestromen I_R

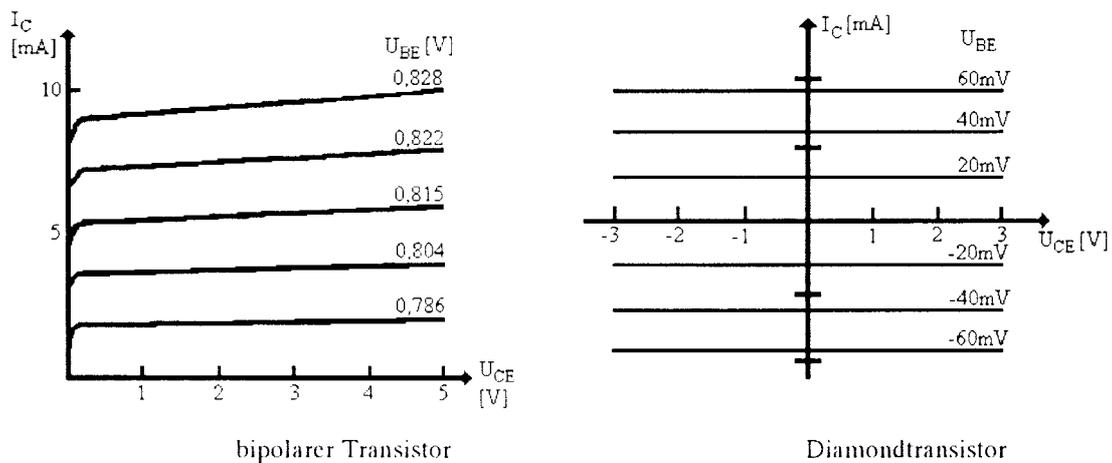


Abbildung 9: I_C als Funktion der Kollektor-Emitter-Spannung

Die Darstellungen 8 und 9 dienen der weiteren Verdeutlichung des Verhaltens des Diamondtransistors.

3. Die Entstehung des Diamondbuffers

Zur Realisierung der verschiedenen Operationsverstärkertypen sind zusätzlich Spannungsfolger (Buffer) mit ähnlich guten Eigenschaften notwendig. Diese lassen sich jedoch leicht dadurch erreichen, daß man die beiden Stromspiegel, die den Kollektoranschluß des Diamondtransistors darstellen, wegläßt und damit eine Schaltung gemäß Abbildung 10 erhält.

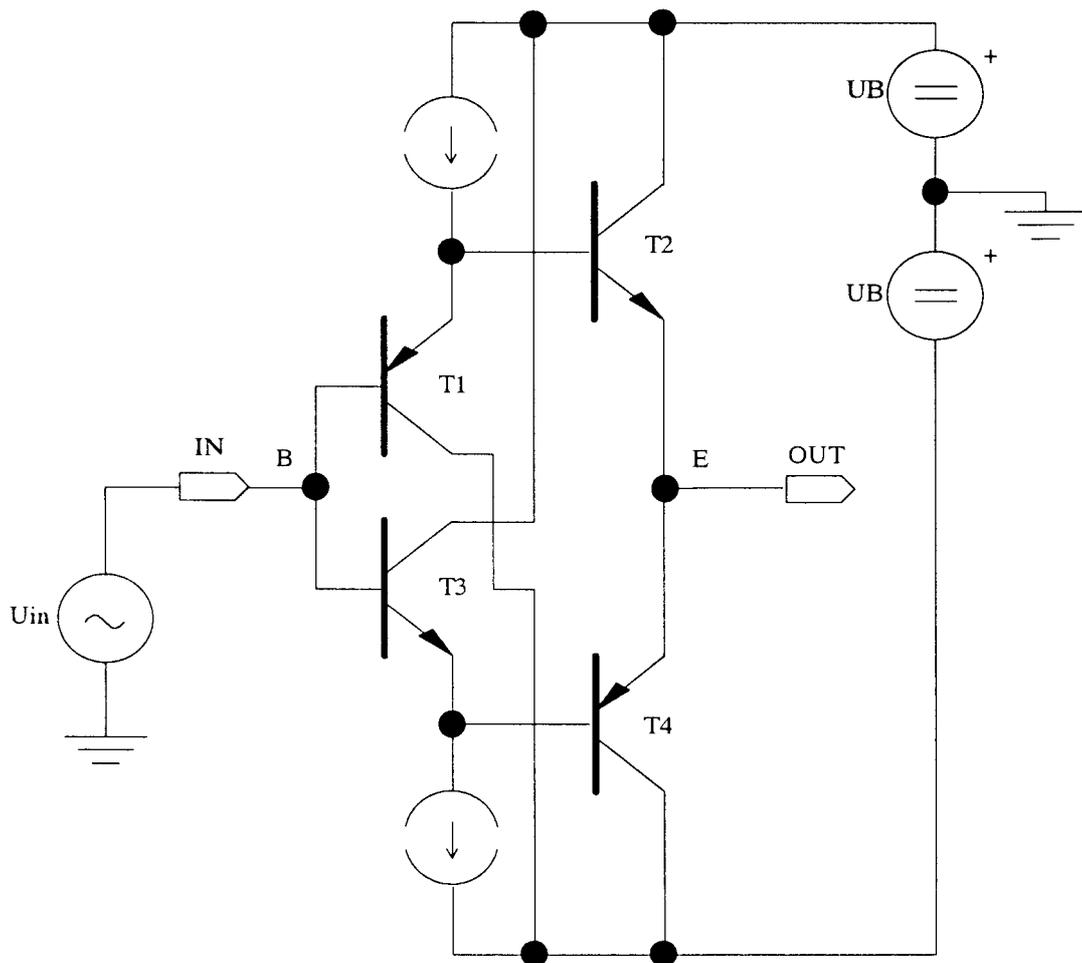


Abbildung 10: Diamondbuffer (DB)

4. Verwendungsbeispiele für Diamond-Transistor und Diamond-Buffer

4.1 spannungsgesteuerte Stromquelle

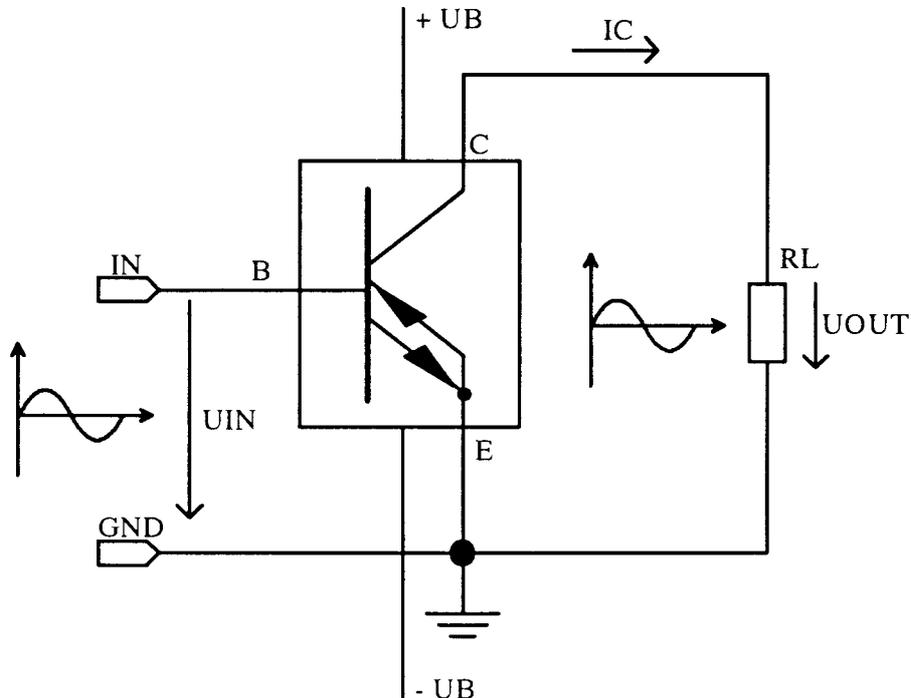


Abbildung 11: Der Diamondtransistor als spannungsgesteuerte Stromquelle

Es gilt: $I_C = S \cdot U_{IN}$ wobei S in einem weiten Bereich eingestellt werden kann

Die Schaltung gemäß Abb. 11 stellt die Grundschaltung des Diamondtransistors dar. Bemerkenswert ist an dieser Stelle, daß die Schaltung bedingt durch die beiden Stromspiegel im Diamondtransistor eine nichtinvertierende Spannungsverstärkung besitzt, es ist daher nicht ohne weiteres möglich, einen herkömmlichen Transistor durch einen DT zu ersetzen, ohne die Gesamtschaltung entsprechend anzupassen. Die Aussteuerbarkeit ist in positive und negative Richtung möglich und idealerweise treten keine Störgrößen wie Offsetspannungen bzw. -ströme auf. Kollektor- und Basisschaltung des Diamondtransistors sind natürlich ebenfalls anwendbar, jedoch nicht so sehr verbreitet.

4.2 Verwendung als Transimpedanzverstärker

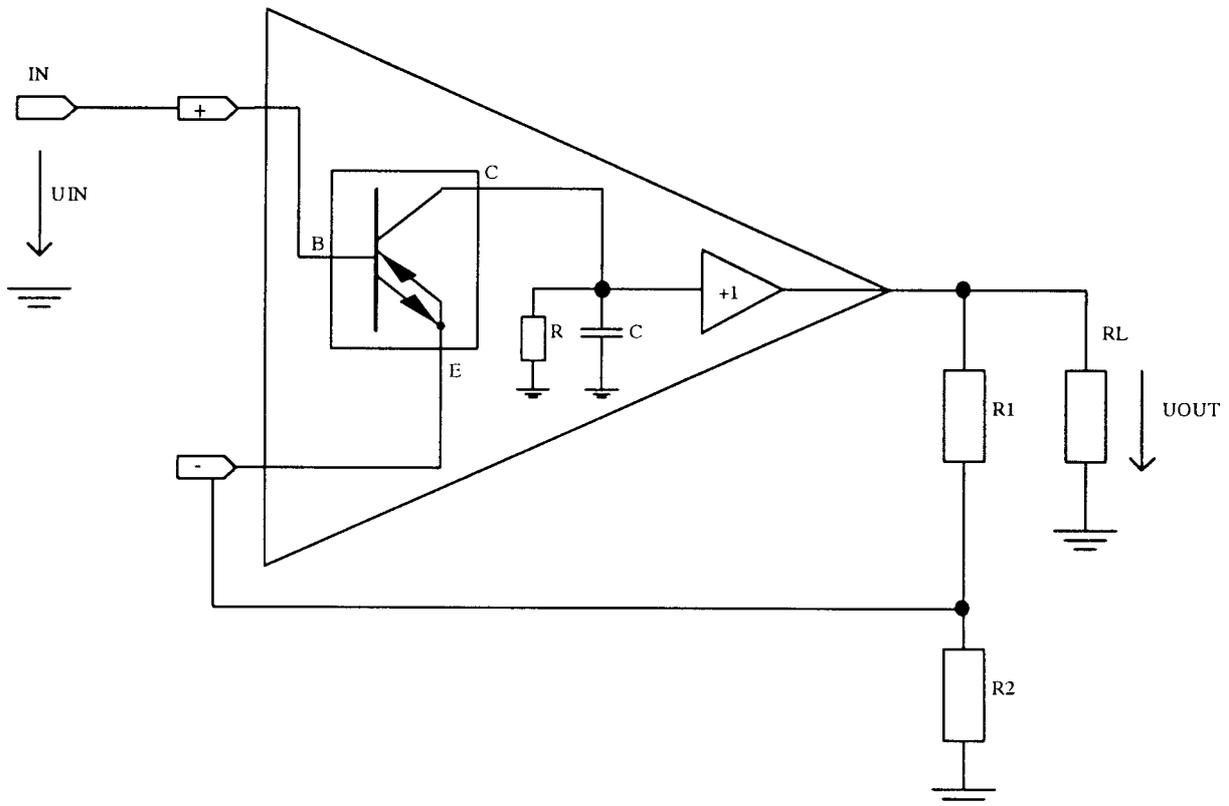


Abbildung 12: Transimpedanzverstärker

Ein Transimpedanzverstärker besitzt bekanntermaßen einen hochohmigen nichtinvertierenden und einen niederohmigen invertierenden Eingang, daher ist der DT eingangsseitig für diese Anwendung geradezu prädestiniert. Um jedoch auch den gewünschten niederohmigen Ausgang zu erhalten, ist ein zusätzlicher Spannungsfolger, d.h. Diamondbuffer notwendig.

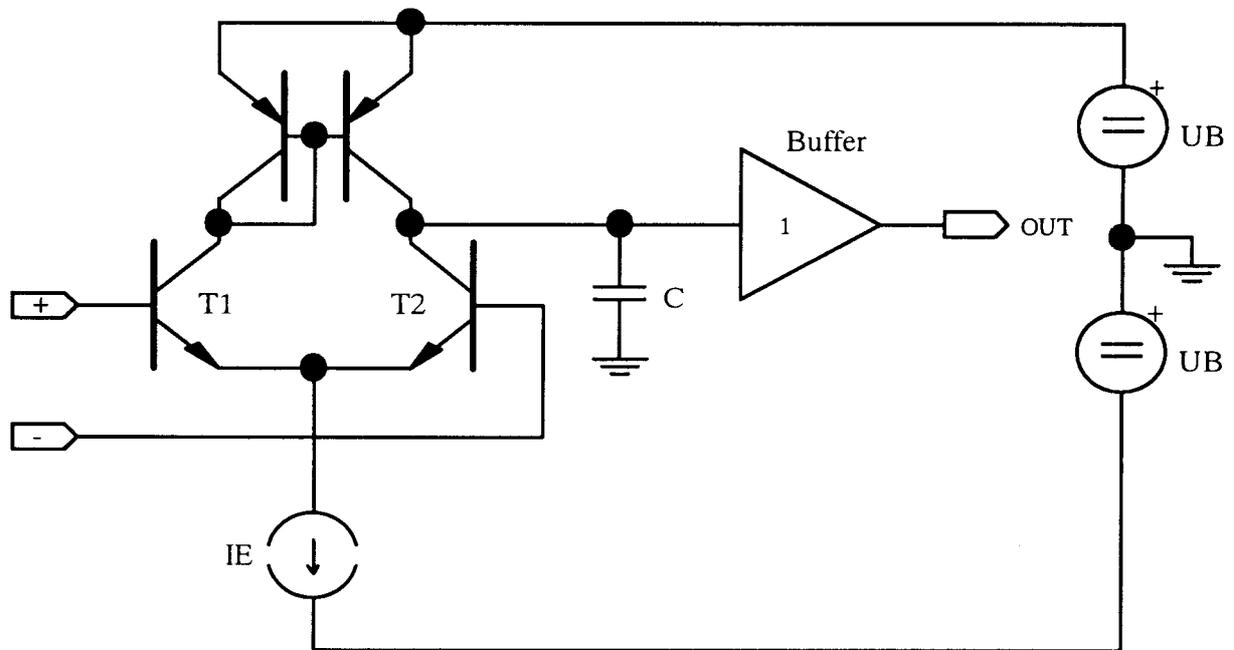
Es gilt:
$$U_{OUT} = V_U \cdot U_{IN} \approx \left(\frac{R_1}{R_2} + 1 \right) \cdot U_{IN}$$
 wie bei einem herkömmlichen OP,

jedoch ist die Bandbreite fast unabhängig von eingestellter Verstärkung, da die Leerlaufverstärkung V_0 abhängig von R_2 ist:

Es gilt nämlich:
$$V_0(j\omega) = \frac{R}{(1 + j\omega RC) \cdot R_2}$$

Dadurch sind bei geeigneter Dimensionierung von R_1 und R_2 zusätzliche Kompensationsverfahren überflüssig, da die Leerlaufverstärkung jeweils angepaßt werden kann, um eine ausreichende Amplituden- bzw. Phasenreserve zu erreichen. Ein weiterer Vorzug des Transimpedanzverstärkers besteht darin, daß die Slew-Rate durch die Stromgegenkopplung besonders hoch ist, da der Umladestrom für C

(theoretisch) nicht begrenzt ist. Zum Vergleich soll die Abbildung 13 dienen, die die Prinzipschaltung eines "normalen" OP's zeigt. Bei diesem ist der maximale Umladestrom für den (parasitären) Kondensator C festgelegt durch den festeingestellten Strom I_E .



Daher gilt: $S_{L,max} = \frac{dU_{OUT}}{dt} \approx \frac{I_E}{C}$ wogegen der Umladestrom beim Transimpedanzverstärker nicht durch den Ruhestrom festgelegt, sondern im Bedarfsfall über den invertierenden Eingang (= Emitter des DT) geliefert werden kann.

5. Probleme bei der Realisierung in integrierten Schaltungen

- a) vergleichbare, komplementäre NPN- und PNP-Transistoren sind notwendig, aber integriert aufwendig herzustellen
- b) hohe Transitfrequenz der Transistoren wünschenswert (auch PNP), um z.B. Vorteile des Transimpedanzverstärkers ausnutzen zu können
- c) Transistor-Array's wie z.B. B500D (Temic) daher schlecht geeignet, da Punkte a) und b) nicht erfüllt

Bei Verwendung von ungleichen Transistoren:

- Folge:
- große Offsetspannungen bzw. -ströme
 - Geschwindigkeitsvorteile der Diamondschaltungen können nicht ausgenutzt werden (langsamer PNP)

6. Stand der Technik heute

- sehr ähnliche NPN- und PNP-Transistoren durch neuartige Prozesse heute integrierbar
- bisherige Hybridbausteine werden durch monolithische IC's ersetzt
- typische Transitfrequenzen der verwendeten Transistoren:

| | NPN | PNP |
|------------|--------------------------|-------------------------------|
| B500D | $f_T = 0,35 \text{ GHz}$ | $f_T = 0,005 \text{ GHz (l)}$ |
| Burr-Brown | $f_T = 3,5 \text{ GHz}$ | $f_T = 2,7 \text{ GHz (v)}$ |
| Maxim | $f_T = 9,3 \text{ GHz}$ | $f_T = 5,5 \text{ GHz (v)}$ |

(l = lateral, v = vertikal)

realisierbare Bandbreiten der Transimpedanzverstärker:

| | |
|----------------------|-------------------------------|
| B500D | $f_G = 1 \dots 3 \text{ MHz}$ |
| OPA 660 (Burr-Brown) | $f_G = 450 \text{ MHz}$ |

Rechnergesteuertes Testen eines Analog-ICs mit „VEE-Test“

A. Diemer, M. Basler, G. Forster
Labor Schaltungsintegration der FH Ulm

1 Einleitung

Angesichts der Komplexität heutiger ICs kommt dem Testproblem weiterhin wachsende Bedeutung zu. Insbesondere bei Analog-ICs oder Mixed Signal-ICs machen die Testkosten einen beträchtlichen Anteil an den Gesamtkosten aus, in Einzelfällen können sie sogar die eigentlichen Chipkosten übersteigen. Aus diesem Grund ist es erforderlich, bereits bei der IC-Entwicklung an den IC-Test zu denken. Optimale Ergebnisse sind zu erwarten, wenn das Prüfprogramm selbst zum Bestandteil der Entwicklungsleistung gemacht wird.

Das Prüfprogramm sollte so gestaltet sein, daß es sich sowohl für die Prototypen- als auch für die Produktionsphase eignet. Während in der Prototypenphase umfangreiche Tests mit ausführlichen Protokollen zum Zwecke der Fehleranalyse bzw. des sog. „Design Centering“ erforderlich sind, geht es in der Produktion um die möglichst schnelle Gut/Schlecht-Selektion. Sie läßt sich als Untermenge des Gesamtprogramms realisieren, so daß bei Bedarf jederzeit der Testumfang erhöht werden kann.

Während Digital-Tester als komplette Einheiten auf dem Markt verfügbar sind, lassen sich Analog-Tester nicht so einfach standardisieren, da sie wesentlich universeller sein müssen. Folgende Anforderungen sind zu erfüllen:

1. Große Vielfalt an Meßparametern
 - Messung von Spannung und Strom in hoher Auflösung
 - Messungen im DC-, Zeit- und Frequenzbereich
2. Ausbaufähigkeit bezüglich der Hardware-Ausstattung
 - Einbinden unterschiedlicher Meßgeräte
 - Einbinden von Geräten verschiedener Hersteller
 - Einbinden von IC-spezifischer Hardware
3. Universell in der Auswertung
 - Berechnung und graphische Auswertung verschiedener Meßdaten

Auf diese Anforderungen zugeschnitten ist die Software VEE von Hewlett-Packard. Dieses Programm kann auch problemlos in das bestehende Konzept der FH Ulm eingebunden werden.

2 Allgemeines über HP VEE

HP VEE (visual engineering environment) ist eine aus Symbolen aufgebaute Programmiersprache und dient zur Lösung von meßtechnischen Problemen. Das Programm bietet die Möglichkeit, ohne konventionell zu programmieren (Textprogrammierung), Daten zu sammeln, zu analysieren und darzustellen. HP VEE beinhaltet HP VEE-Engine und HP VEE-Test. Im Zusammenspiel ergibt sich dann das Gesamtprogramm HP VEE. Um ein Meßprogramm zu erstellen, werden Objekte in die Benutzeroberfläche geladen. Diese Objekte werden durch graphisches Verbinden zu einem Modell zusammengefaßt. So entsteht ohne Textprogrammierung ein lauffähiges Meßprogramm.

Übersicht über alle Funktionen der Entwicklungsumgebung von HP VEE

Bild 1

| File | Edit | Flow | Device | I/O | Data | Math | AdvMath | Display | Help |
|-------------------|---|------------------|------------------|-------------------|-------------------|-----------------|---------------------|-------------------|------------------|
| New | Cut | If/Then/Else | User Object | Instrument ... | Enum | Formula | Array → | AlphaNumeric | On Features ... |
| Open ... | Copy | Conditional → | Function → | Advanced I/O → | Toggle | +*/ → | Matrix → | Logging AlphaNum | On Instruments . |
| Merge ... | Paste | Repeat → | Sequencer | Bus I/O Monitor | Integer Slider | Relational → | Calculus → | Meter | How To ... |
| Merge Library ... | Clone | Junction | Timer | Configure I/O | Real Slider | Logical → | Regression → | XY Trace | Glossary ... |
| Save | Delete Line | Do | Random Number | To → | Constant → | Bitwise → | Data Filtering → | Strip Chart | Short Cuts ... |
| Save As ... | Clean Up Line | Gate | Random Seed | From → | Globals → | Real Parts → | Probability → | Complex Plane | On Help ... |
| Save Objects ... | Line Probe | Sample & Hold | Virtual Source → | To/From NamedPipe | Build Data → | Complex Parts → | Statistics → | X vs Y Plot | On Version ... |
| Preferences → | Select Objects | Confirm (OK) | Counter | Execute Program | UnBuild Data → | String → | Freq Distribution → | Polar Plot | Release Notes |
| Show Description | Move objects | Delay | Accumulator | Print Screen | Allocate Array → | Generate → | Bessel → | Waveform (Time) | |
| Print Screen ... | Add To Panel | Start | Shift Register | | Access Array → | Power → | Hyper Bessel → | Spectrum (Freq) → | |
| Print All ... | Create User Object | Exit Thread | DeMultiplexer | | Access Record → | Polynomial → | Signal processing → | Beep | |
| Print Objects ... | Edit User Function... | Exit User Object | Comparator | | Concentrator | Trig → | | Note Pad | |
| Secure | View Globals ... | Stop | | | Sliding Collector | Hyper Trig → | | | |
| Exit | Breakpoints | Raise Error | | | Collector | Time & Data → | | | |
| | <input type="checkbox"/> Show Data Flow | | | | | | | | |
| | <input type="checkbox"/> Show Exec Flow | | | | | | | | |

2.1 HP VEE-Engine und HP VEE-Test

HP VEE-Engine wurde für Ingenieure und Wissenschaftler entwickelt. Man kann damit Daten analysieren und darstellen. Die Daten können sowohl auf einem File gespeichert, vom Benutzer eingegeben oder mathematisch erstellt worden sein. Dieser Programmteil dient allein der Simulation.

HP VEE-Test wurde speziell für den Test und für Messungen konzipiert. Das Programm enthält alle Komponenten von VEE-Engine, kann aber zusätzlich noch mit Meßgeräten kommunizieren.

Diese Vorgänge geschehen alle von der graphischen Benutzeroberfläche aus. Die verschiedenen Menüs von VEE-Test und VEE-Engine sind in Bild 1 auf Seite 2 dargestellt.

2.2 Das Konzept von HP VEE

2.2.1 Modelle

Um ein spezielles Problem zu lösen, wird ein graphisches Modell erstellt. Es handelt sich dabei um ein sofort ausführbares Blockdiagramm. Hier ist auch der Hauptunterschied zur Textprogrammierung zu suchen. Bei der Textprogrammierung wird in einem Blockdiagramm das Problem zuerst definiert, dann muß diese Definition codiert werden. Mit VEE muß nur das Blockdiagramm erstellt werden. Dieses ist dann sofort ausführbar.

2.2.2 Objekte

Die einzelnen Bausteine in einem Blockdiagramm werden Objekte genannt. Die einzelnen Objekte sind immer gleich strukturiert. Sie können vier verschiedene Arten von Anschlüssen haben (Bild 2). Objekte werden aus den einzelnen Menüs in die Oberfläche geladen.

Bild 2 zeigt die Anschlüsse eines Objekts

1. Dateneingänge: links am Objekt
2. Datenausgänge: rechts am Objekt
3. Steuereingänge: oben am Objekt
4. Steuerausgänge: unten am Objekt

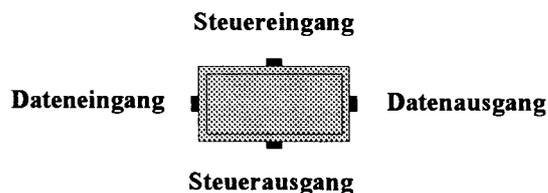


Bild 2 Anschlüsse eines Objektes

2.2.3 User Objects

Ein User Object ist ein erstelltes Modell, das als Funktion verwendet werden kann. Die Funktion hat die gleichen Ein- und Ausgänge wie ein „normales“ Objekt.

2.2.4 Datenfluß

Um Daten zu übertragen, werden Datenausgänge mit Dateieingängen verbunden. Die Verbindung der Steuerein- und Steuerausgänge ist freiwillig. Sind diese Steueranschlüsse verbunden, diktieren sie die Reihenfolge des Programmablaufs. Ein Beispiel für ein Objekt ist das If/Then/Else in Bild 3.

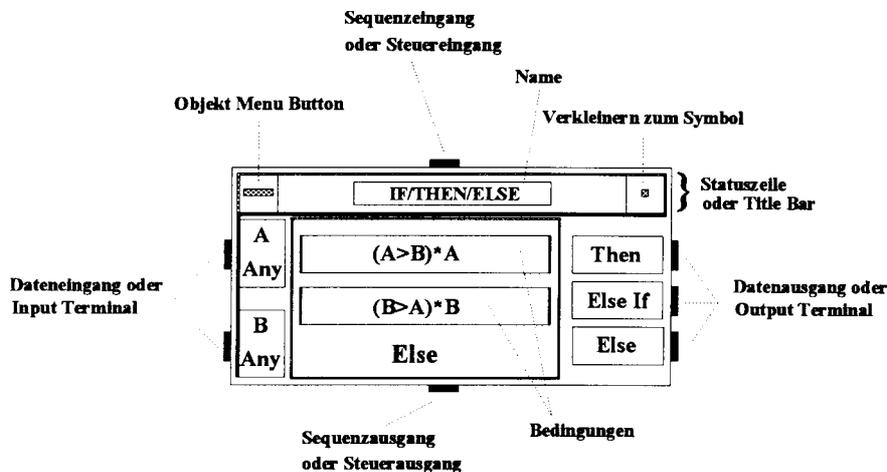


Bild 3 Das Objekt If/Then/Else

3 Das Arbeiten mit Meßgeräten

Mit dem Programm VEE-Test können Meßgeräte angesprochen werden. Diese Meßgeräte müssen eine Schnittstelle für den HP-IB Bus haben. Um diese Geräte über den HP-IB Bus ansprechen zu können, muß ein Treiber (Driver File) vorhanden sein. Meßgeräte, die von der Firma Hewlett Packard hergestellt oder vertrieben werden, besitzen bereits einen zugehörigen Treiber, bzw. sind im Programm bereits eingebunden. Hier kann zwischen 272 Treibern gewählt werden. Von diesen zur Auswahl stehenden Treibern sind allerdings nur 18 von Fremdgeräten (nicht HP).

Es gibt drei Arten von Treibern, um Meßgeräte anzusprechen:

1. State Driver
2. Component Driver
3. Direct I/O

Fremdgeräte können, sofern sie über eine IEEE-488 Schnittstelle verfügen, in das Programm aufgenommen werden. HP VEE stellt hier eine Möglichkeit zur Verfügung, einen eigenen Instrumententreiber zu erstellen.

3.1 State Driver

Mit dieser Art von Treiber kann ein Meßgerät kontrolliert werden, ohne daß man die Syntax des Gerätes kennt. Um mit dem State Driver arbeiten zu können, muß allerdings ein driver file vorhanden sein. Es ermöglicht die automatische Programmierung des Meßgerätes.

State Driver arbeiten als Fernbedienung für die Frontplatte eines Meßgerätes. Dies ist eine nützliche Eigenschaft beim Entwickeln von Modellen, beim Debuggen oder wenn das Meßgerät keine Bedienoberfläche hat.

Bild 4 zeigt die verschiedenen Treiber am Beispiel eines Spektrumanalysators.

3.2 Component Driver

Man nennt jeden gemessenen Wert und jede Funktion einer Treiberroutine „component“. Der Component Driver liest oder schreibt nur die ausgewählten Komponenten. Diese werden als Eingangs- oder Ausgangsterminals festgelegt. Während beim State Driver für viele oder gar alle Variablen Werte geschrieben werden, werden beim Component Driver nur die ausgewählten Funktionen benutzt. Mit dieser Art von Treibern wird also die Ausführungsgeschwindigkeit des Programms erhöht. Dieser Vorteil wird mit einem etwas höheren Programmieraufwand erkauft. Der Treiber ist in Bild 4 ersichtlich.

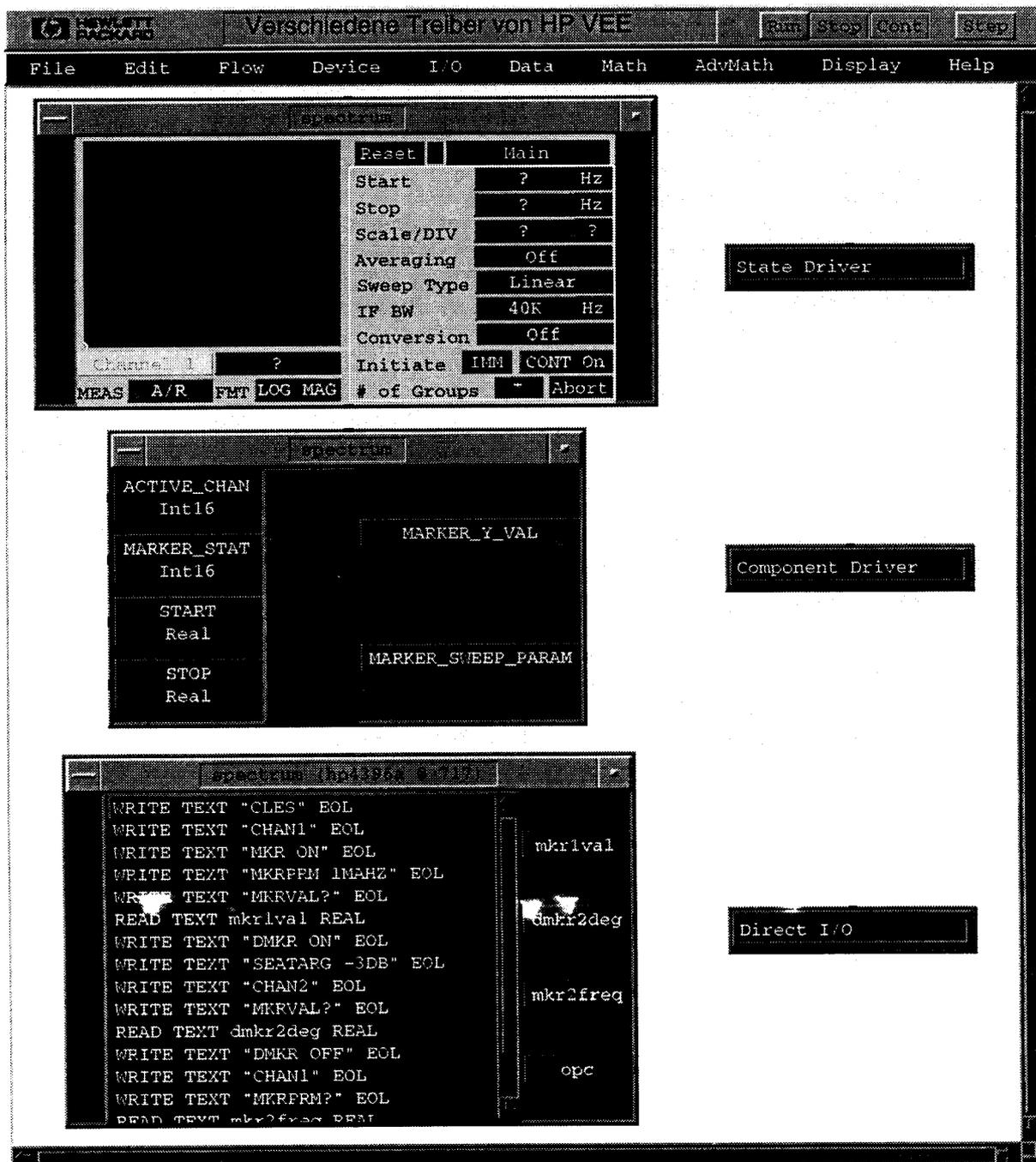


Bild 4 Verschiedenen Treiber am Beispiel eines Spektrumanalysators.

3.3 Direct I/O

Bei dieser Art von Treiber muß kein driver file vorhanden sein. Mit einem Direct I/O Treiber kann das Meßgerät direkt programmiert werden. Man kann willkürlich Daten vom Meßgerät lesen oder auf das Meßgerät schreiben. Dies ist vor allem dann empfehlenswert, wenn die Befehlssyntax des Gerätes bekannt ist und die Messung immer gleich abläuft. Direct I/O Treiber können nämlich gegenüber State Drivern in der Ausführungsgeschwindigkeit bis zu 10mal schneller sein. Die Kenntnis der Syntax des Meßgeräts ist allerdings Voraussetzung.

Bild 4 auf Seite 5 zeigt die verschiedenen Treiber am Beispiel eines Spektrumanalysators.

3.4 HP DWT - Driver Writer's Tool

Dieses Benutzerprogramm dient zum Erstellen von Instrumententreibern für Geräte, die noch nicht Bestandteil der Bibliothek sind. Es ist unter HP VEE-Test installiert. VEE-Engine unterstützt dieses Programm nicht. Um eine Treiberroutine schreiben zu können, muß der Befehlssatz des Meßgerätes vorliegen.

Wird das Programm DWT aufgerufen, befindet man sich im Learning Mode. Hier kann man sich mit dem Programm vertraut machen. Man wird menügeführt durch das Programm geleitet. Über dieses Tool gibt es keine Handbücher. Die Online-Hilfe des Programms selber ist aber gut, so daß man sich recht schnell einarbeiten kann. Wechselt man nun in den Expert Mode, kann eine Treiberroutine geschrieben werden. Man kann hier spezielle Treiber für ganz bestimmte Anwendungen spezifizieren.

4 Gesamtinstallation

Sind für alle Meßgeräte, die in einer Messung benötigt werden, Treiber vorhanden, können die Meßgeräte an den HP-IB Bus angeschlossen und angesprochen werden. Die Gesamtinstallation, wie sie derzeit an der FH Ulm vorliegt, ist in Bild 5 dargestellt. Sie enthält ein Netzgerät mit 4 Analog-Ausgängen, einen Funktionsgenerator, einen Pulsgenerator, ein 4-Kanal Oszilloskop, einen Spektrum-/Netzwerkanalysator, einen Zähler, zwei Digitalmultimeter und eine Relaismatrix.

Die Testsoftware ist auf einer HP Workstation der Serie 9000 installiert. Sie kommuniziert über den HP-IB Bus mit den Meßgeräten, und mit der IC-spezifischen Beschaltung, welche sich auf dem DUT-Board (device under test board) befindet. Der Thermostreamer ermöglicht es, die Tests bei definierten Temperaturen durchzuführen.

Da der Rechner an das Ethernet angeschlossen ist, können die Testergebnisse direkt den auf anderen Rechnern erzielten Simulationsergebnissen gegenübergestellt werden. Somit ist Chipdesign, Testen und Messen optimal miteinander verknüpft.

Bild 5 auf Seite 7 zeigt die Gesamtinstallation, wie sie an der FH Ulm verwendet wird.

Gesamtinstallation

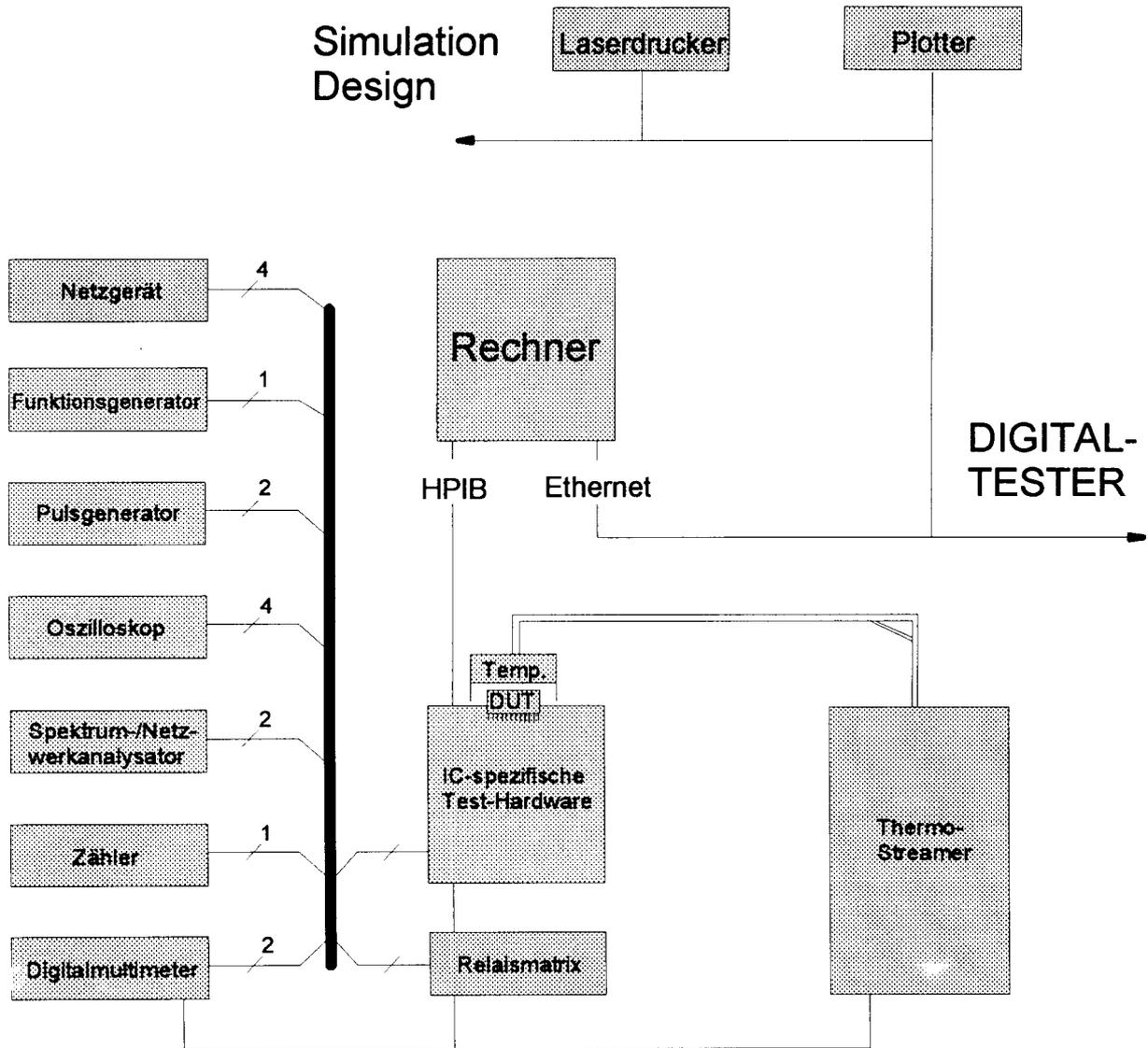


Bild 5 Gesamtinstallation an der FH Ulm

5 Aufgabenstellung

Nach dem Aufbau des Meßplatzes sollte ein Analog-IC, speziell ein Breitbandverstärker für eine Oszilloskop-Eingangsstufe, getestet werden. Um die erforderlichen Messungen festzulegen ist es nötig, die Spezifikationen des Meßobjekts zu kennen. Daraus kann je nach Aussagekraft der einzelnen Meßparameter Aufschluß über die Funktion des Bausteins gegeben werden. So ist es wichtig, die entscheidenden Messungen zuerst durchzuführen. Die Messung der Stromaufnahme sagt wohl am meisten über die Funktion des ICs aus. Bei zu niedriger Stromaufnahme liegt wahrscheinlich ein Fehler des Chips vor, z.B. Ausfall einer Baugruppe, oder die Anschlüsse sind nicht richtig kontaktiert. Dies gilt ebenso für zu hohen Strom. Hier kann nur ein Defekt vorliegen, und die Stromzufuhr muß sofort abgebrochen werden. In beiden Fällen erübrigt sich eine weitere Messung.

Für den Oszilloskop-Eingangsverstärker ergeben sich folgende zu messende Parameter:

- 1.) Stromaufnahme
- 2.) Bodediagramm der Verstärkung
 - Frequenzgang
 - niederfrequente Verstärkung bei 1 MHz
 - Bandbreite (f-3dB)
 - maximale Verstärkung ($V_{max, f}$)
 - Phasengang
- 3.) Kompressionspunkt (V_{-1dB})
- 4.) Impulsantwort
 - Amplitude
 - Anstiegs- und Abfallzeit
 - Überschwinger
 - Slewrate

5.1 Spezifikationen des Breitbandverstärkers

Der Verstärker läßt sich im wesentlichen durch folgende Merkmale charakterisieren:

- zweistufiger Verstärker
- Bandbreite > 200 MHz
- acht verschiedene Verstärkungen (1/4 ... 10) programmierbar über 3 Bit
- Eingangsimpedanz von 50 Ω
- geringe Welligkeit in der Verstärkung
- geringe Überhöhung im Frequenzgang
- hohe Flankensteilheit bei maximaler Aussteuerung

Der Chip wurde in Diplomarbeiten der vorigen Semester entwickelt und enthält außer dem Oszilloskop-Eingangsverstärker mit 3-Bit codierbarer Verstärkung noch einen Operationsverstärker, einen Impedanzwandler und sonstige Testschaltungen. Der Breitbandverstärker wurde bereits bei einem früheren Workshop vorgestellt [1].

5.2 Testplatine

Um den Chip zu testen ist eine äußere Beschaltung nötig. Diese wurde auf einer speziellen Testplatine realisiert, um das beste Übertragungsverhalten auch im Frequenzbereich bis zu einigen 100 MHz zu gewährleisten. Außerdem wurden die Abmessungen der Platine so gewählt, daß eine Messung am Waferprober vorgenommen werden kann.

Die Rückseite ist bis auf einige Durchverbindungen vollkommen mit Kupfer kaschiert, um zum einen eine gute Masseverbindung zu gewährleisten und um die Signalleitungen gegen HF-Einstreuungen abzuschirmen. Auch größere Flächen auf der Bestückungsseite sind mit Massepotential verbunden. Dabei wurden die Versorgungsanschlüsse mit Kondensatoren gegenüber HF-Störungen abgeblockt. Die Ein- und Ausgangssignale wurden so nah wie möglich mit Koaxialkabel an die Pins herangefahren, um der HF-Tauglichkeit gerecht zu werden. Der Eingang ist mit einem 50 Ω -Widerstand an den Wellenwiderstand der Leitung möglichst reflexionsfrei angepaßt.

6 Programmierung des Testprogramms mit HP VEE

Bild 6 stellt das Modell der Gesamtmessung dar. In der Gesamtmessung werden alle in der Aufgabenstellung (Punkt 5) aufgezählten Parameter gemessen. Im Folgenden wird hier nur auf die Messung der Stromaufnahme und des Kompressionspunkts näher eingegangen.

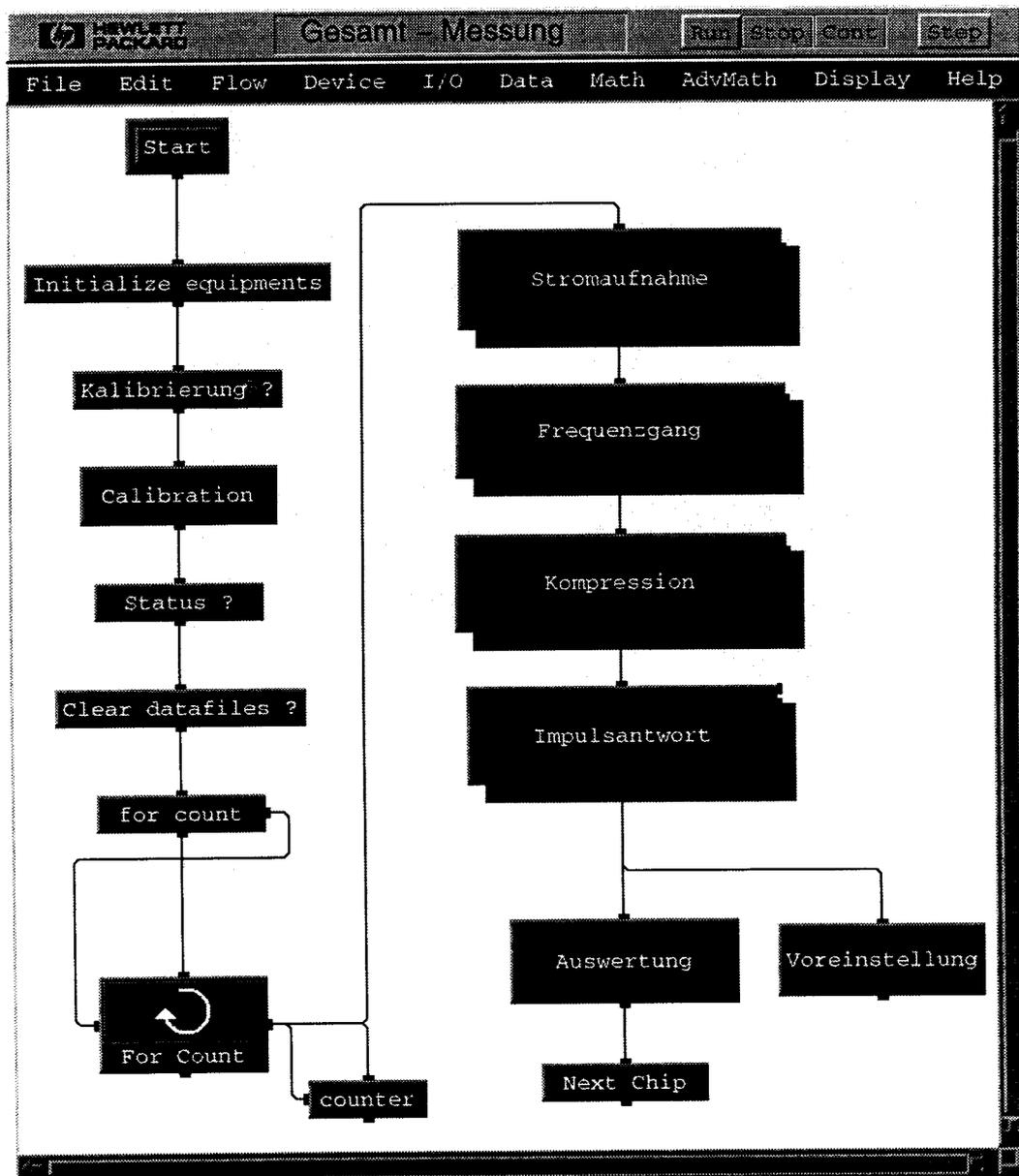


Bild 6 Modell der Gesamtmessung

6.1 Stromaufnahme

Die Messung der Stromaufnahme eines Integrierten Bausteins hat die oberste Priorität in der Reihenfolge der zu messenden Parameter, da ihr, wie bereits erwähnt, die größte Aussagekraft zugesprochen wird. Bei großen Abweichungen des Stroms vom erwarteten Wert kann über die richtige Funktion des ICs entschieden werden. Bei einem zu kleinen Wert (Strom $\rightarrow 0$) liegt entweder ein Kontaktfehler an den Anschlußpins oder ein Funktionsfehler auf dem Chip vor. Bei einer sehr hohen Stromaufnahme (Strom $\rightarrow \infty$) muß auf einen Kurzschluß, entweder durch die Kontaktierung, oder auf dem IC, geschlossen werden.

Beide Varianten, zu hohe bzw. zu niedrige Stromaufnahme, machen eine weitere Messung unnötig, das heißt der Chip ist defekt, und es kann zum nächsten Meßobjekt übergegangen werden. Weiterhin ist bei dieser Messung darauf zu achten, daß der Wert der Versorgungsspannung auf seinen Endwert eingeschwungen ist, oder ob Schwankungen in der Stromaufnahme festzustellen sind.

Für die Strommessung wird folgender Algorithmus angewandt: Es werden zwei Messungen zu verschiedenen Zeitpunkten durchgeführt. Wenn die Differenz der beiden Meßwerte kleiner als der festgelegte Toleranzwert ist, wird der zuletzt ermittelte Wert als Meßergebnis ausgegeben. Wird aber der Toleranzwert nicht eingehalten, so werden weitere Meßwerte vom Multimeter eingelesen, bis ein stabiler Endwert erreicht ist.

Wie bei Stromschwankungen üblich kann ein genauer Wert nicht bestimmt werden, wenn der Toleranzwert nicht unterschritten wird. Deshalb wird nach einer bestimmten Anzahl von Messungen der Vorgang abgebrochen und eine entsprechende Meldung ausgegeben ("current instable"). Bei zu kleinem oder zu großem Stromwert erfolgt die Ausgabe "low current" bzw. "high current".

Es wird die Stromaufnahme an der positiven und an der negativen Spannungsversorgung gemessen, um eine eventuelle starke Abweichung beider Werte beobachten zu können. Bild 7 zeigt das mit VEE-Test erstellte Modell zur Messung der Stromaufnahme.

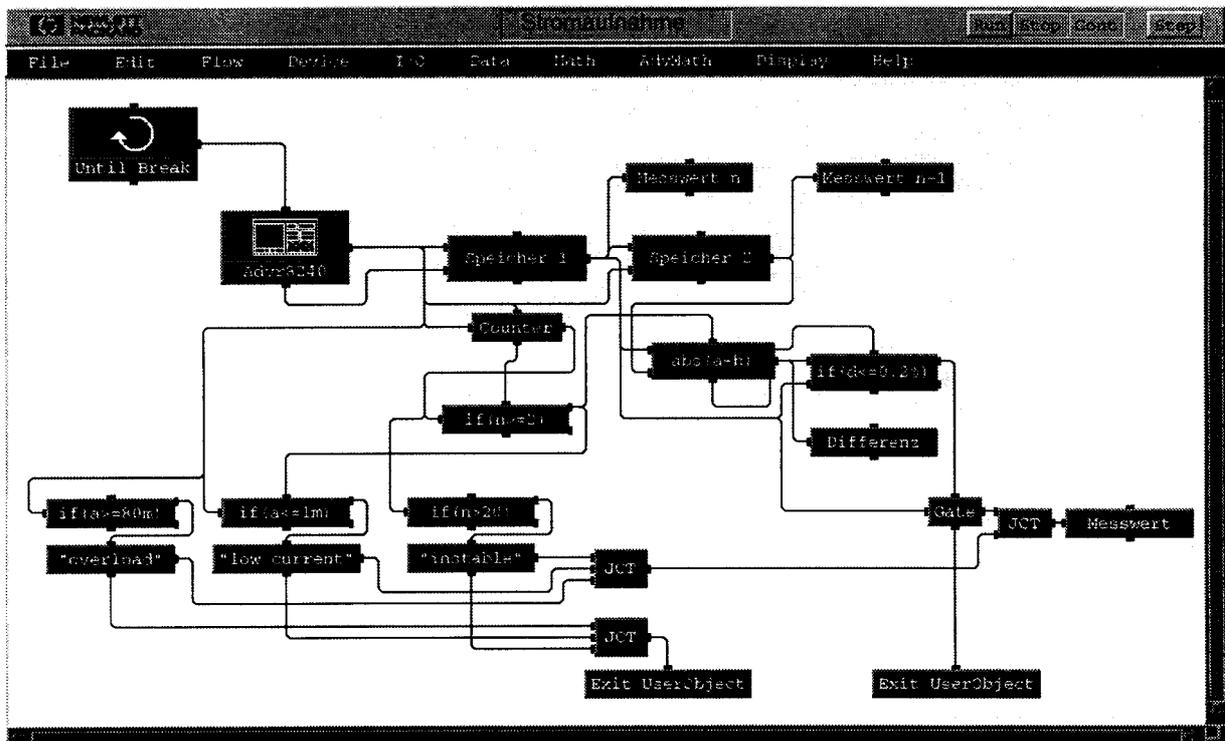


Bild 7 Modell für die Messung der Stromaufnahme

6.2 Kompressionspunkt

Der Kompressionspunkt ist definiert als diejenige Eingangsleistung, bei welcher die Spannungsverstärkung gegenüber ihrem Kleinsignalwert um 1 dB abgefallen ist. Das heißt, daß die Ausgangsleistung nicht mehr linear der Eingangsleistung folgt, bzw. der Eingang übersteuert wird. Die Bilder 8 und 9 beschreiben den Kompressionspunkt.

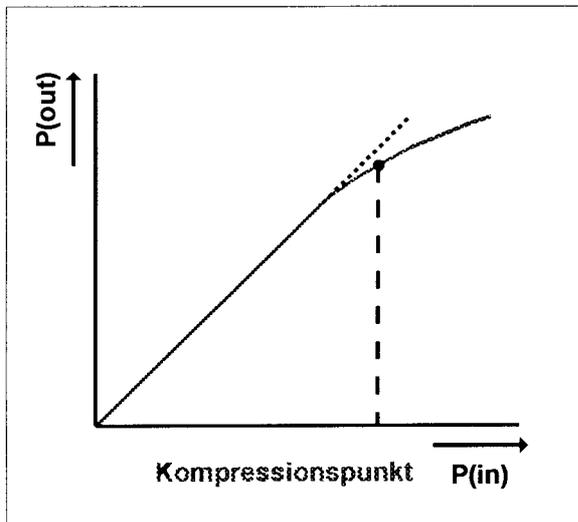


Bild 8 Kompressionspunkt $P(\text{out}) / P(\text{in})$

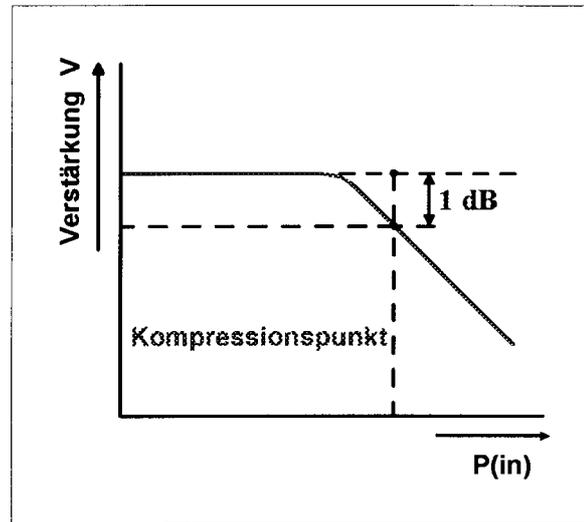


Bild 9 Kompressionspunkt $V / P(\text{in})$

Die Messung des Kompressionspunkts wird bei allen acht möglichen Verstärkungen durchgeführt. Die eigentliche Meßroutine wird dabei 8mal durchlaufen. Vor jeder Messung wird über die digitale Spannungsquelle die jeweilige Verstärkung (über 3-Bit) eingestellt. Es ist darauf zu achten, daß sich die Verstärkung am Chip bereits vor der eigentlichen Messung eingestellt hat.

Weiterer Aufwand muß bei der Voreinstellung jeder Messung betrieben werden, denn für die verschiedenen Verstärkungen müssen jeweils andere Meßbereiche eingestellt werden. Außerdem werden zwei unterschiedliche Kalibrierdaten benötigt, da nur über einen Bereich von 30 dBm kalibriert werden kann, der Power Sweep sich aber im Bereich von -40 bis +10 dBm bewegt. Der Meßablauf muß also so organisiert werden, daß zuerst bei Verstärkungen mit der dazugehörigen Kalibrierung 1 gemessen wird und anschließend mit der Kalibrierung 2. Dies wirkt sich natürlich auch negativ auf die Meßzeit aus, wird aber durch die Optimierung der Reihenfolge in Grenzen gehalten.

Bild 10 auf Seite 12 stellt das mit .E erstellte Modell zur Messung des Kompressionspunktes dar.

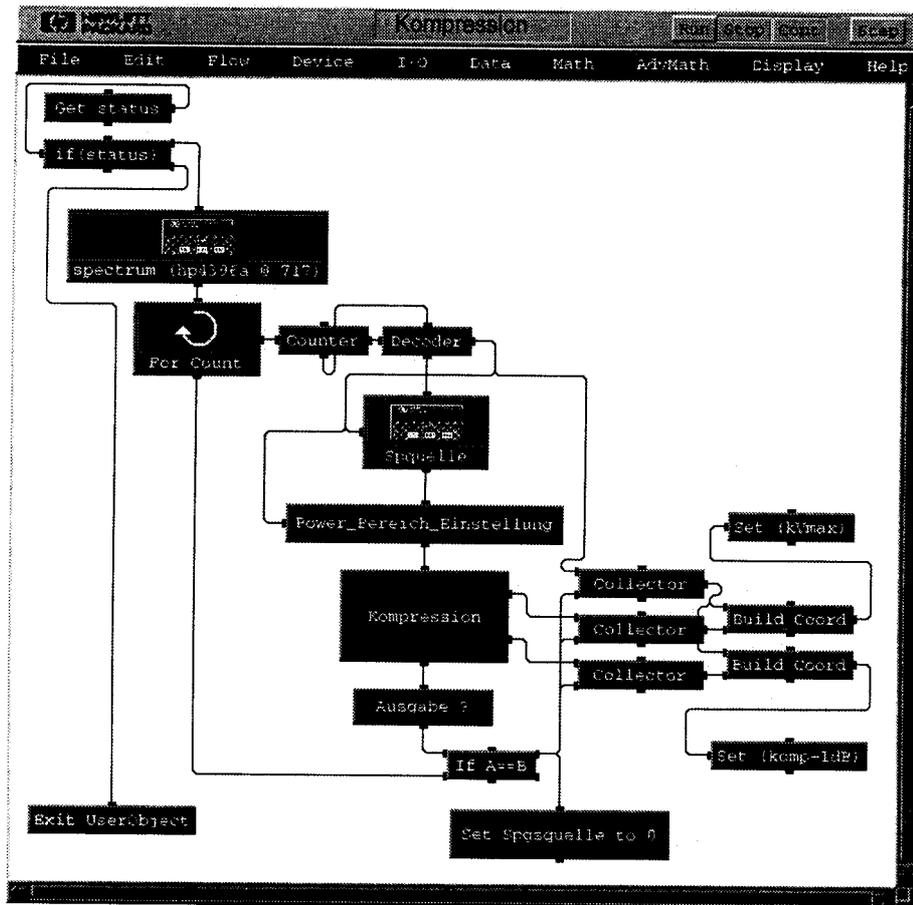


Bild 10 Modell für die Messung des Kompressionspunktes

7 Meßergebnisse

7.1 Stromaufnahme

Die Stromaufnahme des ICs beträgt ca. 52 mA ($I(+vcc) \approx 52 \text{ mA}$, $I(-vcc) \approx 50 \text{ mA}$). Diese Werte sind sehr hoch, lassen sich aber durch den Betrieb bei hoher Verstärkung im Frequenzbereich bis ca. 400 MHz begründen. Die maximale Stromdichte beträgt ca. 1 bis 2 mA pro minimaler Emitterfläche. Weiterhin muß erwähnt werden, daß sich auf der Schaltung ein Ringoszillator befindet, dessen Stromaufnahme entsprechend der hohen Taktfrequenz von 70 MHz bei etwa 20 mA liegt. Für den Breitbandverstärker allein ergeben sich an der positiven Spannungsversorgung 21,5 mA für $I(+vcc)$ und an der negativen Versorgung 17,5 mA für $I(-vcc)$.

Da der Chip mit allen Einzelkomponenten an die Versorgungsspannung angeschlossen ist, wurde bei der Messung auch die gesamte Stromaufnahme gemessen. Dabei können auch Fehler in anderen Schaltungskomponenten des Chips detektiert werden.

7.2 Bodediagramm mit Auswertung

Bei der Messung des Verstärkungsfrequenzgangs kann man sofort erkennen, daß die Abstufung der acht Verstärkungen nicht linear verläuft. Es handelt sich hierbei um einen Dekodierfehler auf dem Breitbandverstärker-IC, der aber beim neuen Design behoben wurde. Leider war der neue Chip zur Zeit der Diplomarbeit noch nicht fertiggestellt.

Meßergebnisse:

| V | V(1MHz) [dB] | f(-3dB) [MHz] | Vmax [dB] | f(Vmax) [MHz] | max Δφ [deg] |
|------|-----------------|------------------|--------------|------------------|-----------------|
| V(0) | 2,47 | 331,09 | 3,36 | 184,73 | -302,9 |
| V(1) | 23,61 | 319,61 | 24,81 | 197,33 | -310,4 |
| V(2) | 25,25 | 315,06 | 26,53 | 178,44 | -306,3 |
| V(3) | 22,64 | 316,78 | 23,50 | 160,43 | -311,1 |
| V(4) | 14,40 | 329,66 | 15,53 | 191,03 | -292,7 |
| V(5) | 35,01 | 255,76 | 36,09 | 209,93 | -281,5 |
| V(6) | 35,87 | 243,85 | 37,12 | 72,37 | -278,0 |
| V(7) | 34,19 | 259,41 | 35,52 | 203,63 | -283,2 |

Aus den Meßergebnissen kann man erkennen, daß die geforderte Bandbreite von 200 MHz leicht eingehalten wird. Allerdings tritt bei der größten Verstärkung V(6) eine maximale Überhöhung von 1,25 dB schon bei einer Frequenz von 70 MHz auf. Bei den Meßkurven mit niedrigerer Verstärkung verläuft der Verstärkungsfrequenzgang bis etwa 70 MHz linear (siehe graphische Auswertung Seite. Die maximale Überhöhung beträgt dort 1,3 dB. Der Frequenzgang unterhalb von 1 MHz kann als linear betrachtet werden.

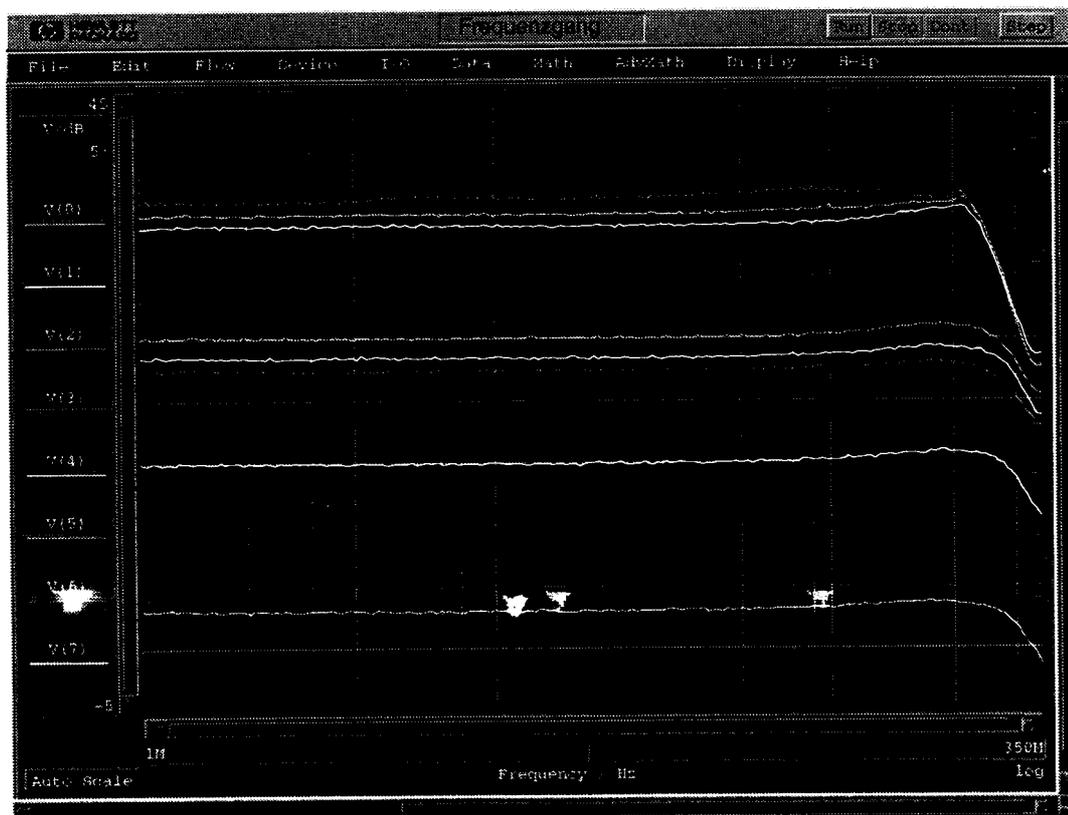


Bild 11 Ergebnisse aus der Messung des Frequenzgangs (bei allen acht Verstärkungseinstellungen)

7.3 1dB-Kompressionspunkt

Die Meßergebnisse beziehen sich auf die Frequenz von 1 MHz:

| V | Vmax bei Kompression [dB] | 1dB-Kompressionspunkt | |
|------|---------------------------------|-----------------------|-------|
| | | [dBm] | [mV] |
| V(0) | 2,45 | + 6,37 | 466 |
| V(1) | 23,81 | -15,40 | 38,0 |
| V(2) | 25,52 | -17,17 | 31,0 |
| V(3) | 22,97 | -14,82 | 40,6 |
| V(4) | 14,40 | - 5,60 | 117 |
| V(5) | 35,44 | -26,91 | 10,09 |
| V(6) | 36,64 | -28,80 | 8,12 |
| V(7) | 34,44 | -25,68 | 11,63 |

Die Meßkurven Bild 12 stellen die Abhängigkeit der Verstärkung gegenüber der Eingangsleistung dar. Man kann sehr gut die verschiedenen Meßbereiche erkennen. Um die Eingangsstufe des Netzwerkanalysators nicht zu übersteuern, dürfen die Eingangsleistungsniveaus bei den verschiedenen Verstärkungseinstellungen nicht zu hoch angesetzt sein. Die 1dB-Kompressionspunkte befinden sich in dieser Darstellung genau 1 dB unterhalb der maximalen Verstärkung.

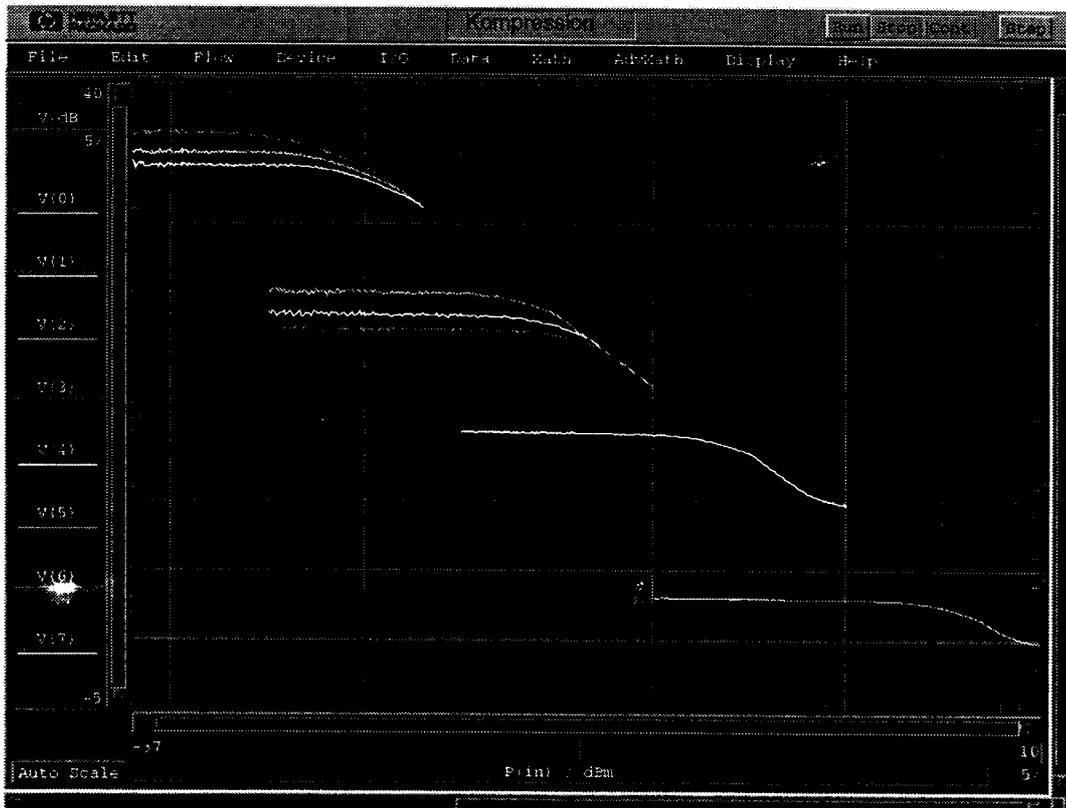


Bild 12 Ergebnisse aus der Messung des 1dB-Kompressionspunkts (bei allen acht Verstärkungseinstellungen)

7.4 Impulsantwort



Bild 13 Ergebnisse aus der Messung der Impulsantwort (Eingangs- und Ausgangssignal)

Die Impulsantwort wurde nur bei der digitalen Verstärkung $V(0)$ gemessen. Da das Signal durch Einstreuungen von UHF-Sendern verrauscht war, wurde hier mit der Average-Funktion über 100 Messungen gemittelt. Bei einem Zeitbereich von 50 ns ist der Zeitbedarf für 100 Mittelungen minimal.

| Messung | Meßwert |
|-------------------------|------------|
| Ausgangsamplitude | 1,86 V |
| Anstiegszeit t_{rise} | 1,23 ns |
| Abfallzeit t_{fall} | 4,60 ns |
| positiver Überschwinger | 30,8 % |
| negativer Überschwinger | 3,6 % |
| positive Slewrate S^+ | 1,210 V/ns |
| negative Slewrate S^- | 0,323 V/ns |

Die Steilheit der positiven Flanke mit 1,21 V/ns ist sehr gut. Es zeigt sich, daß der Verstärker sehr schnell ist, was auch die hohe Bandbreite schon aussagt. Allerdings ist der Wert der negativen Slewrate mit 0,32 V/ns um den Faktor 3,75 bedeutend niedriger. Dieser Effekt ist der Ausgangsstufe (Emitterfolger mit Stromspiegel) des Verstärkers zuzuschreiben. Dabei wird der maximale Entladestrom durch den Stromspiegel begrenzt.

$$\left(\frac{dU}{dt}\right)_{\max} = \frac{i_{MAX}}{C_L} = SR$$

Am Ausgang des Breitbandverstärkers liegt eine kapazitive Last an. Diese Last setzt sich aus folgenden Bestandteilen zusammen:

- Tastkopf (≤ 3 pF)
- Leitungskapazität
- Anschlußpads und Bonddrähte
- Gehäusepins

Die gesamte Kapazität wurde dabei mit etwa 4 pF abgeschätzt. Der eingeprägte Strom am Stromspiegel beträgt 1 mA. Daraus ergibt sich eine theoretische negative Slewrate von 250 mV/ns.

8 Zusammenfassung

Die Testsoftware HP VEE ermöglicht dem Benutzer in relativ kurzer Zeit Meßprogramme zu erstellen, da in dieser Hochsprache bereits sehr komfortable Funktionen und Objekte vorhanden sind. Als sehr positiv haben sich die graphischen Auswertemöglichkeiten und die vielfältigen Mathematik-Funktionen erwiesen. Weiterhin können auch Programme in HP Basic oder C eingebunden werden.

Auch die Möglichkeit eigene Gerätetreiber zu erstellen, muß besonders hervorgehoben werden. Allerdings wurde bereits auf den bedingten Einsatz von State Drivern hingewiesen, da hier sich zeitliche Probleme für die Meßroutinen ergeben.

Negativ ist aufgefallen, daß der Meßablauf mit realen Meßgeräten nicht simuliert werden kann. Das Programm ist nur ausführbar, wenn die Geräte physikalisch über den HP-IB Bus angeschlossen sind. Die I/O Treiber können also nicht debuggt werden, sondern nur im realen Betrieb getestet werden.

Bei der Durchführung der Gesamtmessung hat sich ergeben, daß die Meßzeiten hoch sind. Dies liegt zum Teil auch an der Hochsprache von HP VEE. Hewlett Packard ist aber bereits dabei einen Compiler zu entwickeln, um das Programm auf Quelltext zu minimieren. Zusätzlich muß auch berücksichtigt werden, daß das Bodediagramm und der 1dB-Kompressionspunkt bei allen acht verschiedenen Verstärkungseinstellungen gemessen wurden. Dabei ist dreimal das Einlesen der Kalibrierdaten notwendig. In der Produktion dagegen werden nicht mehr alle Parameter so ausführlich gemessen. Die Verwendung der Meßanordnung für kleine Stückzahlen, bzw. für die Messung von Prototypen in der Entwicklung, rechtfertigt die benötigte Meßzeit.

Bei der Messung haben sich folgende Meßzeiten ergeben:

| Messung | Kalibrierung | Meßroutine |
|---------------------------------|--------------|--------------|
| Stromaufnahme | --- | $\leq 1,0$ s |
| Bodediagramm [V(0)...V(8)] | 14,5 s | 16,9 s |
| Kompressionspunkt [V(0)...V(8)] | 17,0 s | 15,3 s |
| Slewrate | ---- | 5,3 s |
| Menü und Ausgabe | ---- | 3 s |
| Gesamt | 31,5 s | 41,5 s |

Literatur

- [1] Simulation und Design eines Breitbandverstärkers in BiCMOS-Technik, T. Ehrenbeck, M. Schmid, G. Forster, MPC-Workshop, Jan. 1994, Ulm

Ein Demonstrationsprogramm zur Problematik der Temperaturverteilung auf integrierten Schaltungen

Gerhard Albert, Fachhochschule für Technik und Gestaltung Mannheim

Juni 1995

1 Einleitung

Seit der Anfangszeit der monolithischen Integration nahm man näherungsweise an, daß auf einem Mikrochip eine überall konstante Temperatur herrsche. Diese Annahme wurde durch 3 Voraussetzungen begründet :

1. sehr kleine Chipfläche ($< 1mm^2$)
2. geringe in Wärme umgesetzte Leistung
3. recht hohe Wärmeleitfähigkeit von Silizium (etwa $1/3$ so hoch wie bei Kupfer).

Übrigens benutzt das im Entwurfsprozeß von integrierten Schaltungen fast ausschließlich eingesetzte Schaltungssimulationsprogramm SPICE bzw. davon abstammende Nachfolgeprogramme auch heute noch für alle Bauelemente eine einheitliche Temperatur.

Die beiden ersten Annahmen sind inzwischen völlig ins Gegenteil umgekehrt, sodaß die Wärmeableitung und die damit im engen Zusammenhang stehende Temperaturverteilung auf einem Chip zu einem die Komplexität begrenzenden Faktor geworden ist. Dies gilt besonders für den Bereich der digitalen integrierten Schaltungen, da hier die hohe Packungsdichte der Bauelemente (mehrere Millionen Transistoren auf einem Chip) zu erheblichen Verlustleistungen und entsprechenden abzuführenden Wärmeleistungen führt.

Bei analogen Schaltungen kommt zum Problem der Wärmeabführung noch das der Temperaturverteilung hinzu. So ist die Anordnung der Bauelemente auf dem Chip von Bedeutung, da die meisten Halbleiterbauelemente sehr stark temperaturabhängig sind und dadurch Arbeitspunktverschiebungen oder Offsetspannungen entstehen können.

Das klassische Entwurfskonzept des 'Matching' setzt darauf, daß eng benachbarte gleichartige Bauelemente auf einem Chip auch nahezu gleiche Eigenschaften und Temperaturen aufweisen, sodaß durch die Anwendung von Brückenschaltungen bzw. durch die Zurückführung von absoluten Werten auf Verhältnisse von Bauelementwerten die Temperatureinflüsse nachhaltig ausgeschaltet werden können. Sind aber Temperaturgradienten

auf dem Chip vorhanden, so ergeben sich schwer beherrschbare Temperaturabhängigkeiten und Offsetspannungen. Abweichungen von einer konstanten Temperaturverteilung müssen deshalb möglichst genau bekannt sein, um entsprechende Gegenmaßnahmen bzw. Platzierungsänderungen vornehmen zu können.

Ziel der durchgeführten Arbeiten war, ein Demonstrationsprogramm für die Nutzung in Vorlesung und Laborübung zu erstellen, welches die Temperaturverteilung auf einer einfachen Anordnung aus Chip und Trägerschicht (Header) nach Bild 1 näherungsweise errechnen und darstellen kann. Das Programm sollte leicht auf andere Rechner übertragbar sein.

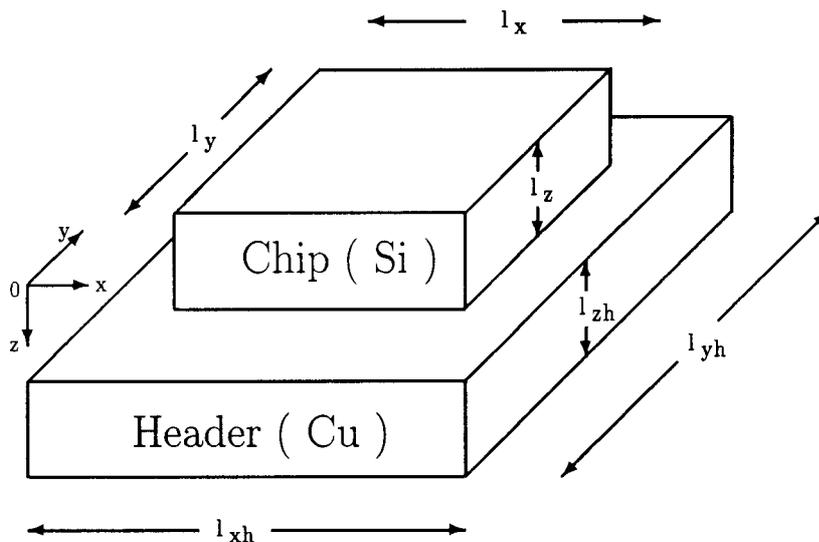


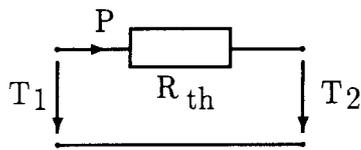
Bild 1 : Betrachtete Anordnung mit Koordinatensystem

2 Berechnungsverfahren

Bei üblichen Chiptemperaturen können Wärmeübergänge durch Strahlung und Konvektion gegenüber der Wärmeleitung vernachlässigt werden. Deshalb wird im Folgenden nur die Wärmeleitung des Chip- und Träger-Materials betrachtet. Die Klebe- bzw. Lot-schicht zwischen Chip und Trägerplatte wird vernachlässigt, könnte aber problemlos durch Erweiterung der betrachteten Anordnung auf drei Medien berücksichtigt werden. Der Wärmeübergang zwischen der Trägerplatte und Umgebung wird durch einen diskreten Wärmewiderstand R_{th} beschrieben.

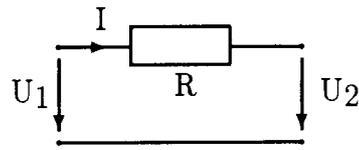
Mathematisch läßt sich die Wärmeleitung durch eine partielle Differentialgleichung mit den Abhängigkeiten von Zeit und Ort darstellen [1]. Eine numerische Lösung ist nur unter Ansatz von gravierenden Vereinfachungen möglich. Hier wird ein Näherungsverfahren verwendet, welches zur Berechnung von eindimensionalen Problemen (z.B. Kühlblechberechnungen) schon lange bekannt ist, nämlich die Analogie zwischen Wärmeleitung und

elektrischer Leitfähigkeit, siehe Bild 2.



$$P = \frac{T_1 - T_2}{R_{th}}$$

Wärmeleitfähigkeit

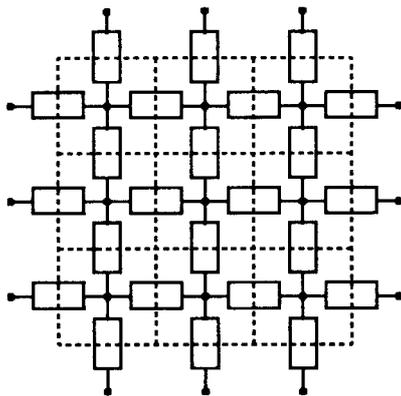


$$I = \frac{U_1 - U_2}{R}$$

Elektrische Leitfähigkeit

Bild 2 : Analogie zwischen Wärmeleitung und elektrischer Leitfähigkeit

Die eindimensionale Analogie kann auch dazu benutzt werden, mehrdimensionale Wärmeleitung zu berechnen [2], indem ein räumliches Modell der Wärmeleitung erstellt wird. Und sogar der zeitliche Verlauf der Wärmeausbreitung ist enthalten, wenn noch die Speicherwirkung der spez. Wärme durch eine elektrische Kapazität nachgebildet wird. Zur Vereinfachung wurden Einschwingvorgänge jedoch nicht betrachtet, sondern nur der eingeschwungene Zustand.



Wärmeleitwerte der Teilvolumen :

$$G_{th} = \frac{\lambda \cdot A}{l}$$

mit

λ Wärmeleitfähigkeit

A Querschnittsfläche eines Teilvolumens

l Länge eines Teilvolumens

Bild 3 : Ebenes Gitter der thermischen Ersatzschaltung

Das hier verwendete räumliche Wärmemodell erhält man, wenn man das Chip- und Trägerplattenvolumen in eine Vielzahl von quaderförmigen Teilvolumen aufteilt und jede Begrenzungsfläche durch je einen Wärmewiderstand entsprechender Größe darstellt. Die Wärmewiderstände treffen sich im Mittelpunkt der Teilvolumen, Bild 3 zeigt einen Ausschnitt (in einer Koordinatenebene) des Widerstandsnetzwerkes. In jedem Knoten treffen sich also Widerstände aus allen sechs orthogonalen Richtungen. Nach der Analogie entsprechen den Temperaturen die Knotenspannungen bezogen auf Nullpotential (Umgebungstemperatur).

Die Wärmequellen auf dem Chip werden durch ihre Lage und Leistung beschrieben.

Verteilte Wärmequellen können berücksichtigt werden, indem ihre Verlustleistung über mehrere Knoten verteilt wird. Der Leistung im thermischen Modell entspricht ein Strom, der in einen Knoten hinein fließt. Den Zusammenhang zwischen den Temperaturen, den Wärmeleitwerten und den Wärmequellen bildet die Knotenleitwertmatrix (Y-Matrix), die mit einem formalen Verfahren aufgestellt werden kann.

Die Lösung der Matrixgleichung

$$(P) = (Y_{th}) \cdot (T)$$

ergibt die gesuchte Temperaturverteilung der Knotenpunkte. Wobei (T) die Spaltenmatrix der Knotentemperaturen, (Y_{th}) die zur Hauptdiagonalen symmetrische $n \cdot n$ Knotenleitwertmatrix mit $Y_{th[i,k]} = Y_{th[k,i]}$ der Wärmeleitwerte und (P) die Spaltenmatrix der Wärmequellen ist (n ist die Anzahl der Teilvolumen).

Die Wärmeleitfähigkeit von Silizium ist selbst stark von der Temperatur abhängig. Deshalb wird die Lösung recht ungenau sein, wenn man zur Aufstellung der Matrix nur von einer mittleren Wärmeleitfähigkeit ausgeht. Viele der bisher verwendeten Berechnungsverfahren vernachlässigen diese Fehlerquelle. Verwendet man ein iteratives Lösungsverfahren für das Gleichungssystem, so kann diese Ungenauigkeit vermieden werden, wenn man während des Iterationsprozesses die thermische Leitfähigkeit in Abhängigkeit von den Zwischenergebnissen (Temperaturen) nachführt. Es erwies sich als ausreichend, die Matrix fünf mal neu zu berechnen. Die Temperaturabhängigkeit der Wärmeleitfähigkeit wurde nach [3] durch eine lineare Beziehung angenähert. Die iterative Lösung nach der 'sparse matrix'-Methode macht sich die Eigenart von Abzweignetzwerken zu Nutzen, daß von den $n \cdot n$ Matrixelementen hier nur maximal $7 \cdot n$ Elemente von Null verschieden sind. Das Unterprogramm 'sparse matrix' wurde aus [4] mit nur geringen Modifikationen übernommen.

3 Programm- Aufbau und Ablauf

Das Programm wurde in ANSI-C geschrieben und es wurde auf eine ausgefeilte Benutzeroberfläche bewußt verzichtet, um das Demonstrationsprogramm möglichst einfach auf verschiedene Rechnersysteme übertragen zu können. So konnte bis jetzt das Programm auf folgenden Rechnertypen problemlos installiert werden : HP-Apollo 425t, SUN-SPARC4, und ATARI-ST (Spielzeugrechner), bei PC's unter dem 'Betriebssystem' MSDOS 6.2 ergaben sich starke Einschränkungen bei der Anzahl der möglichen Teilvolumen (< 120). Das Programm besitzt vier Teile : Initialisierung mit Parametereingabe, Erzeugung des thermischen Netzwerkes, iterative Lösung mit Temperaturnachführung und Ergebnisausgabe. Programmtechnisch gesehen ist die Erstellung des Netzwerkes die komplizierteste Komponente, da viele Grenz- und Übergangsbedingungen geprüft und berücksichtigt werden müssen.

Die Parameter der zu simulierenden Konfiguration werden interaktiv während des Programmablaufes erfragt und auf Plausibilität überprüft. Die Ergebnisausgabe erfolgt numerisch auf Bildschirm und Datei, sodaß eine weiterführende Auswertung bzw. eine vor-

nehme graphische Darstellung daraus erstellt werden kann. Als Option kann bei der Übersetzung angegeben werden, ob das erzeugte Widerstandsnetz als SPICE-Netzliste zusätzlich zur Kontrolle in die Ausgabedatei geschrieben wird.

4 Ergebnisse

Das Konvergenzverhalten des Berechnungsverfahrens wurde anhand der Simulation eines quadratischen Chip's mit 1 mm Kantenlänge, der Dicke 0,3 mm und der Trägerdicke 0,3 mm getestet. Die Chip- und Trägerunterteilung in Dickenrichtung wurde dabei konstant zu 3 bzw. 2 angenommen. Die Unterteilung der Querabmessungen wurden zu 18, 12, 6 und 3 festgelegt. Als Wärmequellen wurden in Abhängigkeit von der Anzahl der Unterteilungen 32 Quellen mit je 1/32 W, 16 Quellen mit je 1/16 W, 4 Quellen mit je 1/4 W bzw. eine Quelle mit 1 W verwendet. Das Ergebnis dieser Betrachtung ist in Bild 4 zu sehen. Aufgetragen ist der Temperaturverlauf entlang einer den Wärmequellen abgewandten Chipkante. Der Kurvenverlauf bestätigt die Aussage nach [2], daß bei Teilvolumenabmessungen unter etwa $100\mu\text{m}$ mit brauchbaren Ergebnissen gerechnet werden kann. Es ist hier auch eine allgemein zu beobachtende Erscheinung zu sehen, nämlich, daß das Konvergenzverhalten zu den Chipecken schlechter wird, hier für $x/l_x < 0,1$ und bei $x/l_x > 0,8$. Dieses Verhalten wird verständlich, wenn man bedenkt, daß an den Chiprändern die Netznachbildung abrupt endet.

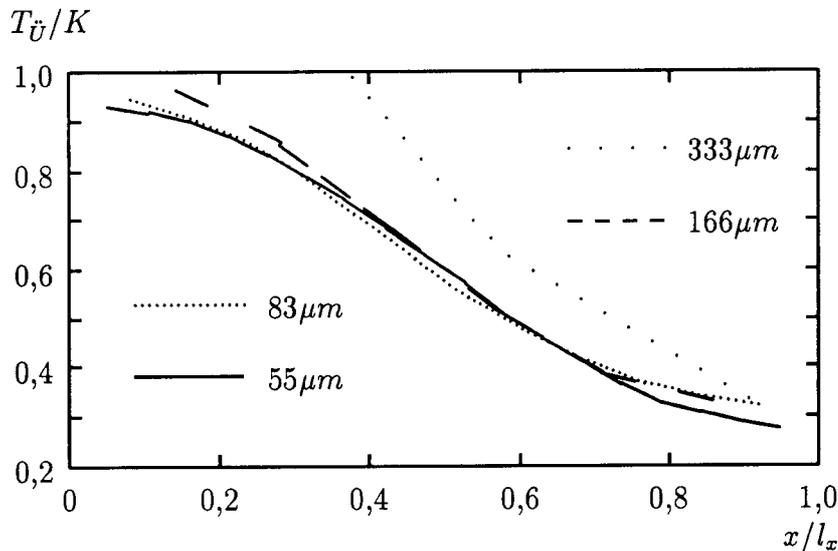


Bild 4 : Konvergenzverhalten als Funktion der Teilvolumenabmessungen

Da keine exakt nachvollziehbaren Temperaturmeßwerte gefunden wurden, waren keine absoluten Vergleiche zwischen Simulation und Messung möglich. Die Tendenz konnte jedoch bei allen Tests bestätigt werden. Exemplarisch soll noch ein Ausdruck des Demonstrationsprogrammes in Bild 5 gezeigt werden (Daten : Chip wie für Bild 4 mit $10 \cdot 10 \cdot 5$ Unterteilungen). Die Wärmequellen mit je 0,25 W liegen an den mit '*' markierten Stellen und sollen die Verlustleistung eines Endstufentransistors einer Gegentaktendstufe darstel-

len. Am linken Bildrand sind an den mit Pfeilen gekennzeichneten Stellen die Differenzverstärkereingangstransistoren an der Mittellinie angeordnet. An den durch '\$' markierten Stellen liegt der zweite Endtransistor. Wird dieser leitend und der vorher leitende Transistor gesperrt, so ergibt sich eine zur Mittellinie spiegelbildliche Temperaturverteilung. Die Eingangstransistoren haben in diesem Beispiel maximal einen Temperaturunterschied von 0,06 K, ein sehr guter Wert, da der Abstand zwischen End- und Eingangsstufe groß ist und die Eingangstransistoren nahezu auf einer Isotherme liegen.

Ergebnis (Temperaturen in Grad Celsius ueber Headertemp.) :
 Headertemperatur = 120.00 Grad Celsius
 Chipoberflaeche

| | | | | | | | | | | | |
|--|---------|------|------|------|------|------|------|--------|--------|-------|--|
| | 0.68 | 0.78 | 1.00 | 1.37 | 1.89 | 2.59 | 3.41 | 4.18 | 4.62 | 4.72 | |
| | 0.68 | 0.79 | 1.03 | 1.44 | 2.06 | 2.95 | 4.12 | 5.37 | 5.90 | 5.76 | |
| | 0.69 | 0.81 | 1.08 | 1.54 | 2.32 | 3.61 | 5.69 | 8.62 | 9.32 | 7.96 | |
| | 0.69 | 0.82 | 1.10 | 1.62 | 2.55 | 4.32 | 8.03 | *17.25 | *18.13 | 11.01 | |
| | -->0.68 | 0.80 | 1.07 | 1.57 | 2.49 | 4.24 | 7.93 | *17.12 | *17.99 | 10.86 | |
| | -->0.65 | 0.75 | 0.99 | 1.41 | 2.13 | 3.35 | 5.36 | \$8.21 | \$8.87 | 7.49 | |
| | 0.61 | 0.69 | 0.88 | 1.21 | 1.71 | 2.47 | 3.50 | \$4.61 | \$5.06 | 4.88 | |
| | 0.57 | 0.64 | 0.78 | 1.02 | 1.36 | 1.82 | 2.38 | 2.91 | 3.20 | 3.25 | |
| | 0.54 | 0.59 | 0.70 | 0.88 | 1.12 | 1.42 | 1.75 | 2.06 | 2.25 | 2.33 | |
| | 0.52 | 0.57 | 0.66 | 0.81 | 1.00 | 1.23 | 1.48 | 1.70 | 1.85 | 1.92 | |

Bild 5: Beispielausdruck (nur Ergebnis an der Chipoberkante)

5 Ausblick

Eine Diplomarbeit soll die Vertrauenswürdigkeit des Demonstrationsprogrammes anhand eines Testchips untersuchen. Weitere Arbeiten, z.B. die der meßtechnischen Bestimmung des Wärmewiderstandes zwischen Chipträger und Umgebung, der Berücksichtigung der Wärmeleitung der Bonddrähte, der Verringerung des Speicherbedarfs (es wird derzeit die Matrix vollständig gespeichert, obwohl dies nicht nötig ist, nur die Berechnung erfolgt nach der 'sparse matrix'-Methode) und der Verbesserung der Benutzeroberfläche werden durchgeführt.

Die Arbeit entstand während des Weiterbildungssemesters des Autors an der Curtin University in Perth, Westaustralien.

6 Literatur

- [1] Haybatolah Khakzar und Günter Schucht
Entwurf, Simulation und Messung von temperaturoptimierten Leistungshybriden ,
Frequenz 46 (1992)7-8 , Seite 203 ff.
- [2] Sang-Soo Lee and David. J. Allstot
Electrothermal Simulation of Integrated Circuits , IEEE Journal of Solid-State Cir-
cuits Vol. 28 No. 12 December 1993, Seite 1283 ff.
- [3] Robert J. Johnson
Thermal Rating of RF Power Transistors, AN790 Motorola Semiconductor Products
Inc.
- [4] William H. Press, Brian P. Flannery, Saul A. Teukolsky and William T. Vetterling
Numerical Recipes in C , Cambridge University Press 1991

FPGA als konfigurierbarer DMA-Controller mit Multiplexer- Demultiplexerfunktion zur Echtzeitspeicherung von Videosequenzen

Frank Imminger¹ und Heinz Sauerburger²
Fachbereich Elektronik, Fachhochschule Furtwangen

Kurzfassung

Für die Entwicklung und Untersuchung geeigneter Codierungsalgorithmen bei Bewegtbildsequenzen ist es unbedingt notwendig, unverarbeitete Videosequenzen von ca. 5 - 12 sec und zum Teil auch länger in einem Rechner zur Verfügung zu haben. Um aber den Einfluß des Algorithmusses auf die subjektive Bildqualität letztlich beurteilen zu können, muß nach erfolgter Codierung und Dekodierung die so verarbeitete Sequenz in Echtzeit wieder ausgegeben werden können. Obwohl die heutigen Rechner immer schneller werden und die Speicherkapazität zunimmt, wird trotzdem für die Echtzeitspeicherung von Videosequenzen zusätzliche Hardware benötigt. Dies liegt vor allem an der hohen Datenrate des unverarbeiteten Videosignals von 216 MBit/s bzw. 27 MByte/s und einer Speicherkapazität von 1 GByte und mehr. Da Speicherbausteine mit kurzer Zugriffszeit deutlich teurer sind, als die in großen Stückzahlen für den PC-Markt vorhandenen DRAMs, sind Speichersysteme für Videosequenzen heute immernoch relativ teuer.

Um die Halbleiterkosten und somit die Gesamtkosten eines Bildspeichersystems erheblich senken zu können, bietet sich die Parallelisierung des Videosignales an. Geht man beispielsweise auf ein Bussystem mit einer Breite von 64 Bit über, so liegt die Taktrate unter 3.5 MHz. In diesem Fall lassen sich dann die preiswerteren DRAM-Bausteine einsetzen, was sich deutlich im Preis bemerkbar macht.

Zur effizienten Umsetzung des Videosignals von ursprünglich 8 Bit für das Helligkeitssignal (Y) und weiteren 8 Bit für die beiden Farbdifferenzsignale (U, V) auf 64 Bit und umgekehrt, bietet sich heutzutage der Einsatz programmierbarer Logikbausteine mit einer

¹ cand. Dipl.-Ing. (FH) Frank Imminger, Fachbereich Elektronik, Fachhochschule Furtwangen, Gerwigstr. 11, 78120 Furtwangen.

² Prof. Dr.-Ing. Heinz Sauerburger, Fachbereich Elektronik, Fachhochschule Furtwangen, Gerwigstr. 11, 78120 Furtwangen.

Vielzahl von Anschlüssen an. Aufgrund ihrer Komplexität läßt sich neben Multiplexer und Demultiplexer auch die gesamte Speicheransteuerung integrieren, sodaß dieser Baustein auch die Aufgaben eines DMA-Controllers in sich vereint.

Bei der Auswahl einer geeigneten Bausteinfamilie hat sich gezeigt, daß neben den technischen Kriterien wie beispielsweise Anzahl der Anschlüsse, Taktraten usw. auch organisatorische Kriterien, wie die Verfügbarkeit eines Systems, leichte Erlernbarkeit und Bedienbarkeit usw. nicht außer Acht gelassen werden dürfen. Da im Rahmen eines Eurochip-Programmes die Software für programmierbare Logikbausteine der Firmen ALTERA und XILINX kostenlos zur Verfügung standen, schieden aus Kostengründen andere Anbieter aus. Aus den verschiedensten Gründen, die hier nicht näher erörtert werden sollen, fiel die Entscheidung für ALTERA.

Zum Einsatz kam ein FPGA der FLEX-8000-Familie. Die logischen Verknüpfungen sind dabei als sog. Look-Up-Tables realisiert, was zum Teil ein Umdenken notwendig machte. Da die Bausteine in SRAM-Technologie gefertigt sind, muß nach der Inbetriebnahme zuerst die Konfigurierung erfolgen. Dies mag auf den ersten Blick als nachteilig angesehen werden, aber dadurch entsteht ein weiterer Freiheitsgrad. Beispielsweise lassen sich je nach Betriebsart andere Schaltungsvarianten in das FPGA laden, die zuvor in einem EPROM gespeichert sind oder in der Testphase direkt vom PC geladen werden.

Der FPGA-Baustein, kurz VIM - Videomultiplexer - genannt, bedient zum einen die digitalisierten Videosignale für Kamera und Monitor und zum anderen den Datenbus mit einer Breite von 64 Bit, den Adreßbus mit 32 Bit und den Steuerbus (siehe "Blockschaltbild des VIM"). Ferner lassen sich verschiedene Betriebszustände unterscheiden, wie beispielsweise Videosignal-EIN, Videosignal-AUS, Farbe, Schwarz-Weiß usw.. Um die Komplexität der zu implementierenden Schaltung in Grenzen zu halten, werden die einzelnen Betriebszustände als Module im EPROM abgelegt. Auf diese Weise kommt man mit einem einzigen Baustein aus, der je nach gewünschter Betriebsart entsprechend konfiguriert wird.

Aus der internen Struktur des VIM (siehe "Interne Struktur des VIM") ist zu entnehmen, daß neben dem Multiplexer/Demultiplexer auch ein Adreßzähler, die DMA-Ablaufsteuerung und Bilderkennungslogik vorhanden sind. Da der eigentliche Bildspeicher an einem Bus betrieben wird, bei dem sowohl der Rechner als auch der VIM darauf zugreifen (siehe "Das digitale Bildspeichersystem DBS") sind DMA-Controller-Funktionen einschließlich

Busvergabesteuerungen notwendig.

Bei der Realisierung der Logik hat sich gezeigt, daß nicht immer der direkte Weg zum gewünschten Ziele führt. Da die im FPGA implementierten Automaten trotz der niedrigen Taktfrequenz des Busses mit einer deutlich darüberliegenden Frequenz getaktet werden müssen, sind interne Taktraten von 30-35 MHz notwendig. Bei den ersten Versuchen hat sich gezeigt, daß die Logik durchaus in einer "Hochsprache" beschrieben werden konnte, die daraus vom Compiler entwickelte Schaltung hielt aber den gewünschten Taktraten nicht stand. Um dieses Problem zu umgehen, wurde teilweise die Struktur der Schaltung vorgegeben, so daß dann die gewünschten Geschwindigkeiten erreicht wurden. Bei Rückfrage bei Firma ALTERA wurde uns versichert, daß solche Schwächen bei einer der kommenden Software-Updates beseitigt werden. Auch hat sich gezeigt, daß beispielsweise durch den Ripple-Carry-Counter als Zähler erheblicher Aufwand gespart werden konnte und trotzdem die Geschwindigkeitsanforderung erfüllt war.

Abschließend kann zusammengefaßt werden, daß durch die Konfigurierbarkeit der FLEX-8000-Familie zahlreiche Betriebszustände mit einem einzigen Baustein abgedeckt werden können und so der Hardwareaufwand sich trotz erzielter Flexibilität in Grenzen hält. Aufgrund geeigneter Architekturen ist der FPGA in der Lage, eine Datenrate von bis zu 27 MByte/s zu multiplexen bzw. demultiplexen und die notwendigen Bussteuerungen zu liefern. Mit Hilfe dieses Bausteines konnte somit eine sehr aufwendige und zeitkritische Schaltung relativ einfach realisiert werden. Durch die Parallelisierung ist somit die Grundlage für den Aufbau eines preiswerteren Bildspeichersystemes geschaffen.

FPGA als konfigurierbarer DMA- Controller mit Multiplexer- Demultiplexerfunktion zur Echtzeit- speicherung von Videosequenzen.

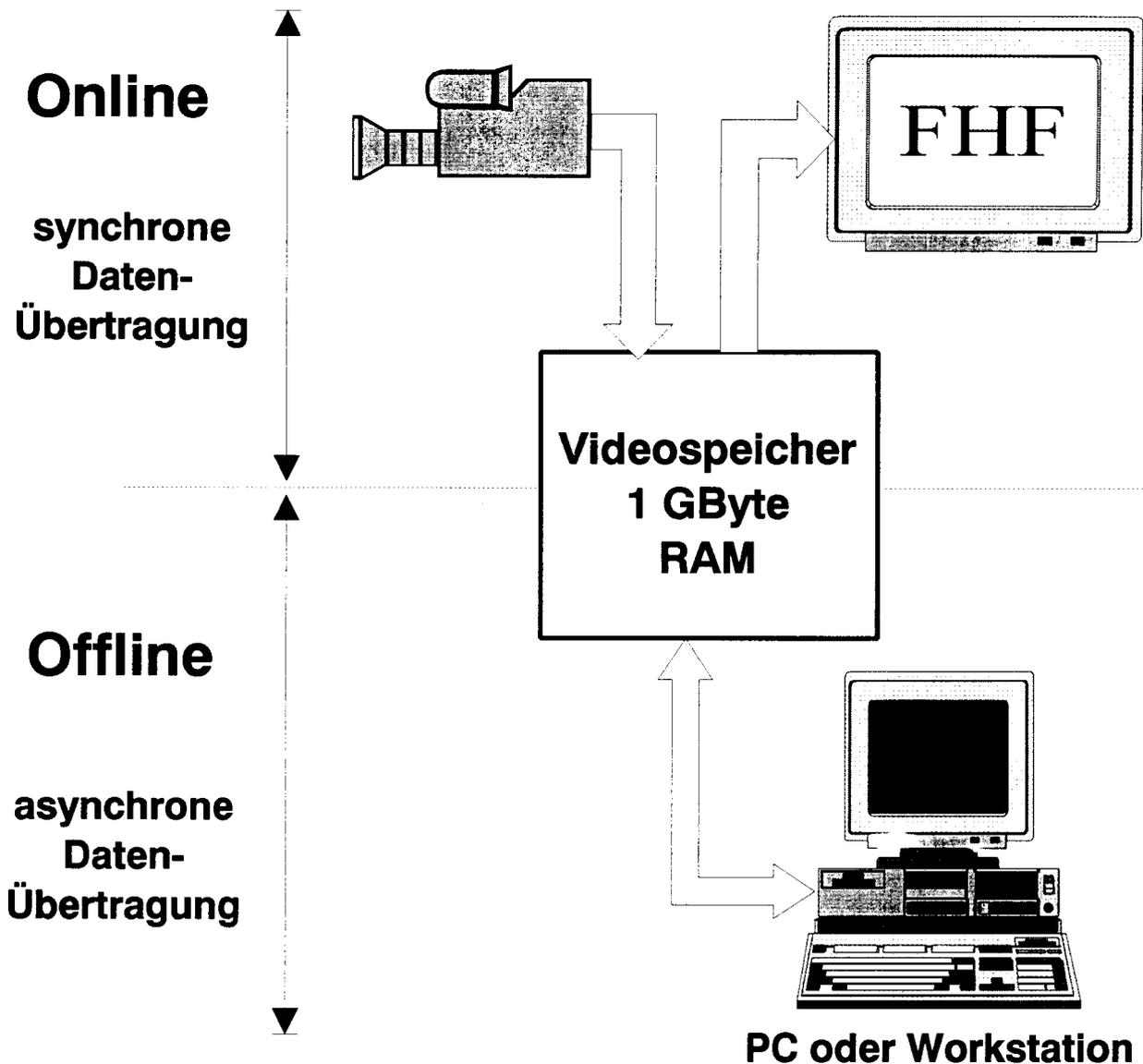
**Frank Imminger
Fachhochschule Furtwangen
Fachbereich Elektronik
Juni 1995**

Überblick:

- Aufgabe**
- Problemstellung / Lösungsansatz**
- Gesamtsystem**
- FPGA- Auswahl**
- Flex 8000 - Familie von Altera**
- Schaltungsentwurf**
- Erfahrungen mit Altera MAX + plus II**
- Zusammenfassung**

Aufgabe:

Speicherung und Ausgabe von Videosequenzen zur Untersuchung von Codier- und Komprimierungsalgorithmen

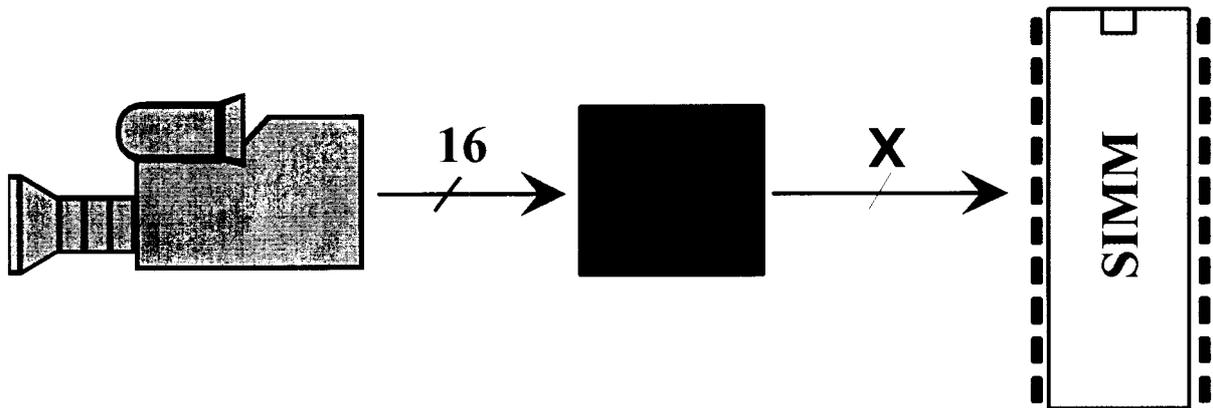


**Videodatenrate (unkomprimiert) ca: 216 MBit/s
 $\hat{=}$ 27 MByte/s**

Das Problem:

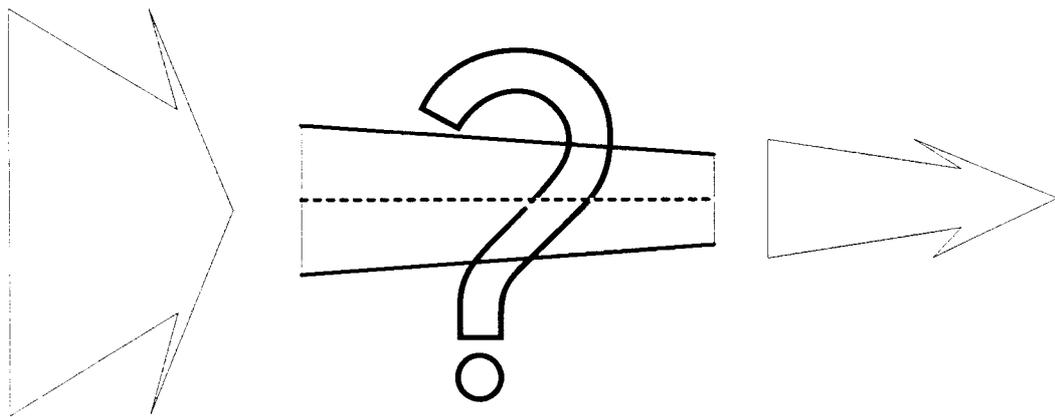
- 16 Bit YUV-Bus
- 4:2:2 Format
- 13,5 ..14,75 MHz

- handelsübliche SIMM's
- 70 bzw. 140 ns Zugriffszeit



Videodatenrate
27 MByte/s

Speicherbausteine
mit Datenrate
ca. 3,3 MByte/s

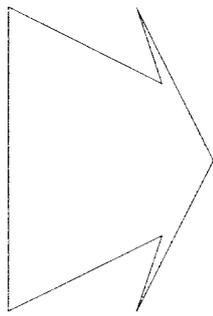


- hohe Quelldatenrate die online zu speichern ist
- geringe Datenrate des einzelnen Speicherchips

Lösungsansatz:

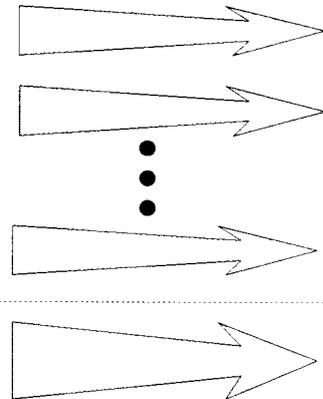
Datenrate

- 16 Bit
- 13,5 MHz



Datenrate

- 8 x 8 Bit <-> 64Bit
- 3,375 MHz



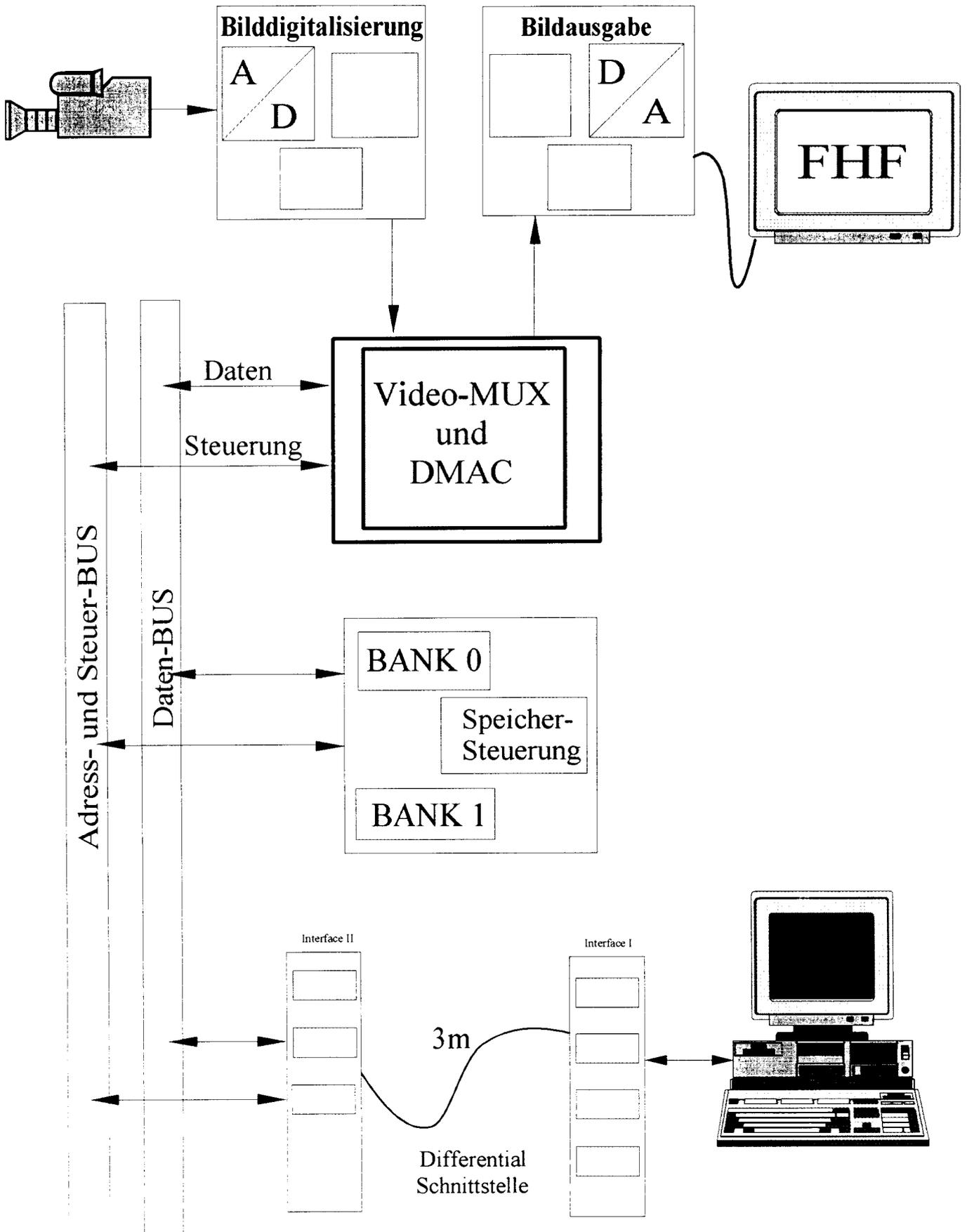
Synchronisation

32 Bit Adressbus

- Multiplexer 16 Bit <-> 64 Bit
- DMA- Controller
- Framegrabberfunktion



Das Digitale Bildspeichersystem DBS



Kriterien zur FPGA-Auswahl

Anforderungen:

- genügend User I/O-Pins
(benötigt ca. 200 Pins)
- hohe Taktrate (ca. 30 MHz)
- gleiche Laufzeiten auf Chip
- Schnelle Einarbeitung in zugehöriges Entwicklungstool
- Konfigurierbarkeit während Laufzeit mit unterschiedlichen Funktionalitäten

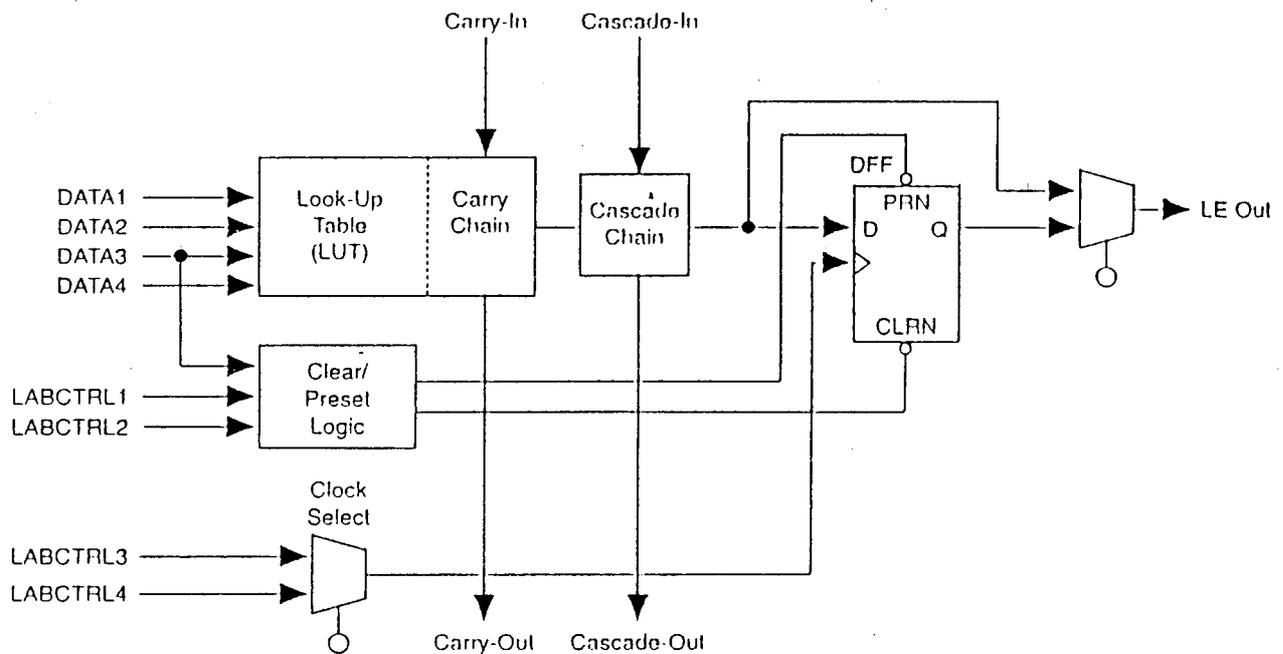
*** Auswahl wurde 1992 getroffen ***

| | Quicklogic | XILINX | ALTERA |
|---------------------------|------------|--------|--------|
| Geschwindigkeit | +++ | 0 | ++ |
| kostengünstig | -- | ++ | +++ |
| Einarbeitung | | 0 | + |
| Erfahrung | | + | ++ |
| gleiche Laufzeiten | | 0 | ++ |

=> Die FLEX8000 - Familie von ALTERA

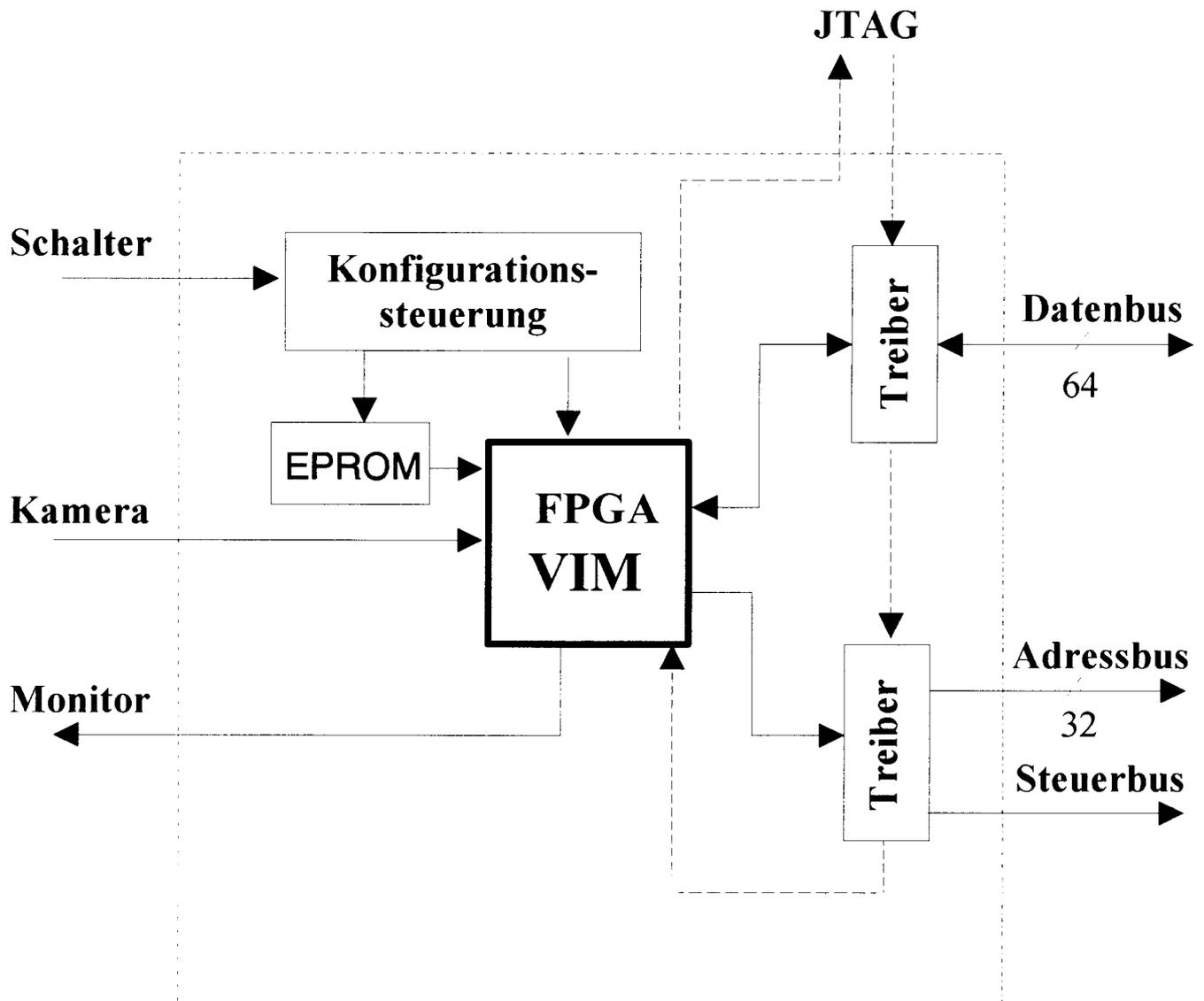
Flex 8000 - Familie von ALTERA

- SRAM- Technologie
- während der Laufzeit rekonfigurierbar
- volle Boundary Scan-Unterstützung
- Bausteine mit 78 bis 208 User I/O-Pins
- 2500 bis 16000 nutzbare Gatter



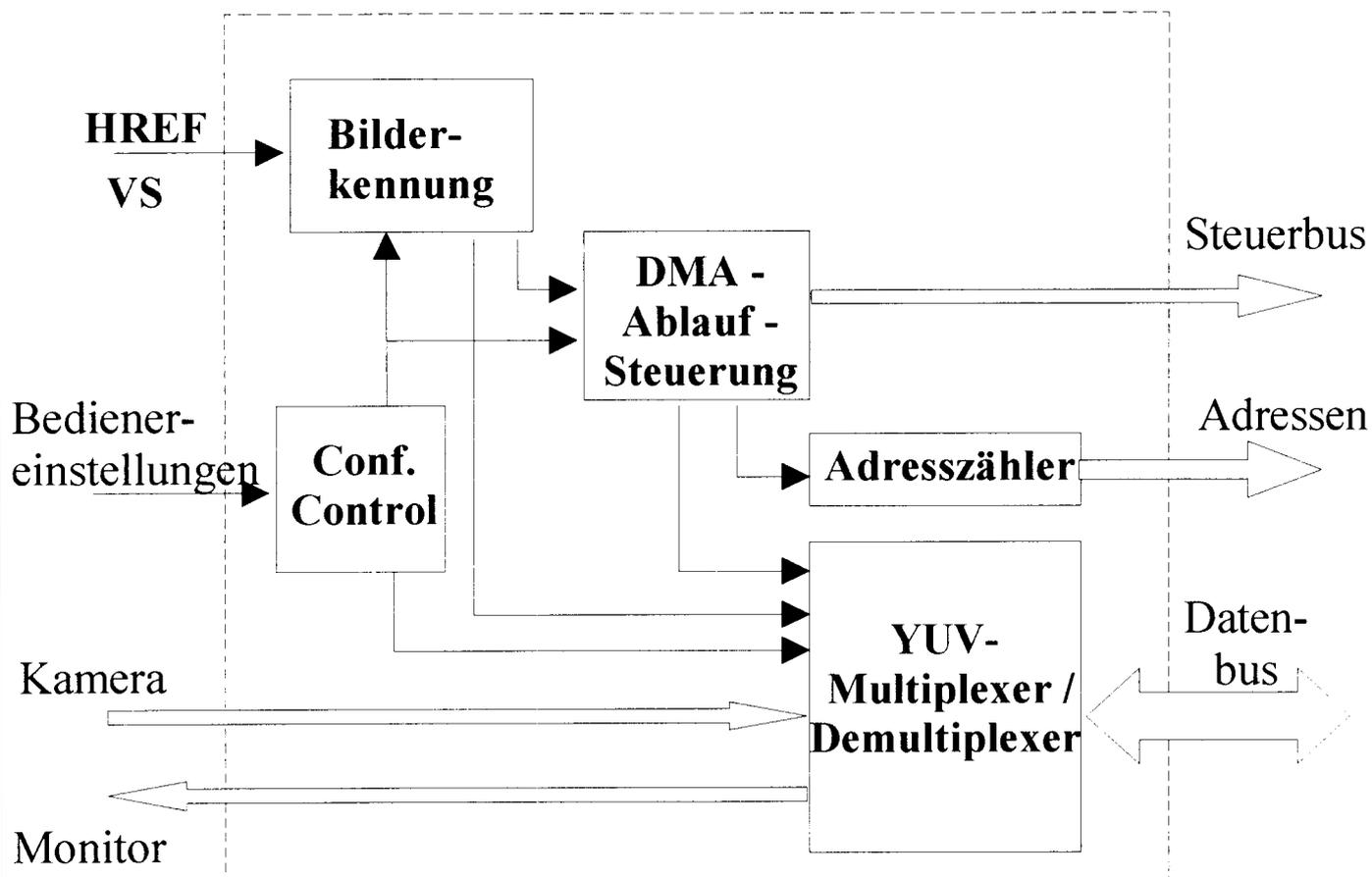
- Gleichungssynthese über LUT
- schnelle Leitungen zur Kaskadierung von Logikelementen
- schnelle interne (metallische) Leitungen in Matrixstruktur (Fast Track Interconnect)

Blockschaltbild des VIM- Boards



- **Konfiguration selbsttätig aus EPROM / in Testphase seriell vom PC**
- **acht verschiedene Konfigurationen ladbar**
- **Generierung aller DMA - Signale**
- **Umsortierung der seriellen Bilddaten (YUV)**
- **64 Bit breiter Datenbus**
- **32 Bit Adressbus => 4 GByte adressierbar**

Interne Struktur des VIM

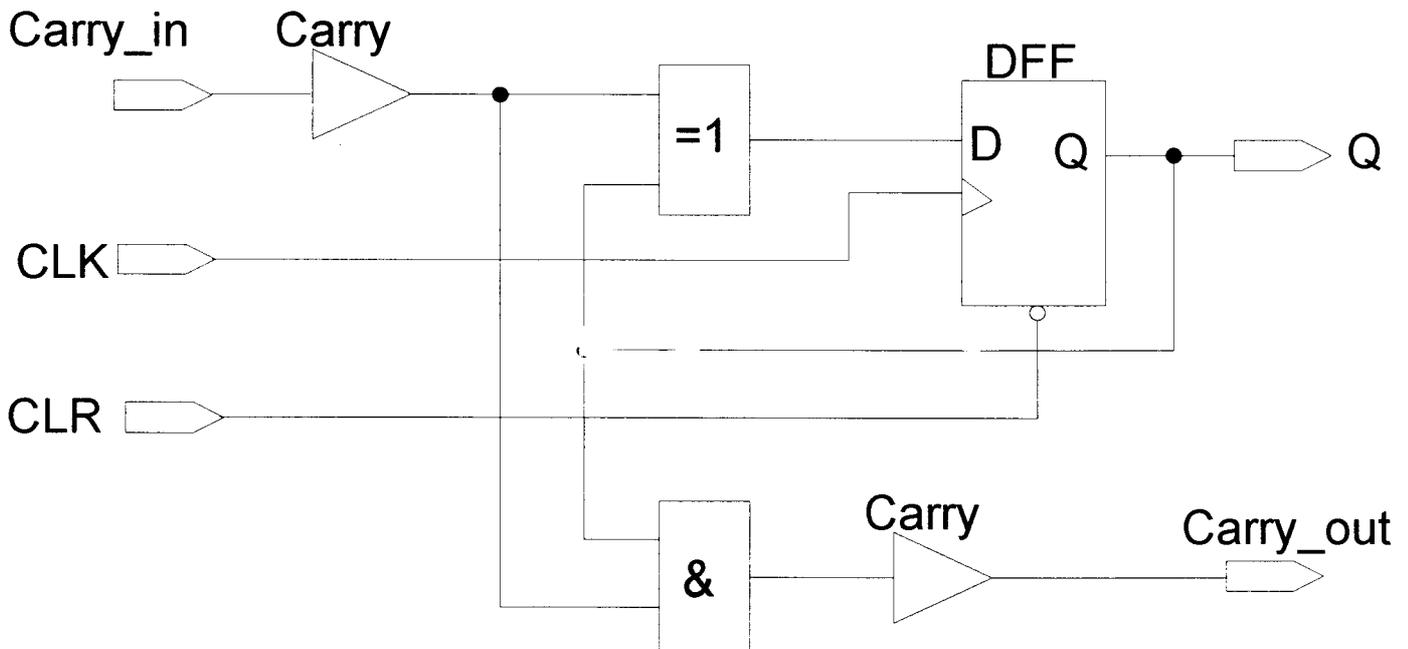


- Speicherung verschiedener Bildgrößen (Voll-, Halb-, viertel und Achtelbilder)
- Generierung eines Farbrahmens bei nicht Vollbilddarstellung
- Funktion des YUV-Multiplexer und der DMA-Ablaufsteuerung für jede Konfiguration anders
- Taktrate der Datenverarbeitung
 - * 27 MHz bei dig. Studiostandard (CCIR 601)
 - * 29,5 MHz für Square Pixel Format

Zählerstruktur

- **Ripple - Carry - Counter**
 - * **Carry-Chain liefert Übertrag innerhalb 1ns**
 - * **Geschwindigkeit des 1 Bit Zählers 200MHz**
 - * **Zähler recourcenintensiv bei Ladbarkeit**
 - * **schnellste Zählerstruktur: Prescaled Counter (sehr recourcenintensiv)**

Blockschaltbild:



Erfahrungen mit Altera MAX+plus II:

- einheitliche Pinbelegung nur wenn kein Mindestfrequenz vorgegeben

- Unter AHDL beschriebene Statemachines:
 - * werden zu langsam, wenn Anzahl der Zustände zunimmt

- => Alle Automaten nach One-Hot-Methode beschrieben
 - * Struktur direkt vorgeben
 - * etwa halbe Anzahl Logikzellen erforderlich

- wichtig, interne Struktur des FLEX8000 beim Entwurf bestimmter Schaltungsmodule zu berücksichtigen

Zusammenfassung

- **Problem des Speichern und Lesens einer Datenrate von 27 MByte/s**
- **Verwendung eines einzelnen FPGA's zum Multiplexen der Daten und Generieren der DMA-Signale**
- **Aufteilung der Gesamtfunktion auf mehrere ladbare Konfigurationen**
- **Realisierung mit FLEX8000 -Familie**
- **volle Ausnutzung der internen Strukturen des FPGA**

Vortrag zum XIII. MPC-Workshop in Offenburg

"Netzlistengenerator als Interface zwischen DA und SPICE"

Diplomand: **P. Xipteras**

Betreuer: **Prof.Dr.-Ing. G. Kampe**

Fachhochschule Esslingen - Hochschule für Technik

0. Inhalt

1. Einführung/ Zielbeschreibung
2. Der DA und die MGC-Netzlist
3. Die Datenstruktur der EDIF-File
4. Der Netzlistengenerator e2s
 - 4.1 Portabilität
 - 4.2 Die Bedienung
 - 4.3 Die Vorgehensweise
 - 4.4 Die Fehlererkennung

1. Einführung/ Zielbeschreibung

Die Entwicklung des Netzlistengenerators war ein Teilprojekt meiner Diplomarbeit.

Das gesamte Ziel war, die Leiterplatte eines D-FF aus Transistoren mit MGC-Tools zu realisieren; ein Laborprojekt zur Vorlesung Digitaltechnik der Fachrichtungen Technische Informatik und Nachrichtentechnik im 4. Semester.

Dabei ist die Vorgehensweise zum Erreichen des Ziels in Form einer Anleitung (im Lab Exercises Layout) ausführlich beschrieben, die von den Studenten Schritt für Schritt verfolgt werden muß. Die Arbeit für die Eingabe der Geometrien (phys. Abmessungen der Komponenten) wurde den Studenten vom Diplomanden abgenommen, indem er sie vorgefertigt hat.

Die Leiterplattenerstellung erfolgt in mehreren Schritten:

- Erstellung der Schaltung mit DA
 - Generierung des PSPICE-Eingabedatensatzes
 - Simulation und Optimierung der Schaltung mit PSPICE
 - Platzierung der Komponente mit MGC-Tools
 - Routen der Leiterplatte
 - Anfertigung und Testen der Leiterplatte im Labor.
- Vergleich mit den Simulationsergebnissen und Beurteilung.

Der Wunsch einer Simulation mit PSPICE liegt an der begrenzten Anzahl von erworbenen Board Station Lizenzen.

Die CAD-Schnittstelle zwischen Design Architect und PSPICE wurde durch eine Eigenentwicklung geschaffen. Es handelt sich hierbei um den Netzlistengenerator e2s.

2. Der DA und die MGC-Netzlistener

Das Softwaremodul *Design Architect* von Mentor Graphics Corporation wird in der Top-Down/ Bottom-Up Entwicklung zur Schaltplanerfassung verwendet (Folie 2). Nach der Schaltplanerstellung mit DA erfolgt die Abspeicherung der Schaltung. Allerdings ist das Abspeicherungsformat vom DA programmspezifisch. Ein Austausch von Verbindungsdaten mit anderen EDA-Systemen und Datenbasen wird nur mit Hilfe von Netzlistern gewährleistet. Diese Netzlister übernehmen die Konvertierung von DA-Schaltungen in das EDIF 2 0 0 Format in beiden Richtungen. Board Station verfügt über zwei solcher Netzlister, die Programme "enwrite" und "enread", die in die Mentor Graphics Datenbasis jeweils Verbindungsdaten ein- und auslesen. In dieser Diplomarbeit wurde das EDIF 2 0 0 Format als Schnittstelle zwischen DA und (P)SPICE verwendet (Folie 3).

3. Die Datenstruktur der EDIF-File

Die EDIF-File einfacher analogen Schaltungen wie z.B das D-Kippglied (auf Transistorebene betrachtet) besteht aus drei Abschnitten (Folie 6):

- Cell

Die Komponentenklasse (z.B Resistor) wird in diesem Abschnitt beschrieben. Der Name der Komponentenklasse und deren Anschlußpins, die Signalflußrichtung usw.. gehören dazu.

- Instance

Die Komponenten mit ihren Werteangaben werden erwähnt. Z.B gehören die Bezeichnung R1 und der SPICE-Wert 10K zu den Angaben für jedes Komponent in diesem Abschnitt.

- Net

Die Komponentenpins zu jedem Knoten werden erwähnt.

Die Komponenten und die Knoten bekommen, wenn nicht anders gewünscht, interne Namen vom DA.

4. Der Netzlistengenerator e2s

4.1 Portabilität

Das Vorhandensein der vielen unterschiedlichen Systeme an der FHTE macht einen portablen Netzlistengenerator wünschenswert. Der "e2s" erfüllt diesen Wunsch, indem es u.a. auf den Betriebssystemen DEC-Ultrix, LinuX, MSDOS und MS-Windows lauffähig ist. Alle Versionen vom e2s werden gleich bedient, die Unix-Versionen sind allerdings, wegen der überlegenen Speicherverwaltung von Unix, besser skalierbar.

4.2 Die Bedienung

Nachdem man die EDIF-Datei aus dem Schematic mit *enwrite* generiert hat, kann man *e2s* einsetzen. Eine Hilfe dazu erhält man mit: *e2s --help* auf dem DOS/Unix-Prompt.

Die Eingabezeile im Betriebssystemprompt sollte etwa wie folgt aussehen:

```
e2s EDIF_file PSPICE_file <-
```

Per Default wird eine PSPICE-Netzliste erzeugt, mit der Option *-s* wird die Generierung einer SPICE-Netzliste ermöglicht.

Nachdem *e2s* die EDIF-Netzliste geladen hat, werden die in der Schaltung gefundenen Knoten auf dem Bildschirm ausgegeben. Dann folgt die Angabe der Masseknotenbezeichnung, denn die Masse wird in der Circuit-Datei von SPICE mit einer **0** dargestellt.

Nach dem Konvertierungsende kann man die Circuit-Datei in PSpice laden und die Kontrollanweisungen je nach Wunsch hinzufügen.

Der Zusammenhang zwischen DA<->EDIF<->PSPICE wird auf der Folie Nr.6 anhand eines Beispiels erläutert.

4.3 Die Vorgehensweise

Der Netzlistengenerator *e2s* (ein C++ Programm) speichert die Abschnittsinhalte und Eigenschaften, die sich daraus ergeben (z.B Anzahl der Ports), in den Eigenschaften dreier gleichnamiger Objekte (Folie 5).

Die in den Eigenschaften dieser Objekte gespeicherten Daten werden mit Hilfe eines Algorithmus, der die SPICE-Regeln berücksichtigt, im geeigneten Format in der Ausgabedatei (.CIR) dargestellt. Dieser Algorithmus ist für zweipolige analoge Komponenten (Ports POS/NEG) und Transistoren (Ports E,B,C) konzipiert, denn das D-Kippglied des Laborversuchs besteht nur aus solchen Bauelementen.

4.4 Die Fehlererkennung

e2s geht von der Richtigkeit, der mit *enwrite* erzeugten EDIF-Netzliste, aus, um seine Tabellen aufzubauen. Aus Sicherheitsgründen wurden aber trotzdem Mechanismen eingebaut, die eventuell eine manuelle fehlerhafte Modifikation der EDIF-Netzliste erkennen. (Folie 4).

Speicherprobleme werden ebenfalls angezeigt, wenn die Anzahl die cells, instances oder nets die Kapazitäten von *e2s* überschreiten sollten. Die maximale Anzahl von 20 Bauteilarten, 100 Bauteilen und 100 Verbindungsknoten sollte als ein Vorschlag vom Entwickler betrachtet werden. Mit diesem Kompromiss kann sogar auf 16-Bit Betriebssystemen gearbeitet werden.

Bei den 32Bit-Versionen von *e2s* sind aber keine derartigen Speicherprobleme zu erwarten.

Netzlistengenerator als Interface zwischen Design Architect und Spice

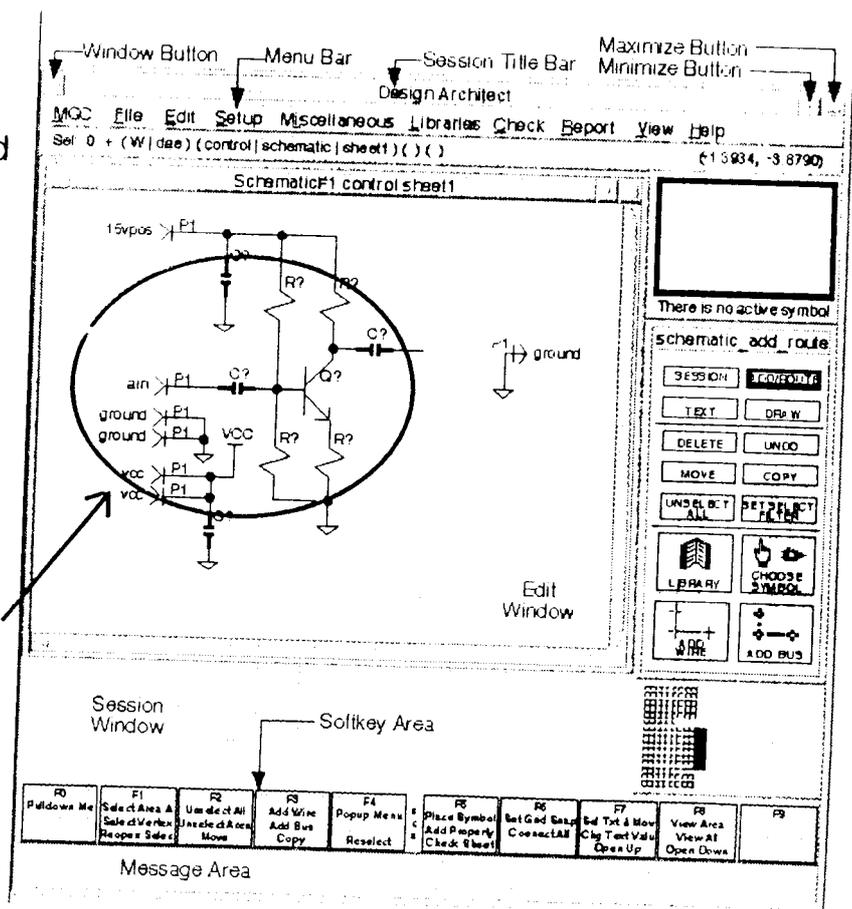


- Das Softwarepaket "Board Station" → Folie 3
- Der Weg zur Simulation → Folie 3
- Ein kurzer Blick auf das Programm "Design Architect" → Folie 2
- Eine Beispielschaltung → Folie 6
- Die Datenstruktur von EDIF → Folie 5
- Die Programmstruktur des Netzlistengenerators → Folie 4
- Zusammenfassung

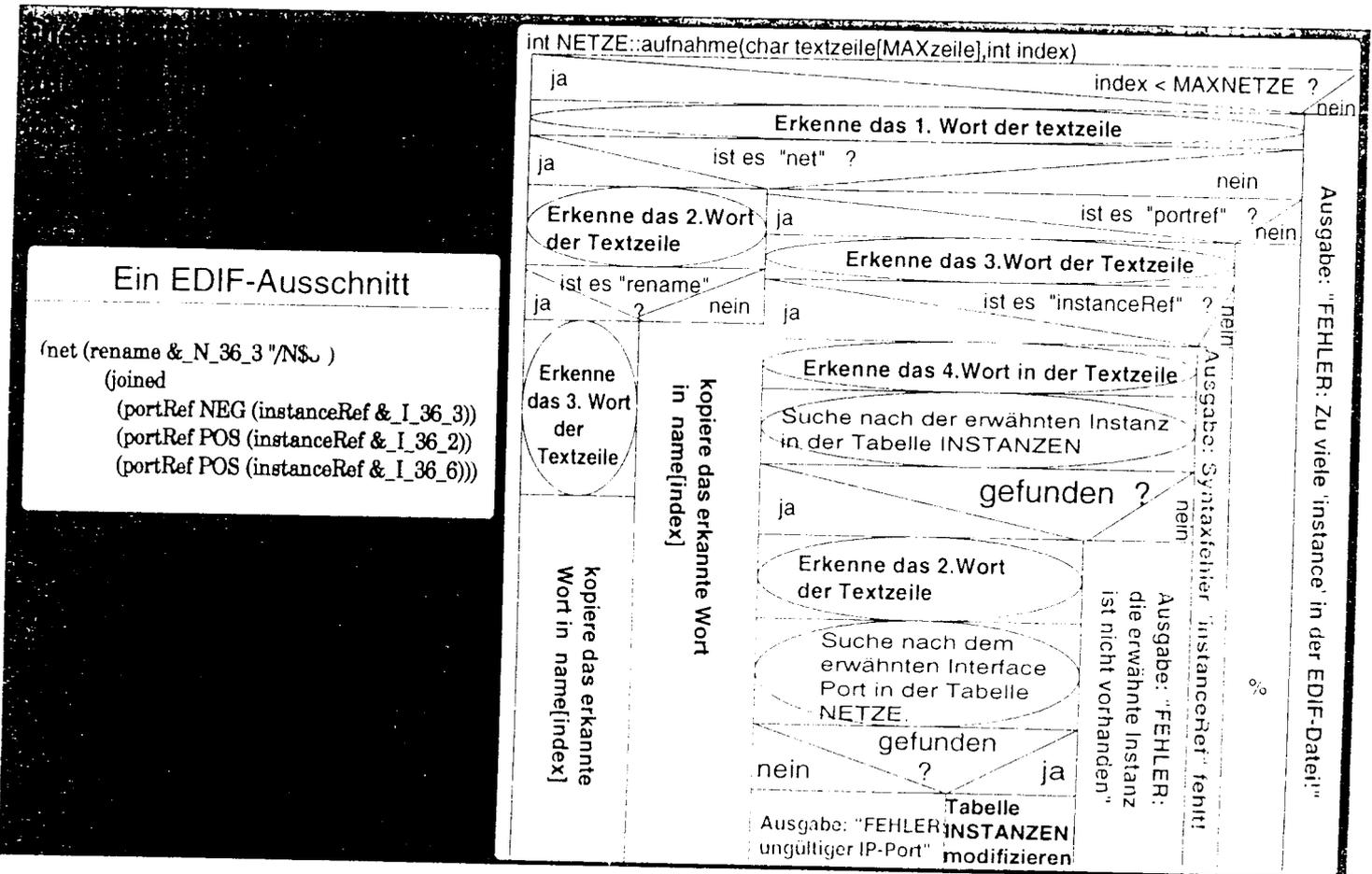
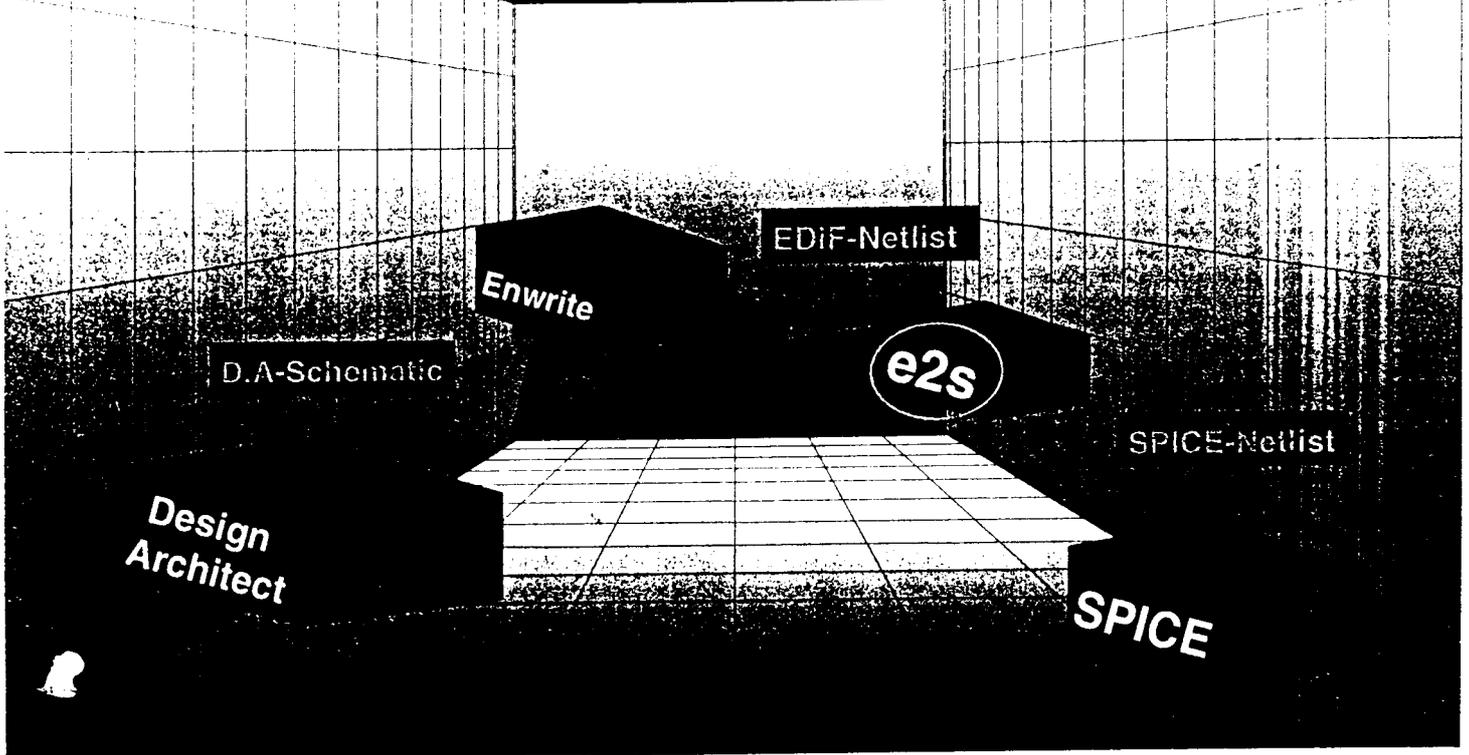
Diplomand: P. Xipteras

Der Design Architect

- Anwendung der Top-Down und Bottom-Up Entwurfsverfahren
 - Architecturebene
 - Logikebene
 - Schaltebene
- Entwurfsbeschreibungen beliebig konfigurieren (back annotations)
- Hineinzoomen in das Schematic
- Analoge und digitale Libraries verfügbar



Der Weg zur Simulation



Tabellenaufbau von e2s

EDIF-Datei

e2s in C++

```
(cell resistor_h (cellType generic)
(view resistor_h (viewType netlist)
(interface
(port NEG
(property pin (string "NEG")))
(port POS (direction INPUT)
(property pin (string "POS")))
)
(property comp (string "RESISTOR_H"))
(property element (string "R"))
(property instpar (string "10K"))
(property ref (string ""))))
```

```
(instance (rename & I_36_3 "R3")
(viewRef resistor_h (cellRef resistor_h .....
(property asim_model (string ""))
(property instpar (string "1K"))
(property model (string "ACCUSIM"))
(property ref (string "R1"))
```

```
(net (rename & N_36_3 "NS3")
(joined
(portRef NEG (instanceRef & I_36_3))
(portRef POS (instanceRef & I_36_2))
(portRef POS (instanceRef & I_36_6)))
```

```
class ZELLEN
string name;
int ipcount;
string ipname[MAXPORT];
string comp_property;
```

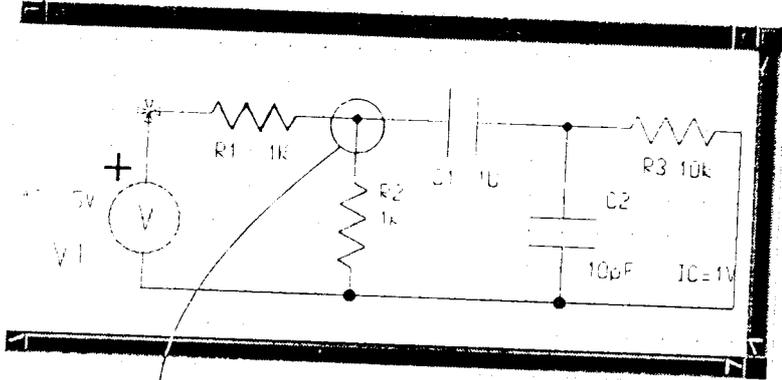
```
class INSTANZEN
int elementtyp; .....
string name;
string wert;
string instanz;
int knoten[MAXPORT];
```

```
class NETZE
int count;
string name[MAXNETZE];
```

6. Folie

Vom Schematic zum SPICE via EDIF

Schematic



PSPICE

| | | | |
|----|--------|--------|------------|
| C1 | N_36_3 | N_36_4 | 1U |
| R2 | N_36_3 | 0 | 1K |
| R3 | N_36_4 | 0 | 10k |
| R1 | N_36_2 | N_36_3 | 1K |
| V1 | N_36_2 | 0 | AC 5V |
| C2 | N_36_4 | 0 | 10pF IC=1V |

```
(cell resistor_h (cellType generic)
(view resistor_h (viewType netlist)
(interface
(port NEG
(property pin (string "NEG")))
(port POS (direction INPUT)
(property pin (string "POS")))
)
(property comp (string "RESISTOR_H"))
(property element (string "R"))
(property instpar (string "10K"))
(property ref (string ""))))
```

```
(instance (rename & I_36_3 "R3")
(viewRef resistor_h (cellRef resistor_h .....
(property asim_model (string ""))
(property instpar (string "1K"))
(property model (string "ACCUSIM"))
(property ref (string "R1"))
```

```
(net (rename & N_36_3 "NS3")
(joined
(portRef NEG (instanceRef & I_36_3))
(portRef POS (instanceRef & I_36_2))
(portRef POS (instanceRef & I_36_6)))
```



1 Bedienung komplexer Programmsysteme

Der Schaltungsentwurf komplexer digitaler Schaltungen erfordert leistungsfähige Programmsysteme, mit deren Hilfe es in kurzer Zeit gelingt, Aufgabenstellungen in konkurrenzfähige Hardware umzusetzen. LOGTUTOR ist ein interaktives Lernprogramm für die Einarbeitung in die Schaltungssyntheseprogramme LOG/iC und STATE/view der Firma ISDATA. An der Fachhochschule Heilbronn folgt der LOGTUTOR in seiner Entwicklung auf die interaktive Einführung in PSPICE, die bereits seit mehreren Semestern erfolgreich zur Vorbereitung einer Laboraufgabe eingesetzt wird.

2 Notwendige Programmierumgebung: LOG/iC Evaluationsversion

Die Studierenden der Produktionstechnik im 5. Semester an der FH Heilbronn verfügen bereits über leistungsfähige PCs (mindestens 386), auf denen die Lernprogramme unter *Windows* lauffähig sind. Wesentlicher Bestandteil der Vorgehensweise ist, daß den Studierenden zusätzlich zum Lernprogramm eine funktionsfähige und lizenzfreie Evaluationsversion von LOG/iC zur Verfügung steht. Diese ist auf das Bauelement GAL16V8 beschränkt. Sie ermöglicht den vollständigen Schaltungsentwurf bis zur Programmierdatei, mit der das Bauelement für die Laborübung programmiert werden kann.

3 Notwendige Vorkenntnisse

Die Grundlagen der Digitaltechnik (Entwurf von Schaltnetzen und Moore-Schaltwerken) sowie Funktion und Aufbau von PLDs müssen bekannt sein.

4 Durcharbeiten des interaktiven Lernprogramms LOGTUTOR anhand vollständig dokumentierter Praxisbeispiele

Die Studierenden werden beim Durcharbeiten des LOGTUTOR aufgefordert, selbständig Felder zu aktivieren und Bezeichnungen und Werte einzugeben. Die Praxisbeispiele *Tankfüllanlage* und *Codeschloß* zeigen exemplarisch die Vorgehensweise beim Schaltungsentwurf. Sie ermöglichen den Studierenden, das gelernte Wissen unmittelbar anzuwenden und zeigen darüberhinaus zielgerichtet den Weg von der Aufgabenstellung bis zur fertigen Schaltung. Um für den Einsteiger die Komplexität einzuschränken, wird nur ein Lösungsweg erlaubt. Bereits bearbeitete Programmteile können jederzeit erneut aufgerufen werden. Zusätzliche Lösungshilfen sollen den Studierenden bei Verständnisproblemen weiterhelfen. Alle Funktionen und Ergebnisse sind mit denen von LOG/iC optisch identisch. Die Hilfefunktion des Programmsystems LOG/iC ist im LOGTUTOR integriert.

5 Vertiefen des Wissens mit der Evaluationsversion von LOG/iC

Im nächsten Schritt können die Studierenden ihr Wissen vertiefen, indem sie dieselben Beispiele wie im LOGTUTOR selbständig mit der Evaluationsversion



LOGTUTOR

Interaktives Lernprogramm für die Einführung in LOG/iC mit PLD

Heilbronn, 2.8.95

durcharbeiten. Bei Fragen kann über die Windows-Oberfläche der Lösungsweg im LOGTUTOR gleichzeitig betrachtet werden, um an schwierigen Stellen Bedien- oder Lösungshinweise zu bekommen. Nach den Erfahrungen an der Fachhochschule Heilbronn verkürzt sich durch diese Parallelarbeit die Lernphase für LOG/iC erheblich.

6 Druckfunktion und Wirtschaftlichkeitsbetrachtungen

Um auch persönliche Notizen sinnvoll anfertigen zu können, kann jeder Bildschirminhalt durch Knopfdruck sofort als Protokoll ausgegeben werden. Das DIN A4 Blatt ist nur halbseitig bedruckt, sodaß genügend Raum für zusätzliche Bemerkungen bleibt.

In der Einarbeitungsphase der Studierenden entfällt das Kopieren der Handbücher. Die Kurzanleitung der Evaluationsversion von LOG/iC ist als Datei ebenfalls vorhanden und kann bei Bedarf ausgedruckt werden. Zusammen mit einer Evaluationsversion des Programmsystems wird das Selbststudium wesentlich erleichtert.

Handbücher sollen durch LOGTUTOR nicht ersetzt werden, da nur diese vollständig sind und für den geübten Anwender zu Verfügung stehen müssen.

7 LOGTUTOR ist kein Ersatz für Vorlesungen und Labor

Der LOGTUTOR ist nicht als Ersatz für Vorlesungen oder Labors gedacht. Er soll hauptsächlich die Vorlesungen und Labors von der Vermittlung der Befehlssyntax von LOG/iC entlasten. Die vielen Einzelschritte beim Schaltungsentwurf mit LOG/iC mit seinen speziellen Funktionen können mit dem Lernprogramm wirkungsvoll im Selbststudium eingeübt werden. Der Dozent kann sich dagegen auf die Lehrinhalte konzentrieren, die nicht oder nur sehr aufwendig mit einem Lernprogramm oder Büchern vermittelt werden können.

8 Zusammenfassung

Mit dem LOGTUTOR ist es Studierenden möglich, sich selbständig in die Schaltungssyntheseprogramme für digitale Schaltungen, STATE/view und LOG/iC, einzuarbeiten. Die Einarbeitung erfolgt auf der Basis des PLD GAL 16V8, mit dem sowohl kombinatorische als auch sequentielle Logikschaltungen realisiert werden können. Mit der legalen Evaluationsversion von LOG/iC kann der Studierende zuhause seine Laboraufgaben vorbereiten.



Logtutor

Eine interaktive Einführung in den Entwurf digitaler Schaltungen für das Programmsystem LOG/iC und STATE/view

Vortrag in Offenburg am 30.6.95 für die
MPC-Gruppe
der Fachhochschulen in Baden-Württemberg

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

CPLD und FPGA in der Lehre

- 10.000 nutzbare Gatter auf einem Bauelement
- Mächtige Logikblöcke (Look up Table, Multiplexer)
- Programmierung des Bauelements auf der fertigen gedruckten Schaltung (20 Jahre)
- Wiederprogrammierbarkeit über JTAG-Schnittstelle (PC-Centronics-Schnittstelle -->FPGA)
 - Rekonfigurierbarkeit: zu verschiedenen Zeitpunkten verschiedene Funktionen einer Hardware
- Prüfbarkeit durch gezielt ladbare Testlogik für Produktions- und Servicetest
- Zahlreiche Hersteller von FPGAs:
 - Actel, Altera, AMD, Lattice, Xilinx

3

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

Einarbeitung in die Bedienung komplexer Programmsysteme

- Bedienoberfläche
- Befehlssyntax
- Systematische Vorgehensweise
- Handbücher werden nicht gelesen
- Handbücher sind nicht verfügbar

4

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470



Interaktive Einführung

- Einarbeitung in LOG/IC und STATE/view vor Laborbeginn
- Selbststudium
- Einfache Beispiele für 16V8
- Bedienoberfläche und Befehlssyntax
- Paralleles Arbeiten im Programmsystem und im Lernprogramm

5

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

Vorkenntnisse

- Grundlagen der Digitaltechnik
 - Entwurfsverfahren für Schaltnetze und Schaltwerke
 - Zustandsdiagramm
 - Schaltungssynthese
 - Verifikation
- Funktion CPLD und FPGA muß bekannt sein

6

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

Nicht gewünschte Einsatzbereiche

- Kein Ersatz für Vorlesungen
- Kein Ersatz für Labors
- Kein Ersatz für Lehrgänge

7

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470



Voraussetzungen für Hard- und Software

- 386-PC mit VGA
- DOS 6.2 und Windows 3.1
- Hauptspeicher 4 MB
- Freier Plattenspeicher 8 MB

8

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

Wirtschaftlichkeitsbetrachtungen

- Experimentierschaltung für jeden Laborteilnehmer:
 - Experimentieren zuhause
- Selbststudium für Bedienung und Syntax unter Kombination von Lernprogramm und Evaluationsversion
- Programmbedienung als Aufgabe der Laborvorbereitung

9

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

Zusammenfassung

- FPGA und CPLD sind interessante Bauelemente für die Lehre
- Interaktives Lernprogramm:
 - Leistungsfähiges Medium für Bedienung komplexer Programmsysteme
 - Verringerung der Einarbeitungszeit durch Parallelarbeit mit Lernprogramm und Evaluationsprogramm

10

Prof.-Dipl.-Ing. E. Prochaska Fachhochschule Heilbronn Fachbereich Produktionstechnik
Max-Planck-Str. 39 74081 Heilbronn Telefon 01731-504-214/266 Fax 252470

