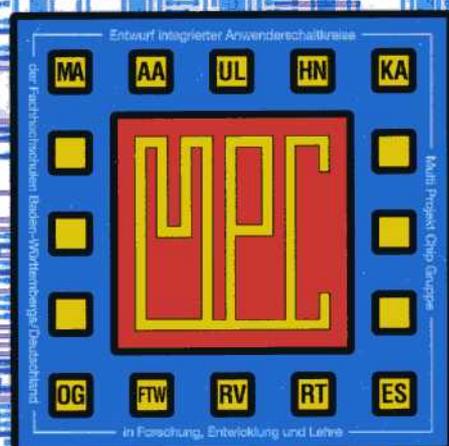


# MULTIPROJEKT CHIP-GRUPPE

BADEN - WÜRTTEMBERG

Workshop Juli 1996

Heilbronn



# **MULTIPROJEKT CHIP - GRUPPE**

## **BADEN - WÜRTTEMBERG**

**Workshop Juli 1996**

**Heilbronn**

**Herausgeber: Fachhochschule Ulm**

© 1996 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.

Adressen der

## **MULTIPROJEKT-CHIP-GRUPPE (MPC-Gruppe)**

### **BADEN - WÜRTTEMBERG**

Fachhochschule Aalen

Prof. Dr. Kohlhammer, Postfach 1728, 73428 Aalen

Tel.: 07361/576-248, Fax: -324, Email: bernd.kohlhammer@fh-aalen.de

Fachhochschule Albstadt-Sigmaringen

Prof. Dr. Rieger, Johannesstr. 3, 72458 Albstadt-Ebingen

Tel.: 07431/579-124, Fax: -149, Email: rieger@fh-albsig.de

Fachhochschule Esslingen

Prof. Dr. Kampe, Flandernstr. 101, 73732 Esslingen

Tel.: 0711/397-4221, Fax: -4212, Email: kampe@ti.fht-esslingen.de

Fachhochschule Furtwangen

Prof. Dr. Rülling, Postfach 28, 78113 Furtwangen

Tel.: 07723/920-501, Fax: -610, Email: ruelling@fh-furtwangen.de

Fachhochschule Heilbronn

Prof. Dr. Clauss, Max-Planck-Str. 39, 74081 Heilbronn

Tel.: 07131/504-401, Fax: /252470, Email: clauss@fh-heilbronn.de

Fachhochschule Karlsruhe

Prof. Ritzert, Postfach 2440, 76012 Karlsruhe

Tel.: 0721/925-2238, Fax: -2259, Email: ritzert@fh-karlsruhe.de

Fachhochschule Konstanz

Prof. Dr. Voland, Postfach 100543, 78405 Konstanz

Tel.: 07531/983-631, Fax: -613, Email: voland@fh-konstanz.de

Fachhochschule Mannheim

Prof. Dr. Albert, Speyerer Str. 4, 68136 Mannheim

Tel.: 0621/2926-351, Fax: -454, Email: gerhard@eis.fht-mannheim.de

Fachhochschule Offenburg

Prof. Dr. Jansen, Badstr. 24, 77652 Offenburg

Tel.: 0781/205-267, Fax: -333, Email: d.jansen@fh-offenburg

Fachhochschule Ravensburg-Weingarten

Prof. Dr. Klotzbücher, Postfach 1261, 88241 Weingarten

Tel.: 0751/501-630, Fax: /49240, Email: klotzbuecher@fbe.fh-weingarten.de

Fachhochschule Reutlingen

Prof. Dr. Kreutzer, Federnseestr. 4, 72764 Reutlingen

Tel.: 07121/341-108, Fax: -100, Email: kreutzer@el.fh-reutlingen.de

Fachhochschule Ulm

Prof. Führer, Postfach 3860, 89028 Ulm

Tel.: 0731/502-8338, Fax: -8363, Email: fuehrer@fh-ulm.de

# Inhaltsverzeichnis

	Seite
1. IrDA kompatible Datenübertragung J. Angerstein, TEMIC Heilbronn	5
2. Die IMS GATE FOREST 0.8 $\mu\text{m}$ CMOS Gate Array Familie J. Gandowitz, IMS CHIPS Stuttgart	11
3. 10Bit $\Sigma\Delta$ -DA-Wandler mit Altera PLD H. Töpfer, FH Esslingen-Standort Göppingen	33
4. Die Gilbert-Zelle als HF-Mischstufe H. Sapotta, FH Karlsruhe	39
5. Dimensionierung von integrierten Anlogschaltungen mit OPSIM H. J. Aleker, G. Forster, FH Ulm	51
6. Applikation des Mikroprozessorkerns FHOP in einem Thermologger ASIC T. Klumpp, D. Jansen, FH Offenburg	63
7. Grafische Eingabe von Zustandsautomaten in PSPICE G. Voland, FH Konstanz	69



## IrDA kompatible Datenübertragung

Serielle Infrarot Datenübertragung war schon seit einigen Jahren im Computermarkt verfügbar. Aber erst die Standardisierung dieser Übertragung durch die IrDA hat erreicht, daß sich unterschiedliche Computer miteinander unterhalten können.

### Wer und was ist IrDA?

IrDA ist die Abkürzung für Infrared Data Association, eine non-profit Organisation, deren Ziel es ist, Standards für die infrarote Datenübertragung zu setzen.

Datenverbindungen mit Datenraten von 2400 bit/s bis zu 4 MBit/s werden in diesem Standard beschrieben.

Typische Anwendungen sind Drucker, Telefone, Desktop - und Laptop/Notebook - PCs. Auch zum Testen und Programmieren von Haushaltsgeräten werden Infrarotverbindungen eingesetzt.

### Wie funktioniert eine IrDA kompatible Verbindung

Im einfachsten Fall nutzt man die RS232 - Schnittstelle, die in jedem Rechner eingebaut ist. Mit einem einfachen Interface wird die Pulslänge des Bits auf 3/16 der Originallänge verkürzt um Strom zu sparen. Beim Empfang wird dann die ursprüngliche Pulslänge durch eine entsprechende Schaltung wiederhergestellt.

Diese Art der Übertragung ist durch die Geschwindigkeit der RS232 - Schnittstelle auf (heute) 115.2 kB/s begrenzt. Im IrDA - Sprachgebrauch wird diese Art der Übertragung „SIR“ (historisch für Serial InfraRed) genannt. Die minimale (und bei manchen Anwendungen einzige) nach dem Standard zu unterstützende Geschwindigkeit ist 9600 bit/s. Jede Übertragung wird aus Kompatibilitätsgründen mit dieser Geschwindigkeit gestartet.

Höhere Geschwindigkeiten erfordern andere Interfaceschaltungen. Beispielsweise bei 1.152 MBit/s (MIR=Medium IR) wird ein dem RS232 ähnliches Übertragungsverfahren verwendet, bei 4 MBit/s (FIR = Fast IR) dagegen hat man eine Pulspositionsmodulation zur Kodierung gewählt (4ppm).

Die optische Sendeleistung und die Empfängerempfindlichkeit sind so gewählt, daß eine Übertragung in einem Winkel von 30° über minimal ein Meter garantiert ist. Eine entsprechende Ausrichtung der Geräte zueinander ist somit nötig, jedoch kein extremes Ziel. Die Empfänger werden relativ unempfindlich ausgelegt um einen entsprechenden Störabstand zu Umweltstörungen (Energiesparlampen z.B.) zu haben.

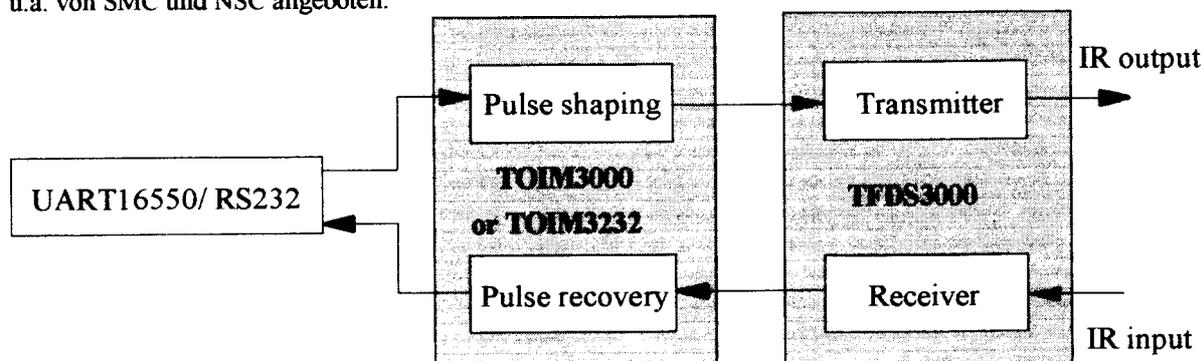
Für SIR ist die minimale Strahlstärke mit 40mW/sr festgelegt, bei MIR und FIR braucht man 100 mW/sr.

Die entsprechenden Empfängerempfindlichkeiten sind dann 40mW/m<sup>2</sup> bzw. 100mW/m<sup>2</sup> um eine Reichweite von 1 Meter zu garantieren.

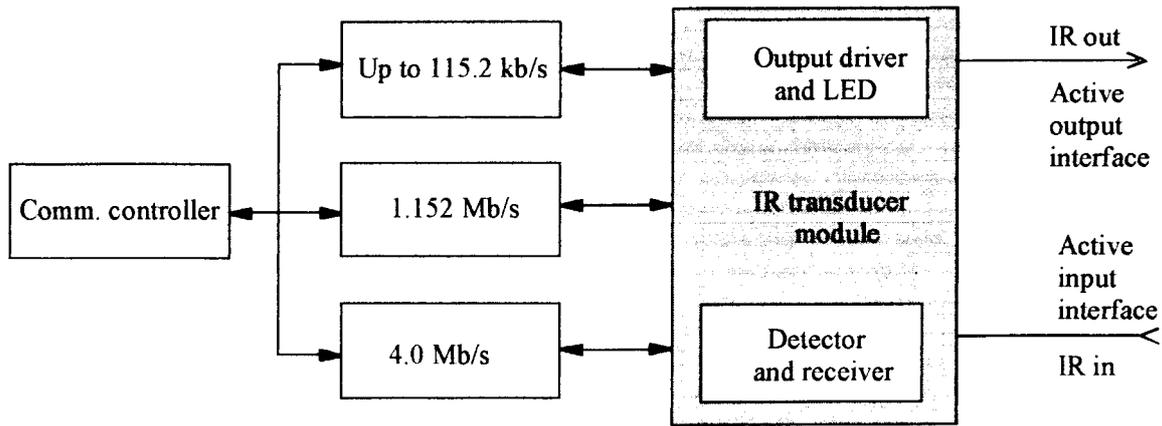
### Was braucht man, um eine IrDA kompatible Datenübertragung zu realisieren?

Der einfachste Weg ist, den RS232-Port zu verwenden (Figur 1). Dazu benötigt man einen zusätzlichen Interface-Baustein, der von verschiedenen Herstellern (im Beispiel der Figur 1: TEMIC) angeboten wird oder schon im UART eingebaut ist (z.B. erhältlich von NSC oder SMC).

Interfaceschaltungen für alle Frequenzen (Schaltung entsprechend Figur 2) von SIR bis FIR werden ebenfalls u.a. von SMC und NSC angeboten.



Figur 1. Blockdiagramm einer IR - Sende/Empfangseinheit an der seriellen Schnittstelle



Figur 2. Blockdiagramm einer IR - Sende/Empfangseinheit mit einem additiven Kommunikationscontroller

### Der IrDA Standard

#### Spezifikation

Im SIR Modus werden die Daten durch einen 1.6µs oder 3/16 der RS232-Bitlänge langen Puls repräsentiert (Figur 3). [1]

Bei den höheren Datenraten 0.576 MB/s und 1.152 MB/s verwendet man Pulse von 1/4 der Bitlänge (Figur 4).

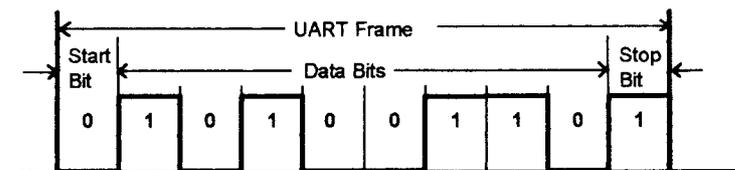
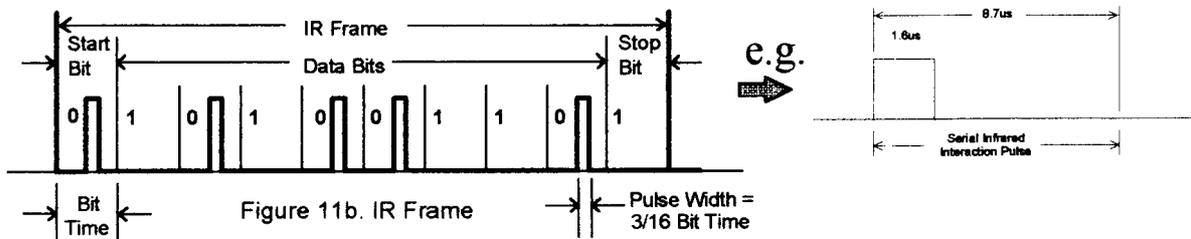
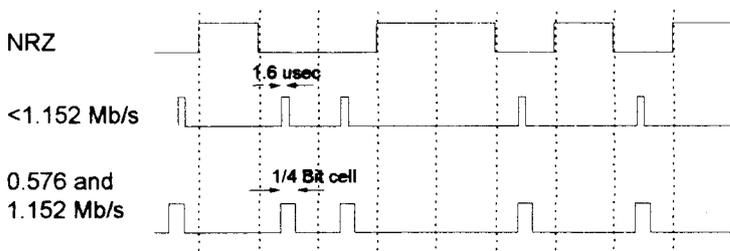


Figure 11a. UART Frame



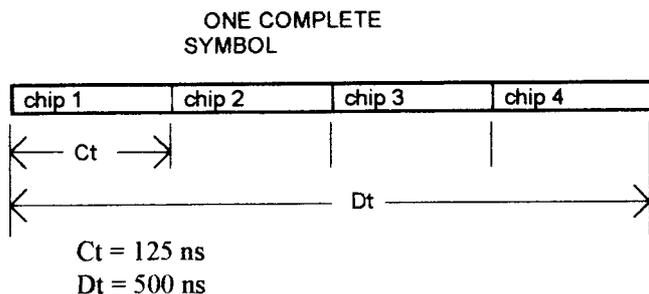
Figur 3 Kodierung eines „NRZ“ - Signals der seriellen Schnittstelle zu einer „Return to Zero Inverted“ - Kodierung für die IR-Übertragung (Figure 11a,b des Standards)



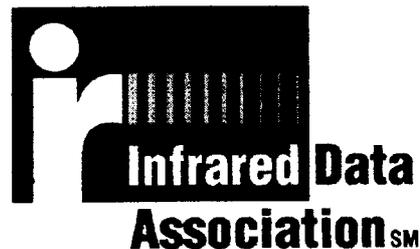
Figur 4 Bei den beiden Frequenzen 0.576 MB/s und 1.152 MB/s wird „0“ durch einen Puls der Länge eines 1/4 Bit repräsentiert

Eine nominelle Pulslänge von 125 ns wird bei einer Übertragungsrate von 4 MB/s verwendet (Figur 5). Hier wird die Information in der zeitlichen Lage des Pulses versteckt. Vier verschiedene Pulspositionen sind

definiert, die Kodierung ist ebenfalls in Figur 5 dargestellt. Die genauen Pulslängen für die verschiedenen Datenraten sind in der Tabelle 1 zusammengestellt.



Data Bit Pair (DBP)	4PPM Data Symbol (DD)
00	1000
01	0100
10	0010
11	0001



Figur 5 4 ppm Kodierung für die Übertragungsrate von 4 MB/s

Figur 6 IrDA-Logo

Tabelle 1. Übertragungsrate und Pulsängenspezifikation

Signaling Rate	Modulation	Rate Tolerance % of Rate	Pulse Duration Minimum	Pulse Duration Nominal	Pulse Duration Maximum
2.4 kb/s	RZI*	±0.87	1.41 µs	78.13 µs	88.55 µs
9.6 kb/s	RZI*	±0.87	1.41 µs	19.53 µs	22.13 µs
19.2 kb/s	RZI*	±0.87	1.41 µs	9.77 µs	11.07 µs
38.4 kb/s	RZI*	±0.87	1.41 µs	4.88 µs	5.96 µs
57.6 kb/s	RZI*	±0.87	1.41 µs	3.26 µs	4.34 µs
115.2 kb/s	RZI*	±0.87	1.41 µs	1.63 µs	2.23 µs
0.576 Mb/s	RZI*	±0.1	295.2 ns	434.0 ns	520.8 ns
1.152 Mb/s	RZI*	±0.1	147.6 ns	217.0 ns	260.4 ns
4.0 Mb/s					
Single pulse	4 PPM	±0.01	115.0 ns	125.0 ns	135.0 ns
Double pulse	4 PPM	±0.01	240.0 ns	250.0 ns	260.0 ns

\* RZI = Return to Zero Inverted

Die Anforderungen an Strahlstärke und Empfindlichkeit wurden schon weiter oben angesprochen. Hinzuzufügen ist noch, daß für die Übertragungswellenlänge ein Bereich von 850nm bis 900nm festgelegt wurde. In den Tabellen 2 und 3 sind die Ausgangs- und Eingangsdaten zusammengestellt.

**Tabelle 2. Spezifikation des IR - Ausgangs**

Specification	Data Rates	Minimum	Maximum
Peak wavelength, $\lambda_p$ , $\mu\text{m}$	All	0.85	0.90
Maximum intensity in angular range, mW/sr	All	-	500
Minimum intensity in angular range, mW/sr	115.2 kb/s and below	40	-
	Above 115.2 kb/s	100	-
Half-angle, degrees	All	$\pm 15$	$\pm 30$
Signaling rate (known as clock accuracy)	All	See table 1	See table 1
Rise time $t_r$ 10 - 90%, fall time $t_f$ 90 - 10%, ns	115.2 kb/s and below	-	600
	Above 115.2 kb/s	-	40
Pulse duration	All	See table 1	See table 1
Optical overshoot, %	All	-	25
Edge jitter, % of nominal bit duration	115.2 kb/s and below	-	$\pm 6.5$
Edge jitter relative to reference clock, % of nominal duration	0.576 and 1.152 Mb/s	-	$\pm 2.9$
Edge jitter, % of nominal chip duration	4.0 Mb/s	-	$\pm 4.0$

**Tabelle 3. Spezifikation des IR - Eingangs**

Specification	Data Rates	Minimum	Maximum
Maximum irradiance in angular range, W/m <sup>2</sup>	All	-	5000
Minimum irradiance in angular range, mW/m <sup>2</sup>	115.2 kb/s and below	40	-
	Above 115.2 kb/s	100	-
Half-angle, degrees	All	$\pm 15$	-
Receiver latency allowance, ms		-	10

Mit den minimalen Anforderungen für die Reichweite von 1 m und dem minimalen Winkel sind die unteren Grenzen des Standards festgelegt. Um eine gegenseitige Störung zwischen gleichzeitig in einem Raum betriebenen Systemen zu verhindern, wurden Obergrenzen für Strahlstärke (500 mW/sr), Empfindlichkeit (4 mW/m<sup>2</sup>) und Winkel ( $\pm 30^\circ$ ) festgelegt.

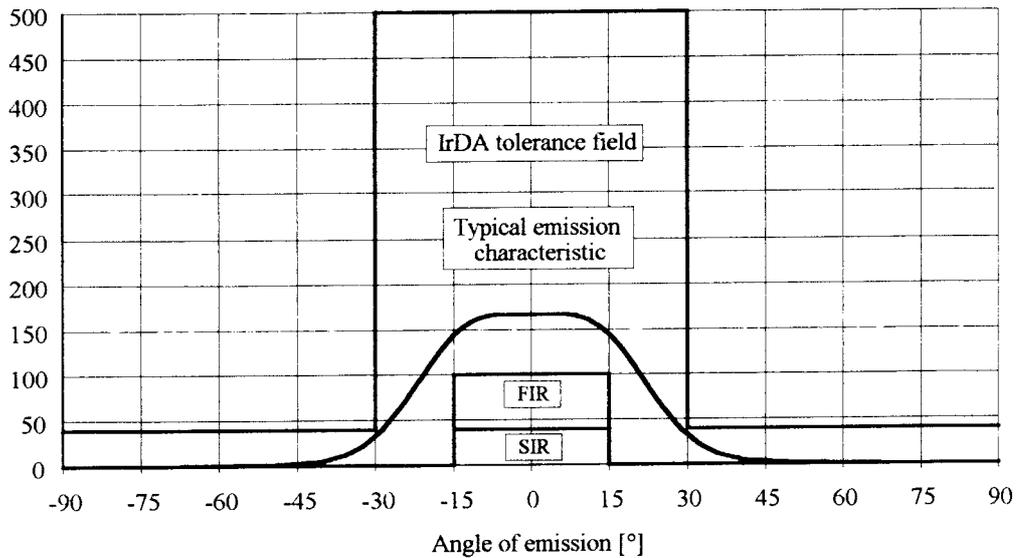
Das Toleranzfeld der winkelabhängigen Emission wird in Figur 6 dargestellt.

Geräte, die dem IrDA Standard entsprechen (die Meßverfahren zur Qualifikation sind in einer weiteren Spezifikation zusammengefaßt [2]), dürfen das IrDA - Logo tragen (Figur 6).

### Zukünftige Entwicklungen

„Schneller, weiter, höher“ kann man erwarten. Die Technologien sind für höhere Frequenzen verfügbar, auch größere Entfernungen wie z.B. bei Fernsehfernsteuerungen sind möglich. Bei der Festlegung weiterer Standards - insbesondere wenn man an diffuse Abstrahlung über ganze Räume denkt - ist Rücksicht zu nehmen auf bestehende Standards und Anwendungen. Hier ist zu erwarten, daß sich die IrDA mit IEC und CENELEC und EIAJ arrangieren muß, um einen weltweiten Standard zu generieren, der auch den Anforderungen der MultiMedia - Anwendungen gerecht wird.

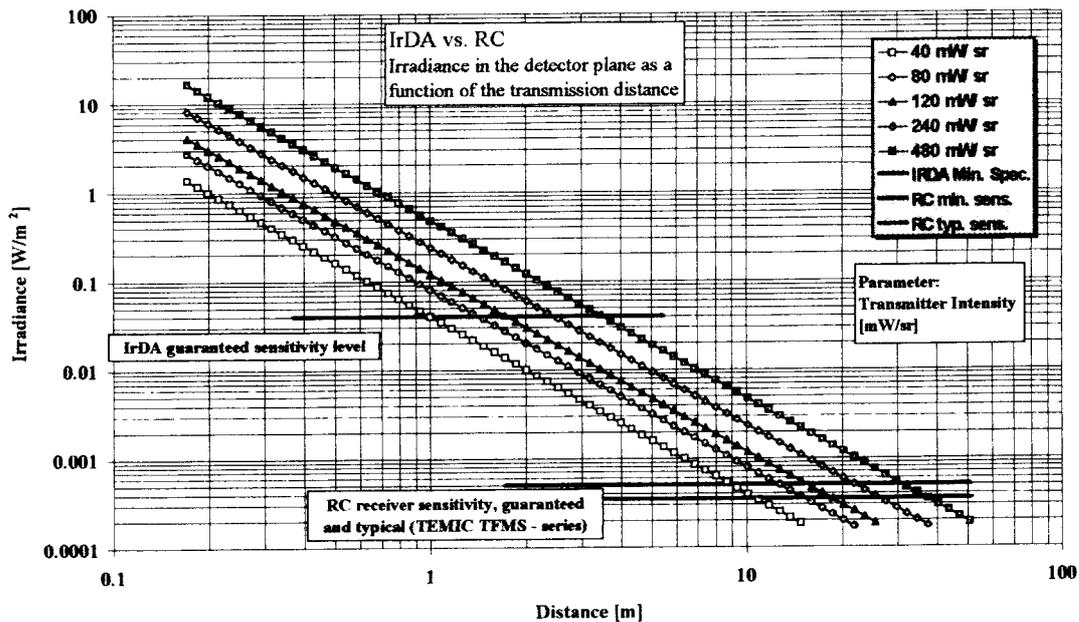
Radiant intensity [mW/sr]



Figur 6 Toleranzfeld der winkelabhängigen Emission

**Übertragungsentfernung**

In Figur 7 ist die Übertragungstrecke bei gegebener Strahlstärke und Empfindlichkeit abzulesen.



Figur 7. IrDA, Übertragungsentfernung

**Literatur**

Die Figuren 3-5 und die Tabellen 1-3 sind aus [1] übernommen

[1] Infrared Data Association, Serial Infrared Physical Layer Link Specification Version 1.1e, October 17, 1995

[2] Infrared Data Association, Serial Infrared Physical Layer Test Specification Proposal Version 0.3, October 11, 1995





# Die IMS GATE FOREST<sup>®</sup> 0.8 $\mu\text{m}$ CMOS Gate Array Familie

Jürgen Gandowitz

## Übersicht

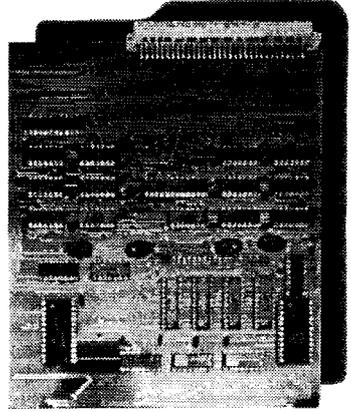
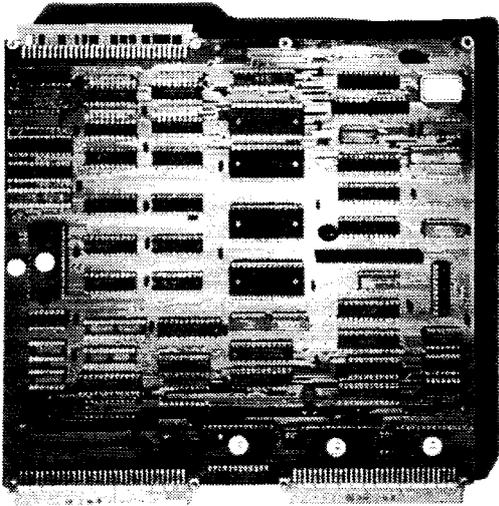
- Einleitung
- Technologie
- Masterfamilie
- Bibliothek
- Design Kits und Schnittstellen zum IMS
- Beispiele
- Zusammenfassung

## Das IMS Stuttgart

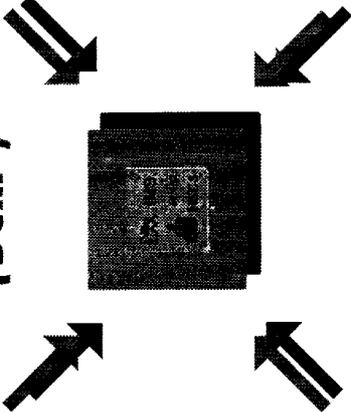
- Förderung und Unterstützung des Mikroelektronik-Einsatzes bei KMUs
- Entwurf und Herstellung mikroelektronischer Schaltungen und Systeme
- 0.8  $\mu\text{m}$  CMOS Technologie, 700  $\text{m}^2$  Reinraum
- Elektronenstrahl Direktschreiben
- Ausbildung, Seminare und Schulungen
- Forschung und Entwicklung



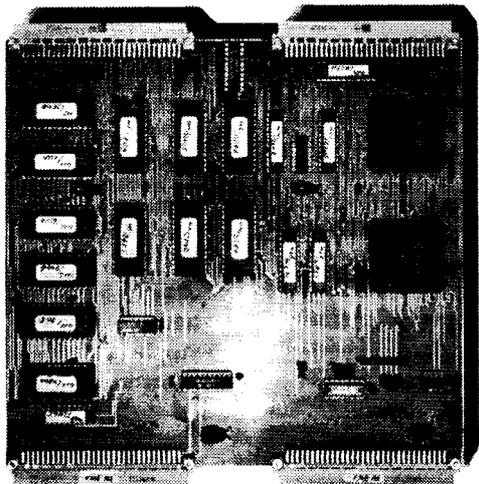
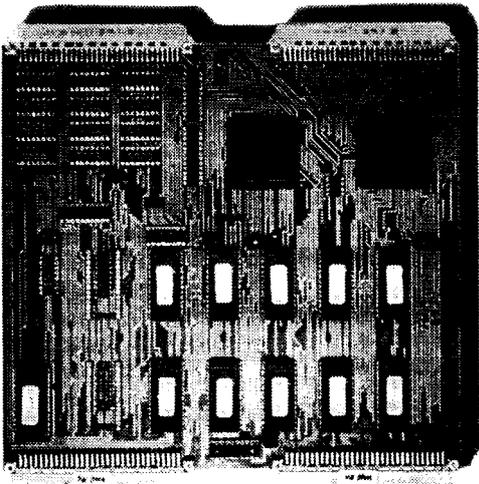
# GATE FOREST ASIC ersetzt Platinen



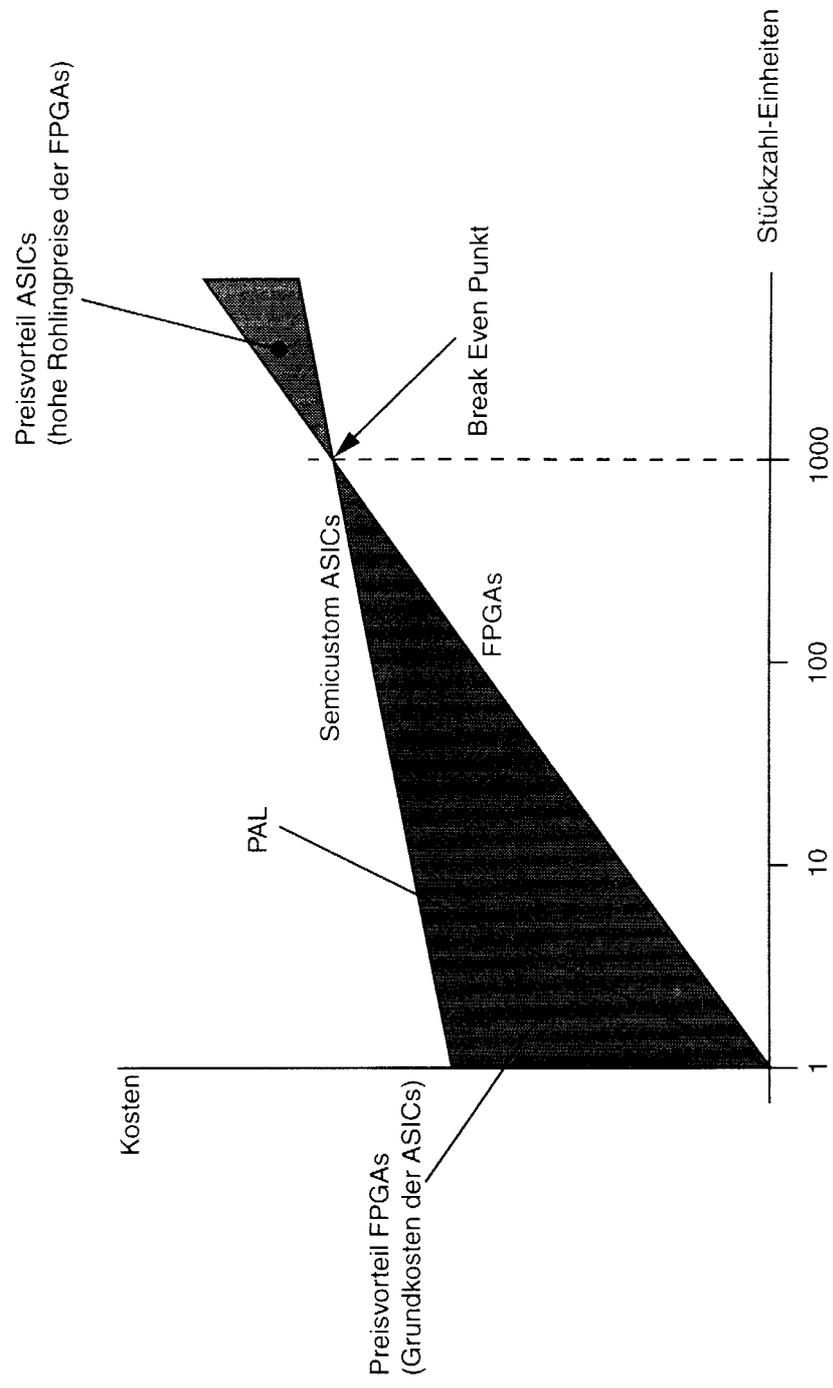
**1 GATE FOREST-ASIC  
(5 cm<sup>2</sup>)**



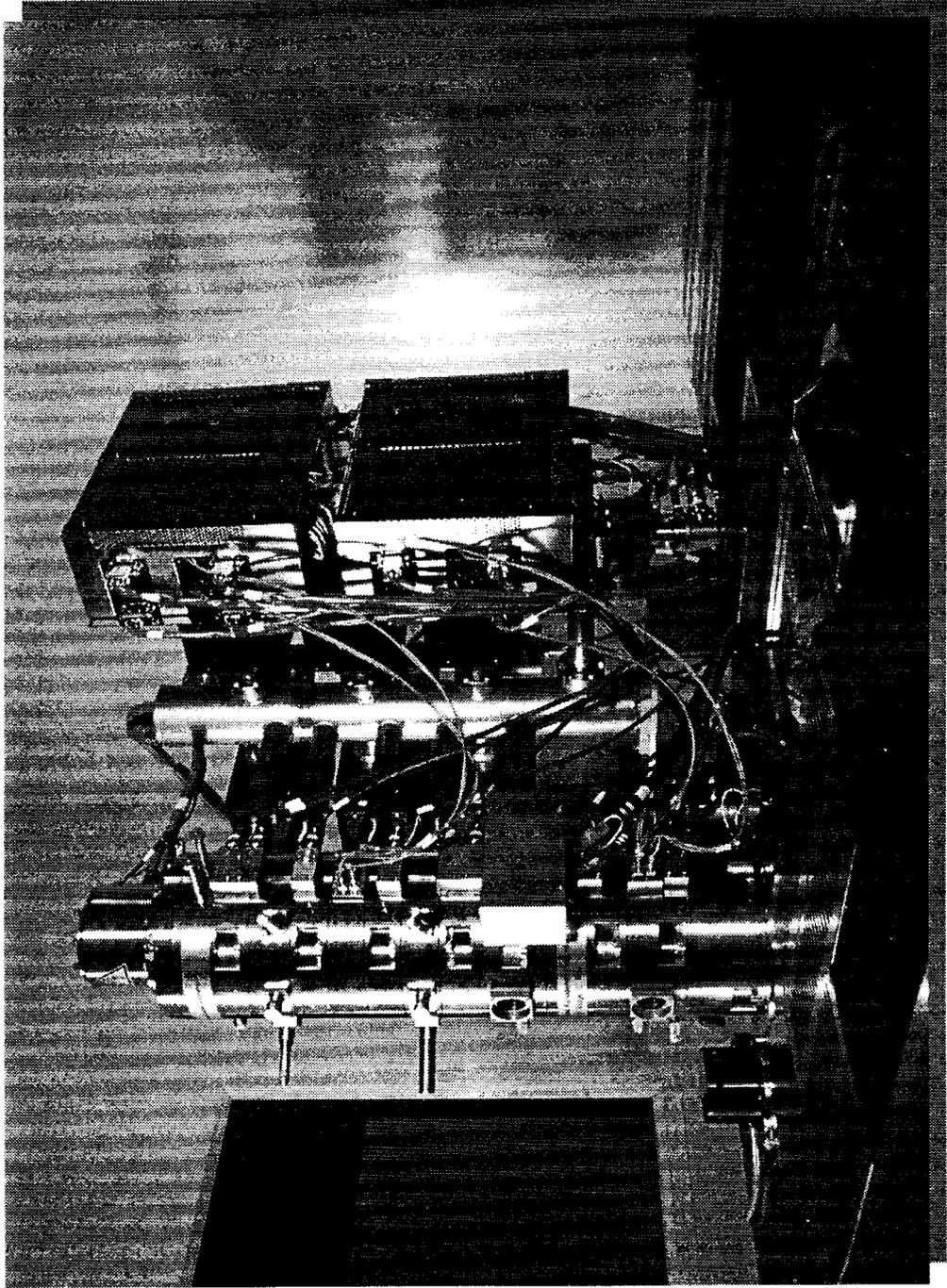
**ersetzt 4 Platinen  
(500 cm<sup>2</sup>)**



# Kostenvergleich FPGA - ASIC



**High-Speed Electron-Beam Writer HL-700D**





## Das Ziel

„Kostengünstige Prototypen und Serien für den Einstieg in die Mikroelektronik“

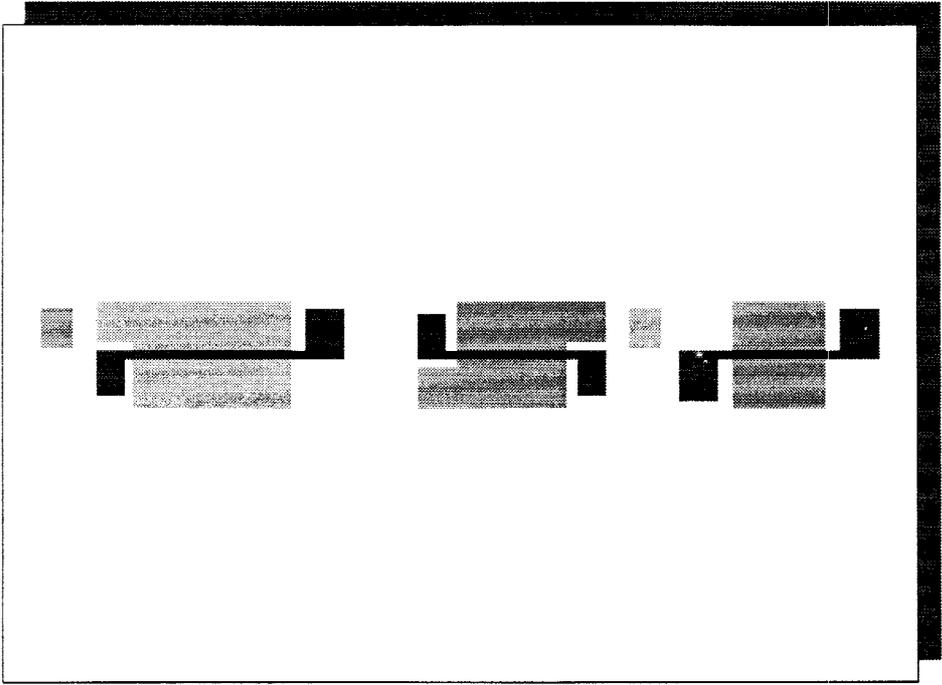
wird erreicht durch die Kombination von

- Semi-Custom Sea-of-Gates Architektur
- CMOS Fertigungstechnologie
- Maskenlose Elektronenstrahl-Lithographie
- Multi-Projekt Fertigungsläufe herunter bis zur Losgröße 1 Wafer

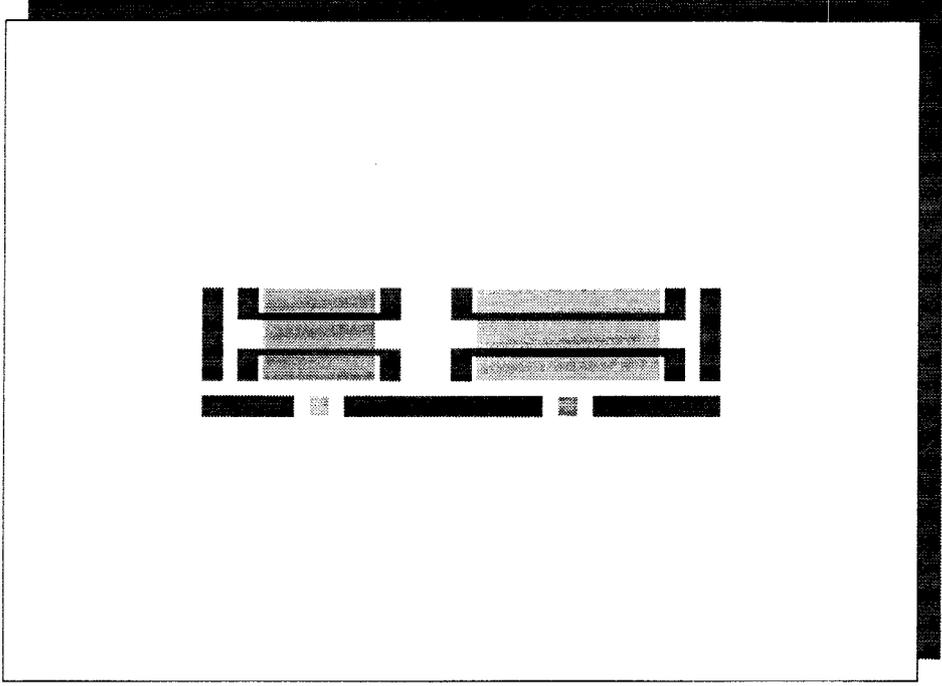
## Entwicklung des IMS GATE FOREST®

<i>Jahr</i>	<i>Technologie</i>	<i>Merkmale</i>
<b>1987</b>	3.0 µm	Erster GATE FOREST® 45° Mikroarchitektur
<b>1988</b>	2.0 µm	Technologie-Shrink aus 3 µm Techn.
<b>1991</b>	1.2 µm	Neue Mikroarchitektur E-Beam Personalisierung
<b>1995</b>	0.8 µm	Verbesserung der Mikroarchitektur Erweiterte Zellbibliothek Analoge Felder

# Gate Array Grundstrukturen



1.2  $\mu\text{m}$ : 3-Transistoren Masterzelle



0.8  $\mu\text{m}$ : 4-Transistoren Masterzelle

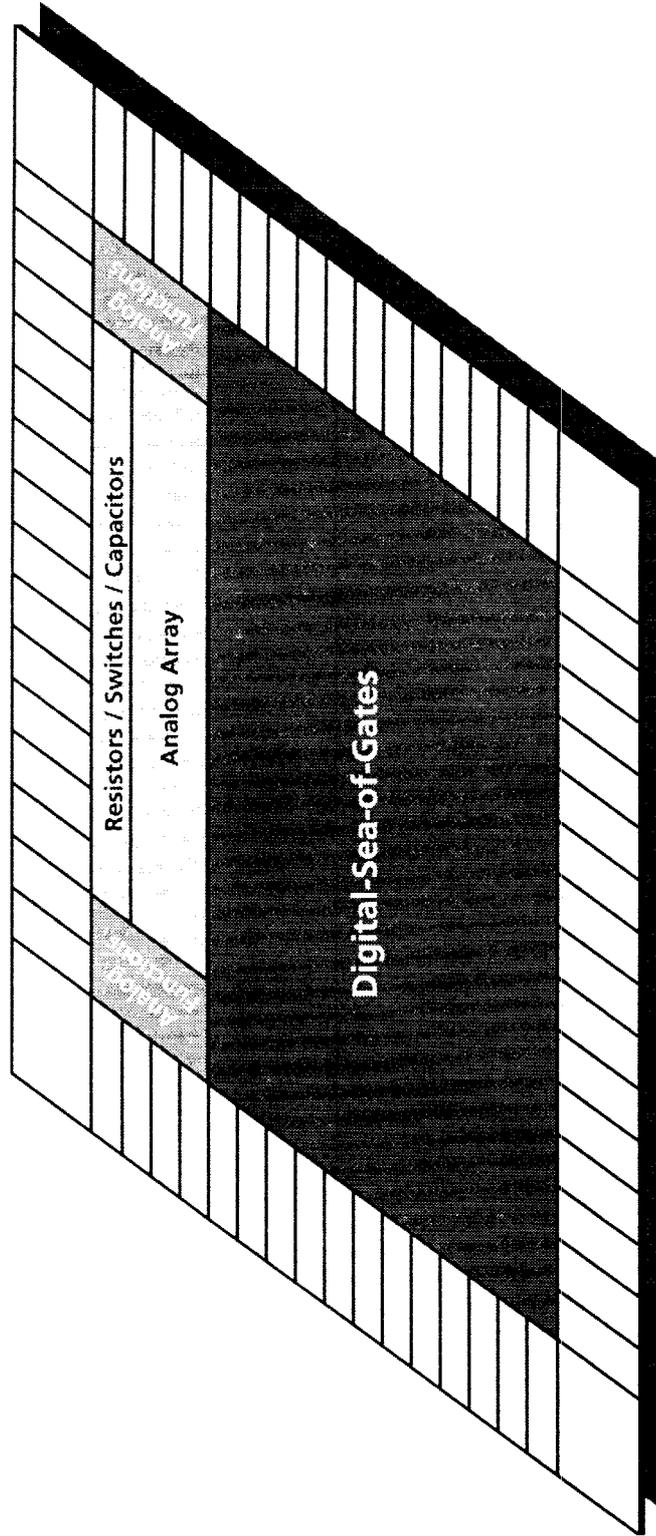


## Entwicklungsziele

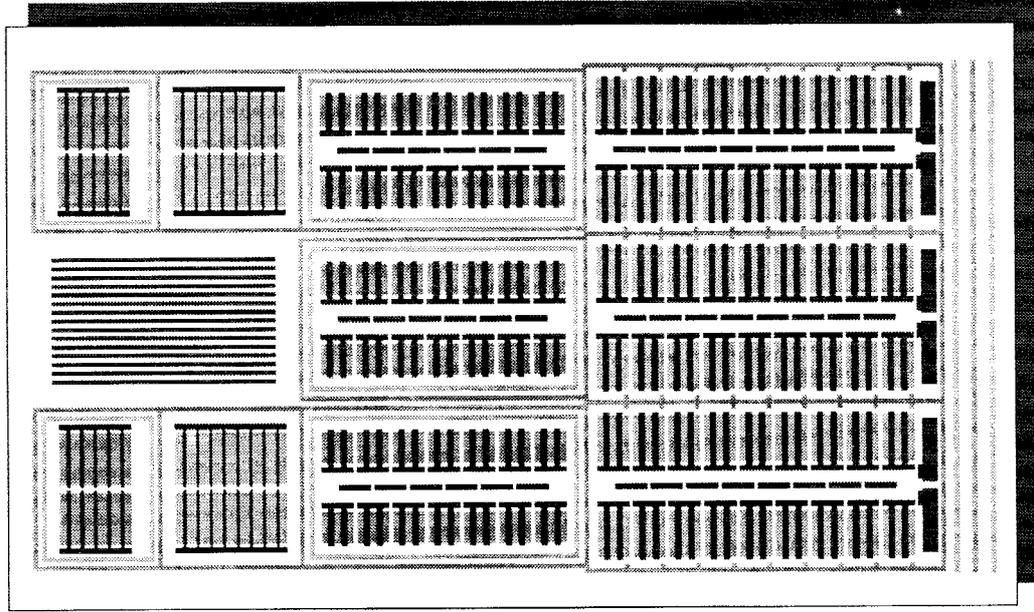
- Ausnutzung der technologischen Weiterentwicklungen
- Einbringen von Anregungen aus dem IMS Kundenkreis
- Erweiterung der Bibliotheken
- Umsetzung von Entwürfen aus der 1.2  $\mu\text{m}$  Technologie
  
- ⇒ Schwerpunkt weiterhin digitale, statische CMOS Schaltungen
  
- ⇒ Realisierung von Designs mit analogen Grundelementen
  
- ⇒ Entwurf von analogen Grundfunktionen / Bibliothekselementen

# IMS GFN120

- Digital Sea-of-Gate
- Analog Array



## Die „Analogfelder“ auf dem IMS 0.8 $\mu\text{m}$ GATE FOREST



- Polysilizium Bahnen
- NMOS Multifinger-Transistoren
- PMOS Multifinger-Transistoren
- NMOS Doppelfinger-Transistoren
- PMOS Doppelfinger-Transistoren
- PMOS Langkanal-Transistoren

## Die Analogfelder II

- Nutzung der Analogfelder vorwiegend durch erprobte Makros  
Beispiele:
    - A/D- und D/A - Umsetzer
    - Komparatoren
    - OpAmps
  - Makroentwicklung
    - IMS intern
    - für kundenspezifische Projekte
    - in Gemeinschaftsprojekten und Kooperationen
- ⇒⇒ Wie simulieren ?
- ⇒⇒ Wie testen ?

## GATE FOREST Masterfamilie

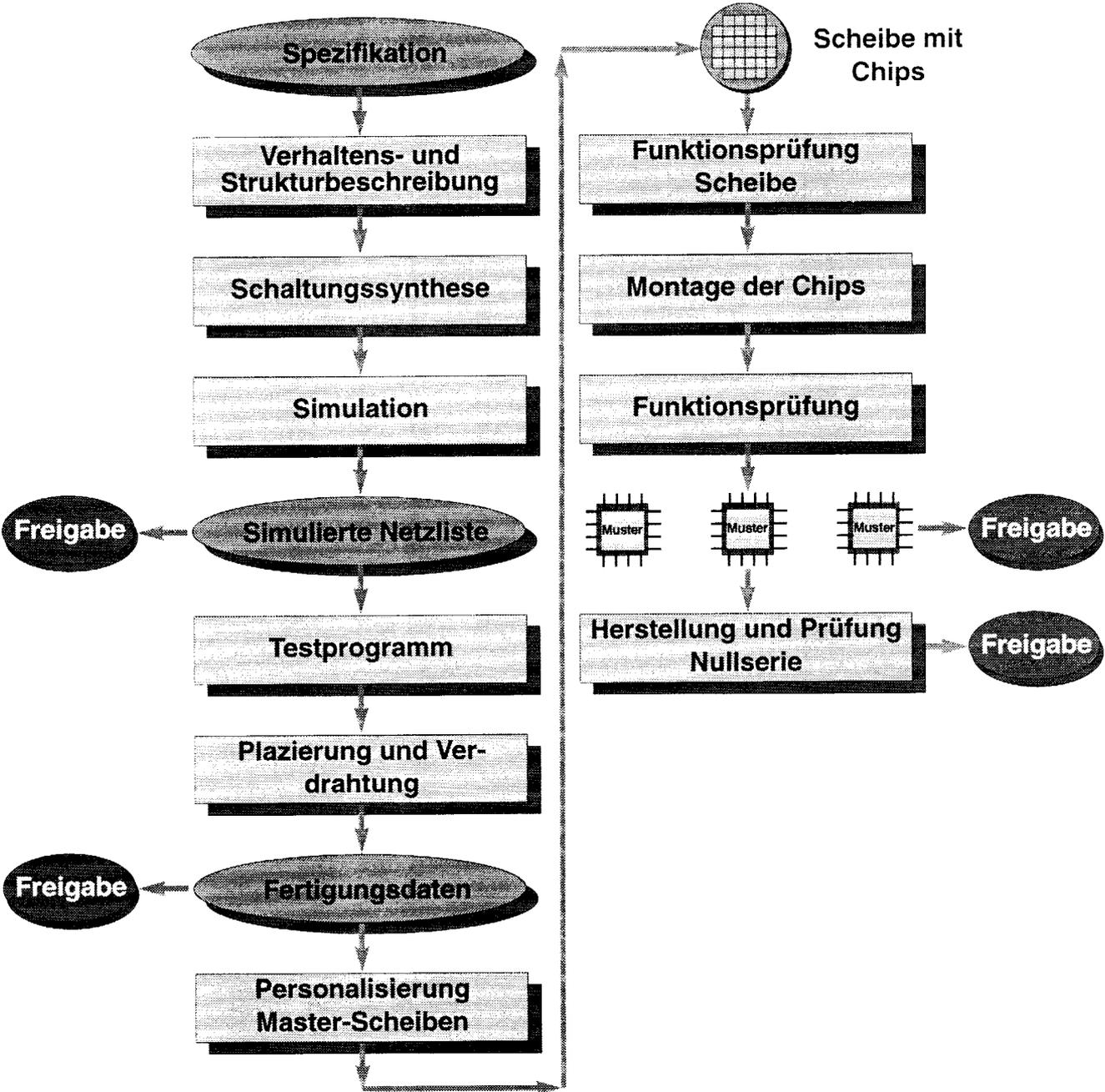
<i>Mastertyp</i>	<i>Max. nutzbare digitale Gatter</i>	<i>Analoge Tiles</i>	<i>Maximum I/O-Pins</i>
<b>GFN004</b>	2,200	8	64
<b>GFN012</b>	6,000	14	112
<b>GFN024</b>	12,000	20	148
<b>GFN036</b>	18,000	24	184
<b>GFN060</b>	30,000	31	256
<b>GFN120</b>	65,000	46	256

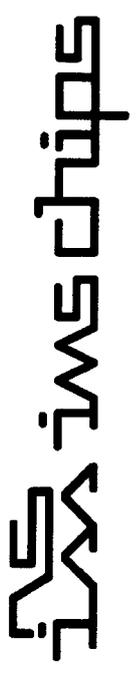
## IMS Standardzellen Bibliothek

- 1.2  $\mu\text{m}$  GATE FOREST: 119 Core-Zellen  
17 I/O Zellen
- 0.8  $\mu\text{m}$  GATE FOREST: 161 Core-Zellen  
92 I/O Zellen
- ⇒ Bibliothekserweiterungen im Core:  
Input Buffer, Complex Gates, Latches,
- ⇒ Bibliothekserweiterungen bei den I/O-Pads:  
z.B. Einführung der Klasse „Limited Slew Rate Pads“

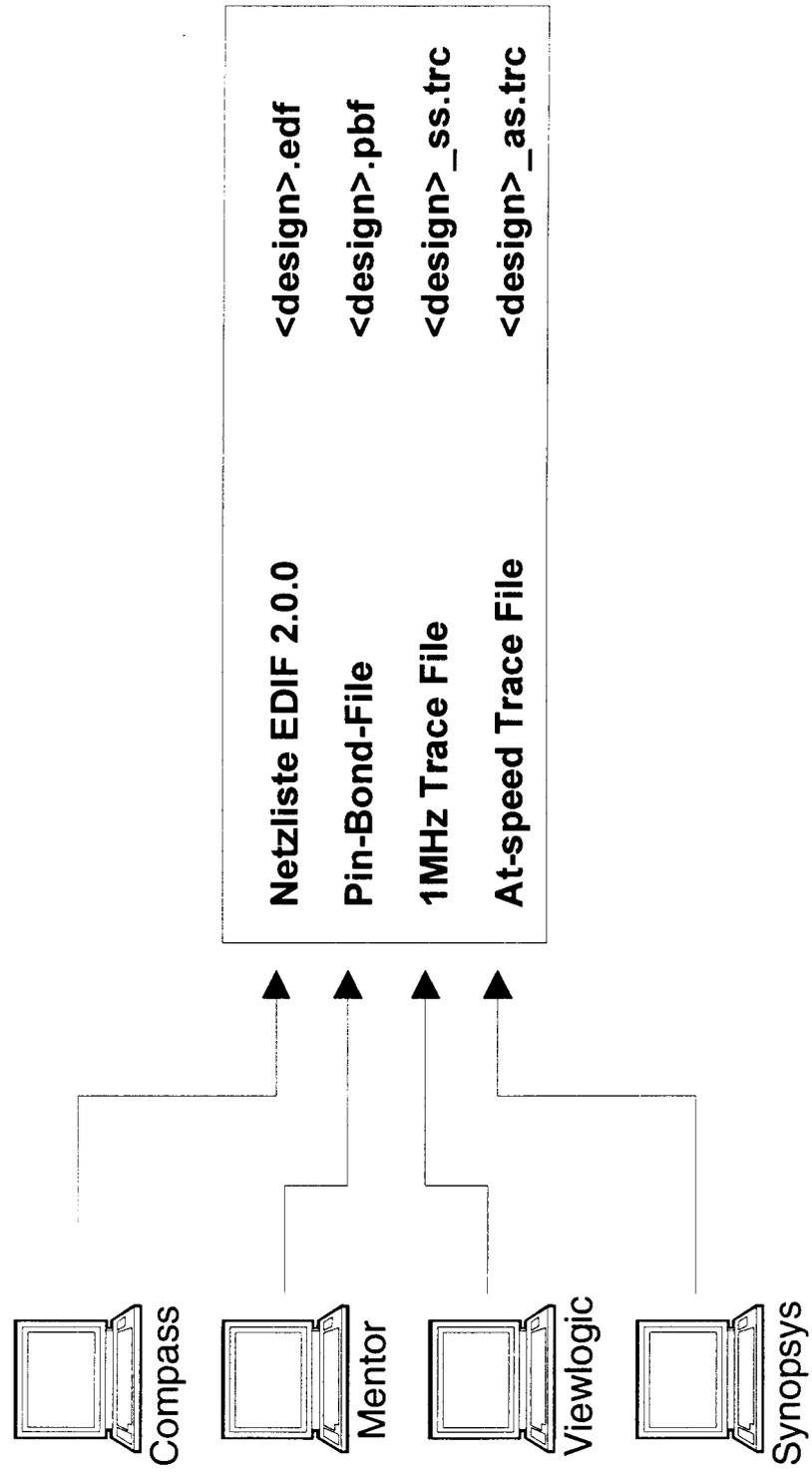
# ASICs in modernen elektronischen Geräten

## Entwicklung und Herstellung von Semicustom-ASICs

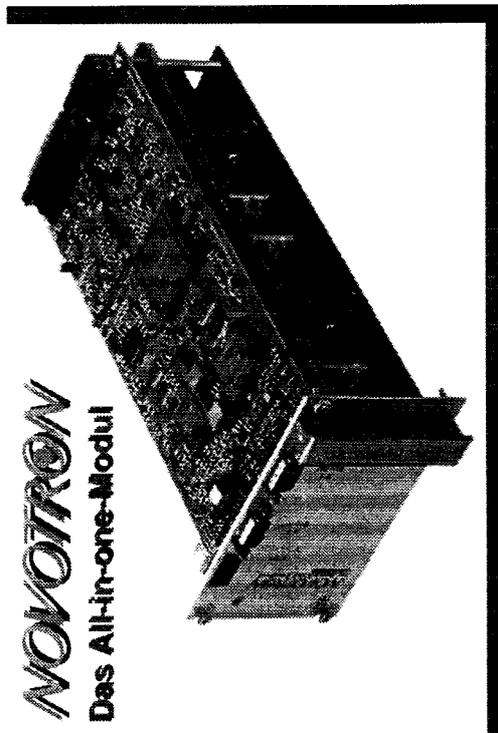




# IMS Design Kits und die Kundenschnittstelle „Simulierte Netzliste“



## **NOVOTRON** Das All-in-one-Modul



### **Novodrive ND 21**

5-kW-Pulsumrichter und Frequenzrichter in einem – für  
bürstenlose Servomotoren und Asynchronmotoren.  
Der kleinste 5-kW-Pulsumrichter der Welt – 229 mm  
lang, 100 mm breit, 65 mm hoch und 1100 g leicht.  
Möglich wurde das Platzwunder durch den ASIC-  
NOVOCHIP, den NOVOTRON zusammen mit dem  
IMS entwickelt hat.

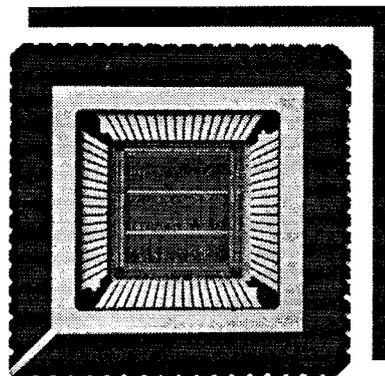
### **ND 2203**

Der Chip dient zur Steuerung von dreiphasigen  
Elektromotoren und besteht grundsätzlich aus  
zwei Hauptteilen: einem Rechenwerk und einem  
Pulsmodulator.

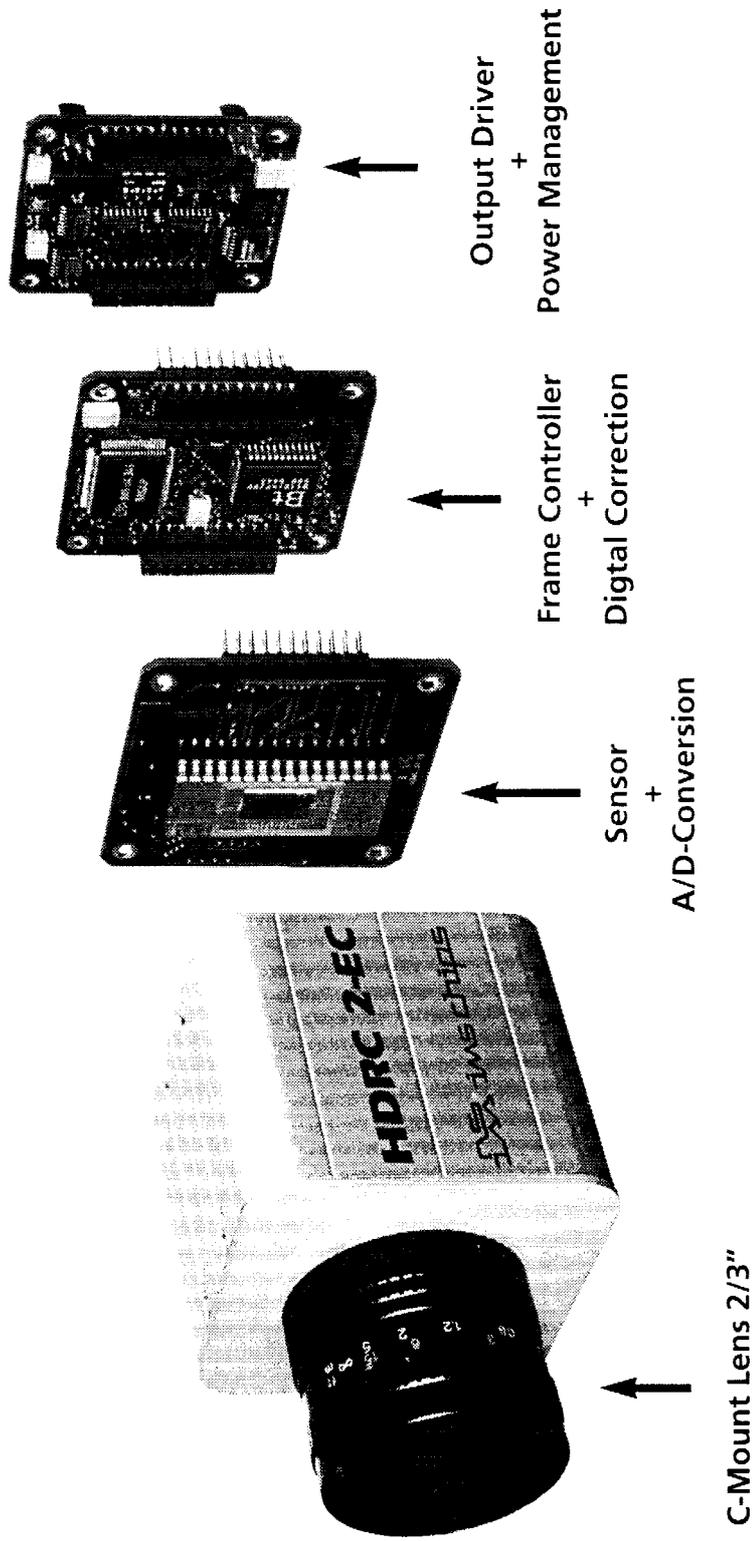
Das Rechenwerk ist ein geschwindigkeitsopti-  
mierter Spezialprozessor, der Additionen und  
Multiplikationen durchführt. Das Programm ist in  
einem 20x6 Bit ROM gespeichert. Der Pulsmodu-  
lator generiert die breitenmodulierte Steuerpulse  
für den Motor.

### **Chipdaten**

Aktive Transit.: 11 600  
Gehäuse: PLCC 68  
Grundtakt: 10 MHz  
Pulsfrequenz: 10–20 kHz



# HDRC Experimental Camera



## Zusammenfassung

- Der 0.8  $\mu\text{m}$  GATE FOREST<sup>®</sup> ermöglicht kostengünstige Lösungen für vorwiegend digitale Schaltungen.
- Die Realisierung von  
RAM / ROM  
Analogen Grundschaltungen  
macht innovative Single-Chip Lösungen möglich.
- Das IMS bietet Kundenschnittstellen von der Spezifikation bis zur fertigen Netzliste an.
- Der Stückzahlbereich von wenigen Prototypen bis zu größeren Serien ist abgedeckt.
- Beratung, Walk-in Service und IMS Designkits für Mikroelektronik-Einsteiger und erfahrene Chipdesigner.



# 10Bit $\Sigma\Delta$ -DA-Wandler mit Altera PLD

H. Töpfer / FHTE Esslingen-Standort Göppingen  
 Vortrag zum MPC-Workshop am 5.7.1996 im Heilbronn

Sigma-Delta-Wandler werden häufig als Analog-Digital- oder als Digital-Analog-Wandler in der Sprachsignalverarbeitung eingesetzt. Aber auch als hochgenaue Wandler für die Meßtechnik finden  $\Sigma\Delta$ -Wandler zunehmend Verbreitung.

Bild 1 zeigt das Modell eines  $\Sigma\Delta$ -Wandlers 1. Ordnung. Der Wandler kann sowohl als AD-, als auch als DA-Wandler ausgeführt werden, ich beschränke mich im Folgendem aber auf DA-Wandler.

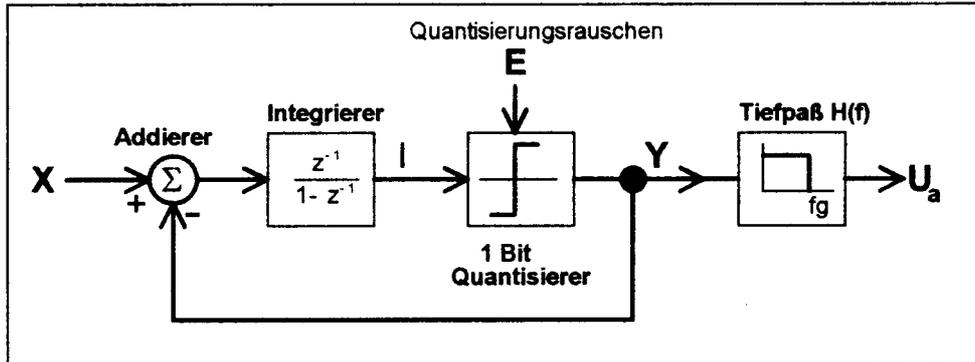


Bild 1: Blockschaltbild eines DA-Wandlers 1. Ordnung

Bei dem DA-Wandler im Bild 1 wird an den Eingang X ein Digitalwort angelegt. Addierer, Integrierer und Quantisierer sind digital. Am Ausgang Y des Quantisierers entsteht ein digitaler 1 Bit Datenstrom, der als MSB-Bit auf den Addierer zurückgekoppelt wird. Gleichzeitig bildet der Tiefpaß aus dem digitalen Bitstrom die analoge Ausgangsspannung  $U_a$ . Der Quantisierer ist denkbar einfach. Er leitet lediglich das Vorzeichen-Bit des Integratorausgangs weiter.

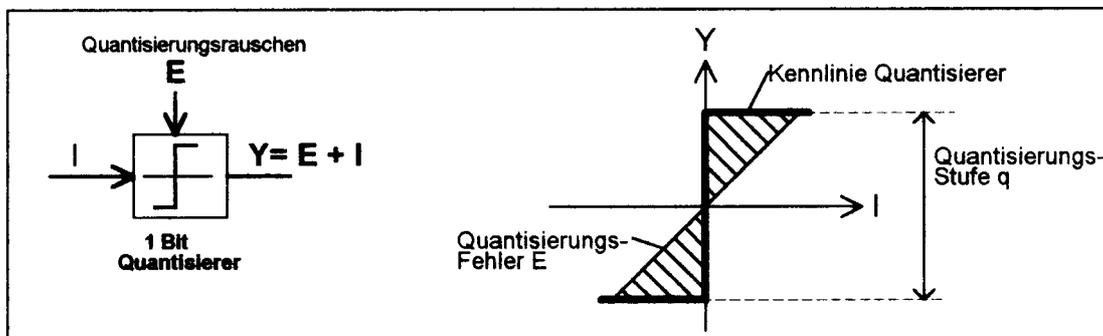


Bild 2: Quantisierer

Bild 2 zeigt den Quantisierungsfehler, der das Quantisierungsrauschen E verursacht. Für den Effektivwert des Quantisierungsfehlers gilt näherungsweise:

$$e_{\text{eff}} = \frac{q}{\sqrt{12}} \quad q \text{ Quantisierungsstufe} \quad (1)$$

Die Rauschleistungsdichte des Quantisierungsfehler ist im Frequenzbereich 0 -  $f_s/2$  ( $f_s$  Samplingfrequenz) gleich verteilt. (siehe Bild 2)

Unter dieser Annahme gilt für Rauschleistungsdichte des Quantisierungsrauschens:

$$E^2(f) = \frac{q^2}{6f_s} \quad 0 < f < f_s/2 \quad f_s \text{ Samplingfrequenz} \quad (2)$$

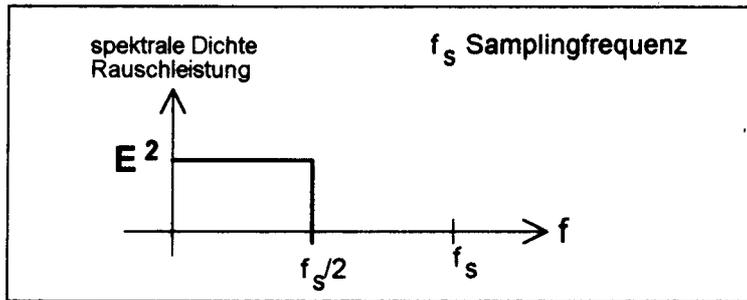


Bild 3: Spektrale Verteilung des Quantisierungsrauschens

Der Sigma-Delta-Wandler 1.Ordnung nach Bild 1 hat folgende Übertragungsfunktion:

$$Y(z) = X(z) \cdot z^{-1} + E(z) \cdot (1 - z^{-1}) \quad (3)$$

Den Term  $(1-z^{-1})$  bezeichnet man auch als Noise-Transfer-Function (NTF), weil er die Übertragung des Quantisierungsrauschens zum Ausgang Y beschreibt. Für den Betrag der NTF gilt:

$$|NTF| = |1 - z^{-1}| = 2 \cdot \sin \frac{\pi \cdot f}{f_s} \approx 2\pi \cdot \frac{f}{f_s} \quad \text{für } f \ll f_s \quad (4)$$

Da die Rauschleistungsdichte des Quantisierungsrauschens konstant ist, gilt für die Frequenzabhängigkeit des Rauschsignals am Ausgang Y:

$$N(f) = E(f) \cdot |NTF| = E(f) \cdot 2 \sin \left( \pi \cdot \frac{f}{f_s} \right) \approx \frac{q}{\sqrt{6f_s}} \cdot \left( 2\pi \cdot \frac{f}{f_s} \right) \quad \text{für } f \ll f_s \quad (5)$$

N spektrale Dichte der Rauschspannung am Ausgang Y

Der Effektivwert der Rauschspannung hinter dem Tiefpaßfilter beträgt:  
(das Tiefpaßfilter sei ein idealer Tiefpaß mit  $H(f)=1$  für  $f < f_g$  und  $H(f)=0$  für  $f > f_g$ )

$$n_{\text{eff}} = \sqrt{\int_0^{f_g} H^2(f) \cdot N^2(f) \cdot df} \approx \frac{q}{6} \cdot \pi \cdot \left( \frac{2f_g}{f_s} \right)^{\frac{3}{2}} \quad (6)$$

Dieses Verhalten nennt man „Noise Shaping“. Das niederfrequente Quantisierungsrauschen wird abgeschwächt und zu höheren Frequenzen hin verschoben.

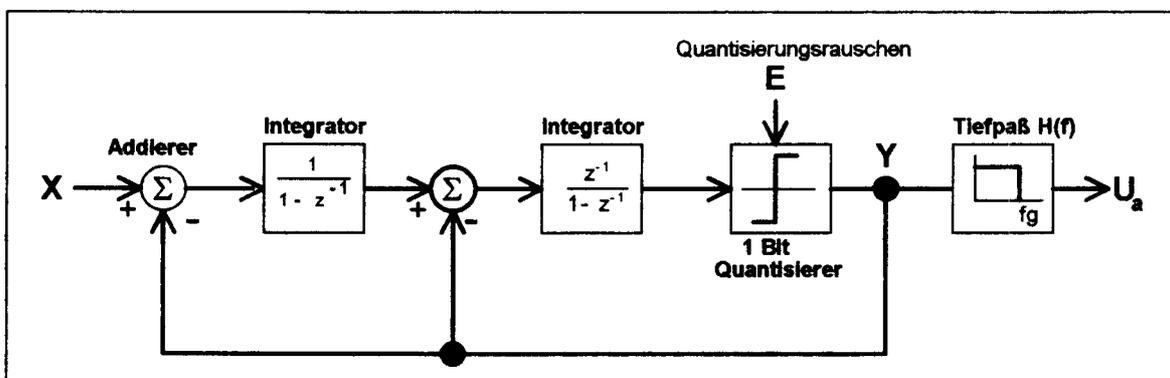


Bild 4: Sigma-Delta-Wandler 2.Ordnung

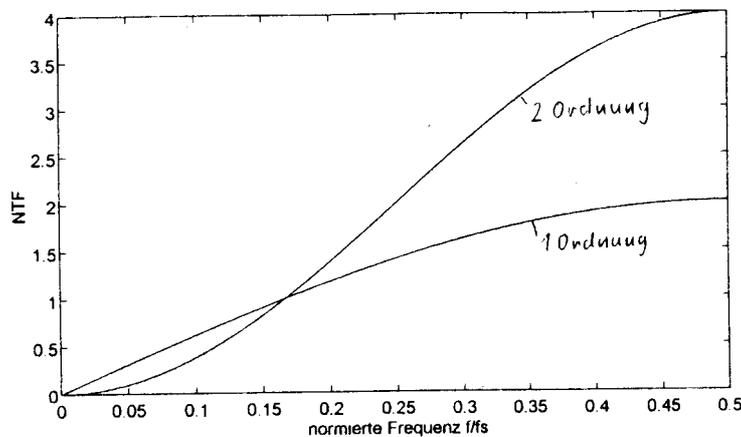
Bild 4 zeigt einen typischen Sigma-Delta-Wandler 2.Ordnung. Der Wandler hat die Übertragungsfunktion:

$$Y(z) = X(z) \cdot z^{-1} + E(z) \cdot (1 - z^{-1})^2 \quad (7)$$

Der Betrag der „Noise Transfer Funktion“ eines Sigma-Delta-Wandler 2.Ordnung ist also:

$$|NTF| = |(1 - z^{-1})^2| = 4 \cdot \left( \sin \frac{\pi \cdot f}{f_g} \right)^2 \quad (8)$$

Bild 5 zeigt den Betrag der NTF für einen Wandler 1.- und 2. Ordnung.



**Bild 5: Betrag der „Noise Transfer Funktion“ des  $\Sigma\Delta$ -Wandlers 1.- und 2. Ordnung (GL 4 und 8)**

Der Wandler 2.Ordnung schwächt das Rauschen bei niedrigen Frequenzen wesentlich besser ab, als der Wandler 1.Ordnung. Allerdings erzeugt der Wandler 2.Ordnung auch bei hohen Frequenzen ein stärkeres Rauschen, als der Wandler 1.Ordnung. Dieses Rauschen muß durch einen entsprechend leistungsfähigen Tiefpaß herausgefiltert werden.

Die spektrale Dichte der Rauschspannung N am Ausgang Y des Wandlers 2.Ordnung wird dann:

$$N(f) = E(f) \cdot 4 \cdot \left[ \sin \left( \pi \cdot \frac{f}{f_s} \right) \right]^2 \approx \frac{q}{\sqrt{6f_s}} \cdot \left( 2\pi \cdot \frac{f}{f_s} \right)^2 \quad (9)$$

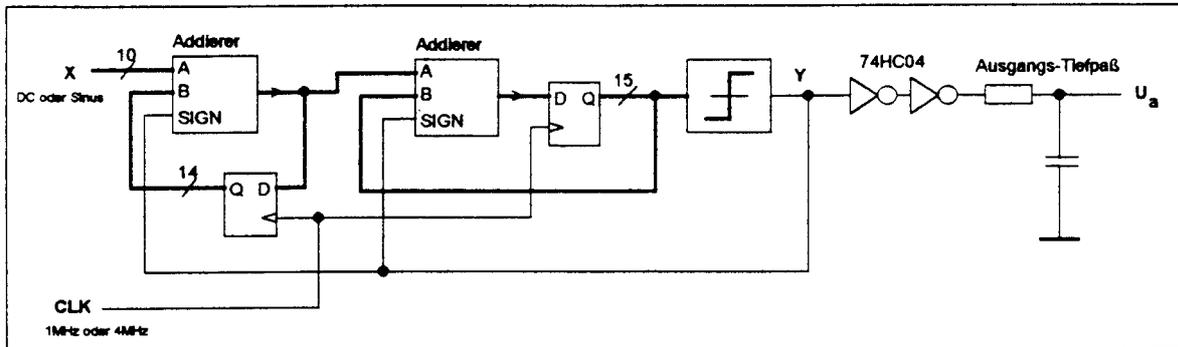
Der Effektivwert der Rauschspannung hinter dem Tiefpaßfilter ergibt sich daraus näherungsweise zu:

$$n_{\text{eff}} \approx \frac{q \cdot \pi}{\sqrt{60}} \cdot \left( \frac{2f_g}{f_s} \right)^{\frac{5}{2}} \quad \text{für } f_g \ll f_s \quad (10)$$

Um das Verhalten dieser Wandlers praktisch zu studieren, wurde ein 10 Bit  $\Sigma\Delta$ -Wandler mit einem Altera PLD vom Typ EPM 7128 aufgebaut. Bild 5 zeigt ein vereinfachtes Blockschaltbild des Wandlers.

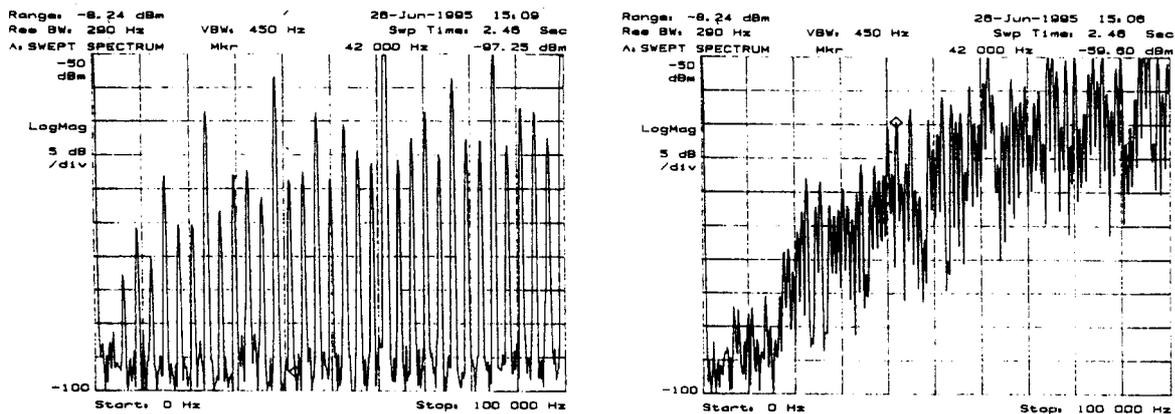
Die Ordnung des  $\Sigma\Delta$ -Wandlers kann durch einen Schalter zwischen 1 und 2 umgeschaltet werden.

Die Clockfrequenz ist zwischen 1MHz und 4 MHz umschaltbar. Damit kann der Einfluß der Samplingfrequenz  $f_s$  auf das Quantisierungsrauschen sehr schön demonstriert werden. Der Wandler funktionierte aber auch noch bei einer Clockfrequenz von 14 MHz. Auf den Eingang des Wandler kann entweder ein mit DIP-Switches einstellbarer DC-Wert, oder ein Sinus gegeben werden, der in einem Signal-Prozessor erzeugt wird.

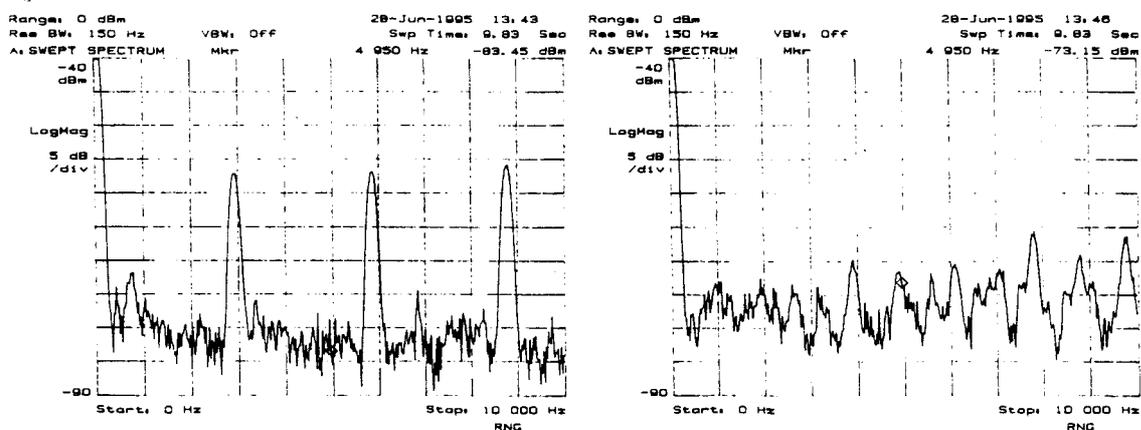


**Bild 6: Blockschaltbild des aufgebauten 10Bit Digital-Analog Wandlers 2.Ordnung**

Die Bilder 7 und 8 zeigen die gemessenen Spektren der Ausgangsspannung  $U_a$  des aufgebauten Sigma-Delta-Wandlers. Das linke Diagramm ist das Spektrum des Wandlers 1.Ordnung. Rechts ist das Spektrum des Wandlers 2.Ordnung gezeigt.



**Bild 7: Spektrum der Ausgangsspannung  $U_a$  des aufgebauten  $\Sigma\Delta$ -Wandlers: links: 1.Ordnung rechts: 2.Ordnung Samplingfrequenz  $f_s=1\text{MHz}$**



**Bild 8: wie Bild 7, aber Frequenzbereich 0-10KHz herausgezoomt**

Die Samplingfrequenz beider Wandler war 1MHz. An den Wandlereingang wurde der DC-Wert „480“ angelegt (Aussteuerungsbereich 0- 1023).

In Bild 7 ist das Spektrum von 0-100KHz dargestellt, im Bild 8 ist der Bereich 0-10 KHz herausgezoomt.

Bei dem Wandler 1.Ordnung handelt es sich nicht um ein kontinuierliches Spektrum, sondern um ein ausgeprägtes Linienspektrum. Die Frequenz der Linien ist abhängig von der Größe des am Eingang angelegten Wertes.

Der Wandler 2.Ordnung „verschmiert“ das Rauschen stärker. Einzelne Linien treten nicht mehr mit so stark hervor. Bei niedrigen Frequenzen ist das Rauschen des Wandlers 2.Ordnung deutlich geringer. Allerdings ist der Unterschied kleiner, als von den Gleichungen (6) und (10) vorhergesagt.

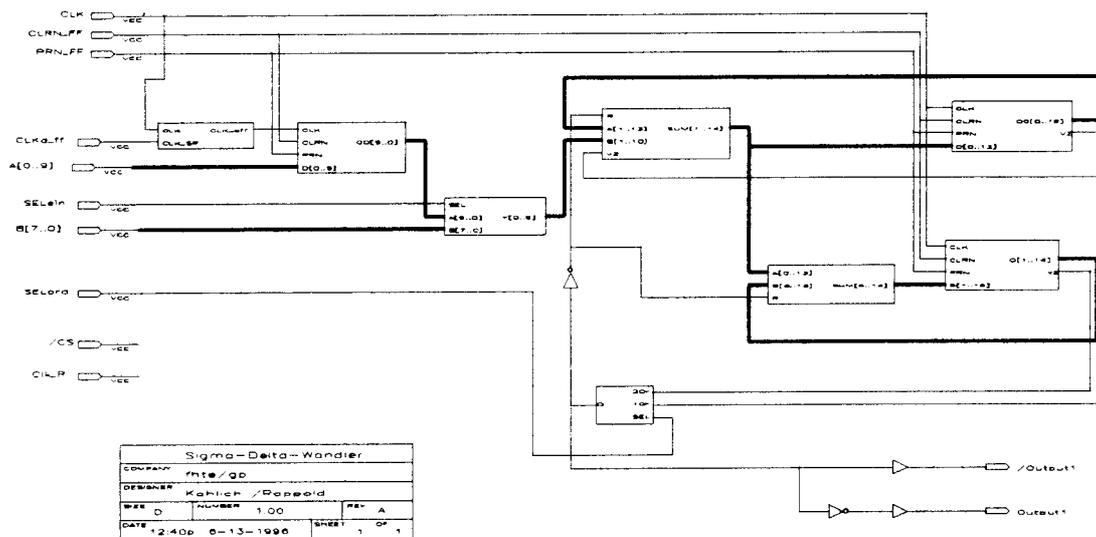


Bild 9: Innenschaltung Altera-PLD mit  $\Sigma\Delta$ -Wandler

#### Literatur:

- [1] Candy  
„Decimation for Sigma-Delta Modulation“  
IEEE Transaction of Data Communications, January 1986, S. 72- 80
- [2] Aziz, Sorensen, van der Spiegel  
„An Overview of Sigma-Delta Converters“  
IEEE Signal Processing Magazine, January 1996, S. 61- 84
- [3] Adamczyk  
„Digitale Verarbeitung eines analogen Winkelsignals unter Verwendung des Sigma-Delta-Verfahrens“, Diplomarbeit FHT Esslingen 1996



# **Die Gilbert-Zelle als HF-Mischstufe**

Analyse und Vermeidung von Störstellen

-

Prof. Dr.-Ing. Hans Sapotta

-

Fachhochschule Karlsruhe - Hochschule für Technik

Gliederung:

1. Repetitorium Intermodulation
2. Rahmenbedingungen im Empfänger
3. Das Störspektrum einer Gilbert-Zelle
4. Intermodulation und LO-Störstrahlung
5. Unterdrückung der 2LO-Störstrahlung
6. Zusammenfassung

# 1. Repetitorium Intermodulation

Die nichtlineare Kleinsignaltheorie beschreibt das Verhalten von realen, nichtlinearen Elementen bei Anregung mit Mehrtonsignalen. Im einfachsten Falle wird die Übertragungskennlinie durch eine Potenzreihe bis zum kubischen Glied angenähert:  $U_{\text{out}} = U_0 + \alpha \cdot U_{\text{in}} + \beta \cdot U_{\text{in}}^2 + \gamma \cdot U_{\text{in}}^3 + \dots$ . Wird diese Übertragungskennlinie nun mit einem Zweitonsignal der Form  $U_{\text{in}} = \mathbf{A} \cdot \cos \omega_1 t + \mathbf{B} \cdot \cos \omega_2 t$  angeregt, dann kann man durch Ausmultiplikation der Cosinus-Terme die folgenden Spektralkomponenten berechnen:

DC	$U_0 + \frac{1}{2}\beta A^2 + \frac{1}{2}\beta B^2$
$\omega_1$	$\alpha A + \frac{3}{4}\gamma A^3 + \frac{3}{2}\gamma AB^2$
$\omega_2$	$\alpha B + \frac{3}{4}\gamma B^3 + \frac{3}{2}\gamma BA^2$
$\omega_1 + \omega_2$	$\beta \cdot AB$
$\omega_1 - \omega_2$	$\beta \cdot AB$
$2\omega_1$	$\frac{1}{2}\beta \cdot A^2$
$2\omega_2$	$\frac{1}{2}\beta \cdot B^2$
$2\omega_1 + \omega_2$	$\frac{3}{4}\gamma \cdot A^2 B$
$2\omega_2 + \omega_1$	$\frac{3}{4}\gamma \cdot A B^2$
$2\omega_1 - \omega_2$	$\frac{3}{4}\gamma \cdot A^2 B$
$2\omega_2 - \omega_1$	$\frac{3}{4}\gamma \cdot A B^2$
$3\omega_1$	$\frac{1}{4}\gamma \cdot A^3$
$3\omega_2$	$\frac{1}{4}\gamma \cdot B^3$

Tabelle 1: Die Spektralkomponenten einer nichtlinearen Übertragungskennlinie bei Zweitonanregung

Neben einer Verschiebung des Arbeitspunktes (DC) findet man auf den Frequenzen  $\omega_1$  und  $\omega_2$  das linear verstärkte Eingangssignal und die Kreuzmodulationsterme. Neue Signale auf neuen Frequenzen werden generiert auf den Summen- und Differenzfrequenzen der beiden Eingangstöne sowie auf deren doppelten Frequenzen. Es handelt sich bei diesen Erscheinungen um Intermodulation 2. Ordnung. Der Intercept-Punkt 2. Ordnung (IP2) läßt sich daraus definieren als der Punkt, bei dem für identische Eingangsamplituden beider Eingangstöne das Signal auf der Differenzfrequenz die gleiche Größe wie das lineare Glied

$$\text{hat: } \alpha \cdot A_{\text{IP2}} = \beta \cdot A_{\text{IP2}}^2 \Rightarrow A_{\text{IP2}} = \frac{\alpha}{\beta}.$$

Neben der Intermodulation 2. Ordnung ist in der Praxis meist die Intermodulation 3. Ordnung von großer Bedeutung. Insbesondere die Signale auf den Frequenzen  $2\omega_1 - \omega_2$  und  $2\omega_2 - \omega_1$  lassen sich bei geringem Frequenzabstand nur unzureichend durch Filter unterdrücken. Daher werden diese Signale zur Definition der Intercept-Punktes 3. Ordnung herangezogen: der Intercept-Punkt 3. Ordnung (IP3) wird erreicht, wenn das Signal auf  $2\omega_1 - \omega_2$  gleich dem

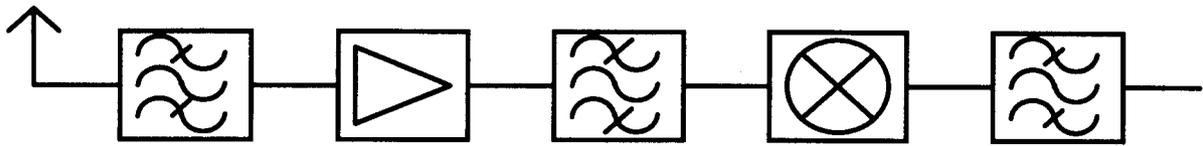
$$\text{linearen Term ist: } \alpha \cdot A_{\text{IP3}} = \frac{3}{4} \gamma \cdot A_{\text{IP3}}^3 \Rightarrow A_{\text{IP3}} = \sqrt{\left| \frac{4\alpha}{3\gamma} \right|}.$$

Wichtig bei alledem sei zunächst die Erkenntnis, daß zum Entstehen eines Störsignals in der Regel 2 Störsender erforderlich sind.

## 2. Rahmenbedingungen im Empfänger

Die Situation für einen Hersteller von Kommunikations-Hardware ist heutzutage im wesentlichen durch einen extremen Preiskampf gekennzeichnet. Daher wird Entwicklungsaufwand nicht verstanden als ein Mittel, um ein Gerät funktionsfähig zum Laufen zu bringen. Vielmehr ist das Handeln der Entwicklungsingenieure gerichtet auf eine Optimierung der Kosten-Performance-Relation bei niedrigsten Kosten für eine Großserienproduktion. Daraus resultiert der immer weiter fortschreitende Einsatz analoger hochintegrierter Schaltkreiskonzepte.

Die einzelnen Blöcke eines modernen Tuners, deren Aufgabe sowie deren technische Realisierung stellt Bild 1 vor.



Block	Vorkreis	Vorverstärker	Zwischenkreis	Mischer	ZF-Filter
Aufgabe	Vorselektion	rauscharme Verstärkung	Selektion und Spiegelunterdr.	Transformierung in ZF-Ebene	Symmetrierung und Hauptselektion
Realisierung	Schwingkreis	FET-Verstärker	Pi-Filter	Gilbert-Zelle	Schwingkreis + Piezo-Filter
Details					

Bild 1: Die Blöcke eines Tuners

Der **Vorkreis** soll zunächst eine grobe Vorselektion des an der Antenne anstehenden Signales erreichen. Aufgrund der endlichen Leerlaufgüte ( $Q_U$ ) des Kreises ergibt sich jedoch eine Einfügedämpfung  $IL$ , die umso geringer wird, je mehr der Schwingkreis durch die Antenne belastet wird. Diese Einfügedämpfung berechnet sich nach

$$IL = \frac{B_L}{B_L - B_U} = \frac{Q_U}{Q_U - Q_L},$$

wobei  $B_L$  die Bandbreite des belasteten Kreises,  $B_U$  die Leerlaufbandbreite des Kreises und  $Q_L$  die Güte des belasteten Kreises sind. Will der Entwickler durch eine schmalbandige Vorselektion die anschließende Signalverarbeitung von Außerbandsignalen entlasten, so muß er eine erhöhte Einfügedämpfung in Kauf nehmen; das heißt, die Rauschzahl des Gerätes steigt dementsprechend. Ist andererseits die Maximierung der Empfindlichkeit (die Minimierung der Rauschzahl) das Entwicklungsziel, dann muß der Vorkreis breitbandig ausgeführt werden. Wegen der zentralen Bedeutung der Leerlaufgüte der Vorkreisspule spricht man auch davon, daß die Performance des Empfängers im wesentlichen von der Güte der Vorkreisspule bestimmt wird. Da Miniaturisierung in modernen Konzepten von den äußeren Rahmenbedingungen vorgegeben ist, bleibt in der Regel nur der Ausweg über eine

Verschlechterung der Vorkreis Selektion, will man das erreichte Empfindlichkeitsniveau nicht verlassen.

An den Vorkreis schließt sich der **Vorverstärker** an. Wegen der geringen Rauschzahl haben sich an dieser Stelle Feldeffekttransistoren in Form von Mosfets oder GaAs-Fets durchgesetzt. Man erreicht Rauschzahlen in der Größenordnung von 0,5dB, die zusammen mit der Einfügedämpfung des Vorkreises von ca. 3,5dB und dem Nachrauschen der folgenden Stufen eine Gesamtrauschzahl von ca. 5dB erzeugen. Das Rauschen eines Feldeffekttransistors wird beschrieben durch eine Rauschstromquelle parallel zur Drain-Source-Strecke, deren Intensität proportional zur Wurzel aus der Steilheit ist. Der Signalstrom als Folge des Eingangssignals ist direkt proportional der Steilheit, woraus folgt, daß die Rauschzahl mit steigender Steilheit abnimmt. Bei gegebenem Drainstrom muß man die Steilheit maximieren, was durch einen Arbeitspunkt in der Nähe der Schwellspannung erreicht wird, wobei das W/L-Verhältnis bis an die Stabilitätsgrenze gesteigert wird. Selbstredend muß man die Kanallänge L minimieren. Durch diese Optimierung in Richtung auf die Rauschzahl verringert sich jedoch die Großsignalfestigkeit des Elementes. Der IP<sub>2</sub> ist ohnehin aufgrund der quadratischen Kennlinie gering. Weniger bekannt ist, daß ein derart betriebener Feldeffekttransistor auch einen bedeutend geringeren IP<sub>3</sub> als ein in Basisschaltung betriebener Bipolartransistor aufweist (0dBm gegenüber +10dBm bei I=10mA). Dies liegt daran, daß die Kennlinie eines Feldeffekttransistors nur abschnittsweise quadratisch verläuft.

Die Kombination aus breitem Eingangskreis mit schwachem Großsignalverhalten stellt eine gefährliche Herausforderung für das Systemkonzept dar.

Im Anschluß an den Vorverstärker soll der **Zwischenkreis** eine weitere, nicht am Rauschen orientierte Selektion durchführen, bevor die Mischstufe, der eigentliche Flaschenhals in der Signalverarbeitungskette, angesteuert wird. Im FM-Bereich erreicht man bei einer Zwischenfrequenz von 10,7MHz die für Geräte mit Antennenbuchse geforderte Spiegelfrequenzdämpfung von 57dB mit einem einfachen Schwingkreis nicht. Früher waren an dieser Stelle Bandfilter verwendet worden, die aus Kostengründen dem Pi-Filter weichen mußten. Ein Pi-Filter wie in Bild 1 gezeigt hat nicht nur eine mit der Frequenz über die Kapazitätsdiode einstellbare Polstelle, sondern gleichzeitig eine ebenfalls mitwandernde Nullstelle, die auf die Spiegelfrequenz gelegt werden kann. So läßt sich kostengünstig die Forderung nach Spiegelfrequenzdämpfung erfüllen. Leider geht gegenüber einem Bandfilter die Weitabselektion verloren; für Frequenzen, die groß gegen die Resonanzfrequenz sind, ist das Pi-Filter weitgehend transparent.

Damit können starke Außerbandsignale, die im Vorkreis nur ungenügend selektiert wurden, anschließend an den Vorverstärker mit geringer Aussteuerfestigkeit gelangen und dort Oberwellen erzeugen, den Mischer in Form von Grund- und Oberwellensignalen übersteuern. Die **Mischstufe** wird heutzutage weitestgehend als Gilbert-Zelle ausgeführt. Diese integrationsfreundliche Lösung hat als multiplikativer Mischer gegenüber additiven Mixern den Vorteil, eine Reihe von Störstellen weniger aufzuweisen. Dies kommt im wesentlichen durch die LO-Unterdrückung zum Ein- und Ausgang zustande. Die andere Alternative, der Diodenringmischer, ist nicht integrationsfreundlich und weist eine Einfügedämpfung gegenüber der Verstärkung in der Gilbert-Zelle auf.

Am Ausgang der Mischstufe kann das Signal dann über **ZF-Kreis** und Keramik-Filter dem Tuner entnommen und auf ZF-Ebene weiterverarbeitet werden.

### 3. Das Störspektrum einer Gilbert-Zelle

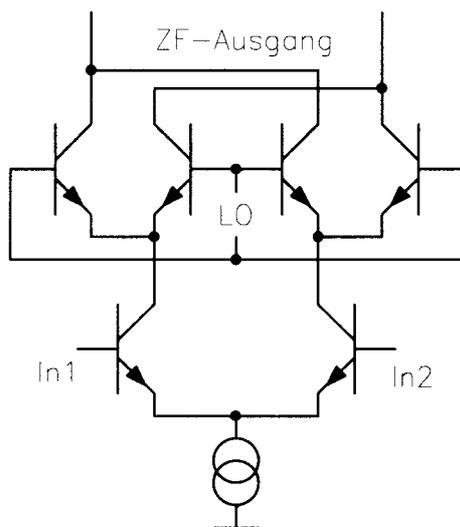


Bild 2: Schaltbild einer Gilbert-Zelle

Bild 2 zeigt nochmals den prinzipiellen Aufbau einer Gilbert-Zelle.

Sie besteht zunächst aus dem unteren **Differenzverstärker**, der mit den beiden Eingängen In1 und In2 angesteuert wird. Ein Vorteil dieses Konzeptes besteht in der Möglichkeit, das asymmetrische Signal des Pi-Filters im Differenzverstärker zu symmetrieren. Dazu wird ein Eingang des Differenzverstärkers gegen Masse kurzgeschlossen.

Das Großsignalverhalten des Differenzverstärkers an sich ist recht bescheiden. Der Intercept-Punkt dritter Ordnung, bezogen auf die Amplitude der Eingangsspannung, beträgt  $4U_T$ . In der Praxis wird daher meist eine Gegenkopplung in Form eines Widerstandsnetzwerkes an den

Emittern oder Strukturen höherer Komplexität angewendet. Durch diese Maßnahmen kann der Dynamikbereich des Mischers an die Verstärkung der Vorstufe angepaßt werden.

Der eigentliche Mischvorgang findet an den 4 **Schaltertransistoren** im oberen Bereich der Gilbert-Zelle statt. Die vier Transistoren invertieren im Rhythmus der Oszillatorfrequenz den Ausgangsstrom der Differenzverstärkerstufe. Zu den Zeitpunkten, an denen der Strom durch alle vier Schaltertransistoren gleich groß ist, beträgt die Verstärkung 0. Das Ausgangsrauschen wird jedoch aufgrund der nicht vorhandenen Korrelation der einzelnen internen Rauschquellen nur unzureichend gedämpft. Daher ist dies ein Betriebszustand, der unerwünscht ist. Um minimale Rauschzahlen zu erhalten, müssen die Schaltertransistoren hart in Form einer Rechteckfunktion angesteuert werden.

Dies hat Konsequenzen für das Störspektrum. Da für jeden Bipolartransistor (und jeden MOS-Transistor) die Einschaltzeit meist kürzer als die Ausschaltzeit ist (das Schaltverhalten asymmetrisch ist), entstehen bei jedem Schaltvorgang Schaltspitzen. Während einer Periode des LO-Signals finden zwei Schaltvorgänge statt, die Schaltspitzen treten also mit der doppelten Oszillatorfrequenz auf. In Bild 3 sind die Vorgänge im einzelnen erläutert.

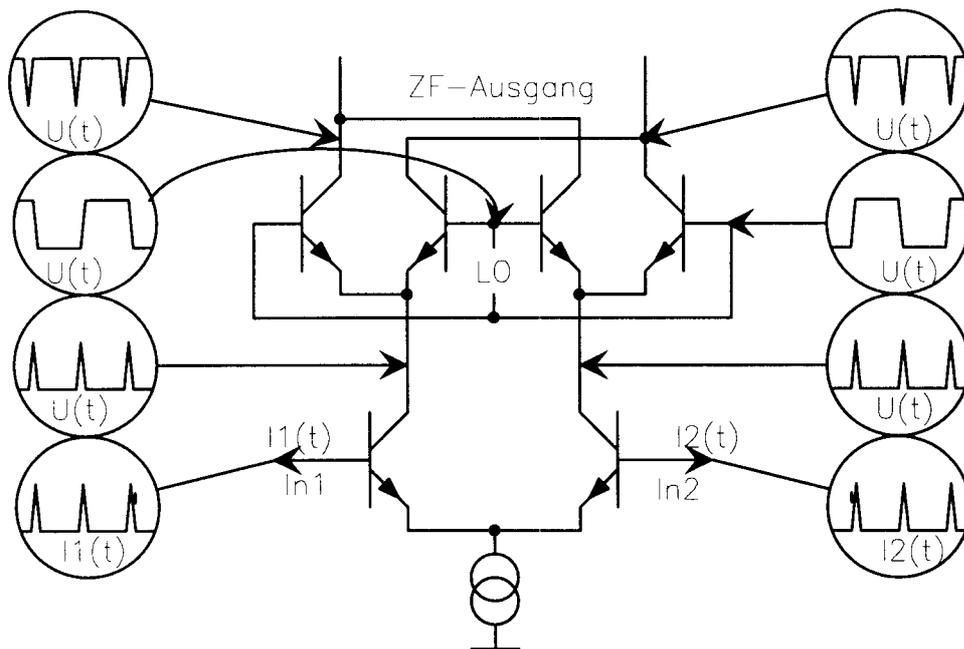


Bild 3: Die Schaltvorgänge in der Gilbert-Zelle

Die 4 Schaltertransistoren werden mit der symmetrischen LO-Spannung angesteuert. An den Kollektoren entsteht ein nadelförmiger Einbruch bei jedem Schaltvorgang. An dem ZF-Ausgang verläßt eine Gleichtaktwelle dieser Nadelimpulse den Schaltkreis.

Auf der anderen Seite entsteht an den Emittern der Schaltertransistoren der gleiche Nadelimpuls in positiver Phasenlage. Da nach dem Kirchhoffschen Gesetz Stromkreise geschlossen werden müssen, äußert sich dieser Nadelimpuls an den Emittern der Schalttransistoren in Form einer Gleichtaktwelle, die über die Kollektor-Basiskapazität der meist groß ausgeführten (Rauschen !) Differenzverstärkertransistoren den Schaltkreis an den Eingängen In1 und In2 verläßt. Der Stromkreis wird über externe Elemente bis zum ZF-Ausgang wieder geschlossen.

An den beiden Eingängen der Mischstufe steht also ein Gleichtaktsignal der doppelten Oszillatorfrequenz mitsamt weiterer Oberwellen an. In Bild 4 wird in schematischer Form dieses Spektrum grafisch dargestellt.

Dies ist solange von untergeordneter Bedeutung, wie dieser Eingangsstrom an beiden Eingängen auf identische Generatorwiderstände trifft. Aufgrund des asymmetrischen Pi-Filter-Konzeptes entsteht an dieser Stelle jedoch ein Problem.

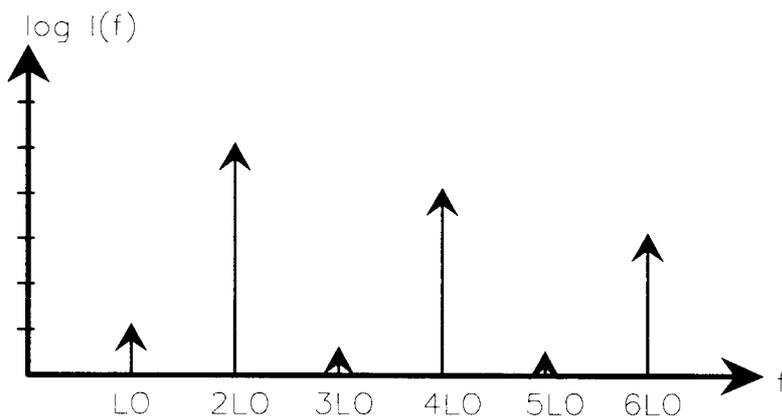


Bild 4: schematische Darstellung des Spektrums des Eingangsstroms (vert. 10dB/div)

## 4. Intermodulation und LO-Störstrahlung

Aufgrund der Forderung nach Rauschminimierung müssen die Schalttransistoren ein hartes Schaltverhalten zeigen. Das führt zu einem Gleichtakt-Ausgangsstrom am Differenzeingang der Mischstufe auf der doppelten Oszillatorfrequenz und deren Oberwellen. Dieser Gleichtakt-Ausgangsstrom trifft auf einen asymmetrischen Generatorwiderstand, so daß durch diesen Mechanismus ein Eingangssignal auf der doppelten Oszillatorfrequenz entsteht.

Die Konsequenz aus dieser Erkenntnis lautet, daß bereits **ein einziger** Störsender ausreicht, um auf ausgesuchten Eingangsfrequenzen Intermodulation **zusammen mit der Oszillatorfrequenz** zu verursachen. Das 2LO-Signal tritt als 2. Partner bei der Generierung von IM-Störsignalen auf. Da die LO-Frequenz in einer festen Beziehung zur eingestellten Eingangsfrequenz steht, treten bei fester Eingangsfrequenz Störsignale auf, wenn der Störsender in einem bestimmten Verhältnis zur Eingangsfrequenz steht. In Bild 5 sind beispielhaft einige mögliche Störkonfigurationen aufgetragen. Dem Praktiker sind die sogenannte ZF/2 und  $2/3$ ZF-Störstelle vertraut, die sich jeweils um den genannten Betrag oberhalb der Empfangsfrequenz befinden. Aufgrund der Breitbandigkeit des Eingangskreises, des ausgeprägten IM2-Verhaltens des Eingangstransistors und der mangelnden Weitabselektion treten insbesondere Intermodulationsprodukte in Kombination mit der doppelten Störfrequenz auf. Sogar auf der Spiegelfrequenz, die eigentlich durch das Pi-Filter besonders unterdrückt werden sollte, befindet sich eine Störstelle.

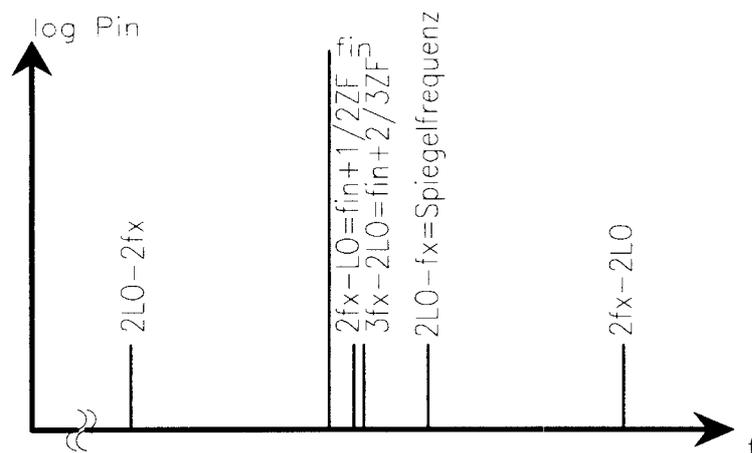


Bild 5: Störstellen eines Störsenders  $f_x$  bei fest auf  $f_{in}$  abgestimmtem Empfänger:

## 5. Unterdrückung der 2LO-Störstrahlung

Aus dem bisher gesagten folgt eindeutig die Notwendigkeit, die Störstrahlung der Mischstufe auf der doppelten LO-Frequenz zu unterdrücken. Als Ansatz dazu bietet sich die Tatsache an, daß alle Nutzsignale in der Mischstufe Gegentaktsignale sind und die LO-Störstrahlung als Gleichtaktsignal auftritt.

Die Störstrahlung kann im Modell als jeweils eine Störstromquelle parallel zu der Kollektor-Emitter-Strecke eines Schalttransistorpaares gesehen werden. Bild 6 verdeutlicht diesen Ansatz.

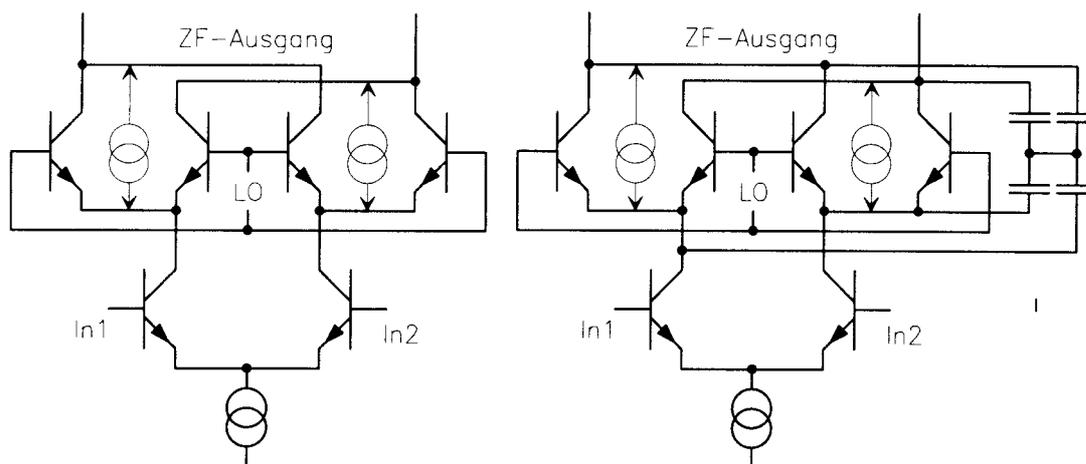


Bild 6: Problem und Lösung der 2LO-Störstrahlung

Auf der rechten Seite dieses Bildes ist gleichzeitig auch die sich ergebende Lösung dieses Problems gezeigt. Ein vierpoliger, symmetrischer Kondensatorstern aus relativ kleinen, integrationsfähigen Kondensatoren stellt für den Ausgangsstrom der beiden Differenzverstärkertransistoren ein hochohmiges Gebilde gegenüber der Eingangsadmittanz der Schalttransistoren dar. Die Gleichheit beider Kondensatoren sorgt dafür, daß kein Strom auf der Eingangsfrequenz an den Schalttransistoren vorbei die Schaltstufe passieren kann.

Auf der Ausgangsseite der Mischstufe ist die zusätzliche kapazitive Belastung in Form zweier, in Serie geschalteter Kondensatoren angesichts des angeschlossenen Ausgangskreises problemlos kompensierbar.

Für die in der Mischstufe vorkommenden Nutzsignale hat dieser Kondensatorstern also keinen wesentlichen Einfluß. Die entstehenden Gleichtakt-Störsignale werden allerdings chipintern auf kürzestem Wege kurzgeschlossen und können die Ein- und Ausgänge nur noch in abgeschwächter Form verlassen.

Die Anordnung wurde simulatorisch untersucht. Für den Fall einer FM-Mischstufe bei 100MHz mit  $I=10\text{mA}$  wurde in einem ersten zaghaften Ansatz der Kondensator zu  $C=3\text{pF}$  gewählt. Es ergab sich eine Reduzierung der 2LO-Störstrahlung um 15dB. Dies an sich ist bereits ein vernünftiger Wert. Einer Erweiterung auf größere Kapazitätswerte steht primär nichts im Wege, gegebenenfalls muß man das relative Toleranzproblem beachten.

In jedem Falle ist diese Lösung erheblich kostengünstiger zu realisieren als alle Symmetrierungsmaßnahmen am Mischereingang.

Neben der integrierten Lösung ist ein vergleichbares Konzept in diskreten Diodenringmischern ebenso denkbar. Das Problem der Symmetrierung aufgrund der größeren relativen Toleranz diskreter Elemente tritt dann nur stärker in den Vordergrund.

## 6. Zusammenfassung

Zusammenfassend die Thesen dieser Präsentation:

- eine Gilbert-Zelle unterdrückt die LO-Strahlung auf der Grundfrequenz
- die 2LO-Strahlung und deren Oberwellen treten am Ein- und Ausgang stark auf
- moderne Empfängerkonzepte weisen eine nur mäßige Weitabselektion auf
- die 2LO-Strahlung bewirkt bereits mit einem Störsender IM2 und IM3
- ein einfaches Kompensationsnetzwerk reduziert die 2LO-Störstrahlung
- die Ergebnisse sind auf Diodenringmischer in gleicher Weise übertragbar



# Dimensionierung von integrierten Analogschaltungen mit OPSIM

H. J. Aleker, G. Forster  
ASIC-Labor, Fachhochschule Ulm

## 1 Einleitung

### 1.1 Aufgabenstellung

Der Entwurf von integrierten Analogschaltungen erfolgt heute fast ausschließlich mit Hilfe von Schaltungssimulatoren. Diese Simulatoren können im allgemeinen nichtlineare Gleichstrom-, nichtlineare Einschwing- und lineare Kleinsignal-Wechselstrom-Analysen bei unterschiedlichen Temperaturen durchführen. Dabei werden die elektronischen Bauelemente durch mathematische Modelle beschrieben. Je detaillierter die Modelle die Eigenschaften der Bauelemente wiedergeben, desto aufwendiger wird die Schaltungsberechnung. Für die Handanalyse sind deshalb nur die einfachen Modelle der Bauelemente geeignet. Zur genaueren Untersuchung des Schaltungsverhaltens ist immer der Einsatz eines Simulators notwendig. Durch die hohe Leistungsfähigkeit der heutigen Rechner und die ständige Verbesserung der Simulationsalgorithmen stellt die Berechnung auch komplexer integrierter Analogschaltungen kein allzu großes Problem mehr dar.

Seit ein paar Jahren wird nun auch die Entwicklung von Optimierungswerkzeugen vorangetrieben, die eine automatische Dimensionierung von integrierten Analogschaltungen möglich machen sollen. Im Rahmen einer Diplomarbeit wurde ein solches Optimierungstool am Beispiel der Dimensionierung eines CMOS-OTAs untersucht. Ausgehend von einer High-Speed-Dimensionierung sollte der CMOS-OTA für eine Low-Power-Anwendung mit Hilfe des Optimierungstools OPSIM automatisch umdimensioniert werden.

### 1.2 Motivation

Ziel der Dimensionierung einer Schaltung ist es, die Schaltungsparameter so einzustellen, daß die vorgegebene Spezifikation erreicht wird. Der Ablauf einer Dimensionierung nach der klassischen Vorgehensweise würde dann wie folgt aussehen: Zuerst versucht der Designer durch eine näherungsweise manuelle Dimensionierung die Spezifikation zu erfüllen. Danach müssen dann die einzelnen Vorgaben Schritt für Schritt überprüft und geeignete Schaltungskorrekturen vorgenommen werden. Sobald alle Einzelziele gleichzeitig der Spezifikation entsprechen, ist die Dimensionierung mit Erfolg abgeschlossen. Dieser Vorgang ist sehr zeitaufwendig, da er in der Regel viele Iterationsschleifen erfordert. Werkzeuge zur Beschleunigung des Dimensionierungsvorgangs sind deshalb von großem Interesse. Ein wichtiges Hilfsmittel ist dabei die lineare Optimierung.

### 1.3 Das Prinzip der Optimierung

Um eine Dimensionierungsaufgabe rechnergesteuert lösen zu können, ist eine mathematische Darstellung in Form eines Optimierungsproblems erforderlich. Die Designparameter sind die unabhängigen Variablen (z. B.  $W$ ,  $L$ ,  $\beta$ ,  $R$ ,  $C$ ...). Die Schaltungseigenschaften sind die abhängigen Größen (z. B. *Aussteuerbereich*, *Offset*, *Transitfrequenz*, *Phasenreserve* ...). Man erhält eine simulierte Istfunktion  $s$  und eine vom Designer vorgegebene Zielfunktion  $z$ .

$$s = f(W_1, \dots, W_n, L_1, \dots, L_n)$$

$Z$  ist eine vom Designer vorgegebene Zielfunktion

$S_i$  ist der simulierte Wert für den diskreten Eingangswert  $X_i$

$Z_i$  ist der Wert der Zielfunktion für den diskreten Eingangswert  $X_i$

Die Abweichung zwischen simuliertem Ergebnis und Zielfunktion ergibt die Fehlerfunktion für eine Zielvorgabe.

$$error = \sum_{i=1}^n \left( \frac{z_i - s_i}{z_i} \right)^2$$

Normalerweise werden immer mehrere Ziele angestrebt, denn es sollen gleichzeitig verschiedene Zielfunktionen erreicht werden (z. B. die Transferkennlinie und der Frequenzgang).

Die Gesamtfehlerfunktion ergibt sich dann bei  $m$  Zielvorgaben zu:

$$error_{total} = G_1 * error_1 + G_2 * error_2 + \dots + G_m * error_m$$

Hierbei sind  $G_1 \dots G_m$  die Gewichtungsfaktoren der verschiedenen Ziele untereinander.

Die Gewichtungsfaktoren sind notwendig, da nicht alle Ziele bei einer Optimierung gleich wichtig sind.

***Aufgabe einer Optimierung ist es die Gesamtfehlerfunktion zu minimieren.***

## 2 Das Optimierungstool OPSIM

### 2.1 Überblick

OPSIM stellt ein Optimierungsprogramm für elektronische Schaltungen dar [1]. Die Werte der Parameter einer Schaltung werden damit automatisch so geändert, daß das Verhalten der Schaltung der vorgegebenen Spezifikation näher kommt. Die Ergebnisse, die OPSIM erzielt, sind also stets besser als die, welche mit der iterativen Methode am Simulator gefunden werden. OPSIM greift ebenfalls auf einen Analog-Simulator zu und berechnet damit die Ist-Funktion. Die Aufgabe von OPSIM besteht darin, den Optimierungsvorgang zu steuern und die Netzliste bei den einzelnen Iterationen korrekt an den Simulator zu übergeben. Der Optimierungsalgorithmus berechnet die Abweichung zwischen Zielfunktion und simulierter Istfunktion und verändert die Schaltungsparameter in der Art, daß die Abweichung verkleinert wird. Wenn die Abweichung auf ein Minimum reduziert worden ist, wird der Optimierungsvorgang beendet, und die neuen Werte der Parameter stehen dem Anwender zur Verfügung. Der Algorithmus arbeitet nach dem Gradientenverfahren und ermöglicht somit nur das Auffinden von lokalen Minima.

### 2.2 Ablauf einer Optimierung

OPSIM benötigt für die Durchführung einer Optimierung nur zwei ASCII-Files, nämlich eine modifizierte Netzliste, die zur Bestimmung der Schaltungstopologie dient, und ein Control-File, das die Optimierung steuert.

#### 2.2.1 Die modifizierte Netzliste

Zuerst wird die Schaltung im DESIGN ARCHITECT gezeichnet. Danach wird das Sheet in ACCUSIM geladen. Im *Pull-Down-Menü SETUP* gibt es einen Befehl *Create OPSIM Netlist*. Nachdem die Netzliste erzeugt wurde, kann sie in einem Editor für OPSIM vorbereitet werden. Dabei müssen die Designvariablen festgelegt und syntaktische Änderungen vorgenommen werden. Die Designvariablen sind diejenigen Parameter, die von OPSIM automatisch geändert werden, um das gewünschte Schaltungsverhalten zu erzielen. Der Anwender muß sich also vor einer Optimierung immer darüber Gedanken machen, welche Parameter überhaupt geändert werden sollen. Die Designparameter müssen sehr sorgfältig ausgewählt werden, da der Erfolg einer Optimierung davon stark abhängt.

### 2.2.2 Das Control-File

Im Control-File werden die für die Optimierungssteuerung notwendigen Informationen eingegeben.

- Festlegen der Anfangswerte und der Wertebereiche der Designvariablen
- Einstellen der Optionen
- Definition der Zielfunktionen
- Auswahl der Analysearten

### 2.2.3 Die Qualität des Ergebnisses

Mit OPSIM kann die Qualität eines Ergebnisses auf verschiedene Arten beeinflusst werden. Durch die Angabe von Gewichtungsfaktoren können den einzelnen Zielen unterschiedliche Prioritäten zugeordnet werden. Es ist auch möglich, ein Ziel in Bereiche mit verschiedenen Gewichtungen einzuteilen. Außerdem kann die Berechnung der Gesamtfehlerfunktion vom Designer über ein spezielles OPSIM-Kommando eingestellt werden.

## 2.3 Was kann optimiert werden?

OPSIM kann gleichzeitig Vorgaben für die Ergebnisse einer AC-, DC- oder Transient-Analyse optimieren. Während der Optimierung werden die simulierten Istfunktionen und die jeweiligen Zielvorgaben online am Bildschirm dargestellt. Dadurch hat der Designer die Möglichkeit, die Änderungen zu beobachten, die bei den einzelnen Iterationsschritten gemacht werden. Durch die Grenzen der Herstellungsverfahren von integrierten Anlogschaltungen können die Schaltungsparameter oft nur in diskreten Schritten geändert werden. Auch dies läßt sich bei OPSIM berücksichtigen. Die Möglichkeiten von OPSIM werden nur durch die Möglichkeiten des verwendeten Simulators eingeschränkt.

### 2.3.1 Die Designparameter

Als Designparameter kommen alle SPICE-Parameter in Frage, die der Simulator als Eingangswerte zuläßt:

- $R, C, L, I_{EE}, U_B \dots$
- $BF, BR, XTF, VTF \dots$
- $W, L, LD, VTO \dots$

### 2.3.2 Die Zielfunktionen

Als Zielfunktionen können alle Ausgangswerte des Simulators benutzt werden. In der Regel sind dies Ströme und Spannungen. Über Funktionsdefinitionen können aber auch Verlustleistung, Chipfläche und andere Größen berechnet und optimiert werden. OPSIM kann eine Optimierung in folgenden Analysearten durchführen:

- DC sweep, temperature sweep
- AC sweep
- Transient sweep

## 3 Die Optimierung eines CMOS-OTAs

### 3.1 Schaltungsbeschreibung

Am Beispiel eines CMOS-OTAs wurde die Arbeitsweise bei einer Dimensionierung mit OPSIM untersucht. Ein OTA (Operational Transconductance Amplifier) ist ein spezieller Operationsverstärker mit hoher Bandbreite. Bild 1 zeigt das Schaltbild des gewählten CMOS-OTAs mit gefalteter Kaskode [2,3]. Die Transistoren M12, M11, M10 und M13 arbeiten als Stromspiegel und dienen der Schaltung zur Einstellung des Arbeitsstroms. Die Differenzstufe, bestehend aus M1 und M2, arbeitet auf eine aktive Last. Diese aktive Last besteht aus Stromspiegel (M3, M4) unten und aus Stromspiegel (M8, M9) oben jeweils mit Kaskode. Durch die Kaskodeschaltung entfällt der Millereffekt. Der Ausgang ist belastet mit einem Lastwiderstand  $R_L$  und einer Lastkapazität  $C_L$ . Da der hochohmigste Knoten des Verstärkers am Ausgang liegt, ist dieser für den dominanten Pol verantwortlich. Ein spezieller Kompensationskondensator ist nicht erforderlich, da der Verstärker schon über den Lastkondensator stabilisiert wird. Über die beiden Spannungsquellen VB1 und VB2 kann der Aussteuerbereich des Verstärkers eingestellt werden. Die Stromquelle IS bestimmt den Strom für die Stromspiegel. Die Dimensionierung der Schaltung geschieht über die Weiten und Längen der MOSFETs. Bei der Simulation mit ACCUSIM müssen die Weiten  $W$  und Längen  $L$  im Property instpar der Transistoren eingetragen werden. Zur Berechnung des dynamischen Verhaltens benötigt ACCUSIM außerdem von jedem Transistor die Drain-Diffusionsfläche  $A_D$ , die Source-Diffusionsfläche  $A_S$ , den Umfang der Drain-Sperrschicht  $P_D$  und den Umfang der Source-Sperrschicht  $P_S$ . Diese Parameter werden ebenfalls in das Property instpar eingefügt. Im Schaltbild sind diese sechs Parameter bei jedem Transistor zu sehen.

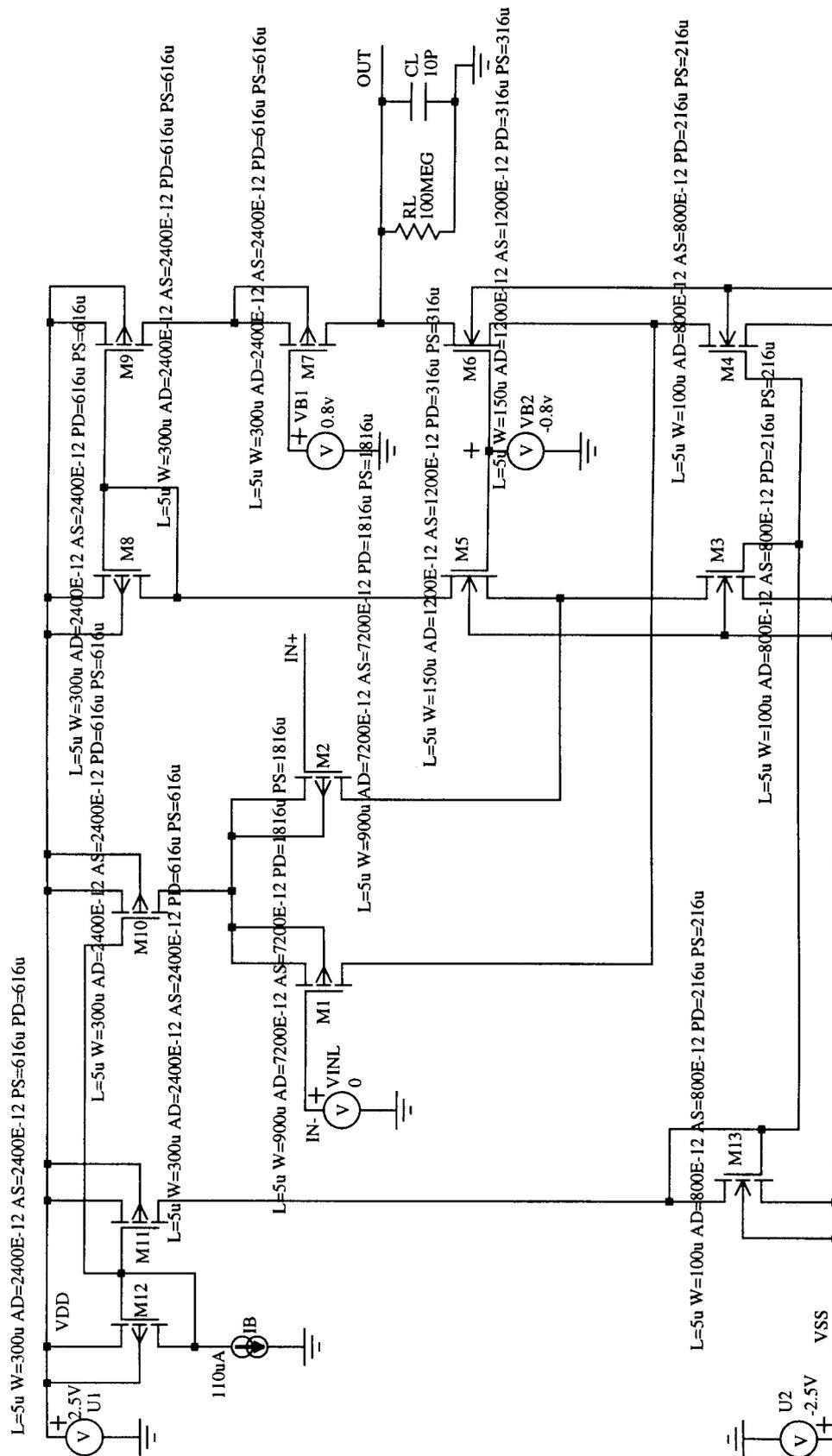


Bild 1: CMOS-OTA mit gefalteter Kaskode

### 3.2 Die Vorbereitung der Optimierung

Ausgehend von einer aktuellen High-Speed-Dimensionierung sollte der CMOS-OTA automatisch mit OPSIM für eine Low-Power-Anwendung umdimensioniert werden. Die Optimierung umfaßte sieben Zielfunktionen. In der ersten Spalte sind die einzelnen Zielfunktionen dargestellt. Die zweite Spalte zeigt die Werte der einzelnen Zielfunktionen für die aktuelle High-Speed-Dimensionierung. Die dritten Spalte beschreibt die erwünschte Dimensionierung. Dabei soll insbesondere die Gesamtstromaufnahme gegenüber der High-Speed-Dimensionierung stark reduziert werden. In der letzten Spalte sind die Simulationsarten für die jeweiligen Ziele angegeben. Das Hauptziel war die Minimierung der Gesamtstromaufnahme auf unter  $15 \mu\text{A}$ .

Zielfunktion	High-Speed <i>aktuelle Dimensionierung</i>	Low-Power <i>erwünschte Dimensionierung</i>	Einstellung von OPSIM
Aussteuerbereich	-1.7V +1.5V	-1.5V +1.5V	DC-Sweep Transferkennlinie
Gesamtstromaufnahme	347 $\mu\text{A}$	$\leq 15\mu\text{A}$	DC-Sweep automatische Berechnung
Chipfläche	73500 $\mu\text{m}^2$	40000 $\mu\text{m}^2$	DC-Sweep automatische Berechnung
Transitfrequenz	7.1MHz	$\geq 1\text{MHz}$	AC-Sweep Amplitudengang Limit bei 1MHz
Phasenreserve	57.2°	$\geq 60^\circ$	AC-Sweep spezieller Opsimbefehl
NF-Verstärkung	87.4dB	80dB	AC-Sweep Amplitudengang 80dB
Slewrates	7.93 $\frac{\text{V}}{\mu\text{s}}$	so groß wie möglich	Transient kein Optimierungsziel nur Simulation

## Besonderheiten

Für diese Optimierung waren nun einige Besonderheiten nötig.

Notwendig war die automatische Berechnung von:

- Chipfläche
- Gesamtstromaufnahme
- AD, AS, PD, PS zur Berücksichtigung der parasitären Kapazitäten

Außerdem mußte der CMOS-OTA zur Berechnung der Slewrate zusätzlich als Spannungsfolger beschaltet werden.

Die automatische Berechnung von Chipfläche, Gesamtstromaufnahme und der parasitären Kapazitätsflächen kann OPSIM über Funktionsdefinitionen durchführen. Außerdem bietet OPSIM die Möglichkeit, verschiedene Beschaltungen gleichzeitig zu optimieren.

Nachdem die modifizierte Netzliste und das Control-File entwickelt sind, kann die Optimierung gestartet werden. Bei der Optimierung treten nun einige Probleme auf. Der Designer muß geeignete Strategien entwickeln, um eine sinnvolle Optimierung zu ermöglichen. Bild 2 zeigt ein Beispiel für ein gutes Optimierungsergebnis. Links oben ist der Verlauf der Fehlerfunktion dargestellt. Die Optimierung wurde nach 36 Iterationen beendet, da sich die Fehlerfunktion nicht mehr verkleinern ließ. Während der Optimierung können die Änderungen der Werte der unabhängigen Variablen ständig in dem mittleren Fenster in der linken Bildhälfte beobachtet werden. Auf der rechten Seite sind die einzelnen Zielfunktionen jeweils in einem eigenen Fenster dargestellt. Die kleinen Quadrate kennzeichnen den gewünschten Funktionsverlauf. Die Grenzen werden durch die schraffierten Balken vorgegeben. Die dünne Linie stellt die simulierte Istfunktion dar.

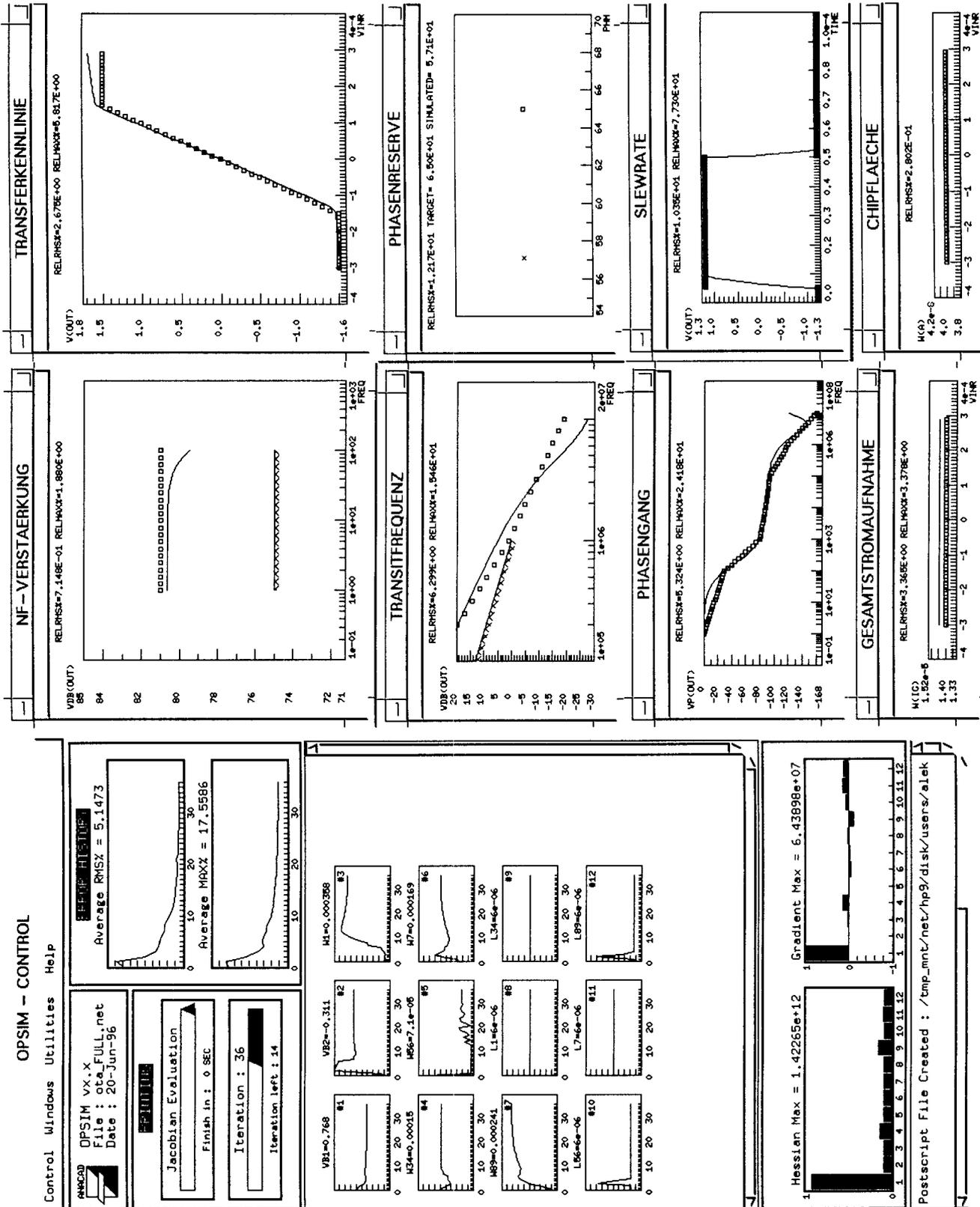


Bild 2: Optimierungsergebnis des CMOS-OTAs mit OPSIM

### 3.3 Die Ergebnisse der Optimierung

Die Vorgaben und die Ergebnisse der Optimierung sind in der Tabelle dargestellt. Die letzte Spalte beschreibt das Verhältnis von Low-Power zu High-Speed für die einzelnen Zielvorgaben. Daraus ist zu erkennen, daß die Gesamtstromaufnahme auf 4,2 % des Anfangswertes verringert wurde.

Zielfunktion	Low-Power erwünschte Dimensionierung	Low-Power erreichte Optimierung	Vergleich Lowp./Highs.
Aussteuerbereich	-1.5V +1.5V	-1.5V +1.5V	0.88 1
Gesamtstromaufnahme	$\leq 15\mu\text{A}$	$\leq 14.6\mu\text{A}$	0.042
Chipfläche	$40000\mu\text{m}^2$	$40500\mu\text{m}^2$	0.55
Transitfrequenz	1MHz	1.6MHz	0.22
Phasenreserve	$\geq 60^\circ$	$57^\circ$	0.997
NF-Verstärkung	80dB	80.7dB	0.46
Slewrate	so groß wie möglich	$0.61 \frac{\text{V}}{\mu\text{s}}$	0.077

### 3.4 Einschätzung und Ausblick

Die Dimensionierung von integrierten Anlogschaltungen mit OPSIM bringt gute Ergebnisse. Der Schaltungsdesigner benötigt aber ein fundiertes Wissen über die Probleme, die bei einer Optimierung auftreten können. Der Optimierer ist nur in der Lage, die Fehlerfunktion zu minimieren. Es ist die Aufgabe des Anwenders, Strategien zu entwickeln, die die Optimierung zum Erfolg führen. OPSIM ermöglicht die gleichzeitige Simulation aller interessierenden Größen. Dadurch kann die Dimensionierungszeit erheblich verringert werden. Die Möglichkeit, unterschiedliche Beschaltungen bei einer Optimierung zu erlauben, spricht ebenfalls für OPSIM. Die Optimierungswerkzeuge sind allerdings noch in der Entwicklungsphase. Eine bessere Optimierungssteuerung und Batchbetrieb wären wünschenswert. Wichtigstes Ziel wäre natürlich das Auffinden von Lösungen in der Nähe des globalen Minimums. Hierzu gibt es Ansätze mit dem sog. Simulated Annealing [4]. Da die Vorteile der Optimierungswerkzeuge sehr deutlich sind, wird die Entwicklung dieser Tools in den nächsten Jahren sicher stark voranschreiten.

## 4 Literatur

- [1] OPSIM User's Manual Revision v. 3.0, ANACAD 1996
- [2] Phillip E. Allen, Douglas R. Holberg, „CMOS Analog Circuit Design“, Holt, Rinehart & Winston, San Francisco 1987
- [3] Kenneth R. Laker, Willy M. C. Sansen, „Design of Analog Integrated Circuits and Systems“, Mc. Graw-Hill, New York 1994
- [4] G. Gielen , H. Walscharts, W. Sansen, „Analog Circuit Design Optimization Based on Symbolic Simulation and Simulated Annealing“, IEEE Journal of Solid-State Circuits, Vol. SC-25, No.3, pp. 707-713, June 1990



# Thermologger

## Applikation des Mikroprozessorkernels FHOP in einem Thermologger ASIC

Dipl. Ing. (FH) Th. Klumpp, Prof. Dr.-Ing. D. Jansen

ASIC Design Center  
Fachhochschule Offenburg

email: klumpp@fh-offenburg.de

Juli 1996

Nach dem Nachweis der Funktionalität des an der Fachhochschule Offenburg entwickelten Mikroprozessorkernels *FHOP* (First Homemade Operational Processor) [2], wird eine Anwendung des Kernels in einem Applikationschip beschrieben.

Der Thermologger-ASIC soll mit Hilfe eines Temperatursensors die Umgebungstemperatur bei technischen Prozessen in regelmäßigen Zeitabständen erfassen und abspeichern. Die Meßwerte werden bei Bedarf über eine serielle Schnittstelle des Thermologger-ASICs an einen PC übertragen und ausgewertet. Zur Verringerung der Leistungsaufnahme wird zwischen zwei Temperaturmessungen in einen Power-Down-Mode geschaltet.

Der ASIC soll später in einer Chipkarte integriert werden.

### Das Gesamtsystem

Die Anordnung in Abb.1 zeigt das Gesamtsystem, wie es in seiner endgültigen Version aussehen soll.

Die *Chipkarte* kennt zwei Betriebszustände. Ist sie vom System abgekoppelt ("Standalone Modus") führt sie in vorgegebenen Zeitabständen Messungen der Umgebungstemperatur durch. Zwischen den Messungen geht sie in einen stromsparenden Modus. Die erfaßten Temperaturwerte werden in einem internen Speicher abgelegt. Dieser Betriebszustand wird solange beibehalten, bis die Chipkarte in das *Kartenzugangsgerät* eingeführt wird. Ist der elektrische Kontakt hergestellt, wird die serielle Schnittstelle des Thermologger-ASICs aktiviert. Mit Hilfe des *PCs* können die Temperaturwerte über die serielle Schnittstelle ausgelesen werden. Weiterhin kann die Chipkarte mit dem PC konfiguriert werden. So kann z.B. die Zeitdauer eines Meßzyklus eingestellt werden.

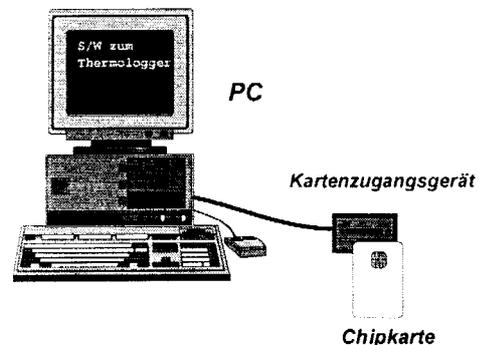


Abb.1: Die auf der Chipkarte abgelegten Temperaturwerte können über das Kartenzugangsgerät mit dem PC ausgelesen werden.

### Aufbau der Chipkarte

Abb.2 zeigt die Anordnung der geplanten Komponenten auf der Chipkarte. Als zentrale Komponente ist der Thermologger-ASIC zu erkennen.

Über das Kontaktfeld nach Norm der Telefonschnittstelle [3] erfolgt die elektrische Kontaktierung an ein Kartenzugangsgerät. Von dort erhält der ASIC im angekoppelten Zustand seine Betriebsspannung und seinen Systemtakt und ist außerdem mit einem angeschlossenen PC seriell verbunden. In dieser Betriebsart wird der Chip konfiguriert.

Befindet sich die Chipkarte im "Standalone-Modus" muß die Taktversorgung mit einem Quarz "on board" vonstatten gehen. Der ASIC wird mit einer Batterie versorgt.

Beim Temperatursensor handelt es sich um einen Baustein der Firma Dallas (DS 1620), der die Temperaturwerte in einem 9-bit-Wert seriell über eine Drei-Leiter-Schnittstelle sendet.

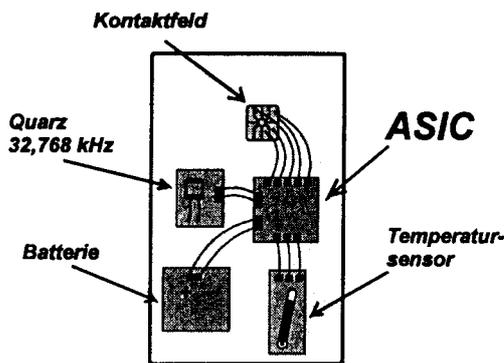


Abb.2: Auf der Chipkarte ist der ASIC, ein Quarz, eine Batterie und ein Temperatursensor zusammengefaßt. Die elektrische Kontaktierung erfolgt über das Kontaktfeld der Telefonschnittstelle.

### Funktionsmerkmale des ASICs

Der Thermologger-ASIC besitzt eine serielle Schnittstelle mit einer Übertragungsrate von 9600 Baud und einem Datenformat bestehend aus einem Startbit (Low), acht Datenbits, einem Paritybit (Parity Odd) und einem Stoppbit (High). Es existiert nur eine Datenleitung für beide Richtungen, die als Open-Kollektor-Leitung ausgeführt ist. Es kann nur abwechselnd gesendet oder empfangen werden (Halbduplex).

Zum Temperatursensor existiert eine Drei-Leiter-Schnittstelle. Über die Datenleitung (DQ), werden Steuerworte und Daten synchron im Takt der Steuerleitung (CLK) gesendet. Eine zweite Steuerleitung ( $\_RST$ ) ermöglicht Zugriffe nur im High-Zustand.

Die vom Temperatursensor kommenden Daten werden in einem RAM abgelegt. Er ist so groß, daß mit Unterstützung eines Kompressionsalgorithmus 40.000 Meßwerte bei einer Genauigkeit von  $\pm 1^\circ\text{C}$  erfaßt werden können.

Zwischen zwei Messungen sollen nicht benötigte Komponenten abgeschaltet werden. Das dafür zuständige Powermanagement hält, nach einer durchgeführten Messung, den Takt für nicht benötigte Komponenten bis zum Ablauf der festgelegten Zeit auf konstantem Potential. Dadurch wird der Leistungsverbrauch auf ein Minimum reduziert.

Für den Wechsel vom angekoppelten Modus in den "Standalone-Modus" wird der Systemtakt durch eine Taktumschaltung von externer Zufuhr auf interne Versorgung geschaltet.

### Blockschaltbild des ASICs

Abb.3 zeigt, welche Funktionseinheiten im Innern des ASICs zu finden sind.

Der Mikroprozessorkernel FHOP steuert das System. Der für den Betrieb des FHOP notwendige Programmcode liegt überwiegend im vier kilobyte

großen ROM. Im ROM sind die unverknüpften Betriebssystemfunktionen (Interrupt-Service-Routinen) zu finden. Der im RAM stehenden Teil des Programmcodes verbindet die einzelnen Interrupt-Service-Routinen miteinander. Dadurch ist das System auf unterschiedliche Applikationen anpaßbar. Der RAM-Speicher ist acht kilobyte groß und enthält Temperaturmeßwerte sowie Programmcode.

In der Interfacebaugruppe sitzen die Schnittstelle zum Kartenzugangsgerät, die Schnittstelle zum Temperatursensor, das Powermanagement und die Taktumschaltung.

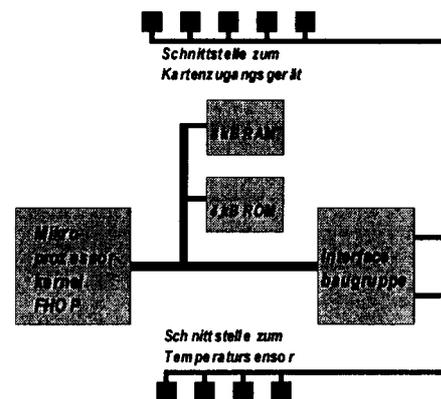


Abb.3: Der Thermologger-ASIC besteht aus einem Mikroprozessor, zwei Speichern und der Interfacebaugruppe.

### Blockschaltbild der Interfacebaugruppe

Abb.4 zeigt die Anordnung der Komponenten, aus der die Interfacebaugruppe besteht.

Die Adressierungslogik leitet das RD- und WR-Signal entsprechend der angelegten I/O-Adresse an die richtigen Baugruppen weiter. So erfolgt z.B. bei einem Schreibzugriff auf die I/O-Adresse 00h das Beschreiben des sechs Bit breiten Steuerregisters. Die einzelnen Bits des Steuerregisters haben Funktionen wie z.B. "Takt umschalten" (Bit 1). Das Statusregister mit seiner Breite von drei Bit zeigt den Zustand der Datenübertragung der seriellen Schnittstelle an. Bei einer noch andauernden Datenübertragung befindet sich z.B. das Statusbit "BUSY" (Bit 0) im Zustand High.

Bei der Schnittstelle zum Temperatursensor handelt es sich um zwei Register - ein 3-bit breites Ausgaberegister und ein 1-bit breites Eingaberegister

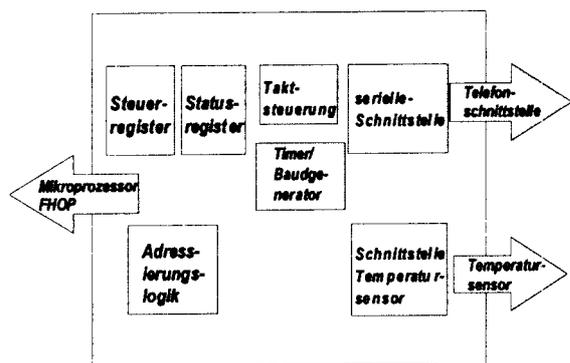


Abb.4: Die Interfacebaugruppe enthält viele weitere Komponenten.

Auf die Einheiten Taktsteuerung, Timer/Baudraten-generator, Powermanagement und serielle Schnittstelle soll im weiteren noch genauer eingegangen werden.

### Taktsteuerung

Bei der Taktsteuerung handelt es sich um die Taktumschaltung kombiniert mit der Taktabschaltung. Abb.5 zeigt symbolisch die Funktionsweise. An der Taktsteuerung stehen zwei verschiedene Takte an. Ein 4,9152 MHz Takt, der über die Telefonschnittstelle vom Kartenzugangsgerät kommt, und ein 32,768 kHz Takt, der mit einem Quarz in Verbindung mit Oszillatorpads erzeugt wird. Mit der Steuerleitung "clock\_int" wird zwischen den Takten umgeschaltet. Während der Takt für den Timer immer ansteht, kann der Takt mit der Steuerleitung "clock\_off", die wie "clock\_int" vom Steuerregister kommt, für die restlichen Systemkomponenten im Power-Down-Modus angehalten werden.

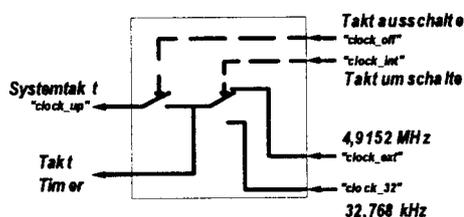


Abb.5: Über die Steuerleitung "clock\_int" kann zwischen zwei Takten umgeschaltet werden. Im Stromsparmodus wird der Systemtakt ausgeschaltet.

### Timer/Baudratengenerator

Daß hier zwei Baugruppen in einem Zuge genannt werden, hat den Hintergrund, daß Timer und Baudratengenerator aus einem einzigen Rückwärtszähler bestehen. Das Blockschaltbild zeigt Abb.6. Durch die Steuerleitung "timer" kann zwischen den beiden Komponenten umgeschaltet werden. Arbeitet das System mit 32,768 kHz - definiert durch den Taktumschalter - kann die Timerfunktion gewählt werden. Soll der Baudratengenerator verwendet werden, muß der Systemtakt auf 4,9152 MHz umgeschaltet werden.

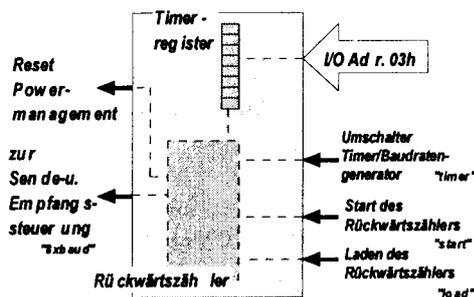


Abb.6: Der Timer/Baudratengenerator sind zwei Baugruppen in einer Einheit. Der Timer bestimmt die Länge eines Meßzyklus. Der Baudratengenerator liefert den Takt für den seriellen Empfang.

#### Timerfunktion

Über das Timerregister bestimmt sich die Länge eines Meßzyklus. Durch Beschreiben des Timerregisters mit einem bestimmten Wert kann die Zykluszeit eingestellt werden. Die Zeit ist einstellbar zwischen 20 Sekunden und einer Stunde. Bevor der Rückwärtszähler nach einer Konfigurationsphase wieder gestartet wird, muß der Rückwärtszähler zunächst mit dem Wert des Timerregisters geladen werden. Hierzu wird die Steuerleitung "load" kurz auf High und dann wieder auf Low gesetzt. Um den Rückwärtszähler zu starten, muß die Steuerleitung "start" gesetzt werden. Dann erfolgt eine Messung. Nach Abspeichern des Meßwertes wird das System in den "Sleep Modus" gebracht. Nach Ablauf der eingestellten Zeit, weckt der Timer das System wieder. Eine neue Messung erfolgt.

#### Baudratengeneratorfunktion

Für den Baudratengenerator reicht ein kleiner Teil des Rückwärtszählers aus. Der restliche Teil wird nicht betrieben. Durch seine Anordnung teilt der Zähler den Eingangstakt von 4,9152 MHz durch 64 und erzeugt so das Eingangssignal für die Sende- und Empfangssteuerung.

## Powermanagement

Das Powermanagement ist die Kombination von Timer und Taktabschaltung. Im Prinzip schaltet der Prozessor seinen eigenen Takt ab (bzw. hält ihn auf einem festen Potential). Alle Register bleiben wegen ihrer statischen Ausführung in ihrem Zustand, was auch für den Programmcounter gilt. Wäre nicht der Timer, würde dieser Zustand nun ewig dauern. Denn bevor der Prozessor seinen Takt anhält, wird der Timer gestartet. Dieser behält im Unterschied zum restlichen System seinen Takt und zählt seine eingestellte Zeit rückwärts. Beim Zählerstand "null" setzt er das für die Taktabschaltung verantwortliche Bit des Steuerregisters zurück, was den Takt wieder am Prozessor anlegt. Der Prozessor arbeitet nun an exakt der Stelle im Programm weiter, an der er stehen geblieben ist. Da nur sehr wenige Funktionsbaugruppen im „Sleep-Mode“ aktiv sind, ist die Stromaufnahme sehr klein. Eine relativ kompakte Lithiumbatterie ist damit in der Lage, den Logger für Wochen oder sogar Monate zu versorgen.

## Serielle Schnittstelle

Über die serielle Schnittstelle erfolgt die Kommunikation mit dem PC. Anhand Abb.7 soll die Funktion erklärt werden.

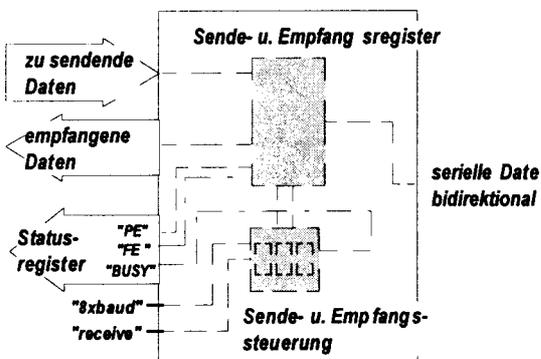


Abb.7: Über das Sende- u. Empfangsregister kann der Thermologger mit dem PC kommunizieren.

Die serielle Schnittstelle besteht im wesentlichen aus dem Sende- und Empfangsregister und der Sende- und Empfangssteuerung. Beim Sende- und Empfangsregister handelt es sich um ein Schieberegister, mit dem gesendet und empfangen werden kann. Der Takt der zum Schieben notwendig ist, wird von der Sende- und Empfangssteuerung erzeugt. Hierbei handelt es sich um eine Kombination von drei State-Maschinen, auf die hier jedoch nicht im Detail eingegangen werden soll. Eingangssignale für die Sende- und Empfangssteuerung ist das

Signal "8x baud", das vom Baudratengenerator kommt und das Signal "receive" vom Steuerregister. Für den korrekten Betrieb des Sende- und Empfangsregisters ist es erforderlich, daß der Baudratengenerator in Betrieb ist.

### Senden

Um Senden zu können, muß "receive" zurückgesetzt sein. Schreibt man nun an das Senderegister seine zu sendenden Daten, werden automatisch Startbit, Stoppbit und Paritybit zum Datenbyte hinzugefügt. Die Sende- und Empfangssteuerung erzeugt elf Schiebepulse im Abstand 1/9600 sec. Um feststellen zu können, wann der Sendevorgang beendet ist, muß der Prozessor das Statusbyte "busy" solange überprüfen, bis es wieder Low ist.

### Empfangen

Für den Empfang muß die Steuerleitung "receive" gesetzt sein. Mit dem achtfachen Baudratentakt wird die Datenleitung abgetastet. Wird ein Startbit erkannt (Wechsel der Datenleitung in den Zustand Low), wartet die Sende- und Empfangssteuerung einen halben Baudratentakt und liefert elf Schiebepulse. Der serielle Datenstrom wird immer in Bitmitte abgetastet. Nach elf Takten steht das Datenwort komplett im Schieberegister. Über eine Logik wird geprüft, ob Start- und Stoppbit korrekt sind, was im Fehlerfall durch das Statusbit "FE" (Frame Error) angezeigt wird. Aus den acht Datenbits wird automatisch ein Paritybit generiert und mit dem gesendeten verglichen. Bei Ungleichheit wird das Statusbit "PE" (Parity Error) gesetzt. Programmtechnisch muß beim Empfang wie folgt vorgegangen werden:

Wird in den Empfangsmodus umgeschaltet muß das Statusbit "BUSY" eingelesen werden, bis dieses High wird. Das ist das Zeichen, daß ein Startbit entdeckt wurde und der Empfang beginnt. Anschließend muß wie beim Senden der Lowzustand des "BUSY"-Bits abgewartet werden. Ist dieser Zustand erreicht, steht das Datenbit im Empfangsregister und kann ausgelesen werden. Erst jetzt stehen auch die Statusbit "FE" und "PE" stabil.

## SW-Konzept Thermologger

Daß die Software in einen ROM-Teil und einen RAM-Teil aufgesplittet ist, wurde zu Beginn schon erwähnt. In Abb.8 sind die zwei Speicher dargestellt. Die zwei Software-Teile sind hierbei als *Interrupt-Service-Routinen* im ROM und als *Hauptprogramm* im RAM bezeichnet. Im RAM ist auch der Stack und die Interrupt-Tabelle angeordnet. Weiterhin ist hier auch der Meßwertspeicher angesiedelt, der jedoch mit seiner Größe

von ca. 7k in der Darstellung relativ zu klein gezeichnet ist. Da der Inhalt des RAM bei jedem Ausschalten der Betriebsspannung verlorengeht, muß dieser Programmteil über die serielle Schnittstelle heruntergeladen werden. Das Hauptprogramm kann so je nach Bedarf an die Anwendung angepaßt werden. Man denke an nachträgliche Änderungen der Software oder an den Einsatz anderer Sensoren auf der Chipkarte. Nach jedem Reset wird eine Interrupt-Tabelle an die unterste Stelle des RAMs geschrieben. Die Betriebssystemroutinen werden durch Softwareinterrupts aufgerufen.

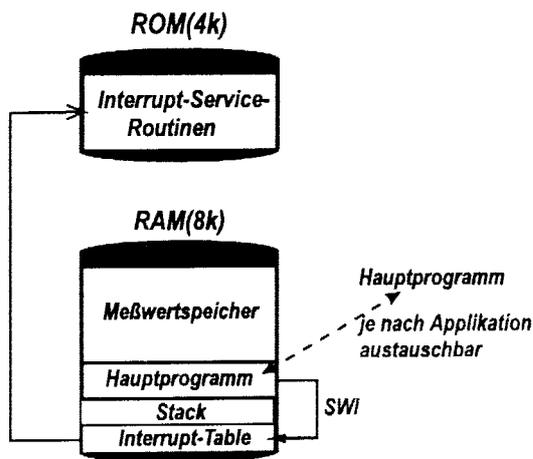


Abb.8: Das System besitzt einen vier kilobyte großen ROM-Speicher und einen acht kilobyte großen RAM-Speicher

## Layout Thermologger

Abb.9 zeigt das Layout der gerouteten Zelle. Die

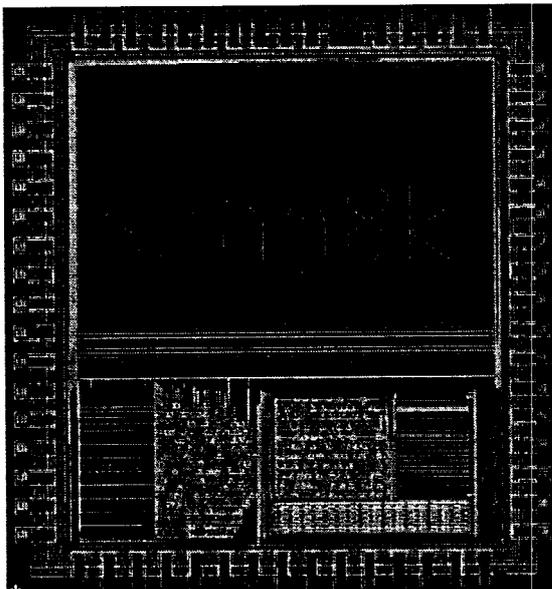


Abb.9: Die geroutete Zelle.

obere Hälfte der Zelle wird durch den RAM-Speicher belegt. Die dunkle Fläche links unten zeigt das ROM. Rechts ist der FHOP-Kern zu erkennen. Die übrigen Baugruppen sind dazwischen in Standardzellenentwurf angeordnet. Die Fläche beträgt 35 mm<sup>2</sup>.

## Stand der Arbeit und Ausblick

Alle ASICs sind getestet und weitgehend voll funktionfähig. In einer weiteren Diplomarbeit soll nun die Entwicklung des Betriebssystems erfolgen. In der nächsten Version des Thermologger-ASICs, wird das entwickelte Betriebssystem im internen ROM integriert. Die Zahl der Pads werden auf ein Minimum reduziert.

Der damit noch einmal deutlich verkleinerte Chip wird direkt auf ein Folienträger der Chipkarte in MCM-Technik gebondet.

## Zusammenfassung

Funktionsmerkmale:

- serielle Schnittstelle
- Schnittstelle zum Temperatursensor
- Powermanagement
- konfigurierbarer Timer
- Erfassung von 40.000 Meßwerten
- Meßgenauigkeit +/- 1°C
- Taktumschaltung
- Download des Hauptprogramms über die serielle Schnittstelle in den RAM-Speicher

## Literaturverzeichnis

- [1] Klöser, Frank:  
*Entwurf eines Ausgabemoduls für den Mikroprozessorkernel FHOP mit VHDL und Integration aller FHOP-Komponenten in einen Test-ASIC; Diplomarbeit, Fachhochschule Offenburg, 1994*
- [2] Jansen, D., Gieringer, T., Zimpfer, F.:  
*A Microprocessor in 4 months; IEEE International ASIC Conference and Exhibit, Rochester USA, 1994*
- [3] ISO/IEC 7816-3



## Grafische Eingabe von Zustandsautomaten in PSPICE

Vortrag beim MPC - Workshop  
4. / 5. Juli 1996 in Heilbronn

Prof. Dr. Gert Voland  
FB Informatik  
FH Konstanz

Tel: 07533 / 98 36 31  
Fax: 07533 / 98 36 13  
voland@fh-konstanz.de

### (1) Überblick

#### Ziele:

- Leichte grafische Eingabe von Zustandsautomaten in Mealy-, Moore- oder gemischter Form in einer bekannten grafischen Umgebung.
- PSPICE bietet sich an, da praktisch bei allen Studenten im Einsatz.
- Die grafische Eingabe soll als didaktische Hilfe zum Verständnis bei der Einführung von Zustandsautomaten dienen.

- Leichter Übergang auf ein Optimierungsprogramm zur automatischen Realisierung der Logik.

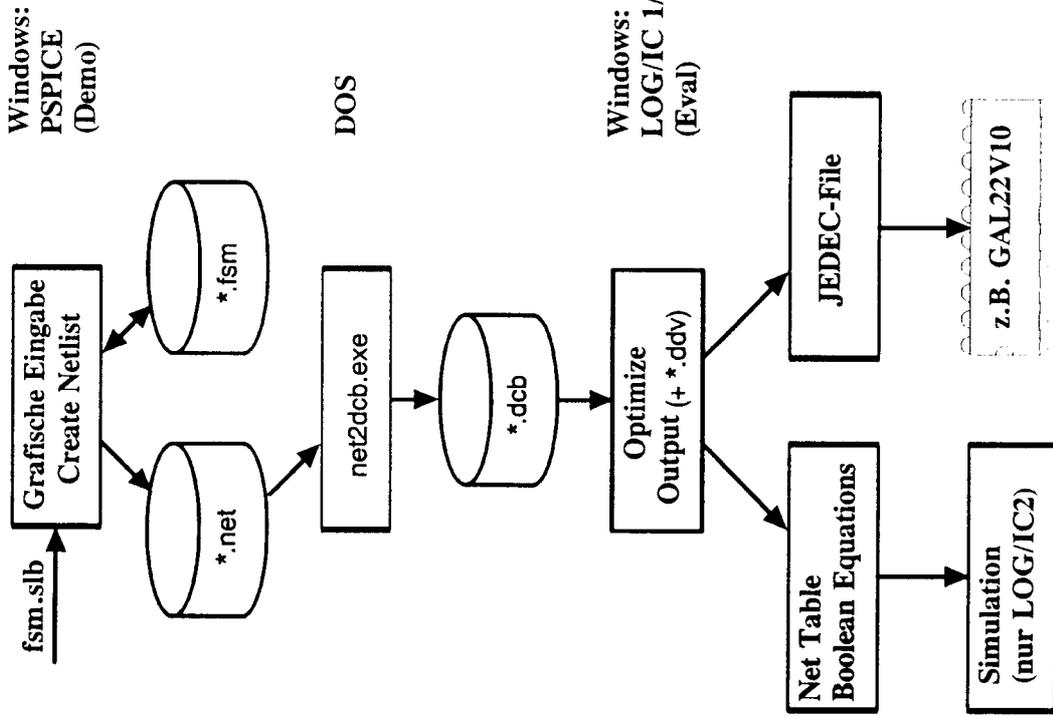
Bisher ist die Schnittstelle zu LOG/IC realisiert, da das Programm in der Lehre eingesetzt wird.

Die "Syntax- und Semantikhürde" der textuellen Eingabe soll überwunden werden.

#### Methodik:

- Durch Platzieren von FSM-Symbolen (fsm.slb) in PSPICE wird ein "Schematic" gezeichnet (\*.fsm zur Unterscheidung von \*.sch).
- Erzeugung einer "Netzliste" mit dem Standard-PSPICE-Mechanismus Create Netlist.  
Eine Parameterbibliothek ist nicht nötig.
- Automatische Übersetzung in das für die jeweilige Form syntaktisch und semantisch richtige Format der Steuerdatei (\*.dcb) des Optimierungsprogramms LOG/IC.

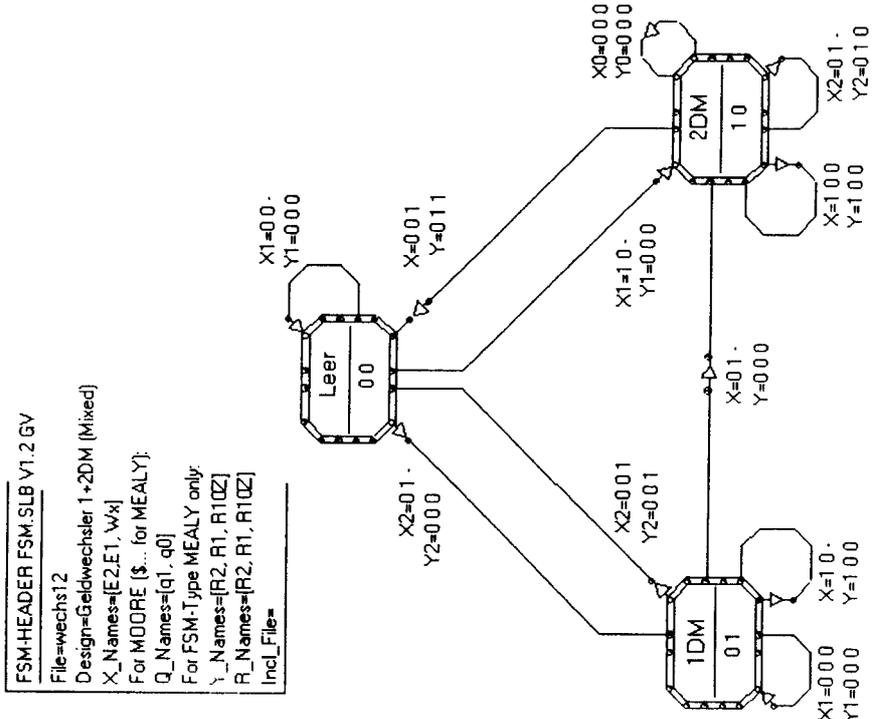
(2) Designablauf



(3) Beispiel

- Schaltung für einen Geldwechsler entwerfen, der eine 1 DM Münze in 10 Zehner wechselt.
- 2 DM werden in 1 DM und 10 Zehner gewechselt.
- Bei Überzahlen werden die Münzen zurückgegeben.

Eingänge: Einwurf 1 oder 2 DM = E1, E2; Wechseln = Wx  
Ausgänge: Rückgabe 1, 2 DM oder 10 Zehner = R1, R2, R10Z



**(4) Erzeugung der Dateien**

In PSPICE wird mit Create Netlist die Datei wechs12.net erzeugt:

```
* Schematics Netlist *
[ fsm-header ] FSM.SLB V1.2 GV
* Identification
File: wechs12 Design: Geldwechsler 1+2DM (Mixed)
*X-Names [E2,E1, Wx]
*Y-Names [R2, R1, R10Z]
*Q-Names [q1, q0]
*R-Names [R2, R1, R10Z]

[ FSM-BUBBLE ] 2DM
*Q-Code
1 0
*Transitions
$N_0001
$N_0002
$N_0003
$N_0004
$N_0005
$N_0006
$N_0007
$N_out22DM
$N_0008
$N_0009
$N_0010
$N_out02DM
$N_in02DM
$N_0011
$N_0012
$N_in12DM

[ FSM-TRANS ] $N_out02DM -> $N_in02DM
*X-Inputs
0 0 0
*Y-Outputs
0 0 0
```

... etc ...

Der Übersetzer net2dcb.exe generiert dann die Datei wechs12.dcb für LOG/IC2:

```
* IDENTIFICATION
FILE: WECHS12 DESIGN: GELDWECHSLER 1+2DM (MIXED)
NETLIST CONVERTER: NET2DCB.EXE V2.2 SYMBOL LIBRARY:
FSM.SLB V1.2 GV
CONVERSION DATE : 06-12-1996 ;

* INTERFACE
IN: [E2,E1, WX] ;
OUT: [R2, R1, R10Z] ;

! FSM Type: MIXED MEALY + MOORE FSM (Outputs at Transitions
and at States)

! DCB- <= User-States
!-----
! S 0 <= LEER
! S 1 <= 1DM
! S 2 <= 2DM

* FLOW-TABLE
$HEADER : X [E2,E1, WX] : Y [R2, R1, R10Z] ;
S 2 : X 1 0 0 : Y 1 0 0 : F 2 ;
: X 0 1 - : Y 0 1 0 : F 2 ;
: X 0 0 0 : Y 0 0 0 : F 2 ;
S 1 : X 0 0 1 : Y 0 1 1 : F 0 ;
: X 0 1 - : Y 0 0 0 : F 2 ;
: X 1 0 - : Y 1 0 0 : F 1 ;
: X 0 0 0 : Y 0 0 0 : F 1 ;
S 0 : X 0 0 - : Y 0 0 0 : F 0 ;
: X 1 0 - : Y 0 0 0 : F 2 ;
: X 0 1 - : Y 0 0 0 : F 1 ;

$HEADER : Q [Q1, Q0] ;
S 0 : Q 0 0 ;
S 1 : Q 0 1 ;
S 2 : Q 1 0 ;

*SPECIAL-FUNCTIONS
R2.REG = DFL ;
R1.REG = DFL ;
R10Z.REG = DFL ;

*END
```

(5) Ergebnis der Optimierung mit LOG/IC (1 = Classic)

Design: wechs12  
14-Jun-96 09:53:35 Date:  
FILE: WECHS12 DESIGN: GELDWECHSLER 1+2DM (MIXED)  
NETLIST CONVERTER: NET2DCB.EXE V2.2 SYMBOL LIBRARY:  
FSM.SLB V1.2 GV  
CONVERSION DATE : 06-14-1996 ;

\*\*\*\*\*  
\*\*\* NET DESCRIPTION TABLE FOR AND/OR STRUCTURE \*\*\*  
\*\*\*\*\*

```

:      RQ Q
:      11 0
EEWQ Q : RR0 .
21X1 0 : 21ZD D
-----
INV  . . . . ;
REG DDDD D ;
1--1 - : A . . . ; 1
1--1 - : A . . . A ; 2
-1-1 - : .A.A . ; 3
--11 - : .A . . ; 4
-011 - : .A . . ; 5
001-1 : .A . . ; 6
-1--1 : . . .A ; 7
1---0 : . . .A ; 8
--01 - : . . .A ; 9
-00-1 : . . .A ; 10
-1-0 0 : . . .A ; 11
1234 5 1234 5

```

27 X\_literals, 13 Y\_literals, 11 P\_terms

MESSAGES: 0 ERROR(S), 3 WARNING(S), 0 INFO(S)

LOG/IC - PLD (PLD) CPU TIME USED: 2 SEC

(6) Einstellungen für Mealy- oder Moore Typen

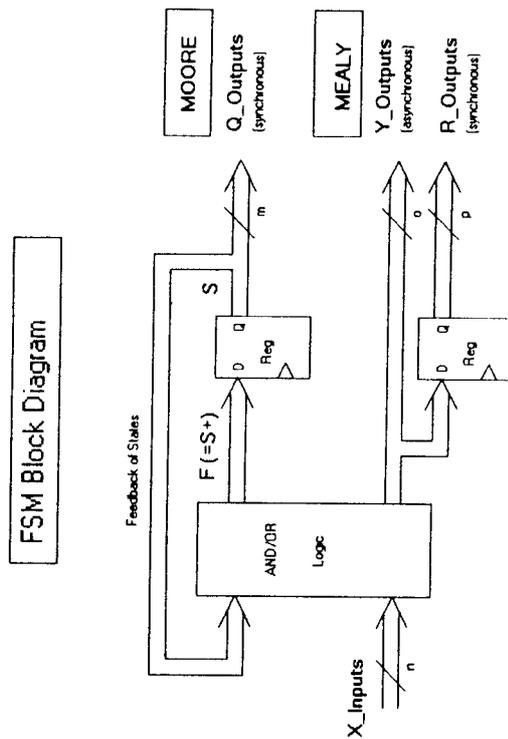
Die Steuerung, ob ein Mealy-, Moore- oder gemischter Automat erzeugt wird, geschieht über die Deklaration der Y\_Outputs und die Q\_Codierung der Zustände:

How to code different FSMs:

	Q_Names: Yes	Y_Names: -none-
MOORE	Q_Codes: Yes	Y_Outputs: -none-
MEALY	Q_Names: only \$...	Y_Names: Yes
MOORE and MEALY Mixed	Q_Codes: -none-	Y_Outputs: Yes
	Q_Names: Yes	Y_Names: Yes
	Q_Codes: Yes	Y_Outputs: Yes

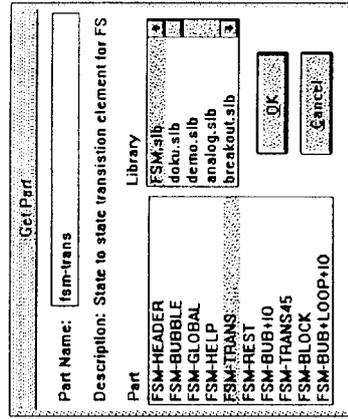
X\_Names and X\_Inputs always have to be coded.

Die resultierende Schaltung hat dann immer folgende Struktur, je nach Deklaration der Ausgänge:



Y\_Names  $\neq$  Q\_Names  
Y\_Names  $\geq$  R\_Names

(7) Inhalt der Symbolbibliothek fsm.slb:

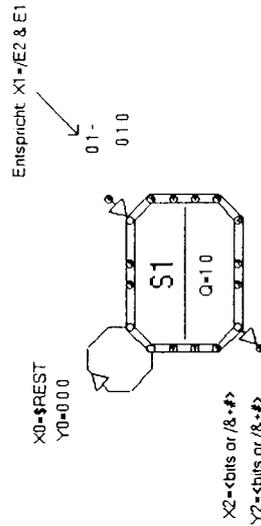


- **FSM-Header:**  
Symbol, das die Deklaration der Namensgruppen beinhaltet.  
Im Prinzip gilt hier die LOG/IC-Syntax. Z.B.: [...] erlaubt Bus-Eingaben.
- **FSM-Bubble:**  
Symbol eines Zustandes (Knoten) mit 16 Anschlüssen. Jeder Anschluß kann entweder Eingang oder Ausgang sein.  
Die Formatierung der \*.net-Datei geschieht mit speziellen Template-Attributen. Beispiel für FSM-Bubble:  

```
\n[PART] @REFDES ?Q|\n*Q-Code \n @Q| \n*nTransitions \n #i01 \n #i02 \n #i03 \n #i04 \n #i05 \n #i06 \n #i07 \n #i08 \n #i09 \n #i10 \n #i11 \n #i12 \n #i13 \n #i14 \n #i15 \n #i16
```
- **FSM-Trans:**  
Symbol eines Überganges (Kante). Sie sind gerichtet und verbinden 2 PSPICE-Netzknoten, die mit Label benannt werden können. Template-Attribut von FSM-Trans:  

```
\n[PART] #in -> #out \n*x-Inputs \n @X ?Y|\n*Y-Outputs \n @Y|
```
- **FSM-Global und FSM-Rest:**  
Zwei Symbole, die Global- und Rest-Übergänge erzeugen.

- Weitere Symbole sind Zusammenfassungen von Knoten und Kanten, um die 25-Teile-Grenze der Demoversion von PSPICE besser ausnutzen zu können. Beispiel: **FSM-Bub+Loop+IO**



Entsprechend der LOG/IC-Syntax können auch Formeln für die Übergangsbedingungen (X\_Inputs) bzw. die Ausgänge (Y\_Outputs) angegeben werden.  
Die Namen (und Werte) der Übergänge und der Codierung können natürlich auch mit Change Display "unsichtbar" gemacht werden.

- Das Programm net2dcb.exe beinhaltet 25 Syntax- und Semantik-Checks.  
Ein möglicherweise im Header angegebenes Include-File wird direkt 1:1 eingefügt.

(8) Gültige Versionen (14.06.96)

fsm.slb	V1.2	51 kB	getestet mit PSPICE 6.0 Demo- und Vollversion und 6.2 Demo.
net2dcb.exe	V2.2	33 kB	getestet mit LOG/IC 1 und 2 Eval, compiliert für ≥ 386.

