

S^E
S_I Studiengruppe für
Elektronische Instrumentierung
der Helmholtz-Zentren

103. Tagung der Studiengruppe elektronische Instrumentierung im Frühjahr 2012

in Dresden vom 12. März -14. März 2012

am



HZDR

Helmholtz-Zentrum Dresden Rossendorf



Editor: Peter Göttlicher (DESY)

Verlag Deutsches Elektronen-Synchrotron

Impressum

103. Tagung der Studiengruppe elektronische Instrumentierung im Frühjahr 2012 12.-14. März 2012, Dresden, Deutschland

Conference Homepage

<http://indico.desy.de/conferenceDisplay.py?confId=5248> oder

https://indico.desy.de//event/SEI_fruehjahr.2012

Online Proceedings auf

<http://www-library.desy.de/confprocs.html>

The copyright is governed by the Creative Commons agreement, which allows for free use and distribution of the articles for non-commercial activity, as long as the title, the authors' names and the place of the original are referenced.

Editor:

Peter Göttlicher

Juli 2012

DESY-PROC-2012-01

ISBN 978-3-935702-65-2

ISSN 1435-8077

Published by

Verlag Deutsches Elektronen-Synchrotron

Notkestraße 85

22607 Hamburg

Germany

Printed by

Kopierzentrale Deutsches Elektronen-Synchrotron

103. Tagung der Studiengruppe elektronische Instrumentierung im Frühjahr 2012

SEI - Studiengruppe elektronische Instrumentierung
der Helmholtz-Zentren
Dresden (HZDR), 12. März - 14. März 2012

Inhaltsverzeichnis

Allgemeines und Zusammenfassendes

Eröffnung und Ausblick	P. Göttlicher	3
Bild der Teilnehmer	HZDR, PR-Abt.	4
Tagungsprogramm		5
Notizen aus der Tagung	D. Notz	8

Vorträge

Ein Framework zur hochperformanten Verarbeitung von Datenströmen mit digitaler Hardware	Prof. B. Lang	10
FPGA framework development for modular high-speed DAQ system	Q. Xia	23
Einführung des Kontrollsystems TANGO am Jülich Centre for Neutron Science	H. Kleines	37
EtherCAT Feldbusknoten: Entwicklung, Systemunterstützung und -kompatibilität	P. Kaever	47
Simplify your LabVIEW-EPICS communication	C. Winkler, J. Kuszynski	54
Customized Off-The-Shelf Technologies for building system platforms for Big Physics experiments	L. Johansson	62
Modulare Ethernet basierte DAQ- und MSR-Systeme	S. Hering V. Thomas	77

IPMI und MicroTCA Module Management - Aktuelle Entwicklungsarbeiten	M. Drochner	93
Model Driven FPGA Design – An approach	M. Penno	107
KIT Kamera System für UFO	M. Balzer	121
Überblick über die Entwicklungen der Datenaufnahme- und Verarbeitungssysteme für Photon Beamlines und Experimente am European XFEL	P. Gessler	133
Developed High Performance Electronics for the Gas Filled Stopping Cell and MR-Time-of-Flight-MS of the FRS Ion Catcher at GSI	S. Ayet san Andres	145
Nachweis von Mikrowellenemissionen aus hochenergetischen kosmischen Luftschauern	O. Krömer	155
Der Modulator Teststand Überblick, Ergebnisse und Ausblick	M. Grimberg	168

Eröffnung

Zu der jährlichen Tagung, die allen Interessierten an Elektronik in der Forschung offensteht, kamen 66 Teilnehmer und -innen. Sie sind von verschiedenen Forschungseinrichtungen, den Helmholtz-Zentren – DESY, FZJ, GSI, HZB, HZDR und KIT –, anderen Forschungseinrichtungen – XFEL, Universitäten und Hochschulen – und der Wirtschaft angereist.

Die Vorträge und Ausstellungen regten zu interessanten Diskussion zwischen den Teilnehmern an, so mal ähnliche Arbeitsmethoden an den verschiedenen Einrichtungen benutzt werden. Die abgedeckten Themen lassen sich zu den Schwerpunkten

- FPGA's und schnelle Datennahme,
- Messen-Steuern-Regeln und damit verbundene langsame Datennahme und Benutzeroberflächen und
- Systeme und Detektoren

zusammenfassen. Sie sind mit ähnlichen Gewichten vertreten gewesen.

Ein weiterer Inhalt wurde durch das lokale Labor HZDR gestaltet. So stellte Prof. Helm das Forschungszentrum als Vortrag vor, und wir konnten das Hochfeld-Magnetlabor auf dem Gelände besichtigen.

Auf einer Exkursion lernten wir auch die Gläserne Manufaktur kennen, wo eine moderne Fertigungsstraße so gestaltet ist, dass Besucher die technischen Tricks und Arbeitsabläufe der Automontage bei der realen Fertigung beobachten können.

Das Tagungsprogramm ist auf dem Internet einzusehen:
<http://indico.desy.de/conferenceDisplay.py?confId=5248> oder
https://indico.desy.de//event/SEI_fruehjahr_2012

Die Homepage der Studiengruppe ist auf <http://sei.desy.de/> zu finden.

Ausblick

Die nächste Tagung wird für das Frühjahr 2013 in Jülich geplant.



Teilnehmer der Tagung SEI_2012

Quelle: HZDR, PR-Abteilung

Tagungsprogramm:

SEI-Tagung am HZDR - Frühjahr 2012

Studiengruppe elektronische Instrumentierung der Helmholtz-Zentren

Monday 12 March 2012

Vortraege 1 - Haus 114 / Raum 201 (14:00-16:00)

time	title	presenter
14:00	HZDR - Helmholtz-Zentrum Dresden-Rossendorf	Prof. HELM
14:30	Eroeffnung	Dr. GOETTLICHER, peter (DESY)
14:45	Lokale Informationen	Dr. KAEVER, Peter (HZDR)
15:00	Ein Framework zur hochperformanten Verarbeitung von Datenströmen mit digitaler Hardware	Prof. LANG, Bernhard (Hochschule Osnabrück)
15:30	FPGA framework development for modular high-speed DAQ system	Ms. XIA, Qingqing (DESY)

Vortraege 2 - Haus 114 / Raum 201 (16:30-18:00)

time	title	presenter
16:30	Einführung des Kontrollsystems TANGO am Jülich Centre for Neutron Science	Mr. KLEINES, Harald (Forschungszentrum Jülich GmbH)
17:00	EtherCAT Feldbusknoten: Entwicklung, Systemunterstützung und -kompatibilität	Dr. KAEVER, Peter (HZDR)
17:30	Simplify your LabVIEW-EPICS communication	Mr. WINKLER, Carsten (Helmholtz-Zentrum Berlin) Mr. KUSZYNSKI, Jens (Helmholtz-Zentrum Berlin)

Busfahrt zum Abendessen, Linie 261 - ---- (18:13-18:20)

time	title	presenter
18:13	Busfahrt zum Abendessen, Linie 261	

Busfahrt zum Hotel: Linie 261 spaetestens 21:17 - ---- (21:17-21:52)

time	title	presenter
21:17	Busfahrt zum Hotel: Linie 261, spaetestens 21:17	

Tuesday 13 March 2012

Vortrage 3 - Haus 114 / Raum 201 (09:00-10:30)

time	title	presenter
09:00	Customized Off-The-Shelf Technologies for building system platforms for Big Physics experiments	Mr. JOHANSSON, Leif (National Instruments)
09:30	Modulare Ethernet basierte DAQ- und MSR-Systeme	Mr. HERING, Stephan (systema computer GmbH) Mr. THOMAS, Volker (systema computer GmbH)
10:00	IPMI und MicroTCA Module Management - Aktuelle Entwicklungsarbeiten	Mr. DROCHNER, Matthias (FZJ)

Firmenausstellungen - Haus 114 / Raum 201 (10:30-13:30)

Parallel pro Firma 1Tisch+Strom , Kaffee

time	title	presenter
10:30	Beckhoff - New Automation Technology	Mr. BURANDT, Nils (Beckhoff Automation GmbH)
10:31	Agilent SA Acqiris Operation	SPELTHANN, Hans Dieter (Agilent SA Acqiris Operation)
10:32	Ausstellung der iseg Spezial- elektronik	DONIX, Maik (iseg Spezialelektronik GmbH)
10:33	Ausstellung computer- gestützter Messtechnik National Instruments	Mr. YI, David (National Instruments Germany GmbH)
10:34	powerBridge Computer: MTCA.4 Solutions	Mr. HOLZAPFEL, Thomas (powerBridge Computer)
10:35	Struck Innovative Systeme GmbH	Dr. MATTHIAS, Kirsch (Struck Innovative Systeme GmbH)
10:36	systema computer GmbH: Modulare DAQ Systeme, Netzwerk- technik und Industrie-Computer	Mr. HERING, Stephan (systema computer GmbH)
10:37	WIENER Plein & Baus GmbH: Produkt- ausstellung	Mr. KOESTER, Andreas (WIENER Plein + Baus GmbH)

Führung durch das Magnet-Hochstrom-Labor - Haus 114 / Raum 201 (13:30-14:30)

time	title	presenter
13:30	...	

Exkursion: Gläserne Manufaktur: Link: <http://www.glaesernemanufaktur.de/> - to be announced (14:45-18:30)

time	title	presenter
14:45	Anfahrt	
15:45	Führung gläserne Manufaktur	
17:30	Gang/ Fahrt zum Abendessen	

Wednesday 14 March 2012

Vortraege 4 - Haus 114 / Raum 201 (09:00-11:00)

time	title	presenter
09:00	Model Driven FPGA Design – An approach	Mr. PENNO, Marek (DESY)
09:30	KIT Kamera System für UFO	Mr. BALZER, Matthias (KIT)
10:00	Überblick über die Entwicklungen der Datenaufnahme- und Verarbeitungssysteme für Photon Beamlines und Experimente am European XFEL	Mr. GESSLER, Patrick (European XFEL GmbH)
10:30	Developed High Performance Electronics for the Gas Filled Stopping Cell and MR-Time-of-Flight-MS of the FRS Ion Catcher at GSI	Mr. AYET SAN ANDRES, Samuel (GSI)

Vortraege 5 - Haus 114 / Raum 201 (11:30-13:00)

time	title	presenter
11:30	Nachweis von Mikrowellenemissionen aus hochenergetischen kosmischen Luftschauern	Dr. KRÖMER, Oliver (Karlsruher Institut für Technologie (KIT))
12:00	Der Modulator Teststand – Überblick, Ergebnisse und Ausblick	Mr. GRIMBERG, Mirko (DESY)
12:30	Abschluss der Tagung	Dr. GOETTLICHER, peter (DESY)

D. Notz
20. 3. 2012

Betr: Gesprächsnotizen über die Frühjahrstagung **der Studiengruppe Elektronische Instrumentierung** vom 12. bis 14. 3. 2012 beim HZDR (Rossendorf) (103. SEI Tagung)

Für DESY von Interesse

Es gibt Überlegungen, einen μ TCA Workshop (bei DESY?) zu veranstalten.
 Es wurden mehrere Tools zur Programmierung von FPGAs vorgestellt.
 DSL = Domain Specific Language; modellgetriebene Modelle. D.R.Y: Don't repeat yourself. Wiederholungen und Redundanzen reduzieren. Eclipse Modelling Framework (EMF) basiert auf Java oder XML (www s. u.). Board related modules, application related frameworks. Simulink für Algorithmen. Das Einarbeiten in die verschiedenen Systeme ist sehr zeitintensiv.
 H. Kleines (FZJ) gab einen Übersichtsvortrag über verschiedene Kontrollsysteme. Das gegenwärtige Kontrollsystem für die Neutronenexperimente ist TACO. Übergang zu TANGO. Es herrscht große Vielfalt: Client/Server Systeme: ILL: MAD, NOMAD (CORBA); NIST: ICP; SNS: pyDAS mit Python; HFIR LabVIEW; HZB Berlin: CARESS (CORBA).
 SCADA Systeme (Supervisory Control and DAQ) : Wonderware InTouch, Siemens WinCC.
 DCS (Distributed Control Systems): Siemens PCS7, EPICS.
 CORBA (Common Object Request Broker Architecture): Von OMG standardisiert. IDL (Interface Definition Language) basiert auf C++ und Java.
 Industrielle Kontrollsysteme basierend auf Windows ActiveX, .Net, VB-Skripting.
 Kontrollsysteme in der Forschung: PVSSII, EPICS, Yokogawa; DESY: TINE, DOOCS, TANGO, EPICS; CERN: FESA, UNICOS, JCOP; FAIR: FESA; ITER TANGO vs. EPICS, das DOE hat sich mit EPICS durchgesetzt. HZB (Berlin) benutzt LabVIEW.
 Real time ist leider noch nicht unterstützt.
 National Instruments (NI) hat enge Zusammenarbeit mit XILINX, TI (Texas Instr) und Analog Devices. Zusammenarbeit mit CERN im Rahmen von White Rabbit mit 10 km Abständen, mehr als 2000 Knoten und Zeitgenauigkeit von < 1ns Skew und 100 ps Jitter. Arbeiten an Multi-Core Prozessoren.
 Systerra beschrieb ihre Zusammenarbeit mit United Electronic Industries (UEI). Module sind klein. System basiert auf Simulink. *Bei Kontrolllampen werden die Einschaltstromkurven gemessen, um rechtzeitig zu sehen, ob Birnen ausgewechselt werden müssen.* Benutzung in Flugsimulatoren. Hohe Verfügbarkeit. Plattformen Linux und VxWorks. Es gibt Quellcodes zu jedem Modul. 10 Jahre Nachkaufgarantie. Alterung von Elektrolytkondensatoren. Geräte nachkalibrieren.

Sonstiges

Rosendorf (HZDR) benutzt EtherCAT mit Feldbusknoten von Siemens und Beckhoff. Die ultra fast imaging Camera (UFO) beim KIT braucht viel Rechenpower: 2*6 Core CPU, 6 GPUs GTX580, 96 GB Memory, XFS File System. Der CMOSIS Sensor CMV2000 liefert bis zu 340 Frames/s. Verlustfreie Kompression.

FZJ baut eigenen Management Controller für μ TCA zum Steuern von Netzteil oder Lüfter. Benutzter Chip: PIC32MX460. Kann mehr als benötigt, ist aber preiswert. Schnelles I/O und auch JPAG. Bei den Backplanes für μ TCA gibt es solche mit IP Adresse 192.168.16.17 oder 0.0.0.0. Empfohlen wird ersteres.

Zur Untersuchung von Luftschauern für das KIT Experiment Crome (Cosmic Ray Observation Microwave Emission) gibt es neue Radioempfänger. Einige Empfänger benutzen logarithmisch-periodische Dipolantennen. Zum Überprüfen der Anlage dient ein Mikrowellensender an einem RadioCopter Hubschrauber. Trigger kommt vom Kaskade Experiment.

WWW Informationen:

<http://zennotz.desy.de/SEI>

<http://www.ueidaq.com/download> für United Electronic Industries Software zum Spielen.

http://de.wikipedia.org/wiki/Modellgetriebene_Softwareentwicklung

<http://www.emftext.org/index.php/EMFText>

<http://www.eclipse.org/modeling/emf/>

<http://www.eclipse.org>

<http://www.acceleo.org/pages/home/en>

<http://www.omg.org>

<http://www.omg.org/spec/MOFM2T/1.0/>

Termine:

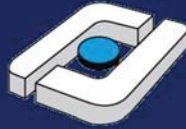
9. – 15. 6. 2012 RT2012 Berkely

27.10. – 3. 11. 2012 IEEE NSSS Anaheim, CA, USA

Nächste Treffen der Studiengruppe:

In Jülich, FZJ, ~März 2013

Stored in notz/ESONE/SEI/MINUTES/hzdr2012.doc (+.pdf)



Hochschule Osnabrück

University of Applied Sciences

Prof. Dr. Bernhard Lang
Dipl.-Inf. (FH) Rainer Höckmann

Ein Framework zur hoch-
performanten Verarbeitung
von Datenströmen mit digitaler
Hardware

1



Hochschule Osnabrück
University of Applied Sciences

Wo kommen wir her?

HS Osnabrück, University of Applied Sciences

Fakultät Agrarwissenschaften und Landschaftsarchitektur

Fakultät Wirtschaft und Sozialwissenschaften

Fakultät Ingenieurwissenschaften und Informatik

...
Laborbereich Elektronik/Kommunikationstechnik

Labor für Digital- und Mikroprozessortechnik:

Prof. W. Gehrke: Mikrorechnertechnik und Digitale Systeme

Prof. B. Lang: Digitale Multimediasysteme

Prof. M. Weinhardt: Hard- und Softwaresysteme zur Informationsverarb.

Dipl.-Ing. (FH) H. Hechler, Dipl.-Inf. (FH) R. Höckmann

Labor für Mikro- und Optoelektronik

Labor für Nachrichtenübertragung und Kommunikationsnetze

Labor für Hochfrequenztechnik und Mobilkommunikation

Labor für Analogelektronik

...
Fakultät Management, Kultur und Technik

2 Fakultät Ingenieurwissenschaften und Informatik



Womit beschäftigen wir uns?

- Programmierung
 - Grundlagen, Objektorientiert
 - Embedded Systeme, Treiberprogrammierung
- Digitale Systeme und Rechnerarchitektur
 - Programmierbare Logik (CPLDs, FPGAs)
 - Hardwarebeschreibungssprachen (VHDL)
 - Mikrocontroller, Soft-Prozessoren
- Bild- und Videoverarbeitung
 - Algorithmen
 - Software-Modellierung
 - Hardware-Realisierung



Inhalt

- Motivation
- Erste Beispiele
- Eigenschaften des Frameworks
- Elementare Datenfluss-Komponenten
- Höherwertige Protokolle
- Applikationen



Motivation

In vielen technischen Anwendungen fallen Massendaten als Datenströme an, die schnell erfasst und verarbeitet werden müssen, z.B.:

- Bildverarbeitung
- Telekommunikation
- Schnelle Sensoren, Detektoren

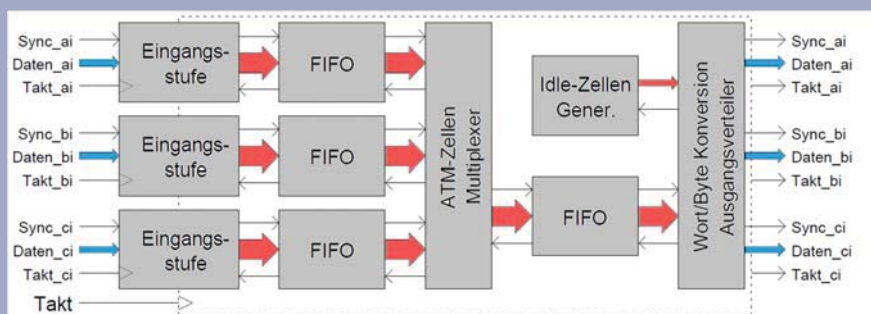
Ziele bei der Erfassung und Verarbeitung dieser Daten sind beispielsweise:

- Reduktion der Datenmenge durch Extraktion interessierender Information
- Lenkung von Datenströmen
- Speicherung von Daten-Bursts für die spätere Offline-Verarbeitung



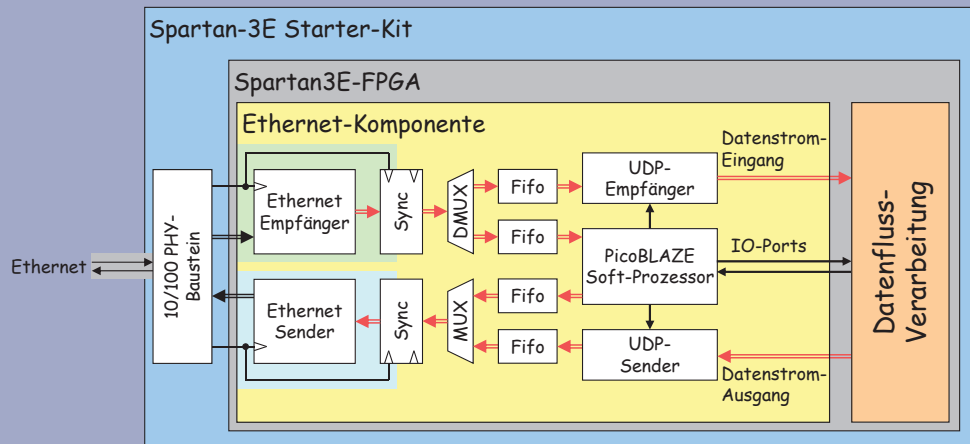
Erste Beispiele

Einfacher ATM-Switch (um 1993)



(Vorstudie für einen komplexen ASIC-Baustein)

Ethernet MAC



Hierarchisch aufgebaut aus kleineren Einheiten, „Multi-Clock“-Design
Multiplexen, Demultiplexen, Speichern und Verarbeiten der Datenströme

7 Fakultät Ingenieurwissenschaften und Informatik

Eigenschaften des Frameworks

Vorstellung eines Framework für die Verarbeitung **synchroner, digitaler** Datenströme.

Daten werde auf einer Schnittstelle unidirektional von einem Sender zu einem Empfänger weitergeleitet.

Es bietet an seinen Schnittstellen folgende elementare Eigenschaften für die Weiterleitung der Datenströme:

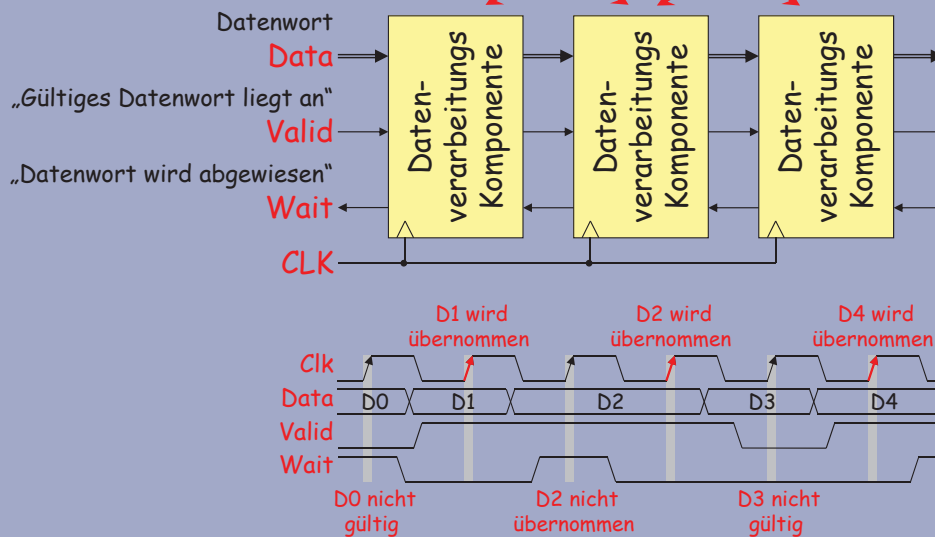
- Es ist schnell:
 - Mit jeder Taktflanke kann die Weitergabe eines Datenworts erfolgen.
- Es ist flexibel:
 - Gültige Datenworte werden vom Sender markiert.
 - Die Übernahme eines angebotenen Datenworts kann vom Empfänger bei jeder Taktflanke abgewiesen werden.

Vergleichbar dem „ready-valid handshake protocol“ (Dynalith Systems) und dem „AMBA AXI4-stream protocol“ (ARM)

8 Fakultät Ingenieurwissenschaften und Informatik

Einheitliche Schnittstelle

Lokale „Handshakes“

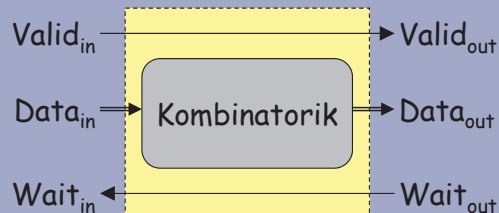


Elementare Datenfluss-Komponenten zur Modellierung von Pipelinesystemen

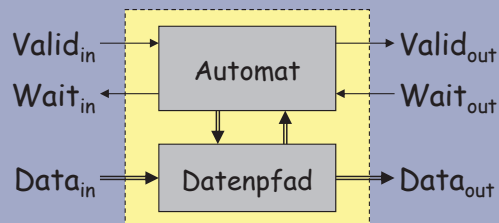
Elementare Datenfluss-Komponenten des Frameworks:

- Verarbeitung der Daten in der Pipeline
- Aufteilen des Datenflusses (Demultiplex, Split)
- Zusammenfügen von Datenflüssen (Multiplex, Merge)
- Synchronisation des Datenflusses
- Synchronisation des „Warte“-Flusses
- Zwischenspeichern von Datenflüssen
- Überführung von Datenflüssen über Taktgrenzen

Elementare Datenfluss-Komponenten: Verarbeitung von Datenströmen

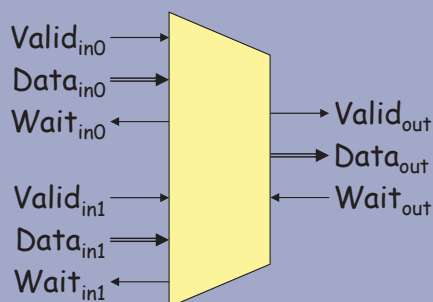


Einfache Komponente
Verarbeitung erzeugt kombinatorische Verzögerung auf den Datenleitungen



Komplexe Komponente
Verzögerungen können auf den Datenleitungen und den Kontrollleitungen „Valid“ und „Wait“ entstehen.

Elementare Datenfluss-Komponenten: Zusammenführen von Datenströmen

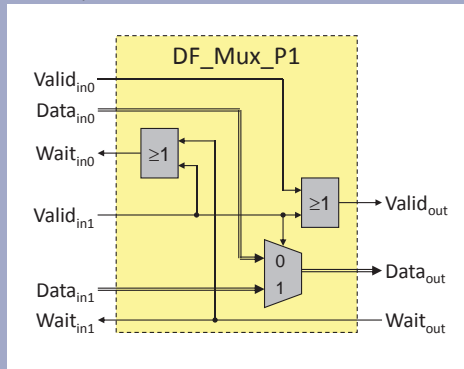


- **Multiplexen**
Zwei Datenströme gleicher Datenwortbreite werden zu einem Datenstrom zusammengefasst. Alle Datenworte beider Ströme bleibt eigenständig erhalten. Konflikte zwischen den Strömen müssen aufgelöst werden.
- **Synchronisieren**
Zwei Datenströme werden aufeinander synchronisiert. Die Datenworte der beiden Ströme werden paarweise zu einem Datenwort zusammengefasst.

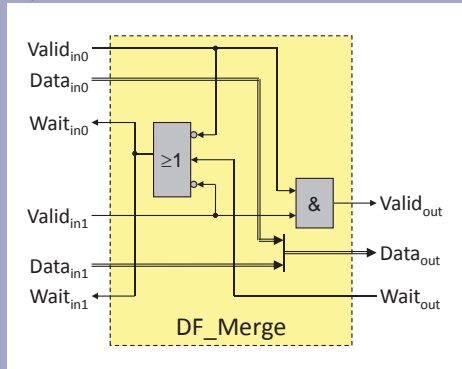
Elementare Datenfluss-Komponenten: Zusammenführen von Datenströmen

Beispiele

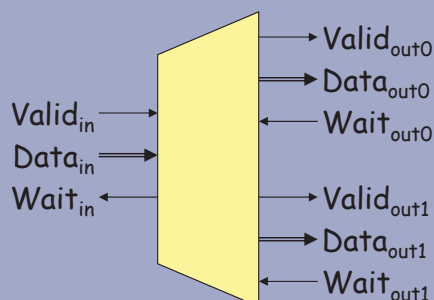
Multiplexen von Einzelworten



Synchronisieren

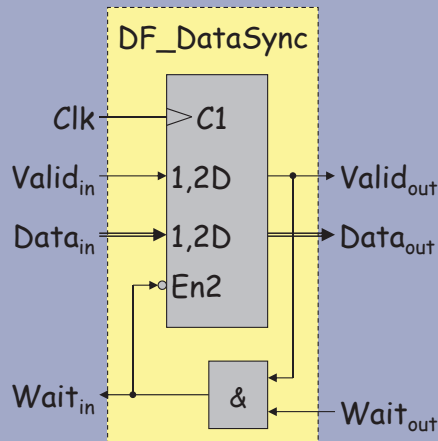


Elementare Datenfluss-Komponenten: Aufteilen von Datenströmen



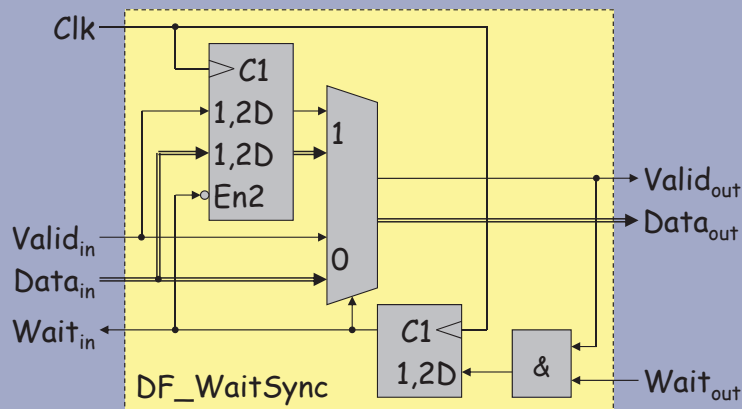
- **Demultiplexen**
Im Element muss ein Algorithmus installiert sein, der die ankommenden Datenworte auf die Ausgänge verteilt.
- **Duplizieren**
Jedes ankommende Datenwort wird unter Beachtung der Warte-Anforderungen auf beiden Ausgängen ausgegeben.

Elementare Datenfluss-Komponenten: Synchronisierung in Datenflussrichtung



- Daten am Ausgang werden **synchron** mit dem Takt bereitgestellt.
- Eine Warteanforderung vom Ausgang wird nur dann an den Eingang weitergereicht, wenn im Synchronisations-Register gültige Daten vorliegen.

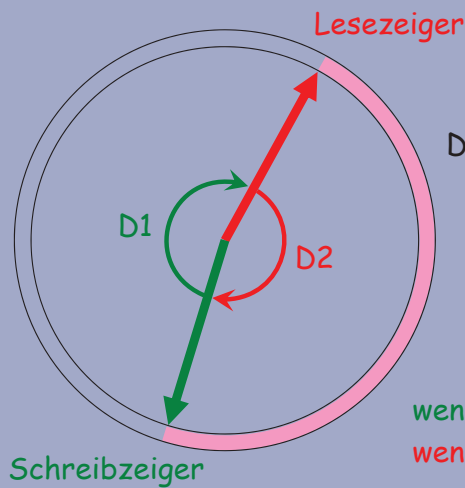
Elementare Datenfluss-Komponenten: Synchronisierung in Warteflussrichtung



- Daten werden vorsorglich in einem Register zwischengespeichert.
- Wenn am Ausgang eine Warteanforderung vorliegt, wird im folgenden Taktzyklus diese am Eingang aktiviert und das aktuelle Datenwort aus dem Register nochmals am Ausgang angelegt.

Elementare Datenfluss-Komponenten: Speichern von Datenströmen mittels Fifo

Prinzip eines Fifos



Differenzen:

$$D1 = \text{Lesezeiger} - \text{Schreibzeiger}$$

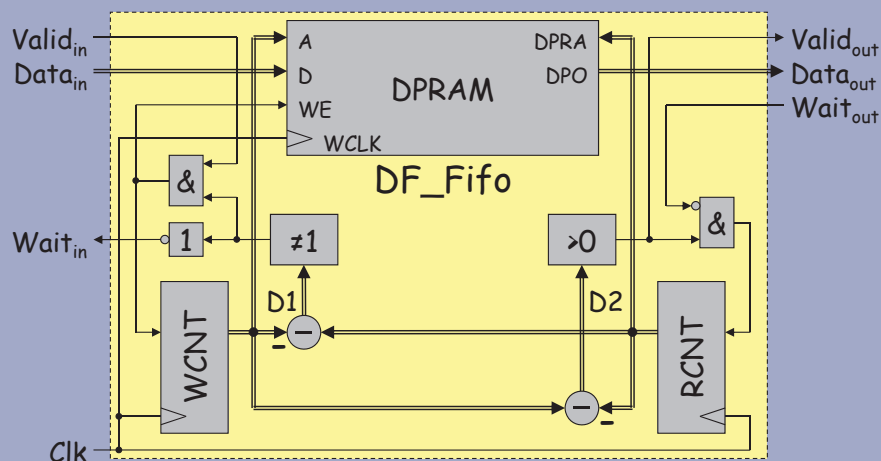
$$D2 = \text{Schreibzeiger} - \text{Lesezeiger}$$

wenn $D1 \neq 1$ darf geschrieben werden

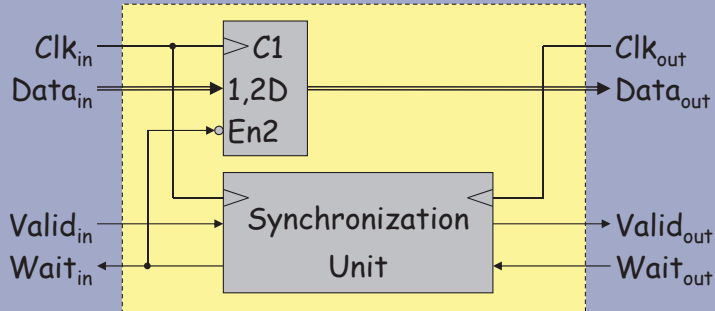
wenn $D2 > 0$ kann gelesen werden

Elementare Datenfluss-Komponenten: Speichern von Datenströmen mittels Fifo

Einfaches Fifo mit nur einem Takt



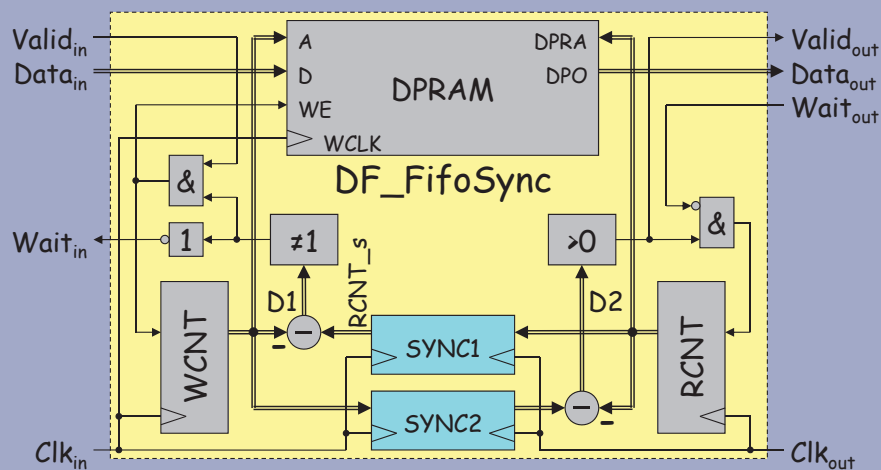
Elementare Datenfluss-Komponenten: Synchronisation über Taktgrenzen



- Speichern von $Data_{in}$ in einem Register.
- Sperren des Registers, bis die Synchronisation abgeschlossen ist. Während der Synchronisation ist das Wait-Signal am Eingang aktiv.
- Sichere Überführung des mit Clk_{in} synchronen Valid-Signals vom Eingang auf den mit Clk_{in} synchronen Ausgang mit Berücksichtigung des $Wait_{out}$ -Signals am Ausgang.

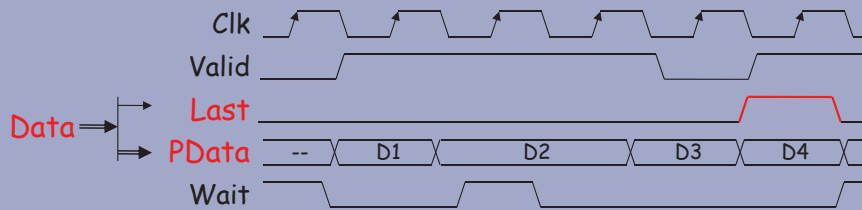
Elementare Datenfluss-Komponenten: Synchronisation über Taktgrenzen

Fifo mit separaten Takten für Schreib- und Leseseite



Höherwertige Protokolle für Paketdaten

Markierung des letzten Datenworts eines Pakets:

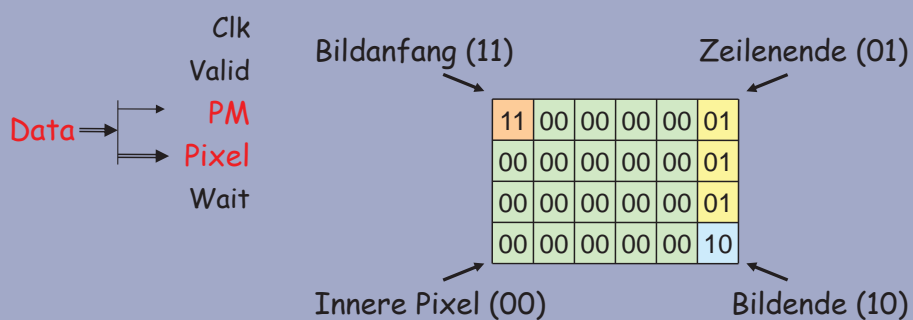


Beispiel eines Pakets aus 4 Datenworten

23

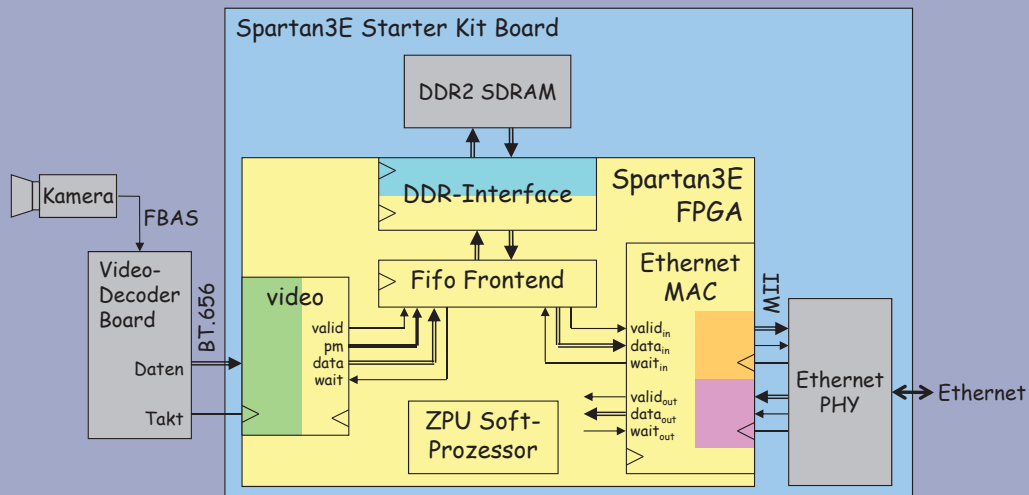
Höherwertige Protokolle für Bilddaten

Markierung der Pixel von Bilddaten (Pixelmarkierung PM):



24 Fakultät Ingenieurwissenschaften und Informatik

Testapplikation:



• Realisierte Komponenten und Anwendungen

- Kopplung paralleler Transputer-Prozessoren
- SIMD-Zeilenprozessor
- ATM Routing-Chip
- UART Sender/Empfänger
- Ethernet MAC mit UDP-Hardware-Port
- Bildverarbeitung
 - Zeilenspeicher
 - Sobel-Filter mit nachgeschaltetem Cordic
 - Rangordnungsfiler
 - Histogramm-Prozessor
- SDRAM-Fifo
- FFT-Prozessor
- Schnittstelle zu ZPU-Softprozessor
- ...

FPGA Framework Development For Modular High-Speed DAQ Systems

Qingqing Xia
FEA, DESY Hamburg
SEI Conference
HZDR Dresden, 12 March, 2012



Outline

> FEA Group

- Digital High Speed Electronics for DAQ Systems at Experiments and Accelerators

> Hardware for Modular High-Speed DAQ Systems

- Modularity of MicroTCA Systems
- Hardware development for MicroTCA Systems

> FPGA Framework Development

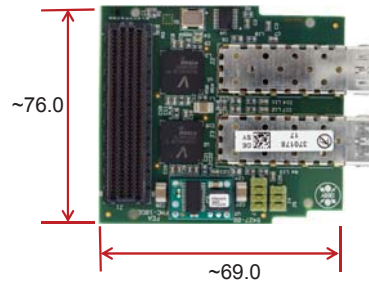
- Concept & Realization
- Framework Example: DAMC2 Framework Development



FEA Group: Digital High Speed Electronics for DAQ Systems at Experiments and Accelerators

> High-speed digital data acquisition and control systems

- System specification, design and development
 - *FPGA-based DAQ System*
 - *xTCA: Modular card-based System (AMC,RTM)*
 - *VMEbus, PXI*
 - *Fast serial links: 10-Gigabit-Ethernet*
 - *PCI-Express*
 - *DDR2 Memories*
- PCB-Design/Layout
- FPGA-Firmware development
- Hardware-oriented software development



FMC with 2x 10-Gigabit-Ethernet Links

> Centralized system administration for Mentor Graphics ECAD

- *Client – server installation*
- *Component library*
- *User project register/tracking*
- *Quality control*
- ...

Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 3



Hardware for Modular High-Speed DAQ Systems

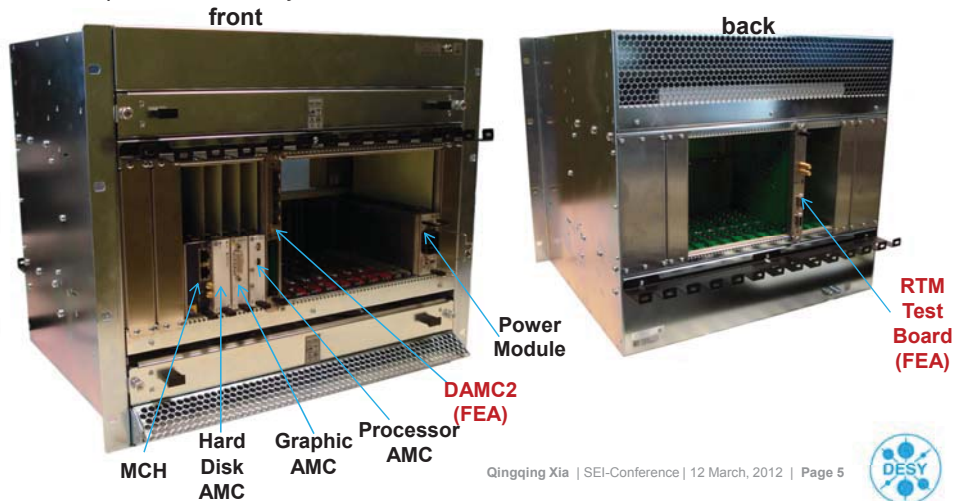
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 4



Hardware for Modular High-Speed DAQ Systems

> MicroTCA system for Physics (MTCA.4)

- MTCA.4: <http://www.picmg.org/v2internal/resourcepage2.cfm?id=5>
- MCH: *MicroTCA Carrier HUB*; AMC: *Advanced Mezzanine Card*
- *Set-Up in our laboratory*

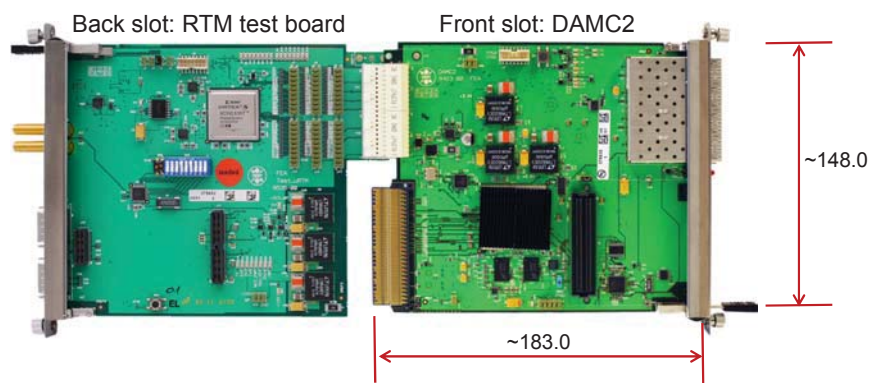


Hardware for Modular High-Speed DAQ Systems

> MicroTCA system for Physics (MTCA.4)

Modularity & Flexibility

- Analog and digital rear transition I/O can be separated from the FPGA based front module

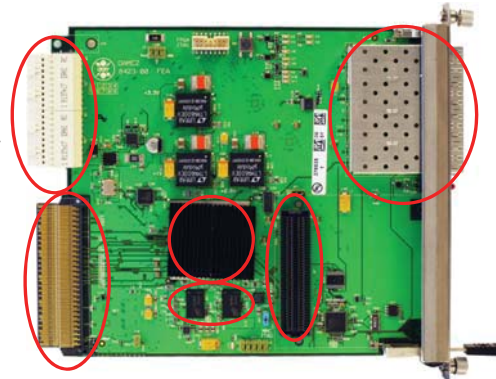


Hardware for Modular High-Speed DAQ Systems

> DAMC2 (DESY Double Size AMC)

Key Features

- FPGA: V5LX50T-1FFG1136
- Interface to “outside world”
 - 4 Lanes PCIe from AMC-edge-connector
 - Four SFP optical links
 - FMC connector
 - μ RTM connector
- On board memories
 - 2 x 512Mb DDR2 memories
- JTAG Interface
- Clocking
 - 100MHz clock from AMC-edge-connector
 - 2x I2C programmable oscillators(10M-810MHz)
 - On-board LVDS 200MHz oscillator
 - ...



Board will be used for different applications!

Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 7



Hardware for Modular High-Speed DAQ Systems

> RTM Test Board

Key Features

- FPGA: Data generator, Emulation of the behavior of different RTM boards
- Interface to front AMC board: 54 differential /108 single-ended signals

Application Example

- ADC Emulation: 14 bit 50MHz Sample Clock, 8 channels, on each channel 700Mbit/s 350MHz DDR (*Double Data Rate*) serial data



Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 8



FPGA Framework Development

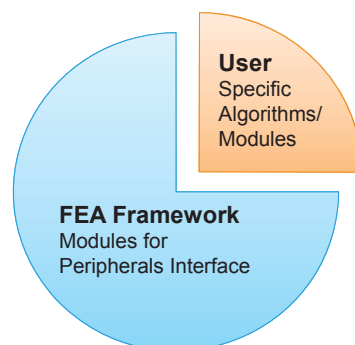
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 9



FPGA Framework Development

> Concept

- FEA provides FPGA framework with VHDL/Verilog – Modules for on board peripherals (PCIe, DDR2, Ethernet,...)
- User copy over the FEA framework and then focus on developing algorithms for various projects

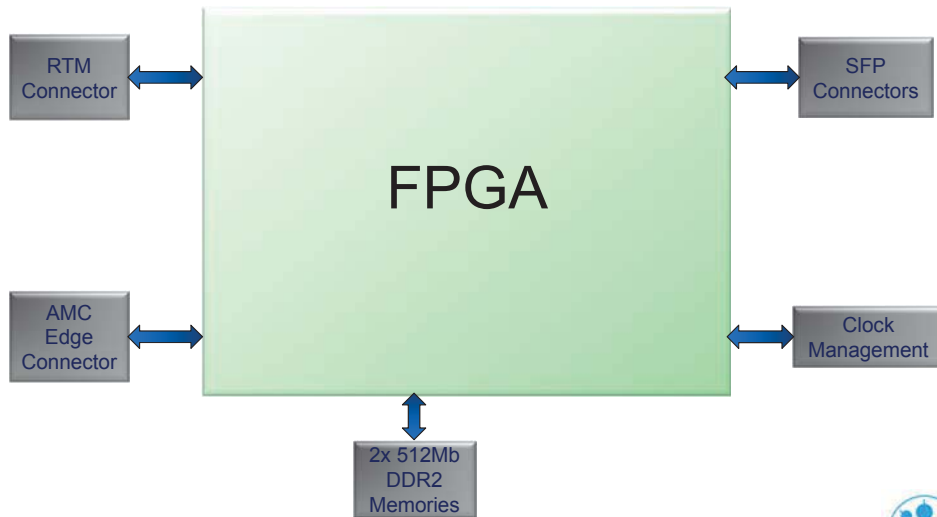


Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 10



FPGA Framework Development: DAMC2 Start-Up

> DAMC2 FPGA On-board Peripherals

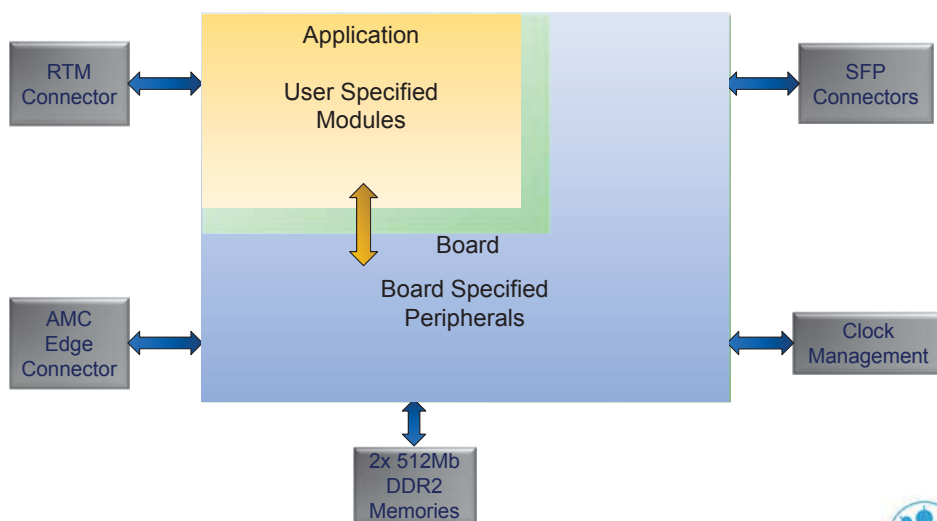


Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 11



FPGA Framework Development: DAMC2 Start-Up

> Concept Realization

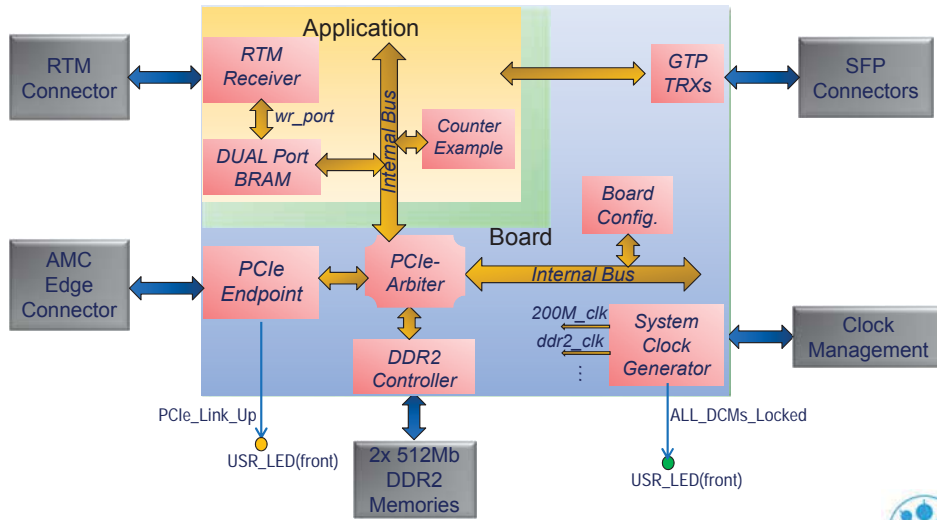


Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 12



FPGA Framework Development: DAMC2 Start-Up

> Framework Block Diagram

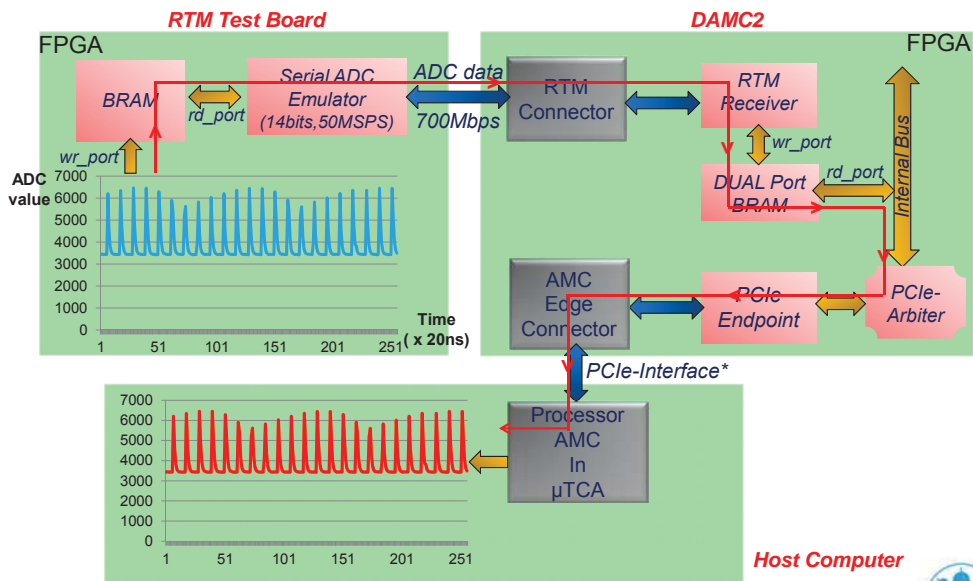


Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 13



FPGA Framework Development: DAMC2 Start-Up

> First Framework Chain-Test



Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 14



FPGA Framework Development: DAMC2 Start-Up

- > **First version of DAMC2-StartUp Framework is ready for user**
- > **Further improvement, optimization, tests will be done in the near future**

Thank you!



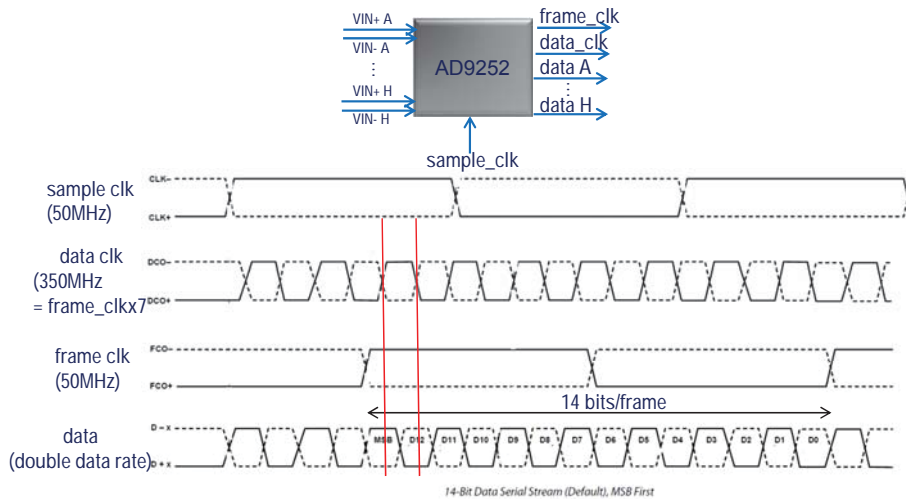
Spare Slides



FPGA Framework Development: DAMC2 Start-Up

> AD9252

- Octal, 14-Bit, 50 MSPS, Serial LVDS ADC
- 8 analog-to-digital converters (ADCs) integrated → 8 data channels



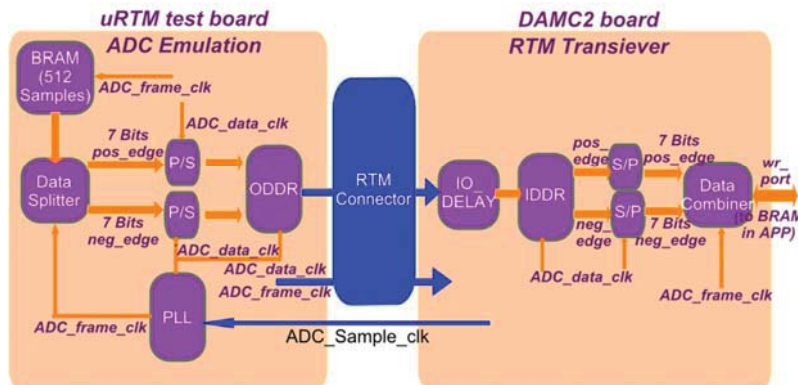
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 17



FPGA Framework Development

- > 8-Channels ADC Emulation using uRTM test mode
- > Block diagram for **one single channel (350 DDR serial data)**

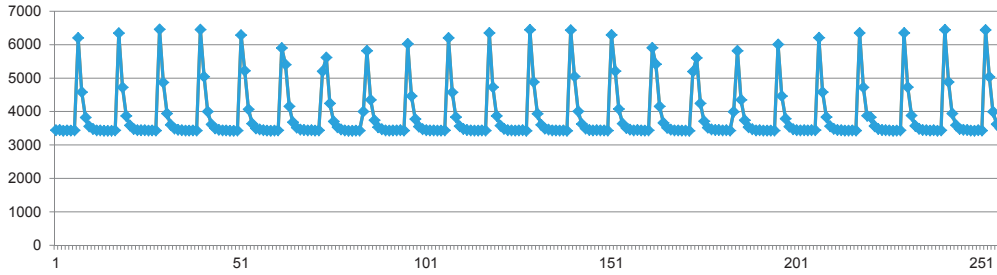
ADC_frame_clk=50MHz
ADC_data_clk=350MHz



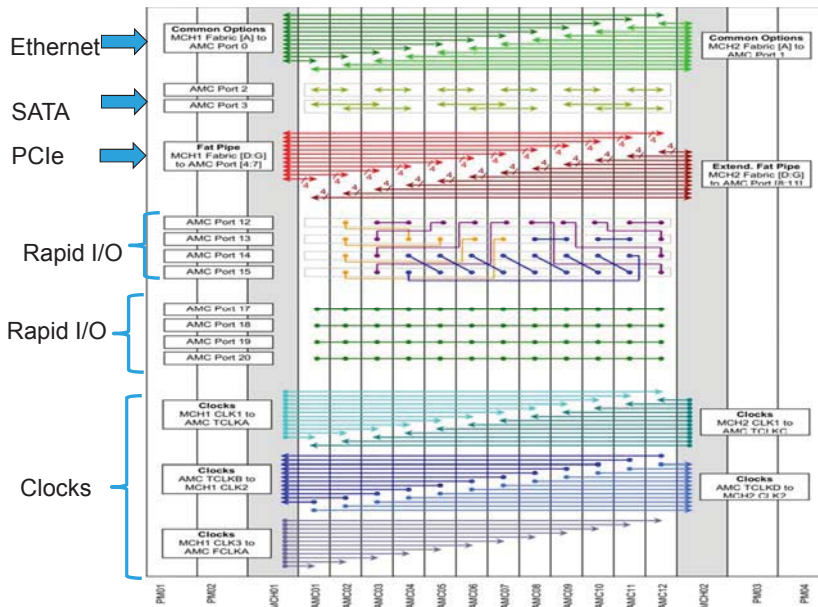
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 18



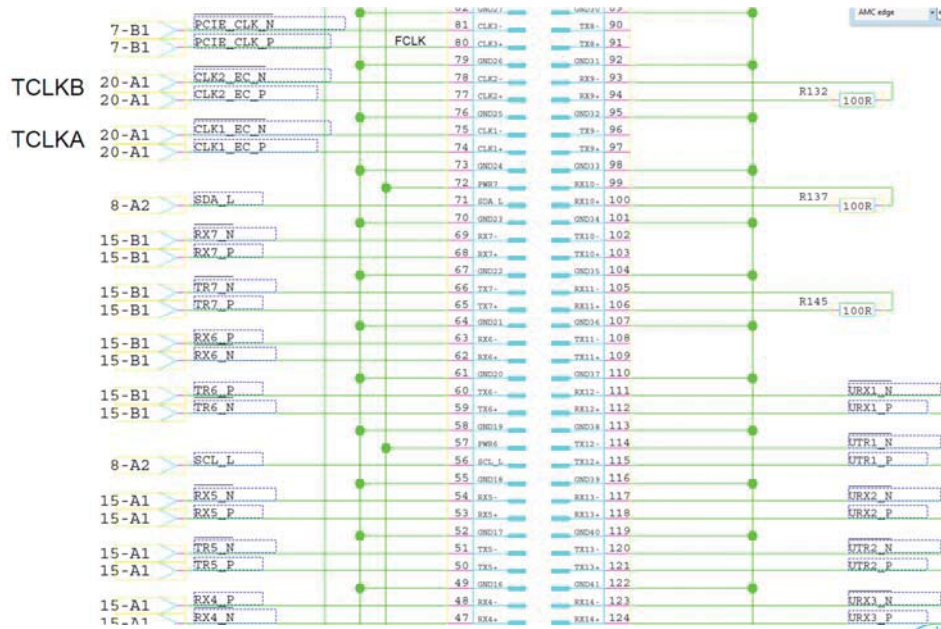
ADC Samples



Notes



Notes



Bruno Fernandes, WP76 (DAQ & Control Systems)

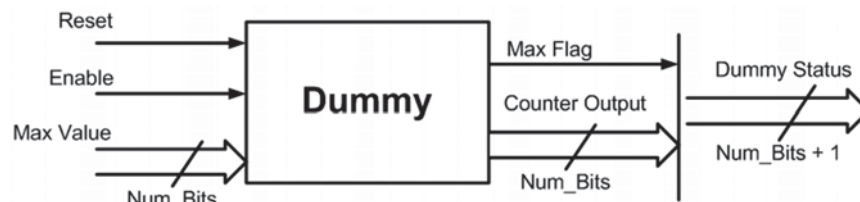
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 21



Dummy Module (Later Exercise)

Simple configurable counter to show how to use the II bus data/strobes signals to configure/monitor user defined registers:

- Asynchronous Reset with II bus Strobe (write or read cycle)
- Enable implemented as Internal register (no Strobe required)
- Max value configurable using II strobe + Data bus (write cycle only)
- Monitor of Max value and Dummy Status



Bruno Fernandes, WP76 (DAQ & Control Systems)

Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 22



Notes

- The MCH is a MicroTCA (uTCA/MTCA) Carrier Hub in the form factor of a single width, mid- or full-size Advanced Mezzanine Card (AMC). It provides the central management and data switching entity for all MicroTCA systems.
- Using the features of AdvancedMC, MicroTCA provides high performance and high availability in one system. MicroTCA has the advantages and technology of AdvancedTCA but in a small form factor. This new technology is used in multiple industries with many different applications. In general, every application of AdvancedMC modules is also possible with MicroTCA modules. Schroff will supply development tools, standard applications and customised systems to further develop this new technology.
- **Advanced Mezzanine Cards** are [printed circuit boards](#) (PCBs) that follow a [specification](#) of the [PCI Industrial Computers Manufacturers Group \(PICMG\)](#), with more than 100 companies participating. Known as AdvancedMC
- **MicroTCA** (auch: **µTCA**) stand for *Micro Telecommunications Computing Architecture*
- **HUB(1):** *IPMI(I2C) Interface connected to every other AMC Modules, so that information(sensor, temperature,...) can be downloaded to HUB and it everthing is fine, HUB will allow PM to supply power to that AMC*
- **HUB(2):** *Forward connection between different AMC with PCIe, because no straight connection between two AMC's Pcie interface.*
- *Hard disk(AMC01) → SATA interface & I2C*
- *Graphic card → PCIe& I2C*
- *Processor card → PCIe to Graphic Card and DAMC2, SATA interface to Hard disk, and I2C*
- *Processor card to HUB, HUB to Graphic Card/DAMC2(Pcie) → Processor to Grahpic Card/DAMC2(Pcie)*
- *8000Euro for crate, HUB, CPU, Graphic Card, Hard disk, PM*
- *carte: 2500-3000Euro*

Bruno Fernandes, WP76 (DAQ & Control Systems)

Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 23



Notes

- Hard disk: Telum 200-SATA is a Serial ATA (SATA) hard disk drive AdvancedMC (AMC) , 200GB
- Graphic Card: Telum™ 3001 is a high-quality graphics adapter AdvancedMC that incorporates a high-performance graphics processing unit (GPU). Integrating x8 PCI-Express (PCI-E) and 128 MB memory in a single package, allows the GPU to deliver exceptional 2D, 3D and multimedia graphics performance.
- Telum NPA-5854 is an intelligent high-performance IP packet processor AdvancedMC (AMC) based on the Cavium OCTEON Plus multi-core CN5850-SCP processor.
- **Advanced Mezzanine Cards** are [printed circuit boards](#) (PCBs) that follow a [specification](#) of the [PCI Industrial Computers Manufacturers Group \(PICMG\)](#)
- RTM Connector: **Advanced CA Zone 2 Z-PACK HM-Zd High Speed Signal Connectors Front Board Connector**

Intel® Core™ 2 Duo L7400 processor, 1.5 GHz core clock, 4 MByte L2 cache

- Single width, full-size or mid-size AMC.0, R2.0 form factor
- AMC.0, R2.0 Hot Swap compliant
- Up to 4 GByte DDR2 SDRAM with ECC in main memory (2 banks of soldered components)

Bruno Fernandes, WP76 (DAQ & Control Systems)

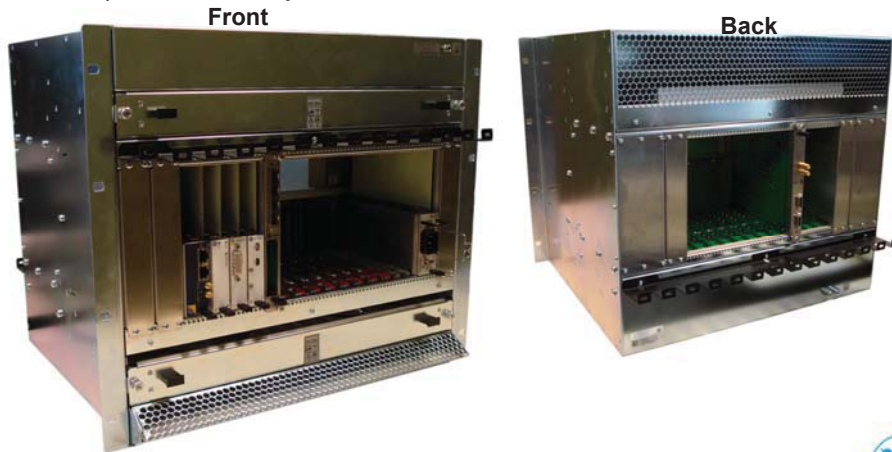
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 24



Hardware for Modular High-Speed DAQ Systems

> MicroTCA system for Physics (MTCA.4)

- MTCA.4: <http://www.picmg.org/v2internal/resourcepage2.cfm?id=5>
- MCH: *MicroTCA Carrier HUB*; AMC: *Advanced Mezzanine Card*
- *Set-Up in our laboratory*

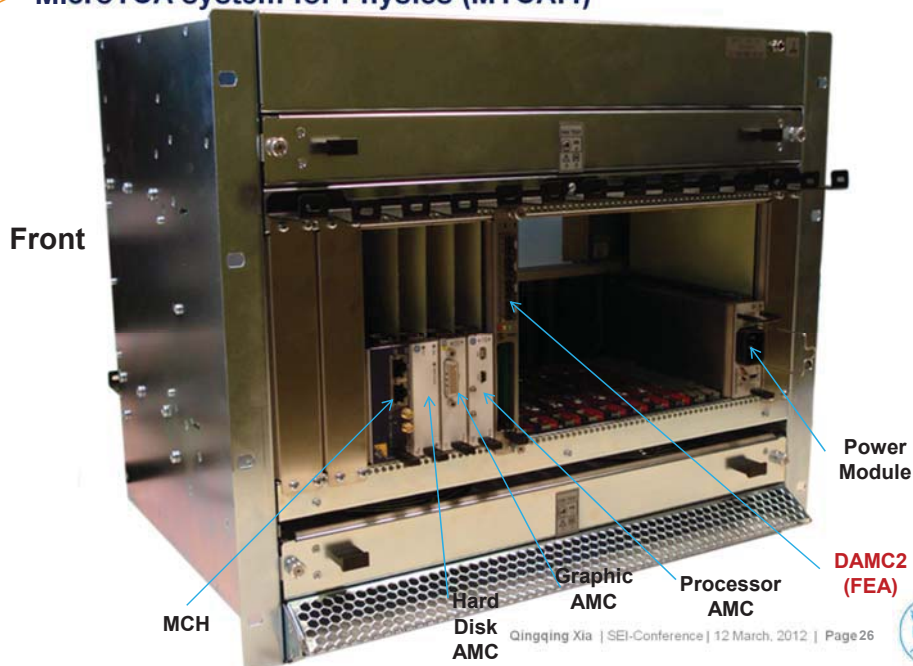


Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 25



Hardware for Modular High-Speed DAQ Systems

> MicroTCA system for Physics (MTCA.4)



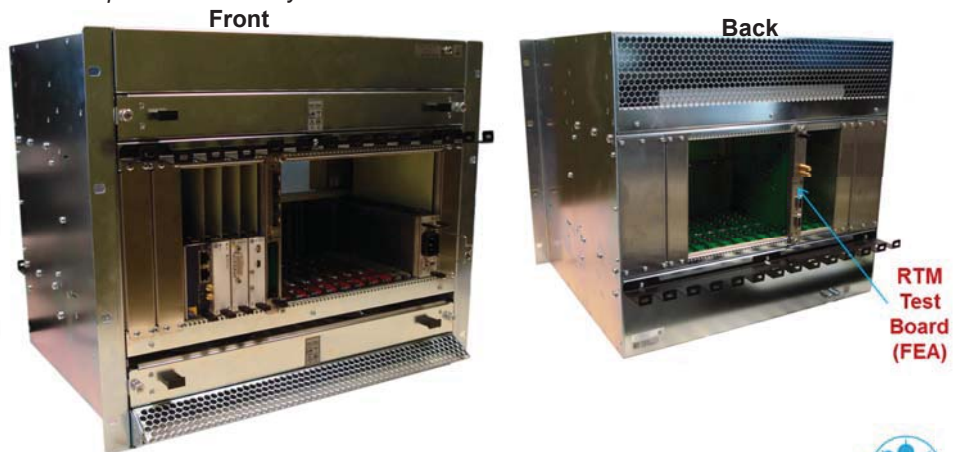
Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 26



Hardware for Modular High-Speed DAQ Systems

> MicroTCA system for Physics (MTCA.4)

- MTCA.4: <http://www.picmg.org/v2internal/resourcepage2.cfm?id=5>
- MCH: *MicroTCA Carrier HUB*; AMC: *Advanced Mezzanine Card*
- *Set-Up in our laboratory*



Qingqing Xia | SEI-Conference | 12 March, 2012 | Page 27



Überlegungen zur Einführung von TANGO im Jülich Centre for Neutron Science

06.03.12 | Harald Kleines

Einführung



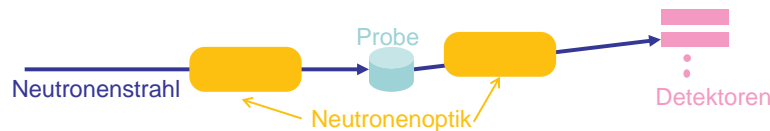
Neutronenleiterhalle des FRMII

- ZEL entwickelt die Kontroll- und Datenerfassungssysteme für die Neutroneninstrumente des Jülich Centre for Neutron Science (JCNS)
 - 11 Instrumente an den 3 Außenstellen (FRMII, ILL, SNS) des JCNS
 - homogene Systemarchitektur: „Jülich-Münchener Standard“
 - Instrument-Software basiert auf dem „Kontrollsystem“ TACO* (ESRF)
- Wechsel zum Nachfolgesystem TANGO**??
 - sehr aufwendig
 - Langjährige Koexistenz
 - Situation FRMII und ESS?

*TACO: Telescope and Accelerator Controlled with Objects

**TANGO: Taco Next Generation Objects

Steuerungs- und DAQ-Software für Neutroneninstrumente



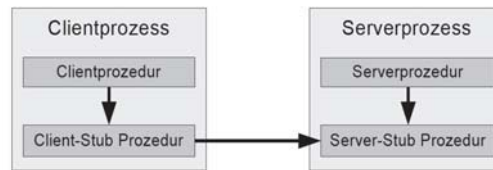
- Aufgaben: “Slow Control” + Datenerfassung + Experimentsteuerung
- Typisch: verteilte Client/Server-Architekturen mit standardisierter Middleware, Bedienung über Skripte, teilweise auch mit GUIs
- Beispiele:
 - ILL: MAD, NOMAD (basiert auf CORBA)
 - NIST: ICP
 - SNS: pyDas mit python/Windows, HFIR: „noname“ mit Labview/Windows
 - HZB: CARESS (basiert auf CORBA)
- Tendenz der letzten Jahre: Verwende standardisiertes **Kontrollsystem** (insbesondere auch in der Synchrotronsteuerung)
 - FRJ2 (TACO), FRMII (TACO)

Kontrollsysteme: Terminologie

- HMI-Systeme
 - Beispiele: ProTool, WinCC Flexible,...
 - Entwicklungsumgebungen für Bedienpanels
- SCADA-Systeme (Supervisory Control and Data Acquisition)
 - Beispiele: Wonderware InTouch, Siemens WinCC,....
 - Graphische Editoren für Prozessbilder, Skripting
 - Datenbanken für Konfiguration, Alarme und Prozessdaten
 - „Variablen-zentriertes“ Modell
- DCS (Distributed Control Systems)
 - Beispiele: Siemens PCS7, EPICS,...
 - Beinhalten auch Frontend-Systeme für Steuerungsaufgaben
 - Verteilte Systeme

Verteilte Systeme

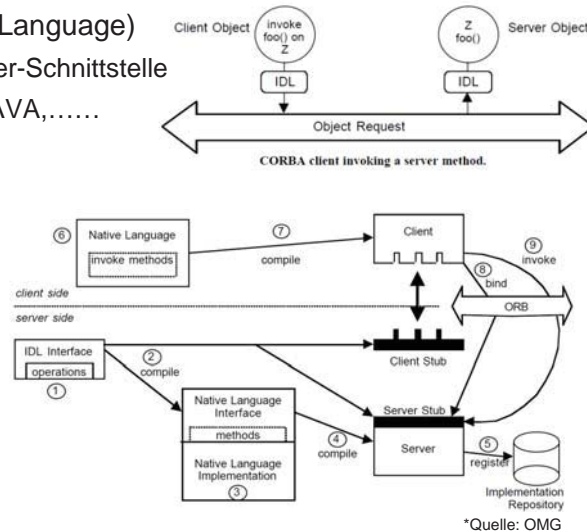
- Middleware:
 - Softwareschicht zwischen HW + OS und Applikation
 - Unterstützung der Verteilung
 - Plattform-übergreifend
- Beispiele: .NET, DCOM, SOAP
- RPC



- Objekt-orientiertes Analogon: Java RMI
 - Ist sprachabhängig!!!

CORBA (Common Object Request Broker Architecture)

- Von der OMG standardisiertes objektorientiertes Middleware-System
- IDL (Interface Definition Language)
 - Beschreibung der Server-Schnittstelle
 - Übersetzung in C++, JAVA,.....
- 3-Tier: ORB
- Protokoll: IIOP (via TCP)
- Adressen: IOR

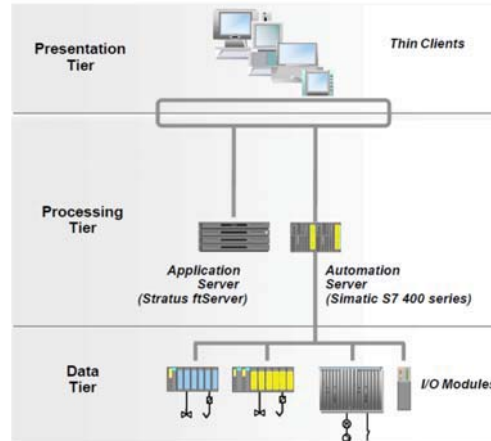


*Quelle: OMG

- Alternativen
 - ZeroC ICE
 - Facebook Thrift
 - ØMQ
 -

Trends bei industriellen Kontrollsystemen

- Stark Windows-basiert (ActiveX, .Net, VB-Skripting, OPC)
- Verteilte Client/Server-Architekturen mit Redundanz
- 3-Tier Modell



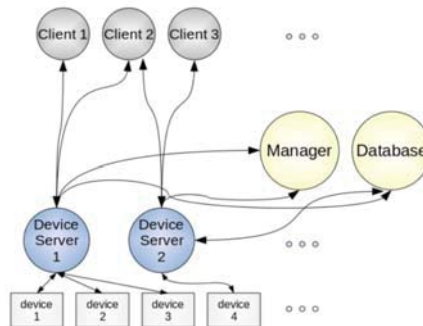
*Quelle: Siemens AG

Situation bei Kontrollsystemen in der Forschung

- Getrieben durch Beschleuniger
- Industrielle SCADA-Systeme meist nur für Subsysteme
 - Ausnahme: CERN mit PVSSII
- **Nordamerika:** Dominierend ist EPICS, wg. Druck des DOE
 - Veraltet, Datenbank und Prozessvisualisierung heterogen
- **Asien:** proprietäre Systeme, EPICS, industrielle Systeme (Yokogawa,..)
- **Europa:** sehr vielfältig
 - DESY: TINE, DOOCS, TANGO, EPICS
 - CERN: FESA, UNICOS(PVSSII) für LHC-Beschleuniger; JCOP (PVSSII) für LHC-Experimente
 - FAIR: FESA für Beschleuniger, EPICS für PANDA
 - ITER: Entscheidung zwischen TANGO und EPICS zugunsten von EPICS

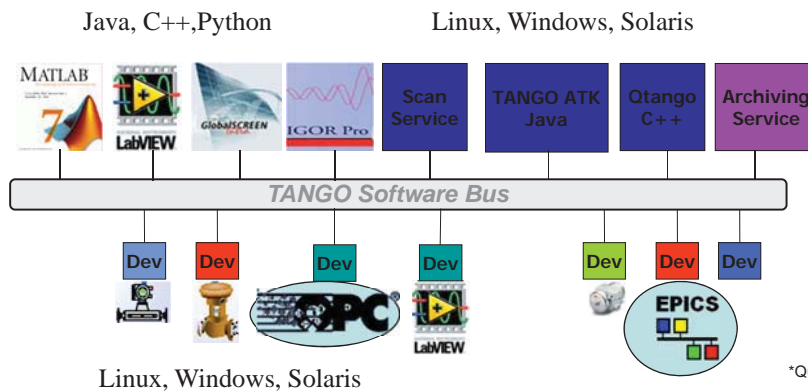
Situation im JCNS

- TACO (ESRF, ca. 1990): Tool für Entwicklung von Kontrollsystemen
- Verteilte Client-Server-Architektur (Sun-RPC)
- Geräteabstraktion
 - `dev_putget(dev,cmd,argin,argin_type,argout,argout_type,error)`
- Name-service: Taco Manager => Ortstransparenz
- Datenbank: dbm (+mysql)
 - TACO-Konfiguration + Parameter
- Probleme bei Multithreading
- Rudimentäre Implementierung von Asynchronität (Events,...)



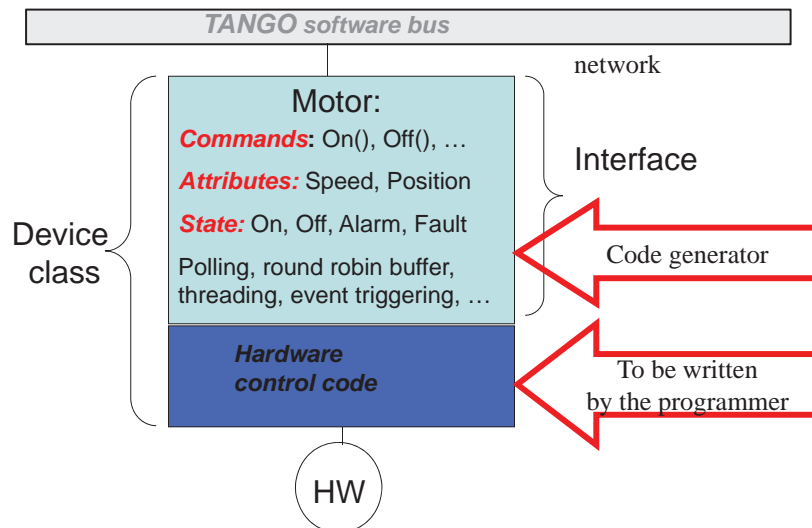
TANGO

- Objektorientierter Nachfolger von TACO (Entwicklung durch ESRF seit 2001)
- CORBA-basiertes Toolkit für die Entwicklung von Kontrollsystemen
- Modell: „Software-Bus“ für verteilte Objekte
- Definiert standardisierte Schnittstelle zu Geräten (Devices)



*Quelle: ESRF

TANGO-Device



*Quelle: ESRF

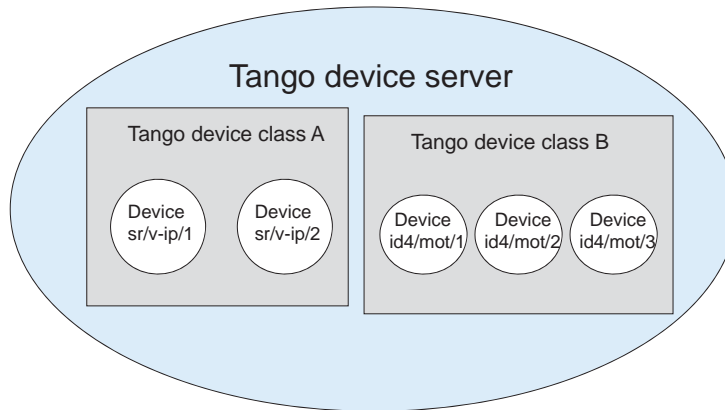
TANGO Device

- Device state
 - 14 vordefiniert Zustände (ON, OFF, FAULT, STANDBY, UNKNOWN....)
 - Endliche Automaten erlauben zustandsabhängige Operationen auf Geräten
- Attribute: Lesen und Schreiben von physikalischen Werten
 - Wert, Status, Zeitstempel, Dimension
- Kommandos: Auslösen von Aktionen auf dem Gerät
 - Ein generischer Aufruf


```
out = dev.command_inout("Cmd name",in);
```
 - 20 verschiedene Datentypen
 - Jedes device besitzt automatisch 2 Kommandos DevState and DevStatus
 - Abfrage der verfügbaren Kommandos


```
dev.command_list_query()
```

Device Server



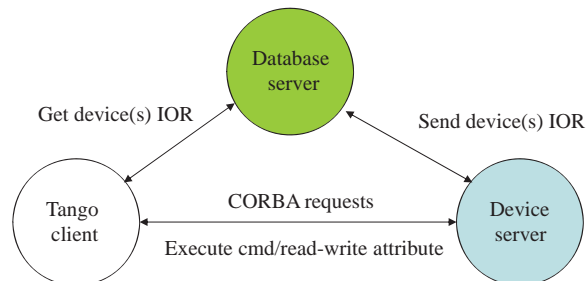
- Server ist Prozess auf Betriebssystemebene, multithreaded

*Quelle: ESRF

Polling und Events

- Polling Thread in jedem Server füllt Round-Robin Buffer
 - Polling Buffer für Kommandos und Attribute
- Client kann vom Gerät oder aus dem Polling Buffer lesen
 - => Data Caching
- Alternativ: Events
 - Polling Thread erkennt Event-Bedingung (Werteänderung,.....)
 - Aufruf von Corba-Methode omniNotify zur Benachrichtigung
 - Aufruf von Callback im Client

TANGO Datenbank



- Datenbank: mySQL oder Oracle
 - Konfiguration
 - Parametrierung
 - Archive

*Quelle: ESRF

TANGO Archive

- HDB (History Data Base)
 - Langzeitarchiv für Prozessdaten
 - Abtast-Intervall $\geq 10s$
 - Unbegrenzt
- TDB (Temporary Data Base)
 - Kurzzeitarchiv für Prozessdaten
 - Abtast-Intervall $\geq 100\ ms$
 - Maximal 3 Tage
- Snap (Snapshot Data Base)
 - Schnappschüsse, die wieder eingespielt werden können??

Tango Tools

- **Astor/Starter:** Administration des Systems (Starten, Stoppen,...)
- **Pogo:** Graphischer Editor zur Generierung von Server-Code
- **Jive:** Graphischer Browser für die Datenbank und Test-Tool für Devices
- **LogViewer:** Graphisches Display für Log-Nachrichten
- **JDraw:** Graphischer Editor für Prozessbilder (JAVA)
- **Canone:** Graphischer Editor für Prozessbilder (Web)
- **DeviceTree:** Graphische Applikation Konfiguration von Device-Attributen und zum Ausführen von Device-Kommandos
- Diverse Tools für die Archive HDB, TDB und Snap
- **E-Giga:** Datendisplay für Webbrowser

Status

- Open Source
- Zusätzlich kann über MOU der Status eines Committers oder Contributors erworben werden
- Wird aktiv weiterentwickelt
- Unterstützte Sprachen

	C++	Java	Python	Matlab	LabView	IgorPro
Client	OK	OK	OK	OK	OK	OK
Server	OK	OK	OK			*

- Unterstützte Betriebssysteme
 - Windows 98 / 2000 / XP, Solaris 7/9, div. releases von Suse, RedHat, Debian und Ubuntu

*Quelle: ESRF

Motivation für die Einführung von TANGO

- Konsistente Objektorientierung
- Beseitigung von Defiziten von TACO
 - Multi-threading
 - Event-Mechanismus
 - Fehlendes Data-Caching
 -
- Funktionalitäts-Erweiterungen
 - Logging/Alarm-System, Graphischer Editor für Prozessbilder, Archive,....
- Migrationsunterstützung
- Aktive Kollaboration: Elettra, Alba, Soleil, DESY, MAX-lab, TUM
- **Modernisierung, Verbesserung des Supports, neue Funktionalitäten, Fit werden für die ESS!!!**

Migration der JCNS-Instrumente zu TANGO

- Vorarbeiten
 - Installation, Server-Entwicklung/Test
 - Test von omniORB (CORBA ORB)
 - Bachelor-Arbeit (Analyse der TANGO-Funktionalität und –Performance, Implementierung Motorserver)
- Exemplarische Migration eines Instruments
 - Kleinwinkelstreuanlage KWS3
 - Test von Performance und Stabilität
 - Ziel: User sollte zunächst keine Unterschiede sehen
 - Zeitrahmen: bis zum Beginn des nächsten Reaktor-Zyklus (Mitte April)
- Danach: „go/not go“ – Entscheidung und Entwicklung eines Zeitplans
- Abstimmung mit der TUM erfolgt kontinuierlich

EtherCAT Feldbusknoten: Entwicklung, Systemunterstützung und -kompatibilität

Peter Kaefer, Helmholtz-Zentrum Dresden-Rossendorf; März 2012

Abstract: Die Verwendung selbst entwickelter Feldbusknoten und deren Einbindung in kommerzielle Automatisierungssysteme erfordert in der frühen Phase des Produktlebenszyklus einen erhöhten Entwicklungsaufwand. Nach erfolgreicher Integration bieten Hersteller von Automatisierungssystemen eine langfristig stabile und leistungsfähige Umgebung zur Projektierung und Programmierung von Anlagen, welche langfristig den Aufwand zur Pflege minimiert. Zur Überprüfung der Funktionsfähigkeit des Gesamtsystems ist die durch Testwerkzeuge unterstützte Systemkompatibilität eine entscheidende Voraussetzung. Die Vorgehensweise bei der Systemintegration eines Slave Device und der Prüfung der Systemkompatibilität wird im Folgenden vorgestellt.

Die Verwendung industrieller Automatisierungssysteme ist im Helmholtz-Zentrum Dresden-Rossendorf (HZDR) fester Bestandteil von Anlagen, für den wissenschaftlichen Gerätebau. Mit der Entwicklung Ethernet-basierter Feldbussysteme [5], [6], [7], [8] bietet sich die Möglichkeit, leistungsfähige Geräte für spezialisierte Funktionen zu entwickeln. Hierzu wurde ein EtherCAT Slave Device mit unterschiedlichen analogen und digitalen Schnittstellen entwickelt und dessen Systemkompatibilität über das Testwerkzeug der EtherCAT Technology Group validiert.

Im HZDR wurden Steuerungslösungen auf Basis industrieller Leitsysteme und Komponenten in einer Vielzahl von Anlagen unterschiedlicher Größe realisiert. Beispielsweise wurde die umfangreiche Automatisierungsinfrastruktur des Beschleunigers ELBE mit mehreren zehntausend I/O-Punkten unter Verwendung eines industriellen SPS-Systems aufgebaut. Die Realisierung der Flüssigmetallanlage LIMMCAST oder der Steuerung für das Hochfeldlabor erforderten gleichfalls sichere und nicht sichere Komponenten, bei denen der Einsatz industriell erprobter und zertifizierter Systeme einen großen Realisierungsvorteil im Bezug auf rasche Umsetzung und langfristige Unterstützung bot.

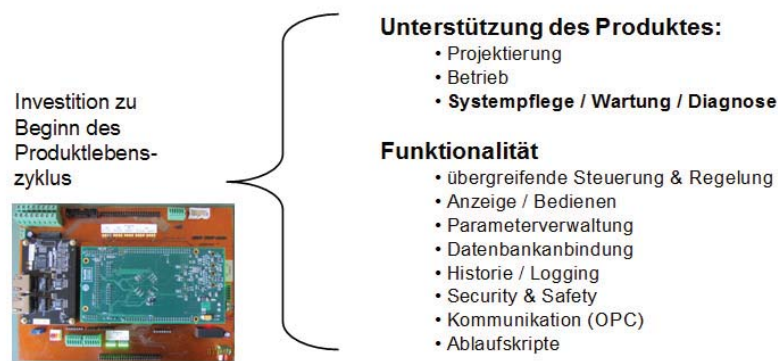


Bild 1: Motivation zur Entwicklung spezialisierter Feldbusknoten

In vielen Forschungseinrichtungen liegt der Schwerpunkt der Entwicklungstätigkeit auf der Bereitstellung der spezifischen Funktionalität. Berücksichtigt man die Langlebigkeit der Experimente und die Notwendigkeit, nachträglich Funktionalität zu erweitern, so ergibt sich auch im wissenschaftlichen Umfeld die Notwendigkeit zum effizienten Einsatz von Ressourcen. Eine

Gesamtbetrachtung des Aufwands zeigt sich auch hier, dass der Aufwand für Integration und Wartung der Anlagen erheblich ist.

Bei systemkonformer Realisierung spezifischer Gerätefunktionen kann, beginnend mit der Projektierung über den Betrieb bis zu Systempflege mit Wartung und Diagnose, der komplette Lebenszyklus des Gerätes unterstützt werden. Die Überprüfung der Systemkonformität stellt somit einen wesentlichen Aspekt der Entwicklung von Feldbusgeräten dar.

Naturgemäß konnte im HZDR das künftige Einsatzgebiet derartiger Feldbusgeräte bei einer ersten Abschätzung von Einsatzmöglichkeiten nicht eingegrenzt werden. Im Umfeld eines Forschungszentrums gibt es Anwendungen, die aufgrund der schnell ablaufenden Prozesse auch kurze Buszyklen erfordern. Andere Anwendungen stellen hohe Anforderungen an die zeitliche Genauigkeit bzw. an den die Obergrenze des Jitters beim Ausgeben von Signalen. Daher erschien es wünschenswert, Feldbusknoten zu entwickeln, die im Falle von EtherCAT statt im Modus „Sync Manager“ zur Minimierung des Jitters auch mit „Distributed Clock“ betrieben werden können. Schnelle Applikationen erfordern es, Buszyklen $\leq 200 \mu\text{s}$ zu nutzen.

Die Entwicklung von Feldbusknoten für EtherCAT ist auf unterschiedlichen Wegen möglich. Kommerziell verfügbare Module erlauben eine raschere Realisierung der Funktionalität innerhalb eines vorgegebenen Rahmens. Der Zeitgewinn wird durch Verzicht auf die Beeinflussung der oben erwähnten Kommunikationsdetails wie Aktualisierungsrate und der Verwendung unterschiedlicher Buszyklen erkauft. Anders als einige am Markt verfügbare Fertigmodule ist der im HZDR entwickelte Feldbusknoten parametrierbar und in der Lage, auch in Systemen mit schnellen oder synchronisierten Buszugriffen zu arbeiten und bietet so die erforderliche Flexibilität für den Einsatz im wissenschaftlichen Gerätebau.

Anhand der Entwicklung dieses EtherCAT Slave Device werden im Folgenden die wichtigsten Schritte bei der Implementierung beschrieben.

Zunächst sind bei der Adaptierung von Software an einen neuen Controller durch Anpassung von Byte-Order und Alignment bzw. dem Packen von Datenstrukturen im Falle von Feldbusknoten zusätzliche Funktionen zu implementieren. Wichtig sind hier der Datenverkehr zum Feldbus sowie die Definition der Prozessdatenobjekte und Datenstrukturen für asynchronen Datenverkehr. Das Abbild der Prozessdatenobjekte auf die Peripherie des Mikrocontrollers ist neben der Einhaltung der zeitlichen Randbedingungen durch die Interruptserviceroutine eines Timers ein weiterer funktionaler Bestandteil. Da die Kommunikation mit dem Feldbus innerhalb eines zeitlich festgelegten Rasters erfolgen muss, wird auch diese in einer Interruptserviceroutine durchgeführt.

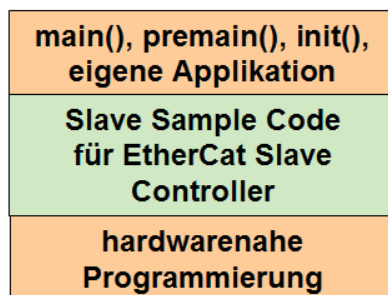


Bild 2: Anpassungsbedarf bei der Realisierung eines Feldbusknotens

Daraus folgt die in Bild 2 veranschaulichte Notwendigkeit, den Slave Sample Code für einen EtherCAT Feldbusknoten mit hardwarenahen Funktionen und solchen Funktionen zu ergänzen, die hier der Applikationsschicht zugeordnet sind. Bei Veränderung des Slave Sample Codes sind ggf. Änderungen in allen Teilen des Programms vorzunehmen.

Innerhalb der ersten Realisierungsstufe wurde in [1] auf einem ARM9 Mikrocontroller die Übertragung der Prozessdatenobjekte die Integration in eine SPS-Steuerung implementiert und

somit die prinzipielle Funktionsfähigkeit gezeigt. Bis März 2011 wurde auch die Einbindung auf einen OPC –Server zum Austausch der Prozessdaten realisiert. Auf Basis dieser Ergebnisse konnte die Entscheidung für eine weitergehende Implementierung getroffen werden. Diese beinhaltet vor allem die Fähigkeit zur asynchronen Übertragung von Strukturinformationen des Data Dictionary über die Mailbox und eine Absicherung der Systemkompatibilität durch das Conformance Test Tool.

Für eine Realisierung von EtherCAT Slave Devices sind der Zugang zu detaillierten Spezifikationen und die Verfügbarkeit des Slave Sample Code wichtige Voraussetzungen. Durch die Mitgliedschaft in der EtherCAT Technology Group wird der Zugang hierzu freigegeben. Ausgehend vom Slave Sample Code V4i30 wurde für das hier realisierte Device eine erste Implementierung durchgeführt und auf einen ARM9 [1], [2] portiert. Im Zuge der Aktualisierung des Slave Sample Code auf V4i42 im März 2011 wurde diese Realisierung in allen Ebenen von Bild 2 angepasst. Eine Realisierung mit Hilfe des zur Zeit neuesten Release des Slave Sample Code Vi50 vom November 2011 wurde bislang noch nicht durchgeführt.

Unterstützend werden Workshops [9],[10] zur Vermittlung der Technologie und Terminologie angeboten, auf denen vor allem die verschiedenen Komponenten und deren Zusammenspiel im System erläutert werden. Naturgemäß bilden Komponenten, welche eine Beschreibung des Verhaltens beinhalten oder an der Kommunikation beteiligt sind, die Basis für das Verständnis und sind zentrale Elemente der Schulung. Neben dem System Manager gehören hierzu die FMMU sowie die Komponenten des EtherCAT Slave Controllers EProm, SyncManager und der Slave Sample Code mit der State Machine. Dies bietet eine gute Basis zur Implementierung des Austauschs von Prozessdatenobjekten. Mailbox-Datenverkehr ist zur asynchronen Übertragung von Strukturinformation des Data Dictionary, wie er beispielsweise in Bild 3 zu sehen ist, erforderlich und wird erst für eine Implementierung mit erweitertem Funktionsumfang von Interesse. Detaillierte Informationen hierzu können abgefragt werden; sie sind als solche aber kein zentraler Bestandteil des Workshops.

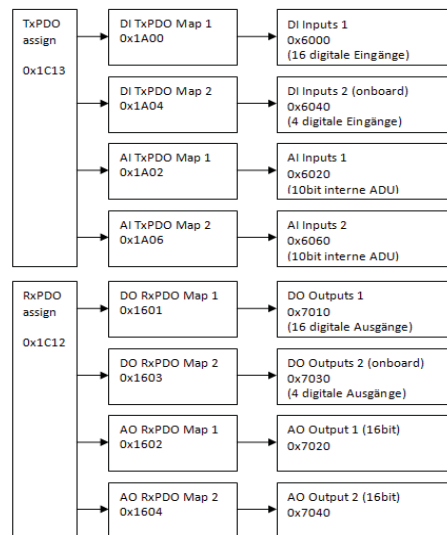


Bild 3 Beispielhaftes Data Dictionary für ein EtherCAT Device des HZDR

An die CANopen Spezifikation angelehnt stellt CANopen over EtherCAT Kommunikationsmechanismen zur Verfügung, wie sie von CANopen her bekannt sind. Damit werden Objektverzeichnis (Data Dictionary), Prozessdatenobjekte und Servicedatenobjekte transferiert. Mit Hilfe der Mailbox wird beispielsweise der Datenaustausch von

Servicedatenobjekten realisiert, der von verschiedenen EtherCAT Mastern zum Auslesen der Struktur des Data Dictionary genutzt wird.

Die Parametrierung von EtherCAT Devices lässt sich über den azyklischen Mailbox-Datenaustausch günstig verwirklichen. Da es möglich ist, die Parametrierung auf sinnvolle Device-Zustände einzuschränken und zudem ein zyklischer Austausch von Parametern unnötig Bandbreite benötigt, bietet dieser Datenaustausch einen sicheren und effizienten Realisierungsweg.

Wie bereits erwähnt ist die Absicherung der Systemkonformität ein wesentlicher Schritt bei der Entwicklung von Feldbusknoten. Nur bei konformer Realisierung lässt sich die beabsichtigt Nutzung der Funktionalität des Hersteller-eigenen Automatisierungssystems absichern. Im Falle von EtherCAT Slave Devices ist die Prüfung der Systemkonformität mit Hilfe des sogenannten Conformance Test Tools bei jedem Device-Hersteller hausintern durchführbar. Das Conformance Test Tool unterliegt ebenfalls der Systempflege und Weiterentwicklung und wurde während der am HZDR laufenden Entwicklung von V1.20.52 im Februar auf V1.20.62 aktualisiert.

Das Conformance Test Tool unterzieht das zu testende Gerät einem Black-Box-Test ohne die Notwendigkeit oder Möglichkeit einer Anpassung auf die spezifische Struktur der implementierten Prozessein- und -ausgangsgrößen. Das Hauptaugenmerk liegt auf einer Überprüfung der systemseitigen Schnittstellen in Bild 4. Hierzu wurden von der EtherCAT Technology Group über 9000 verschiedene Testcases implementiert, welche die Beschreibung des Feldbusknotens prüfen oder Telegramme an diesen versenden und mit erwarteten Antworten vergleichen.

Im ersten Schritt werden zur Identifikation der Geräte die Beschreibungsdateien der Geräte eingelesen (ESI = Electronic Slave Information). Diese werden nach Prüfen der Syntax und wichtiger Teile der Semantik im Cache der ESI-Dateien abgelegt und angezeigt. Ca. 125 Testcases entfallen auf diese Prüfung; etwa weitere 127 auf die Prüfung des in der ESI angegebenen Kommunikationskanals zum EtherCAT Device (Data Link Layer).

■ Conformance Test Tool => CTT Tests

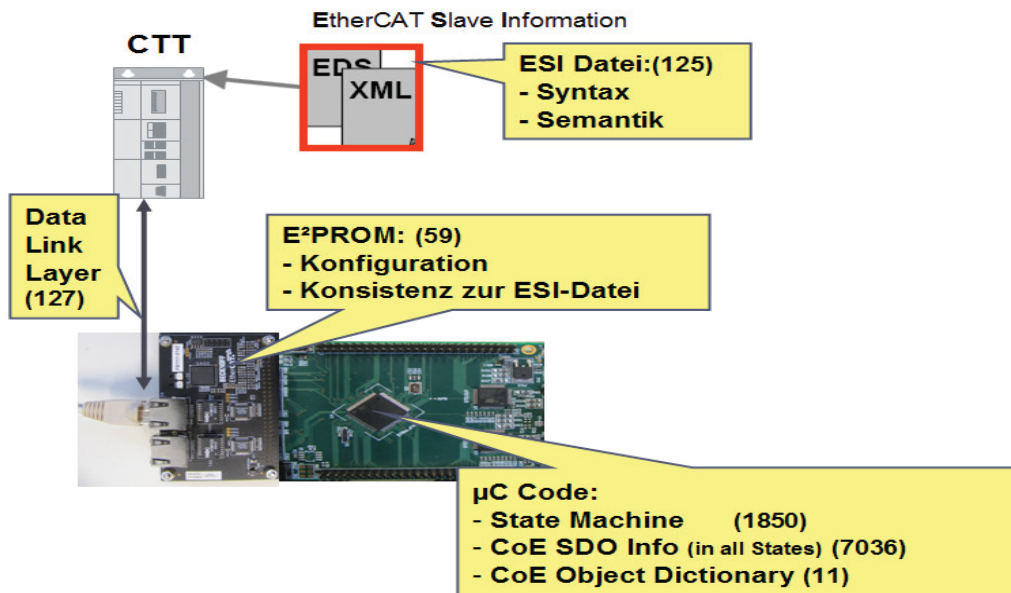


Bild 4: Anwendung des Conformance Test Tool

Im E²Prom sind wichtige Informationen zur Identität des Gerätes (Vendor ID, Product Code, Revision Number, ggf. Serial Number) und zur Adressierung der Kommunikationskanäle für synchronen und asynchronen Datentransfer hinterlegt. Diese werden ausgelesen und mit den Angaben der zugehörigen ESI-Datei im ESI-Cache verglichen.

Mit Hilfe dieser Zuordnung kann zunächst der Zustandsautomat des EtherCAT Device überprüft werden. Die hohe Anzahl der Testcases (1850) belegt die Sorgfalt, mit der an dieser Stelle auf Konformität geachtet wird. Falls CANopen over EtherCAT implementiert ist, wird eine Reihe von Prüfungen durchgeführt, welche Schreib- und Lesezugriffe auf Servicedatenobjekte in den unterschiedlichen Zuständen veranlassen. Abschließend erfolgen einige Zugriffe auf das Objektverzeichnis.

Beim ersten Testlauf unter Verwendung des Conformance Test Tool wurde bei unserer Implementierung ein Fehler während des Herunterladens eines Servicedatenobjektes festgestellt. Auch bei guter Gliederung und detaillierter Angabe von Randbedingungen des Testcase im Conformance Test Tool kann die Auswertung von Black Box Tests und die Beseitigung von Fehlern schwierig sein, da eine weitere Detaillierung von Informationen naturgemäß nicht möglich ist. In dem in Bild 5 gezeigten Fall kam es unter bestimmten Randbedingungen zur zeitlich nicht korrekten Bearbeitung von Interruptserviceroutinen.

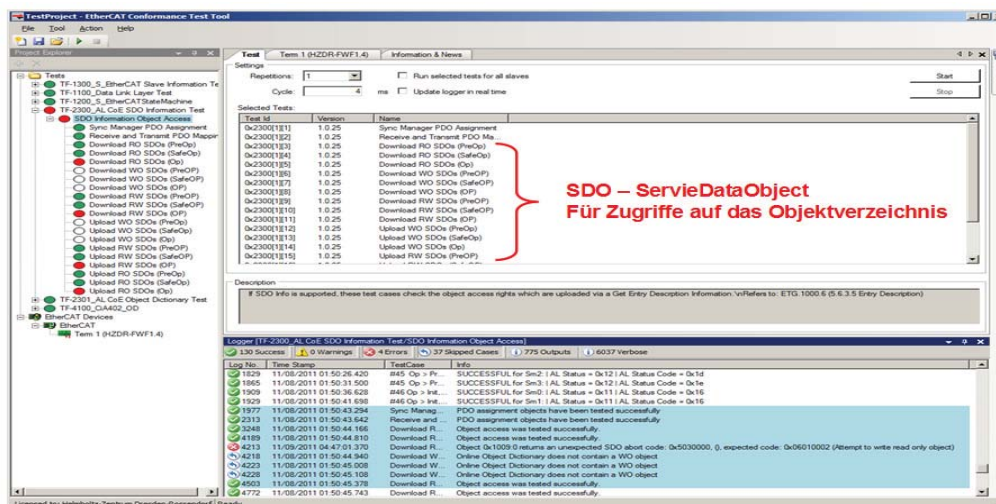


Bild 5: Einsatz des Conformance Test Tool bei einem fehlerhaften Slave Device

Nach Korrektur dieses Fehlers wurden alle Testcases erfolgreich durchlaufen. Das Conformance Test Tool erlaubt darüber hinaus auch die Überprüfung einer Reihe für die Entwicklung wichtiger Informationen. Bild 6 zeigt in dessen Hauptfenster neben den links angeordneten Testcases rechts eine Reihe von Karteikarten, die unterschiedliche Aspekte des Slave Device zeigen. Diese reichen von Inhalten des E²Prom über Inhalte der ESI-Datei, den Mailboxadressen sowie deren Länge und Möglichkeiten zur Anzeige von CANover EtherCAT Kommunikation und Prozessdaten.

Bei der Entwicklung des Slave Device war neben der Realisierung von Interrupt Service Routinen eine weitere Hürde beim Erstellen einer korrekten ESI Datei zu nehmen. Die Dokumentation [4] zur Beschreibung der ESI-Datei enthält auf 92 Seiten eine Fülle von Informationen, wie die einzelnen Teile der ESI-Datei aufgebaut sind. Diese Spezifikation detailliert in der Art eines Nachschlagewerkes oder einer Prüfvorschrift die einzelnen Elemente, kann aber aufgrund der Fülle möglicher Geräte keine vollständige Implementierungsempfehlung für Einzelgeräte geben.

Bei vorliegender Gerätebeschreibungsdatei ist deren Prüfung anhand der Spezifikation möglich, aber in unserem Falle bei einer Länge von etwa 900 Zeilen zeitraubend. Fehler bei der Erstellung

fürten zeitweise zur Instabilität der Testumgebung, welches erst nach Löschen von Konfigurationsdateien wieder funktionsfähig war. Zur Prüfung einer ESI-Datei und des EtherCAT Slave Device stellt das Conformance Test Tool eine wertvolle Hilfe dar. Nebenbei sei erwähnt, dass die EtherCAT Technology Group für die Inverkehrbringung von Geräten einen erfolgreich bestandenen Test vorschreibt.

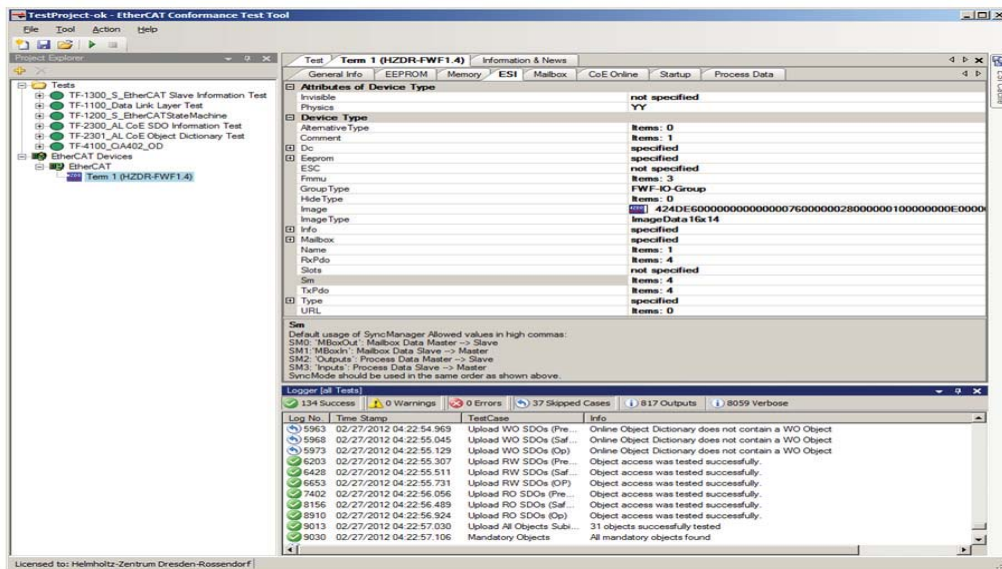


Bild 6: Informationen im Conformance Test Tool

Anhand eines Vergleiches mit Geräten von Beckhoff war eine rasche Anpassung zur Darstellung von Slave Devices des HZDR in der ESI Datei möglich. Diese ist in Bild 7 zu sehen.

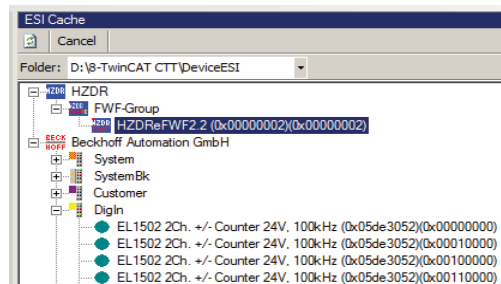


Bild 7: Hierarchische Anordnung von HZDR Feldbusknoten im Conformance Test Tool

Mit Erstellung einer funktionsfähigen ESI-Datei wurde die Entwicklung des Feldbusknotens zunächst abgeschlossen, da hiermit alle wesentlichen Kriterien für einen erfolgreichen Einsatz erfüllt waren:

- Registrierung des Gerätes mit Vendor-Id, Product Code, Revision Number
- Zyklischer Transfer der Prozeßdaten in verschiedenen Zugriffsmodi
- Azyklischer Transfer von Mailboxdaten, z.B. zum Auslesen des Objektverzeichnisses
- Einbinden in OPC – Server
- Einbinden in SPS – Steuerung mit TwinCAT PLC Control

Offen bleibt die Anbindung an andere EtherCAT Master um Anwendern einen leichteren Zugang zur Programmierung zu ermöglichen. Wichtig sind insbesondere die Kopplung zu LabVIEW zur graphischen Programmierung und die Anbindung an den Kithara Master Monitor zur alternativen Verwendung unter Windows. Hier besteht zur Zeit noch Klärungsbedarf, da diese EtherCAT Master unsere Geräte nicht verwenden konnten.

Abschließend sei bemerkt, dass im aktuellen Feldbusknoten die Ankunft von EtherCAT-Frames zur Übertragung der Prozessdaten und der Timer-Interrupt die einzigen Ereignisse im Microcontroller sind. Komplexere Verarbeitungsfunktionen und ggf. darüber hinausgehende Funktionalität zur Übertragung von Massendaten sollen unter Beibehaltung einer klaren internen Strukturierung durch ein Betriebssystem implementiert werden.

Am Helmholtz-Zentrum Dresden-Rossendorf werden aktuell auch Profinet Geräte entwickelt. Siemens bietet ERTEC-Chips zur Realisierung von Profinet Devices an, die intern einen ARM9 verwenden und bis zu Profinet IRT [6] einsetzbar sind. Eine bestehende freie Portierung von ECOS für diesen ARM9 motiviert dessen Auswahl als Betriebssystem auch für EtherCAT Slave Devices. Die Anbindung eines Ethernet-Anschlusses ist dabei in ECOS bereits vorgesehen und erlaubt perspektivisch die Übertragung von Massendaten auf einem separaten Kanal.

Auf diese Weise ist unter Nutzung gleichartiger Module die Realisierung von Feldbusknoten für EtherCAT und Profinet möglich, was den Entwicklungs- und Pflegeaufwand minimiert.

Quellen:

- [1] **Meyer, M.** *Aufbau eines EtherCAT-Slave mit Beckhoff ASIC und Integration in TwinCAT*; Beleg zum praktischen Studiensemester Fachbereich Elektrotechnik der Hochschule für Technik und Wirtschaft Dresden (FH); 4.3.2011
- [2] **Kaever, P.** *EtherCAT for Science – Entwicklung von Feldbusknoten für den wissenschaftlichen Gerätebau*; 102. Tagung der Studiengruppe elektronische Instrumentierung im Frühjahr 2011 DESY-PROC-2011-02 ISBN 978-3-935702-58-4
- [3] **Beckhoff Automation GmbH**: *Hardware Data Sheet ET1100 - EtherCAT Slave Controller*. www.beckhoff.com
- [4] **Beckhoff Automation GmbH**: *ETG2000_S_R_V1i0i2_EtherCATSlaveInformationSpecification.pdf*
- [5] **EtherCAT Technology Group**: *EtherCAT Slave Information Specification*. Mai 2009. www.ethercat.org
- [6] **Siemens Profinet IRT**
www.automation.siemens.com/mcms/automation/de/industrielle-kommunikation/profine
- [7] **B&R Ethernet Powerlink** www.br-automation.com
- [8] **Rockwell EtherNet/IP™**
www.ab.rockwellautomation.com/Networks-and-Communications/Ethernet-IP-Network
- [9] **EtherCAT Seminar: EtherCat Technology Basics for Developers**
- [10] **EtherCAT Evaluation Kit Workshop (Hardware & Software)**

S
E
I 12.-14.03.2012



Simplify your LabVIEW-EPICS communication

© 2012 Helmholtz-Zentrum Berlin

HZB Helmholtz
Zentrum Berlin

S
E
I 12.-14.03.'12

Überblick

HZB Helmholtz
Zentrum Berlin

1. Wozu ein weiteres LabVIEW™-EPICS Interface?
2. Die technische Umsetzung
3. CA Lab in der Praxis



von Carsten Winkler und Jens Kuszynski

2

Motivation

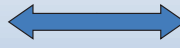
- Gastwissenschaftler nutzen LabVIEW™
- Beamline z.T. via EPICS steuerbar
- Betriebsstatusinformationen via EPICS
- Unkomplizierte EPICS-Schnittstelle aus LabVIEW™ gesucht
- Ohne Zusatzkosten



von Carsten Winkler und Jens Kuszynski

3

Schnittstelle



EPICS I/O Client



LabVIEW™ Channel Access Client



von Carsten Winkler und Jens Kuszynski

4

Auftretende Probleme

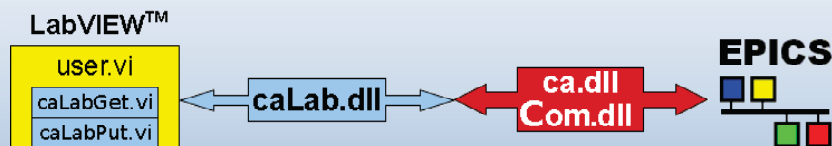


Bisherige Lösungen haben alle einen Haken

- ✘ Abhängigkeit von externen Services
- ✘ Fehlerträchtig für Neueinsteiger
- ✘ Kostenlose Schnittstellen sind unvollständig implementiert (Datentypen, Datenfelder, Enumerations)



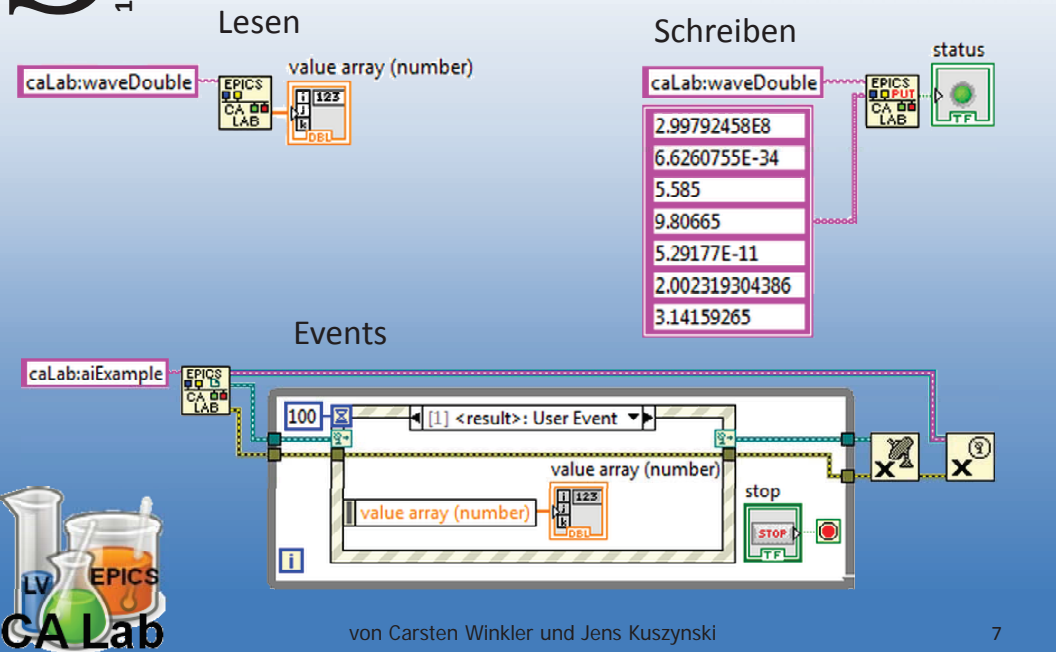
CA Lab



- ✔ Baut direkt auf aktueller EPICS Base auf
- ✔ Support aller EPICS-Datentypen
- ✔ Beliebig viele konsistente Felder pro Variable
- ✔ LabVIEW™ 7.0 bis 2011 unter Windows® und Linux
- ✔ Minimale Netzwerkbelastung durch Monitoring und Caching
- ✔ Open Source



Einfache Umsetzung



CA Lab - Eigenschaften

Arrays ⇒

Konsistenz ⇒

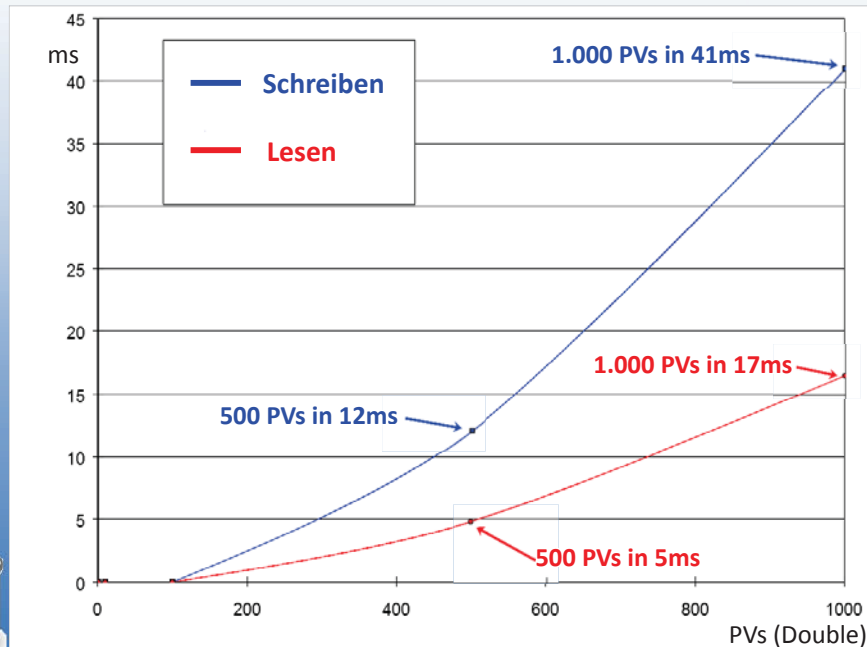
Enumeration

PV names	results	PV name	Elements	PV name	Elements	PV name	Elements
MyMotorRecord	MyMotorRecord 1	MyEnumeration	1	MyArray	2048		
MyEnumeration	value array(string)						
MyArray	value array (#)						
Fields	status (string) status (#)						
DESC	NO_ALARM 0						
VELO	severity (string) severity (#)						
MISS	NO_ALARM 0						
ACCL	time stamp (string)						
	1930-02-11 01:35:40.885898						
	time stamp (number)						
	-1889911460						
	field names field values						
	DESC						
	VELO 0.8						
	MISS 0						
	ACCL 2						
	error IO						
	status code						
	source						
	Normal successful completion						

von Carsten Winkler und Jens Kuszynski



Performance



von Carsten Winkler und Jens Kuszynski

9

Vorteile

- Einfacher Aufbau
- Variablendefinition und Funktionsbaustein beieinander
- Übergabe von Variablenarrays
- Konsistenz durch Clusterung der Ergebnisse
- Automatische Auflösung von Enumerations
- Datentypen werden automatisch erkannt (→Mischung möglich)
- Sehr gute Performance



von Carsten Winkler und Jens Kuszynski

10

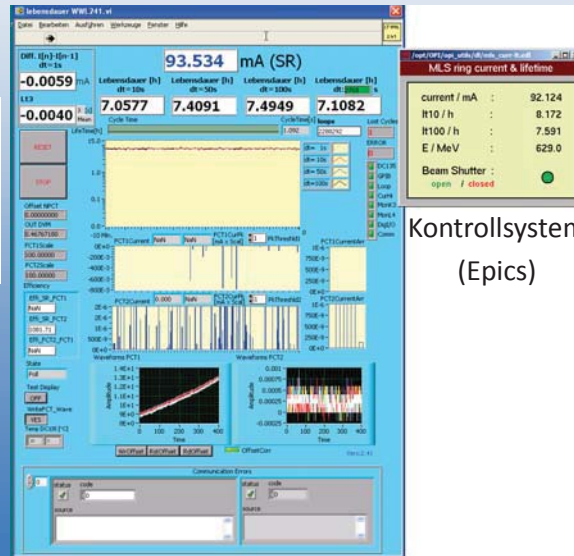
SEI
12.-14.03.'12

Praxis: Epics + LabVIEW™ DSC **HZB** Helmholtz Zentrum Berlin

PXI Chassis
Controller
Einsteckkarten
GPIB DMM



LabVIEW™ Programm „Strommonitor“



Kontrollsystem
(Epics)

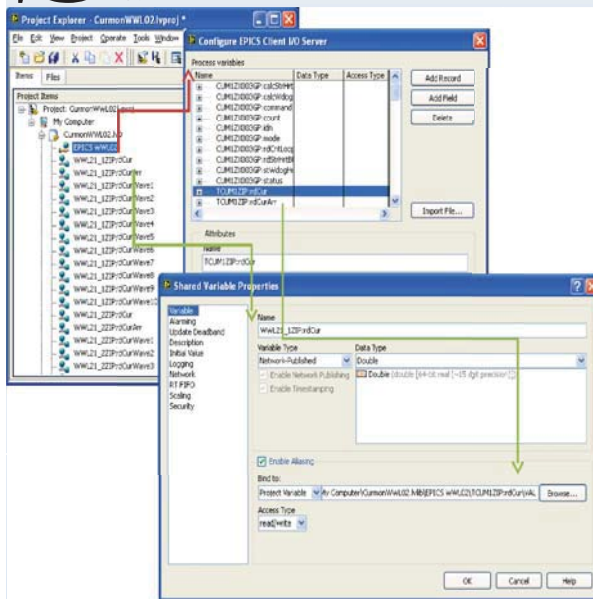


von Carsten Winkler und Jens Kuszynski

11

SEI
12.-14.03.'12

Praxis: Epics + LabVIEW™ DSC **HZB** Helmholtz Zentrum Berlin



Was wird benötigt?

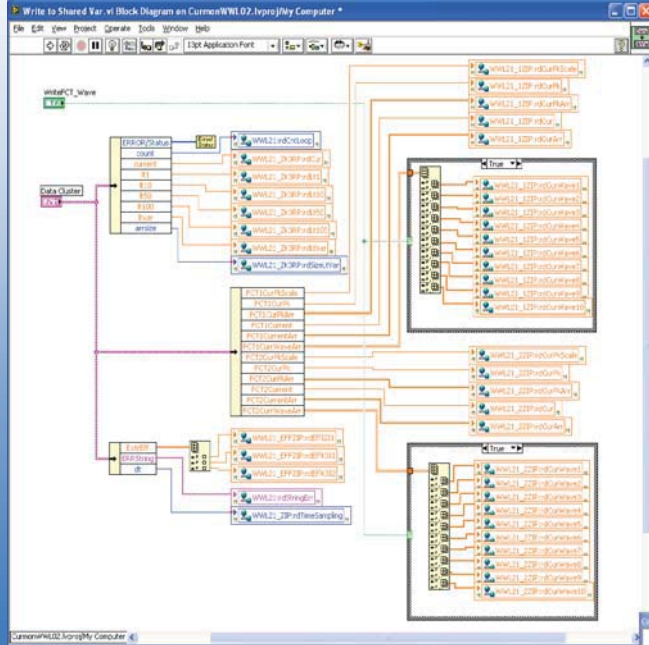
- ein Projekt
- LabVIEW™ Shared Variablen
- Epics Client
- DSC Runtime für EXE



von Carsten Winkler und Jens Kuszynski

12

Übergabe der LabVIEW™ Daten

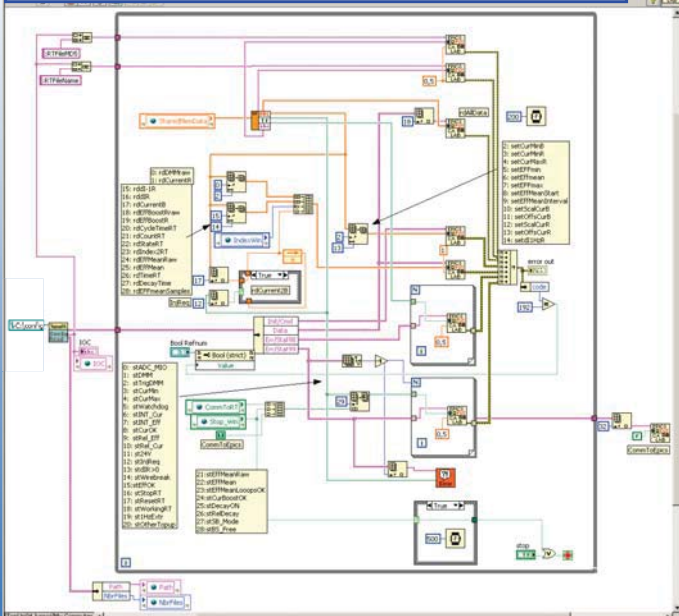


von Carsten Winkler und Jens Kuszynski

Initialisierungsfile

```
[SM]
IOC = "TOPUP1TSG"
[Init/Cmd]
setCurrMinB = Keyname
setCurrMax = Keyname
setCurrMinR = Keyname
setEffMin = Keyname
setEffMax = Keyname
setEffMeanStart = Keyname
setEffMeanInterval = Keyname
setScalCurB = Keyname
setOffScurB = Keyname
setScalCurR = Keyname
setOffScurR = Keyname
setdIHzR = Keyname
[data]
rdDmRaw = Keyname
rdCurrEntR = Keyname
rdIIR = Keyname
rdDdr = Keyname
rdCurrEntB = Keyname
rdEffBoosterRaw = Keyname
rdEffBooster = Keyname
rdCycletimeRT = Keyname
rdCountRT = Keyname
rdStatERT = Keyname
rdIndex2RT = Keyname
rdEffMeanRaw = Keyname
rdEffMean = Keyname
rdTimeRT = Keyname
rdDecayTime = Keyname
rdEffMeanSamples = Keyname
rdIndexwin = Keyname
rdCurrent2B = Keyname
rdAllData = Keyname
[Err/stat]
stADC_MIO = Keyname
stDM = Keyname
stTrigPM = Keyname
stCurFin = Keyname
stCurMax = Keyname
stWatchdog = Keyname
stINT_Cur = Keyname
```

Übergabe der LabVIEW™ Daten (TopUp Effizienzrechner)



von Carsten Winkler und Jens Kuszynski

S
E
I
12.-14.03.'12

CA Lab Vorteile gegenüber DSC aus Anwendersicht

HZB Helmholtz
Zentrum Berlin

- Keine zusätzlichen Kosten durch DSC (und Runtime)
- Einfachere Handhabung während der Programmentwicklung
- Klonen bzw. Überführen der Programme auf andere Rechner ist einfacher, da meist nur die (ASCII-) Initialisierungsdatei angepasst werden muss
- Eigenschaften der Epics PVs werden besser unterstützt

Nachteil:

- LabVIEW™ RealTime wird zur Zeit nicht unterstützt (ein einfacher Epics Client und Epics Server ist im RealTime Modul enthalten)



<http://tinyurl.com/calab2012>

von Carsten Winkler und Jens Kuszynski

15

Customized Off-The-Shelf Technologies for building system platforms for Big Physics experiments

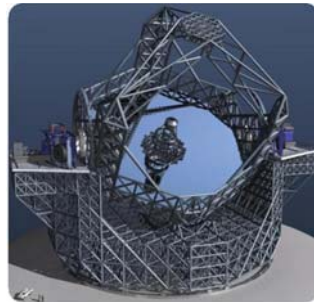
Leif Johansson – European Segment Manager Big Physics



Science Big Physics System



**ITER - Plasma Diagnostics &
Control**



**ESO Extremely Large Telescope -
Mirror Control**

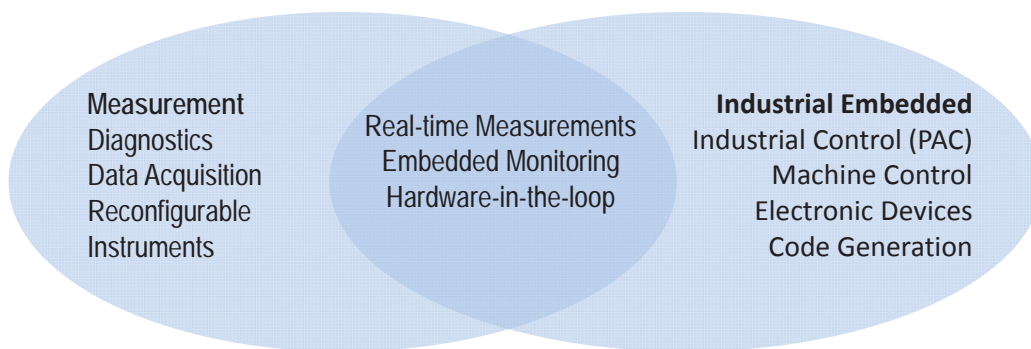


Today's Engineering Challenges

- Doing more with less
- Time to experiment
- Managing global projects
- Adapting to evolving application requirements
- Delivering on increasingly complex initiatives
- Maximizing operational efficiency
- Protecting system and resource investments



The National Instruments Vision, Evolved... Graphical System Design

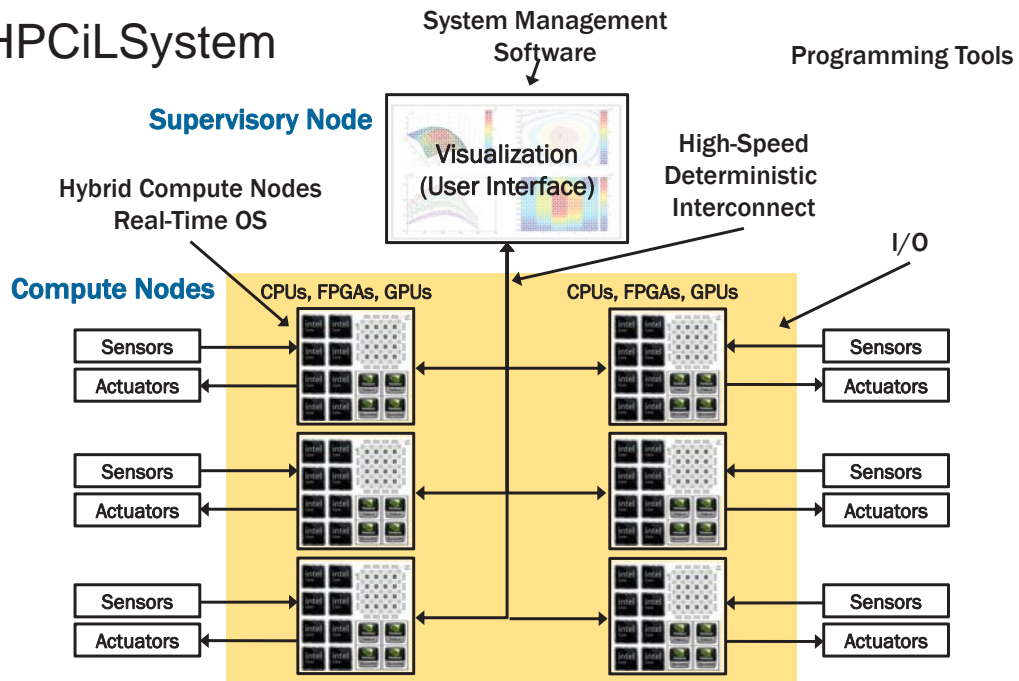


"To do for test and measurement
what the spreadsheet did
for financial analysis."

"To do for embedded what the PC
did for the desktop."

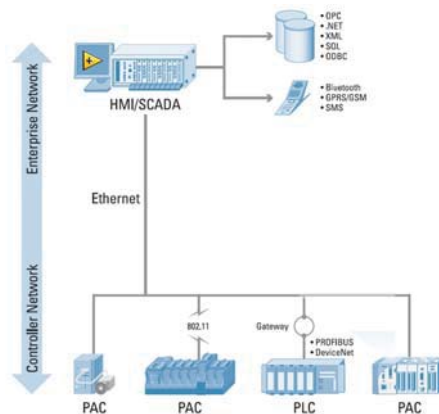


HPCiLSystem

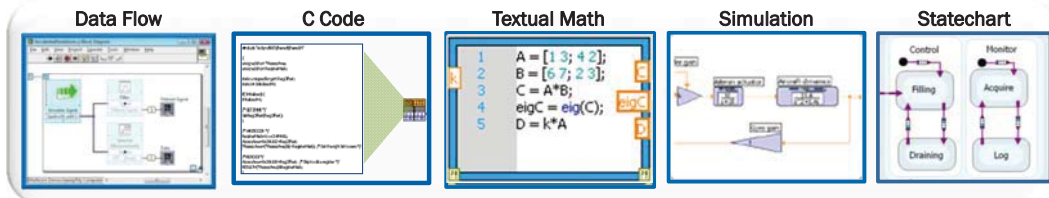


Open Architecture

- Controls standards
 - EPICS, TANGO, CORBA, TINE, C
- Connectivity to different devices
 - OPC, Modbus, TCP/IP, UDP, EtherCAT, Serial
- Flexibility
 - Windows, Linux, RTOS, FPGA



High-Level Design Models



Graphical System Design Platform



OFF-THE-SHELF TECHNOLOGIES LEVERAGING INDUSTRY



Leveraging Industry Relationships

- Apply technologies from wide array of vendors
 - Next generation FPGAs, ADCs, GPUs and processors
- High access to information
 - Regular executive meetings
 - Ability to influence roadmaps

 XILINX

 TEXAS INSTRUMENTS

 ANALOG DEVICES



 NATIONAL INSTRUMENTS

Putting it together.....



Embedded
Controller
(Processor)



Chassis with T&S
(Communication
Bus)



I/O Modules
(ADC/DAC)

 NATIONAL INSTRUMENTS

PXI Platform

PXI Combines Standard Technologies

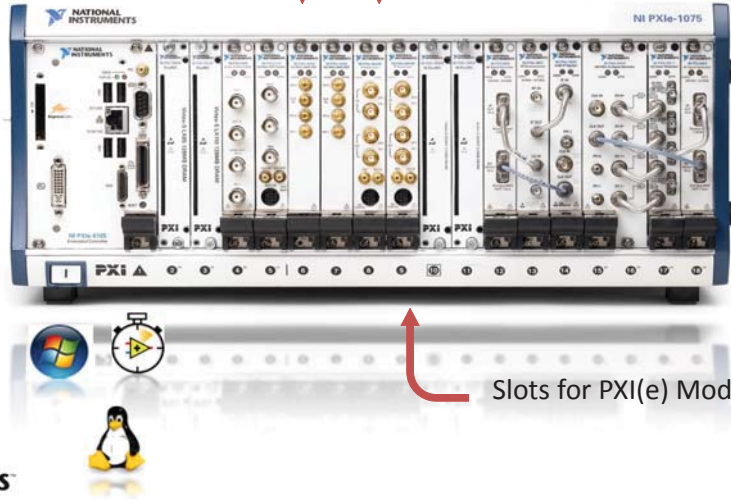
- PXI Controller
- Embedded PC
 - Remote PC

- PXI Chassis
- Ventilation
 - Supply

- PXI Backplane
- PCI or PCI Express Bus
 - Synchronization / Timing

PXI
Systems Alliance

www.pxisa.org



NATIONAL INSTRUMENTS

PXI System to – PXI Express

PXI Express System

PXI Controller



PXI Chassis

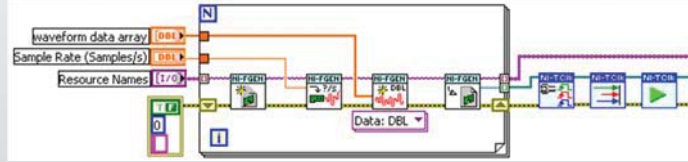


PCI Bus
132 MB/s

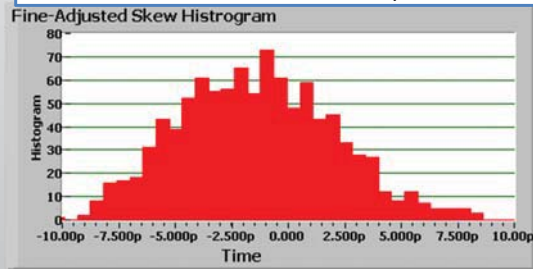
Four x4 PCI Express Links
4 GB/s bidirectional

NATIONAL INSTRUMENTS

Example: Function Generator Synchronisation w/ NI-TCLK

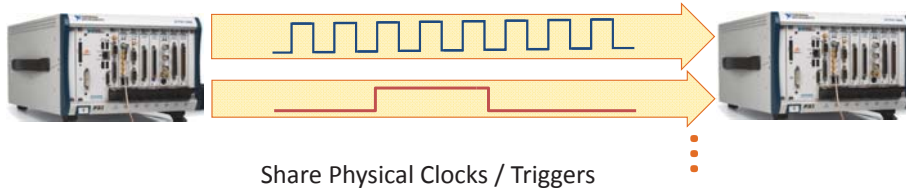


Result: Max. skew = 20 ps

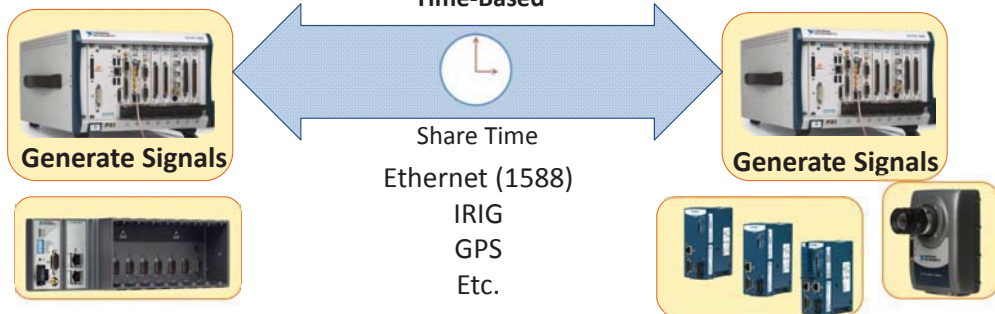


Signal vs. Time-Based Synchronization

Signal-Based



Time-Based



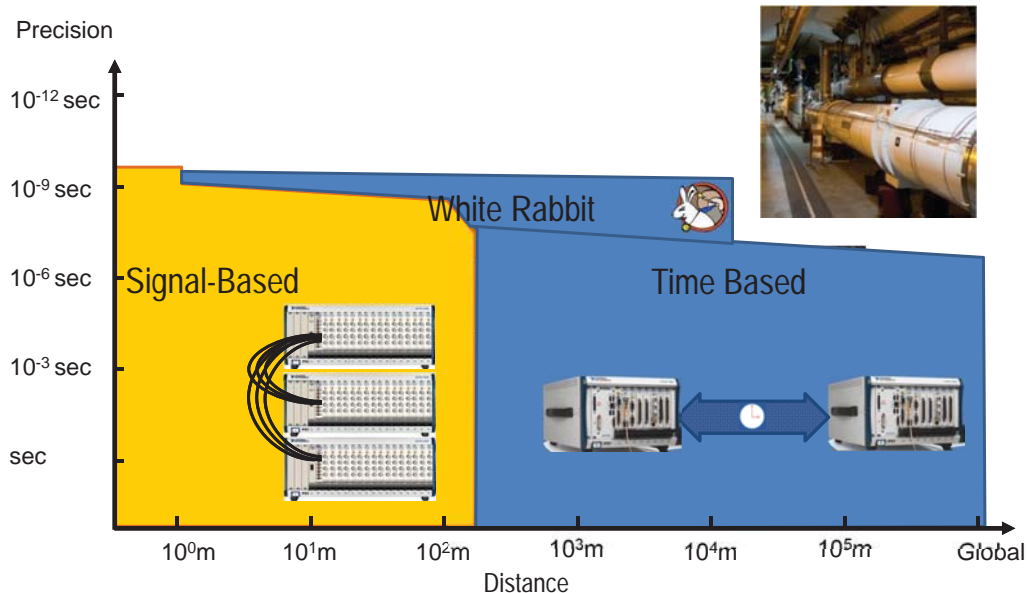
Ongoing project NI and CERN White Rabbit



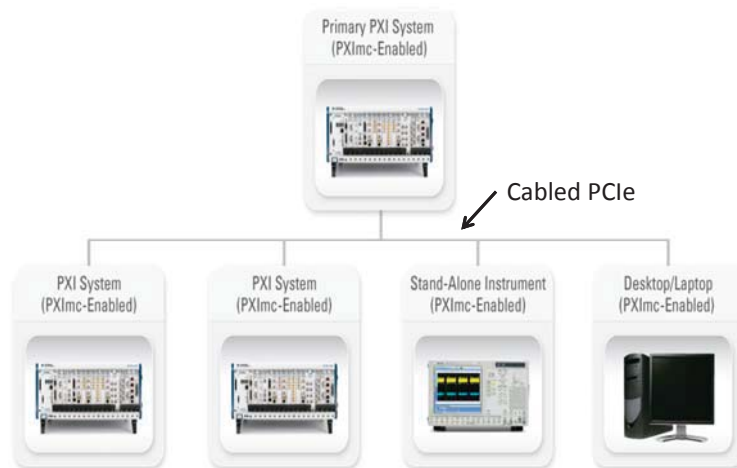
- Partnering with CERN in developing White Rabbit (WR)
- Performance
 - **Distance:** > 10 km
 - **Scale:** > 2000 nodes
 - **Accuracy:** < 1ns skew, < 100 ps jitter
 - Compensates for propagation delay (cable length, temperature variation, etc.)
- Leverage Industry standards (802.x, **IEEE 1588**, SyncE)
 - Gigabit Ethernet communication with deterministic capability
- Generally Applicable
- Leverage for future PXIe modules



White Rabbit: Synchronization over Distance

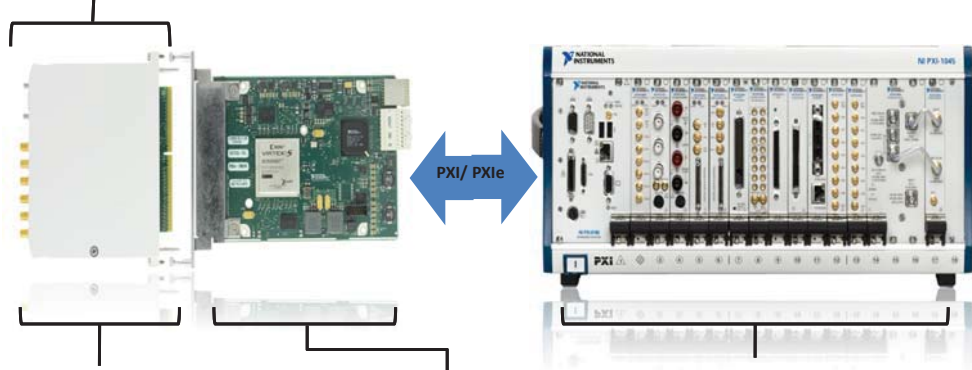
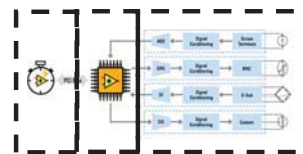


The Next Step PXI Multi-Controller (PXImc)



FPGA on PXI NI FlexRIO Architecture

Customizable
Front-End



NI FlexRIO Adapter Module

- Interchangeable I/O
- Digital or analog
- NI FlexRIO Adapter Module Development Kit (MDK)

NI FlexRIO FPGA Module

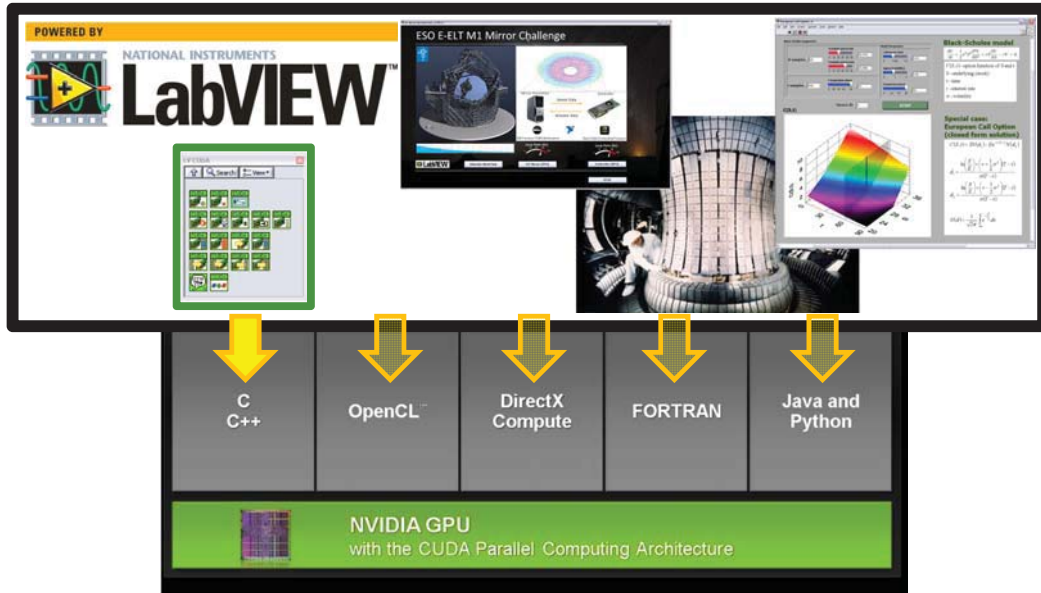
- Virtex-5 FPGA
- 132 digital I/O lines
- Up to 512 MB of DRAM
- Peer-to-peer data streaming

PXIe Platform

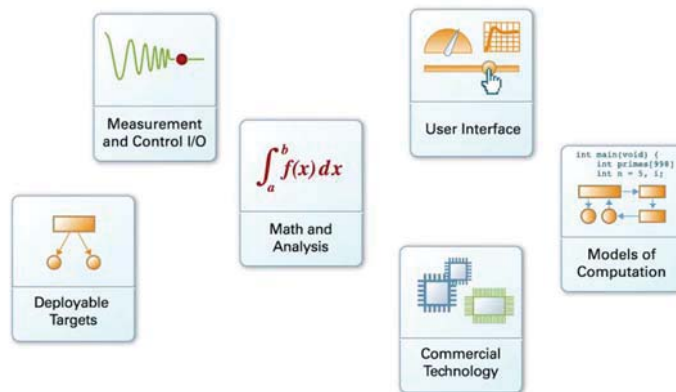
- Data transfer
- Synchronization
- Clocking/triggers
- Power/cooling



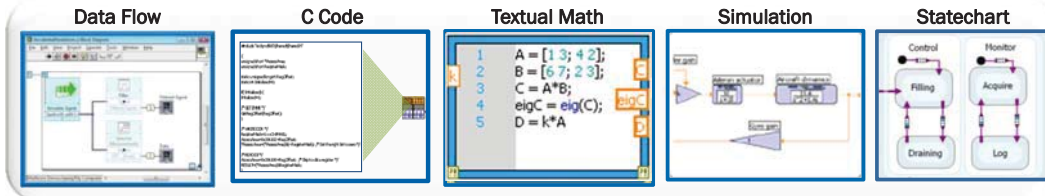
GPU Computing Module



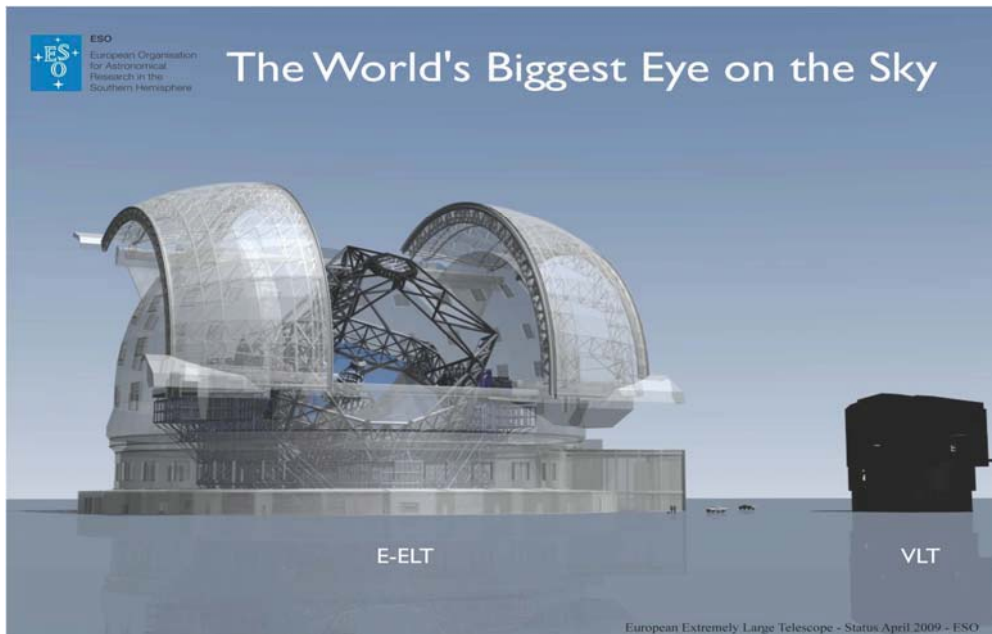
Integrating Elements



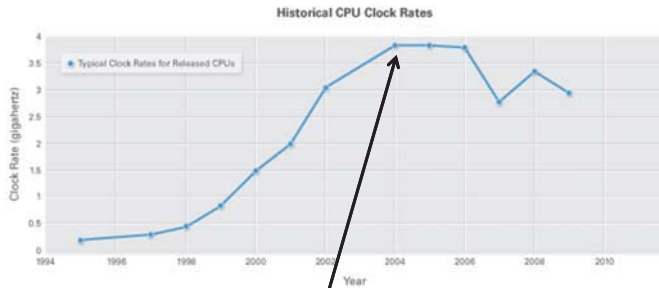
High-Level Design Models



Graphical System Design Platform



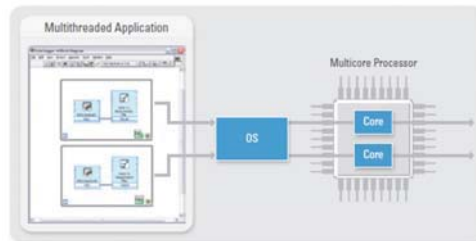
Making parallel computing work



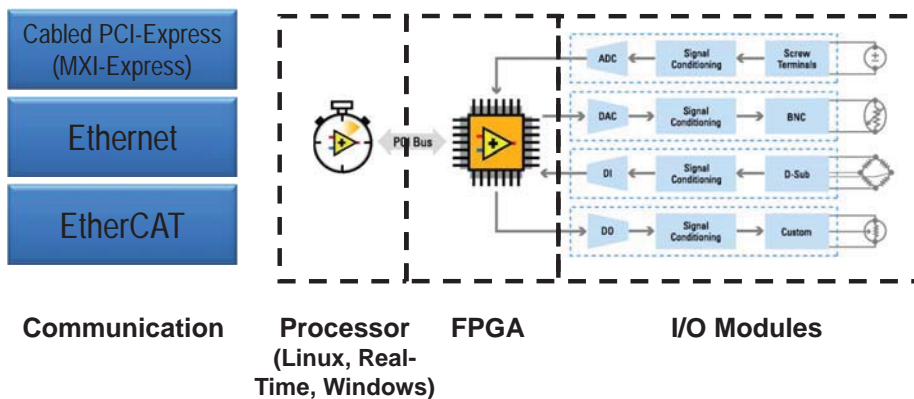
„The free lunch is over“

Herb Sutter
(Head of ISO C++ standardization body)

LabVIEW Automatic Core Assignment

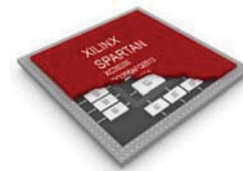
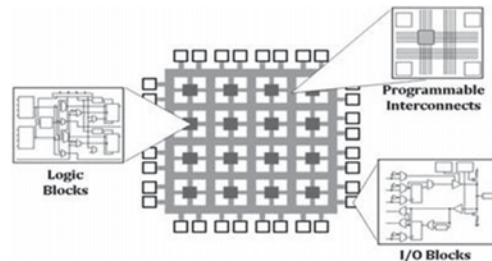


Combining COTS With Your Design: *RIO Architecture*



Benefits of FPGAs

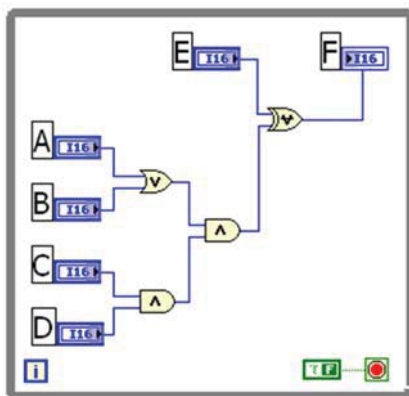
- 👍 Massively parallel
- 👍 Reconfigurable
- 👍 Digital signal processing
- 👍 High-speed control
- 👍 Faster time to experiment
- 👎 Typically require digital design expertise



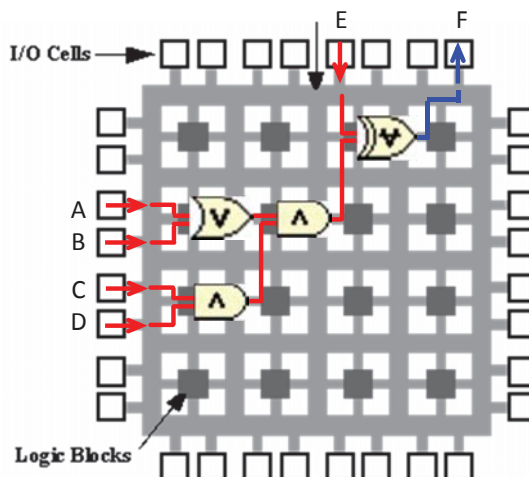
FPGAs are Dataflow Systems

Implementing logic

$$F = \{(A+B)CD\} \oplus E$$

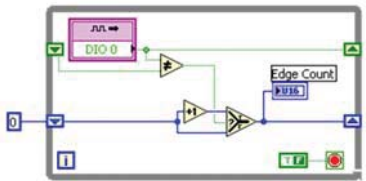


LabVIEW FPGA Code

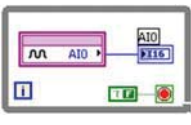


Graphical Programming and Abstraction

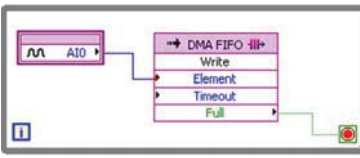
Counter



AIO



AIO with DMA



LabVIEW FPGA

VHDL

```



-- First we synchronize the asynchronous digital input to our clock
-- the output is the flip-flop
process async_clk
begin
    if async_clk'event then
        if async_clk = '0' then
            output <= '0';
        else
            output <= '1';
        end if;
    end if;
end process;

-- Then we take track of when the digital input was on the previous
-- clock cycle by toggling another flip-flop
process async_clk
begin
    if async_clk'event then
        if async_clk = '0' then
            output <= '0';
        else
            output <= '1';
        end if;
    end if;
end process;

-- Then we have a little combinational logic to detect a rising edge
-- and implement it on output and not asynchronous
and ff
begin
    output <= output xor (output and not async_clk);
end and ff;

-- and finally we have a register that increments when that rising
-- edge is detected
process async_clk
begin
    if async_clk'event then
        if async_clk = '0' then
            counter <= counter + 1;
        end if;
    end if;
end process;

-- and process CounterRegister
counter <= counter;
end process;
            
```


66 pages, approx. 4,000 lines



LabVIEW FPGA IP Integration Node

- 1

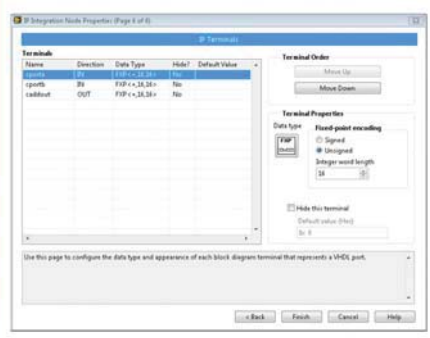
Use Core Generator or Custom VHDL



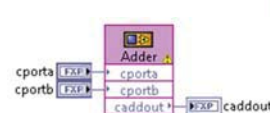
```

aclr : in  std_logic
clk  : in  std_logic
a    : in  std_logic
b    : in  std_logic
q    : out std_logic
            
```
- 2

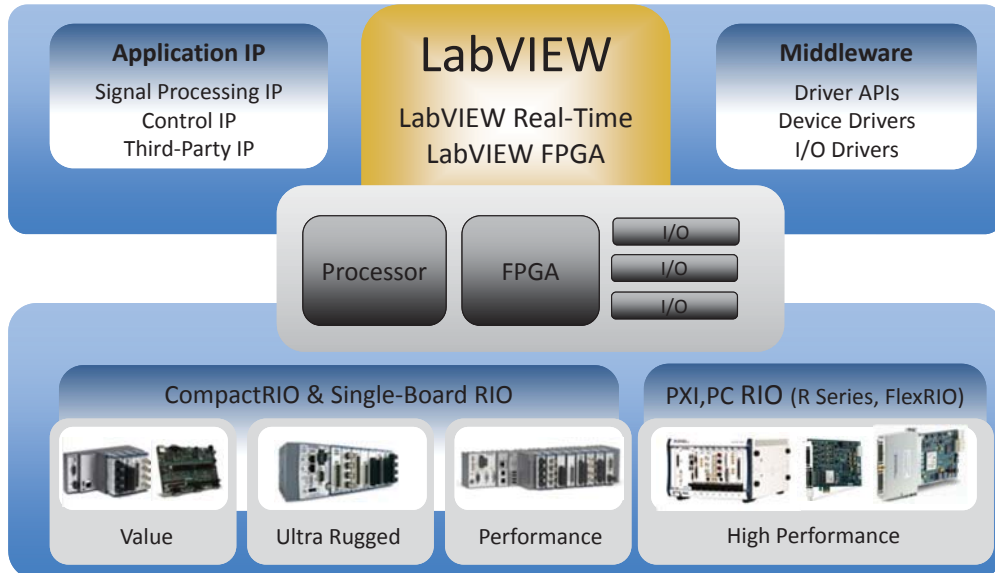
Configure IP Integration Node and Generate Simulation Model


- 3

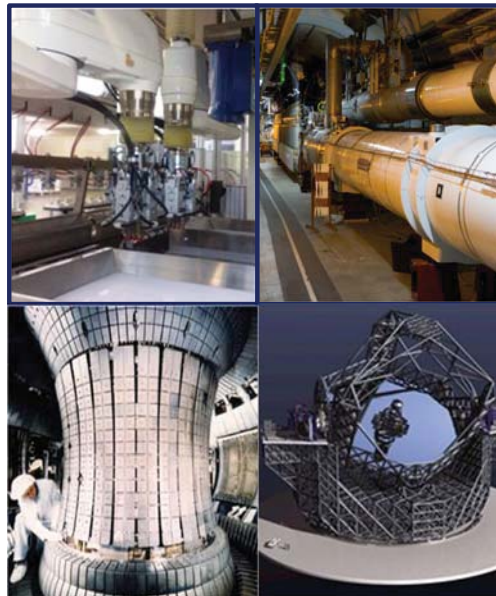
Use the IP Block Using Standard LabVIEW I/O Interfaces



NI Reconfigurable I/O (RIO) Technology Platform



Customized Off-The-Shelf Technologies





Industrielle Computer & Netzwerklösungen

www.systemerra.de

systemerra computer

systemerra
computer



Engineering & Consulting

- Beratung
- Machbarkeitsstudien
- Systemdesign
- Prototypenbau



systemerra computer GmbH

Kreuzberger Ring 22
65205 Wiesbaden
T: 0611 9748-470
E: info@systemerra.de
www.systemerra.de



Systemintegration

- Europas größtes embedded Produktspektrum
- OEM-Dienstleistungen
- Technische Dokumentation

Vertriebsbüros:

- Hannover
- Jena
- München



Life-Cycle-Management

- Revisionsüberwachung
- Ersatzteilversorgung
- Reparatur
- Nachproduktion



Qualitätssicherung

- ISO 9001:2008 zertifiziert
- 8D Reporting

www.systemerra.de

UEI DAQ-Lösungen

Produktspektrum



MicroTCA.4 Systeme



Robuste Box-PCs & Displays



Data Acquisition & Logging



Rugged Enterprise Server & Storage



VPX, VMEbus & CompactPCI



Ethernet Netzwerke und IP-Kameras

www.systemerra.de

UEI DAQ-Lösungen



Data Acquisition & Control

Messdatenerfassung, Datenlogger, Steuerung und Simulation

Dipl.-Ing. Stephan Hering und Dipl.-Ing Volker Thomas



United
Electronic
Industries



www.systemerra.de

4 kompakte Formfaktoren – 5 Basiskonfigurationen



Standalone Applikationen

- **UEILogger**
 - ▶ Auf PC konfigurierte Anwendung läuft standalone
- **UEIPAC**
 - ▶ Embedded I/O Anwendung läuft standalone unter LINUX auf Cube/Rack
- **UEISIM**
 - ▶ Mit Simulink/HIL erzeugte Anwendung läuft standalone auf Cube/Rack

Host basierte Applikationen

- **PowerDNA**
 - ▶ Datenerfassungs-Anwendung läuft auf Host Computer
- **UEIModbus**
 - ▶ Modbus TCP I/O-Anwendung läuft auf Host Computer

www.systemerra.de

UEI DAQ-Lösungen

PowerDNA Cube Architektur



Eigenschaften:

- Zwei Baugrößen: 3 oder 6 I/O Steckplätze
- PowerPC 8347@400 MHz, 128 MB RAM
- Dual Gigabit Ethernet, Dual USB 2.0
- SD-Karten Steckplatz
- Serieller Diagnose-Port
- SYNC Port (Karte zu Karte und Cube zu Cube)
- 9-36 VDC Speisung
- Low Power (ab 15 Watt), AC/DC-Betrieb
- Anbindung: 100 m via Ethernet, 20 km via LWL

www.systemerra.de

UEI DAQ-Lösungen

PowerDNA Cube Features

systemra
COMPUTER

Eigenschaften:

- Kompakt: 102 x 102 x 153 mm
- Hohe Kanaldichte mit bis zu :
 - ▶ 150 A/D, 192 D/A, 288 DIO,
 - ▶ 48 seriellen Schnittstellen
 - ▶ 72 ARINC-429
- Extrem Robust:
 - ▶ -40... +85°C Umgebungstemperatur
 - ▶ 5g Vibration, 50g Schock
 - ▶ MTBF > 34 Jahre
 - ▶ 21.300 m Betriebshöhe
 - ▶ Strahlungstestet für Raumfahrtanwendungen



CPU- und I/O-Karten werden als „Stack“ in das Cube-Chassis eingebaut.

www.systemra.de

UEI DAQ-Lösungen

RACKTangle und Half-RACK

systemra
COMPUTER

Eigenschaften:

- Basiert auf bewährter Technik der Cubes
- 6 oder 12 I/O Steckplätze
- Passive Backplane
- Temperaturregelte Lüfter
- CPU/NIC Karte mit GigE Cube identisch
- 19" Einbau möglich



DNR-6-1G



DNR-12-1G

www.systemra.de

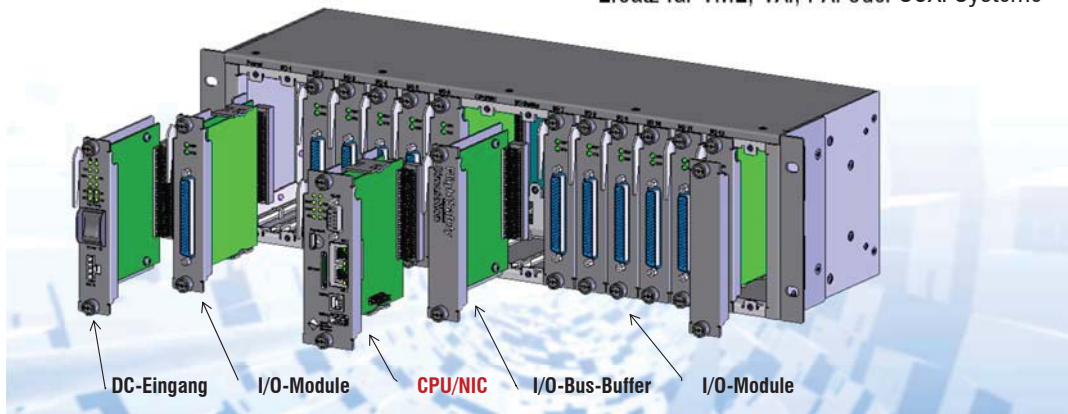
UEI DAQ-Lösungen

RACKtangle Features



Eigenschaften

- 3HE Aluminium Chassis:
- Geringe Masse: 12-Slot Chassis ~2,3 kg
- Dual Gigabit Ethernet, Dual USB 2.0
- PowerPC 8347@400 MHz, 128 MB RAM
- Hohe Kanaldichte mit bis zu:
 - ▶ 300 AIs, 384 AOs, 576 DIOs
 - ▶ 144 ARINC 429 Kanälen
 - ▶ 96 seriellen Ports
- MTBF = 130,000 h
- Ersatz für VME, VXI, PXI oder SCXI Systeme



www.systemerra.de

UEI DAQ-Lösungen

I/O-Module für Cubes und Racks



Eigenschaften:

- Cube und RACKtangle Systeme haben elektrisch identische I/O Karten
- Unterschiede nur in Busstecker/Montage
- Isolation pro Karte oder Kanal
- Integrierte Signal-Konditionierung
- Softwarekonfigurierbar
- 3 oder 6 I/O-Karten pro Cube und 6 oder 12 I/O-Karten pro RACKtangle Chassis
- Standard DB-37 oder DB-62 Stecker
- Beliebige I/O-Kartenkombinationen möglich



Cube I/O-Karte



RACKtangle I/O-Karte

www.systemerra.de

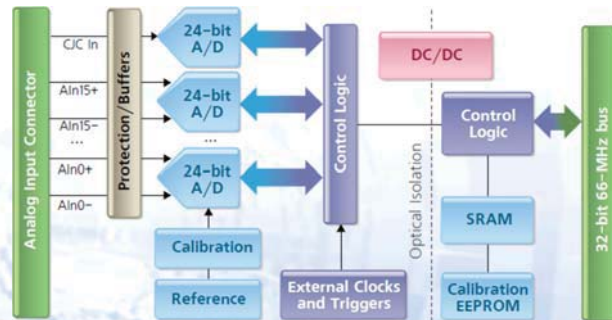
UEI DAQ-Lösungen

Analoge Eingangskarten



Für die PowerDNA I/O-Cubes und RACKtangle I/O-Chassis gibt es eine breite Auswahl an analogen Eingangskarten:

- Alle Karten sind werksseitig vorkalibriert und verfügen über bis zu 25 Kanäle mit max. 24-bit Auflösung und max. 250 kHz Abtastrate.
- Verfügbare Eingangskartentypen:
 - ▶ Universal Eingänge
 - ▶ Thermoelement Eingänge
 - ▶ Dehnungsmesstreifen
 - ▶ LVDT/RVDT
 - ▶ ICP/IEPE
 - ▶ Synchro/Resolver
 - ▶ Simultane Abtastung
 - ▶ Hohe Abtastrate
 - ▶ Hochauflösend



Beispiel: 16-Kanal 24-bit Analog-Eingangskarte

www.systemerra.de

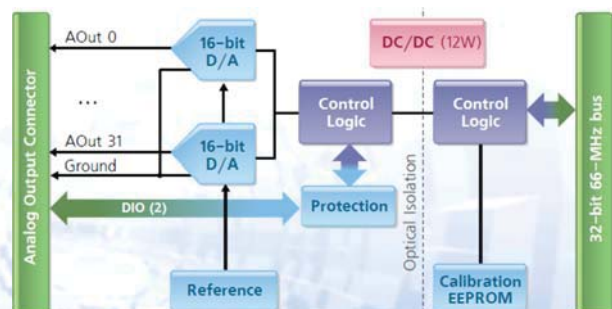
UEI DAQ-Lösungen

Analoge Ausgangskarten



Analoge Ausgangskarten bieten bis zu 32 Kanäle und sind werksseitig vorkalibriert:

- Es stehen zahlreiche Varianten mit unterschiedlichen Auflösungen und Ausgangsspannungen/Strömen zur Verfügung:
 - ▶ ± 10 V
 - ▶ 4...20 mA
 - ▶ Hohe Ausgangsspannungen
 - ▶ Hohe Ausgangsströme
 - ▶ Externe Pufferverstärker



Beispiel: 32-Kanal 16-bit Analog-Ausgangskarte

www.systemerra.de

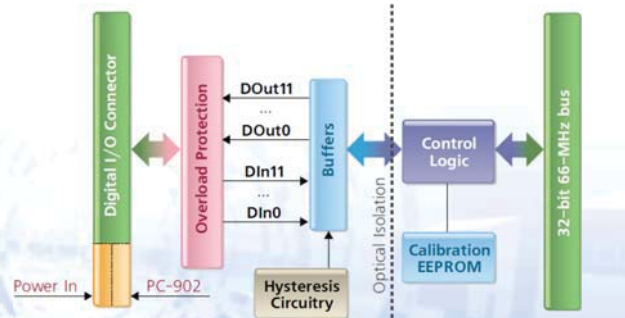
UEI DAQ-Lösungen

Digitale I/O-Karten



Die Palette der digitalen I/O-Karten umfasst neben reinen Ein- und Ausgängen auch intelligente Zähler/Timer und Encoder-Karten:

- Hohe I/O-Dichte
- Industrielle I/O-Pegel
- Hochstrom
- PWM-Ein- und Ausgänge
- Zähler/Timer
- Quadratur Encoder
- LVDT/RVDT
- Synchro/Resolver
- Relais



Beispiel: 24-Kanal Digital-I/O-Karte

www.systemerra.de

UEI DAQ-Lösungen

Serielle I/O-Karten

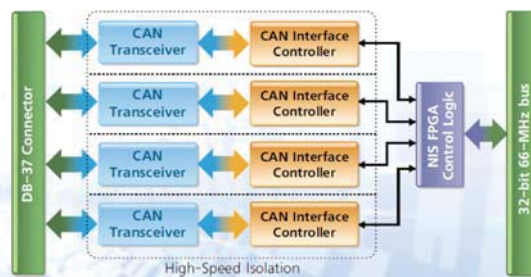


CANbus Schnittstellen:

- DNx-CAN Karten sind elektrisch zu on-board Bussystemen von Kraftfahrzeugen kompatibel
- Support durch UEIDAQ Framework-Bibliothek
- Support für alle gängigen Betriebssysteme und Programmiersprachen inkl. LabVIEW/DASYLab

Serielle Schnittstellen:

- Asynchrone und synchrone RS-232/422/485 Schnittstellenkarten.
- Im synchronen Betrieb werden SDLC, HDLC und generische Clk/Data Konfigurationen unterstützt
- Asynchron bis zu 4 Mbaud



Beispiel: 4-Kanal CANbus-Karte

www.systemerra.de

UEI DAQ-Lösungen

Intelligente I/O-Karten



Avionik Schnittstellen

- AFDX / ARINC664
- MIL-1553, ARINC-429, ARINC-708

Navigation, Position und Zeit

- GPS
- IRIG-B Slave oder Master

Drahtlose Kommunikation

- WLAN und GSM Module

FPGA

- "Custom" FPGA Karten

DC/DC Konverterkarten

- Hilfsenergieerzeugung für Ausgangskarten

Guardian Serie

- Analoge und digitale Ausgänge mit Spannungs-/Stromrücklesefunktion



www.systemerra.de

UEI DAQ-Lösungen

Zubehör



Zum Anschluss von Ein- und Ausgängen, der Signalkonditionierung und der Synchronisation steht eine umfangreiche Zubehörpalette zur Verfügung:

- Anschlussterminals
- Signal- und Datenkabel
- Steckverbinder
- Stromversorgung
- Pufferverstärker
- Synchronisationskarten
- Befestigungs-Kits



www.systemerra.de

UEI DAQ-Lösungen

Software Basiskonfigurationen



Autonome Systeme

- **UEI PAC**
 - ▶ Linux-basierte Automationssteuerung
 - ▶ Cube- oder RACKTangle-Bauform
 - ▶ Standard Linux 2.6 OS mit Echtzeit-Kern
 - ▶ Bis zu 576 KHz Abtastrate
- **UEI Logger**
 - ▶ Standalone oder Netzwerk-Datalogger
 - ▶ Cube-Bauform
 - ▶ Bis zu 1000 Samples/s pro Kanal
- **UEI SIM**
 - ▶ Matlab/Simulink RTW-Target-System
 - ▶ Cube- oder RACKTangle-Bauform
 - ▶ Standard Linux 2.6.x Betriebssystem
 - ▶ Target für Matlab/Simulink Entw.-Umgebung
 - ▶ Bis zu 5 kHz Abtastrate

Host gesteuerte Systeme

- **PowerDNA/DNR**
 - ▶ Messen, Steuern, Regeln via Ethernet
 - ▶ Cube- oder RACKTangle-Bauform
 - ▶ Echtzeitfähig: < 1 ms Abtastzeit für 1000 I/Os
 - ▶ LabVIEW, MATLAB, DASYLab Support
- **UEI Modbus**
 - ▶ Modbus/TCP I/O-Lösung
 - ▶ Cube- oder RACKTangle-Bauform
 - ▶ Modbus/TCP Interface



www.systemerra.de

UEI DAQ-Lösungen

Software Architektur



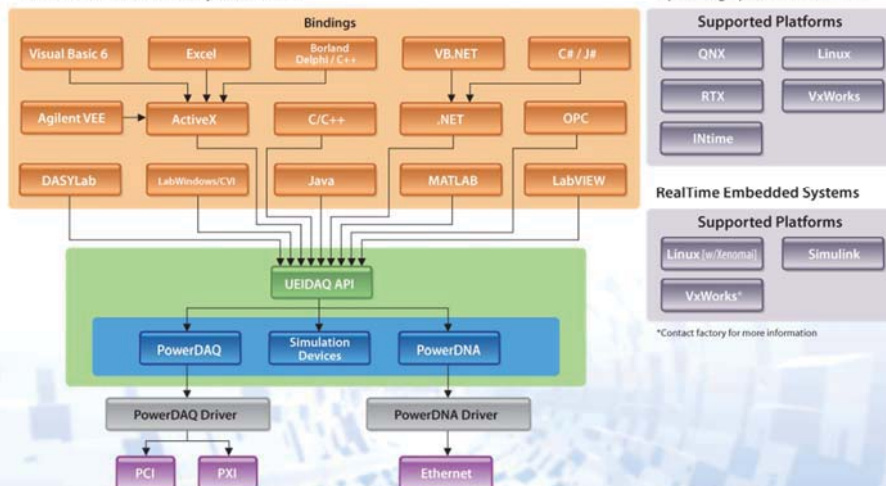
WINDOWS: 32/64-bit XP, Vista, 7

UEIDAQ Framework Library Architecture

[Hosted Systems]

LINUX & REALTIME

Operating Systems/Extensions



Test über simulierte Treiber: www.ueidaq.com/download

www.systemerra.de

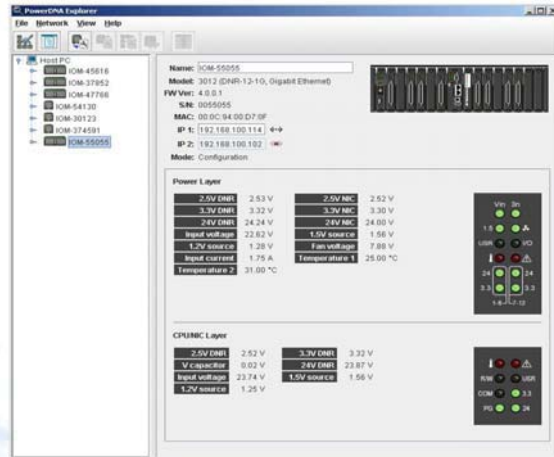
UEI DAQ-Lösungen

PowerDNA Software Suite



Eigenschaften:

- Durchgängige Softwareplattform
- Support für Windows, Linux, RTX, ...
- Keine Lizenzgebühren – Open Source
- Quellcode Beispiele für alle Module
- PowerDNA Explorer
- Spectrum Analyzer Software
- I/O-Simulation unter Windows
 - ▶ Ain, Aout, DIO, Serial, ARINC, 1553
 - ▶ Software kostenlos testen



Kostenloser Download:
www.ueidaq.com/download

www.systemerra.de

UEI DAQ-Lösungen

Langzeitverfügbar



**10 Jahre
Nachkauf-
Garantie**



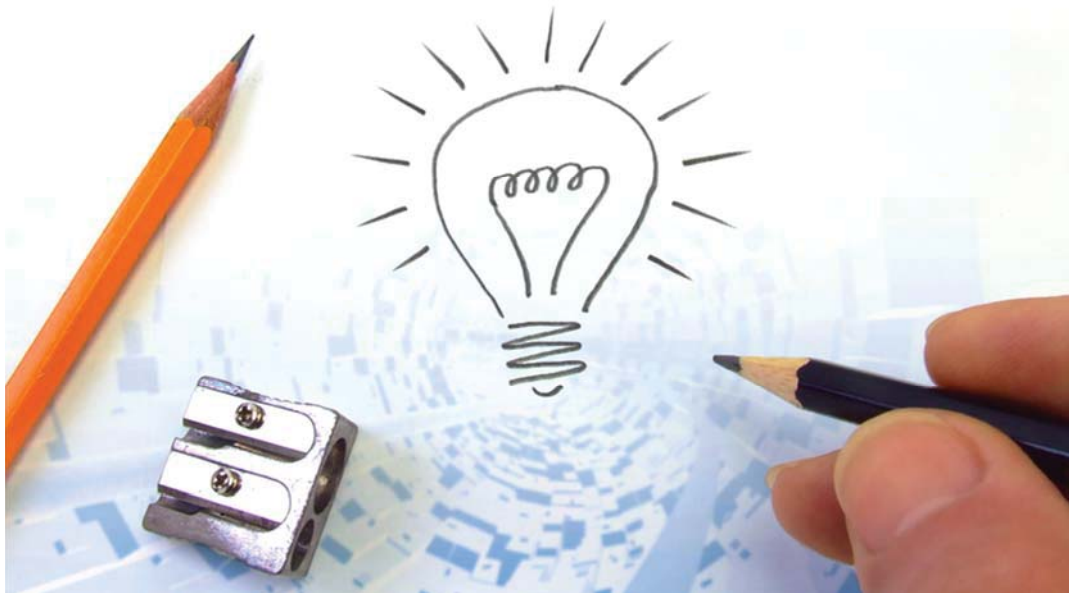
UEI garantiert den Nachkauf aller RACKTangle und Cube Produkte für mindestens 10 Jahre!

- Sofern beim Kauf nicht anders ausgewiesen, können alle DNA/DNR Komponenten mindestens 10 Jahre lang nachgekauft werden
- Langfristige Planungssicherheit für Projekte
- Investitionsschutz für Hard- und Software
- Sichere Versorgung mit Ersatzteilen

www.systemerra.de

UEI DAQ-Lösungen

Applikationsbeispiele



www.systemerra.de

UEI DAQ-Lösungen

Mesdatenerfassung in schweren Tagebaufahrzeugen



Ein führender Anbieter von Flurförderzeugen setzt den UEI Datalogger zur Überwachung einer Vielzahl an Funktionen in seinen Fahrzeugen ein.

- Die Applikation erfordert einen kompakten, äußerst robusten Datalogger
- Neben analogen und digitalen I/Os werden auch CANbus, RS-232 und GPS-Daten aufgezeichnet
- Die Umsetzung ist mit einem UEILogger 600 mit 16 analogen 18-bit Eingängen, 4 CANbus und 4 RS-232 Ports, 8-Kanal Zähler/Timer, GPS und 24-bit digital Eingangskarten realisiert
- Die Datenaufzeichnung erfolgt mit Zeitstempeln



www.systemerra.de

UEI DAQ-Lösungen

MSR-System für Flugsimulatoren



Installation direkt in der beweglichen Kabine

Die Ziele bei der Entwicklung eines Flugsimulators sind neben einer hervorragenden Performance höchste Verfügbarkeit sowie Bediener- und Wartungsfreundlichkeit.

- In enger Zusammenarbeit zwischen FlightSafety und UEI wurde das RACKTangle System in eine neue Simulatorgeneration integriert
- Geringes Gewicht, schnelle GbE Kommunikation, und hohe I/O-Dichte ermöglichen den Einsatz direkt an der Kabine
- Kurze Kabelwege sparen Gewicht und schützen vor Einstrahlung/Übersprechen
- Der wartungsfreundliche Aufbau spart im langjährigen Betrieb enorme Kosten

www.systemerra.de

UEI DAQ-Lösungen

Onboard In-flight Messdatenerfassungssystem



Die Ingenieure von Cessna haben ihr neues In-flight Datenerfassungssystem auf Basis eines UEI Cubes in Rekordzeit aufgebaut.

Das System sammelt im Flug Daten aus dem neuen Modell 162 SkyCatcher™ und erfüllt dabei alle Anforderungen des Kunden:

- 145 analoge und digitale Eingänge
- Systemaufbau auf kleinstem Raum
- Geringe Leistungsaufnahme
- Bedienbarkeit im Flug
- Portabilität und flexible Rekonfiguration
- Verarbeitung von ARINC 429 Telegrammen



www.systemerra.de

UEI DAQ-Lösungen

Referenzen

systemerra
computer



SAIC



Hawker Beechcraft



BAE SYSTEMS



GENERAL DYNAMICS



NORTHROP GRUMMAN

Rockwell Collins

LOCKHEED MARTIN



Raytheon



Sikorsky
A United Technologies Company

Honeywell

FlightSafety
international
Simulation

www.systemerra.de

UEI DAQ-Lösungen

systemerra
computer

**Vielen Dank für Ihre
Aufmerksamkeit!**

www.systemerra.de

UEI DAQ-Lösungen

Zusatzfolien



Details zu den Basiskonfigurationen

www.systemerra.de

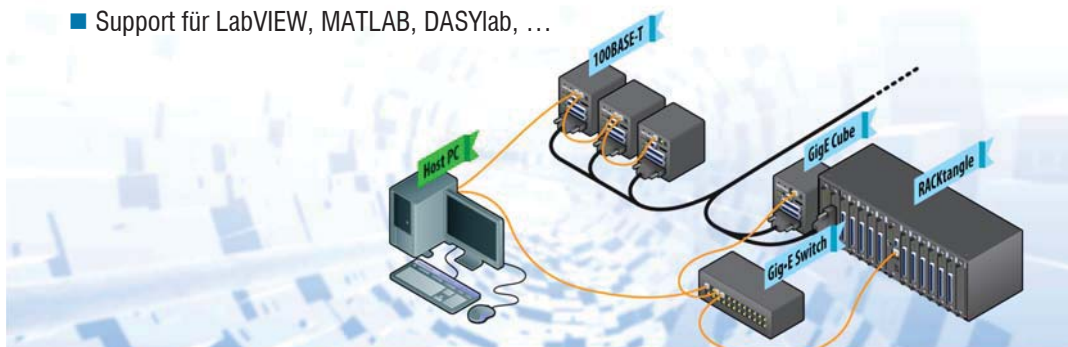
UEI DAQ-Lösungen

Host-controlled Cube oder RACK



Features:

- Standard Netzwerkinterface: (1000BaseT, 100BaseT oder 100BaseFX)
- Hohe Performance: Scan/Update 1000 I/O Messstellen in $< 250 \mu\text{s}$
- Low-Cost: Cubes ab 1000 EUR
- Exzellenter Software Support
- Betriebssystemsupport für Windows, Linux, QNX
- Support für die verbreiteten Programmiersprachen
- Support für LabVIEW, MATLAB, DASylab, ...



www.systemerra.de

UEI DAQ-Lösungen

UEIPAC (Programmable Automation Controller)



Features:

- Standalone-Betrieb (ohne Host-PC)
- Cube- und Rackbauform
- Open source, Applikationsentwicklung in C auf einem Linux-PC oder Windows PC mit Cygwin Umgebung
- Standard Fedora/Suse Linux 2.6.28 mit Xenomai RT extensions



Anwendungsbeispiele:

- Maschinenüberwachung
- Regelungstechnik
- Autonome Land-, See- und Luft-Fahrzeuge
- Avionik Testequipment
- Emulation/Ersatz alter I/O Systeme
- Robuste, portable Messsysteme



www.systemerra.de

UEI DAQ-Lösungen

UEILogger Cube



Features:

- Einfache Anwendung, keine Programmierung erforderlich
- Programmierung via Ethernet oder Start von SD-Karte
- Speicherung auf 4 GByte SD-Karte
- 1000 Samples/s pro Kanal
- Systempreis ab ~1600 EUR
- Download der Daten über Ethernet, WLAN, GSM oder SD-Karte
- Aufzeichnung von A/D, DI, Serial, CAN und ARINC 429 Daten, ...
- Automatische Konvertierung in SI-Einheiten
- Betrieb als PowerDNA Cube an Host-PC möglich

www.systemerra.de

UEI DAQ-Lösungen

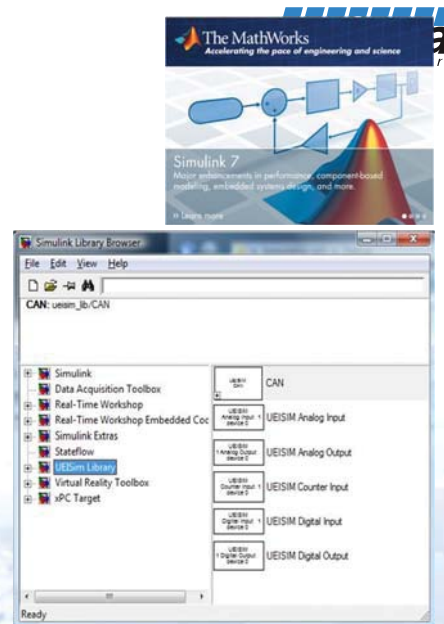
UEISIM (Simulink I/O Target)

Features:

- Simulink/RTW Hardwareinterface
- Basiert auf UEIPAC mit 2.6.28 Kernel und Xenomai Echtzeiterweiterung
- Support für Cube und Rack
- Standalone- und "Monitored"-Betrieb
- Bis zu 5 kHz Abtastrate

Applikationsbeispiele:

- Regelungstechnik
- Systementwicklung und -optimierung
- Prozesssimulation



www.systemerra.de

UEI DAQ-Lösungen

UEIModbus (Modbus® TCP Support)



Konfiguration:

- Kommunikation mit Host-PC oder SPS per Modbus TCP



Features:

- Industriestandard Modbus® TCP Interface
- Flexible I/O-Lösung
- Höhere Genauigkeit und Signalqualität als bei SPS
- Einbindung von I/O-Typen, die für SPS nicht zur Verfügung stehen

www.systemerra.de

UEI DAQ-Lösungen

μTCA Management

Aktuelle Entwicklungen im ZEL

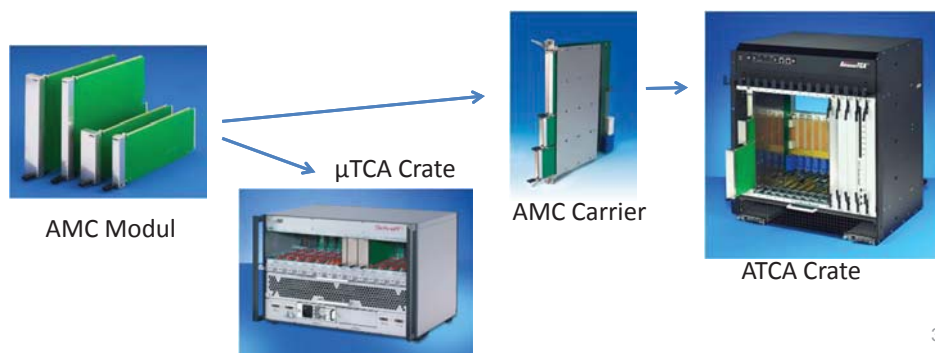
SEI Dresden 13. März 2012 | Matthias Drochner & Peter Kämmerling

Inhalt

- ATCA und μTCA
- IPMI
- IPMI-Testsoftware
- MMC und coreIPM
- Entwicklungsarbeiten

ATCA und μ TCA: kommender Standard für Physik-Instrumente

- Von vielen Labs für Neuentwicklungen favorisiert
- Erlauben den Einsatz moderner serieller Busse: 10Gb/s Ethernet, PCI Express, Serial Rapid IO, Serial ATA
- Für ATCA auch InfiniBand und StarFabric
- Integrierte Fernüberwachung



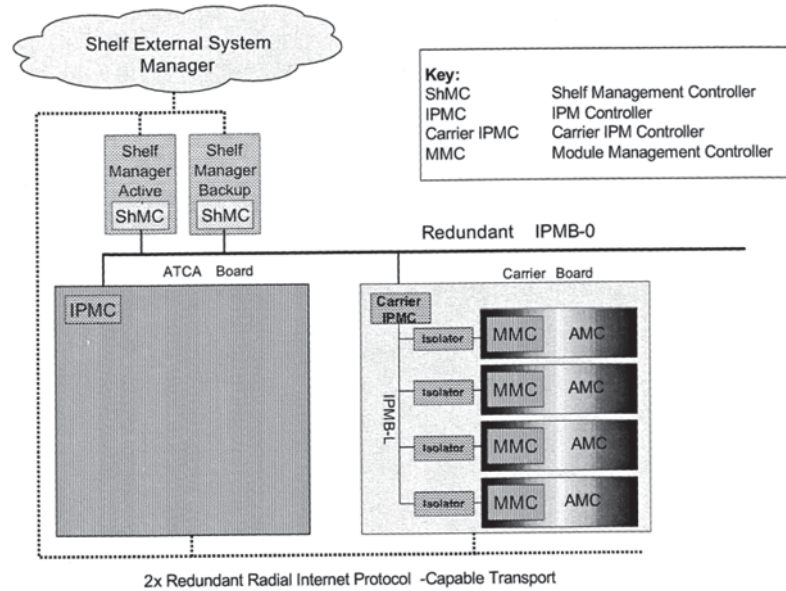
3

ATCA und μ TCA:

- primär Ersatz für VME
 - Signifikant höhere Bandbreite
 - Multiprocessing ist möglich, beschränkt mit PCIe
- Auch Ersatz für compactPCI und serielle Derivate
 - wenig industrielles I/O verfügbar
 - Steckverbinder sind weniger robust
- nur eine Versorgungsspannung: 12VDC oder 48VDC
- Entwicklungen im ZEL sind für μ TCA

4

Managementcontroller ATCA

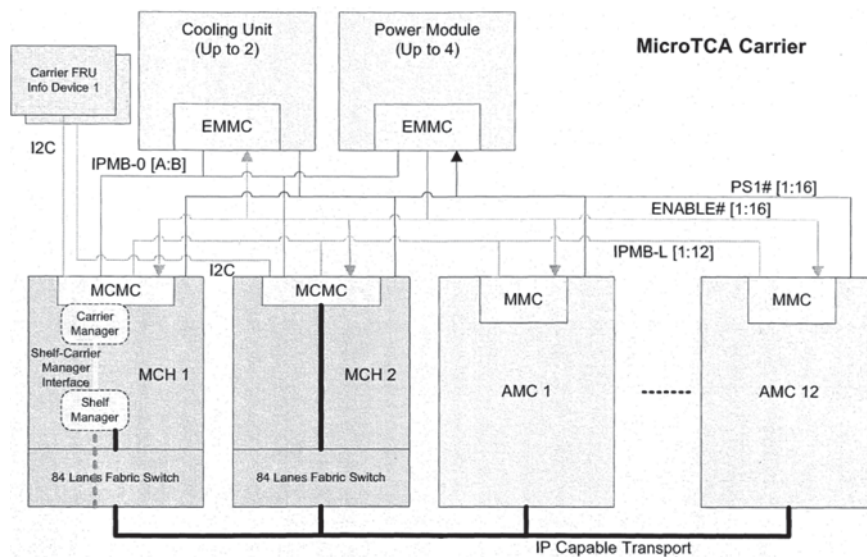


Quelle: Fig.3-1 Module Management Infrastructure, page 3-2 PICMG Advanced Mezzanine Card AMC.0 Specification R2.0, November 15, 2006

5

Mitglied der Helmholtz-Gemeinschaft

Managementcontroller μ TCA



Quelle, Excerpt: Fig.3-5 Example of Shelf Manager located with Carrier Manager, page 3-39 PICMG Specification MTCA.0 R1.0 July 6, 2006

6

Mitglied der Helmholtz-Gemeinschaft

IPMI – Intro

- Protokoll zur Fernwartung von PCs von Intel, HP, NEC, Dell
- Modell: Mikrocontroller als Teil des Systems, Verbindungen zu
 - Aktoren, Sensoren und Meßwerte wie Spannungen, Temperaturen, Lüfter, Schalter ...
 - ROM für „Inventardaten“ (FRU); NV-RAM für Protokoll und veränderliche Daten wie Event-Log, SensorData Record ...
 - Kontrolle von außen über LAN, RS232 ...
 - Kontrolle durch OS im PC „System Interface“: Register, Kommandoblöcke, UART, I2C, SMBus ...
 - Kommunikation mit anderen Management-Controllern, Subsysteme oder übergeordnet
- Kommandos „Messages“ über mehrere Controller geroutet

7

IPMI – Kommandogruppen

- soweit relevant für μ TCA- und ATCA-Management
 - Identifikation, Power, Reset
 - FRU (Field Replaceable Unit) Inventory: r/w
 - SDR (Sensor Data Record) Repository: r/w
 - Sensor device: r, ggf. Schwellen & Events
 - Events: set Receiver, Event Message
 - SEL (System Event Log): r/w/löschen
 - Messaging support: send/receive „raw“ Message

8

IPMI – Transportprotokolle

- IPMB, Intelligent Platform Management Bus
 - I2C-basiert, zwischen Modulen im Crate
- RMCP, Remote Management Control Protocol
 - UDP/IP-basiert, für Kontrolle von außen über LAN
 - Durch Paßwort oder cryptographisch schützbar
 - Unterstützt „Sessions“, d.h. mehrere unabhängige Transaktionen über dasselbe Interface
- Terminal Mode
 - UART-basiert als „System Interface“
 - Üblich für Verbindungen zur CPU auf μ TCA-Processorboards
 - Eingeschränkt für Menschen tauglich, weil Hex-kodiert

9

ATCA & μ TCA, relevante Spezifikationen

- IPMI v2.0: 644 S.
 - Platform Management
 - FRU Information Storage Definition v1.0: 33 S.
 - ATCA Base 3.0 „PICMG3.0“
 - AMC Base 2.0 „AMC.0“
 - μ TCA Base 1.0 „MTCA.0“
 - Ergänzungen für serielle Protokolle „AMC.1“ ... „AMC.4“, Physics „MTCA.4“
 - HPM.1 Hardware Platform Management
- } 800 Seiten

10

xTCA-spezifische Einträge im FRU-Inventory

- Verbindungen Punkt-zu-Punkt
- Multidrop für „Physics“
- Slots inclusive Geometrie
- Belegung der Backplane-Verbindungen in den Modulen
Protokoll, Channels / Lanes, upstream / downstream fuer PCIe
- Ähnlich für Clocks
- Power management: „Channels“ von Netzteilen und Verbindungen mit Maximalstrom, Bedarf der Module, Verzögerungen für sequentielles Einschalten
- I2C-Adressen der Komponenten im Crate
- IP-Adresse des Shelf-Managers für Kontrolle von aussen
HotSwap-Schalter-Sensor, Events für HotSwap

11

IP-Adresse des Shelf-Managers

- EEPROM mit Adress-FRU-Record an Backplane
- Spezifikation empfiehlt deren Bevorzugung
- Backplanes werden mit 192.168.16.17 oder 0.0.0.0 ausgeliefert; Spezifikation empfiehlt ersteres
- NAT-MCHs erlauben Ignorieren der IP-Adresse der Backplane und die eigene Adresse zu bevorzugen
- Standard ist auf Produktionsumgebungen ausgerichtet mit austauschbaren MCHs
- dies erfordert Programmierung des Backplane-EEPROMs, was ohne solide Tools zu Schaden führen kann
- in Laborumgebung ist MCH-gebundene Adresse bequemer

12

Motivation für IPMI-Testsoftware

- Test eigener Modul-Management-Controller
- bei Unklarheiten der Spezifikation Vergleich mit Verhalten industrieller Controller, Referenz Pigeon Point
- Untersuchung bei mysteriösen Interoperabilitäts-Problemen
- Diagnose auch ohne Logicanalyzer am IPMB
- Tool zum Modifizieren von FRU-Records im EEPROM
- Fernüberwachung und –steuerung
- xTCA-Workshops: Bedarf vorhanden, keine bezahlbare Software verfügbar

Basis: ipmitool

- Kommandozeilentool für IPMI-Transaktionen
- populär, z.B. Teil von (Open) Solaris
- kein offizielles Release seit 2007
- unterstützt IPMI über LAN (RMCP) und verschiedene Arten des „System interface“, aber nicht „Terminal mode“
- Textausgabe mit Transaktionscode vermischt, schwer zu parsen -> wenig nützlich
- nützlich: RMCP-Code, Header mit Definitionen für IPMI-Kommandos
- also: Kommunikationsbibliothek abgetrennt
- andere Projekte: freeipmi, ipmiutil: zu dick, zu viel Semantik

Erweiterung für IPMI-Routing

- Problem: Requests vom LAN müssen durch Shelf-Manager und Carrier-Manager zum Modul geroutet werden, 2 Hops
- dazu muss der Request 2x in ein „Send Message“-Kommando eingepackt werden
- ipmitool unterstützt 1 Hop
- also: Hilfsbibliothek zum Einpacken addiert, beliebige Tiefe
- Verwendung
 - `h = ipmiroute_open(i);`
 `/* shelf manager to carrier manager`
 `*/ ipmiroute_addroute(h, 0, 0x20, 0x82);` `/* carrier`
 `manager to AMC on IPMB-L */ ipmiroute_addroute(h, 7,`
 `0x20, slot2ipmi(slot));`

Funktionen

- rudimentär, bisher testweise implementiert:
 - Abfrage von Identifikationsdaten
 - FRU-Inventory: Abfrage, Ausgabe, Integritäts- und Checksummentests
 - SDR-Repository: Abfrage, Ausgabe, Integritätstests, Check der „Device Locators“
 - Sensor: Abfrage
 - Check von IPMI-Channel-Nummern

Verifikation und Ergebnisse

- mit verschiedenen AMC-Modulen über NAT- und Emerson-MCHs getestet
- Fehler in FRU (Checksumme), Device Locators, Header der Antwortpakete gefunden, von Herstellern bestätigt und beseitigt
- Evaluierung der „CoreIPM“-Management-Software in einer simulierten Umgebung unter UNIX, Verbindung zu Testcode über UNIX-Socket, Fehler gefunden, von Autor bestätigt

ipmitool: Pläne und Ideen

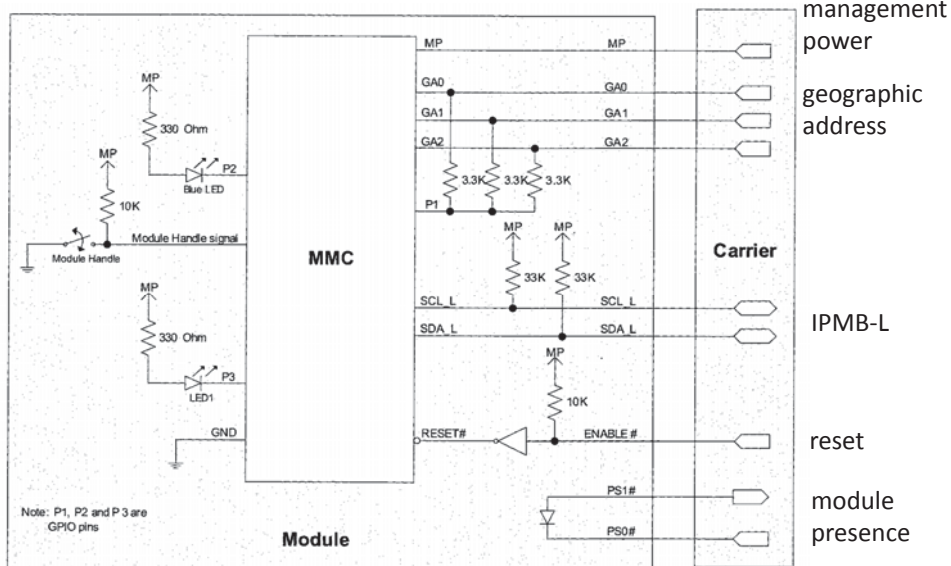
- Ausbau zur Bibliothek mit konsistentem API
- Kommandozeilentools mit allen nötigen Optionen
- Evtl. Scripting-Support, z.B. Python-Wrapper
- Tool zu FRU-Daten-Modifikation, EEPROM-Backup
- Unterstützung von „Terminal Mode“ für Verwendung auf AMC-CPU

Entwicklung eines MMC

- MMC: Module Management Controller
 - steuert Funktionseinheiten wie Lüfter, Netzteil, MCH, FRU wie I/O-Karten ...
 - Vor Ort auswechselbare I/O-Karten (FRU) müssen mit dem System, d.h. dem Crate-Controller, kommunizieren, dafür ist der MMC zuständig
 - Anmelden inkl. „Inventar“, Leistungsbedarf
 - Payload Power ein- und abschalten
 - Statusmeldungen, Befehle entgegen nehmen
 - aufgrund des IPMB (I2C) reines Master/Slave-Protokoll
 - interne Verwaltung möglich, z.B. Power Sequencing, Power Monitor, Temperatur Monitor, Watchdog

19

MMC Module Management Controller



Quelle: Fig.3-3 Module management hardware, page 3-13 PICMG Advanced Mezzanine Card AMC.0 Specification R2.0, November 15, 2006

20

Entwicklung μ TCA GPX Board

- 4x GPX ASIC TDC
- Virtex 5
- PCIx mit 4 Lanes
- MMC
- Hardware teilweise getestet

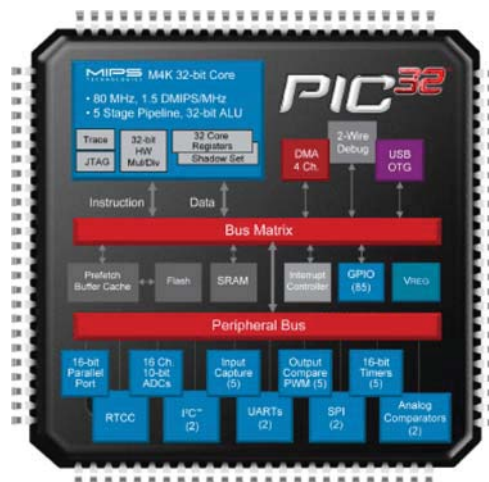


10. Juli 2012

21

MMC μ TCA GPX Board

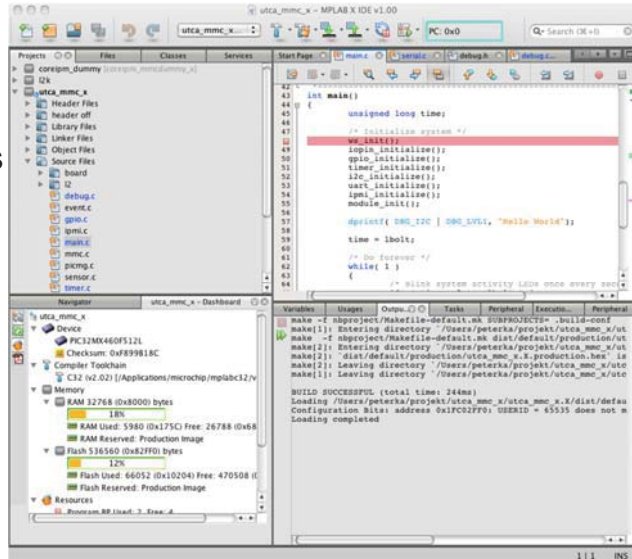
- microchip PIC32MX460
- MIPS32 M4k-Core 80MHz
- 1,56DMIPS/MHz
125DMIPS
- 70 Derivate mit
28, 36, 44, 64, 100 Pins
- sehr schnelles IO, z.B. für
JTAG-Master + SelectMAP
- in-circuit-debugger-
und -programmer
- JTAG-Interface



22

IDE MPLAB X & Libraries

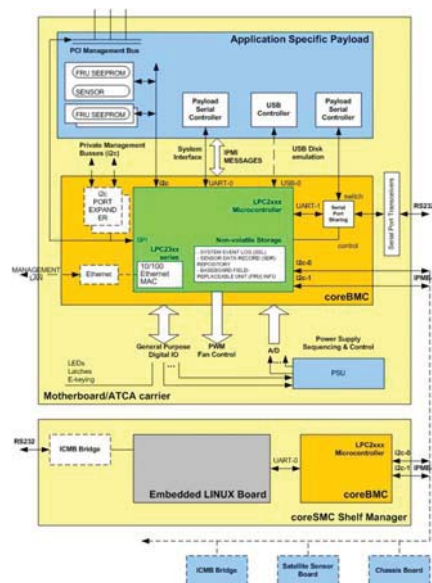
- gcc-toolchain
- IDE unter WIN, Linux, MOX
- Standard-C-Libraries
- DSP-Libraries
- I/O-Libraries
- Stack-Libraries u.a.:
 - USB, CAN
 - TCP/IP
 - graphic / touch
 - audio / speech
 - WiFi, ZigBee



23

coreIPM

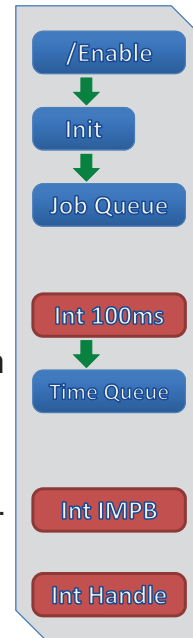
- open source
- coreBMC, kleinerer Teil
 - Baseboard Management Controller Software
 - läuft auf Microcontrollern des CM Carrier Management und des MM Module Management
- coreSMC, größerer Teil
 - Shelf Management Controller Software
 - läuft auf Prozessor des ShM Shelf Management



24

coreBMC

- Kommunikation mit System & Karten-Verwaltung
 - μ TCA-Signale, IPMB, interne Karten-Signale
- Programmablauf
 - initialisieren der CPU, IOs und Interrupts
 - Endlosschleife für Jobs in Warteschlange
 - 100ms Interrupt Timer für zeitgesteuerte Jobs
 - per Interrupt IPMI-Pakete von IPMB bearbeiten
 - per Interrupt Signale bearbeiten
 - Tabelle mit festen Einträgen
 - Variablen-Tabelle mit dynamischen Werten z.B. Zustand, Temperatur, Anzahl Kanäle ...



25

Entwicklungsstand

- coreBMC auf MPLAB X & PIC32 portiert
 - angepasst an CPU und Peripherie
 - Prozessor-Initialisierung
 - IO-Pins, z.B. geographische Adresse, BlueLED, Handle-Switch, /Enable (MMC-Reset) ...
 - IPMB-L (I2C)
 - Interrupts
- Herauslösen der Hardware-abhängigen Programmteile, auslagern in eigene Library
- Implementation zuverlässiger I/O-Protokoll-Libraries für I2C, 100ms Timer-Interrupt, Pin-Interrupt u.ä. hat lange gedauert

26

Erfahrungen

- PIC32 ist eine neue Microcontrollerfamilie
 - erwartete lange Verfügbarkeit
 - Libraries wurden mehrfach überarbeitet
 - Beispielprogramme teilweise für alte Libraries
 - Doku teilweise noch lückenhaft und widersprüchlich
 - „Integration“ dreier Vorgänger-toolchains: neue Entwicklungsumgebung MPLAB X nicht fertig
- mehrfaches Wechseln zwischen alter und neuer Version der microchip IDE aufgrund diverser Probleme
- openBMC enthält viele dynamische Zuordnungen, z.B. zwei mögliche Kanäle für IPMB, es wird immer nur einer benutzt, dynamische Features sind teilweise nicht ausprogrammiert

27

Ausblick

- μ TCA GPX Board Payload einschalten
- im μ TCA-Crate und mit ipmitool testen
- spätere Weiterentwicklung
 - modularer Aufbau
 - IPMI
 - Board-Hardware-Anpassung
 - Einbetten in ein kleines RTOS, z.B. FreeRTOS
- Fern-Diagnose des Board, inkl. FPGA
- FPGA-Firmware laden, updaten, entwickeln

28

Model Driven FPGA Design

An approach using the Eclipse Modeling Framework

Marek Penno (M.Eng.)

SEI-Tagung am HZDR – Frühjahr 2012
2012/03/14



Orientierung.

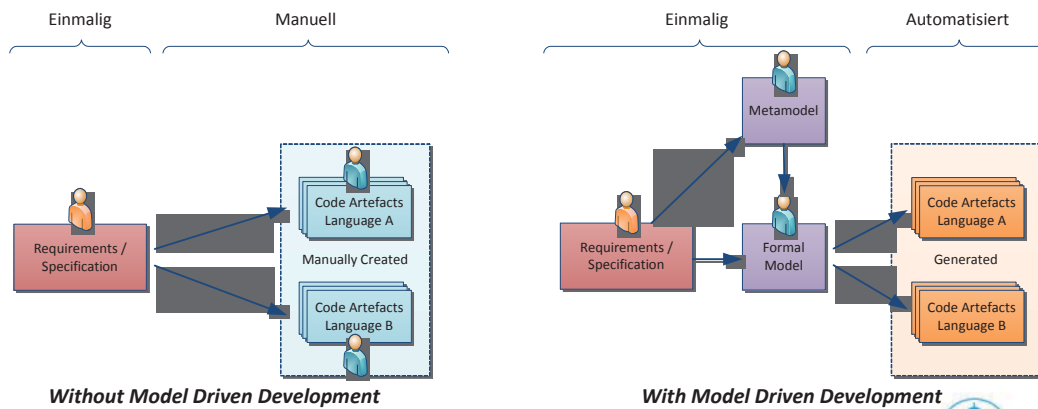
> Modellgetriebene Entwicklung

- > Werkzeuge
- > Beispiel: Register Block
 - Generieren von VHDL und C Code
- > Zusammenfassung



Modellgetriebene Entwicklung.

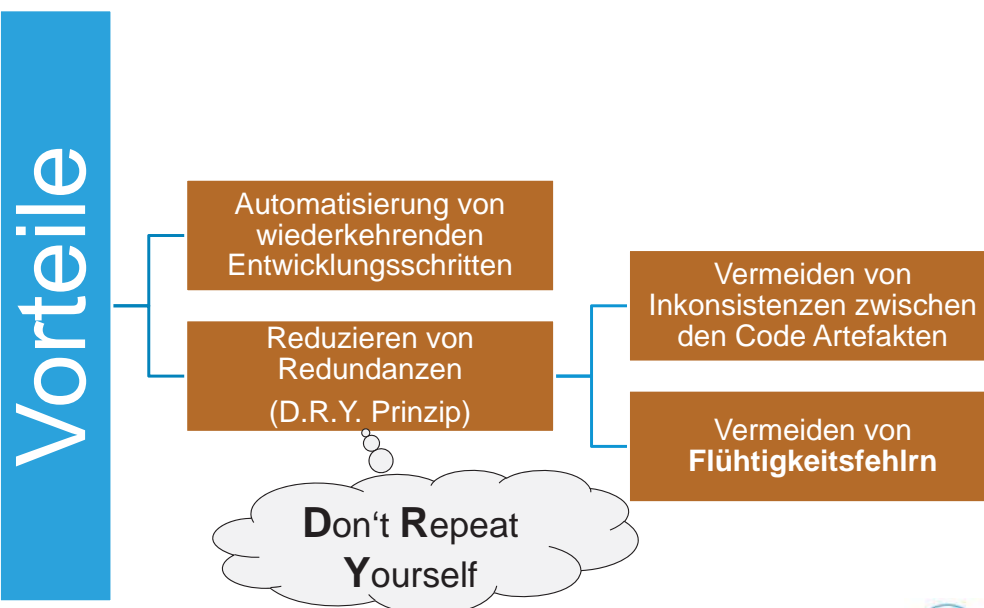
- > Metamodell beschreibt formales Modell der Problemdomäne
- > Formales Modell beschreibt einen Sachverhalt
- > Modell wird mit einer „Domain Specific Language“ (DSL) beschrieben
- > Artefakte werden anhand des Modells generiert



Marek Penno (M.Eng), DESY | Page 3



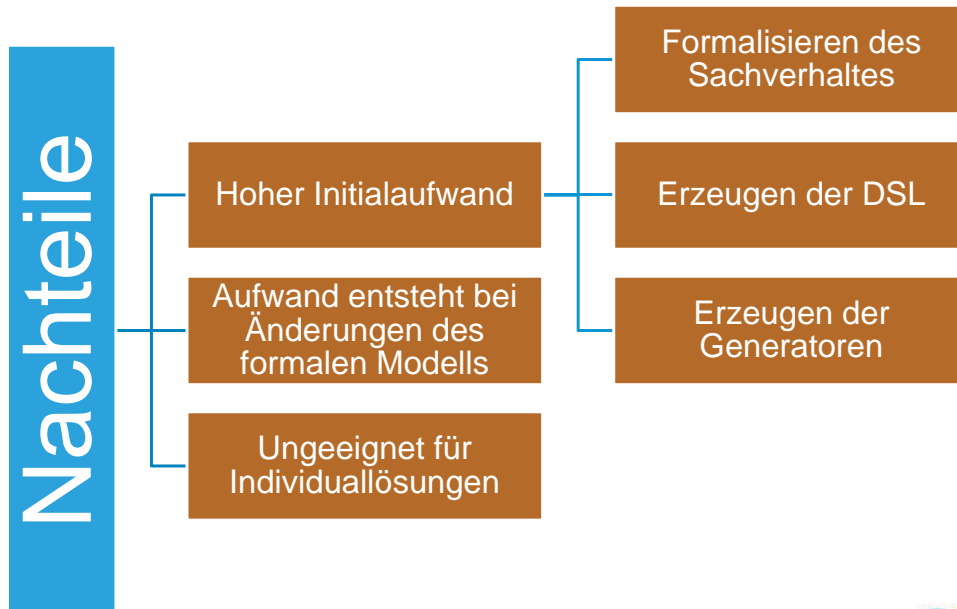
Modellgetriebene Entwicklung.



Marek Penno (M.Eng), DESY | Page 4



Modellgetriebene Entwicklung.



Marek Penno (M.Eng), DESY | Page 5



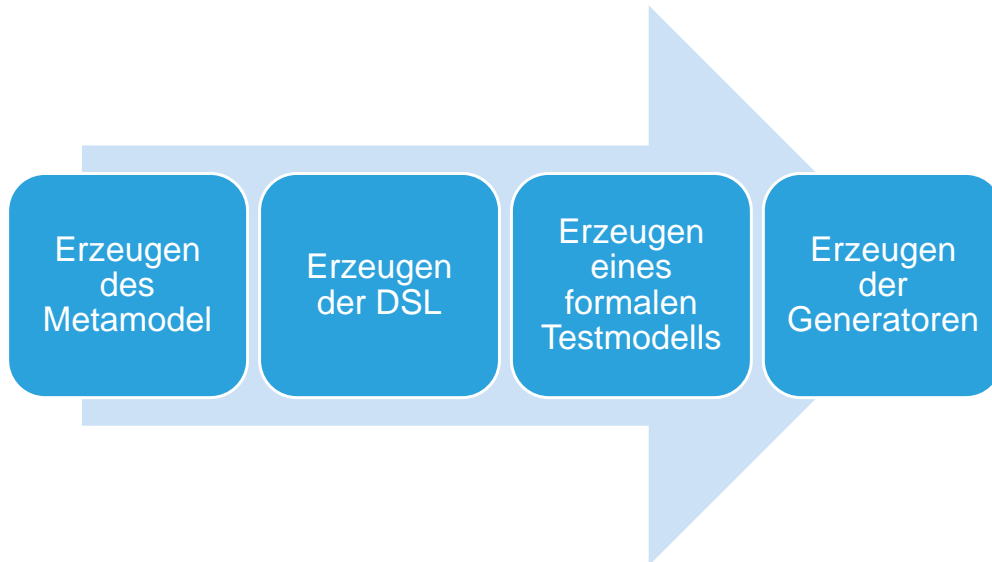
Orientierung.

- > Modellgetriebene Entwicklung
- > **Werkzeuge**
- > Beispiel: Register Block
 - Generieren von VHDL und C Code
- > Zusammenfassung

Marek Penno (M.Eng), DESY | Page 6



Initialaufwand.



Marek Penno (M.Eng), DESY | Page 7



Werkzeuge.

- > Verschiedene Werkzeuge existieren (Siehe Wiki[1])
 - Integrierte Systeme mit graphischen Editor und Generatoren
 - Reine Transformatoren für Modelle
 - XML/XMI basierend (Extended Metadata Interchange)
- > Werkzeuge häufig optimiert zum Generieren von spezifische Sprachen (C, C++, Java, etc.)
- > In diesem Vortrag wird der Eclipse Modelling Framework vorgestellt
 - Zur Definition eigener formaler Modelle
 - Zur Definieren von eigenen "Domain Specific Language"
 - Zur Definieren eigener Generatoren
 - Zum Generieren von Code

Marek Penno (M.Eng), DESY | Page 8



Werkzeugvariante.



Entwicklungsumgebung: Eclipse IDE



Modellbeschreibung:
Eclipse Modelling Framework (EMF)



Generieren der DSL: EMFText



Model zu Text Transformation:
Acceleo 3.0

Marek Penno (M.Eng), DESY | Page 9



Werkzeuge.

- > Eclipse Modelling Framework (EMF) [3]
 - JAVA/XML Framework
- > Unterstützung benutzerdef. Modelle und DSL's
- > Erstellen benutzerdef. Generatoren
- > Viel Unterstützung beim Initialaufwand
 - Graphische Werkzeuge zum modellieren des Metamodells
 - Automatische Generierung der DSL und Editors anhand des Metamodells (EMFText)
 - Template basierende Model zu Text Transformation
- > Java basierend → Plattform unabhängig
- > Open Source
- > Große aktive Entwicklergemeinde

Marek Penno (M.Eng), DESY | Page 10



Orientierung.

- > Modellgetriebene Entwicklung
- > Werkzeuge
- > **Beispiel: Register Block**
 - Generieren von VHDL und C Code
- > Zusammenfassung



Beispiel: Registermodell.

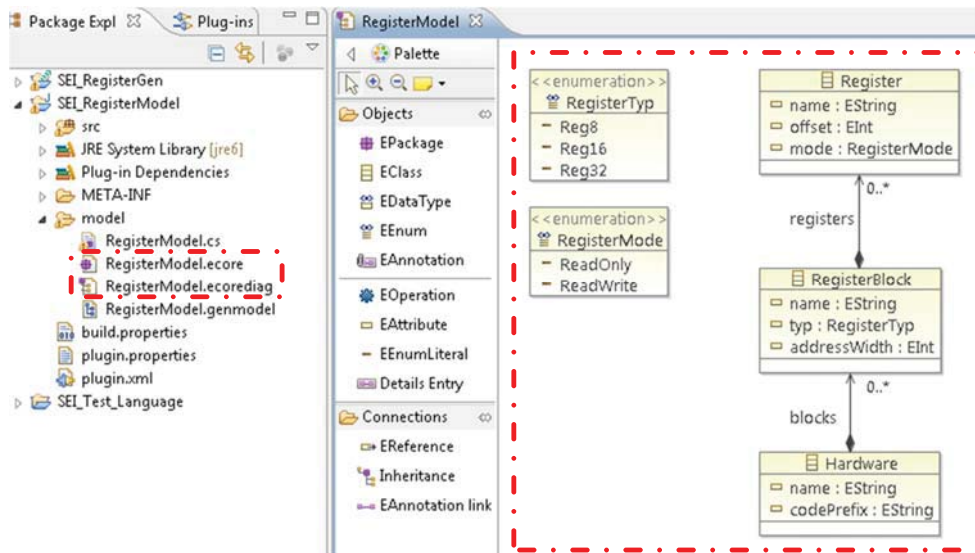
Anforderungen

- > Erzeugen von C und VHDL Code für n Registerblöcke
- > Unterstützen von 8, 16 und 32 Bit Registern
- > Unterstützen von Lese/Schreib und Nur-Lese Registern (RO / RW)
- > Variable Adressbusbreite



Erzeugen des Metamodells.

> Verwenden des EMF Diagramm Editors



Marek Penno (M.Eng), DESY | Page 13



Erzeugen der DSL.

- > Generieren des Metamodels als Eclipse Plugin (Modell Backend)
- > Generieren der DSL Grammatik Spezifikation
- > Generieren des DSL Eclipse Plugins (Text Editor)
- > Installieren der Plugins in die Eclipse IDE

> Man generiert einen Text Editor zum Editieren der DSL

- Implementiert als Eclipse Plugin
- Syntax Hervorhebung
- Autovervollständigung
- Code Hyperlinks
- Automatisches Klammerhandling
- Instant Error Reporting
- More...

Marek Penno (M.Eng), DESY | Page 14



DSL Grammatik Spezifikation.

```

SYNTAXDEF registermodel
FOR <http://registermodel/1.0>
START Hardware

TOKENS {
  DEFINE COMMENT $/*~(\n|\r|\uffff)*$;
  DEFINE INTEGER $('-')?(('1'..'9')('0'..'9')*|'0')$;
  DEFINE FLOAT $('-')?(('1'..'9')('0'..'9')*|'0') '.' ('0'..'9')+ $;
}

TOKENSTYLES {
  "Hardware" COLOR #7F0055, BOLD;
  "name" COLOR #7F0055, BOLD;
  "codePrefix" COLOR #7F0055, BOLD;
  "RegisterBlock" COLOR #7F0055, BOLD;
  "Register" COLOR #7F0055, BOLD;
  "typ" COLOR #7F0055, BOLD;
  "offset" COLOR #7F0055, BOLD;
  "mode" COLOR #7F0055, BOLD;
}

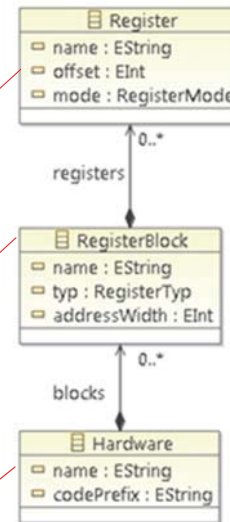
RULES {
  Register ::= "Register" "{"
    ("name" ":" name['"', "'"] |
    "offset" ":" offset[INTEGER] |
    "mode" ":" mode[ReadOnly:"RO", ReadWrite:"RW"])*
  "}";

  RegisterBlock ::= "RegisterBlock" "{"
    ("name" ":" name['"', "'"] |
    "addressWidth" ":" addressWidth[INTEGER] |
    "typ" ":" typ[Reg8:"Reg8", Reg16:"Reg16", Reg32:"Reg32"] |
    registers)*
  "}";

  Hardware ::= "Hardware" "{"
    ("name" ":" name['"', "'"] | "codePrefix" ":" codePrefix['"', "'"] | blocks)*
  "}";
}

```

Metamodell



Marek Penno (M.Eng), DESY | Page 15



Erzeugen des Testmodells mit dem DSL Editor.

```

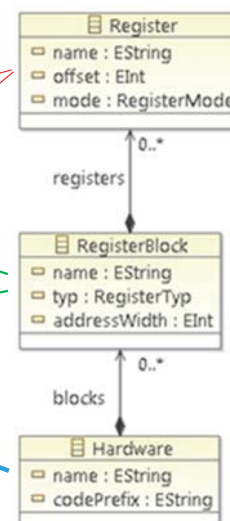
myregistermodel.registermodel
Hardware{
  name:"AnalogBoard"
  codePrefix:"anaio"

  RegisterBlock {
    name: "AdcChannel"
    addressWidth: 3
    typ: Reg16
    Register { name:"min" offset: 0 mode: RW }
    Register { name:"max" offset: 2 mode: RW }
    Register { name:"status" offset: 4 mode: RO }
  }

  RegisterBlock {
    name: "DacChannel"
    addressWidth: 3
    typ: Reg16
    Register { name:"value" offset: 0 mode: RW }
    Register { name:"rampTiming" offset: 2 mode: RW }
    Register { name:"status" offset: 4 mode: RO }
    Register { name:"control" offset: 6 mode: RO }
  }
}

```

Metamodell



Marek Penno (M.Eng), DESY | Page 16



Warum textbasierende DSL?

Nachteile

- Benötigt einen Parser (Text zu Modell)
- Nicht immer formal, benötigt Syntaxprüfung

Vorteile

- I.d.R. menschenlesbar
- Überlebt meist die eigenen Werkzeuge
- Universelles Datenformat
- Versionierbar! (SVN, CVS, GIT, etc.)

Marek Penno (M.Eng), DESY | Page 17



Erzeugen der Generatoren für C und VHDL.

- > Modell zu Text Transformation
- > Generator Framework: Acceleo 3.0 (entwickelt von OBEO [5])
 - 4 Jahre R&D
 - Open Source
 - offizielles M2T Subprojekt von eclipse.org
- > Generatoren werden mit Templates definiert
- > Template Sprache:
 - Model to Text Language (MTL) definiert von der Object Management Group (OMG)
 - <http://www.omg.org/spec/MOFM2T/1.0/>
 - Ermöglicht einfaches Navigieren im Model Baum

Marek Penno (M.Eng), DESY | Page 18



Template zum Generieren von VHDL Code.

```
[file (aRegBlock.name+'.vhd1', false, 'UTF-8')]
-- *****
-- Attention File has been generated!
-- *****

library IEEE;
use IEEE.std_logic_1164.all;
use ieee.numeric_std.all;

-- address map
[for (reg : Register | registers)]
-- [reg.offset/] [reg.name/] ([reg.mode/])
[/for]

-- inputs
[for (reg : Register | registers) ? (mode = RegisterMode::ReadOnly)]
[reg.getInterfaceName()/] : in [reg.getVHDLType()/];
[/for]
```



Template zum Generieren C Code.

```
// ***** Offset Definitions *****
[for (reg : Register | registers)]
#define [getOffsetName()/] [offset/] // offset of register '[name/]'
[/for]

// ***** Getter Definitions for Read Only Register *****

[for (reg : Register | registers) ? (mode = RegisterMode::ReadOnly)]
// reads from register '[name/]'
inline static [reg.registerType()/] [reg.prefix()/]_get[name.toUpperFirst/](unsigned int base)
{
    return [IORD_DIRECT(typ()/)](base, [getOffsetName()/]);
}

[/for]

// ***** Getter/Setter Definitions for ReadWrite Register *****

[for (reg : Register | registers) ? (mode = RegisterMode::ReadWrite)]
// reads from register '[name/]'
inline static [reg.registerType()/] [reg.prefix()/]_get[name.toUpperFirst/](unsigned int base)
{
    return [IORD_DIRECT(typ()/)](base, [getOffsetName()/]);
}

// writes to register '[name/]'
inline static void [reg.prefix()/]_set[name.toUpperFirst/](unsigned int base,[reg.registerType()/] newValue)
{
    [IOWR_DIRECT(typ()/)](base, [getOffsetName()/], newValue);
}

[/for]
```



Generierten Artefakte.

output

AdcChannel.h
AdcChannel.vhdl

bus.h
DacChannel.h
DacChannel.vhdl

```
RegisterBlock {  
  name: "AdcChannel"  
  addressWidth: 3  
  typ: Reg16  
  Register { name:"min" offset: 0 mode: RW }  
  Register { name:"max" offset: 2 mode: RW }  
  Register { name:"status" offset: 4 mode: RO }  
}
```

```
RegisterBlock {  
  name: "DacChannel"  
  addressWidth: 3  
  typ: Reg16  
  Register { name:"value" offset: 0 mode: RW }  
  Register { name:"rampTiming" offset: 2 mode: RW }  
  Register { name:"status" offset: 4 mode: RO }  
  Register { name:"control" offset: 6 mode: RO }  
}
```

Marek Penno (M.Eng), DESY | Page 21



Wiederholung: Das VHDL Template.

```
entity [getVhdlName()] is  
  port(  
    -- inputs  
    [for (reg : Register | registers) ? (mode = RegisterMode::ReadOnly)]  
    [reg.getInterfaceName()] : in [reg.getVHDLType()];  
    [/for]  
    -- outputs  
    [for (reg : Register | registers) ? (mode = RegisterMode::ReadWrite)]  
    [reg.getInterfaceName()] : out [reg.getVHDLType()];  
    [/for]  
    -- bus interface  
    clock      : in std_logic;      -- clock signal  
    RdEn, WrEn : in std_logic;      -- read/write strobe  
    cs         : in std_logic;      -- chip select  
    addr       : in std_logic_vector([addressWidth-1] downto 0);  
    dataIn     : in std_logic_vector(15 downto 0);  
    dataOut    : out std_logic_vector(15 downto 0)  
  );  
end [getVhdlName()];
```

Marek Penno (M.Eng), DESY | Page 22



Generierte VHDL Code.

```
entity AdcChannel is
  port(
    -- inputs
    status : in std_logic_vector(15 downto 0);

    -- outputs
    min : out std_logic_vector(15 downto 0);
    max : out std_logic_vector(15 downto 0);

    -- bus interface
    clock      : in std_logic;      -- clock signal
    RdEn, WrEn : in std_logic;      -- read/write strobe
    cs         : in std_logic;      -- chip select
    addr       : in std_logic_vector(2 downto 0);
    dataIn     : in std_logic_vector(15 downto 0);
    dataOut    : out std_logic_vector(15 downto 0)
  );
end AdcChannel;
```



Wiederholung: Das C Template.

```
[for (reg : Register | registers) ? (mode = RegisterMode::ReadOnly)]
// reads from register '[name/]'
inline static [reg.registerType()] [reg.prefix()/_get[name.toUpperFirst
{
  return [IORD_DIRECT(typ())/(base, [getOffsetName()]);
}
]
[/for]
// ***** Getter/Setter Definitions for ReadWrite Register *****
```

```
[for (reg : Register | registers) ? (mode = RegisterMode::ReadWrite)]
// reads from register '[name/]'
inline static [reg.registerType()] [reg.prefix()/_get[name.toUpperFirst
{
  return [IORD_DIRECT(typ())/(base, [getOffsetName()]);
}
// writes to register '[name/]'
inline static void [reg.prefix()/_set[name.toUpperFirst()/(unsigned int
{
  [IOWR_DIRECT(typ())/(base, [getOffsetName()], newValue);
}
]
[/for]
```



Generierte C Code.

```
// reads from register 'min'
inline static unsigned short anaio_adcChannel_getMin(unsigned int base)
{
    return IORD_16DIRECT(base, OFFS_ANAIO_ADCCHANNEL_MIN);
}

// writes to register 'min'
inline static void anaio_adcChannel_setMin(unsigned int base,unsigned short newValue)
{
    IOWR_16DIRECT(base, OFFS_ANAIO_ADCCHANNEL_MIN, newValue);
}

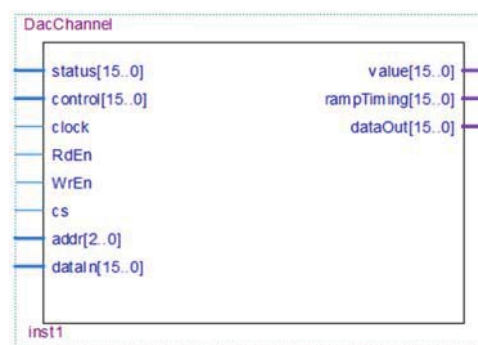
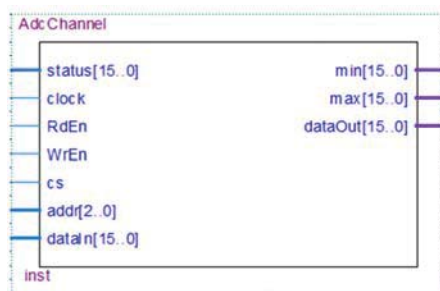
// reads from register 'max'
inline static unsigned short anaio_adcChannel_getMax(unsigned int base)
{
    return IORD_16DIRECT(base, OFFS_ANAIO_ADCCHANNEL_MAX);
}

// writes to register 'max'
inline static void anaio_adcChannel_setMax(unsigned int base,unsigned short newValue)
{
    IOWR_16DIRECT(base, OFFS_ANAIO_ADCCHANNEL_MAX, newValue);
}
```

Marek Penno (M.Eng), DESY | Page 25



Optional: Quartus generiert die Symbole.



Marek Penno (M.Eng), DESY | Page 26



Zusammenfassung.

- > Eclipse + EMF + EMFText + Acceleo
 - Leistungsfähiges modernes Toolkit
 - Entwickeln von eigenen textbasierenden Domain-Spezifischen – Sprachen (DSL)
 - Entwickeln von eigenen Model zu Text Transformatoren (Code Generatoren)
- > MDD - Initialaufwand wird vereinfacht durch die Code Generatoren des EMF Frameworks
 - Keine Eigenentwicklung Parser notwendig
 - Keine Eigenentwicklung Generator notwendig
- > Aber: Am Anfang steile Einstiegskurve
- > Lohnt sich: Wenn formales Modell definierbar und Code Generatoren sehr oft wiederverwendet werden
- > Lohnt sich weniger bei individuellen Lösungen, einmaligen Projekten, unklaren bzw. volatilen Anforderungen



Quellen.

- > [1] Model Driven Development Definiton
http://de.wikipedia.org/wiki/Modellgetriebene_Softwareentwicklung
- > [2] EMFText – DSL Generator
<http://www.emftext.org/index.php/EMFText>
- > [3] EMF – Eclipse Modelling Framework
<http://www.eclipse.org/modeling/emf/>
- > [4] Eclipse – Integrierte Entwicklungsumgebung
<http://www.eclipse.org/>
- > [5] Acceleo 3.0 - Generator Framework implementiert OMG Standard
<http://www.acceleo.org/pages/home/en>
- > [6] OMG – Object Management Group
<http://www.omg.org/>
- > [7] MTL - Model Transformation Language
<http://www.omg.org/spec/MOFM2T/1.0/>



KIT Kamera System für UFO

**M. Balzer, M. Caselle, S. Chilingaryan, A. Herth, A. Kopmann, U. Stevanovic,
M. Vogelgesang (IPE)
T. dos Santos Rolo (ISS)**

Institute for Data Processing and Electronics (IPE)



KIT – University of the State of Baden-Wuerttemberg and
National Laboratory of the Helmholtz Association

www.kit.edu

Einleitung

- Vorstellung des UFO Projektes
- Beschleunigung der Bildrekonstruktion
- Programmierbare High-Speed Kamera
- Embedded High-Speed Kamera

UFO Projekt



- **UFO – Ultra Fast X-ray Imaging of Scientific Processes with On-line Assessment and Data-driven Process Control**

- Deutsch russisches BMBF Projekt

- Kollaboration

- Staatliche Universität für zivile Luftfahrt (St. Petersburg)
- Shubnikov Institut für Kristallographie (Moskau)
- Polytechnische Universität (Tomsk)
- ANKA - Institut für Synchrotron Strahlung (KIT)
- Institut für Prozessdatenverarbeitung and Elektronik (KIT)

UFO Projekt



- **UFO – Ultra Fast X-ray Imaging of Scientific Processes with On-line Assessment and Data-driven Process Control**

- Deutsch russisches BMBF Projekt

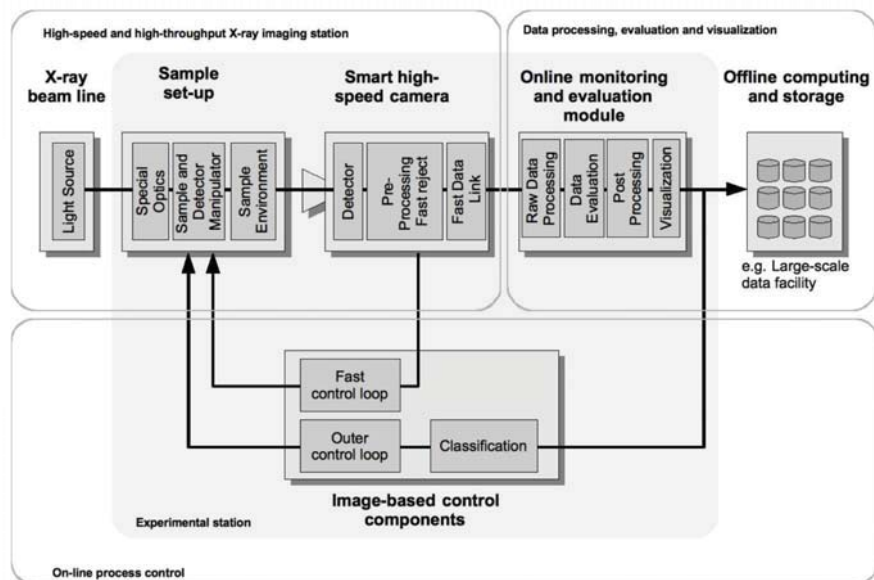
- Kollaboration

- Staatliche Universität für zivile Luftfahrt (St. Petersburg)
- Shubnikov Institut für Kristallographie (Moskau)
- Polytechnische Universität (Tomsk)
- ANKA - Institut für Synchrotron Strahlung (KIT)
- Institut für Prozessdatenverarbeitung and Elektronik (KIT)

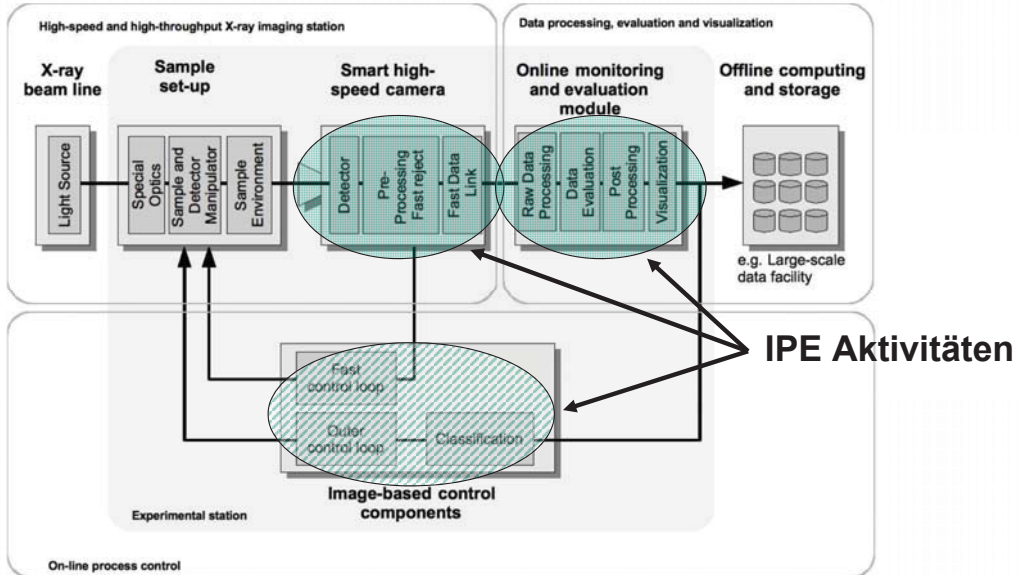
Motivation für das UFO Projekt

- Neue „Image Beamline“ an ANKA (Synchrotron in Karlsruhe)
- Schnelle Radio- und Tomographiegraphie
- Hoher Durchsatz von Proben
 - Materialwissenschaft
 - Biowissenschaft (Zebrafische,...)
- Beobachtung von dynamischen Prozessen
 - In vivo Messungen an Insekten
 - Umwandlungen in Materialien und Flüssigkeiten

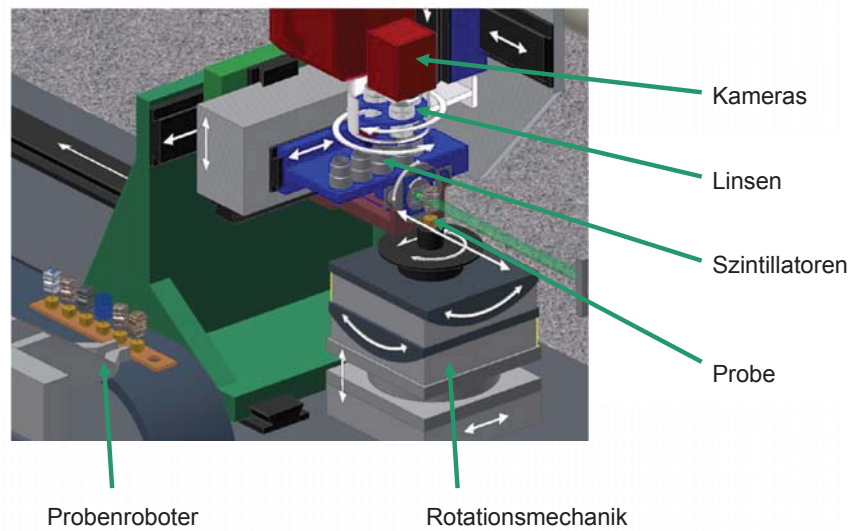
UFO im Überblick



UFO im Überblick



Mechanik und Detektoren



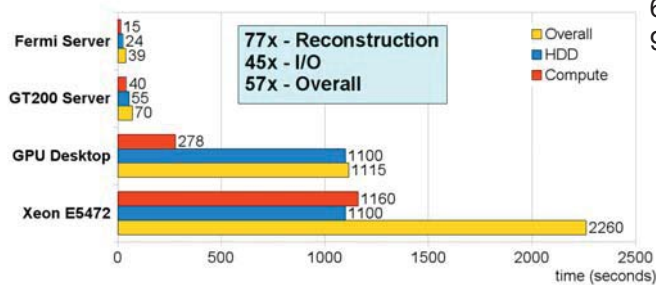
T. dos Santos Rolo ISS

GPU-Server für die schnelle Rekonstruktion

- Beschleunigung der Rekonstruktion
- Optimierte Algorithmen
- Framework



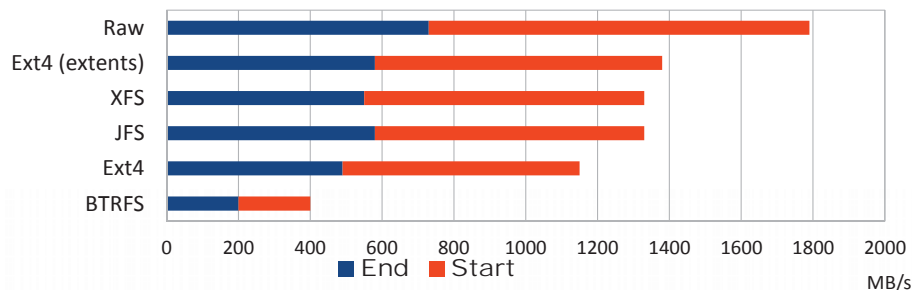
Rekonstruktionsrechner:
 2x Six Core CPU
 6x GPUs GTX580
 96 GB Memory



S.Chilingaryan, M. Vogelgesang

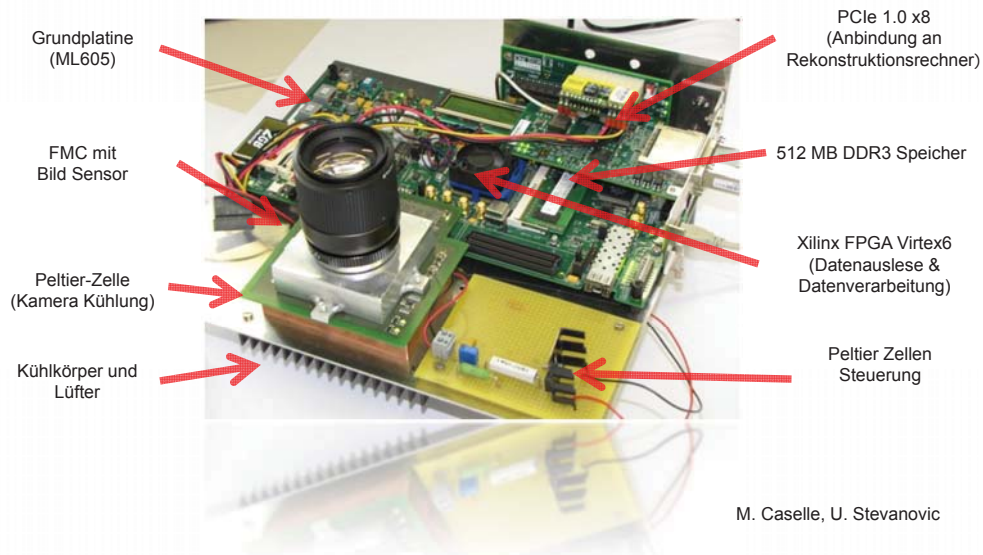
Speicherung der Sensordaten

- Hohe Datenrate (> 1 GB/s)
- Bereitstellung eines schnellen Datenspeichers
- Anbindung an die LSDF (10Gb/s)
- Untersuchung des Einflusses des Filesystems
- 16 Hitachi A7K200 in Raid0



- 800 MB/s erreichte mittlere Speicherdatenrate

KIT Kamera Programmierbare High-Speed Kamera



11

SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

KIT Kamera Sensor



CMOSIS Sensor CMV2000

- 2.2 Megapixel (2048 * 1088)
- Bis zu 340 fps
- 10 oder 12 Bit Auflösung
- Parallele Schnittstelle für die Datenauslese (16 x 480 Mb/s)
- Serielle Schnittstelle (SPI) für die Steuerung
 - Aufnahme Modus
 - Fensterdefinition
 - Offset und Verstärkung
 - ...

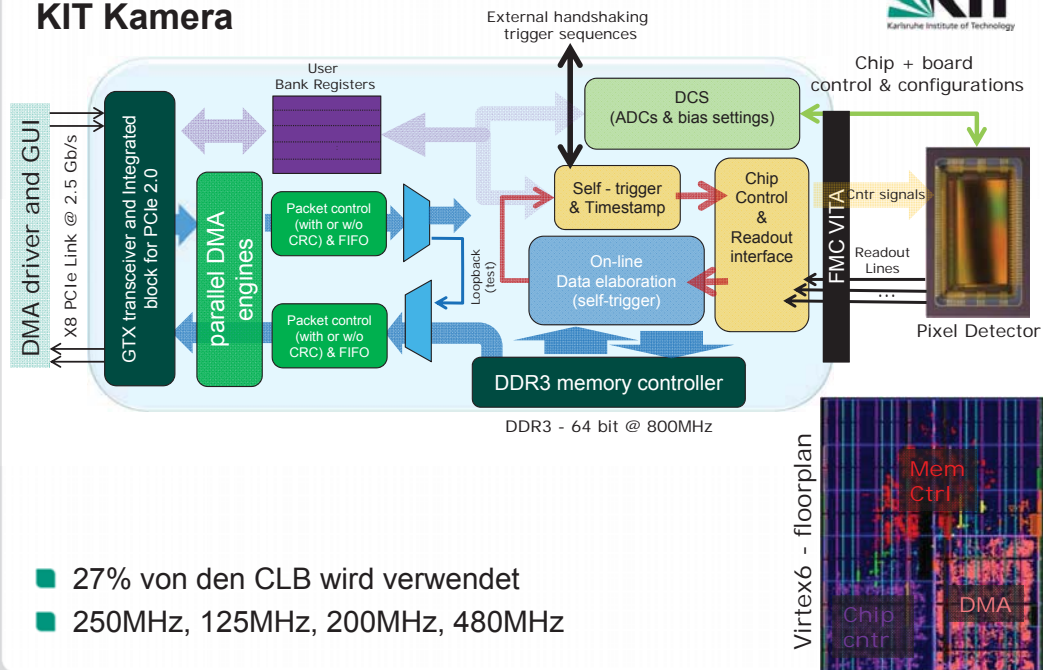


12

SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

KIT Kamera



13

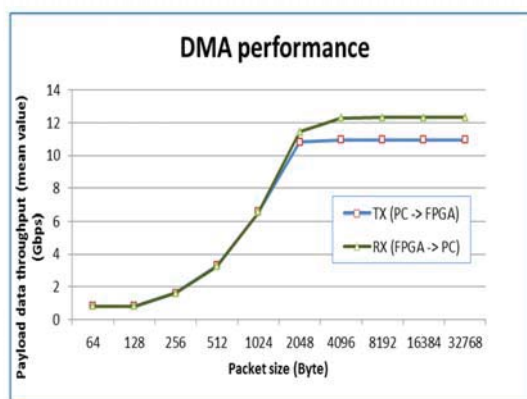
SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

KIT Kamera Eigenschaften

- Full Streaming Mode 340 fps
- Datentransferrate 14 Gbit/s mit PCIe Cabel im DMA Mode
- Netto Datenrate > 12 Gbit/s

- Externe Triggerung
- Programmierbare Datenaufnahme-Szenarien
- Programmierbare Datenverarbeitung

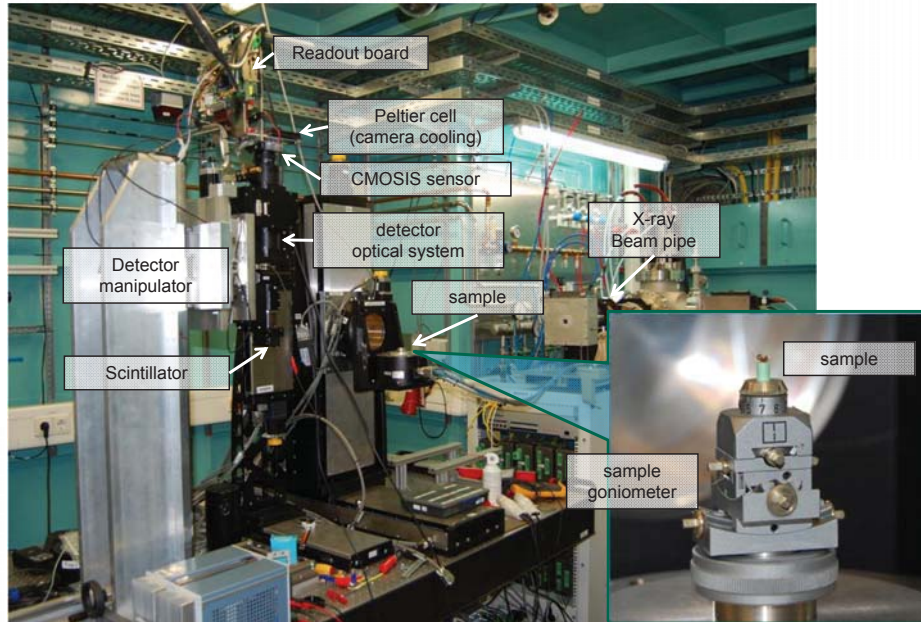


14

SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

Messung an der ANKA Beamline



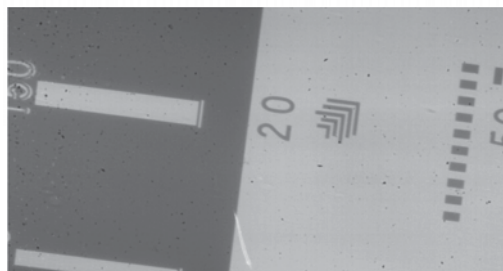
15

SEI Tagung 12.-14.März 2012 M.Balzer

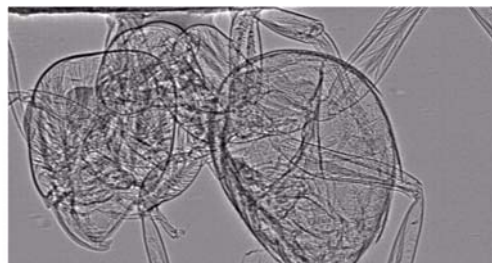
Institute for Data Processing and Electronics (IPE)

Ergebnisse von der 1. Messung an der Beamline

- Auflösung des Detektors (Szintillator, Optik, Sensor)



- Rekonstruierte Ameise



16

SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

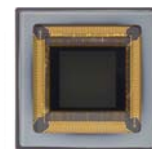
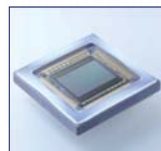
Optische Übertragung der Daten

- Optischer CameraLink
 - Begrenzte Datenrate „nur“ 850 Mb/s
- PCIe
 - Probleme bei den käufliche Produkten
 - Umsetzung auf ein optisches Medium
- Ethernet
 - Standard
 - Relativer hoher Preis für 10G Geräten
 - 10Gb/40Gb
- Infiniband
 - Standard
 - 40Gb
 - Geringer Protokoll Overhead
 - Preiswerte Geräte (Router)



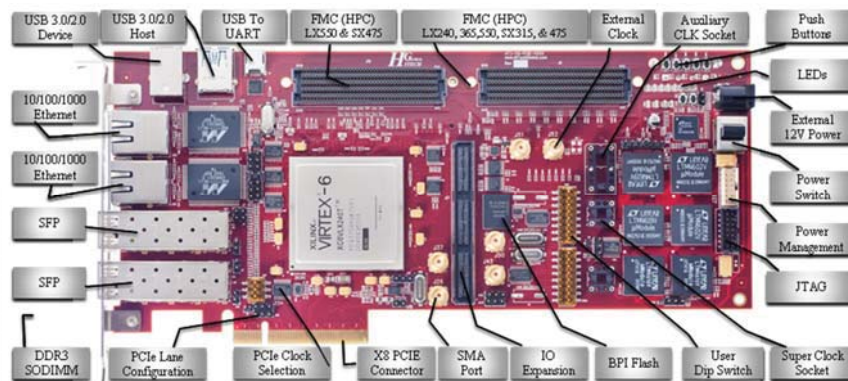
Weitere Bildsensoren

- AWAIBA XF-2336
 - 500 fps
 - 2368 x 1728 aktive Pixel
 - 10 Bit Auflösung
 - Maximale Datenrate 2,5 GB/s
 - Datenschnittstelle 160 Signale mit 133 MHz
 - Effektive $7 \times 7 \mu\text{m}^2$
- IDT AM1X5
 - 5000 fps
 - 1014 x 1024 aktive Pixel
 - 10 Bit Auflösung
 - Maximale Datenrate 6,5 GB/s
 - Datenschnittstelle 160 DDR-Signale mit 180 MHz
 - Effektive Pixelgröße $13 \times 13 \mu\text{m}^2$



Neue FPGA Evaluation Board

- Erhöhte Anforderung an die Auslesegeschwindigkeit
- Schnelleres Speicher Interface (bis 8 GByte)
- PCIe Gen 2.0 x8
- Schnellere interne Logik
- HiTech Global XC6VLX240T-2

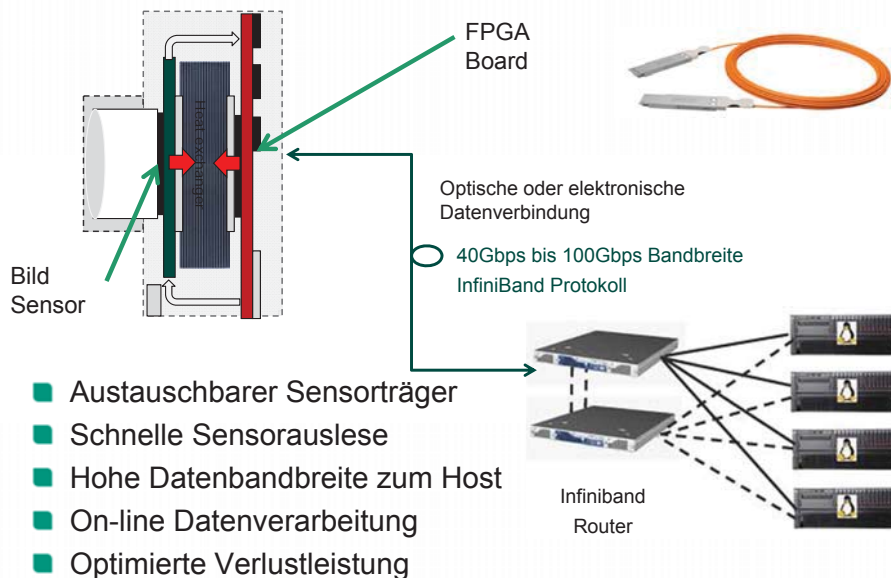


19

SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

Embedded KIT Kamera



- Austauschbarer Sensorträger
- Schnelle Sensorauslese
- Hohe Datenbandbreite zum Host
- On-line Datenverarbeitung
- Optimierte Verlustleistung

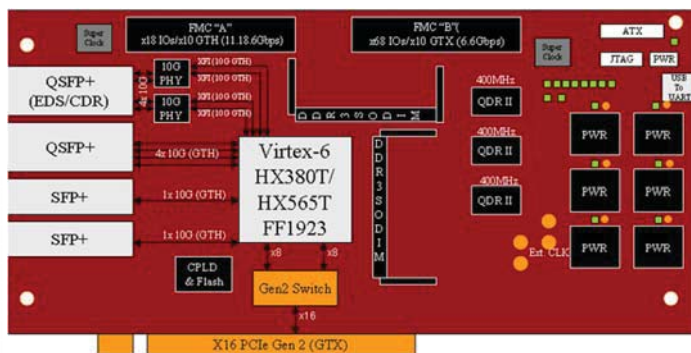
20

SEI Tagung 12.-14.März 2012 M.Balzer

Institute for Data Processing and Electronics (IPE)

Evaluierung der QSFP Schnittstelle

- QuadSFP+ 40 Gb/s (40Gb Ethernet, 20G/40G Infiniband)
- Neue Hochgeschwindigkeitsschnittstelle
- FPGA Board als Datenaufnahmesystem
- Schneller Datentransfer in den Rekonstruktionsrechner (PCIe Gen2 x16)
- HiTech Global mit FPGA XC6VHX380T-2



21

SEI Tagung 12.-14. März 2012 M. Balzer

Institute for Data Processing and Electronics (IPE)

On-line Kompression und Reduktion

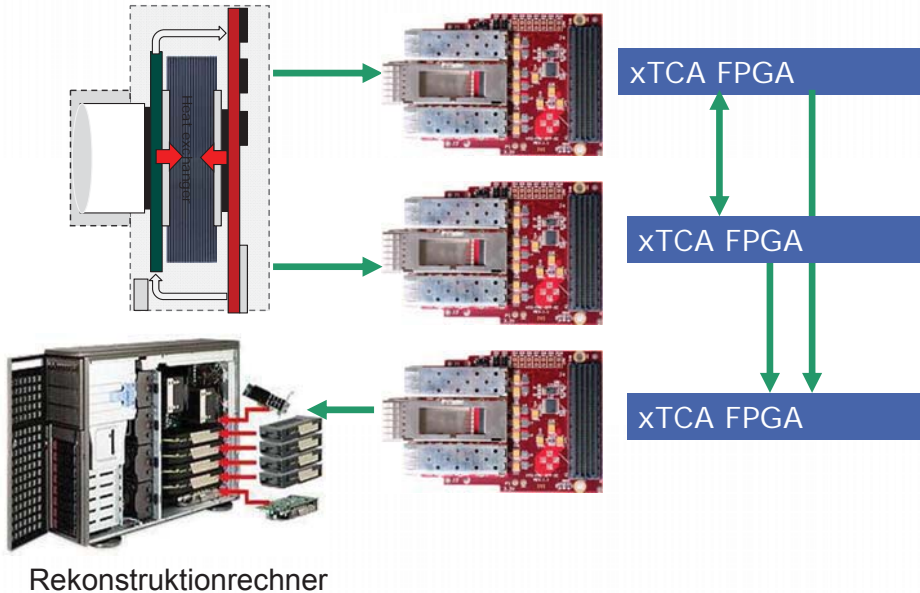
- Kompression
 - Verlustfreie Kompression
 - Verringerung der Anforderungen an die Bandbreite
 - Dekompression kann in parallelen Systemen effektiv durchgeführt werden
 - (Verlustbehaftete Kompression)
- Reduktion
 - Externe Trigger
 - Transienten Recorder
 - Start-Stop Triggierung
 - Selbsttriggierung
 - Differenzverfahren
 - Schwellwerte
 - Objektverfolgung (Region of Interest, ROI)

22

SEI Tagung 12.-14. März 2012 M. Balzer

Institute for Data Processing and Electronics (IPE)

xTCA oder μ TCA System für die parallele On-line Verarbeitung



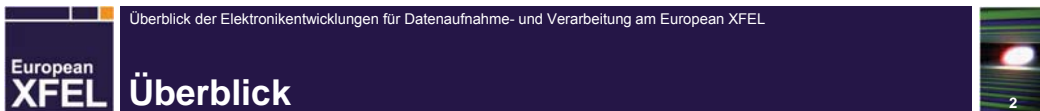
Zusammenfassung

- Beschleunigung der Tomographie Rekonstruktion auf wenige Minuten
- Datenspeicherung 800 MB/s erreicht
- KIT Kamera für UFO einsetzbar
 - 340 fps
 - > 12 Gb/s
- Entwicklung schnellerer Kameras (fps)
- Erprobung von optischen QSFP Verbindungen
- Herausforderungen bei der On-line Vorverarbeitung
- Evaluierung von parallelen Systemen (xTCA)



Überblick über die Entwicklungen der Datenaufnahme- und Verarbeitungssysteme für Photon Beamlines und Experimente am European XFEL

Patrick Gessler
European XFEL



- Kurze Einführung zum European XFEL
- Benötigte Diagnostik und Detektoren für ein Experiment
- Elektronikentwicklungen und Standards
- Verarbeitungs- und Datenflussprinzip
- VETO System zur Datenreduktion und Speicheroptimierung
- Firmware Entwicklung für FPGAs
- Ausblick





Länge	3,4km
Tiefe	6m bis 38m
Kosten (Bau, Inbetriebnahme)	> 1 Mrd. Euro
Wellenlänge Photonen	0.05nm – 6nm
Nutzerbetrieb	ab 2016

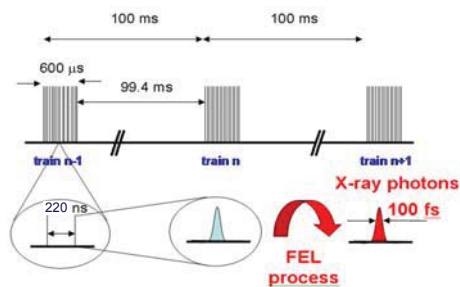
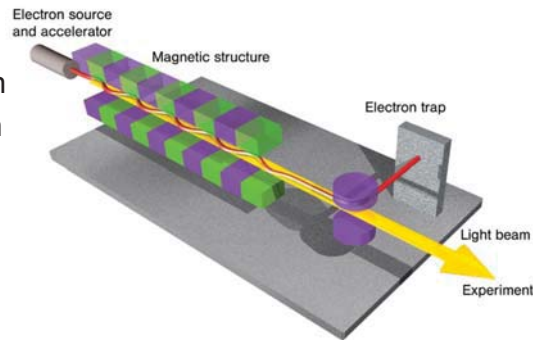
14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler



14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler



- Elektronenbündel erzeugen
- Beschleunigen und komprimieren
- Im Undulator Photonen erzeugen
- Elektronen werden "gedumped"
- Photonen über Beamlines zu Experimenten transportieren



- HF Beschleuniger Feld gepulst
- Alle 10Hz für 600us
- Bis zu 2700 Elektronenbündel (Bunche) pro 600us Puls
- Bunchfrequenz ist 4,5MHz

- Beam Position Monitors
- Foil Backscattering Detectors
- Variable Screens with Camera
- Optical Microscopes
- Intensity Monitors
- Spectrometer
- Peak Detectors
- 2D Detectors



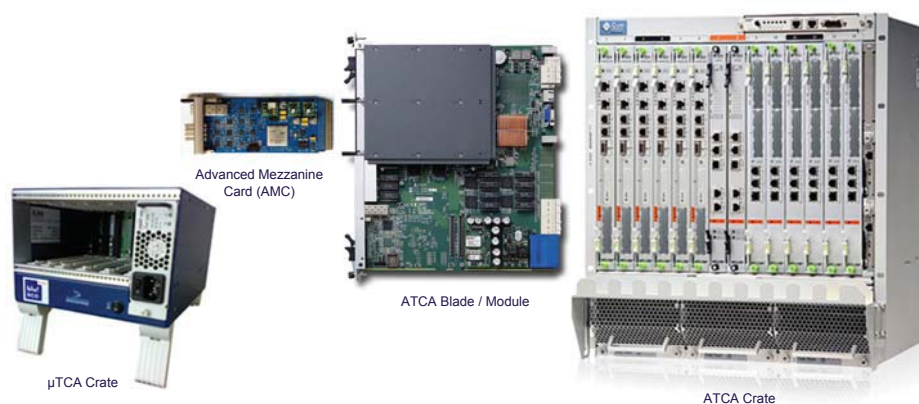
Chamber

- Hohes Datenvolumen zu verarbeiten und speichern

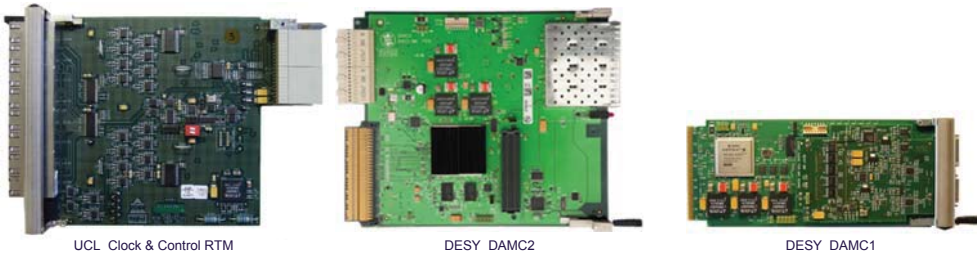
Detector type	Sampling	Data/pulse	Data/train	XFEL/sec	LCLS/sec
1 Mpxl 2D camera	4.5 MHz	~2 MB	~1 GB	~10 GB	~300 MB
1 channel digitizer	5 GS/s	~2 kB	~6 MB	~60 MB	~0.2 MB

- Möglichkeit sämtliche Messdaten zu speichern (sollte aber vermieden werden)
- Skalierbar (Kombination und Anzahl der Anwendungen)
- Fernwartung und Überwachung
- Erweiterung und Austausch während des Betriebs
- Frühe Verarbeitung und Datenreduktion

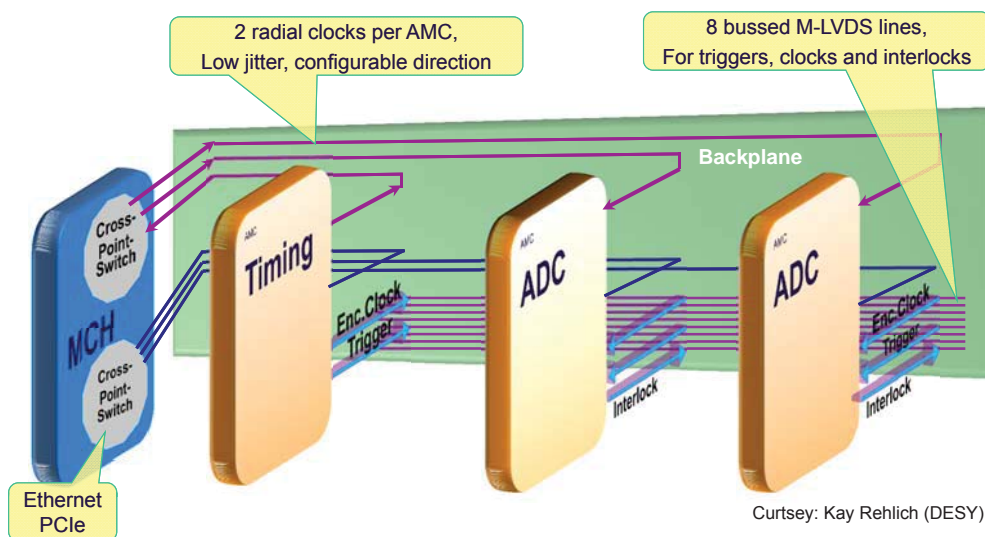
- Advanced Telecommunication Computing Architecture (ATCA)
- Advanced Mezzanine Card (AMC)
- Micro Telecommunication Computing Architecture (μ TCA)



- xTCA nicht für Physikanwendungen vorgesehen (e.g. Timing, Interfacing, ADCs)
- xTCA for Physics Group at PCI Industrial Computing Manufacturers Group (PICMG)
 - Mitglieder sind von Instituten und der Industrie
 - Definition von Erweiterungen des Standards für unsere Anwendungen
 - Timing interface auf der Backplane
 - High speed modules interconnections
 - Mehr Platz → Double size modules
 - Modular interfacing and shaping → Rear Transition Modules (RTMs)

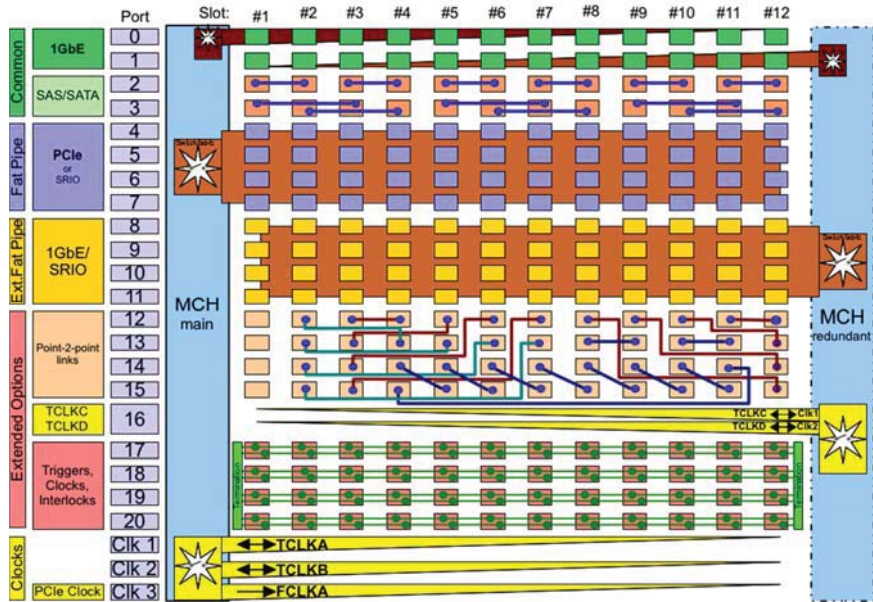


27th October 2011 | DAC Meeting | AER19 5 21 | Patrick Gessler



14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler

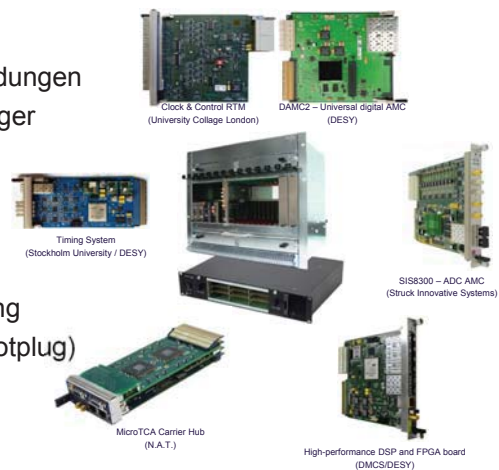




14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler

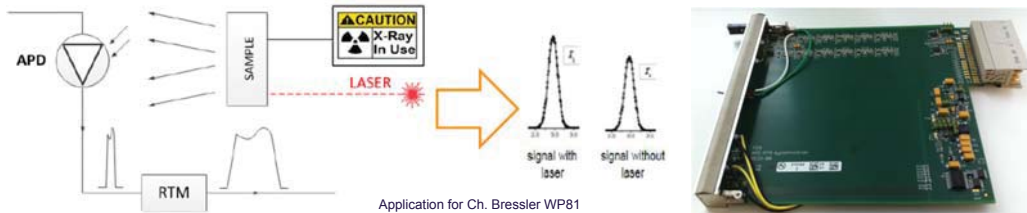


- Verwendung des neuen MTCA.4 Standards erlaubt
 - Hohe Bandbreite zwischen
 - Boards und CPU via PCIe
 - Boards via Punkt-zu-Punkt Verbindungen
 - Synchronisierung via Timing Empfänger
 - Trigger
 - Clocks
 - Maschinenparameter
 - Bunchstruktur
 - Systemüberwachung und Fernwartung
 - Modulaustausch während Betrieb (Hotplug)
 - Funktionserweiterung via RTMs

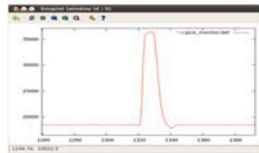


14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler





- Using a COTS ADC Module from Struck - widely used at XFEL (LLRF, BPM,...)
- A customized RTM had been developed in collaboration with Peter Goettlicher (DESY) for
 - Interfacing to the APD detector
 - Shaping the signal - increasing the pulse length and maintain the amplitude information
- Processing in FPGA or CPU and streaming of data
- System was tested with beam and APD module at Petra III (Good contact to P01 BL)
- Besides the foreseen use in WP81 it serves as an important test bed for
 - VETO source and related signal generation
 - Using Simulink for algorithm implementation
 - Device server implementation for MTCA.4



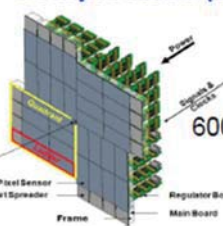
Curtsey: B. Fernandes (XFEL)

AGIPD Adaptive Gain Integrating Pixel Detector (AGIPD)



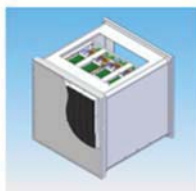
Energy range
3 - 13 keV
Dynamic range
 10^4 @12 keV
Single Photon Sens.
Storage Cells \approx 300

DEPFET Sensor with Signal Compression (DSSC)



Energy range
0.5 - 6 keV (25 keV)
Dynamic range
6000 ph/pix/pulse@1 keV
Single Photon Sens.
Storage Cells \approx 640

Large Pixel Detector (LPD)



Energy range
5 (1) - 20 keV (25 keV)
Dynamic range
 10^5 @12 keV
Single Photon Sens.
Storage Cells \approx 512

Other Detectors

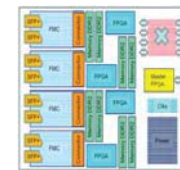
- 0D/1D detectors for high repetition rate applications (e.g. veto, dispersive spectrometers)
- Small areas, low rep. rate, low energy 2D imaging detectors
- Particle detectors (eTOF, iTOF)

- Synchronisierung und Steuerung
 - Clock und Control RTM
 - Synchronisiert durch Timing Empfänger
 - Steuert einen 1MPixel Detektor

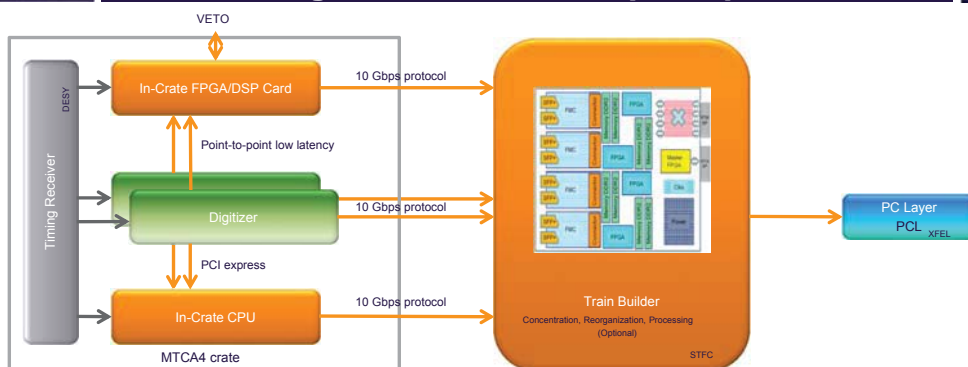
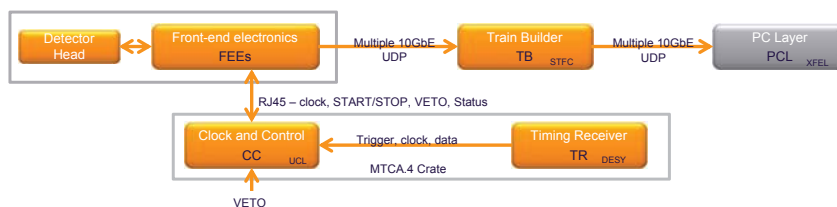
- Daten reorganisieren im Train Builder
 - Teilbilder werden zusammengefügt
 - Ein vollständiger "Train" im Speicher abgelegt
 - Dieser wird per 10GbE an PC Layer übertragen



Prototype Clock and Control RTM (UCL)



Train builder demonstrator board (STFC)

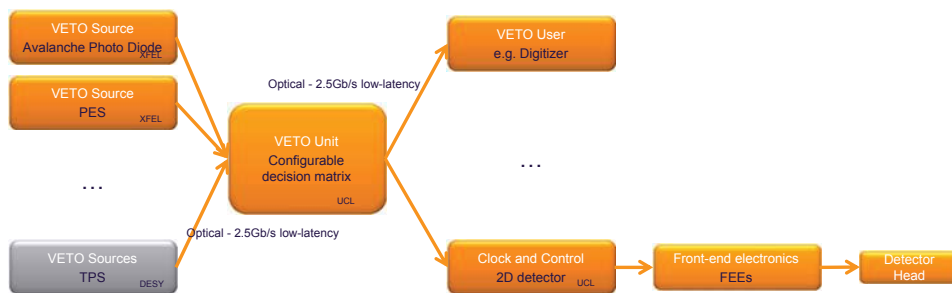


- In-crate Verarbeitung
 - In den FPGAs der Digitizer
 - In der lokalen CPU
 - In einem DSP/FPGA Bord

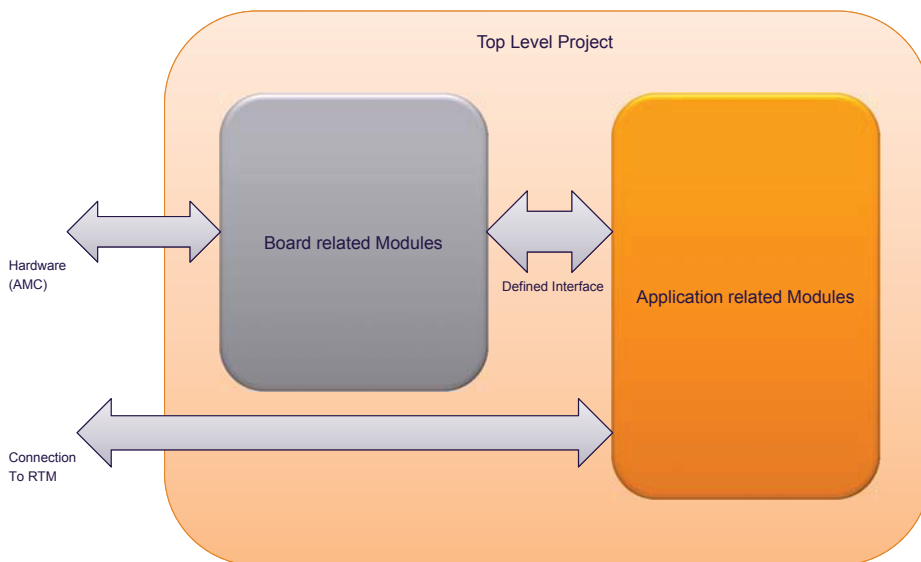
- Verarbeitung mehrerer Quellen
 - Verarbeitung in FPGAs
 - Mehrere Boards
 - Kommunikation zwischen allen FPGAs
 - DDR und QDR RAM

- PC Layer
 - Software
 - CPUs
 - GPUs

- Verbesserung der 2D Detektor Bildqualität
 - Limitierter Speicherplatz im ASIC (~300-700 Bilder)
 - Überschreiben schlechter Bilder im ASIC
- Datenreduktion
 - Verwerfen von unnützen und schlechten Daten
- Umsetzung
 - FPGAs der Diagnostik und Detektoren liefern Messwerte in „Echtzeit“
 - Zentrale VETO Unit per Experiment Entscheidet über die Güte pro Bunch
 - FPGAs der Diagnostik und Detektoren empfangen und verarbeiten die Entscheidung



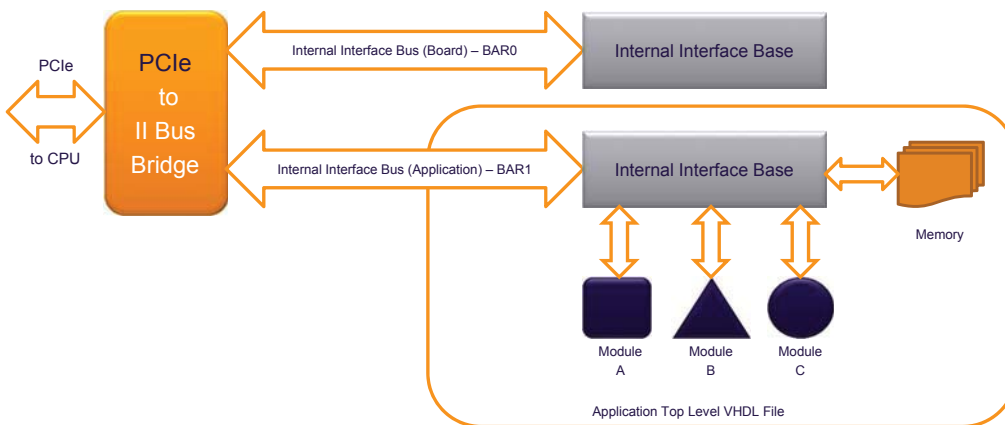
14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler



Original idea from Wojciech Jalmuzna (DESY)
(Board Support Package)

14th March 2012 | SEI-Tagung Frühjahr 2012 | HZDR | Patrick Gessler





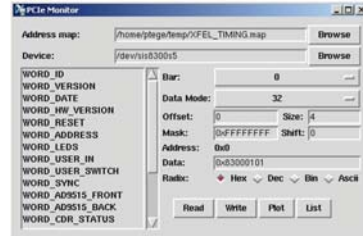
- Vereinfacht Register Zugriff und Memory Mapping
 - Beispiel: Register und Speicher Definition

```
constant VIIItemDeclList : TVIIItemDeclList := (
  VII_WORD , WORD_PROJ_MAGIC_NUM      , 32 , 1 , VII_WNOACCESS , VII_REXTERNAL , 0 , VII_UNSIGNED,
  VII_WORD , WORD_PROJ_ID             , 32 , 1 , VII_WNOACCESS , VII_REXTERNAL , 1 , VII_UNSIGNED,
  VII_WORD , WORD_PROJ_VERSION        , 32 , 1 , VII_WNOACCESS , VII_REXTERNAL , 2 , VII_UNSIGNED,
  VII_WORD , WORD_PROJ_DATE           , 32 , 1 , VII_WNOACCESS , VII_REXTERNAL , 3 , VII_UNSIGNED,
  VII_WORD , WORD_PROJ_RESERVED       , 32 , 1 , VII_WNOACCESS , VII_REXTERNAL , 4 , VII_UNSIGNED,
  VII_WORD , WORD_PROJ_RESET          , 1  , 1 , VII_WACCESS  , VII_RINTERNAL , 5 , VII_UNSIGNED,
  VII_WORD , WORD_PROJ_NEXT           , 32 , 1 , VII_WNOACCESS , VII_REXTERNAL , 6 , VII_UNSIGNED,
  VII_AREA , AREA_MEM_0               , 16 , 1024 , VII_WNOACCESS , VII_REXTERNAL , 1024 , VII_UNSIGNED,
  VII_WORD , WORD_DUMMY_RESET         , 1  , 1 , VII_WACCESS  , VII_REXTERNAL , 10 , VII_UNSIGNED,
  VII_WORD , WORD_DUMMY_ENABLE        , 1  , 1 , VII_WACCESS  , VII_RINTERNAL , 11 , VII_UNSIGNED,
  VII_WORD , WORD_DUMMY_MAX           , 8  , 1 , VII_WACCESS  , VII_REXTERNAL , 12 , VII_UNSIGNED,
  VII_WORD , WORD_DUMMY_STATUS        , 9  , 1 , VII_WNOACCESS , VII_REXTERNAL , 13 , VII_UNSIGNED
);
end PKG_ADDRESS_SPACE_USER;
```

- Beispiel: Verwendung von Registern
 - DUMMY_ENABLE <= IIGetItem(CON_II_AS, SIG_VECINT, WORD_DUMMY_ENABLE, 0)(0)
- Verwendung in Software (z.B. Server)
 - Konvertieren der Tabelle in MAP oder Header Datei

■ PCIe Monitor (Kommunikation mit Hardware)

- Nutzt MAP Datei
- Register zugriffe
- Plotten von Daten
- Tabellen darstellen



■ UTCA Monitor (Kommunikation mit PCIe Hotplugging Treiber)

- Trennen und verbinden mit PCIe Endpunkt
- Informationen über Installierte AMCs
- Zeitweise Testtreiber Zuweisung



■ Kommandozeilen Programme

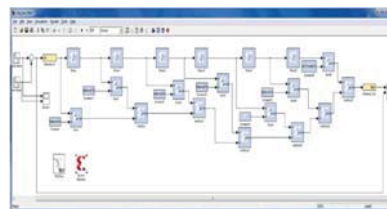
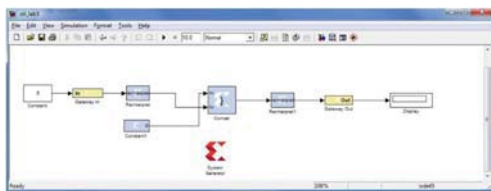
- pcie_rd, pcie_wr, firmware upload
- Skripte für bestimmte AMCs

■ Verwenden von Simulink als Programmierumgebung für Algorithmen

■ Vorteile

- Die meisten Institute haben bereits Matlab/Simulink
- Einfache Block und Flussorientierte Programmierweise
- Spezielle Blöcke von Xilinx bereitgestellt (durch Xilinx System Generator)
- Einfache Simulationsmöglichkeit
 - Eingabe auch durch Matlabdaten möglich
 - Verwendung der selben Quantisierung wie im FPGA
 - Hardware co-simulation möglich

■ Erlaubt Einbinden der Wissenschaftler bis zur Implementierung



- Entwicklung des FPGA Frameworks für Algorithmen
 - vielleicht gemeinsam mit anderen Instituten?
- Organisieren eines 10GbE+ in FPGA Workshops (~May)
- Organisieren eines MTCA.4 Workshops (mit DESY) für
 - Andere Helmholtz Institute
 - Mit Industrie
 - Vorstellen von Implementierungen und Projekten
 - Erfahrungsaustausch
 - ...
 - Besteht Interesse?

Zeit für Fragen und Diskussionen...



Developed High Performance Electronics for the Gas Filled Stopping Cell and MR-Time-of-Flight-MS of the FRS Ion Catcher at GSI

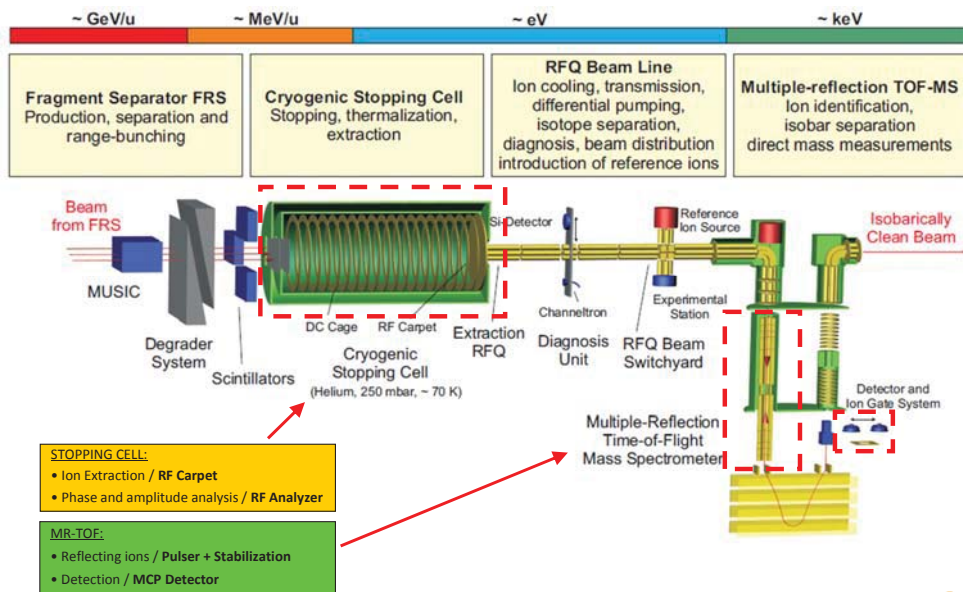
Samuel Ayet San Andrés

s.ayet@gsi.de

S Studiengruppe für
Elektronische Instrumentierung

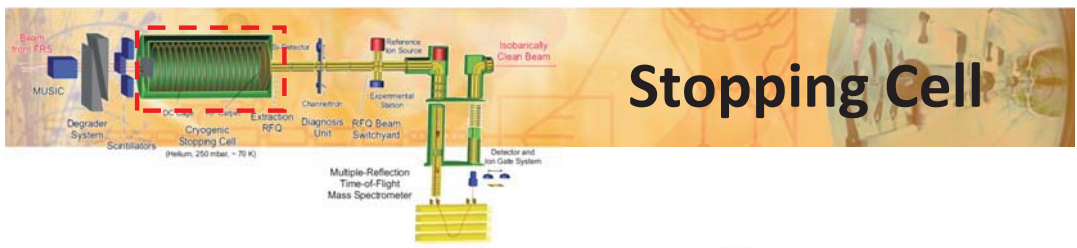


Ion Catcher Overview

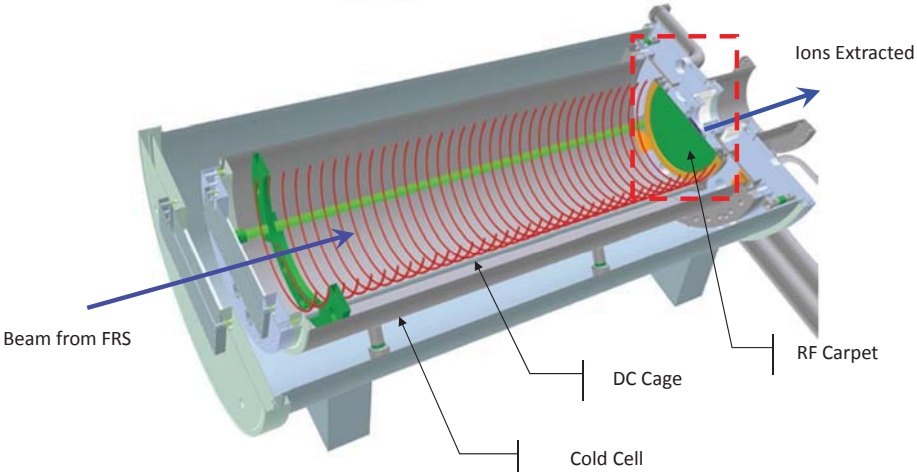


W.Plaß, T. Dickel

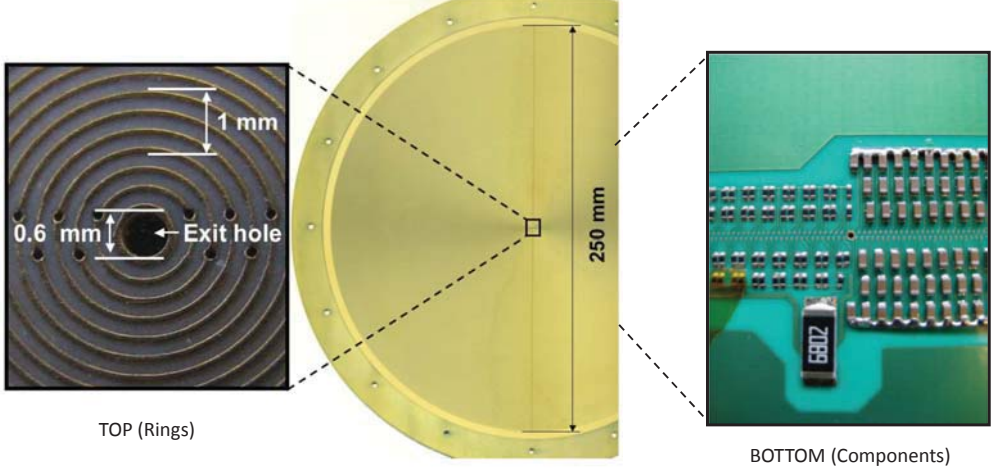




Stopping Cell



RF Carpet

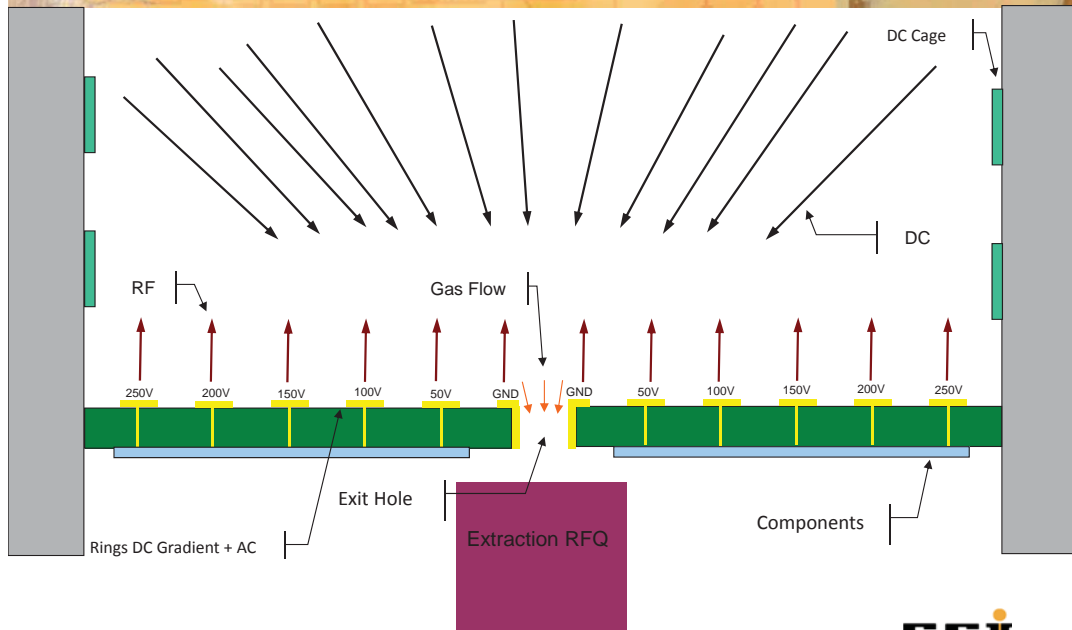


- SPECIFICATIONS:**
- 3 Layers RO4003 Material
 - 500 Rings
 - 1.2mm thickness

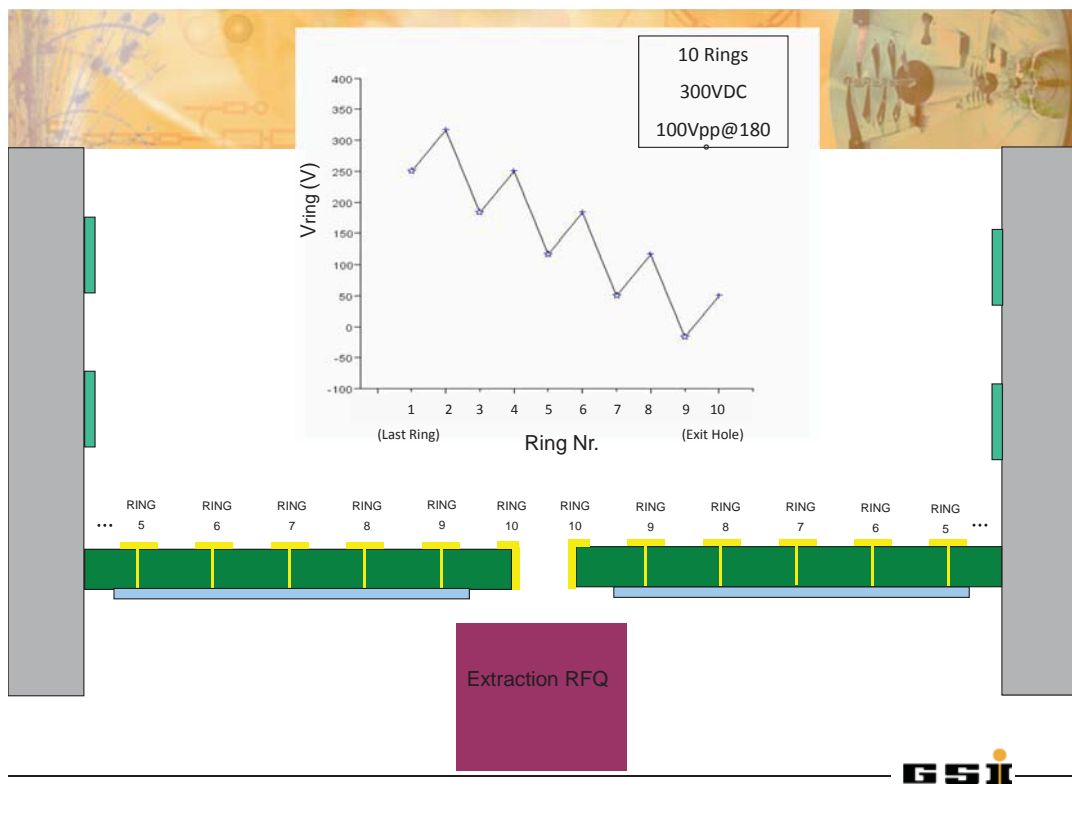
In collaboration with
F. Schreuder (KVI)



Forces: DC + RF + Gas Flow

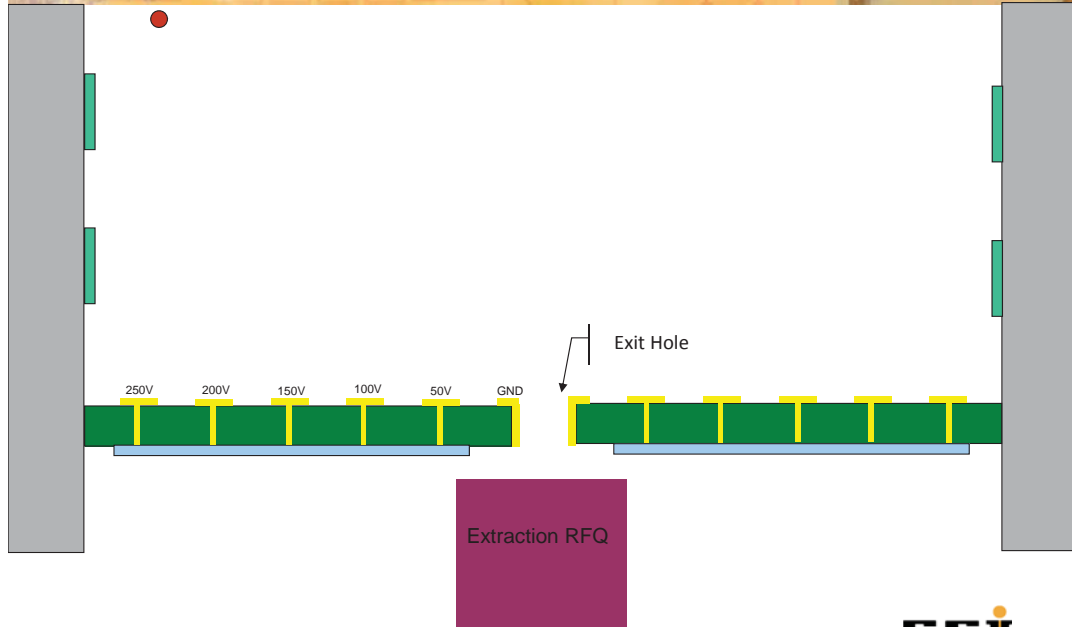


GSII



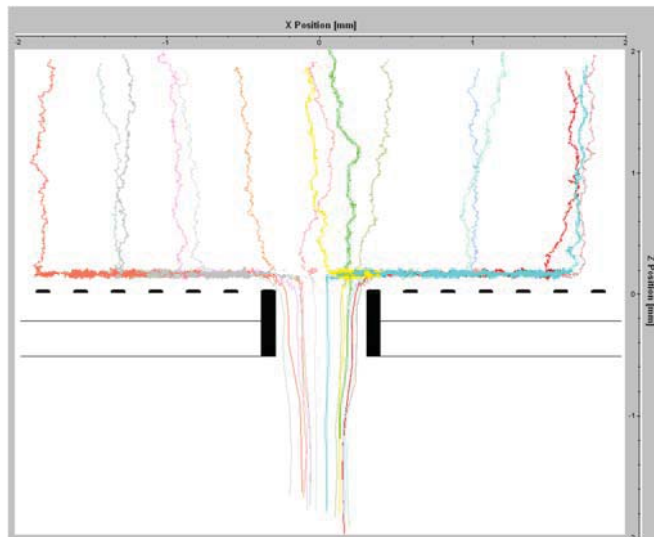
GSII

Ion Guiding



GSII

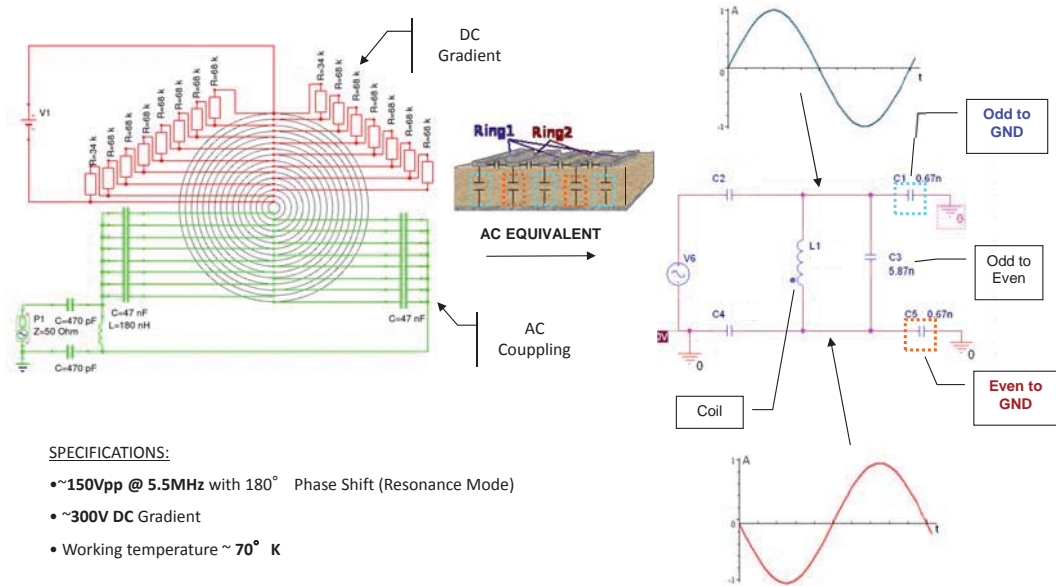
Simulated Ion Trajectories



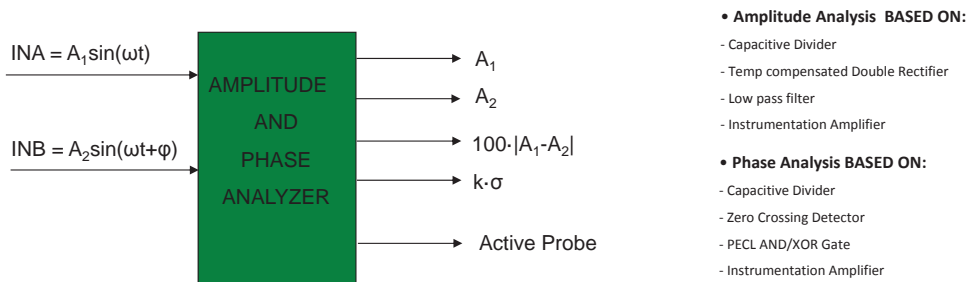
P.Reiter
SIMION Simulations

GSII

RF Carpet Schematic

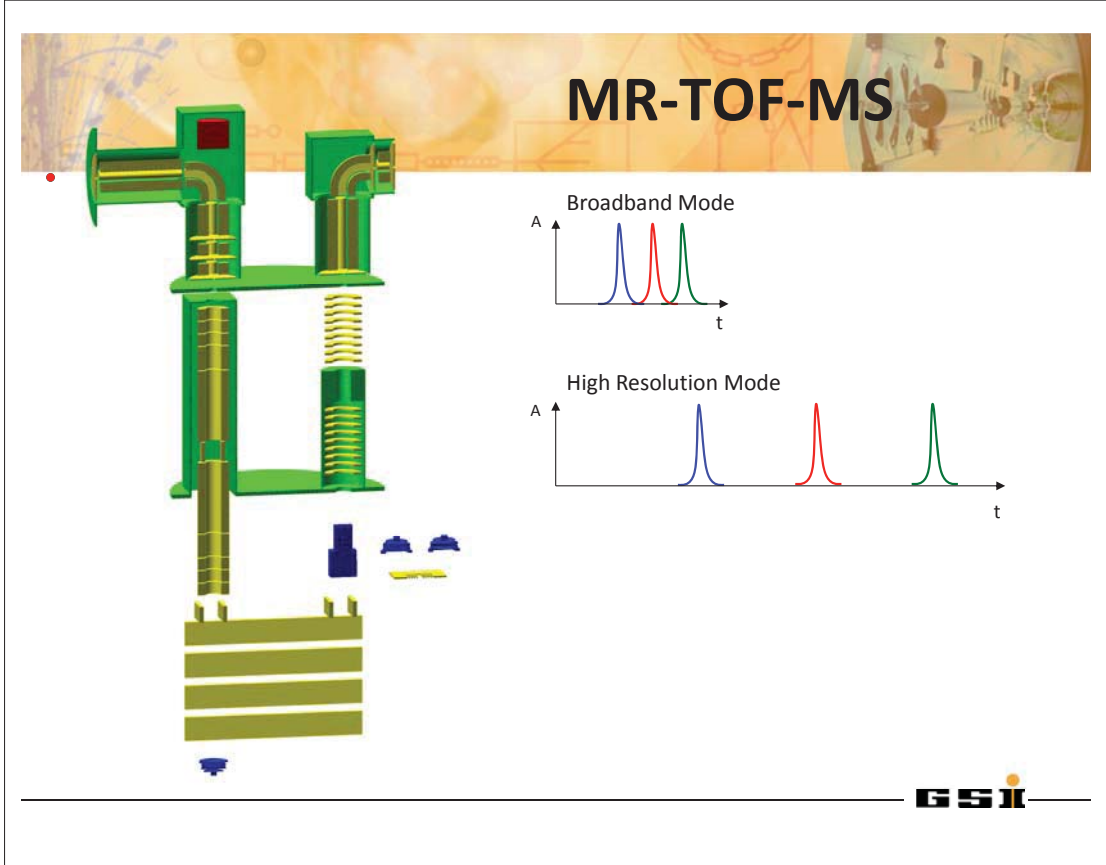
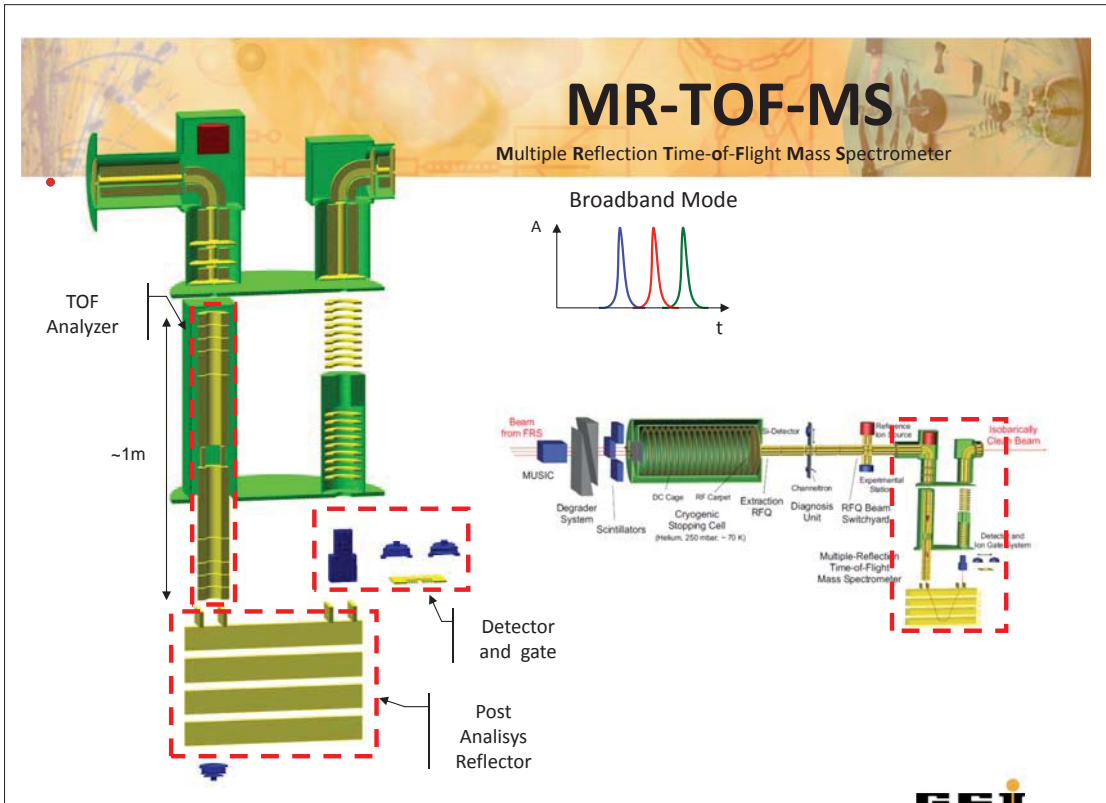


Amplitude and Phase Analyzer

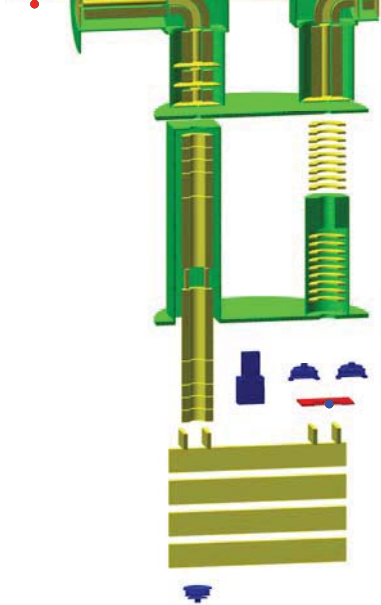


- SPECIFICATIONS:**
- Input frequency range: from 100kHz to 10MHz
 - Input voltage range: from 200mVpp to 10Vpp @ 2MHz
 - Amplitude Error Accuracy of < 0.01mV
 - Phase Error Accuracy < ±0.1° (2MHz: ±0.1° ~ ±135ps)

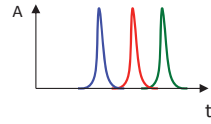




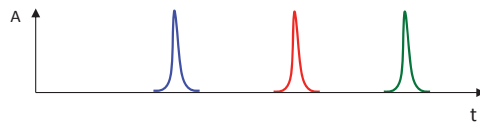
MR-TOF-MS



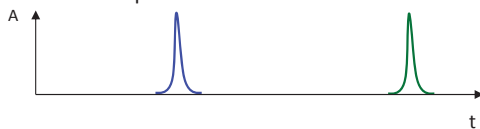
Broadband Mode



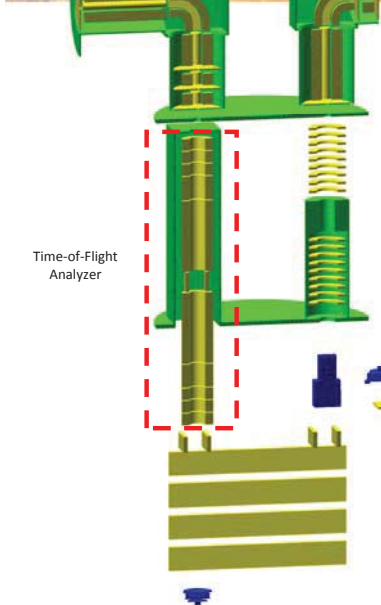
High Resolution Mode



Isobar Separation Mode



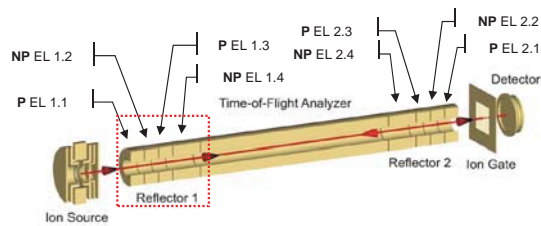
MR-TOF-MS



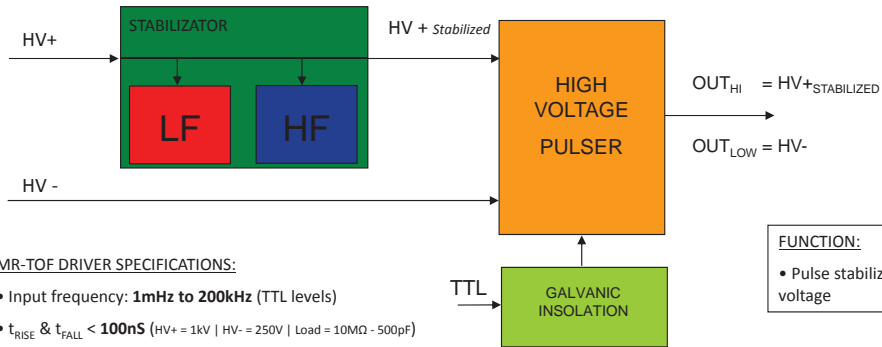
Time-of-Flight Analyzer

Time-of-Flight Analyzer:

- 8 Stabilized Electrodes | Pulsed & Non-Pulsed
- 4 pulsed (~kV) | 4 non pulsed (~kV)
- Symmetric → EL 1.X = EL 2.X (half turn delayed)



Pulsed Reflector Electrode

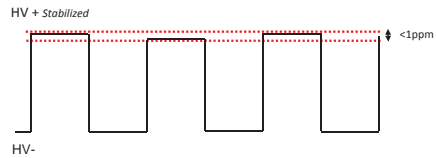


MR-TOF DRIVER SPECIFICATIONS:

- Input frequency: **1mHz to 200kHz** (TTL levels)
- t_{RISE} & $t_{FALL} < 100ns$ (HV+ = 1kV | HV- = 250V | Load = 10MΩ - 500pF)
- Minimum pulse width of **3.5μs** for extreme duty cycle
- Voltage stability (HV+) under **1ppm** for 30 seconds
- Maximum differential voltage HV+ to HV- of **1.7kV (4kV)**
- Maximum absolute voltage HV+ and HV- of **3kV (6kV)**
- Temperature Coefficient $\sim 25ppm/^{\circ}C$

FUNCTION:

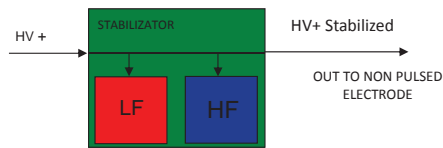
- Pulse stabilized high voltage



Collaboration with U.Czok

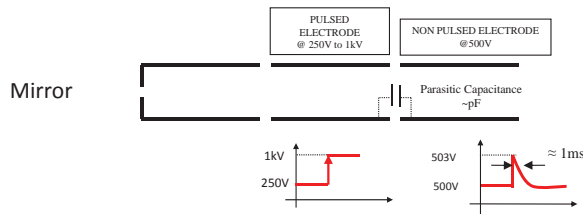


Non Pulsed Reflector Electrode



FUNCTION:

- Reduce noise generated by neighbor pulsing electrode.



Read Out & Temperature Stabilization

MR-TOF READ OUT SPECIFICATIONS:

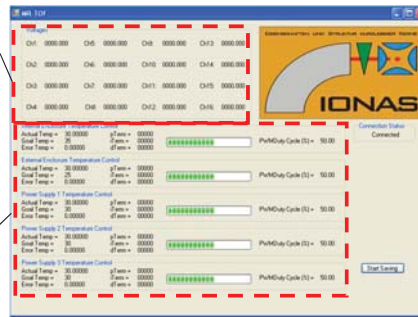
- Max DC input 5kV
- Overall resolution > 10mV (~ 19 bits @ 5V_{REF})
- Number of input channels 16
- Sampling rate ~ 10 SPS
- USB PC Connection with Visual C++ User Interface



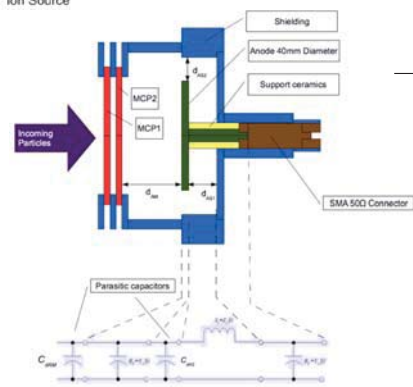
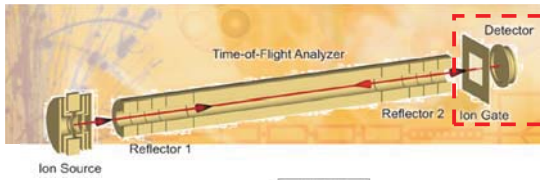
MR-TOF TEMPERATURE STABILIZATION SPECIFICATIONS:

- Temperature stability > 0.1° C
- Cooling (Fan-Speed) & Heating (50W Heaters) with PID control

Voltage Readout
Temperature Regulation

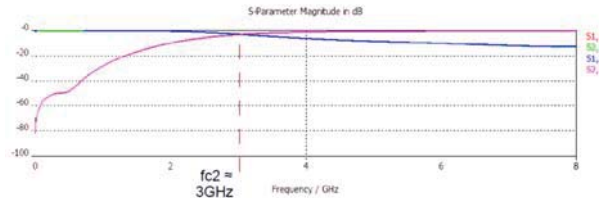
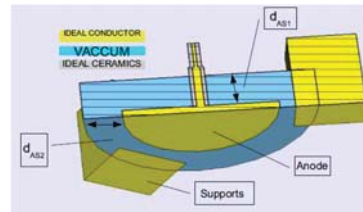


Design Anode



High Frequency Equivalent Circuit

Impedance Matching



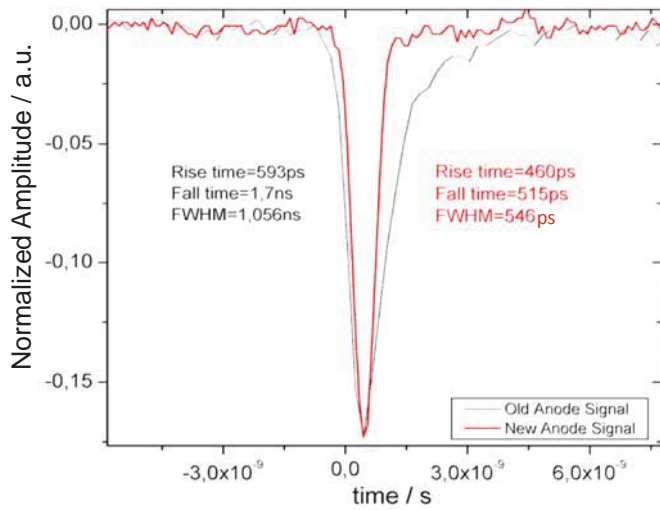
Simulation of S-Parameters – CST Microwave Studio

DESIGN CONSIDERATIONS:

- No ringing effects
- Sharp and symmetric signals
- High bandwidth (~ 2GHz)



Results Anode

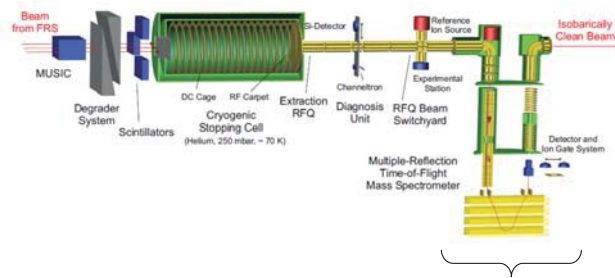


Built Anode

Timing Comparison



Outlook



MR-TOF:

- Mobile Time-of-Flight Mass-Spectrometry (MO-TOF-MS) at **Ambiprobe Project**
- **TRIUMF** Canada's National Laboratory for Particles and Nuclear Physics



Ion Catcher (Stopping Cell + MR-TOF):

- Low Energy Branch of Super-FRS at FAIR



Nachweis von Mikrowellenemissionen aus hochenergetischen kosmischen Luftschauern

Das CROME-Projekt

S. Baur¹, J. Blümer¹, R. Engel¹, A. Haungs¹, T. Huege¹, K.-H. Kampert², H. Klages¹, M. Kleifges¹, O. Krömer¹, S. Mathys², P. Neunteufel¹, M. Ludwig¹, J. Pekala⁴, J. Rautenberg², M. Riegel¹, M. Roth¹, F. Salamida³, H. Schieler¹, J. Stasielak⁴, R. Smida¹, M. Unger¹, M. Weber¹, E. Werner¹, H. Wilczyński⁴, J. Wochele¹

¹Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

²Bergische Universität Wuppertal, Wuppertal, Germany

³Università dell'Aquila and INFN, L'Aquila, Italy

⁴Institute of Nuclear Physics PAN, Krakow, Poland

Gliederung

- **Kosmische Strahlung – Kosmische Luftschauer**
- **etablierte Nachweismethoden und Experimente**
- **Radiodetektion kosmischer Schauer**
- **Das CROME-Experiment**
- **erste Ergebnisse**

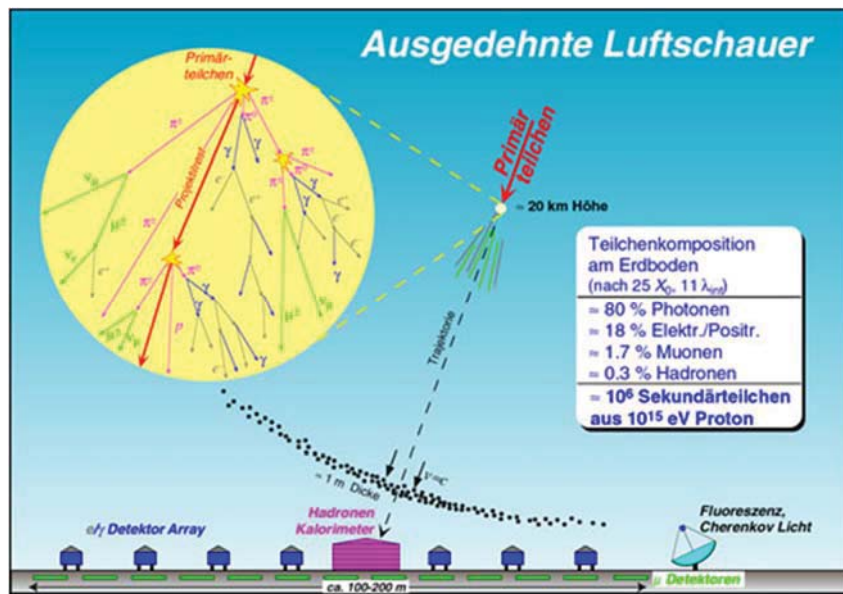
Kosmische Strahlung



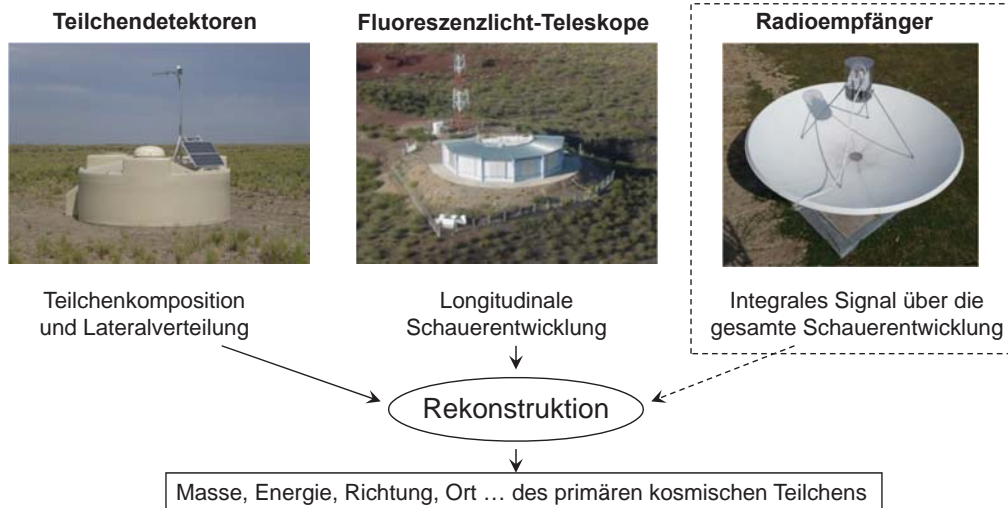
Victor F. Hess, 1912 (Nobelpreis 1936)

- vorwiegend ionisierte Atomkerne (H, He, ... Fe)
- Energie: $E = 10^{10} \text{ eV} \dots 10^{21} \text{ eV}$ (LHC: $7 \cdot 10^{12} \text{ eV}$)
- Teilchenflussdichte fällt mit $dN/dE \sim E^{-3}$
- sehr 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



Nachweismethoden für kosmische Luftschauer



Ziel: Untersuchung und Etablierung der Radioobservation als ergänzende Nachweismethode für kosmische Schauer

Radiodetektion kosmischer Schauer

Radioemissionsmechanismen kosmischer Schauer



• Geomagnetische Strahlungseffekte

- Geosynchrotronstrahlung
- kontinuierliche Ladungstrennung
- translatorische Transversalströme

• molekulare Bremsstrahlung

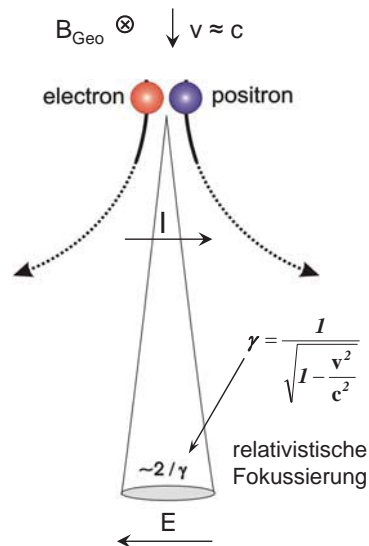
- Streuung an Molekülen der Erdatmosphäre

• Askaryan-Effekt

- Positronen rekombinieren
- negativer Ladungsüberschuss
- Tscherenkow-Emission

• Radioreflexionen an der Schauerfront (RADAR)

- durch Einstrahlung leistungsstarker Sender
- extreme Dopplerverschiebung (MHz → GHz)



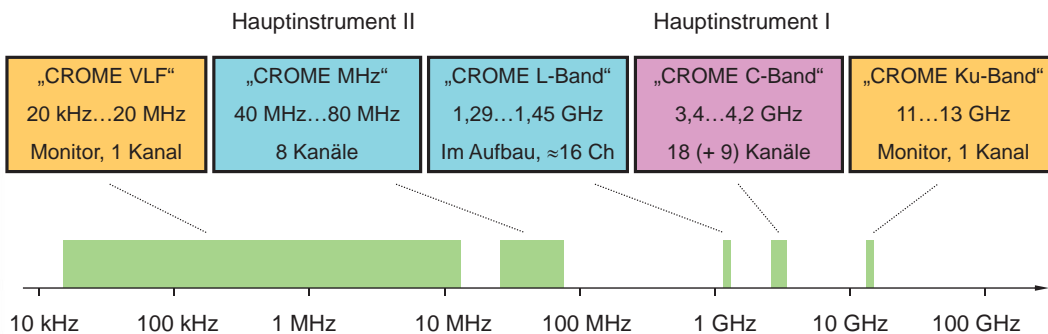
Das CROME-Experiment



CROME: Cosmic Ray Observation via Microwave Emission

- Nahezu alle älteren Radioexperimente (LOPES, AERA, CODALEMA...) arbeiten im 40...80 MHz Bereich
- CROME arbeitet zusätzlich im GHz-Bereich (L-Band, C-Band) und hat Monitorkanäle von 20 kHz...13 GHz

Vorteil: Weitergehende Analyse der zu Grunde liegenden Emissionsmechanismen

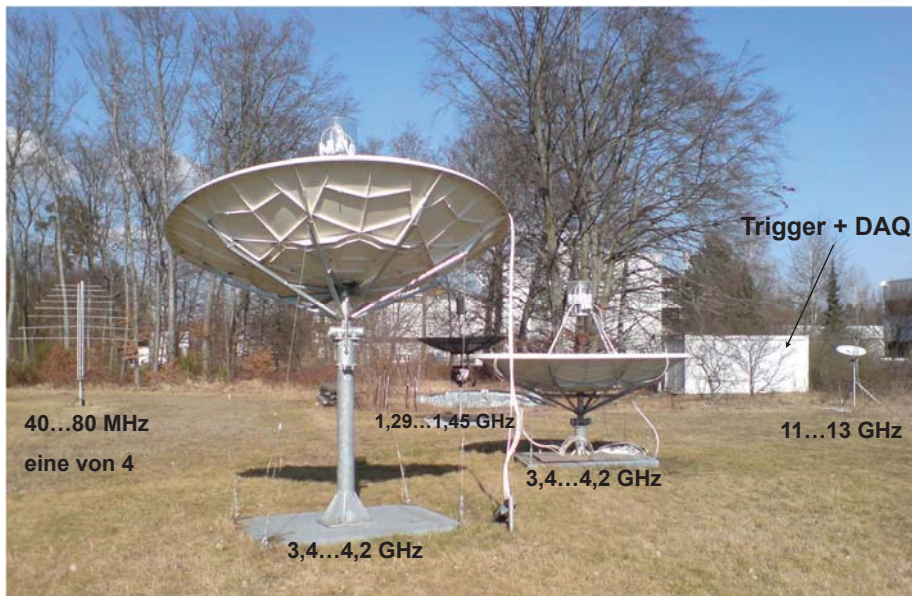


Lageplan KASCADE-Grande und CROME



- 37 KASCADE Grande Teilchendetektorstationen, 700 m x 700 m

Antennenfeld des CROME-Experiments



C-Band Antennen des CROME-Experiments



- zwei C-Band Parabolantennen
- 3,4 m Durchmesser
- 9 Empfänger pro Spiegel (18 Kanäle)
- C-Band: 3,4...4,2 GHz
- Ausrichtung: vertikal and 15° nach Nord
- Strahlbreite $\approx 2^\circ$



11 | Oliver Krömer | Institut für Prozessdatenverarbeitung und Elektronik | 07.04.2008

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



Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

40...80 MHz-Antennen des CROME-Experiments



- Logarithmisch-periodische Dipolantenne
- 40...80 MHz
- Nord-Süd- und Ost-West-Polarisation

- 4 Antennen (8 Kanäle) in ca. 50 Meter Abstand
⇒ Triangulation bzw. Phased-Array

- Antennencharakteristik ist
 - frequenzunabhängig
 - unabhängig von den Bodeneigenschaften⇒ einfach zu kalibrieren
⇒ geringe Kalibrierunsicherheit



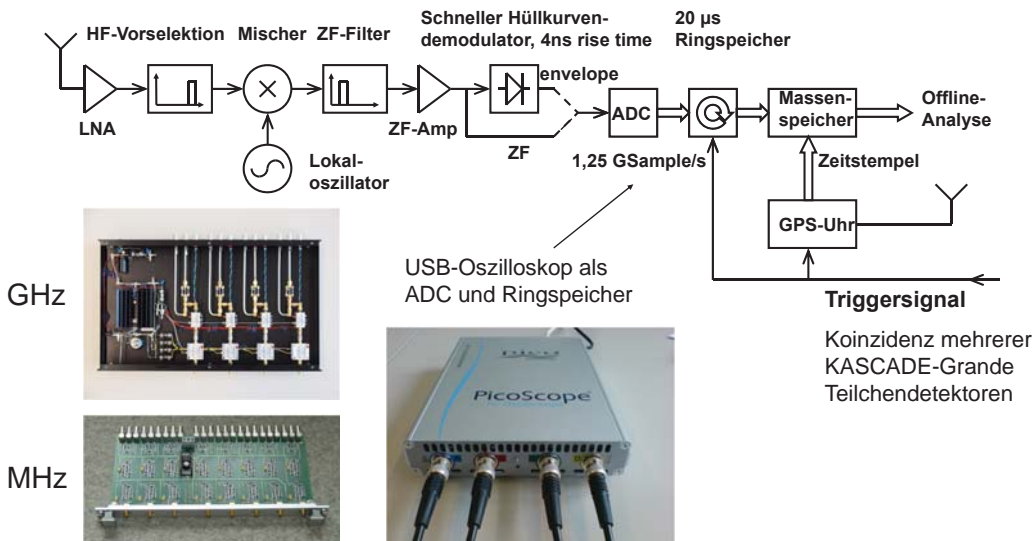
12 | Oliver Krömer | Institut für Prozessdatenverarbeitung und Elektronik | 07.04.2008

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

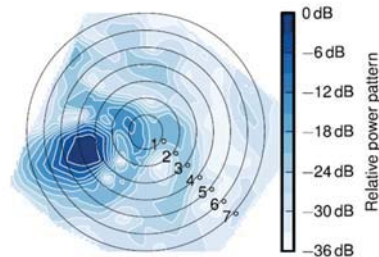


Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

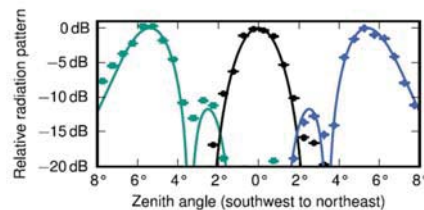
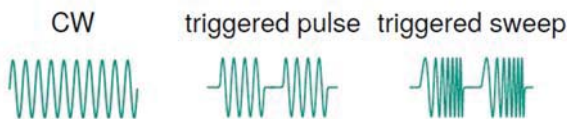
Signalverarbeitung des CROME-Experiments



Kalibration des CROME-Empfängers mittels eines fliegenden Mikrowellensenders - „Radiocopter“



- Radiocopter positionsgeregelt mittels Differential-GPS
- 6 mW, 3...4 GHz, Flughöhe ≈ 300 m
- getriggert durch GPS-Sekundentakt
- Betriebsarten:



Spezielle Entwicklungen aus dem HF-Labor



⇒ TUNKA-Rex ⇒ LOPES-3D



Ergebnisse

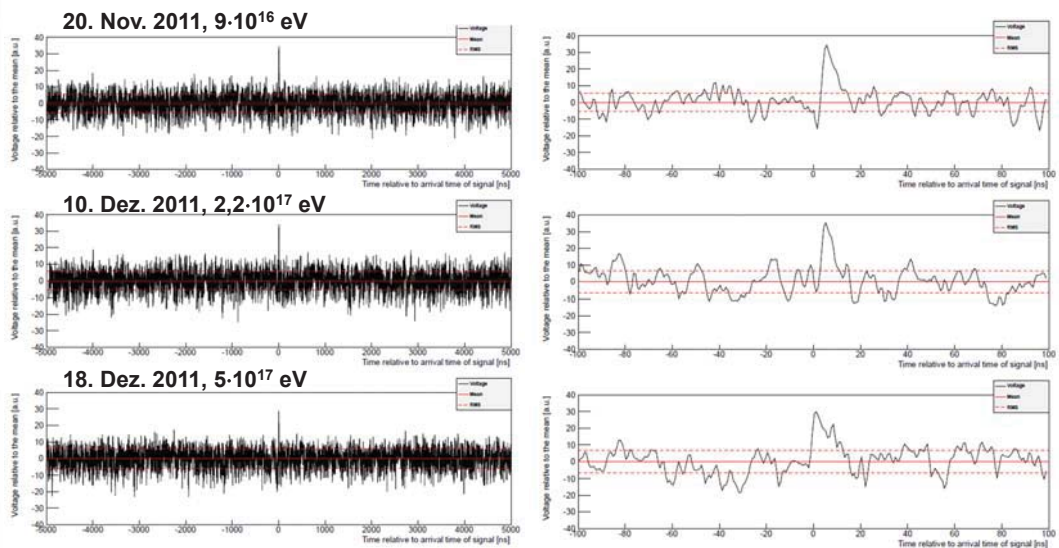


CROME-Projektverlauf

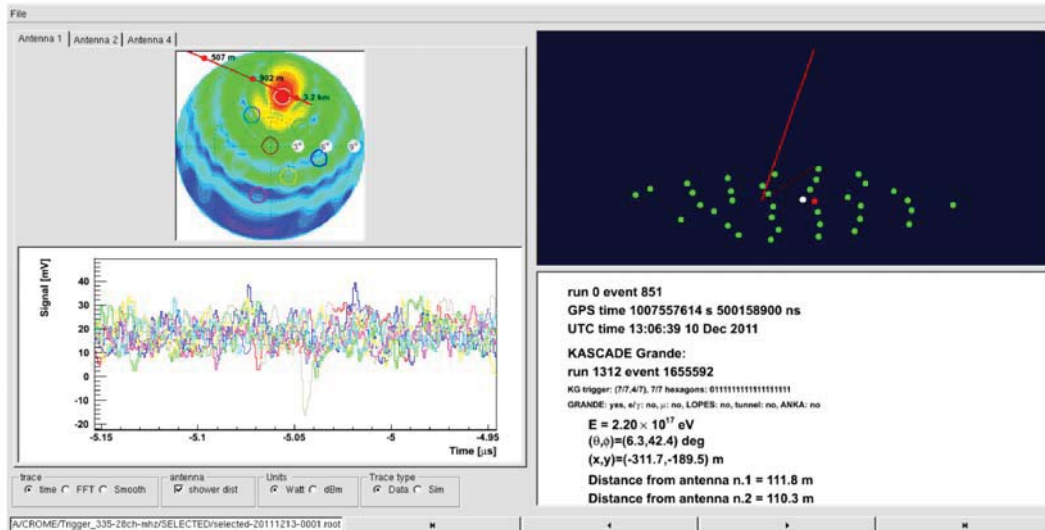


- März 2010: Projektstart
- Juli 2010: erster Empfänger (Prototyp, 4 Kanäle, C-Band + Ku-Band) in Betrieb
- Ende 2010: erster 3,4-Meter-Spiegel mit 9 Kanälen im C-Band
- Anfang 2011: zweiter 3,4-Meter-Spiegel mit 9 Kanälen im C-Band
- Herbst 2011: 4 logarithmisch-periodische Dipolantennen für 40...80 MHz (8 Kanäle)
- März 2012:
 - 1,29...1,45-GHz-Empfänger (L-Band) im Testbetrieb (Prototyp, 1 Kanal)
 - dritter 3,4-Meter-Spiegel (9 Kanäle) im Aufbau
 - VLF-System (20 kHz..20 MHz) im Aufbau
- Anfang 2012:
 - Nachweis von 12 Mikrowellen-Events im C-Band in Koinzidenz mit kosmischen Luftschauern
 - davon mindestens 2 Radio-Hybrid-Events (C-Band + 40...80 MHz-Band)

Signale am Beispiel dreier Mikrowellen-Events



CROME-Event-Viewer / erste Mikrowellenevents



Gefundene Mikrowellen-Events



Table 1: Information about the KASCADE-Grande events with a signal measured by CROME in the C band (3.4–4.2 GHz). $S_{C\text{band}}$ is measured signal in dB above average noise level, D_{core} is the shower core position from the antenna measuring microwave signal from air shower, θ_{tilting} is the tilting angle of the antenna during measurement, $\theta_{EW\text{pol.}}$ is the receiver polarization angle towards EW, θ_{shower} is the angle between shower axis and boresight angle of the receiver measuring the event, $\theta_{\text{shower,4km}}$ is the angle of boresight axis to the position of the shower core at 4 km height. Angles are in geographical coordinates and for receivers with the highest signal.

$S_{C\text{band}}/\text{dB}$	Date	UTC time	Azimuth	Zenith	Energy/eV	D_{core}/m	θ_{tilting}	$\theta_{EW\text{pol.}}$	θ_{shower}	$\theta_{\text{shower,4km}}$
13.8	20.11.2011	22:07:05	97.8°	3.8°	9.0×10^{16}	125.7	0°	80.5°	1.6°	1.6°
13.6	10.12.2011	13:06:39	62.6°	6.3°	2.2×10^{17}	111.8	0°	9.5°	3.5°	1.9°
11.5	18.12.2011	10:57:42	248.6°	4.9°	5.0×10^{17}	98.3	0°	80.5°	1.2°	1.8°
10.7	30.12.2011	08:57:34	355.5°	19.7°	1.6×10^{17}	133.7	15°	58.0°	1.9°	2.1°
9.9	23.06.2011	08:41:23	86.9°	5.9°	3.3×10^{16}	148.7	0°	9.5°	3.1°	1.1°
9.2	21.05.2011	11:18:35	334.5°	7.0°	2.2×10^{17}	89.0	0°	32.0°	2.6°	3.7°
9.7	13.11.2011	07:42:26	263.8°	4.0°	3.2×10^{16}	112.1	0°	80.5°	0.7°	1.3°
9.6	20.11.2011	15:05:39	357.1°	19.2°	4.8×10^{16}	93.6	15°	32.0°	2.0°	1.2°
8.8	30.05.2011	20:39:48	246.1°	5.2°	9.5×10^{16}	133.7	0°	58.0°	1.8°	3.5°
8.3	09.07.2011	03:16:43	59.6°	3.6°	9.2×10^{16}	80.4	0°	58.0°	0.6°	0.7°

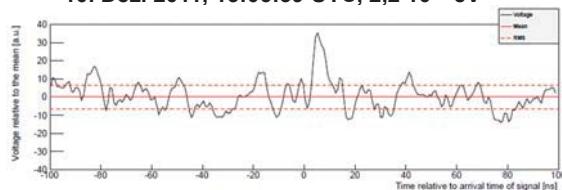
+ mindestens 2 weitere

Hybrid-Radio-Events (3.4...4.2 GHz und 40...80 MHz)

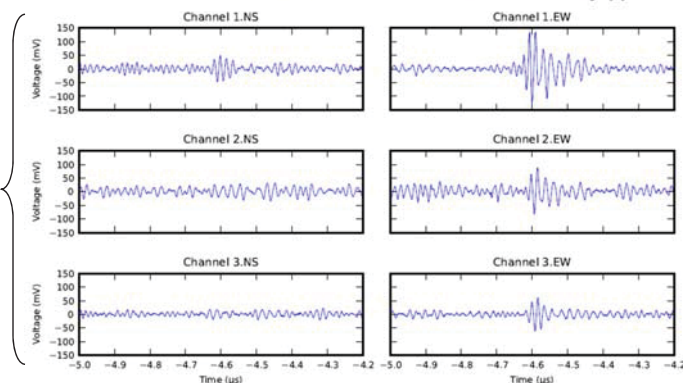


3,4...4,2 GHz

10. Dez. 2011, 13:06:39 UTC, $2,2 \cdot 10^{17}$ eV



40...80 MHz



Zusammenfassung



- CROME seit Herbst 2010 im stabilen Messbetrieb
- externe Triggerung durch KASCADE-Grande-Teilchendetektoren
- 18 (+9) Kanäle 3.4...4.2 GHz (C-Band)
- 8 Kanäle im 40...80-MHz-Band
- kalibriert mittels Radiocopter
- ca. 12 Mikrowellen-Events nachgewiesen
- davon mindestens 2 Radio-Hybrid-Events
- Spitzenstellung hinsichtlich der nachgewiesenen Event-Anzahl
- Diskussion der Radioemissionsmechanismen wurde neu belebt

Vielen Dank für ihre Aufmerksamkeit

KASCADE Teilchendetektorexperiment



200 x 200 m², 256 Detektoren, Abstand 13 m, 10¹⁴ eV...10¹⁷ eV

23 | Oliver Krömer | Institut für Prozessdatenverarbeitung und Elektronik | 07.04.2008

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



Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

KASCADE-Grande Experiment



700 x 700 m², 37 Detektoren, Abstand 130 m, bis 10¹⁸ eV

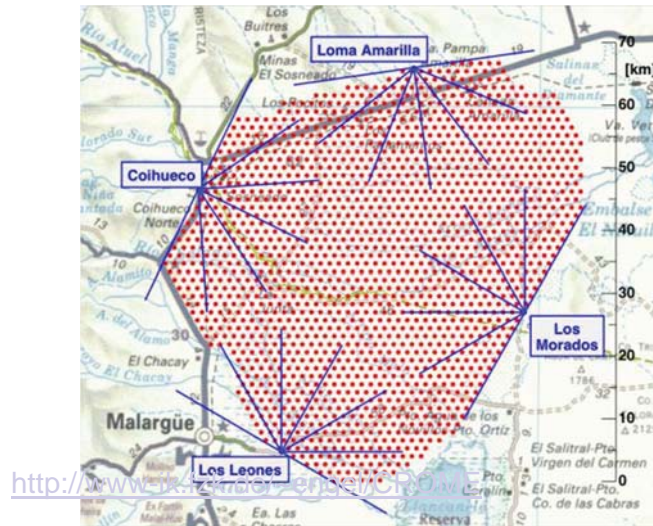
24 | Oliver Krömer | Institut für Prozessdatenverarbeitung und Elektronik | 07.04.2008

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



Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

Pierre Auger Observatorium, Argentinien



50 x 50 km², 1600 Teilchendetektoren, Abstand 1500 m, zusätzlich 24 Fluoreszenzlicht-Teleskope, bis 10²¹ eV

Detektoren des Pierre Auger Observatoriums



1600 Teilchendetektoren



24 Fluoreszenzlicht-Teleskope

Modulator Test Facility

Überblick, Ergebnisse und Ausblick

SEI – Tagung am HZDR

Dresden, 14. März 2012

Mirko Grimberg

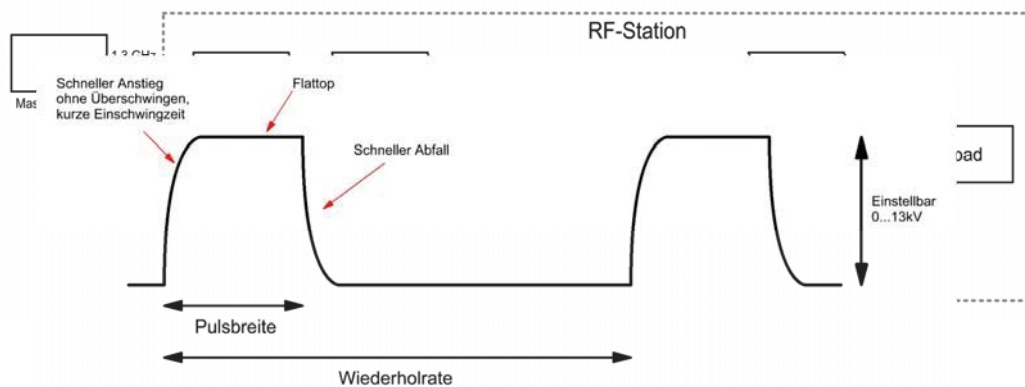
Gliederung

- Einleitung
- MTF
 - Komponenten, Funktionsweise
 - math. Beschreibung, Simulation
 - Messungen, Untersuchungen und Modifikationen
- Ergebnisse, Zahlen & Daten

Einleitung

- Für den XFEL werden 27 RF-Stationen benötigt.
=> Jede RF-Station braucht einen Modulator.
- Test der Modulatoren im Verbund einer kompletten RF-Station
- Sommer 2005: Entscheidung einen Modulator Teststand zu bauen
- Erstellung der Ausschreibungsunterlagen, Spezifikation
- Internationale Ausschreibung
=> Auswahl von zwei Herstellern (Thomson, Imtech-Vonk)
=> Jeder Hersteller fertigt einen Prototypen
- Prototyp muss seine Eignung für den XFEL unter Beweis stellen
=> Messungen, Untersuchungen bei DESY, Standort Zeuthen
=> Modifikation und Anpassung der Modulatoren an die Anforderungen des XFEL
=> Feststellung der Eignung für den XFEL
- Zweite, begrenzte Ausschreibung

Komponenten und Funktionsweise

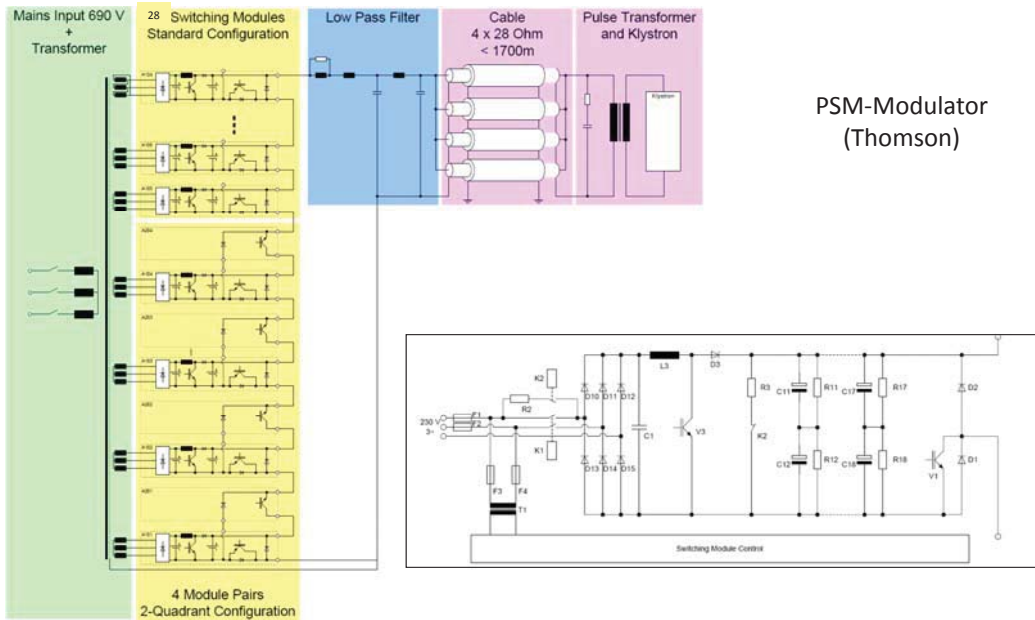


Wiederholrate: 1...30Hz
Pulsbreite: bis zu 1,7 ms

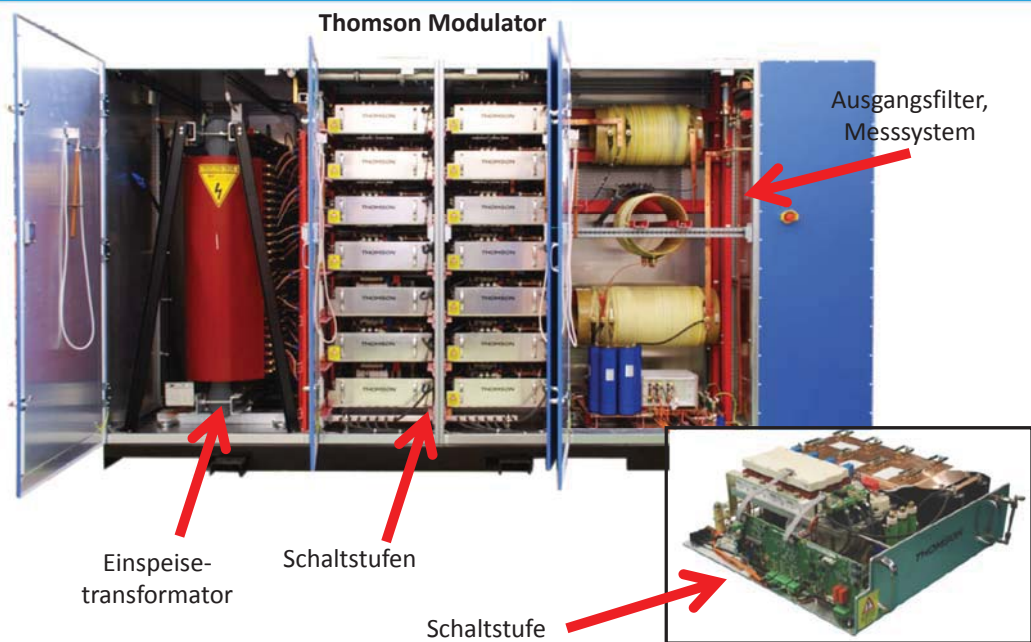
RF-Station MTF
Spannung: 0...13kV
Strom: 0...2kA

Pulspower: 16 MW
Puls Flatness: $\pm 0,3\%$

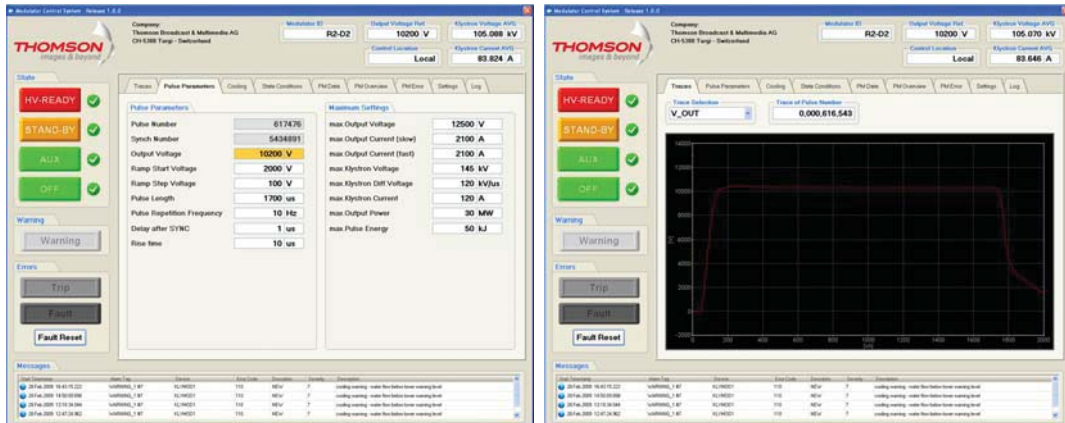
Komponenten und Funktionsweise



Komponenten und Funktionsweise



Komponenten und Funktionsweise



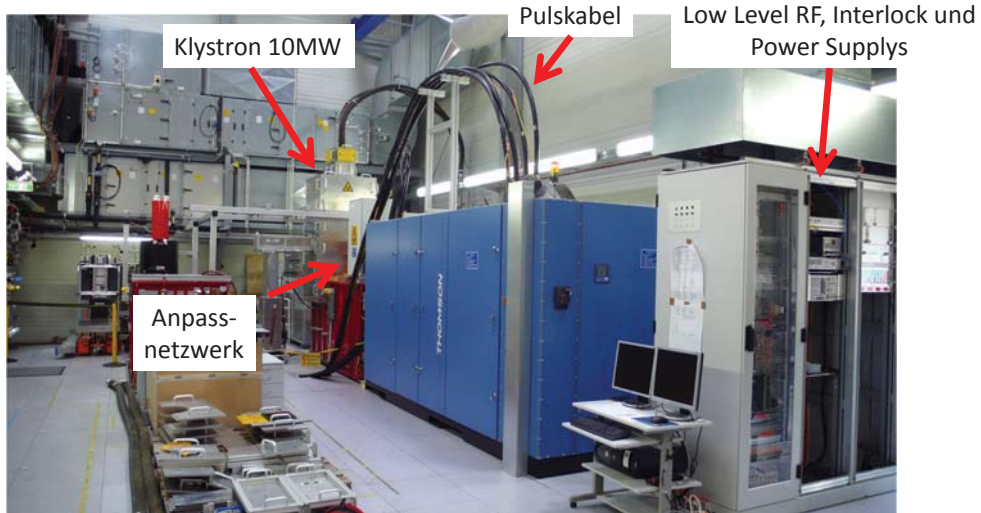
GUI Thomson Modulator

Komponenten und Funktionsweise



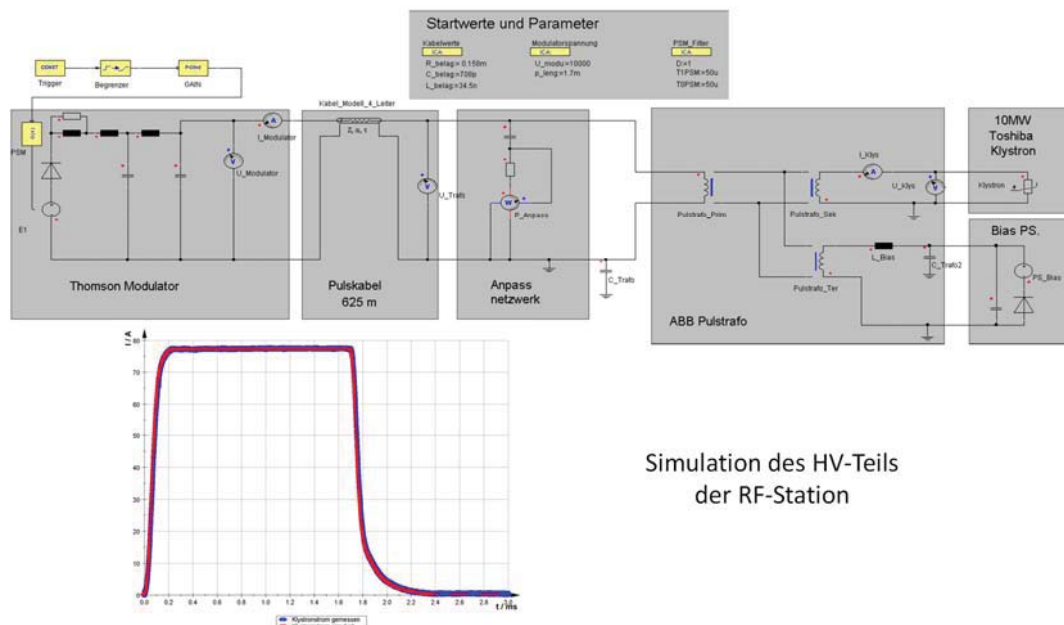
RF-Station mit Thomson Modulator und 5MW Klystron

Komponenten und Funktionsweise



RF-Station mit Thomson Modulator und 10MW Klystron

Simulation



Simulation des HV-Teils der RF-Station

Ergebnisse

Was wurde getestet? (eine kleine Auswahl) Modifikationen?

- Einhaltung der Spezifikation
z.B. Pulsflatness, Puls zu Puls Genauigkeit, MTBF & MTTR
Wirkungsgrad, Störfestigkeit, uvm.
- Langzeittest
- Wärmeentwicklung & Kühlung
- Elektromagnetische Verträglichkeit
- Qualität der RF
- 30 Hz Test
- Kommunikation
- Tine, Doocs

Ergebnisse

Thomson

Thomson hat nach Modifikationen
alle Test's bestanden

- Hardware OK
- Software OK
- Spezifikation wurden eingehalten
- Modulator ist für XFEL geeignet
- Modulares System, MTBF, MTTR, redundant
- Qualitativ hochwertiges Produkt

⇒ Thomson hat die Ausschreibung gewonnen.
Es wurden 22+5 Modulatoren bestellt.

Vielen Dank für Ihre Aufmerksamkeit

Gibt es noch Fragen?



DESY-PROC-2012-01

ISBN 978-3-935702-65-2

ISSN 1435-8077