

Forschungsprojekt

ESANA: Echtzeit-Ethernet für die Sensor/Aktorvernetzung

Zuwendungsempfänger: Hochschule Ostwestfalen-Lippe

Förderkennzeichen: 1742X06

Laufzeit: 01.10.2006 - 30.06.2008

Schlussbericht

(Sachbericht des Verwendungsnachweises nach Nr. 6 ANBest-P)

Prof. Dr.-Ing. Jürgen Jasperneite

Dipl.-Ing. Markus Schumacher

Lemgo, den 12. Februar 2009

Prof. Dr.-Ing. Jürgen Jasperneite
Institut Industrial IT der
Hochschule Ostwestfalen-Lippe
Liebigstrasse 87
D-32657 Lemgo
juergen.jasperneite@hs-owl.de
www.init-owl.de

Inhaltsverzeichnis

1	Aufgabenstellung	3
2	Organisatorische Rahmenbedingungen	4
2.1	Ausführende Stelle	4
2.2	Zusammenarbeit mit anderen Stellen	4
2.3	Planung und Ablauf des Vorhabens	5
3	Stand der Technik	7
3.1	Ausgangssituation	7
3.2	Zeitliche Anforderungen der Sensor/Aktorebene	7
3.3	Echtzeit-Ethernet	8
4	Ergebnisse	11
4.1	Performance	11
4.1.1	Potentialanalyse	11
4.1.2	Optimierungsansätze	21
4.1.3	Leistungsbewertung	32
4.2	Einfachheit	36
4.3	Entwicklungsumgebung	39
4.4	Demonstrator	40
4.5	Verwertbarkeit der Ergebnisse	42
5	Technischer Fortschritt bei anderen Stellen	43
6	Veröffentlichungen	44
6.1	Messeauftritte	45
7	Abschließende Bemerkungen	48

Zusammenfassung

Das Projekt ESANA beschäftigt sich mit dem Einsatz von Echtzeit-Ethernet (RTE) in der Sensor/Aktorvernetzung. Derzeit sind mehr als 25 Echtzeit-Ethernetverfahren in der Diskussion oder Anwendung. Viele dieser Systeme sind sehr spezialisiert, der Anwender wünscht sich jedoch ein möglichst universell einsetzbares Kommunikationssystem. Ein besonders anspruchsvoller Anwendungsbereich für Ethernet stellt die Sensor/Aktorvernetzung dar. Hier stellen die Einfachheit der Implementierung für einen IO-Knoten und die erreichbaren Update-Zeiten wichtige Kriterien dar. Darüber hinaus ist die Feldebene der Automatisierungstechnik durch die Verwendung von Linienstrukturen, sowie einer geringen Payloadgröße pro Knoten gekennzeichnet. Die verschiedenen RTEs benutzen für die Rahmenbildung in der OSI-Schicht 2 entweder einen Individualrahmen oder einen Summenrahmen. Schon von der ersten Generation der industriellen Kommunikationstechnik ist bekannt, dass Summenrahmen für den oben beschriebenen Anwendungsfall vorteilhaft sind. In komplexeren Strukturen hingegen sind Individualrahmen flexibler einsetzbar und skalieren auch bei höheren Bitraten besser. Die erreichbaren Update-Zeiten eines Kommunikationssystems werden durch die Zeit für das Senden der entsprechenden Rahmen (Frametransferzeit) und der benötigten Zeit für die physikalische Ausbreitung vom Sender zum Empfänger begrenzt. Wesentliche Ergebnisse des Projektes sind die Optimierung der Frametransferzeit mit Hilfe eines Dynamic-Frame-Packing (DFP), sowie die Einführung einer topologischen Adressierung, mit dessen Hilfe die Weiterleitungszeit deutlich reduziert werden konnte. Für die Validierung der Verfahren wurde Profinet als Vertreter der Individualrahmen-basierten Verfahren herangezogen. Es konnte sowohl analytisch, als auch experimentell mit Hilfe eines Demonstrators nachgewiesen werden, dass Profinet durch Anwendung der beiden Verfahren zum derzeit leistungsfähigsten RTE in diesem Anwendungsfall avancieren konnte. Die Projektergebnisse werden derzeit in die Standardisierung durch die PNO überführt und stehen damit künftig einem breiten Kreis von Geräteherstellern zur Verfügung. Weiterhin ist ein Leitfaden entstanden, der aufzeigt, wie man einen Profinet-IO Knoten auf Basis eines verfügbaren Starterkits einfach realisieren kann.

Summary

The ESANA project addresses the usage of Real-time Ethernet (RTE) at the sensor/actuator level of industrial automation systems.

Currently more than 25 RTEs are under discussion or in the phase of deployment. While most of them are designed for specific use cases, customers require a general purpose network that is usable as much as possible. A challenging field for Ethernet is the sensor/actuator level (field level) where the simplicity of a device implementation and a low latency for the IO data exchange are important criteria. Furthermore the field level of automation systems is characterized by using line topologies and small payload sizes per node.

The different RTEs are using either an individual frame approach or an total frame approach at the data link layer. From the first generation of industrial communication systems it is well known, that for our use case the total frame approach can be advantageous. For more complex network structures the individual frame approach offers more flexibility and scalability especially if it comes to higher bit rates. The update time of a communication network is limited by two factors: The time for transmitting the frame (frame transfer time) at the source node and the time needed for the propagation of the frame from the source node to the destination node along a line.

The most important outcome of the ESANA project is the optimization of the frame transfer time by introducing a dynamic frame packing (DFP) approach, as well as a topological addressing scheme, which reduces the propagation time significantly.

For validation purposes Profinet was selected as a representative of individual frame based RTEs. By applying both mechanisms we could show analytically as well as with a physical demonstrator that Profinet becomes the most efficient RTE for sensor/actuator networking.

Based on a commercial available starter kit a guideline is published, which shows how to develop a profinet IO device efficiently.

The project results are currently considered by the PNO for the next version of the Profinet specification. Therefore device manufacturers can benefit from the project results in the near future.

Abkürzungsverzeichnis

AT	Automatisierungstechnik
APDU	Application Protocol Data Unit
ASIC	Application Specific Integrated Circuit
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CRC	Cyclic Redundancy Check
DCS	Datagram Check Sequence
DFP	Dynamic Frame Packing
FCS	Frame Check Sequence
FuE	Forschung und Entwicklung
FPGA	Field Programmable Gate Array
GMII	Gigabit Medium Independent Interface
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IFG	Interframe Gap
IuK	Informations- und Kommunikationstechniken
IP	Internet Protocol
IP-Core	Intellectual Property Core
IOCS	Input/Output Object Consumer Status
IOPS	Input/Output Object Provider Status
KMU	Kleines und mittleres Unternehmen
LAN	Local Area Network
LLDP	Link Layer Discovery Protocol
LOI	Letter of Intent
MAC	Medium Access Control
MTU	Maximum Transfer Unit
NRT	non realtime
OSI	Open System Interconnection
Phy	Physical Transceiver
PNO	Profibus Nutzerorganisation
QoS	Quality-of-service
RTE	Real-Time Ethernet
SFD	Start of Frame Delimiter
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
VHDL	Very High Speed Integrated Circuit

VLAN	Hardware Description Language Virtuell Local Area Network
-------------	--

1 Aufgabenstellung

Aufgrund einer durchgängigen Verwendung von Ethernet in der Automatisierungstechnik (AT) werden für den Maschinen- und Anlagenbau künftig Kostensenkungen in der Planung, Inbetriebnahme und Wartung von vernetzten Automatisierungsanlagen erwartet. Für den Einsatz von Ethernet in der AT spricht die große Verbreitung von Ethernet im lokalen Netzwerkbereich (LAN), sowie dessen kontinuierliche Weiterentwicklung. Um Ethernet in der Feldebene der AT einsetzen zu können, sind Erweiterungen des Ethernet-Standards erforderlich. Derzeit sind einige Echtzeit-Ethernet-Konzepte (RTE) basierend auf Fast-Ethernet in der Diskussion und befinden sich in der Markteinführung. Aus Sicht einer zukünftigen Marktverbreitung und des zugrundeliegenden generalistischen Konzeptes ist Profinet ein vielversprechender Kandidat.

Das **übergeordnete Ziel** des Forschungsvorhabens besteht darin, die heute existierende technologische und wirtschaftliche Hemmschwelle für die Verbreitung von Profinet bis hinunter zur Sensor/Aktorvernetzung zu überwinden. Hierdurch ergibt sich ein hohes Innovationspotential aufgrund eines deutlich verbreiterten Anwendungsgebietes von Profinet.

Das **wissenschaftliche Ziel** besteht in der Entwicklung von Konzepten für die Echtzeitkommunikation auf Basis von IuK-Technologien, sowie deren mikroelektronischer Umsetzung. Echtzeitfähigkeit ist eine der elementaren Anforderungen in der Automatisierungstechnik, aber auch angrenzender Disziplinen. In zukünftig mehr und mehr verteilten Systemen wird die Netzwerktechnik zum limitierenden Faktor der Leistungsfähigkeit. Vor diesem Hintergrund ist es von großem wissenschaftlichen Interesse, neue Verfahren zu entwickeln, mit der Technologien aus dem IuK-Bereich für die Echtzeitkommunikation in der Automatisierungstechnik genutzt werden können, ohne sie gänzlich zu verändern.

Das Projekt adressiert hierzu zwei **Leitfragen**:

- Wie kann man für Profinet **kürzestmögliche Latenzzeiten** in typischen Sensor/Aktor-Szenarien erreichen, ohne sich dabei zu weit vom Ethernet-Standard zu entfernen?
- Wie kann man die **Komplexität** von Profinet für die Implementierung in einfache Sensor/Aktor-Knoten reduzieren?

2 Organisatorische Rahmenbedingungen

2.1 Ausführende Stelle

Das Projekt wurde im inIT - Institut Industrial IT der Hochschule Ostwestfalen-Lippe am Standort Lemgo durchgeführt. Das inIT (www.init-owl.de) ist eine Einrichtung der angewandten Forschung und fasst die Forschungsaktivitäten von 6 Professoren unterschiedlicher Fachgebiete zusammen. In dem Institut, dass von Prof. Dr. Jürgen Jasperneite geleitet wird, sind derzeit mehr als 40 Mitarbeiter in den Bereichen industrielle Kommunikation, industrielle Bildverarbeitung und verteilte Echtzeit-Software in öffentlich geförderten Verbundvorhaben oder in Projekten der bi-/multilateralen Auftragsforschung tätig. Beteiligte Personen an dem Projekt ESANA waren in der ausführenden Stelle:

- Prof. Dr.-Ing. Jürgen Jasperneite, Projektleiter
- Dipl.-Ing. Markus Schumacher, wiss. Mitarbeiter
- Dipl.-Ing. Sergej Gamper, wiss. Mitarbeiter
- 2 Studierende als studentische Hilfskräfte
- 6 Studierende im Rahmen von Abschlussarbeiten

2.2 Zusammenarbeit mit anderen Stellen

Als Kooperationspartner haben folgende Firmen das Projekt unterstützt und begleitet:

1. MAZeT GmbH, Jena
2. OWITA GmbH, Lemgo
3. Phoenix Contact GmbH & Co. KG, Blomberg
4. Technische Universität Berlin
5. Siemens AG, Erlangen (ab Oktober 2007)

Die Firma MAZeT mit Sitz in Jena ist ein KMU mit dem Schwerpunkt auf Entwicklung, Produktion und Vermarktung von Produkten und Dienstleistungen für Elektronikanwendungen in der Medizin- und Automatisierungstechnik. MAZeT unterstützte das Projekt durch Beratungsleistungen im Bereich der mikroelektronischen Entwurfsmethodik für IP-Cores.

Die OWITA GmbH mit Sitz in Lemgo ist ein Ende 2005 gegründetes kleines Unternehmen, dass sich auf FuE-Dienstleistungen im Bereich der Anwendung innovativer Technologien in der Automatisierungstechnik spezialisiert hat. Die OWITA GmbH konnte sich im Verlauf des Projektes Know-How im Bereich Echtzeit-Ethernet aufbauen und damit als Dienstleister positionieren.

Phoenix Contact mit Stammsitz in Blomberg ist ein Unternehmen der industriellen Automatisierungstechnik und unterstützte das Projekt neben der Übernahme der Kosten für die Beschäftigung von zwei studentischen Hilfskräften (SHKs), der Bereitstellung und Überlassung von Evaluationsboards für die Durchführung von Systemtests, auch inhaltlich sehr intensiv.

Wissenschaftlich wurde das Projekt durch die Technische Universität Berlin, Lehrstuhl für Mikroelektronik (Prof. Manck), begleitet.

Im Oktober 2007 wurde die Firma Siemens AG, Erlangen, vertreten durch Herrn Dr. Karl Weber mit in das Projekt aufgenommen. Dr. Weber hat dem Projekt wesentliche technologische Impulse gegeben.

2.3 Planung und Ablauf des Vorhabens

Zur Beantwortung der in Kapitel 1 formulierten Leitfragen wurde das Projekt in einem Kick-off Meeting mit den Projektpartnern in die beiden funktionalen Arbeitsbereiche **Performance** und **Einfachheit** strukturiert. Das Projekt besteht strukturell gemäß Vorhabensbeschreibung (VHB) aus den Arbeitspaketen Anforderungsanalyse (AP1), Systemarchitektur (AP2), Implementierung (AP3) und Systemtests (AP4). Im Bereich der **Performance** wurde zunächst im Arbeitspaket 1 eine systematische Analyse der zukünftigen Potentiale heutiger Echtzeit-Ethernet-Systeme, insbesondere unter Berücksichtigung der Gigabit-Technik, durchgeführt. Auf Basis dieser Potentialanalyse wurden Optimierungsansätze erarbeitet (Arbeitspaket 2), implementiert (Arbeitspaket 3) und getestet (Arbeitspaket 4), die sich in den

bestehenden Profinet-Standard einbringen lassen.

Für den Arbeitsbereich **Einfachheit** wurde im Arbeitspaket 1 die Komplexität heutiger Knoten-Implementierungen am Beispiel des ASICs netX der Firma Hilscher und eines vorhandenen ERTEC-Evaluationboards der Firma Siemens untersucht. Im Ergebnis ist ein veröffentlichter Leitfaden entstanden, der Gerätehersteller Schritt für Schritt zu einer ersten Implementierung führt. Die Komplexität von Profinet selbst konnte dadurch natürlich nicht verringert werden. Hierzu sind ASICs notwendig, die die inhärente Komplexität für den Gerätehersteller kapseln. Dazu wurde ein entsprechendes Projekt der industriellen Auftragsforschung parallel zum ESANA-Projekt gestartet.

3 Stand der Technik

3.1 Ausgangssituation

Auf der einen Seite ist die Feldebene der industriellen Automatisierungstechnik heute geprägt von einer Vielfalt unterschiedlicher Netzwerke. Zu Realisierung einer AT-Anwendung kommen heute neben den klassischen Feldbussen wie Profibus [1], Interbus [2], Devicenet [3], zusätzlich spezielle Antriebsbusse (z.B. Sercos [4]), sowie Bussysteme für die Übertragung von sicherheitskritischen Signalen (z.B. Safetybus-P [5]), z.T. gleichzeitig zum Einsatz. Das führt zu einer komplexen Situation für die Systemintegratoren und die Anlagenbetreiber im Maschinen- und Anlagenbau. Für all diese Bussysteme ist spezifisches Know-how für die Planung, Inbetriebnahme und Wartung erforderlich. Darüber hinaus ist eine bussystem-übergreifende Kommunikation nicht vorgesehen, sondern muss immer anwendungsspezifisch gelöst werden. Die Folgen sind unflexible Kommunikationsstrukturen und hohe Kosten für die Systemintegration.

Auf der anderen Seite ist die Übertragung von Sensor-, Aktor- und Parametrierungsdaten in Zukunft nicht mehr ausreichend, um die stetig steigenden Anforderungen an moderne Produktionsanlagen zu erfüllen. Die erweiterte Funktionalität von Feldgeräten und die zunehmende Integration der Prozess- und Unternehmensleitebene, stellen weit höhere Anforderungen an die eingesetzten Kommunikationssysteme, vor allen an deren Durchgängigkeit an den Systemgrenzen.

3.2 Zeitliche Anforderungen der Sensor/Aktorebene

Auf der Sensor/Aktorebene heutiger Automatisierungssysteme, auch Feldebene genannt, ist die Linientopologie sehr bedeutsam [6]. Im Unterschied zur Bürokommunikation muss die Verkabelung in der industriellen Automation den mechanischen Bedingungen der Maschinen und Anlagen folgen. Das führt häufig zu einer linienförmigen Topologie mit einigen Abzweigen. Bei Verwendung von Ethernet stellt sich in diesem Fall eine Kaskade von hintereinandergeschalteten aktiven Komponenten (Switches oder Feldgeräte) ein. Die Anforderungen der Feldebene in den verschiedenen Anwendungsbereichen lassen sich der nachstehenden Tabelle entnehmen.

Anwendungsumfeld	zeitliche Anforderung der Feldebene [ms]
Gebäudetechnik	50 - 100
Prozesstechnik	< 10
Fertigungstechnik	< 10

Tabelle 1: Zeitliche Anforderungen der Feldebene nach [7]

Wirft man einen genaueren Blick in das Anwendungsgebiet Fertigungstechnik mit seinen hohen zeitlichen Anforderungen, so können die Echtzeitforderungen nach den dort vorherrschenden Anwendungen konkretisiert werden:

QoS Klasse	Anwendung	QoS-Anforderung Latenzzeit [ms]	Jitter [ms]
1	Steuerung-zu Steuerung, Visualisierung	10 – 100	./.
2	Steuerung zu dezentrale Peripherie	1 – 10	≈ 1
3	Synchronisierte Bewegungsabläufe	≤ 1 ms	$\leq 10^{-3}$

Tabelle 2: Klassifizierung der Echtzeitanforderungen in der Fertigungstechnik nach [8]

3.3 Echtzeit-Ethernet

Die zweite Generation der industriellen Echtzeit-Kommunikation basiert nach Einführung der Feldbustechnik in den achtziger Jahren auf Ethernet. Eine Einführung in die Prinzipien und die aktuelle Standardisierung von RTEs ist in [9, 10, 11] zu finden. Derzeit sind 25 unterschiedliche Verfahren für RTEs bekannt, die meisten sind für einen spezifischen Anwendungsfall optimiert [12].

Die derzeit diskutierten Herstellerkonzepte für RTEs lassen sich in drei Klassen einteilen (siehe Abb. 1). Die Klassen unterscheiden sich im Hinblick auf die erreichbare Performance und die notwendigen Erweiterungen zum Standard-Ethernet [13].

Die Kategorie 1 übernimmt Ethernet und die TCP/IP-Protokoll-Familie ohne jede Veränderung. Lediglich auf der Applikationsebene wird ein auto-

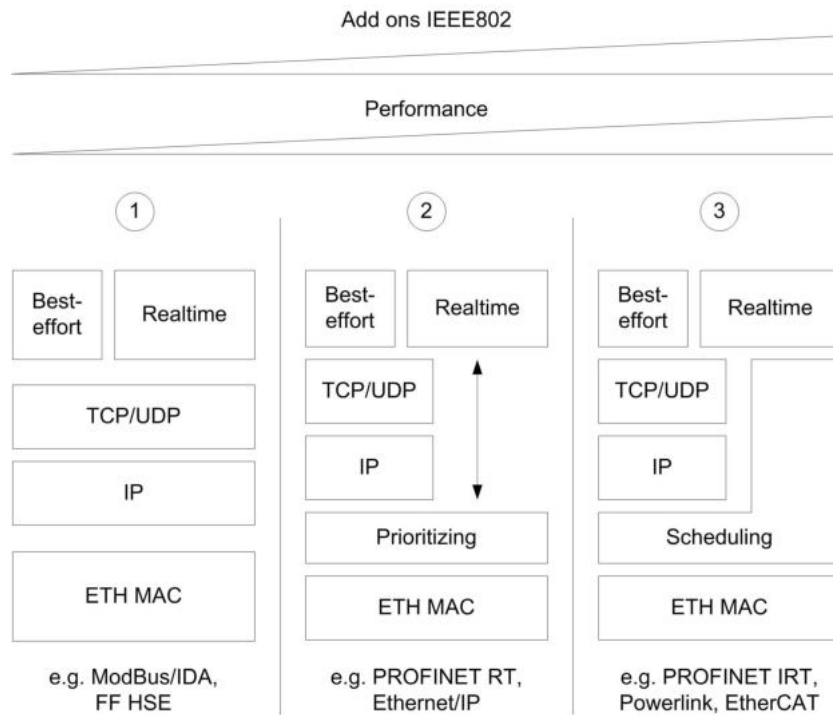


Abbildung 1: Klassifizierung von Echtzeit-Ethernetprotokollen [8]

omatisierungsspezifisches Protokoll hinzugefügt. Zu dieser Klasse zählen die Ansätze ModBus/IDA [14] und High Speed Ethernet (HSE) der Fieldbus Foundation [15]. Mit diesem Ansatz können Echtzeitanwendungen realisiert werden, die in die QoS-Klasse 1 gemäss Tabelle 2 fallen.

In der Kategorie 2 wird von der Priorisierung des Switched Ethernet nach IEEE 802.1Q/D Gebrauch gemacht [16, 17]. Zusätzlich werden die signifikanten Laufzeitanteile der Endgeräte an der gesamten Ende-zu-Ende-Laufzeit durch Umgehung des TCP/IP-Stacks für die Echtzeitdaten um bis zu einer Größenordnung optimiert. In [18] wurde gezeigt, dass mit diesem Ansatz in einer Linientopologie mit 50 Teilnehmern worst-case-Latenzzeiten von 10 ms für die hochprioren Daten garantiert werden können, wenn obere Schranken für die Verkehrslast in dem System angegeben werden können. Für eine optimale Performance sind in diesem Schritt anspruchsvolle Schicht-2 Software-Treiberkonzepte erforderlich, die die Eigenschaften des Ethernet-Controllers berücksichtigen. In diese Klasse gehören Profinet RT [19, 20] und Ethernet/IP

[21].

Die Kategorie 3 greift direkt in das Scheduling von Switched Ethernet ein und stellt damit die größte Veränderung des Standard-Ethernet dar. Hierbei werden Zeitschlitzverfahren (engl.: Time division multiple access, TDMA) benutzt, die wie bei dem hub-basierten Ethernet Powerlink [22] zentral durch ein Master/Slave-Verfahren oder wie bei Profinet mit IRT verteilt durch die Verwendung spezieller Switches realisiert werden. Beim Ethercat-System [23], das ebenfalls dieser Klasse zuzuordnen ist, wird das unter anderem von dem Feldbus Interbus [2] her bekannte Summenrahmenverfahren auf Basis eines Master-Slave-Prinzips, sowie spezieller Slave-Protokollchips genutzt. Nur mit den Ansätzen dieser Klasse wird das Problem gelöst, da ein wichtiges Echtzeittelegramm auf seiner Reise durch das Netzwerk durch niederpriorige Telegramme verzögert werden kann. Daher sind nur hier Latenzzeiten im Sub-Millisekundenbereich sowie ein Jitter im Sub-Mikrosekundenbereich erreichbar.

Im Bereich der Kategorie 3 lassen sich zwei Grundprinzipien ausmachen: Zum einen das Summenrahmenverfahren, in dem mit einem Frame mehrere Teilnehmer gleichzeitig mit Daten versorgt werden, zum anderen der Ansatz der Datenzustellung mit individuellen Rahmen für jeden Teilnehmer. Prominenter Vertreter des Summenrahmenverfahrens ist Ethercat und der individuellen Frames Profinet. Daher konzentrieren sich die Leistungsbetrachtungen im Folgenden auf diese beiden Systeme.

4 Ergebnisse

4.1 Performance

4.1.1 Potentialanalyse

Damit die Vor- und Nachteile von Profinet bewertet werden konnten, wurde das Individualframe-basierte Profinet mit dem Summenrahmen-basierten Ethercat verglichen. Beide RTEs basieren auf dem Ethernet Standard, der prinzipielle Aufbau eines Ethernet-Frames ist in der Abbildung 2 dargestellt. Der Frame beginnt mit einer Präambel. Sie dient den Netzwerkteilnehmern zur Synchronisation und besteht aus einer sieben Byte langen alternierenden Bitfolge (101010...1010) und endet mit dem Start-of-Frame Delimiter (SFD)(Bitfolge. 1010.1011). Darauf folgend werden jeweils sechs Byte für die Ziel- und die Quelladresse übertragen. Im Anschluss daran werden zwei Bytes zur Bestimmung des Ethertypes reserviert. Die Payload des Ethernet-Frames umfasst mindestens 46 Byte und höchstens 1500 Byte. Der Frame endet mit der 4 Byte langen Frame Check Sequence (FCS) für die Fehlererkennung. Sie benutzt das CRC-32 Polynom und garantiert damit die Integrität des übertragenen Frames. Der minimale Abstand zwischen zwei aufeinanderfolgende Frames wird durch die Inter-Frame-Gap (IFG) mit einer Länge von 12 Byte festgelegt.

Der Frameaufbau eines Profinet-Frames entspricht prinzipiell dem eines Ethernet-Frames (siehe Abbildung 3). Der Profinet-Teil beginnt nach dem Ethertype mit der so genannten Frame-ID, die die ID des Zielknotens enthält. Im Anschluss daran werden die Nutzdaten des Knotens übertragen. Darauf folgend wird der Input/Output Provider Status (IOPS) und der Input/Output Consumer Status (IOCS) übermittelt. Dieser Status gibt die Gültigkeit der übertragenen Nutzdaten an. Für die Submodule eines Kno-

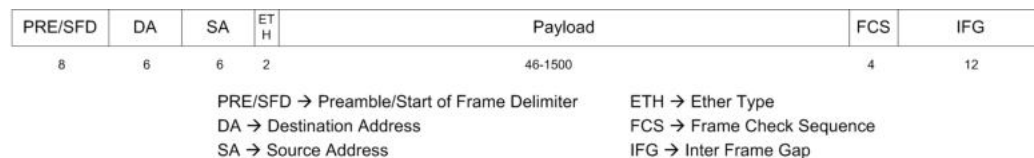


Abbildung 2: Struktur eines Standard Ethernet Frames

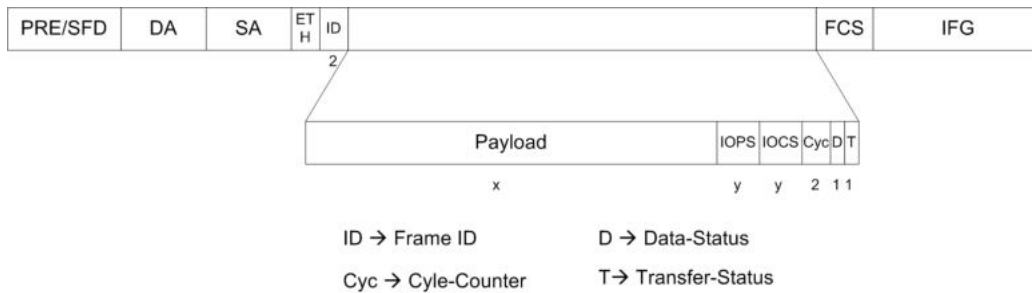


Abbildung 3: Aufbau eines Profinet-Frames

tens wird jeweils ein Byte IOPS und IOCS übertragen. Die Profinet-Struktur endet mit dem APDU-Status, der aus einem Zykluszähler (engl.: Cycle counter) besteht, der in $31,25 \mu s$ Schritten inkrementiert wird und den relativen Sendezeitpunkt angibt. Die Aufgabe des Zählers besteht in der Erkennung alter Frames bei redundanten Strukturen. Nach dem Zähler wird der Status der Daten aus Sicht des Senders übertragen. Ist der Wert in diesem Byte auf 1 gesetzt, dann ist der Zustand des Senders in Ordnung. Der APDU-Status endet mit dem ein Byte langen Transfer-Status. Ist dieses Statusbyte 0, so wurden die Daten fehlerfrei übertragen.

Updatezeit

Definition 1 (Update-Zeit) Die Update-Zeit eines Kommunikationssystems ist diejenige Zeitspanne, die vergeht, um die Nutzdaten von einem Controller zu allen Knoten und von allen Knoten zum Controller zu übertragen.

In der Abbildung 4 wird der Nachrichtenaustausch zwischen einem Profinet Controller und den Knoten in einer Linientopologie in einem Ort/Zeit-Diagramm dargestellt. In dem Beispiel nutzt das System den sogenannten **Windschatteneffekt**. Dieser setzt voraus, daß die Frameübertragung zu dem Knoten mit der grössten Verzögerungszeit als erstes erfolgt. Im Fall einer Linientopologie ist das immer der letzte Knoten einer Linie, da dieser Frame alle Knoten passieren muss, bevor er sein Ziel erreicht. Wenn die Vorteile des Windschatteneffekts nicht genutzt werden und der Frametransfer mit dem Knoten 1 beginnt und mit Knoten 3 endet, erhöht sich die minimale Updatezeit und die vorhandene Bandbreite wird nicht optimal genutzt. Aufgrund des Windschatteneffekts wird die minimale Updatezeit primär von der zu versendenden Datenmenge bestimmt. Die Frametransferzeit t_{frame} ergibt

sich zum einen aus den Nutzdaten (Payload) d_{data} , die zu den Knoten versendet wird, und zum anderen aus dem Ethernet- und Profinet-spezifischen Overhead ($d_{ethernet_ov} + d_{pn_ov}$), geteilt durch die verwendete Bitrate b :

$$t_{frame} = \frac{d_{ethernet_ov} + d_{pn_ov} + d_{data}}{b} \quad (1)$$

mit dem ethernetspezifischen Overhead:

$$d_{ethernet_ov} = d_{PRE} + d_{DA} + d_{SA} + d_{EtherType} + d_{FCS} + d_{IFG} \quad (2)$$

mit der Präambel/SFD $d_{PRE} = 8$ Byte, der Zieladresse $d_{DA} = 6$ Byte, der Quelladresse $d_{SA} = 6$ Byte, dem Ethertype $d_{EtherType} = 2$ Byte, der Checksequenz $d_{FCS} = 4$ Byte und der Interframe Gap $d_{IFG} = 12$ Byte.

Der Profinet-spezifische Overhead d_{pn_ov} besteht aus der FrameID (2 Byte) und dem APDU-Status (≥ 1 Byte IOPS; ≥ 1 Byte IOCS; 2 Byte Cycle Counter; 1 Byte Data-Status; 1 Byte Transfer-Status). Die Weiterleitungsverzögerung t_{fwd_pn} zwischen zwei Knoten besteht aus der physikalischen Laufzeit des Mediums t_{medium} und der Knotendurchleiteverzögerung t_{fwd} :

$$t_{fwd_pn} = t_{medium} + t_{fwd} \quad (3)$$

In der Abbildung 5 wird ein Profinet-Updatezyklus mit einer Bitrate von 1 GBit/s dargestellt. Aufgrund der erhöhten Bandbreite verringert sich der Einfluss aufgrund der Frametransferzeit t_{frame} auf die Update-Zeit. In diesem Szenario verschiebt sich der Flaschenhals des Systems von t_{frame} zu der Weiterleitungsverzögerung t_{fwd_pn} . Der Winkel α kann zur Visualisierung dieser Situation herangezogen und wie folgt berechnet werden:

$$\tan \alpha = \frac{|t_{fwd_pn} - t_{frame}|}{n_{devices}} \quad (4)$$

Das Vorzeichen des Winkels legt fest, ob der Flaschenhals eines Szenarios von der Frametransferzeit, damit auch von der Bitrate, oder von der Weiterleitungszeit dominiert wird. Ist $\alpha \geq 0$ erfährt das Szenario nur einen geringen Leistungsvorteil durch eine Bitratenerhöhung, sondern wird durch die Weiterleitungszeit limitiert.

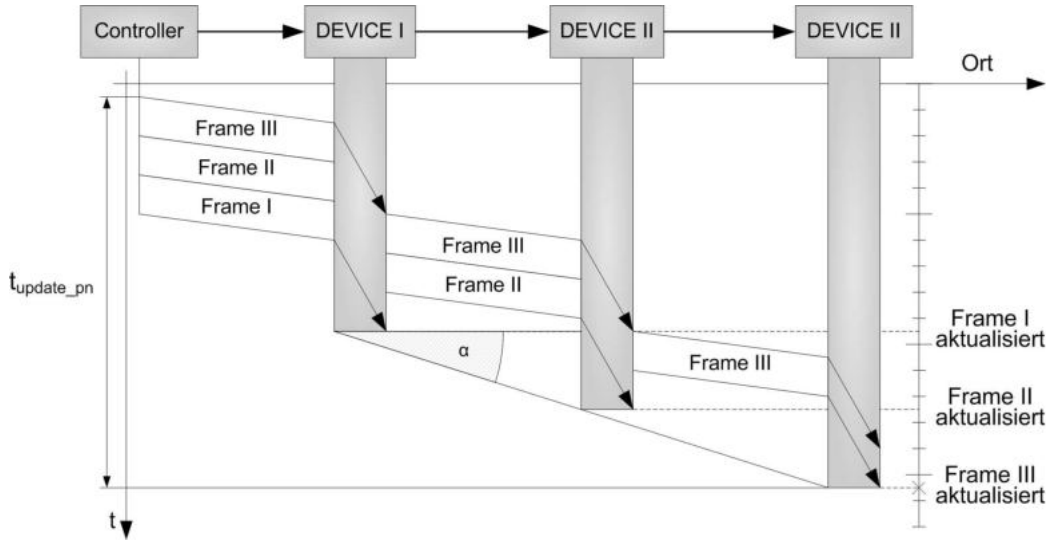


Abbildung 5: Kommunikation Controller → Knoten, Profinet mit 1Gbit/s

$$t_{update_pn} = \max(n_{devices} \cdot t_{frame} + t_{fwd_pn}, t_{frame} + n_{Devices} \cdot t_{fwd_pn}) \quad (7)$$

Als Vertreter für ein Summenrahmen-basiertes Protokoll wurde Ethercat ausgewählt. Wie in der Abbildung 6 dargestellt, basiert ein Ethercat-Frame auch auf einem Ethernet-Frame mit einem speziellen Ethertype und Struktur im Nutzdatenteil. Der Nutzdatenteil beginnt mit einem Ethercat-Header. Darauf folgen die relevanten Daten der vorhandenen Knoten in Form von Ethercat-Telegrammen. Ein Ethercat-Telegramm beginnt mit einem zehn Byte großen Telegramm Header. Im Anschluss daran werden die Nutzdaten für den jeweiligen Teilnehmer übertragen. Das Telegramm endet mit einem zwei Byte großen Working Counter, der die Anzahl funktionierender Teilnehmer wiedergibt und so eine Diagnosemöglichkeit für den Controller bietet. Ein Ethercat-Frame kann beliebig viele Ethercat-Telegramme enthalten. Übersteigt die gesamte Länge 1500 Byte, so müssen die Daten in mehrere Ethernet-Frames segmentiert werden. Die Teilnehmer entnehmen die Daten mit Hilfe eines projektierten Offsets aus dem Telegramm. Jeder Teilnehmer besitzt einen festen Offset, über den er einen Teilbereich des Telegramms, exklusiv für das Versenden von Daten verwenden kann.

Die Update-Zeit von Ethercat t_{update_ecat} wird in der Abbildung 7 dargestellt und setzt sich aus verschiedenen Termen zusammen. Zum einen erfährt ein Ethercat Frame eine Verzögerung aufgrund der Frametransferzeit t_{frame_ecat} und zum anderen aus der Weiterleitungsverzögerung t_{fwd_ecat} der Knoten. Die Frametransferzeit besteht aus dem Datenoverhead aufgrund des Ethernet-Standards $d_{ethernet_ov}$, dem Overhead aufgrund des Ethercat-Protokolls d_{ecat_ov} und der Zeit für die zu übertragene Nutzdaten d_{data} , geteilt durch die Bitrate b . Abhängig von der Applikation kann die Menge der Ausgangsdaten (Controller $\rightarrow$ Knoten) von der Menge der Eingangsdaten (Knoten $\rightarrow$ Controller) abweichen. Daher bestimmt $\max(Eingangsdaten, Ausgangsdaten)$ die zu reservierende Telegrammgröße im Summenrahmen. Die Abbildung 8 zeigt ein Beispiel für eine Asymmetrie zwischen Ein-/ und Ausgangsdaten.

Der Ethercat-spezifische Overhead d_{ecat_ov} besteht zum einen aus dem Ethercat-Header d_{ecat_hdr} (2 Byte) und dem durch die Ethercat Telegramme verursachten Overhead. Der Telegrammoverhead d_{teleg_ov} besteht aus der Anzahl der Ethercat Telegramme multipliziert mit dem Overhead eines Telegramms (12 Byte siehe Abbildung 6).

Die Frametransferzeit für Ethercat t_{frame_ecat} kann wie folgt bestimmt werden:

$$t_{frame_ecat} = \frac{d_{ethernet_ov} + d_{ecat_ov} + d_{data}}{b} \quad (8)$$

Die Weiterleitungsverzögerung t_{fwd_ecat} ist unabhängig von der Framegröße und der verwendeten Bitrate. Sie besteht zum einen aus der Leitungsverzögerung t_{medium} und zum anderen aus der Durchleitungsverzögerung eines Knotens

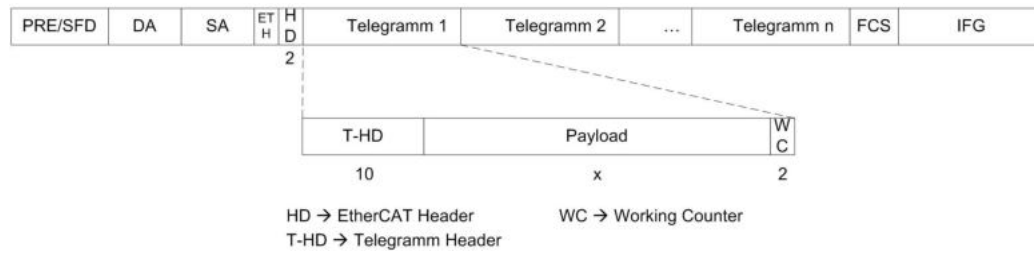


Abbildung 6: Struktur eines Ethercat-Frames

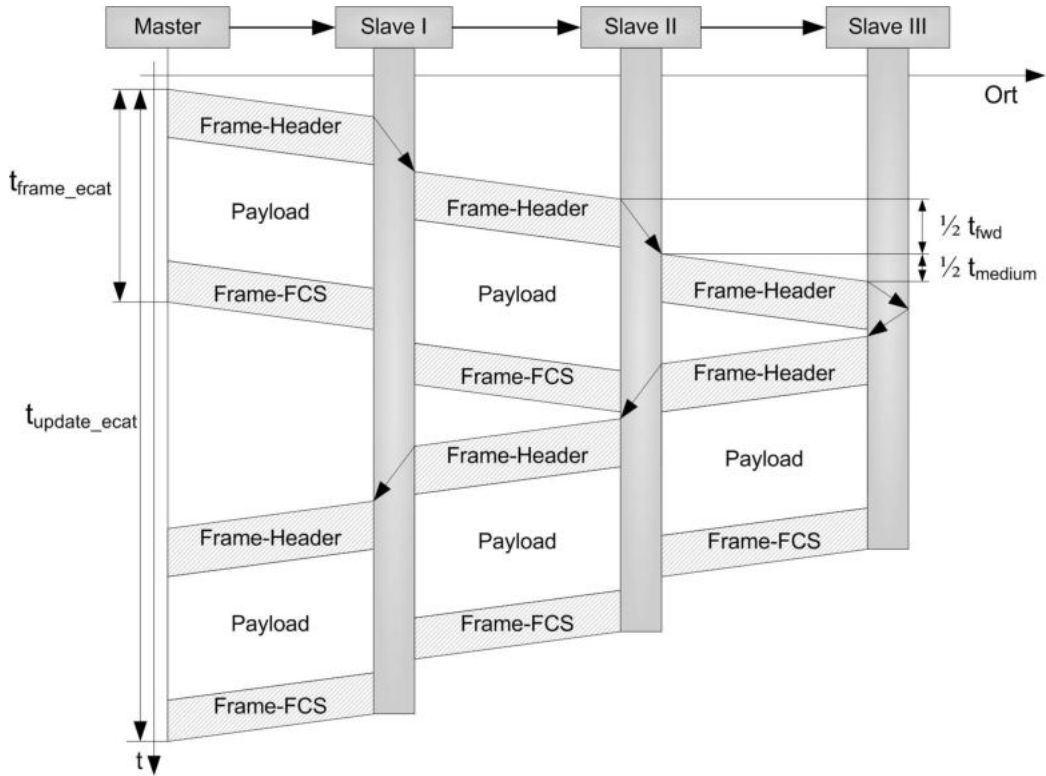


Abbildung 7: Beispiel für einen Update-Zyklus für Ethercat

t_{fwd} multipliziert mit der Anzahl der Knoten $n_{devices}$. Im Fall von Ethercat wird der Summenrahmen in einem Zyklus von dem Controller zu den Knoten und wieder zurück gesendet. Aus diesem Grunde muss die Leitungslänge der Hin- und Rückleitung berücksichtigt werden. Mit der folgenden Formel wird die Weiterleitungsverzögerung t_{fwd_ecat} berechnet:

$$t_{fwd_ecat} = n_{devices} \cdot (t_{medium} + t_{fwd}) \quad (9)$$

Hieraus lässt sich die Update-Zeit für Ethercat berechnen:

$$t_{update_ecat} = t_{frame_ecat} + t_{fwd_ecat} \quad (10)$$

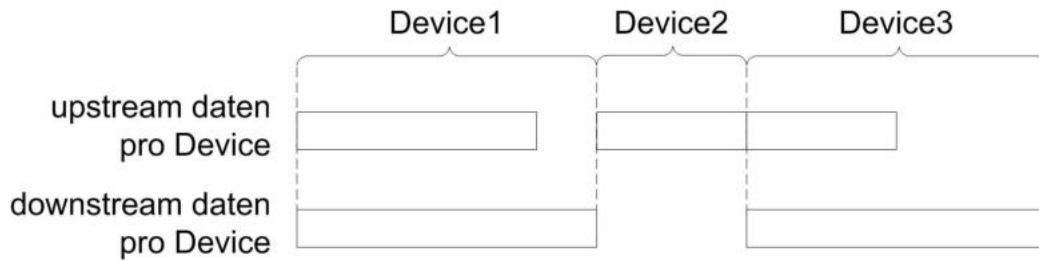


Abbildung 8: Beispiel für eine Asymmetrie zwischen den Ein/- und Ausgangsdaten der drei Knoten

Benchmarking

Allgemeine Annahmen: Der Bewertung der beiden Verfahren liegen die folgenden Annahmen zugrunde:

- Der Abstand zwischen zwei Knoten wird mit 5 Metern festgelegt.
- Für alle Betrachtungen wird die minimal erreichbare Update-Zeit ermittelt, d.h. ohne Berücksichtigung eines NRT-Kanals.
- Da die Menge der zu übertragenden Daten vom Controller zu den Knoten (Downstream) in vielen Anwendungen grösser oder kleiner der Menge von Daten von den Knoten zum Controller (Upstream) ist, wird eine Asymmetrie zwischen den Ein- und Ausgangsdaten eines Knoten von 25% angenommen.

Profinet:

- Die Weiterleitungsverzögerung eines Profinet Knotens in einem Fast-Ethernet Netzwerk wird mit $t_{fwd} = 3\mu s$ angenommen. Diese Annahme stützt sich auf Messergebnissen aus [?]. Bei einem Umstieg auf Gigabit-Technik wird sich die Weiterleitungsverzögerung deutlich verringern. Zum einen verringert sich die Zeit bis die für die Weiterleitungsentscheidung notwendigen Daten eingetroffen sind um den Faktor 10 und zum anderen wird sich die Verzögerungszeit für die Zuweisung der Zieladresse zu den entsprechenden Port aufgrund der fünffachen Taktrate der

GMII-Schnittstelle ebenfalls verringern. Aus diesem Grunde wird die Weiterleitungsverzögerung in einem Gigabit-Szenario mit $t_{fwd} = 600ns$ angenommen.

- Die Verzögerung des physikalischen Transceivers (PHY) t_{phy} wird mit $320 ns$ angenommen und entspricht einer Messung mit dem Broadcom PHY BCM5461S [?].
- Die minimale Framegröße beträgt 64 Byte.
- Die Leitungsverzögerung $t_{medium} = 22,7 ns$.

Ethercat:

- Die Durchleiteverzögerung eines Ethercat Knotens besteht zum einen aus der PHY- Verzögerung und zum anderen aus der Verzögerung, die bei der Framedurchleitung von dem Controller hin zu den Knoten und zurück zu dem Controller entsteht. In einem Fast-Ethernetnetzwerk wird für die Weiterleitungsverzögerung $t_{fwd} = 1,35\mu s$ angenommen. Bei der Betrachtung des Gigabit-Szenarios wird Ethercat ebenfalls einen Performancevorteil erfahren. In diesem Fall wird eine Durchleiteverzögerung von $t_{fwd} = 850ns$ angenommen.
- Aufgrund der Hin- und Rückleitung beträgt die Leitungsverzögerung $t_{medium} = 45,4 ns$.
- Für eine vereinfachte Diagnose eines Ethercat-Systems wird eine Gruppe von 10 Knoten jeweils zu einem Telegramm zusammengefasst.

In der Abbildung 9 wird ein moderates Szenario von 50 Knoten in einer Linientopologie dargestellt. Anhand der Abbildung ist zu erkennen, dass Ethercat bei kleinen Datenmengen einen deutlichen Performancevorteil im Vergleich zu Profinet aufweist. Dieser Vorteil basiert auf der guten Ausnutzung eines minimalen Ethernet Frames (64 Byte) durch Ethercat.

In der Abbildung 10 wird das Szenario aus Abbildung 9 für ein Gigabit Netzwerk betrachtet. Es ist zu erkennen, daß Profinet bis zu einer Nutzdatenmenge von 55 Byte pro Knoten den Performancevorteil ausbaut. Bei einer Nutzdatenmenge von 55 Byte wird der Winkel α null. Bei Datenmengen $\geq$

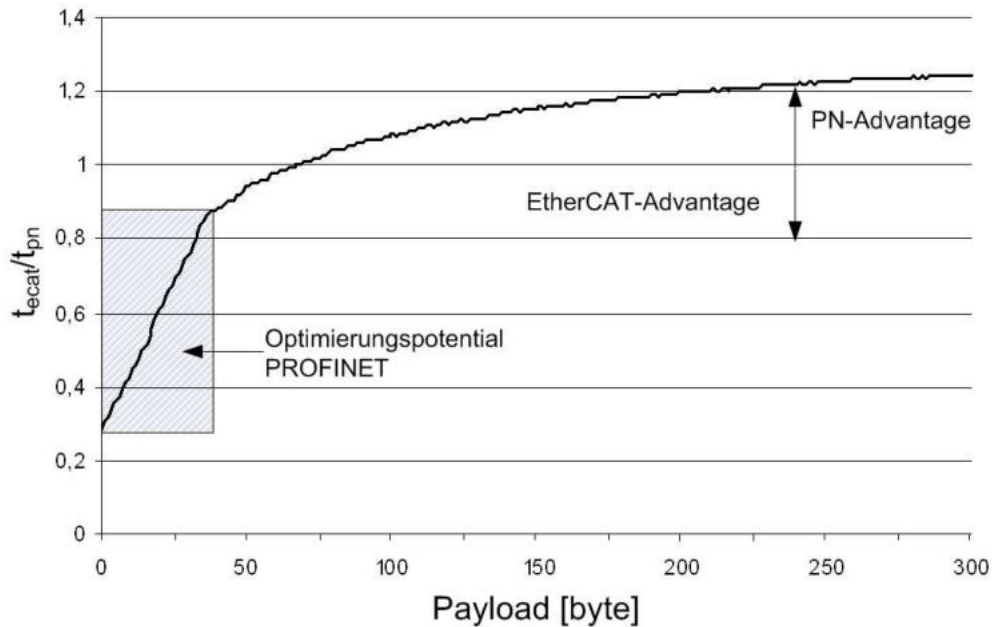


Abbildung 9: Profinet vs. Ethercat in einem Fast-Ethernet-Netzwerk und einer Linientopologie mit 50 Knoten als Funktion der Nutzdatenmenge pro Knoten.

55 Byte wird der Winkel negativ und das System profitiert nicht weiter von einer Erhöhung der Bitrate.

Zusammenfassung der Potentialanalyse: Die wichtigsten Einflussfaktoren für Leistungsbetrachtungen von RTEs sind die physikalische Laufzeit und die Frametransferzeit. Während die physikalischen Laufzeiten im Vergleich zu den Übertragungszeiten der Frames bei den heutigen Feldbussystemen vernachlässigbar sind, wird diese Zeitkomponente bei Ethernet aufgrund der hohen Bitrate schnell zu einem dominierenden Faktor. Dieser Effekt wird sich bei künftigen noch höheren Bitraten weiter verstärken. Im Szenario der einfachen Sensor-/Aktorvernetzung mit reiner Linienstruktur und 100 Mbit/s ist Profinet nur für größere Nutzdatenmengen pro Knoten im Vorteil. Bei kleinen Nutzdatenmengen pro Knoten, üblicherweise bei einfachen Senso-

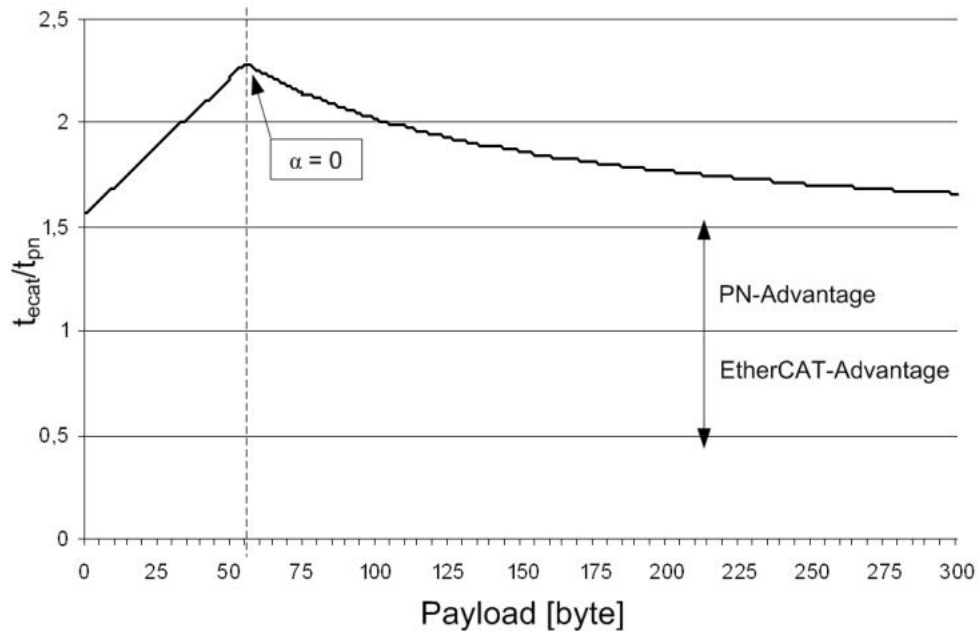


Abbildung 10: Profinet vs. Ethercat in einem Gigabit-Ethernet-Netzwerk und einer Linientopologie als Funktion der Nutzdatenmenge pro Knoten.

ren/Aktoren, besteht Optimierungspotenzial. In diesem Fall ist Ethercat im Vorteil.

4.1.2 Optimierungsansätze

Die Potentialanalyse hat gezeigt, dass für zukünftige RTEs parallel zu der Bitrate, insbesondere die Weiterleitungszeit der einzelnen Knoten optimiert werden muss. Zunächst werden zwei Ansätze zur Verringerung der Weiterleitungsverzögerung vorgestellt. Prinzipiell gesehen kann ein Profinet-Knoten wie eine MAC-Bridge (Switch) mit mindestens drei Ports betrachtet werden. Der dritte Port muss dabei nicht physikalisch ausgeprägt sein, sondern stellt die Verbindung zur Geräteapplikation dar. Die Weiterleitungsverzögerung ei-

nes Knotens besteht aus drei Hauptkomponenten:

- Verzögerung des PHY t_{phy}
- Anzahl von Bytes eines Frames, die benötigt wird, um eine Weiterleitungsentscheidung treffen zu können t_{fwd_data} .
- Logikverzögerung aufgrund der Adress-Lookup und der internen Vermittlungseinheit t_{logik} .

Die Logikverzögerung eines Switches wird primär von dem Suchalgorithmus bestimmt, der benötigt wird um den entsprechenden Port für die Zieladresse ausfindig zu machen. Da die Suchalgorithmen schon seit vielen Jahren optimiert wurden und ausgreift sind, bieten sich hier keine signifikanten Optimierungsmöglichkeiten an.

Topologiebasierte Adressierung Aus diesem Grunde wurde in dem Projekt der Ansatz verfolgt den Suchalgorithmus durch einen topologiebasierten Ansatz zu ersetzen, um die Logikverzögerung zu reduzieren. Damit das topologiebasierte Verfahren zukunftssicher ist, muss es in der Lage sein auch komplexe Netzwerkstrukturen zu bedienen. Die Abbildung 11 zeigt beispielhaft eine Topologie mit einer Tiefe von 3 Ebenen. Prinzipiell basiert das Verfahren auf der Annahme, dass jede Ebene durch ein dediziertes Byte in der Zieladresse repräsentiert ist und die Position eines Knotens der Position des vorhergehenden Knotens (aus Sicht des Controllers) plus 1 entspricht.

In der Abbildung 11 sind die drei Portklassen Primär, Sekundär und Abzweig, sowie deren Zusammenhang mit der Positionsnummer einer jeweiligen Ebene eines Knotens, dargestellt. Der primäre Port verbindet einen Knoten mit dem vorhergehenden Knoten der sich auf der gleichen Ebene befindet. Für den Fall, dass der Knoten der erste einer Ebene ist, wird über den primären Port die übergeordnete Ebene angebunden (z.B. Ebene 2 → Ebene 1). Ein nachfolgender Knoten in der Linie wird über den sekundären Port angeschlossen. Eine tiefer liegende Ebene (z.B. Ebene 1 → Ebene 2) wird über den Abzweig-Port angebunden. Das vorgestellte Adressierungsverfahren kann mit Hilfe lokal administrierter MAC-Adressen realisiert werden. Da eine MAC-Adresse aus 6 Bytes besteht, ist es damit prinzipiell möglich, eine

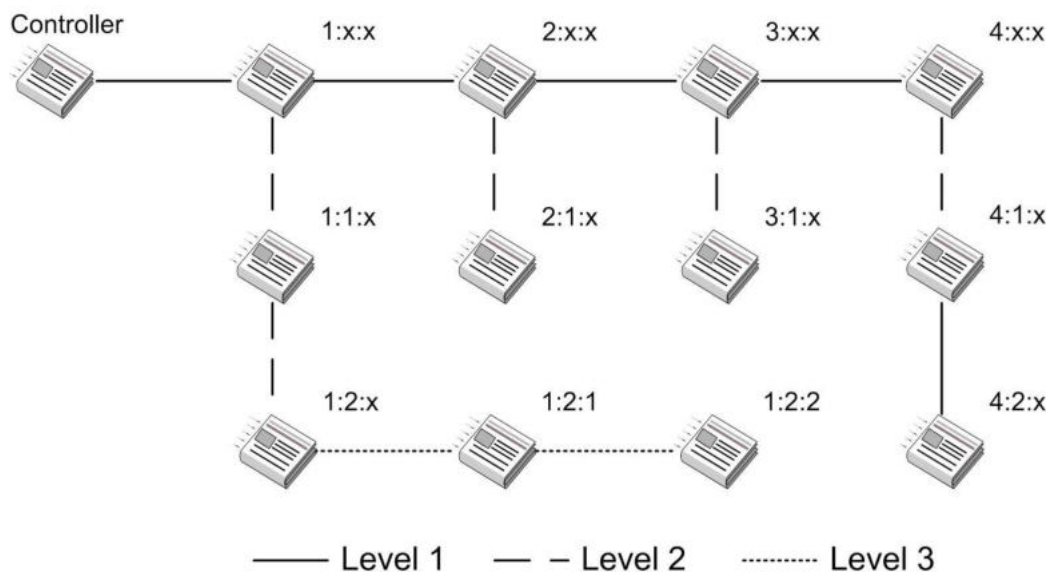


Abbildung 11: Beispiel einer topologiebasierten Adressierung

Tiefe von bis zu 6 Ebenen zu unterstützen. In der Abbildung 13 wird ein Kodierungsbeispiel für eine lokal administrierte MAC-Adresse vorgestellt. Der IEEE-Standard schreibt vor, dass das niederwertigste Bit des ersten Bytes den Frame als Uni-/Multicastframe (Unicast = 0 / Multicast = 1) definiert. Das zweite Bit legt fest, ob die MAC-Adresse lokal oder global administriert ist. Aufgrund dieser durch den IEEE-Standard vorgegebenen Bedingungen, ist die Struktur des ersten MAC-Adressbytes nicht homogen zu den folgenden 5 Bytes. Daher ist es vorteilhafter, das erste Byte nicht für die Adressierung zu verwenden und stattdessen für zukünftige Erweiterungen zu reservieren. Aus diesem Grunde beginnt die Einbettung der Topologieinformation erst ab dem zweiten Byte. Weiterhin zeigt die Abbildung, dass die Positionsnummer in der jeweiligen Ebene durch die Bits [5:0] kodiert wird. Die Bits [7:6] kennzeichnen einen möglichen Abzweig in eine tiefere Ebene.

In der Abbildung 14 wird ein Frametransfer von dem Controller hin zu dem Knoten C2:1:x dargestellt. Die Abbildung 15 zeigt für das Beispiel die entsprechende Ziel-MAC-Adresse des Frames. Der Frame wird zunächst von dem Controller zu dem Knoten (1:X:X) gesendet. Dieser Knoten erkennt an dem gesetzten zweiten Bit im Byte 1, dass es sich um einen lokal administrierten Frame handelt und aktiviert daraufhin das topologiebasierte Weiterlei-

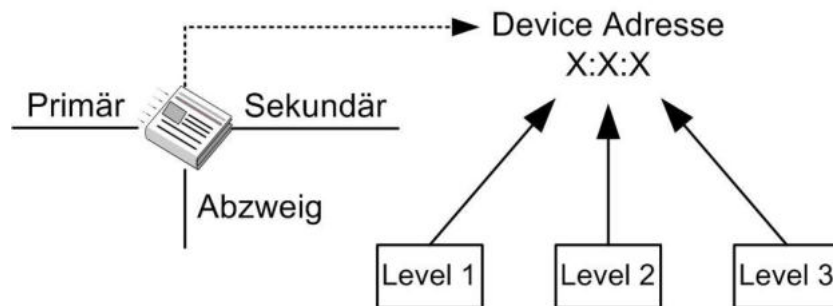


Abbildung 12: Beschreibung des Ports/Adresse

ten. Im Anschluss daran wird das zweite Byte der MAC-Adresse empfangen, der Knoten vergleicht zunächst die eigene Positionsnummer mit der Positionsnummer des Frames (binär "10" → dezimal 2). Da die Positionsnummer der Ziel-MAC-Adresse größer als die eigene ist, wird der Frame über den sekundären Port zu dem Knoten (2:X:X) weitergeleitet. Dieser Knoten vergleicht ebenfalls die Positionsnummer im Frame mit der Positionsnummer des Knotens. Da die beiden Positionsnummern identisch sind, wird zusätzlich der Abzweig (Branch) (Bit [7:6]) ausgewertet und der Knoten sendet den Frame über den Branch Port 3 (Bit[7:6]=11 → Branch 3) weiter zu dem Knoten C2:1:X. Der Frame befindet sich nun auf der Ebene 2 und es wird die Positionsnummer des dritten Bytes der MAC-Adresse zur Weiterleitung mit der Positionsnummer des Knotens verglichen. Aufgrund der identischen Positionsnummern werden zusätzlich die Branch-Bits (00 → Frame ist für diesen Knoten bestimmt) ausgewertet und der Frame hat nun den Zielknoten erreicht. Anhand des Beispiels ist zu erkennen, dass die Ebene eines Knotens direkt mit der Anzahl auszuwertender Adressbytes gekoppelt ist. Ein Knoten auf Ebene 1 wertet das Status- und das erste Adressbyte aus und ein Knoten auf Ebene 2 bewertet das Status- und die nächsten zwei Adressbytes aus. Diese Reihe wird fortgesetzt bis zur Ebene 5, auf der alle Bytes der MAC-Adresse ausgewertet werden müssen.

Eine weitere Möglichkeit mit der die Weiterleitungsverzögerung verringert werden kann, ergibt sich durch die Verkürzung der Präambel. Da heutige Echtzeit-Ethernet Protokolle ein Voll-Duplex Netzwerk voraussetzen, ist die Präambel theoretisch obsolet. Aufgrund angestellter Versuche konnte festgestellt werden, dass trotz einer Vollduplex-Unterstützung ein Teil der Präambel weiterhin von den eingesetzten PHYs vorausgesetzt werden. Es

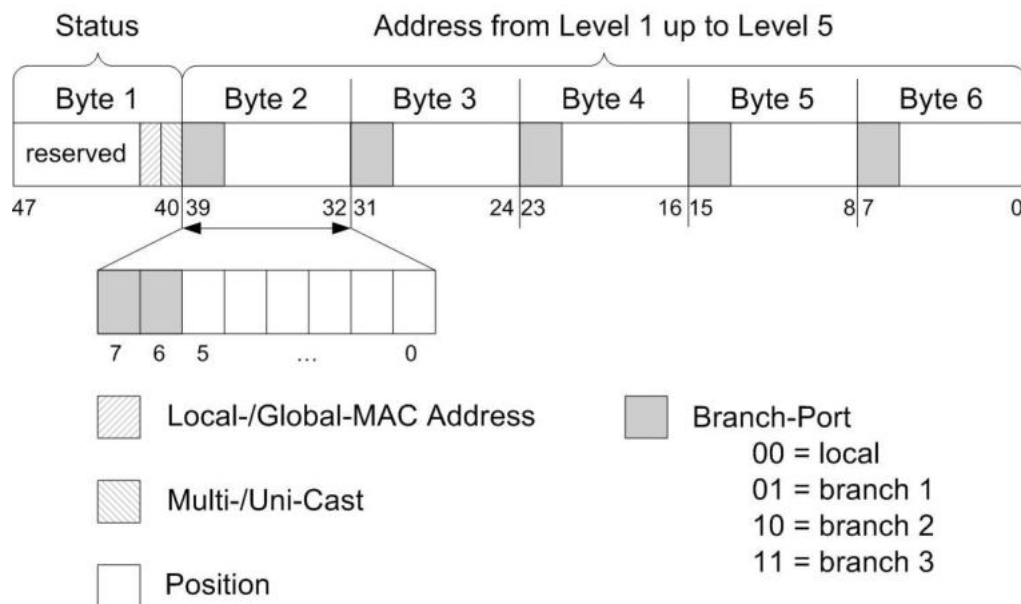


Abbildung 13: Kodierung einer lokalen MAC-Adresse

zeigte sich, dass die verwendeten Netzwerkkarten (Marvel, Belkin, Intel) einen Frame nur dann sicher erkennen, wenn 1 Byte Präambel und SOF vorhanden sind. Damit Fehlfunktionen aufgrund einer Verkürzung der Präambel ausgeschlossen werden können, sollten zwei benachbarte Knoten diese Funktion miteinander aushandeln (z.B. unter Verwendung des LLDP Protokolls). Der Zeitgewinn aufgrund einer Präambelverkürzung beträgt 6 Byte und entspricht 480 ns bei der Verwendung von Fast Ethernet. Die minimal benötigten Informationen eines Frames für eine Weiterleitungsentscheidung bestehen bei einem Switch mit einer klassischen Cut-Through Strategie aus 8 Byte Präambel /SFD und 6 Byte Ziel-MAC-Adresse. Bei der Verwendung von Fast-Ethernet entspricht das einer benötigten Zeit von $14 \cdot 8 \cdot 10ns = 1120ns$. Die Verzögerung der optimierten Variante besteht bei einem Knoten auf Ebene m aus 2 Byte Präambel /SFD und $m + 1$ Byte der Ziel-MAC-Adresse. Das ergibt im günstigsten Fall eine Verzögerung von 320 ns (insgesamt 4 Byte für einen Knoten auf Ebene 1) und im ungünstigsten Fall von 640 ns (insgesamt 8 Byte für einen Knoten auf Ebene 5). Die Logikverzögerung eines Standardswitches wurde mit der folgenden Formel bestimmt und beträgt 1560 ns:

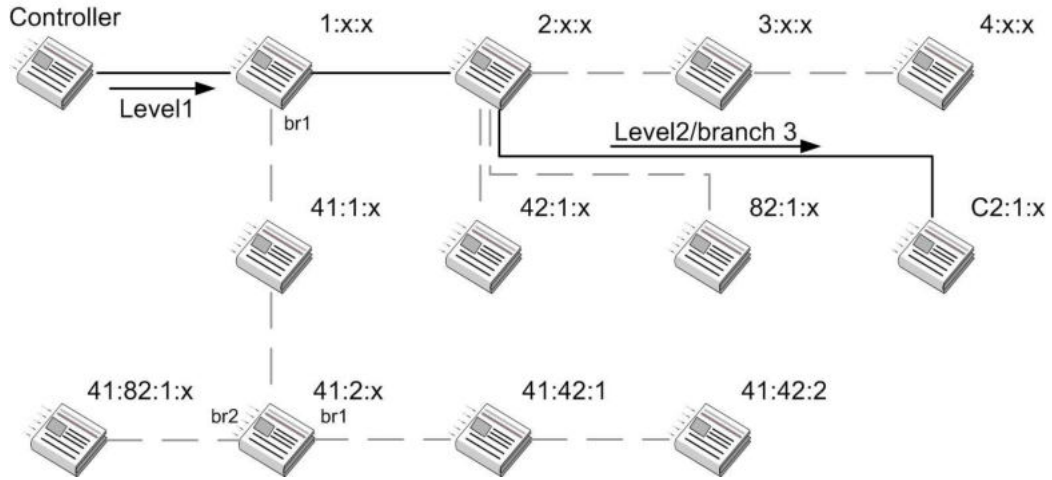


Abbildung 14: Kommunikation vom Controller zum Knoten C2:1:X

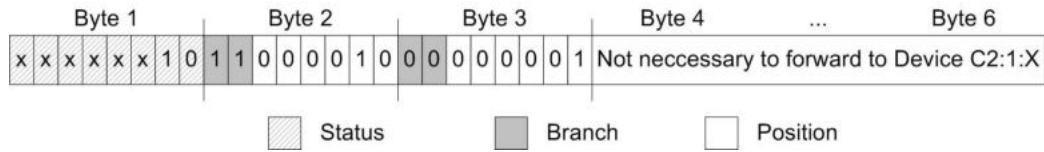


Abbildung 15: Beispiel einer lokal administrierten unicast MAC-Adresse

$$t_{logik} = t_{fwd} - t_{Phy} - t_{fwd_data} \quad (11)$$

Da für die Weiterleitungsentscheidung auf Basis des topologiebasierten Ansatzes nur ein Vergleich durchgeführt werden muss, ist die Logikverzögerung mit 160 ns deutlich kleiner (gemessen mit einem FPGA Prototyp). Aufgrund der vorgestellten Optimierungen beträgt die für eine Weiterleitungsentscheidung notwendige Zeit eines Knotens auf Ebene 1 im Vergleich zu einem Standardswitch nur 26,7 %.

Dynamic Frame Packing (DFP) Zusätzlich zur Optimierung der Weiterleitung wurde ein intelligenter Packmechanismus erarbeitet. Das Ziel des Dynamic Frame Packings (DFP) besteht darin, den Overhead des Protokolls zu minimieren und damit die Frametransferzeit so klein wie möglich zu halten. Im Fall von Profinet entsteht der größte Overhead aufgrund der mini-

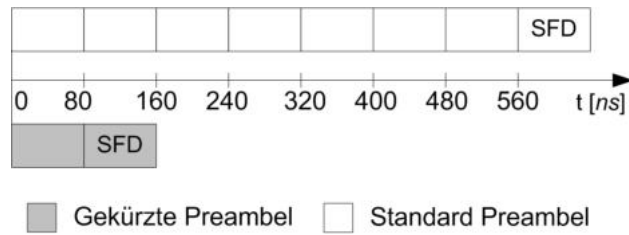


Abbildung 16: Verkürzung der Präambel

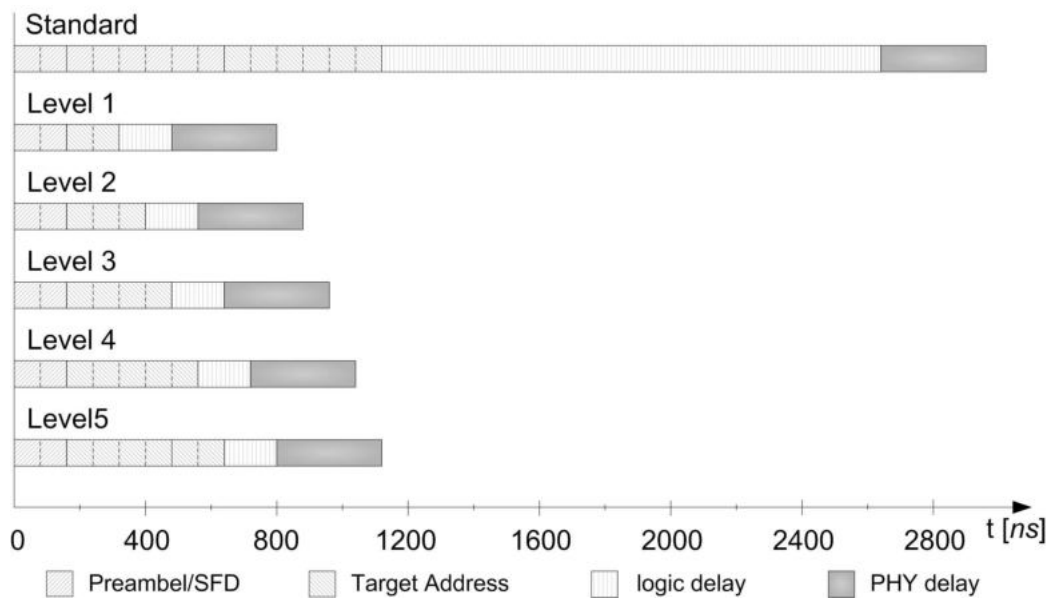


Abbildung 17: Bewertung der Optimierungen zur Weiterleitung

malen Framegröße eines Ethernet Frames (64 Byte). Wenn die Datenmenge eines Knotens 46 Byte (minimale Payload eines Ethernet Frames) unterschreitet, erfolgt ein Paddingvorgang, bei dem die übrigen Bytes mit leeren Bytes ("0") aufgefüllt werden. Damit dieser aus Sicht der Effizienz ungünstige Effekt möglichst klein ist, macht es Sinn die Daten mehrerer Knoten zu einer Gruppe zusammenzufassen und so den Nutzdatenanteil eines Frames im Verhältnis zum notwendigen Overhead zu verbessern. Die Daten jedes teilnehmenden Knotens werden hierzu in ein Datagramm eingebettet. Der Aufbau eines Datagramms ist in der Abbildung 18 dargestellt. Das erste Byte eines Datagramms beinhaltet die Position des Knotens in der entsprechenden

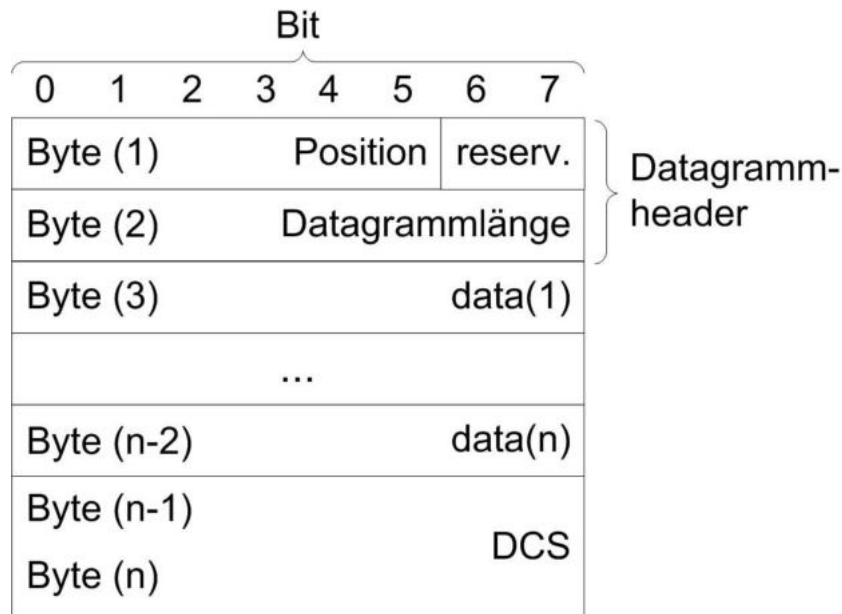


Abbildung 18: Struktur eines DFP-Datagramms

Ebene (Bit 5:0), die Bits [7:6] werden für zukünftige Ergänzungen reserviert. Da die Positionsnummer aus 6 Bit besteht, ist die maximale Linientiefe einer Ebene auf 64 Knoten begrenzt (6 Bit $\rightarrow$ 0-63). Für den Fall, dass eine Linientiefe > 64 Knoten gefordert ist, wird in eine tiefere Ebene gewechselt, in der wieder 64 Knoten adressiert werden können. Auf diesem Wege kann eine maximale Linientiefe von $5 \text{ Ebenen} \cdot 64 \text{ Knoten/Ebene} = 320 \text{ Knoten}$ erreicht werden. Das zweite Byte beinhaltet die Länge des Datagramms. Auf Basis dieser zwei Bytes sind die Grenzen eines Datagramms eindeutig definiert. Die Integrität des Datagramms wird durch einen zusätzlichen CRC-16 in der Datagram Check Sequence (DCS) sichergestellt. Um eine möglichst gute Ausnutzung der Bandbreite zu ermöglichen, bietet es sich an, die Datagramme so zu packen, dass der in Kapitel 4.1.1 beschriebene Windschatten effekt genutzt werden kann. Bei der Verwendung des DFP-Mechanismus wird zwischen der Downstream-Kommunikation (Controller $\rightarrow$ Knoten) und der Upstream-Kommunikation (Knoten $\rightarrow$ Controller) unterschieden. Die Abbildung 19 zeigt ein Beispielszenario zur Downstream-Kommunikation mit einem Controller und 3 Knoten. Der Pack Frame ist in einen Standard Ethernet Frame (Ethernet Header + FCS) eingebettet, wobei die Payload des Frames

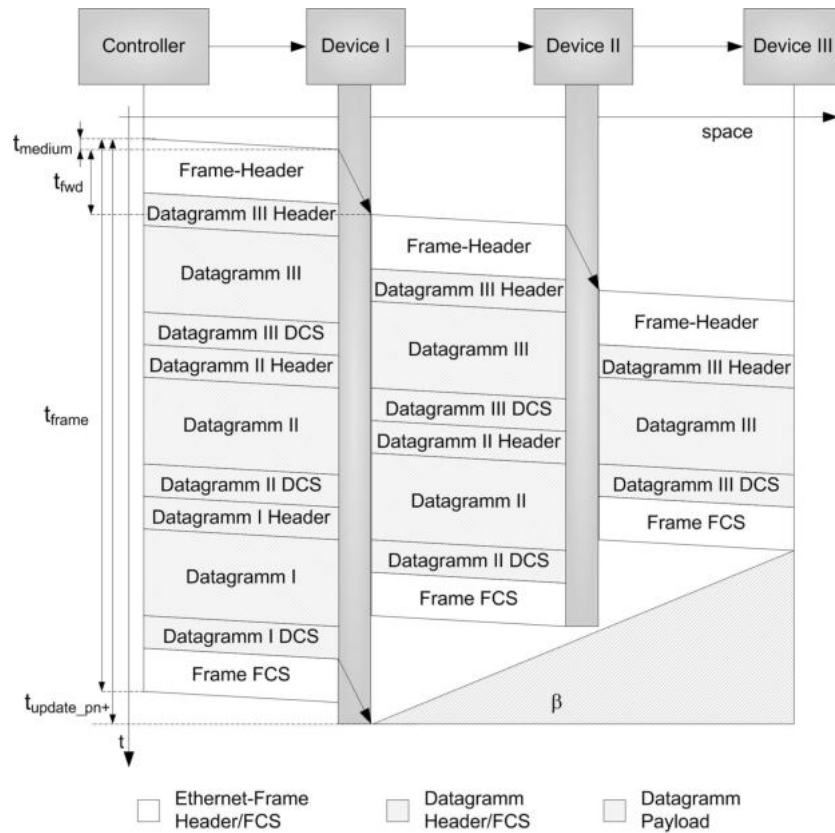


Abbildung 19: Dynamic Frame Packing Controller $\rightarrow$ Knoten (β ist positiv)

aus den Datagrammen der teilnehmenden Knoten besteht. Da jeder Knoten ein dediziertes Datagramm erhält, ist es vorteilhaft, dass ein Knoten das ihm gehörende Datagramm während der Weiterleitung aus dem Frame entfernt und auf diesem Wege die Framelänge aktiv verkürzt. Aufgrund der Anordnung auf Basis des Windschatteneffekts entfernt ein Knoten immer das letzte Datagramm eines DFP-Frames und ersetzt es durch die FCS. Da durch das Verkürzen des Frames der ursprüngliche Frame manipuliert wurde, muss die FCS neu berechnet und eingefügt werden. Um die Integrität während des Durchleitens sicherzustellen, wird die Fehlererkennung von der DCS übernommen. Hierzu beginnt die Berechnung jeweils mit dem ersten Byte des Ethernet Frames (erstes Byte der Ziel-MAC Adresse). Für den Fall, dass aufgrund der Framekürzung die gesamte Framelänge kleiner als die spezifizierte minimale Ethernet-Framelänge wird, muss die Payload des Frames

durch den Einsatz von Paddingbytes auf 46 Byte ergänzt werden.

Wie oben schon erläutert, zeigt das Vorzeichen des Winkels α an, ob die Zykluszeit des optimierten Systems von der Bitrate oder von der Weiterleitungsverzögerung bestimmt wird. Dieser Winkel wird für das DFP-Verfahren im folgenden zur Abgrenzung β genannt. Die bitraten-abhängige Größe $t_{datagram}$ besteht aus der Übertragungszeit eines Datagramms:

$$t_{datagram} = \frac{d_{datagram_ov} + d_{datagram_payload} + d_{dcs}}{b} \quad (12)$$

Die Weiterleitungsverzögerung t_{fwd_pn} besteht aus der durch den Knoten verursachten Verzögerung t_{fwd} und der physikalischen Laufzeit auf dem Medium t_{medium} . Das Vorzeichen des Winkels ist positiv wenn die Bedingung $t_{datagram} \geq t_{fwd}$ erfüllt ist und negativ wenn $t_{datagram} \leq t_{fwd}$ gilt. Aus dieser Bedingung können die folgenden Rückschlüsse gezogen werden:

$\beta \geq 0$: Die Performance des Systems kann verbessert werden, indem die Bitrate b erhöht wird. Für dieses Szenario spielt die Weiterleitungsverzögerung t_{fwd} nur eine untergeordnete Rolle (siehe Abbildung 19).

$\beta \leq 0$: Die Performance des Systems kann verbessert werden, indem die Weiterleitungsverzögerung t_{fwd} optimiert wird. Für dieses Szenario tritt die Bitrate b in den Hintergrund (siehe Abbildung 20).

Der Winkel β kann für grosse $n_{devices}$ mit der folgenden Formel berechnet werden:

$$\tan \beta = \frac{t_{Datagram}}{t_{fwd}} - 1 \quad (13)$$

Die Funktionsweise der Kommunikation in Upstream-Richtung wird prinzipiell in dem Beispiel aus Abbildung 21 vorgestellt, bei dem jeder Knoten zum gleichen Zeitpunkt mit dem Sendevorgang beginnt. Während der Knoten seinen Frameheader und sein Datagramm sendet, wird bereits das Datagramm des folgenden Devices in der Linie empfangen. Betrachten wir hierzu den Knoten I. Während der Knoten sein Datagramm (Datagramm I) sendet, empfängt es bereits den Frame vom nachfolgenden Knoten II. Nachdem der Knoten den Sendevorgang für das sein Datagramm beendet hat, fügt er ohne Unterbrechung die empfangenden Datagramme, hier den Frame von Knoten II, an. Der Mechanismus ist vergleichbar mit einem Zug an dem während

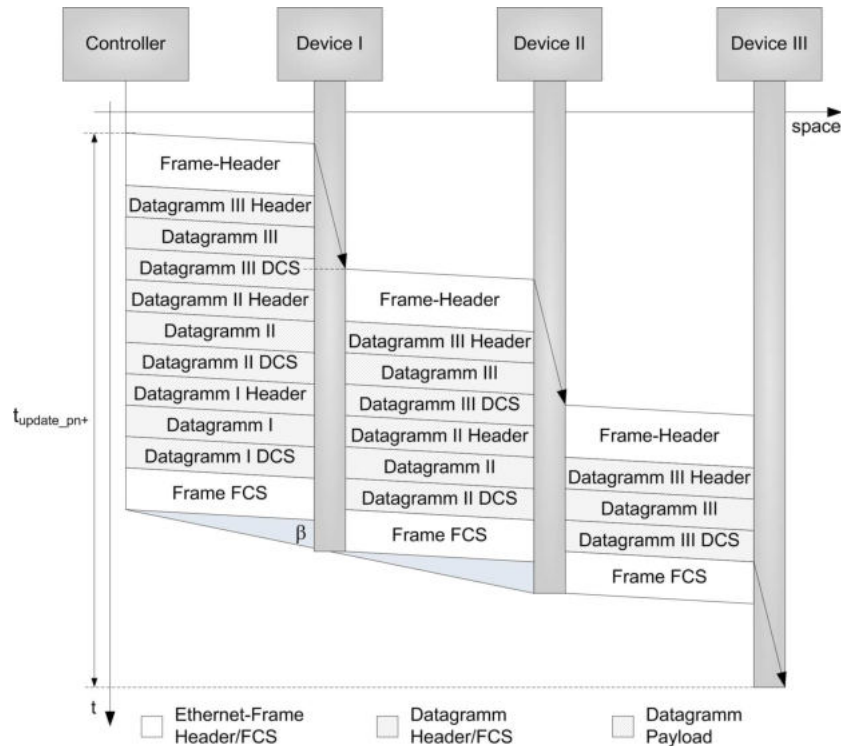


Abbildung 20: Dynamic Frame Packing Controller $\rightarrow$ Knoten (β ist negativ)

der Ausfahrt aus einem Bahnhof die Waggon eines Anschlusszuges ohne Unterbrechung angekoppelt werden. Der Packmechanismus in Upstream-Richtung setzt voraus, dass bis zu dem Zeitpunkt in dem das letzte Byte des dem Knoten zugehörigen Datagramms gesendet worden ist, das erste Byte des Datagramms des Nachfolgers empfangen wurde. In der Abbildung 22 wird ein Beispiel dargestellt, in dem diese Bedingung nicht erfüllt ist. In diesem Fall würde die Übertragung des Frames unterbrochen ohne dass der Frame erfolgreich die Daten des vorhergehenden Devices ankoppeln konnte. Dieses Szenario tritt auf, wenn zwischen den Knoten III und Knoten IV die Frameverzögerung größer ist als die Übertragungszeit des Datagramms III. Um diese Situation beherrschbar zu machen, muss der Sendezeitpunkt des Devices III verzögert werden. Hierzu wird die Verzögerungszeit t_{Offset} eingeführt. Auf Basis von t_{Offset} wird der Sendezeitpunkt des Devices III entsprechend verzögert.

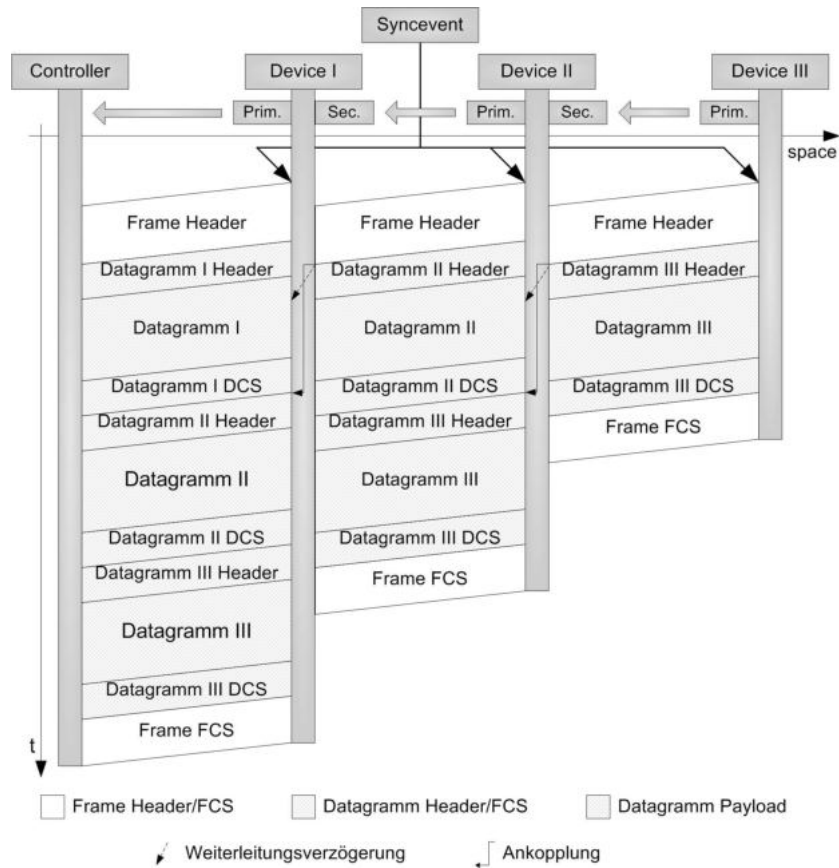


Abbildung 21: Dynamic Frame Packing Knoten → Controller

4.1.3 Leistungsbewertung

Auf Basis der zuvor gewonnenen Ergebnisse der Potentialanalyse werden die vorgestellten Optimierungen nun bewertet. Die Berechnung der minimalen Updatezeit von Ethercat wurde mit der Formel aus Kapitel 4.1.1 berechnet:

$$t_{update_ecat} = t_{frame_ecat} + t_{fwd_ecat} \quad (14)$$

Die Berechnung von Profinet basiert ebenfalls auf der Formel aus Kapitel 4.1.1:

$$t_{update_pn} = \max(n_{Devices} \cdot t_{frame} + t_{fwd_pn}, t_{frame} + n_{Devices} \cdot t_{fwd_pn}) \quad (15)$$

Da bei dem optimierten Ansatz nur ein Frame versendet wird, muss die Berechnung von Profinet angepasst werden. Die Update-Zeit der optimierten Profinet-Variante ergibt sich zu:

$$t_{update_pn+} = \max(t_{fwd_pn+} + t_{frame+}, n_{devices} \cdot t_{fwd_pn+} + \frac{t_{frame+}}{n_{devices}}) \quad (16)$$

Die Frametransferzeit der optimierten Profinet-Variante t_{frame+} wird mit der folgenden Formel berechnet:

$$t_{frame+} = \frac{n_{devices}(d_{datagram_payload} + d_{datagram_ov}) + d_{ethernet_ov} + d_{pn_ov}}{b} \quad (17)$$

Die Größe des DFP-Frames ergibt sich aus der Anzahl der verwendeten Knoten $n_{devices}$, der Payload eines Datagramms $d_{datagram_payload}$, dem Overhead eines Datagramms $d_{datagram_ov}$ (2 Byte Datagramm-Header und 2 Byte DCS) und dem durch Ethernet hervorgerufenen Overhead $d_{ethernet_ov}$. Die Ergebnisse des Vergleichs werden in der Abbildung 23 dargestellt. Die Bewertung betrachtet ein Szenario mit 50 Knoten als Funktion der Payload pro Knoten in einer Linientopologie und einer Bitrate von 100 Mbit/s. Das Diagramm zeigt, dass sich der oben dargestellte Performancenachteil von Profinet bei kleinen Datenmengen unter Verwendung der in dem Projekt erarbeiteten Optimierungen in einen Vorteil wandelt.

Abschließend wird in der Abbildung 24 die erreichbare Update-Zeit für PN+ als Funktion der Knotenanzahl und der Payload pro Knoten dargestellt.

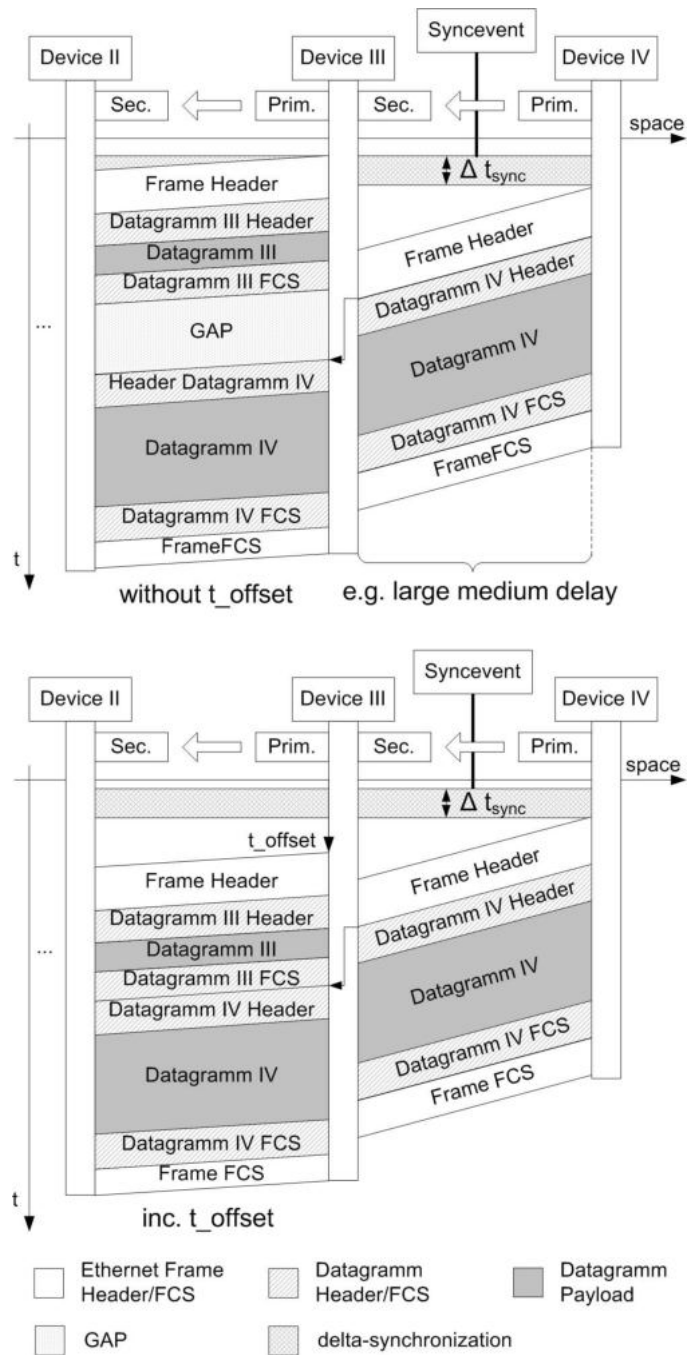


Abbildung 22: Spezialfall DFP in Upstream-Richtung

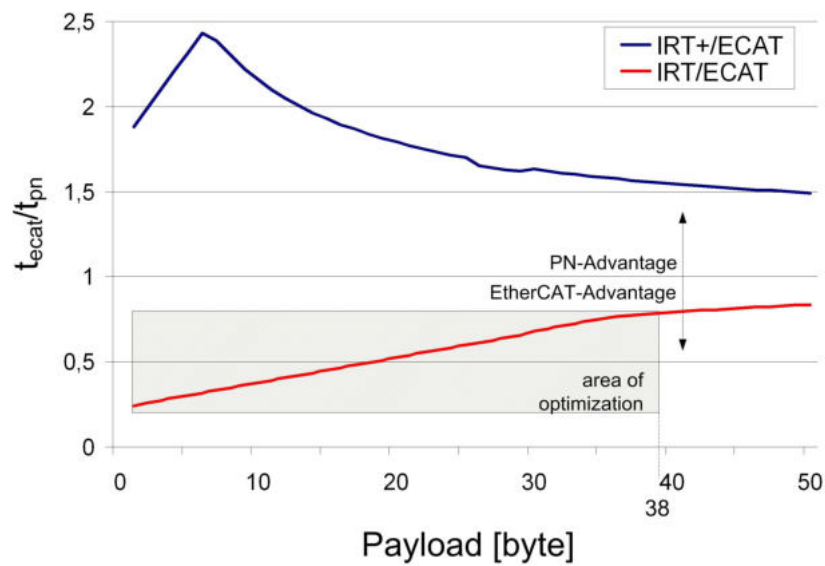


Abbildung 23: Update-Zeit als Funktion der Payload eines Knotens bei der Verwendung von Fast Ethernet, 50 Knoten in einer Linientopologie und einer Asymmetrie von 25% zwischen den Ein- und Ausgangsdaten.

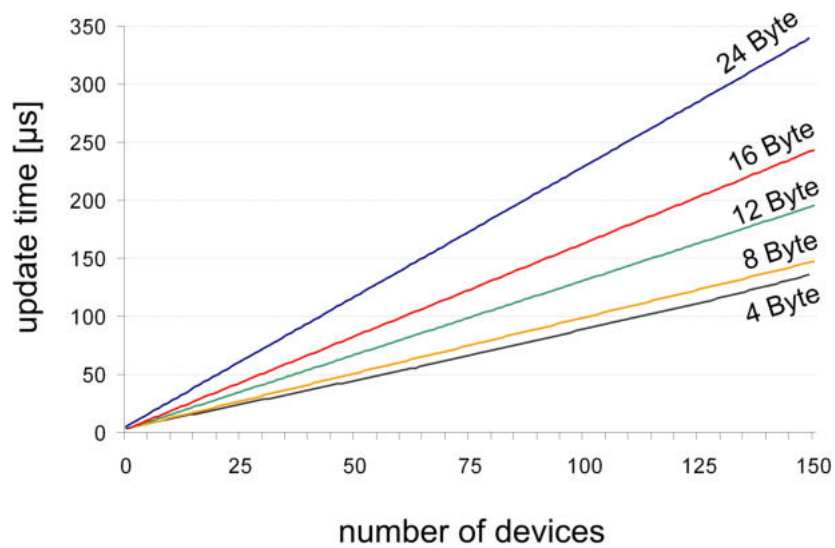


Abbildung 24: Update-Zeit von PN+ als Funktion der Knotenanzahl und der Payload pro Knoten

4.2 Einfachheit

Der zweite Arbeitsbereich des ESANA-Projektes adressiert die in Kapitel 1 beschriebene Leitfrage "Wie kann man die Komplexität von Profinet für die Implementierung in einfache Sensor/Aktor-Knoten reduzieren?".

Zum Zeitpunkt des Projektstarts Ende 2006 konnte noch auf wenig Implementierungsunterstützung in Form von Evaluationsboards für Profinet zurückgegriffen werden.

Um einen ersten Eindruck vom Stand der Implementierungen zu bekommen, wurde zunächst eine Diplomarbeit [24] mit dem ASIC netX der Firma Hilscher [25] durchgeführt. In dieser Arbeit wurde ein RF-ID Lesegerät mit Profinet-IO Devicefunktion realisiert. Der zu diesem Zeitpunkt vorhandene Profinet-IO Protokollstapel war noch in einem sehr frühen Entwicklungsstadium. Diese Arbeit wurde im Mai 2007 von der Firma Hilscher im Rahmen eines bundesweit ausgelobten Wettbewerbs 'netX - go for hightech' mit dem 1. Preis ausgezeichnet¹.

In einem folgenden Arbeitsschritt wurde das ERTEC-Entwicklungskit EB200 der Firma Siemens untersucht. Erstes Teilziel war es, die mitgelieferten Beispielapplikationen in der Entwicklungsumgebung lauffähig zu machen. Danach wurde mit Hilfe des Development Kits und den mitgelieferten Beispielen ein einfaches eigenes Profinet-Device entwickelt. Hier trat die hohe Komplexität von Profinet sehr schnell zu Tage, da es aufgrund der vielen unterschiedlichen Tools sehr mühselig war, eine Beispielapplikation zu entwerfen. In Absprache mit der Firma Siemens wurde vereinbart, dass das inIT die gesammelten Erfahrungen so zu einem Leitfaden aufarbeitet, dass ein Entwickler ohne Vorerfahrung in deutlich kürzerer Zeit seinerseits ein entsprechendes eigenes Profinet-Knoten entwickeln kann. Die wesentliche Herausforderung bestand darin, aus der Flut an Informationen einen kompakten Leitfaden zu erstellen, der in Form einer Schritt-für-Schritt-Anleitung den Entwickler zu einem Profinet-Device führt. Im Ergebnis entstand ein 73 Seiten umfassender Leitfaden, der ebenfalls unter der Webseite (http://www.hs-owl.de/init/uploads/media/Leitfaden_EB200_01.pdf) frei verfügbar ist und seitdem vielfach heruntergeladen wurde. Ebenfalls ist eine

¹s.a. http://www.hs-owl.de/init/no_cache/aktuelles/news/news-einzelansicht/news/beste-diplomarbeit-ausgezeichnet-netx-go-for-hightech/634.html

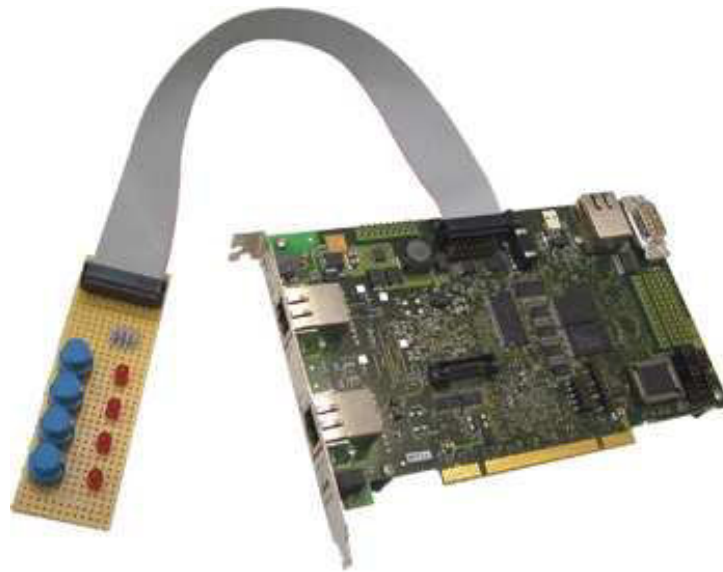


Abbildung 25: Das EB200 mit Beispielapplikation für eine einfache Remote-I/O Anwendung

neue Beispielapplikation für ein Remote-I/O entstanden, die auch auf der angegebenen Webseite abrufbar ist. Abgeschlossen wurde dieses Teilprojekt dadurch, dass ein Diplomand des Instituts, der bis zu diesem Zeitpunkt nicht im Projekt involviert war, die Tauglichkeit des Leitfadens testete. Aufgrund verschiedener Lehrveranstaltungen besaß er Grundkenntnisse im Bereich Profinet und eingebetteter Systeme, hatte also ähnliche Voraussetzungen wie ein Geräteentwickler in der Industrie, der zwar allgemeines Vorwissen hat, sich aber in diesen neuen speziellen Bereich einarbeiten muss. Der Testperson gelang es innerhalb von zwei Arbeitstagen, ein entsprechendes Device zu konfigurieren. Das Teilprojekt ist unter der Webseite <http://www.hs-owl.de/init/research/projects/b/filteroff/76/single.html> dokumentiert.

Durch den Leitfaden wurde die inhärente Komplexität von Profinet natürlich nicht reduziert. Er kann aber kurzfristig dazu beitragen, dass Gerätehersteller effizienter zu einer Device-Implementierung auf Basis des ERTEC kommen.

In Anbetracht der knapp bemessenen Projektlaufzeit wurde in Abstimmung mit den Projektpartnern festgelegt, den Fokus des Projektes auf die Performance zu legen und das Thema Einfachheit aus ESANA auszukoppeln.

Hierzu ist eine ASIC-Entwicklung der Firma Phoenix Contact unter Beteiligung des inITs gestartet worden.

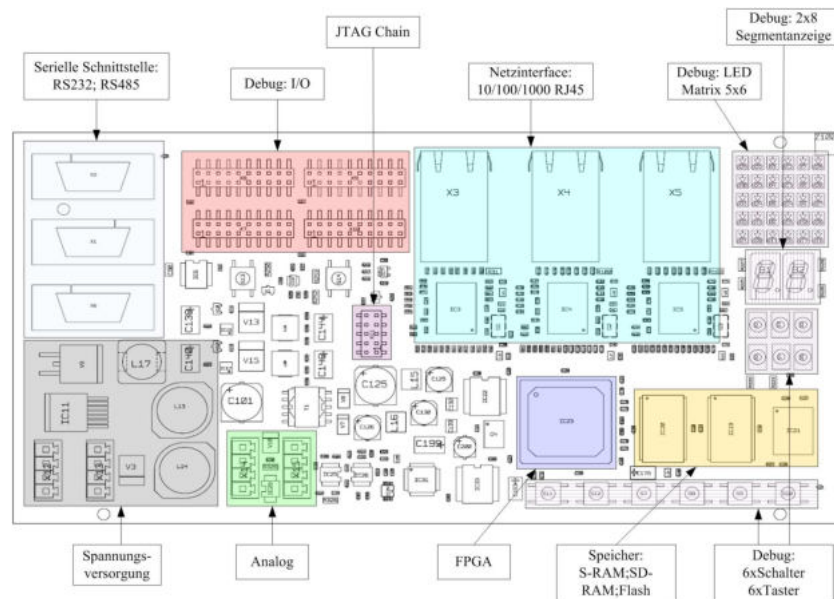


Abbildung 26: Entwicklungs-Bord (Draufsicht)

4.3 Entwicklungsumgebung

Parallel zu den Analysen entstand ein in der Hardwarebeschreibungssprache Verilog entwickelter IP-Core, mit dem die Performanceoptimierungen validiert wurden. Hierzu wurde in Kooperation mit dem Projektpartner Phoenix Contact ein Entwicklungsboard entwickelt. Das Board (siehe Abbildung 26) ist mit den folgenden Eigenschaften ausgestattet:

- Eine Analogeneinheit ist integriert. Auf diesem Weg ist das Entwicklungsboard u.a. auch für regelungstechnische Anwendungen geeignet.
- Die rekonfigurierbare Hardware des Boards ist mit einem Altera Cyclone 2 (EP2C35F484I8N) realisiert.
- Als Speicher stehen 8 MByte SD-RAM, 2 MByte S-RAM und 16 MByte Flash-Speicher zur Verfügung.
- Schnittstellen:
 - drei 10/100/1000 MBit Ethernet Schnittstellen

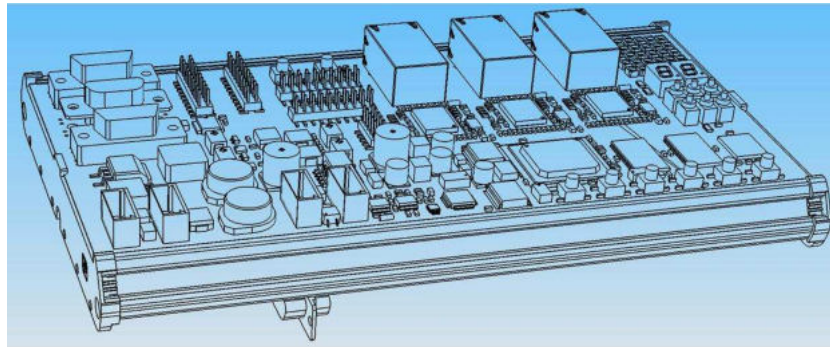


Abbildung 27: Entwicklungsboard (3D-Sicht)

- Serielle Schnittstellen nach RS232/RS485
- 6x5 LED-Matrix
- 48 GPIOs
- 2 Siebensegment-Anzeigen
- 6 Schalter
- 6 Taster

In der Abbildung 27 ist eine 3D-Ansicht des Boards, das während der Entwicklung entstanden ist, dargestellt. Die Abbildung 28 zeigt die Entwicklungsumgebung in der der IP-Core des ESANA-Projekts entwickelt wurde.

4.4 Demonstrator

Damit die Ergebnisse der ESANA-Optimierungen der Fachöffentlichkeit einfacher zugänglich gemacht werden kann, wurde in dem Messebau von Phoenix Contact auf Basis der entwickelten Development-Boards ein Demonstrator gebaut. Der Demonstrator basiert auf einem Beispielszenario mit 1 Controller und 7 Knoten in einer Linientopologie mit einem Abzweig. Aufgrund der Optimierungen erreicht der Demonstrator ohne Berücksichtigung einer Phase für die Übertragung von NRT-Daten eine minimale Update-Zeit von $20 \mu s$. Berücksichtigt man zusätzlich eine NRT Phase erhöht sich die minimale Updatezeit auf $61,25 \mu s$. Es steht in diesem Fall neben der schnellen Prozessdatenübertragung noch eine Bandbreite von 67 MBit/s für NRT-Daten

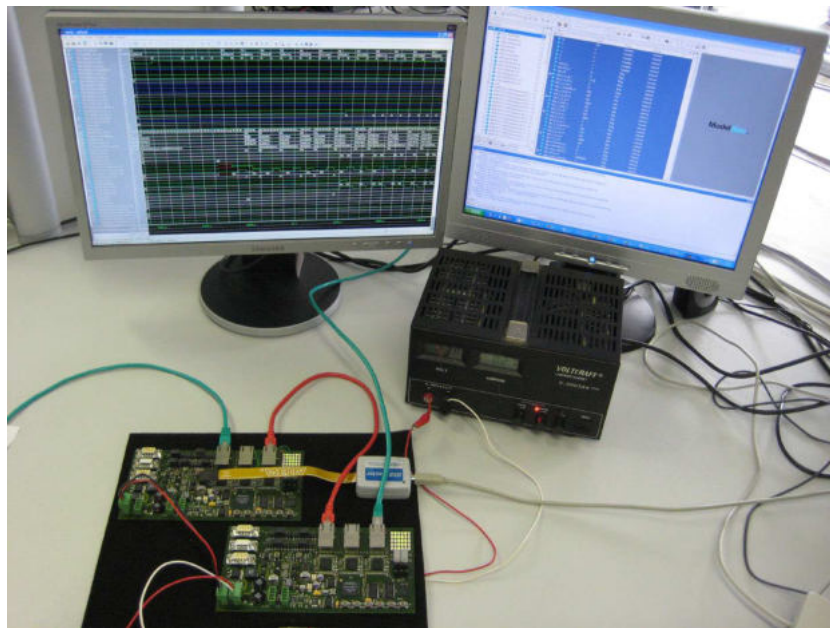


Abbildung 28: Entwicklungsumgebung im Einsatz

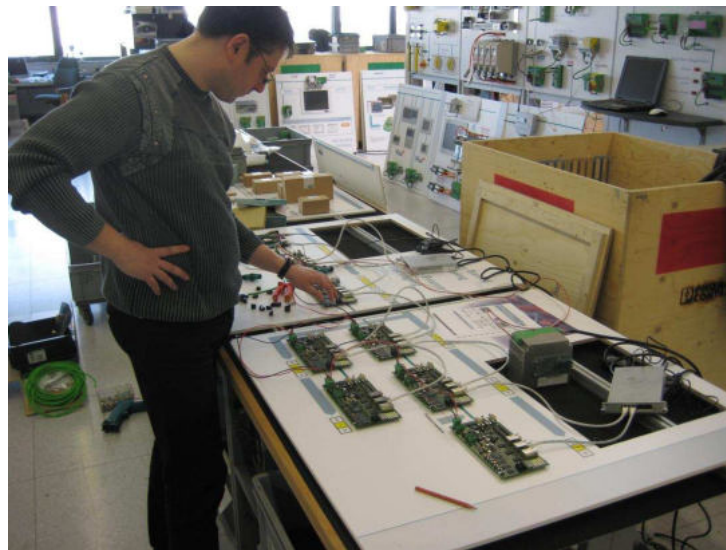


Abbildung 29: Aufbau des ESANA-Demonstrators in der Messebauabteilung von Phoenix Contact (1)

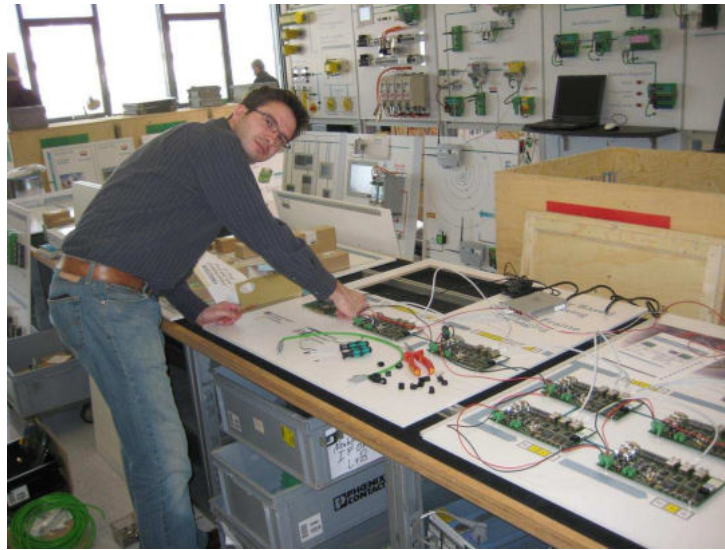


Abbildung 30: Aufbau des ESANA-Demonstrators in der Messebauabteilung von Phoenix Contact (2)

zur Verfügung. In dem Demonstrator wird der NRT-Datenstrom von einer Webcam generiert.

4.5 Verwertbarkeit der Ergebnisse

Die Ergebnisse sind projektbegleitend durch Veröffentlichungen (s.a. Kapitel 6), sowohl in der nationalen und internationalen wissenschaftlichen Gemeinschaft, als auch in den Nutzerorganisationen, diskutiert und verbreitet worden. Die erarbeiteten Optimierungsansätze werden derzeit in die Standardisierung von Profinet über die PNO überführt. Das zeigt die hohe Relevanz der erzielten Ergebnisse. Damit stehen die Verfahren allen Interessierten zur Verfügung und können zum Beispiel in zukünftigen Profinet-ASICs berücksichtigt werden.

5 Technischer Fortschritt bei anderen Stellen

Während der Projektlaufzeit sind keine weiteren Arbeiten bekannt geworden, die den gleichen Fokus wie das ESANA-Projekt hatten.

6 Veröffentlichungen

Im Kontext des ESANA-Projektes wurden folgende nationale und internationale Veröffentlichungen durchgeführt:

- Schumacher, Markus; Jasperneite, Jürgen; Weber, Karl: Optimierte Datenübertragung für Echtzeit-Ethernet. In: VDI Kongress AUTOMATION 2009 Baden-Baden, Jun 2009 (accepted)
- Schumacher, Markus: Beste Zukunftsaussichten - Potenzialanalyse verschiedener Echtzeit-Ethernet-Protokolle (Vortrag). In: 13. Industrial Communication Congress Bad Pyrmont, Sep 2008
- Imtiaz, Jahanzaib; Jasperneite, Jürgen; Weber, Karl; Lessmann, Gunnar; Goetz, Franz-Josef: A Novel Method for Auto Configuration of Realtime Ethernet Networks. In: 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA2008) Hamburg, Germany, Sep 2008
- Schumacher, Markus; Jasperneite, Jürgen; Weber, Karl: A new Approach for Increasing the Performance of the Industrial Ethernet System PROFINET. In: 7th IEEE International Workshop on Factory Communication Systems (WFCS 2008) S.: 159 - 167, Dresden, Germany, May 2008
- Jasperneite, Jürgen; Lessmann, Gunnar; Weber, Karl: Mit Profinet in die Zukunft. In: Profibus & Profinet Journal S.: 8- 9, Profibus Nutzerorganisation e.V. , Karlsruhe, Deutschland, Apr 2008
- Jasperneite, Jürgen: Echtzeit-Ethernet - hohe Geschwindigkeit mit Profinet. In: etz Elektrotechnik + Automation(12) S.: 12-16, Dec 2007
- Jasperneite, Jürgen; Crocoll, Nora: Leitfaden erleichtert den Einstieg. In: atp -Automatisierungstechnische Praxis (10) S.: 26-28 (non-review), Oct 2007
- Jasperneite, Jürgen; Schumacher, Markus; Weber, Karl: Limits of Increasing the Performance of Industrial Ethernet Protocols. In: 12th IEEE Conference on Emerging Technologies and Factory Automation S.: 17 - 27, Patras, Greece, Sep 2007



Abbildung 31: Institutsleiter Prof. Jasperneite stellt in München vor zahlreichen Fachjournalisten die Ergebnisse der Potentialanalyse vor.

Die Ergebnisse der Potentialanalyse im Arbeitspaket 1 waren so interessant, dass die Profibus-Nutzerorganisation (PNO), ein Interessensverbund aus über 1400 Unternehmen der Automatisierungstechnik, am 07. November 2007 einen exklusiven **Presseworkshop** zu diesem Thema in München veranstaltete, der ein grosses Medienecho hervorrief. Unter anderem berichteten die VDI-Nachrichten in ihrer Ausgabe 47/2007 über die ersten Ergebnisse des Projektes ESANA.

Weitere Informationen zum Presseworkshop und die Berichterstattung finden Sie unter den folgenden Links:

<http://www.profibus.com/rpa/germany/events/press/article/02562/>

http://www.hs-owl.de/init/no_cache/aktuelles/news/

[news-einzelansicht/news/init-im-presseworkshop-der-pno/634.html](http://www.hs-owl.de/init/no_cache/aktuelles/news/news-einzelansicht/news/init-im-presseworkshop-der-pno/634.html)

6.1 Messeauftritte

Ein weiteres wichtiges Element für die Verbreitung von Projektergebnissen sind Messeauftritte.

Auf der Hannovermesse 2008 konnten die erarbeiteten Optimierungsansätze



Abbildung 32: Dipl.-Ing. Markus Schumacher präsentiert ESANA-Ergebnisse auf der Hannovermesse 2008 auf dem Stand der PNO.

aus dem Arbeitspaket 2 in Form einer Machbarkeitsstudie der internationalen Fachöffentlichkeit vorgestellt worden. Unsere Projektpartner Siemens und Phoenix Contact unterstützten das inIT bei dem Messeaufbau und der Übernahme der Standmiete auf dem Gemeinschaftstand der Profibus-Nutzerorganisation (PNO).

Auf dem ICC (Industrial Communication Congress) 2008 wurde der Demonstrator ebenfalls gezeigt. Auf der SPS/IPC/Drives 2008 in Nürnberg wurden die Kernaussagen der Optimierungsverfahren nochmals auf dem Stand der PNO dargestellt. Nach der Hannovermesse wurde der ESANA-Demonstrator erweitert und fest im Institut Industrial IT installiert. Seitdem ist er fester Bestandteil bei Besuchergruppen des inITs.

Im Jahr 2009 wird der ESANA-Demonstrator im Rahmen der '7 STATIONEN in OWL' ausgestellt. Die 7 STATIONEN sind ein Projekt der Initiative Innovation und Wissen der OstWestfalenLippe Marketing GmbH. Ziel der von Wirtschaft, Wissenschaft, Verwaltung und Politik getragenen Initiative ist es, die Innovationsfähigkeit der Unternehmen und damit die Innovationskraft der Region zu stärken.



Abbildung 33: NRW-Innovationsminister Prof. Andreas Pinkwart läßt sich am 12. September 2008 den ESANA-Demonstrator erläutern.

7 Abschließende Bemerkungen

Das **übergeordnete Ziel** des Forschungsvorhabens bestand darin, die heute existierende technologische und wirtschaftliche Hemmschwelle für die Verbreitung von Echtzeit-Ethernet am Beispiel von Profinet bis hinunter zur Sensor/Aktorvernetzung zu überwinden. Dieses Ziel ist im Rahmen des Projektes für den Aspekt der Performance in vollem Umfang erfüllt worden. Hierdurch ergibt sich künftig für Profinet ein deutlich verbreitertes Anwendungsgebiet.

Aufgrund der Projektdauer von nur 18 Monaten konnten die beiden im Projektantrag formulierten Leitfragen jedoch nicht in gleicher Intensität bearbeitet werden. In Abstimmung mit den Projektpartnern wurde der Schwerpunkt auf die Performance gelegt, was sich retrospektiv betrachtet als überaus erfolgreich darstellt, da die so erzielten Projektergebnisse nun unmittelbar in die Standardisierung von Profinet einfließen konnten. Die weitere Bearbeitung der Einfachheit wurde aus dem ESANA-Projekt ausgekoppelt und in einem Projekt der industriellen Auftragsforschung fortgeführt.

Das **wissenschaftliche Ziel** bestand in der Entwicklung von Konzepten für die Echtzeitkommunikation auf Basis von IuK-Technologien, sowie deren mikroelektronischer Umsetzung. Dieses Ziel ist in vollem Umfang erreicht worden. Damit hat das Projekt wesentlich dazu beigetragen, die Profinet-Technologie weiterzuentwickeln und das Institut im Forschungsmarkt zu positionieren.

Obwohl ein entsprechendes LOI der TU Berlin vorlag, hat sich gezeigt, dass eine Projektdauer von 18 Monaten deutlich zu kurz ist, um einen wissenschaftlichen Mitarbeiter zur Promotion zu führen. Daher müssen Folgeprojekte akquiriert werden, um diesen wissenschaftlichen Qualifizierungsschritt für den beteiligten Mitarbeiter zu ermöglichen. Diese Erkenntnis sollte in den Begutachtungskriterien des Förderprogramms künftig Berücksichtigung finden.

An dieser Stelle möchten wir uns bei allen Projektpartnern, der AiF und dem BMBF für ihre Unterstützung bedanken.

Literatur

- [1] IEC, Geneva. *Digital data communication for measurement and control – Fieldbus for use in industrial control systems*, 2001. IEC 61158 Ed.3, Type 3 (PROFIBUS).
- [2] IEC, Geneva. *Digital data communication for measurement and control – Fieldbus for use in industrial control systems*, 2001. IEC 61158 Ed.3, Type 8 (INTERBUS).
- [3] Open DeviceNet Vendor Association. <http://www.odva.org>.
- [4] Sercos. *SERCOS Users Web Site*:. www.sercos.de, visited on October 2008.
- [5] Safetybus P. *Safety Network International Users Web Site*:. www.safetybus.com, visited on October 2008.
- [6] S. Rüping, E. Vonnahme, and J. Jasperneite. Analysis of Switched Ethernet Networks with different Topologies used in Automation Systems. In D. Diedrich, P. Neumann, and H. Schweinzer, editors, *Field Bus Technology*, pages 351–358. Springer-Verlag, 1999.
- [7] VDI. *Richtlinie 3687: Auswahl von Feldbussystemen durch Bewertung ihrer Leistungseigenschaften für verschiedene Anwendungsbereiche*. Beuth-Verlag, Berlin, Juni 1997. VDI/VDE-Richtlinie 3687 (Entwurf).
- [8] Juergen Jasperneite. Echtzeit-Ethernet im Überblick. *ATP - Automatisierungstechnische Praxis*, 3:S. 29 – 34, March 2005.
- [9] J.-D. Decotignie. Ethernet based real-time and industrial communications. *Proceedings of the IEEE*, 93 issue 6:1102–17, 2005.
- [10] M. Felser and T. Sauter. Standardization of Industrial Ethernet the Next Battlefield? In *5th IEEE International Workshop on Factory Communication Systems (WFCS'2004)*, Vienna, Sept. 2004.
- [11] M. Felser. Real-time ethernet - industry prospective. *Proceedings of the IEEE*, 93 issue 6:1118ff, 2005.

- [12] J. Schwager. Realtime Ethernet in Industrial Automation. Website. Available online at <http://www.real-time-ethernet.de>; visited on October 2008.
- [13] IEEE. *Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*. IEEE, New York, 2000. ANSI/IEEE Std 802.3-2000 Edition (ISO/IEC 8802-3:2000(E)).
- [14] ModBus. *MODBUS IDA Users Web Site*. www.modbus.org, visited on October 2008.
- [15] FF. *Fieldbus Foundation*. www.fieldbus.org, 2007.
- [16] IEEE. *Virtual Bridged Local Area Networks*. IEEE, New York, 1998. ANSI/IEEE Std 802.1Q-1998.
- [17] IEEE. *Media Access Control (MAC) Bridges*. IEEE, New York, 1998. ANSI/IEEE Std 802.1D-1998.
- [18] Kym Watson and Jürgen Jasperneite. Determining end-to-end delays using network calculus. In *5th IFAC International Conference on Fieldbus Systems and their Applications (FeT2003)*, pages 255–260, Aveiro, Portugal, Jul 2003.
- [19] IEC. *IEC 61158-5-10, Digital data communications for measurement and control - Fieldbus for use in industrial control systems - Part 5-10: Application Layer service definition - Type 10 elements*, 2006.
- [20] IEC. *IEC 61158-6-10, Digital data communications for measurement and control - Fieldbus for use in industrial control systems - Part 6-10: Application Layer protocol specification - Type 10 elements*, 2006.
- [21] Ethernet/IP. *Ethernet Industrial Protocol (EtherNet/IP)*. www.ethernet-ip.org, visited on October 2008.
- [22] Ethernet Powerlink. *Ethernet Powerlink Standardization Group*. www.ethernet-powerlink.org, 2007.
- [23] EtherCAT. *Ethernet Technology Group (ETG)*. www.ethercat.org, visited on October 2008.

- [24] Christoph Rotter. *Entwicklung eines RFID-Systems mit PROFINET IO-Schnittstelle und Erprobung an einer robotergestützten Fließbandapplikation, Diplomarbeit.* Institut Industrial IT, Hochschule Ostwestfalen-Lippe, 2006.
- [25] Hilscher. *Hilscher GmbH Web Site:.* de.hilscher.com/, visited on October 2008.