HELMHOLTZ ZENTRUM BERLIN
für Materialien und Energie

Bericht der Frühjahrstagung der

Studiengruppe für Elektronische Instrumentierung

7. bis 9. April 2008

Forschungszentrum Karlsruhe, Institut für Prozessdatenverarbeitung u. Elektronik

Herausgeber: Dr. F. Wulf

HMI-B 621 Berlin, Juni 2008

ISSN 0936 – 0891

HELMHOLTZ ZENTRUM BERLIN
für Materialien und Energie

Bericht der Frühjahrstagung der

Studiengruppe für Elektronische Instrumentierung

7. bis 9. April 2008

Forschungszentrum Karlsruhe,
Institut für Prozessdatenverarbeitung u. Elektronik

Herausgeber: Dr. Friedrich Wulf

HMI-B 621 Berlin, Juni 2008

ISSN 0936 – 0891

Helmholtz-Zentrum Berlin für Materialien und Energie GmbH
Abteilung Experimentsysteme
Glienicker Str. 100
14109 Berlin

HELMHOLTZ ZENTRUM BERLIN für Materialien und Energie

Teilnehmerliste

Name	Institut
Badura, Dr Eugen	Gesellschaft für Schwerionenforschung, Abt. EE
Balzer, Dipl.-Phys. Matthias	Forschungszentrum Karlsruhe, IPE
Beck, Dr. Dietrich	GSI Darmstadt
Bönisch, Dr. Sven	Helmholtz-Zentrum Berlin
Donner, Irene	Forschungszentrum Karlsruhe GmbH
Drochner, Dr. Matthias	Forschungszentrum Jülich GmbH, ZEL
Ehret, Dr. Günter	privat
Epstein, Dr. Avi	EMBL
Gemmeke, Prof. Dr. Hartmut	Forschungszentrum Karlsruhe IPE
Goettmann, Dipl.-Ing. Walter	Helmholtz-Zentrum Berlin
Hanßler, Dipl.-Ing. Bruno	Mentor Graphics (Deutschland) GmbH
Herbrand, Dr. Frank	Forschungszentrum Rossendorf e.V., Abt. FWFE
Hürttlen, Dipl.-Ing. Werner	Forschungszentrum Jülich GmbH, Abt. IBN-TAE
Kämmerling, Peter	Forschungszentrum Jülich, ZEL
Kirsch, Dr. Matthias	Struck Innovative Systeme GmbH
Kleifges, Dr. Matthias	Forschungszentrum Karlsruhe IPE
Kleines, Harald	Forschungszentrum Jülich
Klockmann, Kay	powerBridge Computer Vertriebs GmbH
Kreidl, Dipl.-Ing. (FH) Harald	Nu Horizons Electronics GmbH
Krömer, Dipl.-Ing. Oliver	Forschungszentrum Karlsruhe, IPE
Kühner, Thomas	Forschungszentrum Karlsruhe GmbH
Lenk, Dr. Stephan	SAAS Systemanalyse u. Automatisierungsservice GmbH
Mauro, Dr. Sergio	Wiener Plein & Baus GmbH
Mexner, Dr. Wolfgang	Forschungszentrum Karlsruhe, ISS (ANKA)
Müller, Dr. Klaus-Dieter	privat
Namaschk, Dr.-Ing. Bernhard	Helmholtz-Zentrum Berlin
Notz, Dr. Dieter	DESY Hamburg, F1
Petzold, Lars	Forschungszentrum Karlsruhe IPE
Podehl, Dipl. -Ing. Martin	MACCON GmbH
Rausch, Dipl. -Ing. Rainer	Forschungszentrum Jülich GmbH, Abt. IBN-TAE
Schmitz, Andreas	Tameq GmbH
Schneider, Dr. Marc	Forschungszentrum Karlsruhe GmbH
Siepermann, Peter	w3l GmbH
Spelthann, Hans-Dieter	Acqiris Data Conversion Instruments
Stotzka, Dr. Rainer	Forschungszentrum Karlsruhe, IPE
Sutter, Michael	Forschungszentrum Karlsruhe GmbH
von Düring, Dipl.-Ing. (Phys.) Arp-Nico	Caen GmbH
von Walter, Dipl.-Ing. Peter	Universität Heidelberg, Physikalisches Institut
Wulf, Dr. -Ing. Friedrich	Helmholtz-Zentrum Berlin
Wüstner, Dr. Peter	Forschungszentrum Jülich GmbH, ZEL
Zillikens, Hans-Peter	Forschungszentrum Jülich GmbH, Abt. IEF-2

III

Teilnehmer der SEI-Frühjahrstagung 2008, FZ Karlsruhe, IPE

Um sich die Namen der Teilnehmer anzeigen zu lassen, wandern Sie mit dem Mauszeiger über die Gesichter

SEI

HELMHOLTZ ZENTRUM BERLIN
für Materialien und Energie

HELMHOLTZ ZENTRUM BERLIN
für Materialien und Energie

Vorträge

	Seite
Zusammenfassung Dr. Friedrich Wulf, Helmholtz- Zentrum Berlin	- VIII -
Aufbau des Katrin-Experiments und seines elektronischen Detektors Lars Petzold, FZK-IPE	- 1 -
Eine Alternative zu einem Supercomputer Dr. Avi Epstein, EMBL Heidelberg	- 18 -
Radio-Detektionstechnik für höchstenergetische kosmische Strahlung Dr. Oliver Krömer, FZK-IPE	- 34 -
Virtuelle Infrastruktur für „Remote Instrumentation“ Michael Sutter, FZK-IPE	- 48 -
Single Source Publishing - Überblick und Konzepte Peter Siepermann, w3l GmbH	- 56 -
Prozessautomatisierung des Versuchstandes TOPFLOW im FZ Dresden-Rossendorf Dr. Stephan Lenk, SAAS GmbH	- 68 -
ALFA -Ein Luminositätszähler für Atlas Dr. Dieter Notz, DESY	- 77 -
Sind Simple µTCA-Systeme die besseren Industrie-PCs? Kay Klockmann, powerBridge Computer Vertriebs GmbH	- 82 -
FPGA Design mit Mentor Graphic Tools Bruno Hanßler, Mentor Graphics GmbH	- 91 -
Signalverarbeitung mit FPGA's: Möglichkeiten und Herausforderungen Harald Kreidl, Nu Horizons Electronics GmbH	- 190 -
High Speed Mixed-Signal ASIC Design Hassan Safdary, advICo microelectronics GmbH	- 227 -
Rauschen von resistiv gekoppelten Ladungsverstärkern Dr. Sven Bönisch, Helmholtz Zentrum Berlin	- 242 -
Strahlprofilmessung mit Lichtwellenleitersensorik am DESY FLASH LINAC W. Goettmann, Helmholtz-Zentrum Berlin, Dr. M. Körfer , DESY, Dr. F. Wulf, Helmholtz-Zentrum Berlin	- 252 -
Entwicklungen mit dem GPX TDC ASIC im FZ Jülich Harald Kleines, FZJ-ZEL	- 263 -
Parallelrechner auf Cell-Basis Dr. Matthias Drochner, FZJ-ZEL	- 273 -


Dr. Friedrich Wulf


Prof. Dr. Hartmut Gemmeke

Lars Petzold

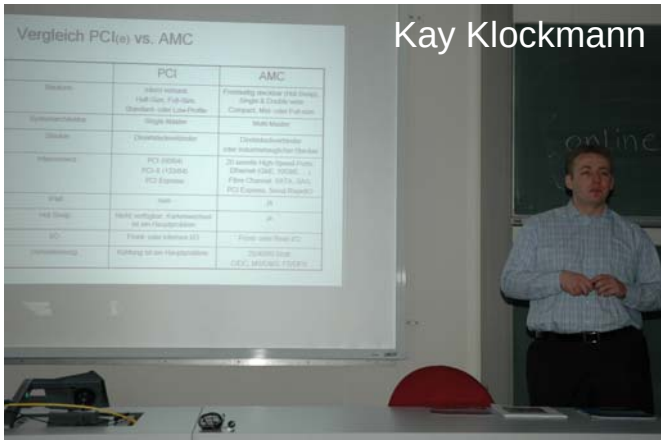
Dr. Avi Epstein

Dr. Oliver Krömer

Michael Sutter

Peter Siepermann

Dr. Stephan Lenk

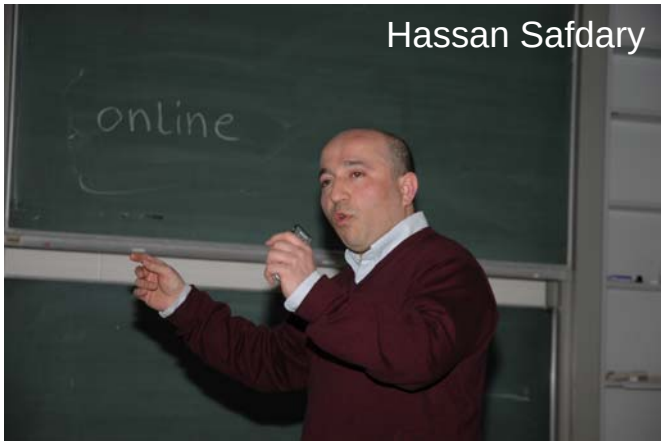

Vergleich PCI(e) vs. AMC
PCI
AMC
Kay Klockmann


uick and Easy
RTL
Bruno Hanßler

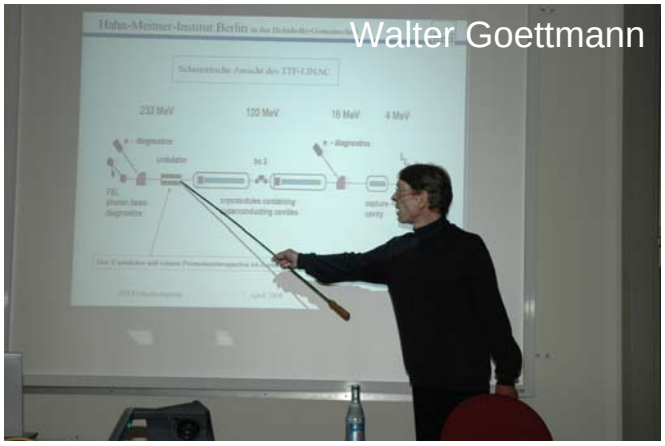

Signalverarbeitung mit FPGA's:
Möglichkeiten und Herausforderungen
Harald Kreidl


MACON
COLLIMATORS
Firmenausstellung

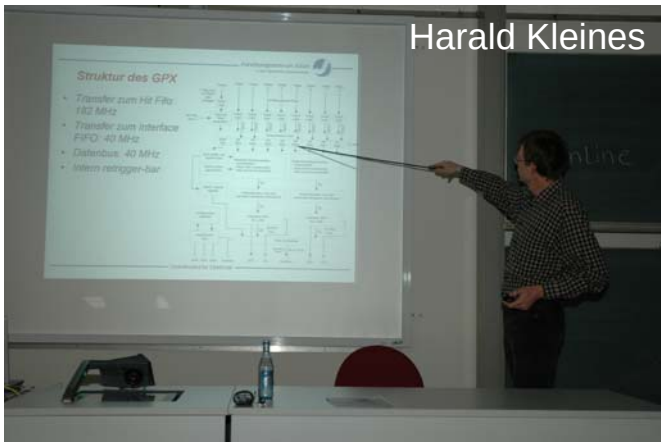

Firmenausstellung


online
Hassan Safdary

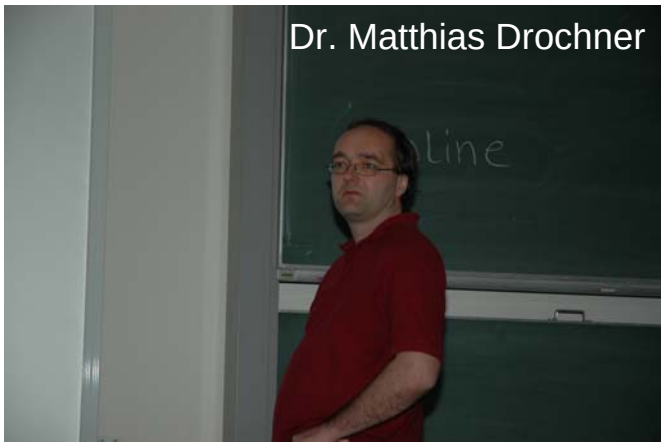

Dr. Sven Bönisch



Walter Goettmann



Harald Kleines


Dr. Matthias Drochner

HELMHOLTZ ZENTRUM BERLIN für Materialien und Energie

Zusammenfassung

Dr.-Ing. F. Wulf, Helmholtz-Zentrum Berlin

Die 96. Tagung der **S**tudiengruppe für **e**lektronische **I**nstrumentierung (SEI-Frühjahrstagung 2008) fand vom 7. bis 9. April 2008 im **I**nstitut für **P**rozessdatenverarbeitung und **E**lektronik (IPE) des Forschungszentrums Karlsruhe statt. Der Tagungsband enthält 15 Beiträge aus den unterschiedlichen Bereichen der Informationstechnik. Ich danke allen Vortragenden und Autoren für die sehr interessanten Beiträge. Mein besonderer Dank gilt Herrn Prof. Dr. H. Gemmeke und Frau Dipl.-Ing. S. Bohrmann für die sehr gute Organisation und interessante Gestaltung des Rahmenprogramms. Die Beteiligung mit 41 Personen aus 19 Forschungseinrichtungen und Industriefirmen führte wieder zu einer intensiven Diskussion über den speziellen Einsatz der Informationstechnik für die Steuerung und Datenerfassung komplexer Experimentsysteme.

In drei Vorträgen wurden die derzeitigen Forschungsschwerpunkte des IPE dargestellt. Der erste Vortrag zeigt die aktuelle Entwicklung der Detektorelektronik für das KATRIN[1] (**KA**rlsruher **TRI**tium **N**eutrino Experiment) Projekt, bei der das Rauschen und die Energieauflösung eine zentrale Rolle spielen. Das Konzept für die Datenerfassung und speziellen Verfahren für die Selbsttriggerung des LOPES^{STAR} [2]Experiments wurde in einem weiteren Vortrag dargestellt. Der steigende Bedarf, Experimente mit einer komplexen IT-Struktur auch über große Entfernungen remote zu steuern, steigt ständig. Eine virtuelle Infrastruktur, die für den Test und die Verifikation sehr nützlich ist, wurde in einem weiteren Beitrag vorgestellt.

Sehr interessant ist die Verwendung von Systolic Arrays (Parallel Architektur) für String Matching Algorithmen. Verglichen mit aktuellen CPUs (von Neumann Architektur) besitzen Systolic Arrays erheblich höhere Rechengeschwindigkeiten und damit sehr viel geringe Durchlaufzeiten. Wie in einem Vortrag vom EMBL gezeigt werden konnte, sind bei der Verwendung heutiger Technologien bis zu 10^{18} OPS erreichbar. Der leistungsfähigste Rechner der Welt[3] ermöglicht z. Z. ca. $5*10^{14}$ FLOPS. Die Möglichkeiten für den Einsatz von Parallelrechnern auf der Basis von Cell Prozessoren[4] zur Berechnung spezieller mathematischer Verfahren wurde in einem Vortrag vom FZJ erläutert.

Aus dem Bereich der Hochenergiephysik wurde der Luminositätsdetektor ALFA (Absolute luminosity-measurement for ATLAS[5]) vorgestellt, der im Experiment ATLAS beim LHC[6] die Rate der elastisch gestreuten Protonen misst, wodurch die Luminosität absolut bestimmt werden kann.

Von der Firma powerBridge Computer wurde das von ihr entwickelte Simple MicroTCA[7] System vorgestellt. Es definiert einen reduzierten und eindeutigen Funktionsumfang für ein MicroTCA-System mit dem Ziel, MicroTCA[8] durch Vereinfachung und Kostenreduzierung für den Industrieeinsatz nutzbar zu machen. Die Kosten eines Simple MicroTCA Basissystems betragen nur etwa ein Drittel der Kosten eines Standard-Systems. Dieses System bietet eine Alternative zu dem cPCI[9]- und PXI[10]-Systemen und ist bei DESY für die Instrumentierung bei XFEL[11] in Verbindung mit MicroTCA geplant.

Detektorelektronik für die speziellen Anforderungen der Neutronstreuexperimente ist nur mit eingeschränkten Funktionen oder gar nicht kommerziell erhältlich. In einem Vortrag wird das Rauschverhalten resistiv gekoppelter Ladungsverstärker für positionsempfindliche Zählrohre auf der Basis von Widerstandsdrähten detailliert erläutert, während ein weiterer Vortrag sich mit der Entwicklungen eines GPX TDC ASIC für Delayline Detektoren beschäftigt.

Seit 2000 wird mit Hilfe von speziell entwickelten Lichtwellenleiter und der dafür entwickelten DAQ-Systeme die Ortsdosis an den Undulatoren und die Strahlverluste (als Maß für die Strahlfokussierung) entlang der gesamten Beschleunigerstrecke des Elektronenbeschleunigers FLASH[12] bestimmt. Das Messsystem zur Überwachung der Strahlqualität kann nun - wie in einem Vortrag gezeigt - auch für die Strahlprofilmessung genutzt werden. Das Strahlstromüberwachungssystem wird zukünftig auch für den XFEL eingesetzt.

Zukünftige Hardwareentwicklungen werden immer stärker FPGAs als Bindeglied zwischen der analogen Frontend-Elektronik, der schnellen Datenvorverarbeitung und den unterschiedlichen Bus-Systemen einsetzen. Für den optimalen Einsatz der zahlreichen FPGAs oder auch Semi Custom ASICs ist der Einsatz herstellerunabhängiger Entwicklungswerkzeuge unabdingbar. In Vorträgen von Systemanbietern wurde über den aktuellen Stand von Design Werkzeugen berichtet. Durch die Verwendung von SystemVerilog[13] wird die Verbindung zwischen der HDL (Hardware Description Language) und HVL (Hardware Verification Language) herstellt und ermöglicht einen deutlich verbesserten Design Zyklus. Dadurch werden die re-spinning Kosten wie auch die Entwicklungszeit reduziert. Der Einsatz dieser Werkzeuge für die Forschungszentren wird nicht durch die Lizenzkosten bestimmt, sondern dadurch, dass die Einarbeitungszeit eine zuverlässige Personalplanung voraussetzt, die in den Zentren z. Z. nicht gegeben ist. Die Entwicklung von FPGA-Designs wird auf der SEI-Herbsttagung 2008 weiter vertieft werden.
Elf Firmen aus den verschiedenen Bereichen der elektronischen Instrumentierung und IT-Branche präsentierten ihre neuen Produkte und standen für eine intensive Diskussion und Beratung zur Verfügung.

Die SEI-Herbsttagung 2008 ist vom 22. bis 24. September 2008 am Max-Planck-Institut für Plasmaphysik, IPP, in Greifswald geplant. Die SEI-Frühjahrstagung 2009 findet voraussichtlich vom 23. bis 25. März 2009 im Zentralinstitut für Elektronik (ZEL) des Forschungszentrums Jülich statt.

Berlin, Mai 2008

---

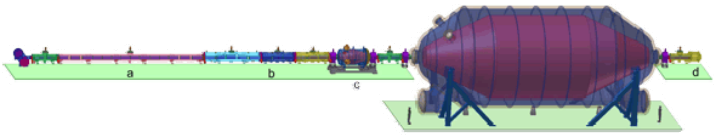
[1] Beim KATRIN-Experiment soll die Masse des Elektron-Antineutrinos durch hochgenaue Energiebestimmung von Tritium-Zerfallselektronen (γ-Strahlung) eingegrenzt werden.
[2] LOPES^{STAR} ist ein vollständig kalibriertes und unabhängiges Detektorsystem zur Radioobservation kosmischer Luftschauer oberhalb einer Primärenergie von 5 10^{17} eV. LOPES^{STAR} ist die konsequente Weiterentwicklung des LOPES-Experimentes (LOfar PrototypE Station) mit dem der prinzipielle Nachweis der Radioemission aus Luftschauern mit externer Triggerung gelang. Die Abkürzung STAR steht für „Self-triggered Array of Radio detectors".
[3] BlueGene von IBM im Lawrence Livermore National Laboratory JUGENE, (Jülich Blue Gene) im Forschungszentrum Jülich, mit 1,7*10^{14} FLOPS
[4] Cell-Prozessor wurde für die the PlayStation 3 von Sony, Toshiba and IBM's entwickelt.
[5] ATLAS: (**A** **T**oroidal **L**hc **A**pparatu**S**) Atlas soll hochenergetische Proton-Proton-Kollisionen untersuchen und dem Higgs-Teilchen auf die Spur kommen; evtl. Teilchennachweis aus Supersymmetriemodellen
[6] LHC: (**L**arge **H**adron **C**ollider) Teilchenbeschleunigern für Protonen bis zu einer Energie von 14 TeV.
[7] Simple MicroTCA beschreibt Gehäusesysteme mit Steckplätzen für AMCModule, die ohne MCH und PM auskommen. Ersatz der notwendigen MCH- (**M**icroTCA **C**arrier **H**ub) und PM-Funktionen (**P**ower **M**odule). durch ein Simple MicroTCA Support Modul (SSM)
[8] MicroTCA (auch: µTCA oder mTCA) steht für **M**icro **T**elecommunications **C**omputing **A**rchitecture und ist ein von der PICMG verabschiedeter modularer Standard, der den Aufbau von Baugruppenträgern und kompletten Grundsystemen regelt.
[9] IEC-1076 international standard.
[10] PXI (PCI eXtensions for Instrumentation)
[11] XFEL: (X-ray free-electron laser) Construction is set to begin in summer 2008, and commissioning will start at the end of 2013.
[12] FLASH: (**F**reie-Elektronen-**LAS**er in **H**amburg)
[13] SystemVerilog ist ein seit 2005 akzeptierter IEEE Standard 1800-2005 und eine Erweiterung von Verilog.

KATRIN - Experiment

- Status -

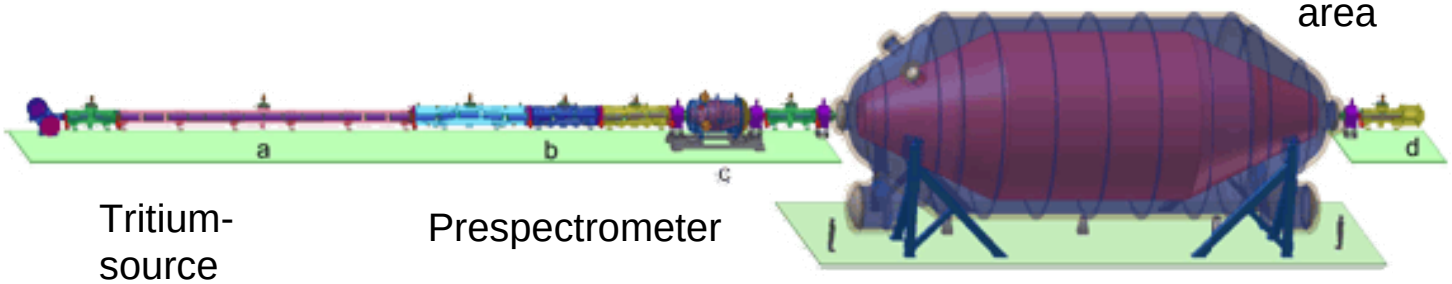
Detection of electrons

Lars Petzold, IPE



Tritium source and main tank

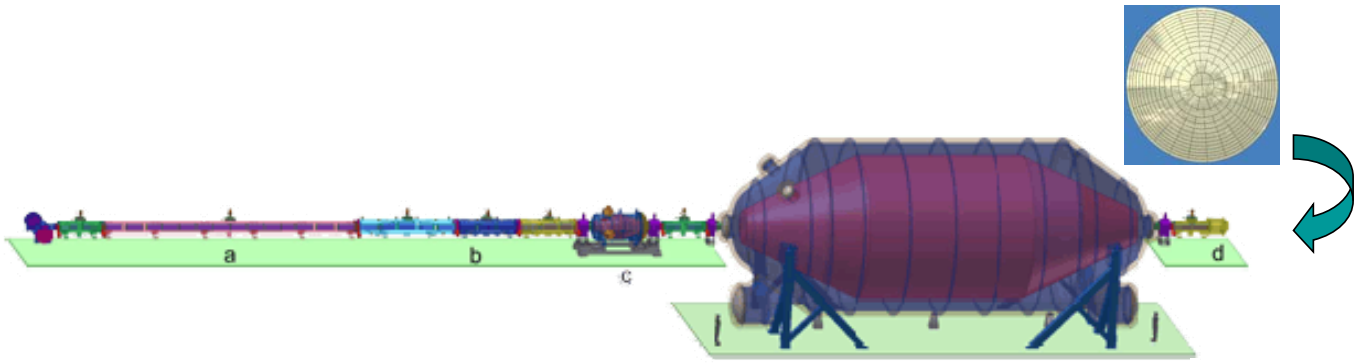
Determine mass of electron-antineutrino with an error of 0.2 eV



Motivation of KATRIN

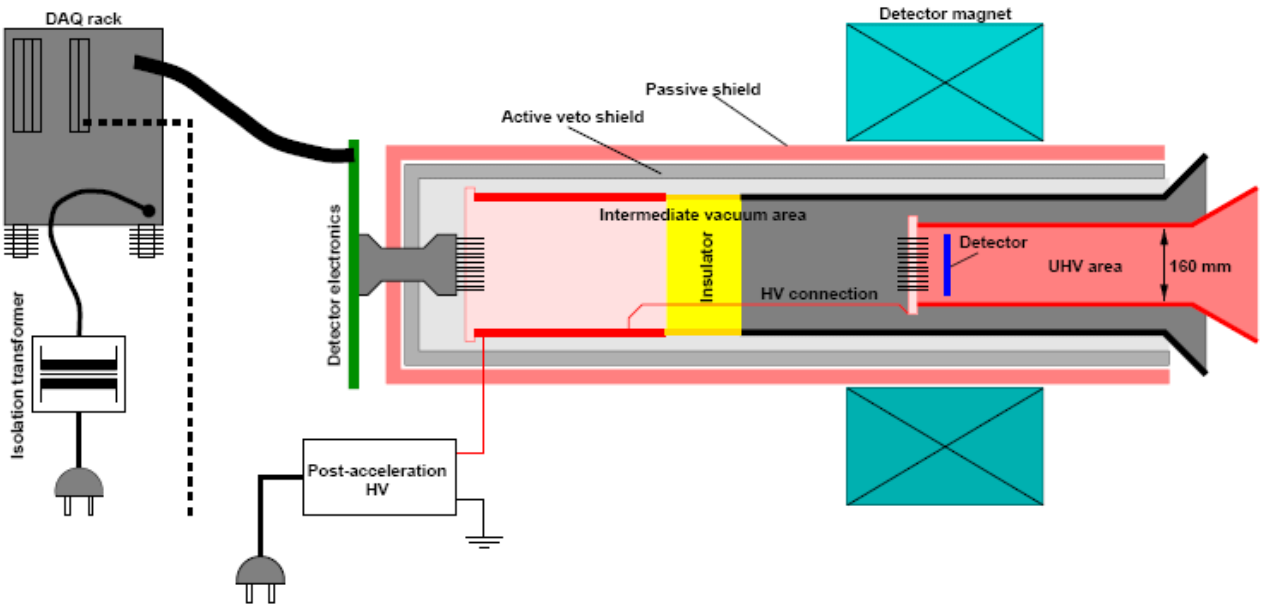
Measurement of electrons with resolution of approx.600eV

- Detector counts the electrons
- Resolution 0.6 …1 keV for background suppression
- Segmented detector

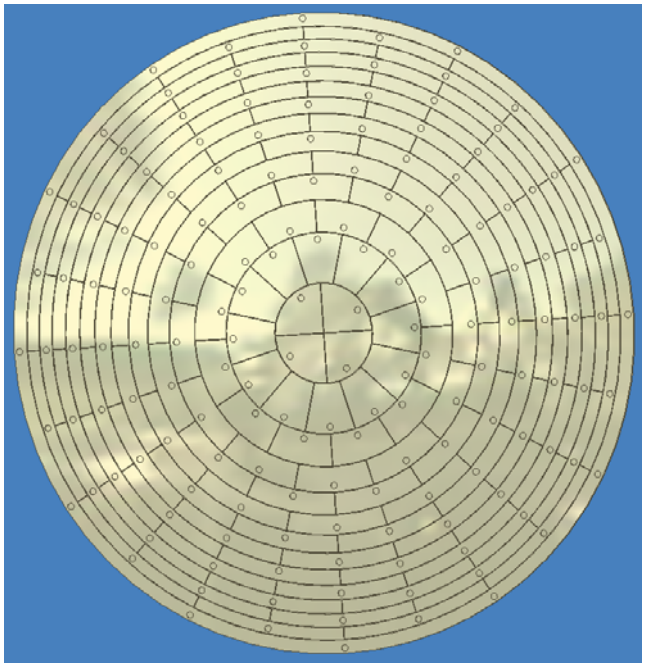


Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

Main detector area

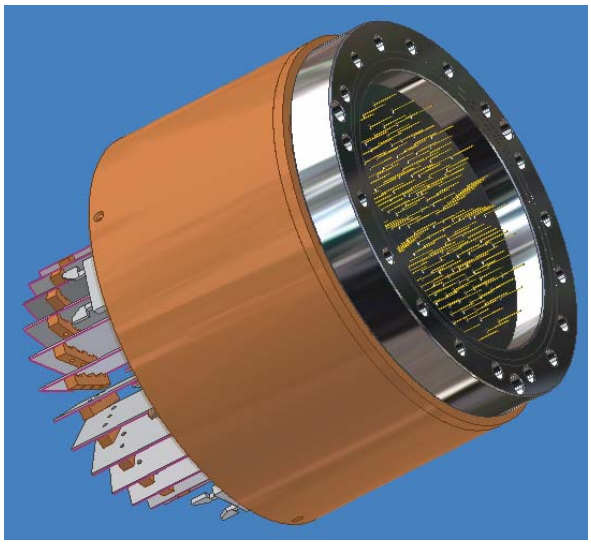


Segmented detector



- 90mm diameter
- 12 pF per pixel
- 148 pixel
- each pixel with same active area
- pixelized mid
- shielded with guard-ring
- charge at 18.6 keV: approx. 0.8 fC

Mechanics



- Specs
- Materials
- CAD-design

Mechanics

Specs:

- Suitable for high vacuum HV (10^{-9} mBar)
- Suitable for low temperatures (210K and less)
- Low activities (α, β, γ) within mHz

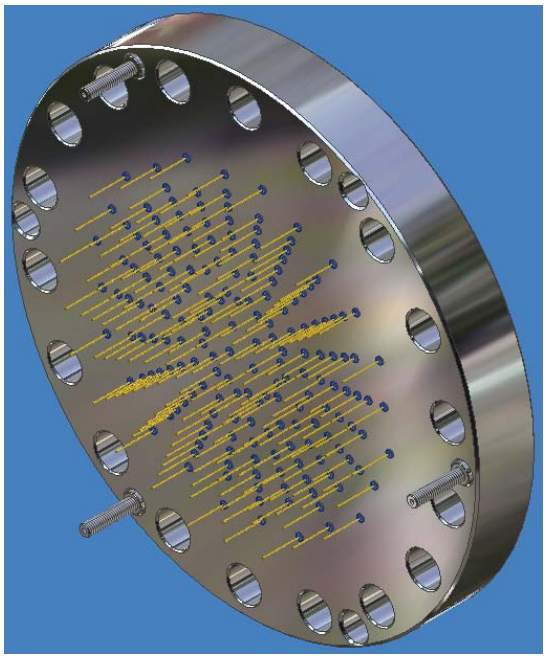
Mechanics

Materials:

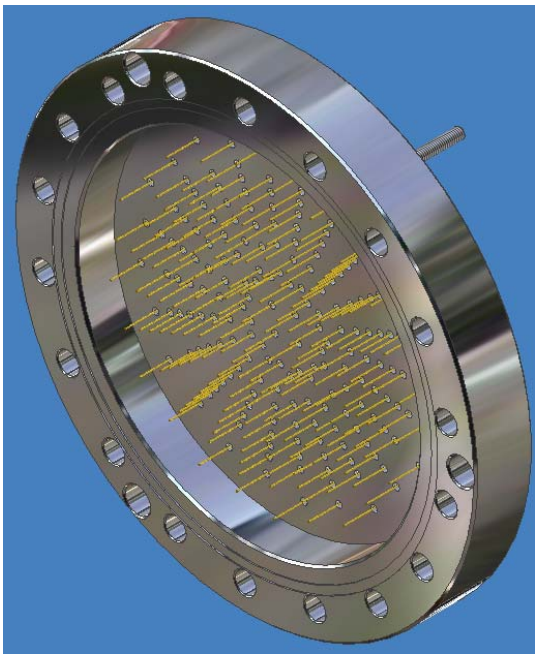
- Preamp boards (RUBALIT 710 S, Ceramtec)
  - Less pollution with Uranium, Thorium and Potassium
  - GEANT simulation (Michelle Leber, UW) estimates count-rate of 80 µHz
- Feed through flange (stainless steel)
- Detector pins (Niobium)
- Front-plate, rods, back-ring, shielding-tube (copper)
  - Radiation shielding
- Remaining pcb-boards (REXOLITE)
  - Less background

Detector feed through

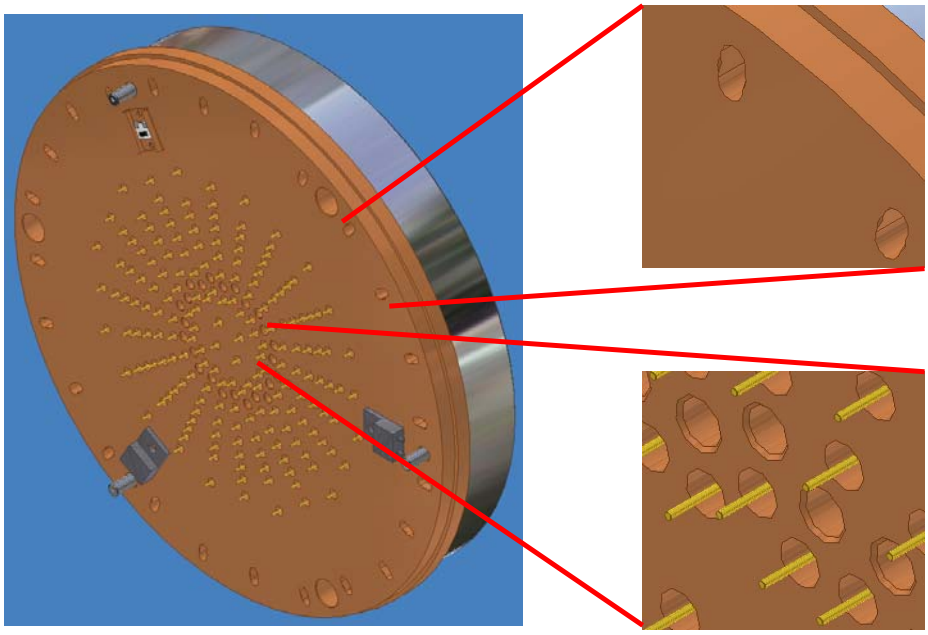
Electronic side



Detector side



Copper front plate



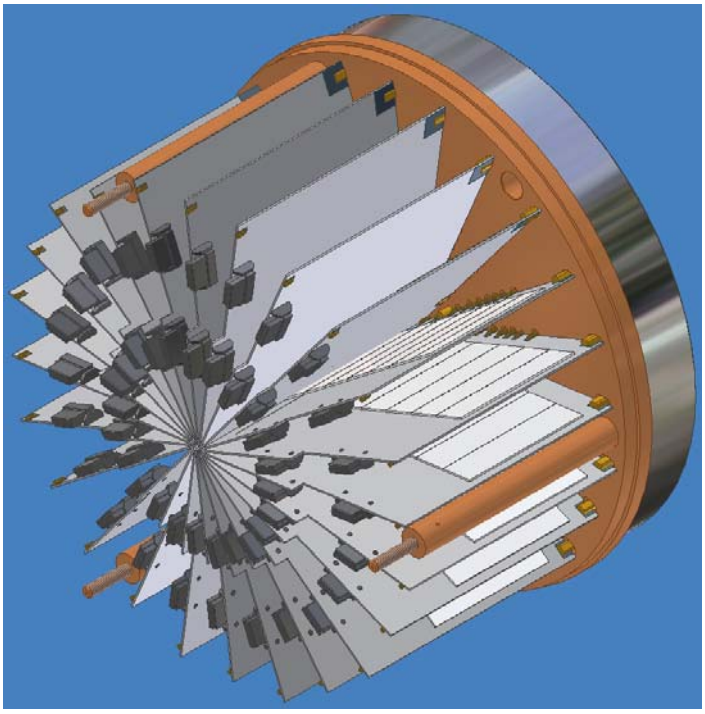
- Cooling and adjustment by using thick pins and drill holes
- Temperature measurement and heating added

Preamp modules

- Arrangement of the preamps
- Modules with 6 ch. (resp. 7 ch.)
- Two cooling pins
- SMC-plugs on bottom side

Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

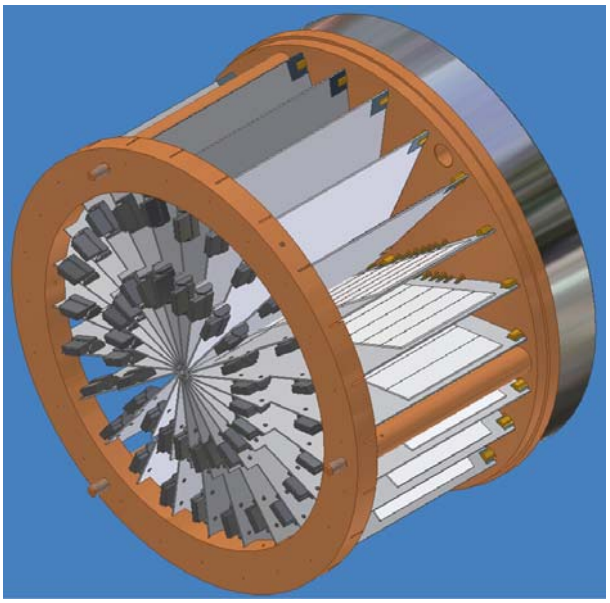
Preamp modules



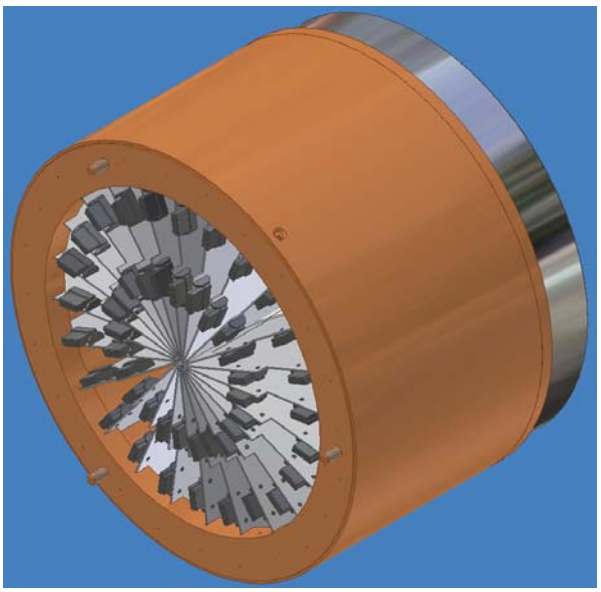
- Radial arrangement of the modules
- Modules are plugged directly onto the feed through pins
- Adjustment is provided by the cooling pins

Fixation and shielding

Fixation with a slotted back ring

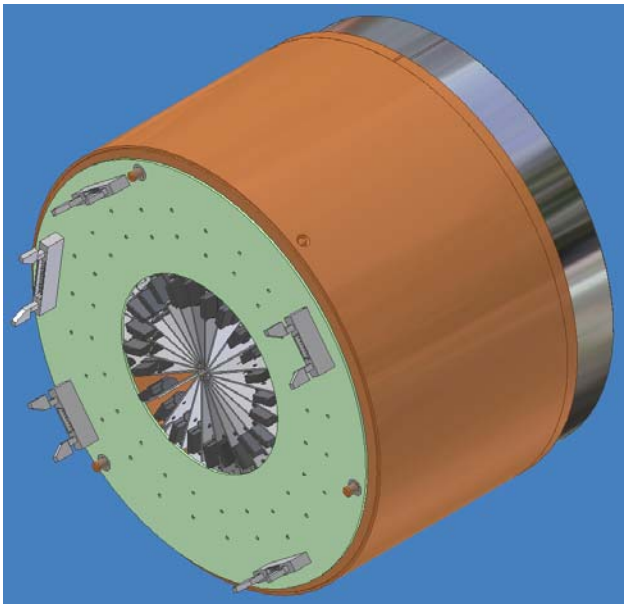


Shielding with a copper tube

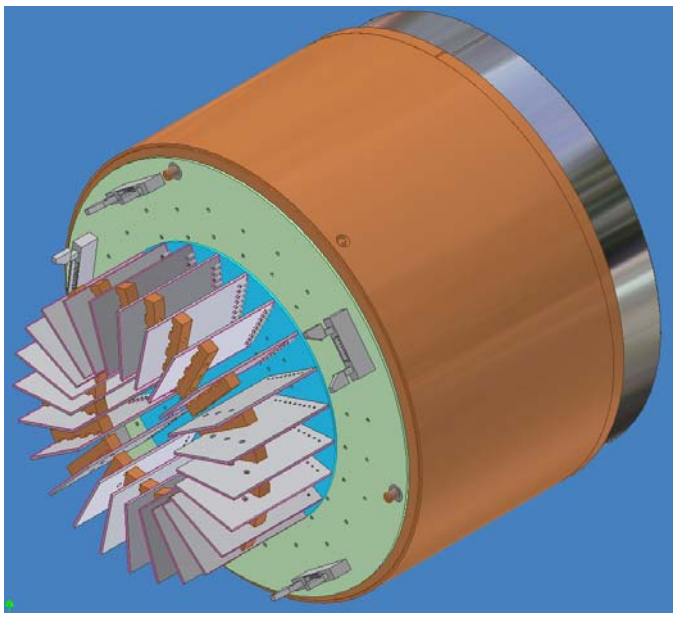


Wiring

Wiring of supplies with a back plane



Wiring of outputs with a cable board



Preamps

- Specs
- Technique
- Testsetups
- Electrical characteristics
- Tests

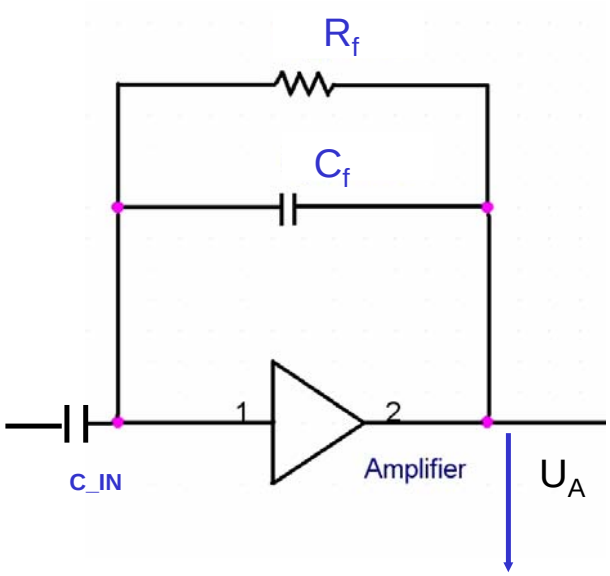
Preamps

Specs for KATRIN:

- Working within vacuum
- Temperature range of approx. 370°K to 210°K
- Influence of magnetic fields
- Low noise (500eV)
- Power consumption < 100 mW
- Bandwidth 5 MHz

KATRIN Preamp

$$U_A = -\frac{Q_{IN}}{C_{FEEDB}} e^{-t/\tau}$$



Charge sensitive preamp:

- Amplification depends on the feedback capacitor
- t means charge collection time ($t << \tau$)
- τ means the time constant of $C_f * R_f$

Evaluation of the KATRIN Preamp

Charge sensitive amplification

(C_f approx 0.75 pF)

$$G = 2mV \cdot \frac{1}{fC}$$

Discharge time constant

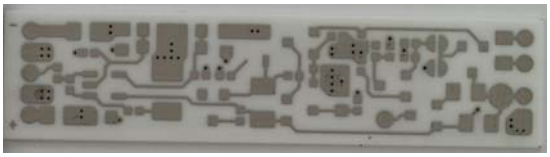
$$\tau = 0.75\,pF \cdot 1G\Omega = 0.75ms$$

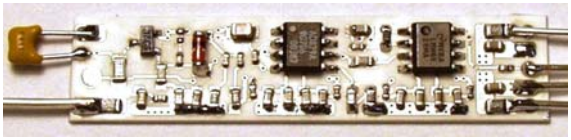
Lower Cutoff frequency

$$f_{LOW} \approx \frac{1}{2\Pi R_f C_f} \approx 220Hz$$

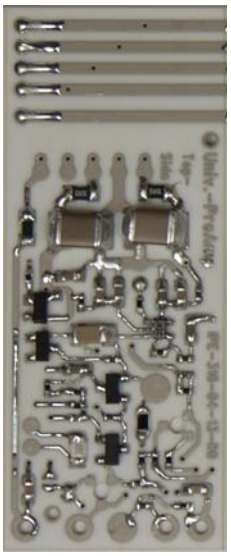
Testsetups single channel

Three different types of single amps





- Sizes of 11mm x 47 mm
- Fabricated on Rubalit 708 (96% Ceramic)
- With printed feedback capacitors



Sensitivity = dV/dC

Charge at 18.6 keV:

E: energy

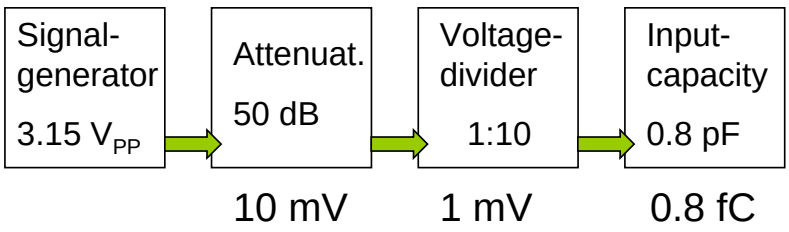
w_T: work to create one pair at 300K

e: elementary charge

$$N_{e^-/p^+} = \frac{E}{w_T} = \frac{18600eV}{3.62eV} \approx 5000e^- / p^+$$

$$Q = 5000 \cdot e = 0.8\,fC$$

Signal chain



Sensitivity

Measured amplitude of the output-peak: 7.5 mV

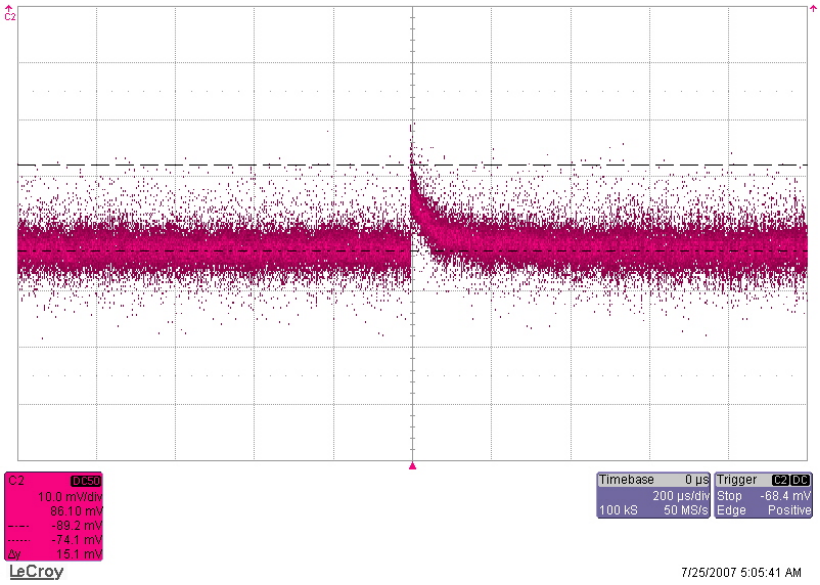
Amplification of input charge result in 1.5 mV.

(amplification second stage: 10 and 50 ohm termination)

Sensitivity: 1 fC cause approx. 2 mV

Or

Sensitivity: 1keV cause approx. 85 µV



Noise

- Function of
  - Temperature
  - Working point
  - used (active) parts
- Noise mainly depends on the first stage

2 Ways to determine:

Theoretical calculation

Measurement with oscilloscope

Noise

Calculation 300K:

- Noise first-stage: 46 nV/sqrt(Hz)

(FET 2.5 nV/sqrt(Hz), G1 =16)

(OPV AD829)

- Bandwidth 5 MHz: 10 µV/sqrt(Hz)
- Noise output: 1 mV_{Eff}; approx 6mV_{PP}

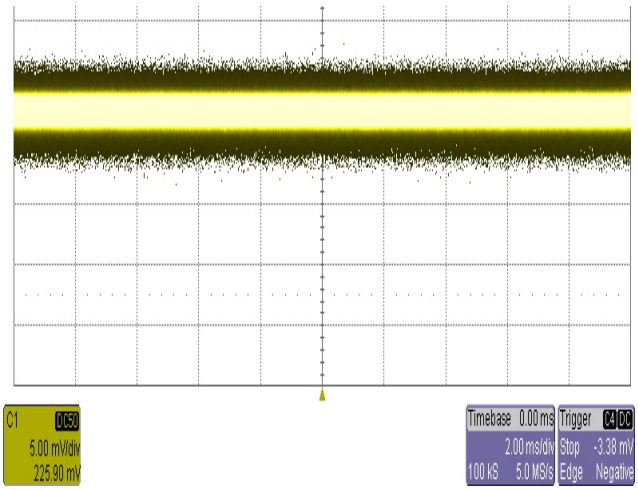
(OPV LMH6628, G2 = 10)

Ratio of 1.3 for benefit of calculation,

means 3nV/sqrt(Hz) noise at input-stage

Measured Noise:

- Approx 8 mV_{PP}



Resolution

The resolution defines an uncertainty of measured energies.

Several ways to determine:

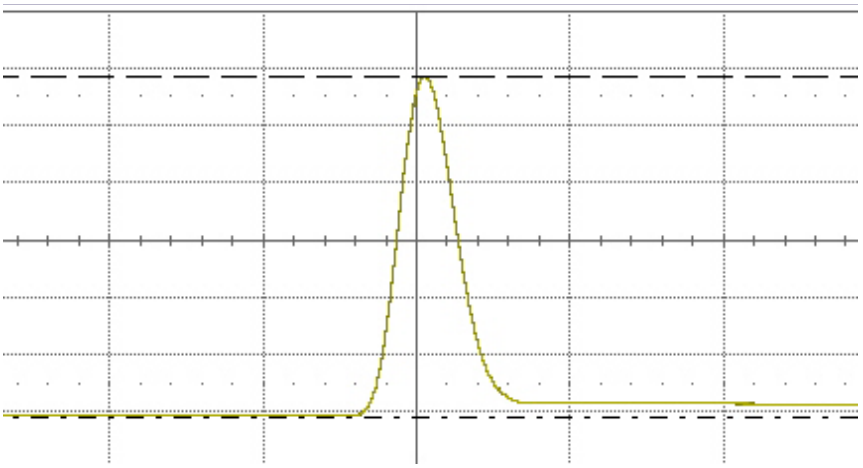
- Roughly with amplitude of certain energy and noise at multichannel analyzer output
- Calculation using shaping constants and type of filter (info: Radeka, Spieler)
- Measurement of the resolution using a shaping-amplifier and software

Resolution, using scope

Signal (A_S) = 123 mV

Noise (A_N) = 7 mV (Canberra Model 2025)

Rough estimation of Resolution (RES):



$$RES = \frac{A_N}{A_S} \cdot E$$

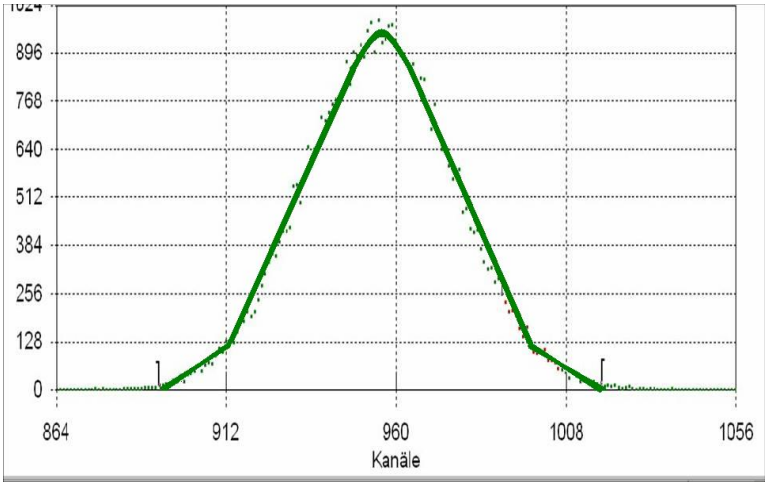
$$RES = \frac{7mV}{123mV} \cdot 18.6keV$$

$$RES = 1.06keV$$

Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

Resolution, multichannel analyzer

- The software GENIE2k uses an PEAK-ADC and logs the distribution.
- The resolution itself is defined as FWHM of the distribution-pulse.

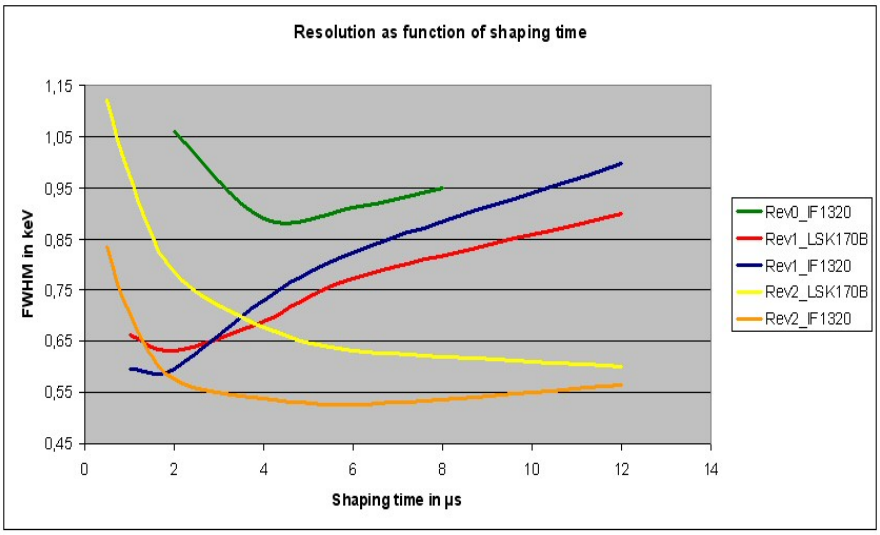


- Every channel corresponds to a certain energy
- The pulse is formed by counts per channel

Resolution, function of shaping time

Results: All revisions of single channel preamps at room temperature

Used input: testinput

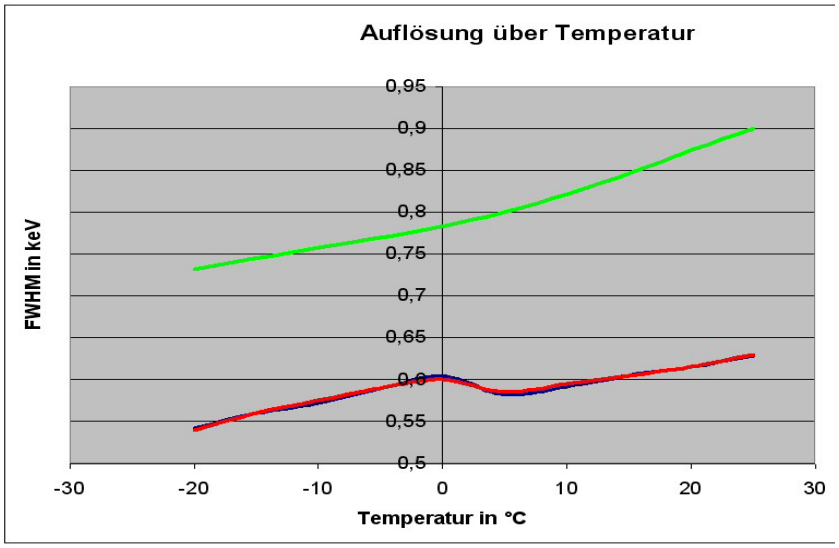


- Best results with revision2 at 6 µs (520 eV)
- shape of curve: dominating current noise changes to voltage noise
- New geometry or parts create new shape of curve

Resolution, function of temperature

Results: at fixed shaping time (6µs), revision0 (green) and revision1 (red)

Used input: testinput

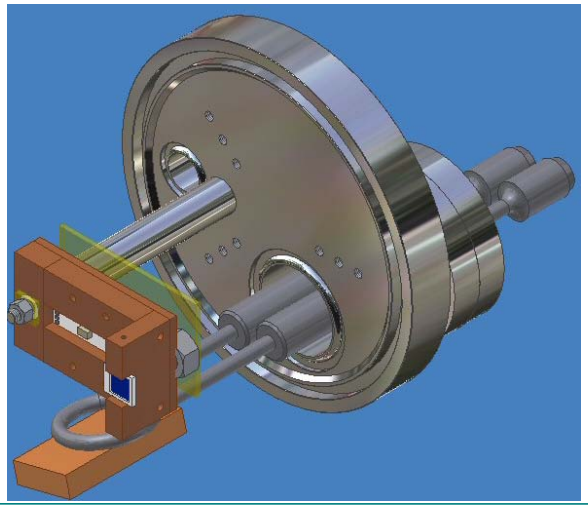


- Best results are with revision1 at -20°C (520 eV)
- At dT of 5°K resolution changes with 15eV
- The boss is a result of condensing effects

Resolution, e-Gun

Task: resolution of the revision2 using a detector and an e - Gun

Used input: Hamamatsu PIN diode



- Cooling with liquid N_2 downto 210K
- Electrons are created with an e - Gun, energies from 20keV up to 50 keV
- Required: resolution approx 500 eV at 210K

Conclusion / Outlook

Mechanics

- Mechanical design is finished
- Materials are chosen
- Fabrication has started

Preamps

- Preamps are well tested and characterized
- After successful finished e - Gun test fabrication will start

Thanks for your attention

Contents

- Motivation of KATRIN
- Simplified concept of KATRIN
- Mechanics
- Preamps
- Conclusion / Outlook

Preamps
Electrical characteristics

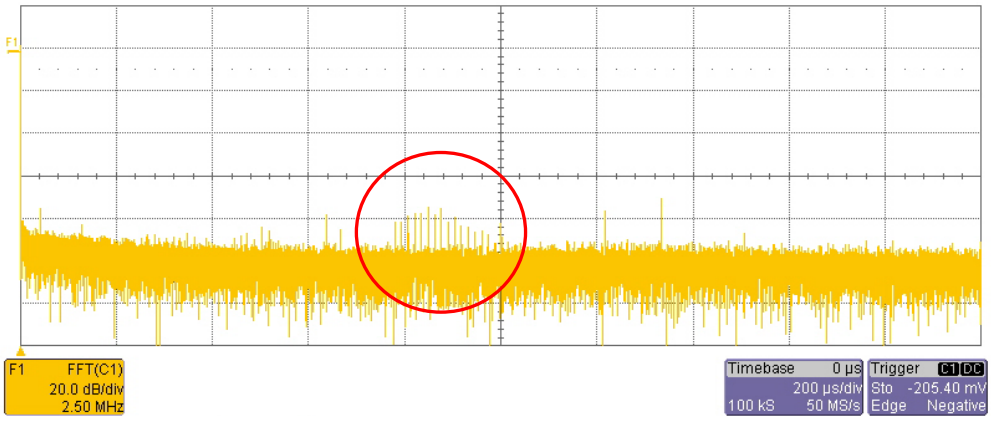
Main issues to measure:

- Amplitude results in the sensitivity
- Noise results in minimal detectable energy
- Noise power spectra gives information about cut-off frequencies and possible resonant circuits (bad compensation, ground layer …)
- Resolution results in an error of the reconstructed energy

More issues:

- Power consumption, cross talk, peripherals

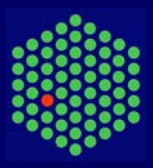
Preamps
Noise Spectra



- Preamps introduce distortions between 2 MHz and 2.5 MHz (center)
- This points to the OPAmps as source, points not to layout.
- Possibility to correct: compensation

Eine Alternative zu einem Supercomputer

Versatile Search Processor Array (VeSPA)



A. Epstein EMBL

Content

- Our Search applications
- Why systolic arrays?
- The VeSPA architecture
- Implementation
- Future Developments
- Performance / Conclusions

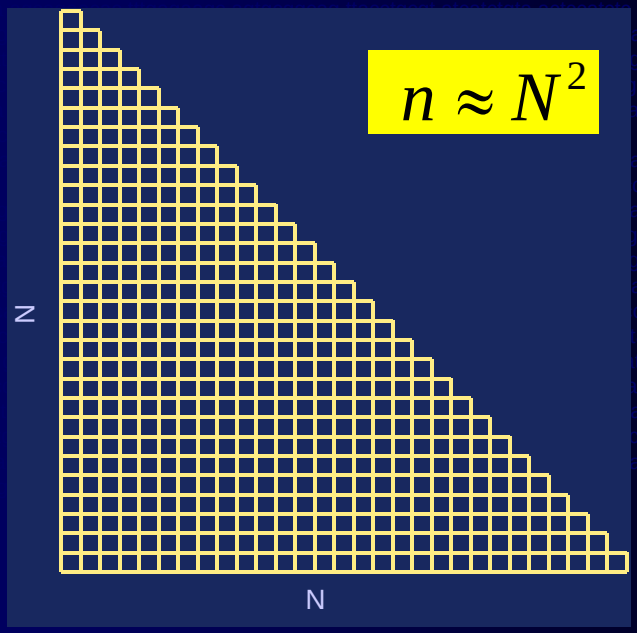
Biological Databases

Source: European Bioinformatics Institute (28.3.2008) http://www3.ebi.ac.uk/Services/DBStats/

http://www.ebi.ac.uk/Databases/

"Prozess, der durch Anwendung von Methoden auf einen Datenbestand Muster entdeckt"

Data Mining



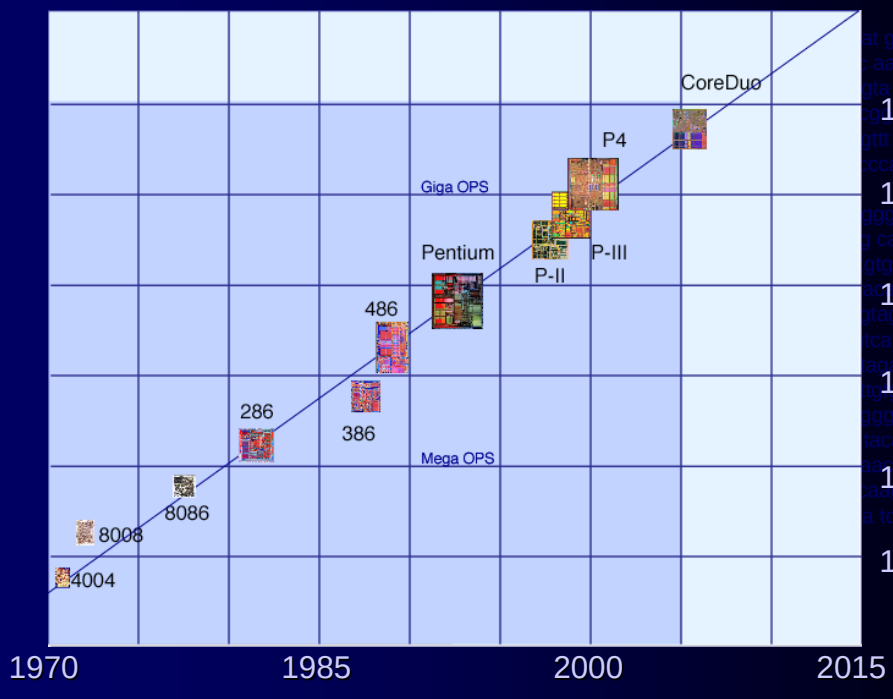
EMBL-Nucleotide Sequence Database

- 5.4.08: 199,792,611,692
- Growth (1982-2007):
  x 1.7 / year
- Required proc. Speed:
  x 3 / year

--> Moore's law does not suffice

QuickTime™ and a
TIFF (Uncompressed) decompressor
are needed to see this picture.

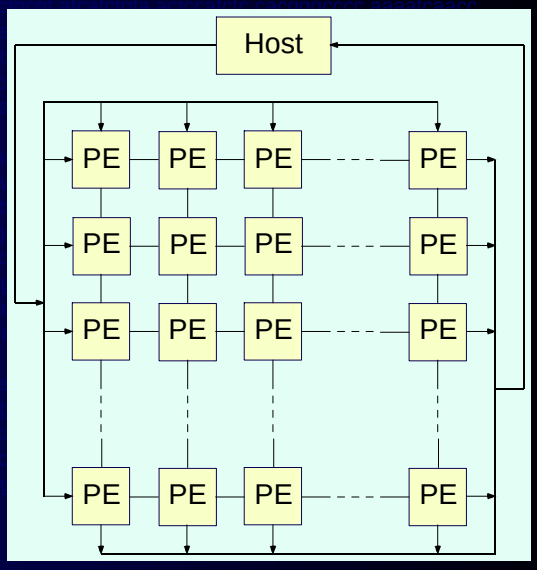
Moore's Law



Why Systolic Arrays ?

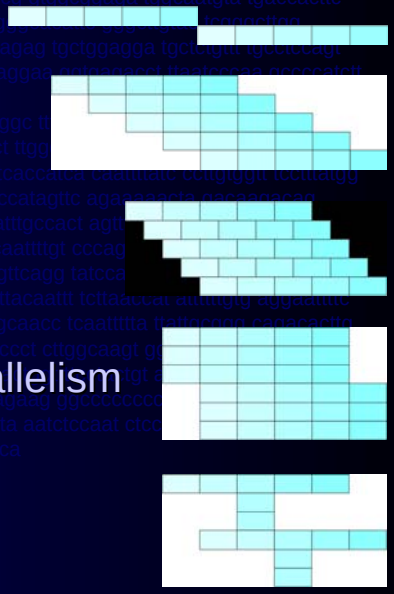
What are SA?

- Large number of PEs
- Single CLK operation
- Neighbor interconnections



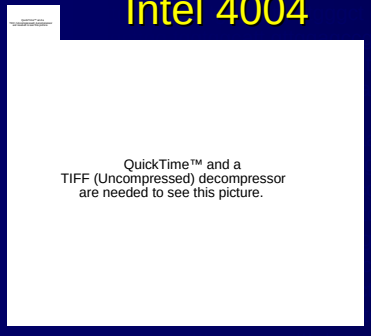
General Purpose Processor Evolution

- CISC multy cycle
- RISC more pipelining
- Superpipelined even more pipelining
- Superscalar more execution units
- VLIW explicit instruction parallelism



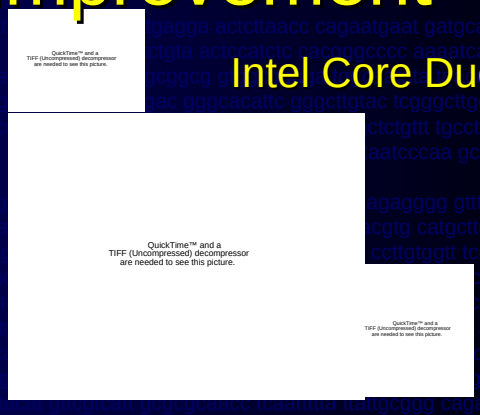
GP Processor Performance Improvement

Intel 4004



1971; 12mm²; 10µm pMOS; 2300 Trans.
- Clock: 0.108 MHz
- MIPS: 0.07
- Bus: 0.108 MHz

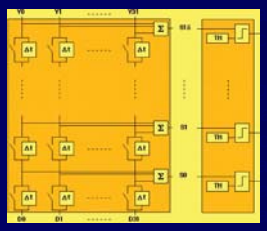
Intel Core Duo



2006; 90.3mm²; 65nm; 151.6 M Trans.
- Clock: 2.33 GHz
- MIPS: 20000
- Bus: 667 MHz

Systolic Array Performance Improvement

Hough Transform

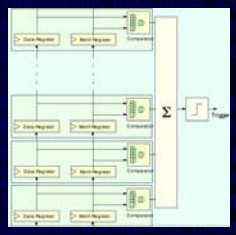


- 0.6µ CMOS
- 256 PE
- 100 MHz

x 4 PEs

x 2 Speed

VeSPA



- 0.35µ CMOS
- 1024 PE
- 200 MHz

x 8 Faster

Moore's Law

Processing speed (GP) doubles every 2 years

Systolic Array Performance Law

SA Processing speed doubles every year!

The VeSPA Systolic Array Architecture

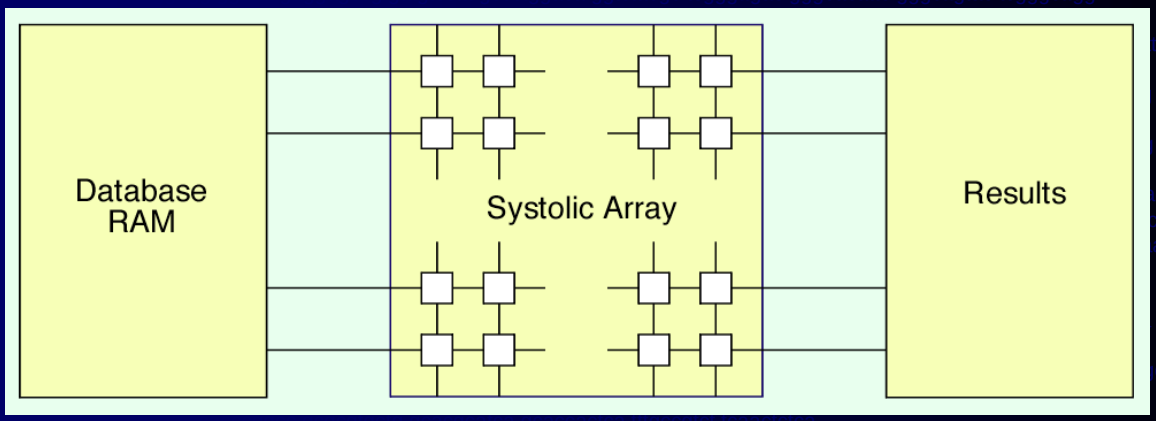
Complex Motif

	Char
1	T
2	S+P
3	E+M+B+L
4	Q..S
5	L+M+P+X
6	S

TSEQLS
TSEQMS
...
TPLSXS

96 valid motifs

Systolic Array Implementation

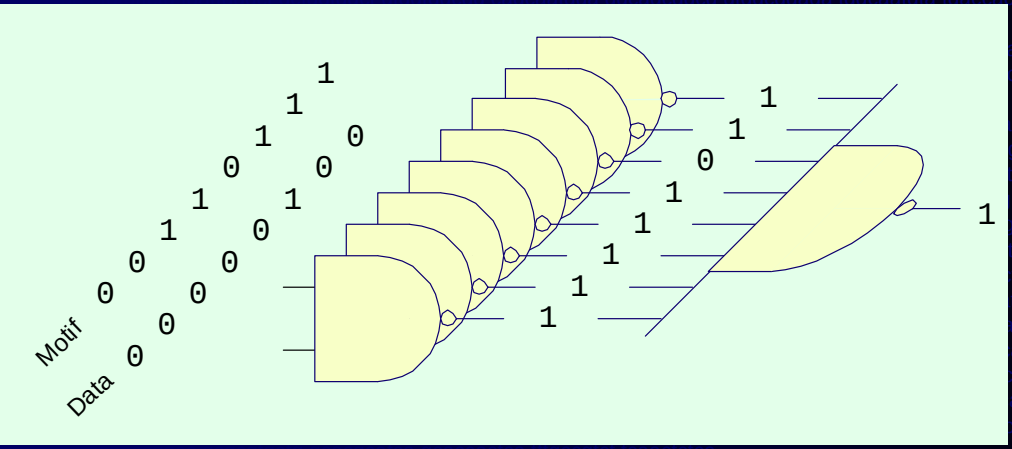


Character Coding

Character	Code
A	…0000000000000001
B	…0000000000000010
C	…0000000000000100

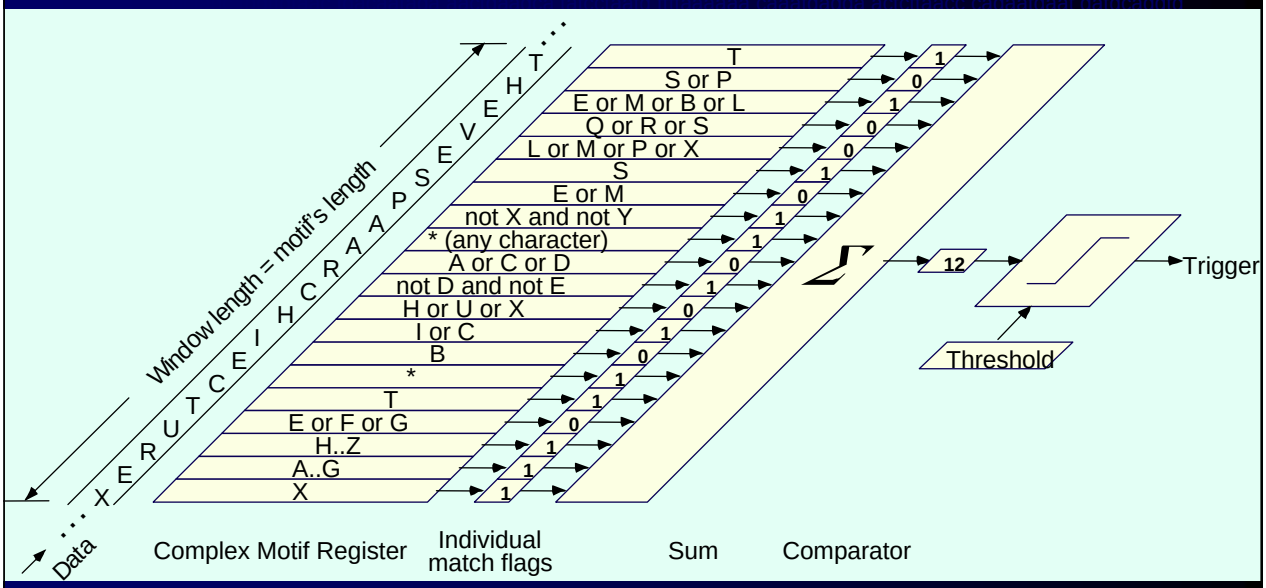
Motif	Code
A or C or E	…0000000000010101
*	…1111111111111111

Single Position Comparator

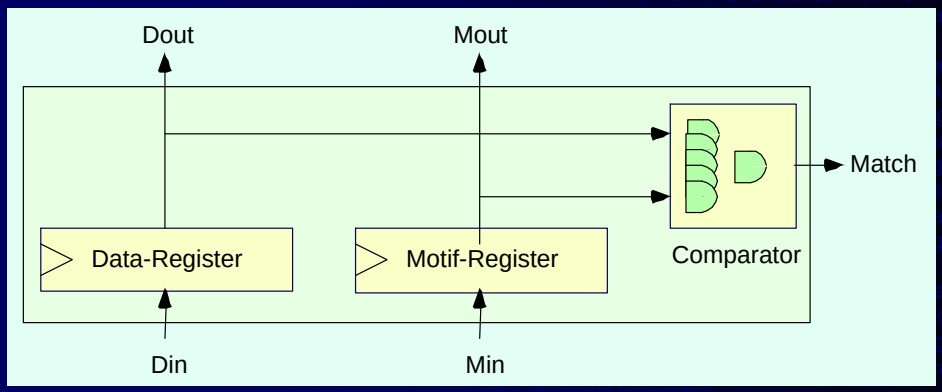


- Data = C
- Motif = A or B or C or E of F

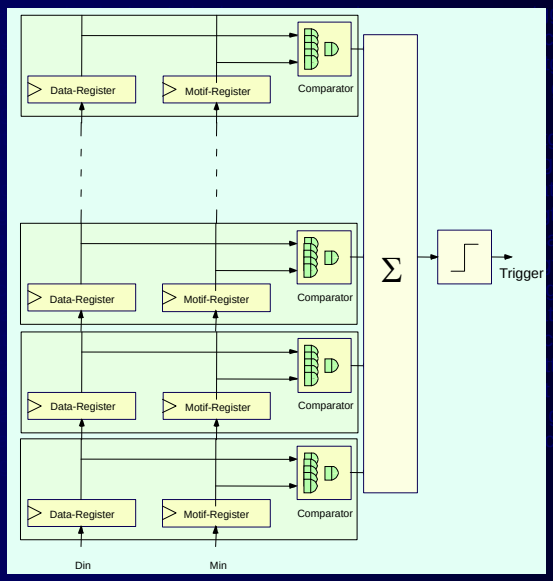
The Data Flow



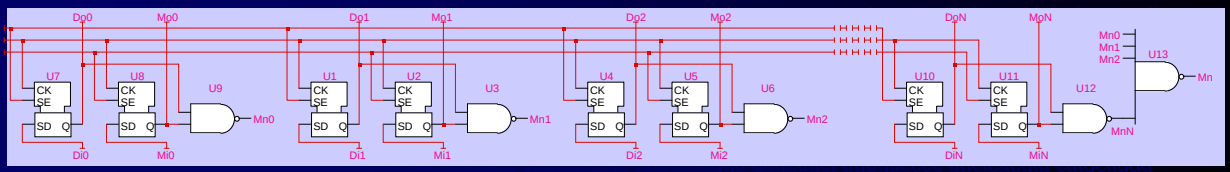
Single Position Comparator



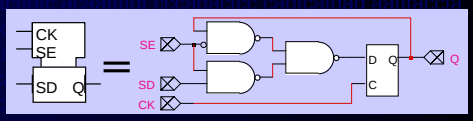
Complex Motif Comparator



Implementation (1 row)



Single Position Element

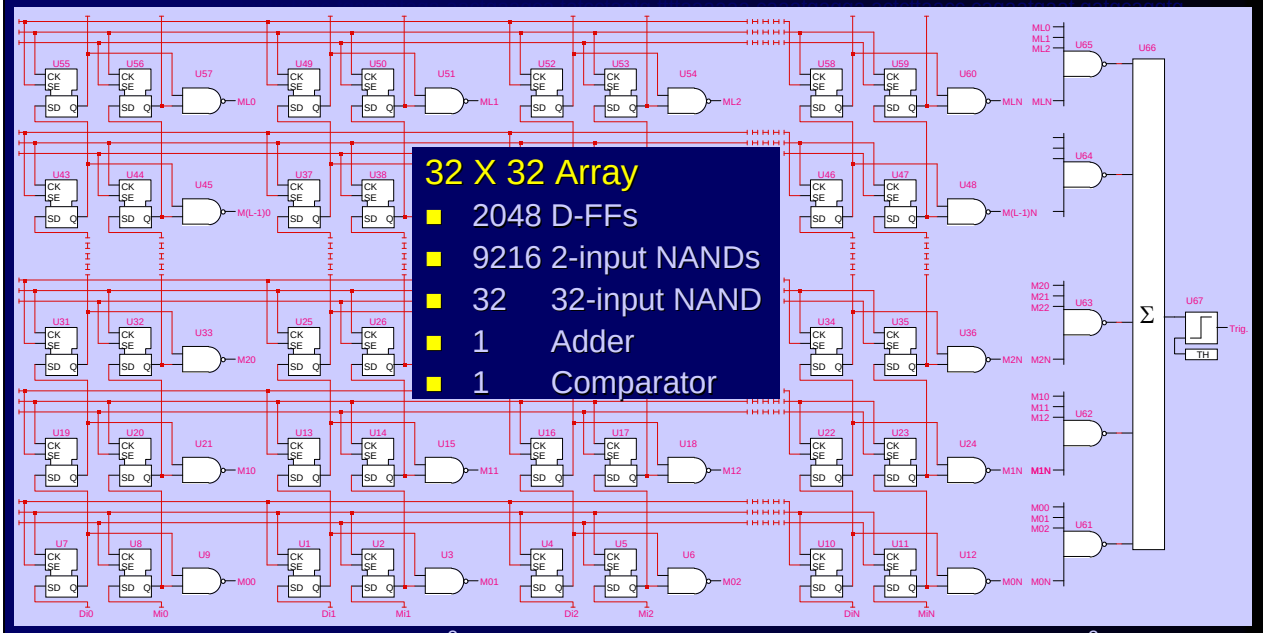


For 32 character-set

- 64 D-FFs
- 288 2-input NANDs
- 1 “32-input NAND”

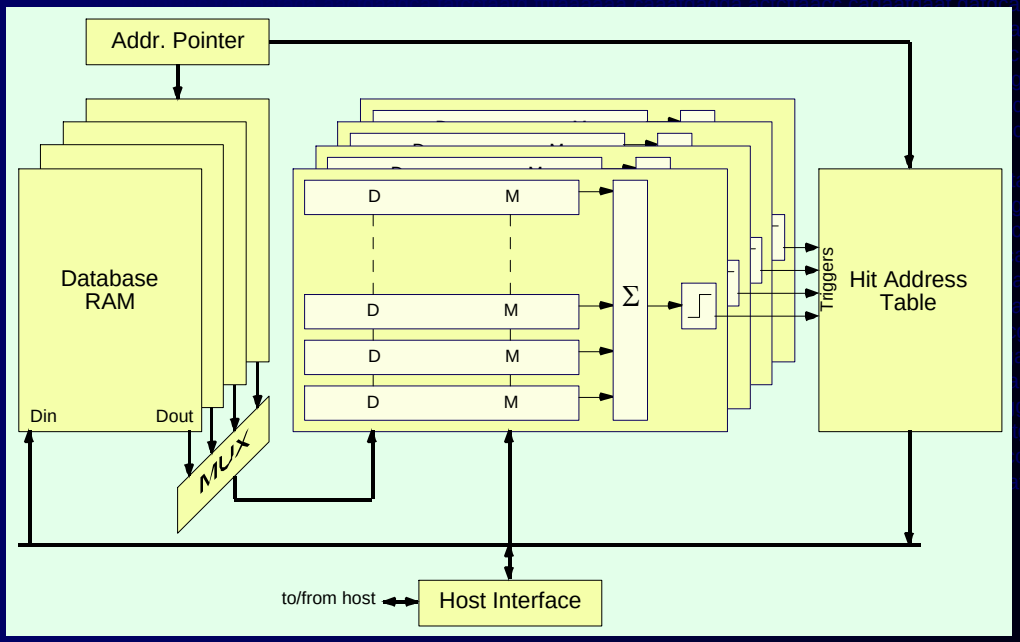
= 1 DFS1 cell

Array Implementation



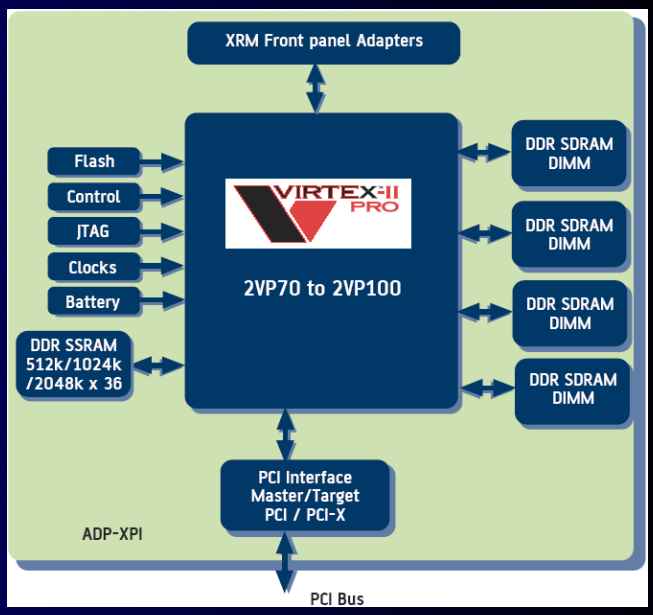
AMS C35B4: < 1 mm^2 UMC L130: < 0.2 mm^2

Functional Implementation



The Alpha-data ADP-XPI

- PCI Interface
- DDR SDRAM
  - Up to 16 GByte
  - 4 banks
- SSRAM
- Clocks
- Mezzanine Interface
- VIRTEX-II FPGA

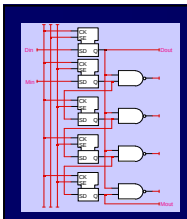


VeSPA Based PCI Board

http://harvester.fzk.de/harvester/index-seq.htm

Present Performance

- Single ASIC, AMS C35B4
  - Size: 2.4 x 1.5 mm = 3.6 mm^2
  - Core: 1 mm^2
  - Data flow: 200 MHz
  - 32 x 32 PEs = 1024 operations / CLK
- 200 Giga OPS
- Board with 8 ASICs: 0.8 Tera OPS



Future Development

- Wider word ➲ larger character set
  512 FF / position for full ASCII search
- Longer array ➲ longer motif
- Dedicated Board ➲ higher throughput
- Multiple Arrays ➲ higher throughput

Expected Performance

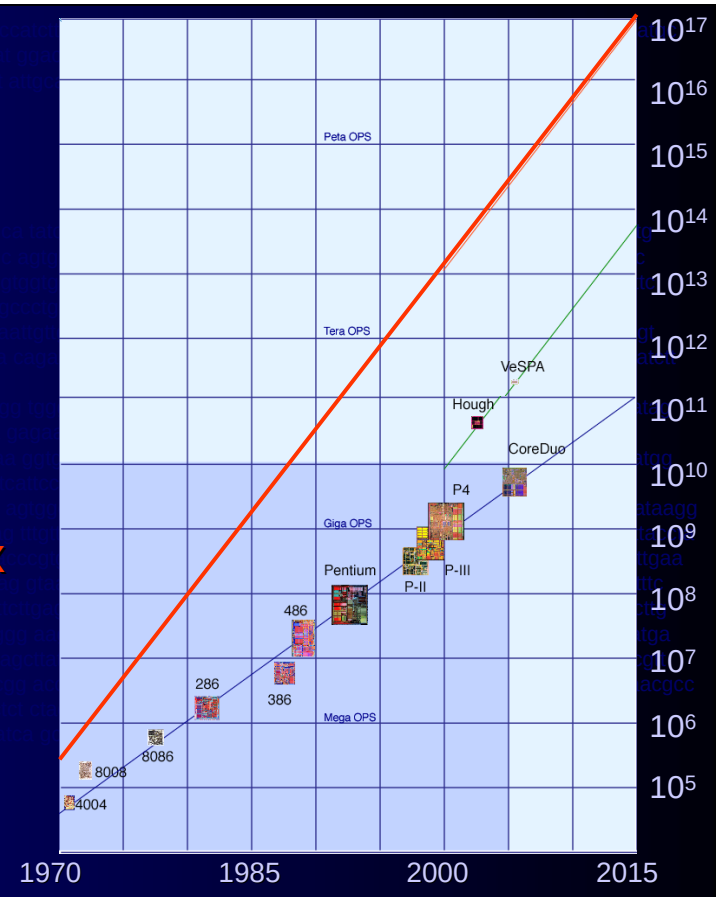
- 16 ASICs, 64 Arrays each (UMC L130)
  - 1 M operations / CLK
  - 1 GHz
- 1 Peta OPS (10^{15})

Performance
(existing technologies)

- Using current custom design technology:
  - 65 nm
  - 3 GHz
  - 1024 arrays / IC
  - 16 ICs / board
  - 22 boards / crate
- 1 Exa OPS (10^{18})

Conclusion

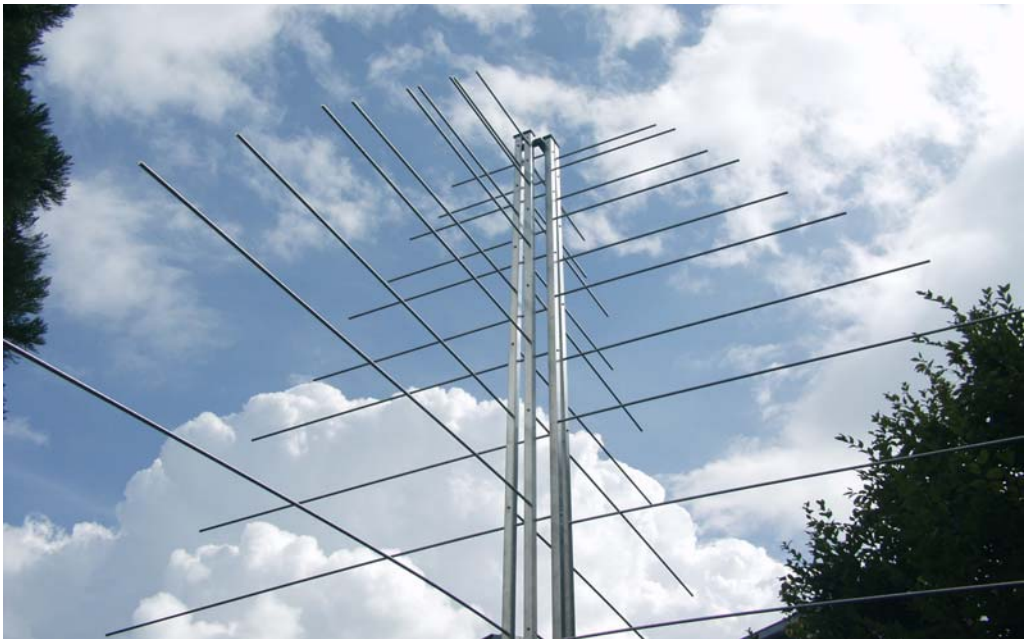
- SA Slope 2 x
- Current VeSPA Acceleration: 80 x
- 2015: 1,000,000 x



Thanks

- Dr. Christian Boulin
- Prof. Dr. Gerd-Uwe Paul
- Dr. Bernd Vettermann
- Dr. Urban Liebel
- Dipl.-Ing. Eduard Gursky
- Dipl.-Ing. Jasmin Lampert
- Mr. Tomas Whitlock B. Eng.

Radio-Detektionstechnik für höchstenergetische kosmische Strahlung



Gliederung

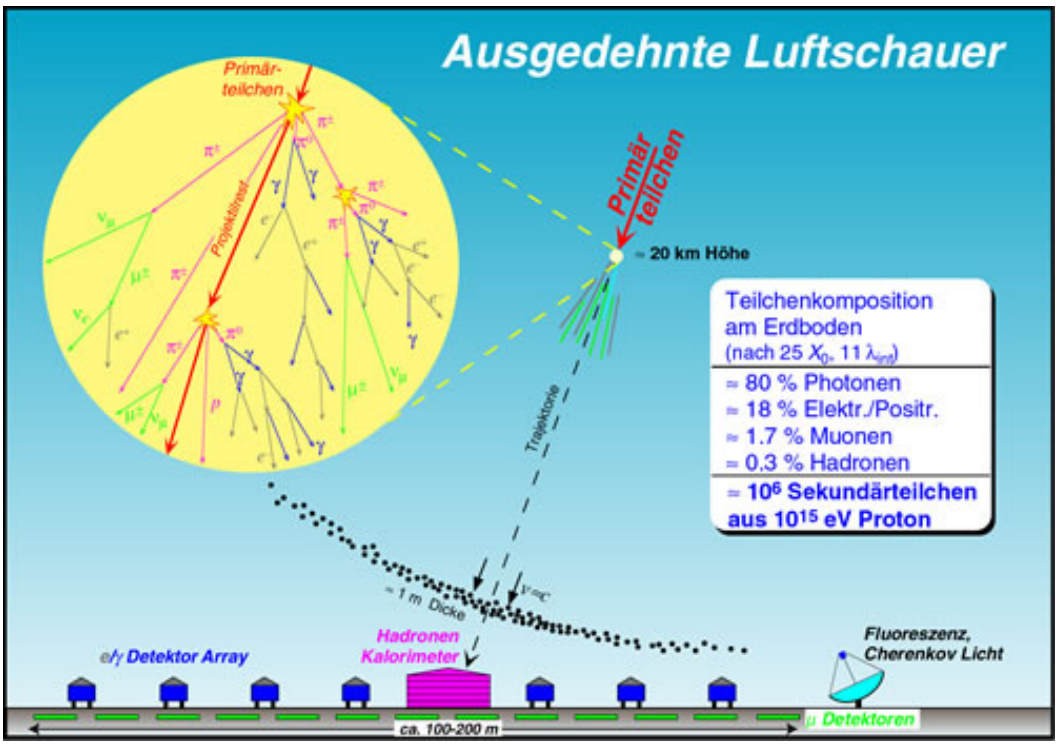
- **Kosmische Strahlung – Kosmische Luftschauer**
- **etablierte Nachweismethoden und Experimente**
- **Radiodetektion kosmischer Schauer**
- **Das Empfangssystem LOPES^{STAR}**
- **Zusammenfassung und Ausblick**

Kosmische Strahlung

Victor F. Hess, 1912 (Nobelpreis 1936)

- vorwiegend ionisierte Atomkerne (H, He, … Fe)
- Energie: E = 10^{10} eV … 10^{21} eV (LHC: $7 \cdot 10^{12}$ eV)
- Teilchenflussdichte fällt mit dN/dE ~ E^{-3}
- geringe Ereignisraten:
  @ 10^{16} eV nur 1 Teilchen / m^2 und Jahr
  @ 10^{20} eV nur 1 Teilchen / km^2 und Jahrhundert
- unterhalb von 10^{14} eV direkter Nachweis (Stratosphärenballons, Satelliten)
- große Messflächen und lange Messzeiten nur am Erdboden realisierbar
- oberhalb von 10^{14} eV nur indirekter Nachweis über den „**kosmischen Luftschauer**“ möglich

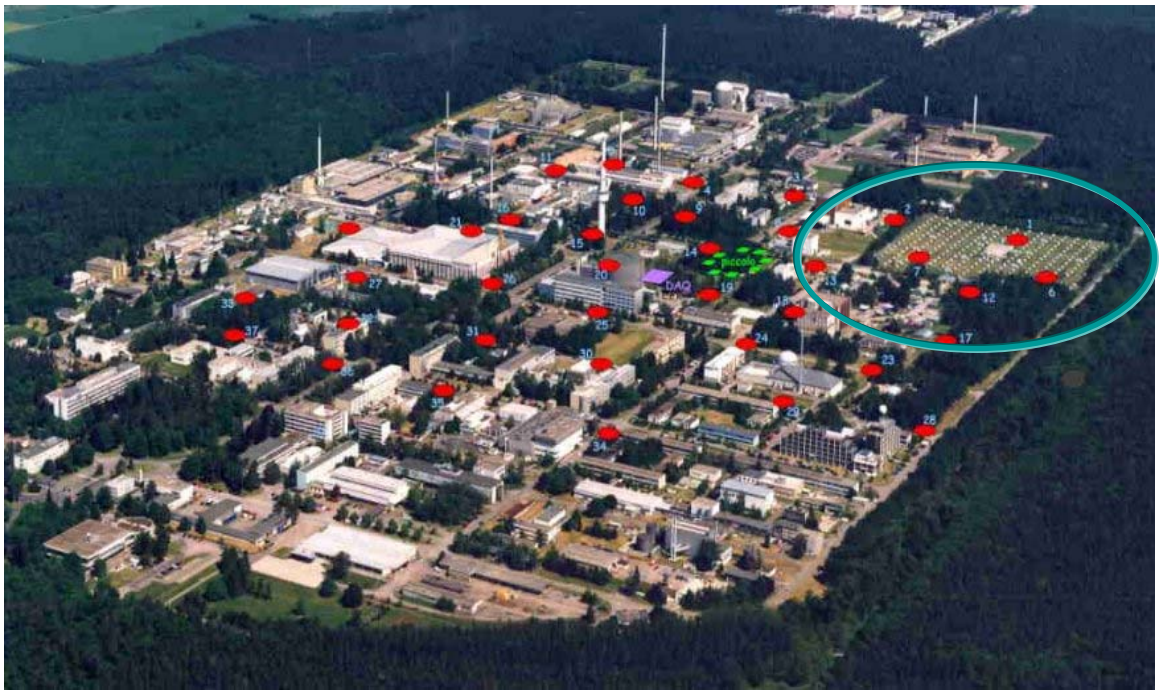
Kosmische Luftschauer



KASCADE Teilchendetektorexperiment

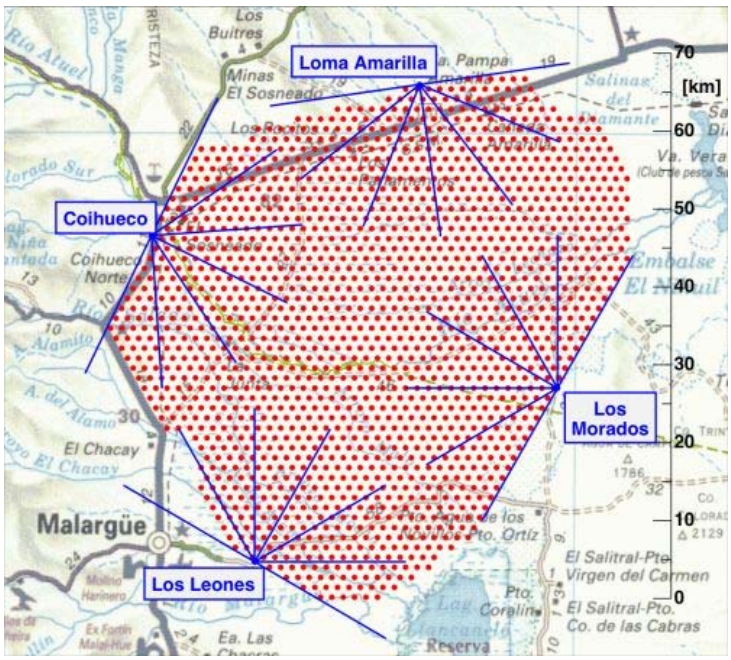
200 x 200 m², 256 Detektoren, Abstand 13 m, 10^{14} eV…10^{17} eV

KASCADE-Grande Experiment



700 x 700 m², 37 Detektoren, Abstand 130 m, bis 10^{18} eV

Pierre Auger Observatorium, Argentinien



50 x 50 km², 1600 Teilchendetektoren, Abstand 1500 m,
zusätzlich 24 Fluoreszenzlicht-Teleskope, bis 10^{21} eV

Detektoren des Pierre Auger Observatoriums

1600 Teilchendetektoren

24 Fluoreszenzlicht-Teleskope

Indirekte Nachweismethoden

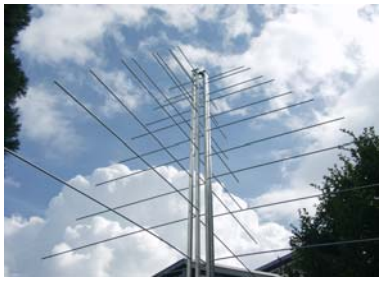
Teilchendetektoren

Teilchenkomposition
und Lateralverteilung

Fluoreszenzlicht-Teleskope

Longitudinale
Schauerentwicklung

Radioempfänger



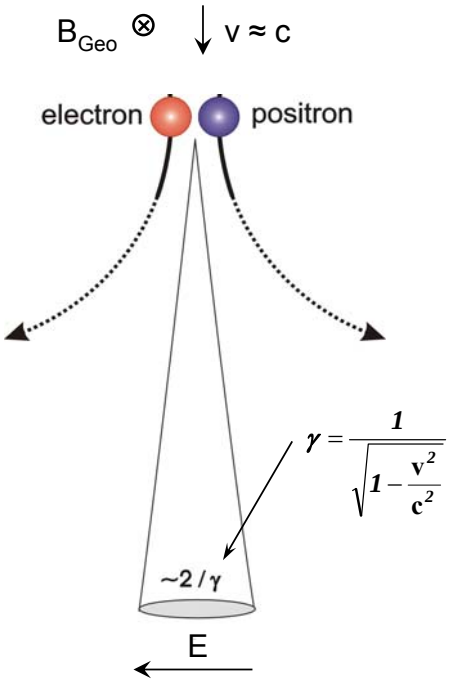
Integrales Signal über die
gesamte Schauerentwicklung

Ziel

Untersuchung und Etablierung der Radioobservation als ergänzende Nachweismethode für kosmische Schauer

Radioemissionen kosmischer Schauer

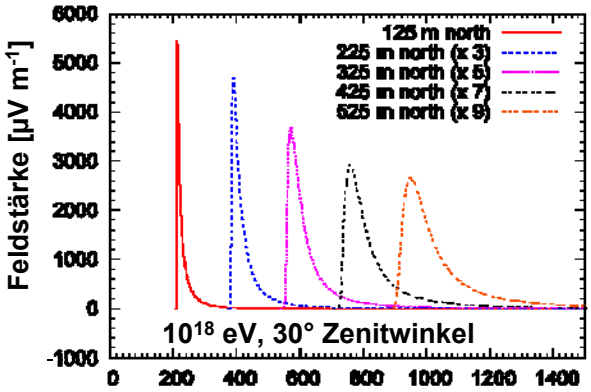
Geosynchrotronstrahlung:



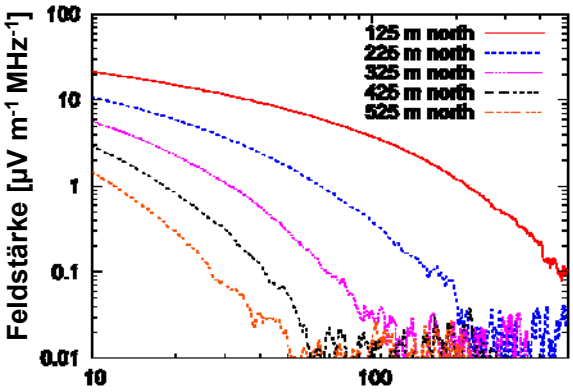
Signaleigenschaften

Modellierung und Simulation des Geosynchrotronemission

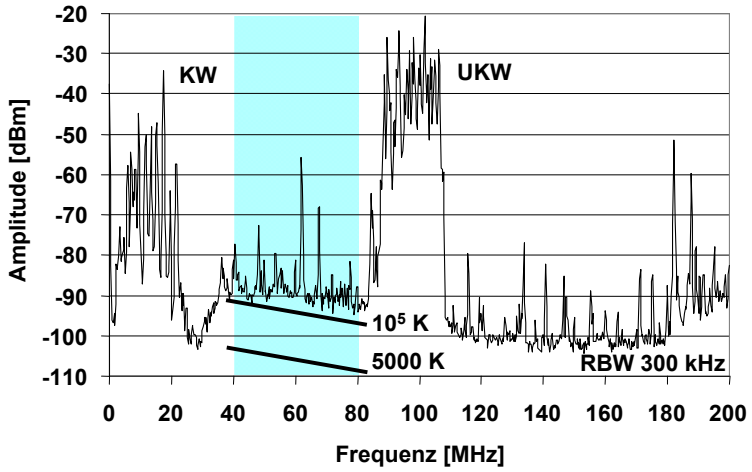
Zeitsignal



Frequenzspektrum



Störungen und Rauschen



- **Bandbegrenzung:** **40 MHz…80 MHz**
- **äußeres Rauschen @ 60 MHz:** **5.000 K** **galaktisch**
  CCIR-Report 670 **30.000 K** **Land**
  100.000 K **Stadt**
- **inneres Rauschen (Empfänger):** **≈ 500 K** **$F_E \cdot T_0$**

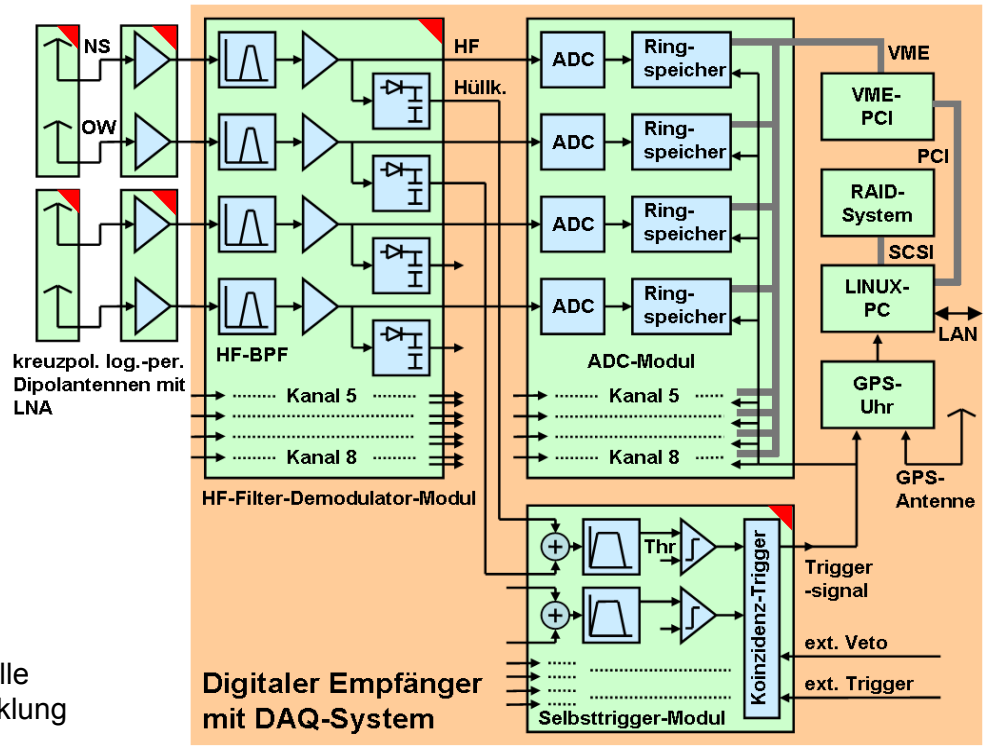
Anforderungen an das Empfangssystem

- Breitbandig (SNR ~ B)
- Störunterdrückung
- Kalibrierung
- OW- und NS-Polarisation
- Selbsttriggerung

Für großflächige Detektorarrays:

- geringer Leistungsbedarf
- moderate Kosten

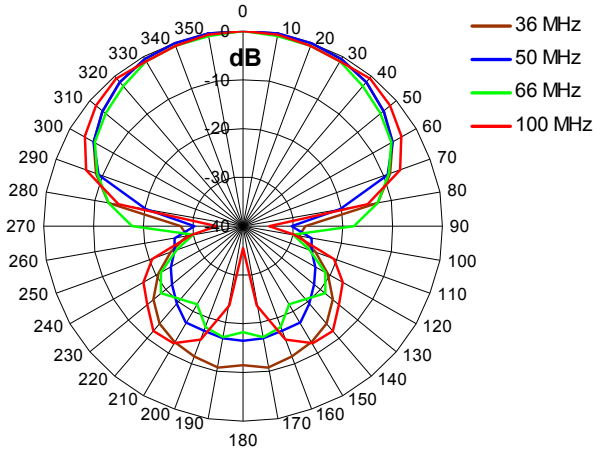
Blockschaltbild der Empfängerstation



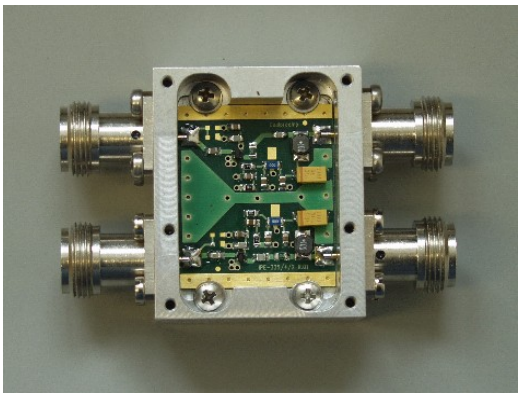
Kreuzpolarisierte log.-per. Dipolantenne

Schlüsselkomponente des Empfängers für:

- Breitbandigkeit
- Störunterdrückung
- Polarisation (NS, OW)
- Kalibrierung
- Unabhängigkeit von der Umgebung

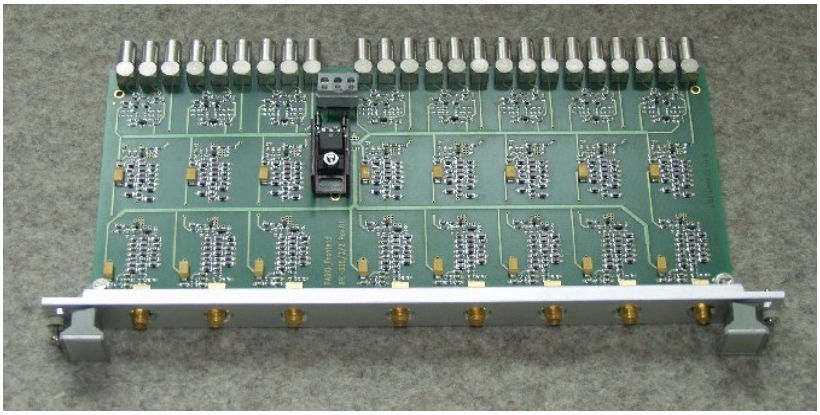


Zweikanaliger Antennenvorverstärker



- Leistungsaufnahme: 22 mW/Kanal
- Fernspeisung über Koaxkabel
- Unbedingt stabil
- Rauschtemperatur: 230 K << T_{GAL} (5000 K)
- Übersprechen: -84 dB

Filter-Demodulator-Modul



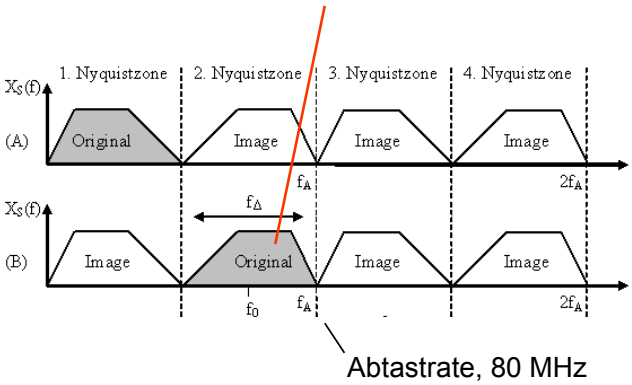
- gesamte analoge HF-Signalverarb. (8 Kanäle / 4 Antennen)
- Leistungsaufnahme: 65 mW/Kanal
- Fernspeisung der LNAs
- Bandpassfilterung, SMD-LC-Filter 34. Ordnung
- Welligkeit: ±1 dB, Sperrdämpfung: > 100 dB
- Nyquistbedingung für ADC (Sub-Sampling)
- Hüllkurvendemodulation → Selbsttrigger

Datenerfassungssystem

VME-Rack mit Empfänger- und DAQ-Komponenten

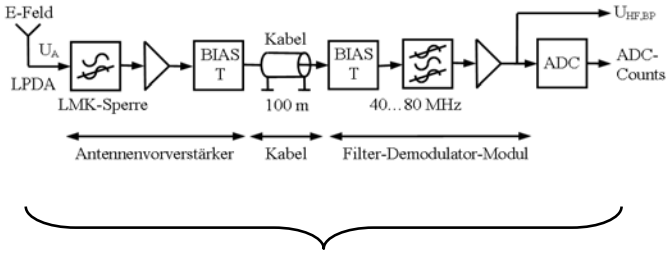
- Abtasttaktgenerator (80 MHz)
- AD-Umsetzer (12 Bit)
- Filter-Demodulator-Modul (vorherige Folie)
- Selbsttrigger-Modul (folgende Folien)
- VME-PCI-Interface

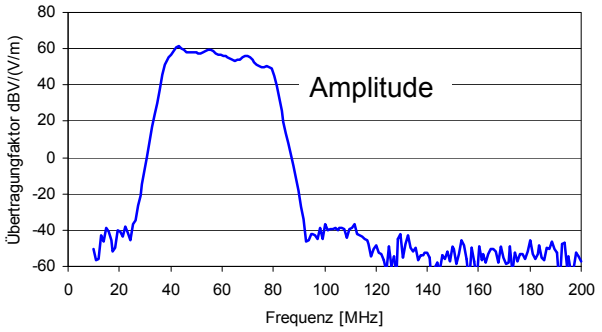
- keine ZF-Verarbeitung, direkte HF-Abtastung
- Unterabtastung (Sub-Sampling), 2. Nyquistzone

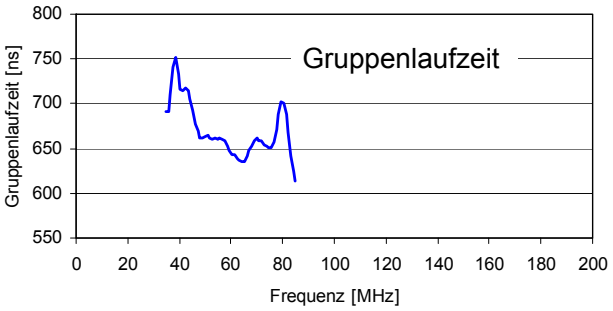


Abtastrate, 80 MHz

Kalibrierung





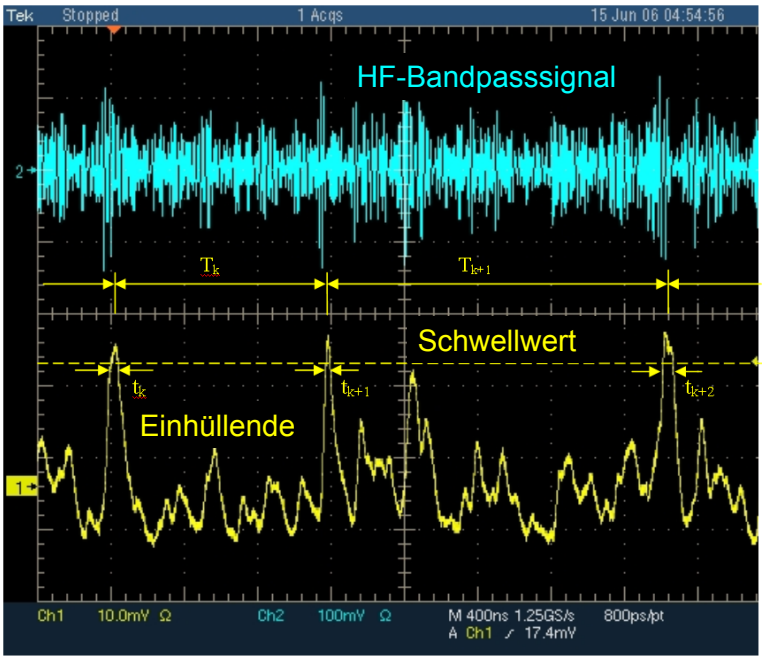


- Labor-Kalibrierung auf das Feld übertragbar
- Kalibrierunsicherheit durch die Antenne: ±0,8 dB
- keine Individual-Kalibrierung erforderlich

Selbsttriggerung

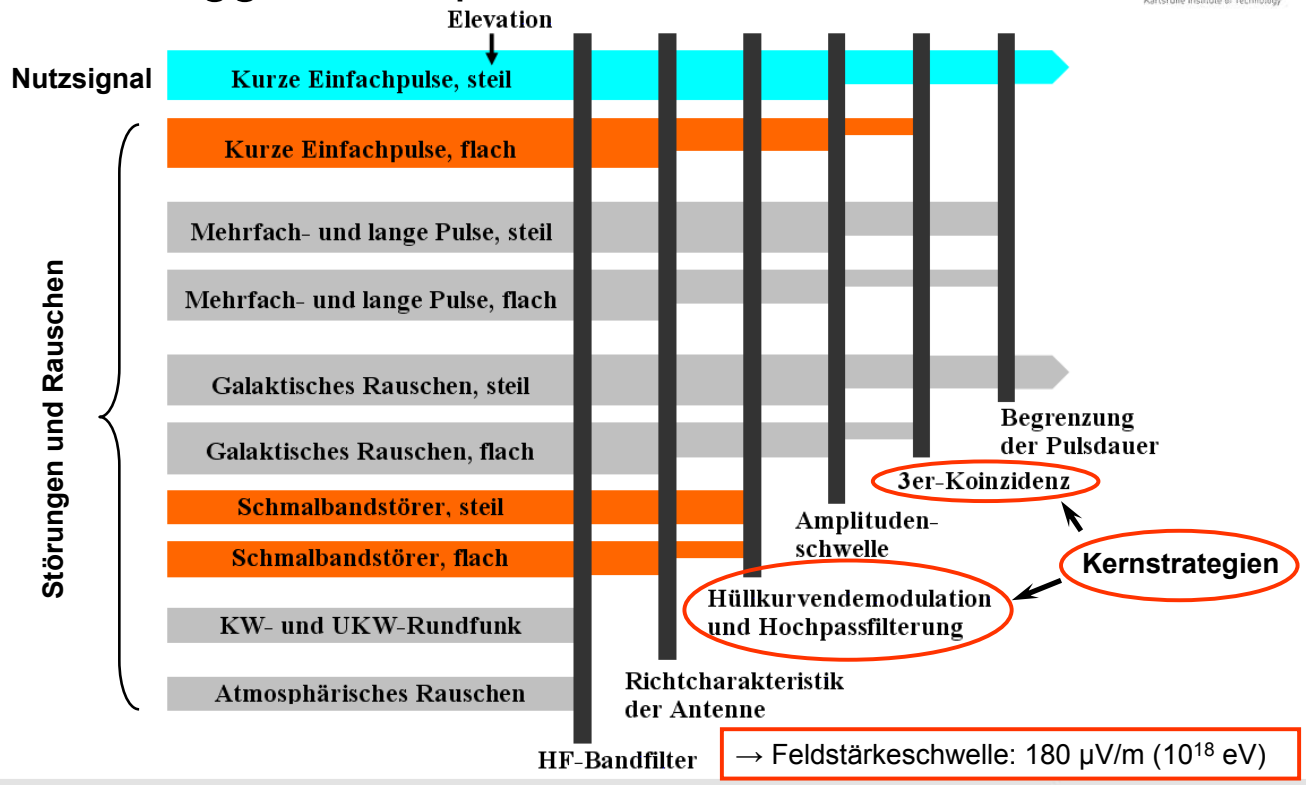
Ziel:
- autarke Radioobservation
- Detektion des Schauers anhand seines Radiopulses

Problem:
- Falschtrigger durch Rauschen und Störungen
- 1…100 Hz pro Station



Grundprinzip der Selbsttr.: Schwellwertverarbeitung

Selbsttriggerkonzept



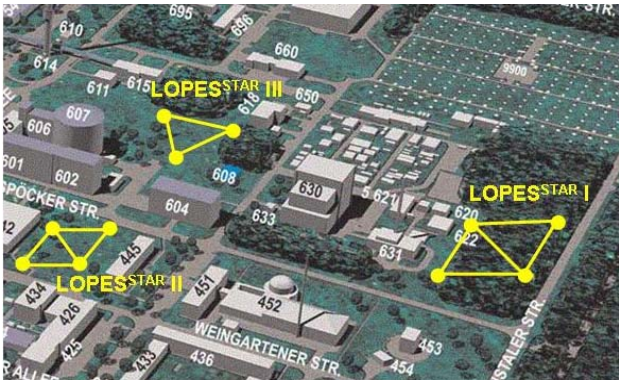
Empfängerstationen

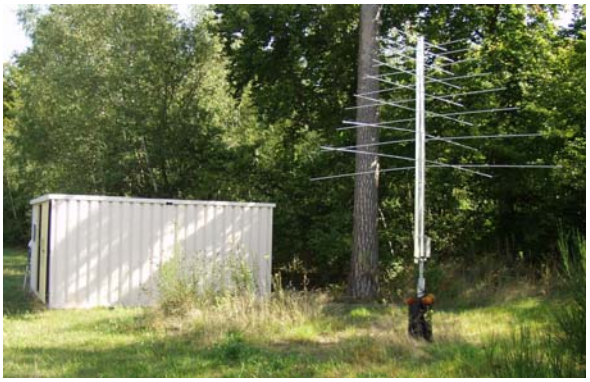
Luftschauerexperiment KASCADE-Grande

- 37 Detektorstationen

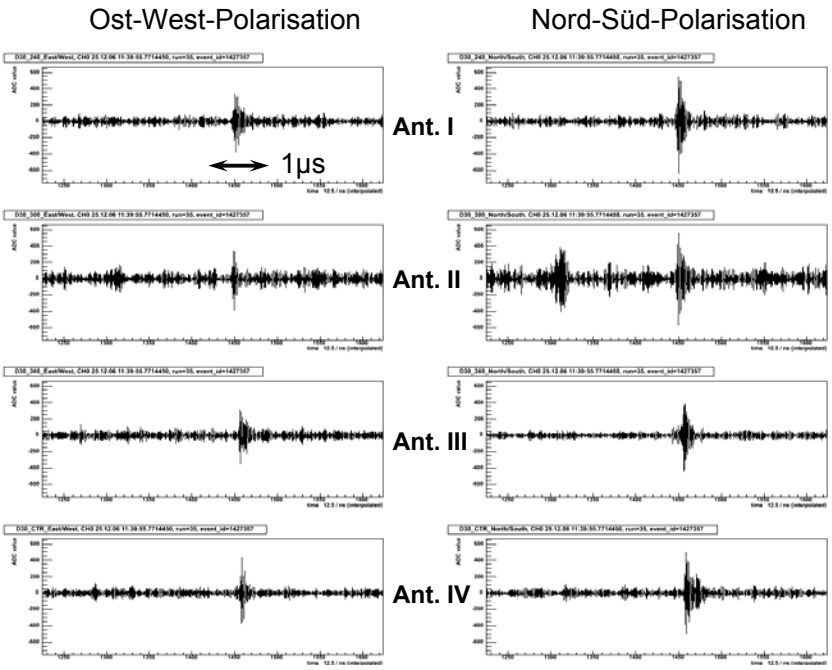
KASCADE-Grande-Station Nr. 17 mit log.-per. Dipolantenne

Drei LOPES^STAR-Empfängerstationen





Messbetrieb



Kontinuierlicher Messbetrieb:
seit Ende 2006 bzw. Mitte 2007

Beispiel:
$6 \cdot 10^{17}$ eV Schauer am 25.12.2006

Typ. Feldstärke @ $5 \cdot 10^{17}$ eV:
≈200 µV/m (gal. Rausch.: 16 µV/m)

Ereignisrate @ $5 \cdot 10^{17}$ eV:
≈ 1 Ereignis / 10 Tage und Station

Statistik:
mehrjähriger, kontinuierlicher Messbetrieb erforderlich für einige 100 typische Schauer

Ergebnisse

- **Konzept, Entwicklung, Aufbau eines Empfangssystems**
  - Antenne
  - Elektronik
- **Vollständige Kalibrierung der Verstärkung und Gruppenlaufzeit**
  - keine individuelle Kalibrierung im Feld erforderlich
- **Komponenten tauglich für großskalige Radiodetektorarrays**
  - Leistungsbedarf
  - Kosten
  - Kalibrierfähigkeit
- **Selbsttriggerung: Konzept, Theorie, Umsetzung in Hardware**
  - Theorie wurde messtechnisch mit kalibriertem Empfänger bestätigt
  - erstmals Angabe der Absolutwerte von Falschtriggerrate und Schwellwert
- **Stabiler, kontinuierlicher Messbetrieb**
  - drei Empfängerstationen
  - Radiosignale in Koinzidenz mit kosmischen Schauern

Ausblick

- **Mehrjährige Fortsetzung des aufgenommenen Messbetriebes**
- **Errichtung eines „Engineering Arrays“ am Pierre Auger Observatorium in Argentinien**

Radio-Detektionstechnik für höchstenergetische kosmische Strahlung

Dr. Oliver Krömer, Forschungszentrum Karlsruhe, IPE

Die Observation der hochenergetischen Komponente der kosmischen Teilchenstrahlung erfolgt durch indirekte Messungen (Bild 3). Dabei dringt das primäre kosmische Teilchen in die Erdatmosphäre ein und erzeugt durch Wechselwirkungen mit den Luftmolekülen einen kosmischen Luftschauer (Bild 4). Die am Erdboden ankommenden Sekundärteilchen werden mit Teilchendetektorarrays nachgewiesen (Bild 5, Bild 6). Das Fluoreszenzlicht der entlang der Schauerachse angeregten Stickstoffmoleküle wird im nahen Ultraviolettbereich mit Spiegelteleskopen beobachtet (Bild 7, Bild 8). Neben diesen etablierten Nachweismethoden wird derzeit die Radioobservation der Geosynchrotronemission des kosmischen Luftschauers als weitere, zusätzliche Beobachtungsmethode untersucht (Bild 9). Geosynchrotronemission entsteht durch Beschleunigung der vom kosmischen Luftschauer erzeugten relativistischen Elektron-Positron-Paare durch Lorentzkräfte im Magnetfeld der Erde (Bild 10). Am Erdboden führt dies zu einem Einzelimpuls der elektrischen Feldstärke mit einem kontinuierlichen Frequenzspektrum, das von wenigen MHz bis über 100 MHz reicht (Bild 11).

Ausgehend von den Signaleigenschaften der Geosynchrotronemission und einer Analyse der überlagerten Stör- und Rauschkomponenten (Bild 12) wurde ein geeignetes Empfängerkonzept erarbeitet (Bild 14). Da dieses in der geforderten Form kommerziell nicht verfügbar war, wurde das Empfangssystem bestehend aus Antenne (Bild 15), Empfängerelektronik (Bild 16, Bild 17) und Datenerfassung (Bild 18) entwickelt und realisiert. Dabei sind bereits auch Aspekte großflächiger Radiodetektorarrays, wie ein geringer Leistungsbedarf für eine photovoltaische Versorgung und Wirtschaftlichkeit, berücksichtigt worden (Bild 13). Ergebnis ist ein kalibrierter, mehrkanaliger, digitaler Breitband-Messempfänger für den lückenlosen Empfang zwischen 40 MHz und 80 MHz (Bild 14). Seine inhärente Störunterdrückung resultiert im Wesentlichen aus der Antennenrichtcharakteristik und der Frequenzselektion und erlaubt die effektive Radioobservation kosmischer Schauer auch in besiedelter Umgebung.

Mehrere der hier konzipierten Empfängerstationen wurden auf dem Gelände des Luftschauerexperimentes KASCADE-Grande im Forschungszentrum Karlsruhe errichtet (Bild 22). Sie empfangen Radiosignale, die eindeutig mit kosmischen Schauern koinzidieren (Bild 23). Die Kalibrierung über den gesamten Signalpfad (Bild 19) und das Polarisationsverhalten gestatten es, die Radioemissionen kosmischer Schauer und die zugrunde liegenden Emissionsmodelle systematisch zu untersuchen.

Für einen autarken Betrieb der Empfängerstationen ist eine Selbsttriggerung gefordert, welche den kosmischen Schauer allein anhand des Radiosignals erkennt (Bild 20). Dabei ist aufgrund additiver Rausch- und Störgrößen mit sinkender Detektorschwelle eine steigende Anzahl von Falschtriggern abzusehen. Die hier konzipierte Selbsttriggerung ermöglicht bei technisch handhabbaren Falschtriggerraten eine geeignet niedrige Schwelle (Bild 21). Das Verhalten und die Grenzen dieses Selbsttriggerkonzeptes werden sowohl theoretisch als auch messtechnisch durch seinen Einsatz unter realen Umgebungsbedingungen nachvollzogen. Auf dem radiolauten Gelände des Forschungszentrums Karlsruhe liegen die Feldstärkeschwellen nur einen Faktor 3 oberhalb des theoretischen Minimums, das in unbesiedelten Gebieten durch das stets präsente galaktische Rauschen gegeben ist. Die resultierende Nachweisgrenze für kosmische Strahlung wird in unbesiedelten Gebieten oberhalb einer Energie des Primärteilchens von ca. $5 \cdot 10^{17}$ eV und in der Umgebung des Forschungszentrums Karlsruhe oberhalb von ca. 10^{18} eV erwartet.

Virtuelle Infrastruktur für „Remote Instrumentation”

Michael Sutter

Forschungszentrum Karlsruhe
Institute for Data Processing and Electronics (IPE)

AugerAccess

- Extension of Pierre Auger Cosmic Ray Observatory
- EU project in the Sixth Framework Program
- Improve access to observatory in Argentina
  - Replication of 7 Tbytes of measured data per year
  - Upgrade of the communication link
- Improve management via remote connection
  - Controllable world wide
  - Simplify software maintenance
  - Software for remote control and remote monitoring
    - Authentication and authorization
    - Secure communication

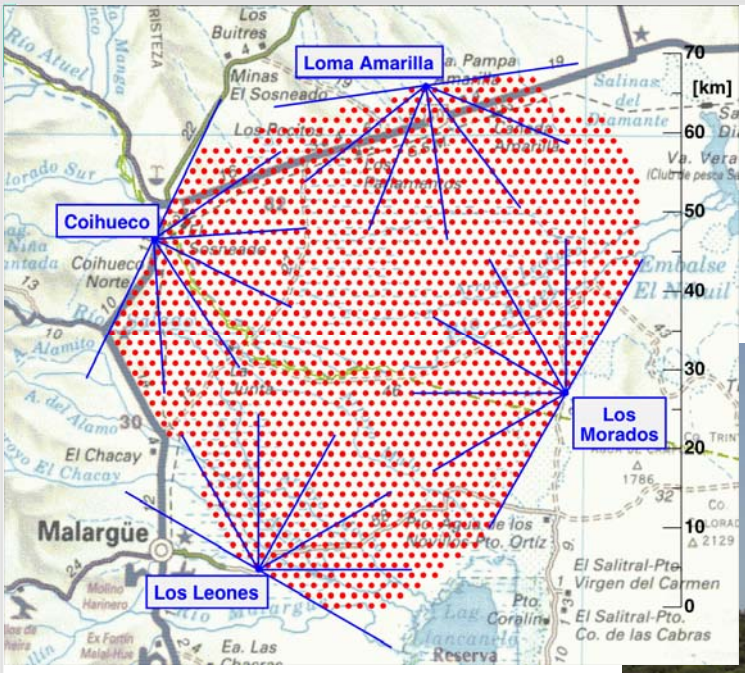
Conditions for Remote Software

- Problems:
  - System for Data Acquisition and Slow Control with
    - Hardware distribution on large area
    - Heterogeneous architecture
    - Multiple network services (DNS, DHCP, …)
    - Network boot (PXE – Preboot Execution environment)
  - ➡ Complex system architecture
  - System already in production
  - ➡ Not available for development and testing
- Objectives:
  - Environment for software development
  - Additional development system in hardware not possible

➡ Virtualization of entire Data Acquisition environment

Auger Observatory

- Study of ultra-high energy cosmic rays
  - Energies up to 10^{20} eV
  - Rare events (one particle per km² and century)
  - ➡ Giant detector needed
- Hybrid measurement
  - Surface detector (SD)
    - 1600 Water-Cherenkov-Tanks
  - Fluorescence detector (FD)
    - 24 telescopes in 4 buildings

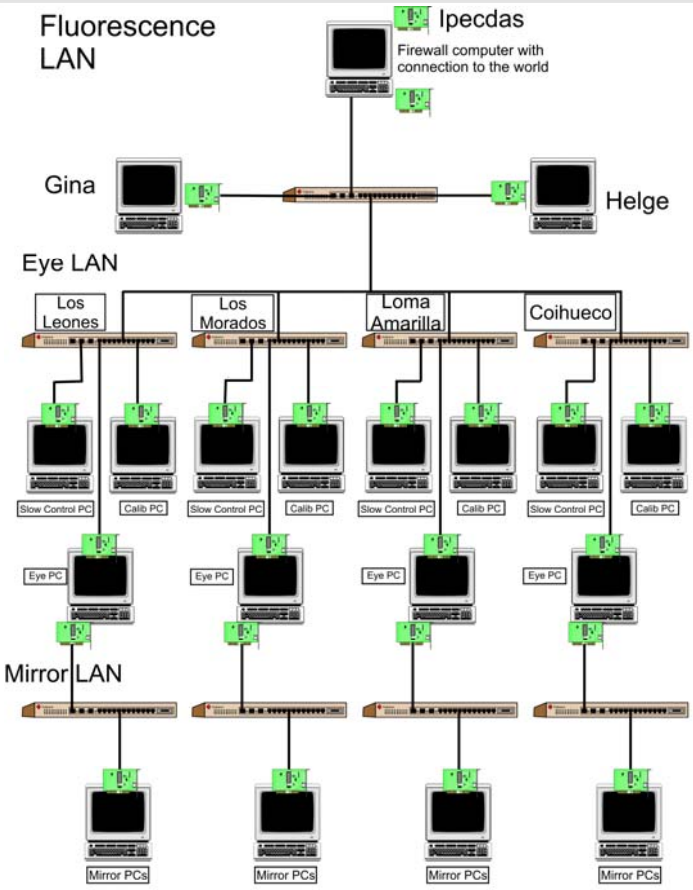


- 3000 km²
- Hardware is distributed

KIT – die Kooperation von Forschungszentrum Karlsruhe GmbH und Universität Karlsruhe (TH)

HELMHOLTZ GEMEINSCHAFT

Forschungszentrum Karlsruhe in der Helmholtz-Gemeinschaft



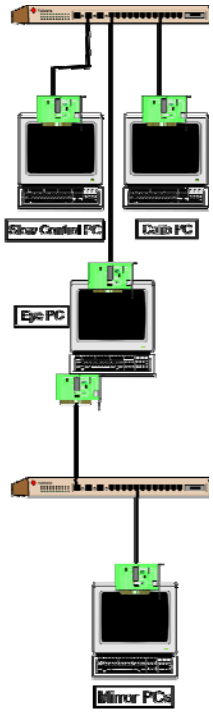
- Internet
- Central campus
- Radio link
- 4 buildings
- 6 computers each

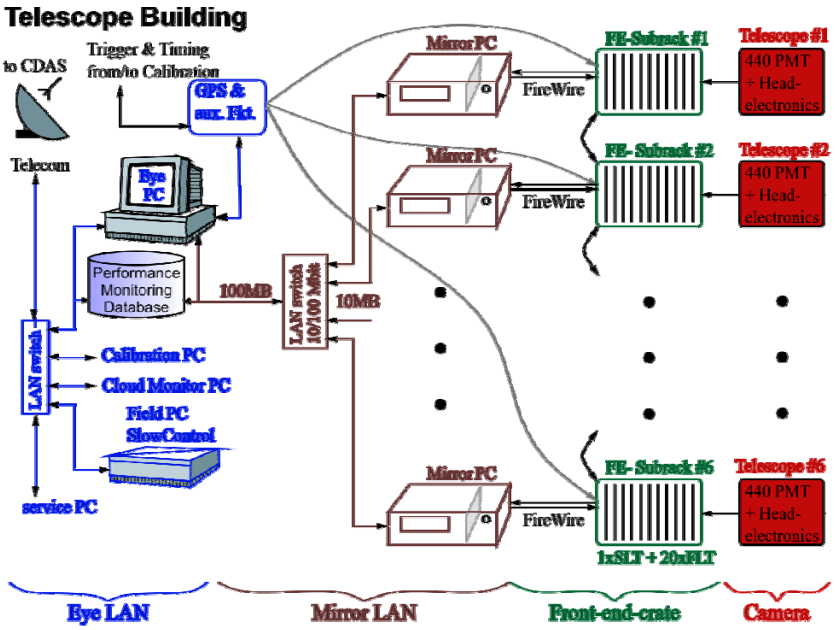
KIT – die Kooperation von Forschungszentrum Karlsruhe GmbH und Universität Karlsruhe (TH)

HELMHOLTZ GEMEINSCHAFT

Forschungszentrum Karlsruhe in der Helmholtz-Gemeinschaft

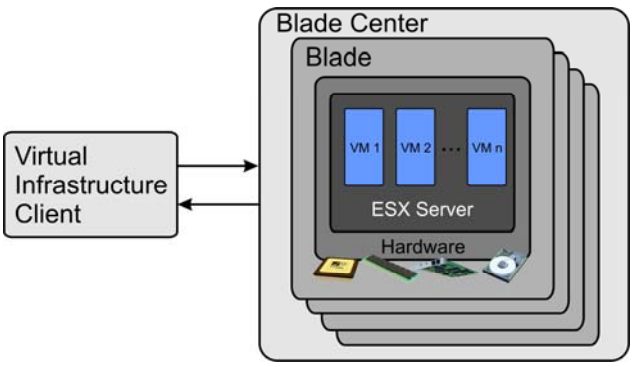
Telescope Building Infrastructure





Realization of Virtualization

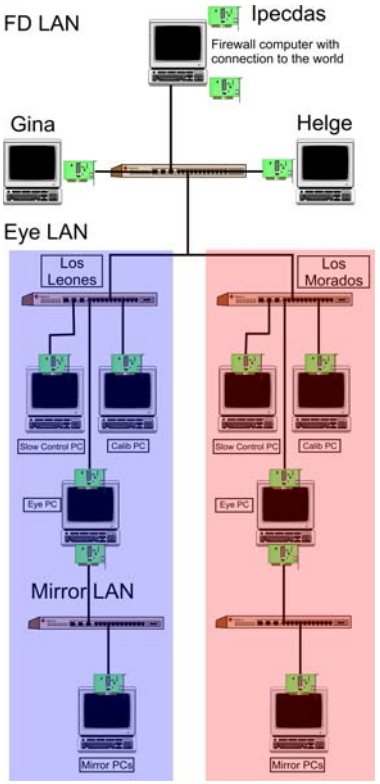
- We use IBM Blade
  - VMware ESX Server as hypervisor on the host
  - Virtual Infrastructure Client for administration and installation of all virtual machines



- 2 dual core Xeons @ 2.33 GHz
- 16 GB Memory

Virtualized development environment

- Masquerading on Ipecdas
- Services for Auger software
  - DNS on Gina and every EyePC
  - PXE boot for every MirrorPC
  - DHCP on every EyePC
  - NFS (Network file system) on every EyePC
- Auger software

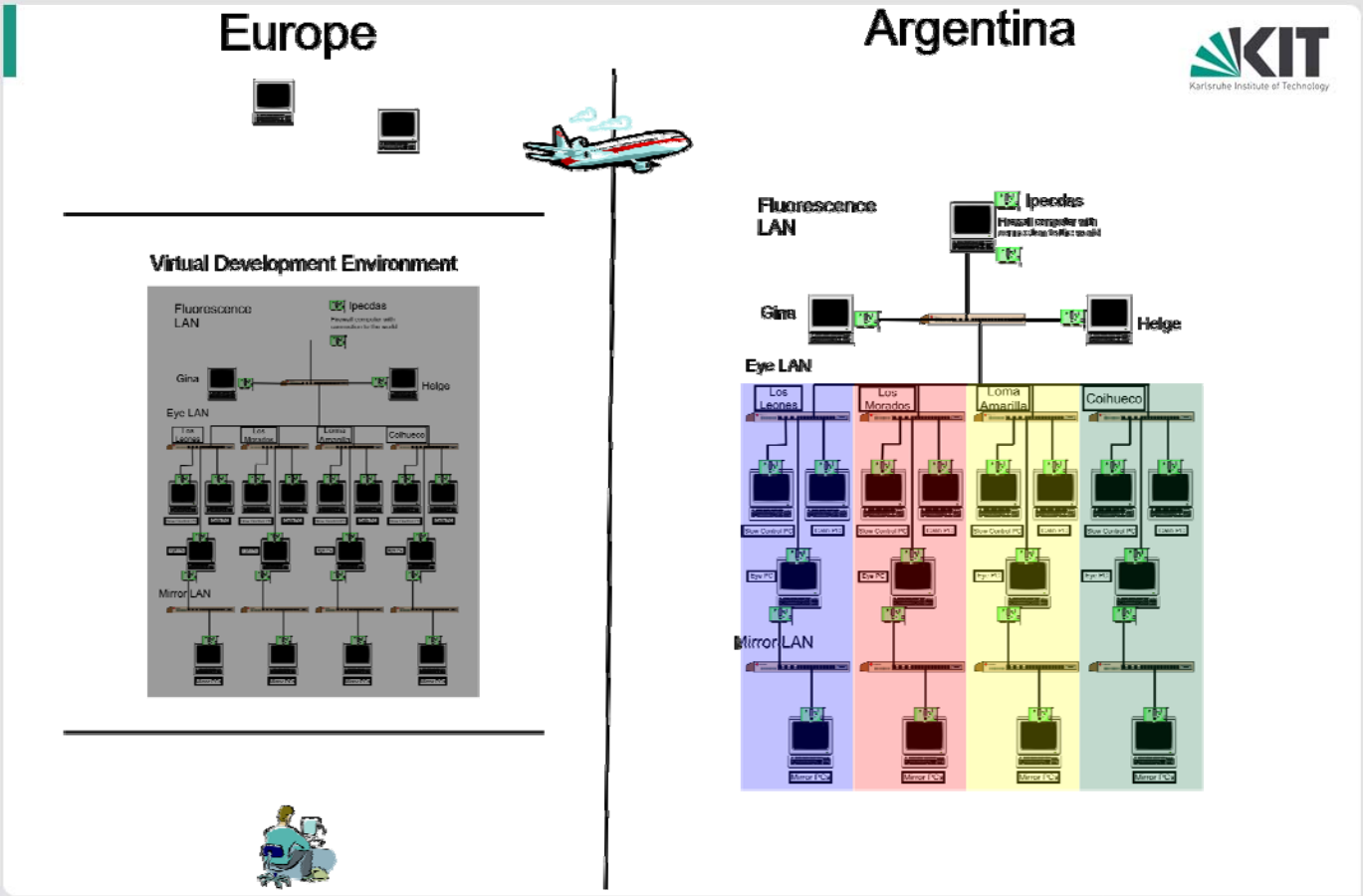


Virtual system offers the same capabilities as the observatory in Argentina

Limitations of Virtualization

- Not possible
  - Radio link
  - Special hardware can not be virtualized (telescopes)
  - Latency time
  - Hardware characteristics
- Possible
  - Simulation of entire FD environment inclusive “measurement”
  - Network communication as in the observatory
  - Easy to copy
  - Extensible with physical hardware

Virtualization fulfills all requirements



Application: Remote Client

Service based communication

GT 4 Grid security used for authentication and authorization from outside

- Problems
  - Auger software in C++, CORBA
  - GT 4 Java
- Solution
  - Apache Axis 2 for coupling of different languages
    - Java Client
    - C Server
    - Problem compiling Axis service with Data Acquisition software

Next steps

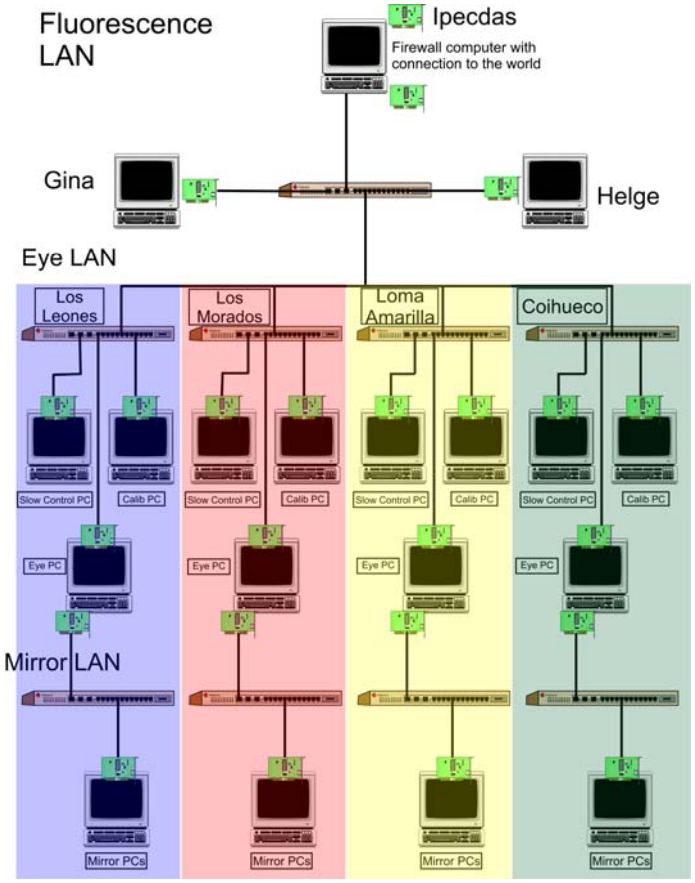
- Installation of remote client
- Extension with physical hardware
  - Slow control system for development purposes
- New version of Operating System
  - At the moment only one dedicated version is possible
- Automatic Data Acquisition
  - Time scheduled from astronomical tables
- Virtualization of existing observatory
  - New release configured on development environment
  - Easy rollback

Conclusion

- Virtualization allows reproduction of entire system
  - Lower costs
  - Nearly same possibilities as the hardware
- Virtualization is flexible solution
  - Simulating complex hardware architecture
  - Software development for Auger observatory
  - Simulation of network architecture
  - Measurement of data with the development environment

→ Virtualization offers new possibilities for projects in regard to software and architecture development

Thank you for your attention!



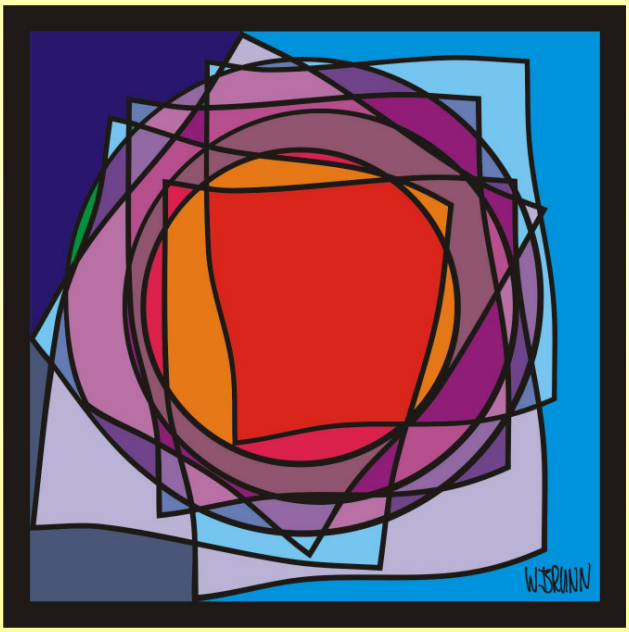
Single Source Publishing – Überblick und Konzepte

SEI 1

Single Source Publishing

Single Source Publishing Überblick und Konzepte

Peter Siepermann



SEI 2

Inhalt

- Single Source Publishing
  - Begriffsbestimmung
  - auf der Basis von Dokumenten
  - auf der Basis von Datenbanken
  - Bewertung
- Fallbeispiel w3l
  - Architektur
  - E-Learning mit w3l.

Single Source Publishing – Überblick und Konzepte

SEI 3

Single Source Publishing – Begriffsbestimmung

- Single Source Publishing
  - eine Quelle
  - mehrere Zielformate, z.B
    - Druckausgaben
    - Online-Darstellungen
- Abgrenzung zum Cross Media Publishing
  - Schwerpunkt auf redundanzarmer Erstellung der Informationen
  - Cross Media Publishing eher aus Sicht der Publikation entstanden
- Datenbank-basiertes Single Source Publishing ähnelt Content Management Systemen.

SEI 4

SSP auf Ebene von Dokumenten

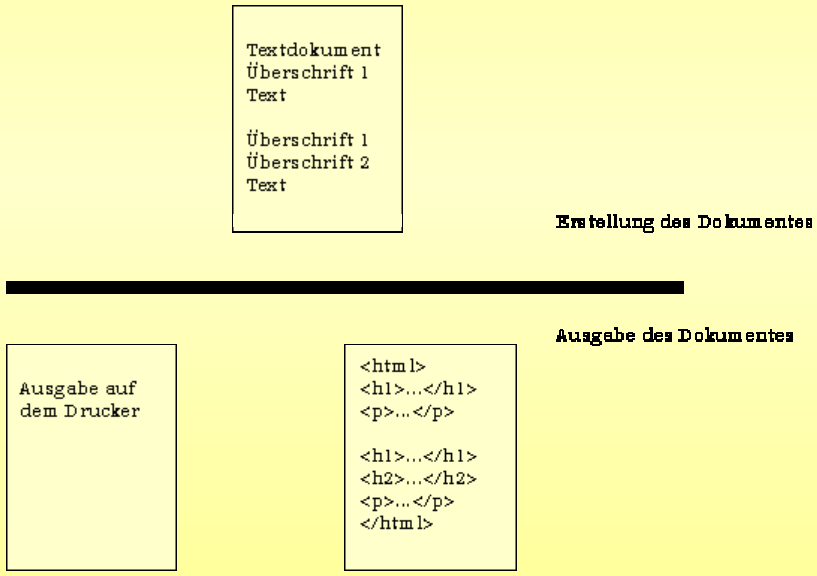
- SSP ursprünglich auf der Basis von Textdokumenten
  - ein Textdokument
  - Druckausgabe und Online-Formate wie WinHelp oder HTML
- Beispiele
  - Doc-To-Help generiert aus Word-Dokumenten u.ä.
  - WinHelp und HTML aus Word-Dokumenten u.ä.
  - HTML aus LaTeX-Dokumenten
  - Textdokumente und HTML aus
    - doxygen
    - Javadoc.

Single Source Publishing – Überblick und Konzepte

SEI 5

SSP auf Ebene von Dokumenten

- 1:1-Abbildung der Absatzformate
  - *Überschrift 1* abgebildet auf *<h1>* in HTML usw.



SEI 6

Datenbank-basierte Verfahren

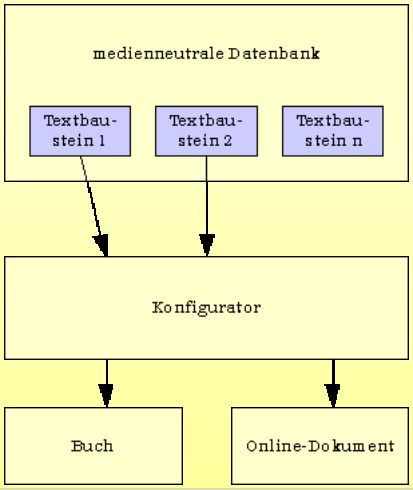
- Weiterentwicklung des SSP
  - nicht nur auf der Basis vollständiger Textdokumente
  - Behandlung von Textbausteinen
- Verwaltung von Textbausteinen in einer Datenbank
  - medienneutral
  - Auszeichnung der Texte in der Regel durch XML-Tags
  - Anreicherung des Textinhaltes mit Metainformationen
- Zieldokument wird zu einer Sammlung einzelner unabhängiger Textbausteine.

Single Source Publishing – Überblick und Konzepte

SEI 7

Datenbank-basierte Verfahren

- Textbausteine
  - dürfen sich inhaltlich nicht überschneiden
  - müssen unabhängig voneinander sein
- => Textbausteine sind häufiger wiederverwendbar



SEI 8

Bewertung

- + Redundanzarme Erstellung von Informationen
  - weniger Aufwand bei der Erstellung
  - weniger Aufwand bei der Pflege
  - Dokumente bleiben synchron
- + Wiederverwendung von Textbausteinen
  - möglich durch modularen Aufbau
  - Kostenersparnis bei Erstellung ähnlicher Dokumente
- + Vorteile für den Konsumenten
  - Informationen in Zielmedien gleich strukturiert
  - gezieltere Suchen möglich.

Single Source Publishing – Überblick und Konzepte

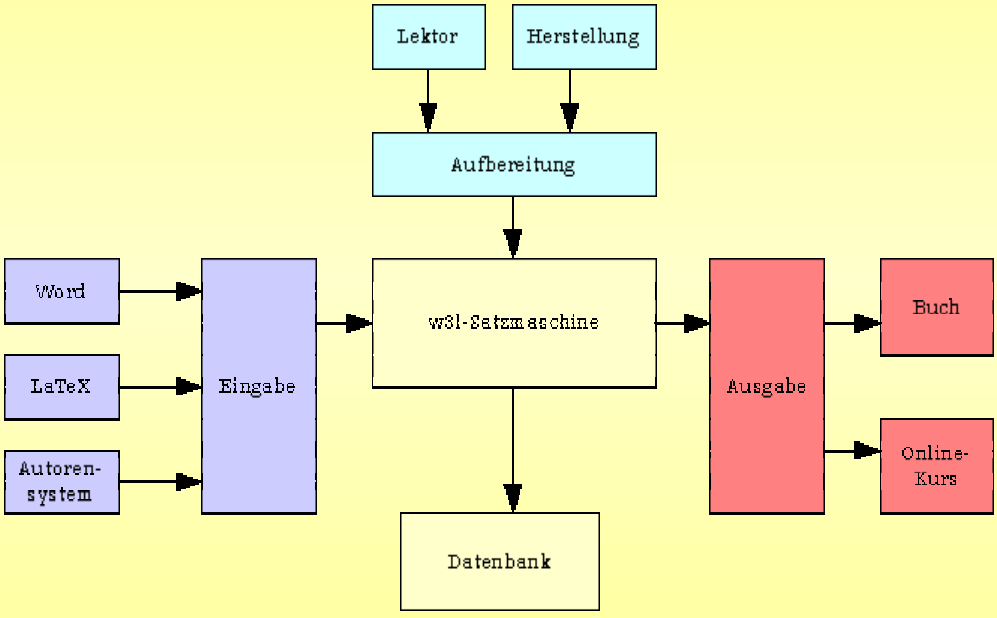
SEI 9

Bewertung

- - hohe Kosten eines SSP-Systems
  - Anschaffung
  - Administration / Wartung
  - Schulung / Einarbeitung
- - höherer Aufwand bei der Erstellung der Texte
  - Modularität muss sichergestellt sein
  - höherer Aufwand durch Bereitstellung von Metainformationen
- - Konfigurationsaufwand bei der Erstellung des Zieldokumentes
  - Wunsch nach einheitlichem Schreibstil bei verschiedenen Autoren.

SEI 10

Fallbeispiel w3l – Architektur



Single Source Publishing – Überblick und Konzepte

SEI 11

Fallbeispiel w3l – Architektur

- Satzmaschine mit medienneutraler Datenbank als zentrales Element (w3l-Server)
- zugehöriger Windows-Client auf der Eingabeseite
- w3l-Server und Windows-Client generiert mit dem Janus-System
  - Generierung einer Stammdatenverwaltung mit Datenbank-Anbindung
  - Client-Server-fähig
  - Generierung auf der Basis angereicherter UML-Modelle
  - Generierung der Oberflächen für den Windows-Client.

SEI 12

Fallbeispiel w3l – Architektur

- Auf der Eingabeseite
  - Windows-Client
  - LaTeX-Importer
    - vor allem für bereits bestehende Systeme
  - Word-Add-In
    - bestehende Texte können genutzt werden
    - Autor kann in vertrauter Umgebung arbeiten
- Aufbereitung
  - Lektoren und Hersteller arbeiten wie auch der Autor auf gemeinsamer Datenbasis
  - w3l-Server dient als zentrales Repository
  - keine weitere Versionsverwaltung erforderlich.

Single Source Publishing – Überblick und Konzepte

SEI 13

Fallbeispiel w3l – Architektur

- Ausgabeseite
  - Generierung von TeX-Dokumenten auf der Basis der Wissensbausteine
    - Ausgabe als Buch
  - Online-Veröffentlichungen
    - Generierung von JSPs mit Hilfe mehrstufiger Transformationen der Textbausteine
    - Einsatz von XSLT.

SEI 14

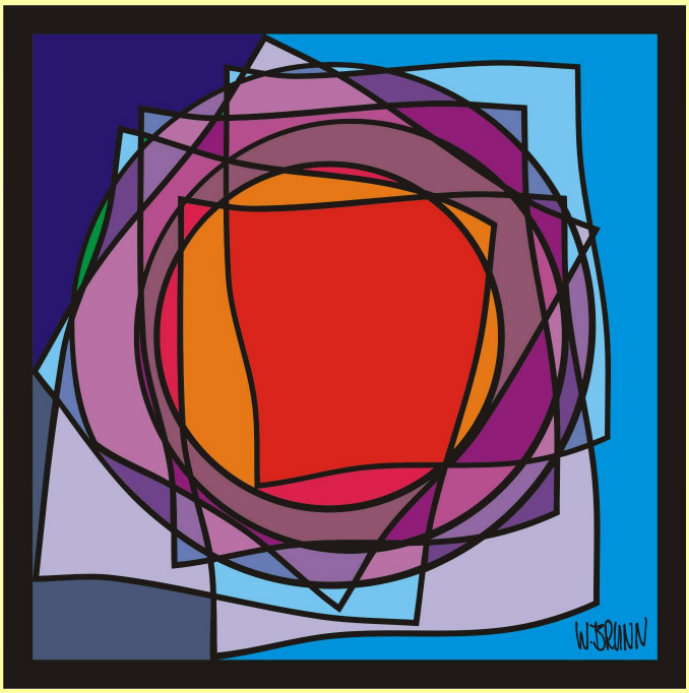
E-Learning mit w3l

- Einsatz der w3l-Plattform im E-Learning-Bereich
  - Kurs und Buch werden aus einer Quelle generiert
  - durchgehendes didaktisches Konzept
  - Kurselemente werden von derselben Satzmaschine erstellt
    - einheitliches Layout
    - keine reine »Dokumentensammlung«
- E-Learning erfordert weitere Komponenten
  - Unterstützung von Tutoren über die Plattform
  - Konzept für die Bearbeitung von Übungsaufgaben, Online-Klausuren usw.
  - virtuelles Klassenzimmer.

Single Source Publishing – Überblick und Konzepte

SEI 15

Vielen Dank!



Single Source Publishing

Überblick und Konzepte

Peter Siepermann
w3l GmbH
Witten
07.04.2008

Abstract

Single Source Publishing beschreibt eine Menge von Konzepten und Methoden, um aus einer Quelle mehrere Ausgabeformate zu generieren. Erste Anwendungen waren Textdokumente, mit deren Hilfe man sowohl die Druckausgabe als auch eine Online-Version generieren konnte.

Im Laufe der Zeit wandelte sich das Single Source Publishing dahingehend, dass nicht mehr vollständige Textdokumente als Ausgangspunkt verwendet wurden, sondern Textmodule in einer Datenbank erfasst wurden. Diese Textmodule lassen sich mit Metainformationen anreichern, sodass sie einfach wiederverwendet werden können.

Am Beispiel der Anwendung *w3l* sollen die Konzepte und die Auswirkungen des Single Source Publishing demonstriert werden.

Single Source Publishing auf der Ebene von Dokumenten

Die ursprüngliche Bedeutung des *Single Source Publishing* besteht darin, dass ein Textdokument sowohl auf einem Drucker als auch online zum Beispiel in Form von *WinHelp* oder *HTML* zur Verfügung gestellt werden soll. Dabei werden dann die Absatzformate 1:1 auf die Formatierungen im Online-Dokument übertragen. `Überschrift 1` wird zum Beispiel in HTML auf `<h1>` abgebildet, `Überschrift 2` auf `<h2>` usw. Abbildung 1 veranschaulicht noch einmal diesen Prozess.

Die Vorteile dieses Konzeptes liegen auf der Hand: Die Informationen werden nur einmal erfasst und liegen nicht mehr redundant vor. Das erspart nicht nur Aufwand bei der Erstellung der Dokumente, sondern auch bei deren Pflege. Es gibt zahlreiche Beispiele für ein solches Vorgehen:

- Doc-to-Help arbeitete bis ca. 2003 nach diesem Vorgehen
- Erstellung von WinHelp- und HTML-Dokumenten aus Word-Texten
- Erstellung von HTML-Dokumenten aus LaTeX-Dokumenten
- Erzeugung von Textdokumenten und HTML-Dokumenten aus Quelltexten mit Hilfe von Doxygen, Javadoc u.ä

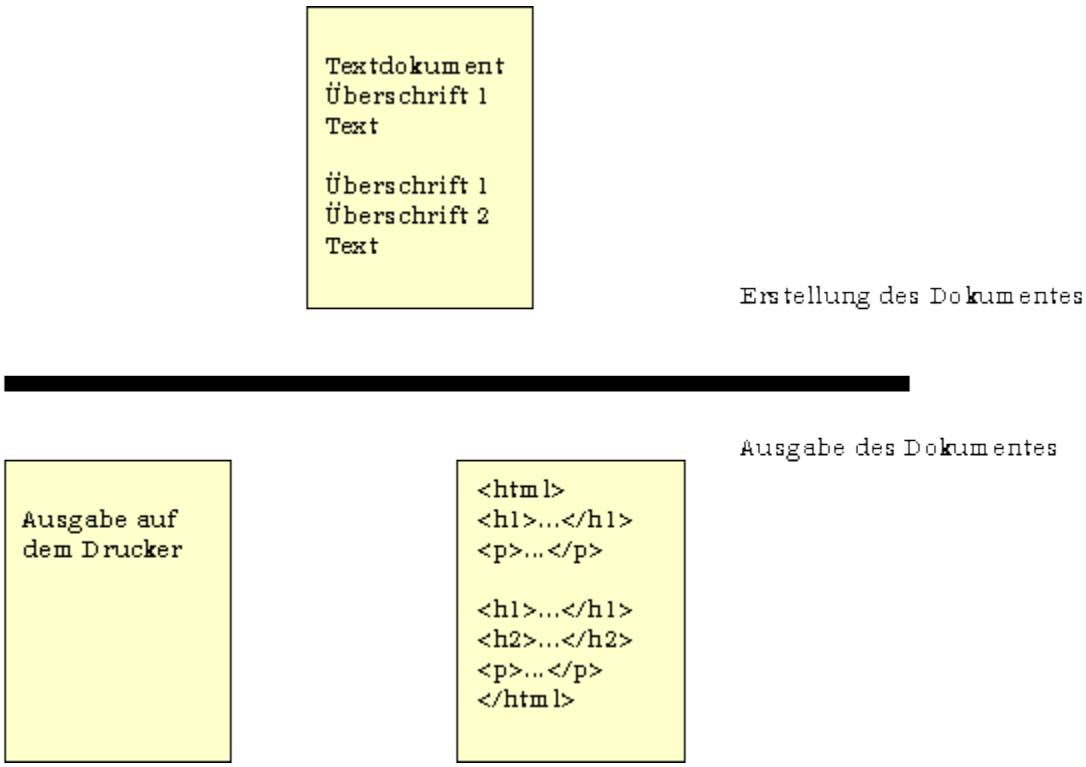


Abbildung 1: Single Source Publishing auf Dokumentenebene

Datenbank-basierte Verfahren

Die Idee, ein Textdokument in mehreren Medien zu veröffentlichen, lässt sich dahingehend weiterentwickeln, dass als Basis nicht ein vollständiges Dokument betrachtet wird, sondern eine Menge von Textbausteinen. Diese Bausteine werden zu einem Dokument kombiniert, das dann veröffentlicht wird.

Abbildung 2 stellt das Verfahren graphisch dar. Einzelne Textbausteine werden in einer medienneutralen Datenbank gespeichert. Die in der Regel proprietären Absatzformate in den Textbausteinen treten zugunsten einer XML-Beschreibungssprache zurück. Neben den eigentlichen Formatinformationen können dabei auch Metainformationen über den Inhalt abgelegt werden. Das eigentliche Dokument enthält dann nur noch Referenzen auf seine Textbausteine. Damit einher geht eine stärkere Wiederverwendung der einzelnen Module. Das Single Source Publishing erhält auf diese Weise Berührungsflächen mit Content Management-Systemen. Bei der Wiederverwendung muss man allerdings bedenken, dass gegebenenfalls verschiedene Autoren Rechte an den einzelnen Modulen haben. Außerdem dürfen sich die Module inhaltlich nicht überschneiden.

Mit Hilfe eines Konfigurators werden die benötigten Textbausteine zum gewünschten Zieldokument zusammengesetzt, das dann zum Beispiel als Buch oder auch als Online-Dokument publiziert werden kann.

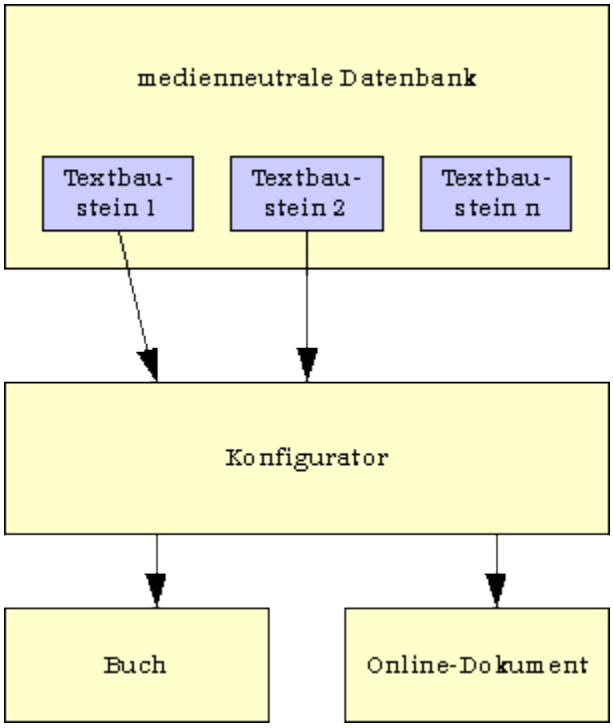


Abbildung 2: Datenbank-basierte Verfahren

Bewertung

Single Source Publishing stellt ein wichtiges Hilfsmittel zur Verfügung, um Redundanzen bei der Bereitstellung von Informationen zu vermeiden. Einmal erstellte Informationen können beliebig oft wiederverwendet werden. Dadurch ergibt sich auf der Erzeugerseite ein deutlich reduzierter Aufwand gegenüber herkömmlichen Verfahren. Auch für den Nutzer ergeben sich hieraus Vorteile, da zum Beispiel bei technischen Dokumentationen die gedruckten Fassungen mit den Online-Versionen synchron laufen.

Dem gegenüber stehen auch einige Nachteile: In der Regel ist die Anschaffung eines Single Source Publishing-Systems mit hohen Kosten verbunden. Neben den Kosten für das eigentliche System sind auch die Kosten für die Administration und die Schulung / Einarbeitung zu berücksichtigen. Schwieriger wird auch die Erstellung der Texte selbst. Da diese wiederverwendet werden sollen, müssen sie modularer aufgebaut sein. Außerdem sollte sichergestellt werden, dass der Schreibstil einheitlich bleibt. Häufig kann daher die Wiederverwendung nur bei ein und demselben Autor eingesetzt werden.

Der Aufwand, Texte mit Metainformationen auszustatten, sollte nicht unterschätzt werden. Zusätzlich zum eigentlichen Text werden in der Regel noch Kurzbeschreibungen, Schlagworte usw. erwartet.

Das Fallbeispiel w3l

Am Beispiel der w3l-Satzmaschine soll das Single Source Publishing verdeutlicht werden. Die Autoren erstellen die Informationen in modularen Textbausteinen, die mit Metainformationen angereichert werden. Diese können im Falle eines E-Learning-Kurses zum Beispiel in der Angabe des Schwierigkeitsgrades bestehen. Die Eingabe kann im Fall von w3l entweder über einen eigens zur Verfügung gestellten Windows-Client geschehen, oder aber über einen LaTeX- bzw. Word-Importer. Dieses Verfahren hat den Vorteil, dass der Autor seine gewohnte Umgebung nicht verlassen muss.

Nach der Erstellung der Texte werden diese von einem Lektor bearbeitet. Der Lektor arbeitet ebenfalls mit dem w3l-System. Das hat den Vorteil, dass keine externe Versionsverwaltung auf den Texten durchgeführt werden muss. Die aktuellen Versionen werden vom w3l-System verwaltet, und alle Personen arbeiten auf denselben Daten. Im Herstellungsprozess wird aus den einzelnen Dokumenten das Zieldokument zusammengestellt.

Aus dem Dokument kann anschließend ein Buch generiert werden. Dies geschieht im Fall von w3l mit Hilfe von LaTeX. Außerdem können Online-Darstellungen erzeugt werden. Dazu werden aus den Textbausteinen mit Hilfe von XSLT-Transformationen JSPs generiert. Abbildung 3 zeigt diesen Prozess schematisch.

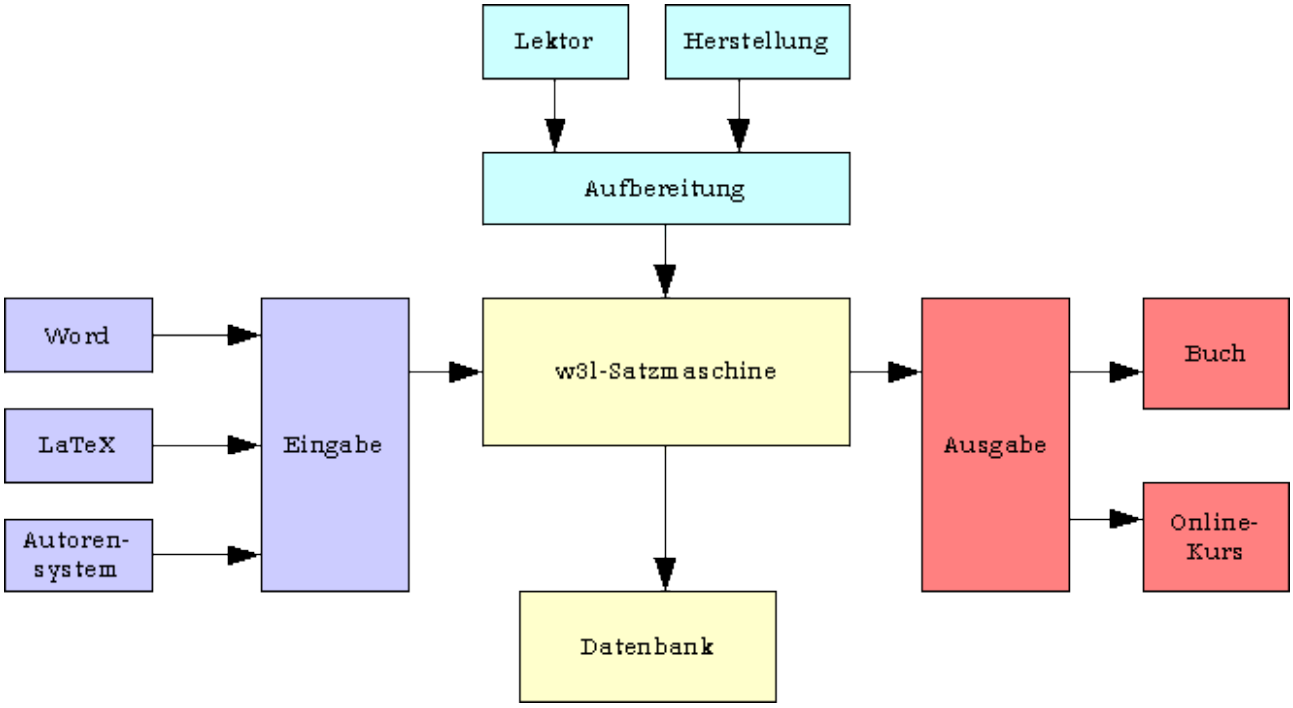


Abbildung 3: Schematische Darstellung der w3l-Satzmaschine

Stärken zeigt w3l vor allem im Bereich von E-Learning-Kursen. Viele E-Learning-Systeme stellen auf der Dokumentenseite lediglich eine Sammlung einzelner Textdokumente (zum Beispiel Word-Dokumente) zur Verfügung. Damit ist die Gestaltung der Inhalte einzig den Autoren überlassen was dazu führen kann, dass ein und derselbe Kurs mit völlig verschiedenen Konzepten und Layouts erscheint. Da im Fall von w3l die Dokumente mit dem eigenen Autorensystem erstellt werden, ist so innerhalb eines Kurses ein einheitliches Erscheinungsbild deutlich eher sichergestellt.

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

Prozessautomatisierung des Versuchsstandes TOPFLOW des Forschungszentrums Dresden Rossendorf

Dr. Ing. habil. H. Carl FZD
Matthias Beyer FZD

Gliederung

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

- Der Versuchstand TOPFLOW (**T**ransient Tw**o** **P**hase **Flow** Test Facility
- Instrumentierung des Versuchstandes Automatisierungslösung Ausbau seit 2002
- Zeitverhalten der Prozessdatenverarbeitung unter den Bedingungen der OPC Kopplung und Ethernet-kommunikation
- Instrumentierung – Softwaremodul SAAS MSuE (**M**assen**s**trom **u**nd **E**nthalpieberechnung)

Dr. Ing. habil. H. Carl FZD
Matthias Beyer FZD

Anlage TOPFLOW im Forschungszentrum Dresden – Rossendorf (FZD)

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

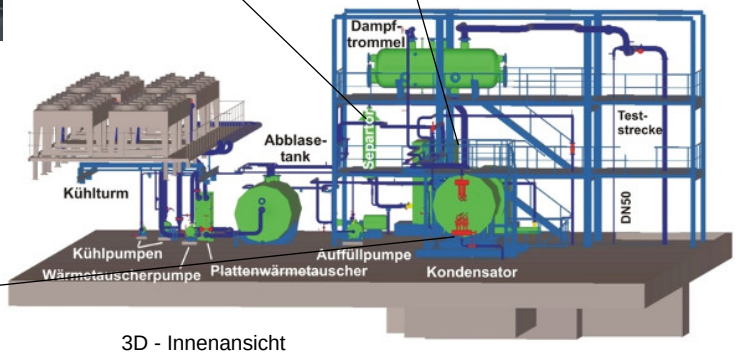
Außenansicht

Separator

Elektrokessel

Drucktank mit horizontalem Versuchsaufbau

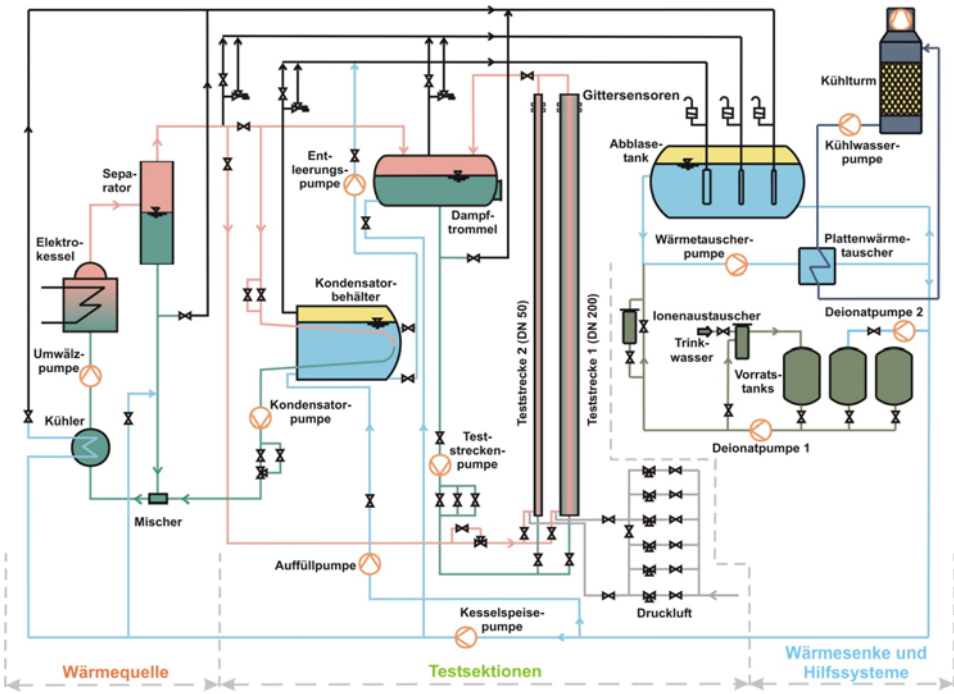
Kondensator



3D - Innenansicht

Hauptkomponenten Anlage TOPFLOW

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



Entwicklungsstufen des Automatisierungssystems

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

Planungsbeginn : Juni 2000

Maßgaben: ca.300 analoge Prozessvariablen, ca. 500 binäre Prozessvariablen
Aktualisierungszeit im PLS < 1s

Messunsicherheit Temperatur :0,5% vom Messbereich

Messunsicherheit Massendurchfluss :2% vom Messbereich eines Messkanals

Leichte Nachrüstbarkeit einzelner Antriebe und Messstellen

Aktualisierbarkeit der Dokumentation entsprechend der Qualitätssicherung des Institutes für Sicherheitsforschung

Systementscheidung:

SPS Ebene: Fabrikat PhoenixContact Interbus Inline

PLS Ebene: Fabrikat Wonderware Intouch, Kopplung über OPC

Feldbus: Interbus-S Fernbus

Druck- , Differenzdruckmesstechnik: Fabrikat Fischer-Rosemount

Antriebstechnik: EMG mit Interbus-S

Fertigstellung: August 2002

Sensordatenbank und Loopdokumentation

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

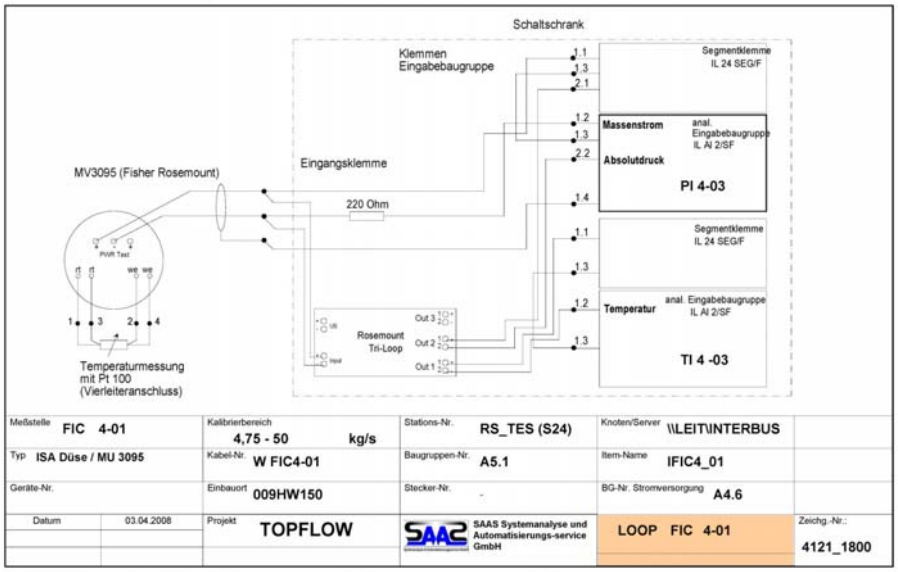
Basis: MS Access

Enthält die wesentlichen elektrischen und konstruktiven Parameter

Der Einbauort und die Beziehungen zu anderen Messstellen sind erkennbar.

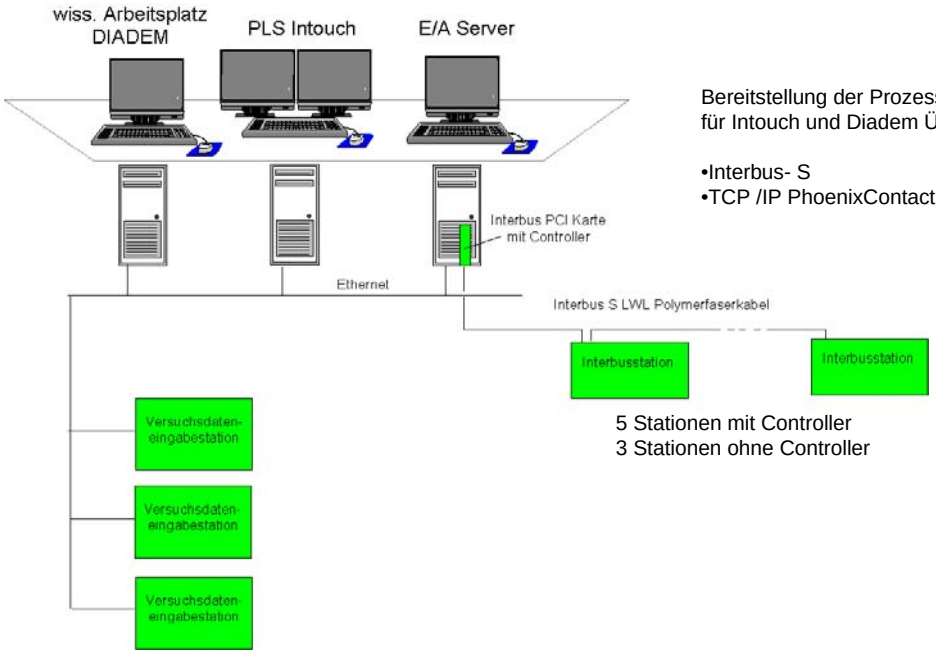
Datenbankbericht Loopzeichnung

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



Systemkonfiguration 2002

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



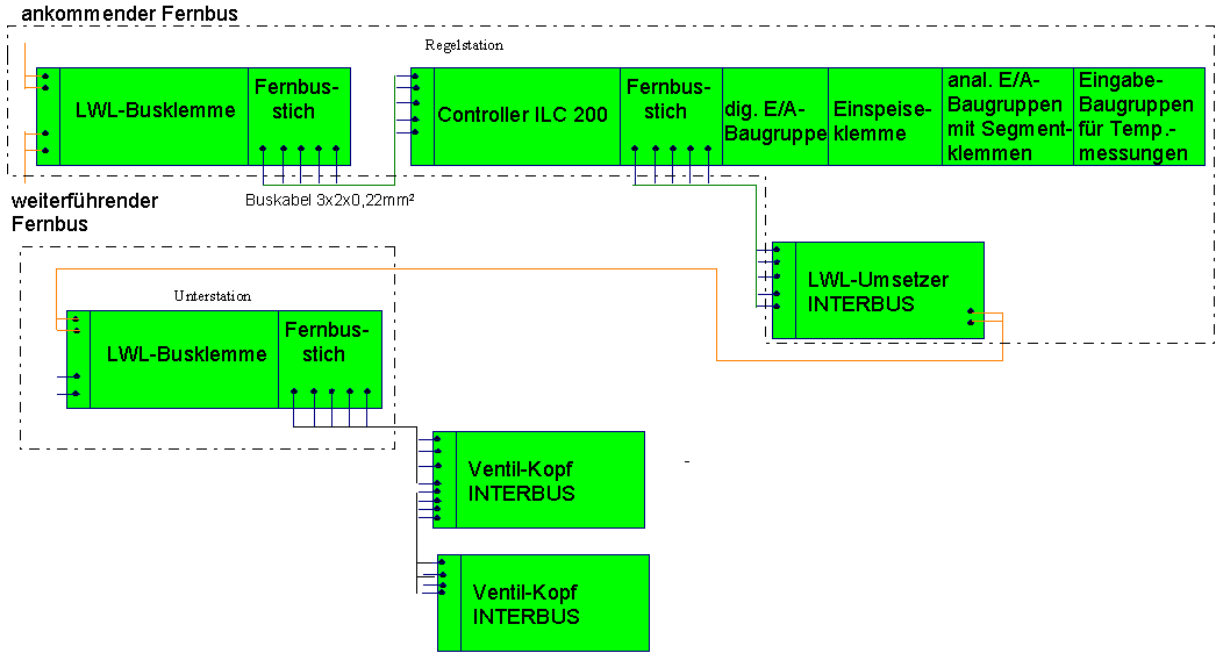
Bereitstellung der Prozessdatenkopplung
für Intouch und Diadem Über 2 OPC Server

•Interbus- S
•TCP /IP PhoenixContact Ethernetklemme

5 Stationen mit Controller
3 Stationen ohne Controller

Typische Konfiguration einer Interbusstation

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



Erweiterungen des Systems bis 2008

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

Maßgaben: ca.450 analoge Prozessvariablen, ca. 700 binäre Prozesvariablen
Aktualisierungszeit im PLS < 1s

Modernisierung des Automatisierungssystems
mit Baugruppen der Firma Phoenix Contact:

Ersatz der SPS Controller ILC 200 mit Interbus-S durch ILC 350 mit Ethernetkopplung

Erweiterung der Anlage um einen Profibusstich zur Anbindung einer Stickstoffvergasungsanlage

Neue Leistungsmerkmale:

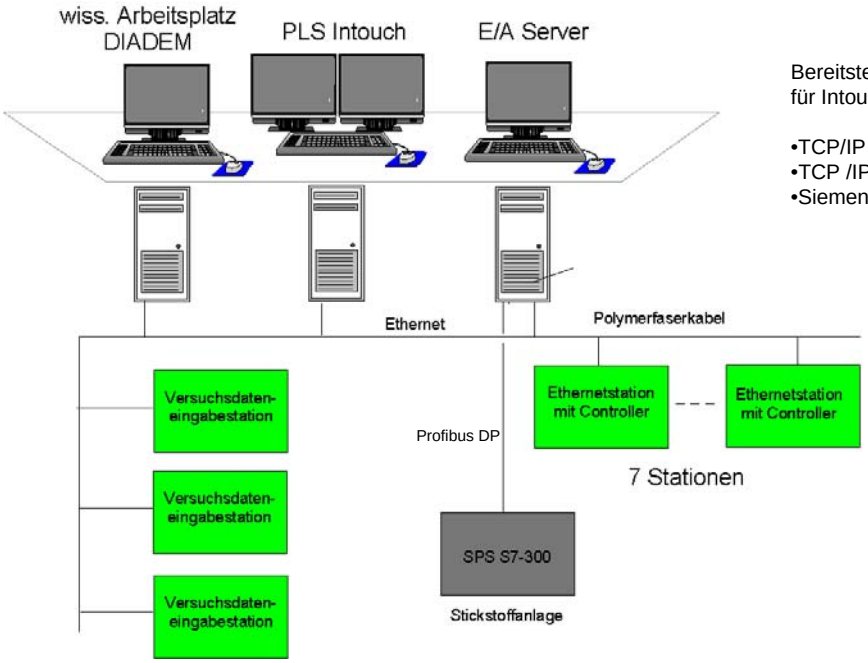
Aktualisierungszeit von Variablen im PLS bei ca. 300 ms

Querkommunikation zwischen den SPS Einheiten

Möglichkeit der Datenarchivierung in Datenbanken direkt aus der SPS heraus.

Konfiguration Automatisierungssystem TOPFLOW 2008

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



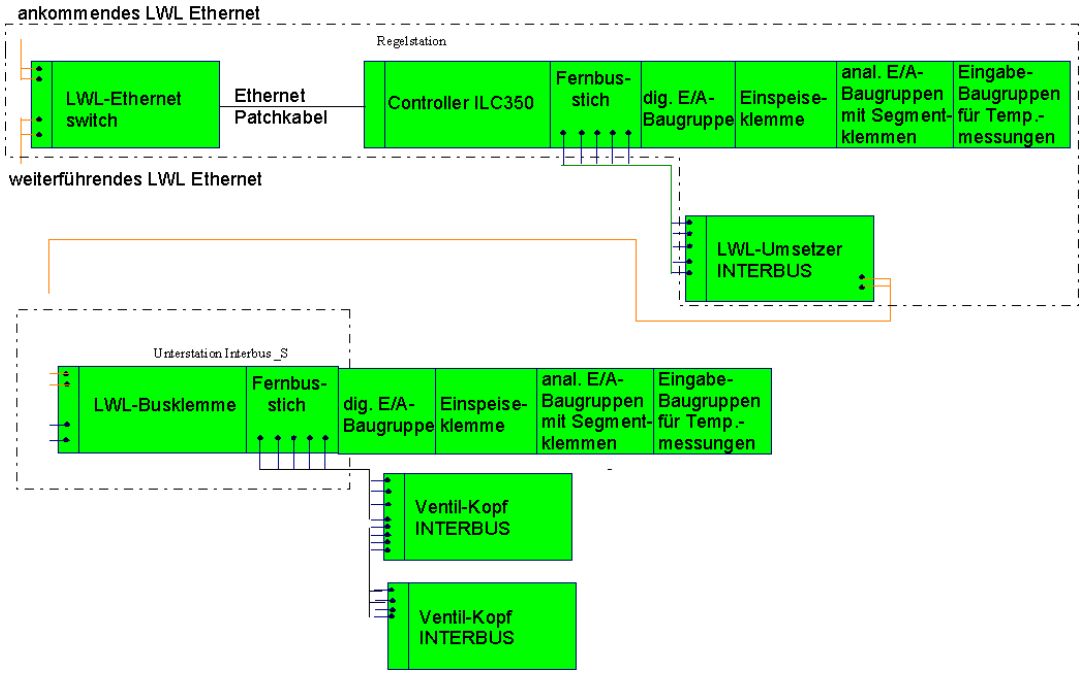
Bereitstellung der Prozessdatenkopplung
für Intouch und Diadem Über 3 OPC Server

- TCP/IP AX Server PhoenixContact
- TCP /IP PhoenixContact Ethernetklemme
- Siemens S7 OPC Server

Dr. Ing. habil. H. Carl FZD
Matthias Beyer FZD

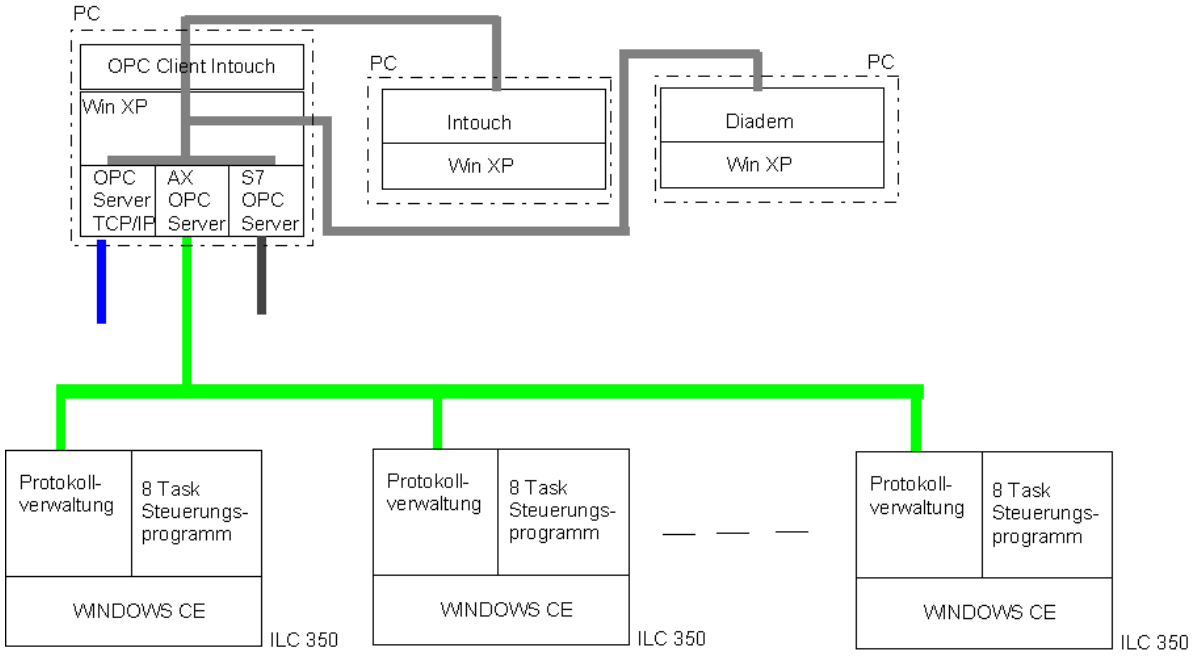
Typische Konfiguration einer Ethernet SPS Station

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



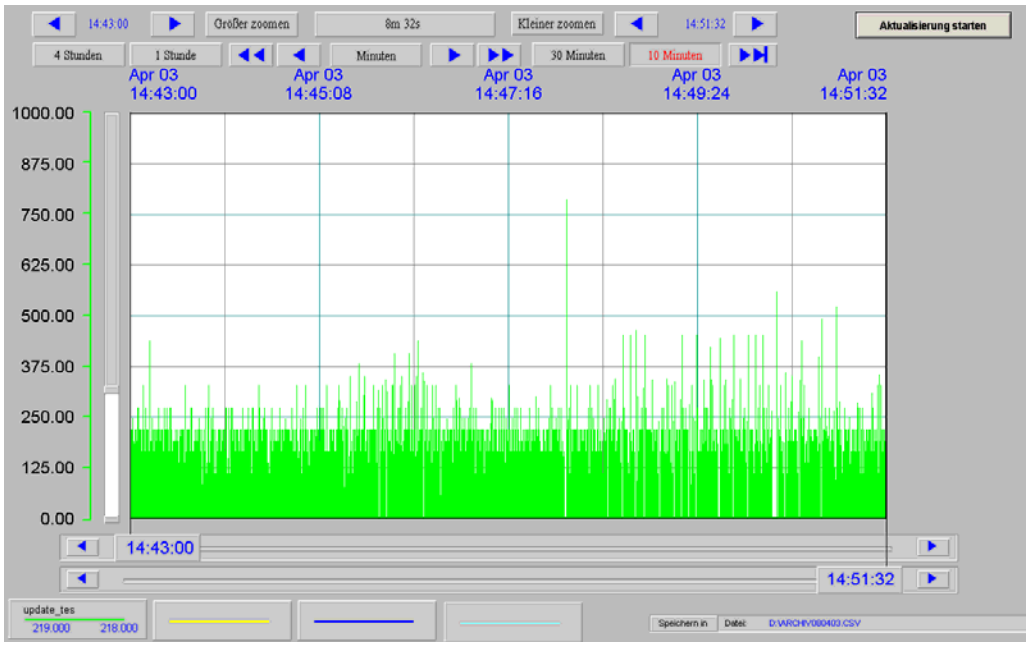
Aktuelle Softwarestruktur des Automatisierungssystems

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



Messung der Updatezeit des Prozessleitsystems

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de



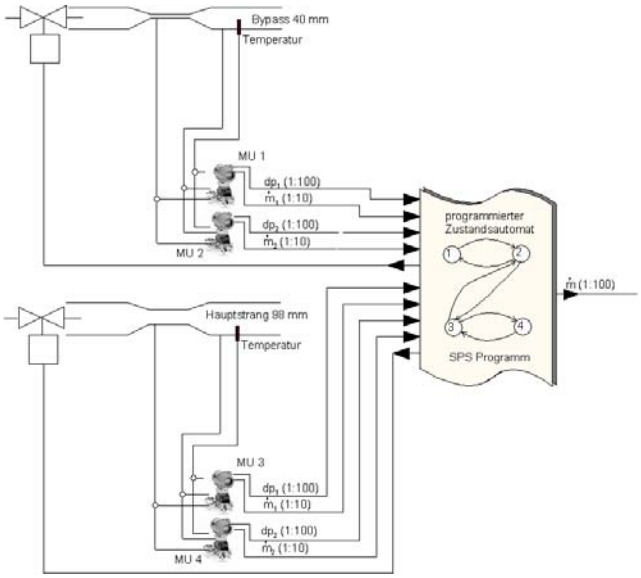
Schwerpunkte der Instrumentierung

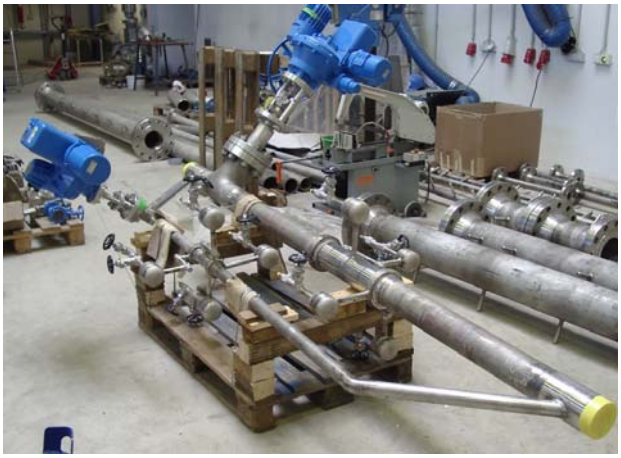
SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

- Durchflussmessungen: Wirkdruckprinzip / Vortex
  Messspanne: bis 1:10^4
- Temperatur: bis 286 °C
- Absolutdruck: bis 7,0 MPa
- Differenzdruck: bis 2000 kPa

Beispiel für eine Massenstrommessstelle

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de





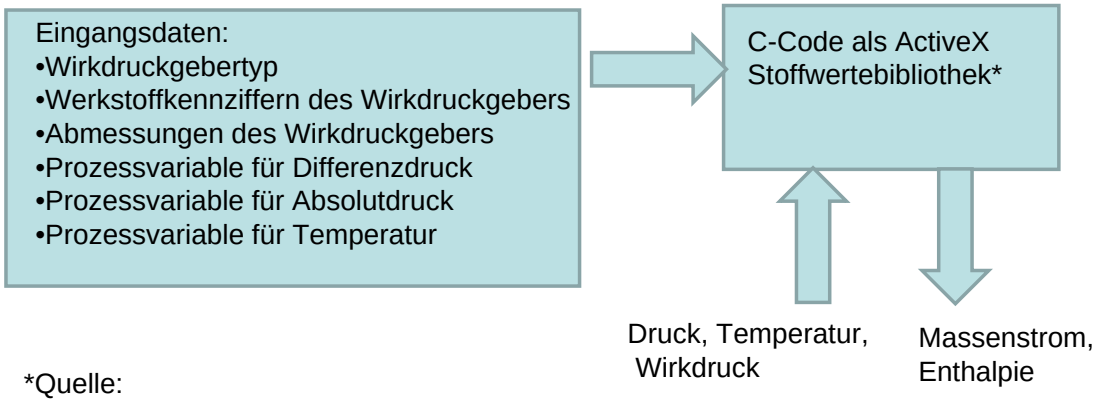
ActiveX Modul
MSuE für Windowsbasierte PLS

SAAS GmbH
Rippiener Str. 1
01728 Bannewitz
Dr. S. Lenk
www.saas-online.de

Idee:
Echtzeitberechnung von Stoffdaten erspart teure Instrumentierung

Projektierung:

PLS Laufzeitsystem:



*Quelle:
IH Zittau Görlitz 2006, Stoffwertebibliothek für Wasser und Dampf

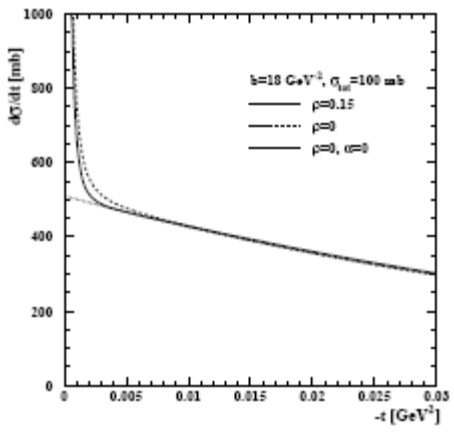
ALFA – Ein Luminositätszähler für ATLAS

Dieter Notz
Deutsches Elektronen-Synchrotron, DESY, Hamburg, Germany
EMAIL: Dieter.Notz@desy.de
SEI Tagung FZK Karlsruhe, 7. 4. – 9. 4. 2008

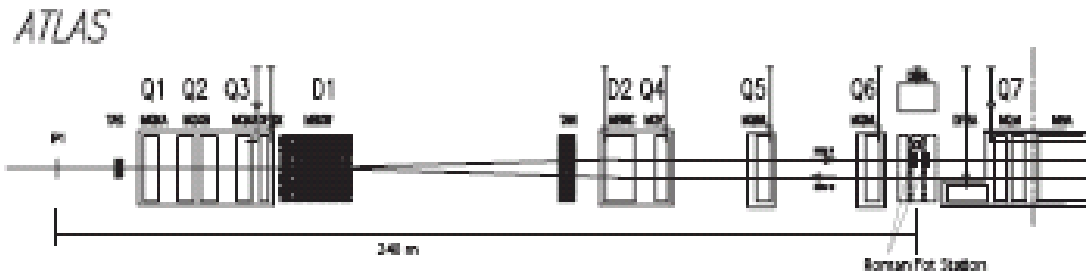
Zusammenfassung

Bei einem Speicherring treffen Teilchenpakete aufeinander. Die Ereignisrate bei physikalischen Reaktionen ist abhängig vom Wirkungsquerschnitt der Reaktion und von der Luminosität der Strahlen. Zur Messung der Luminosität misst man die Ereignisrate für eine bekannte und berechenbare Reaktion.

Die Designluminosität beim LHC (Large Hadron Collider) am CERN beträgt etwa 10^{34}cm^{-2} s^{-1}. Der Wirkungsquerschnitt einer Reaktion hat die Dimension einer Fläche. In der Hochenergiephysik benutzt man dafür die Einheit barn = 10^{-24} cm^2. Hätte also eine Reaktion den Wirkungsquerschnitt von 1 barn, so hätte man bei LHC 10^{10} Ereignisse/s. Die Luminosität steigt mit der Zahl der Teilchen pro Strahlpaket (Bunche), der Häufigkeit (Frequenz), mit der die Bunche aufeinandertreffen und dem Kehrwert des Strahlquerschnitts: Je besser man einen Sonnenstrahl mit einer guten Linse fokussieren kann, um so leichter fängt Papier an zu brennen; um so höher ist seine Luminosität.
In einem Hochenergiephysikexperiment werden neue Teilchen mit einer bestimmten Rate erzeugt. Um im Experiment ATLAS beim LHC die Wirkungsquerschnitte unterschiedlicher Prozesse messen zu können, benutzt man einen Luminositätsmonitor genannt LUCID. Dieser zählt die Zahl von Teilchen, die ihn pro Zeiteinheit treffen. Durch Vergleich der Raten von neuen Ereignissen mit der Rate des 17 m entfernten LUCID Detektors bekommt man das relative Verhältnis der Wirkungsquerschnitte.
Mit dem hier vorgestellten Luminositätsdetektor misst man die Rate der elastische gestreuten Protonen. Da man den elastischen Wirkungsquerschnitt für Proton-Proton-Streuung berechnen kann, hat man die Möglichkeit, die Luminosität absolut zu bestimmen. Daher kommt der Name ALFA = *Absolute luminosity-measurement for ATLAS*.
Beim ALFA Detektor handelt es sich um einen Detektor, der in etwa 240 m Entfernung vom Wechselwirkungspunkt Protonen misst, die etwa 2 – 8 mm vom ungestreuten Strahl entfernt sind. Die Protonen haben also am Wechselwirkungspunkt einen Streuwinkel von 0.002m/240m = 0.001 mrad erfahren. Man stellt also in 240 m Entfernung einen ortauflösenden Detektor auf, der auf etwa 1.5 mm Abstand an den Strahl herangefahren werden kann.



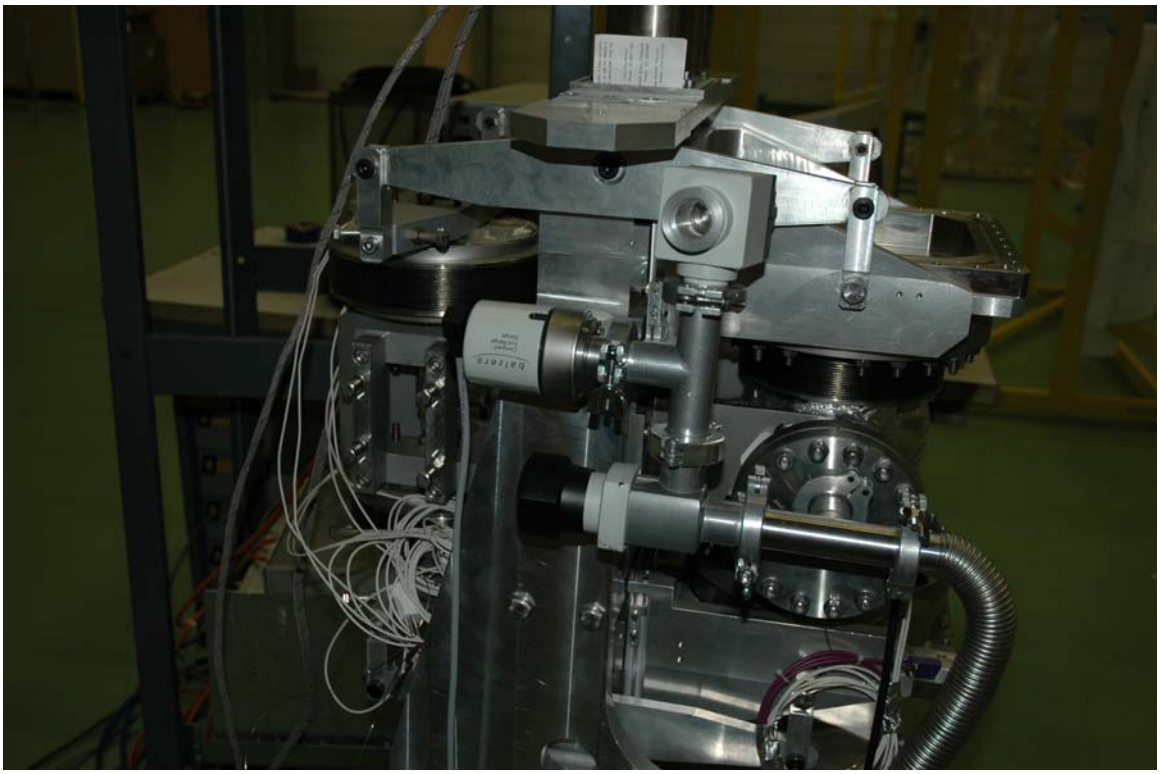
Die Abszisse -t ist ein Maß für den Winkel. Die Ordinate zeigt den Wirkungsquerschnitt. Für kleine t ist der Wirkungsquerschnitt für elastisch gestreute Protonen aufgetragen.



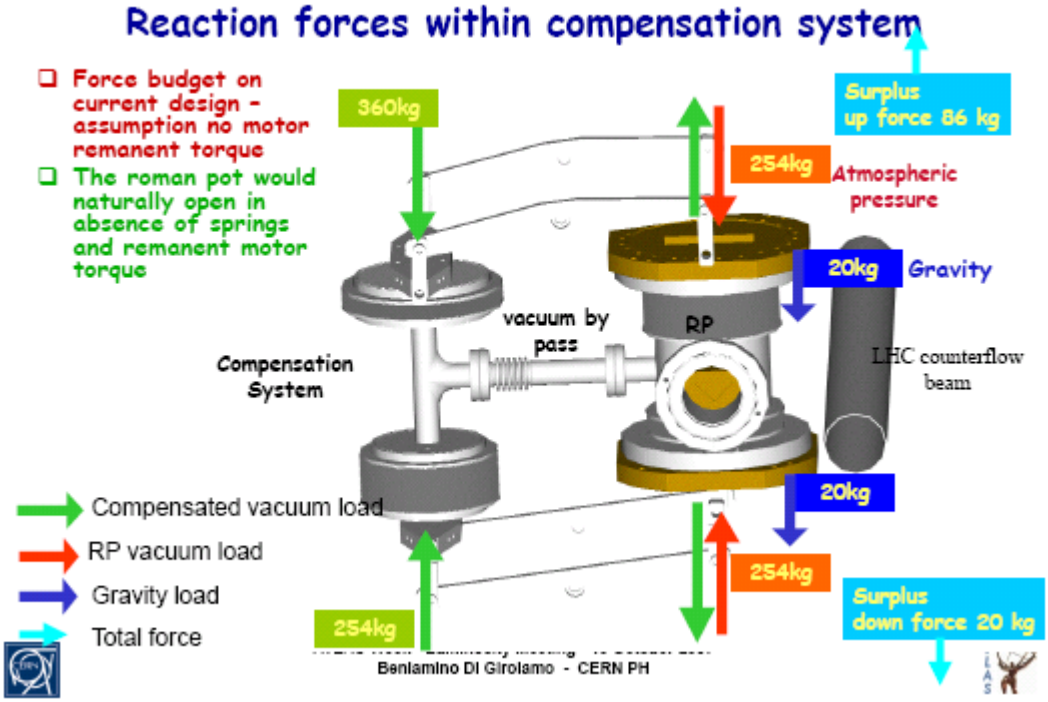
Links im Bild ist der Wechselwirkungspunkt des ATLAS Experiments. 17 m davon entfernt befindet sich der LUCID Detektor. Q und D stellen Quadrupole und Dipolmagnete dar, die für den Strahltransport zuständig sind. Nach etwa 240 m hat man ein freies Stück. Hier wird der ALFA Detektor installiert.

Um mit dem ALFA Detektor den LUCID Detektor eichen zu können, müssen die Strahlen in LHC möglichst parallel laufen. Für den normalen Physikbetrieb ist man aber an stark fokussierten Strahlen interessiert. Um also eine Eichung vornehmen zu können, muss die Strahloptik in LHC geändert werden. Hierzu wird einer der Quadrupole umgepolt. Die Strahlen laufen dann weitgehend parallel, so dass man mit LUCID und ALFA gleichzeitig die selben Reaktionen und Wirkungsquerschnitte messen kann.

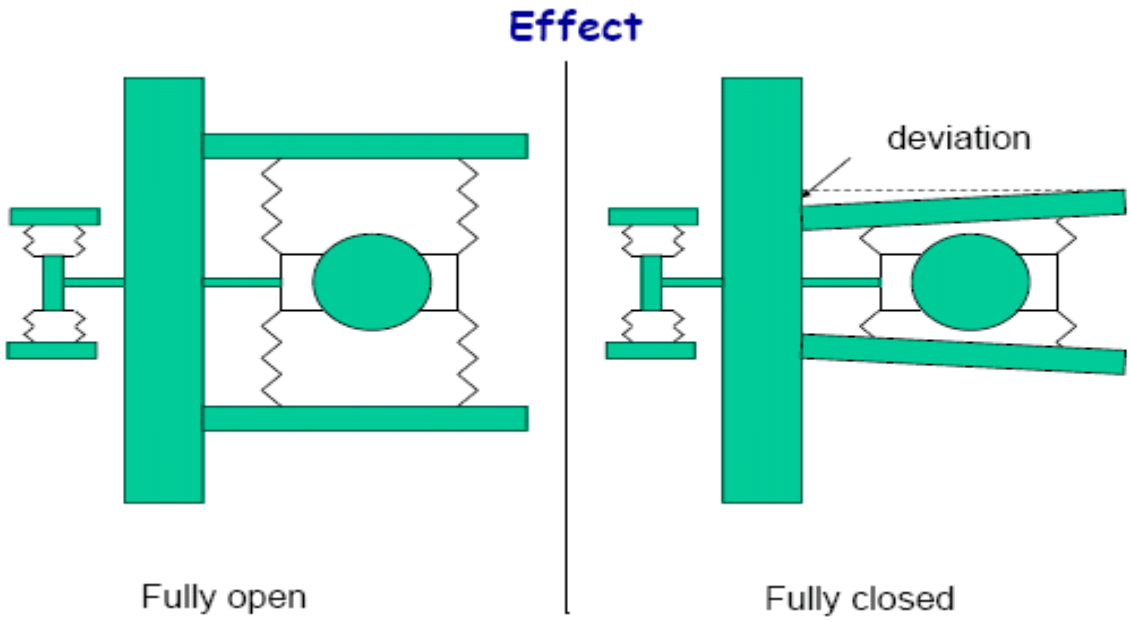
Die Apparate, die den Detektor am Strahlrohr beherbergen, heißen "Roman" Pot, genannt nach einer Gruppe aus Rom, die solche Geräte das erste mal verwendete.



Das Bild zeigt einen Roman Pot, wie er am Strahlrohr angebracht wird. Eine Station beinhaltet zwei Detektoren: Einen von oben und einen von unten. Im linken Teil ist ein zweites Vakuumsystem installiert. Dies dient dazu, dass der Detektor auch in Notsituationen ohne elektrisch Strom in eine offene Position fährt.



Sicherheitsvakuum im linken Teil des Bildes.

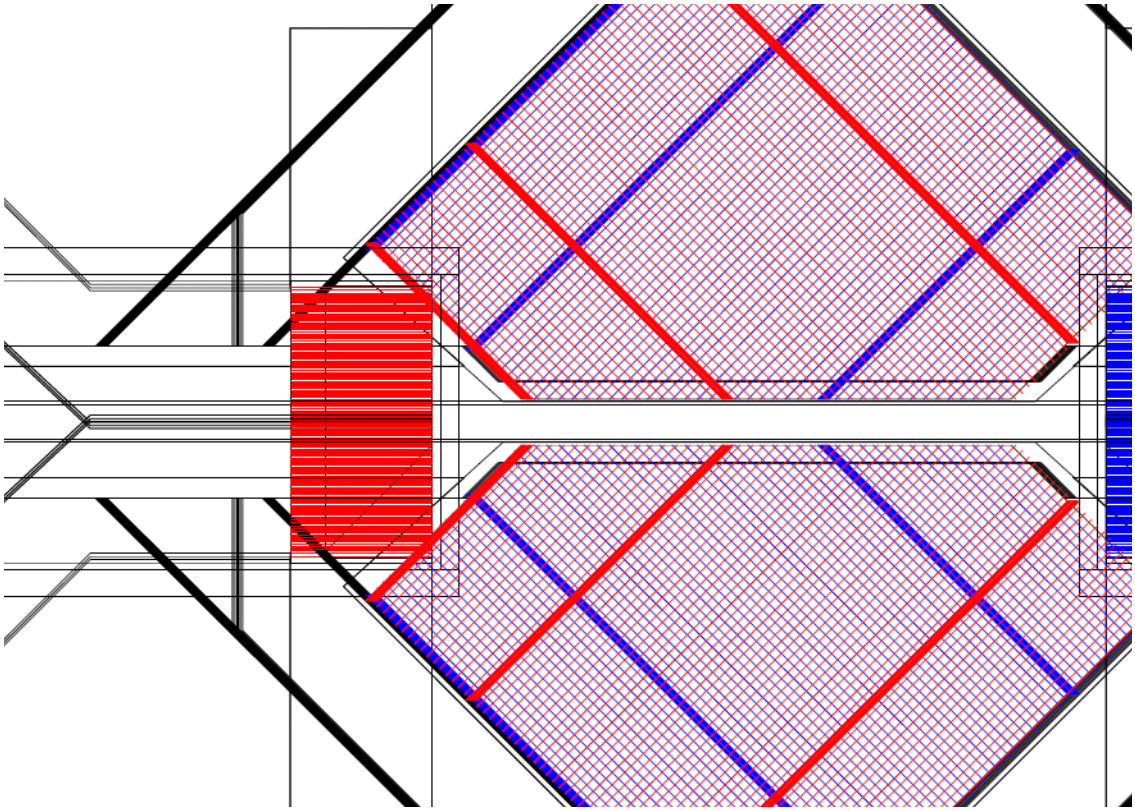


Beim Schließen können sich die Flansche leicht verbiegen.

Es ist eine Forderung der Maschinengruppe, dass die Detektoren in Notsituationen oder bei Stromausfall sofort in ihre Parkposition fahren. Zu diesem Zweck wird ein zweites

Vakuumsystem wie an einer Balkenwaage installiert, das die Kräfte des Detektorvakuums (über-)kompensiert.
Wahrend des Strahleinschusses in den Speicherring sind die Detektoren in Parkposition weit weg von Strahl. Für die Luminositätsmessung werden die Detektoren dann an den Strahl herangefahren. Es hat sich gezeigt, dass sich dabei die Flansche etwas verziehen. Die Detektoren befinden sich ein einem Vakuumbehälter, der vom Strahlvakuum durch ein 0.2 mm dickes Fenster separiert ist.

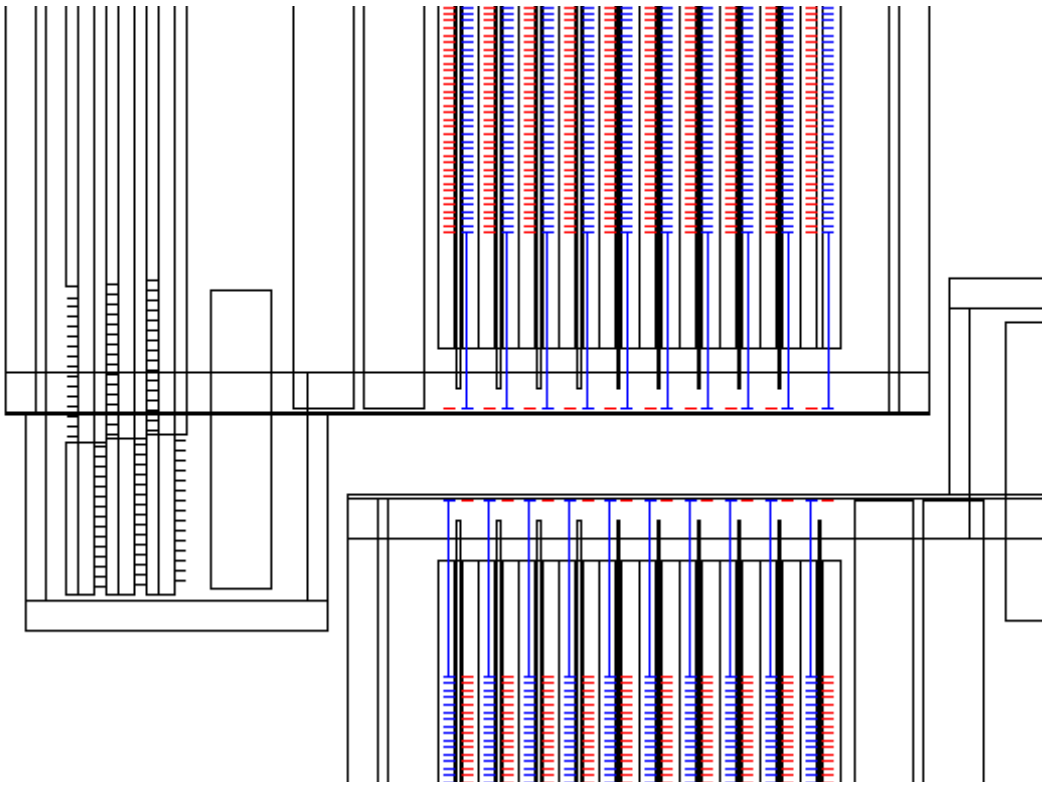
Roman Pots
Die Roman Pots bestehen aus Edelstahl. Sie sind mit dem oberen Flansch verbunden und können an diesem in der Höhe bewegt werden. Der Flansch beinhaltet die Photomultiplier und Anschlüsse an die Elektronik. Im Zwischenbereich werden die Szitillationsfiber und die Lichtleiter geführt. Im unteren Bereich befind sich in der Mitte der Detektorbereich mit 64 um 45 Grad abgewinkelten Szintillationsfibern. Zwei weitere Stutzen rechts und links des Strahles bieten Platz für den Overlapdetector. Die Overlapdetectoren bieten die Möglichkeit, dieselben gestreuten Protonen mit dem oberen und unteren Detektor messen zu können. Auf diese Weise kann man den Abstand der Detektoren recht genau bestimmen.



Oberer und unterer Detektor (Mitte) mit u-(blau) und v-(rot) Koordinate. In schwarz-weiß Darstellung ist u im oberen Detektor von oben links nach unten rechts orientiert.

Detektoren
Jeder Roman Pot enthält zehn Ti- Platten, ca. 2 mm dick. Diese sind abgefräst und tragen je Koordinate 64 Szintillationsfiber unter einem Winkel von 45 Grad (u- und v- Koordinate). Die

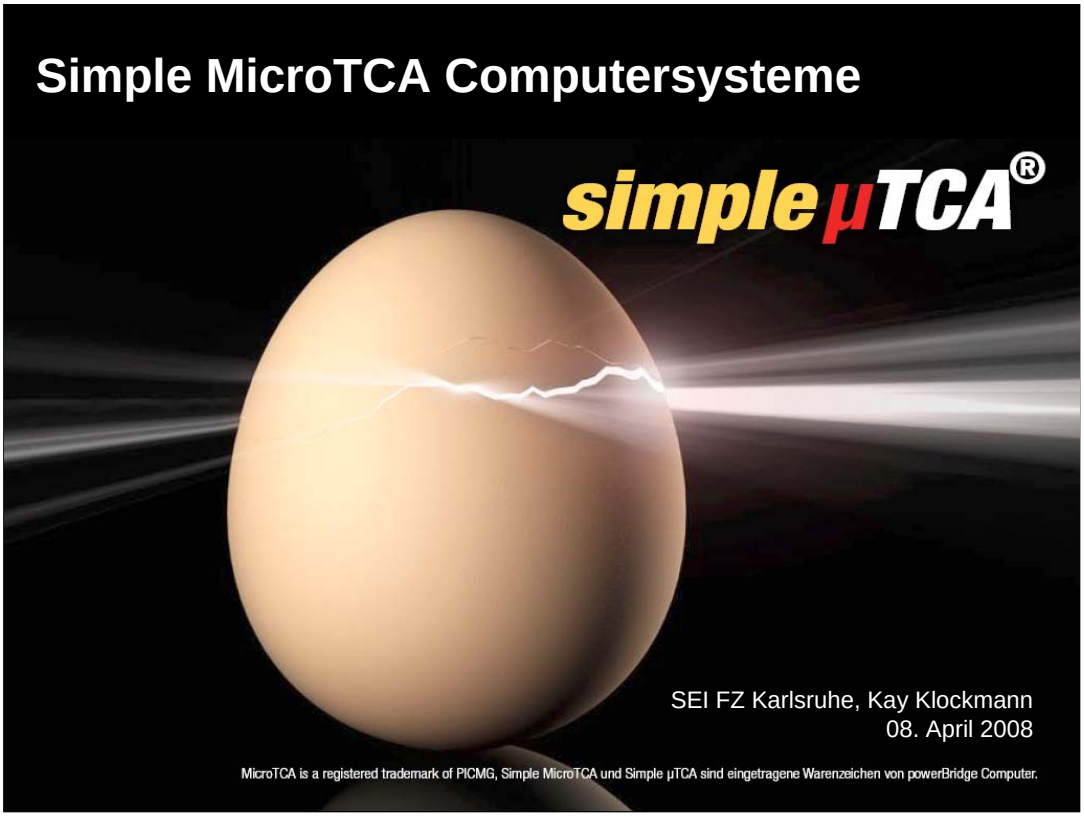
Fiber sind 0.5 mm dick und aluminisiert. Sie werden durch einen Adhäsionsprozess auf die Ti-Flächen geklebt, anschließend unten auf der Strahlseite abgefräst und genau vermessen.
Die zehn Platten sind untereinander um einige Mikrometer versetzt, um eine höhere Auflösung zu erreichen.
Bei den Overlapdetektoren sind auf drei Ti-Platten je 30 Fiber je rechts und links vom Strahl horizontal angeordnet. Der Versatz der Fieber auf den einzelnen Platten beträgt 0.3 mm.



Der Detektor von der Seite gesehen. Der untere und obere Detektor sehen dieselben Teilchen.

Elektronik
Die Fiber enden in einer Hamamatsu Photoröhre mit 64 fast unabhängigen Multipliern. Nach Aufbereitung der Signale werden diese über etwa 300 m lange Kabel zur zentralen Elektronik transportiert und ausgelesen. Ein Problem stellt das Triggern da. Nach einer Wechselwirkung werden die Originalsignale gespeichert. Ein Triggersignal muss nach 1.8 µs erscheinen, um das Ereignis auszulesen. Diese Forderung kann vom ALFA Detektor nicht erfüllt werden. Nach einer Wechselwirkung müssen die Teilchen erst einmal. 240 m (800 ns) weit fliegen. Dann werden sie verarbeitet und wieder zurückgesand. Das ganze nimmt 1.9 µs in Anspruch. Es müssen extra Vorkehrungen getroffen werden, damit das Triggersignal noch akzeptiert wird.

Weiterführende Informationen
http://atlas.cern.ch



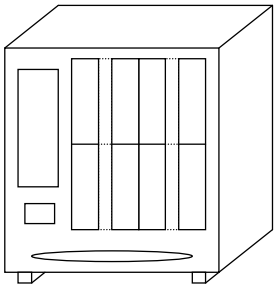
uTCA ist ein Industriestandard, der zunächst für Telekom Applikationen gedacht war. uTCA ist dabei nichts anderes als Mezzanines Boards (AMCs) auf eine Backplane zu stecken. Die Definition der Backplane sowie weitere Spezifikas sind in der uTCA Spezifikation beschreiben. uTCA ist damit ein sehr modulares, granulares System, welches aktuelle Bustechnologien unterstützt (u.a. wie PCI-Express), sowie weitere Vorteile in Form von Remote Parametrierung, Servicefreundlichkeit und Hot-Swap Funktionalität im Industrie Automationsumfeld ermöglicht. Wegen der grossen Akzeptanz in vielen Bereichen (abseits der Telekommunikation), ist uTCA die einzige Plattform im industriellen Umfeld, welche das Zeug hat, die nächsten 10 Jahre (und mehr) zur Verfügung zu stehen. Das Ecosystem von verfügbaren Produkten wächst dabei konstant, industrielles IO steht bereits heute breitbandig zur Verfügung (mit Umweg über IP TrägerAMC)

Computersysteme auf Basis von AMC-Karten

RackPak/SM2

Centellis-1000

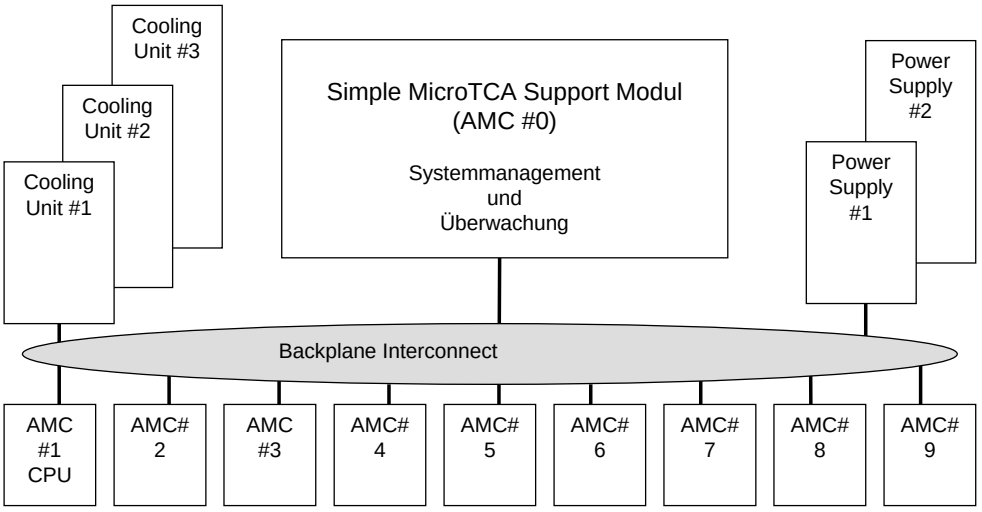
Centellis-500



MicroPak/6

powerBridge Computer

Aufbau eines Simple MicroTCA Systems



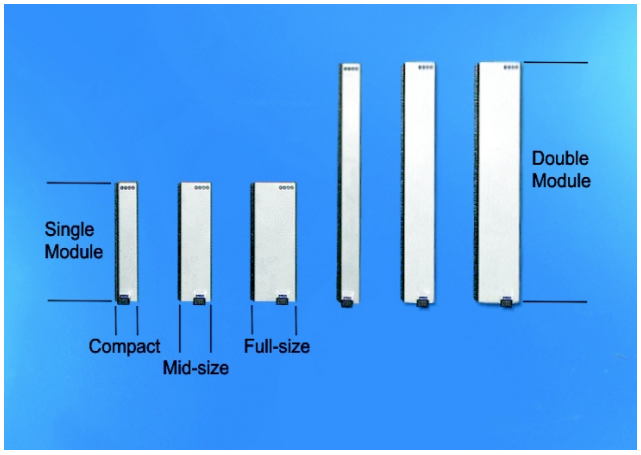
powerBridge Computer

Vergleich PCI(e) vs. AMC

	PCI	AMC
Bauform	Intern verbaut, Half-Size, Full-Size, Standard- oder Low-Profile	Frontseitig steckbar (Hot-Swap), Single & Double wide Compact, Mid- oder Full-size
Systemarchitektur	Single-Master	Multi-Master
Stecker	Direktsteckverbinder	Direktsteckverbinder oder industrietauglicher Stecker
Interconnect	PCI (66/64) PCI-X (133/64) PCI Express	20 serielle High-Speed-Ports: Ethernet (GbE, 10GbE, …) Fibre Channel, SATA, SAS, PCI Express, Serial RapidIO
IPMI	nein	ja
Hot Swap	Nicht verfügbar, Kartenwechsel ist ein Hauptproblem	ja
I/O	Front- oder internes I/O	Front- oder Rear-I/O
Verlustleistung	Kühlung ist ein Hauptproblem	20/40/80 Watt C/DC, MS/DMS, FS/DFS

Bauformen von AMC Modulen

- 6 unterschiedliche Bauformen verfügbar
  - Compact (C)
  - Double Compact (DC)
  - Mid-Size (MS)
  - Double Mid-Size (DMS)
  - Full-size (FS)
  - Double Full-size (DFS)
- Modultiefe 181,5 mm

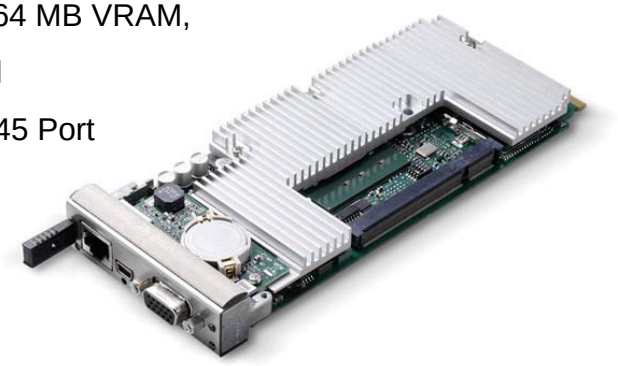


powerBridge Computer

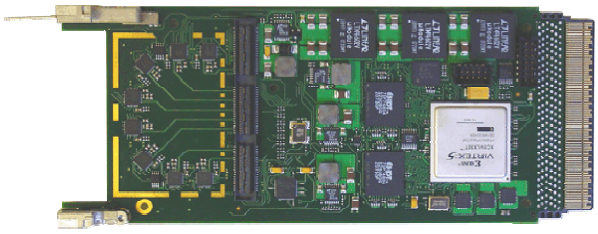
AMC Prozessorkarte

Adlink AMC-1000

- Intel Core 2 Duo Prozessor AMC, bis 2.16 GHz Takt, 667 MHz FSB
- Server Class Intel 3100 Chipsatz
- max. 2 GB ECC 400 MHz DDR2 RAM
- ATI ES1000 Grafikchipsatz mit 64 MB VRAM,
- Auflösung bis 1600 x 1200 Pixel
- VGA, USB 2.0 und serieller RJ-45 Port frontseitig
- 4GB Flash onboard
- -20°C .. +75°C Betriebs-temperaturbereich

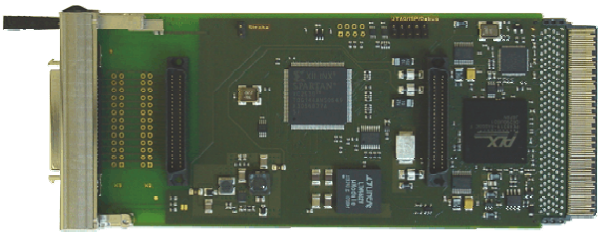


Schnittstellen



- Analog-I/O
- Carrier Boards
- Counter
- Digital-I/O
- Ethernet / Gigabit Ethernet
- Extender
- Feldbus Module
  (Arcnet, CAN, Interbus, LON, Sercos, ProfiBus …)

Schnittstellen

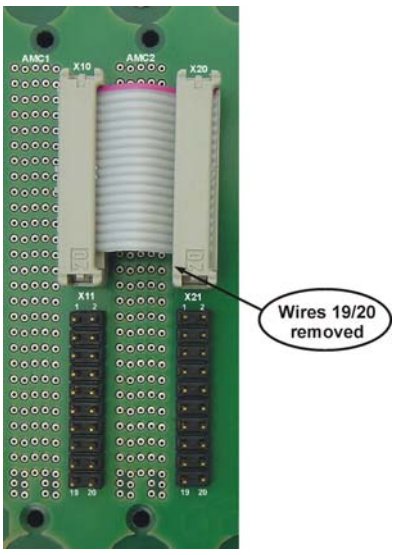


- FPGA-Module
- Motorcontroller
- Netzwerkprozessoren
- Prozessor Module (x86, PowerPC)
- serielle Ports
- Speichermodule
- Synchro/Resolver
- Telekommunikation (ATM, E1/T1, DSP)
- Video / Grafik

powerBridge Computer

Rear-I/O

- AMC Ports 12 bis 15 und 17 bis 20 stehen für Rear-I/O zur Verfügung
- Zwei 20-pol. Stiftleisten mit Dual GND, lokaler MP und PP
- Anwendung: Rear-I/O Inter-Slotverbindungen z.B.: TDM „Bus“, Taktverteilung, …)

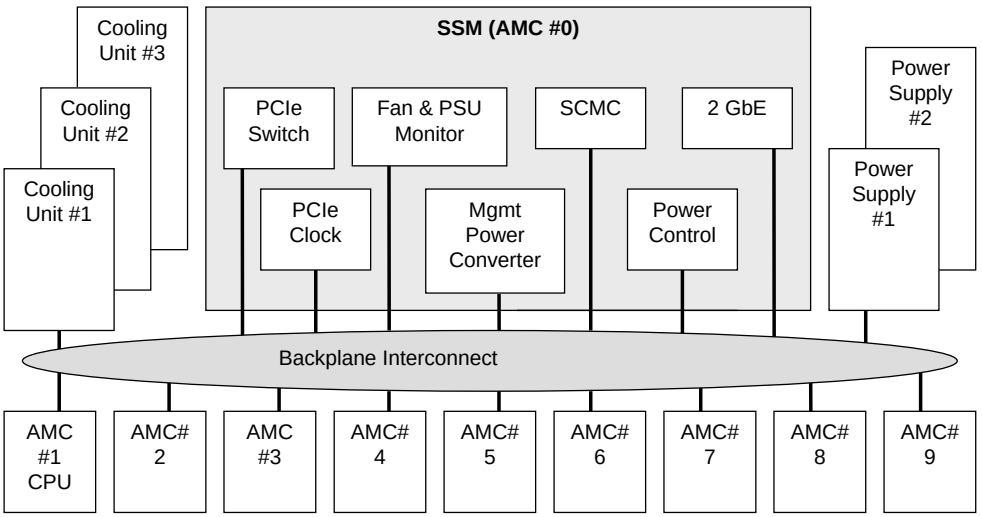


powerBridge Computer

Aufbau eines Simple MicroTCA Systems

powerBridge Computer

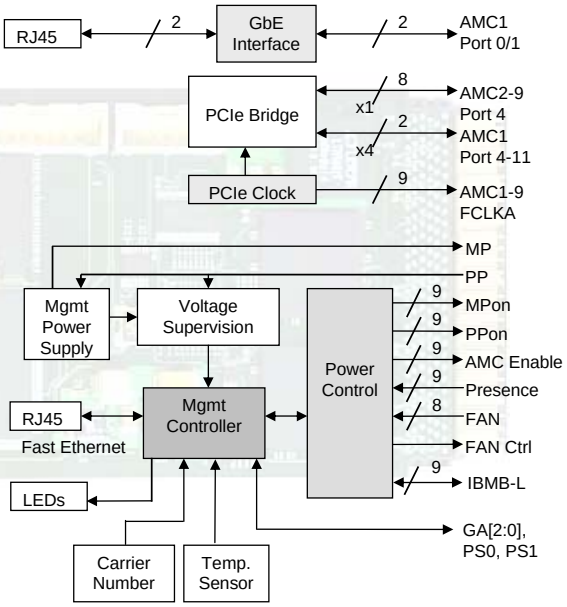
Blockdiagramm eines SMTCA Systems



powerBridge Computer

Simple MicroTCA Support Modul

- AMC mid-size Bauform
- PCIe Switch x8 zu 8 x1
- PCIe Takterzeugung
- DC/DC Konverter für Managementversorgung
- Power Control
- Überwachung für Lüftung und Betriebsspannungen
- Lüftersteuerung
- Management-Controller mit Ethernet Anschluss
- 2 GbE Anschlüsse für AMC Ports 0 und 1 der CPU



powerBridge Computer

System-Management

- Fernüberwachung des Systemzustandes: Temperatur, Lüftung, Stromversorgung, Kartenzustand
- Remote Firmware- oder BIOS-Update
- Remote Power Control: Steckplätze einzeln abschaltbar und/oder rücksetzbar
- Kontinuierliche Überwachung aller internen Temperaturmessstellen und Steuerung der Lüftung
- Leistungsmanagement: Überwachung des Leistungsbudgets
- Integrierbar in vorhandene Managementarchitekturen (SNMP)

powerBridge Computer

Spezifikation AMC Module

- AMC.0 (Advanced Mezzanine Module)
  - Defines a mezzanine building block approach for the addition of crucial functionality to a PICMG 3.0 carrier card available from a number of third party suppliers.
- AMC.1 (Advanced Mezzanine Module PCI Express and Advanced Switching)
  - Defines port usage for PCI Express and Advanced Switching environments on AMC.0
- AMC.2 (Advanced Mezzanine Module Ethernet)
  - Defines port usage for Ethernet on AMC.0
- AMC.3 (Advanced Mezzanine Module Storage)
  - Defines port usage for Fibre Channel, SATA or SAS on AMC.0
- AMC.4 (Advanced Mezzanine Module Serial RapidIO)
  - Defines port usage for Serial RapidIO on AMC.0

Vorteile SMTCA vs. IPC

- Kleine kompakte Bauform bei hoher Systemleistung
- Einfaches Upgrade und Wartung
- Hohe mögliche Verlustleistung
- Hoher Datendurchsatz (20 serielle High-Speed-Ports)
- System-Management
- Robustheit: Schock, Vibration, Temperatur
- Lange Produktlebensdauer
- Skalierbarkeit durch Upgrade zu µTCA und ATCA
- Stabiler Industriestandard für 10 Jahre und länger
- Deutliche Reduzierung der Systemkosten
- Investitionssicherheit

Simple µTCA Anwendungen

- Bildverarbeitung (CT/MRT, QS, Personenschutz)
- High-Speed Datenerfassung (Messtechnik, Radar)
- Kommunikationssysteme
- Inspektionssysteme
- Verkehrstechnik
- Medizintechnik
- Wehrtechnik
- Maschinensteuerungen
- Telekommunikationssysteme

powerBridge Computer

Zusammenfassung

Simple µTCA ist die richtige Lösung für Anwender,

- die mittel- und langfristig planen,
- die Gesamtkosten minimieren wollen
- und stabile, zuverlässige Produkte herstellen.

powerBridge Computer

Mentor Graphics is THE ONLY company offering a complete solution !

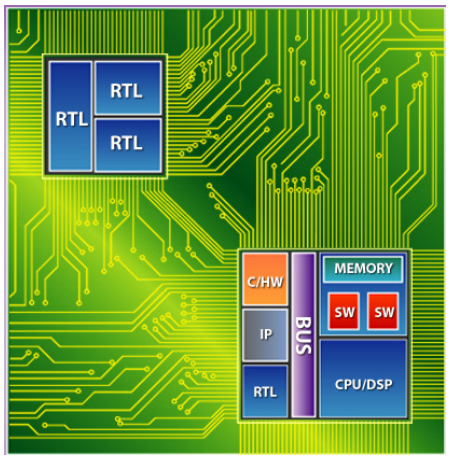
Complex FPGA Systems Demand New Design Approaches

Mainstream / Ready-to-use FPGA/PLDs

-Synthesize

-Place & Route

-Simulate/Debug



High-end FPGA/FPSoC

- Complex timing
- Timing closure problems
- IP integration
- HW/SW design
- Complex debug
- Prototyping
- SDC constraints
- C-based design
- PCB integration issues
- Team design

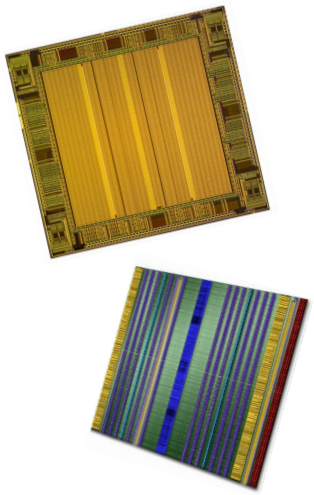
Low-cost tools
Push-button flow

Sophisticated
Complex design flow

Mentor Graphics design tools are best suited for complex FPGA designs.

Too Much to Validate?

- **Majority of all ASIC/FPGA designs required re-spins due to logic/functionality flaws***
  - **Insufficient verification**
  - **Inadequate test points**
  - **Not enough visibility**
  - **Scarce test patterns**
  - **Lacking testbenches**
  - **Deficient coordination with PCB**



* Source 2004 IC/ASIC Functional Verification Study, Collett International Research

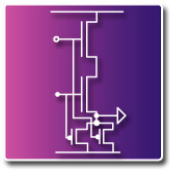
With the explosion in FPGA device size over the last few years, the task of validating designs in silicon has become a daunting task. There simply is just too much to validate using the traditional method of “design-burn-test”

"Design-Burn-Test" When It Was Simple

- Smaller devices → logic analyzer & test board
  - Limited gates to verify
  - Accessible pins
  - Could "spin" around design errors
  - Many iterations in one day

…Re-spinning was Quick and Easy

Transistor 

Gate 

RTL 

With smaller devices it was plausible that a designer could actually debug their design by programming their device and bringing signals out to pins. Re-spinning the design was very easy and quick

"Design-Burn-Test" No Longer Works

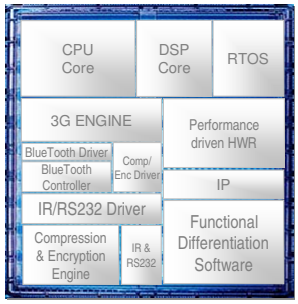
- **FPGAs are complex**
- **Newest FPGA architectures → virtual verification**
  - **Platform based architectures**
  - **Huge number of pin connections, pin limited debug**
  - **Limited internal accessibility/visibility**
  - **One "Spin" per day**

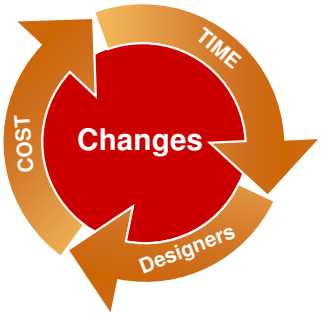
…Re-spinning is Painful and Costly

The size of today's FPGAs have placed an enormous strain on the old tried and true methodology of "design-burn-test", where pulling signals out and inspecting them in the lab with hardware was somewhat feasible on earlier generations of FPGA hardware. Today, however, we need to be thinking about more effective ways of verifying the functionality inside and out of devices. The simple act of re-spinning the design could cost as much as 5-6 hours for large designs, almost an entire work day. With this in mind, we need to work smarter in the way we validate, and avoid unnecessary design re-spins, which are costly in terms of lost engineering productivity. FPGA design re-spins are no longer "free".

CHANGE$$$

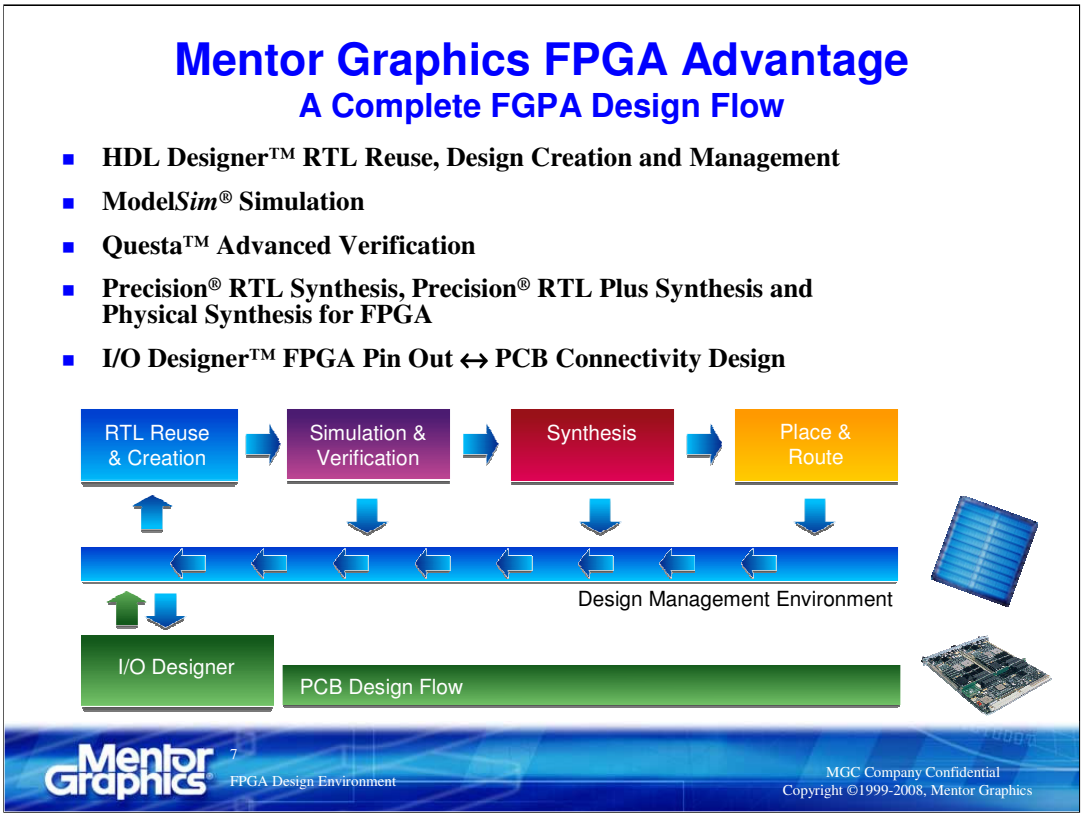
- **Re-spins cost *$$$***
  - **Engineering time**
  - **Engineering effort**
  - **Missed schedules**
  - **Delayed product delivery**
  - **System integration**





- **Design changes are expected**
  - **Functionality requirement changes**
  - **Area restrictions**
  - **Timing budgets**
  - **System parameters**
    - **High speed effects and board timing issues**

The cost of a re-spin is not only lost engineering productivity, but other factors like missing schedules, the product's time to market and possibly even re-spins to accommodate the PCB. Change, however, is a natural part of FPGA design. Change is why FPGAs are attractive for us to use, develop with, and deploy. We need to manage change so that it does no become a limiting factor in our design process.

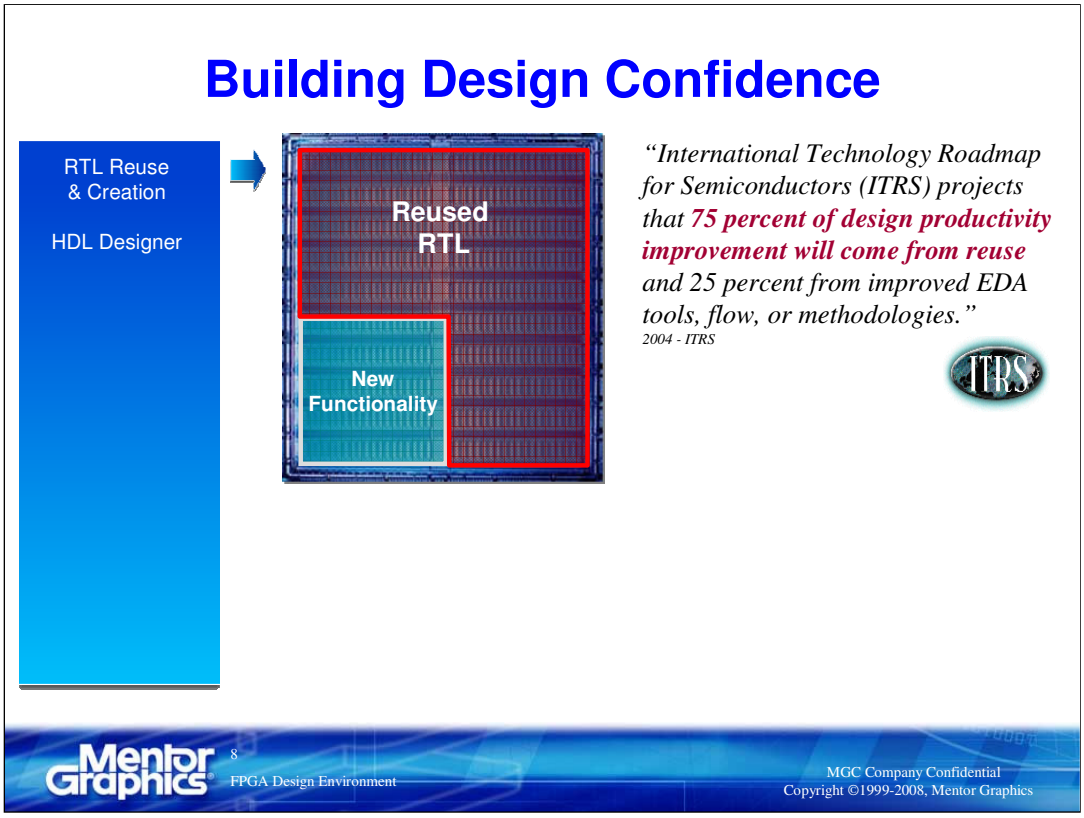


HDL Designer™ - Integrated HDL design with powerful design re-use

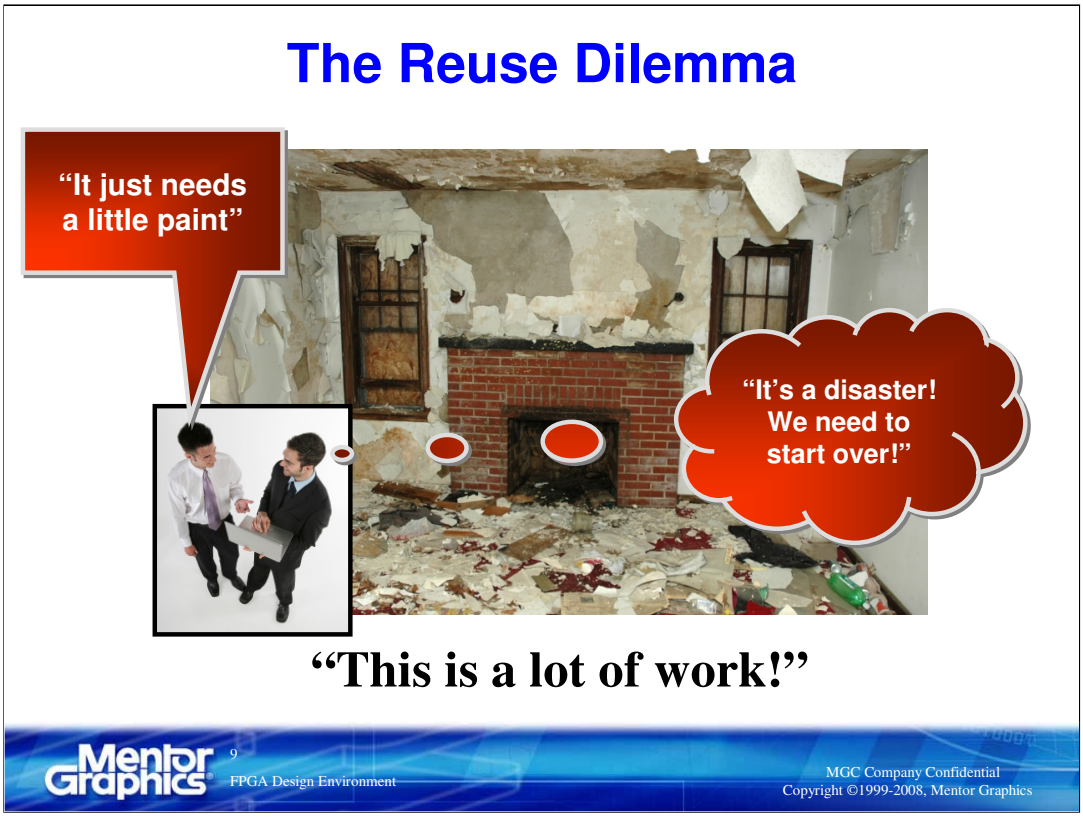
Questa™/ModelSim® - Industry leading simulation environment with Assertions

Precision™ Synthesis – Vendor independent HDL synthesis for FPGA design

I/O Designer™ - I/O placement and integration with the PCB design environment



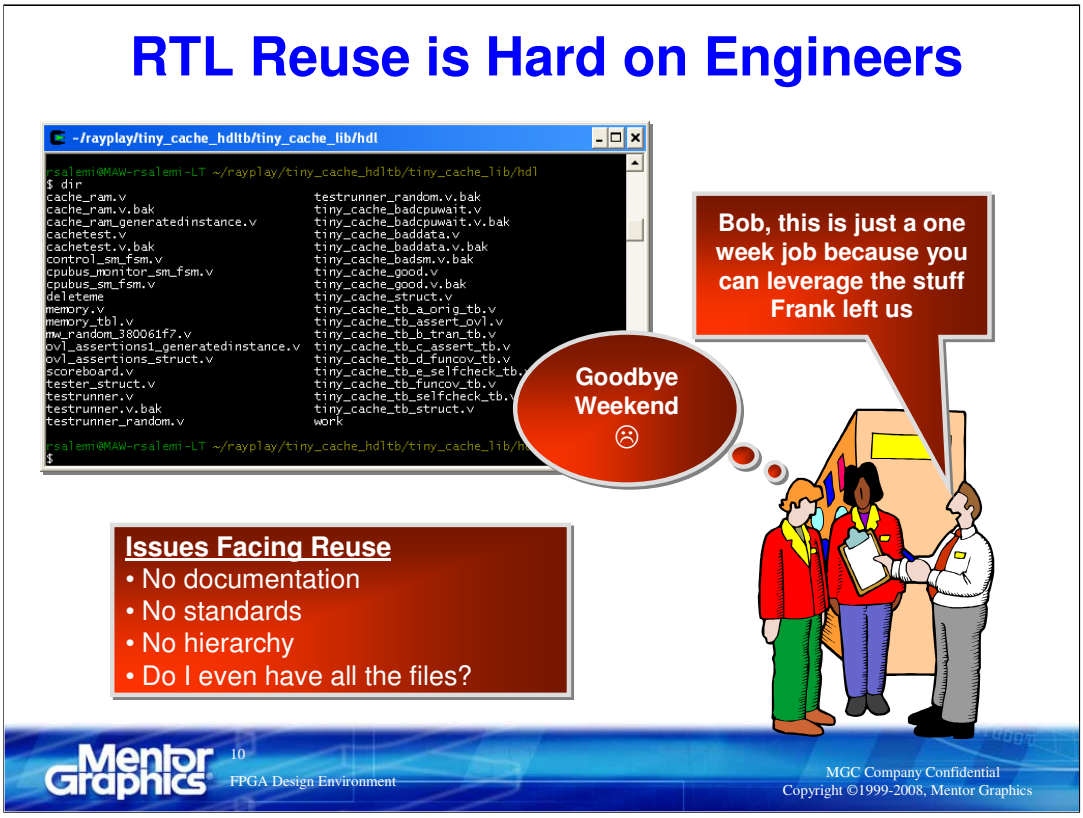
The majority of big design projects use a large portion of reused RTL and add new functionality. This leads to the ITRS position that the majority of productivity improvements will occur from reusing existing code – not in improved tools.



There are two big problems with RTL reuse.

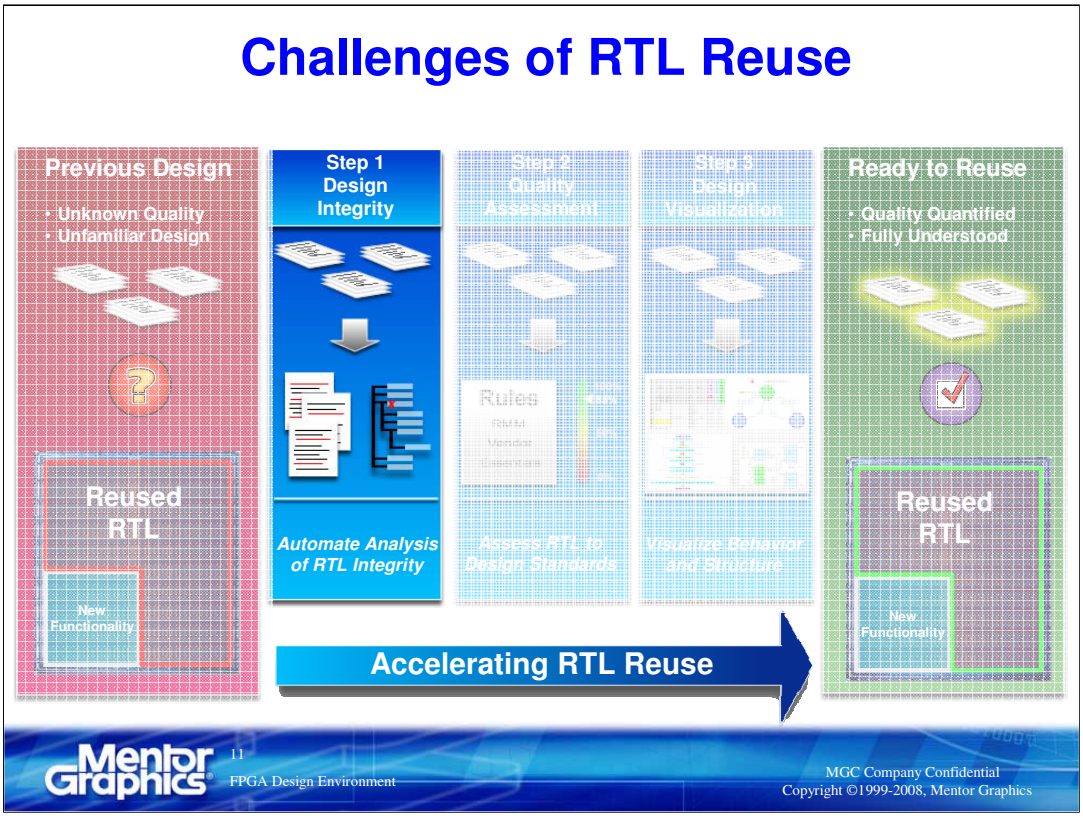
The first is answering the question, should we reuse the code at all? This question is critical to hitting schedules with reused RTL. It takes longer to reuse bad code than to rewrite it, but it is a waste to rewrite good code.

The second problem has to do with the difficulty of reusing code. It is hard to understand someone else's code and engineers need help with this task.



RTL Reuse can be hard to do well. Often the engineer only gets a pile of code with no documentation or comments. The management may expect that this code will save tremendous amounts of time, but without the right tools engineers can lose a lot of time.

This often leads to resistance to using new code. Nobody picks up someone else's code and says, “This is great. I'll use this.” More often than not, engineers feel they are taking less of a risk by writing all the functionality themselves. The key is to have a tool that helps engineers understand reused RTL quickly so they can make it their own.



A clear method of reusing RTL saves time through quickly understanding the task of whether legacy RTL will be used in the project. Regardless of the origins of the code that you are trying to reuse, we propose a simple, 3 step, automated process to help you quickly decide whether the code is worthy of reuse:

Design integrity: is the design complete?

Quality assessment: an automated way to score RTL quality.

Design visualization: graphically examining source code in order to understand it's function.

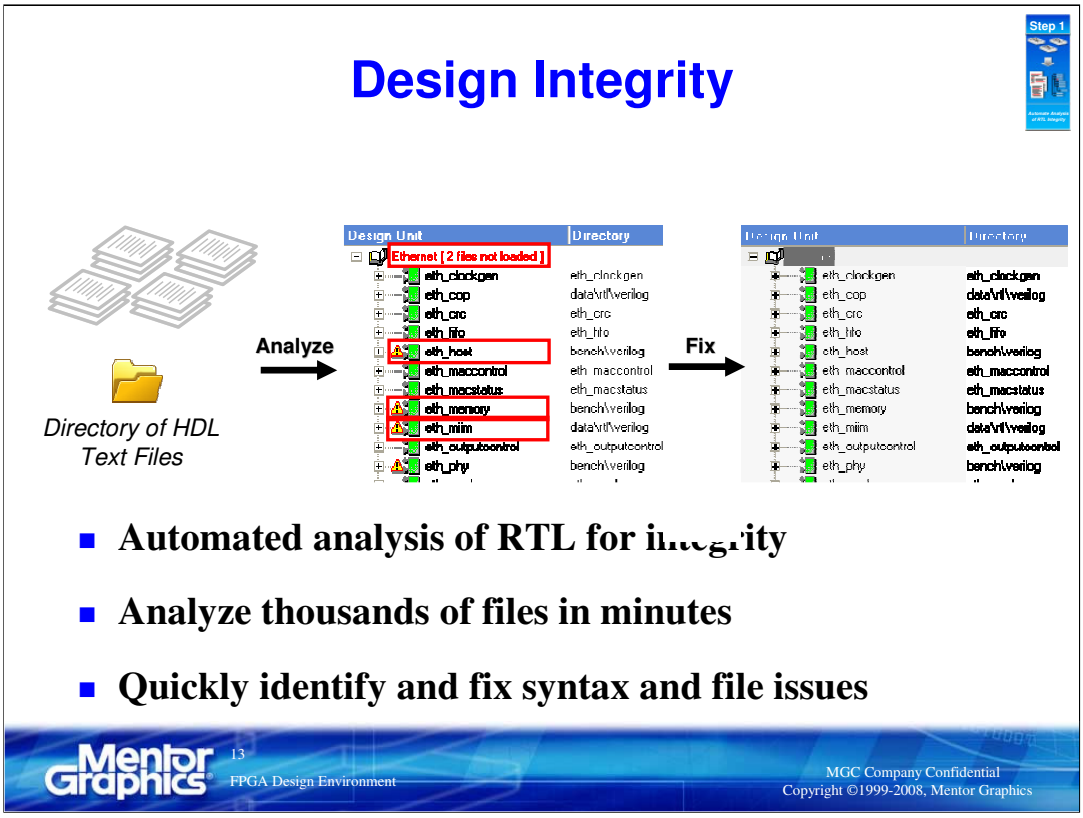
Design Integrity

Common Problems when Encountering Legacy RTL

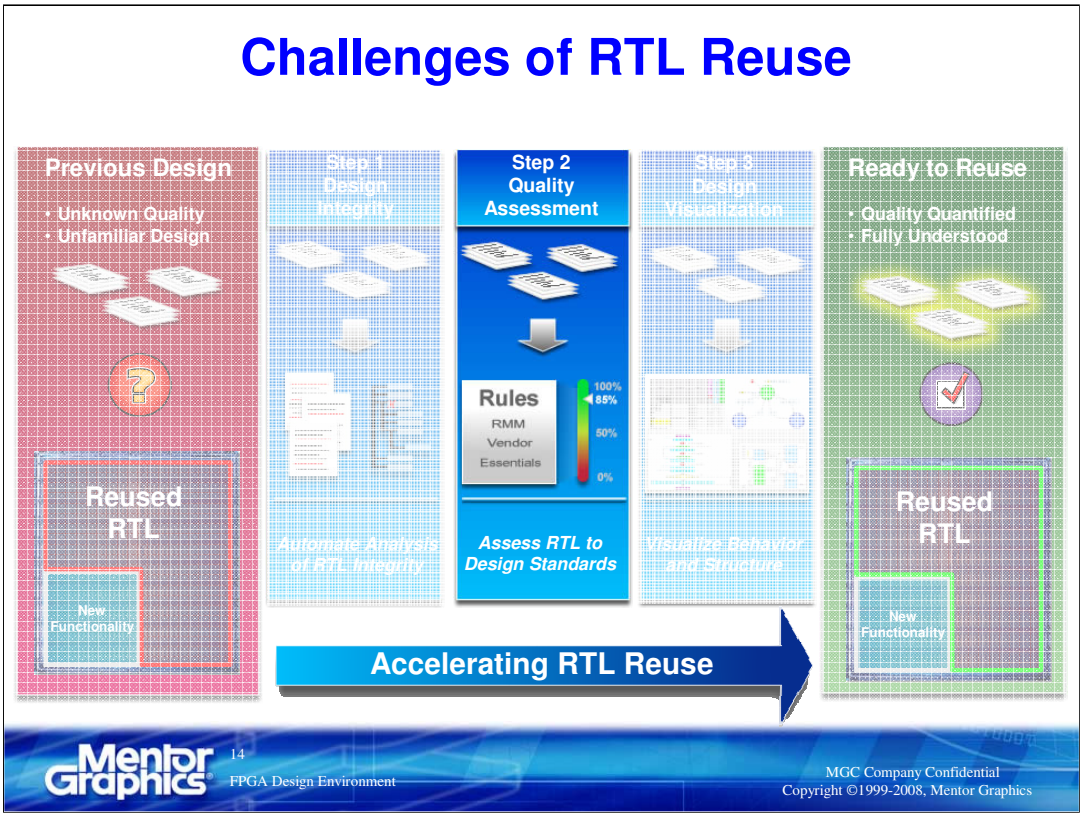
- Syntax errors
- Missing files
- Un-resolved paths
- Lost libraries
- Poor structure for reuse
- Entangled with other code

Do you have all the files in the design? This can be a difficult thing to discover. Often you need to rely upon simulators and other compilation tools to see if they generate errors.

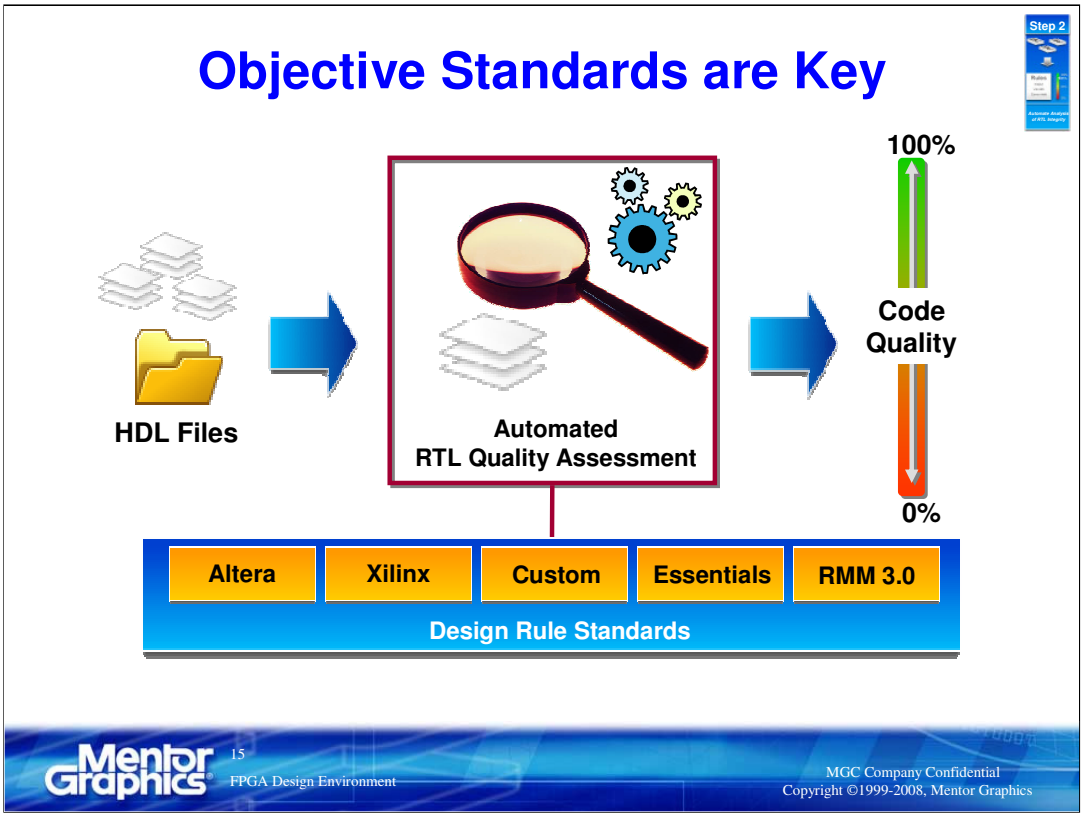
What if a tool could just read your code and point you right to the errors?



HDS answers the question about the design integrity: is the design complete? While importing HDL files HDS parses and analyses all these files and identifies missing pieces.



Next, let's look at how to automatically grade the quality of the RTL.



The key to success is to have objective standards. If you have a mechanized way of analyzing code, you can use these objective measures to fully understand the impact of potential reused RTL.

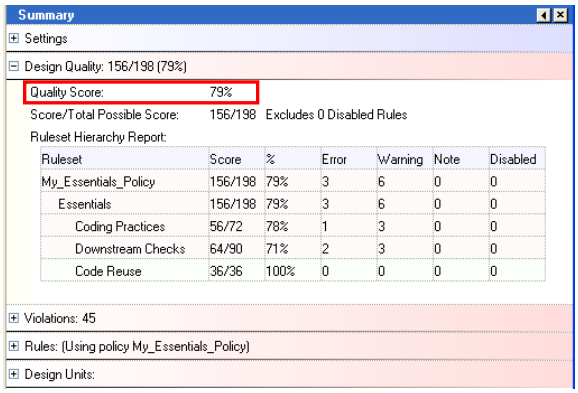
We have Altera and Xilinx rules built, as well as a set of common rules called Essentials and the Reuse Methodology Manual rules.

If you have your own standards, you can easily add them into the tool as well.

Code Quality Assessment

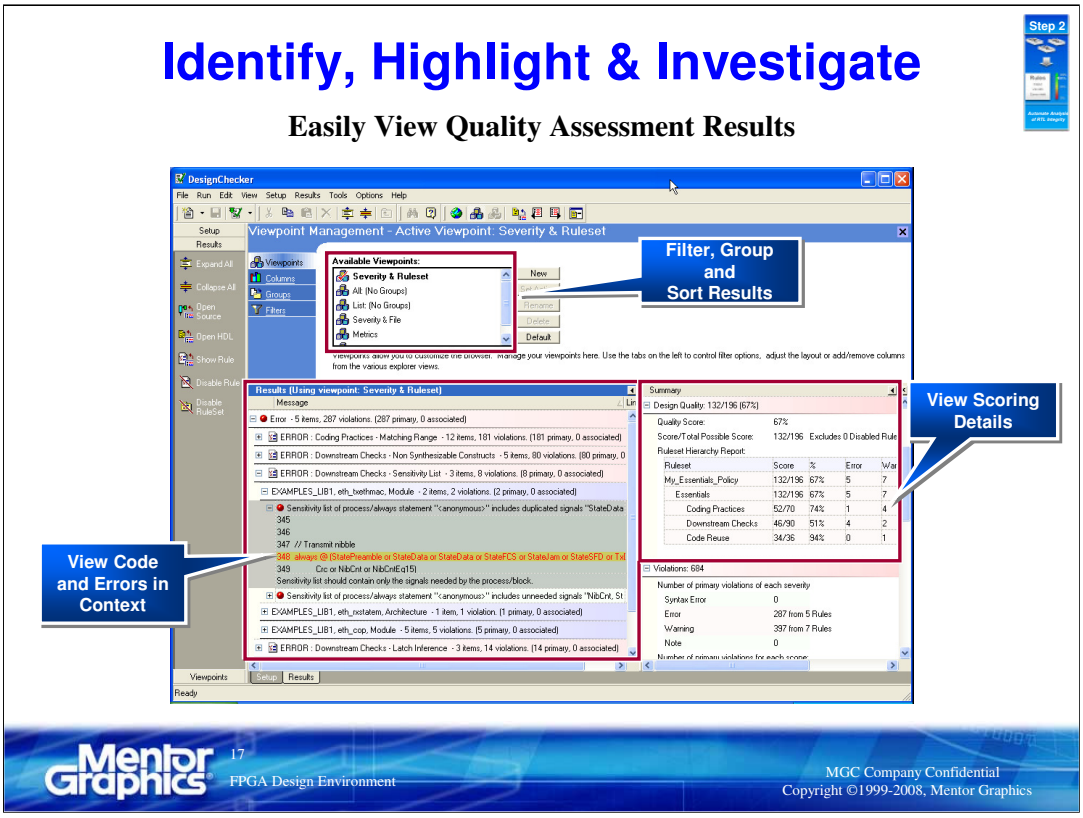
Directory of HDL Text Files

Analyze



- **Determine schedule impact of reusing RTL**
- **Automated scoring of RTL to design rule standards**
- **Powerful GUI for analysis of results**

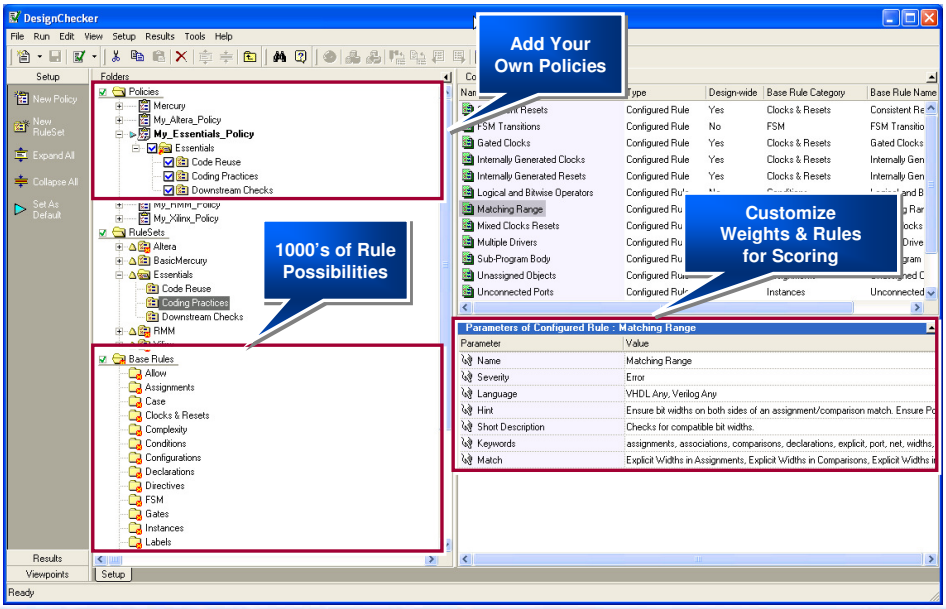
With the click of a button, you can see a score for your RTL code. The score is based on the total score for each rule in play. Your score is formed by deducting points for each violation, providing you with a percentage for the design (out of 100%). Based on this score, you can determine how much work it will take to get the code into proper shape for reuse.



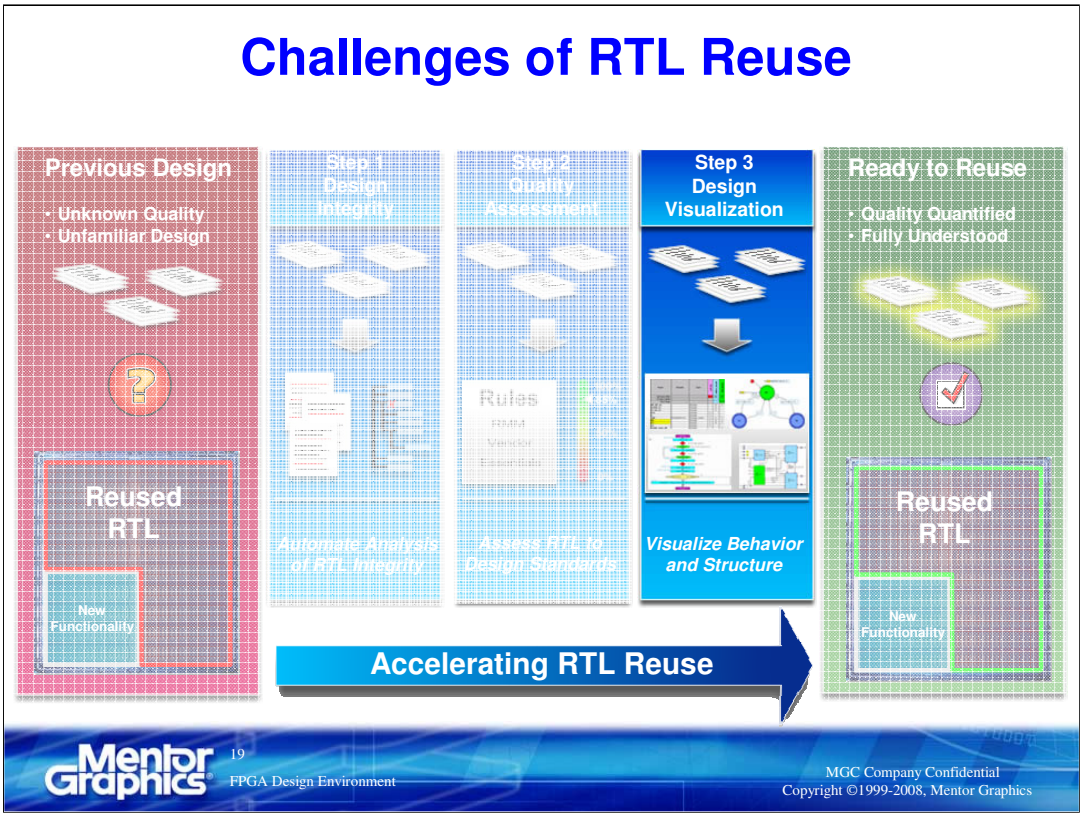
Once you run the code through the checker, it is easy to dig into the results and find out where you need to work on the code. There are three parts of HDS that help:

1. HDS shows you the errors in context.
2. HDL Designer shows you details on which rules and rulesets were missed
3. HDL Designer let's you filter, group, and sort all results. For example you can view results sorted by file, rule, design unit, or ruleset. You can filter on any part of the design hierarchy to zoom in on specific problems.

Create Your Own Custom Rules & Policies



You can create your own rulesets and policies. HDS ships with hundreds of common base rules. You can use these base rules to create your own coding standards, or use the ones that ship with HDS.



The final step is to visualize your RTL code in order to understand what it does…

Design Visualization Challenges

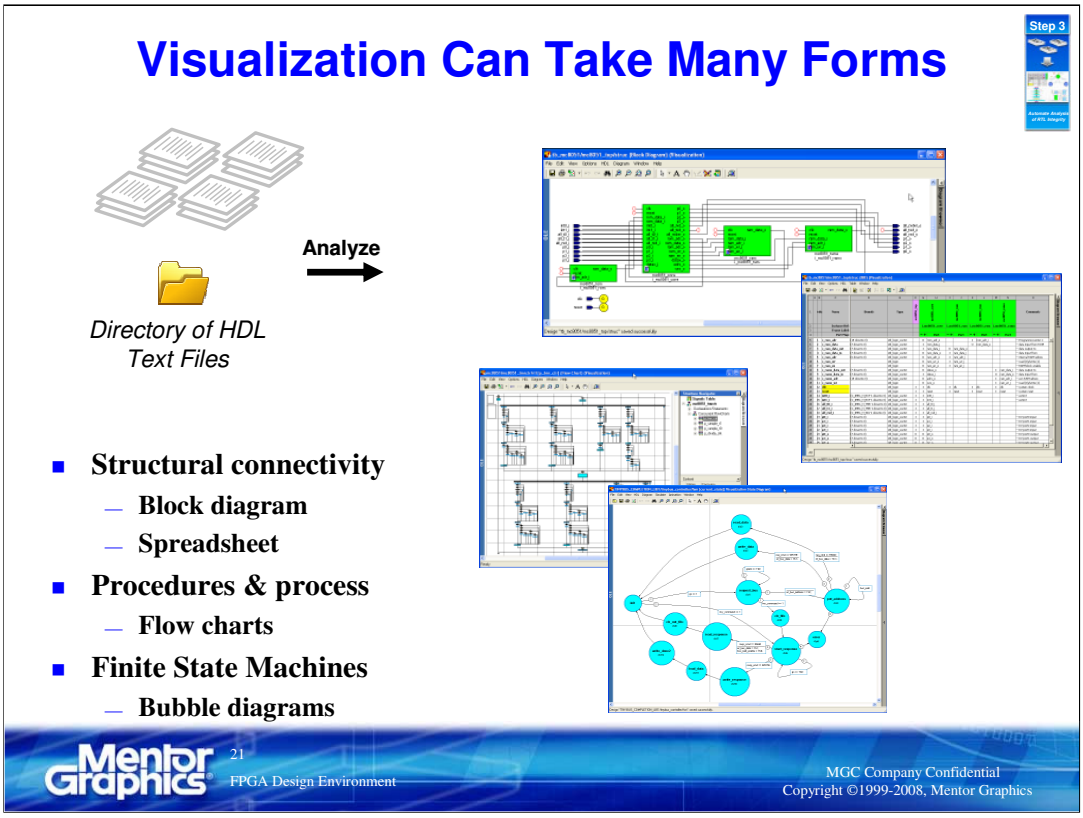
Learning & Integrating Legacy Designs

- **Designs are easier to *WRITE* in text, but harder to *READ***
  - **Design structure is hard to "see" in text**
  - **Design functionality is hard to trace across text files**
  - **Interfaces are often poorly documented**
- **Re-drawing by hand is Inefficient**
  - **Very time consuming**
  - **Error prone**
  - **Un-maintainable**

Existing RTL Code

Reading through all the code that represents the design can be tedious. High-level information, like the structure of the design, is buried within all the RTL constructs.

Some folks read the code and draw pictures, such as block diagrams, by hand in order to figure out the code. However, this is error-prone, takes and long time, and if you alter the design, these drawings become out of date.



Before you go into code with a text editor, it is important to have an overall map of the code and what it looks like. Hacking around in text with out the big picture will only be dangerous.

HDS creates Block Diagrams, Flow Charts, and Bubble Diagrams to help you understand your code.

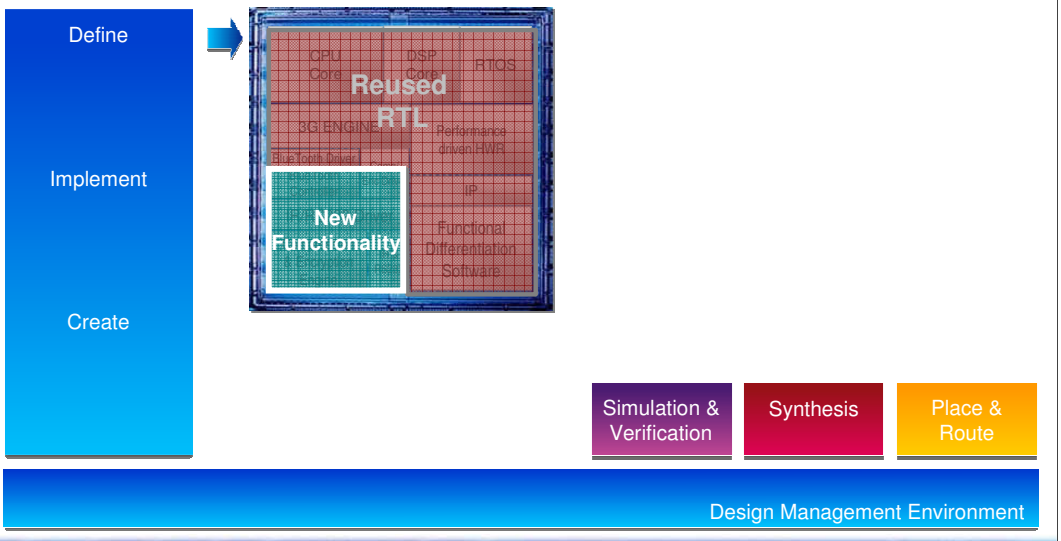
How HDL Designer *Accelerates RTL Reuse*

- ✓ **Saves time by analyzing & correcting design integrity**
- ✓ **Ensures the best reuse decision by measuring code quality**
- ✓ **Accelerates design understanding of structure & behavior**
- ✓ **Reduces pain when integrating legacy code**
- ✓ **Makes designs more reusable for the future**

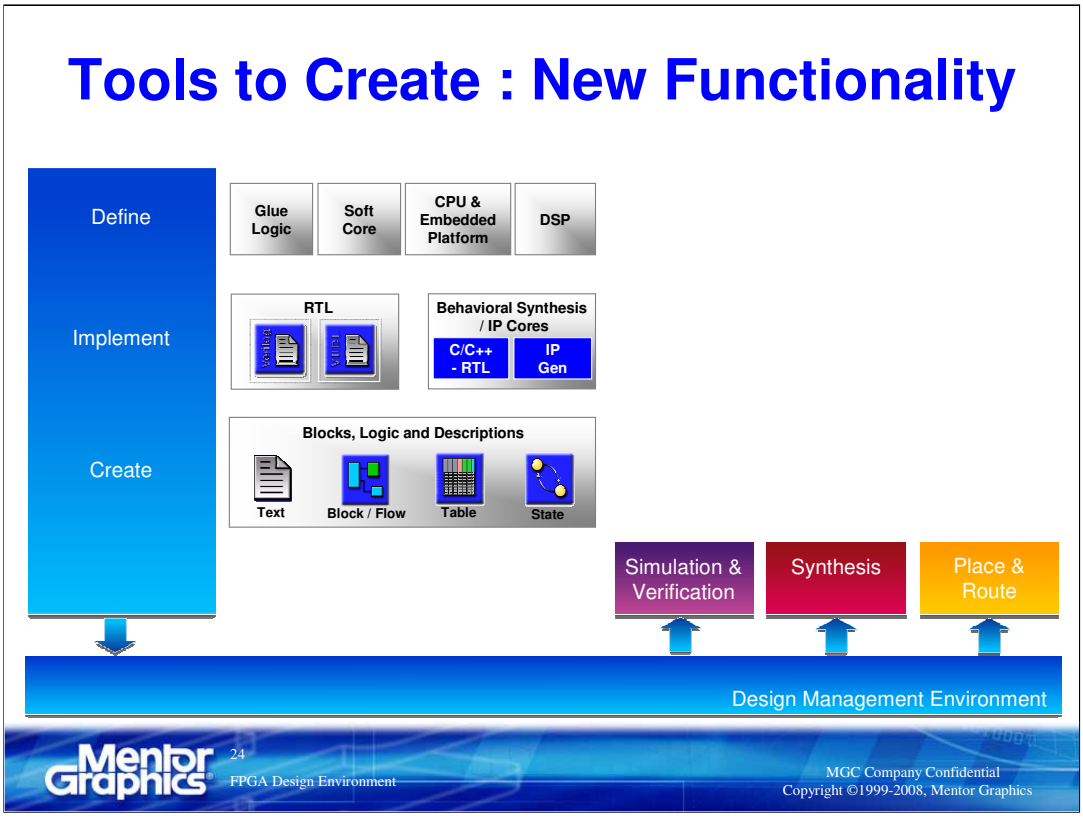
The benefits of our 3 step reuse process are <on the slide>

Design Creation

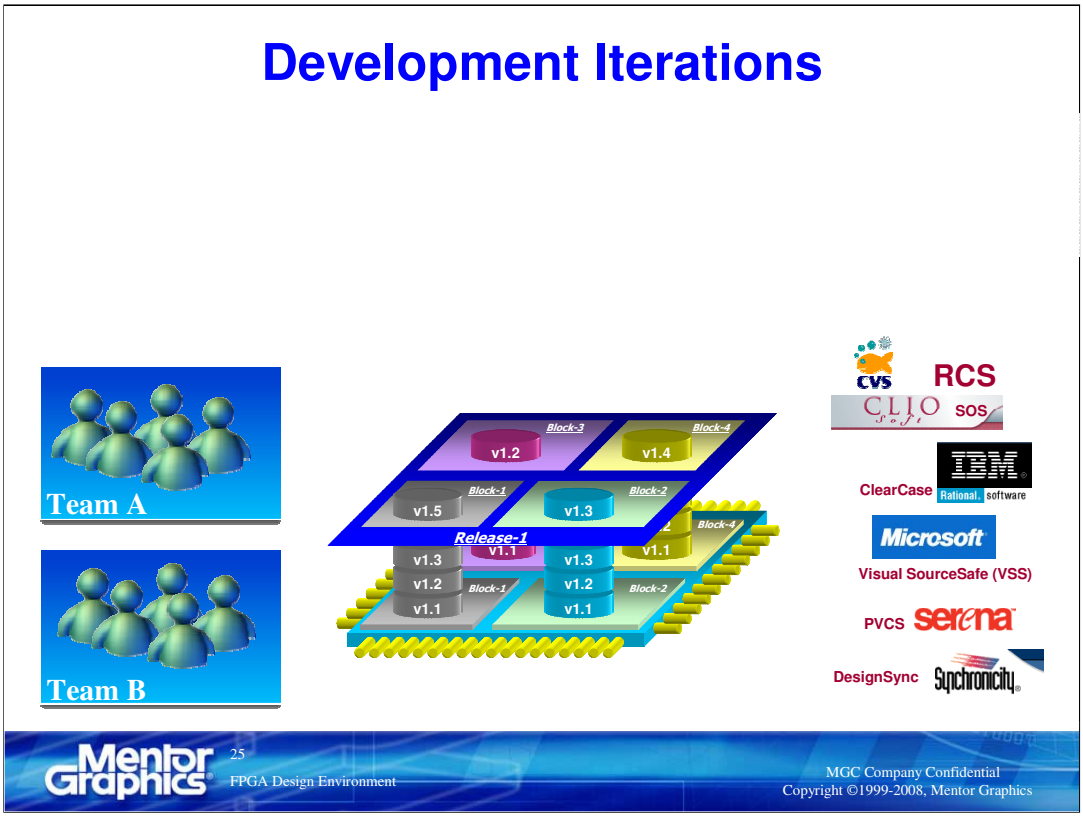
Design for downstream success…manage all design descriptions and controls



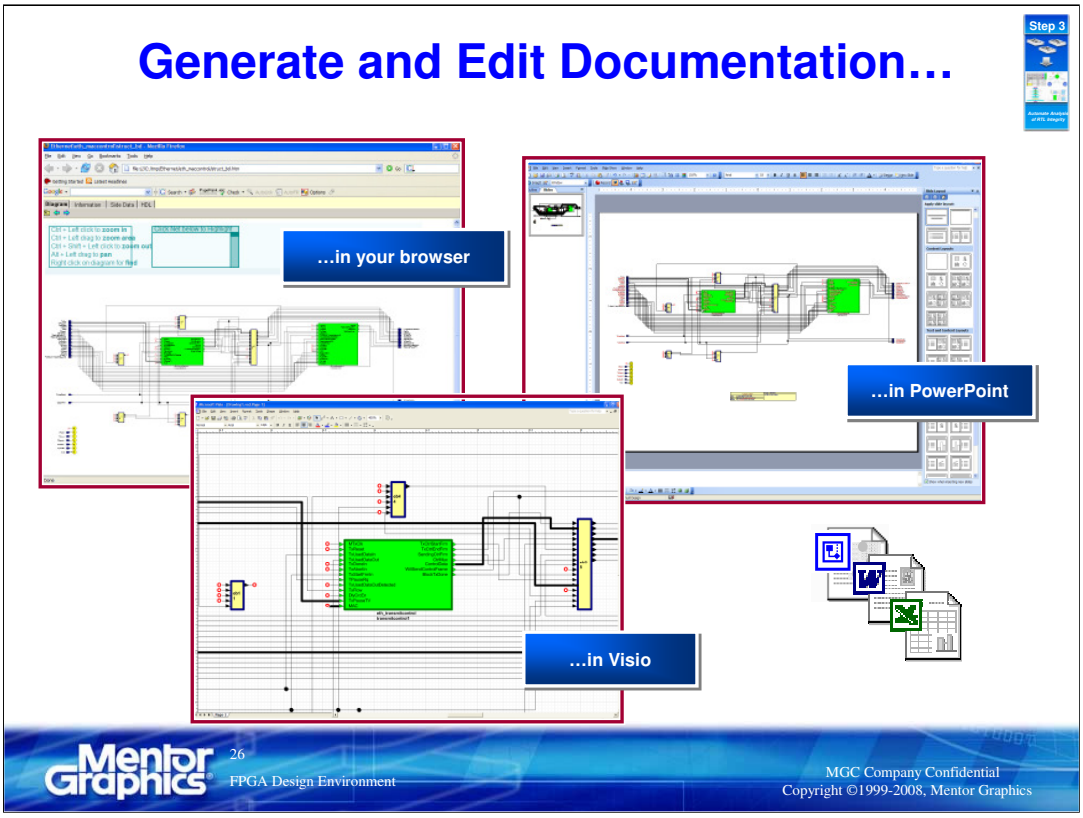
Now that we have an effective method for reusing RTL from previous work, we'll take a look at creating new functionality.



As systems are defined and consequently specified from the block level and then implemented, we must get all blocks to the RTL level prior to synthesis for specific technology implementation. HDL Designer manages the definition of the design, most common approach is starting at the block level, then develop the logic and descriptions with text and/or graphics and ultimately generate the system level RTL description of the design.



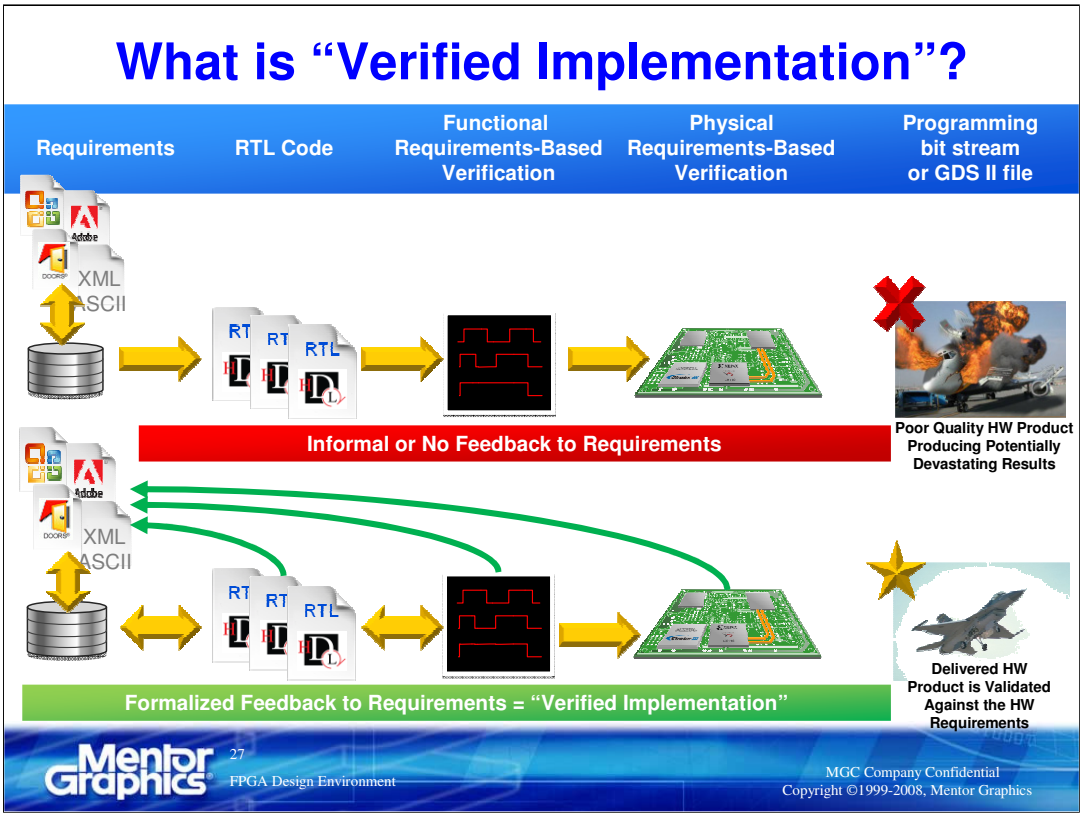
The development cycle includes the simulation and verification phase then the synthesis stages. In the meantime a design might be divided by functional blocks amongst several designers or teams. To effectively manage the design iterations, HDL Designer includes CVS and RCS opensource version management tools and is integrated with other commonly used tools as well. Version management is the easiest way to maintain the integrity of the design code and track change history.



HDL Designer creates documentation in a number of formats. You can create dcoumentation that you can read in your internet browser. In fact HDL Designer will generate a web site for your entire design so you can browse the hierarchy. The drawings can be written in Adobe's SVG format which allows infinite zooming with completely clear drawings.

You can also generate views that can be read and edited in PowerPoint and Visio.

- Automated design documentation and website publication
- Generates directly from design files
- Easy drag-and-drop into Microsoft Visio & Office
- Designer decides which files should be included in the website



The need of having a tight link between the specification and downstreaming design

and verification tools is mandatory for tracking tasks.

Recommended Solution

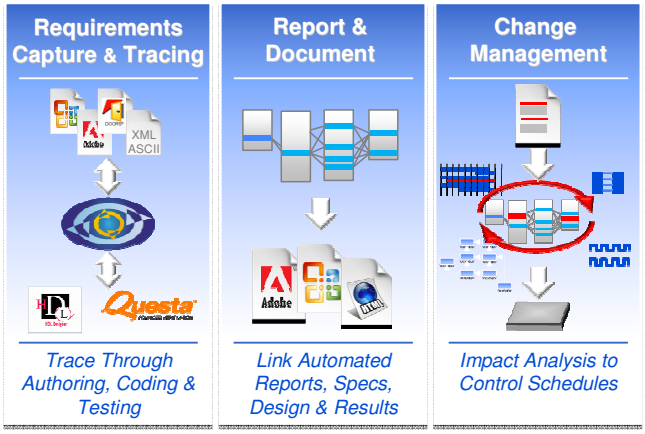
Repeatable & Verifiable Design Process

- **What**
  - **A development environment that facilitates good project management practices for improved overall product quality**
- **Solution comprised of…**
  - **Quality code checking & measurement**
  - **Requirements tracking & management**
  - **Configuration management**
  - **Audit support / project documentation**
- **Critical attributes for any chosen solution**
  - **Requirements tracking down through design & back to requirement**
  - **Design checking of syntax, semantics & rules**
  - **Design visualization to aid in design code understanding**
  - **Ability to incorporate new and changed requirements reliably**
  - **Configuration management of project design, requirements & artifacts**
  - **Minimal manual error-prone interaction**

ReqTracer™

Managing Requirements in Your Design Flow

- **Control & predict project schedules**
- **Trace requirements through HW design process**
- **Clearly communicate via visualization & intuitive reports**
- **Manage impact of requirement changes**
- **Meet safety critical & DO-254 certification**



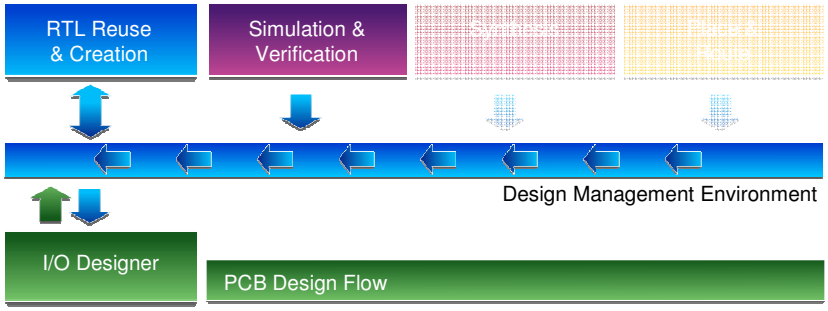
KEY

"A tool that can facilitate traceability would be a much needed help to developers who must comply with DO-254 objectives for traceability."

Tammy Reeve, President, PATMOS Engineering

Enhancing Functional Verification

- **Typical verification methodology:** (used in many companies today)
  - Directed tests *or* "burn-and-test" in the system lab!
- Verification must accommodate
  - Raising the level of coverage
  - Self-checking verification

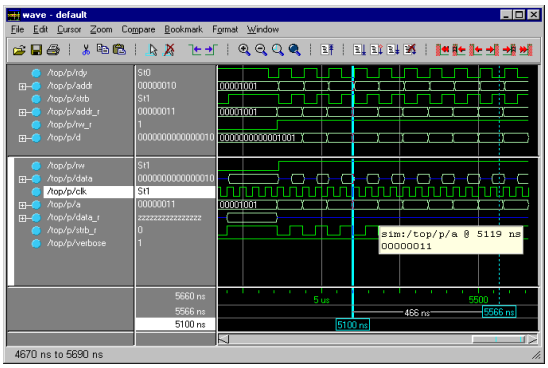


If we consider the entire FPGA design flow, the task of verification deals with the design itself, the actual task of verification, which uses an appropriate testbench and simulation to prove correct functional operation. Validating that the functional operation is correct, and to the prescribed specifications can be a difficult, time consuming and an incomplete process (how do you know when your are done?).

Many companies (and engineers) are still using "ad-hoc" or "burn-and-test" methods in the lab to verify a design. While it is very gratifying to see a piece of hardware come up and work in the lab, especially with a minimum of testing on the software side, and the "I can just re-program the FPGA" temptation can be very real, the quality of the design, under all operating conditions is still unknown.

Changes Are Cheap During Virtual Verification

- **Start analysis and verification early**
  - **HDL coding**
  - **Target FPGA device**
  - **Area modification***
  - **Timing constraints***
  - **I/O design configuration**
  - **Micro architecture**
  - **HW/SW partitioning**



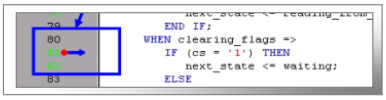
Iterations take longer the closer you get to silicon!

*** Most useful timing and area results will be obtained post P&R**

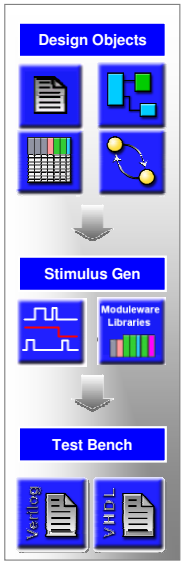
The earlier in the design process that necessary changes can be implemented, the less expensive that change will be. The closer to silicon the greater the impact of the change.

Dynamic Debug & Analysis

- **Boost your simulation efficiency**
  - Incrementally compile, launch, run
  - Visualize HDL code
- **Analyze interactively**
  - Live debug visualization
  - Cross probe
  - Set breakpoints



- **Automate your stimulus generation**
  - Shorten stimulus creation times
  - Parameterize stimuli
  - Easy waveform entry



In the area of Debug and Analysis, HDL Designer enhances the simulation task by enabling live visualization of simulation data back to the graphics. Links from within HDS allows the control of the simulation so that you can track down and analyze the resulting data to quickly isolate the design errors/bugs. HDS also can be used to easily create testbenches utilizing the Moduleware libraries or via Designwave for easy hand creation of test benches.

Enhancing Functional Verification

How complete is your testbench for verification?

- **FPGA complexity makes lab debug Difficult**
  - **Multi-million gate design anyone?**
- **Logic probes change timing**
  - **100's or 1000's of pins**
- **Schedules slip as debug continues**

How can you validate more of your design *PRIOR* to "tapeout" in the lab?

Testbenches are able to provide a robust method for validating the design, but it is difficult to know just how complete a hand written testbench is for verification. The complexity of FPGAs make it increasingly difficult to know that you have covered all of the possible cases. The same goes for actual hardware debugging in the lab. Logic probes attached to the circuit change the loading, and in doing so the drive characteristics and timing will also be changed.

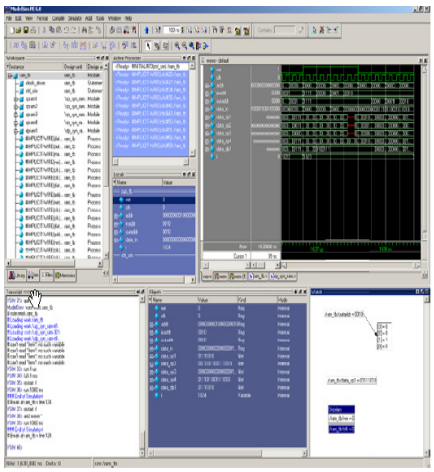
It is estimated that verification consumes about 70% of the project schedule and this figure is growing. An effective, repeatable verification process is required. For FPGA-based designs, the temptation is to program ("blow") the device and to start lab testing as soon as possible, although this is not a viable option for ASIC designs, unless FPGA prototypes are developed. Lab testing quickly gets you to the point where you can stimulate the design with "real-life" data and monitor the actual behaviour with the device running at full speed. This approach results in many additional debug pins (sometimes hundreds) on the device with a corresponding large number of probes.

These probes, in turn, change the timing of the device. It is difficult to ensure that sufficient stimulus has been applied, not only to cover expected, but also unexpected behaviour and corner cases.

Perhaps the biggest drawback is that once a problem occurs, it is very difficult to locate the cause. This results in never-ending iterations with the team unsure when the device has been sufficiently tested. This is particularly true of intermittent problems. Is there a pin loading problem, where the signal is not making it to the other chips, or is the problem some complex behaviour that was unexpected?

Simulate Lab Behavior

- Simulators are easy to probe
- Changes are easy to make
- Debug is more efficient
- Creating a good test bench is a challenge



Raising the level of functional verification is *ESSENTIAL* for reduced spins and lab debug time

Although logic simulation is slower than running the real device, by applying advanced verification techniques, you can rapidly create and apply realistic stimulus which exercises the design in both expected and unexpected scenarios. In addition, the testbench itself can tell you when a problem occurs (without having to scan through waveforms) and makes it easy to locate the cause and fix it. By using Functional Coverage techniques, the testbench can also tell you when you have fully exercised the required functionality. The challenge is the creation of a testbench with known coverage.

Methodology Explosion Targeting Verification

- Assertion-based verification
- Functional coverage
- Constrained-random testing
- Coverage-driven verification
- Dynamic-formal verification
- Transaction-level verification
- Model checking
- And more . . .

The size explosion in the designs that are being implemented in today's FPGAs, has also lead to an explosion in the techniques and methodologies in validating these designs.

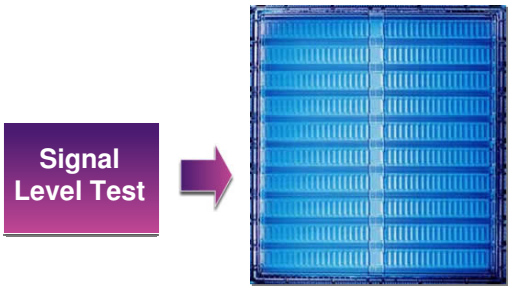
Value of Testbench Automation

- **Generate more stimuli & tests**
  - Assume constrained random test bench takes longer to write
  - Equivalent to 10's or more equivalent directed tests
- **High reuse of test bench code**
  - Write at higher abstraction level
    - Fewer lines = fewer bugs in the testbench
  - Language features (e.g., OOP) increase reuse opportunities
- **Standards verification languages**
  - SystemVerilog
  - SystemC
  - Work with all design languages
    - VHDL, SystemC, Verilog, SystemVerilog

Constrained Random Testbenches may not take longer to write as the testbench writer need not program all the specifics into the testbench. Most of the additional upfront effort in testbench writing is in making the testbench reusable. Tools and methodologies used to assist in the assembly of testbenches and generation of stimulus which, ensures that the testbenches can be used effectively and reduces the number of tests that must be done by hand. The testbench is essentially the scaffolding around the design. All the stuff you need to build and verify a design should be specified in the verification plan, which can cost as much or more than the design itself. For this reason, efficiently creating a testbench that is reusable is an important task.

Directed Tests Benches

- **Basic RTL test bench**
  - **Simple stimulus from a HDL module**
- **Author maintains understanding of bus protocol**

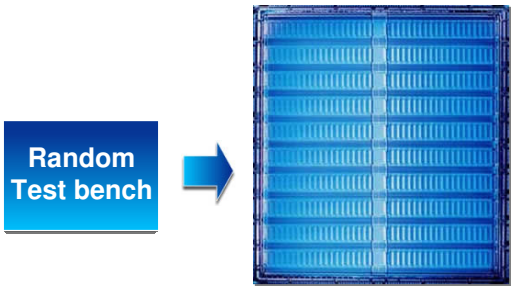


Difficult to expand or randomize

In their simplest form, testbenches can produce a direct stimulus with a number of directed tests that are applied to the ports of the device under test. Each directed test consists of a series of individual signal value changes at specific relative times which effectively means the bus protocol is embedded within each and every test. The stimulus can be implemented in a variety of ways, from reading in files of values, creation in behavioral code, application of waveforms or even simulator force files and scripts. As we shall see, the problem with all of these approaches is the difficulty of expanding and randomizing the tests.

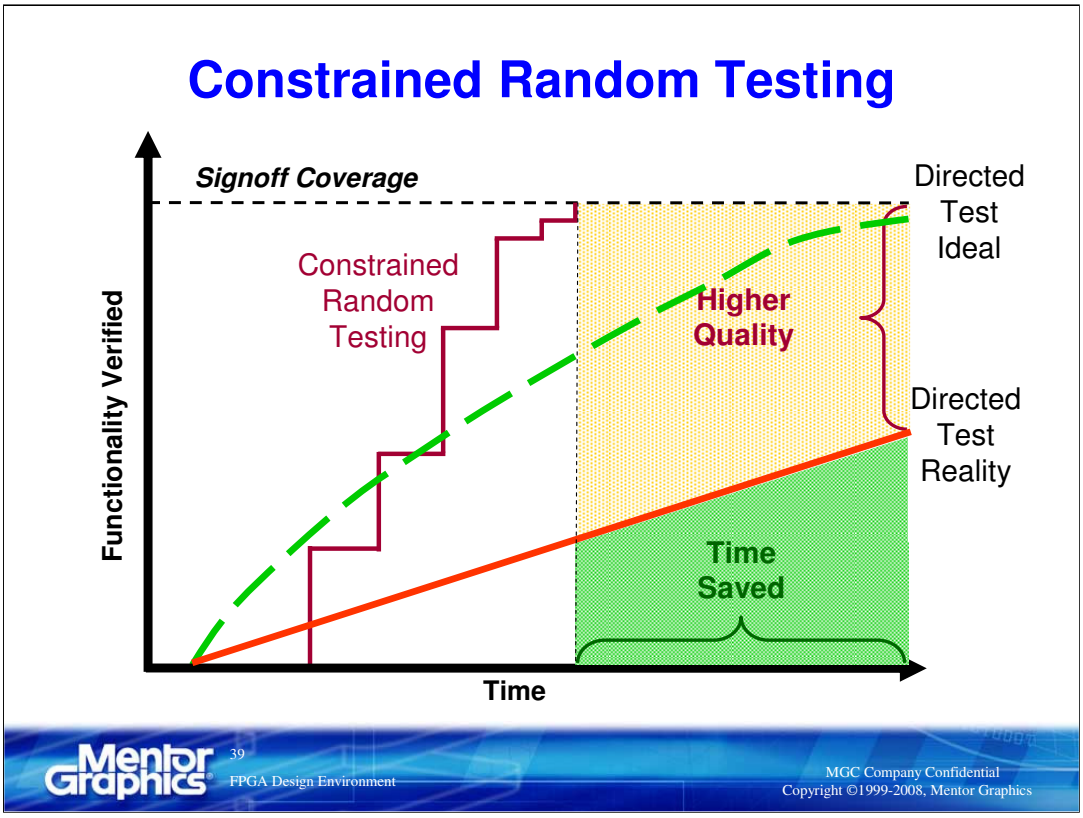
Directed Random Test Benches

- Generate random values to apply to the device under test
- Can be run repeatedly
  - Deterministic
  - All possible combinations covered (how long to run?)
    - Insures coverage
  - Equivalent to the running in lab
- No HW implementation required



Infinite Tests = Infinite Results
How do you detect errors? Must constrain!!!

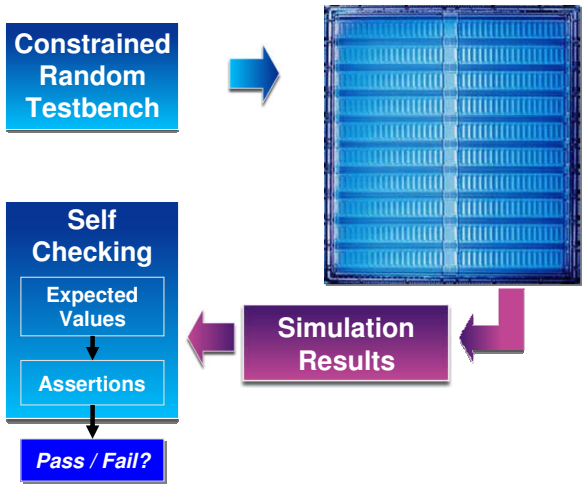
Taking the directed testbench one step further, and we can randomize the stimulus that we apply to the device under test. When this is done, we call it Directed Random Testing. A directed test that uses pseudo-random numbers for parameters/data/etc. Generally one would write random data to a random address; Read back from same address and compare. As could be imagined, going through every combination and permutation of all inputs would produce a very complete, if not very large set of test stimulus. We still need to be able to detect any errors that occur, and if possible reduce the redundancy by using a totally random methodology



With the use of constrained random testing, we are able to reduce the total time in testing, while meeting our sign-off coverage goals earlier.

Self-Checking Testbenches

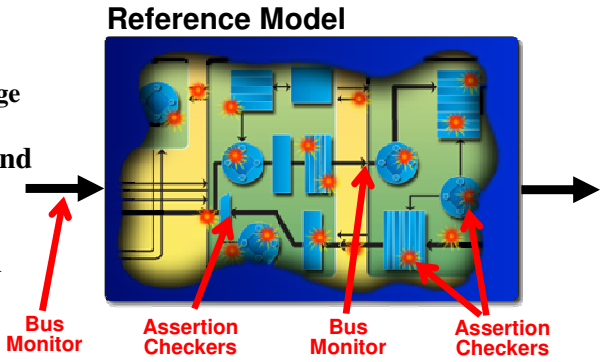
- **Self-checking test bench**
  - **Checks values against known valid response**
- **Combine with constrained random test bench**
  - **A powerful testing methodology**



We can take the constrained random testbench that we have, and add code into the design that will allow us to monitor that it is behaving correctly. We can do this with assertions.

Questa™ Assertion-Based Verification

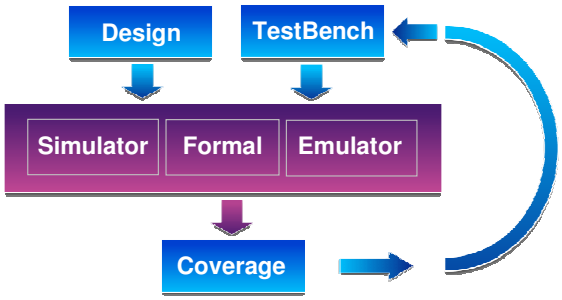
- Assertions
  - Find more bugs
  - Shorten repair times
  - Basis for functional coverage
- Questa supports both PSL and SystemVerilog assertions
  - VHDL and SystemVerilog
  - Mentor-donated OVL SVA
- Native assertion engine in Questa ensures high performance



Assertions allow us to expose problems that would only be seen with the most rigorous of testing.

Coverage-Driven Verification (CDV)

- Every verification cycle counts
  - Eliminate simulation cycles that don't add to coverage
  - Identify "holes" in verification
- Effectiveness of methodology becomes the new benchmark
- Integrated coverage collection across multiple tools is key
  - Results in merging of coverage data

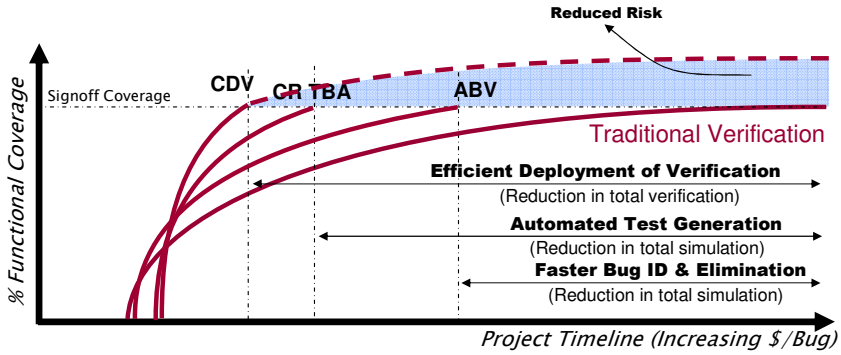


Another weapon in our arsenal is coverage-driven verification. This provides answers to some of the most common nagging questions most of us have a bout the completeness of our verification strategy and the tests used to implement that strategy.

Have I verified all functional requirements?
Have I covered the entire test plan?
Are my assertions doing anything?
Have I exercised corner cases in my design?
Am I using my verification resources efficiently?
Is every testcase adding unique coverage?
Do I have sufficient observability of my design?
How can I generate stimulus to reach uncovered areas?
Am I done?
If I'm not done, do I have a clue as to how much risk remains?

Integrated coverage maximizes the benefits of assertion based verification (ABV). Verification requires a Total Coverage Model as it is effectively metric-less. Few designers know if their strategy is adequate or efficient. The sign-off criteria are typically ad-hoc and vary by company. The simple metric of code coverage is not a functional verification metric, it can tell you how many lines you have hit, but it can't tell you want states you have missed.
Without these metrics we don't know if the Test Plan is complete, how much remains untested or even what is the meaning of sign-off? Coverage of C-R testing is unknown without metrics. If it isn't verified, it's broken.

Adopting Verification Methodologies Works



Fastest route to verification closure requires new methods!

We have seen a number of different methodologies that we can use to improve the coverage we have in our verification strategy.

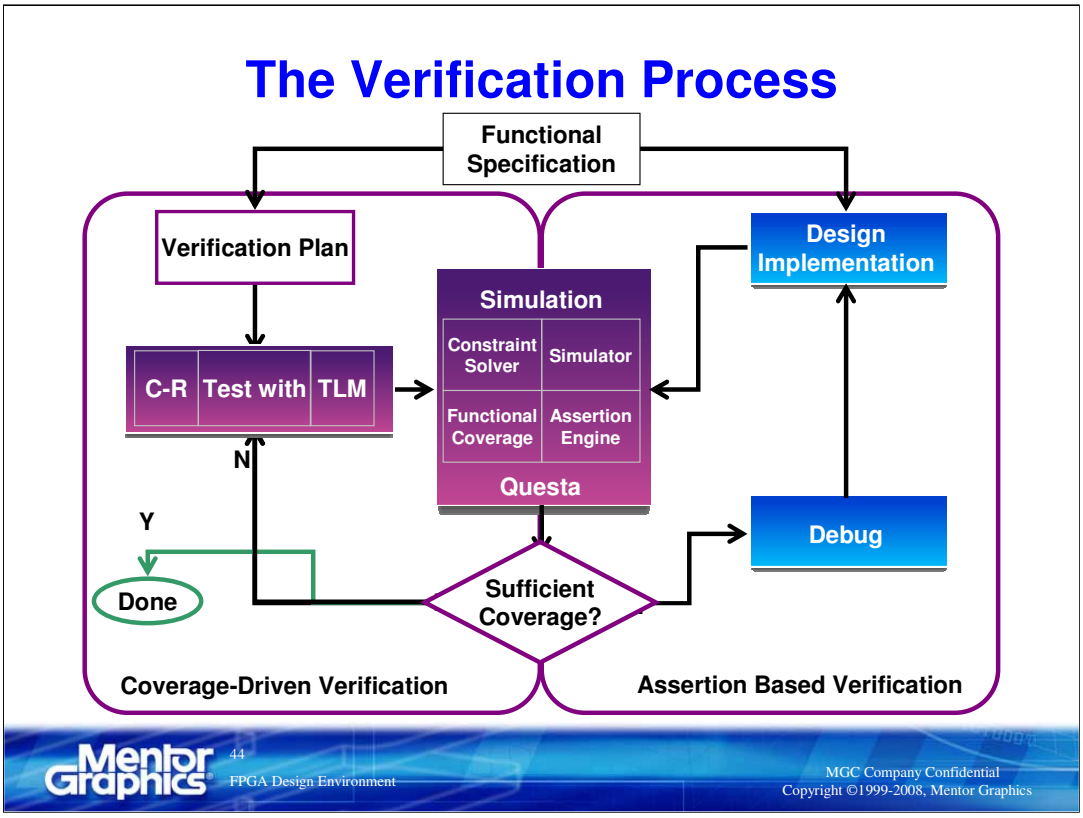
ABV: Assertion Based Verification

CR: Constrained Random

TBA: Testbench automation

CDV: Coverage-Driven Verification

Using one, or a combination of these improves the total coverage we have for the entire verification process.

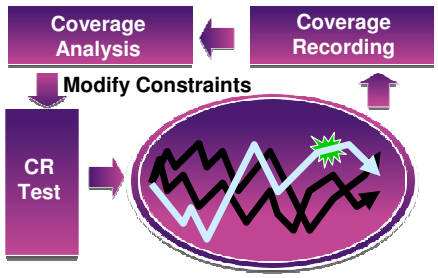


If we look at the entire verification process, we start with the functional specification. From that we have our verification test plan and the design implementation. We can implement our test bench and use simulation to find any bugs in our implementation. We can iterate around the debug and sufficient test coverage paths until we have our design at the desired quality. This can all be accomplished using industry standards.

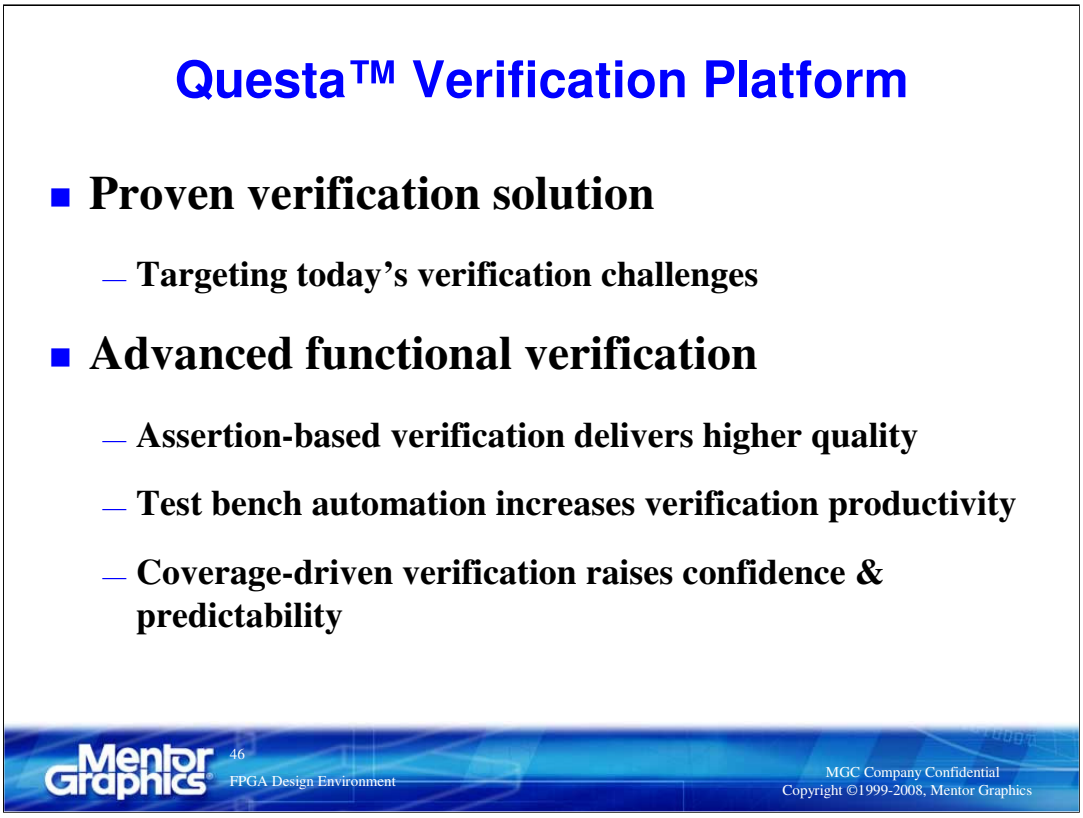
Synergy of Coverage & Test Automation

The "How" of effective test

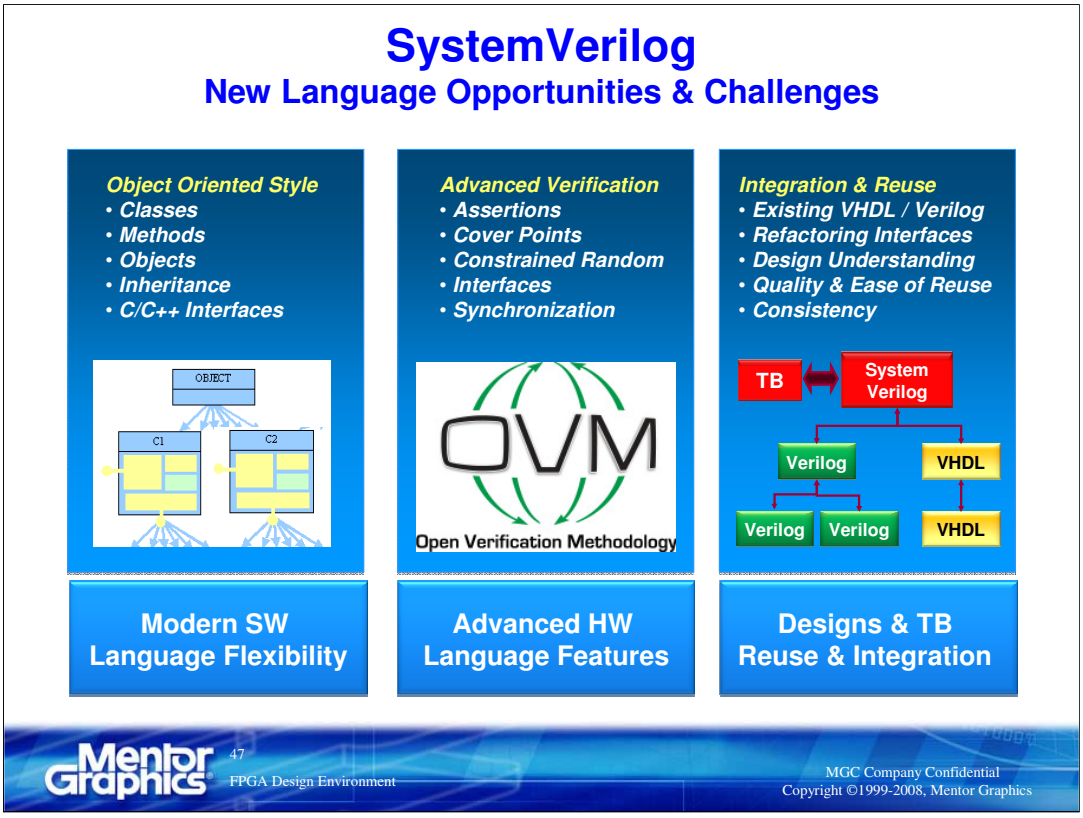
- SystemVerilog and SystemC randomization features automatically vary the stimulus to explore more of the design
- Coverage analysis compares the test against the target goal and modifies the stimulus constraints as needed
- Result is verification engineers can write one test to do the work of many



Test creation productivity is critical, as testbench sizes can exceed design size. The synergy created by using a tool to assist with testbench creation can help overcome this burden. Based on the advanced verification features of SystemVerilog and SystemC, we can create a verification environment which takes advantage of Constrained-random stimulus generation, Reactive testbenches and Transaction-level modeling to increase the value delivered by your verification team



Mentor Graphics has the proven verification solution that can target today's FPGA and IC verification challenges by using advanced functional verification techniques.



object oriented languages used for design description and functional verification need to be supported tool vice. Certe does it.

Advanced Verification Methodologies Adoption Problems

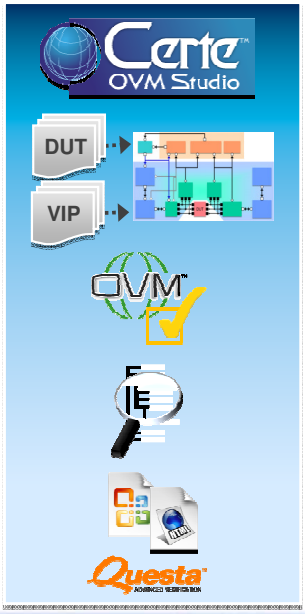
- HW Designers not versed in Object-Oriented (OO) languages
- Time consuming process to learn OO language, types & techniques
- Understanding overall testbench infrastructure
- Assertions, coverage, randomization – creation, understanding
  - Interface / System level, Internal architecture, Functional coverage
  - Constrained random stimulus generation
- Refactoring legacy code to take advantage of new language features

Recommended Solution
Assisted Testbench Development

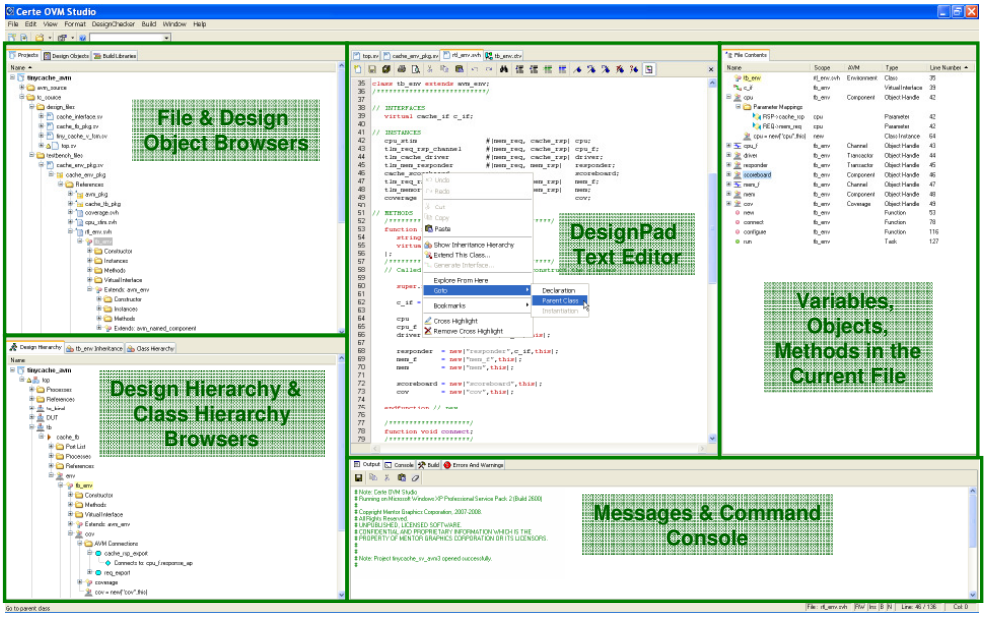
- **What**
  - An enabling environment to ease adoption & use of SystemVerilog, OVM/AVM, & VIP
- **Solution comprised of…**
  - A foundation for OO languages, i.e. SystemVerilog
  - Testbench design & assembly assistance
  - Analysis tools to understand composition, inheritance & relationships
  - SystemVerilog, OVM, AVM compliance checking
  - Links to DUT creation & verification engine
- **Critical attributes for any chosen solution**
  - Non-disruptive process – must fit in any current RTL design flow
  - Testbench creation that enables VIP creation & usage
  - Design checking of syntax, semantics & rules
  - Visualization to aid in understanding & documentation of code

Certe OVM Studio

- **Testbench Design**
  - Advanced verification language text-based assisted editing
  - VIP reuse & creation
  - Class browsing
- **Analyze**
  - File contents & relationships
  - X-referencing
  - Verification & OVM rule checking
- **Visualize**
  - Testbench structure & connectivity
  - Class inheritance
  - Documentation generation
- **Output**
  - Questa Makefile



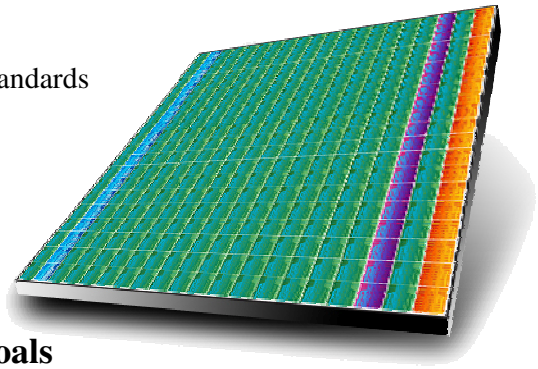
The Certe OVM Studio UI



Precision Synthesis

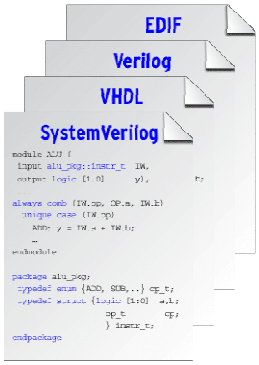
Standards Support + Advanced Optimizations + Analysis Leadership = Meeting Design Goals

- **Start right**
  With industry's best support for HW languages and constraint standards
- **Synthesize**
  Out-of-the-box advanced optimizations and user control
- **Meet aggressive design goals**
  With award-winning analysis & debug



Start Right

Industry-Leading Support for HW Languages & Constraint Standards



- **"What"**
- **Language support leadership**
  - SystemVerilog, Verilog, VHDL, EDIF

Technology Leadership

- **Comprehensive constraint support**
  - Synopsys Design Constraint (SDC) industry standard
  - Quick design constraining
- **ASIC prototyping**
  - Same HDL and constraint files
- **Integrated & optimized flow with Mentor tools**
  - ESL-to-gates (Catapult Synthesis)
  - Language compatibility with verification (ModelSim, Questa)
  - Design reuse, creation, and management (HDL Designer)
  - FPGA-to-PCB (I/O Designer)

Technology Leadership

The Need for FPGA Vendor Independence

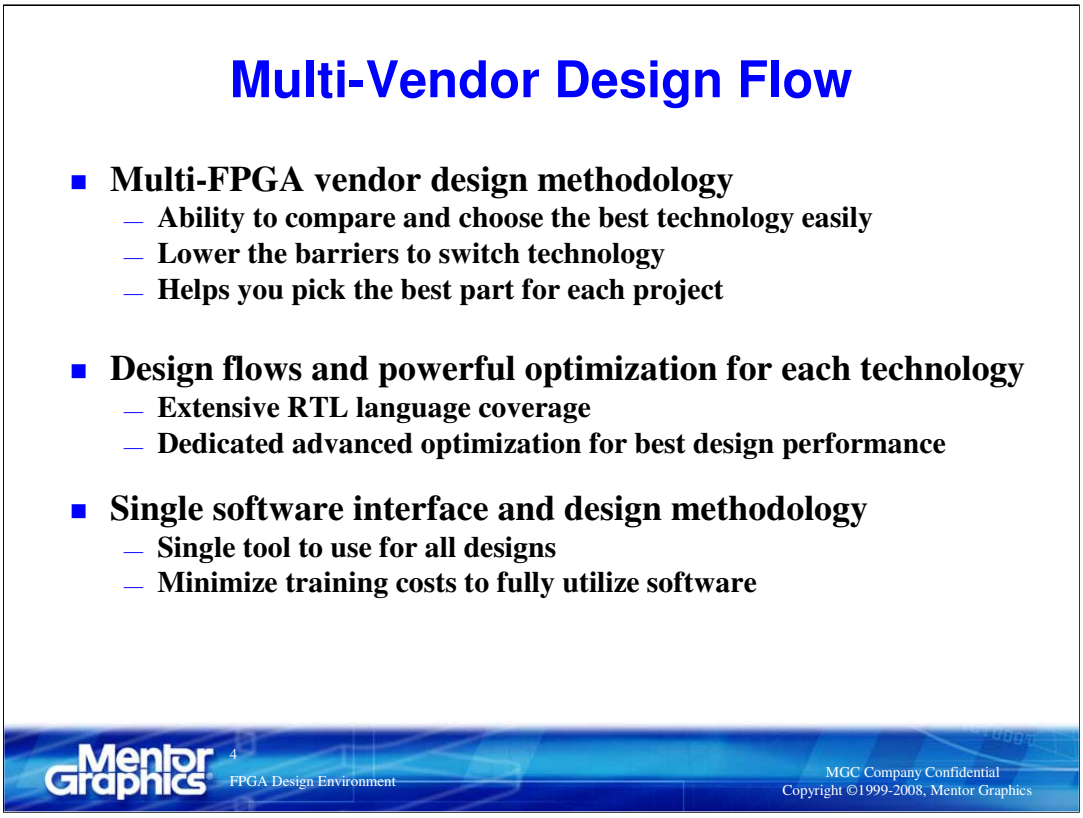
- **The correct technology for your project**
  - **Fastest system performance**
  - **Faster design cycles**
  - **Least cost to manufacture**

Vendor Independent Design Methodology

Best Technology for Your Design

Using a vendor independent methodology for evaluation, you will find the best technology for you design that will produce the fastest system performance and thus reduce your design cycle. You will also be able to compare technologies to find the least expensive device that meets your needs.

Having many choices between vendors will give you the best technology for your design.



A multi-vendor design flow has advantages over FPGA vendor tools and single-vendor OEM tools. With a multi-vendor design flow, you are able to compare different FPGA technologies side by side, using the same source code and the same timing constraints. You are able to see the same timing reporting in a consistent format and easily do a comparison to choose the best technology for the design.

Multi-vendor tools provide design flows and optimization flows specific for each technology supported. Precision synthesis provides extensive RTL language coverage and is available to understand most language constructions while still providing an optimal solution. Technology mapping has also progressed to a level where every advantage of an FPGA technology can be utilized.

Another consideration in using a multi-vendor solution is that a single tool is used to implement any of the FPGA architectures. The cost of switching between architectures is not as high because you will not have to be trained on new tools to use a new architecture.

Generic Code = Choice of Device

- Ability to choose the best device for the design
  - Easily compare devices from different vendors
- Flexibility with generic coding styles
  - Memory and DSP elements inferred
    - Do not rely on a fixed architecture
- *Rely on Synthesis* to implement best structures for target architectures
  - Silicon vendor independence ensures best implementation
  - Interference vs. Instantiation

Vendor Independent RTL Optimization

- Independent of target architecture
  - Logic must be interpreted and optimized
    - RTL can be mapped to technology gates
- Most beneficial to both area and performance
  - RTL optimization
    - Boolean optimization
    - Constant propagation
  - Resource sharing
    - Duplication reduction
  - Inferencing
    - Memory, DSP, arithmetic operators

Before RTL can be mapped to technology gates, the logic must be interpreted and optimized, regardless of the target architecture

RTL Inferencing Advancements

- RTL code now used to generically describe
  - Logic
  - Operators
    - Arithmetic functions, counters, muxing
  - Memories
    - Flip-flops, shift registers
    - Single-port, dual-port, and multi-port RAM
  - DSP elements
    - Multipliers, multiply-accumulate, pipelining
- Benefits
  - Improved performance
  - Smaller implementations
  - Decreased runtime during technology mapping

Timing Constraints

- Timing constraints describe "When" events should happen
- Important to fully constrain a design
  - "Unconstrained" also means no violations are seen
    - Failures only seen after testing on the board
  - Downstream tools do not alert you …even after running for many, many hours
    - You have to find them manually
      - Iterations and wasted time

I/O Timing Affects System Performance

- Have to account for I/O timing
  - Internal reg-to-reg is NOT system frequency
- Aggressive timing on the I/O will make performance on the board a challenge
- I/O timing in large FPGA systems is key to system performance and reducing iterations
  - Fewer RTL changes
  - Fewer synthesis iterations
  - Fewer place and route iterations

Analyzing Your Timing

- **Report all clocks and 'n' worst slacks**
  - **Clocks and edge separation reported**
- **Interactive reporting**
  - **What happened to paths that are no longer critical?**
  - **GUI interaction and/or script-based**
  - **Investigate all the timing paths of interest**
  - **No need to re-run synthesis every time**

Timing is a critical part of getting the design working. Timing closure, and knowing where problems may be, is an important piece of quickly improving the design. We only need to look at the parts that are having problems.

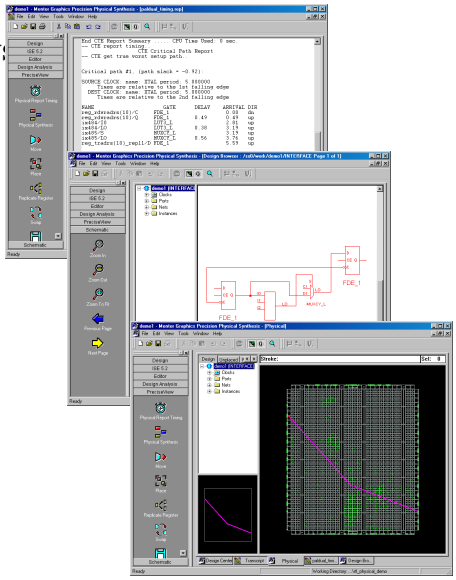
Analyzing Your Timing (cont)

- **Changing constraints & re-running synthesis?**
  - **Adds iteration time**
  - **Does not give the same netlist results**
  - **Is a hit-and-miss process**
- **Preferable to do comprehensive timing analysis *before* P&R**
  - **Change constraints - re-run timing analysis**
    - **Save a synthesis/P&R iteration**
  - **Add constraints - re-run timing analysis**
    - **Save a synthesis/P&R iteration**

While synthesis is a powerful technology, it takes time and by producing a new and possibly different netlist, it opens the door for producing different results. Re-running synthesis in Precision is unnecessary when we are looking at timing. We can very easily change a timing constraint and just re-evaluate timing of the design against the new constraint

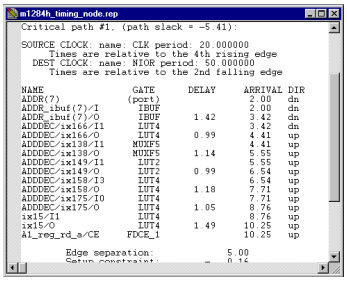
Accurate Timing Analysis & Debugging

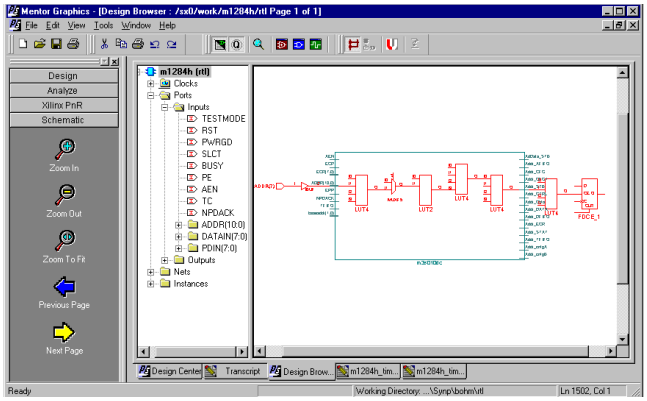
- Powerful Integrated Timing Analy
  - Incremental timing
    - Fast optimization feedback
  - Perform “what-if” analysis
    - Without re-synthesizing
  - Easily identify design issues
    - Critical instances
    - Long interconnects
  - Report clock domain crossings
  - Missing constraints report



Critical Path Viewing and Tracing

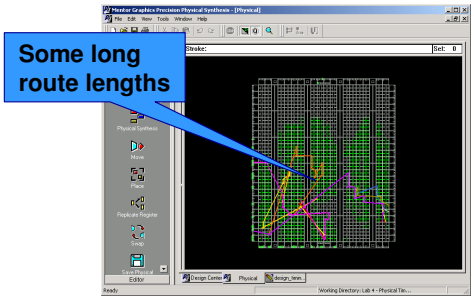
- Logic level timing closure issues easily identified with critical path fragment schematics



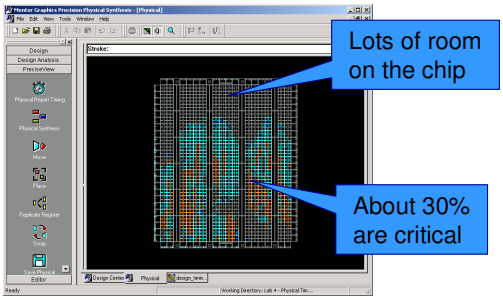


Integrated Timing and Physical Views

Display Critical Paths



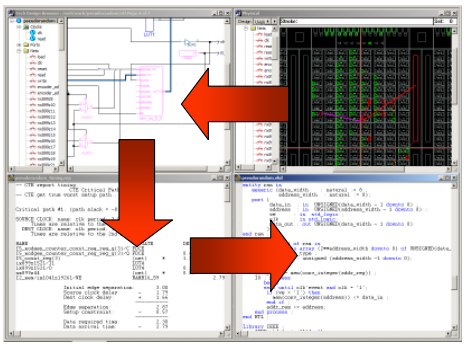
Display Critical Instances



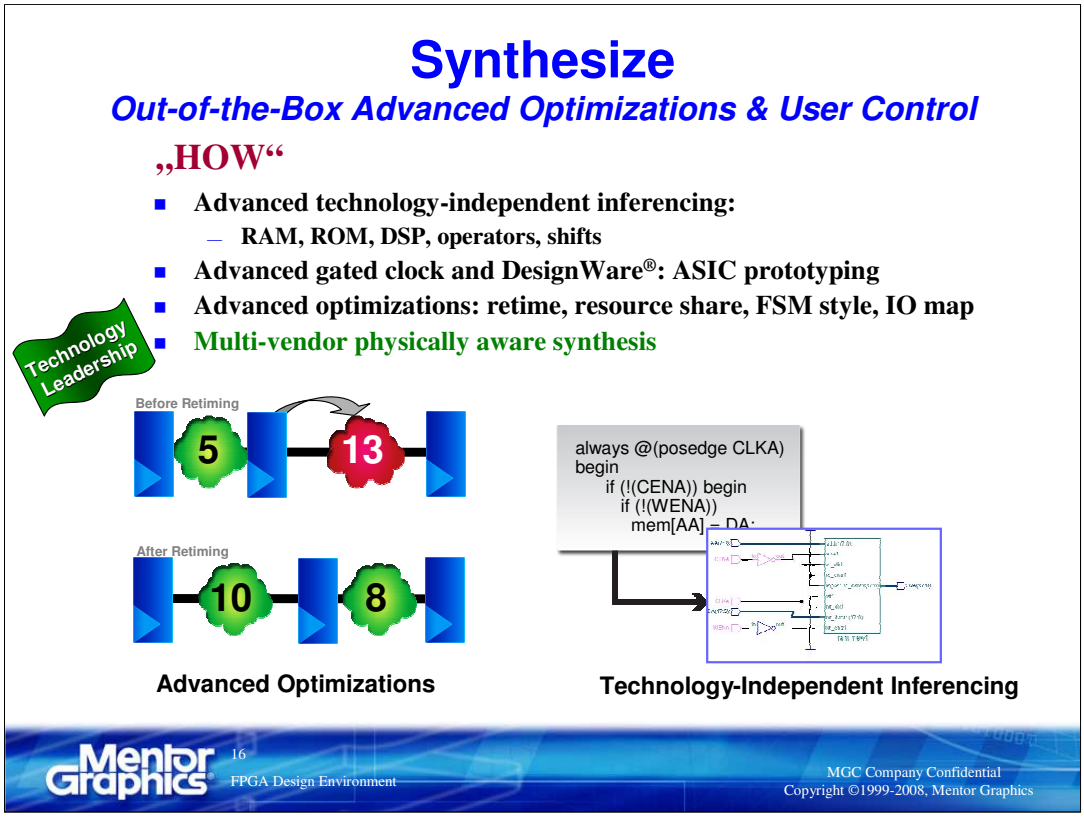
- Easily comprehend the extent of a design's timing closure issues
  - Timing information in easy to understand colorized physical format
- Assists in determining best solution to timing closure issues
  - Are there non-critical regions?
  - Are critical paths spread across a chip?

Efficient Analysis with Cross Probing

- **Analyze the Actual Physical Design Implementation**
  - **Post-P&R information**
    - **Placement**
    - **Routing delay**
  - **Cross-probing between 3 domains**
    - **Physical domain**
    - **Gate-level domain**
    - **RTL domain**
  - **Critical paths views**
    - **Physical view**
    - **Gate level technology view**



Precision includes a complete physical design analysis environment. Precision includes a powerful analysis environment that can annotate physical timing and placement information from place and route. In a single integrated environment Precision can quickly determine how best to solve a timing closure problem through timing, schematic and placement views of the design.



Precision has capabilities that help customers get their designs started on the right foot with strong push-button results, but our tools

go beyond that. Precision provides the tools and flows for our customers to be able to more reliably meet their design requirements, on schedule and at cost

targets. Unlike other synthesis products, our tools are compatible with industry standard ASIC design constraints so that as more

designers move over from the ASIC design space, our tools work more closely in the way customers think and that is familiar to them.

As designs become more complex, it becomes equally important to help the designer solve functionality, timing and performance problems in the design quickly.

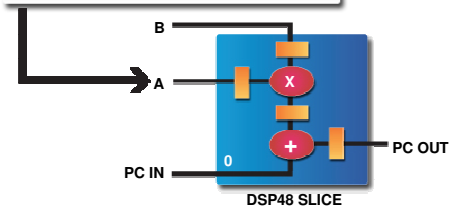
By leveraging our physically-aware synthesis process, we are able to provide great design insight to customers, so that customers can identify and solve functional or performance bottlenecks. For example ... You can cross probe from implementation and timing issues to the specific HDL code that creates the issue and fix problems at their source.

Advanced Technology-independent DSP Inferencing

Enables users to write FPGA Vendor neutral HDL code → making designs portable across FPGA Vendors and devices

- DSP inferencing
  - Significant area and performance gains
  - Full support for DSP functions in advanced FPGA architectures
  - More efficiently cascade DSP blocks → major improvements in digital filter designs

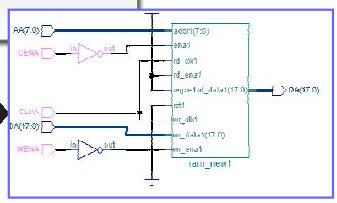
```
always @(posedge clk)
begin
mult0_result <= A * B;
PCOUT<= mult0_result + PCIN;
end
```



Advanced Technology-independent FSM and Memory Inferencing

- **FSM control**
  - **Optionally map into available synchronous RAM elements**
- **Memory inferencing**
  - **Better device utilization & improved performance**
  - **RAM, ROM, Dual Port RAM**
  - **Technology neutral code enables device independence**

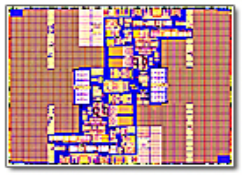
```
always @(posedge CLKA) begin
    if (!(CENA)) begin
      if (!(WENA))
        mem[AA] = DA;
      QAi = mem[AA];
      end
    end
assign QA=QAi;
```

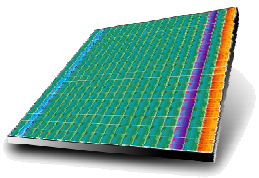


For significant area and performance gains Precision uses advanced optimization techniques like DSP inference.

Extensive ASIC Gated Clock Conversions

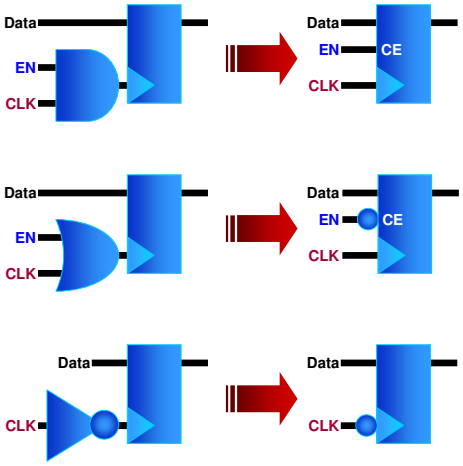
- **Automatic gated clock conversion**
  - ASIC clock networks transitioned to FPGA global clock networks
  - Gated output re-routed to clock enable
    - Block RAM
    - DSP & multipliers
    - Latch
  - Through hierarchies
- **Advanced gated clock conversion**
  - Complex gated clocks
  - Generated clocks
  - Multiplexed clocks
  - Inverter gated clocks
- **Easy RTL reuse**





For Latches, Shift-registers, RAM, Block Multipliers and DSP blocks
Minimizes clock skews

Gated Clock Conversion



- **Benefits**
  - Minimize clock skews in FPGA designs
  - Use the same RTL ASIC source files
- **Automatic Conversion**
  - Re-route to clock enables (when applicable)
  - Re-implement with functional equivalence
- **Support**
  - Flip-Flops, latches
  - RAMs
  - Counters, multipliers, DSPs

Synopsys DesignWare® Support

```
//VERILOG
// No includes needed

DW01_add #(32) myadder (.A (a),
   .B (b), .CI (c_in), .SUM (sum),
   CO (c_out));

-- VHDL
-- LIBRARY DECLARATION
LIBRARY  DW01;
USE DW01.DW01_COMPONENTS.ALL;

component DW01_add
-- Description
  end component;
```

- DW support enables HDL reuse between ASICs & FPGAs
- Minimal user intervention
  - Transparent flows for DW
  - Auto detects DW01, DW02 & DW RAM
  - Automatically exchanges with optimal & compatible implementations
  - Black boxes unsupported DW

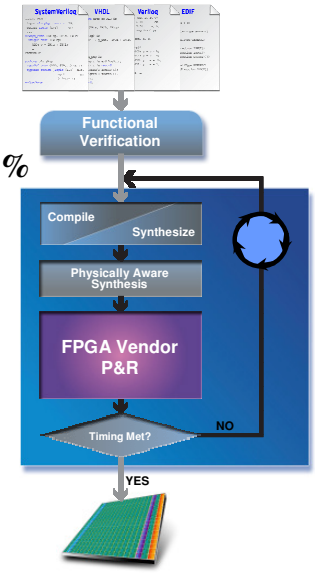
Technology Leadership

Physically Aware Synthesis

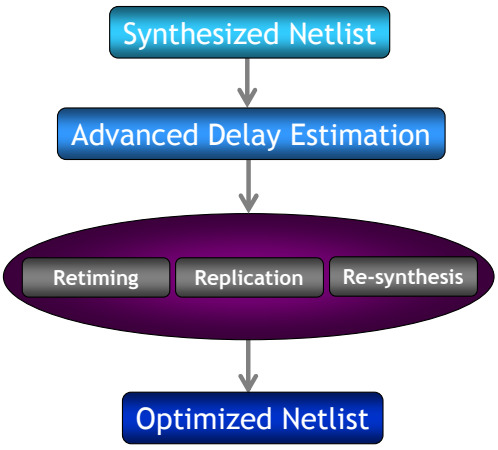
The Only Multi-Vendor Physical Synthesis

Reach design goals faster, in fewer iterations

- **Average 10% Fmax improvement, typical 5-40%**
- **Industry's only multi-vendor physical synthesis**
  - **19 FPGA device families supported**
    - **5X the nearest competitor**
    - **Actel, Altera, Lattice, Xilinx**
  - **Push-button so every designer benefits**
  - **Pre-P&R physical synthesis creates an optimized netlist**
    - **Fewer P&R iterations**
    - **Improved netlist → faster placement & routing**

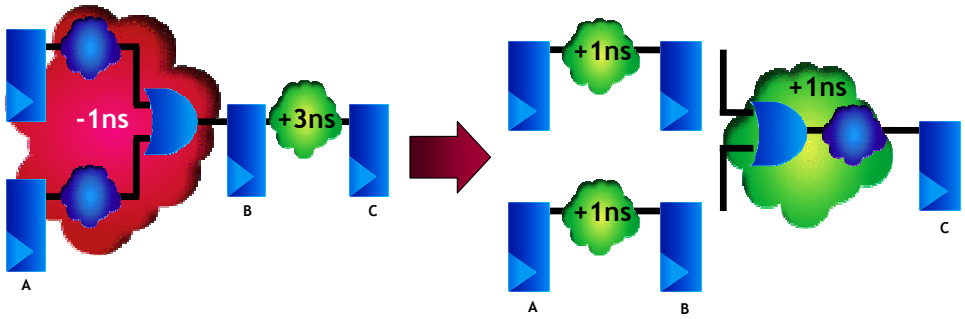


Physically Aware Synthesis - Technology



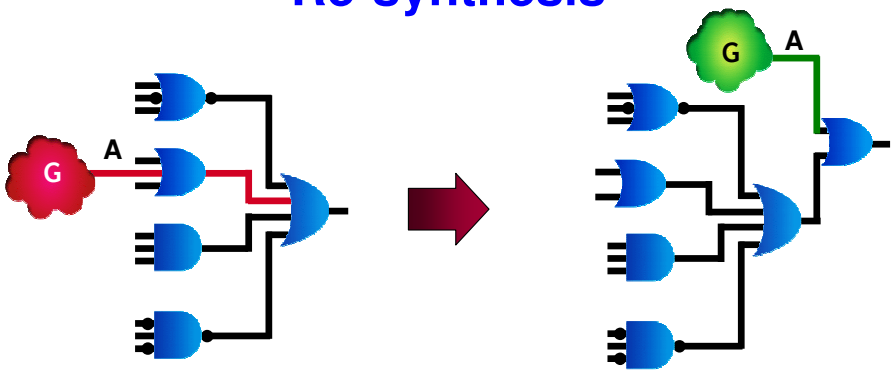
- **Advanced delay estimation**
  - Estimated location
  - Routing resources
  - Design rules
- **Netlist optimizations**
  - Pre Place & Route
  - Uses more accurate net delays
  - Scalable across devices
- **No placement sent to P&R**
  - No packing violations
  - Maximum flexibility for P&R

Register Retiming



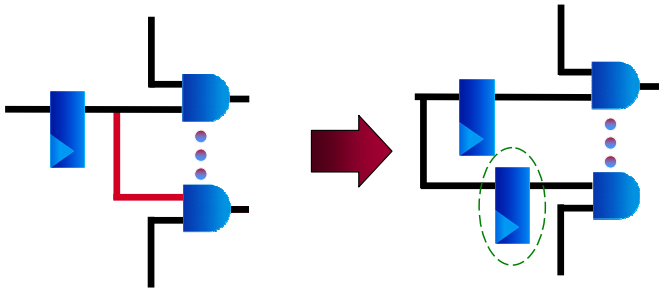
- Register retiming to balance slack
- Reduce logic levels between registers
- Retiming in both forward & backward directions
- Multiple iterations until negative slack reduced or eliminated

Re-synthesis



- Restructure logic to improve timing
  - Local Logic Optimization
- Transformed ‘A’ from critical to non-critical path
  - Shorten path by logic re-synthesis

Replication



- Replicate registers to reduce routing delay
- Duplicate and move registers across interconnect

Meet Aggressive Design Goals

Award-winning Analysis & Debug

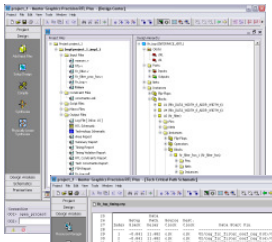
- Graphical critical paths views
- Cross-probing to source
- Missing constraint report
- Domain crossing report
- Interactive physical controls
- "What-if" analysis: no re-synthesis
- Fast/predictable incremental flows
- Resource analysis and control

Technology Leadership

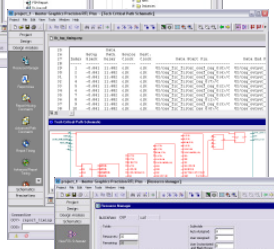
Technology Leadership

Precision voted *highest satisfaction* in logic synthesis analysis by

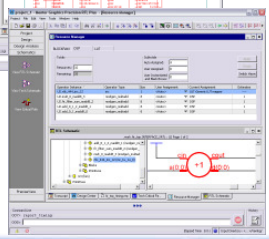
FPGA and Structured ASIC JOURNAL



HDL Source Code, Missing Constraints & Timing Reports



RTL & Schematic Views



Resource Manager

Mentor Graphics 27 FPGA Design Environment

Precision has capabilities that help customers get their designs started on the right foot with strong push-button results, but our tools

go beyond that. Precision provides the tools and flows for our customers to be able to more reliably meet their design requirements, on schedule and at cost

targets. Unlike other synthesis products, our tools are compatible with industry standard ASIC design constraints so that as more

designers move over from the ASIC design space, our tools work more closely in the way customers think and that is familiar to them.

As designs become more complex, it becomes equally important to help the designer solve functionality, timing and performance problems in the design quickly.

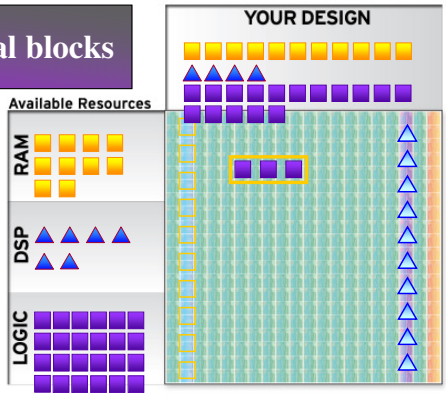
By leveraging our physically-aware synthesis process, we are able to provide great design insight to customers, so that customers can identify and solve functional or performance bottlenecks. For example ... You can cross probe from implementation and timing issues to the specific HDL code that creates the issue and fix problems at their source.

Technology Leadership

Resource Manager

Patent-Applied-For Technology

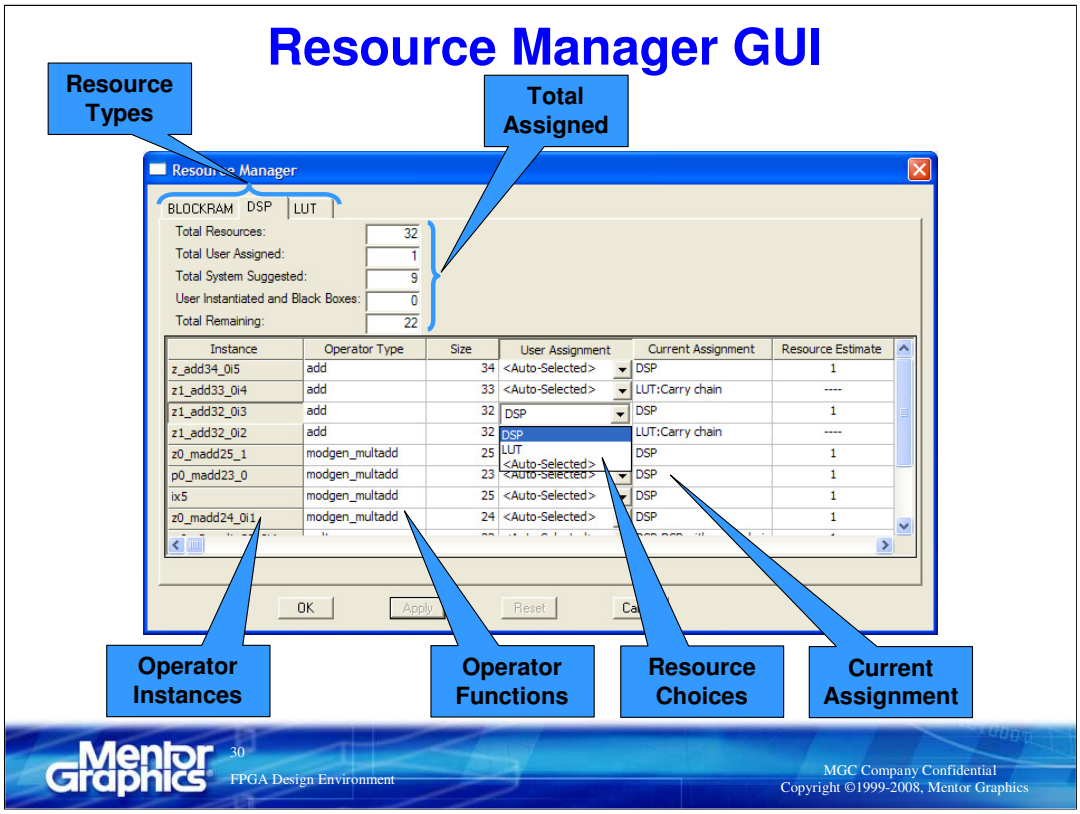
Make efficient use of FPGA architectural blocks



- **Industry-first technology in resource analysis and control**
  - **Identifies available resources**
  - **Allows remapping resources for**
    - **Trading-off fabric vs. dedicated block utilization**
    - **Critical path timing improvement**
  - **Provides resource budgeting control for team-based design**
  - **Cross-probes to HDL and schematic views**
  - **Intuitive, easy-to-use interface**

Resource Manager Technology

- Operates on technology-specific embedded blocks
  - RAM blocks
  - Multipliers
  - DSP blocks
- Synthesis allocates resources using heuristics
  - Considers embedded blocks to be available for logic
  - Size of the operator is considered
    - For example, small RAMs mapped to logic core
- Multiple implementations on available resources
  - Map operators to various hardware resources
  - Example: Virtex-4 adder can be mapped to DSP48, LUT-based carry chains, or LUTs only


Resource Manager GUI
Resource Types
Total Assigned
Resource Manager
BLOCKRAM
DSP
LUT
Total Resources: 32
Total User Assigned: 1
Total System Suggested: 9
User Instantiated and Black Boxes: 0
Total Remaining: 22
Instance
Operator Type
Size
User Assignment
Current Assignment
Resource Estimate
z_add34_0i5 add 34 <Auto-Selected> DSP 1
z1_add33_0i4 add 33 <Auto-Selected> LUT:Carry chain ----
z1_add32_0i3 add 32 DSP DSP 1
z1_add32_0i2 add 32 LUT:Carry chain ----
z0_madd25_1 modgen_multadd 25 DSP 1
p0_madd23_0 modgen_multadd 23 DSP 1
ix5 modgen_multadd 25 <Auto-Selected> DSP 1
z0_madd24_0i1 modgen_multadd 24 <Auto-Selected> DSP 1
DSP
LUT
<Auto-Selected>
OK
Apply
Reset
Operator Instances
Operator Functions
Resource Choices
Current Assignment
30
FPGA Design Environment
MGC Company Confidential
Copyright ©1999-2008, Mentor Graphics

Minimizing the Impact of ECOs

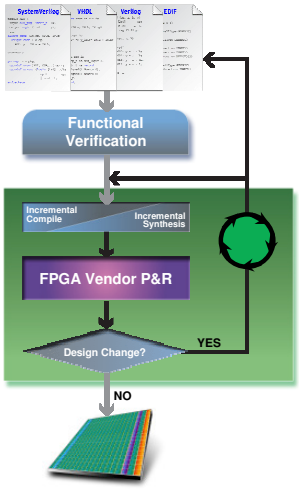
- **Engineering Change Order (ECO):**
  - **A small functional change**
- **Problem:**
  - **Design changes → netlist changes**
  - **Back to square one: re-run Place & Route**
    - **Long iteration cycles**
    - **Performance not retained**
- **Solution:**
  - **Incremental synthesis and Place & Route flows**
  - **Re-synthesize and Place & Route only the modified portions of the design**
    - **Unaffected areas retain performance and area**

ECOs generate fear of losing design closure especially on designs that have already met timing. Changes to a netlist can significantly affect the place and route performance and predictability. What if there was a way to save the placement and re-apply it before place and route? There is! with Precision's Reuse Placement feature you give the vendor based place and route a much better starting point than just starting from scratch.

Incremental Synthesis

Handle Late Cycle Design Changes Quickly & Predictably

Minimize the impact of late cycle design changes



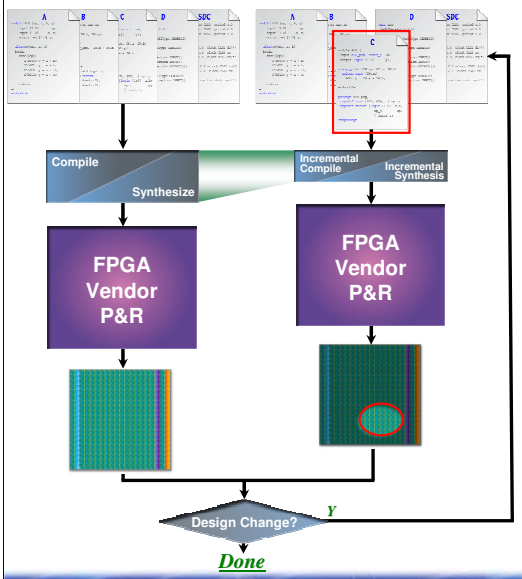
- Two incremental flows
  - Automatic incremental synthesis
    - No setup or partitioning necessary
    - All FPGA families supported
    - Fully automatic through P&R with Xilinx SmartGuide flow
  - Partition-based incremental synthesis and P&R
    - Efficient for team-based design
    - For Altera & Xilinx flows

Technology Leadership

Technology Leadership

Automatic Incremental Synthesis

Industry First, No Partitioning Needed

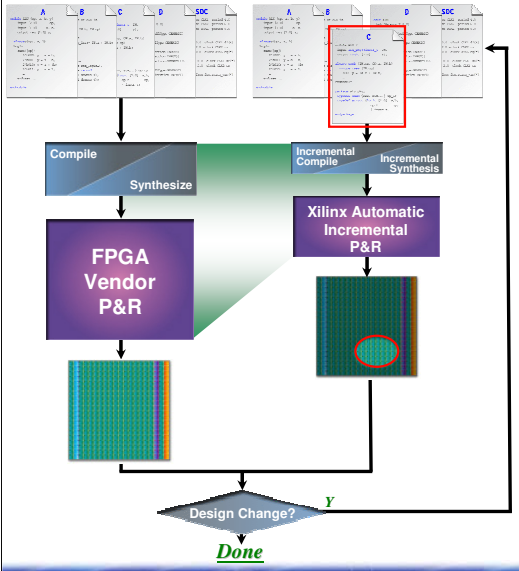


- Up to 60% synthesis runtime savings
  - Design, change dependent
- Industry's only automatic incremental synthesis
  - No partitioning or prior planning
  - Maintains QoR
    - Cross-hierarchy optimization
  - True incremental
    - Based on parse tree
- All FPGA families supported

Technology Leadership

Automatic Incremental Synthesis w/ Xilinx SmartGuide

Industry First – Fully Automatic Incremental Synthesis + P&R Flow

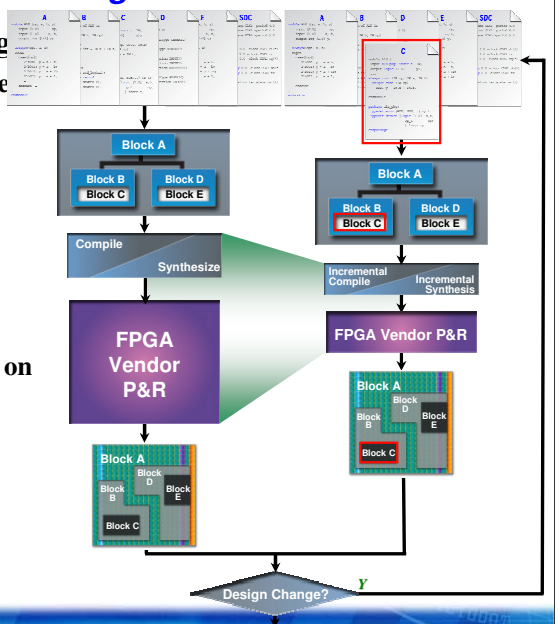


- **3X average and up to 10X runtime savings**
  - **Greater than 90% predictability of instances/net changes**
  - **Design, change dependent**
- **Industry's only automatic incremental synthesis + P&R**
  - **Performs synthesis and P&R only on changes**
  - **No partitioning or prior planning**

Partition-Based Incremental Synthesis & P&R

Localizes Changes

- Up to 6X synthesis runtime saving
  - 100% predictability for unchange blocks
  - Design, change dependent
- Divide & conquer for team-based design
  - Localized effects of changes
    - Preserves unaffected partitions
  - Automatic incremental synthesis on changed partition
- Supports Altera and Xilinx incremental P&R



Partition-Based Incremental Flow

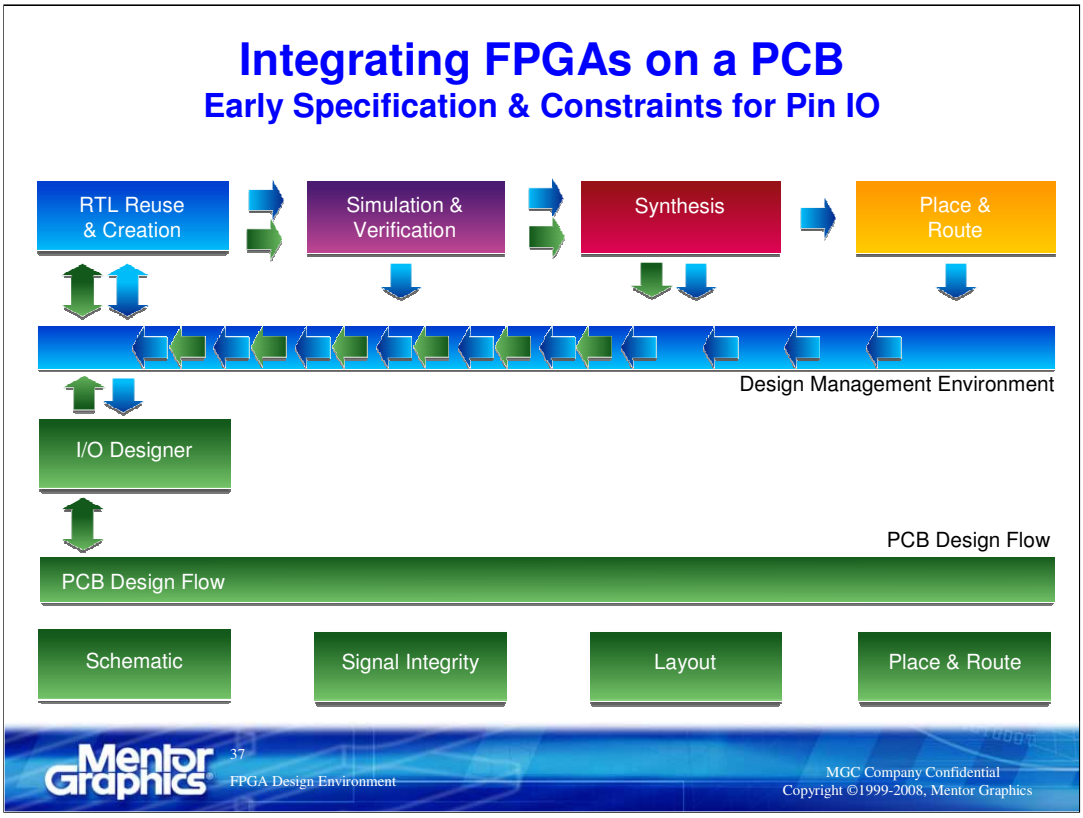
- User knows logical partitions of design
- User sets attributes within HDL
- Precision honors the partition boundaries
  - Synthesizes partition only
  - Applies automatic incremental synthesis within the partition
  - No cross boundary optimization between partitions

ALTERA®

- Altera Flow
  - Precision creates separate EDIFs for partitions
  - Forwards EDIFs and TCL script
  - Quartus MAP processes the design

XILINX®

- Xilinx Flow
  - Precision puts attributes in EDIF
  - ISE reads attributes to understand partitions
  - ISE MAP processes the design



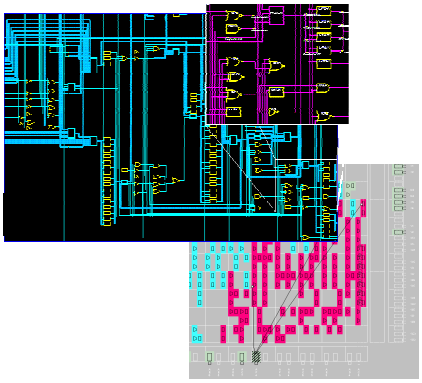
In the typical FPGA and PCB design process it is serial in that the FPGA design starts and then after the initial place and route the pin out is known and then the PCB design process can start to create the required symbols and elements for PCB design.

Early consideration of the IO characteristics and using I/O Designer with HDL Designer will impact the system integration success and reduce the redesign cycles that often occur once the FPGA is sent over to PCB layout designers. The system performance, PCB layers and routing are all directly impacted by the pin locations and characteristics.

- **Add Some Physical IO Definitions (clocks, etc..)**
- **Complete Physical IO Definitions (inside the chip)**
- **PCB Schematic Integration**
- **PCB Symbol and Package Creation**

FPGA Design Challenges…
from the I/O Perspective

- Increasing levels of complexity inside the FPGA
  - Multiple I/O banks with varying power requirements
  - Ever increasing number of I/O standards (50+)
  - High-speed differential signaling
  - Embedded IP
- Increasing physical complexity
  - Increasing pin-count per device
  - Shrinking feature sizes
  - Faster clock rates
  - Multiple FPGAs per PCB



Serial Design Process Not Effective

- **Traditionally, FPGA and PCB teams collaborate *after* the FPGA logical design is complete**
  - **FPGA pin-out will not be optimal for the PCB**
  - **The PCB may not meet its timing constraints**
  - **Changes at this point impact time-to-market**
- **Two disciplines are responsible for the design**
  - **FPGA Design Engineer for the logical design**
  - **PCB Layout Engineer for physical implementation**

FPGA Design Challenges…
from the PCB Perspective

- **Generating the schematic symbols is tedious and difficult**
  - **A time-consuming manual process**
  - **Difficult to fit large symbols on one page**
  - **Mapping the symbol to the physical device is error prone**
- **Connecting the device to the rest of the design**
  - **A time-consuming manual process**
- **No control over the pin assignments**
  - **Difficult to meet High Speed signal constraints**
  - **Difficult to route and optimize the board**
  - **Fixed pin-out so no means of optimizing routing on the PCB**

And…if anything changes it has to re-done!

Problems from PCB perspective:

Generating symbols and mapping them to physical cell definition

Manual, time consuming and error prone.

Physically 1500+ pin components are too big for page.

Device has to be wired up manually.

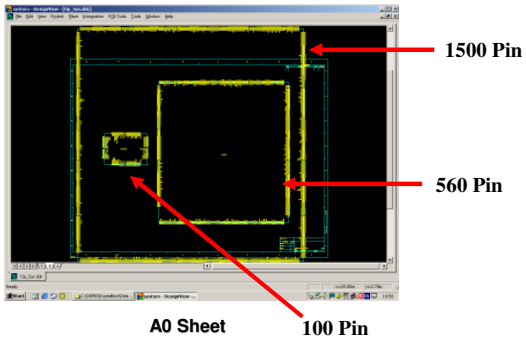
Optimisation:

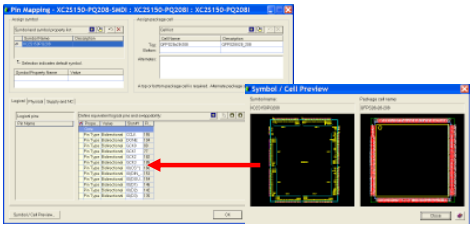
Given a fixed pin-assignment with no information about pin swappabilty.

Makes it difficult to optimise the FPGA for routability or high-speed constraints.

Symbol Creation Challenges

- Data entry
  - Excel
  - Paper
- Symbol physical size
  - Unreadable
  - Unplaceable
  - Unusable
- How to:
  - Split up (fracture) the symbol?
  - Update the symbol?
  - Physically map the symbol?





Where does the symbol information come form, device data sheet, excel or some other file if lucky.

Large devices cannot be created as monolithic symbols as they don't fit on the page, how do you determine the best way to split up the symbol.

If something changes how is the information communicated.

Traditional FPGA Pin Assignment

- **Automatically by FPGA Place & Route tools**
  - **No PCB layout constraints**
  - **No complex constraints**
  - **Little designer control/influence**
  - **Optimized for FPGA, not the PCB**
- **Text editing or Excel spreadsheet**
  - **No pin information available**
  - **Time consuming**
- **Manually in FPGA Place & Route tool**
  - **Just for single FPGA vendor**
  - **No PCB Layout constraints**

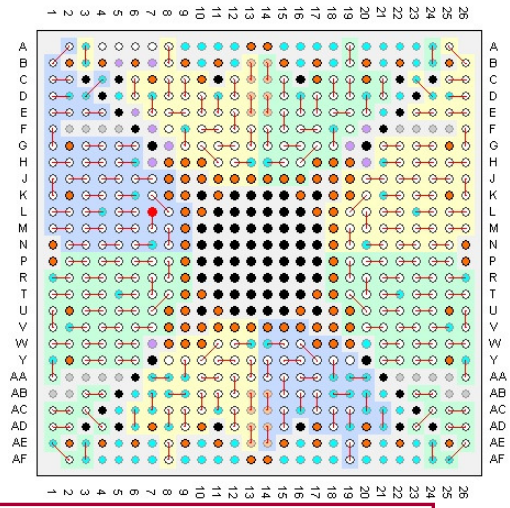
```
#PINLOCK_BEGIN
CONFIG PART=2v3000-ff1152-6;
#PINLOCK_END


#I/O Pin Assignments
NET "A_EDGE(0)"  LOC = "H16" ;
NET "A_EDGE(1)"  LOC = "H15" ;
NET "A_EDGE(2)"  LOC = "H14" ;
NET "A_EDGE(3)"  LOC = "H13" ;
NET "A_EDGE(4)"  LOC = "H12" ;
NET "A_EDGE(5)"  LOC = "H11" ;
NET "A_EDGE(6)"  LOC = "H10" ;
NET "A_EDGE(7)"  LOC = "H9"  ;
NET "ALUOUT(0)"  LOC = "AF3" ;
NET "ALUOUT(1)"  LOC = "AF4" ;
NET "ALUOUT(2)"  LOC = "AF5" ;
NET "ALUOUT(3)"  LOC = "AF6" ;
NET "ALUOUT(4)"  LOC = "AF7" ;
NET "ALUOUT(5)"  LOC = "AF8" ;
NET "ALUOUT(6)"  LOC = "AF9" ;
NET "ALUOUT(7)"  LOC = "AF10" ;
NET "B_EDGE(0)"  LOC = "AJ13" ;
NET "B_EDGE(1)"  LOC = "AJ12" ;
NET "B_EDGE(2)"  LOC = "AG14" ;
NET "B_EDGE(3)"  LOC = "AF14" ;
NET "B_EDGE(4)"  LOC = "AG13" ;
NET "B_EDGE(5)"  LOC = "AF13" ;
NET "B_EDGE(6)"  LOC = "AG12" ;
NET "B_EDGE(7)"  LOC = "AF12" ;
NET "Error_flag" SLEW = FAST ;
```


In a typical pin assignment task for an FPGA you will not have any constraints or restrictions. Also, the pin assignments are typically created and passed via text or a spreadsheet to the PCB designer after they have manually entered the information into the P&R tool. With the complexity as we spoke about earlier, this process has gotten much more complex due to the different I/O standards and banks that the FPGAs now have available to the designer.

FPGA Pin Swapability

- **FPGAs are fully configurable, but their pins have**
  - **Different I/O standards**
  - **Different banks – Voltages**
  - **Differential pair pin-pairs**

  So not all pins can be freely swapped → bank rules



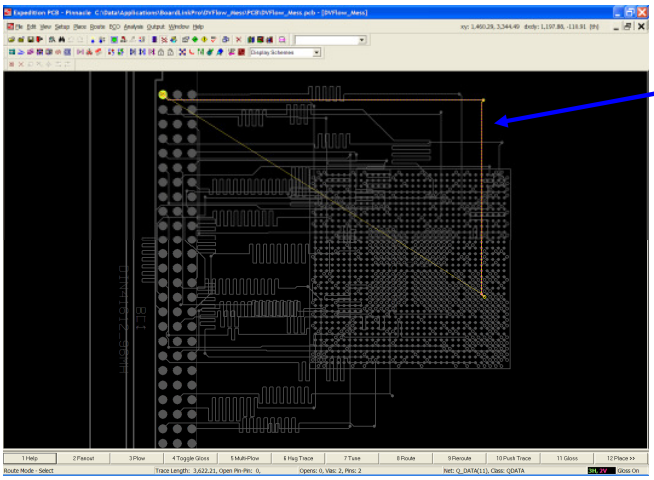
Each FPGA design can have a different pin swap matrix for the same device!

FPGA's are highly configurable and flexible making them ideal for creating complex systems.

The down side is they are no-longer design independent from the PCB design perspective. They have different I/O standards, voltages and pin assignments. And unique pin-swap compatibility.

We need some level of automation to ensure pin-swap data is valid.

FPGA Pin Assignment – P&R



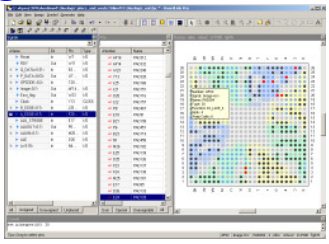
- **Automatic P&R of pin outs is not optimal – scatter of IO for a bus all over the chip**
- ***An intelligent solution is required…***

Resulting routing using automatic P&R

- **Matched BUS all nets same length**
- **Longest net – Q_Data(11) 3.622 inches**

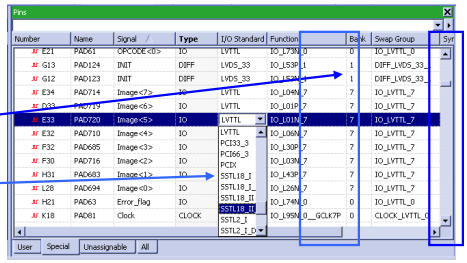
Typically the PCB designer would match the route lengths to reduce skew especially in the case of a bus. So, scattered pin outs and poor placement choices can seriously effect the performance as well as the PCB complexity (layers, vias, signal integrity, etc)

Intelligent I/O Design



- **Pin assignment controlled by both the FPGA and the PCB designers**
  - **Built-in I/O design data management**
  - **Support for scalability (multiple devices)**
  - **Built-in guidance for FPGA constraints**
  - **Support early PCB prototype (No synthesis or P&R step or dummy run needed)**

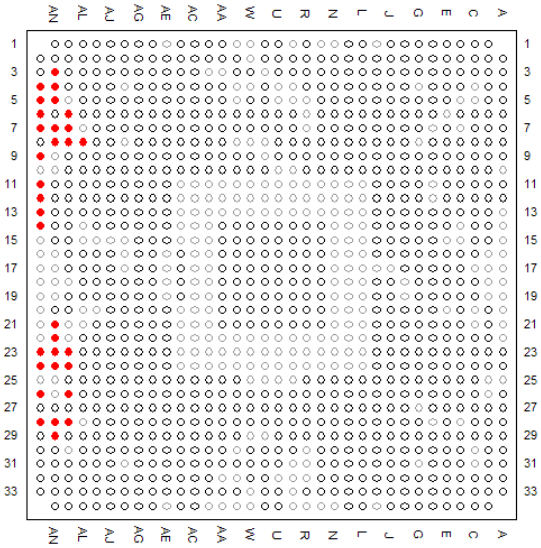
- **All pin information available during I/O design**
  - **Pin number**
  - **Pin function**
  - **Pin type i.e. single-ended or differential pin types**
  - **I/O bank**
  - **I/O standard**



A solution to this solve this problem is the use of intelligent I/O Design. The I/O design must be a managed process and can facilitate the concurrent design/development of the FPGA as well as the PCB. All of the pin information is known and can be set up ahead of time to allow the PCB designer the freedom to optimize the routing while adhering to the rules of the given device.

A tool such as I/O Designer enables the user to intelligently assign signals to pins on the device considering the rules of the device architecture. Hence, enabling the concurrent FPGA and PCB design.

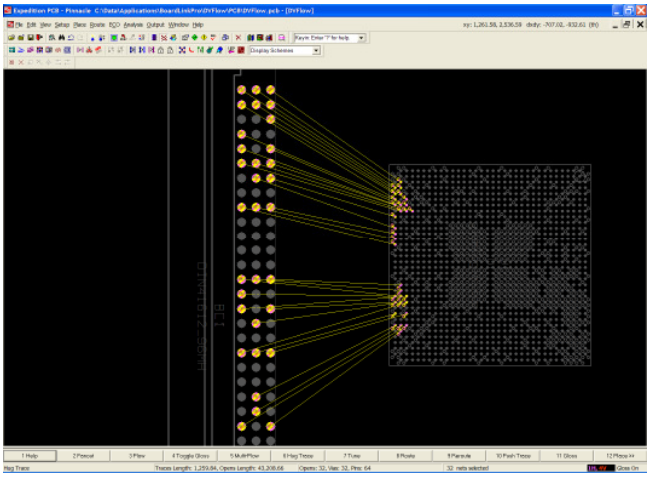
FPGA Pin Assignment



32-bit bus assigned by I/O Designer

In this example the user determined 32 bit bus placement. Logically this makes sense as the signals are all routed towards the side of the chip that the connections are made.

FPGA Pin Assignment



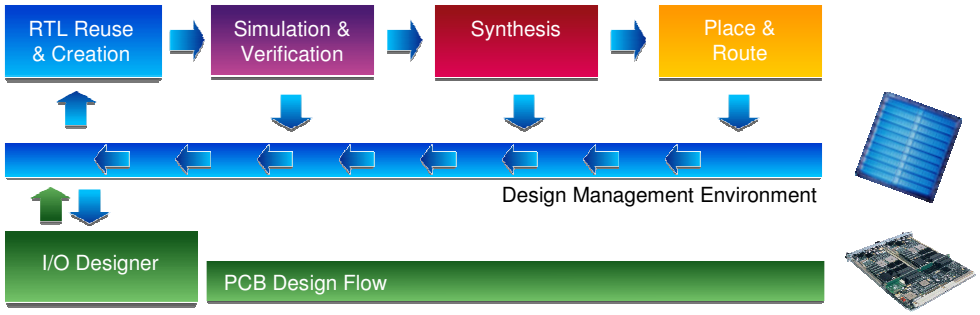
- **Result of further pin swap**
  - **Pin swaps are back annotated to I/O Designer and schematic**
  - **Pin assignment could begin within layout**

Now, with the ability to intelligently manage the I/O's, decisions to swap are based on rules not guess work or going back to the FPGA designer to ask. Nets can be uncrossed thus simplifying routing and the PCB design process.

Mentor Graphics FPGA Advantage

A Complete FGPA Design Flow

- **HDL Designer™ RTL Reuse, Design Creation and Management**
- **Model*Sim*® Simulation**
- **Questa™ Advanced Verification**
- **Precision® RTL Synthesis, Precision® RTL Plus Synthesis and Physical Synthesis for FPGA**
- **I/O Designer™ FPGA Pin Out ↔ PCB Connectivity Design**



Mentor Graphics 48 FPGA Design Environment

MGC Company Confidential
Copyright ©1999-2008, Mentor Graphics

HDL Designer™ - Integrated HDL design with powerful design re-use

Questa™/ModelSim® - Industry leading simulation environment with Assertions

Precision™ Synthesis – Vendor independent HDL synthesis for FPGA design

I/O Designer™ - I/O placement and integration with the PCB design environment

February 20, 2008

Signalverarbeitung mit FPGA's: Möglichkeiten und Herausforderungen

Harald Kreidl, harald.kreidl@nuhorizons.com
Technical Director

Nu Horizons Electronics GmbH
Elektrastraße 6
D-81925 München
Germany
Tel: +49 89 9233345 0

NU HORIZONS ELECTRONICS

Agenda

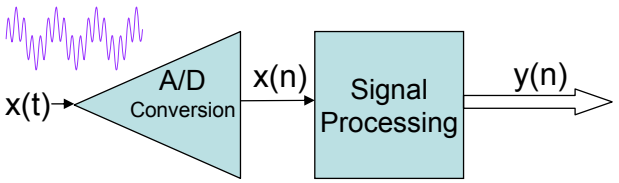
- *The System View*
- Why FPGA's for Signal Processing?
- Typical Signal Processing Algorithms
- Basic Building Blocks of Signal Processing
- What is an FPGA?
- DSP Blocks in the FPGA
- Tools
- Summary

XILINX®

NU HORIZONS ELECTRONICS

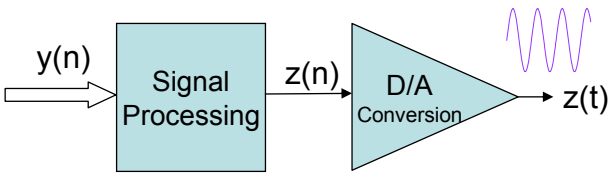
Digital Signal Processing Systems

Signal Analysis



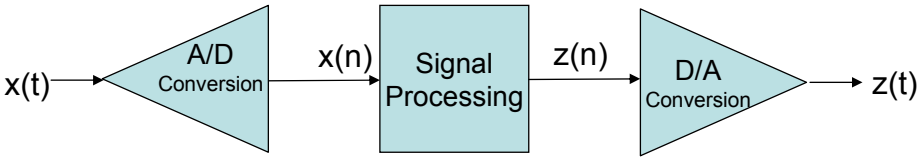
Examples: Receiver/Encoder

Signal Synthesis



Examples: Transmitter/Decoder

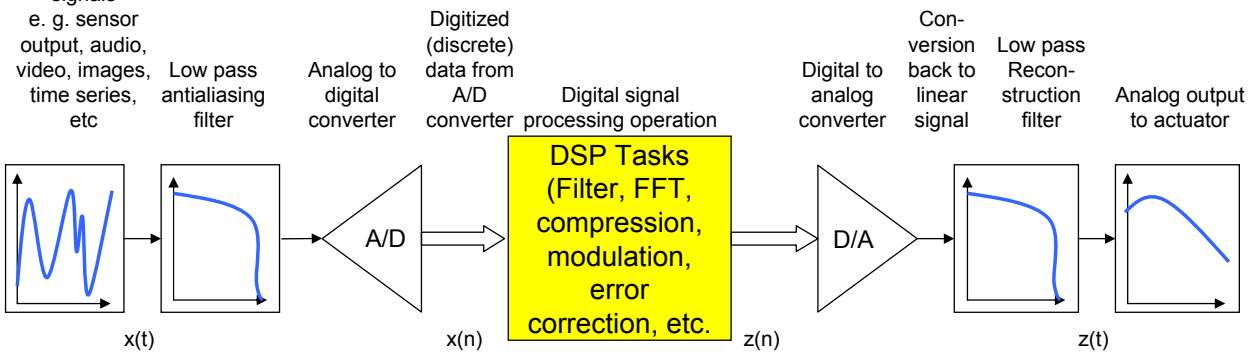
Signal Processing/Control



XILINX®

Digital Control Systems

Signal Processing/Control



Advantages of digital control system:

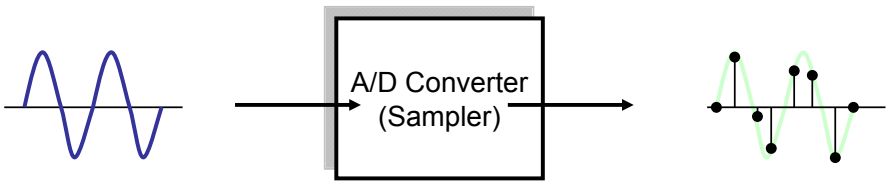
- Flexibility (any algorithms) & programmable & modifyable
- Simulateable ("perfect world" creation)
- Controllable
- No aging (long term stability) & no drift over time
- No adjustments (calibration)
- Self test possibilities
- High noise immunity
- High power supply rejection
- Temperature independent
- Reproducible results, repeatable
- High reliability
- Multiplex possible (multiple hardware utilization)
- Use of techniques not available in nature
- Utilization of transmission media
- Alternative storage techniques
- Secrecy

Examples:

- Modulation/Demodulation (Phone Modems, DSL, GSM, SDR, etc)
- Radar Signal Processing
- Multimedia Coding (MP3, DivX, JPEG, M-PEG
- Noise Cancelation
- Video Processing (pattern recognition, edge filter, etc.)
- Adaptive filters
- Data compression
- Music synthesis

Sampling & Nyquist

Sampling



Analog Signal

Digital Representation of the Analog Signal

Requirements and assumptions

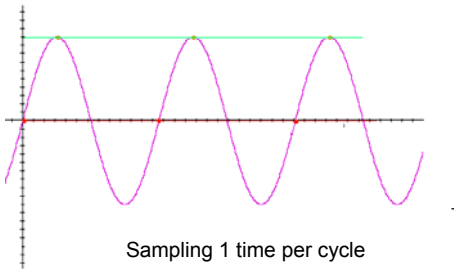
- Digital samples must be accurate in time
- Digital samples must be accurate in amplitude
- Adequate samples to approximate to waveform
- Simulation models

Sampling & Nyquist

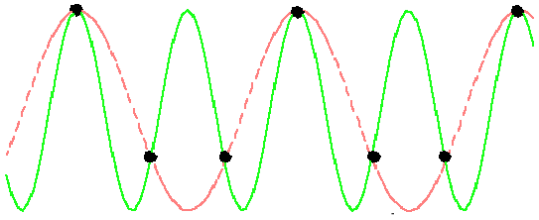
Nyquist

$$f_{sample} > 2 \cdot f_{max}$$

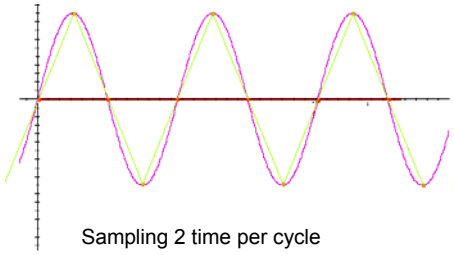
$$f_{sample} \approx 2.2 \cdot f_{max}$$



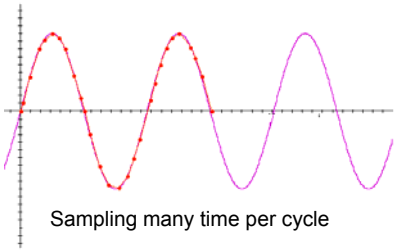
Sampling 1 time per cycle



Sampling 1.5 time per cycle



Sampling 2 time per cycle

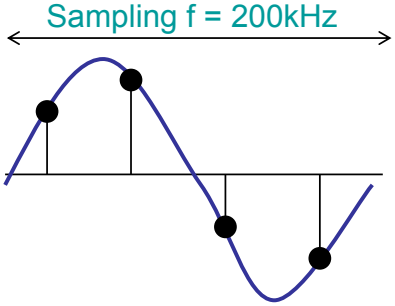


Sampling many time per cycle

XILINX

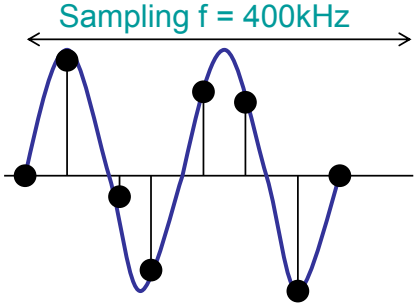
Nyquist Frequency

Input signal = 100kHz
Sampling f = 200kHz



Sampling Frequency needs to be at least 2x the highest frequency of the incoming signal (this is the called the Nyquist frequency)
i.e. => 200kHz
This ensures that the signal can be correctly reconstructed after it has been modified using DSP.

Input signal = 200kHz
Sampling f = 400kHz

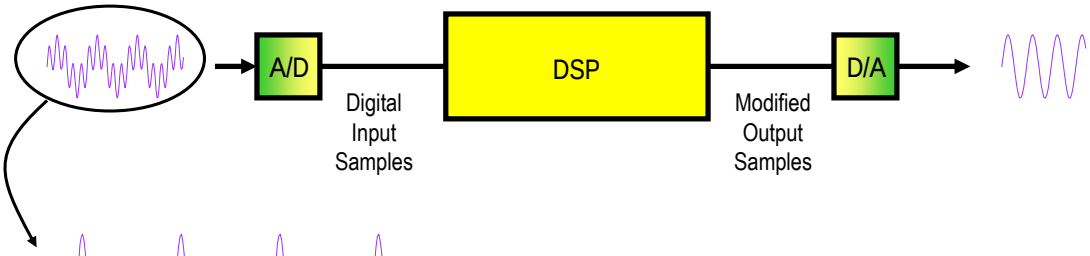


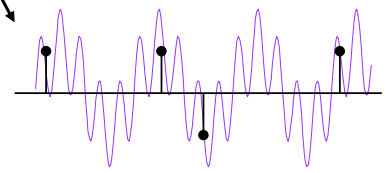
Now a sample rate of 200kHz is not enough!
We need to sample at a rate => 400kHz

XILINX

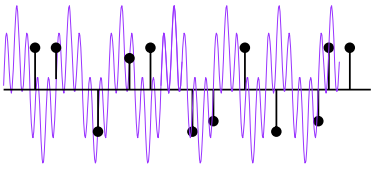
NU HORIZONS ELECTRONICS

Low & High Sample Rate





Low Sample Rate
→ OK for low frequency signals (kHz)
→ Not OK for high frequency signals

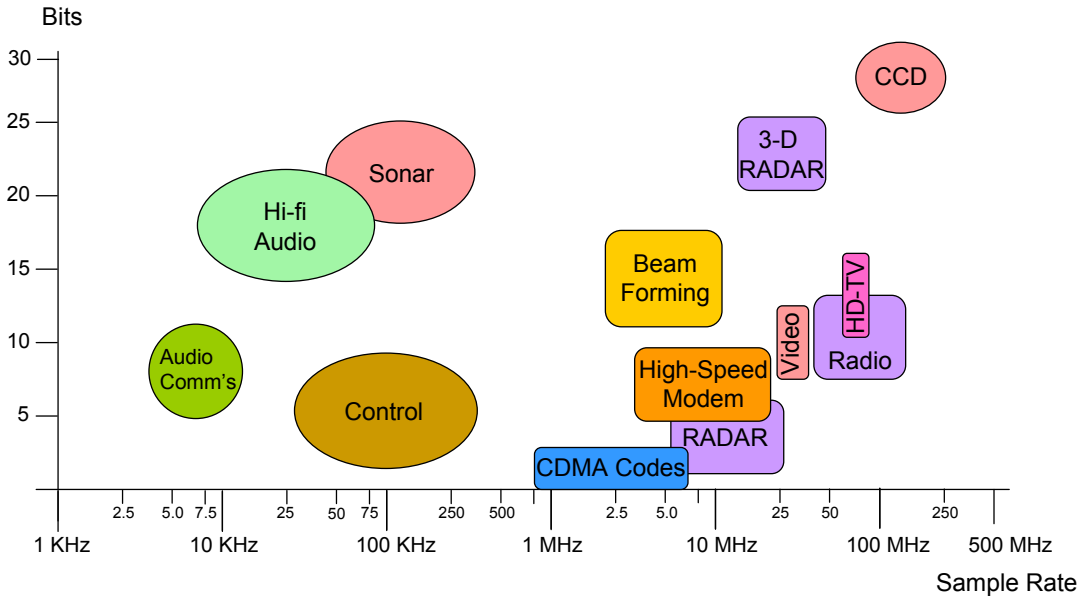


High Sample Rate
→ Needed for high frequency signals (lots of MHz)

XILINX

NU HORIZONS ELECTRONICS

Sample Rates and Bit Widths



XILINX

NU HORIZONS ELECTRONICS

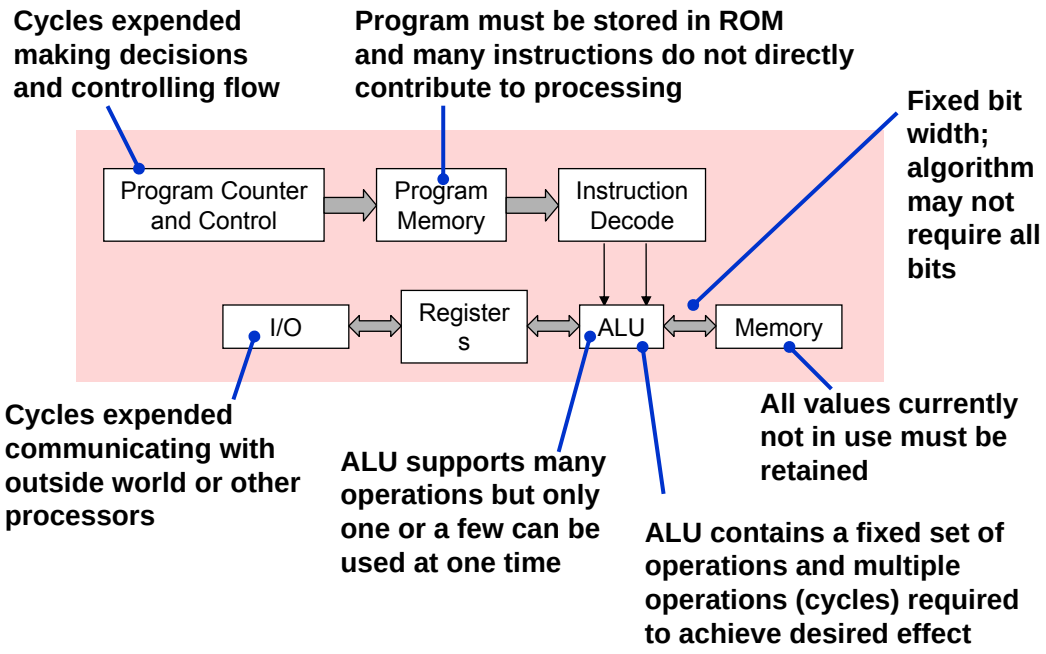
Agenda

- The System View
- *Why FPGA's for Signal Processing?*
- Typical Signal Processing Algorithms
- Basic Building Blocks of Signal Processing
- What is an FPGA?
- DSP Blocks in the FPGA
- Tools
- Summary

XILINX®

NU HORIZONS ELECTRONICS

Essence of a DSP Processor

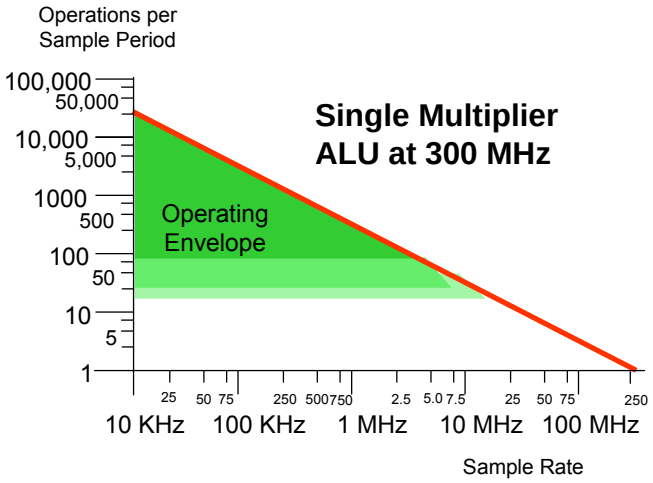


XILINX®

NU HORIZONS ELECTRONICS

Limits of a DSP Processor

Sample Rate	Operations per sample period	
	Single Multiplier ALU at 300 MHz	Two Multipliers ALU at 600 MHz
8 KHz	37,500	150,000
44.1 KHz	6,803	27,210
300 KHz	1,000	4,000
1.2288 MHz	244	976
3.84 MHz	78	312
27 MHz	11	44
74 MHz	4	16
102.4 MHz	3	12
300 MHz	1	4

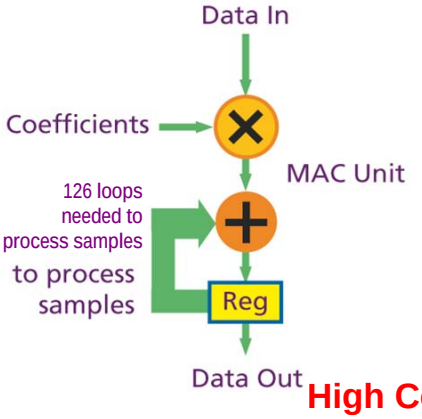


XILINX®

NU HORIZONS ELECTRONICS

Why FPGAs for DSP?

Conventional DSP Processor - Serial

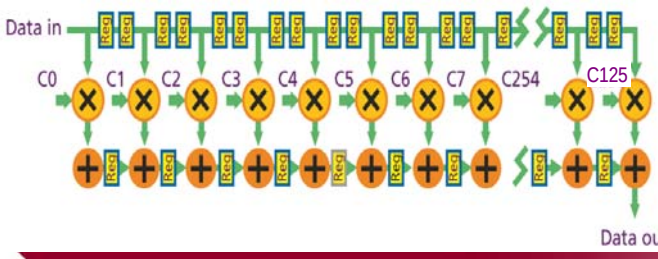


1 GHz / 126 clock cycles = 8 MSPS / MAC unit

MSPS = Megasamples per second

High Computational Workloads

FPGA-based DSP - Parallelism



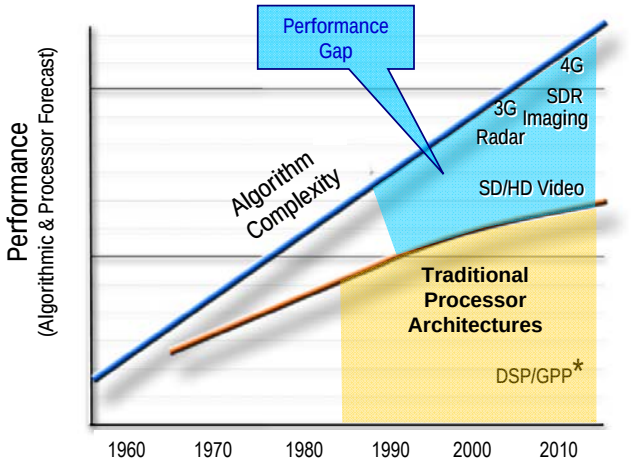
Virtex FPGA: 550 MHz / 1 clock cycle = 550 MSPS x5 filters

Spartan FPGA: 250 MHz / 1 clock cycle = 250 MSPS

XILINX®

NU HORIZONS ELECTRONICS

FPGA's Address DSP Performance Gap



* Source: Jan Rabaey, BWRC (Berkeley Wireless Research Center)

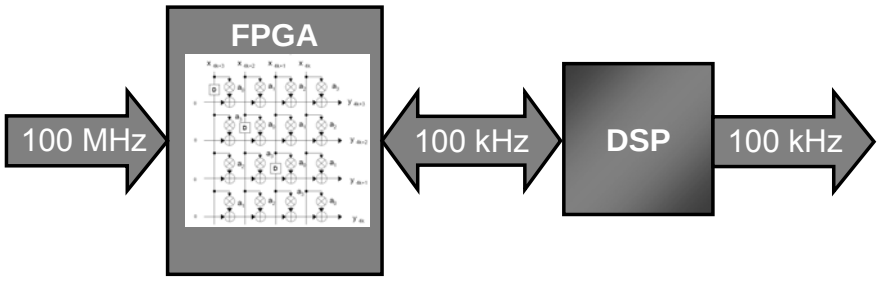
Parallelism	→	Performance increase
Flexible architecture	→	Low risk
Easy changes/design migration	→	Handle changing/adding standards
Programmability	→	Customized & differentiated solutions
Technology shrinks	→	Price decreases
Lower power consumption	→	Portable applications

XILINX®

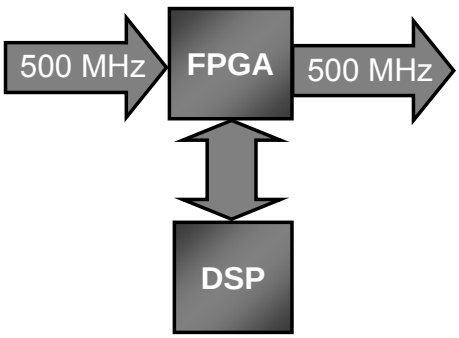
NU HORIZONS ELECTRONICS

How To Address the DSP Performance Gap?

Pre-processor



Co-processor



XILINX®

Agenda

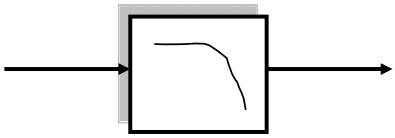
- The System View
- Why FPGA's for Signal Processing?
- *Typical Signal Processing Algorithms and their Presentation*
- Basic Building Blocks of Signal Processing
- What is an FPGA?
- DSP Blocks in the FPGA
- Tools
- Summary

What are DSP Filters?

- Filters are DSP functions to remove unwanted signals such as electrical noise that is added during transmission
- Filters are one of the most common DSP functions
- In FPGAs, filters are built up using
  - → Adders
  - → Multipliers
  - → Flip Flops (delays)
- DSP Designers talk about "Filter Taps" or "Filter Order"
  - → Generally speaking the more the # of Taps, or the higher the filter order, the better the filter at removing unwanted signals.
  - → The catch is higher order and more taps usually means more resources hence cost
- Implementing filters in FPGAs is relatively easy to do these days using tools like Xilinx FIR compiler

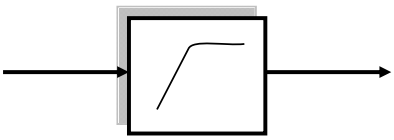
Common DSP Filter Types

Low Pass



Filter lets LOW frequencies PASS through. Rejects all others.

High Pass

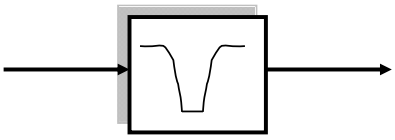


Filter lets HIGH frequencies PASS through. Rejects all others.

Band Pass

Filter only lets a specified BAND of frequencies to PASS through. Rejects all others.

Band Stop



Filter STOPS a specific BAND of frequencies from passing through. Lets all others pass.

FIR Filter

Viewed as an Equation

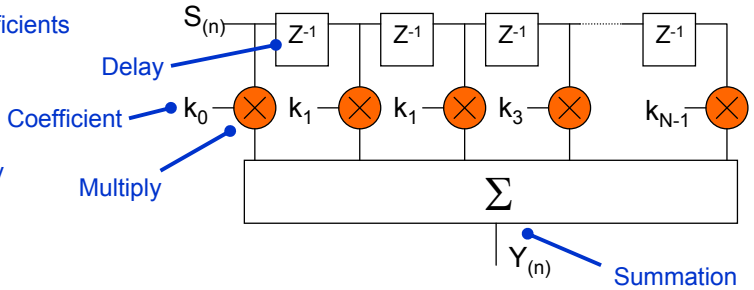
$$Y_{(n)} = \sum_{i=0}^{i=N-1} k_i . S_{(n-i)}$$

Coefficients

Multiply

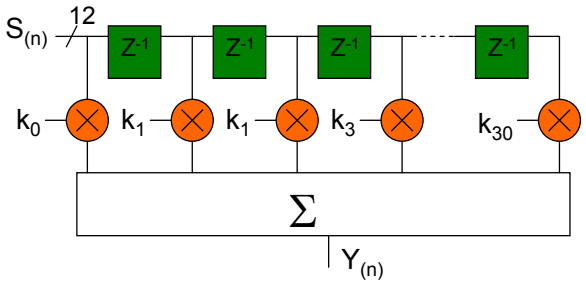
Accumulate N times

Viewed as a Diagram



Inside the FIR Filter

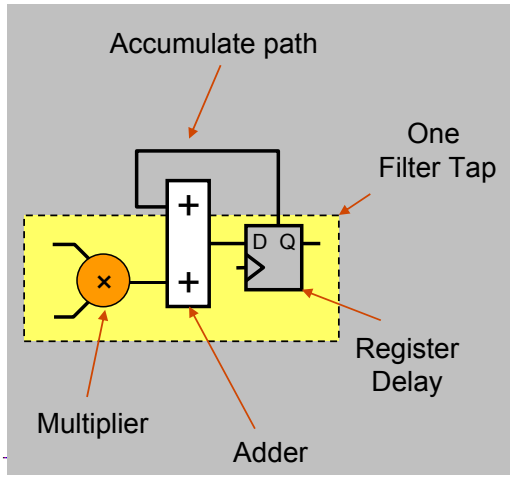
- Basic specifications
  - Number of taps
  - Sample rate (Hz)
  - Sample size (Bits)

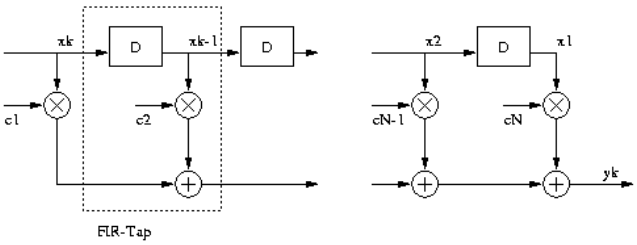


- Although some decisions could already be taken with the basic specifications, understanding some more about the inside of the filter can be helpful—which means that coefficient values should be studied

Filter Architectures: MAC FIR Filter

- MAC FIR Filter
  - Single multiplier, adder & register
  - Time shared
  - Common in DSP processors (e.g. TI)
- Pros
  - Few resources
  - Easy to implement
- Cons
  - Suitable for low sample rates (typically 1-few hundred kHz)
- Example Applications
  - Audio (20kHz) filtering
  - Low rate motor control

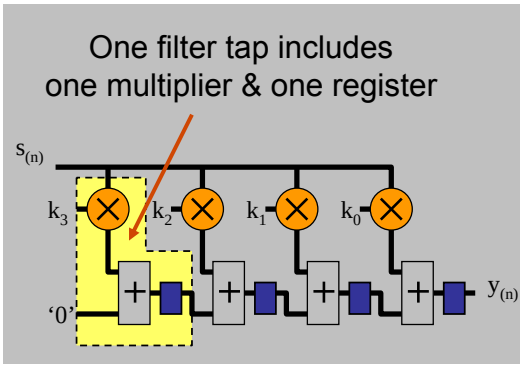




To implement a 4 Tap Filter you need to accumulate 4 times around the MAC FIR Filter

Filter Architecture: Parallel FIR Filter

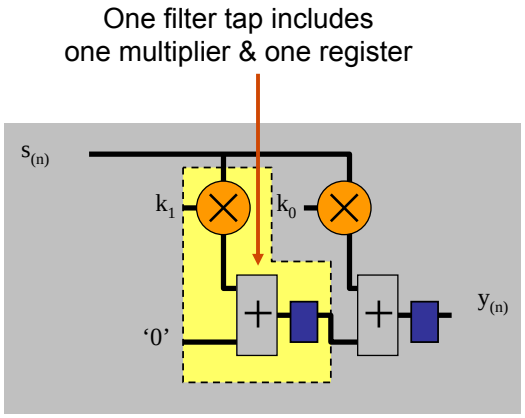
- Parallel FIR Filter
  - Multiple multipliers, adders & registers
  - Filter Taps are NOT time shared
  - Cannot do this in DSP processors for > 4 taps
- Pros
  - Parallel = High Performance
  - Suitable for high sample rates (typically 10 - 500MHz)
- Cons
  - More difficult to design as architecture needs to be constructed first
- Example Applications
  - HD Video filtering
  - RF stage filtering in wireless applications



To implement a 4 Tap Filter you use each tap only once and all at the same time

Filter Architecture: Semi-Parallel FIR Filter

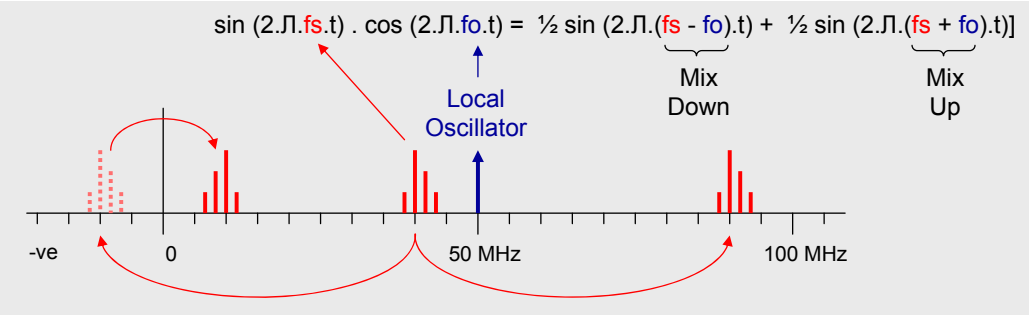
- Semi-Parallel FIR Filter
  - Multiple multipliers, adders & registers
  - Filter Taps are time shared
  - Can do this in some high-end DSP processors that have 4 MAC units
- Pros
  - Parallel = High Performance
  - Suitable for moderately high sample rates (typically 1-200MHz)
- Cons
  - Architecture needs to be constructed first
- Example Applications
  - SD Video Filtering
  - IF Stage in Wireless



To implement a 4 tap filter you use each tap 2x

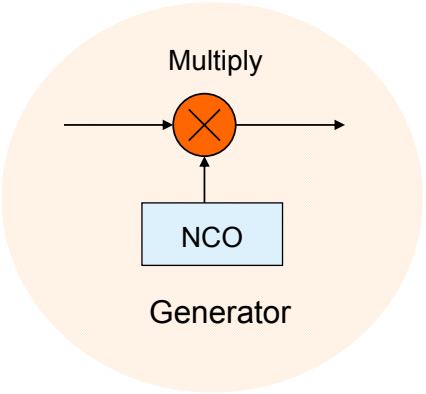
Digital Mixing

- Mixing is common in frequency up and down conversion
- An input signal ‘fs’ is mixed with a local oscillator frequency of ‘fo’
  - → The resulting signal contains components at frequencies that are the sum and difference of the original frequencies
  - → Filters are then used to reject the unwanted of the two spectrums



Digital Mixing

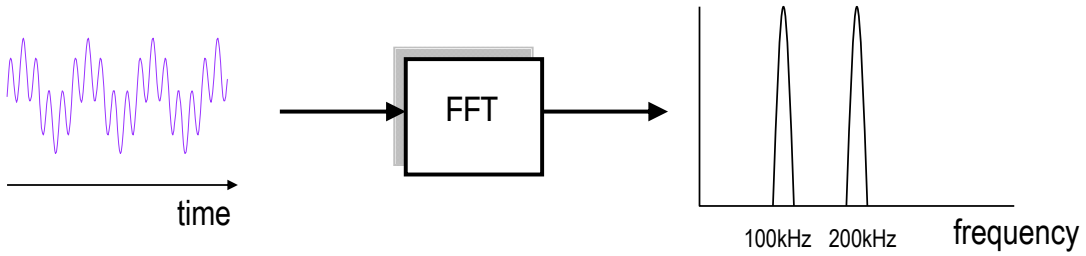
- At the implementation level, you see the requirement of a single multiplier inserted in the datapath
- Only one operation is required per sample
- The oscillator normally takes the form of a Numeric Controlled Oscillator (NCO) such that the mixing frequency can be easily selected or even changed during operation
- The NCO is required to generate samples representing a sinusoidal waveform at the same rate as the input data samples



NU HORIZONS ELECTRONICS

FFT

- The Fast Fourier Transform (FFT) is the most important operation in digital signal processing.
- The FFT converts from Time to Frequency domain and is used for analyzing the frequency content of signals, the frequency response of signals
- The FFT helps you to understand what signals are present and hence determine which ones you want to keep and which ones to filter out



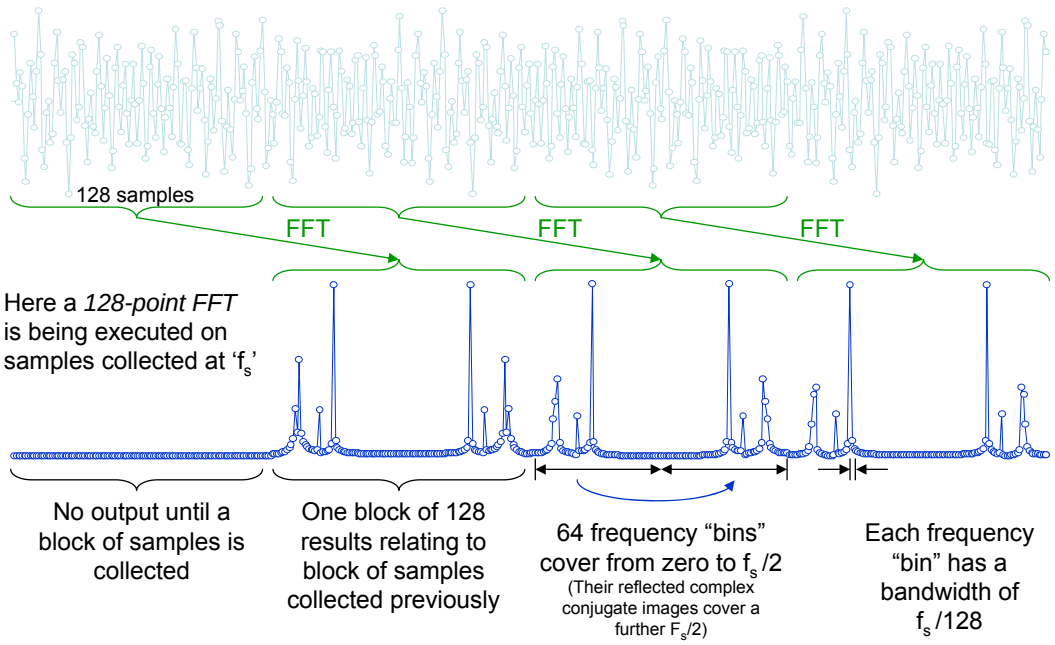
Time Varying Signal that consists of multiple sine waves at different frequencies

Information shows 2 sine waves are present in the signal

XILINX

NU HORIZONS ELECTRONICS

Fast Fourier Transform (FFT)



XILINX

NU HORIZONS ELECTRONICS

Agenda

- The System View
- Why FPGA's for Signal Processing?
- Typical Signal Processing Algorithms
- ***Basic Building Blocks of Signal Processing***
- What is an FPGA?
- DSP Blocks in the FPGA
- Tools
- Summary

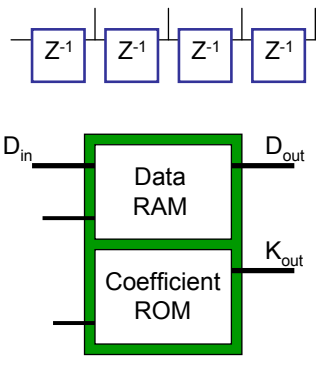
XILINX®

NU HORIZONS ELECTRONICS

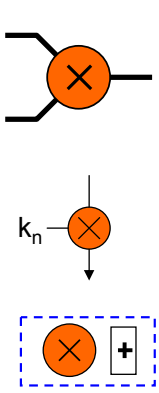
Basic Building Blocks of DSP

From the FIR and FFT, you have seen that DSP is supported by some very basic mathematical functions

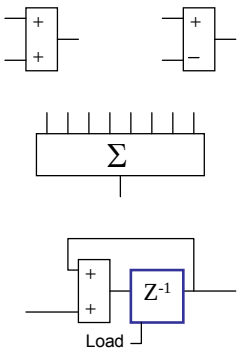
Delays and Data Storage



Multiplication



Addition, Subtraction, Summation, and Accumulation

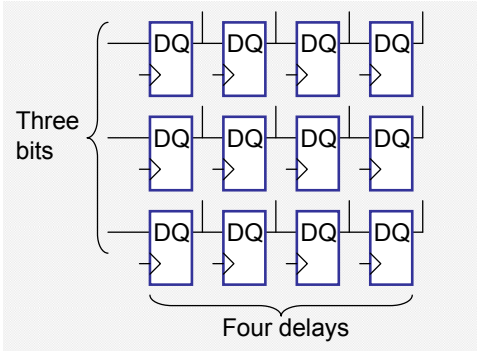


DSP48 can be used in multiplication, accumulation, summation, and MAC modes

XILINX®

Delays and Data Storage

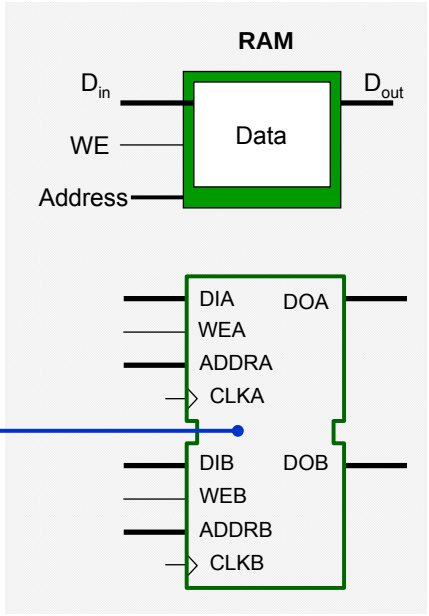
A flip-flop is a primitive storage element capable of retaining one bit of information when clocked



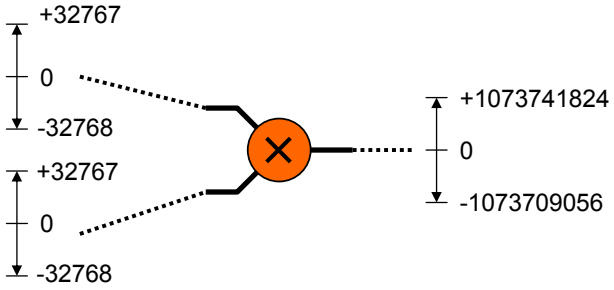
- Four delays × three bits = 12 flip-flops
- Full parallel access to data
- Maximum possible bandwidth
- Simple (or possibly no) control logic

Delays and Data Storage

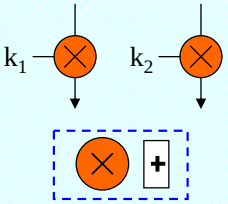
- Memory (RAM) is organized into locations, with each location containing a number of bits
  - → For example, a RAM of 512 × 8 would have 512 addressed locations, each containing 8 bits of information
- Sequential access to data
- Bandwidth limited to one location at a time
- Dual port, if available, provides two locations at a time
- Can only write to one location at a time
- Binary coded address needs to be generated



Multipliers



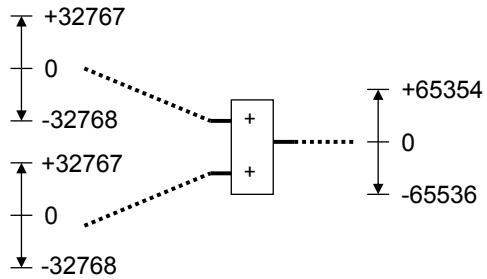
Multiplication is the foundation of DSP

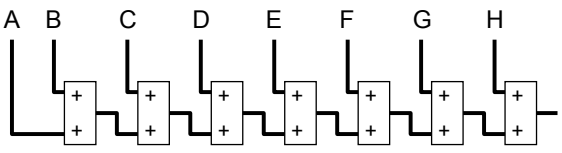


- A multiplier is used with coefficients in many algorithms
- In full parallel structures, each multiplier is only working with one coefficient, which leads to the Constant(k) Coefficient Multiplier (KCM)
- The DSP48 slice provides a high-performance multiplier

XILINX®

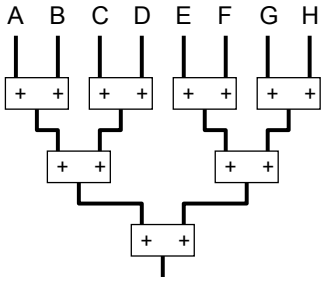
Addition and Summation





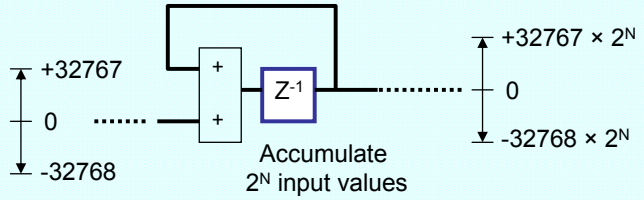
Addition Chain

Adder Tree



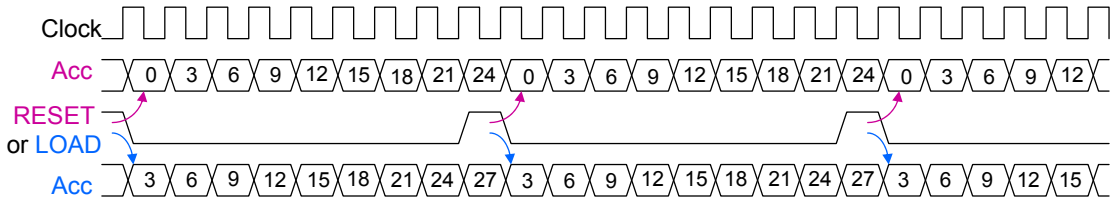
XILINX®

Accumulation



Accumulation is a combination of addition and data storage

Because only one value is retained, you associate this storage with a flip-flop-based register and use a clock to indicate the point of accumulation



Constructing a 128-Tap FIR Filter

A 128-tap constant coefficients FIR filter is to be constructed in a Virtex™-II Pro device, assuming design clocking at 200 MHz

Requirements	MAC Engine Style	Full Parallel Style
Number of processing clock cycles per sample	**128**	**1**
Memory size (words) and requirements	**256 dual port**	**-**
Number of flip-flop based registers	**1**	**128**
Number of multipliers	**1**	**128**
Number of accumulators	**1**	**-**
Number of adders	**-**	**127**
Maximum sample rate achievable (Hz)	**1.5625 MHz**	**200 MHz**

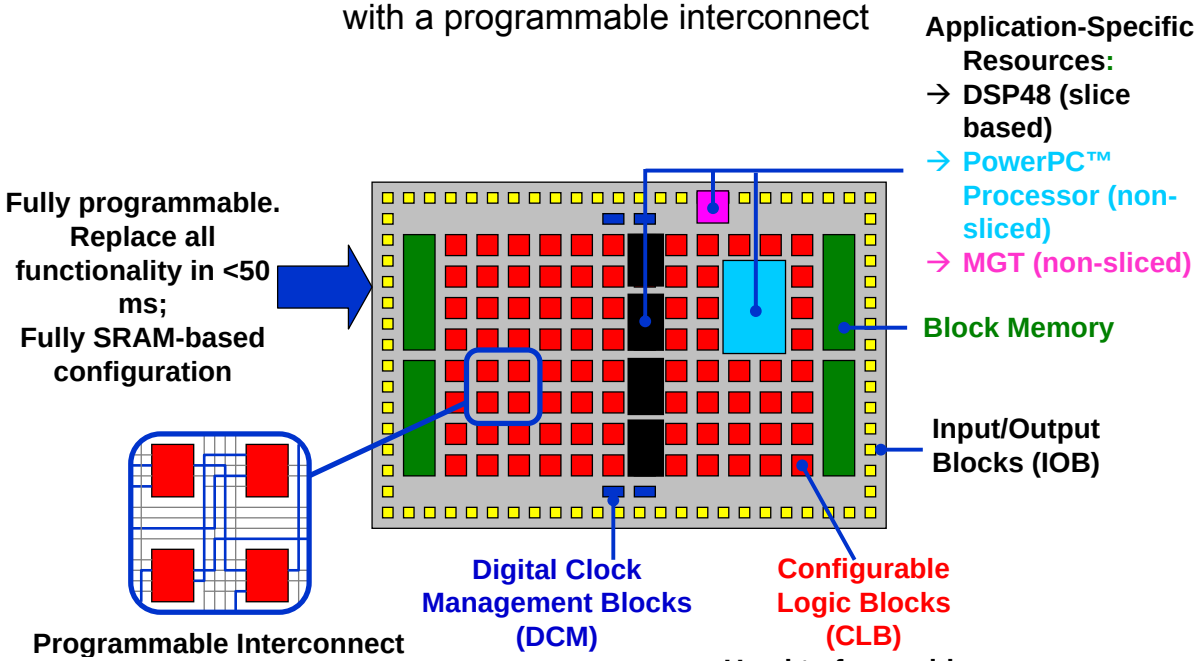
Agenda

- The System View
- Why FPGA's for Signal Processing?
- Typical Signal Processing Algorithms
- Basic Building Blocks of Signal Processing
- *What is an FPGA?*
- DSP Blocks in the FPGA
- Tools
- Summary

XILINX®

Xilinx FPGA Device Architecture

Uniform structure of programmable blocks, which can be connected together with a programmable interconnect

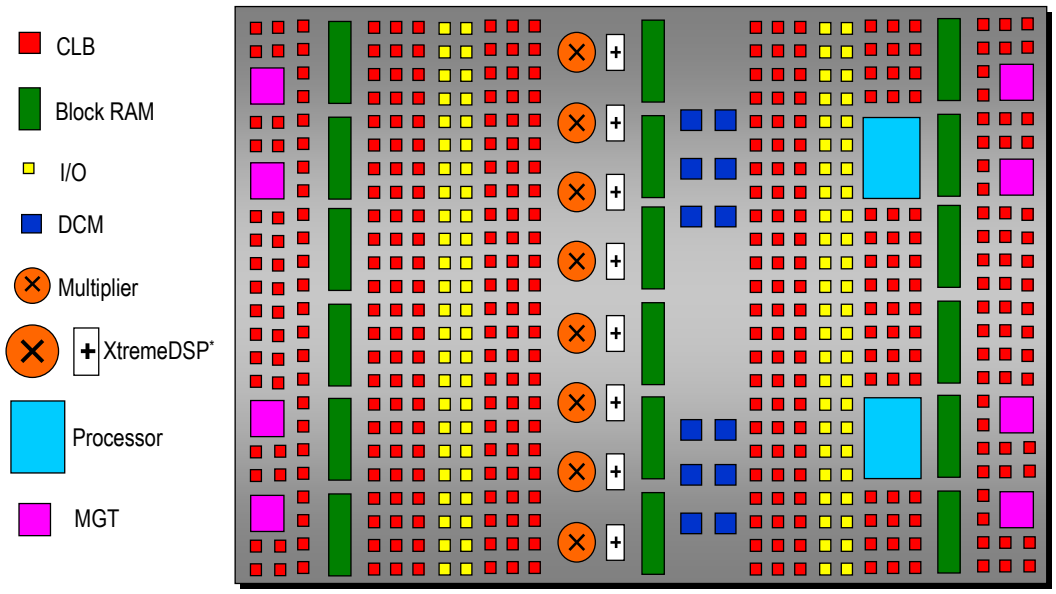


And have evolved through several generations
- Adding various capabilities
- Incorporating system integration-oriented resources
- Improving silicon efficiency

XILINX®

NU HORIZONS ELECTRONICS

Range of Resources



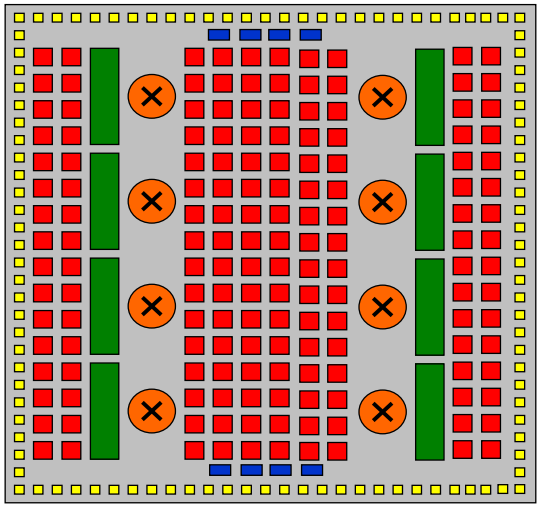
* XtremeDSP™ technology ≡ DSP48

XILINX®

NU HORIZONS ELECTRONICS

Spartan-3E FPGA

- Industry's first sub-$2 FPGA
- Optimized for gate-centric designs (lowest cost per logic)
- Multiplier/block RAM and DCMs separated and "embedded" in array
- Improved DDR interface for differential I/O interface
- Input and output pipeline stages in multipliers for improved performance
- Register cascading in multiplier useful for some DSP structures



Smallest device – XC3S100E	240 CLB	4 BRAM	4 Multipliers
Largest device – XC3S1600E	3688 CLB	36 BRAM	36 Multipliers

XILINX®

NU HORIZONS ELECTRONICS

Spartan-3E Platform

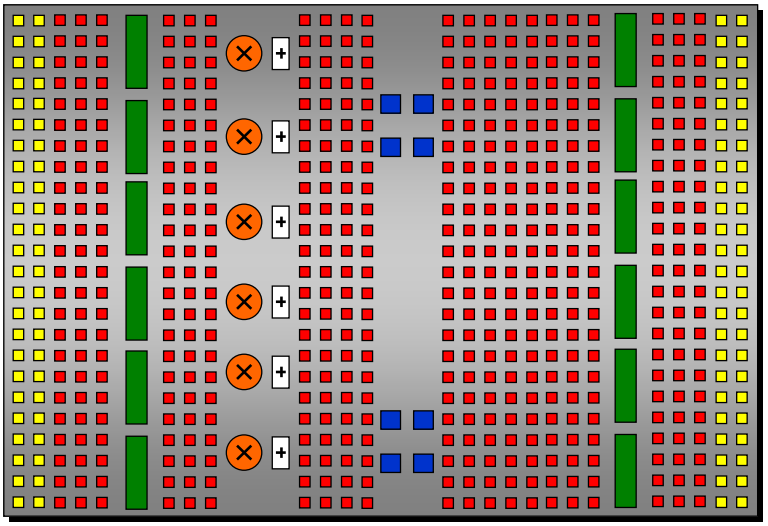
	Spartan-3 Platform Optimized for High Density and High I/O Designs								Spartan-3E Platform Logic Optimized				
Part Number	XC3S50	XC3S200	XC3S400	XC3S1000	XC3S1500	XC3S2000	XC3S4000	XC3S5000	XC3S100E	XC3S250E	XC3S500E	XC3S1200E	XC3S1600E
System Gates [1]	50K	200K	400K	1000K	1500K	2000K	4000K	5000K	100K	250K	500K	1200K	1600K
Slices [2]	768	1,920	3,584	7,680	13,312	20,480	27,648	33,280	960	2,448	4,656	8,672	14,752
Logic Cells	1,728	4,320	8,064	17,280	29,952	46,080	62,208	74,880	2,160	5,508	10,476	19,512	33,192
CLB Flip-Flops	1,536	3,840	7,168	15,360	26,624	40,960	55,296	66,560	1,920	4,896	9,312	17,344	29,504
Maximum Distributed RAM (Kbits)	12	30	56	120	208	320	432	520	15	38	73	136	231
Block RAM Blocks	4	12	16	24	32	40	96	104	4	12	20	28	36
Total Block RAM (Kbits)	72	216	288	432	576	720	1,728	1,872	72	216	360	504	648
Digital Clock Managers (DCMs)	2	4	4	4	4	4	4	4	2	4	4	8	8
Maximum Single Ended I/Os	124	173	264	391	487	565	712	784	108	172	232	304	376
Maximum Differential I/O Pairs	56	76	116	175	221	270	312	344	40	68	92	124	156
I/O Standards Supported	LVTTL, LVCMOS33, LVCMOS25, LVCMOS18, LVCMOS15, LVCMOS12, GTL, GTL+, HSTL15 Class I, HSTL15 Class III, HSTL18 Class I, HSTL18 Class II, HSTL18 Class III, PCI 3.3V 32/64bit 33MHz, SSTL2 Class I, SSTL2 Class II, SSTL18 Class I, Bus LVDS, LDT (ULVDS), LVDS_ext, LVDS25 & 33, LVPECL25, RSDS25								LVTTL, LVCMOS33, LVCMOS25, LVCMOS18, LVCMOS15, LVCMOS12, HSTL18 Class I, HSTL18 Class III, PCI 3.3V 32/64bit 33MHz, PCI 3.3V 64bit/66MHz, PCI-X 3.3V, SSTL2 Class I, SSTL18 Class I, Bus LVDS, LVDS25, LVPECL25, Mini-LVDS25, RSDS25				
Dedicated Multipliers	4	12	16	24	32	40	96	104	4	12	20	28	36
Commercial	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5	-4, -5
Industrial	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4
Configuration Memory Bits (Mbits)	0.4	1.0	1.7	3.2	5.2	7.7	11.3	13.3	0.6	1.4	2.3	3.8	6.0

XILINX®

NU HORIZONS ELECTRONICS

Virtex-4 LX FPGA

- Innovative Application Specific Modular Block architecture
- The LX family has a heavy skew toward logic fabric
- The DSP48 slice replaces embedded multipliers
- Enables DSP functions to be implemented efficiently with slices



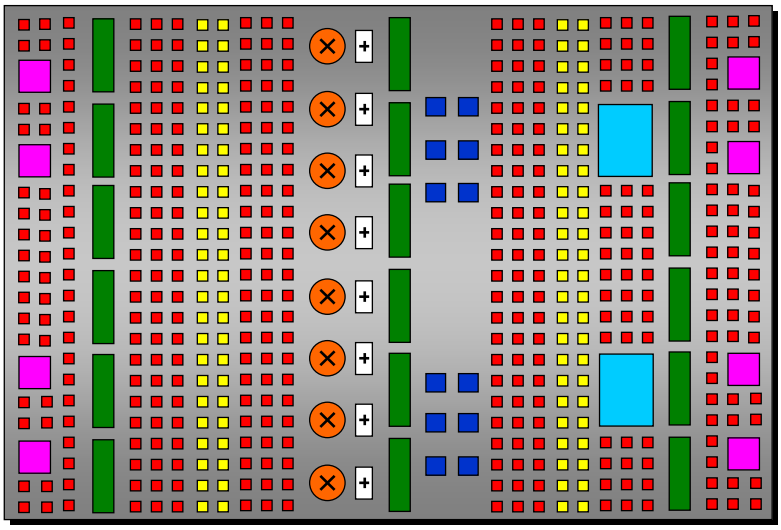
Smallest device – 4VLX15 1,536 CLB 48 BRAM 32 DSP48 Slices

Largest device – 4VLX200 22,272 CLB 336 BRAM 96 DSP48 Slices

XILINX®

Virtex-4 FX FPGA

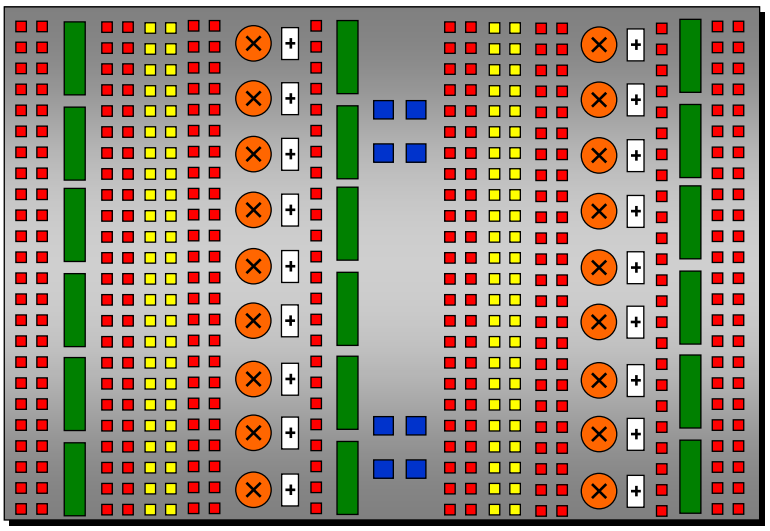
- The FX family emphasizes system integration, much like the Virtex™-II Pro FPGA
- Similarly, there are 1-2 PowerPC™ 405 processors and MGTs available to support a high-speed serial interface
- An Ethernet MAC is also available for networking communications



Smallest device - 4VFX12	1,368 CLB		36 BRAM			32 DSP48 Slices
Largest device - 4VFX140	15,792 CLB		552 BRAM			192 DSP48 Slices

Virtex-4 SX FPGA

- The SX family emphasizes DSP applications by providing a strong skew toward the dedicated arithmetic units versus logic
- The DSP48 slice resource redefines techniques to implement functions
- Unlike LX, this family only has three devices



Smallest device – 4VSX25	2,650 CLB		128 BRAM			128 DSP48 Slices
Largest device – 4VSX55	6,144 CLB		320 BRAM			512 DSP48 Slices

Virtex-5 FPGA

- Higher DSP Performance
  - DSP Performance = GMACs + Bandwidth (Memory + IO)
- XtremeDSP™ Slices
  - Enable greater parallelism for faster performance
  - Up to 1,056 built-in 18x25 DSP slices delivering
    - 560 GMACs at 550MHz
    - 215 GFLOPs floating-point
- Increased Memory Bandwidth
  - 18.6Mb of memory, delivering >100 terabits/s memory bandwidth
- Higher I/O Bandwidth
  - DSP-optimized FPGA with built-in serial transceivers & PCIe® technology
  - Support for CPRI, OBSAI, SRIO, PCIe technology, etc.
  - 163 GBytes/s aggregate IO performance

XtremeDSP Slice – DSP48E

XtremeDSP

XILINX®

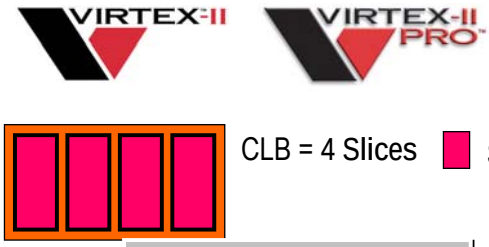
Agenda

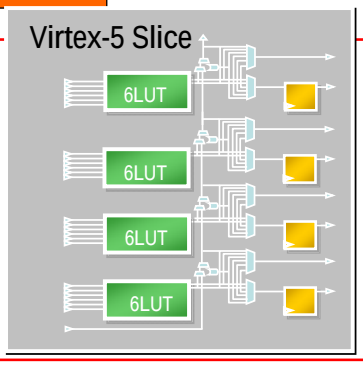
- The System View
- Why FPGA's for Signal Processing?
- Typical Signal Processing Algorithms
- Basic Building Blocks of Signal Processing
- What is an FPGA?
- *DSP Blocks in the FPGA*
- Tools
- Summary

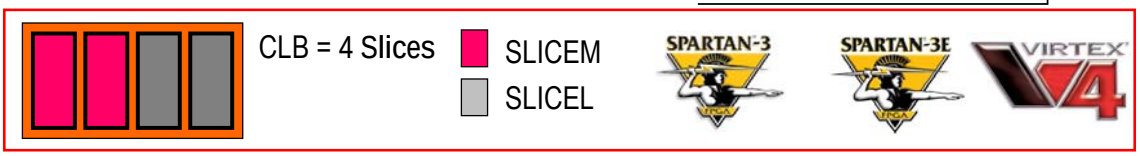
XILINX®

Configurable Logic Block (CLB)

- Four slices per CLB
- Wide-input functions
  - → 16:1 multiplexer in one CLB
  - → 32:1 multiplexer in two CLBs
- Fast arithmetic functions
  - → Two carry chains per CLB
- Four 16-bit addressable shift registers
  - → Cascadable



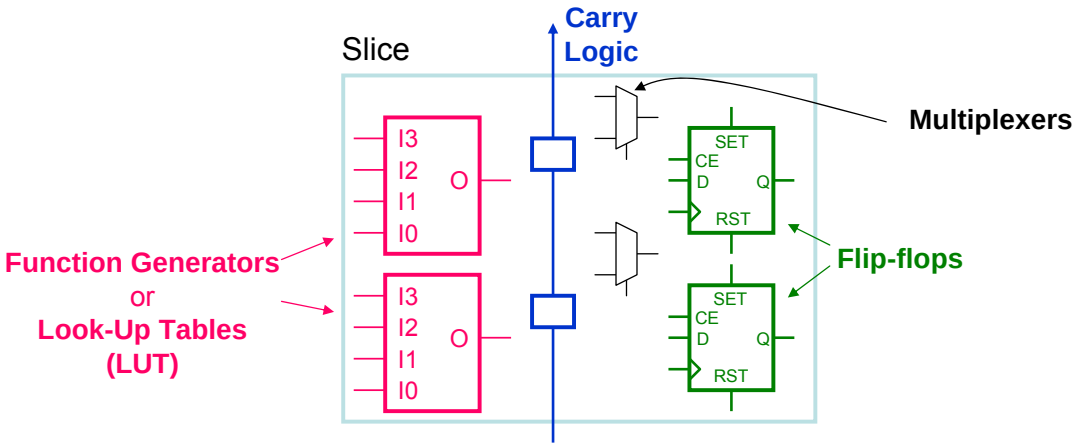




Slice

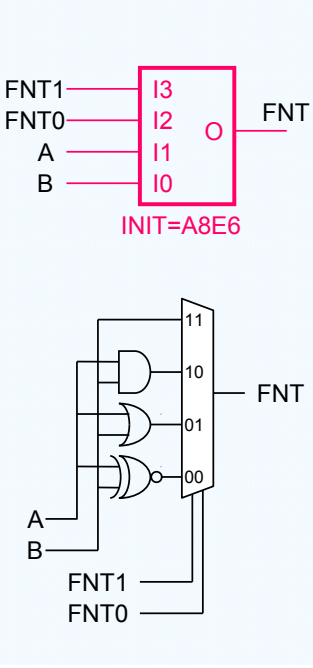
Output of the LUT (Look-up Table) can be routed either straight or through a flip-flop

- → Flip-flops provide synchronous functionality, improving clocking rate without costing additional resources



Look-Up Tables

- The Look-Up Tables (LUTs), or function generators, are the elements of the FPGA that enable logic gates to be implemented
  - This example shows how a complex logical function equivalent to 14 two-input gates and 5 inverters is formed into the initialization attribute of an LUT



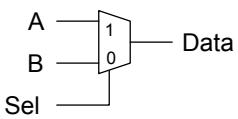
My Logic Selector

I3	I2	I1	I0	O	INIT=A8E6
FNT1	FNT0	A	B	FNT	
0	0	0	0	0	6
0	0	0	1	1	
0	0	1	0	1	
0	0	1	1	0	
0	1	0	0	0	E
0	1	0	1	1	
0	1	1	0	1	
0	1	1	1	1	
1	0	0	0	0	8
1	0	0	1	0	
1	0	1	0	1	
1	0	1	1	1	
1	1	0	0	0	A
1	1	0	1	1	
1	1	1	0	0	
1	1	1	1	1	

Multiplexer LUT

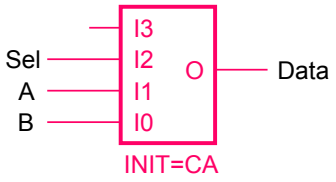
The multiplexer is a fundamental part of many DSP techniques

Sel	Data
0	B
1	A



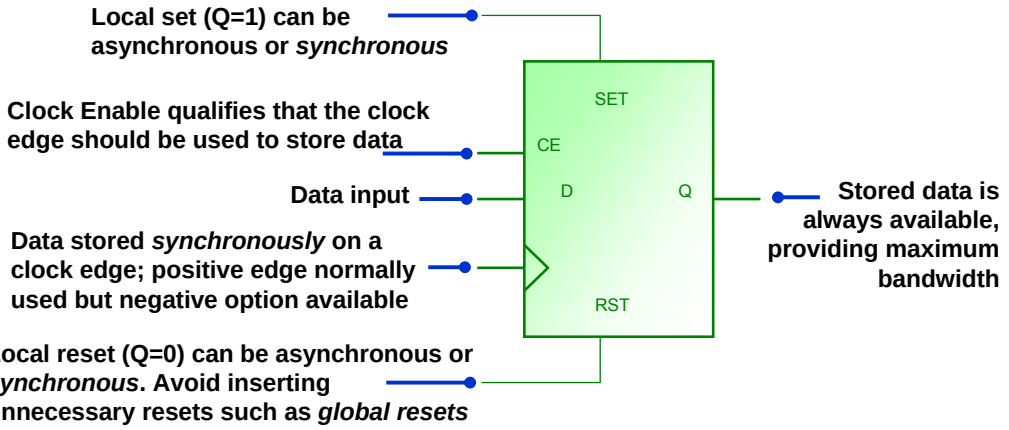
2:1 Multiplexer

I3	I2	I1	I0	O	INIT=CA
	Sel	A	B	Data	
x	0	0	0	0	A
x	0	0	1	1	
x	0	1	0	0	
x	0	1	1	1	
x	1	0	0	0	C
x	1	0	1	0	
x	1	1	0	1	
x	1	1	1	1	



Flip-Flops for Delay

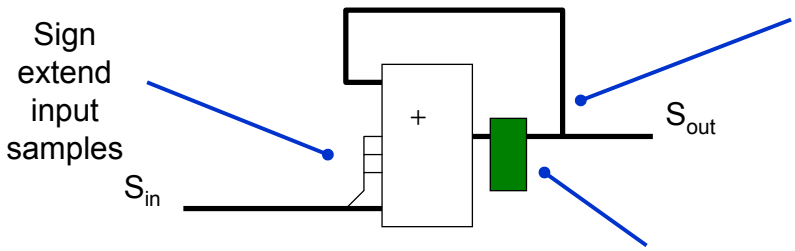
- Sample delay is the most simple of the DSP functions
  - → Flip-flops provide the sample delay functionality and locally have all the controls that you could require



Accumulation

- Accumulation is a fundamental operation
  - It can be used to perform integration
  - ½ slice per bit

Support virtually any bit width and avoid the dreaded overflow.
(Can also detect carry and disable clock enable on register)

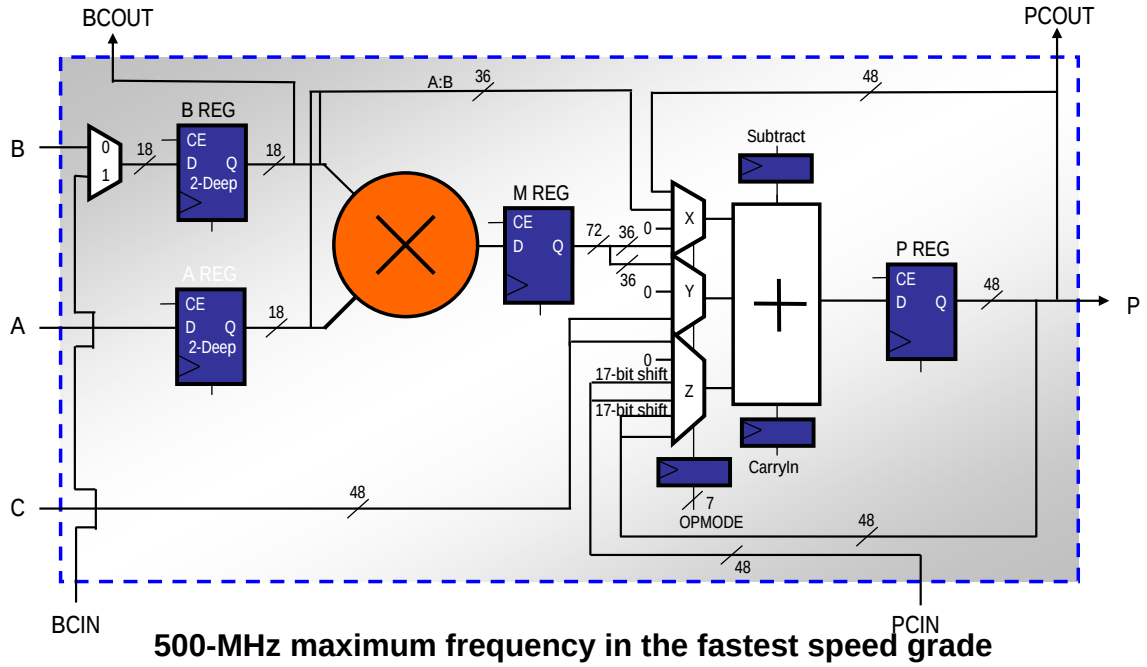


Register is local data storage
• No bandwidth issues
• Maximum performance and sample rates

Multiplication

- Multiplication is a time-consuming but fundamental operation in DSP applications
- Multiplication greatly affects the sample rate
- Various implementation methods exist and can be used depending on the sample rate, clock rate, and sample width
  - Parallel multiplication
    - Using embedded multipliers or DSP48 slices
    - Using LUTs
  - Semi-parallel multiplication
  - Shift and add
  - ROM-based
  - Constant coefficients multipliers

DSP48 Slice (Virtex-4)



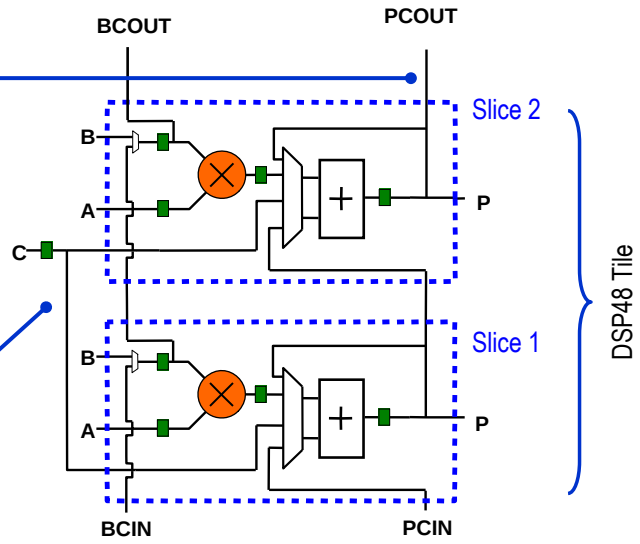
500-MHz maximum frequency in the fastest speed grade

Refer to the *DSP: Designing for Optimal Results – High-Performance DSP Using Virtex-4 FPGAs* guide for additional information on DSP48 slice functionality and usage

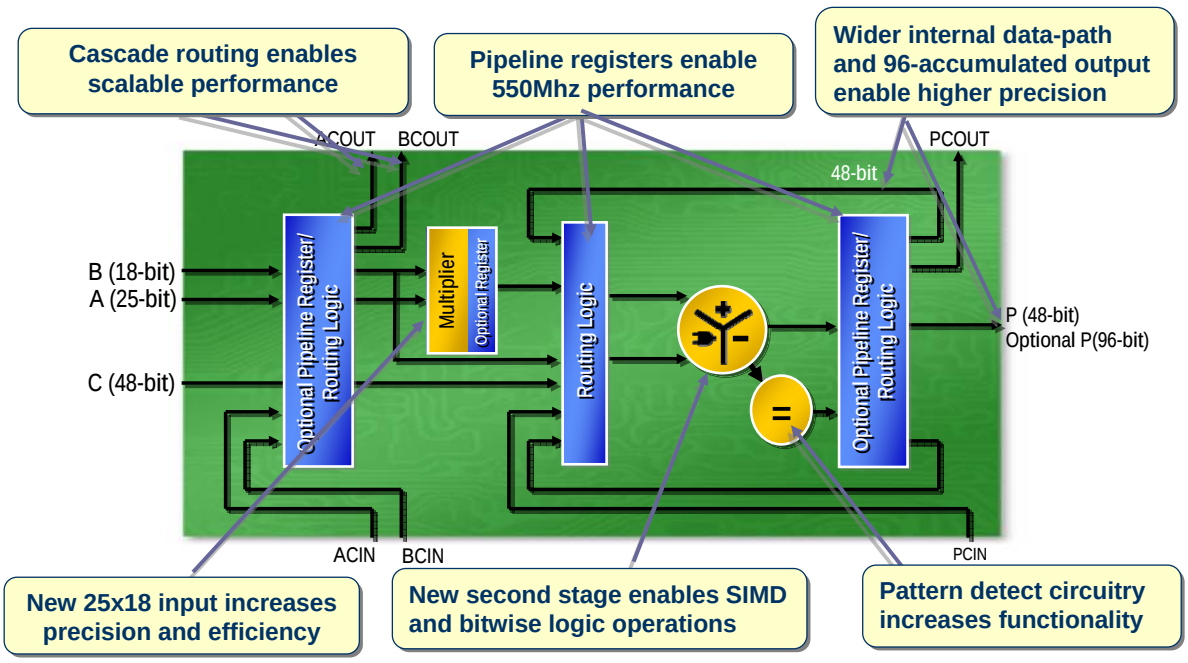
Cascading Slices → DSP48 Tile

DSP48 slices can be cascaded together with a direct dedicated interconnect from slice to slice. The column's length is determined by the number of DSP48 slices in the column

The C input is shared between two DSP48 slices or a single tile. Be careful to select algorithms that do not depend on a C input for each DSP48 slice



Virtex-5 DSP48E



NU HORIZONS ELECTRONICS

Agenda

- The System View
- Why FPGA's for Signal Processing?
- Typical Signal Processing Algorithms
- Basic Building Blocks of Signal Processing
- What is an FPGA?
- DSP Blocks in the FPGA
- *Tools*
- Summary

XILINX®

NU HORIZONS ELECTRONICS

Who Designs DSPs?

Architects

- Algorithm development
- HW/SW partitioning
- C/C++, Matlab, Simulink

System Engineers

Algorithm & Software Developers

- Algorithm development
- C/C++, Matlab development
- Compilers, SW debuggers
- Prog. DSP orientation

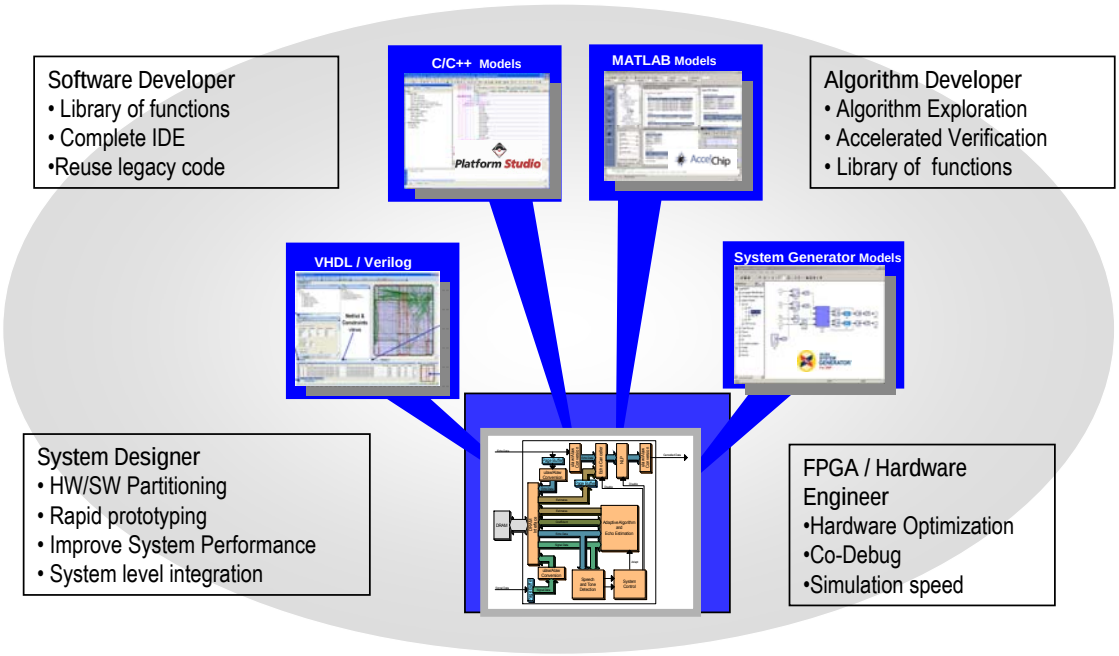
Hardware Engineers

- Hardware implementation
- RTL design, verification
- Synthesis, HDL simulation
- ASIC orientation

XILINX®

NU HORIZONS ELECTRONICS

The Programming Challenge

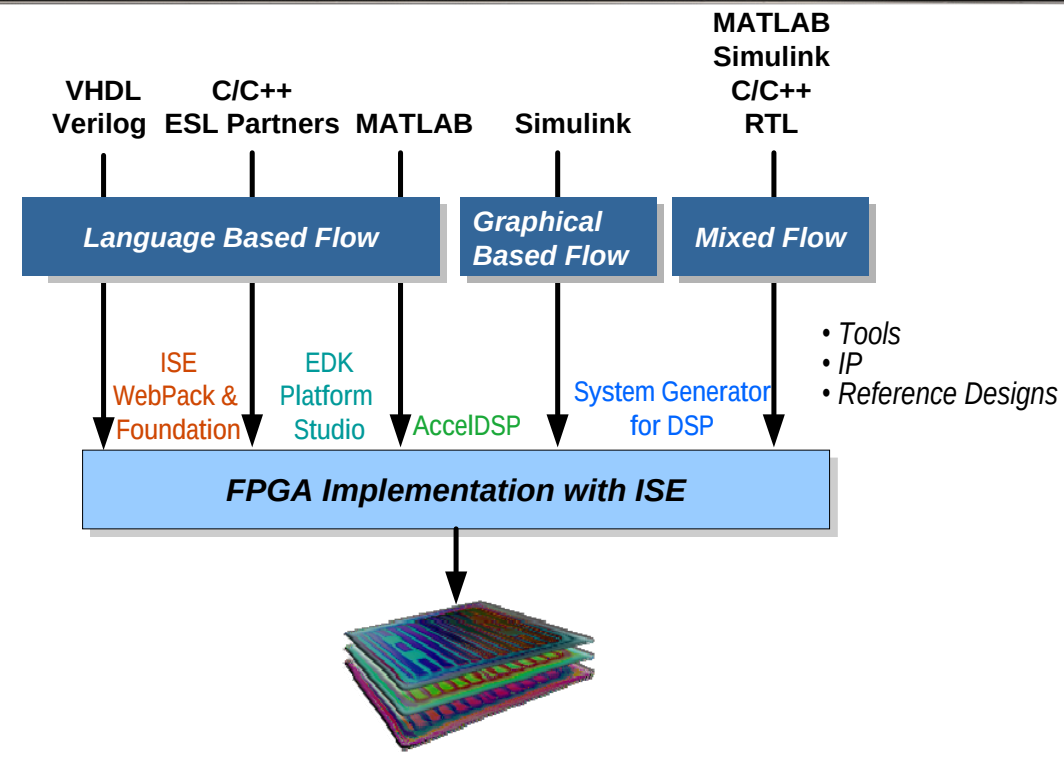


#1 Focus ~ Ease-of-use: Make FPGA Invisible

XILINX®

NU HORIZONS ELECTRONICS

Xilinx DSP Tools and Flows Accelerate DSP Design



XILINX®

Extending The Solution

Tools Vendors
- Mathworks
- Synplicity
- Mentor Graphics

3rd Party IP Vendors
- Qpixel (H.264)
- 4i2i (H.264)
- AXIS (DPD)

Core Partners
- Texas Instruments
- Mathworks

Board / Subsystem Vendors
- ISR Technologies *(JTRS Kit)*
- Spectrum Digital *(VCK)*
- AXIS *(DFE)*
- Nallatech *(XtremeDSP Kit)*

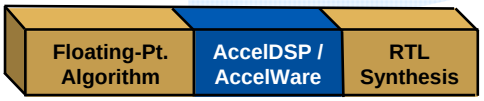
Academia
- Rice University
- Strathclyde University
- UC Berkeley
- San Diego State

XILINX

AccelDSP + AccelWare Design Flow

Typical MATLAB DSP Design Flow

Steps performed by AccelDSP



AccelDSP Design Flow

"We saw a 30% reduction in the design cycle time. This equated to an overall project development reduction of 15 percent, which provides two very significant benefits: we get our products to market faster and our teams are freed up to work on other projects sooner."

Dr. Paul Turner
Principal Systems Engineer
Powerwave Technologies

- ***Replaces manual steps***
  - Floating to fixed-point conversion
  - RTL creation
  - RTL and gate verification back to the original algorithm
  - IP creation and integration

XILINX

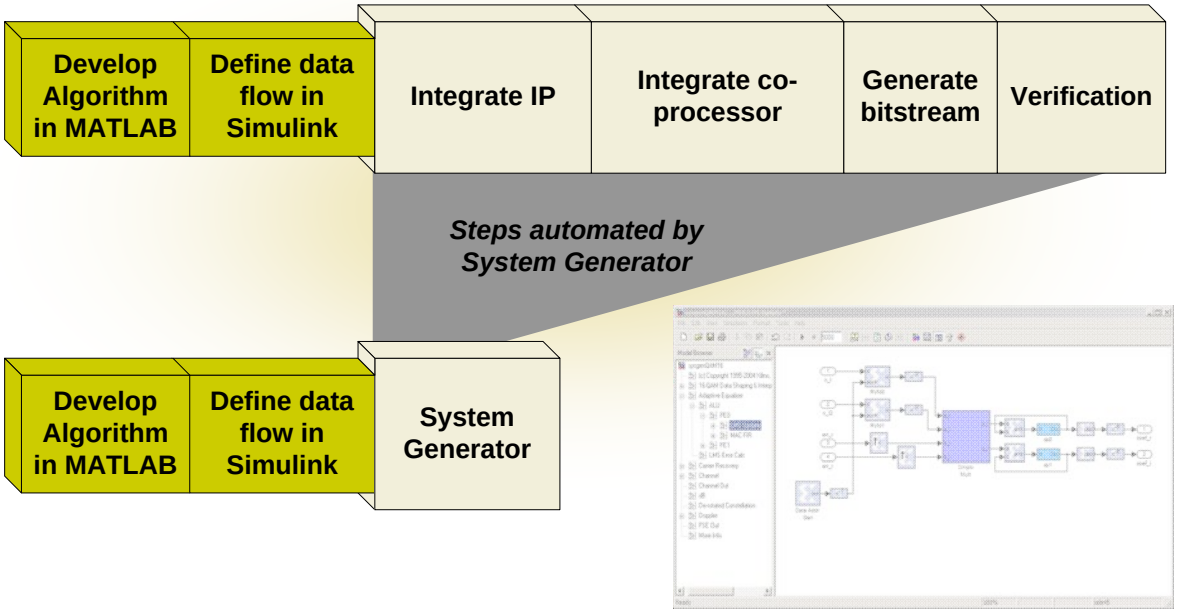
AccelWare DSP IP Core Generators

AccelWare Advanced Math Toolkit	AccelWare Signal Processing Toolkit	AccelWare Communications Toolkit
Polynomial Evaluator	CIC Decimator / Interpolator	A/D Sinc Compensation Filter
QRD-RLS Spatial Filter	Decimating FIR filter	Convolutional Interleaver / Deinterleaver
Givens Array Rotation	FFT, IFFT	Convolutional Encoder
Matrix inversion	FIR Filter	Direct Digital Synthesizer
QR method	Half-Band FIR Filter	Reed-Solomon Decoder
Cholesky	Polynomial Evaluation	Reed-Solomon Encoder
Triangular	Polyphase FIR Filter	BCH Encoder / Decoder
Matrix factorization		Root-raised Cosine Filter
QR method		Viterbi Decoder
Cholesky		Scrambler / Descrambler
SVD		
Triangular System of Equations Solver		

Each Toolkit core provides multiple silicon architectures that are fully parameterized to meet market-specific needs

System Generator Design Flow

Typical graphical based design flow



Rapid Integration of DSP IP

- Over 90 DSP building blocks pre-optimized for Xilinx devices
  - Developed by Xilinx DSP hardware design experts
- Automatic generation of the hardware interface
  - Clocks
  - Resets
  - handshake logic

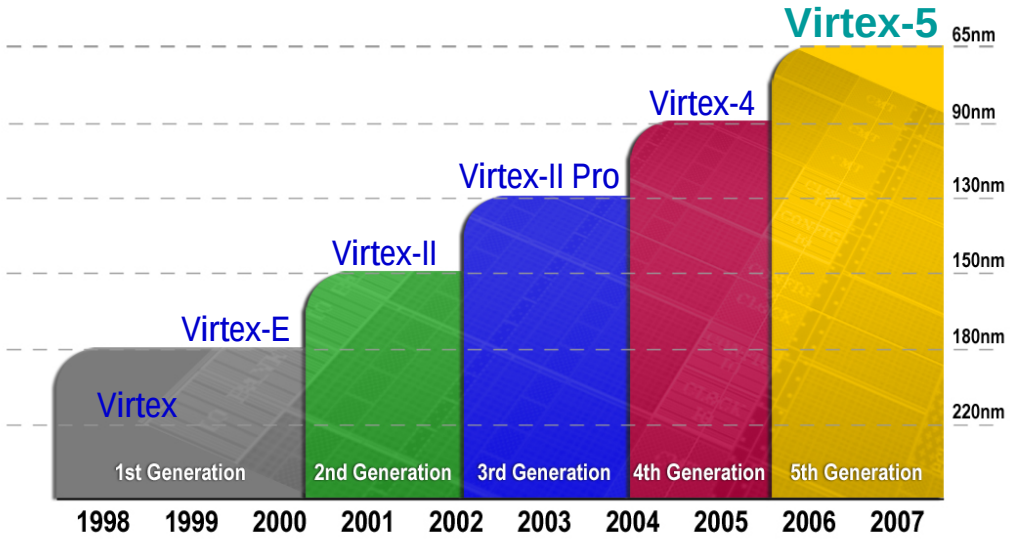
Category	Blocks
Math	mult, adder, accumulator, divider, trig
Filters	FIR, CIC, DAFIR
Memory	RAM, register, FIFO, shift register
Transforms	FFT, IFFT
Processors	TI DM642, MicroBlaze
Video	scalar, 2D FIR, color-space converter, chroma resampler, 2D rank order filter, VOIP
Communications	DDS, Viterbi Dec, Reed-Solomon Enc/Dec, DDC / DUC, digital front end, digital pre-distortion, interleaver / deinterleaver

Agenda

- The System View
- Why FPGA's for Signal Processing?
- Typical Signal Processing Algorithms
- Basic Building Blocks of Signal Processing
- What is an FPGA?
- DSP Blocks in the FPGA
- Tools
- *Summary*

NU HORIZONS ELECTRONICS

Virtex Product & Process Evolution



XILINX®

NU HORIZONS ELECTRONICS

Highest DSP Performance per Family

Largest Family Member	Highest # of MACs	Max Clock Rate	GMACS
Virtex-4 4VSX55	512*	500 MHz	256
Virtex-4 FX140	192*	500 MHz	96
Virtex-4 4VLX200	96*	500 MHz	48
Virtex-II Pro 2VP100	444**	300 MHz	133
Spartan-3 3S5000	104**	185 MHz	19
DSP Processor	4	1 GHz	4

* Using DSP 48 slices only (18x18 multiply, 48-bit add)
** Using MACs built from hard embedded multipliers only

XILINX®

NU HORIZONS ELECTRONICS

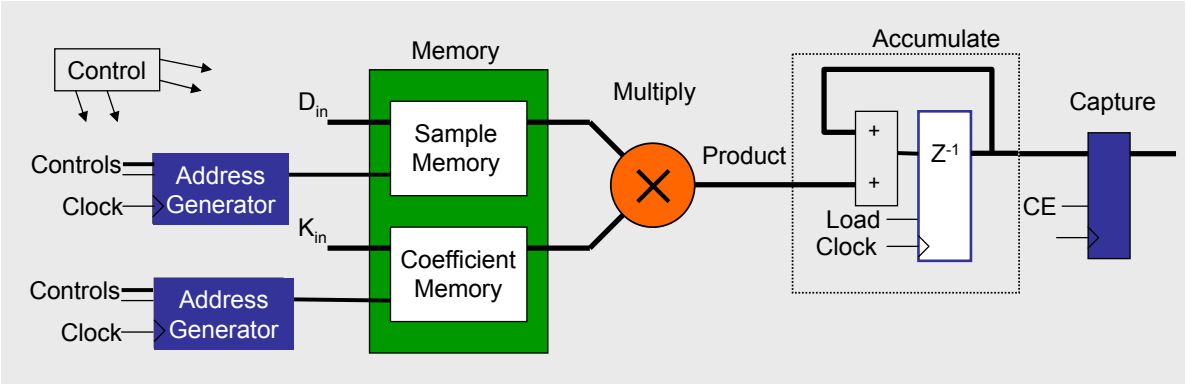
NU HORIZONS ELECTRONICS

XILINX®

NU HORIZONS ELECTRONICS

MAC Engine

- A simplified DSP processor
- Support of the MAC is where most of the considerations must be focused



MAC Rate = Sample Rate * Number of Taps

XILINX®

MAC Rates and Memory Requirements

$$Y_{(n)} = \sum_{i=0}^{i=N-1} k_i . S_{(n-i)}$$

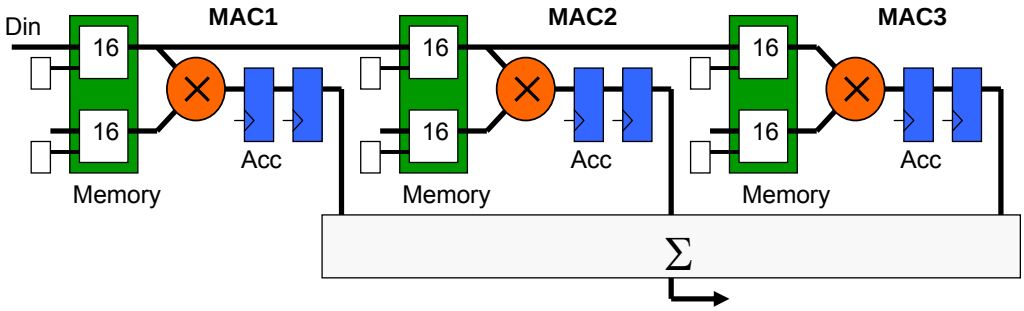
For the filter specifications given below, fill in the requirements of the MAC engine structure. What observations can you make?

Specification / Requirements	Taps (N) = 512 Sample Rate = 8 KHz	Taps (N) = 64 Sample Rate = 2 MHz	Taps (N) = 16 Sample Rate = 100 MHz
Data Samples (Nº of words)	**512**	**64**	**16**
Coefficients (Nº of words)	**512**	**64**	**16**
Memory Total (Nº of words)	**1024**	**128**	**32**
MAC Rate (Hz)	**4.096 MHz**	**128 MHz**	**1.6 GHz**
Memory Bandwidth (Words per second)	**8.192 MHz**	**256 MHz**	**3.2 GHz**
Dual Port Memory Access Rate	**4.096 MHz**	**128 MHz**	**1.6 GHz**
Single Port Memory Access Rate	**8.192 MHz**	**256 MHz**	**3.2 GHz**

XILINX®

"MAC Farm"

When NxSample_Rate indicates a frequency that is too large, one solution is to split the number of taps (N) between a "farm" of MAC engines

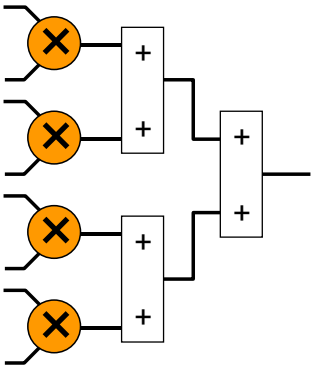


Hint: 16 taps will make distributed RAM very attractive

XILINX®

Sample Rate Affects Architecture

Parallel

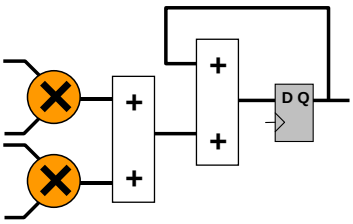


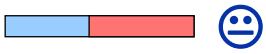
Speed 

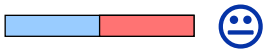
Area 

Great for High Sample Rates (Many MHz)

Somewhere in between

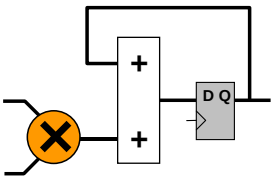


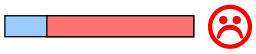
Speed 

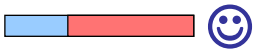
Area 

Great for Medium Sample Rates (Few MHz)

Serial



Speed 

Area 

Great for Low Sample Rates (kHz)

advico microelectronics

High-Speed Mixed Signal ASIC Design

SEI-Frühjahrstagung
7.-9.04.2008
Forschungszentrum Karlsruhe (FZK)
in der Helmholtz-Gemeinschaft,
Institut für Prozessdatenverarbeitung und Elektronik (IPE)

High-Speed Mixed-Signal ASIC Design

Hassan Safdary, Günter Grau
hs@advico.de
+49(0)2361/90438-73

advico microelectronics High-Speed Mixed Signal ASIC Design

Presentation Overview:

- About advICo
- Introduction
- ASIC Design Styles & Design Flow
- Standard Cell Design
- Structured ASIC
- Mixed-Signal ASIC Design

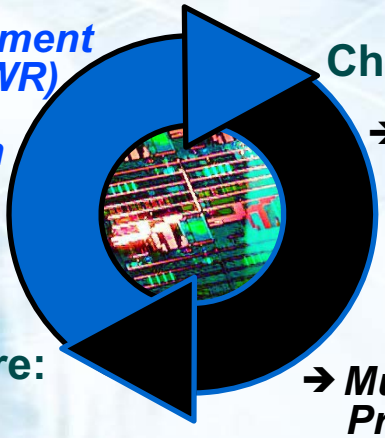
High-Speed Mixed Signal ASIC Design

advICo is an IC Design House located in Recklinghausen,Germany

Core competences:

- ➔ *tool Development (Cadence, AWR)*
- ➔ *HDL-Based System Design*
- ➔ *Design Kit Development and Support*

EDA Software:



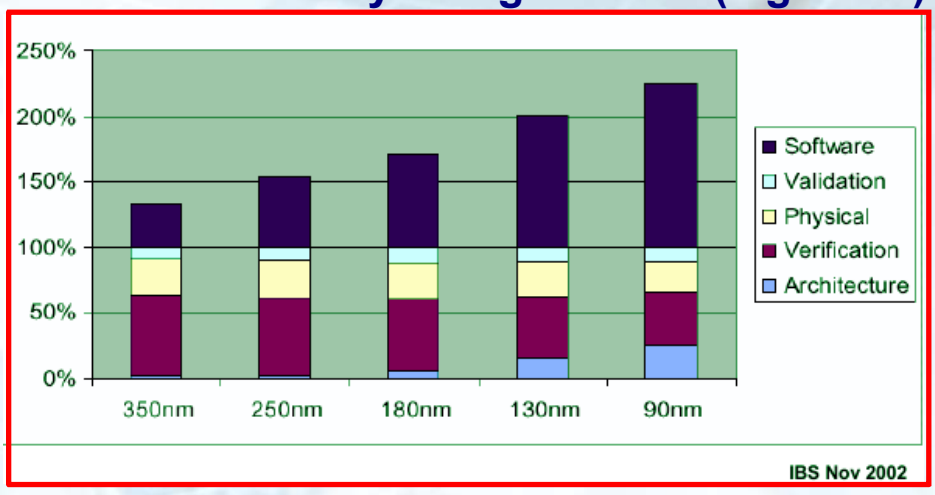
Chip Design:

- ➔ *SiGe BiCMOS and CMOS (0.25-0.13µm)*
- ➔ *IC Design Services (Mixed-Signal Analog IP/ broadband/RF)*
- ➔ *Multi project Wafer (MPW)/ Prototyping service/ Measurement*

High-Speed Mixed Signal ASIC Design

- **Introduction**

Relative Effort by Designer Role (e.g. SOC)



source:STMicroelectronics

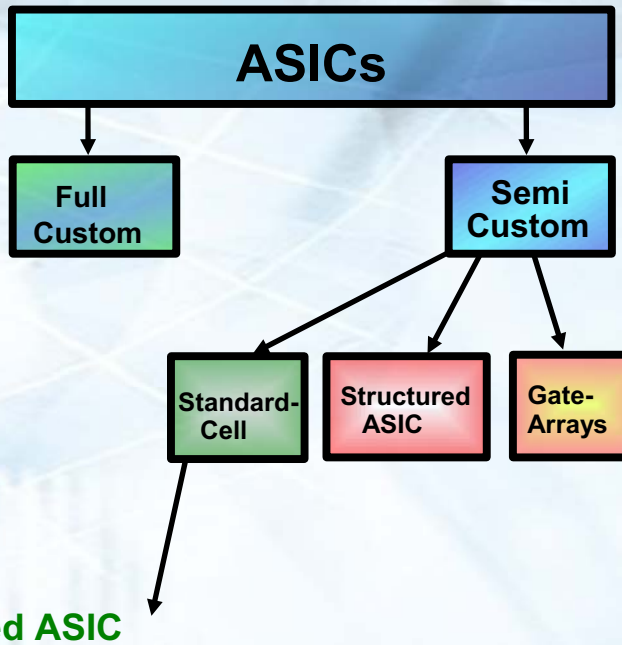
- ➢ **Software costs overtakes total hardware costs at 130nm**
- ➢ **part of Architecture overtakes physical design at 90nm**

High-Speed Mixed Signal ASIC Design

Chosen Design Styles depends on:

High-Speed Mixed Signal ASIC Design

ASIC Design Styles:



Full- Custom ASICs:

All mask layers are customized
Highest performance
Lowest power consumption
Optimum of area
Lowest per-unit cost

Cell-based ASIC
Standard Cell ASIC (SC ASICs)

advico microelectronics

High-Speed Mixed Signal ASIC Design

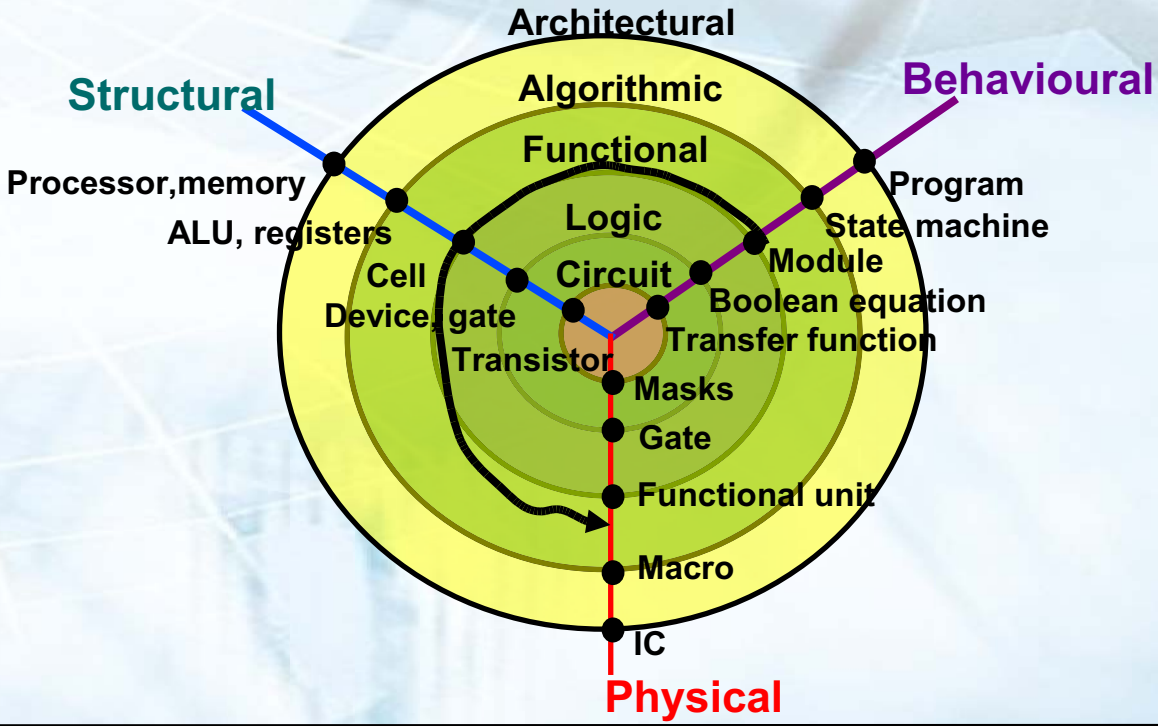
Comparison of Design Styles:

	Full custom	Standard Cell	Structured ASIC	FPGA	Gate array
Density	Very high	High	Medium	Medium	High
Performance	Very high	High	High	Medium	High
Flexibility	Very high	High	high	Low	Medium
Design time	Very long	Short	short	Very short	Short
Manufacturing time	Medium	Medium	Very short	Very short	Short
Unit cost-same quantity	Very high	High	Medium	Low	High
Unit cost-large quantity	Low	Low	Low	High	Low

advico microelectronics

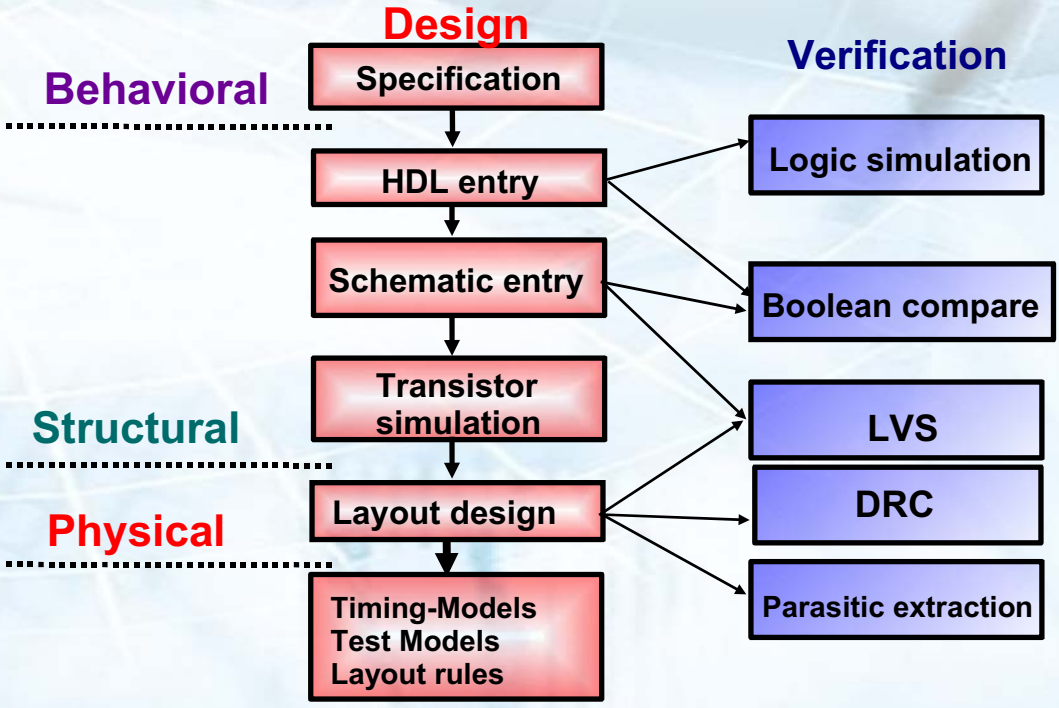
High-Speed Mixed Signal ASIC Design

Design-Strategy in "Y-Diagram":



High-Speed Mixed Signal ASIC Design

Standard-cell-based Design Flow:



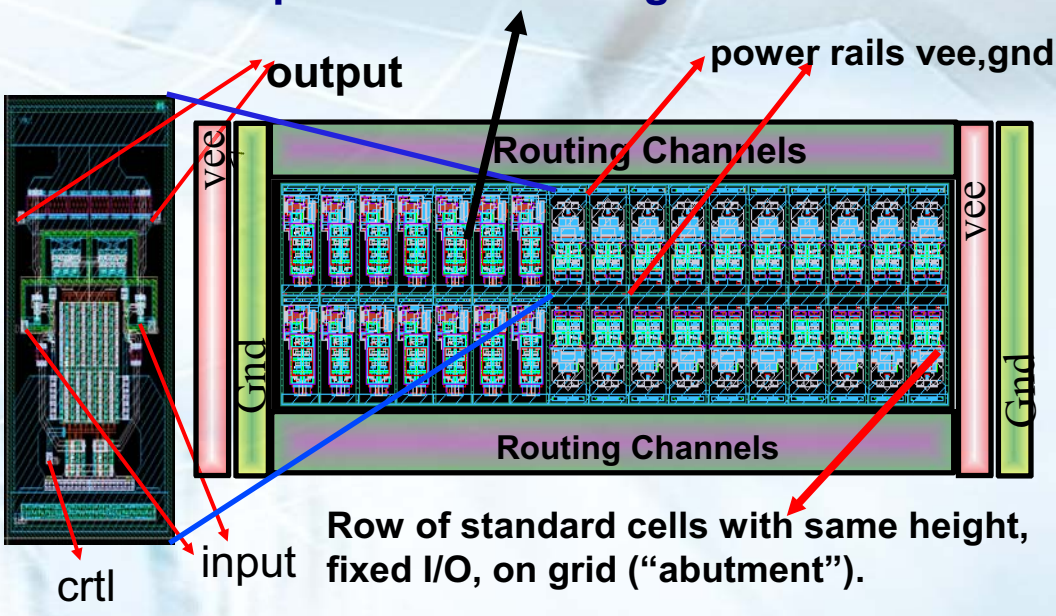
source: IBM

High-Speed Mixed Signal ASIC Design

Standard cell Design: A library of pre-characterized cells which customer can use to implement own design.

Benefit:
- Lower Risk
- Performance
- Power
- Capacity
- IP Integration
- Verified (Timing..)

Drawback:
- NRE-Cost
- Time to market



Row of standard cells with same height, fixed I/O, on grid ("abutment").

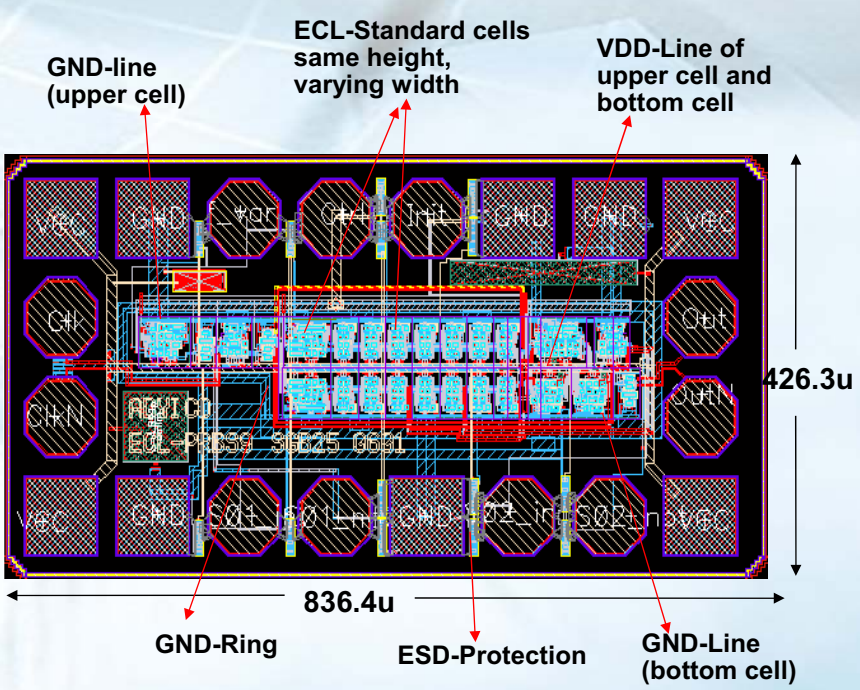
Cooperation: IHP microelectronics, FH-Brandenburg, Humboldt University, advICo microelectronics

High-Speed Mixed Signal ASIC Design

Pseudo-Random Binary Sequence Generator (PRBS)

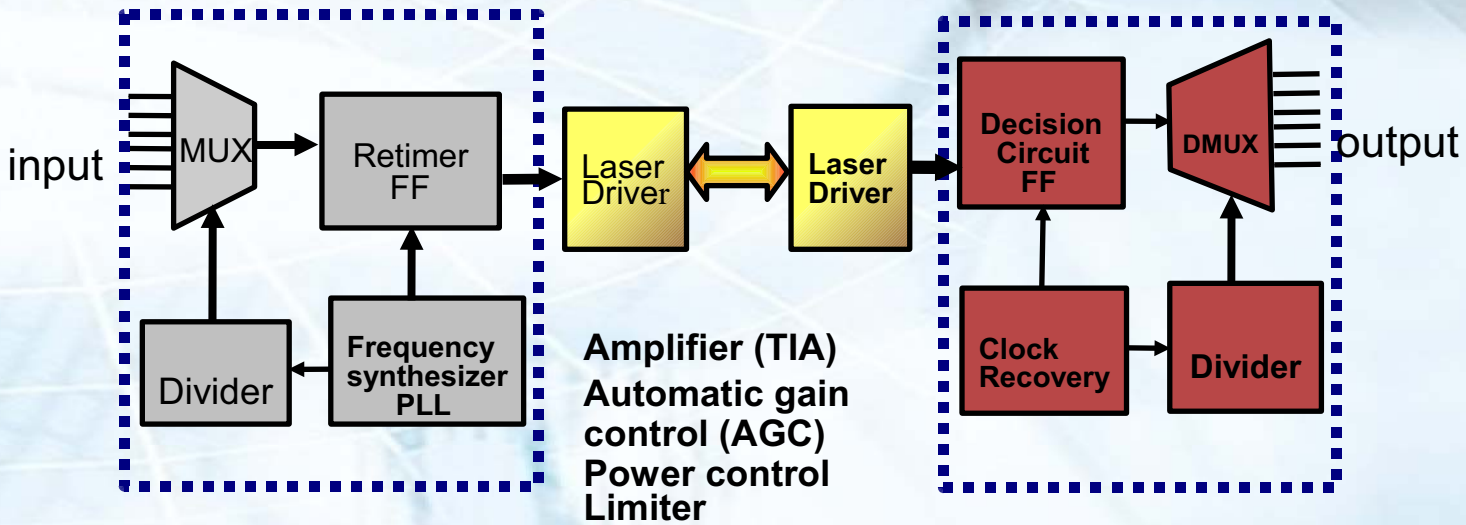
Test Chip for:

- Verification of Design specification (T, fmax, Timing, Jitter, swing, ...)
- Verification of Design specification depending on marginal conditions (MIN, TYP, MAX)
- Measurement of VDD-/GND -Drop and therewith the effect on Power-Grid respectively its max. length and min. line width



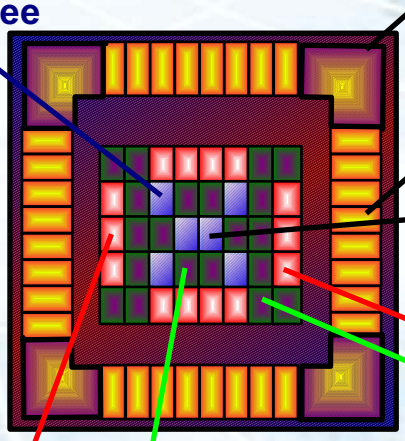
High-Speed Mixed Signal ASIC Design

An application example: Components of Fiber Optic Transceiver



High-Speed Mixed Signal ASIC Design

Structured ASIC: Alternative architecture structured ASIC



Hard IP:standard I/O and Interface (Pads, power rails,Buffer, ECL to CMOS...

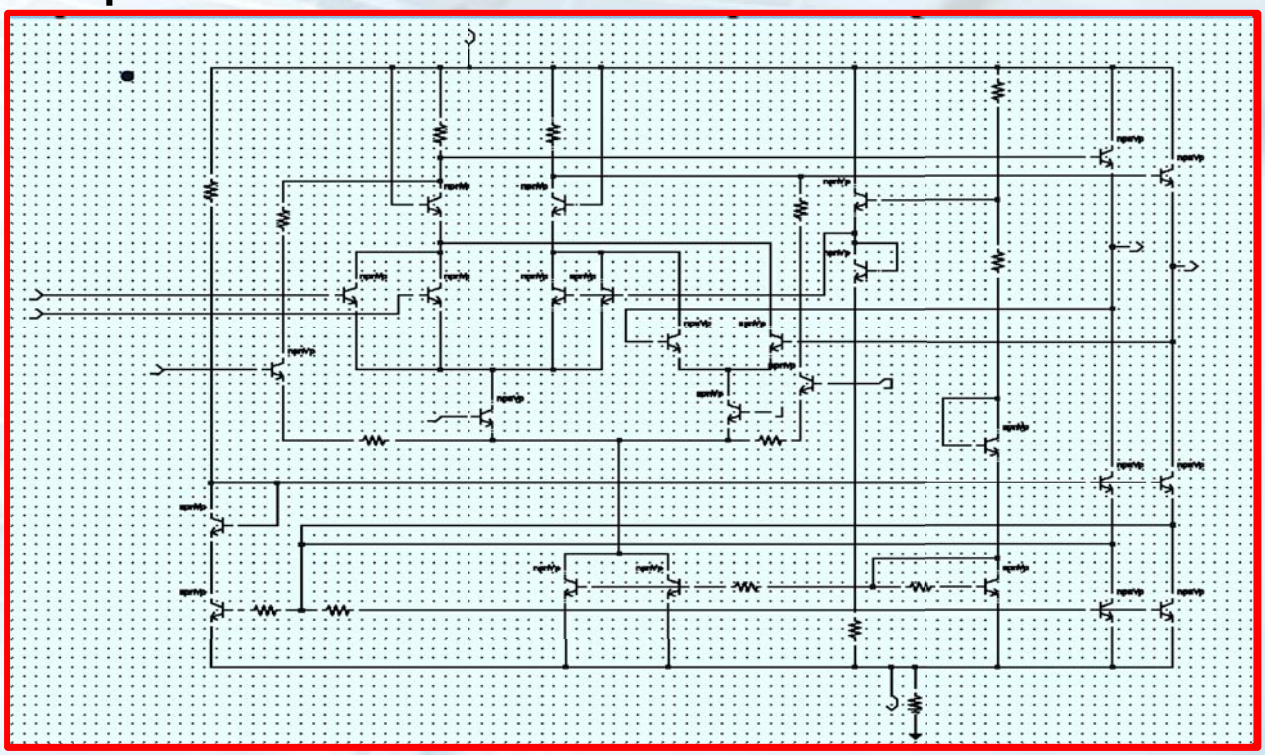
Firm IP: High-level function, optimally mapped, placed, routed (Clock Tree, RAM...

Soft IP:source-level library of high-level function , designed and routed by customer (synthesizable)

Benefit: **Low NRE and TTM ,Prototype design, design flexibility, Short development time medium volume,high performance, IP Integration, Power consumption,fast design re-spins**

High-Speed Mixed Signal ASIC Design

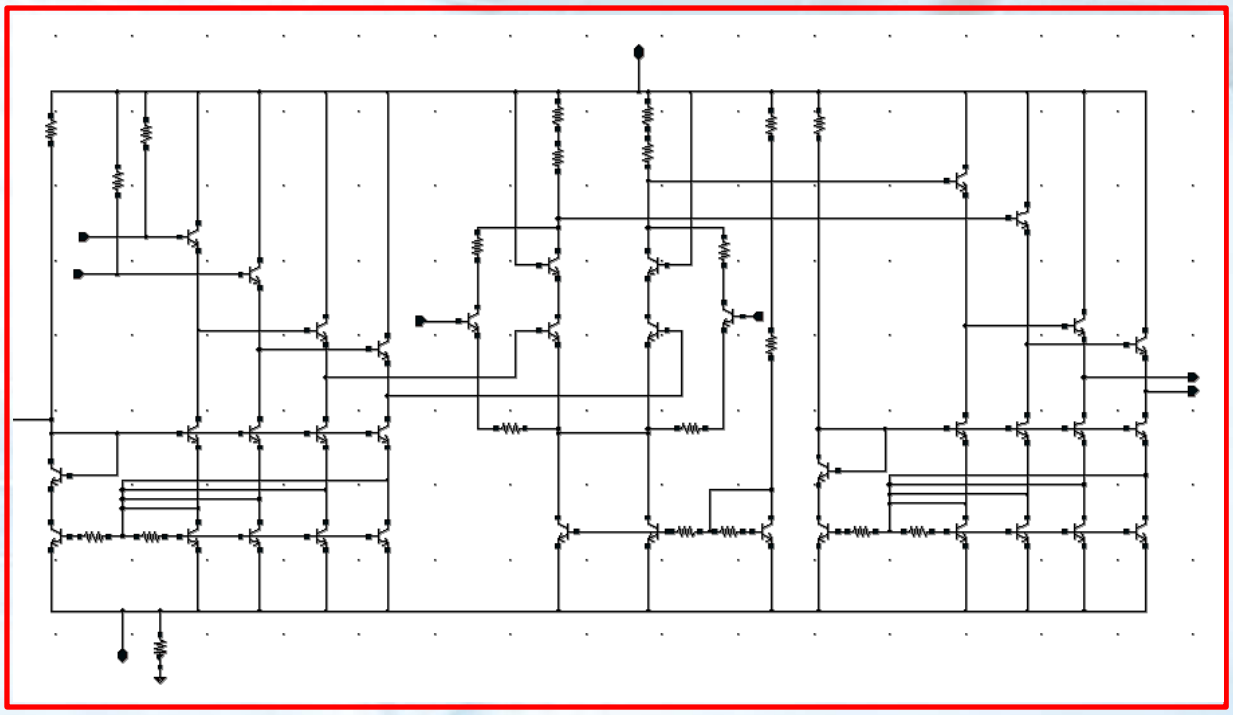
Simplified Circuit of NOR/OR Latch:



advico microelectronics

High-Speed Mixed Signal ASIC Design

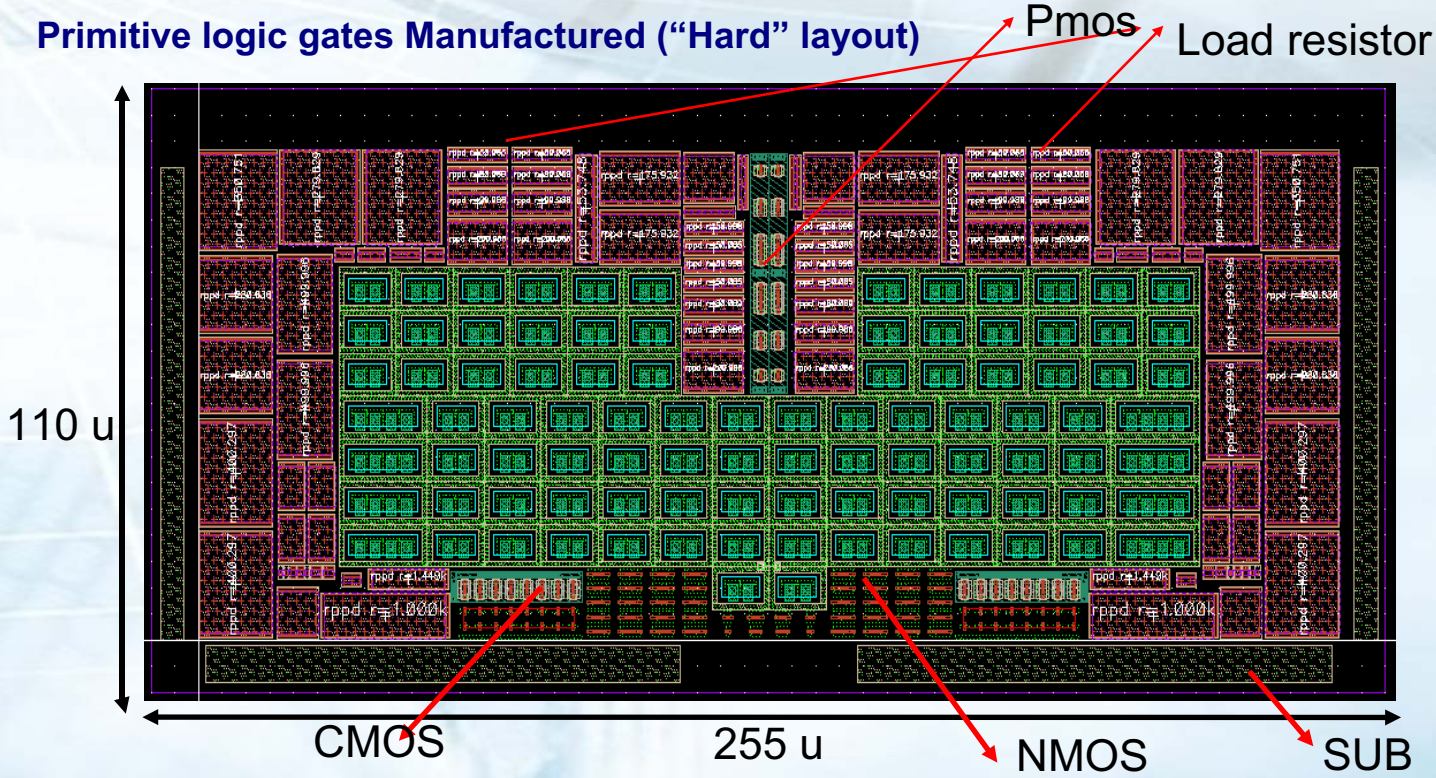
Simplified Circuit of clock Driver:



advico microelectronics

High-Speed Mixed Signal ASIC Design

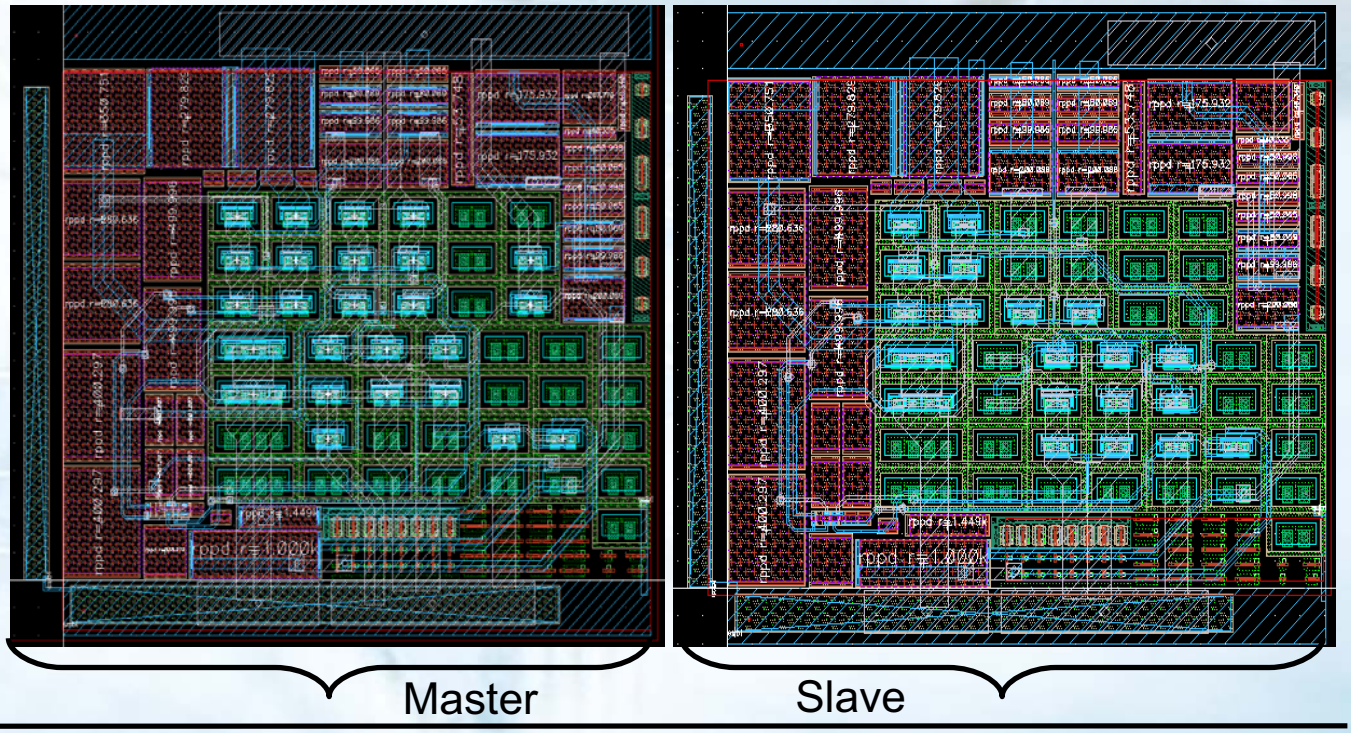
Primitive logic gates Manufactured (“Hard” layout)



advico microelectronics

High-Speed Mixed Signal ASIC Design

NOR/OR Latch



advico microelectronics

High-Speed Mixed Signal ASIC Design

- I/O- Master Cell

Diode

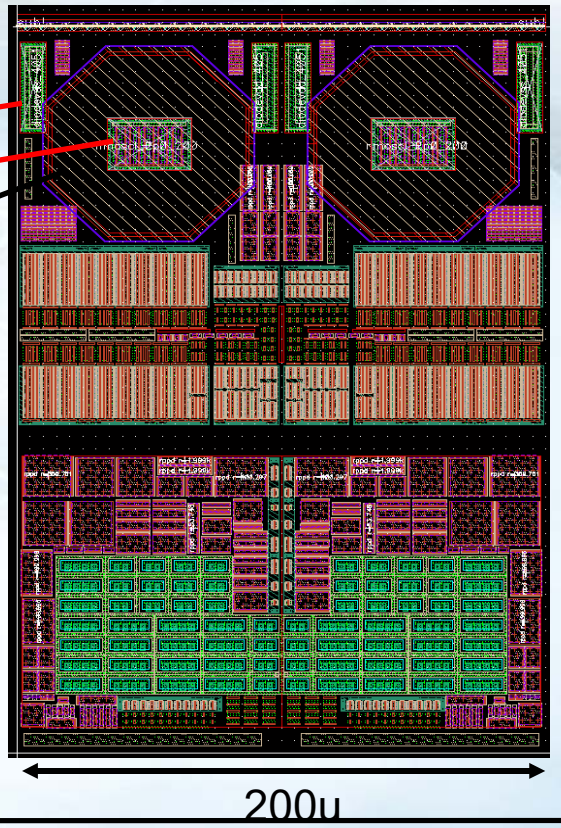
ESD

Pad with Pitch 100u

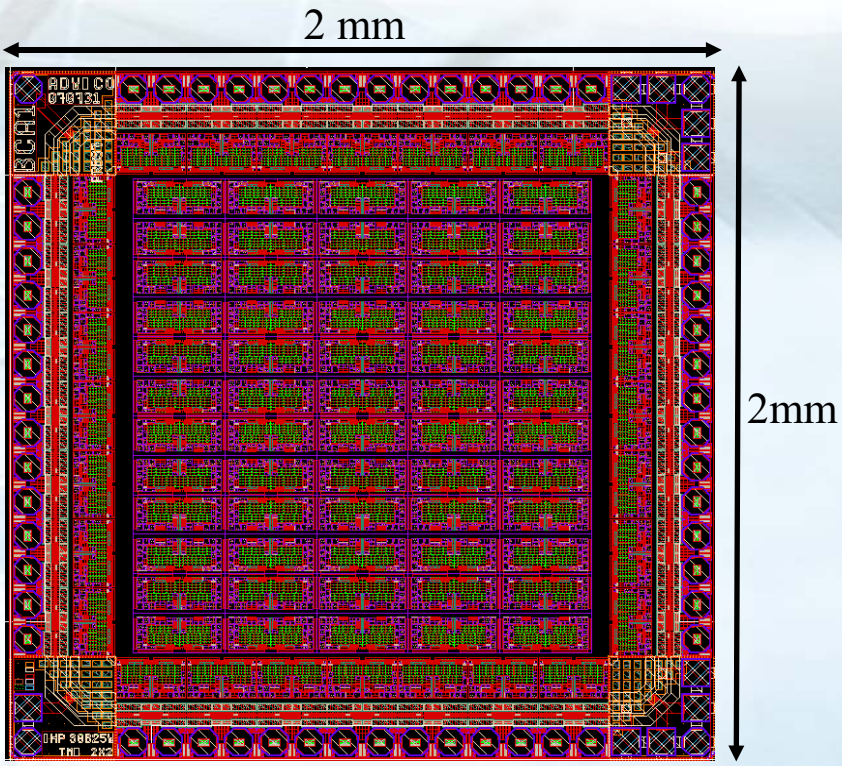
CMOS Driver

50 R Buffer
ECL to CMOS

CMOS Gate



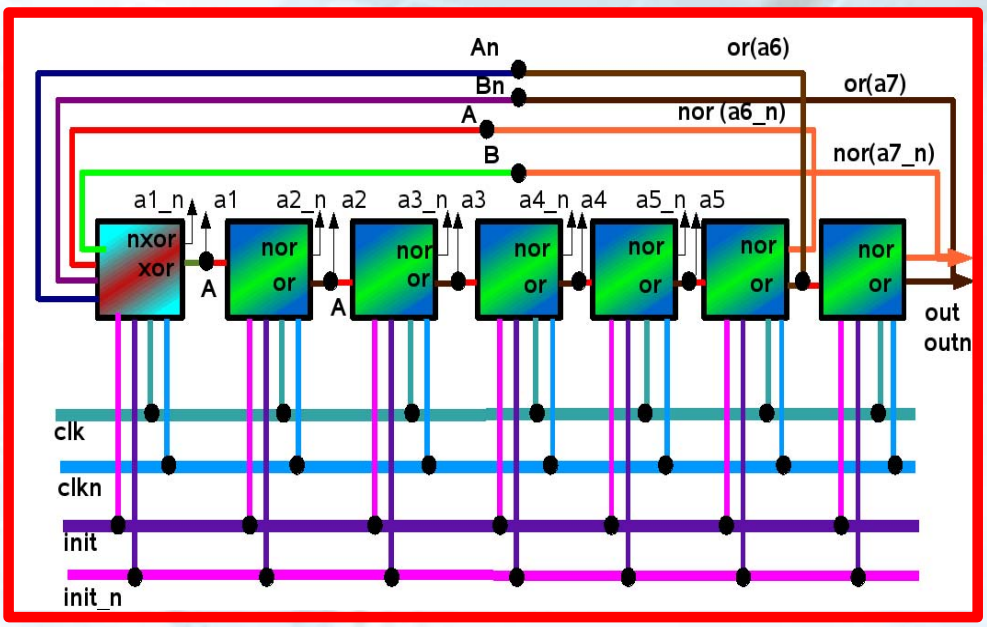
High-Speed Mixed Signal ASIC Design



- Master slice contains 60 basic cells in a matrix by 12 rows and 5 columns.
- 28 Differential I/O (ECL_logic, CMOS-logic)
- VCC, VCCIO, GND-Power rails
- 4 Corner cells
- Technology SiGe:C BiCMOS SGB25VD

High-Speed Mixed Signal ASIC Design

PRBS-7 (Block diagram):

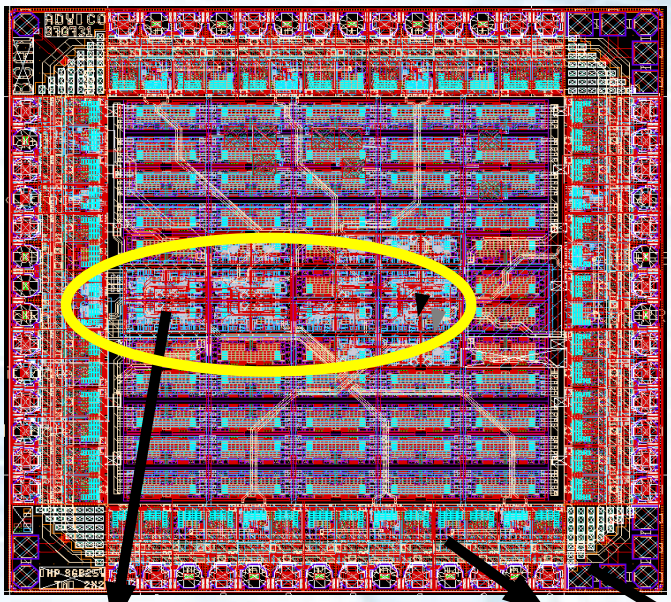


Pseudo random bit sequence generator

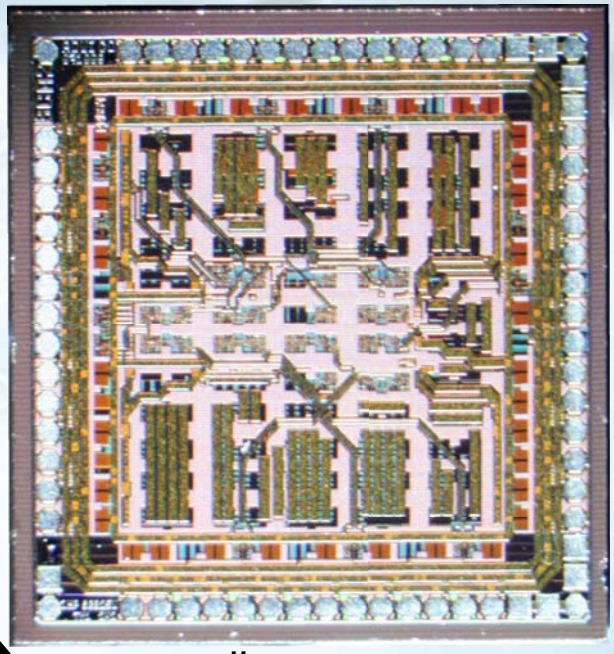
High-Speed Mixed Signal ASIC Design

Pad limited Chip: Design of PRBs-7 Based on structured ASIC (10GHz)

Layout



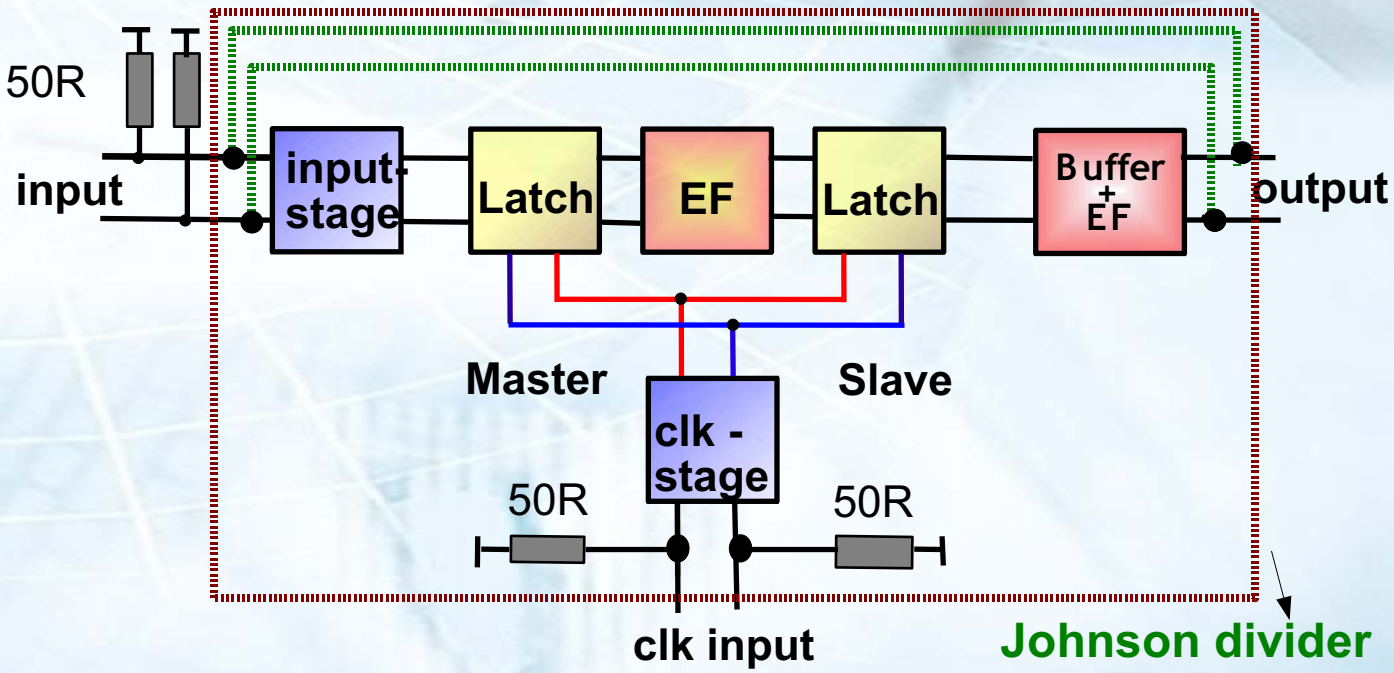
Chip photo



High-Speed Mixed Signal ASIC Design

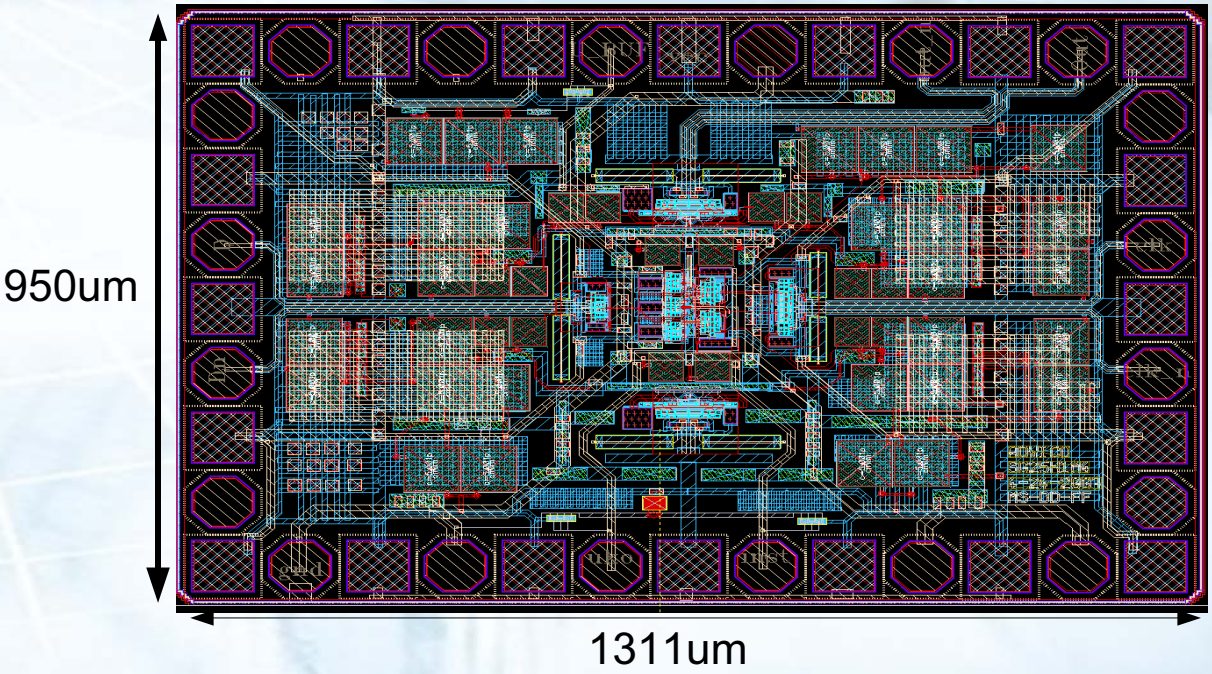
Mixed-Signal ASIC Design:

Block Diagram of a Master- Slave D- Filpflop (MS-D-FF)



High-Speed Mixed Signal ASIC Design

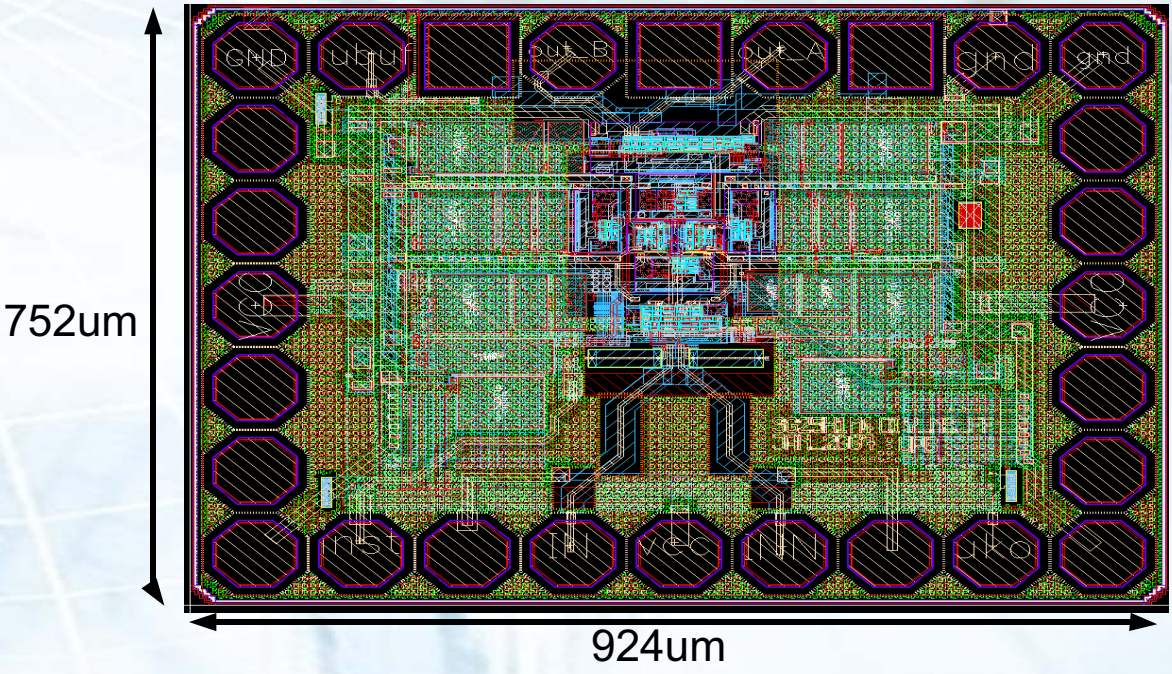
- **ECL Master- Slave-D-Flip-flop (40 GHz)**



Technology:High-Performance 0.25 um SiGe BiCMOS SG25H1 IHP

High-Speed Mixed Signal ASIC Design

Layout of Divider

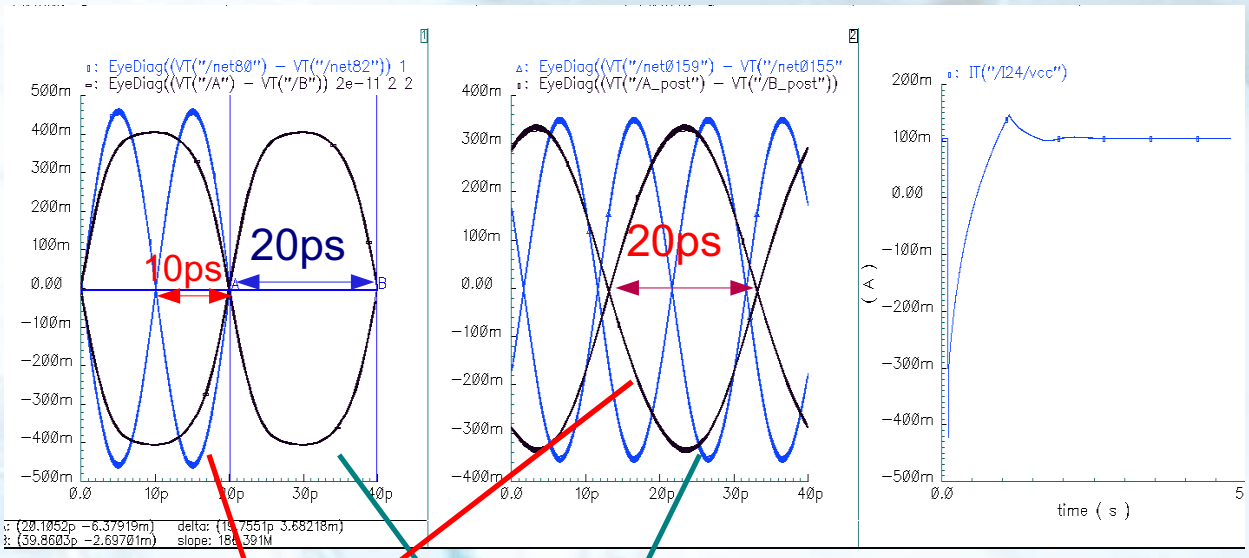


Technology:High-Performance 0.25 um SiGe BiCMOS SG25H1 IHP

High-Speed Mixed Signal ASIC Design

100 G/bs Divider: (SG25h1m4)

pre_simulation post_simulation current=148 mA

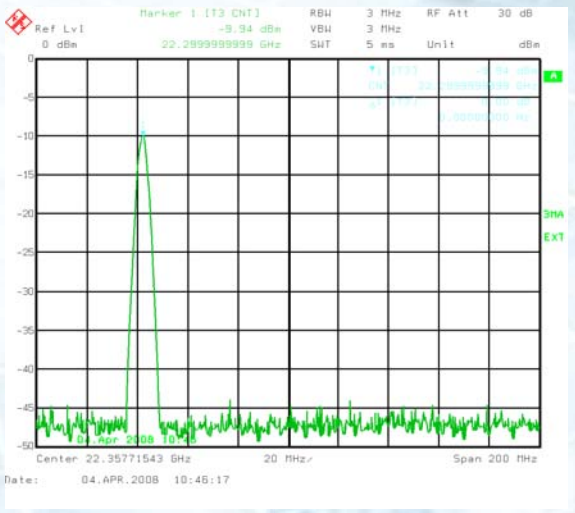


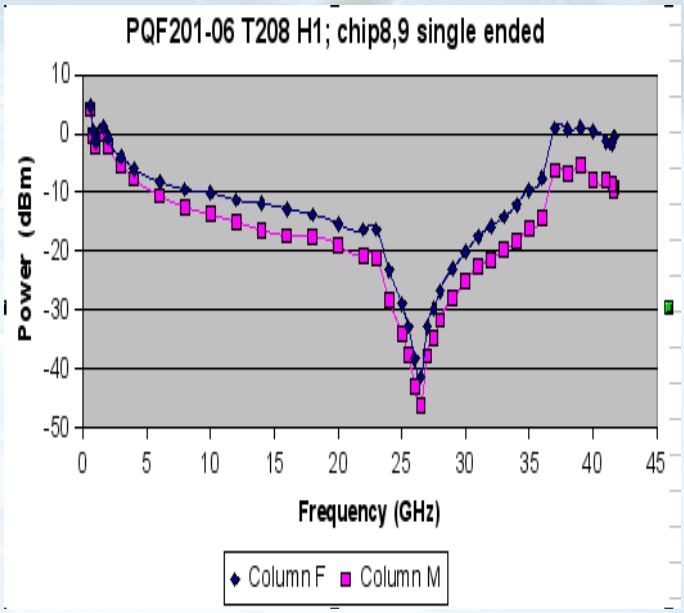
input=50 GHz output=25 GHz swing 600mV

Differential input and output

High-Speed Mixed Signal ASIC Design

Preliminary Measurements Results:



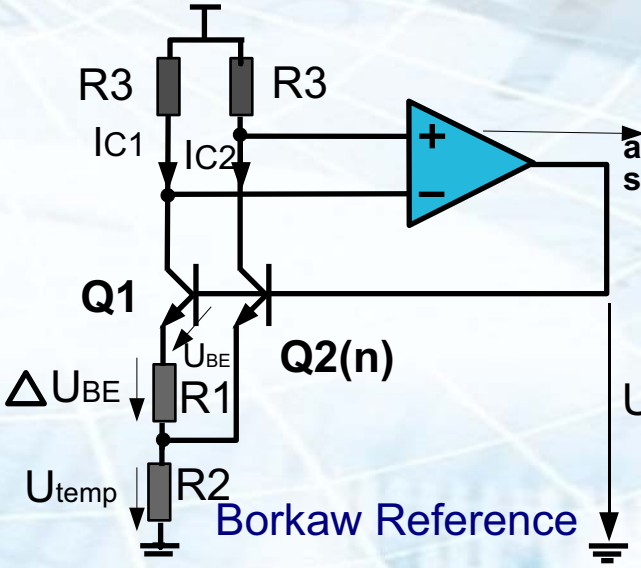


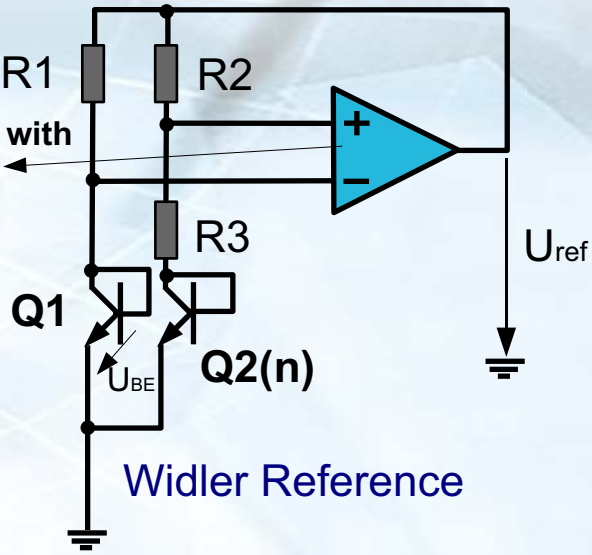
Measured output spectrum at 44.6 GHz input frequency.

Input power sensitivity

High-Speed Mixed Signal ASIC Design

- **Block Diagram of Bandgap Reference Circuit**





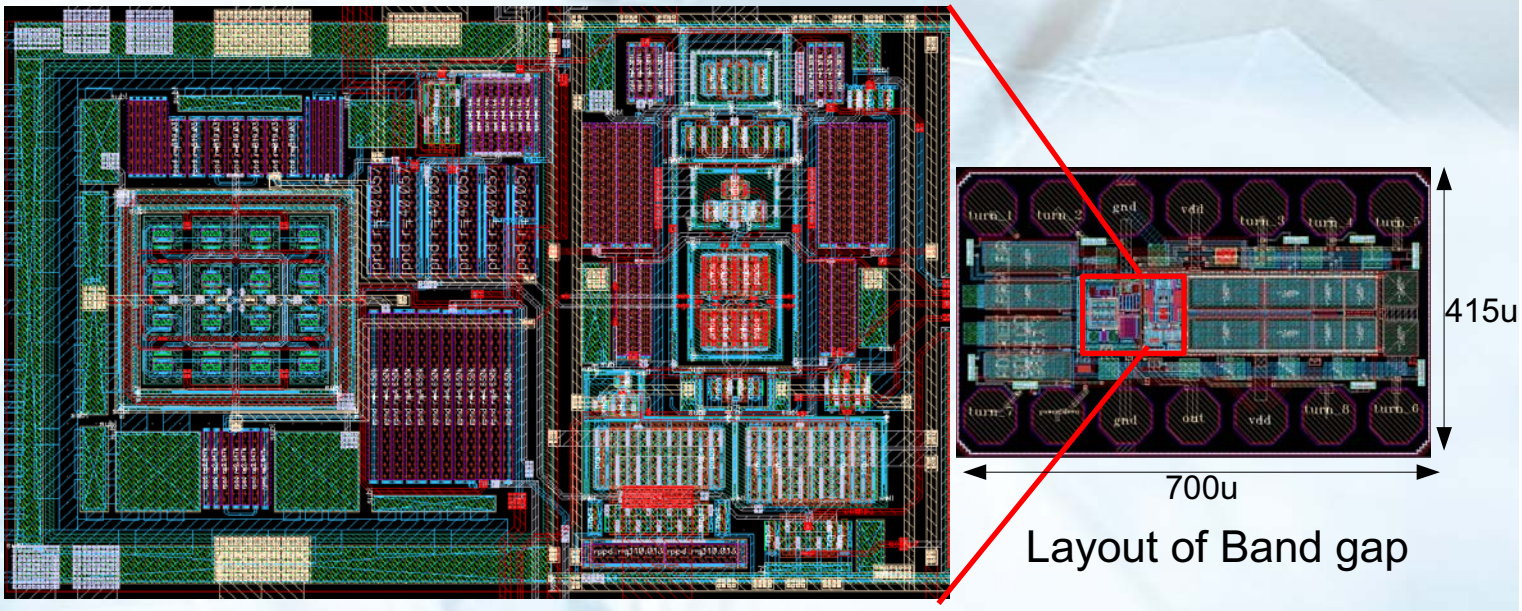
$\Delta U_{BE}=U_T*\ln(I_{C2}/I_{C1})$, $(I_{C2}=nI_{C1})$, $U_T=KT/e$

$U_{temp}=U_T*(R2/R1)*(1+n)*\ln(n)=U_T*A$ **PTAT (Proportional-to-absolute-temperature)**

$U_{ref}=U_{temp}+U_{BE}$ $dU_{BE}/dT=-2$ mV/K $dU_{temp}/dT=A*U_T/T=+2$mV/K

High-Speed Mixed Signal ASIC Design

Specification : T - 40..120 C°, power: 1.8V..3.3V, TC: 9ppm/C°

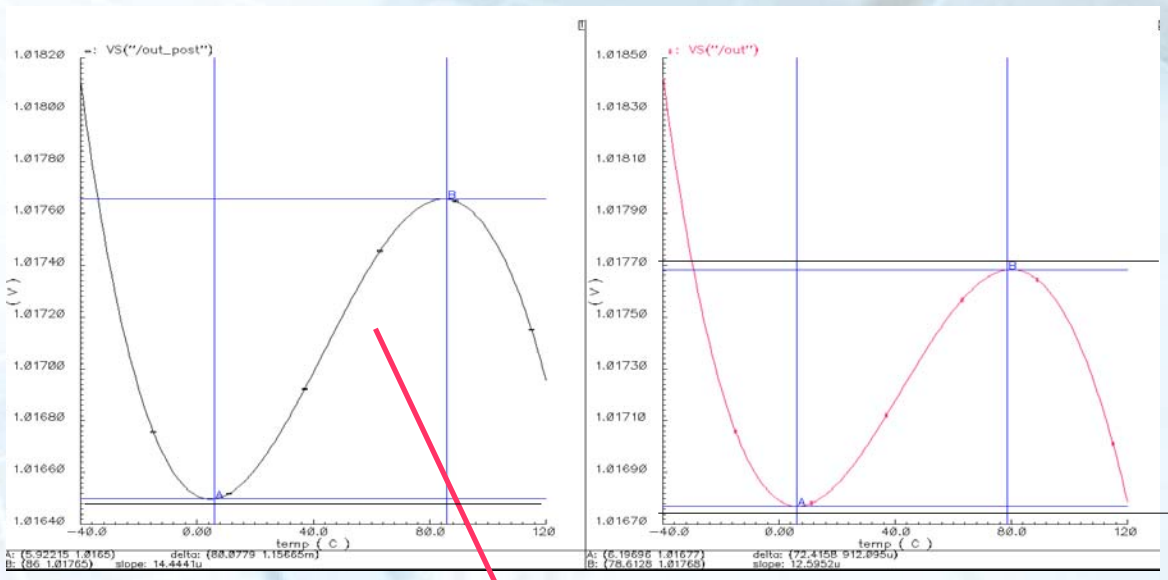


Layout of Band gap

Technology:High-Performance 0.25 um SiGe BiCMOS SG25VD IHP

High-Speed Mixed Signal ASIC Design

- **Pre- Versus Post-Simulation of curvature corrected bandgap references**



High-Speed Mixed Signal ASIC Design

Contact

advICo microelectronics GmbH
Steinstr. 12
45657 Recklinghausen
Germany
Tel.: +49-2361-90438-73
Fax: +49-2361-90438-69
info@advico.de
www.advico.de
Hassan Safdary
hs@advcio.de

Rauschen von resistiv gekoppelten Ladungsverstärkern

SEI-Tagung Forschungszentrum Karlsruhe 2008
7. - 9. 4. 2008

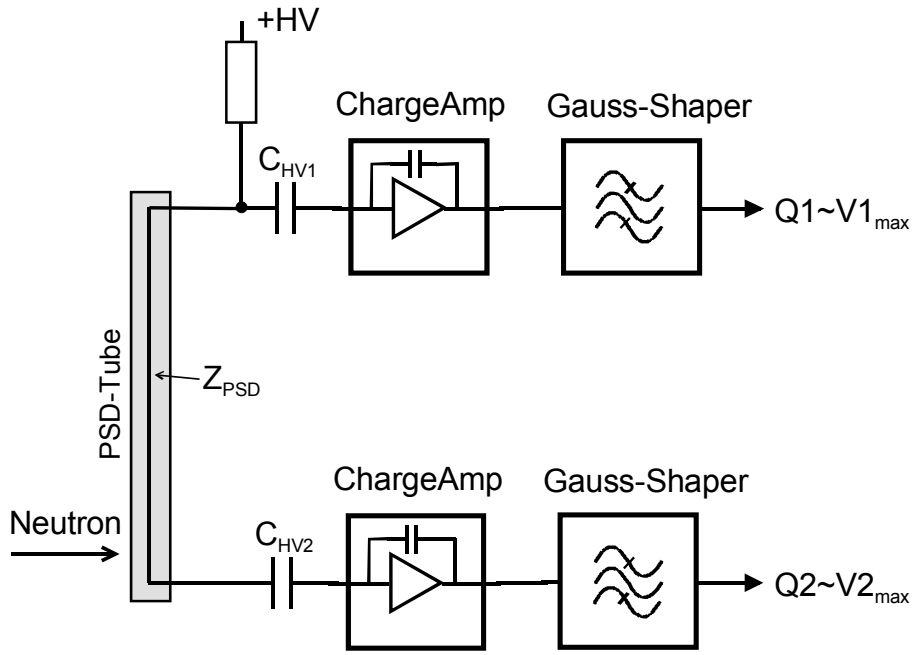
Dr. Sven Bönisch
Dr. Bernhard Namaschk
Dr. Friedrich Wulf

Hahn-Meitner-Institut Berlin in der Helmholtz-Gemeinschaft

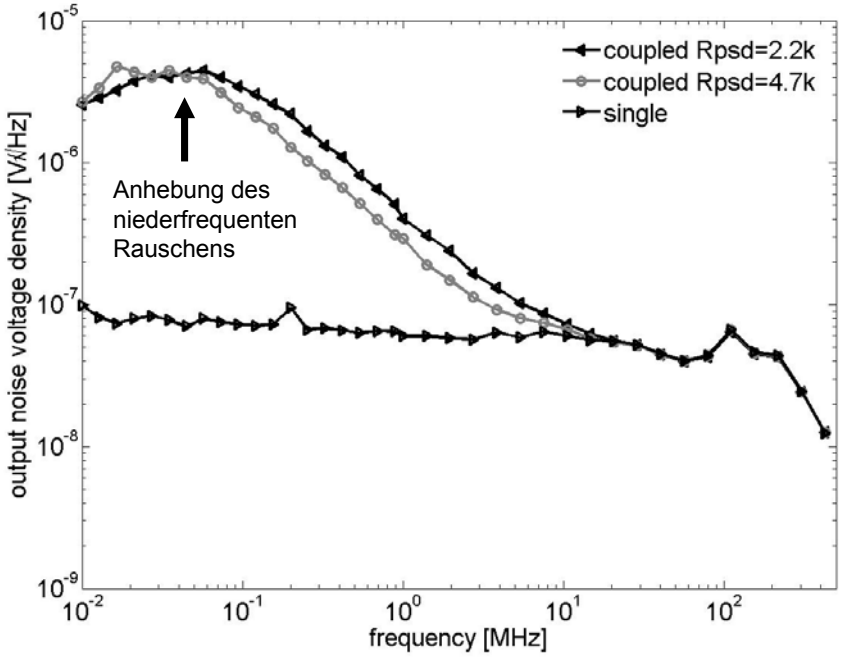
Inhalt

- Kopplung von Ladungsverstärkern
- Rauschanhebung im Niederfrequenzbereich
- Rauschanalyse im Frequenzbereich: Frequenzabhängige Rauschverstärkung
- Rauschanalyse im Zeitbereich: Rauschspannungskorrelation
- Zusammenfassung

Ortsauflösender Neutronendetektor mit Auswerteelektronik



Messung des Ausgangsrauschens eines Ladungsverstärkers

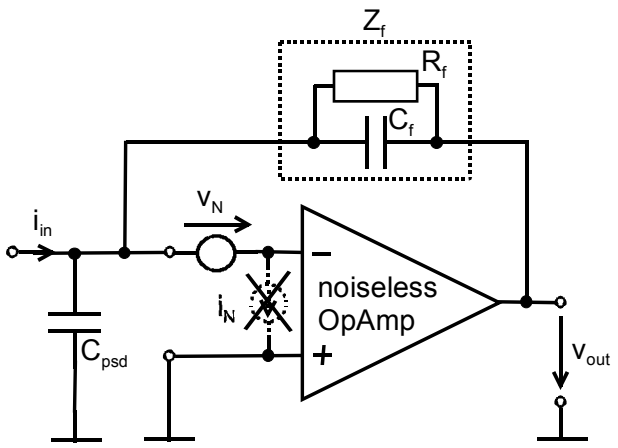


Konfiguration Ladungsverstärker

Equivalente Eingangsrausch-spannungsdichte 4.8nV/√Hz (OPA657)

Bauteil	Wert
R_{psd}	2.2kΩ
C_{psd}	15pF
C_{HV}	3.3nF
R_f	100MΩ
C_f	1.5pF

Rauschmodellierung Operationsverstärker

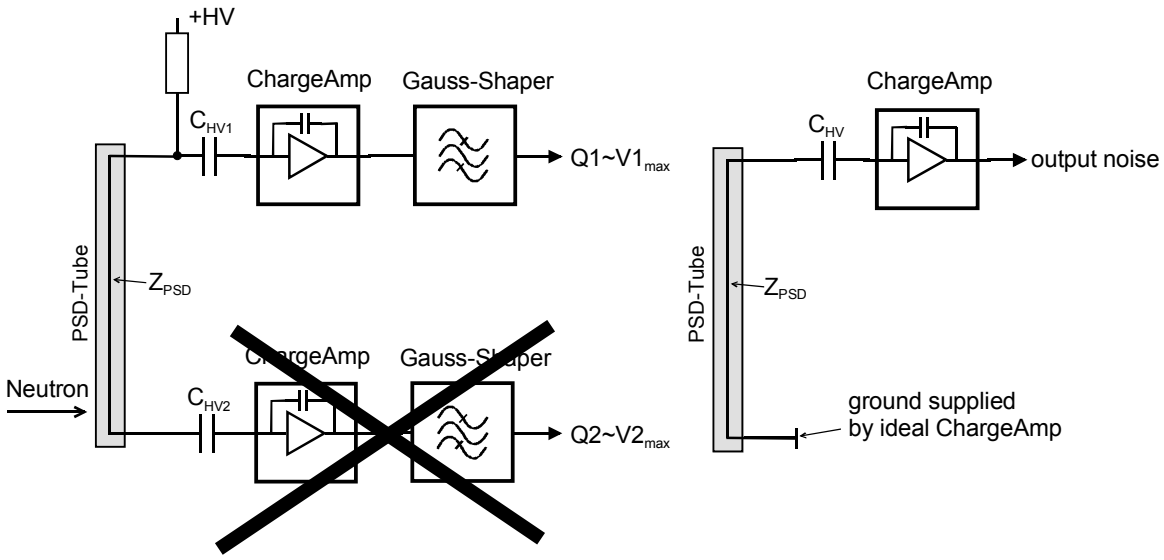


Konfiguration

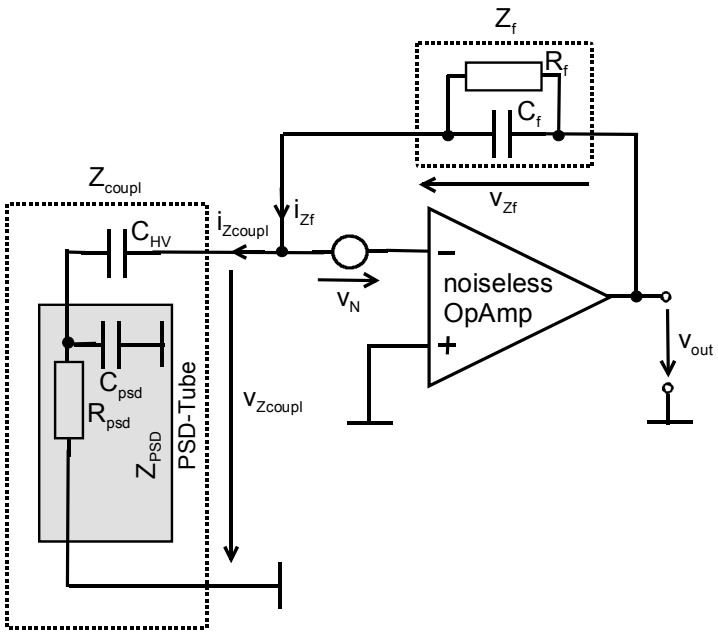
Parameter	Wert
OpAmp	OPA657
v_N	4.8nV/√Hz
i_N	1.2fA/√Hz
C_{psd}	15pF
R_f	100MΩ
C_f	1.5pF
T	300K
B_{Lim}	$1/(2\pi R_f C_f)$
i_{EQ}	eq. Eingangs-rauschsstrom

$$i_{EQ} = \sqrt{i_N^2 + \frac{4kT}{R_f} + \left(\frac{v_N}{R_f}\right)^2 + \frac{\left(2\pi C_{psd} B_{LIM} v_N\right)^2}{3}} \approx 13\,fA/\sqrt{Hz} = v_N\,@\,28kHz$$

Vereinfachung für Rauschanalyse



Berechnung der Rauschverstärkung



$$g_{Noise} = \frac{v_{out}}{v_N}$$

$$g_{Noise} = \frac{v_{Zcoupl} + v_{Zf}}{v_{Zcoupl}}$$

$$i_{Zf} = i_{Zcoupl}$$

$$\frac{v_{Zf}}{Z_f} = \frac{v_{Zcoupl}}{Z_{Coupl}}$$

$$g_{Noise} = 1 + \frac{Z_f}{Z_{Coupl}}$$

$$g_{Noise} = 1 + \frac{X_{Cf} \,\|\, R_f}{X_{CHV} + (X_{Cpsd} \,\|\, R_{psd})}$$

Rauschverstärkung

$$g_{Noise} = 1 + \frac{X_{Cf} \,\|\, R_f}{X_{CHV} + (X_{Cpsd} \,\|\, R_{psd})}$$

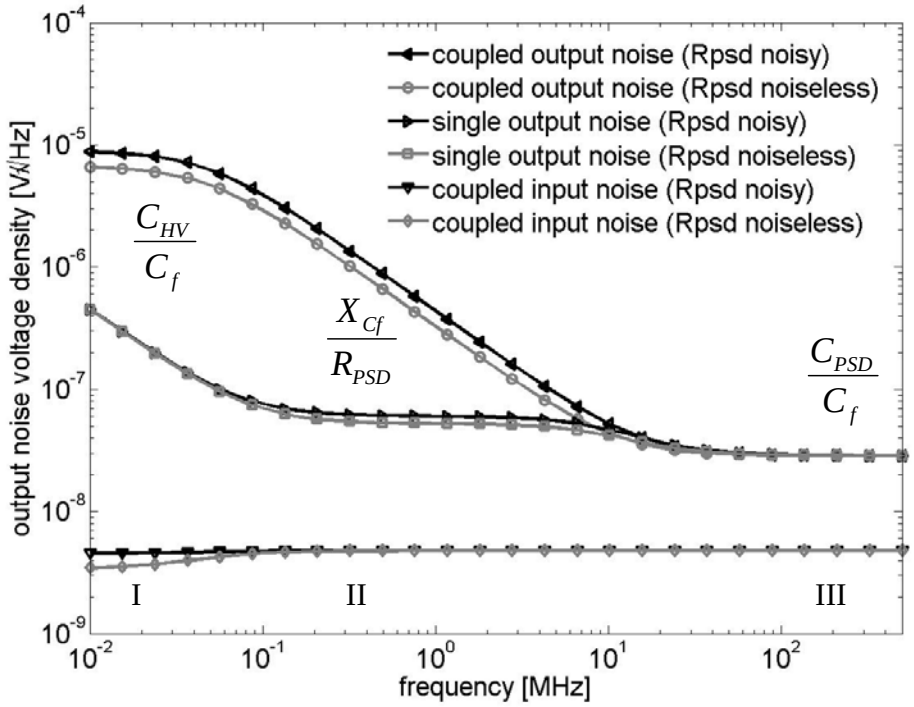
Bereich	**I**	**II**	**III**
Frequenz	<20kHz	<5MHz	>5MHz
X_{CHV}	$>R_{psd}$	$<R_{psd}$	$<R_{psd}$
X_{Cf}	$>R_{psd}$	$>R_{psd}$	$>R_{psd}$
X_{Cpsd}	$>R_{psd}$	$>R_{psd}$	$<R_{psd}$
g_{Noise}	$\approx \frac{C_{HV}}{C_f}$	$\approx \frac{X_{Cf}}{R_{PSD}}$	$\approx \frac{C_{PSD}}{C_f}$
	2200	>12	11
Anstieg	konstant	~1/f	konstant

Konfiguration Ladungsverstärker

Equivalente Eingangsrausch-spannungsdichte 4.8nV/√Hz (OPA657)

Bauteil	**Wert**
R_{psd}	2.2kΩ
C_{psd}	15pF
C_{HV}	3.3nF
R_f	100MΩ
C_f	1.5pF

Frequenzabhängiges Rauschverhalten

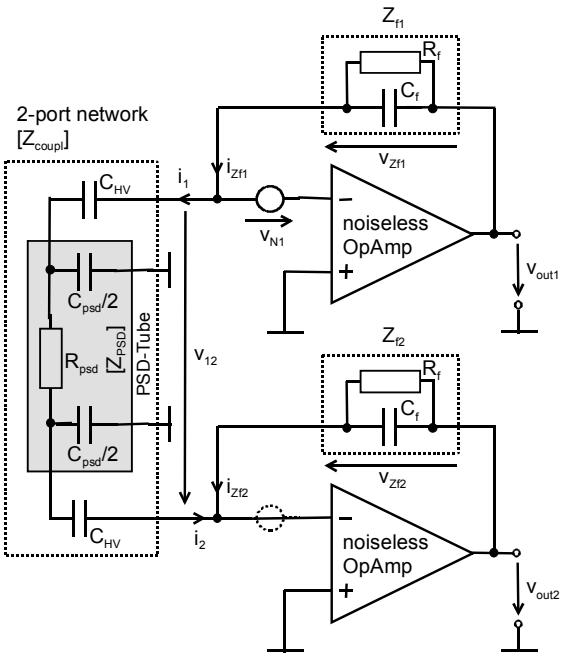


Konfiguration Ladungsverstärker

Equivalente Eingangsrausch-spannungsdichte 4.8nV/√Hz (OPA657)

Bauteil	Wert
R_{psd}	2.2kΩ
C_{psd}	15pF
C_{HV}	3.3nF
R_f	100MΩ
C_f	1.5pF

Berechnung der Rauschkorrelation



$$i_{Zf1} = i_1$$

$$v_{out1} = i_1 \cdot Z_{f1} + v_{N1}$$

$$i_1 = \frac{v_{N1}}{X_{CHV} + 2 \cdot X_{Cpsd} \,\|\, (R_{PSD} + 2 \cdot X_{Cpsd} \,\|\, X_{CHV})}$$

$$v_{out1} = v_{N1} \cdot \left(1 + \frac{X_{Cf} \,\|\, R_f}{X_{CHV} + 2 \cdot X_{Cpsd} \,\|\, (R_{PSD} + 2 \cdot X_{Cpsd} \,\|\, X_{CHV})}\right)$$

$$i_{Zf2} = -i_2$$

$$v_{out2} = -i_2 \cdot Z_{f2}$$

$$i_2 = \frac{v_{N1}}{X_{CHV} + 2 \cdot X_{Cpsd} \,\|\, (R_{PSD} + 2 \cdot X_{Cpsd} \,\|\, X_{CHV})}$$

$$v_{out2} = -v_{N1} \cdot \frac{X_{Cf} \,\|\, R_f}{X_{CHV} + 2 \cdot X_{Cpsd} \,\|\, (R_{PSD} + 2 \cdot X_{Cpsd} \,\|\, X_{CHV})}$$

Rauschkorrelation

$$v_{out1} = v_{N1} \cdot \left(1 + \frac{X_{Cf} \,\|\, R_f}{X_{CHV} + 2 \cdot X_{Cpsd} \,\|\, (R_{PSD} + 2 \cdot X_{Cpsd} \,\|\, X_{CHV})} \right) \qquad v_{out2} = -v_{N1} \cdot \frac{X_{Cf} \,\|\, R_f}{X_{CHV} + 2 \cdot X_{Cpsd} \,\|\, (R_{PSD} + 2 \cdot X_{Cpsd} \,\|\, X_{CHV})}$$

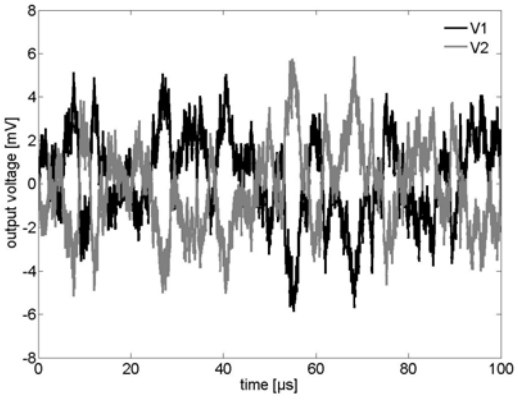
Bereich	**I**	**II**	**III**
Frequenz	<20kHz	<5MHz	>5MHz
X_{CHV}	$>R_{psd}$	$<R_{psd}$	$<R_{psd}$
X_{Cf}	$>R_{psd}$	$>R_{psd}$	$>R_{psd}$
X_{Cpsd}	$>R_{psd}$	$>R_{psd}$	$<R_{psd}$
v_{out1}	$\approx v_{N1} \frac{C_{HV}}{2C_f}$	$\approx v_{N1} \frac{X_{Cf}}{R_{PSD}}$	$\approx v_{N1} \left(1 + \frac{C_{PSD}}{2C_f} \right)$
v_{out2}	$\approx -v_{N1} \frac{C_{HV}}{2C_f}$	$\approx -v_{N1} \frac{X_{Cf}}{R_{PSD}}$	$\approx -v_{N1} \frac{C_{PSD}}{2C_f}$
Rausch-korrelations koeffizient	≈ -1	≈ -1	$\approx -\frac{C_{PSD}}{2C_f + C_{PSD}}$ $\approx -0.8\overline{3}$

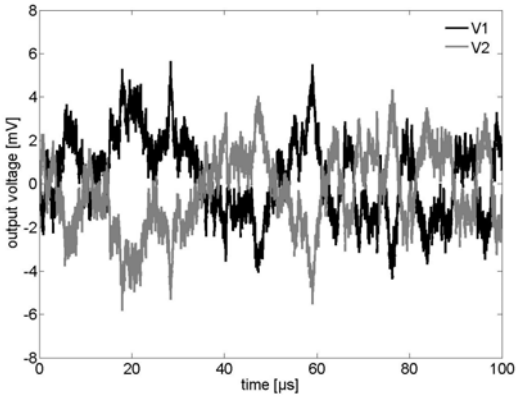
Konfiguration Ladungsverstärker

Equivalente Eingangsrausch-spannungsdichte 4.8nV/√Hz (OPA657)

Bauteil	**Wert**
R_{psd}	2.2kΩ
C_{psd}	15pF
C_{HV}	3.3nF
R_f	100MΩ
C_f	1.5pF

Transiente Rauschsimulation

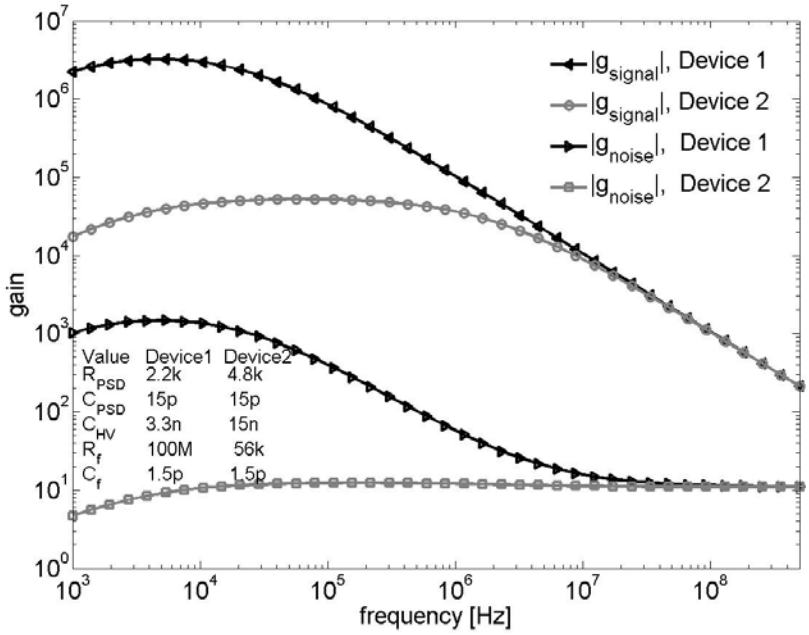




Bauteil	**Wert**
OPA657	4.8nV/√Hz
R_{psd}	2.2kΩ
C_{psd}	15pF
C_{HV}	3.3nF
R_f	100MΩ
C_f	1.5pF

Bauteil	**Wert**
OPA657	4.8nV/√Hz
R_{psd}	2.2kΩ ***!! rauschfrei !!***
C_{psd}	15pF
C_{HV}	3.3nF
R_f	100MΩ
C_f	1.5pF

Vergleich von verschiedenen Konfigurationen

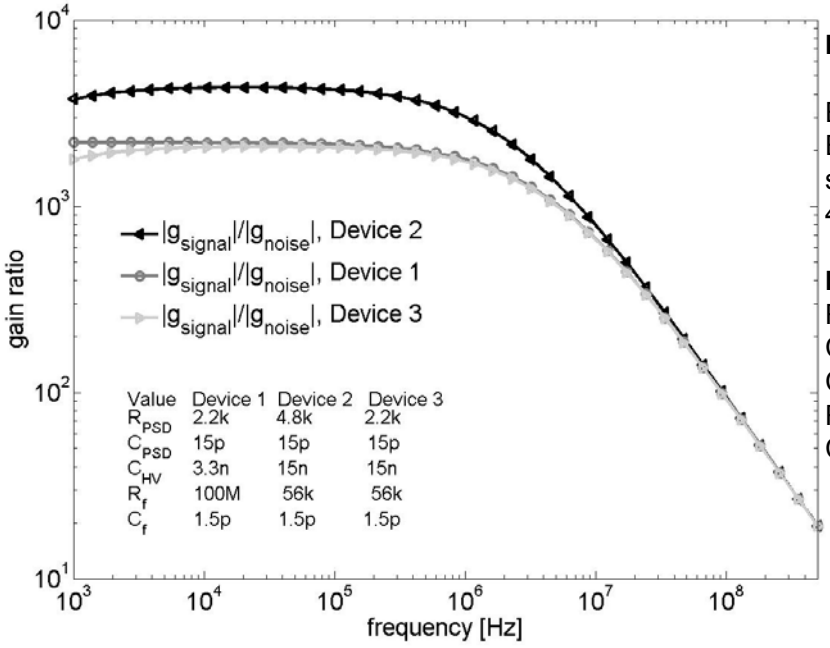


Konfiguration

Equivalente Eingangsrausch-spannungsdichte 4.8nV/√Hz (OPA657)

Bauteil	Dev1	Dev2
R_{psd}	2.2kΩ	4.8k
C_{psd}	15pF	15p
C_{HV}	3.3nF	15n
R_f	100M	56k
C_f	1.5pF	1.5p

Verhältnis von Signalverstärkung zu Rauschverstärkung



Konfiguration

Equivalente Eingangsrausch-spannungsdichte 4.8nV/√Hz (OPA657)

Bauteil	Dev1	Dev2	Dev3
R_{psd}	2.2k	4.8k	2.2k
C_{psd}	15p	15p	15p
C_{HV}	3.3n	15n	15n
R_f	100M	56k	56k
C_f	1.5p	1.5p	1.5p

Zusammenfassung

- Anhebung des Niederfrequenzrauschens durch Kopplung über die Detektorimpedanz
- Der Mechanismus der Rauschanhebung ist gültig, unabhängig vom tatsächlichen Wert des Detektordrahtwiderstandes und unabhängig von der Annahme eines idealen rauschfreien Rohrwiderstandes bzw. der Berücksichtigung des Rauschanteils des Rohrwiderstandes
- Der Effekt der Rauschanhebung aufgrund der Kopplung dominiert gegenüber der reinen Addition des Rauschens eines angekoppelten Verstärkers
- Oberhalb einer Eckfrequenz ist das Rauschen teilweise unkorreliert
- Die Rauschkorrelation ist unabhängig vom Drahtwiderstand des Detektorrohres
- Das Detektorrohr kann mit konzentrierten Bauelementen hinreichend gut beschrieben werden
- Der äquivalente Eingangsrauschstrom des Operationsverstärkers kann vernachlässigt werden

Rauschen von resistiv gekoppelten Ladungsverstärkern

Dr. S. Bönisch, Dr. B. Namaschk, Dr. F. Wulf
Hahn-Meitner-Institut Berlin

Für die Optimierung der analogen Auswerteelektronik eines ortsauflösenden Neutronendetektors wurden theoretische Untersuchungen durchgeführt, die es gestatten das Rauschverhalten von resistiv gekoppelten Ladungsverstärkern analytisch zu beschreiben. Hierzu wurden die frequenzabhängige Rauschverstärkung sowie die zeitliche Korrelation des Rauschens analytisch hergeleitet und mit Hilfe von Simulationen und Messungen verifiziert. Die analytische Herleitung soll helfen, die Mechanismen des Rauschens von solchen gekoppelten Systemen besser zu verstehen und ein Optimum des Signal-Rauschabstandes zu finden.

Die Rauschmodellierung eines Operationsverstärkers erfolgt mit Hilfe von auf den Eingang bezogenen äquivalenten Rauschspannungs- und Rauschstromquellen. Es zeigt sich jedoch, daß im hier betrachteten Fall die äquivalente Eingangsrauschstromquelle vernachlässigt werden kann. Zusätzlich ist oberhalb einer Frequenz von ca. 280kHz das Stromrauschen aller beteiligten Bauteile gegenüber dem Spannungsrauschen vernachlässigbar. Zur weiteren Vereinfachung der analytischen Ableitung der Rauschverstärkung eines Ladungsverstärkers ist es möglich nur diesen einen Vorverstärker mit seiner zugehörigen Eingangsrausch-spannungsquelle am Rohr anzuschließen und den anderen als ideal, d.h. als rauschfrei und mit einem idealen Eingangswiderstand von 0Ω anzunehmen. Das Detektorrohr wird als komplexe Impedanz, bestehend aus Drahtwiderstand und Rohrkapazität nach Masse berücksichtigt.

Die Berechnung der Rauschverstärkung erfolgt mit Methoden der Netzwerkanalyse für statische Netzwerke mit komplexwertigen Impedanzen. Die gewonnene Funktion der Rauschverstärkung enthält komplexe und frequenzabhängige Parameter und kann abhängig von den betrachteten Frequenzbereichen weiter vereinfacht werden. Es zeigt sich, daß die Rauschverstärkung bei Frequenzen unterhalb etwa 20kHz vergleichsweise hoch und näherungsweise konstant ist. Der Anhebungsfaktor (ca. 3 Größenordnungen) ist durch das Verhältnis von Hochspannungskoppelkapazität C_{HV} und Gegenkopplungskapazität C_f bestimmt. Zwischen 20kHz und etwa 5MHz fällt die Rauschverstärkung proportional 1/f zu größeren Frequenzen hin ab. In diesem Bereich ist die Rauschverstärkung durch das Verhältnis von Blindwiderstand der Gegenkopplungskapazität X_{Cf} zu Drahtwiderstand R_{PSD} bestimmt. Bei Frequenzen oberhalb etwa 5MHz ist die Anhebung auf etwa eine Größenordnung zurückgegangen. Die Rauschverstärkung ist hier wieder konstant und durch das Verhältnis von Rohrkapazität C_{PSD} zu Gegenkopplungskapazität C_f bestimmt. Es zeigt sich auch, daß es keinen Einfluß auf den Mechanismus der Rauschanhebung infolge der frequenzabhängigen Rauschverstärkung hat, ob der Rohrwiderstand als ideal rauschfrei oder als real rauschend angesehen wird.

Die Berechnung der Rauschkorrelation erfolgt ebenfalls mit den o. ä. Methoden. Die Impedanz des Detektorrohres ist hier jedoch mit einem 2-Tor Netzwerk zu beschreiben, da auch der herausfließende Strom zu berücksichtigen ist. Zusätzlich wird hier mit einem verfeinerten Modell mit symmetrisch verteilten Rohrkapazitäten gearbeitet. Untersuchungen zeigten jedoch, daß die Rohrimpedanz ausreichend genau mit dem oben dargestellten Modell beschrieben werden kann. Die gewonnenen Ausdrücke für die Spannungen an den Ausgängen der beiden Vorverstärker lassen sich auch hier in bestimmten Frequenzbereichen stark vereinfachen. Im Frequenzbereich <5MHz ist die Rauschkorrelation konstant -1. Oberhalb 5MHz ist die Rauschkorrelation ebenfalls konstant aber betragsmäßig <1, d.h. das Rauschen ist teilweise unkorreliert. Die Rauschkorrelation ist in diesem Bereich nur von der Rohrkapazität C_{PSD} und der Gegenkopplungskapazität C_f abhängig.

Im Vergleich verschiedener Konfigurationen des Vorverstärkers (z.B. als Ladungsverstärker oder als Strom-Spannungswandler) kann kein Optimum des Signal-Rauschabstandes bezüglich der Konfiguration gefunden werden. Ein lokales Optimum des Signal- Rauschverhältnisses wird bei Verwendung von Strom-Spannungswandlern und großen Drahtwiderständen (R_{PSD}>100kΩ) erwartet.

Folgende allgemeingültige Aussagen können aufgrund der bis jetzt gefundenen Ergebnisse getroffen werden:

- Anhebung des Niederfrequenzrauschens durch Kopplung über die Detektorimpedanz
- Der Mechanismus der Rauschanhebung ist gültig, unabhängig vom tatsächlichen Wert des Detektordrahtwiderstandes und unabhängig von der Annahme eines idealen rauschfreien Rohrwiderstandes bzw. der Berücksichtigung des Rauschanteils des Rohrwiderstandes
- Der Effekt der Rauschanhebung aufgrund der Kopplung dominiert gegenüber der reinen Addition des Rauschens eines angekoppelten Verstärkers
- Oberhalb einer Eckfrequenz ist das Rauschen teilweise unkorreliert
- Die Rauschkorrelation ist unabhängig vom Drahtwiderstand des Detektorrohres
- Das Detektorrohr kann mit konzentrierten Bauelementen hinreichend gut beschrieben werden
- Der äquivalente Eingangsrauschstrom des Operationsverstärkers kann vernachlässigt werden

Strahlprofilmessung mit Lichtwellenleitersensorik am DESY FLASH LINAC

Mit Bezug auf die Messtechnik die als Teil der HMI-DESY-Kollaboration am TESLA-Strahlrohr installiert wurde, und mit freundlicher Unterstützung der DESY Wire Scanner Group.

Kollaborations-Initiatoren:

Dr. U. Nielsen[3]; Dr. D. Trines[2]

Projektmitarbeiter:

Dr. M. Koerfer[2]; Dr. F. Wulf [3]; W. Goettmann[3]

[2] DESY-Hamburg; [3] HMI-Berlin

Hahn-Meitner-Institut Berlin in der Helmholtz-Gemeinschaft

Inhalt:

Der DESY Flash LINAC

Warum Strahlprofilmessung

Die Mess-Systematik

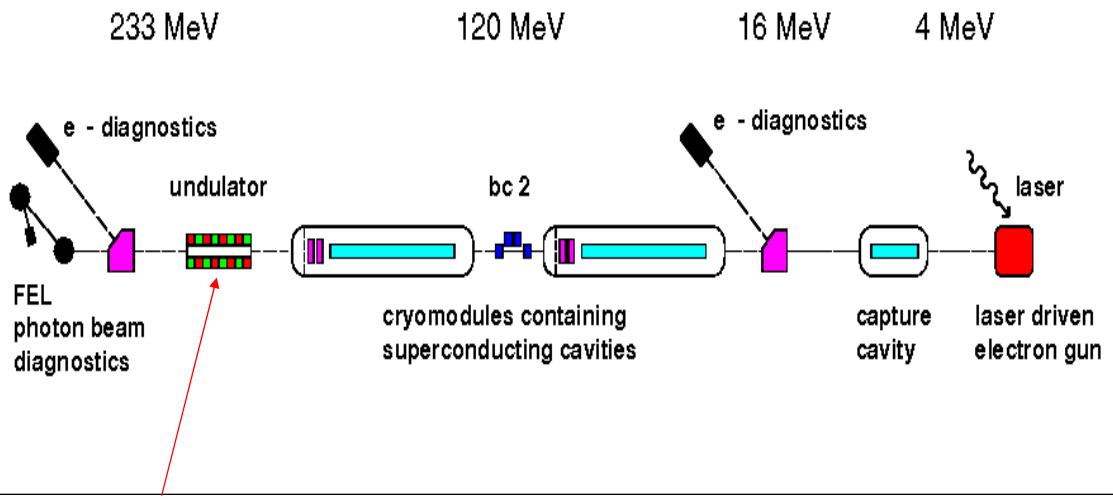
Die Messanlage im Überblick

Die Vorzüge der Lichtwellenleitersensorik

Grafiken einiger Messergebnisse

Verbesserungsmöglichkeiten

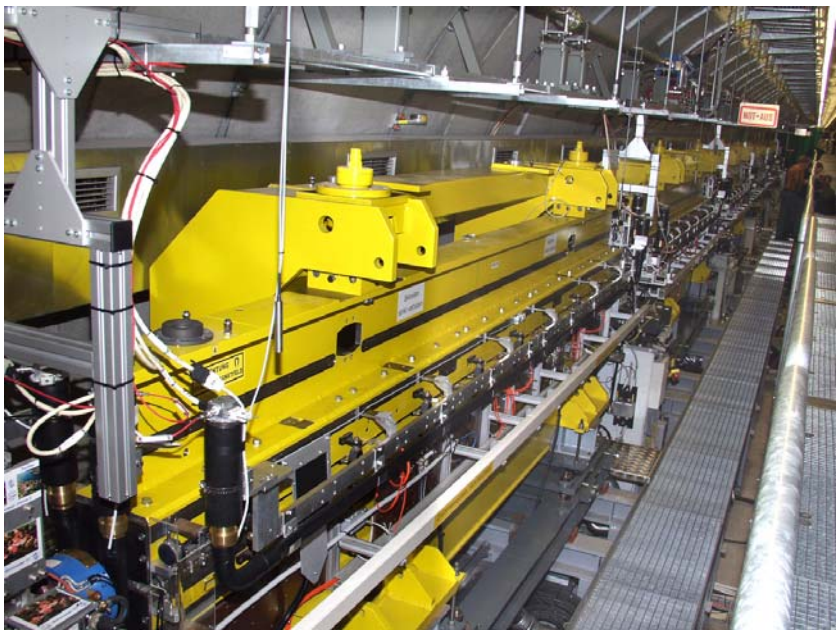
Schematische Ansicht des TTF-LINAC



Der Undulator mit seinen Permanentmagneten ist eminent wichtig, teuer, und auch strahlungsempfindlich.

Hahn-Meitner-Institut Berlin in der Helmholtz-Gemeinschaft

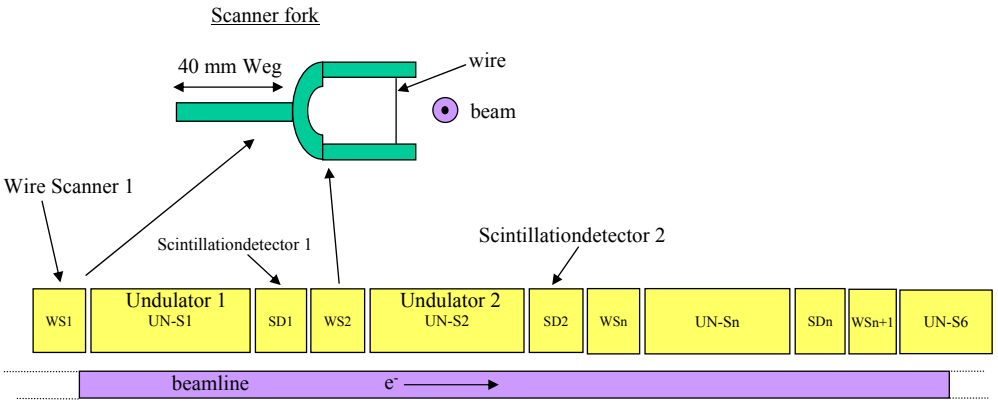
Ansicht der Undulator-Segmente im Tunnel



Beam profile measurement using wire scanner and scintillation detectors

FLASH beamline wire scanner / undulator-section

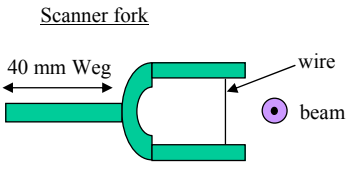
6 Undulator Segmente; Gesamtlänge = 30m

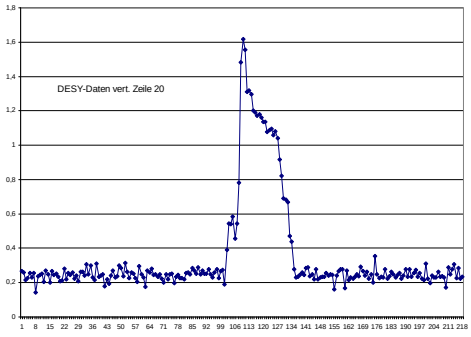


This beam profile measurement system (based on wire scanners and scintillation detectors) has been developed by physicists of DESY-Hamburg and DESY-Zeuthen. The motion of the scanner fork is realized by a programmable driven stepper motor.

Die Strahlprofilmessung gibt Hinweise auf die Ausdehnung und die Lage des Strahls im Strahlrohr.

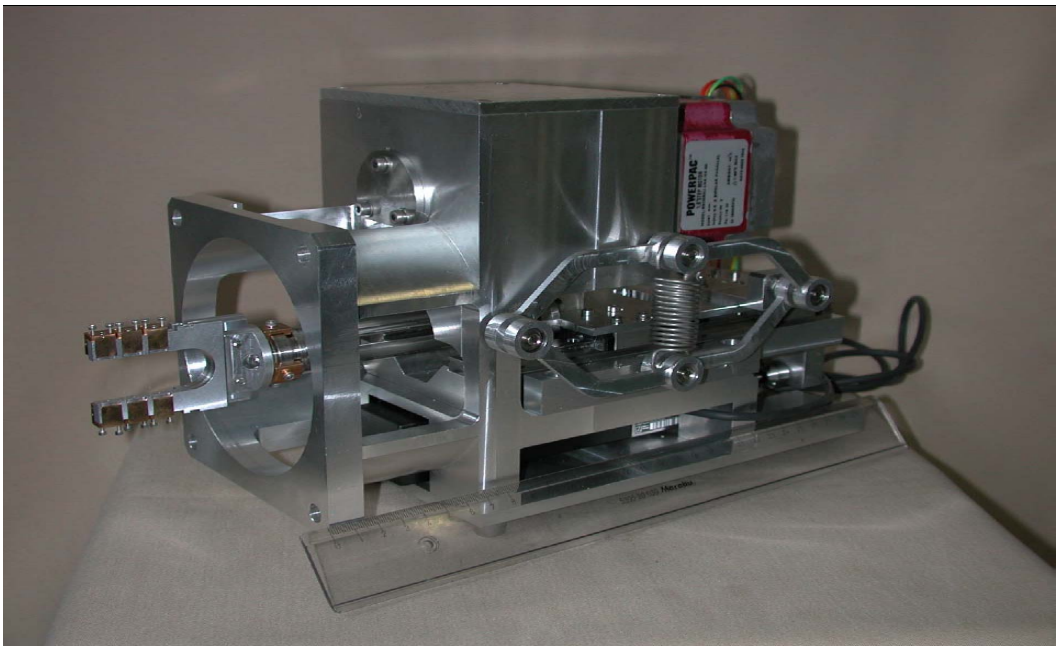
Arbeitsweise des Wire Scanners





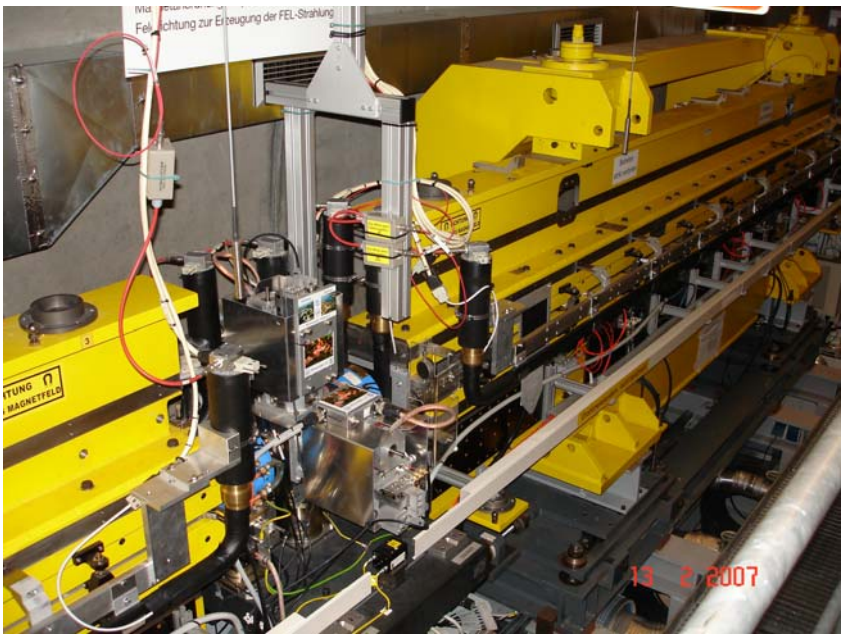
Der Draht wird mit vorbestimmter Geschwindigkeit und mit vorbestimmten Strecken in den Strahl gefahren (erzeugt Bremsstrahlung). An vorbestimmten Positionen des Drahtes wird die Intensität der Verluststrahlung ausserhalb des Strahlrohres gemessen. Jede Messung ergibt genau einen Punkt des Profil-Graphen.

Der Rücklauf des Drahtes in die Ausgangsposition erfolgt mit hoher Geschwindigkeit (bis 10 m/s), um die Stahlungsbelastung des Undulators gering zu halten, und um die Lebensdauer des Drahtes zu erhöhen.



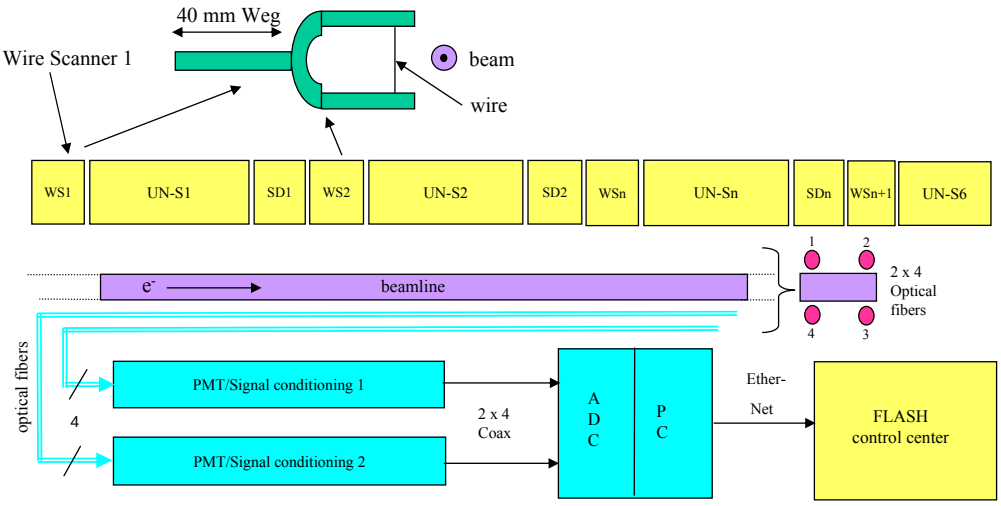
Wire scanner used at DESY-FLASH. Developed and assembled at DESY-Zeuthen.

(Das Bild wurde mit freundlicher Genehmigung dem TESLA Report 2002 – 2008 entnommen)



A complete x-y wire scanner system positioned in the gap betwenn two undulator segments

Additional components of thr Cherenkov BLPM system

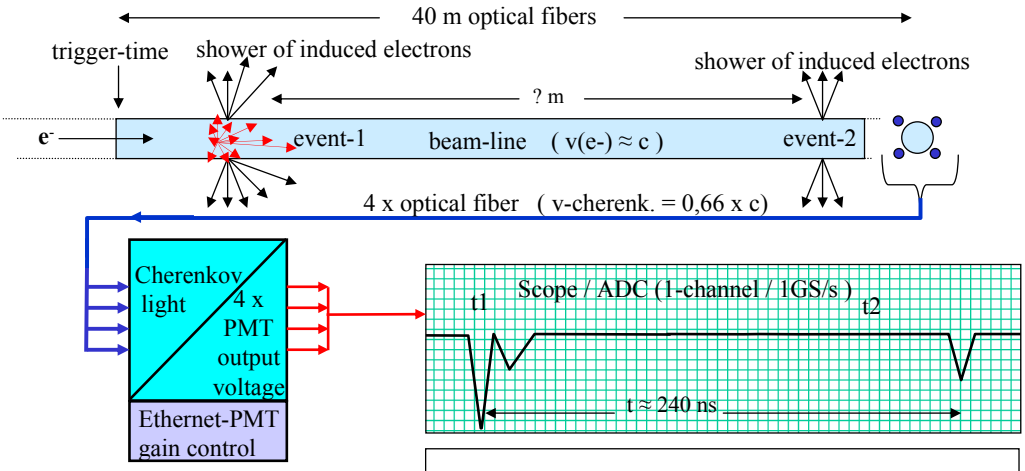


Zum Schutz des Undulators wurde ergänzend installiert das Beam Loss Position Measurement System der HMI-DESY Collaboration (Bestimmung von Strahlverlusten durch Messung des in Licht-Wellenleitern auftretenden Cherenkov Lichtes).

Beam Loss Monitoring with optical fibers

W.G. 2007
HMI-Berlin

Showers of electrons outside the beam line are generating Cherenkov light inside the optical fibers. Localization of these events by evaluating the run-time in the beam line / optical fiber, with respect to the system trigger .

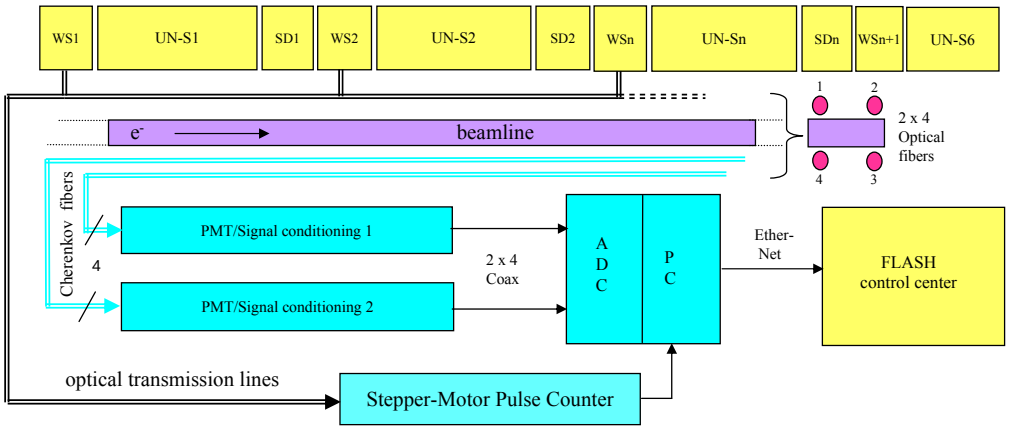


$v_{vac} \approx c$ (≈3ns/m)
$v_{cher} \approx 0{,}66 \times c$ (≈5ns/m)

measuring time = 320 ns (≡ 320 samples bei 1GS/s)
(results from transition time in the beamline and opt. fibers)

1 m ≡ 8 samples ≡ 8 ns
→ t2 – t1 = 240 ns event 2 - event 1 ≈ 30 m

Cherenkov beam profile measurement system

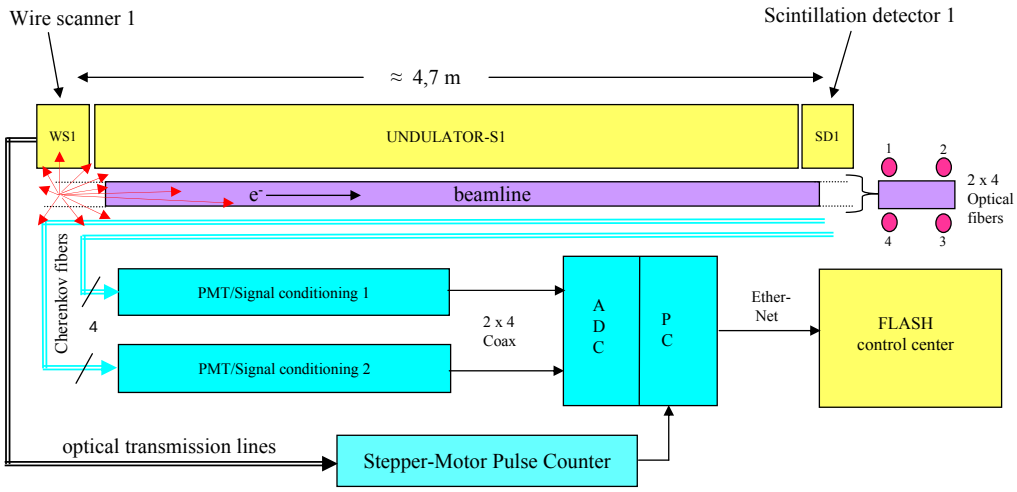


Nur wenige Hard-und Software Zusätze sind erforderlich, um das Cherenkov BLM System als Beam Profile Measurement System nutzen zu können : Ein Counter Modul zählt die Stepper Motor Pulse, als Positionsinformation. Die aufsummierten Strahlverluste jeder Messung werden der aktuellen Position des Scanners zugeordnet, und ergeben genau einen Punkt des Beam Profile Graphen.

In use are two Cherenkov setups synchronously. One is working with very high PMT gain, to measure the peripheral parts of the beam and beam tail. The second system is working with an appropiate lower sensitivity, to show the results in a continuous graph

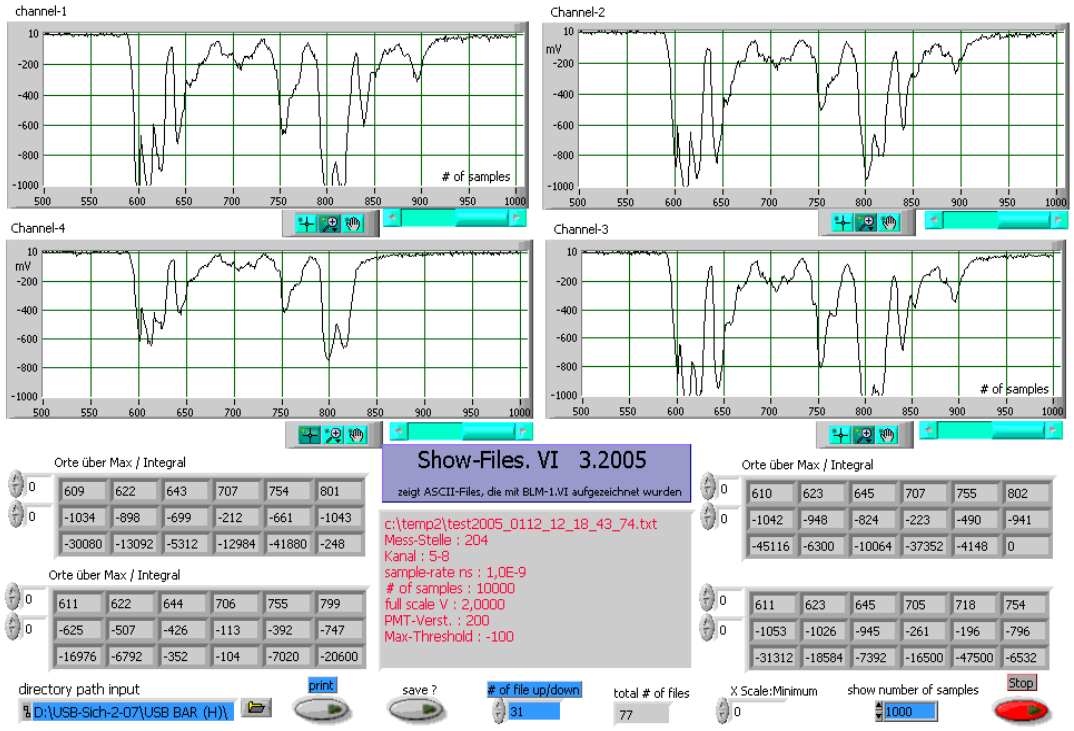
Hahn-Meitner-Institut Berlin in der Helmholtz-Gemeinschaft

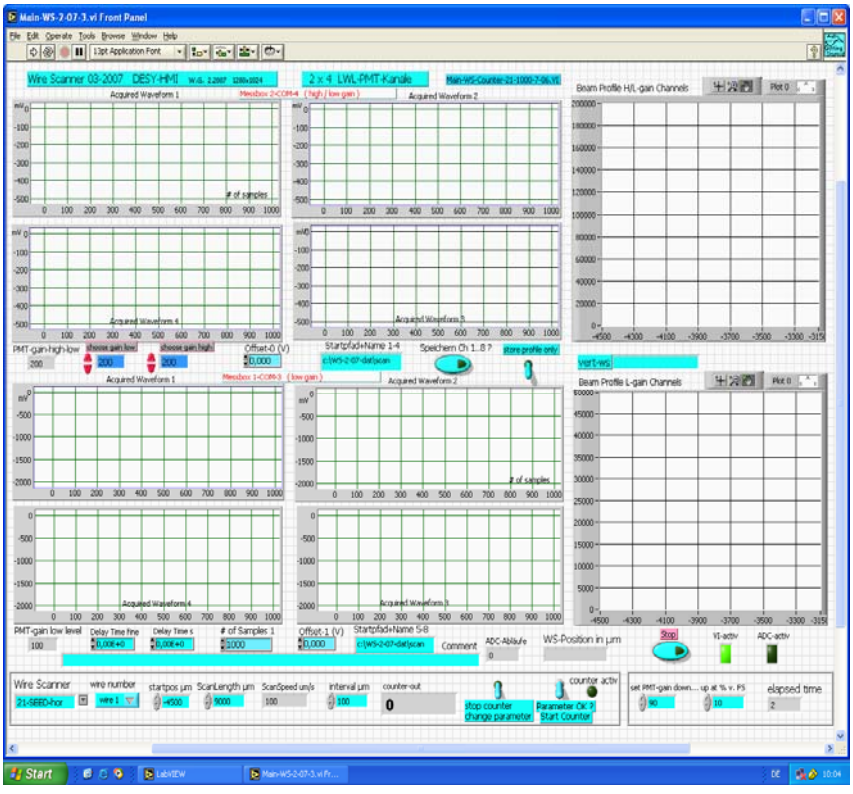
Diskreter Sensor (Scinti) vers. distributed Sensor (LWL)



The scintillation detector is positioned about 4,7 m downstream of the wire scanner, and is partially shielded by the undulator. As an advantage, the Cherenkov system is measuring loss showers at the entire length of the undulator segment.

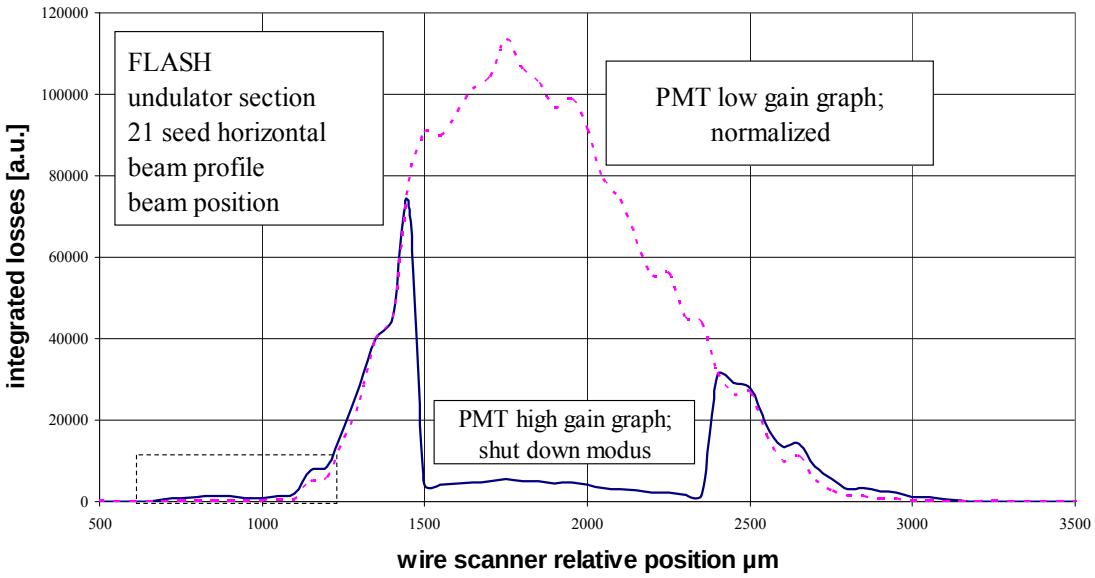
Beam losses detected with 4 optical fibers at the beamline





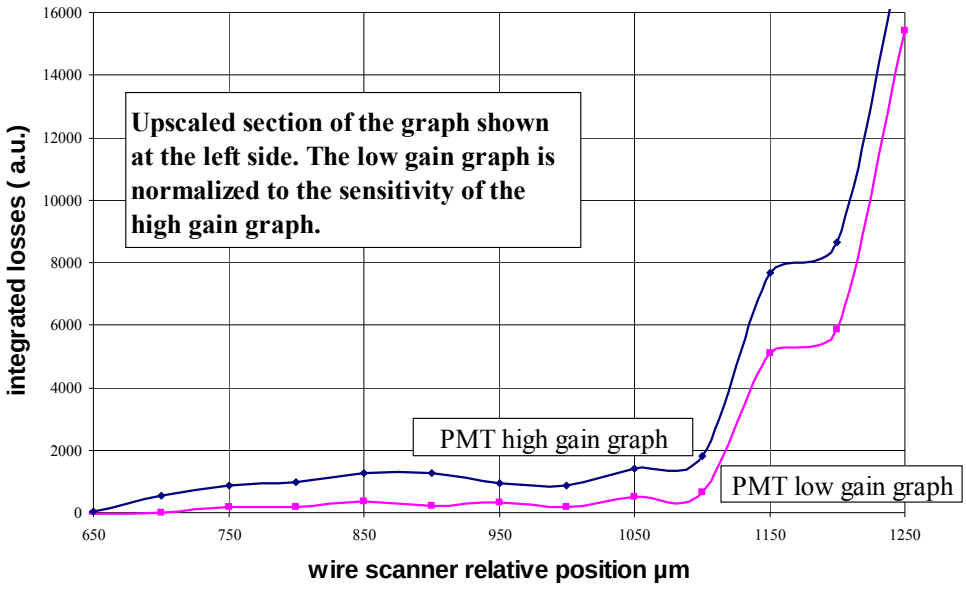
Cherenkov beam profile measurement display

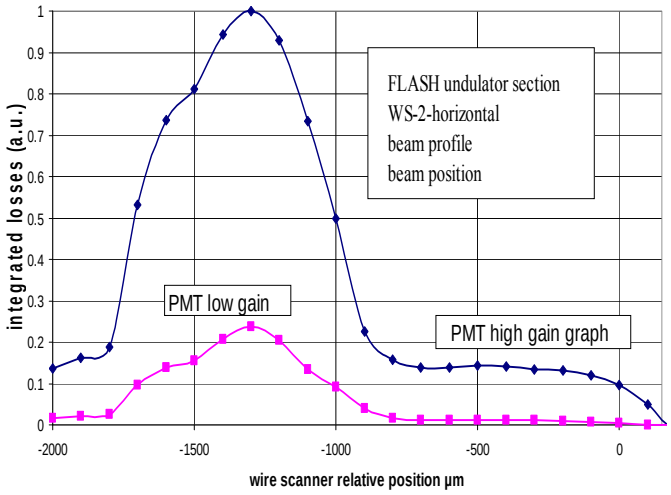
Look and feel of the operators control panel. At the right side is monitored the beam profile, while at the left side are displayed beam losses in the undulator section. The entire system is controlled by use of Lab View software.



In use are two Cherenkov setups synchronously. One is working with very high PMT gain, to measure the peripheral parts of the beam and beam tail. To protect the photomultipliers against outstanding high levels of Cherenkov radiation, PMT gain is reduced at prefixed levels of incoming light, and vice versa if Cerenkov light decreases. The second system is working with an appropiate lower sensitivity, to show the results in a continuous graph.

Hahn-Meitner-Institut Berlin in der Helmholtz-Gemeinschaft





Beam profile measurement showing beam core and beam tail region.

Light intensity is very small, thus the sensitivity of the high gain PMT is not reduced during the scan.

At predetermined positions of the wire scanner, all PMT output signals are measured (digitized with a sample rate of 1 ns) . Summing up these values leads to one point of the beam profile. Most interresting are the peripheral zones of the beam, and it is noticeable that the sensitivity of the entire system is sufficient, to show even the beam tail region.

Advantages of fiber sensors

- **beam loss measurement in narrow slits, which are inaccessible for conventional systems (shielded fibre diameter: 800 µm)**
- **measuring of particle losses in transverse and longitudinal dimensions**
- **online radiation detection downstream of collimation section**
- **recording of collimator “leaks” and online particle loss tuning**
- **simple sensitivity adjustment by switching photomultiplier gain**

Verbesserungsmöglichkeiten / Improvements

Für Messungen im beam tail Bereich und für HALO Messungen ist entscheidend eine hohe (Nach-)Verstärkung des PMT –Signals, bei sehr gutem S/N ratio. Die eigentliche PMT-Gain kann dann niedriger gehalten werden (weniger HV-Peaks der PMTs).

Die bisher verwendeten OP-HF-DC-Verstärker sind dafür nicht besonders gut geeignet.

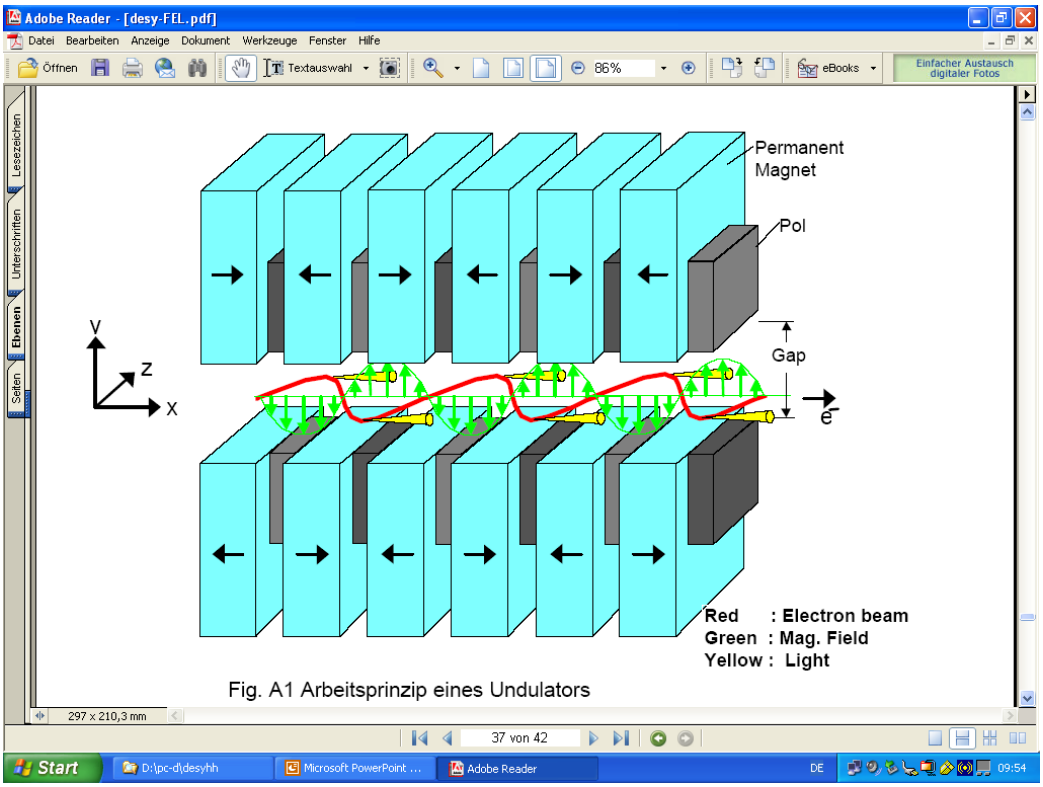
Deshalb wurden Verstärker (AC) mit diskreten Bauteilen entwickelt, die im Labor sehr gute Ergebnisse erzielen.

Die neuen Verstärker werden in Kürze in Hamburg getestet werden.

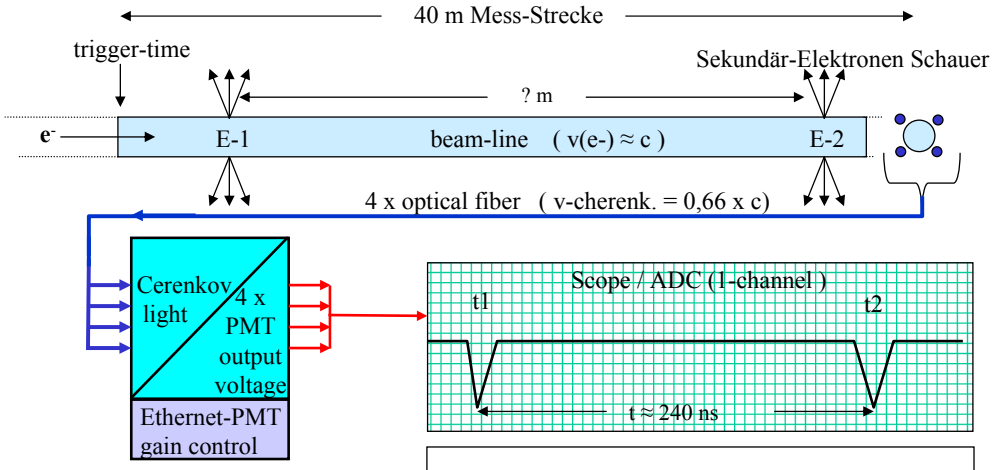
Auch wurde softwaremässig eine Umstellung der Anlage auf nur 4 LWL vorgenommen, was eine deutlich höhere Ablaufgeschwindigkeit zur Folge hat.

Eine Bündelung von 4 LWL so, dass alle Signale von einem PMT aufgenommen werden, würde eine weitere gravierende Verbesserung des Signal-Rauschverhältnisses bringen.

ENDE



Cherenkov-Mess-Systematik : Elektronenschauer ausserhalb des Strahlrohres erzeugen Cherenkov Strahlung in Lichtwellenleitern. Ort und Intensität der Ereignisse können bestimmt werden .



$v_{vac} \approx c$ (≈3ns/m)
$v_{cer} \approx 0{,}66 \times c$ (≈5ns/m)

Messzeit (40m) = max. Laufzeit – min. Laufzeit
= (40 x 3ns/m) + (40 x 5ns/m) – 0 ns
= 320 ns (= 320 samples bei 1GS/s)

⟹ 1 m Messstrecke ≡ 8 samples ≡ 8 ns
⟹ t2 – t1 = 240 ns ⟹ E-2 - E-1 ≈ 30 m

Entwicklungen mit dem GPX TDC ASIC im Forschungszentrum Jülich

H. Kleines[1], A. Ackens[1], P. Wüstner[1], W. Erven[1], F.-J. Kayser[1], M. Drochner[1], P. Kämmerling[1], G. Bertschinger[2]

[1]Zentralinstitut für Elektronik (ZEL), Forschungszentrum Jülich

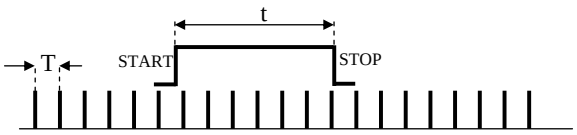
[2]Institut für Energieforschung /Plasmaphysik (IEF-4), Forschungszentrum Jülich

- *Zeitmessung mit TDCs*
- *TDC-Baugruppen für WASA*
- *MWPC-Auslese mit dem GPX*
- *uTCA-Baugruppe mit dem GPX*

Forschungszentrum Jülich
in der Helmholtz-Gemeinschaft

Zeitmessung

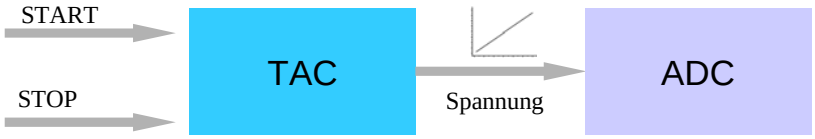
- *Binärer Zähler:*



  - *T= 1/f ⇒ Messfehler <= 2* T*

  ⇒ Hohe Anforderung an externe Clock

- *Time-to-Amplitude-Converter (TAC)*

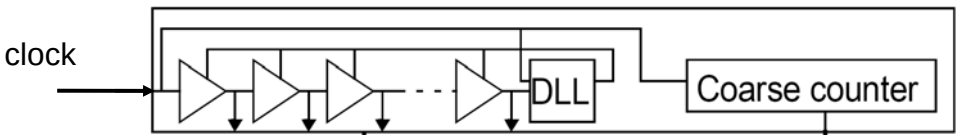


  - *Hohe Genauigkeit (10ps) und Linearität*
  - *Hohe Totzeit*
  - *Aufwendig*

Zentralinstitut für Elektronik

Digitale TDC ASICs

- *Interne Taktinterpolation durch Ringoszillator (PLL-stabilisiert) oder Delayline (DLL-stabilisiert)*

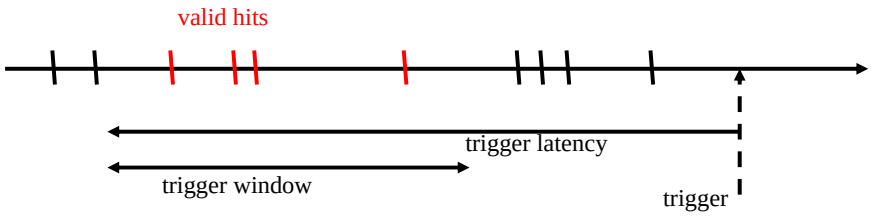


  - *Steuerung der Gatterlaufzeiten über die Corespannung*
  - *Kompensation von Temperatur und Produktionstoleranzen durch Selbstkalibrierung*
- *Wichtige Beispiele: F1 (acam), CERN HPTDC, AMS110 (ESRF), GPX (acam)*

Beispiele I

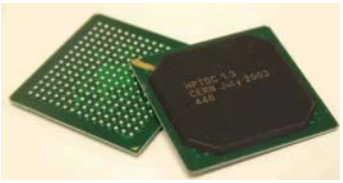
(PQFP160)

- *F1: entwickelt von Uni Freiburg + acam messelectronic für Compass (CERN)*
  - *mittlerweile obsolet*
  - *8 Kanäle (LVPECL) + gemeinsamer START/STOP*
  - *bin-Breite: 120ps (standard mode)*
  - *Messbereich: 16bit*
  - *4-fach multihit-fähig, Doppelpuls-Auflösung: 20ns*
  - *Trigger-Matching Unit:*

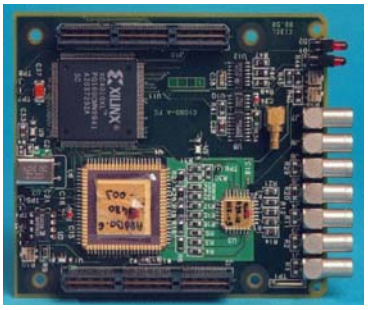


Beispiele II

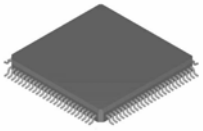
- *CERN HPTDC (mehrere Speed Grades)*
  - *Mehrere Modi: 32/8 Kanäle, Bin-Breite: 24ps, 98 - 781ps*
  - *Messbereich: 20 bit (high resol.)*
  - *Mulithitfähig*
  - *10ns Doppelpulsauflösung*
  - *Hitrate/ Kanal: 2/4/8 MHit/s*



- *ESRF AMS110(AMS111): optimiert für 2D delay-line Auslese*
  - *4 Kanäle + gemeinsamer Start/Stop*
  - *Pileup-Erkennung (2D-Mode)*
  - *Bin-Breite: 125 ps*
  - *Messbereich: 14 bit*
  - *40 MHit/s (100 MHit/s)*



GPX ASIC



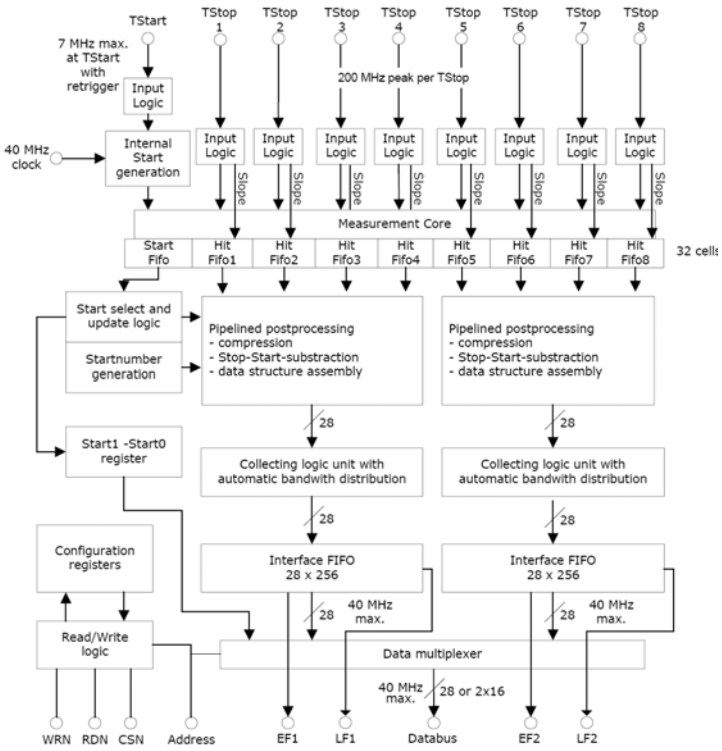
TQFP100

TFBGA120

- *2 Packages:*
- *Kern: PLL-stabilisierter Ring-Oszillator*
- *Mehrere Modi: Bin-Breite 10 ps bis 81 ps*
- *I-Mode:*
  - *Bin-Breite: typ. 81 ps*
  - *Messbereich: 17 Bit*
  - *32fach multihit-fähig, Doppelpulsauflösung ca. 5,5 ns*
  - *Peak-Rate: 182 MHit/s*
  - *Kontinuierliche Rate pro Kanal: 10 MHit/s*
  - *Kontinuierliche Rate über alle Kanäle: 40 MHit/s*
  - *Eingänge: LVTTL*
  - *Keine Trigger-Matching-Unit*

Struktur des GPX

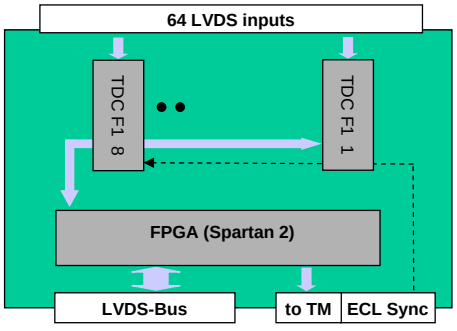
- *Transfer zum Hit Fifo: 182 MHz*
- *Transfer zum Interface FIFO: 40 MHz*
- *Datenbus: 40 MHz*
- *Intern retrigger-bar*



WASA TDC Baugruppe mit F1 ASIC

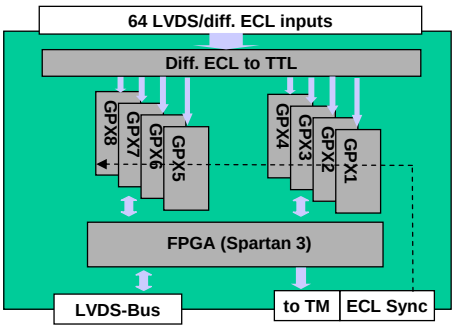
- *WASA: Detektor an COSY (ca. 1600 PMTs , 3700 straws)*
- *Bewährtes Design für TDC aus ANKE-Experiment in Kombination mit CMP16 preamp/discriminator*
- *„Privater“ Backplanebus (SCSI)*
- *8 F1-Chips (8 Kanäle) an einem Bus zum FPGA (=> 64 Kanäle)*
- *F1-Performance unzureichend für Zeitstempelung von Plastik-szintillatoren*

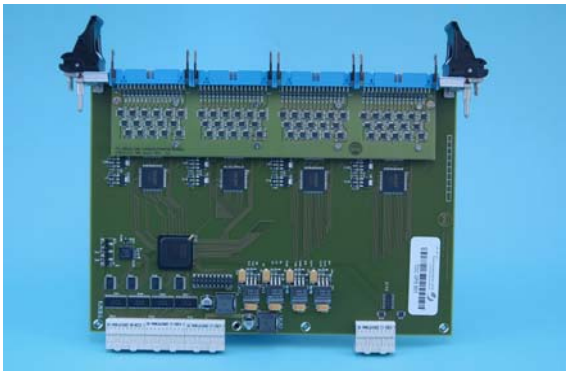
=> Neuentwicklung notwendig



WASA TDC Electronics with GPX ASIC

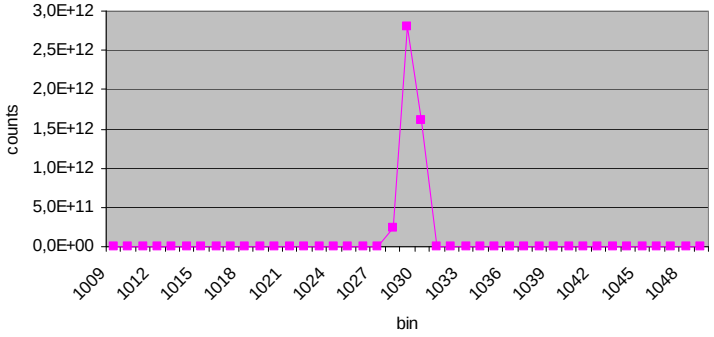
- *GPX (Nachfolger des F1 von acam messelectronic GmbH in I-Mode): 8 Kanäle, 81ps Bin-Breite, 10 MHit/s je Kanal*
- *Crate-weite Synchronisation über ECL-Signale auf der Backplane*
- *Kontroll-Signal für Frontend-Eletronik via Transition-Module*
- *Inputs LVTTL => Mezzanines für LVDS und diff. ECL*
- *Tigger-Matching im FPGA*



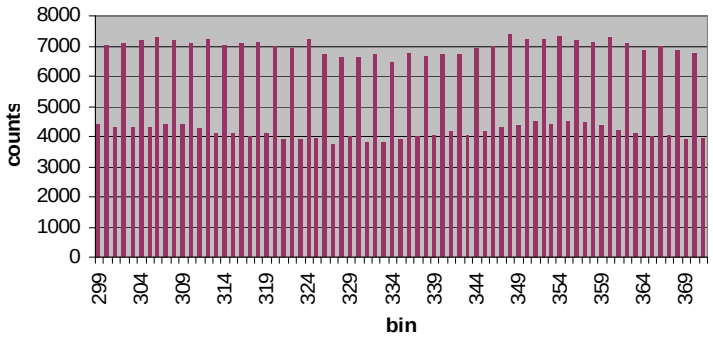


Messungen am WASA TDC (GPX)

- *Auflösung: 0,56 LSB (Standard-abweichung)*



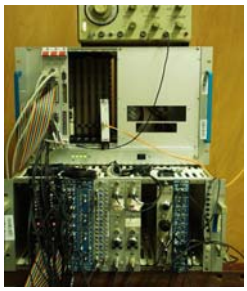
- *Differentielle Nicht-Linearität*

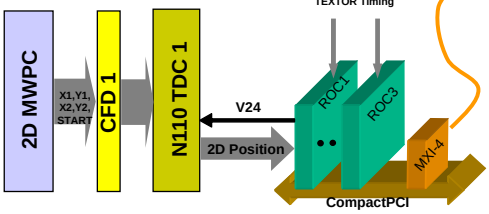


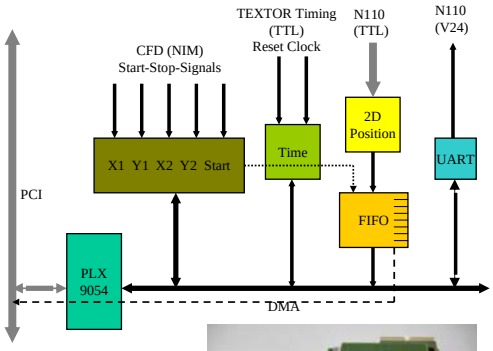
Forschungszentrum Jülich
in der Helmholtz-Gemeinschaft

Auslese von MWPC mit Delayline

- *Klassische Lösung des ZEL (für Neutronen + Gammas):* ***N110***
- ***N110****: NIM-Modul der ESRF mit AMS110 TDC ASIC (Pileup-Erkennung, Berechnung der 2D-Koordinaten)*



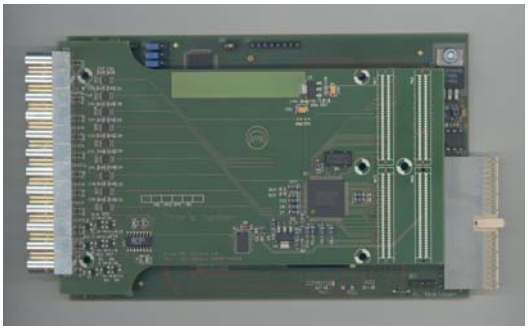


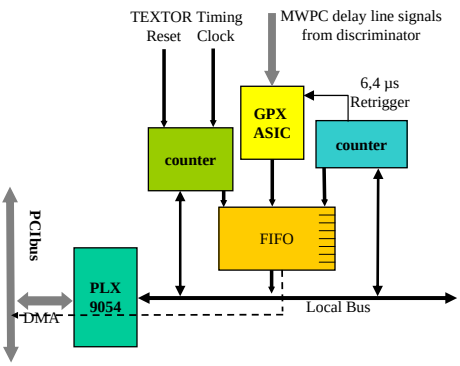


Forschungszentrum Jülich
in der Helmholtz-Gemeinschaft

MWPC-Auslese mit GPX

- *Mezzanine, usprünglich für SIS1100 geplant*
- *GPX Timing schwierig mit Spartan II => Jumiom Board*
- *Zeitstempelung aller MWPC Inputs + Timing System*
- *Rekonstruktion, Pileup-Erkennung, etc. im PC*
- *Erste erfolgreiche Tests im März*





Vorlaufentwicklung für PANDA

- *PANDA: Detektorsystem am HESR (FAIR)*
- *FZ Jülich verantwortlich für zwei Subdetektoren*
  - *Micro Vertex Detector (MVD): Pixeldetektor mit extremen Anforderungen an DAQ*
  - *Straw Tube Tracker(STT): Eine Option für den Central Tracker, ca. 10.000 Kanäle*
  - *STT: Messung der radialen Entfernung track-to-wire mit Diskriminatoren + TDCs*
    (in Diskussion: Messung der Z-Position durch Ladungsteilung)

Forschungszentrum Jülich
in der Helmholtz-Gemeinschaft

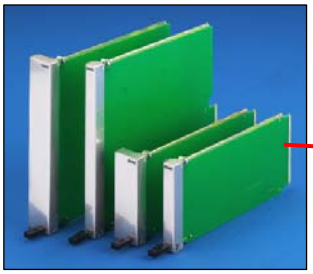
Pragmatischer Ansatz beim Design der PANDA DAQ Elektronik

- *Gewünschter Formfaktor für PANDA DAQ: ATCA*
  - *Teure Infrastruktur*
  - *Hohe Enwicklungskosten wegen Board-Größe, Management und Power Supplies*
  - *Offene Frage: Protokoll für die seriellen Links(PCIe, GbE,..)*
  - *Offene Frage: Timing System*
  - *Offene Frage: Rear IO*
- *Zwischenlösung: AMC-Module*
- *Laborsysteme: uTCA*

Zentralinstitut für Elektronik

AdvancedMC™ Advanced Mezzanine Cards

*Source: Ganninger + Lenkisch
Schroff®



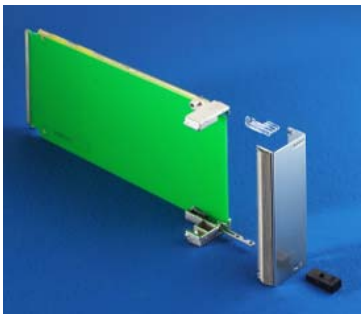
AMC form factors

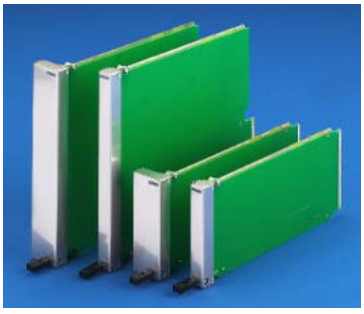
Carrier Blade

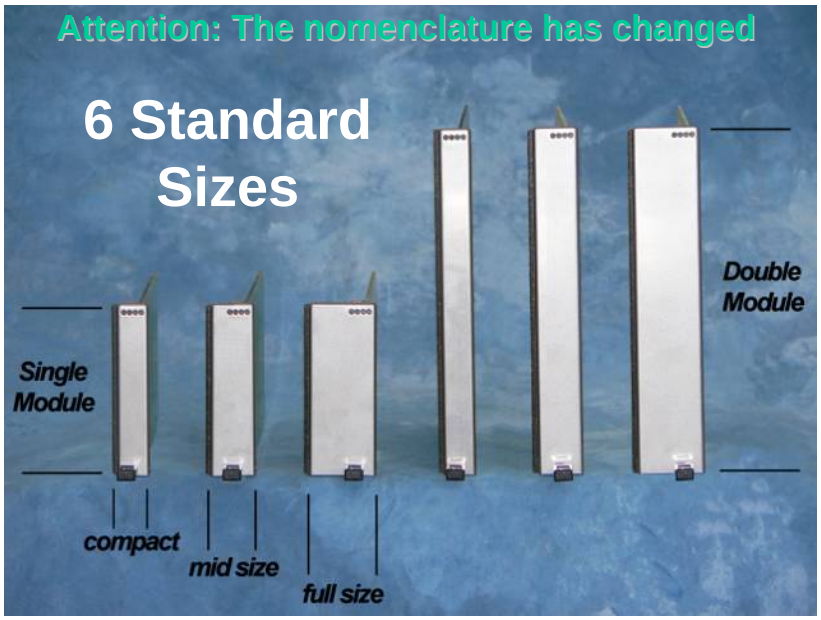
Carrier blades in ATCA chassis

PICMG® AMC.0 Specification
PICMG® AMC.0 R2.0 RC1.1

*Source: Ganninger + Lenkisch
Schroff®





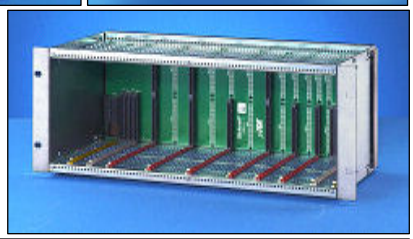


µTCA™ System components

*Source: Ganninger + Lenkisch

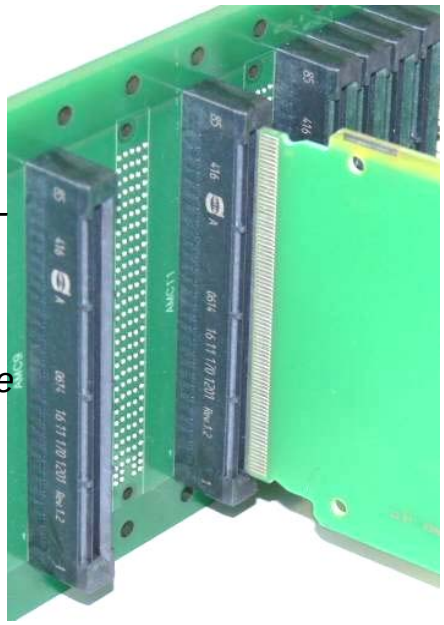
Schroff®

- ***AdvancedMC Modules***
- ***Subrack, chassis and accessories***
- ***Optional Cooling System***
- ***Power Supply***
- ***Backplane***
- ***Optional MicroTCA Carrier Hub (MCH)***



Backplane

- *kein paralleler Bus*
- *Datentransport basiert auf "Switch Fabrics"*
- *„Common Options": 1000BASE-BX und SATA vorgeschrieben*
- *„Fat Pipe": PCIe oder SRIO oder 10GBASE-BX4 (4 Lane)*
- *Backplane unabhängig von „Switch Fabric"*
- *Physical Layer: 100 Ohm differentielle Paare*
- *Stecker: 170pol, direkte Steckung*
- *Komponenten-Seite 1 ist links!!*



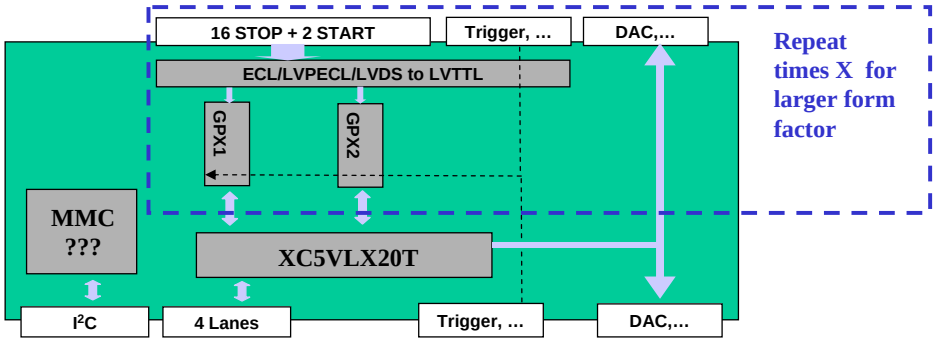
*Source: Ganninger + Lenkisch

Schroff®

Design-Entscheidungen für den PANDA uTCA TDC

- *Format: Compact + Single => Maximale Kanalzahl im AMC carrier*
- *Verfügbarer Platz: 2 GPX ASICs => 16 Kanäle*
- *Serielle Kommunikation auf Backplane: PCIe*
- *FPGA: Xilinx Virtex5-LXT Familie => Identische HW für PCIe und GbE*
- *Wähle den preiswertesten Virtex5-LXT:* **XC5VLX20T**
- *PANDA Timing System noch nicht definiert => Signale CLOCK und START als ECL/LVPECL/LVDS Eingänge von der Vorderseite. (zusätzlich als LVDS von der backplane)*
- *Zusätzliche LVDS Signale für Frontend (DACs, etc.)*
- *Eingänge wählbar als diff. ECL/LVPECL/LVDS (durch Bestückungsoptionen)*

Architektur des PANDA uTCA TDC



- *Module Management Controller (MMC)*
  - *von Pidgeon Point???*
  - *Alternative: MicroBlaze + I²C in Virtex 5 + SW*
- *GbE oder PCIe: FPGA Code*
- *"Lern-Projekt" for die uTCA Technologie*

Cell Processor: Introduction and possible use areas

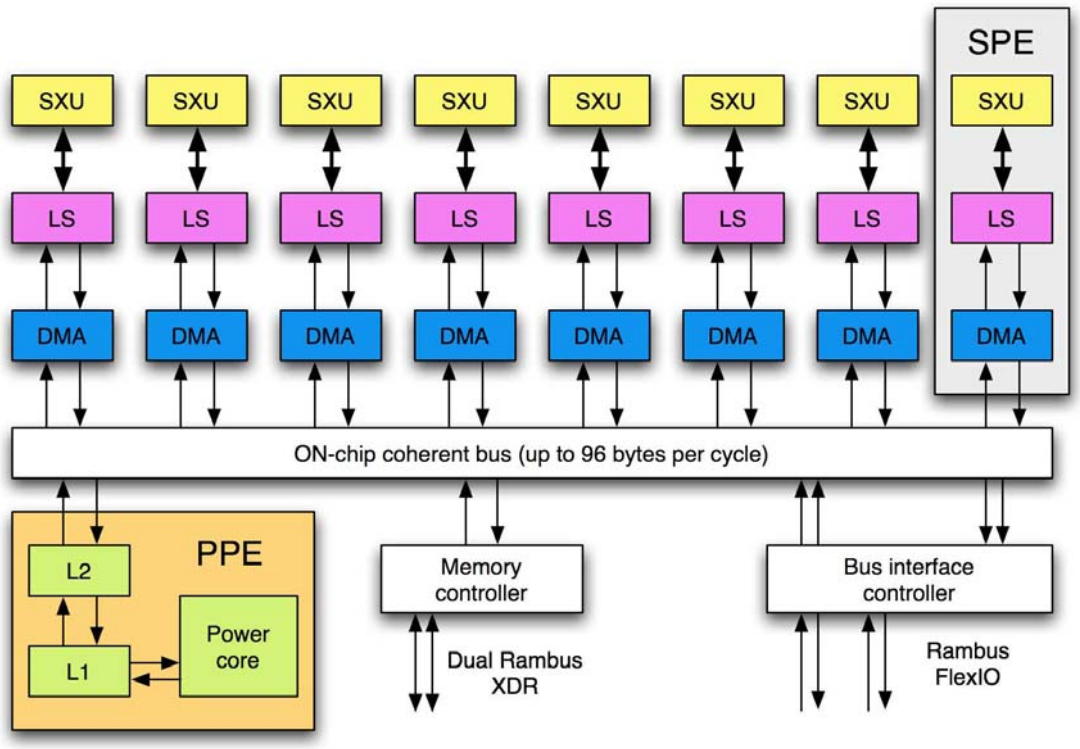
Matthias Drochner

FZ Jülich / ZEL

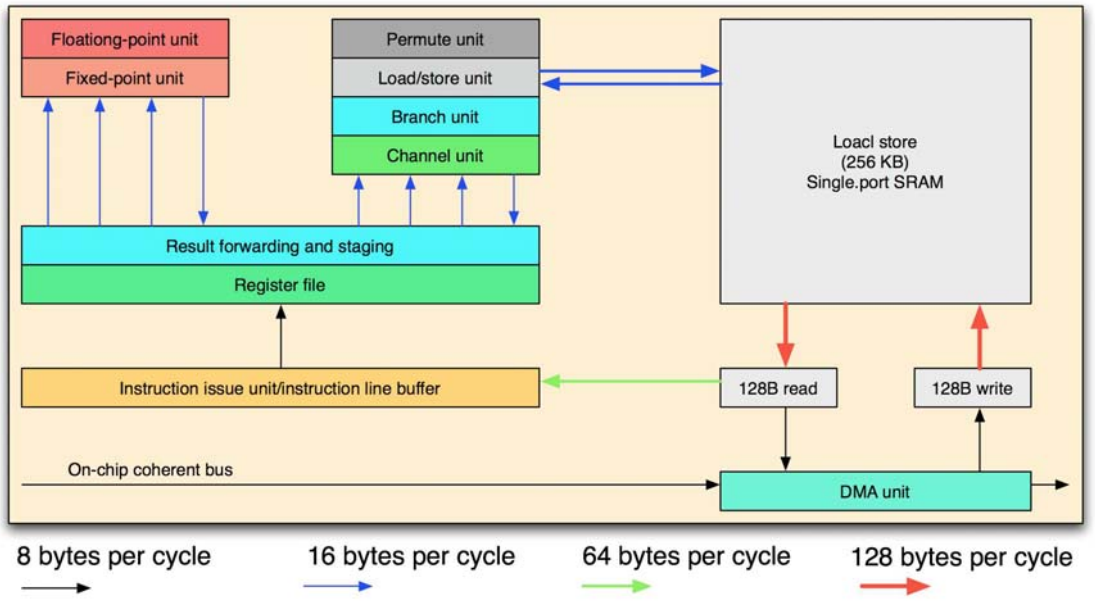
Contents

- Introduction to CBE and other acronyms
- Host bridge chips, use in existing machines
- IBM's roadmap
- The QPACE project
- Useful for particle physics?

Cell schema



Cell SPU schema



What makes it different

- Heterogenous architecture
- While the PPCs are already fast, the real computing power comes from non-general-purpose processors.
- High FLOP / W
- No IEEE rounding for SP, slow DP (in initial version)
- SPUs only 256K local store (seems architected), non-coherent connection to main memory
- Wide registers (128 bit) — SIMD optimisation for optimal performance
- Ring interconnect — needs planning for optimal throughput

Machines

- PS3 (even clustering attempts, but no low-latency interconnect)
- IBM BladeCenter: QS20+, also in FZJ (Infiniband interconnect)
- Mercury accelerator board (in ZEL), appears to be closed system
- Mercury 1U dual Cell
- Roadrunner (Coprocessor for Opteron)

Host bridge chips

- Generally: hard to get information, data here might be inaccurate
- Toshiba chip
  - Used in PS3 and QS20
  - Some common periphery interfaces
  - PCI / PCIexpress
  - Security features (e.g. harddisk encryption), therefore extra IDE chip in QS20
- AXON
  - Used in Mercury accelerator board and QS2x
  - DDR2 RAM controller
  - PCIexpress
  - Reports about poor DMA performance (might be interoperability problems with PC chipsets)

QPACE

- SFB of Uni Regensburg and Wuppertal, with IBM, FZJ and DESY
- Goal: build fastest / cheapest machine for QCD
- Successor of QCDOC and APEnext
- 2048 nodes, shared between Jülich and Wupppertal
- 3-dimensional network, low latency, and some special global signals

Use for particle physics? General Usefulne

- For pure event building, there are cheaper / more suitable solutions. (memory bandwidth likely limiting)
- Interesting: track reconstruction involving FP math.
- But: problem should fit into 256k, or modularized into pieces of that size.
- 8 SPUs per Cell, 2 Cells can be coupled coherently, more needs special periphery.
- Either the problem makes good use of max. 2 Cells, so that eg. a trigger latency is achieved which is impossible with PCs, or the problem is so big that power consumption is an issue.
- Competes with FPGA for integer / static problems.

Use for particle physics? Practical issues

- No general availability of Cell processors yet.
- Board construction needs significant support from IBM (or Toshiba / Sony), there must be some value to get them interested.
- QPACE board might be a starting point
  - Board will be built by IBM, don’t know about their plans.
  - Firmware could be adapted to other needs.
  - 10GE compatibility (possible, but not ensured now)
  - Water cooling needs infrastructure.