

# **Beitrag zur Verhaltensanalyse und Synchronisation von steuerungstechnischen Prozessen durch verteilte echtzeitfähige Kommunikationssysteme**

**Von der Fakultät für Maschinenbau, Verfahrens- und Energie-  
technik der Technischen Universität Bergakademie Freiberg**

**genehmigte**

## **DISSERTATION**

**zur Erlangung des akademischen Grades**

**Doktor-Ingenieur (Dr.-Ing.)**

**vorgelegt**

<b>von:</b>	Dipl.-Ing. Gert Anders	
<b>geboren:</b>	am 29.07.1966 in Elsterwerda	
<b>Gutachter:</b>	Prof. Dr.-Ing. habil. Peter Löber	TU Bergakademie Freiberg
	Dr.-Ing. Dietmar Sobottka	Siemens AG Chemnitz
	Prof. Dr.-Ing. habil. Karl Heß	TU Chemnitz
	Prof. Dr.-Ing. Dietmar Römer	HTW Mittweida
<b>Verleihung:</b>	am 08.12.2006	

# **Bibliographische Beschreibung**

Anders, Gert:

Beitrag zur Verhaltensanalyse und Synchronisation von steuerungstechnischen Prozessen durch verteilte echtzeitfähige Kommunikationssysteme

Eingereicht von Gert Anders 2006

Freiberg, Technische Universität Bergakademie, Fakultät für Maschinenbau, Verfahrens- und Energietechnik, Dissertation

154 Seiten, 57 Abbildungen, 13 Tabellen

## **Schlagwörter:**

Feldbussysteme, Industrial Ethernet, industrielle Kommunikation, Architekturkonzepte, Client-Server, verteilte echtzeitfähige Steuerungssysteme, Webservices, CSMA/CA/CD, PROFIBUS, Antriebssysteme, Web-basierte Steuerungstechnik

## **Referat:**

Aufbauend auf dem voranschreitenden Übergang zentralistischer Steuerungskonzepte über die Dezentralisierung hin zum verteilten System soll ein echtzeitfähiges Steuerungskonzept für die Antriebssynchronisation zur Patientenbewegung als lokal abgeschlossenes System erarbeitet werden. Dabei sollen Grundlagen echtzeitfähiger Steuerungstechnik, verteilter Systeme, der Prozesssynchronisation sowie entsprechende Kommunikationssysteme vorgestellt, bestehende Lösungen diskutiert und aus den Erkenntnissen entsprechende Konzeptvorschläge für die Lösung der Aufgabenstellung gemacht werden. Im zweiten Teil dieser Arbeit, soll aus den gewonnen Erkenntnissen und dem voranschreitenden Erfordernis der weltweiten Vernetzung technischer Systeme ein Konzept für die Integration und den echtzeitfähigen Zugriff auf dezentrale Peripheriekomponenten in das Internet erarbeitet werden. Für beide Teilaufgaben sollen Lösungsschritte aufgezeigt und evaluiert werden. Eine kritische Betrachtung der vorgestellten Konzepte erfolgt auf Basis bereits erfolgter industrieller Anwendung sowie in der Vorstellung einer zum Patent angemeldeten Lösung für den webbasierten Zugriff auf dezentral angeordnete Steuerungskomponenten.

## Vorwort

Die vorliegende Arbeit entstand im Rahmen meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Automatisierungstechnik der Technischen Universität Bergakademie Freiberg, sowie aus Ergebnissen meiner Tätigkeit als Entwicklungsingenieur für ein mittelständiges Unternehmen auf dem Gebiet der mechatronischen Medizintechnik.

Bedanken möchte ich mich an dieser Stelle für die konstruktiven inhaltlichen und fachlichen Vorschläge von Herrn Professor Dr.-Ing. habil. Karl Hess von der Technischen Universität Chemnitz und Herrn Professor Dr.-Ing. Dietmar Römer von der Ingenieurhochschule Mittweida.

Mein ganz besonderer Dank gilt Herrn Professor Dr.-Ing. habil. Peter Löber von der Technischen Universität Bergakademie Freiberg, als einen immer verständnisvollen Vorgesetzten und betreuenden Erstgutachter für die stets wertvollen Hinweise und Vorschläge für das Gelingen dieser Arbeit.

Ganz herzlich möchte ich mich für die gute Zusammenarbeit des Institutes für Automatisierungstechnik mit der Siemens AG, Geschäftsbereich A&D am Standort Chemnitz bedanken. Die freundliche Unterstützung von Herrn Dr.-Ing. Dietmar Sobottka, Herrn Dr.rer.nat. Bernhard Weißbach, Herrn Dr.-Ing. Hagen Böhme sowie der Entwicklungsabteilung für die dezentrale Peripherie war für die Untersuchungen auf dem Gebiet verteilter echtzeitfähiger Automatisierungssysteme von unschätzbarem Wert. Die Untersuchungen des Verhaltens echtzeitfähiger dezentraler Peripheriekomponenten bei der Integration in das definitiv nicht echtzeitfähige Internet wären ohne die Forschungsprojekte „Internetmechanismen in verteilten Systemen der Automatisierungstechnik“ mit der Siemens AG nicht zustande gekommen. Ganz besonders möchte ich mich bei allen beteiligten Kollegen, der Siemens AG und der Bergakademie Freiberg für die positiv zustande gekommene Anmeldung der Patentschrift „Verfahren zur Nutzung von Webservices in Systemen der Automatisierungstechnik“ im Rahmen der Forschungstätigkeiten bedanken.

Ich möchte mich weiterhin für die freundliche Unterstützung der TRUMPF-Medizintechnik GmbH Saalfeld für die Nutzung von Forschungs- und Entwicklungsergebnissen auf dem Gebiet der medizinischen Mechatronik, vertreten durch Herrn Dipl.-Phys. Falk Georgi und Dipl.-Ing. Christian Streitberger bedanken.

Ein weiterer Dank gilt Herrn Dr.-Ing. Swen Schmeißer für die gute Zusammenarbeit auf dem Gebiet der webbasierten Ausbildung an industrieller Steuerungstechnik, sowie den Herrn M.Sc. Matthias Strobelt und Dipl.-Ing. Daniel Brechter für ihre tatkräftige Mitarbeit bei der Implementierung von Softwaremodulen zum Test des Verhaltens und der Diagnose echtzeitfähiger dezentraler Peripheriekomponenten im Internet. Abschließend bedanke ich mich für die die vielen kreativen Diskussionen mit Herrn Dr.-Ing. Michael Martin auf dem Gebiet mobiler verteilter Steuerungstechnik im Rahmen seiner Dissertation.

# 0 Inhaltsverzeichnis

1	Motivation und Aufgabenstellung.....	1
2	Grundlagen für die Realisierung verteilter echtzeitfähiger Steuerungssysteme .....	3
2.1	Begriffsdefinitionen .....	3
2.2	Wesentliche Anforderungen an verteilte industrielle Steuerungssysteme .....	5
2.2.1	Verfügbarkeit und Zuverlässigkeit.....	5
2.2.2	Flexibilität und Offenheit .....	5
2.2.3	Leistungsfähigkeit .....	6
2.2.4	Schlussfolgerungen .....	6
2.3	Grundlegende Kommunikationsstrukturen in der industriellen Automation.....	7
2.3.1	Client-Server-Architektur.....	7
2.3.2	Peer-to-Peer-Architektur .....	8
2.3.3	Producer-Consumer-Konzept.....	9
2.3.4	Publisher/Subscriber - Konzept.....	10
2.3.5	Weitere automatisierungstechnische Architekturen und Zugriffsprinzipien ...	11
2.3.5.1	Ereignisgesteuerte Architekturen .....	11
2.3.5.2	Zeitgesteuerte Architekturen .....	11
2.4	Echtzeitanforderungen in verteilten Systemen der Automatisierungstechnik .....	12
2.4.1	Begriffe.....	12
2.4.2	Fallunterscheidungen und Auswirkungen auf das Verhalten verteilter Systeme. .....	13
2.4.3	Schlussfolgerung und Bewertung der Echtzeitanforderungen .....	18
3	Kommunikationssysteme für den Aufbau verteilter Systeme.....	19
3.1	Grundlegender Begriff .....	19
3.2	Einteilung der Kommunikationssysteme .....	19
3.2.1	Einteilung nach der Übertragungstechnik .....	19
3.2.2	Einteilung nach den Medienzugangsverfahren .....	20
3.3	Ausgewählte parallele Kommunikationssysteme.....	21
3.4	Ausgewählte serielle industrielle Kommunikationssysteme.....	21
3.4.1	Ausgewählte Kommunikationssysteme aus dem Feldbereich .....	22
3.4.1.1	PROFIBUS.....	22
3.4.1.2	CAN .....	23
3.4.2	Ausgewählte Kommunikationssysteme der Officekommunikation.....	25
3.4.3	Ausgewählte Kommunikationssysteme der Gerätetechnik.....	25
3.5	Ausgewählte drahtlose Kommunikationssysteme.....	27
3.5.1	Funkbasierte Kommunikationssysteme.....	27
3.5.2	Optische Kommunikationssysteme .....	28
3.6	Industrial Ethernet.....	29
3.6.1	Einführung.....	29
3.6.2	Aufgabenstellung .....	31
3.6.2.1	Bestehende Ausgangssituation für Industrial Ethernet .....	32
3.6.2.2	Gestellte Anforderungen für Industrial Ethernet.....	32
3.6.2.3	Schlussfolgerungen .....	33
3.6.3	Wichtige Protokolle des Ethernet.....	34
3.6.3.1	Address Resolution Protocol (ARP) .....	34
3.6.3.2	Internet Protocol (IP).....	34
3.6.3.3	Transmission Control Protocol (TCP).....	35
3.6.3.4	User Datagram Protocol (UDP) .....	35

3.6.4	Probleme und Verfahren für die Realisierung deterministisches Ethernet .....	36
3.6.4.1	Methode der Lastbegrenzung .....	36
3.6.4.2	Zeitschlitzverfahren.....	37
3.6.4.3	Zeitsynchronisationsverfahren .....	37
3.6.4.4	Bewertung von Zeitsynchronisationsprotokollen.....	39
3.6.4.5	Codierungsverfahren von Fast Ethernet.....	40
3.6.5	Grundlegende Architekturkonzepte für echtzeitfähiges Ethernet .....	41
3.7	Technische Realisierungen von Industrial Ethernet.....	44
3.7.1	Einführung.....	44
3.7.2	Standard-Konzepte .....	44
3.7.2.1	PROFINET .....	44
3.7.2.2	EtherCAT .....	46
3.7.2.3	Ethernet/IP mit CIPSync .....	47
3.7.2.4	Powerlink .....	48
3.7.2.5	Modbus-TCP-IDA.....	49
3.7.2.6	SERCOS-III .....	50
3.7.2.7	IAONA .....	51
3.7.3	Weitere Konzepte .....	51
3.8	Erkenntnisse für den Entwurf verteilter echtzeitfähiger Systeme .....	53
4	Modellierung von verteilten Kommunikationssystemen .....	54
4.1	Grundlagen der Modellbildung .....	54
4.1.1	Eigenschaften von Modellen und deren Klassen .....	55
4.1.2	Ziele, Vorteile und Probleme der Modellbildung .....	56
4.1.3	Simulationsstudie .....	57
4.2	Modellierungsmittel und Methoden .....	58
4.2.1	Modellierung durch Petri-Netze.....	59
4.2.2	Graphenbasierte Modellierung.....	60
4.2.3	Algorithmische Modellierung .....	62
4.2.4	Weitere Modellierungsarten .....	63
4.3	Modellvalidierung .....	63
4.4	Problemstellungen bei der Modellierung feldbusbasierten Netzstrukturen .....	64
4.4.1	Grundaufgabe .....	64
4.4.2	Unicast-Kommunikationsbeziehungen ohne stochastischen Einfluss .....	65
4.4.3	Unicast-Kommunikationsbeziehungen mit stochastischem Einfluss.....	66
4.4.4	Broad- und Multicast-Kommunikationsbeziehungen .....	67
4.4.5	Multipeer-Kommunikationsbeziehungen.....	68
4.4.6	Concast-Kommunikationsbeziehungen.....	68
4.4.7	Abbildung von Graphen in Matrizenform.....	69
4.5	Stabilitätsbetrachtung von verteilten Kommunikationssystemen .....	72
4.5.1	Der Stabilitätsfall.....	73
4.5.2	Der Grenzlastfall .....	74
4.5.3	Der Überlastfall.....	74
4.6	Verteilte Kommunikation am Beispiel von PROFIBUS-DP .....	74
4.7	Probleme bei der Modellierung ethernetbasierter Kommunikationssysteme .....	76
4.8	Zusammenfassung und Bewertung .....	77
5	Synchronisation von technischen Prozessen in MIKRO-LAN's .....	78
5.1	Aufgabenstellung .....	78
5.2	Entwicklungsstand im medizinischen Gerätebau.....	78
5.3	Ausgangssituation und Lösungsansätze .....	79
5.3.1	Konventionelles zentralisiert organisiertes Steuerungskonzept.....	82

5.3.2	Kommunikationsorientierte verteilte organisierte Steuerungskonzepte .....	83
5.3.3	Schlussfolgerungen und Festlegung des Steuerungskonzeptes.....	83
5.4	Algorithmische Umsetzungen (Auszugsweise) .....	85
5.5	Integration des Teilsystems in ein Gesamtsystem.....	87
5.6	Projekthistorie der Entwicklung des Teilsystems .....	88
5.7	Zusammenfassung .....	89
6	Integration des Steuerungssystems SIMATC-S7 in das WAN .....	90
6.1	Motivation und Aufgabenstellung.....	90
6.2	Ausgangssituation und Stand der Informationstechnik.....	91
6.2.1	Auswirkungen der Globalisierung der Kommunikation auf die Industrie .....	92
6.2.2	Problemstellungen von Internettechnologien in der Automatisierungstechnik	92
6.2.3	Projektspezifische Problemstellungen.....	93
6.2.4	Dezentrale Peripherie als wesentlicher Bestandteil verteilter Automatisierungssysteme .....	94
6.2.5	Konzepte für den internetbasierten Zugriff auf die dezentrale Peripherie .....	95
6.2.5.1	Webbasierter Zugriff über die SPS und den IT- Kommunikationsprozessor . .....	95
6.2.5.2	Webbasierter Zugriff durch Klasse-2-Master mit integriertem Web-Server	97
6.2.5.3	Bewertung beider webbasierter Zugriffskonzepte .....	99
6.3	Webservices in verteilten Automatisierungssystemen.....	99
6.3.1	Grundlegende Konzepte und Definitionen von Web-Services .....	99
6.3.2	Vorstellung eines Konzeptes für den Einsatz von Web-Services in echtzeitfähigen verteilten Steuerungssystemen .....	100
6.3.3	Prinzipieller Ablauf des Web-Services Firmwareupdate .....	103
6.4	Versuchsanlagen zur Analyse, Realisierung und Evaluierung.....	104
6.5	Leistungsbewertung des prototypisch realisierten Konzepts .....	107
6.6	Zusammenfassung .....	108
7	Abschließende Betrachtungen und weiterführende Arbeiten.....	110
8	Thesen .....	112
9	Literaturverzeichnis.....	114
10	Abkürzungsverzeichnis .....	127
11	Abbildungsverzeichnis .....	130
12	Tabellenverzeichnis.....	132
13	Eidesstattliche Selbstständigkeitserklärung .....	133
14	Veröffentlichungen – Forschungsberichte .....	134
15	Anhang .....	137

# 1 Motivation und Aufgabenstellung

In der modernen Industrieautomation ist der Übergang vom zentralistisch aufgebauten Steuerungssystem zu dezentral realisierten Automatisierungsstrukturen im vergangenen Jahrzehnt vollzogen worden. Leistungsfähige Hardware, intelligente Algorithmen in Embedded-Systemen sowie leistungsfähigere Diagnosekonzepte beschleunigen den Trend, immer komplexere verteilte Systeme in automatisierungstechnischen Anlagen zu realisieren. Dieser Entwicklungstrend lässt sich als hochdynamischer Prozess einstufen und kann als unumkehrbar betrachtet werden.

Die Konzeption des Aufbaus verteilter Steuerungssysteme setzt sich aber nicht nur im prozessnahen Bereich der Automation durch, sondern kann auch für die Modellierung und Realisierung von Steuerungssystemen in der Gerätetechnik eingesetzt werden. Damit kommen die Vorteile der Konzeption (Flexibilität, Kosten, Modularität u.s.w.) auch in diesem Aufgabebereich zum Tragen.

In der modernen Industriekommunikation steht, unabhängig vom gewählten Realisierungskonzept, die zentrale Frage nach der echtzeitfähigen Bedienung des zu steuernden Prozesses im Vordergrund. Bei der Realisierung von regelungstechnischen Algorithmen für die Steuerung industrieller Abläufe, bei denen ein zeitlich präzises abgestimmtes Kommunikationsmanagement von grundlegender Bedeutung ist, besteht die Forderung nach determiniert harter Echtzeit.

Die Kommunikation über serielle Informationskanäle, mit den oben formulierten Anforderungen an die Echtzeitfähigkeit, stellt eine wesentliche Grundlage für die Realisierung verteilter Automatisierungssysteme dar. Mit bestehenden Feldbussystemen kann bisher grundsätzlich der Forderung nach Echtzeitfähigkeit bei der Steuerung von industriellen Prozessen entsprochen werden. Mit ausgewählten seriellen Kommunikationssystemen, welche über entsprechende Zugriffsprinzipien und Protokollspezifikationen verfügen, können steuerungstechnische Aufgaben mit determiniert harter Echtzeitforderung realisiert werden. Solche Forderungen sind bei jeder Art der Synchronisation von technischen Prozessen Grundvoraussetzung, ebenso bei jeglicher Art der Realisierung von regelungstechnischen Algorithmen über serielle Kommunikationssysteme. Das trifft insbesondere auf die Problematik der Synchronisation von Bewegungsabläufen in der modernen Antriebstechnik zu.

Die Modellbildung und Untersuchung des Verhaltens von, auf serieller Kommunikation basierenden, echtzeitfähigen Steuerungssystemen, wurden zu einem Teil bei der Entwicklung und Realisierung eines Steuerungskonzeptes auf dem Gebiet der medizinischen Gerätetechnik realisiert. Grundlegende Zielstellung bei der Konzeption des verteilt aufgebauten Steuerungssystems bestand in der Reduzierung bzw. Ablösung von mechanischen, hydraulischen oder pneumatischen Systemen zur Synchronisation von Bewegungsabläufen im Bereich der Operationstischkonstruktion. Das vorgestellte Modell ist durchgängig nach dem Konzept eines verteilten echtzeitfähigen Antriebssystems aufgebaut und modular konzipiert.

Zielstellung dieser Arbeit ist es, durch Entwicklung entsprechender Regelalgorithmen den Einsatz von „offengelegten“<sup>1</sup> seriellen Kommunikationssystemen für die Steuerung von dynamischen Prozessen in räumlich begrenzt ausgedehnten Systemen zu ermöglichen.

Das Konzept basiert dabei auf der präzisen Einhaltung des Zeitmanagements in modular erweiterbaren Kommunikationssystemen. Die Untersuchungen und Entwicklungen entsprechender Algorithmen werden dabei am Beispiel der Bewegungssynchronisation durch Koordinierung von drei Einzeltriebssystemen über IIC-Kommunikation beim Patientenbewegungssystem „Jupiter“ durchgeführt und realisiert.

Im zweiten Teil der Arbeit sollen basierend auf diesen gewonnenen Erkenntnissen echtzeitfähiger Steuerungssysteme Konzepte zur Integration echtzeitfähiger Steuerungssysteme der SIMATIC S7 Gerätefamilie in das nichtechtzeitfähige verteilte System „Internet“ untersucht und entsprechende Lösungsvorschläge vorgestellt werden. Dabei stellt die Bewertung bestehender Lösungen für den Web-Zugriff sowie das Verhalten der dezentralen Peripheriekomponenten im Vordergrund. Für die Evaluierung der Ergebnisse soll prototypisch die Firmware einer ET200 im laufenden Anlagenbetrieb durch Nutzung von Internettechnologien auf einen aktuelleren Stand gebracht werden.

Im Rahmen der Forschungsk Kooperation zwischen der Siemens AG und dem Institut für Automatisierungstechnik auf dem Gebiet der IT-Integration dezentraler Peripheriekomponenten wurden unter anderen die Schwerpunkte auf folgende Themen gelegt:

- Studie zum Thema „Offene verteilte Automatisierungssysteme“
- Analyse und Bewertung des Verhaltens echtzeitfähiger dezentraler Peripheriekomponenten der Geräteklasse ET200 bei der Integration von SIMATIC S7 basierender Steuerungssysteme in das Internet.
- Analyse und Modellierung möglicher Konzepte für den Zugriff auf echtzeitfähige dezentrale Peripherie in verteilten Systemen der Automatisierungstechnik über das Internet am Beispiel der Diagnose und des Firmwareupdates einer ET200S.
- Dynamische Adressvergabe dezentraler Peripheriekomponenten in Profibussystemen
- CAN - PROFIBUS - Gateway

---

<sup>1</sup> Unter „offengelegt“ sollen Systeme verstanden werden, bei denen die Spezifikationen für Hard- und Software für „jedermann“ frei zugänglich sind.



## 2 Grundlagen für die Realisierung verteilter echtzeitfähiger Steuerungssysteme

### 2.1 Begriffsdefinitionen

Zunächst soll die Bedeutung des Begriffs *echtzeitfähiges verteiltes* Steuerungssystem, im Zusammenhang mit dieser Arbeit, auf dem Gebiet der Automatisierungstechnik festgelegt werden. Aus den Wissenschaftsgebieten Informatik und Informationstechnik existieren verschiedene Definitionen für die Begriffe

*Echtzeitsystem, verteiltes System und offenes verteiltes System:*

#### **Echtzeitsystem:**

Es sind verschiedene Versuche unternommen worden den Begriff Echtzeitsystem zu definieren. Bei der Bestimmung des Begriffs wird dabei vorrangig auf einzelne Computer mit echtzeitfähiger Verarbeitung entsprechender Algorithmen auf diesem lokal abgeschlossenen Systemen eingegangen. Die folgende Definition ist in der DIN 44300 festgelegt und kann als Standard für die Beschreibung von Echtzeitsystemen angesehen werden.

Definition Echtzeitsystem nach DIN 44300

*Ein Betrieb eines Rechnersystems, bei dem Programme zur Verarbeitung anfallender Daten ständig betriebsbereit sind, derart, dass die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind. Die Daten können je nach Anwendungsfall nach einer zeitlich zufälligen Verteilung oder zu vorherbestimmten Zeitpunkten anfallen.<sup>2</sup>*

#### **Verteiltes System:**

Analog zum Begriff Echtzeitsystem sind mehrere Ansätze der Definition „Verteiltes System“ durchgeführt worden. Die Definition von Andrew S. Tanenbaum von 1996 kann als Grundlage für die Bestimmung des Begriffs verteilte Systeme, auf dem Wissenschaftsgebiet der technischen Informationsverarbeitung, angesehen werden.

*Ein verteiltes System ist eine Kollektion unabhängiger Computer, die den Benutzern als Einzelcomputer erscheinen [2.03].*

Die Definition von A.S.Tanenbaum beschreibt den Sachverhalt „Verteiltes System“ in einer allgemeingültigen abstrakten Weise, ohne sich in einzelnen Details festzulegen. So ist beispielsweise weder eine Architektur noch ein zeitliches Verhalten eines Systems festgeschrieben. Damit kann diese Definition als Basis für die Definitionen spezieller verteilter Systeme

---

<sup>2</sup> DIN 44300

z.B. auch verteilter auf Kommunikation basierender industrielle Steuerungssysteme verwendet werden.

*Ein verteiltes System ist ein System, mit dem ich nicht arbeiten kann, weil irgendein Rechner abgestürzt ist, von dem ich nicht einmal weiß, dass es ihn überhaupt gibt [2.09].<sup>3</sup>*

Die Definition aus dem Jahre 1987 kann aus philosophischer Sicht als sicherlich sinnvoll eingestuft werden. Für die Beschreibung des Sachverhaltes der modernen auf Informationsverarbeitung basierenden Automatisierungstechnik ist diese Definition aus heutiger technischer Sicht nicht treffend und eindeutig genug.

### **Offenes verteiltes System:**

Die Informationsverarbeitung in automatisierungstechnischen Anlagen ist mit dem Übergang von der zentralen Struktur über die Dezentralisierung hin zu verteilten Systemen gravierenden Veränderungen unterlegen. Moderne Komponenten der Automatisierungstechnik unterliegen heute der Forderung nach herstellerunabhängiger Offenheit.

*Ein offenes verteiltes System ist ein verteiltes System, mit offen gelegten Spezifikationen für Schnittstellen und Formate [2.04].*

Aus den verschiedenen Versuchen die Begriffe Echtzeitsystem und verteiltes System zu definieren kann folgende Charakterisierung eines verteilten echtzeitfähigen Steuerungssystems für die Automatisierung technischer Prozesse abgeleitet werden.

Ein *verteilt* **echtzeitfähiges Steuerungssystem** ist ein informationsverarbeitendes technisches Gebilde für die Bearbeitung eines technischen Prozesses unter der Maßgabe der geforderten Einhaltung des vorgegebenen Zeitmanagements, der geforderten Redundanz und der geforderten Skalierbarkeit der Funktionalität des Gesamtsystems.

Die Größe der räumlichen Ausdehnung des Systems spielt bei dieser Betrachtung eine unwesentliche Rolle. Die Anforderungen an Systeme zur Steuerung industrieller Prozesse sind im Grundsatz vom Mikrosystem bis zum Makrosystem identisch.

Zwei wesentliche Begriffe der Automatisierungstechnik Steuerung und Regelung werden hinsichtlich ihres offenen und geschlossenen Wirkungsprinzips in dieser Arbeit nicht näher

---

<sup>3</sup> Zitat Leslie Lamport, Microsoft

explizit unterschieden, da die Anforderungen an das informationsverarbeitende technische System hinsichtlich Echtzeitfähigkeit und Verteilung für beide identisch sind.

## **2.2 Wesentliche Anforderungen an verteilte industrielle Steuerungssysteme**

### **2.2.1 Verfügbarkeit und Zuverlässigkeit**

Durch den Einsatz verteilter informationsverarbeitender Steuerungssysteme wird eine definitiv höhere Verfügbarkeit des Gesamtsystems, gegenüber dem Aufbau nach dem zentralistischen organisierten Architekturkonzepten, erreicht [2.01][2.10]. Mit dem Konzept der verteilten Verarbeitung von technischen Prozessen lassen sich informationsverarbeitende Automatisierungssysteme realisieren, welche auch unter Ausfall von Teilkomponenten im erforderlichen Maße arbeitsfähig bleiben<sup>4</sup>. Im Normalfall wird die Funktionalität des Gesamtsystems eingeschränkt, sowie die zeitliche Vorhersagbarkeit verschlechtert. Die Forderung nach einer hohen Verfügbarkeit und Zuverlässigkeit steht in Korrelation mit dem hard- und softwaremäßigen Aufwand der Realisierung zueinander.

### **2.2.2 Flexibilität und Offenheit**

Verteilte Systeme können durch ihren modularen, strukturiert organisierten Aufbau in ihrer Funktionalität erweitert, verändert aber auch entsprechend abgerüstet werden. Grundvoraussetzung für Flexibilität und Offenheit verteilter Systeme sind definierte und dokumentierte Schnittstellen der einzelnen Systemkomponenten. Die Veränderung der Funktionalität eines informationsverarbeitenden Steuerungssystems kann dabei auf hardwaremäßiger und softwaremäßiger Basis oder aus einer Kombination aus beiden erfolgen.

Derartige Funktionserweiterungen lassen sich in zwei Klassen einteilen:

#### **Statische Funktionserweiterung des verteilten Systems:**

Das technische System wird beim Anlauf auf den zu steuernden technischen Prozess angepasst. Während der Betriebsphase werden keine Veränderungen an der hard- und softwaremäßigen Ausstattung des Gesamtsystems vorgenommen. Diese Art der Funktionserweiterung wird bei einem Adressvergabeverfahren für dezentrale Peripheriegeräte, welches in der Patentschrift [2.6] „Adressvergabeverfahren für mindestens einen neu an ein Bussystem angeschlossenen Busteilnehmer“ beschrieben und urheberrechtlich geschützt ist, eingesetzt. Eine prototypische Umsetzung des Verfahrens wurde im Forschungs- und Entwicklungsvorhaben „Dynamische Adressvergabe für dezentrale Peripheriekomponenten in Profibusssystemen“ am Institut für Automatisierungstechnik realisiert [2.7][6.08][6.09].

---

<sup>4</sup> 1957 Gründung der Organisation ARPA (Advanced Research Projects Agency) mit der Zielstellung ein redundantes verteiltes Kommunikationssystem mit militärischen Hintergrund zu schaffen, welches auch bei Ausfall einzelner Knoten arbeitsfähig bleiben sollte. Das ARPANET kann als Urvater heutiger weltweiter Kommunikation durch Internettechnologien angesehen werden.

### **Dynamische Funktionserweiterung des verteilten Systems:**

Das technische System wird beim Anlauf auf den zu steuernden technischen Prozess angepasst. Es können während der Betriebsphase Erweiterungen am hardwaretechnischen Aufbau des Gesamtsystems vorgenommen werden. Die softwaremäßige Funktionalität des Gesamtsystems kann im Betriebszustand erweitert und verändert werden.

Nach dem gegenwärtigen Erkenntnisstand wird diese Art der Funktionserweiterung nur von Systemen der Officekommunikation unterstützt. Dazu gehören insbesondere die Systeme USB und FireWire<sup>5</sup>.

Dynamische Funktionserweiterungen von verteilten Steuerungssystemen sind aber auch in der industriellen Automation sinnvoll. Steuerungssysteme, welche den Anspruch einer hohen Verfügbarkeit unterliegen, werden gegenwärtig in verschiedenen Formen redundant ausgelegt. Für erforderliche Wartungsaufgaben (z.B. Austausch von Geräten) müssen jedoch gegenwärtig die meisten Anlagen heruntergefahren werden.

Bei mobilen verteilten Steuerungssystemen, wie in den Arbeiten nach [2.18], macht eine dynamische funktionelle Erweiterung des Gesamtsystems nicht nur Sinn, sondern ist aufgrund der besonderen Gegebenheiten zwingend erforderlich. Dazu gehören insbesondere die nicht eindeutig vorhersagbare Anzahl der aktiven Kommunikationsteilnehmer sowie die nicht deterministische Vorhersagbarkeit der sicheren Kommunikation der einzelnen mobilen Stationen im Gesamtsystem.

### **2.2.3 Leistungsfähigkeit**

Die Leistungsfähigkeit von verteilt aufgebauten industriellen Steuerungssystemen ist durch das modulare Grundkonzept praktisch beliebig skalierbar. Die Grenze der Leistungsfähigkeit des Gesamtsystems wird im Wesentlichen nur durch den betriebswirtschaftlichen Faktor „Kosten“ bestimmt.

### **2.2.4 Schlussfolgerungen**

Zentrales Problem und Triebkraft der modernen informationsverarbeitenden Automatisierungstechnik:

Der aktuell erreichte Entwicklungsstand informationsverarbeitender Systeme ermöglicht eine ausreichend gute Befriedigung der technologischen Bedürfnisse der Anwender bei der Realisierung automatisierungstechnischer Aufgabenstellungen. Steigende Leistungsfähigkeit verteilter informationsverarbeitender Steuerungssysteme schafft bei den Anwendern den Wunsch, Prozesse noch komplexer, anspruchsvoller und präziser automatisieren zu wollen. Der hohe Entwicklungsstand der Steuerungstechnik wirkt dabei als Triebkraft für die Ent-

---

<sup>5</sup> Norm IEEE 1394a/b auch als FireWire und i.Link bekannt

wicklung der Prozess- und Fertigungsautomatisierung. Im Gegenzug treiben die steigenden Anforderungen aus der Prozess- und Fertigungsautomatisierung die Entwicklung der modernen Steuerungstechnik weiter voran.

## 2.3 Grundlegende Kommunikationsstrukturen in der industriellen Automation

Moderne Automatisierungsanlagen stellen grundsätzlich informationsverarbeitende Systeme dar. In verteilten Systemen der Automatisierungstechnik erfolgt die Verarbeitung an verschiedenen Orten gleichzeitig und durch verschiedene zusammengefasste Gruppen von Geräten. Beim Einsatz von Kommunikationsmodellen ist eine der wichtigsten Forderungen automatisierungstechnischer Anlagen von Bedeutung, die Frage nach der Echtzeitfähigkeit. Die unterschiedlichen Anforderungen an echtzeitfähige Steuerungssysteme werden im Abschnitt 2.4 detailliert beschrieben und gegenübergestellt.

In der Informationstechnik sind zum gegenwärtigen Zeitpunkt folgende vier Kommunikationsmodelle von wesentlicher Bedeutung:

- Client-Server-Architektur
- Peer-to-Peer-Architektur
- Producer-Consumer-Architektur
- Publish-Subscriber-Architektur

### 2.3.1 Client-Server-Architektur

Unter der Client-Server-Architektur<sup>6</sup> (engl.: client-server architecture) versteht man eine kooperative Informationsverarbeitung, bei der die Aufgaben zwischen Programmen und Prozessen auf miteinander verbundenen Rechnern aufgeteilt werden. In einem solchen Verbundsystem können über standardisierte Schnittstellen und Protokolle (u.a. Sockets, TCP/IP/UDP) Rechner mit verschiedener Hardware- und Softwareausstattung zusammenarbeiten.

Die Kommunikation zwischen einer Client-Anwendung und einer Server-Anwendung basiert auf Transaktionen, die vom Client generiert und dem Server zur Verarbeitung überstellt werden. Eine Transaktion (engl.: transaction) stellt dabei eine Folge logisch zusammengehöriger Aktionen dar.

Client und Server können über ein lokales Netz (LAN, Feldbusse, Intranet) verbunden werden. Über große Entfernungen erfolgt die Kommunikation zwischen den Einheiten über das Internet, dem globale Datennetzverbund. Dabei kann es sich um Systeme verschiedener Größenordnung handeln. Das Leistungsvermögen von Clients kann das des Servers ohne Probleme auch übersteigen. Das ist insbesondere dann der Fall, wenn so genannte Embedded-Web-

---

<sup>6</sup> Liba Svobodova: Client/Server Model of Distributed Processing, 1985

Server zum Einsatz gebracht werden. Solche Geräte sind in der Regel mit relativ wenig Datenspeicher und geringen Verarbeitungsleistungen ausgerüstet und modernen PC-Systemen in dieser Hinsicht fast immer unterlegen.

Der grundlegende Ansatz für den Aufbau von Client-Server-Strukturen basiert auf der Überlegung, die Ressourcen von einzelnen beteiligten Rechnern optimal auszunutzen und dem daraus folgenden Aufbau von leistungsfähigern verteilten Gesamtsystemen zu realisieren.

**Server:** Ein Server (Dienstleister; Backend) tritt gegenüber dem Netz als Dienstanbieter auf und stellt den Clients entsprechende Funktionalität bereit. Solche Dienstleistungen können zum Beispiel Datenverwaltung, Rechnen, Drucken, Kommunikation und vieles andere sein. Der Server hat in einem Netz somit die Aufgabe, Anfragen von anderen Rechnern (Clients) zu beantworten und wenn möglich die gewünschten Informationen zurückzuschicken. Ein Server kann grundsätzlich mehrere Clients gleichzeitig bedienen.

**Client:** Ein Client (Dienstnutzer, Kunde) nimmt die Dienstleistungen, die ein Server erbringt, in Anspruch. Der Client verarbeitet die angeforderten Informationen und stellt diese in entsprechender Form dar (z.B. in einem Browser). Durch die Anwendung dieses Prinzips wird der Server entlastet.

### 2.3.2 Peer-to-Peer-Architektur

In einem Peer-to-Peer-Netzwerk (P2P-Netzwerk) ist jeder beteiligte Rechner ein Server bzw. ein Client. Jeder Rechner stellt den anderen Rechnern seine Ressourcen (Dienste, Dateien) zur Verfügung. Einen zentralen Netzwerkverwalter gibt es in solchen Systemen nicht, deshalb muss jeder Netzwerkteilnehmer selber bestimmen, welche Ressourcen er anderen Teilnehmern freigeben will.

Peer-to-Peer-Dienste sind Funktionen auf Arbeitsplatz-Computern, die anderen IT-Systemen im lokalen Netz Ressourcen zur Verfügung stellen, beispielsweise gemeinsamen Zugriff auf die Festplatte oder auf Drucker.

P2P-Netzwerke stellen kein neues Architekturmodell dar. In der Computer-Industrie basierte die Vernetzung lange Zeit vorrangig auf dieser Technologie, bis die zentralisiert organisierte Client-Server-Architektur eingeführt wurde.

Da Peer-to-Peer-Dienste wesentlich geringere Sicherheitsfunktionalitäten bieten als durch dedizierte Server bereitgestellte Dienste, sollten Peer-to-Peer-Dienste innerhalb servergestützter Netze nicht verwendet werden.

Ein Peer-to-Peer-Netzwerk eignet sich für den Aufbau von automatisierungstechnischen Strukturen bis zu 10 Computer (Clients). Steigt die Anzahl der Klienten, wird das Netz sehr schnell unübersichtlich und seine Nachteile treten zunehmend in Erscheinung. Das betrifft

besonders die räumliche Ausdehnung des Netzes, bei denen die Kommunikationspartner möglichst dicht zueinander aufgestellt sein sollten. Eine konsistente Versionsverwaltung für Dokumente ist bei diesem Netzwerktyp ebenfalls nur mit viel Aufwand realisierbar. Die erforderliche Datensicherung ist beim Peer-to-Peer-Netzwerk vollständig vom teilnehmenden Nutzer durchzuführen. Dafür lassen sich Netze nach diesem Prinzip relativ schnell und kostengünstig aufbauen.

### 2.3.3 Producer-Consumer-Konzept

Bei diesem Kommunikationsmodell erfolgt der Datenverkehr zwischen den beteiligten Geräten ohne Umweg über Drittgeräte. Das Modell basiert auf Consumern, welche Bedarf an Daten haben und Producern, welche Daten auf das Netz aussenden. Am Netz beteiligte Geräte können sowohl als Consumer als auch als Producer auftreten. Diese Eigenschaft bildet die Grundvoraussetzung für den Aufbau von flexiblen verteilten Automatisierungssystemen.

Wichtige Vertreter des Kommunikationsprinzips sind Ethernet, ControlNet, DeviceNet, Foundation Fieldbus sowie CAN. Diese Bussysteme basieren auf einem stochastischen Zugriff der Kommunikationsteilnehmer auf das Übertragungsmedium.

Von diesem Kommunikationsmodell werden folgende Kommunikationsformen unterstützt:

- M1:N1      Kommunikation      (Unicast-Prinzip)
- M1:Nj      Kommunikation      (Multicast-Prinzip)
- M1:Nj\*     Kommunikation      (Broadcast-Prinzip)
- Mj:N1      Kommunikation      (Concast-Prinzip)
- Mj:Nj      Kommunikation      (Multipeer-Prinzip)

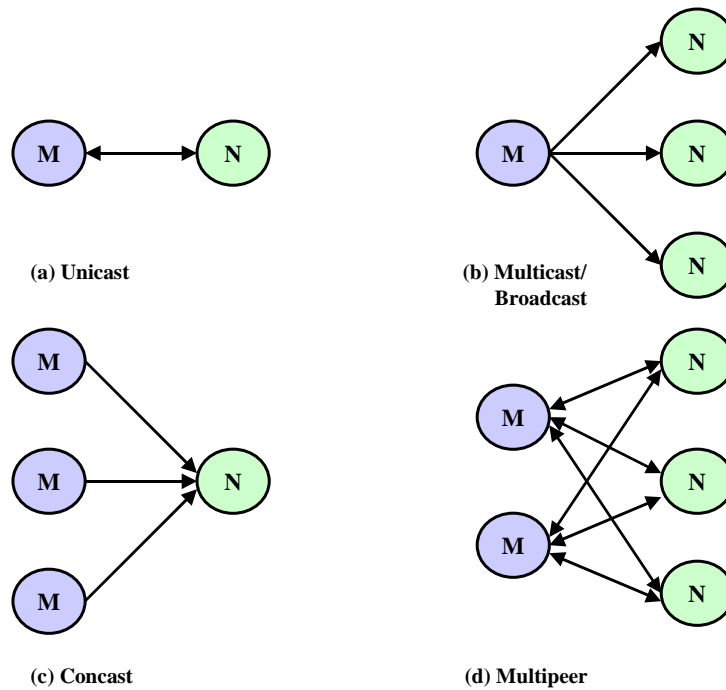


Abbildung 2.1: Kommunikationsformen in informationsverarbeitenden Systemen [2.19]

Die Variable  $M$  repräsentiert die Anzahl der Sender, die Variable  $N$  die Anzahl der Empfänger. Damit lassen sich mit diesem Kommunikationsmodell alle in der industriellen Kommunikation erforderlichen Kommunikationsbeziehungen beschreiben. Das Verteilen von Nachrichten an einzelne Gruppen ( $1:N$ ) von Busteilnehmern oder auch an alle Busteilnehmer ( $1:N_j^*$ ) erfolgt sehr effektiv. Für die moderne, auf Kommunikation basierende Automatisierungstechnik ist dieses Prinzip optimal geeignet.

Für die Umsetzung von Producer-Consumer-Diensten auf Ethernet empfehlen sich die Protokolle TCP/IP für gesicherte Verbindungen und UDP/IP für alle Echtzeit I/O Zugriffe. Damit kann die gesamte Producer-Consumer-Funktionalität auf Ethernet über TCP/IP und UDP/IP realisiert werden. Die praktische Umsetzung wird als Ethernet/IP (Ethernet/Industrial Protokoll) bezeichnet.

### 2.3.4 Publisher/Subscriber - Konzept

Das Publisher/Subscriber - Konzept ähnelt dem Producer-Consumer-Konzept. Es wird eine Menge  $M$  der Kommunikationsteilnehmer, die zu einer Gruppe gehören von einem Publisher verwaltet. Jeder zur Gruppe gehörende Teilnehmer kann dabei entsprechende Anforderungen beim Publisher stellen und wird danach vom Publisher ebenfalls mit Daten versorgt. Das Versenden der Daten kann dabei ereignisgesteuert oder zeitgesteuert erfolgen. Die Kommunikation erfolgt im Regelfall über Telegramme mit Multicastcharakter [2.11].



### 2.3.5 Weitere automatisierungstechnische Architekturen und Zugriffsprinzipien

Heutige dezentrale bzw. verteilt realisierte industrielle Steuerungstechnik basiert zu einem überwiegenden Anteil auf einer Master-Slave-Kommunikation mit verschiedenen möglichen Ausprägungen (z.B. dem Token Passing Zugriffsverfahren).

Eingesetzte Zugriffsprinzipien auf die fast ausschließlich seriellen Kommunikationskanäle in der industriellen Steuerungstechnik lassen sich hinsichtlich der Modellierung und des Zeitverhaltens in zwei wesentliche Architekturkonzepte einteilen [2.13][2.14][2.15]:

- **Ereignisgesteuerte Architekturen (event triggered)**
- **Zeitgesteuerte Architekturen (time triggered)**

#### 2.3.5.1 Ereignisgesteuerte Architekturen

Nach diesem Architekturkonzept können alle Kommunikationsteilnehmer selbständig entscheiden, ob und wann sie den Kommunikationskanal belegen wollen. Charakteristische Eigenschaft ist dabei ein stochastischer Buszugriff mit den gängigen Zugriffsverfahren CSMA/CD und CSMA/CA<sup>7</sup>.

Das Architekturkonzept ist durch das Zugriffsprinzip mit dem wesentlichen Nachteil behaftet, dass keine eindeutige deterministische Angabe über die Reaktionszeit des Kommunikationssystems getroffen werden kann. Der wesentliche Nachteil dieses Architekturkonzepts<sup>8</sup> bildet aber eine ganz wesentliche Voraussetzung für die Realisierung vollständig verteilter Systeme ohne zentrale Intelligenz. Wichtige Vertreter für den ereignisorientierten Zugriff auf das Kommunikationssystem sind zum gegenwärtigen Zeitpunkt das CAN-Bussystem mit den verschiedenen Ausprägungen<sup>9</sup> der Schicht 7 des OSI-Modells sowie das Ethernet.

#### 2.3.5.2 Zeitgesteuerte Architekturen

Zeitgesteuerte Architekturen lösen das Problem der eindeutigen Vorhersagbarkeit des Zeitverhaltens eines Kommunikationssystems aus ihrem Ansatz heraus. Klassische Vertreter zeitgesteuerter Architekturen stellen zum gegenwärtigen Zeitpunkt alle industriellen Kommunikationssysteme dar, auch als Feldbussysteme bezeichnet, welche typischerweise nach dem Master-Slave-Prinzip arbeiten. Dazu zählt unter anderem das Profibusssystem sowie der Inter-

---

<sup>7</sup> Ist aufgrund des zerstörungsfreien stochastischen Buszugriffs trotzdem als bedingt echtzeitfähig einzustufen (CAN)

<sup>8</sup> In Abschnitt 5 wird die Evolution von der zentralisierten zur verteilten Automatisierung mit ihren wichtigsten Merkmalen vorgestellt.

<sup>9</sup> Bei CAN sind nur die Schichten 1 und 2 nach dem OSI-Modell spezifiziert. Die Schicht 7, bei gängigen Feldbussen ebenfalls von den Herstellern spezifiziert, unterliegt bei CAN keiner zwingenden Festlegung. Erst mit der Einführung des CAN-OPEN-Konzepts wurde eine einheitliche Kommunikationsschnittstelle geschaffen. Deren Nutzung erfolgt dabei auf freiwilliger Basis.

bus-S, bei denen der Einsatz dezentraler Peripheriekomponenten eine wesentliche Rolle spielt.

Im klassischen Sinne haben zeitgesteuerte Architekturen den entscheidenden Vorteil der genauen zeitlichen Vorhersagbarkeit des Kommunikationssystems für die einzelnen Fälle harter Echtzeitforderungen. Nachteilig erweist sich die Tatsache, dass durch die zentrale Steuerungskomponente (Industrie PC oder SPS) Realisierungen flexibler verteilter Steuerungssysteme stark erschwert werden<sup>10</sup>.

Es kann an dieser Stelle die Schlussfolgerung getroffen werden, dass zukünftige industrielle Kommunikationssysteme mit einer Kombination aus beiden Konzepten arbeiten werden. Ansätze wie die Erweiterung zu TTCAN oder Industrial Ethernet untermauern diese Feststellung.

## **2.4 Echtzeitanforderungen in verteilten Systemen der Automatisierungstechnik**

### **2.4.1 Begriffe**

Echtzeitbetrieb ist eine Betriebsart eines Rechensystems bei dem Programme zur Verarbeitung anfallender Daten ständig betriebsbereit sind, in der Form, dass die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind. Die Daten können je nach Anwendungsfall nach einer zeitlich zufälligen Verteilung oder zu vorbestimmten Zeitpunkten anfallen [2.02] (siehe auch Abschnitt 2.1).

Industrielle Kommunikationssysteme unterscheiden sich in ihren Grundprinzipien nicht von anderen informationsverarbeitenden Systemen. Eine wesentlich strengere Forderung ist die garantierte Einhaltung der Rechtzeitigkeit bzw. Gleichzeitigkeit der Verarbeitung der Informationen. Das heißt, im industriellen Bereich besteht nicht nur die Notwendigkeit Informationen sicher zu Übertragen und zu Verarbeiten, sondern zusätzlich die Forderung dass die Verarbeitung von Informationen zum richtigen Zeitpunkt stattfindet. Die für die sichere Kontrolle eines Prozesses erforderliche Zeit bestimmt die maximale Zeitschranke (Deadline), welche ein als echtzeitfähig eingestuftes Automatisierungssystem einhalten muss. Die einzuhaltende Zeitschranke in ihrer zeitlichen Dimension ist primär von dem zu steuernden bzw. zu regelnden technischen Prozess abhängig.

---

<sup>10</sup> Bei PROFIBUS DPV1 besteht die Möglichkeit des Querverkehrs der dezentralen Peripherie. Es handelt sich dabei jedoch um einen Pseudomechanismus bei dem eine Gruppe von Slaves alle Informationen mithören, welche für die Lösung der Aufgabenstellung dieser Gruppe erforderlich ist. Ein echter Querverkehr besteht aufgrund des Master-Slave-Prinzips nicht.

## 2.4.2 Fallunterscheidungen und Auswirkungen auf das Verhalten verteilter Systeme

Es müssen grundsätzlich zwei prinzipielle Formen der Echtzeitfähigkeit von informationsverarbeitenden industriellen Steuerungssystemen unterschieden werden:

- Harte Echtzeit (hard real time)
- Weiche Echtzeit (soft real time)

In einem weichen Echtzeitsystem sind Zeitschranken grundsätzlich einzuhalten. Überschreitungen der vorgegebenen maximalen Zeitschranke werden in gewissen Maße zugelassen. Die Zeitschranke kann also in solchen Anwendungsfällen, in denen trotz der Verletzung der Echtzeitforderung, das informationsverarbeitende System nicht in eine instabile Lage versetzt wird, überschritten werden.

Die Überschreitung der Zeitschranke kann in den folgenden zwei Formen auftreten:

### **Stochastische Überschreitung**

Die Überschreitung der Zykluszeit unterliegt stochastischen Einflüssen (kurzzeitige nicht ausreichende Performance, Störungen des Kommunikationskanals u.s.w.). Die Überschreitung der Zykluszeit erfolgt stets zu einem Prozentsatz kleiner 100 % ( $p < 1$ ).

Technisch zulässig ist beispielsweise eine Überschreitung der Zykluszeit von ca. 0.1% der Zyklen [2.17].

### **Deterministische Überschreitung**

Die Überschreitung der Zykluszeit unterliegt determinierten Einflüssen (ständig zu geringe Performance, ständig zu geringe Kanalkapazität des Kommunikationssystems u.s.w.). Die Überschreitung der Zykluszeit erfolgt zu 100%, faktisch wird jedes Intervall überschritten. Ein technisches System kann damit, je nach Aufgabenstellung, auch als dauerhaft instabil betrachtet werden und stellt einen Projektierungsfehler des Anlagensystems dar.

In modernen Automatisierungssystemen, welche hochdynamische und timingmäßig kritische Prozesse zu bedienen, beobachten und zu steuern haben, spielt diese Form der Echtzeitfähigkeit eher eine untergeordnete Rolle. Bei der Integration industrieller Steuerungssysteme in das Internet stellt diese Form der Echtzeitfähigkeit den Regelfall dar.

In einem harten Echtzeitsystem besteht die zwingende Notwendigkeit, die durch den zu steuernden Prozess vorgegebene Zeitschranken, unter allen möglichen Bedingungen einzuhalten. Eine Überschreitung der Zeitschranke bringt das Automatisierungssystem in eine instabile Lage. In diesem Zustand kann keine eindeutige Zuordnung von Prozesseingangsgrößen zu Systemausgangsgrößen durch das System getroffen werden. Mit den Begriffen der Verkehrs- und Bedientheorie kann das System den erforderlichen Anforderungsstrom nicht mehr bewäl-

tigen. Ein technisches System welches sich in diesen Zustand manövriert hat, kann als versagt bezeichnet werden.

Für den Aufbau von industriellen Steuerungssystemen ist die Realisierung eines harten Echtzeitverhaltens von entscheidender Bedeutung.

Dabei bedeutet harte Echtzeit, die Reaktionszeit  $T_e$  des Systems auf ein Ereignis  $e$  muss in jedem Fall kleiner als die maximale Antwortzeitdauer  $P_e$  sein.

Harte Echtzeitsysteme lassen sich aus Sicht der Informatik in vier Fälle mit speziellen Anforderungsprofilen einteilen. Nach [2.01] sind dabei folgende Fallunterscheidungen getroffen worden:

**Fall 1:** *Eine vom Programm auszuführende Funktion (z.B. Erfassen eines Messwertes, Ausgabe eines Stellsignals) hat genau zu bestimmten Zeitpunkten  $t_1, t_2, \dots, t_n$  zu erfolgen.*

**Fall 2:** *Eine Funktion hat während eines Zeitintervalls zu erfolgen, das durch einen Zeitpunkt  $t_1$  und einen oberen und unteren Toleranzbereich gegeben ist.*

**Fall 3:** *Eine Funktion hat spätestens zu einem bestimmten Zeitpunkt zu erfolgen.*

**Fall 4:** *Eine Aktion darf frühestens zu einem bestimmten Zeitpunkt erfolgen.*

Im folgenden Abschnitt sollen die typischen Anforderungsprofile industrieller Kommunikationssysteme in die vier Fallunterscheidungen der Echtzeitfähigkeit eingestuft werden.

### **Fall 1:**

Bei der Automatisierung von Prozessen mit sehr hohen Anforderungen an die präzise Einhaltung von Zeiten ist der Fall 1 von grundlegender Bedeutung. Dieser Fall tritt genau dann ein, wenn technische Prozesse hochgenau synchronisiert werden müssen. Insbesondere ist dieser Fall aber dann von Bedeutung, wenn das Kommunikationssystem sich als ein Teil in technische Systeme integriert, welche regelungstechnische Aufgaben zu lösen hat.

Das  $\Delta$  welches sich aus der Summe von  $\Delta t_1, t_2, \dots, t_n$  zusammensetzt ist um Potenzen kleiner als  $\sum t$ .

In informationsverarbeitenden Steuerungssystemen, welche nach zeitgesteuerten Architekturkonzepten arbeiten, setzt sich die Systemzykluszeit  $t_{\text{Summe}}$  prinzipiell aus der Summe der Zeiten von  $t_1, t_2, \dots, t_n$  zusammen.

Für die Realisierung regelungstechnischer Aufgabenstellungen ist die genaue Kenntnis der Schwankungsbreite des Zeitintervalls von ausschlaggebender Bedeutung.

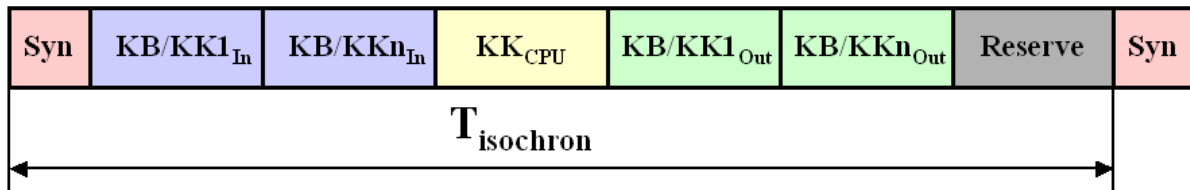


Abbildung 2.2: Belegungszeiten in industriellen Kommunikationssystemen (Beispiel)

In der Abbildung 2.2 ist der prinzipielle Ablauf der Kommunikation zwischen einzelnen Kommunikationsknoten in einem zeitgesteuerten informationsverarbeitenden Steuerungssystem dargestellt. Charakteristisch ist bei der Realisierung dieser Art von Steuerungssystemen die Verwendung einer speziellen Kommunikationsbeziehung, mit der eine hochpräzise Zeitbasis im Gesamtsystem gewährleistet wird. Die Reihenfolge der Kommunikationsbeziehungen hat bei den einzelnen Kommunikationssystemen unterschiedliche Anordnungen und soll nur das Prinzip verdeutlichen. Ebenfalls soll die Größe des Zeitintervalls an dieser Stelle nicht weiter berücksichtigt werden, da dieser Faktor von der steuerungstechnischen Aufgabenstellung sowie vom speziell eingesetzten industriellen Kommunikationssystem abhängig ist.

Die einzelnen Belegungszeiten haben dabei folgende grundlegende Bedeutung:

- **Syn** Synchronisationstelegramm an alle Busteilnehmer
- **KB/KK1In** Bearbeitung Eingangsdaten von Kommunikationsknoten 1
- **KB/KKnIn** Bearbeitung Eingangsdaten von Kommunikationsknoten n
- **KKCPU** Bearbeitung Kommunikationsknoten CPU
- **KB/KK1Out** Bearbeitung Ausgangsdaten von Kommunikationsknoten 1
- **KB/KKnOut** Bearbeitung Ausgangsdaten von Kommunikationsknoten n
- **Reserve** variable Zeitreserve

Taktsynchron bzw. isochron arbeitende industrielle Kommunikationssysteme stellen charakteristische Realisierungen dieses Falls der Echtzeitfähigkeit in der Automatisierungstechnik dar. In taktsynchron arbeitenden Systemen, einen speziellen Sonderfall zeitgesteuerter Architekturen, setzt sich die Systemzykluszeit aus der Summe von  $t_1, t_2, \dots, t_n$  zusammen.

Aus regelungstechnischer Sicht muss das Verhältnis  $\frac{\Delta T_{zyklus}}{T_{zyklus}} \ll \frac{1}{10}$  oder besser gewählt werden um den Einfluss des Jitters<sup>11</sup> auf das Systemverhalten vernachlässigen zu können.

<sup>11</sup> Jitter: Fluktuieren oder Schwankung der Zyklus bzw. Durchlaufzeit (typische Werte  $\leq 1 \mu s$ )

In der heutigen industriellen Kommunikation spielt diese Form der Echtzeitfähigkeit von Steuerungssystemen eine immer bedeutendere Rolle. Gegenwärtig sind bzw. werden von verschiedenen Anbietern industrieller Kommunikationssysteme Mechanismen für zeitsynchrone Betriebsarten eingeführt, um die gestiegenen Anforderungen der Echtzeitfähigkeit gerecht zu werden. In den Abschnitten 3, 5 und 6 wird auf spezielle Realisierungen in den Systemen detaillierter eingegangen.

### **Fall 2:**

Dieser Anwendungsfall tritt bei industriellen Kommunikationssystemen immer dann ein, wenn technische Prozesse gesteuert und geregelt werden sollen, welche unbedingt echtzeitfähig bearbeitet werden müssen, aber nicht präzise zeitsynchronisiert sein brauchen. Das Kommunikationssystem bewältigt den Anforderungsstrom in einem genau festgelegten Intervall. Die Summe aus der Zeitdauer des Intervalls und der maximalen positiven Toleranzgrenze muss stets kleiner als die geforderte Prozesszykluszeit sein. In diesem Betriebsfall kann das Kommunikationssystem als stabil betrachtet werden. Ist die Summe gleich der Prozesszykluszeit, dann befindet sich das Kommunikationssystem an der Stabilitätsgrenze. Eine Ausführliche Betrachtung dieses Falls der Echtzeitfähigkeit ist nicht erforderlich, da das zeitliche Verhalten industrieller Kommunikation mit Fall 1 und 3, je nach Anforderung, modelliert werden kann.

### **Fall 3:**

Das Kommunikationssystem bewältigt alle Anforderungen bis spätestens zu einem festgelegten Zeitpunkt. Unter Einhaltung der Bedingung  $T_{\text{Zyklus}} \leq T_{\text{Prozess}}$  besteht zu jeder Zeit eine 100%ige Kontrolle über den zu steuernden technischen Prozess. Solange diese Bedingung eingehalten wird, besteht kein zwingend notwendiges Erfordernis der genauen Kenntnis der Schwankungsbreite  $T_{\text{Zyklus}}$ .

Für die Anwendung in regelungstechnischen Aufgabenstellungen muss diese Bedingung zwingend schärfer formuliert werden

$$T_{\text{Zyklus}} \ll T_{\text{Prozess}}$$

Technisch erforderliche Werte sind

$$T_{\text{Zyklus}} = T_{\text{Prozess}} / T_{\text{Differenz}} \quad \text{mit der Bedingung } T_{\text{Differenz}} > 10.$$

Da das Abtasttheorem unter dieser Bedingung in jedem Fall eingehalten wird, darf die Zykluszeit in der zulässigen Weise schwanken.

Der Fall 3 der Echtzeitfähigkeit wird von den meisten industriellen Steuerungssystemen unterstützt<sup>12</sup>. Der Anwendungsfall kann zum gegenwärtigen Zeitpunkt als Stand der Technik auf dem Gebiet der industriellen Kommunikationstechnik für die Prozess- und Fertigungsautomatisierung angesehen werden.

---

<sup>12</sup> z.B. SIMATIC PROFIBUS DP, Interbus-S

#### Fall 4:

Das Ausführen einer Aktion „frühestens“ zu einem bestimmten Zeitpunkt, also künstliches warten, muss aus Sicht der echtzeitfähigen Informationsverarbeitung in Automatisierungssystemen aus zwei Richtungen betrachtet werden. Aus Sicht der erforderlichen echtzeitfähigen Kommunikation, welche bei Feldbussen in den Schichten 1 und 2 des OSI-Modells und bei Ethernet zusätzlich in den Schichten 3 und 4 des OSI-Modells stattfindet, spielt diese Form der Echtzeitfähigkeit eine unwesentliche Rolle. Nach der Definition ist die zeitliche Dimension der Echtzeitfähigkeit nicht festgelegt und ist primär vom zu steuernden Prozess abhängig.

Dieser Fall beschreibt eine der typischen Problemstellungen aus dem Bereich der Echtzeitbetriebssysteme, z.B. dem Warten bis ein benötigtes Betriebsmittel<sup>13</sup> frei ist, bevor eine anstehende Aufgabe weiterbearbeitet werden kann bzw. darf. Im Extremfall kann eine solche Situation zu einem so genannten Deadlock<sup>14</sup> führen.

Nach dem OSI-Modell ist dieser Fall bei Kommunikationssystemen in die Schicht 7 (Applikation) einzustufen.

Derartige Systemzustände werden in Echtzeitsystemen vermieden, wenn von den folgenden vier Bedingungen nur eine einzige nicht erfüllt ist:

- Wechselseitiger Ausschluss
- Belegung und Wartebedingung
- Nicht Unterbrechbarkeit
- Zyklische Wartebedingung

Dieser Fall spielt bei Betrachtungen zu verteilten echtzeitfähigen Steuerungssystemen in dieser Arbeit keine wesentliche Rolle.

---

<sup>13</sup> Als Betriebsmittel bezeichnet man alle Komponenten der Hardware und Software eines Rechnersystems, die ein Prozess bis zur seiner Terminierung benötigt. Diese Betriebsmittel werden dem Prozess durch das Betriebssystem zugeteilt.

<sup>14</sup> Deadlock: (tödliche Umarmung) Eine Menge von Prozessen befindet sich in einem Deadlock-Zustand, falls jeder Prozess der Menge auf ein Ereignis wartet, dass nur ein anderer Prozess der Menge auslösen kann. Es wartet beispielsweise eine Transaktion auf eine andere bei beiderseitigem lock (Verriegelung).

### 2.4.3 Schlussfolgerung und Bewertung der Echtzeitanforderungen

Mit weicher Echtzeitforderung lassen sich alle technischen Kommunikationssysteme modellieren, bei denen die Aktualisierung von Prozessdaten innerhalb eines vorgegebenen Zeitintervalls erfolgen muss. Die Zykluszeit des zu steuernden Verarbeitungsprozess stellt hierbei die primäre Vorgabegröße für das informationsverarbeitende System dar. Es besteht dabei die grundsätzliche Forderung, dass bei der Informationsverarbeitung eine vom Prozess geforderte Zeitschranke nicht überschritten werden sollte. Es kann jedoch zu einem gewissen Prozentsatz die Zeitschranke überschritten werden, solange ausreichende Informationen über die Sensorik und Aktorik im Gesamtsystem verfügbar sind.

Einsatzgebiete sind in der Regel die Steuerung von niedrig dynamischen Prozessen wie z.B. die Regelung einer Warmwasserheizung, bei denen ein kurzzeitiger stochastisch auftretender Informationsverlust das System und seine Umwelt in keinerlei Gefahr bringt.

Mit harter Echtzeitfähigkeit müssen alle industriellen Kommunikationssysteme betrieben werden, bei denen ein genau festgelegtes Zeitregime für den zu steuernden Prozess zwingend erforderlich ist. Eine Verletzung des Zeitmanagements bringt dabei immer das System sowie seine Umwelt in jedem Fall in eine gefährliche Situation.

Automatisierungstechnische Aufgabenstellungen unterscheiden dabei prinzipiell zwischen den Forderungen Rechtzeitigkeit und Gleichzeitigkeit.

#### **Begriff Rechtzeitigkeit**

Rechtzeitigkeit ist für die Steuerung und Regelung von technischen Prozessen immer erforderlich. Die vorgegebene Zeitschranke darf unter keinem Fall überschritten werden. Dabei spielt die zeitliche Konstanz der Einhaltung der vorgegebenen Zeitschranke keine Rolle. Rechtzeitigkeit wird heute von fast allen industriellen Kommunikationssystemen (Feldbusse) unterstützt und stellt somit den Stand der Technik dar. Typische Anwendungsfelder sind Prozessautomatisierung, Verkehrsleittechnik, Energie- und Umwelttechnik und große Teile der Fertigungsautomatisierung. Diese industriellen Kommunikationssysteme werden in Fall 3 eingruppiert.

#### **Begriff Gleichzeitigkeit**

Gleichzeitigkeit ist Grundvoraussetzung für die Modellierung und Realisierung von verteilten industriellen Steuerungssystemen, welche für alle Arten der Synchronisation von Prozessen eingesetzt werden sollen. Die zeitliche Konstanz des Zeitmanagements ist von ausschlaggebender Bedeutung für die Güte des Systems. Es wird von isochroner bzw. äquidistanter Betriebsweise gesprochen. Die wichtigste treibende Kraft für die forcierte Entwicklung dieser Technologie ist der Wunsch, Antriebssysteme auf elektronischem Wege verteilt zu synchronisieren. Das Prinzip der Gleichzeitigkeit bildet aber auch die Grundlage für Kommunikationssysteme, welche auf stochastische Buszugriffe basieren, durch entsprechende informationstechnische Maßnahmen echtzeitfähig zu gestalten. Dazu zählt insbesondere die Konzepte für Industrial Ethernet und das CAN mit der Erweiterung zu TTCAN. Kommunikationssysteme mit dieser Forderung entsprechen Fall 1 der harten Echtzeitkriterien.



### 3 Kommunikationssysteme für den Aufbau verteilter Systeme

#### 3.1 Grundlegender Begriff

Der Begriff Kommunikation umfasst den Austausch von Informationen. Nach der ISO/IEC wird der Begriff Information wie folgt definiert:

“The meaning that is currently assigned to data by means of the conventions applied to data.”

#### 3.2 Einteilung der Kommunikationssysteme

Industrielle Kommunikationssysteme lassen sich nach verschiedenen Gesichtspunkten einteilen. Für Betrachtungen über die Einsatzfähigkeit der verschiedenen Systeme in verteilten echtzeitfähigen Steuerungssystemen ist eine Einteilung nach dem genutzten Übertragungsmedium sowie nach dem Prinzip des Zugriffes auf das Übertragungsmedium zwingend erforderlich. Bei weiterführenden Betrachtungen des Verhaltens echtzeitfähiger informationsverarbeitender Steuerungssysteme hinsichtlich der Größe des Faktors „Zeit“ müssen seitens der eingesetzten Systeme zusätzliche Einflüsse berücksichtigt werden. Dazu gehören die Effizienz des Protokolls (entspricht dem Verhältnis aus Brutto- und Nettodatenrate), der Datensicherungsmechanismen und letztendlich der Bitübertragungsrate des Kommunikationssystems.

##### 3.2.1 Einteilung nach der Übertragungstechnik

Die Übertragungstechnik hat einen wesentlichen Einfluss auf das Echtzeitverhalten der industriellen Kommunikation. Grundsätzlich kann zwischen leitungsgebundener und drahtloser Kommunikation unterschieden werden.

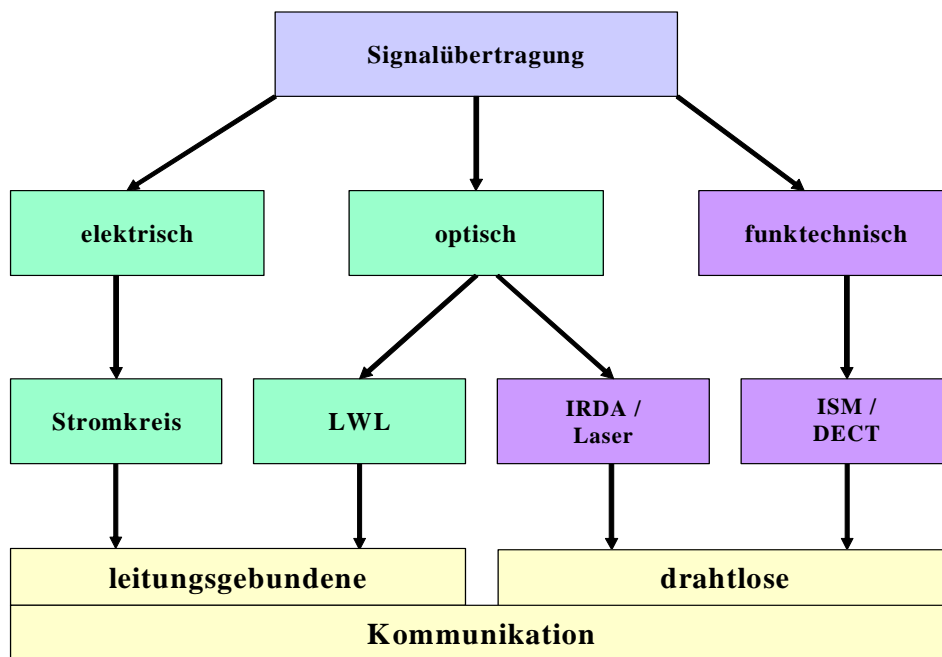


Abbildung 3.1: Einteilung nach den Übertragungssystemen (Auszug)

In der Abbildung 3.1 wird ein Überblick über wichtige Übertragungssysteme, welche für den industriellen Einsatz geeignet sind, gegeben. Prinzipiell kann mit jedem Übertragungssystem echtzeitfähige Kommunikation realisiert werden. Echtzeitfähige drahtlose Kommunikation<sup>15</sup> zu realisieren gestaltet sich prinzipbedingt schwieriger, als bei leitungsgebundener Übertragungstechnik [2.18][3.11][3.12].

### 3.2.2 Einteilung nach den Medienzugangsverfahren

Das Echtzeitverhalten von industriellen Kommunikationssystemen wird von der Organisation des Zugriffs auf das Übertragungsmedium maßgeblich beeinflusst. Grundsätzlich wird dabei zwischen stochastischem und deterministischem Zugriff unterschieden. In der feldbusbasierten industriellen Kommunikation herrschen zum gegenwärtigen Zeitpunkt deterministische Zugriffsverfahren mit unterschiedlichen Ausrichtungen vor. Ein typischer Vertreter dieses Zugriffsverfahrens ist das Profibussystem mit seinen verschiedenen Profilausprägungen. Stochastische Zugriffsverfahren, welche in der Officekommunikation stark verbreitet sind, tragen zum gegenwärtigen Zeitpunkt nur zu einem geringen Anteil in der industriellen Kommunikation bei. Wesentlicher Grund für diese Situation ist das prinzipbedingt nicht vorhersagbare Zeitverhalten dieser Zugriffsverfahren. Wichtige Vertreter des CSMA-Zugriffsprinzips sind CAN (CSMA/CA) und Ethernet (CSMA/CD). Durch entsprechende informationstechnische Maßnahmen<sup>16</sup> kann ein deterministisches Verhalten von Ethernet erreicht bzw. von CAN verbessert werden. Die Abbildung 3.2 gibt einen Überblick über die gängigen Zugriffsverfahren für die industrielle Kommunikation [3.04].

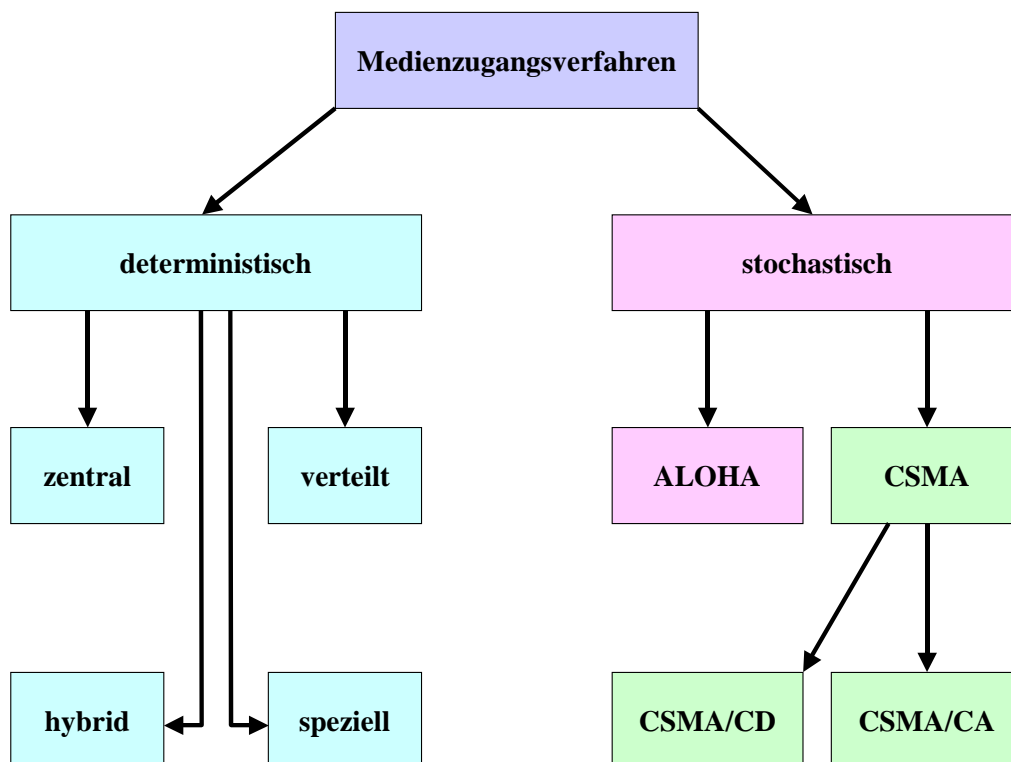


Abbildung 3.2: Einteilung nach den Medienzugangsverfahren (Auszug)

<sup>15</sup> Siehe 3.5

<sup>16</sup> Siehe 3.6.4

### 3.3 Ausgewählte parallele Kommunikationssysteme

**Definition:** Bitparallel (engl. bit parallel) stellt eine Art der Datenübertragung dar, bei der die Datenbits nebeneinander (parallel) übertragen werden [2.16].

Im Normalfall wird bei Systemen, welche nach diesem Prinzip arbeiten, eine Trennung zwischen Datenbus (je nach System 8, 16 oder 32 Bit breit) und Steuerbus durchgeführt. Somit können je nach Breite des Datenbusses ein oder mehrere Bytes gleichzeitig übertragen werden. Die Synchronisation zwischen den kommunizierenden Geräten erfolgt dabei durch entsprechende Signale auf dem Steuerbus.

Parallele technische Kommunikationssysteme eignen sich, hinsichtlich ihres echtzeitfähigen Betriebsverhaltens, für die Realisierung verteilter Steuerungssysteme. Durch die parallele Übertragung der Informationen entwickelt sich mit zunehmender Breite der Bussysteme das Problem des Anstiegs des Verkabelungsaufwandes. Die übertragenen Informationen werden im Regelfall asymmetrisch übertragen, d.h. das Signal richtet sich gegen ein festes Potential, meist die Masse und unterliegt daher größeren elektrischen Störeinflüssen als bei symmetrischer Übertragung. Aus diesen beiden Gründen ist die räumliche Ausdehnung von verteilten Kommunikationssystemen dieser Art relativ stark begrenzt.

Sender	Empfänger	wichtige Gerätesysteme
1	1	Centronics
1	n	IEEE488 GPIB
m	n	SCSI VME

Abbildung 3.3: Klassifizierung paralleler Kommunikationssysteme

In der Abbildung 3.3 sind die wichtigsten Vertreter paralleler Kommunikationssystemen dargestellt [3.03]. Systeme wie z.B. ISA oder PCI als typische Vertreter der Gerätetechnik wurden aufgrund des speziellen Einsatzgebietes nicht in die Betrachtung aufgenommen.

### 3.4 Ausgewählte serielle industrielle Kommunikationssysteme

**Definition:** Bitseriell (engl. bit serial) stellt eine Art der Datenübertragung dar, bei der die Datenbits hintereinander (seriell) auf dem Übertragungsmedium übertragen werden [2.16].

### 3.4.1 Ausgewählte Kommunikationssysteme aus dem Feldbereich

#### 3.4.1.1 PROFIBUS

Mit der Entwicklung des PROFIBUS (Process Field Bus) wurde ein offenes, herstellerunabhängiges Konzept verfolgt. PROFIBUS steht heute in den Derivaten FMS, DP und PA zur Verfügung. Die Bedeutung von PROFIBUS FMS ist insbesondere durch die Einführung der DP-Variante, mit ihren speziellen Profilausrichtungen V1 und V2, stark eingeschränkt worden. PROFIBUS ist in den Normen DIN 19245, Teil 1 bis 4 und in der Europäischen Norm EN 50170-2 und auch in der IEC 61158 spezifiziert. Durch diese einheitliche Norm können Geräte verschiedener Hersteller über dieses Bussystem miteinander kommunizieren.

Mit dem Erscheinen der Europäischen Norm haben die nationalen Normen, aus denen die EN 50170 hervorgegangen ist, ihre Bedeutung verloren. Die PROFIBUS-DP-Variante wurde mit dem Ziel entwickelt, das Bussystem im Peripheriebereich als auch im Sensor-Aktuator-Bereich einsetzen zu können. Diese Variante bietet dem Anwender die Multimasterfähigkeit (vergleichbar mit der FMS-Multimasterfähigkeit). Multimasterbetrieb wird aber in einem DP-System eher selten eingesetzt, es dominiert die Master-Slave-Struktur. PROFIBUS-DP stellt eine feste Master-Slave-Struktur dar. Eine direkte Kommunikation von Slave zu Slave ist in der ursprünglichen Norm DIN 19245 nicht vorgesehen. Die Kommunikation zwischen zwei DP-Slaves erfolgt grundsätzlich über den DP-Master. Mit der Verabschiedung der Norm IEC 61158 wird die Kommunikationsart Querverkehr zwischen Slaves standardisiert.

PROFIBUS-DP ist topologisch als Linie ausgelegt. Als Übertragungsmedien kommen verdrehte Zweidrahtleitungen und Lichtwellenleiter zum Einsatz. Die maximale Übertragungsrate von 500 KBit/s wurde mit der EN 50170 auf 12 MBit/s erhöht. Ab 1,5 MBit/s Datenübertragungsrate werden an die Übertragungstechnik erhöhte Anforderungen<sup>17</sup> gestellt. Im OSI-Modell sind, wie bei den meisten Feldbussen, nur die Schichten 1, 2 und 7 festgelegt worden. Die Effizienz des Protokolls erlaubt eine schnelle, zentral organisierte Datenübertragung.

Zurzeit wird das Konzept des Feldbusses PROFIBUS um folgende drei wichtige Profilausprägungen erweitert:

- PROFINet-Profil
- PROFISave-Profil
- PROFIDrive-Profil

Die Anforderungen an die industriellen Kommunikationssysteme sind in den vergangenen Jahren stetig angestiegen. Komplexe Automatisierungslösungen werden auch in Zukunft immer höhere Datenübertragungsraten, sowie Verbesserungen bei Feldbusprotokollen und -diensten erfordern. Um mit dieser Entwicklung Schritt zu halten, wurden in der Norm IEC 61158 die Themen Äquidistanz und Querverkehr standardisiert [2.31]. Nachfolgend werden das Äquidistanz- und das Querverkehrsprinzip im Überblick dargestellt:

---

<sup>17</sup> Geschirmtes Kabel sowie der Verzicht auf Stickleitungen

## Äquidistanz-Prinzip

Das Äquidistanz-Prinzip<sup>18</sup> realisiert Mechanismen, mit denen beim Feldbussystem PROFIBUS ein definiertes vorhersagbares Zeitverhalten garantiert werden kann. Ziel der Normung ist es, Buszykluszeiten mit einer maximalen Schwankung der Zykluszeit von 1µs zu erreichen. Diese Forderung stellt die Grundvoraussetzung für die Realisierbarkeit verteilter regelungstechnischer Strukturen dar. Dazu zählt insbesondere die Synchronisation von Antrieben in verteilten Mehrachssystemen.

Die praktische Realisierung berücksichtigt im Buszyklus die Übertragung folgender spezifischer Daten, Dienste und sonstiger Anforderungen:

- Übertragung von zyklischen Daten
- Übertragung von azyklischen Daten
- FDL\_Request-Realisierung<sup>19</sup>
- Realisierung einer Reservezeit

## Querverkehrs-Prinzip

Unter dem Querverkehrs-Prinzip versteht man das Versenden von Nettodaten eines Slaves, der als Publisher (Datenquelle) arbeitet, an einen oder mehrere andere Slaves, die als Subscriber (Empfänger) arbeiten. Ein Publisher wird durch ein DXCHG-Telegramm vom Master aufgefordert, seine Antwortdaten nicht mehr nach dem Unicastprinzip (Master-Slave), sondern nach dem Multicastprinzip (Slave sendet direkt an eine Gruppe von Teilnehmern) zu senden. Das heißt, Slaves, welche als Subscriber parametrieren wurden, können dem Antworttelegramm des antwortenden Slaves direkt Informationen entnehmen. Daraus folgt, dass eine zusätzliche Kommunikation über den Master nicht mehr zwingend erforderlich ist. Diese Erweiterung der Spezifikation setzt aber trotzdem das deterministische Master-Slave-Prinzip nicht außer Kraft. Die Slaves dürfen auf Protokollebene nach wie vor nur auf Anforderung durch den Master das Übertragungsmedium belegen. Allerdings muss beachtet werden, dass als Subscriber nur Slaves eingesetzt werden können, die die DPV1-Spezifikation erfüllen. Als Publisher können alle Arten von Slaves (DP und DPV1) eingesetzt werden.

### 3.4.1.2 CAN

CAN (Controller Area Network) wurde 1981 von Bosch entwickelt und erstmals 1993 von der ISO standardisiert. Intel stellte als erster Halbleiterhersteller 1985 den ersten Protokoll IC vor. Zum jetzigen Zeitpunkt werden von fast allen Herstellern von Mikrocontrollern integrierte Lösungen angeboten. Im Jahr 2000 erfolgte eine Überarbeitung des Standards, welche in die ISO 11898 überging.

---

<sup>18</sup> die isochrone Arbeitsweise erlaubt Echtzeitfähigkeit nach Abschnitt 2.4.1 Fall 1

<sup>19</sup> FDL Fieldbus Data Link

## Echtzeitbewertung von CAN

*Durch die Art der Busarbitrierung (CSMA/CA with AMP) werden zwar Zerstörungen<sup>20</sup> am Bus vermieden weil sich die höherpriorisierte Nachricht bei gleichzeitigem Sendestart durchsetzt und den Kommunikationskanal belegt. Dies kann aber dazu führen, dass niederpriorisierte Nachrichten zeitweilig bis dauerhaft blockiert werden können. Für „harte“ Echtzeitanforderungen in sicherheitskritischen Applikationen kann die prioritätsgesteuerte Datenübertragung problematisch sein [3.69][6.20].*

Das Grundproblem des CAN, hinsichtlich der Echtzeitfähigkeit, liegt bei Version 2.0A als auch bei Version 2.0B im Arbitrierungskonzept (Event Triggered) für den Zugriff auf das Übertragungsmedium. Die Protokollspezifikation sieht einen zerstörungsfreien nachrichtenorientierten Zugriff auf das Übertragungsmedium vor, bei dem höher priorisierte Nachrichten vorrangig den Bus belegen. Bei sehr hohem Verkehrsaufkommen können niederpriorisierte Nachrichten das Kommunikationssystem mit einer Wahrscheinlichkeit von  $0 < p < 1$  bzw. von  $p=0$  belegen.

## **Wesentliche Merkmale und Eigenschaften des Ansatzes der Erweiterung zu TTCAN:**

Durch die Einführung des Konzeptansatzes TTCAN lässt sich die Echtzeitfähigkeit des Kommunikationssystems unter Nutzung des Zeitschlitzverfahren<sup>21</sup> verbessern. Die erforderliche Zeitsynchronität wird dabei durch das Senden einer Referenznachricht mit hoher Priorität, welche als Broadcastnachricht alle Kommunikationspartner adressiert, realisiert. Der Time Triggered Protokollaufsatz wird auf die bestehende CAN-Infrastruktur aufgesetzt, so dass Time-Triggered und Event-Triggered Kommunikationen über das gleiche physikalische Bus-system zusammenarbeiten können. Es besteht die Möglichkeit diese Erweiterung auf software- und hardwaremäßiger Basis zu realisieren [3.39].

Der Protokollaufsatz sieht folgende wesentliche Erweiterungen vor:

- Es wird im Kommunikationssystem ein Zeitmaster eingeführt, welcher in genau definierten Abständen eine Referenznachricht sendet. Mit dieser Referenznachricht müssen sich alle Kommunikationsteilnehmer synchronisieren. Dadurch wird im Gesamtsystem ein einheitliches Zeitmanagement (unter Vernachlässigung der Laufzeiten auf dem Übertragungsmedium) realisiert.
- Aus Redundanzgründen können im Kommunikationssystem mehrere Zeitmaster betrieben werden. Die verfügbare Bandbreite zwischen den Synchrontelegrammen wird in einen deterministischen und einen stochastischen Bereich aufgeteilt. In einem

---

<sup>20</sup> Bei gleichzeitigen Versuch mehrerer Kommunikationsknoten auf das Kommunikationsmedium zuzugreifen, belegt immer das Telegramm mit der niedrigsten Priorität den Übertragungskanal. Deshalb wird vom zerstörungsfreien Buszugriff gesprochen.

<sup>21</sup> Siehe Echtzeitfähigkeit Fall 2 Abschnitt 2.4.2

TTCAN-System wird eine Systemmatrix für das Senden von Nachrichten für jeden Kommunikationsteilnehmer eingeführt.

### 3.4.2 Ausgewählte Kommunikationssysteme der Officekommunikation

Das Ethernet, mit seinem 1973 [3.13] entwickelten und nach der Norm IEEE 802.3 standardisiertem Übertragungsprotokoll TCP/IP, bildet die Grundlage für die Entwicklungsarbeiten am Industrial Ethernet.

Auf Ethernet basierende Kommunikationssysteme werden in der Regel in zwei Topologien ausgelegt. Die Bus- bzw. Linienstruktur ist typischerweise bei physikalischen Realisierungen mit Koaxialkabel ausgeprägt. Diese Struktur ist in der industriellen Kommunikation (Feldbusysteme) vorherrschend. Im Office-Bereich wurde von dieser Art der Verkabelung aus verschiedenen Gründen abgegangen. Dazu gehören unter anderem die Sicherheit der elektrischen Verbindung, sowie der umständlichen An- und Abkopplung von Teilnehmern.

Moderne Systeme<sup>22</sup> werden heute nach der Methode Twisted-Pair (verdrillte Zweidrahtleitungen) aufgebaut. Damit sind die Nachteile der Koaxialverkabelung umgangen worden. Topologisch lassen sich Ethernetnetze jetzt nur noch als Stern aufbauen. Es ist also ein zentrales Kommunikationsgerät (Hub oder Switch) erforderlich, welches die Kommunikation der Teilnehmer in seinem „Subnetz“ koordiniert und die Kommunikation zu anderen Subnetzen verwaltet. Im Office-Bereich stellt diese Architektur kein Problem dar. Im industriellen Bereich ist der Verkabelungsaufwand eine schwierigere Angelegenheit und führt in gewisser Weise zum Rückschritt in den zentralisierten Steuerungsaufbau. Für den industriellen Einsatz ist es sinnvoll, diese sternförmige Topologie durch bewährte Topologien aus dem Feldbusbereich zu ersetzen.

#### **Schlussfolgerungen:**

Ethernet muss sich hinsichtlich der Topologie möglichst wie ein standardisiertes Feldbussystem verhalten, um sich rasch im industriellen Einsatz durchsetzen zu können. Diese Feststellung betrifft nur die Art und Weise des Systemaufbaues (Kabel, Steckverbinder, Schutzklassen usw.), nicht jedoch die ohnehin geforderte Echtzeitfähigkeit der industriellen Steuerungstechnik.

### 3.4.3 Ausgewählte Kommunikationssysteme der Gerätetechnik

#### **I<sup>2</sup>C - Interface**

Das I<sup>2</sup>C-Interface wurde von der Firma Phillips in den 1980er Jahren für den Einsatz in der Konsumgüterindustrie entwickelt. Mit dieser Entwicklung wurde das Ziel verfolgt, die Anzahl von Leitungen für analoge und digitale Signale in räumlich abgeschlossenen Geräten zu be-

---

<sup>22</sup> 1 Gbit-Ethernet Übertragung elektrisch und optisch, 10 Gbit-Ethernet vorrangig optische Übertragung (gegenwärtig erfolgen Arbeiten am Standard IEEE 802.3an für elektrische Kommunikation[3.01])

grenzen. Mit dem Einsatz dieses Interfaces wurde der Trend zur Verteilung von Funktionalitäten in der elektronischen Gerätetechnik eingeleitet. Der erste Standard ist durch eine 7 Bit Adressierung mit 100 Kbit/s Übertragungsgeschwindigkeit charakterisiert. Mit dem Standard V1.0 von 1992 wurde die Adressierung auf 10 Bit erweitert und die maximale Baudrate auf 400 Kbit/s erhöht. Im Jahr 1998 erfolgte die Standardisierung zur V2.0 mit einer maximalen Baudrate von 3,4 Mbit/s. Die Erweiterung zum Standard V2.1 im Jahr 2001 enthält keine Änderungen hinsichtlich der Baudrate und der Adressierung.

Prinzip:	serielle synchrone Kommunikation über zwei Leitungen
Adressierung:	7 Bit für 112 Knoten (16 Knoten sind als Sonderadressen reserviert) 10 Bit für 1024 Knoten
Übertragungsrate:	100 Kbit/s im Standard-Mode 400 Kbit/s im Fast-Mode 3,4 Mbit/s im High-Speed-Mode
Netzausdehnung:	mit Repeater maximal 10 m
Teilnehmeranzahl:	Begrenzung „logisch“ durch Anzahl der Adressen (7 Bit bzw. 10 Bit) Begrenzung „elektrisch“ durch Lastkapazität von 400 pF
Konzept:	Multimasterfähiges Verfahren

Das I<sup>2</sup>C Interface eignet sich konzeptionell für den Aufbau räumlich begrenzt ausgedehnter echtzeitfähiger Steuerungssysteme auch für den Einsatz in der industriellen Steuerungstechnik. Es bildet die technische Grundlage für das Kommunikationsmodell des verteilten Steuerungssystems zur Synchronisation von Antriebssystemen nach Kapitel 5.

### **One-Wire-Interface**

Für den Einsatz in der Gerätetechnik wurde das On-Wire-Interface von der Firma Dallas Semiconductor entwickelt. Die Datenübertragung erfolgt bei diesem Konzept asynchron über eine Leitung. Auf dieses Kommunikationssystem soll nicht näher eingegangen werden, da die Realisierung verteilter Steuerungssysteme für die Gerätetechnik durch die Konzepte I<sup>2</sup>C und SPI in ausreichender Weise abgedeckt werden.

### **SPI-Interface**

Von der Firma Motorola wurde für die Gerätetechnik das SPI (Synchron Peripher Interface) entwickelt. Das Interface ist in 4-Draht-Technik ausgelegt. Die Kommunikation erfolgt im klassischen Master-Slave-Prinzip. Die Realisierung der Kommunikation in Kapitel 5 erfordert ein flexibleres Kommunikationskonzept (Multimasterfähigkeit). Deshalb soll dieses Kommunikationssystem an dieser Stelle nicht näher betrachtet werden.



### 3.5 Ausgewählte drahtlose Kommunikationssysteme

#### 3.5.1 Funkbasierte Kommunikationssysteme

In Abbildung 3.4 sind wichtige Vertreter von funkbasierten Systemen mit ihren wesentlichen technischen Daten gegenübergestellt, die zum gegenwärtigen Zeitpunkt Einzug in die industrielle Automation nehmen. Grundsätzlich müssen funkbasierte Kommunikationssysteme dabei hinsichtlich der Anforderungen in der industriellen Steuerungstechnik (siehe Abschnitt 2) ohne Nutzung zusätzlicher informationstechnischen Maßnahmen, als nicht echtzeitfähig eingestuft werden.

Protokoll	GSM	DECT	IEEE802.11x	433/868Mhz	Bluetooth
Frequenzband	900/1800 Mhz	1,88-1,90 Ghz	2,4/5,0 Ghz	433,05-434,79 868-870 Mhz	2,4 Ghz
Sendeleistung	max. 2 W	250 mW	50 mW – 1 W	10/25 mW 500 mW	1 mW 2,5 mW 100 mW
Datenrate	9,6 Kbits/s	1,15 Mbit/s	11/54 Mbit/s	19,2 Kbit/s	1,00 Mbit/s
Einsatzgebiete	Handy/SMS	Telefonie	WLAN	Allgemeiner Datenfunk	WPAN
Ausdehnung	35 Km	300 m	20-150 m	> 1 Km	10/20/100 m
Supporter	Europa/Asien	Siemens, Phillips, etc.	IEEE, IBM 3Com, Dell, etc.	allgemein	Nokia, 3Com, Intel, Ericsson Microsoft, etc.
Organisation	gsmworld.org	dectweb.com	ieee.org		bluetooth.com

Abbildung 3.4: Wichtige funkbasierte Systeme für den industriellen Einsatz [3.11] (Auszug)

Funkbasierte Kommunikationssysteme können als stationäre<sup>23</sup> oder mobile<sup>24</sup> verteilte Systeme ausgelegt werden.

<sup>23</sup> Funkkommunikation erfolgt durch einen stochastischen Zugriff auf das Kommunikationsmedium. Unter der Bedingung „stationär“ also ortsunveränderlich bestehen im Verhalten entsprechende Analogien zu den Zugriffsverfahren wie CSMA/CD/CA, welche z.B. bei Ethernet und CAN angewendet werden. Für ein vorhersagbares Zeitverhalten in verteilten funkbasierten Systemen können die in Kapitel 3 analysierten Verfahren für die Realisierung echtzeitfähiger Kommunikationssysteme mit stochastischen Zugriffsverfahren am Beispiel des Industrial Ethernet angewendet werden.

<sup>24</sup> Siehe Forschungsarbeiten „Mobilfunkbasiertes Steuerungskonzept für das Flottenmanagement in Tagebauen und Minen“ [2.52]

### **Schlussfolgerungen:**

Funksysteme für den Einsatz in echtzeitfähigen verteilten stationären Automatisierungssystemen, unter der Bedingung der Gewährleistung einer störsicheren Kommunikation, müssen zwei wesentliche Forderungen erfüllen:

- Die Ausdehnung muss den gängigen industriellen Anforderungen entsprechen.
- Die Systeme müssen eine hohe Bruttodatenübertragungsrate aufweisen, um Verfahren nach Abschnitt 3.6.4 anwenden zu können

Die vorgestellten funkbasierten Systeme können dabei für ihre Industrietauglichkeit wie folgt bewertet werden:

Die GSM-basierte Kommunikation lässt sich aufgrund der globalen Ausrichtung des Funksystems mit dem Echtzeitverhalten des Internets vergleichen. Sie sollte aus diesem Grund in der industriellen Automation nur für Bedien- und Beobachtungsfunktionalität eingesetzt werden, wie es zum gegenwärtigen Zeitpunkt schon realisiert wird (z. B. SMS für Wartungsarbeiten und Fehlermeldungen).

Die Funksysteme 433/868 Mhz und Bluetooth sind hinsichtlich der Größe der Netzausdehnung einerseits und durch die geringe Bruttodatenrate andererseits nicht die optimale Wahl für die Realisierung echtzeitfähiger Funkkommunikation im industriellen Bereich.

Die Standards DECT und IEEE 802.11xx bieten die aussichtsreichsten Voraussetzungen für die Echtzeitkommunikation, unter Nutzung der Verfahren Zeitschlitz- und Zeitsynchronisation, in lokal begrenzten funkbasierten Systemen. Von beiden Systemen wird eine für industrielle Zwecke ausreichende Netzausdehnung spezifiziert. Die Bruttobandbreite erweist sich hinsichtlich der künstlichen Reduzierung der Nettobandbreite durch den Einsatz der Synchronisationsmechanismen ebenfalls als ausreichend.

### **3.5.2 Optische Kommunikationssysteme**

Optische drahtlose Datenübertragungssysteme wie z.B. IRDA<sup>25</sup> oder Laser-Link-Kommunikation werden an dieser Stelle nur der Vollständigkeit halber genannt. Das Haupteinsatzgebiet der Systeme liegt zum gegenwärtigen Zeitpunkt in der Office-Kommunikation. Untersuchungen des Echtzeitverhaltens hinsichtlich der industriellen Kommunikation wurden aufgrund der Aufgabenstellungen nicht vorgenommen.

---

<sup>25</sup> Infared Date Association  
Verfahren zur Informationsübertragung mit begrenzter Reichweite durch moduliertes Infrarotlicht

## 3.6 Industrial Ethernet

### 3.6.1 Einführung

Ethernettechnologien und echtzeitfähige industrielle Steuerungssysteme ein unlösbarer informationstechnischer Widerspruch? Mit Real-Time-Ethernet in die moderne auf Informationsverarbeitung basierende Automatisierungstechnik?

**Thematisiert wurden diese grundlegenden Fragestellungen spätestens seit der HMI 2000 durch die Firma Jetter mit der These „Das Netz ist die Steuerung“.**

Die moderne Informationsverarbeitung in Systemen der Automatisierungstechnik ist durch einen hierarchischen Aufbau gekennzeichnet. Dabei existieren zum gegenwärtigen Zeitpunkt zwei Formen der Hierarchiepyramide:

- Die klassische Hierarchiepyramide der Automatisierungstechnik mit ihrer Ausprägung in den Ebenen: Leitstand, Zellensteuerung, Steuerung und Regelung sowie der Sensorik und Aktorik, welche aus technischer Sicht das eigentliche Interface zum zu steuernden Prozess darstellen.
- Das abstraktere aus nur den drei Ebenen MES, ERP und PCS<sup>26</sup> bestehende Modell. Bei genauerer Betrachtung der Hierarchieebenen kann festgestellt werden, dass die zwei unteren Ebenen sowie Teile aus der Zellensteuerungsebene zur PCS-Ebene zusammengefasst werden. Die Ebenen Zellensteuerung und ein Teil der Leitstandsebene werden in die ERP-Ebene integriert. Und schließlich wird aus den heutigen Anforderungen der Produktionsautomatisierung eine Ebene zusätzlich aufgesetzt, die ERP-Ebene.

Das klassische Hierarchiemodell lässt sich durch zwei grundlegende gegensätzlich gerichtete Merkmale charakterisieren:

- steigende Ebene -> steigender Kommunikationsbedarf -> sinkende Anforderung an die Einhaltung von Zeitschranken
- sinkende Ebene -> sinkender Kommunikationsbedarf -> steigende Anforderung an die Einhaltung von Zeitschranken

Die Kommunikation in den Ebenen Steuerung/Regelung und Sensorik/Aktorik wird nach dem jetzigen Stand der Technik mit Feldbussystemen<sup>27</sup> ausgeführt, welche zu Beginn der 1990er Jahre, die bis dahin bestehenden zentralistischen Automatisierungsstrukturen aufbrachen. Mit dem Einsatz von Feldbussen verschiedener Hersteller können nach Abschnitt 2 alle gängigen

---

<sup>26</sup> Process Control System z.B. PCS7 von der Siemens AG

<sup>27</sup> Unter Feldbussystemen können verschiedene physikalische Bussysteme für die Automatisierungstechnik, der Fertigungstechnik, der Gebäudeautomation und der Automotivtechnik verstanden werden. Eine wesentliche Eigenschaft der Systeme ist die zeitliche Vorhersagbarkeit des Kommunikationsverhaltens durch entsprechende Zugriffsverfahren auf das Übertragungsmedium wie z.B. TDMA und andere.

Anforderungen der Echtzeitfähigkeit eines technischen Systems zur Steuerung und Regelung von Prozessen aus der Verfahrens- und Fertigungstechnik abgedeckt werden. Für die Lösung automatisierungstechnischer Aufgabenstellungen werden von den eingesetzten industriellen Kommunikationssystemen verschiedene Zugriffsverfahren, Protokollspezifikationen, Übertragungsraten sowie Übertragungsmedien genutzt.

Die Realisierung der Kommunikation in den Ebenen Leitstand und Zellensteuerung erfolgt im Normalfall mit Kommunikationssystemen, welche nicht speziell auf Echtzeitkommunikation ausgelegt sind. In sehr vielen Einsatzfällen wird das im Office-Bereich dominierende Kommunikationssystem Ethernet mit den Protokollen TCP/IP/UDP eingesetzt. Analysen nach [3.18] besagen, dass in ca. 80-90 % solcher Einsatzfälle Ethernet zur Anwendung kommt. Ein großer Teil des verbleibenden Anwendungsfälle wird durch das ATM-Verfahren abgedeckt.

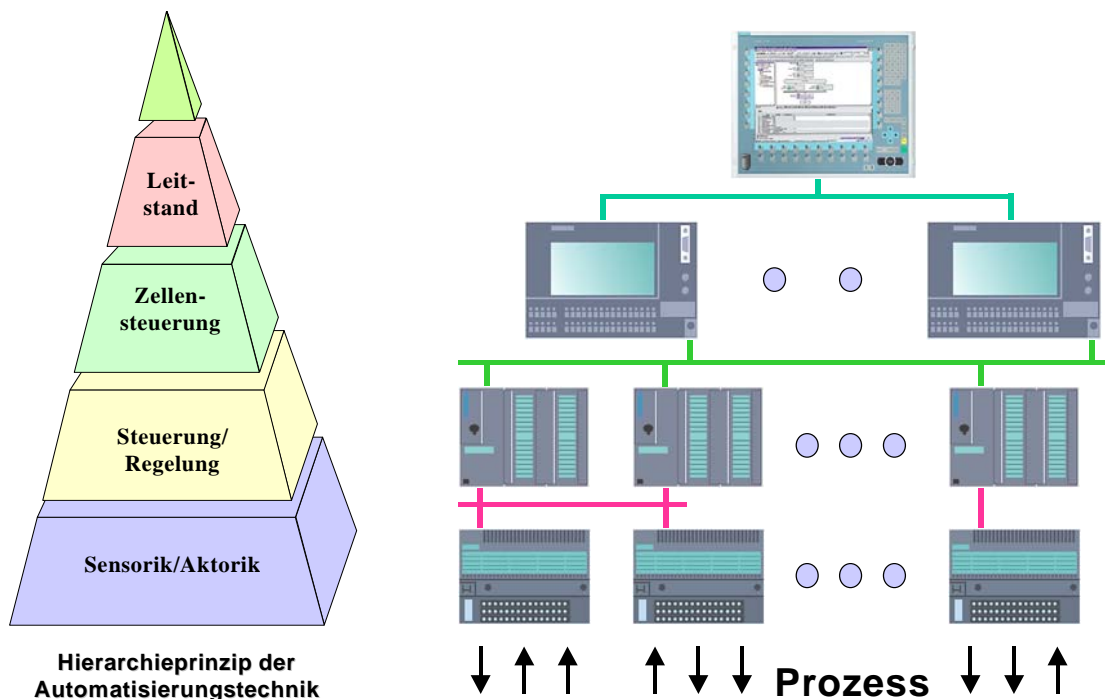


Abbildung 3.5: Klassisches Hierarchieprinzip der Automatisierungstechnik

Um eine durchgängige Kommunikation aus der ERP-Ebene über die MES-Ebene zur PCS-Ebene, letztendlich auf jeden einzelnen Sensor bzw. Aktor, zu gewährleisten, sind nach dem jetzigen Stand der Technik informationsverarbeitende Geräte erforderlich, welche die verschiedenen Kommunikationssysteme miteinander koppeln. Die Verbindung der Netze kann im einfachsten Fall über Gateways bis hin zu Proxy-Servern erfolgen. Dabei erfolgt eine vollständige Umsetzung des Protokolls, sowie die fast immer erforderliche Anpassung der Übertragungsgeschwindigkeit (Realisierung bis in die Ebenen 7 des OSI-Modells).

Dabei ergeben sich folgende wesentliche Anforderungen:

- Die Kopplung in den Bereich der PCS-Ebenen aus einem nichtechtzeitfähigen Kommunikationssystem in ein hart bzw. determiniert hart arbeitendes Kommunikationssystem muss zeitlich gezielt und zerstörungsfrei erfolgen. D. h., ein Zugriff aus den Ebenen ERP/MES muss sich in jedem Fall dem Zeitmanagement der Prozesssteuerungsebene unterordnen. Die erforderliche Protokollumsetzung macht leistungsfähige Rechnerkomponenten erforderlich, da die Durchlaufzeiten in Gateways und Switches starken Einfluss auf die Performance des Gesamtsystems ausüben.
- Da in der modernen Industrieautomation zunehmend die Notwendigkeit besteht, ständig über alle erforderlichen Information des zu steuernden Prozess verfügen, ist eine durchgängige Kommunikation von ganz oben bis nach ganz unten in der Hierarchiepyramide zwingend erforderlich. Die Kommunikation soll dabei so transparent wie möglich erfolgen. Dazu wird von den führenden Herstellern von Komponenten für die Automatisierungstechnik das aus der Office-Welt seit Jahren bewährte Ethernet favorisiert.
- Der Einzug der Dezentralisierung in die Automatisierungstechnik führte nach [3.22] zur Existenz von über 40 bis zu 150 unterschiedlichen Systemen verschiedener Hersteller. Wenige Systeme wurden erst später in der europäischen Feldbusnorm standardisiert. Solche Entwicklungen sollten bei der Einführung von Industrial Ethernet möglichst vermieden werden.
- Zum gegenwärtigen Zeitpunkt zeichnen sich verschiedene Konzepte und Verfahren unterschiedlicher Hersteller und Gremien ab, welche sich mit der Aufgabenstellung „Industrial Ethernet“ beschäftigen. Um den besagten Wildwuchs aus dem Feldbusbereich vorzubeugen, wurde die IAONA ins Leben gerufen, welche sich als Non-Profit-Organisation der Koordinierung der Arbeiten zum aktuellen Thema widmet.

### **3.6.2 Aufgabenstellung**

Aus den bisherigen Vorbetrachtungen lassen sich folgende grundlegende Aufgabenstellungen ableiten:

- Eine Analyse der Problemstellung Ethernet und Echtzeitfähigkeit sowie der Vergleich mit ausgewählten speziell auf die Echtzeitproblematik ausgelegten Feldbussystemen.
- Ein Vergleich der wichtigsten Ansätze Ethernettechnologien für echtzeitfähige industrielle Steuerungsaufgaben einsetzbar zu machen.
- Die Bewertung der Ansätze ethernetbasierter Kommunikationssysteme für die Synchronisation von technischen Prozessen, insbesondere der Bewegungssynchronisation.

### 3.6.2.1 Bestehende Ausgangssituation für Industrial Ethernet

- Es sollen offene Standards aus der Bürokommunikation verwendet werden.
- Ethernet mit den wichtigen Protokollen TCP/IP und UDP.
- Ethernet ist wegen des CSMA/CD- Zugriffsprinzips nicht echtzeitfähig.
- Ethernet weist bei geringen Nutzdatenmengen ein schlechtes Verhältnis von Brutto- und Nettodatenrate auf (Protokolleffizienz).

### 3.6.2.2 Gestellte Anforderungen für Industrial Ethernet

- Das Netz muss vollständig geswitched sein.
- Die Kommunikation muss Full Duplex erfolgen.
- Es müssen spezielle Eingriffe in den Protokollstacks der Ethernetkommunikation vorgenommen werden.

Von der Arbeitsgruppe JTWG<sup>28</sup> der IOANA wurden die Anforderungsprofile an echtzeitfähige Ethernetkonzepte in den folgenden vier Klassen spezifiziert [3.67][3.73]. In [3.14] werden die Forderungen an echtzeitfähiges Ethernet in drei QoS Klassen definiert. Nach [3.59] erfolgt die Klassifizierung ausgewählter Ethernet-Realtime-Verfahren im vier Quadrantenprinzip.

Klasse	1	2	3	4
Hardware	Standard	Standard	Standard	Speziell
Software	Standard	Speziell	Speziell	Speziell
Protokoll	Standard	Standard	Speziell	Speziell
Jitter	> 1 ms	100 µs - 3 ms	10 µs - 400 µs	0,5 µs - 15 µs

Abbildung 3.6: Klassifizierung der Echtzeitanforderungen bei Industrial Ethernet (IAONA) [3.67]

<sup>28</sup> Joint Technical Working Group

Aus der Abbildung 3.6 geht deutlich hervor, dass mit zunehmender Forderung der präzisen Einhaltung des Zeitmanagements in Kommunikationssystemen, der Aufwand für die Implementierung der Gerätekomponenten zunimmt. So lassen sich Geräte für Automatisierungssysteme in den Klassen 1 und 2 mit vergleichsweise vertretbarem Aufwand realisieren. Dazu lassen sich Standard-Ethernetcontroller einsetzen. Bei etwas stärkeren Forderungen an die zeitliche Konstanz im Gesamtsystem müssen jedoch die Protokollstacks für UDP und TCP/IP durch spezielle, laufzeitoptimierte Implementierungen ersetzt werden. Das im Standard IEEE 802.3 festgelegte Ethernettelegramm bleibt dabei erhalten. Die Klassen 1 und 2 nach der Festlegung der IAONA entsprechen den typischen Leistungswerten von Feldbussystemen wie beispielsweise PROFIBUS, Interbus oder CAN.

Ein wesentlich höherer Aufwand für die Realisierung industrietauglicher Ethernetkommunikation muss in den Klassen 3 und 4 betrieben werden.

Für derartige Anforderungen ist ein reiner Austausch des Protokollstacks nicht mehr ausreichend. Dabei versuchen die Hersteller Echtzeit nach Klasse 3 noch ohne spezielle Protokoll-ASIC's zu realisieren, indem spezielle optimierte Protokolle in den höheren Schichten des OSI-Modells eingesetzt werden. In [3.17] wird davon ausgegangen, dass Motion Control Anwendungen<sup>29</sup> mit reinen Softwarelösungen nicht mehr realisiert werden können<sup>30</sup>.

In Klasse 4 erfolgt dann als logische Konsequenz der vollständige Austausch aller drei Komponenten der Firmware. Aus der Sicht der Hardware lässt sich die Aufgabenstellung auf der einen Seite durch spezielle Großserien-ASIC's lösen. Auf der anderen Seite können Standard-FPGA mit entsprechenden VHDL-Designs, welche vom Hersteller zur Verfügung gestellt werden, das angepasste Ethernetprotokoll abwickeln. Um die Durchlaufzeiten für die implementierten Protokollstacks zu minimieren, werden diese für solche Anwendungsfälle in Hardware realisiert. Verschiedene Ansätze Ethernet nach dieser Einstufung echtzeitfähig zu machen setzen auf ein spezielles, angepasstes Ethernettelegramm. Damit geht der bei den drei Vorgängerklassen wesentliche Vorteil in der Schicht 2 des OSI-Modells standardisierten einheitlichen Telegramms verloren und es entwickeln sich wieder abgeschlossene Systeme ähnlich wie bei den Feldbussen. Es muss dabei aber angemerkt werden, dass Automatisierung nach Klasse 4 der IAONA-Spezifikation hinsichtlich seiner geforderten zeitlich präzisen Abarbeitung von Algorithmen ohnehin nur in lokal begrenzten Ausdehnungen realisiert werden dürfte. Der typischste Einsatzfall für solche Arten von verteilten Systemen sind Anwendungen wie Synchronisation von Antriebssystemen z.B. in der Druckmaschinenindustrie.

### 3.6.2.3 Schlussfolgerungen

Nach der Klassifizierung der IAONA können Automatisierungssysteme, welche zum heutigen Zeitpunkt durch die klassische Feldbustechnik realisiert werden, mit Ethernettechnologien rea-

---

<sup>29</sup> D. Herian Siemens AG in IEE 5/2003

<sup>30</sup> Spezielle Ethernetcontroller z.B. bei PROFINET, EtherCAT und SERCOS-III, FPGA basierte Hardware bei Powerlink

lisiert werden. Die Bandbreite des Aufwandes reicht dabei von reinen Softwarelösungen mit Standardethernetcontrollern bis hin zu speziell entwickelten Protokollchips. Bei sehr hohen Anforderungen an die zeitliche Konstanz der Kommunikation im Steuerungssystem ist eine steigende Inkompatibilität zum ursprünglichen Ethernetstandard unvermeidbar. Für diese Anwendungsfälle ist eine Analogie zu den bisherigen Feldbussystemen erkennbar. So können z.B. Antriebsregelungen bis zu einem gewissen Grad mit Feldbussen realisiert werden. Wird die Aufgabenstellung komplexer, so muss in diesem Bereich auf spezielle Systeme z.B. SER-COS-III ausgewichen werden. Die Klassifizierung des Echtzeitverhaltens moderner Automatisierungstechnik durch die IAONA kann als allgemeingültig betrachtet werden und sollte nicht nur auf die Kommunikation mit Ethernet beschränkt werden. Die vorgenommene Eingruppierung hinsichtlich des Zeitverhaltens kann auf alle Arten von verteilten Steuerungssystemen angewendet werden. Dazu gehören insbesondere das System IEEE 1394 a/b, sowie spezielle Systeme wie Flexray aus dem Automotivsektor.

### **3.6.3 Wichtige Protokolle des Ethernet**

#### **3.6.3.1 Address Resolution Protocol (ARP)**

Das Address Resolution Protocol (ARP) ist in der RFC<sup>31</sup> 926 veröffentlicht und dient der Konvertierung von Software-Adressen (IP-Adressen) in Hardwareadressen (MAC-Adressen). Dabei wird vor einer Übertragung von Daten in einem ethernetbasierten Netzwerk vom IP-Protokoll das Vorhandensein eines Eintrages in der ARP-Adresstabelle geprüft. Existiert kein Eintrag, erfolgt durch Anfrage an alle Kommunikationsknoten die Hardware-Adresse zur bekannten Destination-IP. Nur der Kommunikationsknoten mit dem Eintrag zur IP-Adresse antwortet auf diese Anfrage. Danach erfolgt der Eintrag in die ARP-Adresstabelle.

#### **3.6.3.2 Internet Protocol (IP)**

Das IP ist ein kommunikationsloses Protokoll, welches lediglich für jedes Datagramm den Weg zum Ziel bestimmt und es absendet. Dabei wird von IP keine Überprüfung des Empfangs vorgenommen, so dass Datagramme durchaus auch verloren gehen können.

Das Protokoll ist durch folgende Eigenschaften charakterisiert:

- Routing von Datagrammen durch das Netz
- Fragmentierung und Zusammensetzen von Datagrammen
- Definition des Adressierungsschemas
- Übermittlung der Daten von der Transportebene zur Netzwerkschicht

---

<sup>31</sup> Request for Comment



### 3.6.3.3 Transmission Control Protocol (TCP)

Aufbauend auf dem verbindungslosen, nicht sicheren Dienst der Schicht 3, stellt das Transmission Control Protocol als Protokoll der Schicht 4 einen verbindungsorientierten, sicheren Transportdienst zur Verfügung. Die Sicherheit dieser Verbindung wird durch Rückmeldungen und wiederholtes Versenden fehlerhafter Datenblöcke erreicht.

Wie bei allen verbindungsorientierten Diensten muss zuerst eine Verbindung aufgebaut werden, bevor die Nutzdaten transportiert werden können. Diese muss danach abgebaut werden. Aus logischer Sicht wird über TCP ein unstrukturierter Strom von Datenbytes transportiert, der für die Übertragung in einzelne nummerierte Segmente aufgeteilt wird. Dabei sollte die Größe der Segmente auf die Rahmenbedingungen der darunter liegenden Schicht abgestimmt sein.

Das Protokoll ist durch folgende Eigenschaften charakterisiert:

- datenstromorientiert
- verbindungsorientiert
- zustandsorientiert
- Handshakeverfahren

Das Protokoll ist sehr gut für die Kommunikation in nicht echtzeitfähigen Kommunikationssystemen einsetzbar. Durch die Verbindungsorientierung können Pakete nicht verloren gehen oder sich überholen, da in jedem Fall auf eine Bestätigung vom Kommunikationspartner gewartet wird. Die Verbindungsorientierung hat allerdings den gravierenden Nachteil, keine Broadcast - Nachrichten versenden zu können. Broadcast-Kommunikation stellt die grundlegende Voraussetzung isochron arbeitender Kommunikationssysteme<sup>32</sup> im Feldbereich dar. Das TCP Protokoll ist in der RFC 793 spezifiziert.

### 3.6.3.4 User Datagram Protocol (UDP)

Das User Datagram Protocol stellt einen verbindungslosen, unzuverlässigen Transportdienst in der Schicht 4 zur Verfügung, welcher der Anwendung im Wesentlichen nur die Funktionalität der darunter liegenden Schicht 3 offeriert. Im Gegensatz zu TCP wird allerdings auf Rückmeldungen aller Art verzichtet, so dass für den Sender zum Beispiel auch unbekannt bleibt, ob alle Pakete angekommen sind. Der große Vorteil des Protokolls liegt in der verbindungslosen Kommunikation, mit der die Möglichkeit besteht Broadcast-Nachrichten im Subsystem zu versenden. Mit Broadcast-Nachrichten kann in asynchron arbeitenden Kommunikationssystemen ein einheitliches Zeitmanagement (Zeitschlitzverfahren) realisiert werden. Außerdem zeichnet sich das Protokoll durch einen geringen Protokoll-Overhead aus.

---

<sup>32</sup> z. B. äquidistante Betriebsweise bei PROFIBUS - DP oder der Time Triggered Protokollaufsatz von CAN

Das Protokoll ist durch folgende Eigenschaften charakterisiert:

- verbindungslos
- zustandslos

Das UDP Protokoll ist in der RFC 768 spezifiziert.

Als ein wichtiges Problem beim Einsatz von Ethernettechnologien in der industriellen Steuerungstechnik kann die begrenzte rechentechnische Verarbeitungsleistung der dezentralen Peripheriekomponenten angesehen werden. Die gegenüber Feldbustechnologien wesentlich aufwändigeren Protokollstacks können die Embedded-Hardware dieser Komponenten eines Automatisierungssystems zum Teil erheblich belasten. Beim Vergleich der Industrial Ethernettechnologien ist eindeutig erkennbar, dass von vielen Systemen die Bereitstellung von UDP- und TCP-Funktionalität in Hardware realisiert werden soll. Die Durchlaufzeiten können somit erheblich reduziert werden.

### 3.6.4 Probleme und Verfahren für die Realisierung deterministisches Ethernet

Um die Vorteile des Ethernetkonzepts für die moderne Industriekommunikation nutzen zu können, bieten sich einige wichtige Verfahren für die Synchronisation der Kommunikation in verteilten echtzeitfähigen Systemen an. Das Einsatzgebiet einzelner Verfahren ist dabei nicht zwingend auf Kommunikationssysteme mit CSMA/CD/CA Zugriffsprinzipien begrenzt. Auch äquidistant arbeitende Steuerungssysteme nach dem Master-Slave-Verfahren<sup>33</sup> arbeiten mehr oder weniger nach diesen Verfahren, um ein präzises Zeitmanagement im Gesamtsystem zu gewährleisten.

#### 3.6.4.1 Methode der Lastbegrenzung

Als ein sehr frühzeitiger Ansatz das zeitliche Verhalten von Ethernet vorhersagbar zu machen, stellt die Begrenzung der Busauslastung auf eine genau zu definierende prozentuale Größe dar. Mit dieser Methode kann mittels mathematischer Statistik eine Aussage über eine mittlere Reaktions- und Bearbeitungszeit für eine Kommunikationsbeziehung getroffen werden.

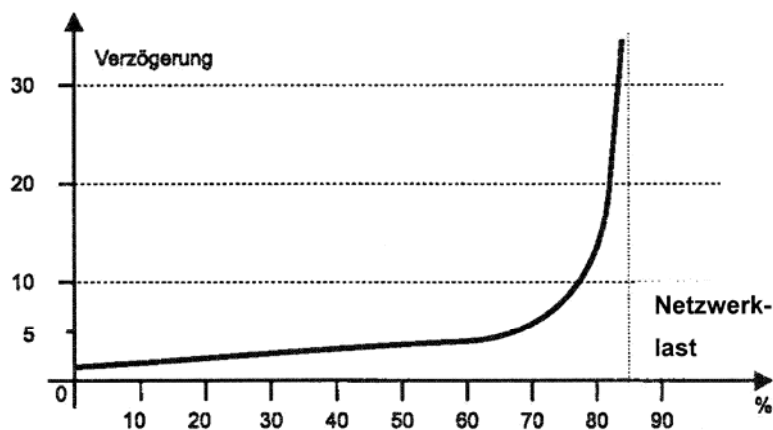


Abbildung 3.7: Relative Zeitverzögerung als Folge der Busbelastung bei Ethernet [3.07]

<sup>33</sup> Gegenwärtig im Feldbusbereich vorherrschendes Prinzip z.B. PROFIBUS mit dem Profil DPV1

In [3.02] wird davon ausgegangen, dass bei einer Begrenzung der Netzbelastung auf maximal 10 % Shared Ethernet praktisch als pseudo-deterministisch betrachtet werden kann. Gegenüber Ethernet-TCP/IP ist das Verfahren „predictable Ethernet“ hinsichtlich Hardware und Protokollspezifikation identisch und zueinander kompatibel.

### 3.6.4.2 Zeitschlitzverfahren

Bei diesem Verfahren erfolgt die Kommunikation synchron in festgelegten Zeitrastern. Die informationstechnische Basis bilden spezielle Telegramme mit Multicast-Charakter, welche von einem als Zeitmaster arbeitenden Kommunikationsknoten an alle im Subnetz vorhandenen Kommunikationsknoten verschickt werden.

Das Zeitschlitzverfahren ist dabei nicht nur auf Kommunikationssysteme beschränkt, welche nach dem CSMA/CD-Zugriffsverfahren arbeiten. In äquidistant arbeitenden Steuerungssystemen aus dem aktuellen Feldbusbereich wird das Zeitmanagement ebenfalls mit diesem Konzept realisiert. Dazu wird z.B. beim SIMATIC-System ein Global-Control-Telegramm (siehe Abbildung 2.1) eingesetzt, mit dem alle Slaveknoten vom Masterknoten synchronisiert werden. Der wesentlichste Nachteil des Verfahrens liegt in der Tatsache begründet, dass in jedem Fall Laufzeiten in den Kommunikationsknoten berücksichtigt werden müssen.

Die Durchlaufzeit setzt sich aus den Komponenten

$$T_{\text{isochron}} = T_{\text{channel}} + T_{\text{transmit}} + T_{\text{receive}} + n * T_{\text{switch}} \quad (3.1)$$

zusammen.

Aus Gleichung (3.1) geht hervor, dass die Präzision der Zeitsynchronisation wesentlich von den Schwankungen der Durchlaufzeiten der beteiligten Kommunikationsgeräte abhängt. In kaskadierten verteilten echtzeitfähigen Systemen erhöht sich die Gesamtzeit mit dem Hinzufügen jeder weiteren Ebene, da die Durchlaufzeiten durch Switches eine wesentliche Einflussgröße darstellen.

Das Zeitschlitzverfahren eignet sich gut für die Prozesszeitsynchronisation in lokalen industriellen Kommunikationssystemen mit den Zugriffsverfahren CDMA/CD/CA. Für die Gewährleistung einer hohen zeitlichen Konstanz im Gesamtsystem müssen die Anzahl der Kaskadierungsstufen so klein wie möglich sein, im günstigsten Fall gleich Null gehalten werden.

### 3.6.4.3 Zeitsynchronisationsverfahren

Dieses Verfahren basiert auf präzise synchronisierten, autonom arbeitenden Uhren in jedem eingesetzten Kommunikationsknoten. Das Precision Time Protocol (PTP) wird durch den Standard IEEE 1588 “Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems“ [3.75], der im Jahr 2002 veröffentlicht wurde, beschrieben. Mit Hilfe dieses Protokolls kann in multicastfähigen Netzwerken eine präzise Uhrensynchronisation erreicht werden. Der IEEE-Standard ist geeignet für kleine Netze und benötigt

nur wenig Verwaltungs- und Rechenaufwand. Die beste<sup>34</sup> Uhr im Netzwerk wird als Referenz (Masterclock) definiert. Dieser Zeitserver sendet per Multicast Synchronisationsinformationen an die anderen Teilnehmer. Der im Telegramm übermittelte Zeitstempel dient als Zeitbasis für die Clients. Über eine Delay Request bestimmen die Knoten mit Hilfe der Masterclock die Verzögerung durch die Netzwerkübertragung. Symmetrische Übertragungsverzögerungen<sup>35</sup> lassen sich auf diesem Wege beseitigen. Für die exakte Bestimmung der Sende- und Empfangszeitpunkte muss eine spezielle Hardware eingesetzt werden. Sie filtert alle eingehenden Telegramme und versieht PTP-Telegramme mit einem Zeitstempel, der später vom Protokolltreiber ausgewertet wird. Mit dem PTP ist eine Genauigkeit von 1  $\mu$ s erreichbar. Voraussetzung ist allerdings, dass die Synchronisation ständig wiederholt wird<sup>36</sup>. Für kaskadierte Netzwerke sind Verfahren mit Boundary- und Grandmasterclocks spezifiziert, die das zeitliche Verhalten von Switches und Hubs beherrschbar machen. Für die Nutzung in Netzwerken von Automatisierungssystemen wird statt der Boundaryclock die Verwendung von PTP-Bridges<sup>37</sup> diskutiert, da diese keine zusätzliche Regelung einführen, sondern nur die Durchlaufzeit der Telegramme durch einen Switch messen und das Telegramm entsprechend korrigieren.

Wesentlicher Vorteil des Verfahrens:

Unterschiedliche Signallaufzeiten auf dem Kommunikationsmedium sowie schwankende Durchlaufzeiten der Protokollstacks zur Bearbeitung der Kommunikationsbeziehungen nehmen keinen Einfluss auf das einheitliche Zeitmanagement im verteilten System.

Wesentliche Merkmale des Verfahrens:

- Zeitkonstanz (Jitter) ist von der eingesetzten Hardware abhängig.
- Quarzstabil mit Nachsynchronisation über DCF77<sup>38</sup> (Zeitliche Konstanz ist von der Toleranz des Quarzes abhängig).
- Atomstabil durch Einsatz der UTC in GPS-Modulen (Zeitbasis wird aus dem GPS-Signal abgeleitet mit einer theoretischen Toleranz (Jitter) von +/- 200 ns, praktisch von +/- 1  $\mu$ s) [2.18].

Um Jitter kleiner 1  $\mu$ s zu realisieren reichen diese Maßnahmen allein noch nicht aus. So wird von PROFINET IO die Uhrzeitsynchronisation durch das PTCP-Protokoll nach der IEEE 61158 durchgeführt, welches als Ergänzung in die IEEE 1588 integriert werden wird<sup>39</sup>. Als grundlegende Erweiterung des Verfahrens besteht in der zusätzlichen Berücksichtigung der realen Laufzeiten im Switch sowie der realen Laufzeiten auf der Leitung (bei elektrischer als auch optischer Übertragung muss eine Laufzeit des Signals von ca. 0,5  $\mu$ s bei einer Segment-

---

<sup>34</sup> bezüglich Drift und Gangabweichung

<sup>35</sup> Hin- und Rückweg einer Meldung verursacht die gleiche zeitliche Verzögerung

<sup>36</sup> mit 1/64 bis 1 Hz

<sup>37</sup> wird auch als Bypass Clock bezeichnet

<sup>38</sup> Langwellensender mit dem Rufzeichen DCF77 in Mainflingen mit der Aufgabe der Aussendung eines präzisen Zeitsignals

<sup>39</sup> Popp Manfred, Expertengespräch PROFINET IO, Siemens AG Standort Fürth, September 2005

länge von 100 m angenommen werden). Dazu werden zusätzlich von zwei benachbarten PROFINET IO Komponenten die genauen Verzögerungszeiten gemessen.

### 3.6.4.4 Bewertung von Zeitsynchronisationsprotokollen

Eine präzise Zeitbasis ist in einem auf serieller Kommunikation basierendem verteilten Automatisierungssystem von grundlegender Bedeutung. In automatisierungstechnischen Spezialgebieten wie der Antriebssynchronisation und der Messtechnik stellt eine hochgenaue Zeitbasis im Gesamtsystem sogar eine zwingend notwendige Voraussetzung dar.

Für die Synchronisation von Zeiten in verteilten Systemen sind verschiedene Verfahren und Systeme entwickelt worden bzw. befinden sich in ständiger Weiterentwicklung. Wichtige leitungs-basierte Systeme sowie ein funkbasierteres System sind in der Abbildung 3.8 mit wesentlichen Merkmalen gegenübergestellt.

	NTP/SNTP	PTP	PTCP	GPS
Ausdehnung	Wide Area	Subnetze	Subnetz	Wide Area
Kommunikation	IP/UDP	IP	IEEE802.3	Satellit
Präzision	< 10 ms	< 1 ms	< 1 µs	< 1 µs
Realisierung	Software	Soft&Hardware	Hardware	Receiver
Übertragung	Internet	Intranet	Ethernet	Funk
Standardisierung	RFC1305 RFC2030	IEEE1588	IEEE1588	

- SNTP/NTP : (Simple) Network Time Protocol
- PTP : Precision Time Protocol
- PTCP : Precision Transparent Clock Protocol
- GPS : Global Positioning System

Abbildung 3.8: Wichtige Parameter von Zeitsynchronisationsprotokollen

Die Systeme und Protokolle unterscheiden sich im Wesentlichen in der Netzausdehnung und der zeitlichen Präzision der Uhrensynchronisation. Die grundlegende Gemeinsamkeit besteht darin, dass alle Systeme und Verfahren auf Multicast-Kommunikation durch Synchronisationsnachrichten basieren.

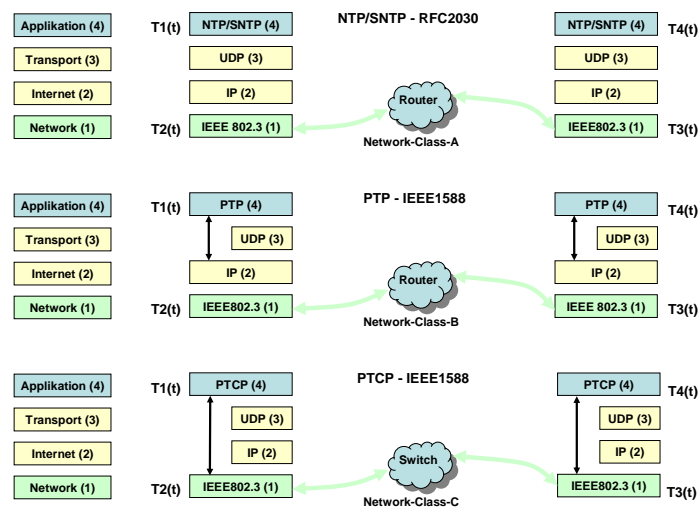


Abbildung 3.9: Klassifizierung leitungsgebundener Verfahren nach dem DOD-Modell

Obwohl NTP/SNTP auf Internettechnologien aufsetzt und PTP in der IEEE1588 nur für Ethernet-Kommunikation spezifiziert wurde sind die Prinzipien in jedem anderen multicastfähigen Netzwerk anwendbar. Die Übernahme der PTCP-Ergänzung in die IEEE1588-Norm erfolgt zum gegenwärtigen Zeitpunkt. In der Abbildung 3.9 sind die wichtigsten leitungsgebundenen Verfahren in das DOD-Modell eingeordnet.

Die Synchronisation zwischen PTP-Master<sup>40</sup> und PTP-Slave<sup>41</sup> erfolgt durch vier Telegramme mit Multicastcharakter. Im ersten Teil wird der Offset zwischen PTP-Master und PTP-Slave durch das wiederholte Senden von SYNC- und FOLLOW-UP- Telegrammen korrigiert. Im zweiten Teil erfolgt die Kompensation der Laufzeit zwischen PTP-Master und PTP-Slave durch die Telegramme DELAY-REQUEST und DELAY-Response. Im Anhang „Zeitsynchronisation in ethernetbasierten verteilten Systemen auf Basis der IEEE 1588“ sind die Kommunikationsbeziehungen an den Beispielen Slave-Uhr geht vor und Slave-Uhr geht nach schematisch dargestellt.

### 3.6.4.5 Codierungsverfahren von Fast Ethernet

Fast Ethernet wird unter Nutzung einer Kombination aus einer 4B5B Codierung mit einer MLT3 Codierung über das physikalische Medium übertragen. Mit der 4B5B Codierung werden 4 Nutzdatenbits durch 5 zu übertragene Bits ersetzt. Es wird also ein zu übertragenes Byte mit 8 Bit durch 10 Bits repräsentiert. Die Codeeffizienz reduziert sich somit auf  $4/5 = 80\%$ . Die Bildungsvorschrift des Codes besagt: Maximal eine Null am Anfang des Codeworts sowie maximal zwei aufeinander folgende Nullen am Ende des Codeworts. Damit wird eine ständige Änderung der Codefolge gewährleistet, welche eine gleichspannungsfreie Übertragung (erforderlich für die Rückgewinnung der Taktinformation) gewährleistet.

Durch die Codeeffizienz von 80 % erfolgt bei Fast-Ethernet die tatsächliche Übertragung der Informationen mit 125 Mbit/s. Die 4B5B Codierung weist gegenüber dem Verfahren des Bit-Stuffing den Vorteil auf, dass das Verhältnis der Eingangscodierung zur Ausgangscodierung hinsichtlich der Informationsmenge immer konstant ist und somit eine deterministische Berechnung der Übertragungszeiten unabhängig von den zu übertragenden Codeworten durchführen lässt.

Hexadezimal	4_Binär	5_Binär
0x00	0000	11110
0x01	0001	01001
0x02	0010	10100
0x03	0011	10101
0x04	0100	01010
0x05	0101	01011
0x06	0110	01110
0x07	0111	01111

<sup>40</sup> PTP-Master Zeitmaster des verteilten Systems

<sup>41</sup> PTP-Slave stellt seine Uhr auf die Master-Uhr ein

0x08	1000	10010
0x09	1001	10110
0x0A	1010	10110
0x0B	1011	10111
0x0C	1100	11010
0x0D	1101	11011
0x0E	1110	11000
0x0F	1111	11101

Tabelle 3.1: Codetabelle für ein Halbbyte (Nibble) bei einer 4B5B Codierung

Für die Reduzierung der erforderlichen physikalischen Bandbreite wird die MLT3-Codierung eingesetzt. Bei der MLT3-Codierung (ternäre Codierung) werden zusätzlich Informationen in Amplitudenstufen codiert. MLT3 verwendet dabei die Signalpegel  $-V$ ,  $0$ ,  $+V$ .

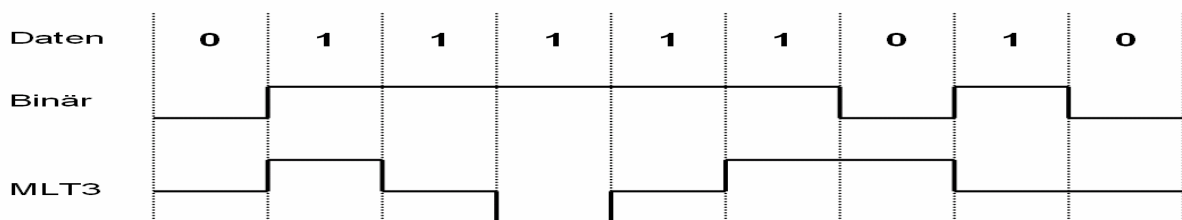


Abbildung 3.10: Beispiel einer MLT 3-Codierung

Durch die Amplitudenmodulation bedingt, erfolgt die Übertragung von 4 Bit in einer Periode. Die physikalische Frequenz beträgt beim Fast-Ethernet-Standard, welcher zum größten Teil für die Anwendung im industriellen Bereich angewendet wird, mit  $125 \text{ Mbit/s} * 1/4 = 31,25 \text{ MHz}$ .

### 3.6.5 Grundlegende Architekturkonzepte für echtzeitfähiges Ethernet

Architekturkonzepte für die Realisierung einer echtzeitfähigen industriellen Kommunikation mit Ethernet lassen sich prinzipiell in drei Klassen einteilen. In Abbildung 3.11 sind die drei möglichen Architekturkonzepte gegenübergestellt. Nach Variante 1 erfolgt die Kommunikation weiterhin über die bekannten Protokolle des Ethernet (TCP/IP/UDP). Die Verbesserung des Zeitverhaltes wird durch den zusätzlichen Eingriff in der Applikationsebene erreicht.

Die Gewährleistung echtzeitfähiger Kommunikation nach Variante 2 erfolgt durch eine Aufteilung in einen deterministischen und einen offenen, stochastischen Kanal beim Zugriff auf das Kommunikationssystem. Für die Realisierung eines einheitlichen Zeitmanagements in industriellen Kommunikationssystemen werden das Zeitschlitz- bzw. das Zeitsynchronisationsverfahren nach der IEEE 1588 eingesetzt. Charakteristisches Merkmal von Konzepten dieser Art ist die quasi parallele Kommunikation von IP-basierten Standardprotokollen und proprietären Echtzeitprotokollerweiterungen einzelner Systeme.

Echtzeitethernet nach Variante 3 ist mit gravierenden Veränderungen am Ethernetstandard verbunden. Die Anwendung der Verfahren Zeitschlitz und Zeitsynchronisation erfordern, dass sich die Protokolle TCP/IP/UDP vollständig einem proprietären Kommunikationsmanage-

ment unterordnen müssen. Die gewünschte IP-Kommunikation wird durch Untertunnelung (z. B. EtherCAT) realisiert. Um die Forderungen der Klasse 4 der IAONA zu erreichen, wird im Extremfall der komplette Protokollstack<sup>42</sup> ersetzt bzw. sogar das Ethernettelegrammformat verändert<sup>43</sup>.

Für die Architekturkonzepte des Ethernet in der industriellen Kommunikation gilt: Das deterministische Verhalten, also die präzise zeitliche Vorhersagbarkeit der Kommunikation, steigt von Variante 1 nach Variante 3. Die Kompatibilität zum Ethernet aus dem Officebereich sinkt dabei in gleicher Reihenfolge. Mit den drei Architekturkonzepten sowie den daraus folgenden Realisierungen lassen sich ethernetbasierte Kommunikationssysteme für industrielle Anwendungen realisieren, welche gegenwärtig durch die Feldbuskommunikation abgedeckt werden.

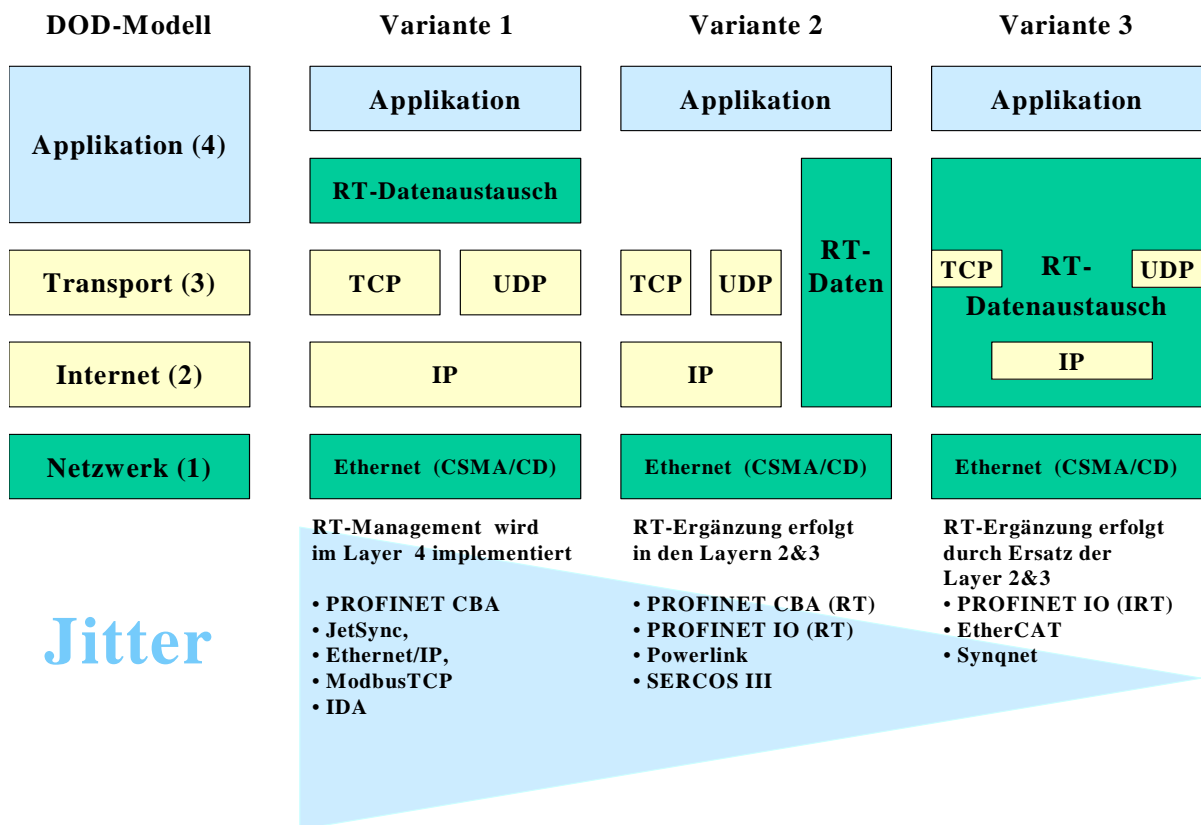


Abbildung 3.11: Gegenüberstellung Architekturkonzepte für Industrial Ethernet

<sup>42</sup> z.B. bei PROFINET IRT-Kommunikation als Hardwarerealisierung mit ERTEC200/400 als Protokoll-IC (verfügbar seit Anfang 2005)

<sup>43</sup> z.B. Ethernetkommunikation für die Antriebssynchronisation bei Synqnet [3.65]



Echtzeitklasse	Architekturkonzept	Typische Vertreter
1	1	Ethernet/IP, Modbus/TCP, IDA
2	2	Ethernet/IP, Modbus/TCP
3		Powerlink, PROFINET V2
4	3	PROFINET V3, EtherCAT, SERCOS-III

**Tabelle 3.2: Klassifizierung ausgewählter Verfahren nach der IOANA (siehe Abbildung 3.6)**

**Schlussfolgerung:**

Der Widerspruch aus Kompatibilität und hoher zeitlicher Präzision der Kommunikation führt zu der Schlussfolgerung, dass es „ein“ Industrial Ethernet für die Lösung aller industriellen Kommunikationsaufgaben nicht geben kann.

### 3.7 Technische Realisierungen von Industrial Ethernet

#### 3.7.1 Einführung

Gegenwärtig arbeiten mehrere Firmen und Organisationen an Konzepten die Ethernettechnologie für den industriellen Einsatz tauglich zu machen. Nach [3.44] existieren sieben verschiedene Ansätze für die Realisierung der gestellten Aufgabe. In [3.45] wird von neun unterschiedlichen Konzepten ausgegangen. Unter Berücksichtigung von Überschneidungen in den Literaturstellen kann zum gegenwärtigen Zeitpunkt von dreizehn Firmen und Konsortien ausgegangen werden, welche an der Aufgabenstellung „Industrial Ethernet“ arbeiten. Nach [3.12][3.61] sind von fünf Standards gegenwärtig Lösungen verfügbar, welche die Echtzeitforderungen der industriellen Kommunikation erfüllen.

#### 3.7.2 Standard-Konzepte

Als Standard-Konzepte sollen die Ansätze der Realisierung von Industrial Ethernet verstanden werden, mit denen ein große Bandbreite<sup>44</sup> industrieller steuerungstechnischer Aufgabenstellungen bedient werden kann.

##### 3.7.2.1 PROFINET

Bei der Entwicklung der Spezifikation von PROFINET wurde der Versuch unternommen, die unterschiedlichen Anforderungen der industriellen Kommunikation mit einem System zu realisieren. Ein wesentlicher Schwerpunkt der Entwicklungsarbeiten, welche 1999 begonnen wurden, ist die Integrationsfähigkeit bestehender feldbusbasierter industrieller Kommunikationssysteme in dieses Konzept<sup>45</sup>. PROFINET wird dabei durch die Profile V1, V2 und V3 repräsentiert und verwendet in diesen Versionen ein einheitliches Realtimeprotokoll. PROFINET ist zum gegenwärtigen Zeitpunkt mit 100 Mbit/s Übertragungsrate spezifiziert und wird in geschwichteter Ethernettechnologie ausgelegt.

##### **PROFINET V1:** Real-Time-Konzept (RT)

Real-Time Ethernet (RT-Ethernet) mit einer Zykluszeit von durchschnittlich 10-100 ms (ausreichend für Aufgabenstellungen wie zum Beispiel der Gebäudeautomatisierung). Der Ansatz aus dem Jahr 1999 arbeitet dabei nach der Methode der Lastbegrenzung. Die Schwankungsbreite der Zykluszeit kann dabei durchschnittlich 100% betragen.

##### **PROFINET V2:** Software Real-Time-Konzept (SRT)

Soft-Real-Time Ethernet (SRT-Ethernet) mit einer Zykluszeit zwischen 1-10 ms und einer durchschnittlichen Schwankungsbreite von 15 %. Die Verbesserung der Durchlaufzeiten wird

---

<sup>44</sup> Vorrangig die Fertigungs- und Prozessautomation

<sup>45</sup> Über PROXY - Server (z.B. PROFIBUS, Interbus)

ausschließlich durch softwaretechnische Maßnahmen erreicht. Die Reduzierung der Durchlaufzeiten des Protokollstacks ist dabei von wesentlicher Bedeutung.

**PROFINET V3:** Isochrones Real-Time-Konzept (IRT)

Die isochrone Betriebsart ist das charakteristische Merkmal von PROFINET V3. Dabei wird die verfügbare Bandbreite wie folgt in zwei Kanäle aufgeteilt:

- Deterministischer Kanal für zyklische echtzeitfähige Kommunikation
- Offener Kanal für Standardkommunikation nach IEEE 802.3 (TCP/IP)

Bei PROFINET handelt es sich um einen offenen Standard. D.h., die Spezifikationen für RT, SRT und IRT Kommunikation sind durch die PROFIBUS Nutzerorganisation offengelegt.<sup>46</sup> Damit stehen allen Mitgliedern der Organisation die erforderlichen Informationen für die Realisierung eigener PROFINET-Komponenten zur Verfügung. Wegen der sehr hohen Anforderungen an die Zykluszeiten und dem Jitter werden bei dieser Ethernetrealisierung spezielle ASIC's für die Protokollabwicklung eingesetzt. Erste Muster der ASIC's ERTEC 200/400 sind seit Ende 2004 verfügbar. Die Serienproduktion hat Mitte 2005 begonnen. Das wesentlichste Kriterium von PROFINET V3 ist die Realisierung kleiner Zykluszeiten mit einem Jitter kleiner 1 µs. Es ergibt sich z.B. bei 250 µs Zykluszeit eine Schwankungsbreite von maximal +/-0,4%. Mit dieser geringen Schwankungsbreite lassen sich taktsynchrone Antriebsregelungen analog PROFIBUS DPV1(2) realisieren. In Abbildung 3.12 sind ausgewählte zeitliche Eckwerte des Zusammenhanges zwischen Teilnehmeranzahl, Zykluszeit und Jitter zusammengestellt.

**Leistungsdaten des isochronen Real-Time-Ethernet**

Zykluszeit	1 ms		250 µs
Jitter	< 1 µs		< 1 µs
Anzahl Teilnehmer	70	150	35
Verbleibende Bandbreite für Standardprotokolle	75 %	50 %	50 %

**Abbildung 3.12: Leistungsdaten bei PROFINET V2**

<sup>46</sup> Lorenz: Vortrag Industrielle Kommunikation mit PROFINET, PNO, Hannover Messe, 2005

Charakteristischer Protokollaufbau von PROFINET SRT:

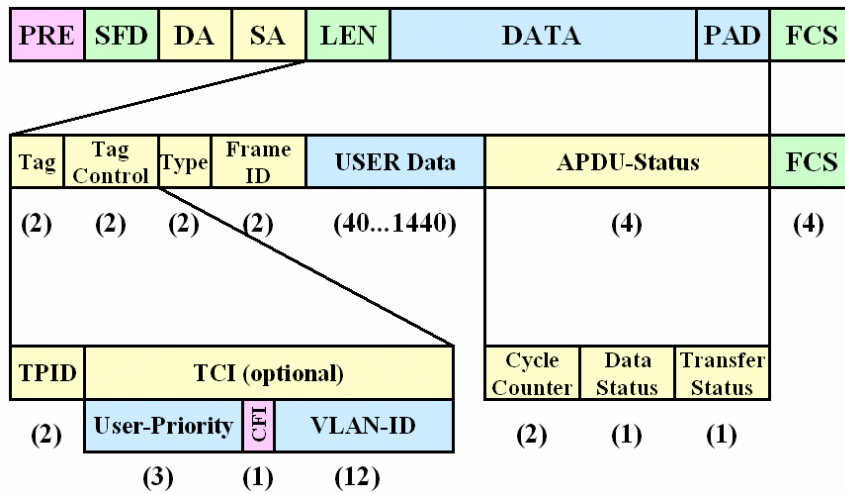


Abbildung 3.13: Telegrammformat für die Kommunikation bei PROFINET SRT [3.41]

Die Kommunikation von PROFINET wird nach dem Stand der Technik in die Profile PROFINET CBA und PROFINET IO eingeteilt [3.58]. Die unterschiedlichen Ausprägungen für das Erreichen der Echtzeitklassen können in verschiedenen Kombinationen bei beiden Geräteklassen ausgeprägt sein. PROFINET IO stellt sich dabei hinsichtlich der zeitlichen Konstanz der Kommunikation eindeutig anderen Anforderungen als PROFINET CBA gegenüber.

### 3.7.2.2 EtherCAT

EtherCAT wurde von der Firma Beckhoff entwickelt. Charakteristisches Merkmal der Umsetzung ist die Tatsache, dass die Telegramme alle Kommunikationsknoten durchlaufen. Durch die Kommunikation im Vollduplex-Modus entsteht eine physikalische und logische Ringstruktur bei der die Knoten für sie bestimmte Eingabedaten entnehmen und Ausgabedaten hinzufügen. Das EtherCAT-Telegramm hat dabei einen Aufbau nach Abbildung 3.14 und wird direkt auf das Ethernettelegramm aufgesetzt. Die TCP/IP- Kommunikation wird bei diesem Ansatz durch das Prinzip der Untertunnelung gewährleistet.

Charakteristischer Protokollaufbau von EtherCAT:

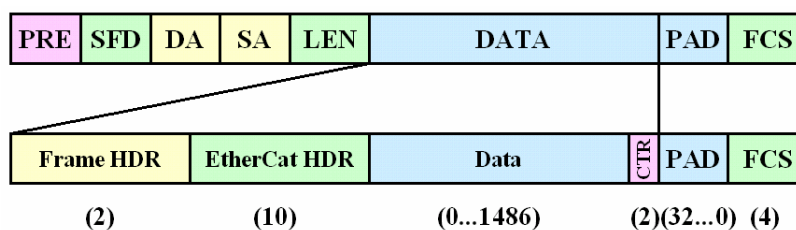


Abbildung 3.14: Telegrammformat für die Kommunikation mit EtherCAT [3.35][3.36]

Charakteristische Leistungsparameter:

➤ Übertragungsrate 100 Mbit/s
➤ Echtzeitfähigkeit nach IEEE 1588
➤ Stern-, Ring-, Baum- und Doppelringtopologie
➤ Kompatibel zu Ethernet IEEE 802.3
➤ Kommunikation vollständig in Hardware
➤ Durchlaufverzögerung von 60 ns pro Teilnehmer
➤ Protokollstack nach Variante 3 in Abbildung 3.11
➤ Segmentlänge max.100 m mit max. 65536 Teilnehmern
➤ 11888 Boolesche E/A in 350 μs

Tabelle 3.3: Leistungsparameter von EtherCAT (Auszug)

### 3.7.2.3 Ethernet/IP mit CIPSync

Charakteristischer Protokollaufbau von Ethernet/IP:

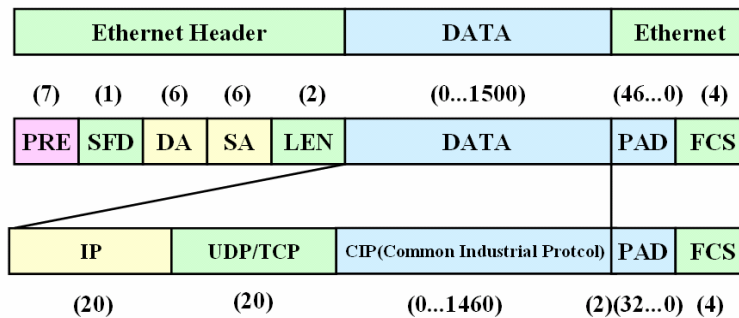


Abbildung 3.15: Telegrammformat für die Kommunikation mit Ethernet/IP[]

Charakteristische Leistungsparameter:

Übertragungsrate 100 Mbit/s
Echtzeitfähigkeit nach IEEE 1588
Stern- und Linientopologie
Kompatibel zu Ethernet IEEE 802.3
Kommunikation Standard-Ethernet-Controller
Zykluszeit 10 ms mit 30μs Jitter
Protokollstack nach Variante 1 in Abbildung 3.11

Tabelle 3.4: Leistungsparameter von Ethernet/IP (Auszug)

### 3.7.2.4 Powerlink

Powerlink ist das Realtime-Verfahren der Firma B&R<sup>47</sup> Ethernet für die industrielle Kommunikation einsetzbar zu machen.

Der Protokollstack bei Powerlink ist in zwei Teile geteilt: Azyklischer Datenverkehr wird über den Standard TCP/IP Protokollstack realisiert. Zyklischer isochroner Datenaustausch erfolgt über den Powerlink Protokollstack. Powerlink basiert auf einer klassischen Master-Slave-Kommunikationsstruktur.

Charakteristischer Protokollaufbau von Powerlink:

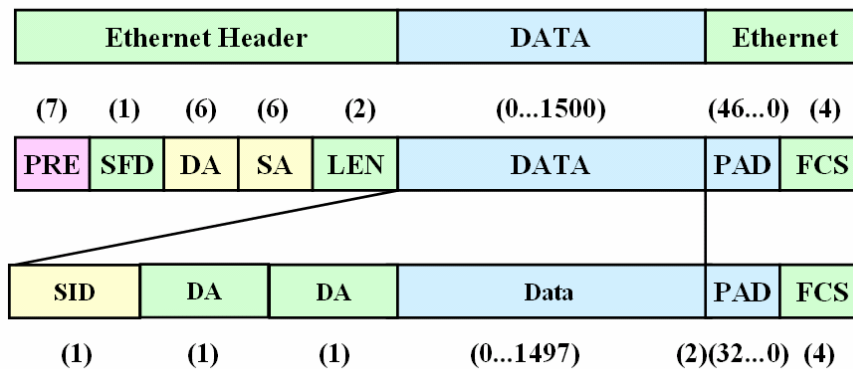


Abbildung 3.16: Telegrammformat für die Kommunikation mit Powerlink[3.26, 3.47]

Charakteristische Leistungsparameter:

Übertragungsrate 100 Mbit/s
Echtzeitfähigkeit nach IEEE 1588
Stern- und Linientopologie
Kompatibel zu Ethernet IEEE 802.3
Kommunikation vollständig in Hardware (FPGA)
Protokollstack nach Variante 2 in Abbildung 3.11

Tabelle 3.5: Leistungsparameter von Powerlink (Auszug)

<sup>47</sup> Bernecker + Rainer Industrieelektronik GmbH

### 3.7.2.5 Modbus-TCP-IDA

Modbus-TCP ist ein klassischer Vertreter der Encapsulation-Technik, bei dem das Prinzip der Untertunnelung<sup>48</sup> angewendet wird. Die Spezifikation der Protokolle ist im Standard „Modbus over IP Release 1.0“ veröffentlicht [3.23].

Modbus-TCP kann wie folgt eingeteilt werden:

- Modbus-RTU (asynchrone Übertragung über RS232 oder RS485)
- Modbus-Plus (high speed Kommunikation nach dem Token Passing Prinzip)
- Modbus-TCP (Ethernet-TCP/IP basierende verbindungsorientierte Client-Server Kommunikation über Port 502 mit 100Mbit/s)

Wesentlicher Vorteil: Das Anwenderprotokoll ist bei den drei Profilen einheitlich. Mit Modbus-TCP können theoretisch 3,6 Millionen 16-Bit-Werte pro Sekunde übertragen werden und erreicht eine Datentransferleistung eines PROFIBUS-DP-Systems mit 1,5 Mbit/s [3.45]. Modbus/TCP verwendet nach Abbildung 3.11 die Variante 1 der möglichen Architekturkonzepte.

Charakteristischer Protokollaufbau von Modbus/TCP

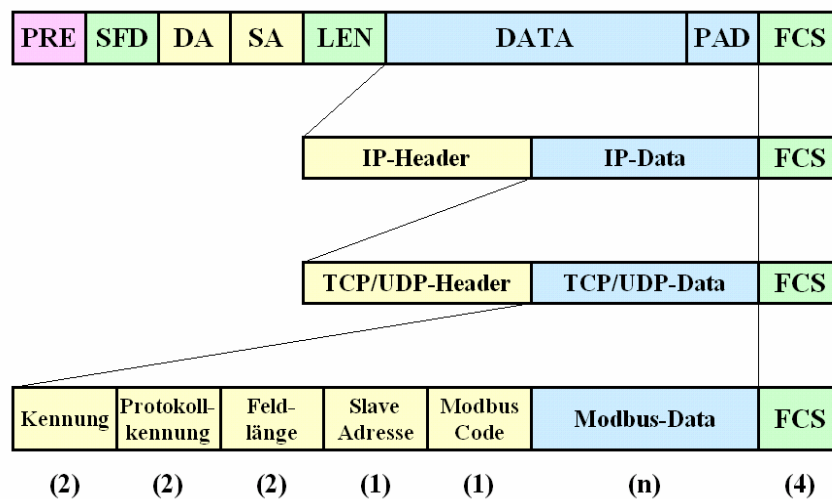


Abbildung 3.17: Telegrammformat für die Kommunikation mit Modbus/TCP [ 3.51]

<sup>48</sup> Ein naheliegender und auch häufig benutzter Ansatz ist es, ein bestehendes Busprotokoll einfach in die Ethernetpakete einzubetten. Auf diese Art und Weise lässt sich prinzipiell jedes Bussystem über einen Ethernetkanal „tunneln“ [3.73].

### 3.7.2.6 SERCOS-III

Das SERCOS-Interface wurde bei seiner Entwicklung vorrangig als Antriebsschnittstelle ausgelegt. Die taktsynchrone Arbeitsweise des Systems (Prinzip der Gleichzeitigkeit der Informationsverarbeitung) ist seit Entwicklungsbeginn ein grundlegendes Merkmal von SERCOS. SERCOS steht in drei Versionen zur Verfügung:

- SERCOS-I 2-4 Mbit/s Standardisiert 1995 in der IEC 61491
- SERCOS-II 16 Mbit/s Standardisiert 1999 in der IEC 61491
- SERCOS-III 100 Mbit/s Standardisierung erfolgt in der IEC 61491

Während bei der Entwicklung von SERCOS-I/II auf eigene Hard- und Softwarekomponenten gesetzt wurde, setzt die IGS mit der Entwicklung von SERCOS-III auf die zukunftsweisende Ethernettechnologie mit 100 Mbit/s Übertragungsrate. Die taktsynchrone Arbeitsweise des Systems wird dabei durch ein Zeitschlitzverfahren sichergestellt mit denen das System 254 Teilnehmer bedienen kann (Class-C-Netz). Um den zeitlichen Anforderungen zu genügen erfolgt die Protokollabwicklung durch spezielle ASIC's<sup>49</sup>, welche seit Mitte 2005 als Muster zur Verfügung stehen. Das SERCOS-III-Interface kombiniert die Funktionalitäten zur Antriebssynchronisation mit dem erforderlichen echtzeitfähigen E/A-Datenaustausch wodurch auch in Stand-Alone-Systemen (z.B. Werkzeugmaschinen) auf getrennte Bussysteme verzichtet werden kann.

Charakteristischer Protokollaufbau von SERCOS-III:

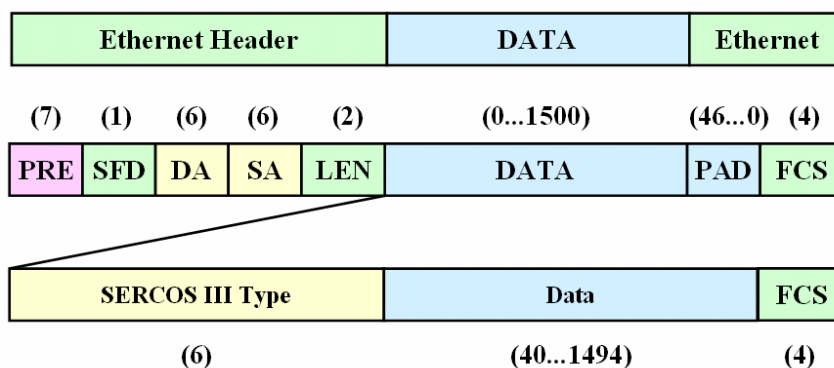


Abbildung 3.18: Telegrammformat für die Kommunikation bei SERCOS-III [3.37][3.38]

<sup>49</sup> SERCON 816



Charakteristische Leistungsparameter:

➤ Übertragungsrate 100 Mbit/s
➤ Echtzeitfähigkeit durch Zeitschlitzverfahren
➤ Line- und Doppelringtopologie
➤ Kompatibel zu Ethernet IEEE 802.3
➤ Kommunikation vollständig in Hardware (SERCON 816)
➤ 31,25 $\mu$ s Zykluszeit für 8 Teilnehmer mit 8 Byte Nutzdaten
➤ 250 $\mu$ s Zykluszeit für 70 Teilnehmer mit 12 Byte Nutzdaten
➤ 1000 $\mu$ s Zykluszeit für 150 Teilnehmer mit 32 Byte Nutzdaten
➤ 1000 $\mu$ s Zykluszeit für 254 Teilnehmer mit 16 Byte Nutzdaten
➤ Protokollstack nach Variante 3 in Abbildung 3.11
➤ Segmentlänge max.100 m mit max. 254 Teilnehmern
➤ Ethernettypkennung 0x88CD

**Tabelle 3.6: Leistungsparameter von SERCOS-III (Auszug)**

### 3.7.2.7 IAONA

Die IAONA wurde 1999 gegründet und stellt eine unabhängige Plattform dar, die sich mit der Einführung des Ethernetstandards IEEE 802.3 für die industrielle Steuerungstechnik befasst. Die IAONA bietet keinen eigenen Ansatz zur Lösung der Aufgabenstellung „Industrial Ethernet“ an, sondern koordiniert und unterstützt die Firmen und Vereinigungen bei den Entwicklungsarbeiten, um einen möglichst einheitlichen Standard zu definieren. Eine Vielfalt von Standards ähnlich der Entwicklung von Feldbussen soll auf diese Art und Weise verhindert werden.

### 3.7.3 Weitere Konzepte

#### SynqNet (Motion Engineering Inc. (MEI))

SynqNet stellt topologisch eine Ringstruktur dar, wodurch redundante Systemrealisierungen möglich sind. SynqNet verzichtet auf Switches und Hubs. Der Jitter wird im Bereich zwischen 0,5  $\mu$ s und 15  $\mu$ s angegeben womit der Ansatz gemäß der Anforderungen der IAONA in Klasse 4 einzustufen ist. Der Telegrammaufbau ist bei SynqNet zugunsten einer Reduzierung der Latenz- und Zykluszeiten von minimal 72 Bytes nach IEEE 802.3 auf minimal 24 Bytes reduziert worden. Diese Reduzierung ist aus Sicht der Entwickler erforderlich um die gewünschten kurzen Zykluszeiten zu erreichen, machen aber dadurch die Lösung herstellerabhängig. Die Protokollabwicklung kann aufgrund dieser Vorgabe nicht mit Standard-Ethernet-Controllern realisiert werden.

Synqnet ist nach den spezifizierten Echtzeitanforderungen eindeutig für hochdynamische verteilte Anwendungen, insbesondere Motion Control-Anwendungen, vorgesehen. Diese Einschätzung kann zusätzlich mit dem Firmenprofil des federführenden Herstellers Motion Engineering Ing. sowie seinen Partnerfirmen begründet werden. Der Ansatz zielt auf das Einsatz-

gebiet der Synchronisation von hochdynamischen Bewegungsabläufen und konkurriert in dieser Hinsicht mit dem SERCOS-III-Interface.

### **Interbus on Ethernet**

Vom Interbusclub wird der Ansatz PROFINET als umfassendste Kommunikationslösung auf Basis von Ethernet eingestuft. Im März 2004 wurde der Arbeitskreis Integration Interbus in PROFINET gegründet. Unter Leitung von Dr. Jasperneite von der Firma Phoenix Contact soll die Umsetzung des PROFIBUS-PROXY-Konzepts von Interbus in PROFINET realisiert werden [3.79]. Phoenix Contact beschränkt sich bei den Arbeiten am Industrial Ethernet auf den Bereich der Schnittstelle zwischen Ethernet und seinem hauseigenen Feldbussystem Interbus-S. Das Unternehmen verzichtet gegenwärtig auf die Option eine eigene Industrial Ethernet Variante zu entwickeln.

### **JetSync**

JetSync wurde von der Firma Jetter entwickelt. Das Konzept basiert auf TCP/IP Kommunikation. Prozesssynchronisationen können mit einem Jitter  $<10 \mu\text{s}$  über Mechanismen des TCP/IP Protokolls realisiert werden. Die Echtzeitfähigkeit wird im Layer 4 des DOD-Modells unter Nutzung des Zeitsynchronisationsprotokolls nach der IEEE 1588 realisiert.

### **Rtnet**

Stellt ein Open Source Projekt dar, welches vom Institut für System Engineering, der Universität Hannover (ISE) entwickelt wird. Die Arbeitsgruppe „Real Time Systems Group“ der Universität Hannover befasst sich mit der Entwicklung eines echtzeitfähigen Ethernetstandards auf Basis von Open Source Software. Dieses Projekt mit dem Namen „Rtnet“ unterliegt den Lizenzbedingungen der GPL.

Gegenwärtig erfolgen auch im asiatischen Raum Arbeiten an der Entwicklung von Standards für die Realisierung einer echtzeitfähigen Ethernetkommunikation für den industriellen Einsatz. Wichtige Ansätze, zum jetzigen Zeitpunkt sind zwei japanische und ein chinesischer bekannt, sind in Tabelle 3.7 zusammengefasst. Mit den gegenwärtig verfügbaren Informationen über diese Systeme ist keine aussagekräftige Analyse und Bewertung der in Abbildung 3.11 und Tabelle 3.2 vorgestellten Konzepte möglich.

<b>Vnet/IP</b>	Firma Yokogawa (Japan)
<b>Tcnet</b>	Firma Toshiba (Japan)
<b>EPA</b>	ITEI Instrumentation Technology & Economy Institute (China)

**Tabelle 3.7: Ansätze industrieller Ethernetkommunikation im asiatischen Raum**

### **3.8 Erkenntnisse für den Entwurf verteilter echtzeitfähiger Systeme**

Aus den Betrachtungen über industrielle Kommunikationssysteme wird ersichtlich, dass es nicht einfach möglich ist, Systeme verschiedener Hersteller und der damit verbundenen Standards in inhomogenen verteilten Automatisierungssystemen zu verbinden. Diese Tatsache wird auch der verstärkte Einsatz von Ethernettechnologien in der Automatisierungstechnik, welche auf der Zielstellung der Realisierung flacherer homogener Kommunikationsstrukturen basieren, zwar entschärfen aber nicht lösen.

Verteilte industrielle Kommunikationssysteme können nach folgenden drei wesentlichen Merkmalen optimiert ausgelegt werden:

- Nach der maximal zulässigen Zykluszeit des Systems und der damit verbundenen effektiv erforderlichen Datenübertragungsrate
- Nach der räumlichen Ausdehnung des Systems und der damit verbundenen Signallaufzeiten
- Nach der Anzahl der Kommunikationsteilnehmer im automatisierungstechnischen System

## 4 Modellierung von verteilten Kommunikationssystemen

### 4.1 Grundlagen der Modellbildung

Um ein technisches System analysieren zu können, muss es zunächst beschrieben werden. Eine Modellbildung von Kommunikationsstrukturen, die für den Informationsaustausch in größeren bzw. großen Automatisierungssystemen einsetzbar sein soll, kann zunächst ohne Berücksichtigung des angekoppelten Prozesses erfolgen. Allgemein muss berücksichtigt werden, dass jedes Automatisierungssystem grundsätzlich dem Anwendungsfall gerecht werden muss. Das heißt, dass die Reaktionszeit des Automatisierungssystems jeder denkbaren Situation der Zeitanforderung des zu steuernden Prozesses gerecht werden. Die Reaktionszeit ist die wichtigste Prämisse, die beim Entwurf von Automatisierungsstrukturen zu beachten ist.

Genau diese Forderung muss bei wachsender Komplexität und Größe einer Automatisierungsanlage ein Modellierungsansatz erfassen. Anzahl, Art und Vernetzungsprinzip der Automatisierungsgeräte, die Leistungsfähigkeit des verwendeten Feldbussystems bzw. Feldbusysteme und die Dynamik der Teilprozesse des Gesamtprozesses müssen im Modellierungsansatz und im Simulationslauf berücksichtigt werden können.

Anzumerken ist, dass beim Entwurf verteilter Strukturen die wachsende Komplexität und der ebenfalls rasch wachsende Informations- und Prozessdatenaustausch zwangsläufig dazu führt, immer leistungsfähigere Bussysteme bzw. Kommunikationsnetze einsetzen zu müssen. Ist es immer notwendig, die Leistungsfähigkeit zu erhöhen und ist nicht auch der Attraktivitätsnachweis, der Beweis über die maximal erreichbare Leistungsfähigkeit eines verteilten Systems der Automatisierungstechnik, in der Regel ein treibendes Element?

Die Beschreibung für die Modellierung verteilter Kommunikationsstrukturen muss praktikabel gehandhabt werden können und sich durch einen übersichtlichen Aufbau auszeichnen. Das heißt, das Modell muss eine hohe Transparenz für den Anwender besitzen, große Flexibilität beinhalten und der Realität möglichst nahe kommen. Die auf Basis des Modells erreichbaren Simulationsergebnisse müssen mit den experimentellen Ergebnissen in einer Versuchs- bzw. Demonstrationsanlage korrelieren. Nur so kann erreicht werden, dass die seit langem bestehenden Vorurteile eines Modellierungs- und Simulationssystems wesentlich entschärft werden.

Generell gilt, dass ein Modell eine abstrahierte Abbildung aller wesentlichen Zusammenhänge, Eigenschaften und Abläufe eines technischen Systems beinhalten muss.

Wesentliche Ziele der Modellbildung eines verteilten Automatisierungssystems sind:

- das Erreichen eines ausreichenden Verständnisses für die im Modell ablaufenden zeitlichen und logischen Abläufe
- eine wesentliche Unterstützung bei der Planung und Projektierung eines Systems auf Basis modellhafter Vorstellungen

- eine nachvollziehbare Leistungsbewertung
- die Validation des Modells und dessen Verifikation

Die Grundelemente eines Modellierungs- und Simulationssystems, die unter anderem durch Algorithmen, Tabellen, mathematischen Funktionen und Schemata dargestellt werden können, bilden die Ausgangsbasis, um Modelle zu entwickeln, die trotz des hohen Abstraktionsgrades ihre Praxistauglichkeit im Simulationslauf überzeugend nachweisen können.

Eine Modellierung erfolgt üblicherweise in folgenden Arbeitsschritten:

<b>1. Schritt</b>	Festlegung der Grenzen des zu modellierenden Systems Bestimmung der Eingangs und Ausgangsgrößen
<b>2. Schritt</b>	Zerlegung des Systems in überschaubare Teilsysteme Festlegung der Schnittstellen zwischen den Teilsystemen
<b>3. Schritt</b>	Modellentwicklung für jedes Teilsystem durch theoretische und /oder experimentelle Untersuchungen
<b>4. Schritt</b>	Vereinfachung des Modells und Implementierung der Teilsysteme
<b>5. Schritt</b>	Zusammenbau der Teilsysteme zum Gesamtmodell

**Tabelle 4.1: Schritte der Modellbildung [4.25]**

Für die Modellbildung eines feldbusneutralen Beratungssystems ist ein allgemeingültiger abstrakter Ansatz erforderlich, der den oben genannten Forderungen entsprechen muss.

#### **4.1.1 Eigenschaften von Modellen und deren Klassen**

Modelle lassen sich entsprechend Abbildung 4.1 in verschiedene Klassen einteilen. Für die Modellierung technischer Systeme ist die Klasse der begründbaren Modelle bedeutsam. Durch die rasante Entwicklung der Computertechnik können mit dieser Modellklasse immer leistungsfähigere Simulatoren realisiert werden.

Eine Simulation eines technischen Sachverhaltes bietet folgende Vorteile:

- Einsparung von aufwendigen Versuchsanlagen
- Analyse und Optimierung eines Systems einer nach vorgegebenen Zielfunktion
- Reduzierung des Projektierungsaufwandes

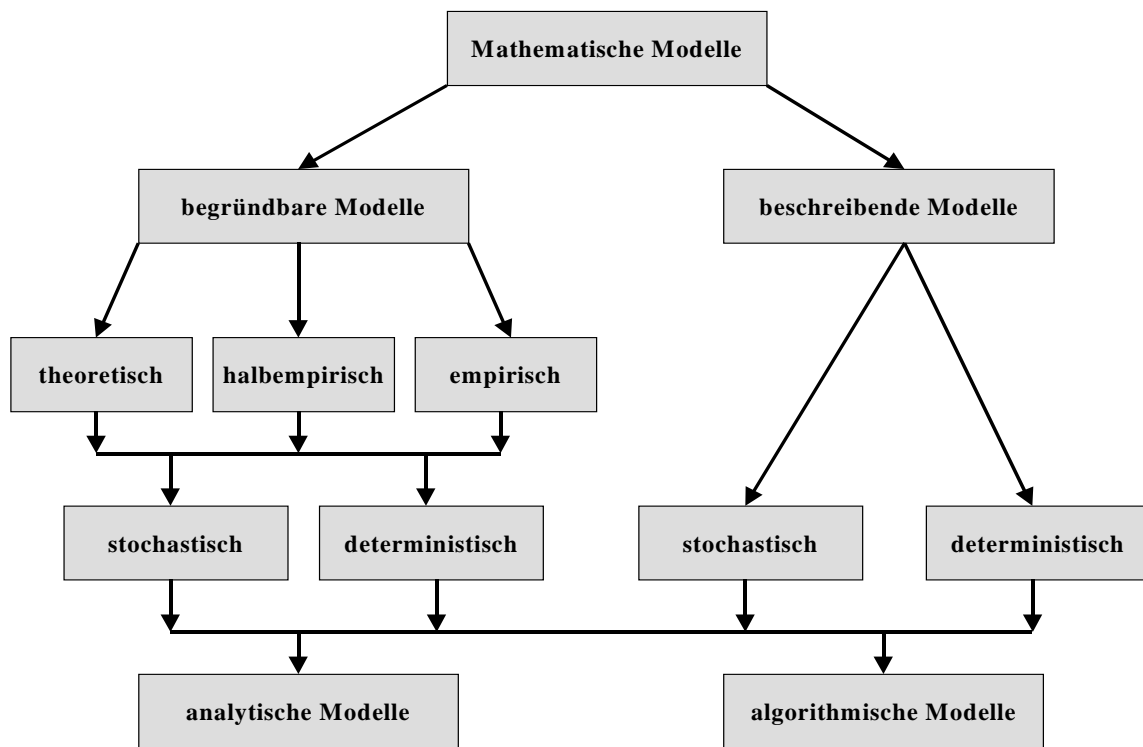


Abbildung 4.1: Modellklassen [4.26]

#### 4.1.2 Ziele, Vorteile und Probleme der Modellbildung

Die Modellbildung verfolgt folgende Ziele:

- Analyse des zeitlichen Verhaltens eines fest definierten Systems
- Analyse des zeitlichen Verhaltens eines fest definierten Systems bei unterschiedlichem Einsatz
- Analyse von Entwurfalternativen des Systems bzw. Ergebnisse
- Bestimmung von Eingaberandbedingungen und Verhaltensprämissen, um vorgegebene Ausgaben zu erhalten.

Wesentliche Vorteile der Modellbildung sind unter anderem:

- Änderungen des Modells sind weniger kostspielig als Änderungen am realen System
- Das Modell und/oder die Eingaben sind leichter änderbar als das reale System und/oder im Prozess
- Das Modell lässt die Untersuchung von mehr Alternativen zu, weil das Modell schneller ablaufen kann als die reale Zeit
- Modellbildung erleichtert die exakte Darstellung der theoretischen Kenntnisse über das Verhalten eines sehr komplexen Systems

- Durch Simulation und Vergleich der Ergebnisse mit dem realen System lässt sich die Theorie überprüfen
- Das Systemverständnis wird verbessert und der Einfluss verschiedener Faktoren auf das System veranschaulicht.

Probleme bei der Modellierung:

Die Modellbildung wird bei entsprechender Größe und Komplexität von realen Systemen in der Regel schwieriger. Es besteht die Gefahr, dass u.a. folgende Probleme auftreten können:

- das Modell führt zu hohen Rechenzeiten, weil das Modell zu viele Parameter und Sachverhalte berücksichtigt
- ein Modell widerspiegelt die abzubildende Wirklichkeit nur unzureichend, da wichtige Parameter oder Sachverhalte fehlen bzw. fehlerhaft modelliert wurden
- das Modell enthält eine ungünstige Beurteilung der Wirklichkeit und wird dadurch schwer verständlich oder unsicher
- der Modellierungsaufwand wird zu hoch
- der Kommunikationsablauf wird über Gateways, Stacks und entsprechende Funktionen des Betriebssystems abgewickelt (deren Ausführung- bzw. Abwicklung nicht bekannt ist und auch aufwendig sein kann)
- das Simulationssystem den realen Randbedingungen des zu analysierenden Kommunikationssystems nicht gerecht wird (sowohl bei homogener aber insbesondere bei inhomogener Kommunikationsform)
- die ausgewählte Modellierungsmethode widerspiegelt nicht oder aber nur unzureichend das reale System

### 4.1.3 Simulationsstudie

Ein allgemeingültiger Zusammenhang für die Handhabung des Simulationsproblems bzw. die Entwicklungsstufen zeigt folgende Abbildung 4.2.

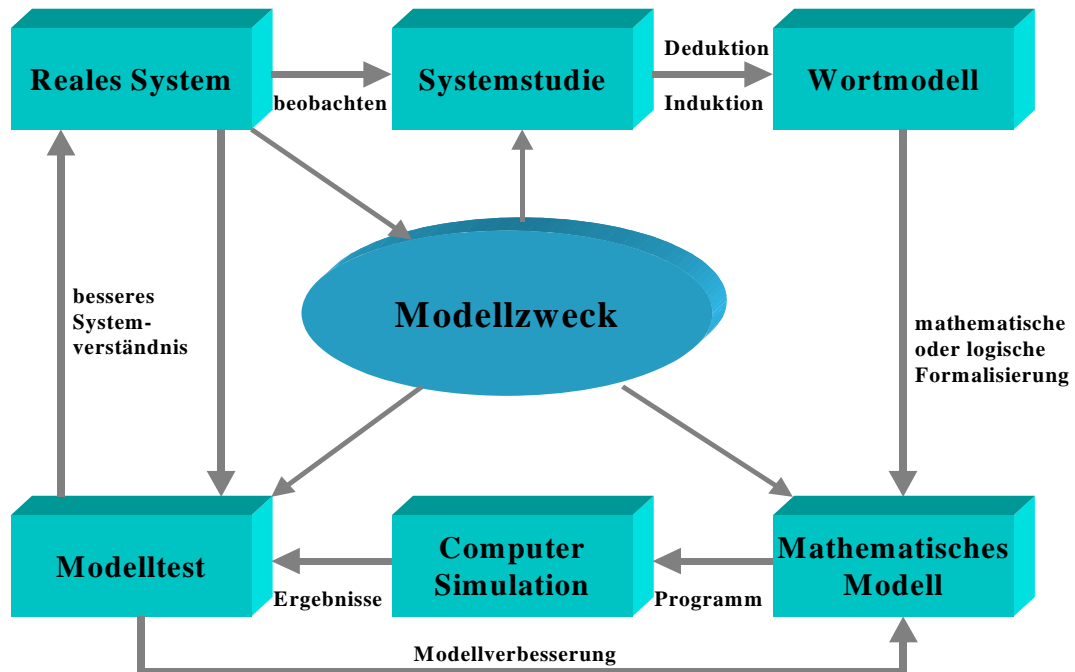


Abbildung 4.2: Modellierungsstufen [4.21]

Für eine Simulationsstudie, die als Basis für ein zu planendes Kommunikationssystem verwendet werden soll, sollten die im folgendem angegebenen Entwurfsstufen abgearbeitet werden (ein existierendes Simulationssystem wird dabei vorausgesetzt).

- Transparente Darstellung der Simulationsergebnisse
- Analyse der bestehenden Aufgabenstellung
- Aufstellen der Struktur des Systems und Platzierung der benötigten Komponenten
- Aufstellen der erforderlichen Kommunikationsbeziehungen zwischen den Teilnehmern
- Syntaktischer Test des Modellaufbaus
- Durchführung der Simulationsläufe und Auswertung der Resultate
- Analyse bzw. Auswertung der Simulationsergebnisse

Tabelle 4.2: Simulationsstudie

## 4.2 Modellierungsmittel und Methoden

Modellierungsmittel und Methoden unterstützen die Entwicklung eines abstrakten Modells. Wichtige Modellierungsmittel sind unter anderem:



- **Zustandsgraphen:** Endliche Automaten ermöglichen, die Aktivität eines Programms in Abhängigkeit von Zuständen und aktuellen Ereignissen zu spezifizieren. Dies spielt sowohl beim Dialogentwurf für interaktive Anwendungssysteme als auch im Bereich der Echtzeitsysteme eine wichtige Rolle. Das dynamische Verhalten von Echtzeitsystemen wird auch heute noch durch Zustandsgraphen, Zustandsmaschinen und endliche Automaten beschrieben.
- **Prozessmodellierung:** Prozesse sind kausal vernetzte Ereignisketten, die in der Regel per Nachrichtenaustausch synchronisiert werden müssen. Ein wichtiges Modellierungsmittel auf der Anwendungsebene ist das Petri-Netz [4.31][4.33].
- **Ablaufmodellierung:** Sequentielle Prozesse (Abläufe) können sowohl mittels dem klassischen Flussdiagramm als auch mit Struktogrammen (Nassi-Shneiderman-Diagrammen) dargestellt werden. Eine Darstellung durch Petri-Netze ist ebenfalls möglich.
- **Datenmodellierung:** Für persistente Daten sind sämtliche Verfahren der Datenmodellierung relevant, insbesondere das relationale Datenmodell und das Entity-Relationship-Modell. Anwendungsbezogene Datenmodelle werden in Datenlexika aufgenommen. Zunehmend werden auch die Typisierung und die objektorientierte Modellierung relevant [4.34].
- **Entscheidungsstrukturen:** Die einfachste Form, die Reaktion von Programmen in Abhängigkeit von Ereignissen darzustellen, ist die tabellarische Form (Belegungstabellen, Schaltfolgetabellen). Bei der tabellengesteuerten Programmierung wird eine Fallunterscheidung zur Grundlage der ausgewählten Programmaktivität.
- **Kommandosprachen:** Bei der Entwicklung der sprachlichen Teile von Benutzungsschnittstellen lassen sich die Verfahren zur formalen Beschreibung der Syntax von Programmiersprachen mit Hilfe regulären Ausdrücken und kontextfreien Grammatiken einsetzen. Die Semantik wird meist operational durch Interpretation der Sprach-elemente bei der Ausführung definiert.

#### 4.2.1 Modellierung durch Petri-Netze

Der Begriff des Petri-Netzes und die Netztheorie wurden auf der Suche nach transparenten, einfachen und effizienten Methoden zur Beschreibung und zur Analyse des Informations- und Kontrolldatenflusses in informationsverarbeitenden Systemen entwickelt. Petri-Netze sind abstrakte Modelle des Informationsflusses, die es gestatten, Systeme und Prozesse auf unterschiedlichen Abstrahierungsebenen und damit in unterschiedlicher Detailliertheit in einer einheitlichen Sprache zu beschreiben. Das Bedürfnis nach einer solchen Theorie entstand, als sich zeigte, dass Computer mehr können, als Funktionen numerisch auszuwerten [4.19][4.36] und [4.37].

Das Hauptanwendungsgebiet der Petri-Netze bildet die Modellierung von Systemen, in denen Systemereignisse kausal unabhängig und damit auch nebenläufig eintreten können. Nicht nur Informationssysteme arbeiten nach diesem Grundprinzip, derartige Wirkungsformen findet

man auch in Fertigungseinrichtungen und Produktionssystemen, wo die Belegung der Ressourcen bzw. Bediensysteme nach netz- bzw. graphenbasierten Prinzipien in sequentieller und/oder paralleler Art und Weise erfolgt. Die große Abstraktionskraft derartiger Entwurfs- bzw. Modellierungsmethoden führt nahezu automatisch dazu, dass netz- bzw. graphenbasierte Modelle auch in vielen Bereichen außerhalb der Informationstechnik angewendet werden.

Diskrete Systeme bilden auch einen Anwendungsbereich der Automatentheorie. Insbesondere die Zustandsgraphen werden zur Untersuchung und beim Entwurf diskreter Systeme angewendet (im folgenden Abschnitt wird auf dieses Modellierungsprinzip spezieller eingegangen). Die damit verbundenen Methoden sind sowohl in der Netz- als auch in der Graphentheorie in Form des Erreichbarkeitsgraphen aufbewahrt und weiterentwickelt worden. Man kann sagen, alles was mit Automaten modellierbar ist, kann auch mit Netzen modelliert werden.

Der Hauptvorteil der Verwendung von Petri-Netzen bei der Modellierung bzw. beim Entwurf komplexer Automatisierungssysteme besteht darin, dass die graphische Darstellung prinzipiell alle möglichen Wirkungsabläufe widerspiegelt und dass die Darstellungsform auch für den Anwender eine hohe Transparenz aufweist.

Ausgangspunkt einer jeden petrinetzbasierten Modellierung sind die zwei folgenden Grundelemente:

- Plätze bzw. Stellen
- Transitionen bzw. Ereignisse.

#### 4.2.2 Graphenbasierte Modellierung

Das graphenbasierte Modellierungsprinzip kennt folgende zwei Grundelemente:

- den Knoten
- den gerichteten Verbindungsbogen bzw. Kante.

Die Knoten repräsentieren die möglichen Prozesszustände, wobei die Prozessart zunächst bei der Modellierung nicht berücksichtigt wird. Die Kanten beschreiben bzw. beinhalten den Prozessablauf von einem Prozesszustand zu einem möglichen Nachfolgezustand. Der Zustandsübergang vom Prozesszustand  $n$  zum Prozesszustand  $n+1$  wird wie in Abbildung 4.3 gezeigt, durch Prozessaktionen erreicht.

Für die Modellierung von Kommunikationsbeziehungen erscheint es zweckmäßig, dieses Modellierungsprinzip einzusetzen. Folgende Grundbegriffe ergeben sich bei Einhaltung des oben genannten Modellierungsprinzips nahezu zwangsläufig:

- Knoten             $\Rightarrow$     Kommunikationsknoten KK
- Kante              $\Rightarrow$     Kommunikationsbeziehung KB

Ein Zustandsgraph besteht aus einer Verknüpfung von Knoten und gerichteten Kanten. Prozesszustände werden im Graphenmodell durch die Knoten dargestellt. Mit den Kanten werden die Prozessaktionen beschrieben. Für die Modellierung von Kommunikationssystemen erscheint die Einführung der Begriffe Kommunikationsknoten und Kommunikationsbeziehung als sinnvoll. Die Kommunikationsknoten stellen eigenständige, komplexe Subsysteme mit definierten Eigenschaften dar. Über die gerichteten Kanten werden die verschiedenen Kommunikationsbeziehungen zwischen den Knoten abgewickelt.

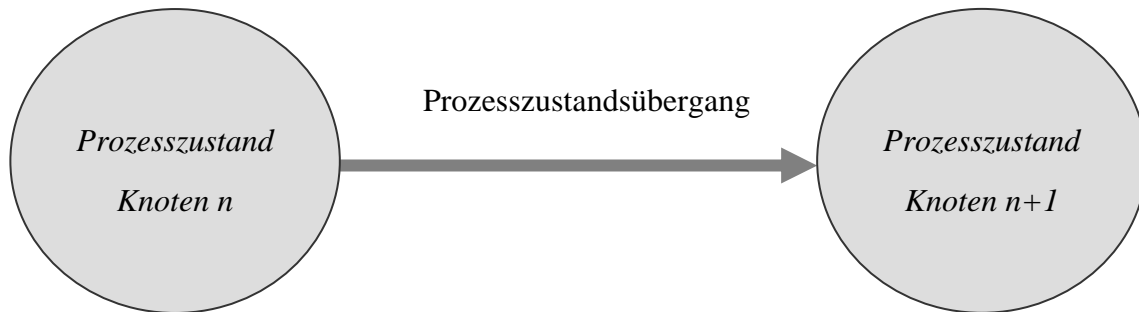


Abbildung 4.3: Darstellung eines Zustandsgraphen

Grafische Modelle haben sich in vielen Wissenschaftszweigen als Standard durchgesetzt [4.26]. Diese Darstellungsform bleibt auch bei größeren Netzausdehnungen für Menschen transparent. Graphen lassen sich mathematisch gut handhaben und können als Matrix oder in Tabellenform dargestellt werden [4.32].

Wenn man die Kommunikationsbeziehungen des Zustandsgraphen durch eigenschaftsbehaftete Kanten erweitert, dann erhält man ein Phasenmodell [4.26]. Diese Modellerweiterung ist für die eigenschaftsbehaftete Kommunikation erforderlich.

Die Struktur lässt sich aus dem, von der Projektierung geforderten, Kommunikationsverhalten ableiten.



Abbildung 4.4: Darstellung eines Phasenmodells

### 4.2.3 Algorithmische Modellierung

Algorithmische Modelle werden vorrangig zur Beschreibung von diskreten Prozessen verwendet. Dieses Modell gestattet eine detaillierte (Schritt für Schritt) Nachahmung des Systemverhaltens und hat die Form eines sehr komplexen, vermaschten Algorithmus.

Definition Algorithmus:

*Ein Algorithmus ist eine geordnete endliche Folge von elementaren Operationen und Bedingungen zur Umformung von Eingangsgrößen in Ausgangsgrößen, wobei nach der Ausführung einer Operation eindeutig feststeht oder an Hand einer Bedingung entschieden werden kann, welche Operation beim nächsten Schritt auszuführen oder das Verfahren abzubrechen ist [4.38].*

Das algorithmische Modell ermöglicht eine hochpräzise schrittweise Abbildung eines Prozessablaufes auf der Basis von Algorithmen. Entsprechend der gegebenen Algorithmendefinition basiert das algorithmische Modell auf einer Menge von elementaren Operatoren, die dem Prozessablauf entsprechen.

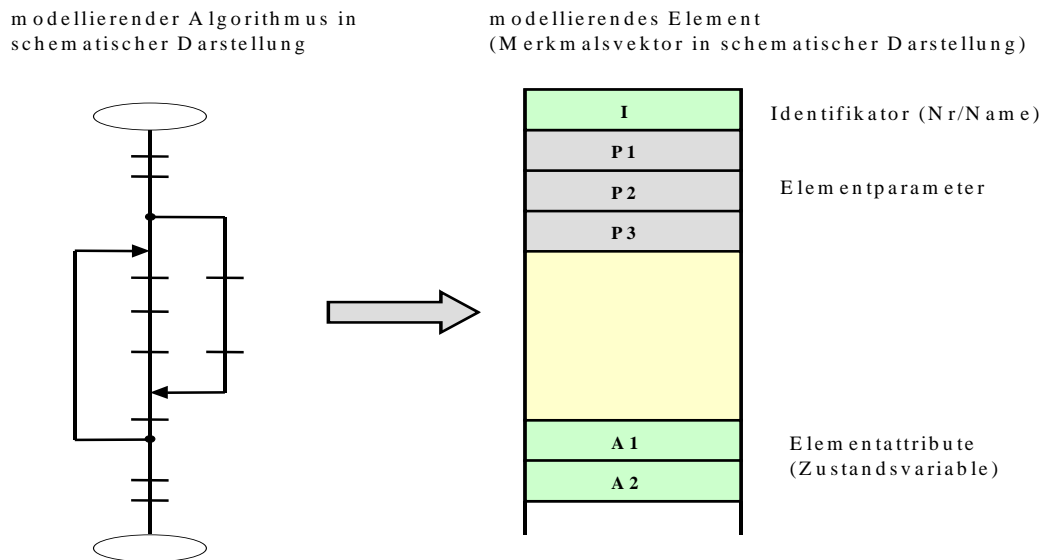


Abbildung 4.5: Grundelemente eines algorithmischen Modells

Die Grundelemente der algorithmischen Modellierung sind:

- das Ablaufelement
- das Datenelement.

In Abbildung 4.5 sind beide Grundelemente dargestellt worden. Beide Grundelemente erinnern an das im Programmablaufplan bzw. in der Ablaufkette realisierte Prinzip.

Auf der Basis der beiden Grundelemente werden in Simulationssystemen algorithmische Basen hinterlegt, die eine Ablaufanalyse in komplexen verteilten Prozessstrukturen prinzipiell ermöglichen.

Das zentrale Anliegen eines Simulationssystems stellt die Verhaltensanalyse von modellierten Systemen dar. Die schrittweise Abarbeitung des Ablaufes in einem modellierten System durch einen Simulator wird durch Ereignisse (Events) realisiert. Diese Ereignisse signalisieren dem Simulator zeitdiskrete bzw. diskrete Zustandsänderungen in modellierten Systemen. Die Ereignisse stellen weiterhin ein wesentliches Element für die Verhaltensanalyse von modellierten Systemen dar.

Dieser vermaschte Algorithmus kann bei entsprechender Vorbereitung von einem Computer abgearbeitet werden. In diesem Modell spielt der Begriff „Ereignis“ eine wesentliche Rolle. Ein „Ereignis“ ist die zu einem bestimmten Zeitpunkt erfolgende Zustandsveränderung eines Systems. Es führt zu Werteveränderungen einer oder mehrerer zusammenhängender Zustandsgrößen des Systems und beansprucht selbst keine Zeitdauer.

Ein wesentlicher Bestandteil algorithmischer Modelle sind Mechanismen zur Zufallszahlengenerierung als Basis für die Nachbildung zufälliger Einflüsse auf den zu simulierenden Prozess. Zentraler Bestandteil des Modells ist die adäquate zeitliche Nachbildung des realen Prozessverlaufes.

#### 4.2.4 Weitere Modellierungsarten

- Mathematische Modelle sind problembezogene, mathematische Beschreibungen des Zusammenhangs von Zustandsvariablen und Ausgangsgrößen eines Systems mit seinen Eingangsgrößen, Parametern und Anfangsbedingungen. Als Grundlage für die Modellierung werden verschiedene mathematische Methoden eingesetzt. Dazu gehören insbesondere Differential- und Integralgleichungen, Mengenlehre, mathematische Logik, Wahrscheinlichkeitsrechnung, mathematische Statistik und Optimierung [4.25].
- Als komplexes Werkzeug für die Modellierung von Softwaresystemen wurde die UML (Unified Modelling Language) Mitte der 1990er Jahre entwickelt. Sie ist eine Sprachdefinition zur Spezifikation, Konstruktion und Dokumentation für die Entwicklung objektorientierter Softwaresysteme. In der Version 2.0 stehen 13 Diagrammtypen für die Modellierung von komplexen Systemen zur Verfügung [4.12][4.39][4.40][4.41].

#### 4.3 Modellvalidierung

Syntaktisch getestete und für richtig befundene Modelle müssen einer sehr wichtigen Kontrolle unterzogen werden. Man spricht von der sogenannten Validierung (Gültigkeitserklärung) eines Modells. Für die Validierung eines Modellansatzes zur Simulation verteilter Kommunikationsstrukturen gibt es zwei grundsätzliche Möglichkeiten:

- Die Validierung des Modells unter der Bedingung eines bereits gegebenen verteilten Systems
- Für noch nicht existierende verteilte Systeme kann eine Simulation nur mit angenommenen Daten vorgenommen werden.

Für die Validierung eines Feldbusberatungssystems kommt sinnvollerweise die erste Möglichkeit in Frage. Der Prototyp des Feldbussimulationssystems wird dazu mit bekannten Eingangswerten gestartet, wobei deren dazugehörige Ausgangswerte (z.B. durch Messungen) bereits bekannt sind. Diese Messungen können aus einer bestehenden Versuchsanlage gewonnen werden. Damit lässt sich mit Einschränkungen die Korrektheit des Simulationsmodells absichern. Das Simulationsmodell muss in den folgenden Stufen für verschiedene Feldbussysteme überprüft werden. Für die Validierung des Modellansatzes kann im Extremfall nur die zweite Möglichkeit zur Verfügung stehen, denn ein Aufbau einer Versuchsanlage für alle üblichen Feldbussysteme stellt einen sehr hohen Aufwand dar.

#### **4.4 Problemstellungen bei der Modellierung feldbusbasierten Netzstrukturen**

Industrielle Kommunikationssysteme arbeiten im Basisbandbereich und sind immer echtzeitfähig ausgelegt. Im gesamten Kommunikationssystem kann demzufolge das Übertragungsmedium immer nur mit einer einzigen Information gleichzeitig belegt sein. In praktischen Realisierungen solcher Systeme kann und darf es nicht vorkommen, dass mehrere Informationen gleichzeitig das Kommunikationssystem belegen können. Da immer nur eine Information im System im Unlauf ist kann es auch nicht vorkommen, dass sich diese überholen können. Daraus folgt, dass keine Nebenläufigkeiten wie sie hervorragend durch Petrinetze modelliert werden können, auftreten können. Da derartige Nebenläufigkeiten in der echtzeitfähigen Kommunikation nicht erwünscht sind und das System hinsichtlich des Zeitverhaltes der Kommunikation immer nur in einem Zustand befindet kann die graphenbasierte Modellierung für die Lösung derartiger Aufgabenstellungen favorisiert werden. In den folgenden Abschnitten wurden Ansätze für die Modellierung unterschiedlicher Kommunikationsformen nur mit graphentheoretischen Überlegungen durchgeführt. Die streng deterministische Kommunikation in verteilten echtzeitfähigen Systemen führt auch dazu, dass mathematische Modellierungsverfahren nicht in die näheren Untersuchungen einbezogen wurden. Sollen Kommunikationssysteme mit gemischt deterministischen und stochastischen Kommunikationsanteil modelliert werden, sind tiefgründigere mathematische Betrachtungen, insbesondere für die Kommunikation im stochastischen Teil der Kommunikation erforderlich.

##### **4.4.1 Grundaufgabe**

Aufgrund dieser Analysen und den Erkenntnissen aus Kapitel 2 und 3 ist es möglich, Simulatoren zu bauen, die das Zeitverhalten von realen Kommunikationsstrukturen auf der Grundlage von modellierten Kommunikationssystemen ermöglichen.

Die meisten Feldbussysteme verwenden ein stationsorientiertes, deterministisches Buszugriffsverfahren. In diesen Systemen kommen Unicast, Multicast und Broadcast als typische Kommunikationsform zum Einsatz. Nachrichtenorientierte und stochastische Buszugriffsverfahren gestatten unmittelbar den Einsatz von Multipeer- und Concast-Kommunikation. Dadurch wird echter Querverkehr zwischen den Teilnehmern ermöglicht. Für eine allgemeingültige Modellierung müssen alle informationstechnisch möglichen Kommunikationsbeziehungen zwischen den einzelnen Kommunikationsknoten des Gesamtsystems berücksichtigt werden.

#### 4.4.2 Unicast-Kommunikationsbeziehungen ohne stochastischen Einfluss

Für die Modellierung eines verteilten Kommunikationssystems eignet sich das Phasenmodell. In diesem Modell können mit den Kommunikationsknoten ( $KK_{xx}$ ) die Eigenschaften der beteiligten Automatisierungsgeräte beschrieben werden. Die Kommunikationskanten ( $KB_{xx}$ ) enthalten die Eigenschaften für den Telegrammaustausch zwischen den Kommunikationsknoten. Das Phasenmodell enthält somit die vollständigen Informationen über die zu realisierende Automatisierungsaufgabe.

Am einfachsten gestaltet sich die Abwicklung eines Netzes mit einem rein deterministischen Verkehrsangebot. Am folgenden Beispiel (siehe Abbildung 4.6), bestehend aus einem Masterknoten und zwei Slaveknoten, soll der Sachverhalt verdeutlicht werden. Da das System auf keinerlei äußere Einwirkungen reagieren muss, entsteht ein linearer (streng sequentieller) Kommunikationsablauf.

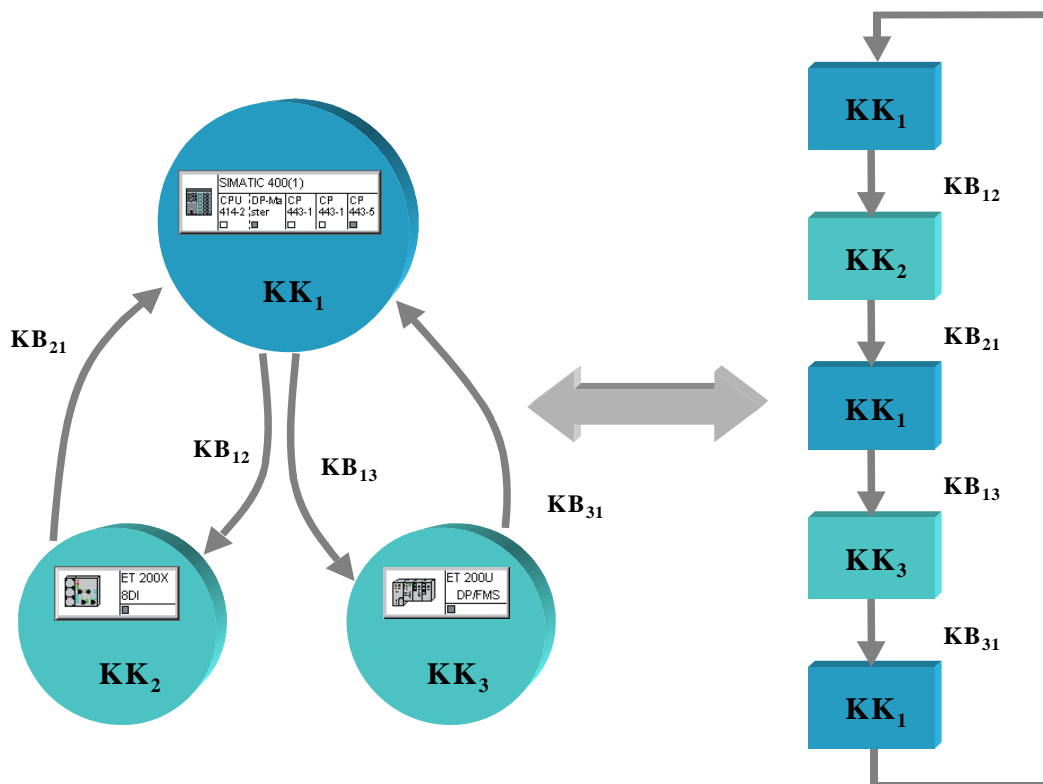


Abbildung 4.6: Netzabwicklung mit determiniertem Verkehrsangebot

Der Ablauf der Kommunikation kann beispielsweise in folgender Reihenfolge durchgeführt werden. Der Master (KK<sub>1</sub>) beginnt die Kommunikation mit dem Slave (KK<sub>2</sub>) über die Kommunikationskante KB<sub>13</sub>. Der Slave bearbeitet die gestellte Aufgabe und teilt sein Ergebnis über die Kommunikationsbeziehung KB<sub>31</sub> dem Master (KK<sub>1</sub>) mit. Damit hat ein vollständiger Telegrammaustausch zwischen diesen beiden Teilnehmern stattgefunden. Danach wird die Kommunikation zwischen dem Master (KK<sub>1</sub>) und dem zweiten Slave (KK<sub>3</sub>) entsprechend bearbeitet. Die Abarbeitungsreihenfolge der Kommunikationen ist im Beispiel zufällig ausgewählt worden.

#### 4.4.3 Unicast-Kommunikationsbeziehungen mit stochastischem Einfluss

Reale Kommunikationssysteme unterliegen grundsätzlich äußeren Störeinflüssen. Auch der zu steuernde Prozess unterliegt immer unvorhergesehenen Schwankungen. In industriellen Einsatzumgebungen spielen zusätzlich Störungen auf dem Übertragungsmedium eine wichtige Rolle. Das Verhalten eines verteilten Automatisierungssystems auf solche Einflüsse, wird durch die Systemprojektierung sowie durch die Architektur des Bussystems bestimmt.

Der wirkende stochastische Anteil kann den Programmablaufplan entscheidend verändern. Für ein feldbusneutrales Analyse- und Simulationssystem muss deshalb ein sehr allgemeingültiger Ansatz gefunden werden. Einem Codegenerator müssen unter Umständen auch Informationen über die Automatisierungsaufgabe zur Verfügung stehen, um alle Ablaufvarianten berücksichtigen zu können.

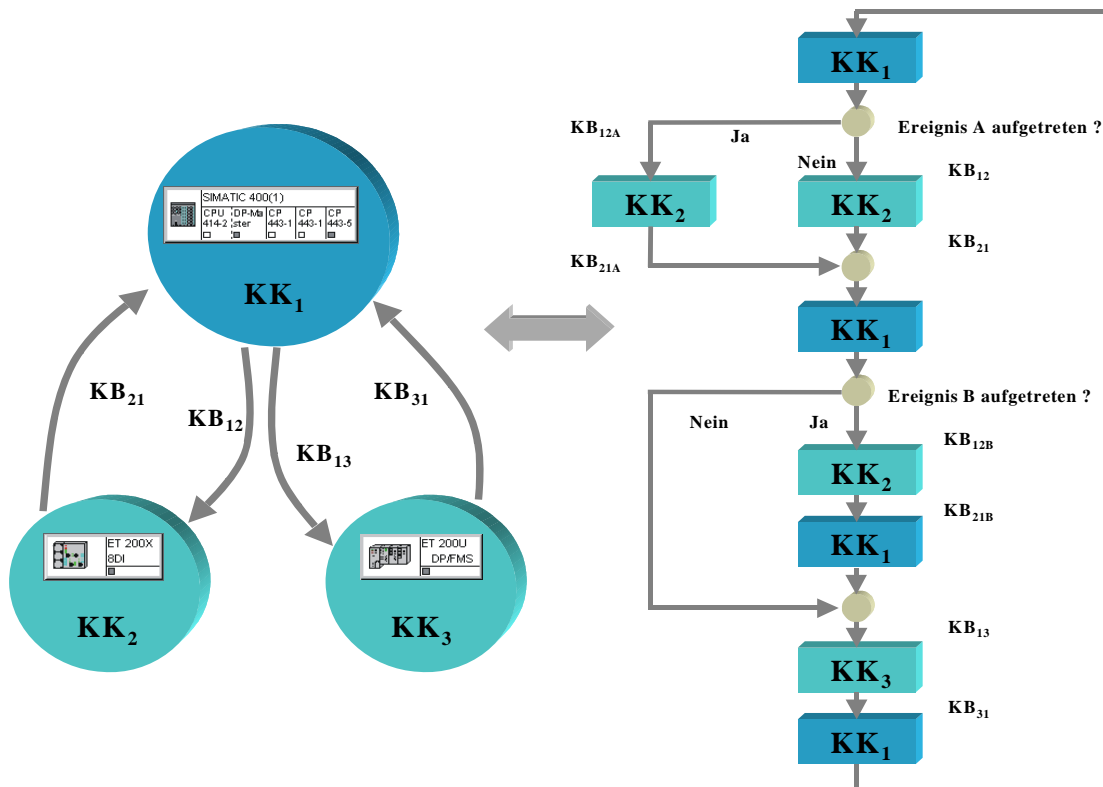


Abbildung 4.7: Netzabwicklung mit kombiniertem Verkehrsangebot



Der deterministische Kommunikationsablauf verhält sich wie im vorherigen Beispiel. Das Verkehrsaufkommen kann jetzt durch äußere Einflüsse erhöht oder verändert werden. Diese Änderungen werden durch das Eintreten der Ereignisse A und B beispielhaft dargestellt. Warum diese Unterscheidung?

Soll das Modell unabhängig vom eingesetzten Feldbussystem sein, müssen alle Änderungsmöglichkeiten des Kommunikationsablaufes berücksichtigt werden. Der normale Ablauf der Kommunikation zwischen  $KK_1$  und  $KK_2$  kann sich prinzipiell auf zwei Arten ändern. Tritt Ereignis A ein, wird im folgenden Zyklus das normale Telegramm durch ein anderes Telegramm ersetzt. Beim Auftreten von Ereignis B wird eine Telegrammwiederholung noch im selben Zyklus durchgeführt. Die Änderungen des Kommunikationsverhaltens lassen sich durch Verteilungsfunktionen der mathematischen Statistik beschreiben. Im Beispiel sind die prinzipiellen Verzweigungen von Kommunikationsverläufen dargestellt. Diese können, je nach eingesetztem Bussystem, aber auch sehr viel komplexere Formen annehmen.

#### 4.4.4 Broad- und Multicast-Kommunikationsbeziehungen

Broad- und Multicastnachrichten werden von vielen Feldbussystemen unterstützt. Mit dieser Kommunikationsmethode können alle Teilnehmer oder einzelne Teilnehmergruppen angesprochen werden. Die Teilnehmer werden nicht sequentiell, sondern parallel bedient. Gruppen von Teilnehmern lassen sich somit über das Bussystem gut synchronisieren. Die Zeitpunkte mit der alle Teilnehmer erreicht werden, unterscheiden sich maximal durch die Signallaufzeit auf dem Übertragungsmedium.

Feldbusse arbeiten durchgängig mit Basisbandübertragung. Die parallele Kommunikation über das Basisband stellt jedoch einen formalen Widerspruch dar. Deshalb entfällt bei Broad- und Multicastnachrichten die Rückantwort der angesprochenen Teilnehmer. Diese spezielle Kommunikationsform muss im Feldbusmodell gesondert berücksichtigt werden. Im Beispiel ist der Kommunikationsknoten  $KK_2$  durch den Kommunikationsknoten  $KK_0$  ersetzt worden.  $KK_1$  und  $KK_2$  verhalten sich wie in den beiden anderen Beispielen. Der Kommunikationsknoten  $KK_0$  stellt eine Multicastnachricht an zwei Teilnehmer dar, die über die Kommunikationsbeziehung  $KB_{10}$  geschickt wird. Da es keine Rückantwort gibt, würde das Phasenmodell im Knoten  $KK_0$  stecken bleiben. Dieses Problem lässt sich durch die Einführung einer imaginären Kommunikationsbeziehung  $KB_{01}$  umgehen. Diese stellt bei der Berechnung einen zeitlosen Übergang zwischen zwei Zuständen dar. Somit lassen sich mit dem Phasenmodell Broad- und Multicastnachrichten modellieren.

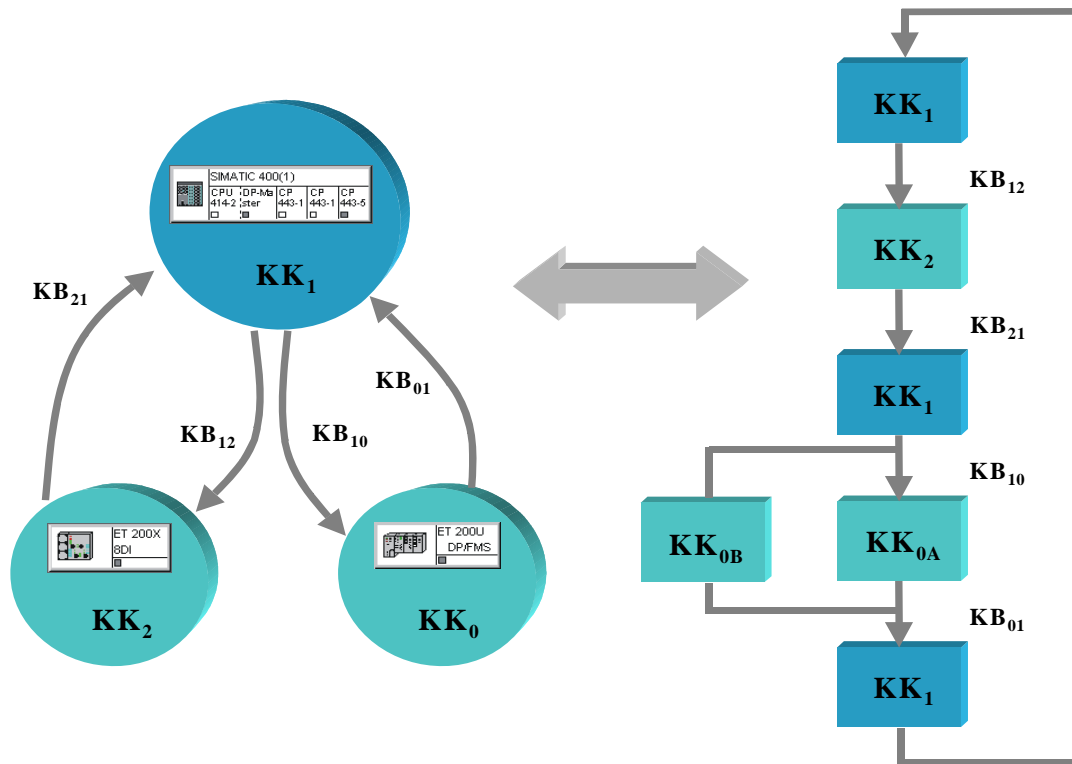


Abbildung 4.8: Netzabwicklung mit Broad- und Multicastnachrichten

#### 4.4.5 Multipeer-Kommunikationsbeziehungen

Bei Multipeer-Kommunikation können mehrere Sender mit mehreren Empfängern kommunizieren. Im Phasenmodell lassen sich die Multipeer-Kommunikationen wie Unicast-Kommunikationen betrachten. Mit der steigenden Anzahl von Sendern und Empfängern dehnt sich der zu generierende Programmablaufplan aus. Bei determinierten Buszugriffsverfahren wird diese Kommunikationsmethode durch das Token-Passing-Verfahren repräsentiert.

#### 4.4.6 Concast-Kommunikationsbeziehungen

Bei Concast-Kommunikation senden mehrere Sender ihre Daten an einen Empfänger. Diese Kommunikationsmethode kann nur bei nachrichtenorientierten oder stochastischen Buszugriffsverfahren verwendet werden. Die Telegrammeingänge werden vom Empfänger nicht quittiert. Eine Quittung an die Sender kann aber über eine Multicast-Kommunikation erfolgen. Im Phasenmodell sind keine nebenläufigen Prozesse definiert. Das Modell muss sich also stets genau in einem Zustand befinden.

Im folgenden Beispiel sind die Kommunikationsbeziehungen KB<sub>21</sub> und KB<sub>31</sub> die eigentlichen Träger der Information. Um Verklemmungen zu vermeiden, müssen imaginäre Kommunikationsbeziehungen, wie bei der Multicast-Kommunikation, eingefügt werden. Dazu gehören die Kommunikationsbeziehungen KB<sub>12</sub>, KB<sub>23</sub>, KB<sub>13</sub> und KB<sub>32</sub>. Durch ihre Einführung wird

ein geschlossener Umlauf des Systems garantiert. Das Simulationsergebnis wird durch die „zeitlose“ Abarbeitung der Transitionen nicht beeinflusst.

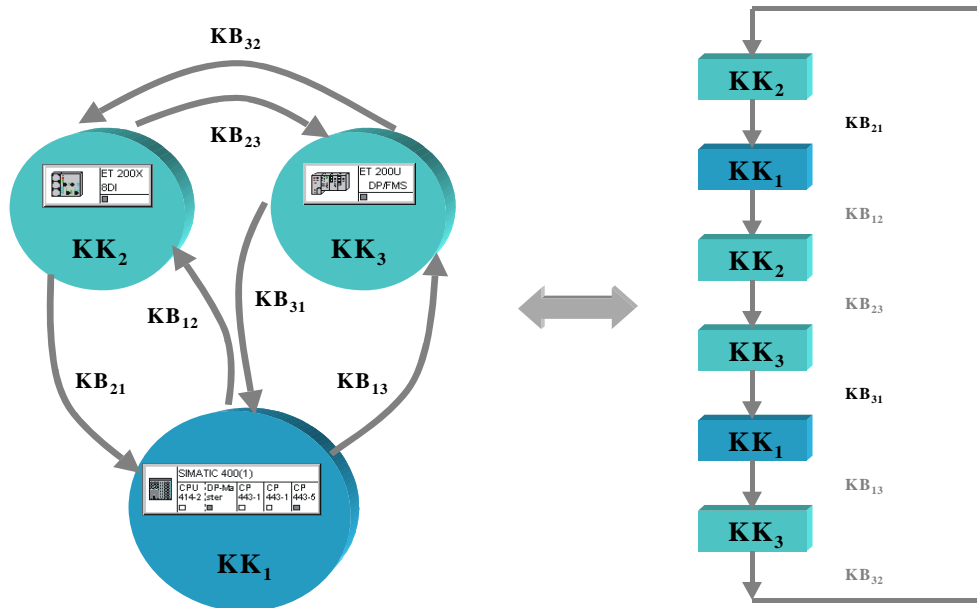


Abbildung 4.9: Netzabwicklung von Concast-Kommunikationsbeziehungen

#### 4.4.7 Abbildung von Graphen in Matrizenform

Ein Graph wird als Zusammenfassung zweier Mengen definiert.

Grundlegende Definitionen der Graphentheorie:

**Definition:** Ein Graph  $G$  besteht aus einer Menge  $X$  (deren Elemente Knotenpunkte genannt werden) und einer Menge  $U$ , wobei jedes Element  $u \in U$  in eindeutiger Weise ein geordnetes oder ungeordnetes Paar von (nicht notwendig verschiedenen) Knotenpunkten  $x, y \in X$  zugeordnet ist.

**Definition:** Ist jedem Element  $u \in U$  ein geordnetes Paar von Knotenpunkten zugeordnet, so heißt der Graph gerichtet, und man kann schreiben  $G = (X, U)$ . Die Elemente von  $U$  werden in diesem Fall als Bögen bezeichnet.

**Definition:** Ist einem Element  $u \in U$  ein ungeordnetes Paar von Knotenpunkten zugeordnet, so heißt der Graph ungerichtet, und man kann schreiben  $G = [X, U]$ . Die Elemente von  $U$  werden in diesem Fall als Kanten bezeichnet.

Für die Anwendung der Graphentheorie für eine Verhaltensanalyse und Leistungsbewertung in Kommunikationssystemen werden folgende Vereinbarungen bzw. Zuordnungen getroffen:

- Knotenpunkt bzw. Knotenmenge  $X \cong$  Kommunikationsknoten  $KK_i$  der Kommunikationsknotenmenge  $KK$

- Bogen  $U_{ij}$  der Bogenmenge  $U$  einer gerichteten Kommunikationsbeziehung  $KK_{ij}$  vom Kommunikationsknoten  $KK_i$  zum Kommunikationsknoten  $KK_j$
- Beschränkung auf die Analyse endlicher Graphen, das sind reale Graphen bzw. Kommunikationssysteme, bei denen sowohl die Menge der  $KK$  als auch die Menge der  $KB$  nur endlich viele Elemente angehören. Die Anzahl der  $KK$ -Elemente wird mit  $KK$  und die Anzahl der  $KB$ -Elemente mit  $KB$  bezeichnet
- Die einer  $KB_{ij} \in KB$  zugeordneten Kommunikationsknoten  $KK_i$  und  $KK_j$  sind Endpunkte der  $KB_{ij}$  (Endpunkte der Kommunikation = Betrachtungsende für die Modellierung)
- Gilt für eine Kommunikationsbeziehung  $KB_{ij}$   $KK_i = KK_j$ , dann bezeichnet man die dazugehörigen Bögen als Schlinge. Die kommunikationstechnische Interpretation ist für die Modellierung und Analyse von Kommunikationssystemen, insbesondere für den Buszugriff und die Buszuteilung bedeutsam (Senden und gleichzeitiges Mithören).
- Zwei Kommunikationsknoten  $KK_i$  und  $KK_j$  heißen adjazent (benachbart), wenn die Endpunkte ein und die selbe Kommunikationsbeziehung  $KB_{ij}$  sind (s.a. Darstellung von Graphen in Form einer Adjazenzmatrix)
- Eine  $KB_{ij}$  ist mit den  $KK_i$  inzident, wenn sie ein Endpunkt von  $KK_i$  ist (s.a. Darstellung von Graphen als Inzidenzmatrix)
- Graphen die keine Schlingen und Mehrfachbögen (parallele Bögen) bzw. Mehrfachkanten (parallele Kanten) besitzen, nennt man schlichte Graphen.

Für die Darstellung von Graphen in Matrizenform existieren zwei grundlegende Beschreibungsformen:

- die Adjazenzmatrix
- die Inzidenzmatrix.

**Definition:** Als Adjazenzmatrix  $A(G) = (A_{ij})$  des Graphen bezeichnet man die  $(n,m)$ -Matrix mit den Elementen

$$A_{ij} = \begin{cases} 1, & \text{falls } (X_i, X_j) \in U \\ 0, & \text{sonst} \end{cases} \quad i, j = 1, \dots, n \quad (4.1)$$

**Definition:** Als Inzidenzmatrix  $S(G) = (S_{ij})$  des Graphen  $G=(X,U)$  bezeichnet man die  $(n,m)$ -Matrix mit den Elementen

$$S_{ij} = \begin{cases} +1, & \text{falls } X_i \text{ Anfangspunkt des Bogens } U_j; \\ -1, & \text{falls } X_i \text{ Endpunkt des Bogens } U_j; \\ 0, & \text{falls } U_j \text{ nicht mit } X_i \text{ indiziert} \end{cases} \quad \begin{matrix} i = 1, \dots, n \\ j = 1, \dots, m \end{matrix} \quad (4.2)$$

Für die Darstellung von Kommunikationssystemen in Form von Graphen wird vorausgesetzt, dass alle KK und alle KB durchnummeriert sind bzw. werden. Für die Knotenbeziehung gilt:

$X = \{X_1, X_2, X_3, \dots, X_n\}$  bzw.

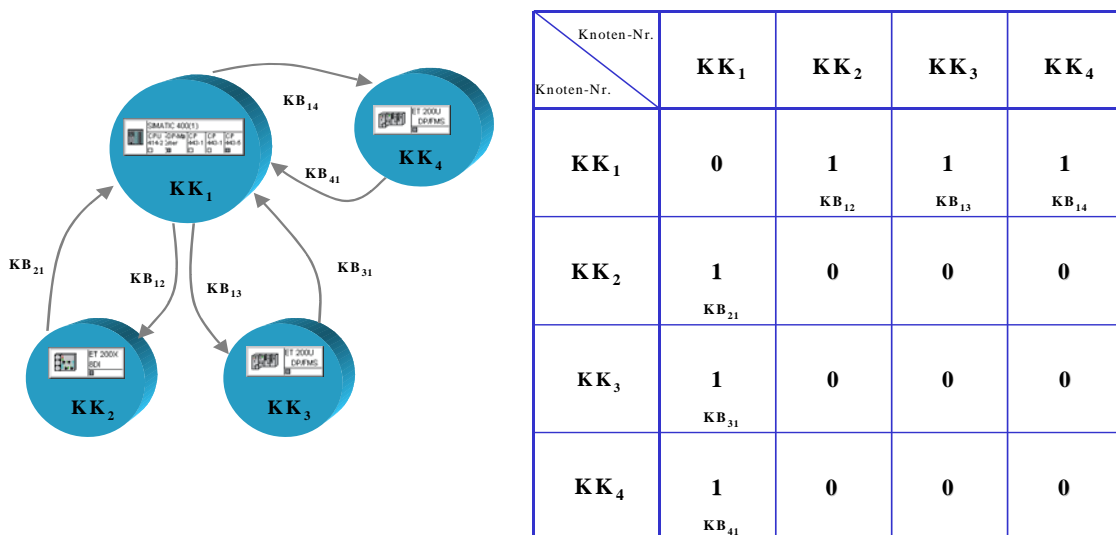
$KK = \{KK_1, KK_2, KK_3, \dots, KK_n\}$

Entsprechend wird zunächst analog die Menge der gerichteten Verbindungen der Bögen bzw. Kommunikationsbeziehungen  $KK_i$  formuliert.

$U = \{U_1, U_2, U_3, \dots, U_n\}$  bzw.

$KB = \{KB_1, KB_2, KB_3, \dots, KB_n\}$

Das folgende Beispiel zeigt, wie ein Graph als Adjazenzmatrix abgebildet werden kann. Die Teilnehmer bilden die Menge  $KK$  der Kommunikationsknoten  $KK_i$ . Die Summe der Kommunikationsbeziehungen  $KB_{ij}$  bilden die Menge der Kommunikationsbeziehungen  $KB$ .



$KK_i$ : Kommunikationsknoten

$KB_{ij}$ : Kommunikationsbeziehung

Adjazenzmatrix des Beispielgraphen

Abbildung 4.10: Abbildungsbeispiel eines Graphen als Adjazenzmatrix

Die Abbildung des unvollständigen asymmetrischen Graphen durch eine Adjazenzmatrix, der vier Knotenelemente bzw. Knoten kennt, besitzt maximal  $2^n = 16$  Verbindungsbögen ( $n =$  Anzahl der Knotenelemente) der Knotenmenge  $KK$ . Bedingt durch die Schlichtheit des Graphen folgt, dass alle Diagonalelemente der Matrix 0 sind. Die unvollständige Symmetrie des abgebildeten Graphen führt dazu, dass die für alle vorhandenen  $K_{b_{ij}}$  auch die entsprechenden  $K_{b_{ji}}$  vorhanden sein müssen (siehe Abbildung 4.10).

#### 4.5 Stabilitätsbetrachtung von verteilten Kommunikationssystemen

Das Verhalten eines verteilten Kommunikationssystems wird durch stochastische und deterministische Einflussfaktoren bestimmt.

Unter deterministischen Einflussfaktoren versteht man unter anderem die Telegrammlänge, Anzahl verschiedener Telegrammtypen welche durch die Projektierung bedingt sind, Bearbeitungszeiten von Geräten und Signallaufzeiten auf dem Übertragungsmedium.

Als wichtigste stochastische Einflüsse sind Telegrammwiederholungen durch Störungen und eine prozessbedingte Steigerung des Datenaufkommens zu nennen.

Auf ein reales technisches System wirkt immer eine Kombination aus beiden Einflussfaktoren. In einem Simulationsmodell müssen beide Einflussfaktoren berücksichtigt werden. Sinnvollerweise sollte die Simulation eines Feldbussystems im ersten Durchlauf nur mit deterministischen Größen vorgenommen werden. In der zweiten Phase können dann Schritt für Schritt einzelne stochastische Größen hinzugefügt und verändert werden. Als Ergebnis dürften sich die im folgendem beschriebenen Betriebsfälle und der Auslastungsgrad des Systems ermitteln lassen.

Für die Stabilität eines verteilten Kommunikationssystems gilt allgemein:

$$I(t) < C \quad (4.3)$$

$I(t)$  : Informationsfluss zum Zeitpunkt  $t$

$C$  : Zur Verfügung stehende maximale Kanalkapazität

Diese Gleichung drückt aus:

Die Summe der Belegungszeiten aller Kommunikationsbeziehungen muss stets kleiner als die geforderte maximale Reaktionszeit des Systems sein. Nur so können alle Anforderungen rechtzeitig bedient werden.

Es kann unterscheiden werden:

- **Stabilitätsfall:** Anforderungen werden sofort bedient. Reserven für einen erhöhten Anforderungsstrom sind stets vorhanden
- **Grenzfall:** Anforderungen können gerade noch bedient werden. Reserven für einen erhöhten Anforderungsstrom sind nicht mehr vorhanden
- **Überlastfall:** Teile von Anforderungen können nicht mehr bedient werden. Reserven für einen noch höheren Anforderungsstrom sind nicht mehr vorhanden. Es gehen Informationen über den zu steuernden Prozess verloren.

Der Zustand eines verteilten Kommunikationssystems lässt sich in Stabilitätsfall, Grenzlastfall und Überlastfall einteilen.

#### 4.5.1 Der Stabilitätsfall

$$I(t) \ll C \quad (4.4)$$

Befindet sich ein verteiltes System in diesem Betriebsfall, kann es als stabil betrachtet werden. Das System hat ausreichend Reserven um alle Anforderungen, in der vom Prozess vorgegebenen Zeit, zu bedienen. Eine Erhöhung des Anforderungsstromes führt zu keinem unkontrollierten Systemverhalten. Der Anforderungsstrom kann z.B. durch Störungen auf dem Übertragungskanal hervorgerufen werden, da jetzt zusätzliche Telegramme übertragen werden müssen. Einen weiteren Einfluss haben auch durch den Prozess bedingte Erhöhungen der Buslast. In einem möglichen Fehlerfall kann die Menge der Prozessdaten, die erfasst, transportiert und verarbeitet werden sollen, rapide ansteigen.

Wenn alle zeitlich bekannten Größen eines Kommunikationssystems zusammengefasst werden, lässt sich eine minimale Reaktionszeit bzw. maximal mögliche Zykluszeit des Systems bestimmen. Es werden noch keine stochastischen Einflüsse, wie sie in realen Umgebungen auftreten, berücksichtigt. Dazu müssen in jedem Fall die Worst-Case-Werte der Busteilnehmer verwendet werden. Wenn Laufzeiten bei sehr großen Netzen berücksichtigt werden müssen sind diese eindeutig bestimmbar und sollten mit einem Sicherheitszuschlag versehen werden.

Der Ansatz die Worst-Case-Zeiten zu verwenden, führt immer zu einer Verschwendung von Übertragungskapazität (Alle Teilnehmer haben nicht die ungünstigsten Reaktionszeiten. Wie soll aber in diesem Fall reagiert werden?). Diese Verschwendung stellt aber eine zusätzliche Reserve (durchschnittlich sind die Teilnehmer besser als im Datenblatt) dar. Mit diesem Ansatz widerspiegelt ein Beratungssystem reale Komponenten der Automatisierungstechnik und ist unabhängig vom gewählten Feldbus. Bei PROFIBUS-DP können die maximalen Verzögerungszeiten beispielsweise aus den Gerätestammdateien der Hersteller entnommen werden.

#### 4.5.2 Der Grenzlastfall

$$I(t) < C \quad (4.5)$$

Der stochastische Anteil an Forderungen ist so groß geworden, dass sich das System an der Stabilitätsgrenze befindet. Es werden aber noch alle Forderungen bedient. Muss der Forderungsstrom noch weiter erhöht werden, können einzelne Forderungen nicht mehr bedient werden.

#### 4.5.3 Der Überlastfall

$$I(t) \geq C \quad (4.6)$$

Wenn sich ein verteiltes Kommunikationssystem in diesen Zustand befindet, kann vom Überlastfall gesprochen werden. Der Forderungsstrom ist jetzt größer als die reale Kapazität des Übertragungskanal. Das Gesamtsystem ist nicht mehr in der Lage alle Prozessdaten in der geforderten Zeit zu transportieren und zu verarbeiten. Das Prozessabbild im Steuerrechner entspricht nicht mehr zu 100% der Realität, da Informationen verloren gegangen sind. Fehlentscheidungen des Steuerrechners mit unterschiedlichen Auswirkungen können die Folge sein. Dieser Betriebsfall stellt ein unerwünschtes Ereignis in einem echtzeitfähigen Kommunikationssystem dar.

Das Konzept eines zu planenden Beratungssystems sieht vor, die drei Betriebsfälle mit mathematischen Methoden zu bestimmen. Der Überlastfall sollte durch Simulation mit einer gewissen Restfehlerwahrscheinlichkeit vermieden werden. Eine Restfehlerwahrscheinlichkeit kann nie vollständig ausgeschlossen werden, da immer unvorhergesehene Einflüsse zur Wirkung kommen die nicht vollständig erfasst werden können. Sie ist aber in ihrer Größe durch statistische Methoden bestimmbar.

Durch diese Fallunterscheidung wird keine Aussage über die Dauer der Lastzustände getroffen. Im Stabilitätsfall und Grenzfall ist diese Aussage auch ohne Bedeutung. Beim Überlastfall hat die Dauer schon einen gewichtigen Einfluss auf die Anlagensicherheit. Eine kurzzeitige Überlastung des Systems z.B. über nur eine Zykluszeit kann möglicherweise ohne Schwierigkeiten toleriert werden. Länger andauernde Überlastungen können auch Schäden an der Anlage hervorrufen. Dieses Problem erfordert eigenständige Untersuchungen auf dem Gebiet der Prozess- und Anlagensimulation.

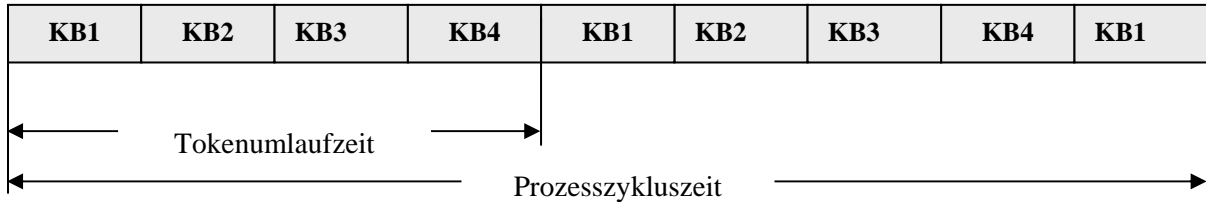
### 4.6 Verteilte Kommunikation am Beispiel von PROFIBUS-DP

Die ständige Aktualisierung des Prozessabbildes führt zu einer laufenden Belegung des Übertragungsmediums. Innerhalb der Tokenumlaufzeit wird in allen beteiligten Mastergeräten das Prozessabbild aktualisiert. In einem echtzeitfähigen PROFIBUS-System muss die Tokenum-



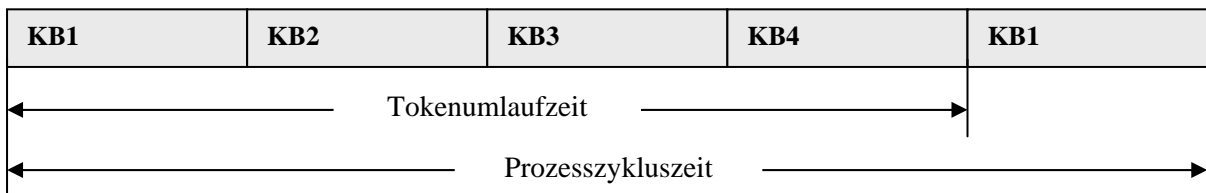
laufzeit stets kleiner als die geforderte Prozesszykluszeit sein. Die folgenden Abbildungen zeigen die typischen Betriebszustände eines verteilten Systems.

In den folgenden Abbildungen werden die typischen Betriebsfälle eines verteilten Systems am Beispiel von vier Kommunikationsbeziehungen innerhalb der Tokenumlaufzeit dargestellt.



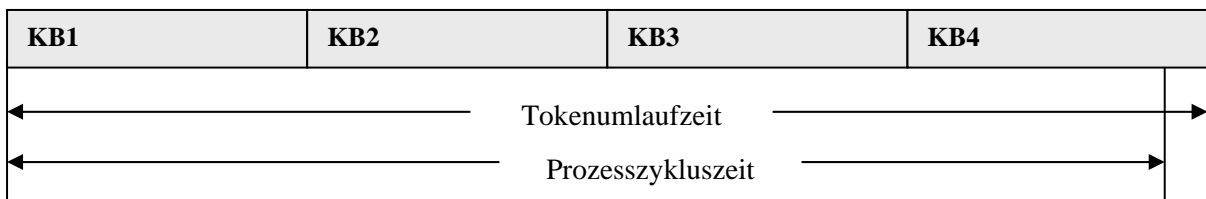
**Abbildung 4.11: Stabilitätsfall des PROFIBUS**

Die Tokenumlaufzeit ist viel kleiner als die geforderte Prozesszykluszeit. Das System kann alle Anforderungen bedienen.



**Abbildung 4.12: Grenzlastfall des PROFIBUS**

Die Tokenumlaufzeit ist noch immer kleiner als die geforderte Prozesszykluszeit. Das System kann alle Anforderungen gerade noch bedienen.



**Abbildung 4.13: Überlastfall des PROFIBUS**

Die Tokenumlaufzeit ist größer als die Prozesszykluszeit. Das System kann nicht mehr alle Anforderungen bedienen. Informationen über den zu steuernden Prozess gehen verloren.

#### 4.7 Probleme bei der Modellierung ethernetbasierter Kommunikationssysteme

Bei der Realisierung moderner Automatisierungssysteme steigt der Anteil ethernetbasierter Kommunikation in zunehmendem Maße. Die Kommunikationsknoten in einem verteilten industriellen Steuerungssystem kommunizieren ausschließlich in Echtzeit. Das klassische Ethernet als auch industrietaugliche Ethernettechnologien basieren grundsätzlich auf einen stochastischen Zugriff auf das Übertragungssystem. Die verschiedenen Verfahren für die Lösung des Widerspruchs Echtzeitkommunikation und stochastischer Zugriff auf das Übertragungsmedium sind in Abschnitt 3 dieser Arbeit vorgestellt worden. Für eine abstrakte Modellbildung derartiger Kommunikationssysteme mittels graphenbasierter Methoden ist eine Aufteilung der Belegungszeiten des Kommunikationskanals in einen deterministischen und einen stochastischen Anteil von zwingend notwendiger Voraussetzung. Eine präzise Einhaltung des Zeitmanagements im Kommunikationssystem stellt dabei eine wesentliche Grundvoraussetzung dar.

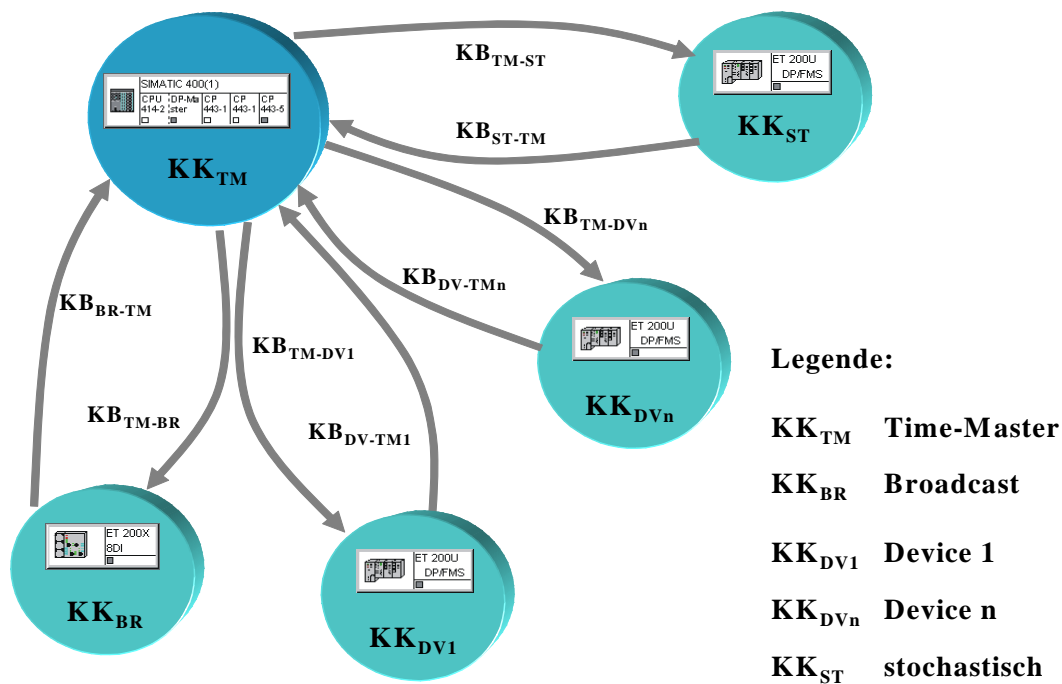


Abbildung 4.14: Netzabwicklung ethernetbasierter Kommunikationssysteme

Die meisten ethernetbasierten echtzeitfähigen Kommunikationssysteme können in abstrakter Weise mit dem Zustandgraphen nach Abbildung 4.14 beschrieben werden. Für die Modellierung durch Phasenmodelle gelten dabei die grundsätzlichen Feststellungen für die Abwicklung von Concast-Kommunikationsbeziehungen nach Abschnitt 4.4.6. Für die Modellierung des deterministischen Teils der Kommunikation ist das Phasenmodell in gleicher Weise geeignet. Ein Phasenmodell befindet sich laut Definition dabei immer genau in einem Zustand, was mit der zwingend notwendigen zeitlichen Vorhersagbarkeit der Kommunikation in technischen Steuerungssystemen korreliert. Im Beispielgraph sind das die Kommunikationsbeziehungen  $KB_{TM-BR}$ ,  $KB_{BR-TM}$ ,  $KB_{TM-DV1}$ ,  $KB_{DV1-TM}$ ,  $KB_{TM-DVn}$  und  $KB_{DVn-TM}$ . Wobei die Broad-

castkommunikation  $KB_{BR-TM}$  als imaginäre Kommunikationsbeziehung zu betrachten ist, um eine Verklemmung des Phasenmodells zu verhindern. Gegenüber der Modellierung von Master-Slave basierten Kommunikationssystemen ist eine wesentliche Erweiterung hinzugekommen, welche sich auf die Modellierung ethernetbasierter Kommunikationssysteme durch Phasenmodelle erschwerend auswirken kann. Viele Ansätze von Industrial Ethernettechnologien basieren auf der Koexistenz eines deterministischen und stochastischen Anteils des Verkehrsaufkommens, wobei sich der zweite Anteil immer dem deterministischen unterordnet. Wird auf eine ausführliche Modellbildung im stochastischen Zeitfenster verzichtet, wozu in jedem Fall Methoden der mathematischen Statistik angewandt werden müssen, kann dieser Teil der Kommunikation mit  $KB_{TM-ST}$  und  $KB_{ST-DV1}$  mit dem imaginären Kommunikationsknoten  $KK_{ST}$  modelliert werden. Der Kommunikationsknoten  $KK_{ST}$  wird dabei als Black-Box betrachtet, welcher in seinem fest zugewiesenen Zeitfenster das stochastische Verkehrsaufkommen (Anforderungsströme) des Kommunikationssystems beschreibt. Auf eine Darstellung möglicher Forderungslaufpläne wie in den Abbildungen 4.6, 4.7, 4.8 und 4.9 vorgenommen muss an dieser Stelle verzichtet werden, da die Entwicklung von Industrial Ethernet zum gegenwärtigen Zeitpunkt ein dynamisches Themengebiet darstellt und viele Spezifikationen noch nicht veröffentlicht bzw. in laufender Überarbeitung sind.

#### 4.8 Zusammenfassung und Bewertung

Die Modellierung eines verteilten Systems ist mit einem Graphenmodell prinzipiell möglich. Die Kommunikationsknoten  $KK$  repräsentieren die einzelnen Automatisierungsgeräte im verteilten System. Die Kommunikationsbogen  $KB$  beschreiben die Art und Weise der Kommunikationsbeziehungen zwischen den Automatisierungsgeräten. Mit dem Graphenmodell lässt sich ein allgemeiner Ansatz für die Modellierung eines verteilten industriellen Kommunikationssystems finden. Die Kommunikationsformen Multicast, Multipeer, Broad- und Unicast lassen sich im Graphenmodell gut darstellen. Die Kommunikationsform Concast, gegenwärtig nur bei wenigen Feldbussystemen verwendet, kann durch Einführung von imaginären Kommunikationsbeziehungen ebenfalls modelliert werden.

Die aufgabengemäße, notwendige Aufteilung einer Kommunikationsbeziehung in Teilabläufe kann in einem Phasenmodell weiter dekomponiert werden. Die für die Abwicklung des Ablaufes in graphenbasierten Modellen notwendigen Abarbeitungsalgorithmen stellen eine klassische wissenschaftlich technische Aufgabe dar. Eine Modellierung des Zeitverhaltens der Kommunikation zwischen den Kommunikationsknoten ist mit Verhaltensdiagrammen der UML 2.0 Spezifikation ebenfalls möglich, wurde aber zum Bearbeitungszeitpunkt nicht näher untersucht. Für die Modellierung echtzeitfähiger ethernetbasierter Kommunikationssysteme ist das Graphenmodell bedingt geeignet. Der deterministische Teil der Kommunikation kann analog Master-Slave basierter Kommunikationssysteme graphenbasiert beschrieben werden. Eine Aussage über das Verhalten im stochastischen Teil der Kommunikation ist nur mit Methoden der mathematischen Statistik möglich<sup>50</sup>.

---

<sup>50</sup> Siehe Abschnitt 3.6.4.1 [3.07]

## 5 Synchronisation von technischen Prozessen in MIKRO-LAN's

### 5.1 Aufgabenstellung

Die zentrale Aufgabenstellung dieses Teilgebietes der Arbeit war die Entwicklung einer neuen Operationstischgeneration für die Trumpf Medizinsysteme GmbH Saalfeld [5.03][5.04]. Folgende Anforderungen standen dabei im Vordergrund:

- Einsatz wartungsarmer intelligenter Antriebssysteme
- Modulare Erweiterbarkeit des Gesamtsystems
- PC-basierte Diagnose des Systems mit der Option zur Fernwartung
- Bereitstellung eines erweiterten Arbeitsraums gegenüber vergleichbaren Systemen

In diesem Kapitel wird ein Konzept und die Realisierung eines Steuerungssystems als verteiltes echtzeitfähiges System für die zentrale Säulensteuerung als Teilkomponente des Gesamtsystems vorgestellt und auszugsweise mit heute verfügbaren Systemen zur Synchronisation von Bewegungsabläufen verglichen.

### 5.2 Entwicklungsstand im medizinischen Gerätebau

Es gibt verschiedene Konzepte für die Realisierung von Gerätessystemen zur Patientenbewegung im stationären Bereich (Operationssäle). Im europäischen Raum werden von ca. 20 [5.11-5.30] Herstellern<sup>51</sup> Systeme mit unterschiedlichen Leistungsparametern angeboten. Dazu gehören rein mechanische Lösungen, elektrohydraulische Realisierungen, sowie elektromechanische Konzepte.

Sowohl in der allgemeinen industriellen Steuerungstechnik als auch im allgemeinen Gerätebau ist seit Beginn der neunziger Jahre der Übergang von der zentralistischen Steuerungstechnik zu dezentralen Steuerungskonzepten eingeleitet worden. Der Trend zu verteilten Steuerungssystemen ist dabei eindeutig erkennbar.

Viele Ansätze für die Synchronisation von präzisen Bewegungsabläufen basieren dabei gegenwärtig noch immer auf zentralistischen Steuerungskonzepten, welche selten modularisiert und verteilt sind, sowie kaum eine dynamische Funktionserweiterung nach Abschnitt 2.2.2 zulassen.

Die Ursachen für den Einsatz noch stark zentralistischer Steuerungskonzepte bei der Realisierung solcher automatisierungstechnischer Aufgabenstellungen liegen an verschiedenen Stellen:

- Es handelt sich hierbei meist um räumlich sehr lokal abgeschlossene Systeme, bei denen die Probleme der allgemeinen industriellen Steuerungstechnik, wie die Reduzie-

---

<sup>51</sup> Auszug, vorrangig europäische Hersteller

rung des Verkabelungsaufwandes oder die Erhöhung der Störsicherheit nicht primär im Vordergrund stehen.

- Steuerungssysteme zur Synchronisation von Prozessen, insbesondere der Antriebssynchronisation erfordern, ein hochpräzises Zeitmanagement mit einer möglichst geringen Schwankung der physikalischen Größe Zeit. Die primäre Forderung bei solchen Steuerungssystemen lautet dabei: Das System muss sich in jedem Fall zur richtigen Zeit am richtigen Ort befinden. Hochpräzise Antriebssynchronisationen stellen nach dem gegenwärtigen Stand der Technik die höchsten Anforderungen an die Echtzeitfähigkeit bei zentralistisch aufgebauten als auch bei verteilt realisierten Steuerungskonzepten<sup>52</sup>.
- Für die Realisierung verteilter Steuerungssysteme zur Bewegungssynchronisation sind im Normalfall spezielle Kommunikationsprotokolle und Strukturen erforderlich.

### 5.3 Ausgangssituation und Lösungsansätze

Zum Zeitpunkt der Konzipierung einer neuen Operationstischgeneration wurden Bewegungsabläufe, sowie die erforderlichen Synchronisationen von Teilbewegungen nach Abschnitt 5.2 auf mechanischer, elektromechanischer oder hydraulischer Basis, sowie mit Kombinationen daraus realisiert. Steuerungstechnisch leicht beherrschbare Achsanordnungen sind insbesondere im Bereich der zentralen Säulensteuerung international patentrechtlich geschützt. Als Antriebssysteme werden in elektromechanischen Systemen meist geregelte Elektromotoren ohne Kommunikationsinterface eingesetzt. Anwenderspezifische Funktionserweiterungen, welche sehr vorteilhaft durch modulare, komplexitätsreduzierende Konzeptionen realisierbar wären, lassen sich nur in bedingtem Umfang realisieren.

Es ergeben sich daraus folgende wichtige Problemstellungen:

- *Hydraulikbasierte Systeme können wegen des erforderlichen Mediums Öl als nicht wartungsarm eingestuft werden*
- *Komplizierte Arbeitsraumüberwachungen lassen sich mit rein elektromechanischen Systemen schwierig realisieren.*
- *Ein modularer skalierbarer Aufbau ist mit elektromechanischen oder hydraulischen Systemen nicht oder nur schwer realisierbar.*

Zum Vergleich: Für die Automobiltechnik, deren Steuerungssysteme nach Abschnitt 2 ebenfalls als lokal echtzeitfähig eingestuft werden können, sind folgende charakteristische Eigenschaften formuliert [3.70]:

- *Durch elektronische Verkabelung werden aufwendige Mechanik- und Hydraulikteile überflüssig.*

---

<sup>52</sup> Siehe auch Abbildung 3.6 Klassifizierung der Echtzeitforderungen der IAONA

- Die Rechenleistung für das Gesamtsystem kann auf alle im System vorhandenen Knoten aufgeteilt werden.
- Die Anzahl der Sensoren kann verringert werden, weil ihre Messdaten im gesamten Netzwerk zur Verfügung stehen.
- Die Software ist bei Einbau entsprechender Sensorik- und Sicherheitsfunktionen erweiterbar.
- Softwareentwicklung für echtzeitfähige verteilte Steuerungssysteme unterliegt besonderen Anforderungen

Moderne industrielle Steuerungskonzepte lassen sich nach [3.72] wie folgt klassifizieren:

- zentrale Steuerung mit zentralen I/O
- zentrale Steuerung mit dezentralen I/O
- zentrale Steuerung mit intelligenten Modulen (Intelligente Devices)
- verteilte Steuerung mit intelligenten Modulen (Intelligente Devices)

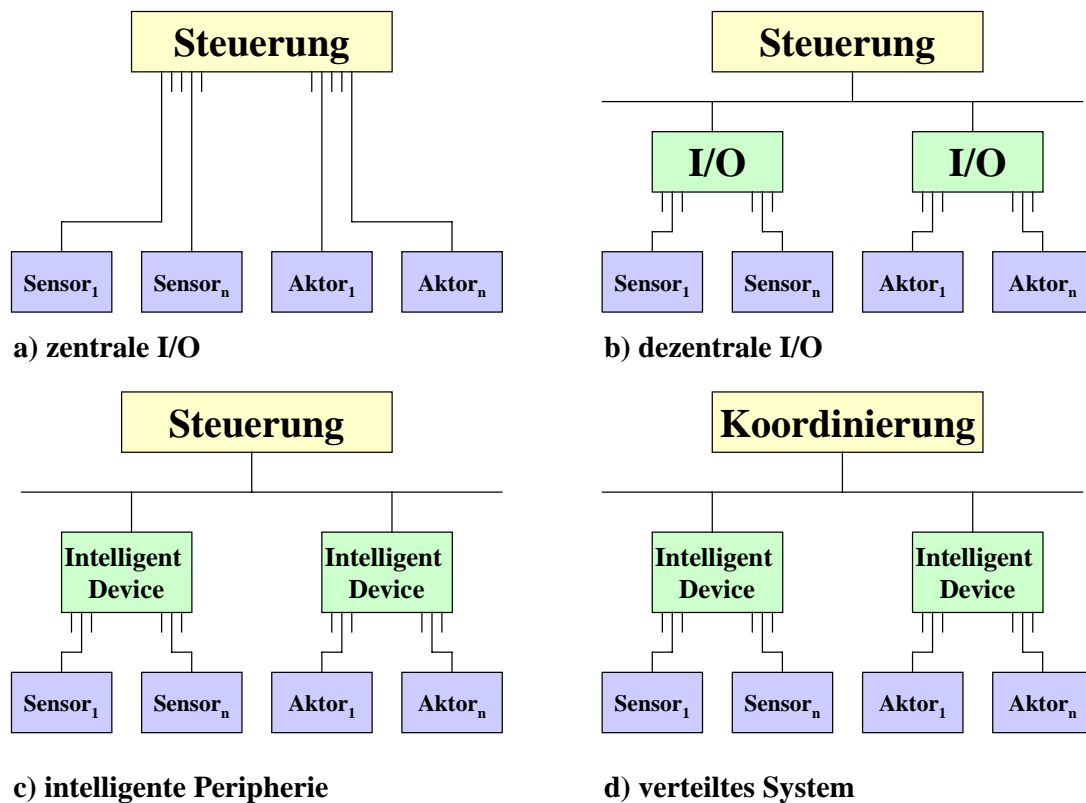


Abbildung 5.1: Klassifikation von industriellen Steuerungskonzepten

In Abbildung 5.1 sind die, in der industriellen Steuerungstechnik vorherrschenden Steuerungskonzepte in ihrem prinzipiellen strukturellen Aufbau vorgestellt. Die Konzepte nach Abbildung 5.1a/b sind durch klar definierte Unterscheidungsmerkmale charakterisiert. Mit

beiden Konzepten können echtzeitfähige Steuerungssysteme mit den verschiedenen Anforderungen, welche in Abschnitt 2.4 betrachtet werden realisiert werden. Den Anspruch verteilt zu sein, erfüllt das Konzept Abbildung 5.1b nach der Definition von Tanenbaum (siehe Abschnitt 2.1) nur im bedingten Maße. Bei diesem Steuerungskonzept muss davon ausgegangen werden, dass aufgrund der Busankopplung, welche eine elektrische Informationsverarbeitung erforderlich macht, die dezentralen I/O-Komponenten mit Computern (heute fast ausschließlich mit Mikrocontrollern) ausgestattet sind. Aus Sicht der Verteilung von Funktionalität, zwecks skalierbarer Erweiterbarkeit des Gesamtsystems, stellt dieses Steuerungskonzept kein verteiltes System dar.

Dem Gegenüber stehen die Konzepte nach Abbildung 5.1c/d in einer fließenden Korrelation zueinander. Charakteristisch für beide Konzepte ist die Verlagerung von Funktionalität in intelligente Devices. Diese werden bei dem Konzept nach Abbildung 5.1c durch ein zentrales Steuerungssystem verwaltet. Für die Realisierung solcher Steuerungskonzepte werden aus informationstechnischer Sicht Kommunikationssysteme eingesetzt, welche mit zentralen Zugriffsverfahren arbeiten. Die Echtzeitfähigkeit des Gesamtsystems wird durch den zentralen Zugriff auf das Kommunikationsmedium grundsätzlich gewährleistet. Die Einhaltung der Echtzeitanforderungen nach den vier Fällen nach Abschnitt 2.4.2 wird durch entsprechende Protokollspezifikationen realisiert.

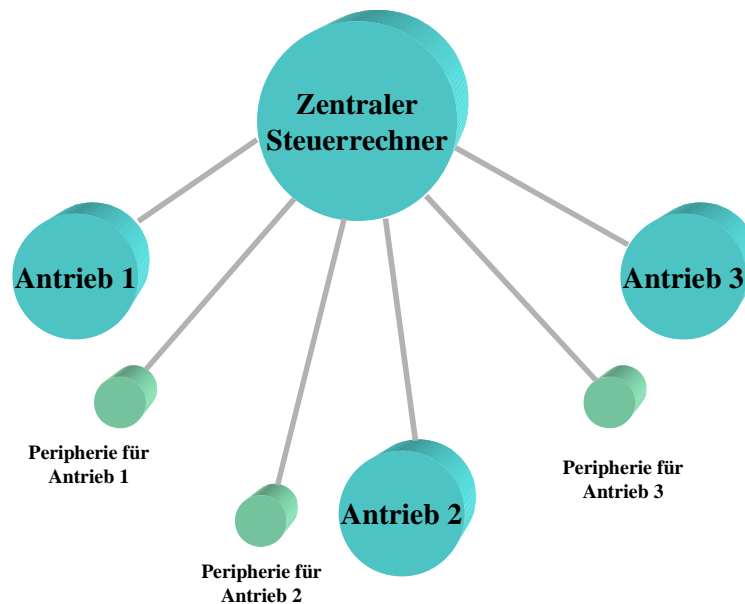
Das Steuerungskonzept, mit den Merkmalen von Abbildung 5.1c, kann nach der Definition in 2.1 als verteiltes System der industriellen Steuerungstechnik eingestuft werden.

Das Steuerungskonzept nach Abbildung 5.1d ist dadurch charakterisiert, dass es im Sinne der Automatisierungstechnik keine zentrale Steuerung gibt. Einzelne Devices können sich zu Gruppen zusammenschließen um eine gestellte automatisierungstechnische Aufgabenstellung zu realisieren. Diese Art von Steuerungskonzepten erfordert aus informationstechnischer Sicht grundsätzlich stochastische Zugriffsverfahren auf die Kommunikationskanäle (z.B. Ethernet und CAN), da die Devices selbständig miteinander kommunizieren müssen. Unter Echtzeitbedingungen, wie in der industriellen Steuerungstechnik zwingend erforderlich, ist bei diesem Konzept dennoch eine zentrale Steuerung erforderlich. Der wesentliche Unterschied zu Abbildung 5.1c besteht jedoch darin, dass diese Steuerung keine Aufgaben hinsichtlich der bisherigen Automatisierungstechnik mehr übernimmt. Die Steuereinheit wird mit der Funktionalität der Aufgabenverteilung, aber viel wichtiger mit der Gewährleistung eines einheitlichen Zeitmanagements des Gesamtsystems beauftragt und kann als Koordinator betrachtet werden.

Daraus lässt sich folgende Schlussfolgerung für das wissenschaftliche Teilgebiet der Automatisierungstechnik „Verteilte industrielle Steuerungssysteme“ ableiten:

Vollständig verteilte auf Kommunikation basierende technische Systeme und echtzeitfähige Steuerungstechnik stehen ohne besondere informationstechnische Maßnahmen im Widerspruch zueinander.

### 5.3.1 Konventionelles zentralisiert organisiertes Steuerungskonzept



**Abbildung 5.2: Zentralisiert organisiertes Steuerungskonzept für die Bewegungssynchronisation**

Steuerungskonzepte für die echtzeitfähige Synchronisation von Bewegungsabläufen im Sinne der Aufgabenstellung lassen sich prinzipiell nach der Strukturen, welche in Abbildung 5.2 dargestellt sind, realisieren. Hinsichtlich des Echtzeitverhaltes weisen derartige Strukturen, eine entsprechende Verarbeitungsleistung der CPU vorausgesetzt, für den Einsatz in Bewegungssteuerungen hervorragende zeitliche Eigenschaften auf. In [5.06] ist eine entsprechende Realisierung durchgeführt worden, wo bekannte Grafikalgorithmen unter Echtzeitbedingungen nach Fall 1 die Bahnsteuerung von Schrittantriebssystemen realisieren. Die Berechnung der nächsten folgenden Position, die Ermittlung der aktuellen Ansteuerfrequenz des aktuellen Zyklus sowie der entsprechende Ausgabe an die Aktorik erfolgt dabei unter harten Echtzeitbedingungen (12 Khz Schrittfrequenz -> 84µs).

- Vorteile:** Sehr gutes Echtzeitverhalten des Gesamtsystems
- Geringe Zykluszeiten mit variabler Länge möglich (sehr hohe Dynamik)
- Es erfolgt keine externe Kommunikation zwischen den Teilsystemen. Eine Analyse, Bewertung und Optimierung von Belegungszeiten auf Bussystemen sowie Durchlaufzeiten durch entsprechende Protokollstacks ist wesentlich vereinfacht.
- Nachteil:** Die Softwareentwicklung wird mit zunehmender Anzahl der Sensorik und Aktorik-Komponenten unübersichtlich
- Es erfolgt keine räumliche Verteilung der Funktionalität
- Das System wird nicht modular realisiert
- Alle regelungstechnischen Algorithmen werden zentral abgearbeitet



### 5.3.2 Kommunikationsorientierte verteilte organisierte Steuerungskonzepte

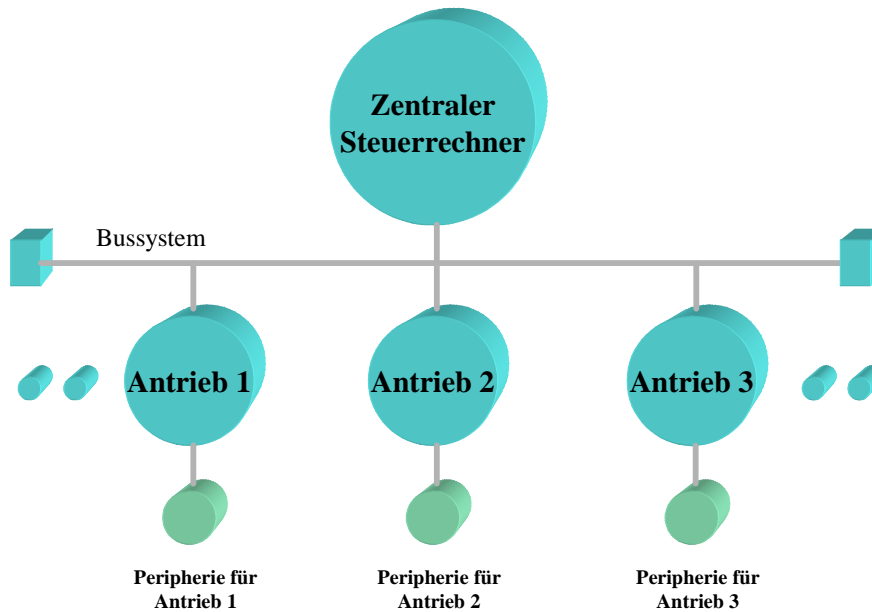


Abbildung 5.3: Dezentralisiert bzw. verteilt organisiertes Konzept für Bewegungssynchronisationen

Moderne Steuerungskonzepte zur Synchronisation von Bewegungsabläufen werden bei größeren räumlichen Ausdehnungen (z.B. bei Druck- und Verpackungsmaschinen) mit Konzepten der Dezentralisierung und zunehmend durch Verteilung von Funktionalität realisiert. Zentralisierte Realisierungen wie in Abbildung 5.2 dargestellt lassen sich aus Gründen der steigenden der Komplexibilität der Softwaresysteme in der zentralen Steuerungseinheit als auch des erhöhen Aufwandes der Verkabelung mit der Sensor- und Aktorebene zunehmend schwerer realisieren. Die Forderungen nach der Reduzierung des mechanischen Aufwandes als auch die zunehmende Steigerung der Präzision der Synchronisation von Bewegungsabläufen beschleunigen den Entwicklungstrend. Realisierungen derartiger Steuerungskonzepte wie in Abbildung 5.3 prinzipiell darstellt, waren zu Beginn der Arbeiten speziellen, nicht offenen Kommunikationssystemen<sup>53</sup> vorbehalten. Heute stellt der Einsatz derartiger Konzepte den Stand der Technik dar und wird zunehmend von Standard-Feldbussystemen als auch von Industrial Ethernettechnologien unterstützt.

### 5.3.3 Schlussfolgerungen und Festlegung des Steuerungskonzeptes

Nach der Analyse der speziellen Aufgabenstellung und den Untersuchungen zu einsetzbaren Architekturkonzepten, Echtzeitverhalten und verfügbaren Kommunikationssystemen wurde für das Konzept des Gesamtsystems sowie insbesondere für das Teilsystem „Zentrale Säulensteuerung“ die Realisierung nach Abbildung 5.1c favorisiert.

Im folgenden Abschnitt soll das Client/Server-Konzept als Modellierungsstrategie für die echtzeitfähige Kommunikation zur verteilten Synchronisation von Bewegungsabläufen am

<sup>53</sup> Wesentlichster Vertreter ist SERCOS I /II (III)

Beispiel der drei Antriebssysteme für die Zentralsäule des Operationstischs Jupiter betrachtet werden.

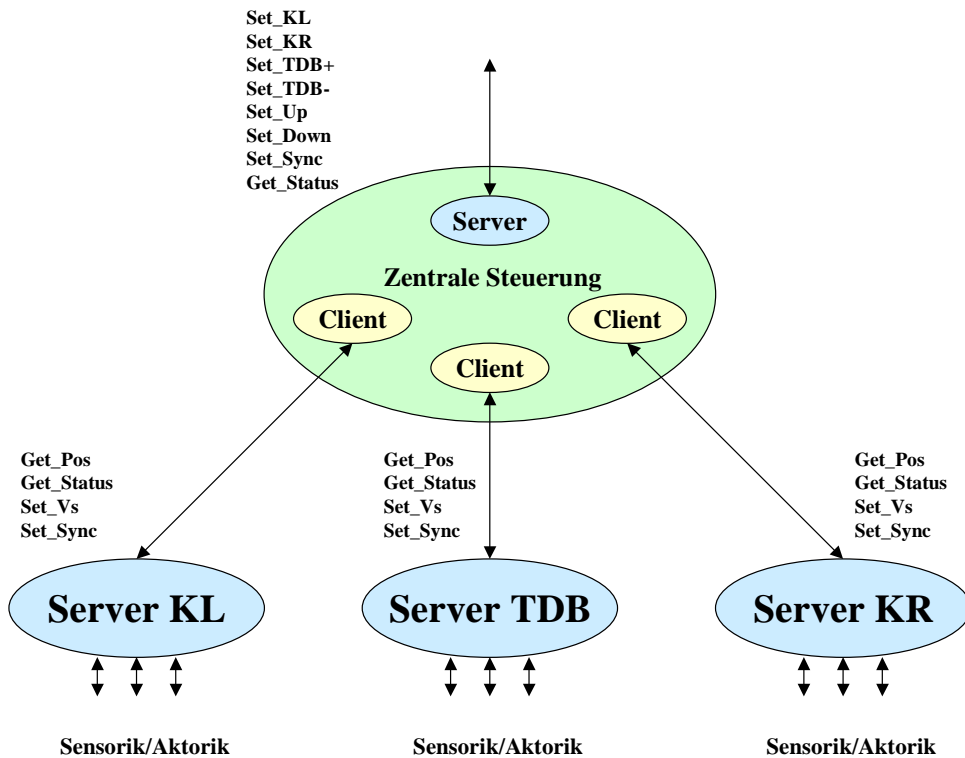


Abbildung 5.4: Client/Server Model für die Säulensteuerung als verteiltes System

Für die Kommunikation zwischen der zentralen Steuerungskomponente mit seiner übergeordneten Steuerungskomponente und seinen intelligenten Devices sind folgende wesentliche echtzeitfähige Dienste erforderlich:

Set_TDB+	Dienst System Trendelenburg <sup>54</sup>
Set_TDB-	Dienst System Anti-Trendelenburg
Set_KL	Dienst System Kantung Links
Set_KR	Dienst System Kantung Rechts
Set_Up	Dienst System nach oben
Set_Down	Dienst System nach unten
Set_Sync	Dienst Zeit synchronisieren
Set_Start	Dienst Start Motion
Set_Vs	Dienst Geschwindigkeit setzen
Get_Status	Dienst Status abfragen
Get_Pos	Dienst aktuelle Position lesen

Tabelle 5.1: Wichtige bereitgestellte Dienste (Auszug)

<sup>54</sup> Friedrich Trendelenburg Chirurg (1895-1911) Direktor der Chirurgischen Universitätsklinik Leipzig  
Trendelenburg entwickelte die Methode der Beckenhochlagerung. Nach ihm wurde dieser spezielle Bewegungsvorgang bei Operationstischen benannt.

#### 5.4 Algorithmische Umsetzungen (Auszugsweise)

Um die geforderte Funktionalität in der gewünschten Güte bereitzustellen und die einzelnen Komponenten hinsichtlich Zeit und Ort wie in Abschnitt 2 beschrieben zu koordinieren, sind entsprechende echtzeitfähige Algorithmen erforderlich.

Wie aus Abbildung 5.3 und Tabelle 5.1 hervorgeht, sind dabei verschiedene Dienste mit entsprechenden komplexen Ablaufvorschriften in das Steuerungssystem zu implementieren. Auf eine detaillierte Beschreibung softwaretechnischer Realisierungen soll jedoch an dieser Stelle verzichtet werden.

Ein Teilalgorithmus für die Synchronisation von Bewegungsabläufen durch verteilte Komponenten ist in Abbildung 5.5 dargestellt. Die kinematischen Grundlagen sind auszugsweise im Anhang dargestellt. Im Wesentlichen wird dabei wie folgt vorgegangen:

Bei jeder eingehenden Dienstanforderung wird zuerst die Prüfung durchgeführt, ob der Systemstatus in Ruhe oder Bewegung ist. Im Ruhezustand kann somit jede gültige Dienstanforderung bearbeitet werden, z.B. System nach „Oben“ fahren. Es werden dabei entsprechende Kommunikationsbeziehungen mit den untergeordneten Systemen durchgeführt und der Dienstanforderung an das Zentralsystem entsprechend quittiert.

Befindet sich das System in Bewegung erfolgt zunächst die Überprüfung ob der angeforderte Dienst im aktuellen Zustand zulässig ist oder nicht. Ist der Dienst zum aktuellen Zeitpunkt nicht zulässig wird, er nicht ausgeführt und entsprechend quittiert. Wenn der Dienst zum aktuellen Zeitpunkt zugelassen ist, werden die entsprechenden Kommunikationen mit den Antriebsknoten durchgeführt um die erforderlichen Informationen über aktuelle erreichte Positionen sowie Betriebeszustände einzulesen. Danach wird der dem geforderten Dienst zugehörige Synchronisationsalgorithmus abgearbeitet und die Sollwerte für den nächsten Zyklus in die Antriebsknoten geschrieben. Abschließend wird das übergeordnete System mit der entsprechenden Quittung über die Ausführung des Dienste informiert.

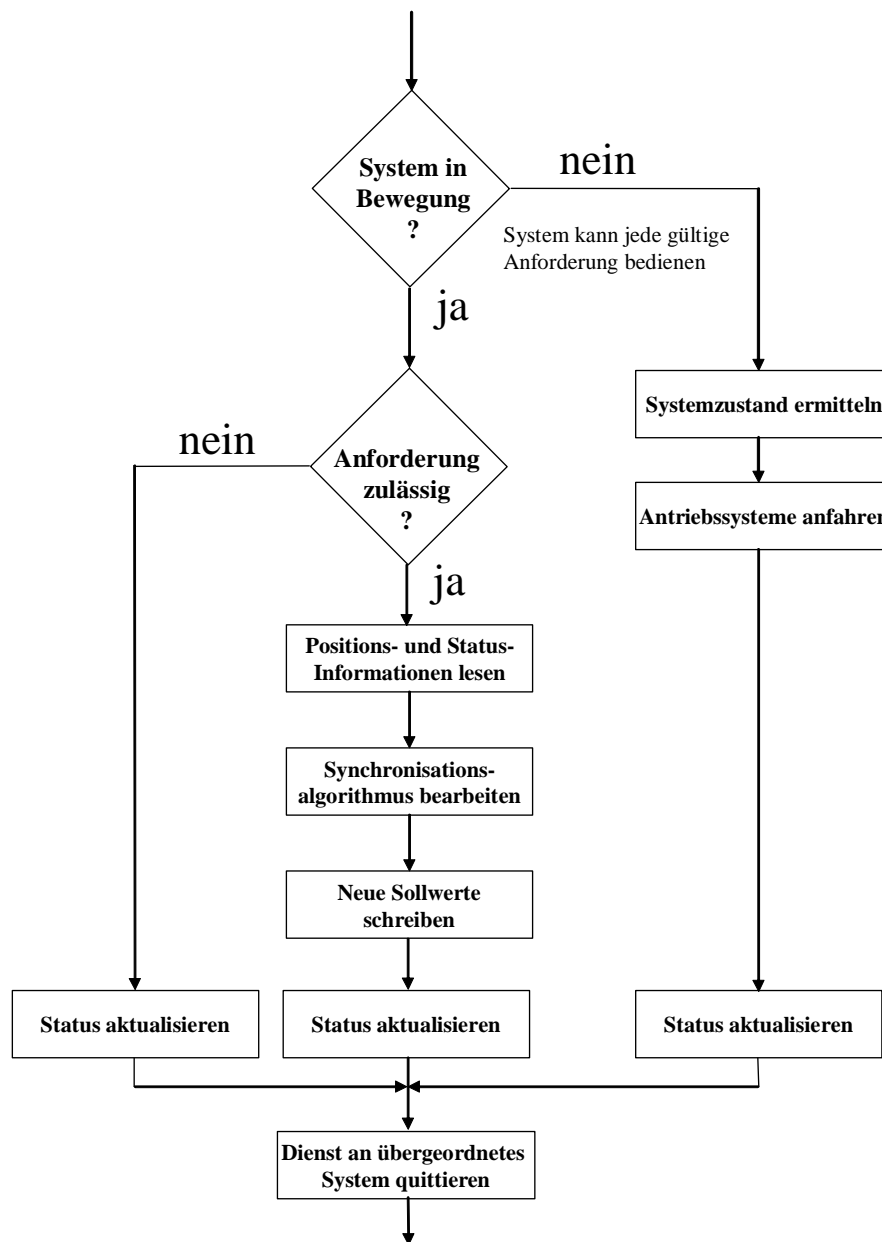


Abbildung 5.5: Algorithmus für die Bewegungssynchronisation (Auszug)

Das Steuerungssystem für die Realisierung der Aufgabenstellung „Feinpositionierung eines Patienten“ ist als lokal abgeschlossenes verteiltes echtzeitfähiges System in ein weiteres verteiltes System integriert.

Dieses Subsystem charakterisiert sich durch folgende Merkmale:

- Verteilter Aufbau der Funktionalität
- Funktionalität eines Gateways
- echtzeitfähige Realisierung des Synchronisationskonzepts

## 5.5 Integration des Teilsystems in ein Gesamtsystem

Die vorgestellte Lösung zur Synchronisation von Bewegungsabläufen als verteiltes Steuerungssystem integriert sich, wie in Abbildung 5.6 gezeigt in ein übergeordnetes Gesamtsystem. Das informationstechnische Gesamtkonzept wurde dabei in Zusammenarbeit mit der Firma Trumpf Medizinsysteme entwickelt, zentral koordiniert und die Anforderungen an die Subsysteme festgelegt. Das mechanische Konzept des Gesamtsystems wurde bei diesem Forschungs- und Entwicklungsvorhaben ausschließlich durch die Trumpf Medizintechnik entwickelt. Alle Softwarekomponenten zur Bewegungssynchronisation außerhalb der Steuereinheit für den Zusatzhub wurden von Herrn P. Schütz von der Tetra GmbH Ilmenau konzipiert und realisiert.

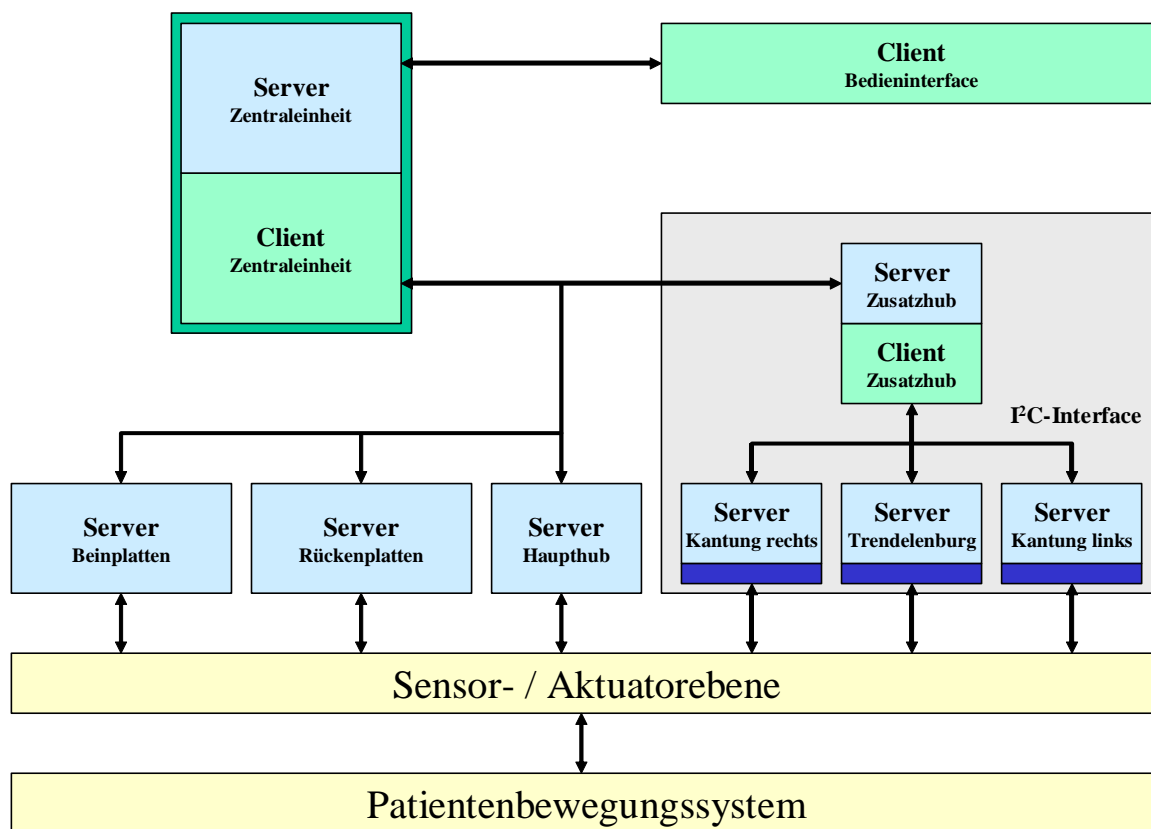


Abbildung 5.6: IT-Struktur des Gesamtsystems

Die Funktionsblöcke sind dabei im Wesentlichen wie folgt aufgeteilt:

- Benutzerschnittstelle (Bediener vor Ort oder Fernwartung)
- Steuerungssystem (Zentrale Intelligenz des Systems)
- Steuerungssystem für die Zentrale Säule (Koordination des Dreiachsensystems)

## 5.6 Projekthistorie der Entwicklung des Teilsystems

Firmwarestand	Entwicklungsjahr	Wesentliche Merkmale
V 5.01	I/1998	Entwicklungsbeginn und Konzepterstellung
V5.02	II/1998	Entwicklungsbeginn und Konzepterstellung Erste Tests unter Laborbedingungen
V5.03	IV/1998	Entwicklungsbeginn und Konzepterstellung Erste Tests des Steuerungskonzepts in der Praxis
V5.04	II/1999	Realisierung und Praxistests abgeschlossen - > Beginn der Serienproduktion
V5.05	I/1999	Erweiterung des Softwaresystems um zusätzliche Module für die Eigendiagnose Konzept der Synchronisation von Bewegungsabläufen in MIKRO-LAN dessen praktische Realisierung wird mit der Überleitung in die Serienproduktion bestätigt
V5.05	I/2000	Abgabe der logistischen Betreuung des Steuerungssystems durch den Autor <sup>55</sup>
V5.06	III/2000	Fehler im I <sup>2</sup> C Management korrigiert
V5.07	I/2003	Fehler in der Arbeitsraumüberwachung korrigiert Kern des Konzepts der Bewegungssynchronisation durch MIKRO-LAN ist nach wie vor im praktischen Einsatz

**Tabelle 5.2: Entwicklungsstände mit wesentlichen Merkmalen [5.32]**

Produktionsjahr	Produzierte Stückzahl
1998 (Nullserie)	
1999	
2000	
2001	
2002	
2003	
2004	
2005	
2006	

**Tabelle 5.3: Produzierte Einheiten des „Jupiter“ seit Beginn der Markteinführung [5.32]**

<sup>55</sup> Der Autor ist seit 04/2000 wissenschaftlicher Mitarbeiter des Institutes für Automatisierungstechnik der TU-Freiberg

## 5.7 Zusammenfassung

Zu Beginn der Arbeiten wurden vorrangig mechanische, elektromechanische bzw. elektrohydraulische Systeme zur Lösung von derartig gelagerten Aufgabenstellungen eingesetzt. Steuerungskonzepte für elektromechanisch basierte Systeme wurden und werden nach wie vor nach den Konzepten zentralisierter Steuerung mit zentralen I/O ausgelegt<sup>56</sup>.

Als wesentliches Ergebnis der Forschungs- und Entwicklungsarbeiten steht die Erkenntnis, dass sich steuerungstechnische Aufgabenstellungen wie die Synchronisation von Prozessen insbesondere der Bewegungssynchronisation durch verteilte echtzeitfähige Steuerungssysteme realisiert werden können. Im vorgestellten Einsatzfall, der Bewegungssynchronisation in einem Patientenbewegungssystem, leistet die vorgestellte Steuerungskonzeption zum Zeitpunkt der Bearbeitung des Themengebietes einen Beitrag zur Ablösung zentralistisch orientierter Steuerungssysteme in diesem Bereich des medizinischen Gerätebaues.

Weiterhin lässt sich aus den Recherchen über andere verfügbare Systeme zur Bewegungssynchronisation eine zweite wichtige Erkenntnis gewinnen. Es ist trotz verfügbarer, öffentlicher zugänglich gemachter Systeme zur Antriebssynchronisation (zum Zeitpunkt der Arbeiten SERCOS-I) sinnvoll, für solche sehr spezielle Anwendungsfälle an alternativen Konzepten zur Synchronisation von Bewegungsabläufen zu arbeiten. Diese Feststellung kann einerseits mit der sehr umfangreichen Funktionalität von SERCOS-I, welche nicht in jedem Einsatzfall erforderlich ist, begründet werden. Andererseits erschweren die nicht offen bzw. nur ansatzweise offen gelegten Algorithmen und Protokolle dieser Systeme die Realisierung von Applikationen wie in diesem Kapitel vorgestellt. Auch trotz der schnellen Weiterentwicklung auf diesem technologischen Gebiet (SERCOS-III, Simotion, Synqnet, Äquidistanz bei PROFIBUS-DP usw.) handelt es sich auch heute um meist sehr universelle Systeme (mit vielfach besseren Leistungsparametern), deren Spezifikationen durch die Globalisierung der Automatisierungstechnik heute öffentlicher zugänglich sind, als zum Zeitpunkt der Projektrealisierung.

Es wird mit dem vorgestellten und realisierten Konzept der Antriebssynchronisation zur Patientenbewegung im stationären Bereich nachgewiesen, dass Architekturkonzepte aus der Informationstechnik wie die Client-Server Architektur sowie Modellansätze zur Verteilung von Funktionalität aus der Informationstechnik und Informatik zur Steuerung von dynamischen Prozessen auch in lokal abgeschlossen Systemen geeignet sind. Es wurde gezeigt, dass derartige Systeme, welche von ihrer räumlichen Ausdehnung jederzeit zentralisiert realisiert werden können, mit den in dieser Arbeit vorgestellten Konzepten modelliert, validiert und technisch realisiert werden können. In [3.51] sind 2004 wichtige Aussagen über Entwicklungstrends im modernen Maschinenbau veröffentlicht worden, welche in sehr guter Weise mit den realisierten Konzepten für dieses sehr spezielle Gebiet des Maschinenbaues korrelieren.

---

<sup>56</sup> Siehe Hersteller nach Abschnitt 5.2

## 6 Integration des Steuerungssystems SIMATC-S7 in das WAN

### 6.1 Motivation und Aufgabenstellung

Die moderne Fertigungs- und Prozessautomation stellt immer höhere Anforderungen an die Anlagenverfügbarkeit, die Reduzierung von Stillstandszeiten durch bessere Diagnoseverfahren, der Optimierung von Prozessabläufen, der Langzeitarchivierung von Prozessdaten um nur einige wesentliche zu nennen. Durch derartige Forderungen und der anhaltenden Globalisierung wirtschaftlicher Prozesse der heutigen Industriegesellschaft entsteht zunehmend die Notwendigkeit, dass lokal abgeschlossene echtzeitfähige steuerungstechnische Kommunikationssysteme miteinander weltweit kommunizieren müssen. Die Kommunikation zwischen den Systemen kann dabei auf verschiedenen Wegen, z.B. Modemverbindungen, erfolgen. Das Internet, mit seiner global gut ausgebauten Infrastruktur, bietet sich ebenfalls für die Lösung dieser Aufgabenstellung an und wurde von Herstellern automatisierungstechnischer Komponenten zu einem wichtigen Schwerpunkt der Forschung auf diesem Gebiet erklärt.

Im Jahr 1999 begann die Siemens AG mit einer deutschlandweiten Aktion, deren Start eine „KICK OFF“<sup>57</sup> Veranstaltung in Chemnitz war, webbasierte SPS-Applikationen auf der Grundlage der Steuerungsfamilie SIMATIC S7 zu fördern. Auf der darauf folgenden Tagung in Mittweida wurde beschlossen, verschiedene Projektgruppen zu bilden. Daraufhin begannen diese Projektgruppen die Möglichkeiten und Grenzen dieser sich rasant entwickelnden Kommunikationstechnologie zu untersuchen [6.01].

Basierend auf den Analysen zu echtfähigen Kommunikationssystemen, Architekturkonzepten und aktuellen Entwicklungstrends der Informationstechnik, wurden in Verbindung mit dem Industriepartner die Forschungsschwerpunkte auf den Bereich internetbasierter Zugriffe für dezentrale Peripheriekomponenten im Feldbussystem PROFIBUS gelegt. Dabei sollen bestehende Zugriffskonzepte analysiert und bewertet sowie neue Konzepte des Zugriffs auf diese Geräteklasse in verteilten Systemen entworfen werden.

#### **Internettechnologien und moderne industrielle Steuerungstechnik:**

**Es können folgende grundlegende Feststellungen getroffen werden, welche bei Arbeiten auf diesem Gebiet von primärer Bedeutung sind:**

Das Internet und heutige moderne Realisierungen auf Kommunikation basierender industrieller Steuerungssysteme weisen eine grundlegende Gemeinsamkeit auf:

- Verteilte Realisierung der Kommunikationssysteme

Es besteht aber auch ein wesentlicher Unterschied zwischen dem weltweit verfügbaren Kommunikationssystem und den Anforderungen der industriellen Kommunikation:

- Gegensätzliche Anforderungen hinsichtlich der Echtzeitfähigkeit

---

<sup>57</sup> Auftaktveranstaltung für die Kooperation der Siemens AG, der TU-Dresden, der TU-Freiberg und der Hochschule für Technik und Wirtschaft Mittweida auf dem Gebiet des Einsatzes von Internettechnologien in der Automatisierungstechnik.



Das Internet muss prinzipbedingt aus Sichtweise der industriellen Steuerungstechnik als nicht echtzeitfähig eingestuft werden [6.10]. Das betrifft insbesondere Applikationen aus dem Bereich der Automatisierung von hochdynamischen Prozessen, bei denen extreme Anforderungen an die Präzision der Zeitkonstanz (Gleichzeitigkeit) gestellt werden.

Dennoch kann und sollte sich die auf Informationsverarbeitung basierende echtzeitfähige Steuerungstechnik nicht vor den Möglichkeiten des weltweit verfügbaren Kommunikationsmediums verschließen.

In diesem Kapitel der Arbeit „Beitrag zur Verhaltensanalyse und Synchronisation von steuerungstechnischen Prozessen durch verteilte echtzeitfähige Kommunikationssysteme“ werden eine Analyse, Bewertung, Modellierung sowie eine prototypische Realisierung des „Ortsunabhängigen Zugriffes auf Automatisierungssysteme“ (OZA) über das Internet am Beispiel des SIMATIC S7 Steuerungssystem vorgenommen.

Dabei fließen wesentliche Ergebnisse aus der Forschungskooperation der TU - Bergakademie Freiberg, Institut für Automatisierungstechnik und der Siemens AG Geschäftsbereich A&D Standort Chemnitz ein.

Außerdem werden Ergebnisse aus dem Projekt „Interaktives Web-basiertes Labor Automatisierungstechnik“ als ein Teilprojekt im „Sächsischer internetbasierter Hochschulverbund der Automatisierungstechnik“, welche die Forschungsarbeiten tangieren, vorgestellt und bewertet.

## **6.2 Ausgangssituation und Stand der Informationstechnik**

Zum Zeitpunkt der Untersuchungen auf dem Gebiet webbasierter Zugriffskonzepte für Automatisierungskomponenten, welche prototypisch für die Diagnose von ET200-Komponenten realisiert werden sollten, stand ein Kommunikationsprozessor<sup>58</sup> [6.06], mit TCP/IP Interface und integriertem Web-Server in einer Prototypenversion, zur Verfügung. Mit dieser Komponente wird der Zugang eines SIMATIC-basierten verteilten echtzeitfähigen Steuerungssystems mit der entsprechenden Funktionalität zum Internet gewährleistet.

Für den Zugriff auf dezentrale Peripheriekomponenten stehen im Profibusssystem entsprechende Dienste zur Verfügung, welche zu einem Teil durch das Anwenderprogramm in der SPS als auch durch die Protokoll-Asic's autonom genutzt werden können. Dazu gehören (siehe Kapitel 3) zyklische Dienste für die ständige Aktualisierung des Prozessabbildes, azyklische Dienste für das Lesen und Schreiben von Datensätzen, sowie Dienste für die Diagnose von Peripheriebaugruppen. Da das Profibusssystem multimasterfähig ist, existieren auch entsprechende Dienste für die Master-Master-Kommunikation. Auf diese soll in dieser Arbeit nicht näher eingegangen werden, da Untersuchungen der Master-Master-Kommunikation nicht Gegenstand der Forschungsaufgabenstellungen waren.

---

<sup>58</sup> CP343-1 IT bzw. CP443-1 IT

### **6.2.1 Auswirkungen der Globalisierung der Kommunikation auf die Industrie**

Die Globalisierung der Kommunikation durch das Internet auf Basis von TCP/IP hat in der zweiten Hälfte der 1990iger Jahre, wie kaum eine andere Technologie, starke Veränderungen in gesellschaftlichen Bereichen eingeleitet. Ab dem Jahr 2000 zeichnet sich bei den Herstellern von automatisierungstechnischen Systemen als auch durch die Erfordernisse der Anlagenbetreiber der Trend ab, die Internettechnologie auch im industriellen Bereich einsetzbar zu machen. Ebenso wie in den 1990iger Jahren die Dezentralisierung in der Steuerungstechnik veränderte wird die global möglich gewordene Vernetzung ganzer Anlagensysteme die Automatisierungstechnik verändern. Der zunehmende Einsatz von echtzeitfähiger Kommunikation über industrielle Ethernettechnologien ab Mitte der 2005er Jahre stellt dabei einen weiteren wichtigen Meilenstein dar, bewährte Kommunikationskonzepte aus dem Office-Bereich für die Automatisierungstechnik nutzbar zu machen.

Globale Kommunikation und industrielle Steuerungstechnik einige Zitate(Auszug):

*Das Internet hat in den letzten Jahren die Erwartungen an Applikationen drastisch verändert, so dass die angebotenen Dienste weit mehr bereitstellen müssen als die Möglichkeit des Surfens. Die vorhandene Infrastruktur des WWW wird in den nächsten Jahren die Grundlage für diverse interaktive Formen der Kommunikation sein [6.03].*

*Ethernet und Web scheinen eine Art Allheilmittel zur Bekämpfung allen Automatisierungsübels zu sein. Der eigentliche Nutzen der IT- und Web-Technologien wird sich vor allem mittel- und langfristig bemerkbar machen. Immer mehr abgeschlossene Projekte zeigen, dass verteilte Intelligenz mit Ethernet und Web-Technologien die Bewährungsprobe bestanden haben und reif für den Einsatz in der breiten Masse sind [6.02].*

*Nicht-PC-Geräte werden ab 2002 für fast die Hälfte aller Internetzugriffe verantwortlich sein[6.20].*

### **6.2.2 Problemstellungen von Internettechnologien in der Automatisierungstechnik**

Betrachtet man den historischen Hintergrund der Entwicklung der Internettechnologie sowie die Vielzahl von unterstützten Diensten und Protokollen bis hin zu der Web-Servicetechnologie, so lassen sich für die Nutzung in automatisierungstechnischen Anlagensystemen folgende wesentliche Vorteile und Nachteile gegenüberstellen:

### Wesentliche Vorteile

- Das Kommunikationsmedium steht global<sup>59</sup> zur Verfügung.
- Das Kommunikationsmedium steht mit reduzierter Dienstgüte auch nach einem Ausfall einzelner Knoten zur Verfügung.
- Die Browsernutzung kann in vielen Einsatzfällen teure Visualisierungssysteme ersetzen.
- Über Funknetze kann die gewünschte Funktionalität, unter Berücksichtigung reduzierter Sicherheit<sup>60</sup>, dem Anwender auch mobil zugänglich gemacht werden.
- Durch Webservices können verschiedenste, speziell angepasste Dienste für Automatisierungstechnische Systeme angeboten werden.
- Die Bandbreite ist in weiten Bereichen skalierbar (Kbit, Mbit, Gbit).

### Wesentliche Nachteile

- Die Kommunikationstechnologie muss als nicht echtzeitfähig eingestuft werden.
- Dynamische und interaktive Inhalte sind nur mit zusätzlichem Aufwand realisierbar.
- Eine garantierte Dienstgüte (QoS) ist nicht möglich

### Grundlegende Lösungsansätze

- Steigende Datenübertragungsraten entschärfen das Problem der fehlenden Echtzeitfähigkeit des Internets, lösen es aber nicht.
- Interaktive Bedienung und Beobachtung von technischen Prozessen in Web-Seiten sind mit Java, JavaScript, ActiveX und Flash realisierbar.

### 6.2.3 Projektspezifische Problemstellungen

Ausgangspunkt für den internetbasierten Zugriff auf dezentrale Peripheriekomponenten ist eine prototypische automatisierungstechnische Lösung der Firma Siemens. Die Kommunikationsprozessoren der Produktreihe CP343-1 IT und CP443-1 IT bilden das Interface des Automatisierungssystems zum Internet und kommunizieren über den Rückwandbus<sup>61</sup> mit einer SIMATIC S7-CPU. Bei der bestehenden Lösung des Zugriffes aus dem Internet auf die dezentrale Peripherie besteht der wesentliche Nachteil, dass keine direkte Kommunikation zwi-

---

<sup>59</sup> Die betriebswirtschaftliche „Größe“ Kosten soll an dieser Stelle ausgeklammert und nicht näher betrachtet werden.

<sup>60</sup> WLAN-Kommunikation kann leichter durch Unbefugte analysiert und manipuliert werden als drahtgebundene LAN-Kommunikation. Die Störsicherheit gegenüber Übertragungsfehlern ist gegenüber leitungsgebundener Kommunikation ebenfalls geringer.

<sup>61</sup> Firmeninternes lokales Bussystem der Siemens AG für die Kopplung von Baugruppen

schen Ethernet und dem PROFIBUS DP erfolgt und alle Zugriffe über die SIMATIC S7-CPU weitergereicht werden müssen.

In den genannten Kommunikationsprozessoren sind nach Herstellerangaben aus verschiedenen Gründen (vorrangig Netzwerksicherheit für die Automatisierungsanlage) Dienste und Portzugriffe nicht implementiert bzw. gesperrt. Zugelassen sind nur Zugriffe auf Port 80 (http) und Port 102. Damit soll auch ohne zusätzliche Firewall ein Automatisierungssystem bei der Integration in das Internet, vor ungewollten Zugriffen geschützt werden.

Durch die Zusammenarbeit mit der Hochschule für Technik und Wirtschaft Mittweida (FH) auf dem Gebiet der Schaffung eines „Internetbasierten Labors für industrielle Steuerungstechnik“ im Rahmen des Sächsischen internetbasierten Hochschulverbundes (Federführend: TU-Dresden, TU-Chemnitz, TU-Freiberg, Siemens AG) wurden dazu unabhängige Untersuchungen hinsichtlich der Netzwerksicherheit unternommen. Durch die Arbeiten der Mittweidaer Kollegen wurde nachgewiesen, dass über den CP343-1 IT eine direkte Manipulation an den Programm- und Datenspeicherbereichen in der SPS über das Internet möglich ist. Diese Ergebnisse sind unter anderem als ein wichtiger Bestandteil in die Arbeit „Beitrag zu offenen Automatisierungsnetzen am Beispiel eines Web-basierten Labors Automatisierungstechnik“ von Herrn Dr.-Ing. Schmeißer [3.09] eingeflossen. Während der laufenden Forschungsarbeiten auf dem Gebiet der Integration von ET200-Peripheriekomponenten in die genannten Datennetze, wurde damit die Notwendigkeit der Untersuchung neuer Zugriffskonzepte, deren Ergebnisse in diesem Kapitel vorgestellt werden, bestätigt.

Ausgehend von den theoretischen Betrachtungen zu echtzeitfähigen Systemen, zur Modellierung von Kommunikationssystemen sowie zu Architekturkonzepten von echtzeitfähigen Kommunikationssystemen und den öffentlich zugänglichen Informationen über das SIMATIC S7 Steuerungssystem zuzüglich der entsprechenden PROFIBUS-Spezifikationen werden im folgenden die Arbeitsschritte der Entwicklungsvorhaben bis zum Ergebnis beschrieben und evaluiert.

#### **6.2.4 Dezentrale Peripherie als wesentlicher Bestandteil verteilter Automatisierungssysteme**

Auf serieller Kommunikation basierende industrielle Steuerungssysteme mit zentralisierter Informationsverarbeitung (SPS bzw. IPC) können als Stand der Technik in der modernen Industrieautomation angesehen werden. Eine wesentliche Rolle spielt dabei der Einsatz von dezentralen Peripheriekomponenten, welche mit dem zentralen Steuerungssystem(n) kommunizieren und prozessnah die Sensorik und Aktorik bedienen. Die steigende Komplexität verteilter Automatisierungssysteme verändert das Anforderungsspektrum dieser Komponenten im Gesamtsystem. Moderne intelligente dezentrale Peripheriekomponenten bilden ein wesentliches Standbein heutiger und zukünftiger Industrieautomation.

## **6.2.5 Konzepte für den internetbasierten Zugriff auf die dezentrale Peripherie**

Ausgehend von den analysierten Problemen der derzeit realisierten Zugriffskonzepte auf dezentrale Peripheriekomponenten in verteilten Systemen der Automatisierungstechnik soll ein neues Konzept für den webbasierten Zugriff auf ET-200 Baugruppen erarbeitet werden. Dazu wird zuerst das bisherige Kommunikationskonzept für den Zugriff auf dezentrale Peripheriekomponenten vorgestellt und dessen Nachteile analysiert. Im Folgenden wird aus den aufgestellten Schlussfolgerungen ein neues erweitertes Konzept für den Zugriff auf die genannten Komponenten, unter den Bedingungen der Integration in das Internet/Intranet diskutiert und mit dem derzeitig verfügbaren Konzepten verglichen.

Im Folgenden werden die Begriffe „Zugriff auf dezentrale Peripheriekomponenten“ und „Diagnose dezentraler Peripheriekomponenten“ mit der gleichen Bedeutung versehen, entsprechende Nachweise der Funktionalität der Architekturkonzepte werden am Beispiel der Diagnose der Baugruppen realisiert.

### **6.2.5.1 Webbasierter Zugriff über die SPS und den IT- Kommunikationsprozessor**

Der Internet-Kommunikationsprozessor wird als Gerät, ebenso wie beispielsweise eine I/O-Baugruppe, über den Rückwandbus der SPS angeschlossen. Er tritt gegenüber dem Internet als Web-Server auf und arbeitet nach seiner Projektierung unabhängig von der SPS. Das heißt, auch wenn sich die SPS im Stop-Zustand befindet, können die Web-Seiten, welche auf dem Kommunikationsprozessor hinterlegt wurden, durch einen Browser genutzt werden.

Die Kommunikation zwischen beiden Geräten erfolgt in der Regel über STEP7-Datenbausteine. Ein direkter Zugriff auf Merker, Ein- und Ausgänge ist ebenfalls möglich. Befindet sich die SPS jedoch im Stop-Modus, ist eine Kommunikation zwischen Kommunikationsprozessor und dezentraler Peripherie nicht mehr möglich, da kein Anwenderprogramm abgearbeitet wird, welches das Handshaking<sup>62</sup> übernimmt. Das heißt, die Kommunikation, welche durch die SPS weitergeleitet wird, ist unterbrochen. Der Kommunikationsprozessor kann dem Anwender keine aktuellen Daten über das verteilte Automatisierungssystem zur Verfügung stellen. Damit entsteht das Problem, dass in einem solchen Fehlerfall eine Diagnose des Systems, insbesondere der Peripheriebaugruppen, nicht mehr möglich ist. In Abbildung 6.1 ist die Kommunikationsstruktur des bisherigen Konzeptes dargestellt. Auf der linken Seite ist die Bedien- und Beobachtungskomponente als Clientanwendung zu sehen. Auf der rechten Seite ist symbolisch ein verteiltes Automatisierungssystem, bestehend aus einer zentralen Steuerungskomponente mit dezentraler Peripherie, als Server dargestellt.

---

<sup>62</sup> auch als Quittungsbetrieb bezeichnet

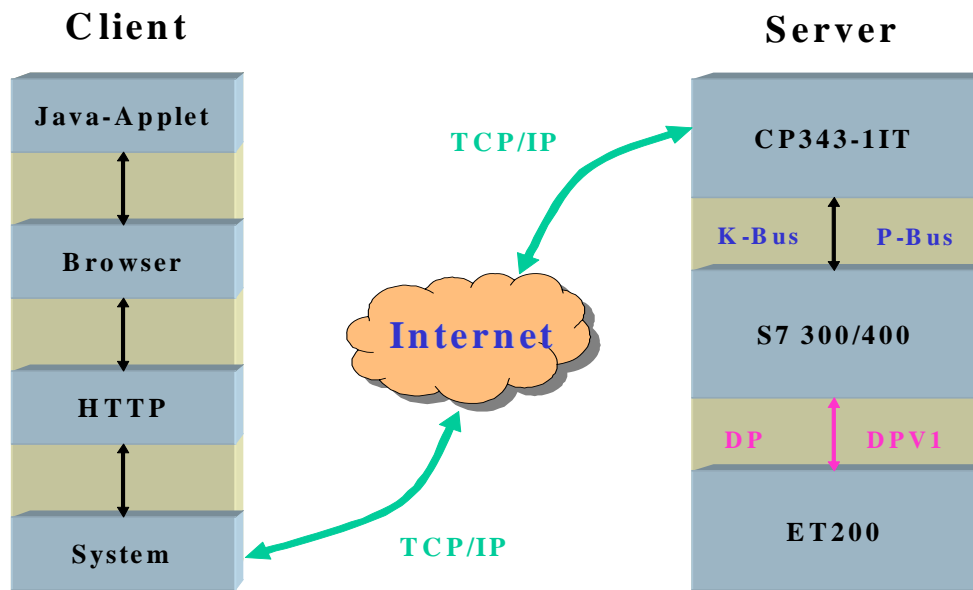


Abbildung 6.1: Webbasierter Zugriff über S7300 und CP343-1 IT<sup>63</sup>

Die bestehende Kommunikationsstruktur weist, für den Einsatzfall der Diagnose von ET200-Baugruppen, folgende grundlegende Schwachstellen auf:

- Die Diagnose während des Stop-Zustandes der SPS ist nicht möglich, da über die S7 300/400 Komponente geroutet werden muss.
- Die Kommunikation über den Rückwandbus wird zu einem Flaschenhals. Insbesondere der Rückwandbus der S7-300 kann mit den 12 Mbit/s des Profibussystems nur bedingt mithalten. Bei Verwendung noch höherer Baudraten, wie z.B. mit der LVDS-RS485-Technologie (Low Voltage Differential Signaling) oder mittels Lichtwellenleiterübertragung möglich, spitzt sich dieses Problem weiter zu.
- Es können mit den bestehenden Kommunikationsprozessoren nur SIMATIC-STEP7 Funktionen genutzt werden. Verschiedene Dienste, welche in den Profibusnormen festgelegt sind, können nicht oder nur eingeschränkt über Umwege genutzt werden. Dazu zählen insbesondere MSAC-C2-Dienste.

Die Erkenntnisse aus den genannten Schwachstellen sind ebenfalls bei der Konzipierung des Architekturmodells und der prototypischen Realisierung des Web-basierten Updates der Firmware von ET200 Komponenten von Bedeutung.

<sup>63</sup> Die Kommunikation zwischen Client und Server über das Internet ist schematisch dargestellt.

### 6.2.5.2 Webbasierter Zugriff durch Klasse-2-Master mit integriertem Web-Server

Wie kann nun in einem SIMATIC S7-Steuerungssystem eine Diagnose im möglichen Stop-Zustand der SPS erfolgen?

Die PROFIBUS-Norm, insbesondere ab der Erweiterung DPV1, unterscheidet Klasse-1-Master (SPS) sowie Klasse-2-Master (Programmiergeräte, Diagnosegeräte). Mit den gegenwärtigen Kommunikationsprozessoren wird über den P- bzw. K-Bus über die SPS auf den PROFIBUS zugegriffen.

Wenn ein Diagnosegerät einen direkten Zugriff auf den PROFIBUS bekommt, kann mit einzelnen dezentralen Peripheriebaugruppen des Gesamtsystems auch ohne laufendes SPS-Programm kommuniziert werden. Diese Möglichkeit besteht, da die Protokoll-ASIC's (SPC3 und ASPC2) nach ihrer Initialisierung das Profibusprotokoll autonom abwickeln. Das heißt, es wird unter anderem auch der wichtige Token im Stop-Zustand einer SPS an weitere Master, welche im System projiziert sind, weitergegeben.

Nach erfolgter theoretischer und praktischer Analyse des Profibussystems lassen sich die festgestellten grundlegenden Schwierigkeiten nur durch den direkten Durchgriff auf die dezentrale Peripherie umgehen. Vom Profibussystem werden dabei prinzipiell zwei Masterklassen unterstützt.

Auf Protokollebene können MSAC-C2-Dienste (Master-Slave-Acyclic-Communication, class 2), welche in den Normen festgelegt sind, genutzt werden. Es handelt sich dabei im Wesentlichen um die Dienste Datensatz lesen und schreiben sowie das Diagnosetelegramm laden. Im Gegensatz zu MSAC-C1-Diensten, welche Slaves nur vom Parametriermaster annehmen, müssen MSAC-C2-Dienste vom Slave grundsätzlich ausgeführt werden. Damit ist selbst in einem Doppelmastersystem nur ein internetfähiges Kommunikationsgerät erforderlich.

Der Typ Klasse-2-Master eignet sich aufgrund seiner Spezifikation aus folgenden Gründen für die Realisierung eines internetbasierten Diagnosesystems:

- Der Klasse-2-Master kann bei laufendem Betrieb in ein SIMATIC basiertes Automatisierungssystem integriert werden.
- Der Klasse-2-Master hat auch in Multimastersystemen den Zugriff auf alle dezentralen Peripheriekomponenten im verteilt aufgebauten Automatisierungssystem.
- Der Klasse-2-Master kann nach der Spezifikation (DPV0 und DPV1) nicht die zyklische (echtzeitfähige) Kommunikation eines Slaves mit dem Parametriermaster stören.

Schlussfolgernd soll an dieser Stelle der Begriff „Diagnosemaster“<sup>64</sup> eingeführt werden. Es handelt sich dabei um ein Gerätekonzept welches eine Kombination aus einem Klasse-2-Master und einem Web-Server darstellt und einen direkten Zugriff auf dezentrale Peripheriekomponenten in PROFIBUS-DP-Systemen gestattet.

---

<sup>64</sup> Unter dem Begriff Diagnosemaster soll ein Gerät verstanden werden, welches als Klasse-2-Master arbeitet.

Realisieren lassen sich solche Gerätekonzepte nach dem gegenwärtigen Stand der Technik auf zwei verschiedenen Wegen:

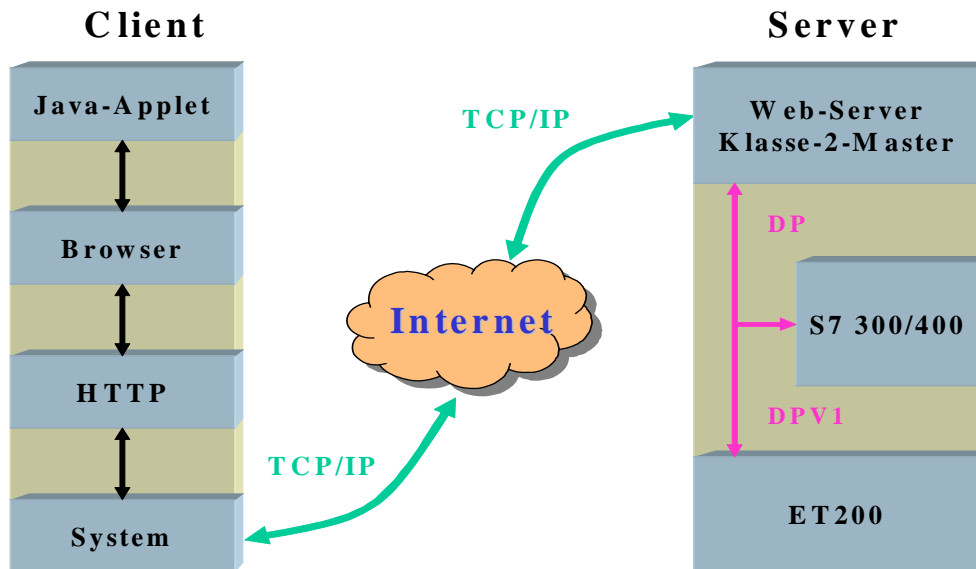


Abbildung 6.2: Webbasierter Zugriff durch direkte Kommunikation mit der dezentralen Peripherie

- Ein PC wird mit einem Kommunikationsprozessor (z.B. CP5611/CP5613 mit PCI-Interface) ausgerüstet. Auf diesem Rechner wird ein Web- oder FTP-Server aufgesetzt. Die Unterstützung der Funktionalität RPC (Remote Procedure Call) stellt für das eingesetzte Serversystem (z.B. Apache-Webserver) eine zwingende Voraussetzung dar. Die softwaremäßige Kopplung zwischen Web-Interface und PROFIBUS-DP-Interface wird über entsprechende Gerätetreiber realisiert und stellt damit eine eigenständige Windows-Applikation dar.
- Die Realisierung erfolgt als autonom arbeitendes Embedded-System. Die Funktionalität entspricht der wie im ersten Punkt beschrieben. Der wesentliche Unterschied liegt in der Nutzung eines Embedded-Web-Servers (z.B. SC12@CHIP) in Kombination mit einem PROFIBUS Master-ASIC (z.B. ASPC2). Als nachteilig erweist sich bei dieser Variante die Tatsache, dass die Gerätetreiber für den Master-ASIC individuell entwickelt werden müssen.



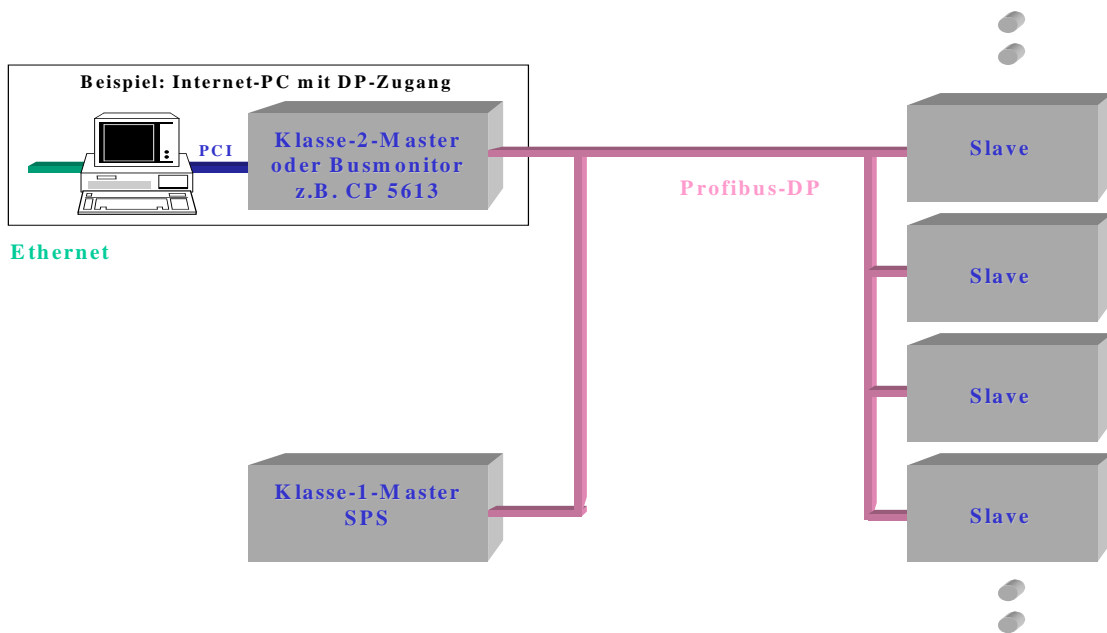


Abbildung 6.3: Klasse-2-Master basierte Zugriffsstruktur auf ET200-Komponenten

In der Abbildung 6.3 ist die prinzipielle Struktur eines SIMATIC basierten verteilten Automatisierungssystems, welches Client- und Serververhalten aufweist, dargestellt. Auf das System kann, wie in Abbildung 6.2 vorgestellt, über das Web-Interface zugegriffen werden.

### 6.2.5.3 Bewertung beider webbasierter Zugriffskonzepte

Die Analyse der Kommunikationsmechanismen auf dem Feldbussystem PROFIBUS führt zu dem Ergebnis, dass genau zwei Zugriffsverfahren für den Fernzugriff auf dezentrale Peripheriekomponenten im SIMATIC S7 System zur Verfügung stehen. Es kann dabei nachgewiesen werden, dass Internet-basierte Zugriffe über einen speziell ausgerüsteten Klasse-2-Master möglich sind und eine sinnvolle Alternative zum bisherigen Konzept darstellt.

## 6.3 Webservices in verteilten Automatisierungssystemen

### 6.3.1 Grundlegende Konzepte und Definitionen von Web-Services

Web-Service Definition:

*Ein Webservice ist eine über ein Netzwerk zugängliche Schnittstelle zu Anwendungsfunktionen, die mit Hilfe von Standardtechniken des Internet realisiert wird [6.05].*

Durch die Nutzung von Standardprotokollen wie TCP/IP können Web-Services sehr universell eingesetzt werden. Die Webservice-Technologie ist dennoch nicht zwingend notwendig an ein bestimmtes Protokoll gebunden. Folgende wesentliche Entwurfs- und Programmierme-

thoden werden zum jetzigen Zeitpunkt für Internet-basierte Zugriffe auf entfernte Kommunikationssysteme eingesetzt.

- SOAP:** Das Simple Object Access Protocol dient als Nachrichten- und RPC-Austauschprotokoll von Applikationen und Softwarekomponenten. SOAP basiert auf der XML-Spezifikation [6.23].
- WSDL:** Web Services Description Language stellt eine Beschreibung dar, wie eine hergestellte Verbindung mit einem Web-Service kommunizieren soll. WSDL ist XML-basiert und dient zur Identifikation und zum Auffinden des Services [6.24].
- UDDI:** Universal Description, Discovery, and Integration stellt ein webbasiertes Informationssystem für Web-Services dar. Es bietet ein Verzeichnis von Adress- und Produktdaten sowie von Anwenderschnittstellen.
- XML:** Die eXtensible Markup Language (XML) wird aufgrund ihrer Flexibilität und Erweiterbarkeit in der Web-Service-Technologie angewendet.

### **6.3.2 Vorstellung eines Konzeptes für den Einsatz von Web-Services in echtzeitfähigen verteilten Steuerungssystemen**

Der Stand der Technik ist durch komplexe Systeme der Automatisierungstechnik gekennzeichnet, deren Komponenten über spezielle Kommunikationssysteme (z.B. Aktor-Sensor-Busse, Feldbusse, Industrial Ethernet) vernetzt sind.

Mit dem Internet existiert parallel dazu ein weltweit verfügbares und standardisiertes Kommunikationssystem, über das zahlreiche Dienste (im folgenden Webservices genannt) bereitgestellt werden können. Ziel des vorgestellten Konzeptes ist es, Webservices für Systeme der Automatisierungstechnik auf eine flexible Art und Weise verfügbar zu machen, wobei diese Systeme selbst nicht zwingend eine Internetarchitektur mit den entsprechenden Mechanismen und Protokollen aufweisen müssen.

Die Nutzung von Webservices beruht zum gegenwärtigen Zeitpunkt üblicherweise auf der manuellen Anfrage bei einem bekannten Dienstleister (Server, Provider), der nach dem gewünschten Service abgefragt wird. Anschließend wird über die Inanspruchnahme des Services entschieden und dieser gegebenenfalls realisiert. Dabei ist aus Sicht des Automatisierungssystems das Vorgehen nicht von einer Vor-Ort-Serviceleistung (ohne Webservice) zu unterscheiden.

Es muss vorausgesetzt werden, dass eine Architektur nach Abbildung 6.4 und Abbildung 6.5 existiert, die dadurch gekennzeichnet ist, dass es im Internet eine zentrale Registratur („Gelbe Seiten“) gibt, die Angaben über verfügbare Webservices enthält. Dafür existieren entsprechende Lösungsvorschläge. Ein Dienstanutzer muss dabei diese Registratur nicht zwangsweise kennen. Über eine Anfrage wird er zu dieser Registratur geführt und erhält Auskünfte, welche Services für sein Automatisierungssystem verfügbar sind. Ein Dienstanutzer kann dabei sowohl eine Person oder ein technisches System sein.

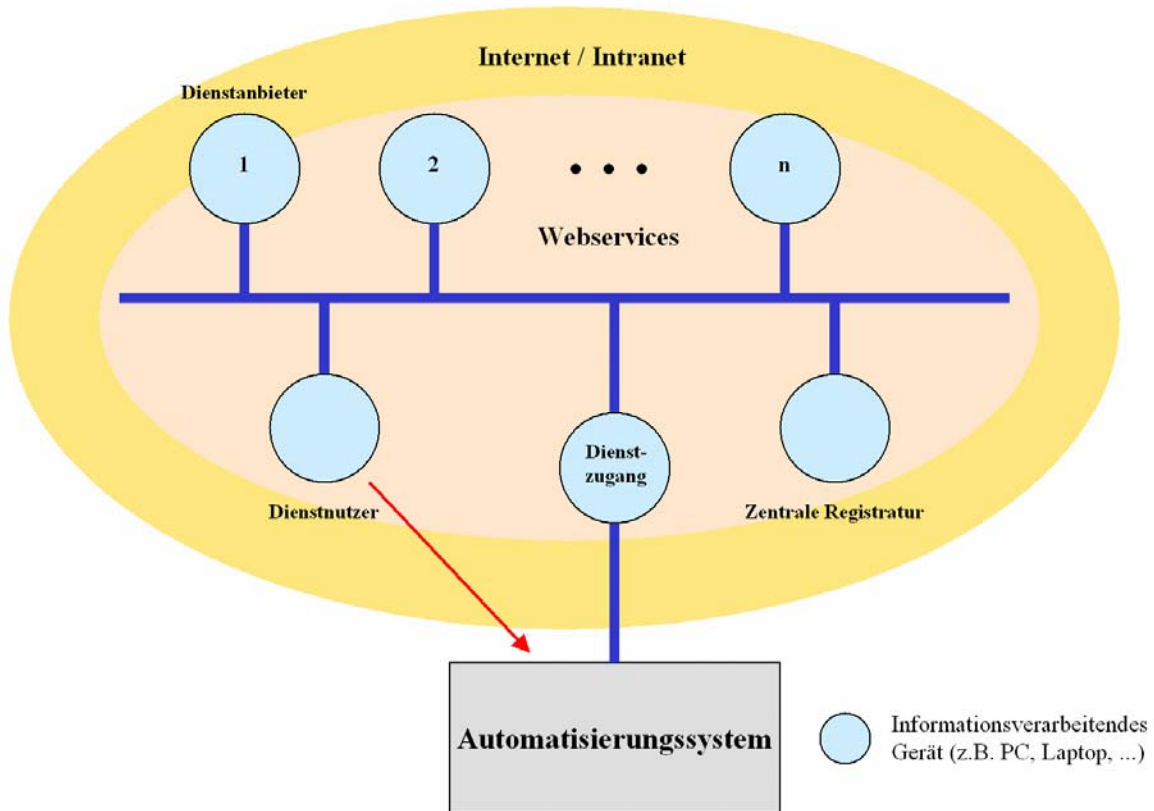


Abbildung 6.4: Webservicearchitektur für verteilte Automatisierungssysteme [6.17]

Ein Nutzer (Client  $i, j$ ) an einem beliebigen Standort mit TCP/IP fähigen Kommunikationsanschluss, dabei kann es sich um eine Person an einem Computer oder auch um eine autonom agierende Maschine handeln, möchte einen Webservice (Firmwareupdate) nutzen und kennt entweder die Adresse einer zentralen Registratur (Server  $n, m$ ) oder beschafft sich diese über die Nutzung von Suchmaschinen. Daraufhin verbindet sich der Client mit einem der ermittelten Registratur-Server und fragt nach einem Webservice mit der gewünschten Funktionalität (Firmwareupdateservice) an. Als Antwort erhält er von der zentralen Registratur eine Übersicht möglicher Dienstanbieter sowie deren Adressen und eine Kurzbeschreibung der angebotenen Dienstleistungen.

Er wählt entsprechend seiner eigenen Anforderungen einen Dienstanbieter aus und verbindet sich mit diesem Server (Server  $l$ ). Mit dem Dienstanbieter werden die Details der Dienstausführung (z.B. die Adresse der für die Wartung vorgesehenen Baugruppe oder weitere Details zum Update der Firmware) ausgehandelt.

Der Dienstanbieter wiederum besitzt unter Umständen noch nicht alle Daten (z.B. Firmwareupdatedateien), die für die Ausführung der angebotenen Dienstleistung notwendig sind und muss sich diese bei einem anderen Webservice besorgen. Dazu geht er analog vor: Er sucht jetzt in Funktion eines autonomen Clients nach einem Verzeichnis (Server  $m, n$ ). Dort fragt er nach einem geeigneten Dienst (Firmwarehalter) an und erhält eine Bestätigung. Er wählt einen Dienstanbieter (Server  $o, p, q$ ) aus, verbindet sich mit diesem und vereinbart alle nötigen

Details (Daten zur benötigten Firmware). Nachdem er (Client = Dienstanbieter des Webservices „Firmwareupdate“) die notwendigen Daten für die Ausführung des Webservices zur Verfügung hat, startet er den angebotenen Dienst und informiert den Dienstanbieter über seine Fortschritte.

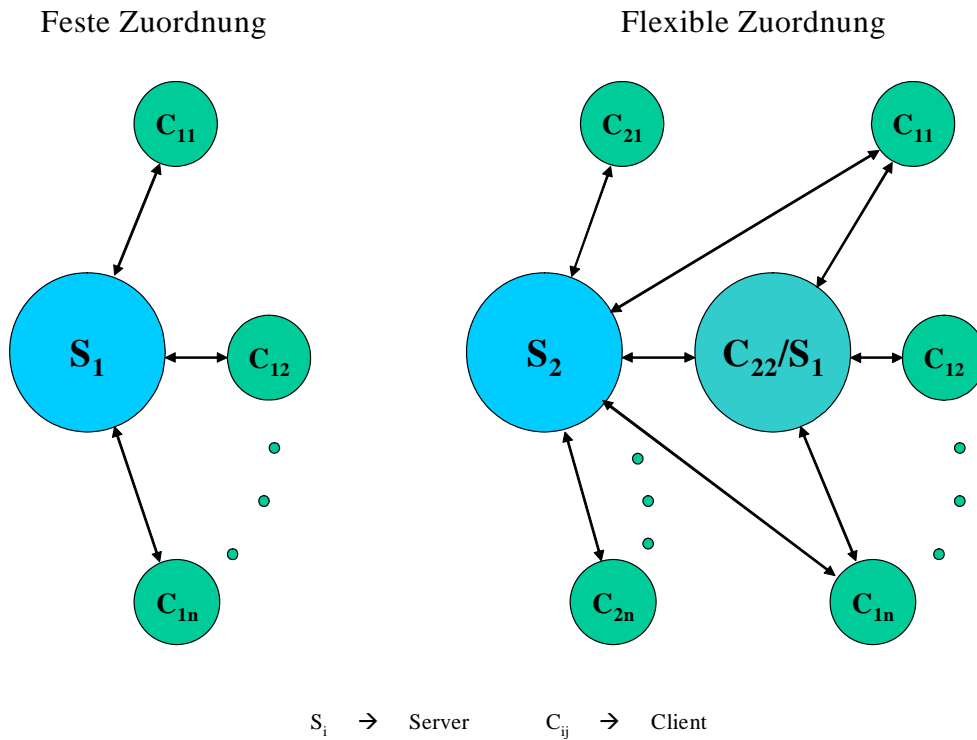


Abbildung 6.5: Dynamisierung der Client-Server-Architektur für Automatisierungssysteme

Bei der prototypischen Realisierung des Webservices „Firmwareupdate“ werden die neuen Firmwaredaten im laufenden echtzeitfähigen Betrieb des Versuchsaufbaues übertragen. Die Tests erfolgen unter den Bedingungen der Betriebsart rechtzeitige Kommunikation im PROFIBUS-System. Das vorgestellte Konzept ist ohne weiteres auch in der äquidistanten Betriebsart des PROFIBUS einsetzbar.

Der Webservices „Firmwareupdate“ hat im wesentlichen die Aufgabe ein neues Betriebssystem auf einer Peripheriekomponente zu installieren. In der Zeitspanne von der Anforderung zum Neustart bis zur vollständigen Betriebsbereitschaft der Baugruppe gehen selbstverständlich Informationen über den zu steuernden Prozess verloren. Untersuchungen zum Echtzeitverhalten in dieser Wideranlaufphase sind nicht Bestandteil der vorliegenden Arbeit. Für entsprechende Analysen und Betrachtungen zu diesem Teil der echtzeitfähigen Kommunikationstechnik erfordern in jedem Fall die detaillierte Kenntnis über den Hard- und Softwaretechnischen Aufbau der einzelnen Komponenten.

### 6.3.3 Prinzipieller Ablauf des Web-Services Firmwareupdate

In den vorhergehenden Abschnitten wurden die wesentlichsten Anforderungen an die Architekturen und softwaretechnischen Methoden für die Realisierung von Webservices im industriellen Steuerungsbereich vorgestellt. Dazu wurden im Vorfeld entsprechende Architekturuntersuchungen durchgeführt sowie das Kommunikationsverhalten im Simatic-S7 System analysiert und bewertet. In der Abbildung 6.6 ist der sequentielle Ablauf der Kommunikationsbeziehungen zwischen den einzelnen Komponenten dargestellt, welche erforderlich sind, derartige webfähige Dienste für industrielle Kommunikationssysteme anzubieten.

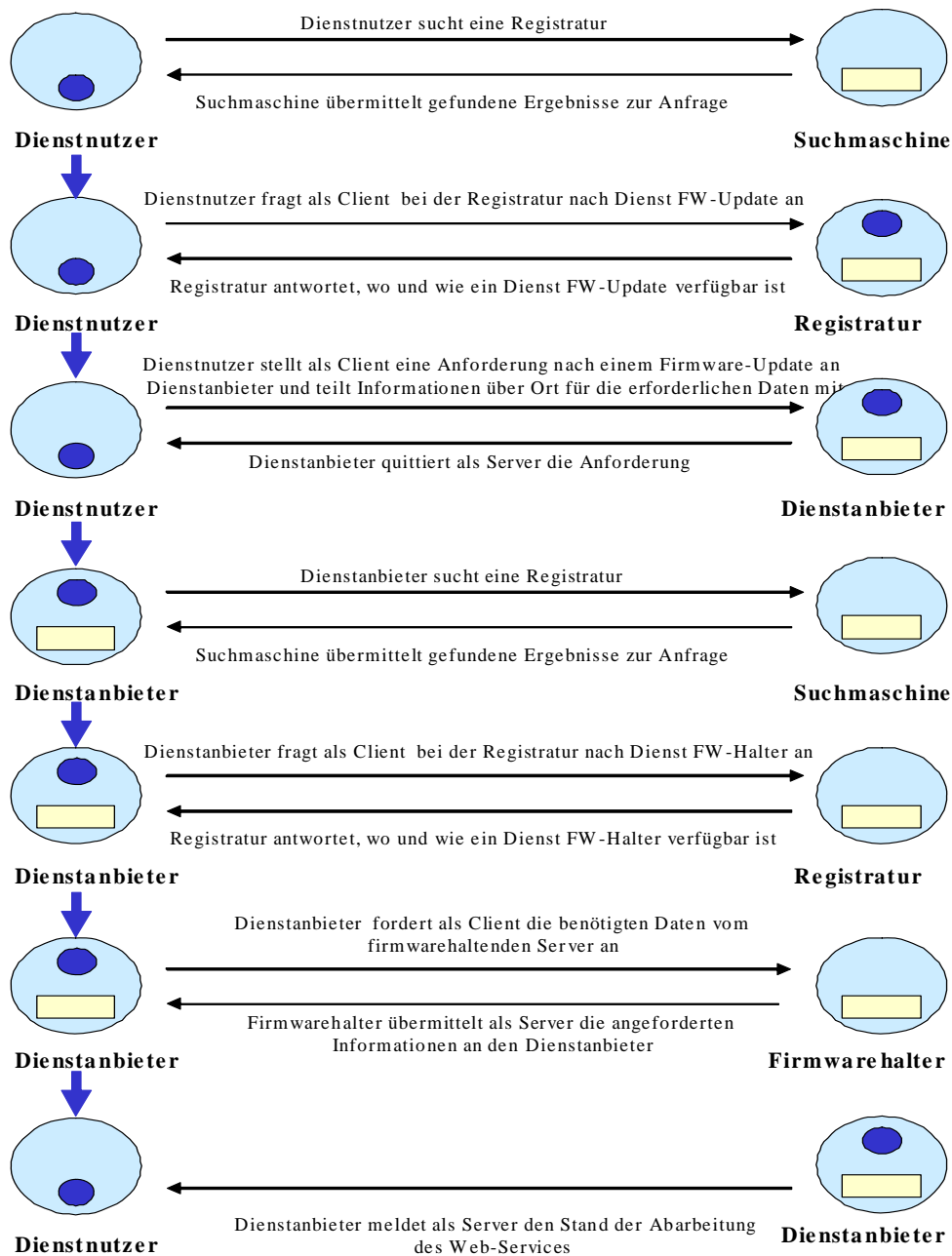


Abbildung 6.6: Schematischer Ablauf eines internetbasierten Firmwareupdates [6.17]

Legende:



Komponente mit reiner Clientfunktionalität



Komponente mit reiner Serverfunktionalität



Komponente mit kombinierter Client- und Serverfunktionalität

Wie man bei näherer Betrachtung sehen kann, erfolgt die Kommunikation zwischen Komponenten zu einem Teil mit streng getrennter Client- oder Serverfunktionalität. Wesentliches neues Merkmal ist jedoch die Komponente, welche beide Funktionalitäten ausfüllen kann, in diesem Fall das industrielle Steuerungssystem. Ein entscheidendes Merkmal ist die Tatsache, dass in der Abbildung 6.6 explizit keine SIMATIC S7 Komponenten dargestellt sind. Die Bereitstellung webbasierter Dienste dieser Art ist in keiner Weise dabei auf dieses spezielle industrielle Steuerungssystem beschränkt, obwohl die positive prototypische Realisierung eines Dienstes explizit mit diesem System durchgeführt wurde. Mit entsprechenden Analysen und Bewertungen der Funktionsweise industrieller Steuerungssysteme anderer Hersteller<sup>65</sup> können solche Dienste ebenfalls bereitgestellt werden. Die primäre Voraussetzung bildet in jedem Fall ein internetfähiger TCP/IP/UDP-basierter Zugangspunkt auf das industrielle Kommunikationssystem.

#### 6.4 Versuchsanlagen zur Analyse, Realisierung und Evaluierung

Um die analysierten Problemstellungen bei webbasierten Zugriffen auf dezentrale Peripheriekomponenten insbesondere der ET200 Komponenten zu synthetisieren, verifizieren und bewerten zu können, sind neben ausführlichen Literaturrecherchen Versuchsausbauten mit den entsprechenden Automatisierungskomponenten erforderlich.

Im Rahmen der Forschungs Kooperation ist die bestehende Versuchsanlage des Instituts für Automatisierungstechnik mit den erforderlichen zusätzlichen Komponenten ausgerüstet worden. Die Konzeption der Versuchsanlage ist dabei so ausgelegt, dass einerseits die Forschungsschwerpunkte der Themen (siehe Kapitel 6.1) bearbeitet werden konnten, als auch im Rahmen des „Sächsischen internetbasierten Hochschulverbundes“ Arbeiten auf dem Gebiet der web-basierten Ausbildung an verteilten Steuerungssystemen realisiert werden konnten sowie zukünftig können.

Das mit industriellen Standardkomponenten realisierte Kommunikationssystem umfasst dabei den in Abbildung 6.7 dargestellten strukturellen Aufbau.

---

<sup>65</sup> Bezüglich Zugriffsprinzipien, Kommunikationsstrukturen, Echtzeitverhalten usw.

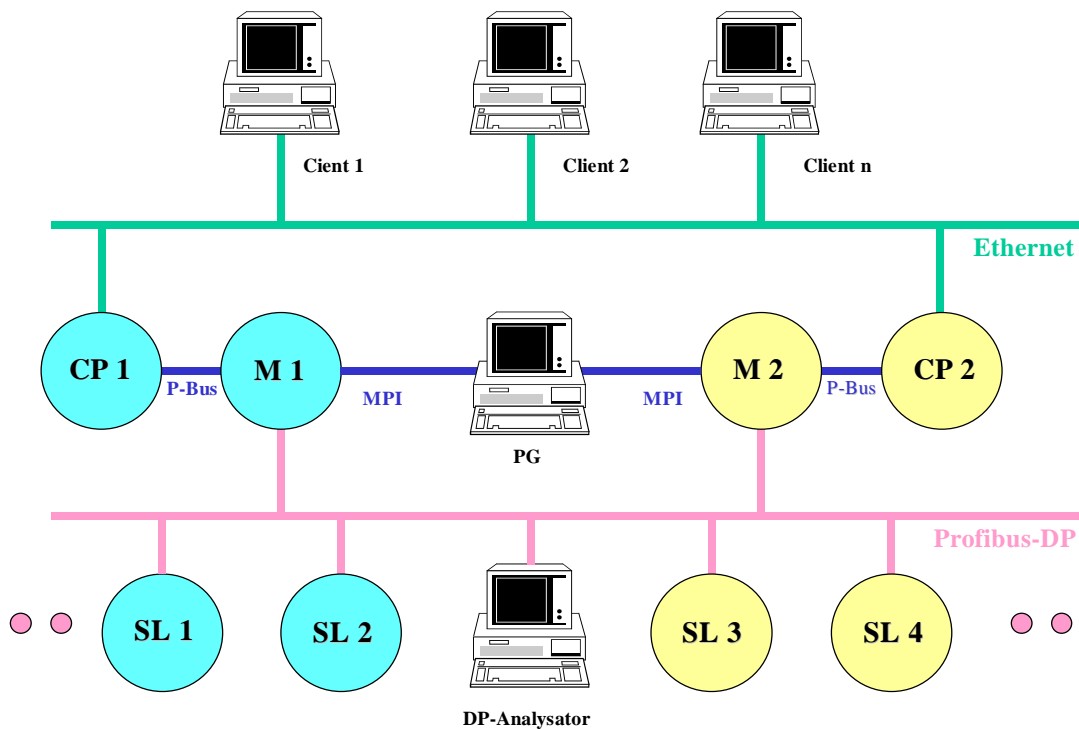


Abbildung 6.7: Struktur der Versuchsanlage des Instituts für Analysezwecke

Die Versuchsanlage ist durch folgende wesentliche Merkmale charakterisiert:

- S7-Dualmastersystem
- Separater Internetzugang beider Steuerungen über TCP/UDP/IP
- Profibusanalysetool „Amprolyzer“ [6.07] zur Zeitverhaltensanalyse

Für die prototypische Realisierung als auch zur Evaluierung der Modellansätze (Verteidigung der Ergebnisse nach dem Zeitplan der Meilensteine sowie der Veröffentlichung der Ergebnisse auf Messen und wissenschaftlichen Konferenzen) für den web-basierten Zugriff auf ET200-Komponenten ist ein separater Versuchsaufbau erforderlich. Diese Versuchsanordnung für die prototypische Realisierung des Web-Services „Firmwareupdate für eine ET200S“ ist komplexer ausgelegt und wird durch spezielle Prototypenbaugruppen ergänzt. Der strukturelle Aufbau ist in Abbildung 6.8 dargestellt und erfüllt alle Anforderungen für die Realisierung der Aufgabenstellung und entspricht dem Architekturkonzept, welches aufgrund der gemachten Analysen zur Aufgabenstellung favorisiert wurde. Die Struktur des Versuchsaufbaus zeigt sehr gut den laufenden Entwicklungstrend in der zukünftigen industriellen Kommunikation, welcher auf der Erhöhung der Transparenz in der klassischen Automatisierungspyramide beruht. Er repräsentiert modellhaft ein verteiltes industrielles Steuerungssystem, welches aus einer Kombination echtzeitfähiger Kommunikation in der Steuerungsebene und der nichtechtzeitfähigen Kommunikation in der MES- und ERP-Ebene besteht.

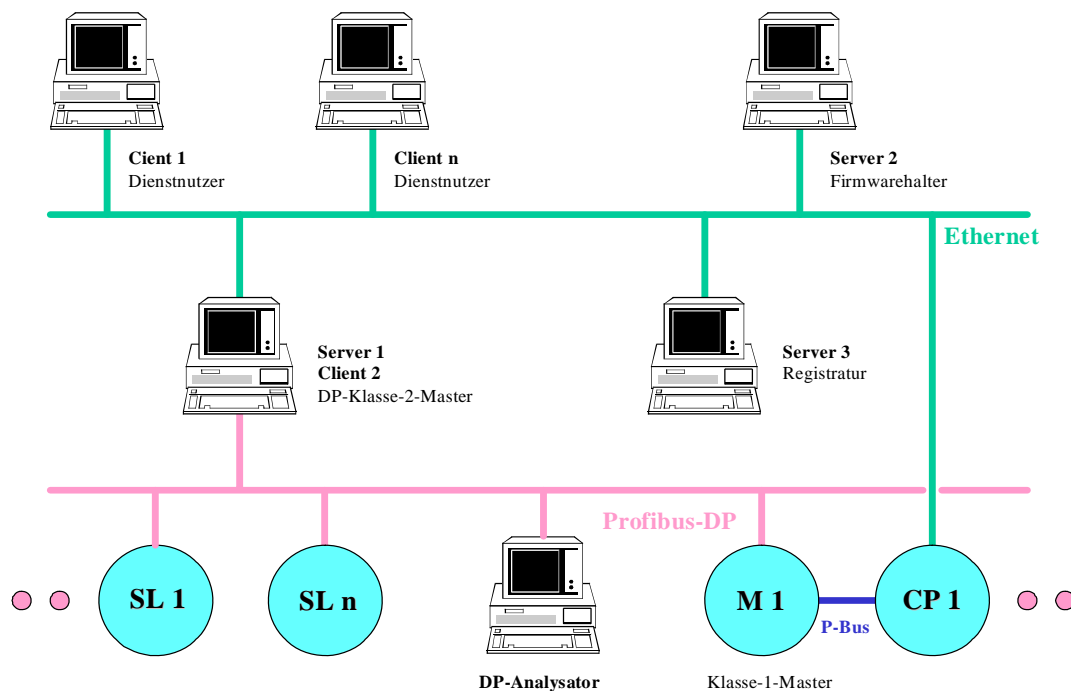


Abbildung 6.8: Versuchsanlage für die web-basierte Zugriffsfunktionalität auf ET200-Komponenten

Das verteilte System besteht dabei im Wesentlichen aus den Komponenten:

- Steuerungssystem S7/315 2DP mit Internetzugang über CP343-1IT
- Dezentrale Peripheriebaugruppen ET200
- ET200S (Prototypenbaugruppe der Entwicklungsabteilung RD52 A&D Siemens)
- Rechner für die Nutzung des angebotenen Dienstes
- Rechner mit Apache-Web-Server und Datenbank für Firmwarehaltung und Registratur
- Rechner mit Apache-Web-Server und DP-Klasse-2-Mastersystem

Optional erforderlich:

- Profibusanalysetool „Amprolyzer“ [6.07] zur Zeitverhaltensanalyse
- Rechner mit PCS7-Projektierungssystem für Klasse-1-Master



## 6.5 Leistungsbewertung des prototypisch realisierten Konzepts

Das vorgestellte Konzept des web-basierten Zugriffs auf dezentrale Peripheriekomponenten weist gegenüber eines vor Ort Zugriffs (im Normalfall durch das Softwaresystem PCS 7) prinzipbedingt schlechtere zeitliche Leistungsparameter auf. Das prototypisch realisierte Zugriffskonzept ist dabei durchschnittlich um den Faktor 10 langsamer (siehe Anhang).

Prinzipiell wirken sich auf das zeitliche Verhalten eines webbasierten Zugriffs auf dezentrale Peripheriekomponenten in einem Profibusssystem folgende zwei wesentliche Faktoren aus:

### 1. Nicht echtzeitfähiger web-basierter Zugriff auf das Automatisierungssystem

- Die effektive Datenübertragungsrate zwischen Web-Client und Web-Server
- Die Durchlaufzeiten in den beteiligten Kommunikationsknoten
- Die Menge der Telegrammverluste<sup>66</sup> bei der Kommunikation zwischen den Kommunikationsknoten

### 2. Echtzeitfähiger Zugriff auf die dezentrale Peripherie

- Die aktuelle Datenübertragungsrate auf dem Feldbussystem
- Die maximal zulässige Belegungszeit für azyklische Dienste unter Einhaltung der durch den zu steuernden Prozess geforderten oberen Zeitschranke. Arbeitet das Steuerungssystem mit konstanten Telegrammlängen, nach der DP-Norm maximal 246 Nutzdatenbytes, lässt sich die maximal mögliche Protokolleffizienz nutzen.
- Arbeitet das verteilte Steuerungssystem taktsynchron, entsteht das Problem, dass möglicherweise der zugewiesene Zeitslot für die azyklischen Dienste nicht ausreicht, um Telegramme mit maximaler Nutzdatenlänge pro Taktzyklus zu übertragen. Für diesen Fall kann eine Reduzierung der Nutzdatenbytes für ein azyklisches Datentelegramm, nach den Spezifikationen für PROFIBUS-DP möglich, vorgenommen werden. Mit der Schwankung der Anzahl der Nutzdatenbytes verändert dabei sich die Protokolleffizienz. Diese Schwankungsbreite muss bei der Zeitverhaltensanalyse in solchen Systemen je nach Art der Implementierung zusätzlich berücksichtigt werden.

---

<sup>66</sup> Beim Durchlaufen eines Ethernettelegramms durch einen Router wird ein Zähler im Header dekrementiert. Beim Zählerstand 0 wird das Telegramm verworfen (vom Router nicht mehr weitergeleitet). Erst Protokolle wie TCP/IP gewährleisten eine, aus Sicht der Übertragungstechnik, sichere Verbindung zwischen den Kommunikationsknoten.

## 6.6 Zusammenfassung

Die weltweite Vernetzung von Computern durch das gegenwärtig größte bekannte verteilte System, dem Internet, hat bereits viele Bereiche des gesellschaftlichen Lebens verändert. Die Tatsache dass das Internet, dessen Grundlage Anfang der siebziger Jahre durch das ARPANET geschaffen wurde und erst mit der wesentlich späteren Einführung des WWW solche gravierenden Veränderungen hervorgerufen hat, bestätigt einen hochdynamischen Entwicklungsprozess auf diesem Fachgebiet. Die ständige Weiterentwicklung dieses verteilten Systems hinsichtlich der Zuverlässigkeit und Geschwindigkeit der Kommunikation sowie seine Erweiterung um neue Dienstangebote, wie Web-Services führen dazu, dass es kaum einen Hersteller von Automatisierungskomponenten gibt, in dessen Produktspektrum sich nicht entsprechende Komponenten zur Nutzung dieser neuen Technologie befinden.

Es wurde in dieser Arbeit mit der prototypischen Realisierung des vorgestellten webbasierten Zugriffskonzepts nachgewiesen, dass Zugriffe in echtzeitfähige verteilte SIMATIC basierte Steuerungssysteme realisiert werden können. Dabei muss grundsätzlich die Trennung von echtzeitfähiger Steuerungstechnik und web-basierten Zugriff gewährleistet werden. Durch die Realisierung wurde nachgewiesen, dass in jedem Fall zum gegenwärtigen Zeitpunkt als auch zukünftig eine präzise zeitliche Synchronisation zwischen der Kommunikation im Internet und der Kommunikation im industriellen Steuerungssystem durchgeführt werden muss.

Die in Kapitel 3 für den Einsatz in der industriellen Kommunikation vorgestellten Ethernet-technologien entschärfen dabei eine Reihe von Problemstellungen, wie z.B. durch die Reduzierung der eingesetzten Telegrammformate. Auch wenn Ethernet in naher Zukunft durch entsprechende Maßnahmen<sup>67</sup> echtzeitfähig realisiert werden kann, besteht auch bei dieser modernen Kommunikationstechnologie das Problem den nichtechtzeitfähigen Teil des verteilten Systems Internet mit der Teilmenge des echtzeitfähigen verteilten Systems zu synchronisieren.

Neben den positiven Ergebnissen (siehe Abschnitt 14) aus den Projekten für den Web-basierten Zugriff auf dezentrale Peripheriekomponenten sollen aber auch negative Fakten aufgeführt werden.

- Es kann trotz des Einsatzes eines zusätzlichen Web-Servers als Klasse-2-Master nicht ausgeschlossen werden, dass ungewollte bzw. bewusst vorsätzliche lesende Zugriffe im Datenverkehr zwischen Klasse-1-Master und DP-Slaves vorgenommen werden können.
- Mit dem Klasse-2-Master als Zugangspunkt ins DP-System kann nicht, unter der Bedingung dass in der Applikationssoftware nur Dienste für den Zugriff auf dezentrale Peripherie – Komponenten implementiert sind, auf den Klasse-1-Master zugegriffen werden.

---

<sup>67</sup> Methode der Lastbegrenzung, Zeitschlitz- und Zeitsynchronisationsverfahren

- Mit der Einführung des S7-Online Zugangspunktes<sup>68</sup> über PROFIBUS-DP entsteht aber auch bei diesem vorgestellten Zugriffskonzept das Problem, dass im DP-System Dienste existieren, welche die Kommunikation zwischen den beiden Masterklassen erlauben. Der entscheidende Vorteil des Zugriffskonzepts, nur auf dezentrale Peripheriebaugruppen zugreifen zu können, wird dadurch leider aufgeweicht.

Eine detaillierte Betrachtung sicherheitstechnischer Aspekte soll an dieser Stelle nicht weiter durchgeführt werden, da dies nicht Schwerpunkt der Aufgabenstellungen war. Sicherheitstechnische Aspekte spielen jedoch eine immer bedeutendere Rolle bei der Integration von industrieller Steuerungstechnik in das Internet. Durch Maßnahmen wie genau ein definierter Zugangspunkt über einen Firewall lässt sich eine skalierbare Sicherheit gewährleisten. Bei der Kommunikation über IWLAN ist der sicherheitstechnische Aspekt von grundlegender Bedeutung, da solche industrielle Kommunikationssysteme theoretisch unbegrenzt viele Zugangspunkte bieten.

Während der Bearbeitung der Forschungs- und Entwicklungsthemen wurde darauf geachtet, dass erreichte Zwischenstände evaluiert und verifiziert werden. Dieser Prozess wurde durch die gestellten Meilensteine, bei denen die Ergebnisse den verantwortlichen Kollegen des beteiligten Industriepartners vorgestellt und diskutiert wurden, wesentlich unterstützt. Außerdem wurden ausgewählte Teilergebnisse der Forschungsk Kooperation auf dem Gebiet des webbasierten Zugriffes dezentraler Peripheriekomponenten in SIMATIC basierten Systemen auf wissenschaftlichen Kolloquien und Fachmessen vorgestellt und somit eine kritische Bewertung durch Fachkollegen gewährleistet.

Einzelne detaillierte Schritte der Bearbeitung der Themengebiete können den Forschungsberichten sowie den Protokollen und Sachstandsberichten zu den Projektverteidigungen entnommen werden.

---

<sup>68</sup> Schnittstelle für die Programmierung einer SPS über den PROFIBUS

## 7 Abschließende Betrachtungen und weiterführende Arbeiten

Die zentrale Zielstellung der vorgelegten Arbeit besteht in der Bereitstellung eines offenen Beitrages zur Verhaltensanalyse, Bewertung und Modellierung verteilter echtzeitfähiger Kommunikationssysteme für die Automatisierungstechnik. Die Bandbreite umfasst dabei räumlich sehr lokal abgeschlossene Systeme<sup>69</sup> mit sehr hohen Anforderungen an die Echtzeitfähigkeit (insbesondere der Gleichzeitigkeit der Kommunikation) bis hin zum größten gegenwärtig bekannten verteilten System, dem Internet bei dem die Frage der Echtzeitfähigkeit prinzipbedingt eine untergeordnete Rolle spielt. Im Zeitraum der Bearbeitung des Themengebietes kristallisierten sich dabei folgende drei wesentliche Forschungs- und Entwicklungsschwerpunkte heraus, welche nachstehend kurz zusammengefasst werden:

Die Konzeption, Modellbildung, Bewertung sowie Realisierung eines Steuerungskonzepts für die verteilte Synchronisation von dynamischen Prozessen am Beispiel einer Antriebssynchronisation des Operationstisches Jupiter auf dem Gebiet des medizinischen Gerätebaus. Die Funktionalität des vorgestellten und realisierten Steuerungskonzeptes für die zentrale Säulensteuerung und deren Integration in die Gesamtkonzeption des OP-Tisches Jupiter ist dabei seit Anfang 1999 durch den praktischen Industrieinsatz nachgewiesen.

Die Konzeption, Verhaltensanalyse sowie Leistungsbewertung eines internetbasierten Zugriffskonzepts für dezentrale Peripheriekomponenten der Baureihe ET200M aus der SIMATIC-S7 Steuerungsfamilie.

Dazu wurden die Forschungsprojekte „Studie zum Thema: Offene verteilte Automation“, „Internetbasierte Bedien- und Beobachtungskonzepte für ET200-basierte Systeme“ und „Internetmechanismen für ET200-basierte Systeme“ in Zusammenarbeit mit der Siemens AG Standort Chemnitz bearbeitet.

Ein wesentliches Ergebnis aus den Forschungsarbeiten stellt dabei die Patentanmeldung „Verfahren zur Nutzung von Web-Services-Technologien in Systemen der Automatisierungstechnik“ dar.

Die prototypische Realisierung zeigt, dass es möglich und sinnvoll ist, solche webbasierten Dienste<sup>70</sup> unter Nutzung von Standardkomponenten der industriellen Steuerungstechnik über ein Netzwerk anzubieten. Geschwindigkeit und Bedienkomfort liegen bei der prototypischen Realisierung jedoch unter denen der aktuellen Lösung von STEP7 mit Bedienung vor Ort. Eine Angleichung der Leistungsmerkmale der bestehenden STEP7-Lösung mit der webbasierten Lösung ist durch eine verbesserte softwaretechnische Umsetzung jederzeit möglich und würde eine mögliche produkttechnische Umsetzung des vorgestellten Architekturkonzepts darstellen.

---

<sup>69</sup> können auch als MIKRO-LAN bezeichnet werden

<sup>70</sup> Im Rahmen der Forschungsk Kooperation mit dem Industriepartner wurde der praktische Nachweis der neuen Funktionalität auf den Dienst „Firmware Update“ für ET200-Komponenten beschränkt.

Das vorgestellte und prototypisch realisierte Architekturkonzept stellt einen Beitrag zur IT-Integration von ET200-basierten Steuerungssystemen dar. Das neuartige Konzept wird und kann dabei die bestehenden meist lokal ausgelegten Verfahren des Zugriffs auf dezentrale Peripheriekomponenten nicht ersetzen, Es soll aber eine sinnvolle Ergänzung z.B. für die Ferndiagnose oder das Ferndebugging ET200 - basierter Komponenten darstellen.

Die klassische Form der Internetnutzung erfolgt von Anwendern durch einen Browser auf einem Standard-PC. Zum gegenwärtigen Zeitpunkt kann der Nutzer auch über PDAs, Laptops oder Mobiltelefonen, also auch mobil, auf das Internet zugreifen. Aus den Ergebnissen der Forschungs Kooperation mit der Siemens AG auf dem Gebiet der Internetmechanismen für dezentrale Automatisierungskomponenten kann folgende allgemeingültige Schlussfolgerung getroffen werden:

Die Nutzung des Internets bleibt nicht nur „natürlichen Personen“<sup>71</sup> über entsprechende Zugriffsmöglichkeiten vorbehalten. Zukünftige intelligente verteilte Automatisierungssysteme werden autonom im Internet Dienste als Server anbieten und Dienste als Client nutzen können.

Aus den Untersuchungen zu den verschiedenen Ansätzen der Realisierung von Industrial Ethernet kann die wesentliche Erkenntnis gewonnen werden, dass auch in der ethernetbasierten industriellen Kommunikation mit einem System alle Kommunikationsaufgaben abgedeckt werden. Die meisten bekannten Ansätze basieren auf dem IEEE802.3 Standard in dem die Schichten 1 und 2 des OSI-Modells bzw. in der Schicht 1 des DOD-Modells definiert sind. Die unterschiedlichen Auslegungen der Anwenderprotokolle sowie die verschiedenen Verfahren der Zeitsynchronisation der Systeme erschweren jedoch einen umfassenden wissenschaftlichen Vergleich. Für eine detaillierte Bewertung des Zeitverhaltens der Kommunikation sowie den Durchlaufzeiten einzelner Komponenten und Protokollstacks sind daher Versuchsaufbauten der wichtigsten Vertreter von Industrial Ethernettechnologien erforderlich, welche zum gegenwärtigen Zeitpunkt am Institut für Automatisierungstechnik nicht verfügbar sind. Mit dem Voranschreiten der Hersteller und Gremien zur Standardisierung der Systeme und der beginnenden Verfügbarkeit von Komponenten (z.B. ERTEC200/400 für PROFINET, ESC10 für EtherCAT, SERCON 819 für SEROCS-III) werden weiterführende Arbeiten auf diesem Gebiet der Kommunikationstechnik erheblich erleichtert.

---

<sup>71</sup> Im juristischen Sinne wird zwischen natürlichen (lebenden Personen) und nicht natürlichen Personen (Gesellschaften) unterschieden.

## 8 Thesen

- [1] Die Synchronisation von technischen Prozessen ist mit verteilt aufgebauten informationsverarbeitenden Automatisierungssystemen in Echtzeit grundsätzlich realisierbar und stellt zukünftig den Trend in der Entwicklung auf diesem Wissenschaftsgebiet dar.
- [2] Die Synchronisation von Bewegungsabläufen in verteilt aufgebauten mechanischen Systemen kann durch echtzeitfähige informationsverarbeitende Steuerungssysteme ersetzt werden. Die Basis für die Synchronisation solcher Abläufe erfordert echtzeitfähige Kommunikation im gesamten Steuerungssystem.
- [3] Für sehr spezielle Aufgabenstellungen in lokal abgeschlossenen Systemen (MIKRO-LAN), charakteristisch für den Gerätebau, ist es trotz bestehender Systeme wie SERCOS und SIMOTION zur Antriebssynchronisation sinnvoll, alternative Kommunikationsstrukturen zu realisieren.
- [4] Die bitserielle Übertragung zwischen Kommunikationsknoten mit entsprechender Bandbreite und echtzeitfähigen Zugriffsprinzipien auf das Übertragungsmedium stellen zwei wichtige Grundlagen für die Realisierung moderner Automatisierungssysteme dar.
- [5] Gegenwärtig verfügbare serielle Bussysteme unterscheiden sich durch verschiedene Zugriffsprinzipien, unterschiedliche Protokollspezifikationen und Übertragungsraten. Die technisch sinnvolle Anzahl lässt sich nach dem Zugriffsprinzip, der Protokolleffizienz und der möglichen Datenübertragungsrate auf relativ wenig standardisierte Systeme reduzieren, um den größten Teil industrieller Anforderungen abzudecken.
- [6] Die Reduzierung auf ein einziges Kommunikationssystem zur Lösung aller kommunikationstechnischen Aufgabenstellungen der Automatisierungstechnik ist aus gegenwärtiger Sicht technisch bereits möglich, kann aber als nicht sinnvoll eingestuft werden.
- [7] Der direkte Zugriff aus der Leitebene in die Sensor/Aktorebene ist informationstechnisch möglich sowie aus Unternehmenssicht erforderlich. Dabei sind spezielle Zugriffsmechanismen im informationsverarbeitenden Gesamtsystem erforderlich. Die Organisation des Zugriffes ist dabei grundsätzlich so zu gestalten, dass Prozesse welche echtzeitfähig bedient werden müssen in keiner Art und Weise gestört werden.
- [8] Kommunikationssysteme mit deterministischem zentralisiertem Zugriff auf das Kommunikationssystem, charakteristisch für Feldbusse, können mit Methoden der Graphentheorie modelliert werden.

- [9] Kommunikationssysteme mit stochastischem Zugriff auf das Kommunikationssystem, charakteristisch für Ethernet und CAN, können im deterministischen Bereich der Kommunikation mit Methoden der Graphentheorie modelliert werden.
- [10] Mit ethernetbasierten industriellen Kommunikationssystemen wird die klassische Hierarchiepyramide der Automatisierungstechnik transparenter. Als eine zwingend notwendige Voraussetzung für entsprechende Realisierungen von Industrial Ethernet kann die Einhaltung des IEEE802.3 (auch Ethernet II) Standards angesehen werden.
- [11] Der verstärkte Einsatz von Kommunikationssystemen mit stochastischem Zugriff auf das Kommunikationsmedium (CSMA/CA/CD) bildet eine wesentliche Grundlage für den Übergang von dezentralisierten zu verteilten Automatisierungskonzepten.
- [12] Kommunikationssysteme mit wahlfreiem Zugriff auf das Übertragungsmedium bilden die Grundvoraussetzung für die Integration von bekannten Funktionalitäten wie z.B. Plug and Play in echtzeitfähige industrielle Steuerungssysteme.
- [13] Mit zunehmendem Grad der Kaskadierung in verteilten Systemen verschlechtert sich die zeitliche Konstanz (Jitter) der Kommunikation. Prozesse mit sehr hohen Anforderungen an die präzise Einhaltung des Zeitmanagements insbesondere der Forderung nach Gleichzeitigkeit, lassen sich nur mit Steuerungskomponenten in abgeschlossenen Subnetzen steuern und regeln.
- [14] Das Internet als weltweites Kommunikationsmedium ist aus der Sicht der Anforderungen der informationsverarbeitenden Steuerungstechnik als definitiv nicht echtzeitfähig einzustufen. Das trifft sowohl auf weiche, harte und determiniert harte Echtzeitanforderungen der auf Kommunikation basierender industrieller Steuerungstechnik zu.
- [15] Die Integration von komplexen verteilten Automatisierungssystemen in das Internet ist informationstechnisch möglich und beginnt sich im stärkeren Maße durchzusetzen.
- [16] Komplette Automatisierungssysteme können im Internet sowohl als Client als auch als Server fungieren. Das Automatisierungssystem kann dabei als Client bereits heute eine große Anzahl von bereitgestellten Diensten nutzen. Im Gegenzug kann das gleiche Automatisierungssystem auch als Server fungieren und Dienste unterschiedlichster Art in das Internet stellen und anderen Clients zugänglich machen.
- [17] Die Nutzung von Webtechnologien im industriellen Steuerungsbereich unterliegt aufgrund bekannter Schwachstellen, vorrangig der Sicherheit, zum gegenwärtigen Zeitpunkt einer Stagnation. Die Automatisierungstechnik sollte und kann sich dennoch in Zukunft nicht vor der Integration seiner Steuerungssysteme in das weltweite Datennetz verschließen.

## 9 Literaturverzeichnis

- [2.01] Lauber, R.; Göhner, P.: Prozessautomatisierung 1 und 2, 3.Auflage, Springer-Verlag, Berlin Heidelberg New York, 1988 and 1999, ISBN 3-540-65318-X
- [2.02] Anonym: Stankovic, J.; Ramamritham, K.:
- [2.03] Tanenbaum, Andrew S.; Maarten van Steen: Verteilte Systeme, Pearson Studium Verlag & Addison-Wesley, ISBN 3-8273-7057-4
- [2.04] Irmscher, K.: Scriptum zur Lehrveranstaltung Verteilte Systeme, Universität Leipzig, Institut für Informatik, 2004  
<http://www.informatik.uni-leipzig.de/rnvs/lehre/vs/VerteilteSystemeScriptum.pdf>
- [2.05] Schild, G.-H; Kastner, W.: Prozessautomatisierung, Springer-Verlag, Wien New York, ISBN 3-211-82999-7, 1998
- [2.06] Patentschrift Siemens AG 2002 Internationale Veröffentlichungsnummer WO 02/15452 A2 „Adressvergabeverfahren für mindestens einen neu an ein Bussystem angeschlossenen Busteilnehmer“  
Anmelder: Heinrich, A.; Weißbach, B.:
- [2.07] Anders, G.; Görsch, R.; Strobelt, M.:  
Dynamische Adressvergabe dezentraler Peripheriekomponenten in Profibussystemen, TU-Freiberg, Institut für Automatisierungstechnik, 2004
- [2.08] Coulouris, G.; Dollimore, J ; Kindberg, T. : Verteilte Systeme Konzepte und Design, Pearson Studium Verlag & Addison Wesley, ISBN 3-8273-7022-1, 3.Auflage 2002
- [2.09] Leslie Lamport: Email an das DEC SRC Forum vom 28.Mai 1987  
<http://research.microsoft.com/users/lamport/pubs/distributed-system.txt>
- [2.10] Emmerich, W.: Konstruktion von verteilten Objekten, dpunkt Verlag, Heidelberg, 2003, ISBN 3-89864-140-6
- [2.11] Lüder, A.(Editor); IAONA Handbuch – Industrial Ethernet, Magdeburg, IAONA e.V., 1. Auflage, 2004
- [2.12] Schlichter, J.: Script zur Vorlesung „Verteilte Anwendungen“, TU-München, Institut für Informatik, Sommersemester 2000



- [2.13] Krauß, P.; Boshnakov, K.: Kfz-Busse CAN und FlexRay, Technische Universität München, Lehrstuhl für Datenverarbeitung, 2003  
[http://www.ldv.ei.tum.de/media/files/hauptseminar/ss2003/D12\\_CANvsFlexRay.pdf](http://www.ldv.ei.tum.de/media/files/hauptseminar/ss2003/D12_CANvsFlexRay.pdf)
- [2.14] Bärz, R.: Kommunikation in Mobilien Systemen, Seminar Institut für Softwaretechnik, Universität Koblenz-Landau, 2003
- [2.15] Kühnel, C.; Neve, A.: Verteilte Echtzeit, Hauptseminar Automotive Software Engineering, Technische Universität München, Lehrstuhl Software und Systems Engineering, 2005
- [2.16] Bachmann, B.: Grosses Lexikon der Computerbegriffe, IWT-Verlag, München, 1990, ISBN 3-88332-257-7
- [2.17] Fischer, R.: Beitrag zur Entwicklung eines Prozessdatenmanagementsystems, Dissertation A, Technische Universität Bergakademie Freiberg, 2003
- [2.18] Martin, M.: Mobilfunkbasiertes Steuerungskonzept für das Flottenmanagement Technik in Tagebauen und Minen, Dissertation A, Technische Universität Bergakademie Freiberg, 2005
- [2.19] [www.informatik.tu-muenchen.de/publications/html/geiger2003/node16.html](http://www.informatik.tu-muenchen.de/publications/html/geiger2003/node16.html)
- [3.01] Braach, B.: Kupfer für 10 GbE, Computer & Automation, 06-2005  
[www.elektroniknet.de](http://www.elektroniknet.de)
- [3.02] Simon, R.: Ethernet/IP, Besondere Merkmale - Anwendungsbereiche Echtzeit, IOANA, ODVA, Rockwell Automation, 2003, [www.ethernetip.de](http://www.ethernetip.de)
- [3.03] Schnell, G.(Hrsg.): Bussysteme in der Automatisierungs- und Prozesstechnik, Vieweg Verlagsgesellschaft, 5.Auflage, 2003, ISBN 3-528-46569-7
- [3.04] Staub, M.: Untersuchungen zum Echtzeitverhalten von Feldbussystemen mit Zugriffsverfahren CSMA/CA unter besonderer Berücksichtigung des CAN-Protokolls, Chemnitz Zwickau, Dissertation A, 1997
- [3.05] Fiedler, K.: Zeitverhaltensanalyse serieller Bussysteme mittels Lastmethode, Dissertation A, Magdeburg, 1990
- [3.06] Lingenthal, T.: Echtzeitverhalten von Feldbussystemen, München, 1993
- [3.07] Furrer, F.: Ethernet-TCP/IP für die Industrieautomation: Grundlagen und Praxis, Hüthig Verlag, Heidelberg, 2.Auflage 2000

- [3.08] Etschberger, K.: Controller Area Network (CAN) Grundlagen, Protokolle, Bausteine, Anwendungen. Dritte Auflage. Hanser Verlag, 2002
- [3.09] Schmeißer, S.: Beitrag zu offenen Automatisierungsnetzen am Beispiel eines Web-basierten Labors Automatisierungstechnik, Dissertation A, Technische Universität Chemnitz, 2004
- [3.10] Jasperneite, J.: Leistungsbewertung eines lokalen Netzwerkes mit Class-of-Service Unterstützung für die prozessnahe Echtzeitkommunikation, Dissertation A, Otto-von-Guericke-Universität Magdeburg, 2003
- [3.11] Wagner, T.: Konzeption eines Praktikumsversuches auf Basis des Aerocomm-Funksystems, Studienarbeit, TU Freiberg, 2004
- [3.12] PROFINET – offener Standard für die Automatisierung, Veranstaltung 5-7, Siemens AG 2005, [www.siemens.com/profinet](http://www.siemens.com/profinet)
- [3.13] [http://home.t-online.de/home/hans-guenter.kahl/nw/nw\\_91.htm](http://home.t-online.de/home/hans-guenter.kahl/nw/nw_91.htm)
- [3.14] Jasperneite, J.: Echtzeit-Ethernet im Überblick, atp, 2005, Heft 3
- [3.15] Schneider, K.: Ethernet für die Industrie, Automatisierung im IT-Netz, IT&Production, Zeitschrift für die industrielle Informationstechnologie, Ausgabe 2/2005, [www.it-production.com](http://www.it-production.com)
- [3.16] Tragl, K.: Der echtzeitfähige Kommunikationsstandard für Motion Control, Praxis Profiline – SERCOS Interface, Vogel Industriemedien Würzburg, 2004, ISBN 3-8259-1926-9
- [3.17] Herian, D.: IEE Automatisierung und Datentechnik, Alfred Hüthig Verlag, 5/2003 [www.iee-online.de](http://www.iee-online.de)
- [3.18] Reisner, M.(Hrsg.): Ethernet - Das Grundlagenbuch, Franzis Verlag GmbH, Poing, 2002, ISBN 3-7723-6670-8
- [3.20] Dumotier, B.: Industrial Ethernet – Ein Trend etabliert sich – Volume D/E, Vogel Industrie Medien & IAONA e.V., Würzburg, 2004, ISBN 3-8259-1925-0
- [3.21] Euler, S.; Hartman, M.: Netzwerk-Know-How, tec-Channel-Compact, IDG Interactive GmbH, München, Sonderheft 1/2005, [www.tecChannel.de](http://www.tecChannel.de)

- [3.22] Göhler, R.; Hermann, M.; Jarosz, P.; Löber, P.: Vergleichsstudie PROFIBUS – Fire Wire, Bergakademie Freiberg, Institut für Automatisierungstechnik, 1998
- [3.23] Modbus/TCP Release 1.0 Schneider Electric 1999
- [3.24] Modbus-IDA, Modbus-IDA News, Member Company News, Hannover – Messe, April 2005
- [3.25] Sikora, A.: Technische Grundlagen der Rechnerkommunikation, Internet-Protokolle und Anwendungen, Fachbuchverlag Leipzig, 2003, ISBN3-446-22455-6
- [3.26] Wimmer,H.; Meidndl,A.: Ethernet Powerlink gestern, heute morgen, SPS Magazin, HMI-Spezial, 2006
- [3.27] Beckhoff Automation GmbH, EtherCAT-Drives, SPS Magazin, HMI-Spezial, 2006
- [3.28] Lutz,P.; Selbig,A.: SERCOS III – Steuerungsvernetzung mit Echtzeit-Ethernet, SPS-Magazin, HMI-Spezial, 2006
- [3.29] Gramann, T.; Mohl.S.: Precision Time Protocol – IEEE 1588 in der Praxis, Elektronik 24/2003
- [3.30] Erweiterung des Protocolls PTP zu PTCP, Internes Dokument der PNO, 2005
- [3.31] <http://www.fh-luh.de/fb5/netlab/paper.php> März/2006
- [3.32] [www.epl-tools.com](http://www.epl-tools.com) März/2006
- [3.33] [www.br-automation.com](http://www.br-automation.com) März/2006
- [3.34] [www.ixxat.de](http://www.ixxat.de) März/2006
- [3.35] [www.ethercat.org](http://www.ethercat.org) März/2006
- [3.36] [www.bechhoff.de](http://www.bechhoff.de) März/2006
- [3.37] [www.sercos.com](http://www.sercos.com) März/2006
- [3.38] [www.sercos.de](http://www.sercos.de) März/2006
- [3.39] [www.microchip.com](http://www.microchip.com) Dezember/2006  
Datenblatt CAN-Controller MCP2515

- [3.40] [www.can-cia.org](http://www.can-cia.org) Juli/2005
- [3.41] [www.profibus.de](http://www.profibus.de) März/2006
- [3.42] [www.ttagroup.org](http://www.ttagroup.org) Juni/2005
- [3.43] [www.ttpgroup.org](http://www.ttpgroup.org) Juni/2005
- [3.44] [www.semiconductors.philips.com/i2C](http://www.semiconductors.philips.com/i2C)
- [3.45] <http://www.hms-networks.de/Technologies/indethernet.shtml>
- [3.46] <http://www-pdv.fh-reutlingen.de/rte/> März/2006
- [3.47] [www.ethernet-powerlink.org](http://www.ethernet-powerlink.org) März/2006
- [3.48] [www.fieldbus.org](http://www.fieldbus.org) März/2006
- [3.49] [www.hartcomm.org](http://www.hartcomm.org) März/2006
- [3.50] [www.interbusclub.com](http://www.interbusclub.com) März/2006
- [3.51] [www.modbus.org](http://www.modbus.org) März/2006
- [3.52] Bormann, A.; Hildenkamp.: Industrielle Netze, Ethernetkommunikation für Automatisierungsanwendungen, Alfred Hüthig Verlag Heidelberg, 2005, ISBN 3-7785-2950-1
- [3.53] Winkler, L.: Grundlagen der Kommunikationstechnik, Teil 4, Vorlesungsskript, Hochschule Mittweida, 2005
- [3.54] Riggert, W.: Rechnernetze, Grundlagen-Ethernet-Internet, 2. Auflage, Fachbuchverlag Leipzig, 2002, ISBN 3-446-21984-6
- [3.55] Performance Parameter von IRT, PROFIBUS Nutzerorganisation, 2005
- [3.56] Popp, M.: PROFIBUS-DP/DPV1, Grundlagen, Tipps und Tricks für Anwender, Hüthig GmbH & Co. KG, Heidelberg, 2000, ISBN 3-7785-2781-9
- [3.57] Popp, M.; Weber K.: Der Schnelleinstieg in PROFINET, PROFIBUS Nutzerorganisation e.V., 2004
- [3.58] Popp, M.: Das PROFINET IO-Buch, Grundlagen und Tipps für Anwender, Hüthig GmbH & Co. KG, Heidelberg, 2005, ISBN 3-7785-2966-8

- [3.59] Zwerina, K.: Realtime-Ethernet, Standard bringt Vorteile, IEE 7/8 2005
- [3.60] Institut für Automation und Kommunikation e.V. Magdeburg  
an der Otto-von-Guericke-Universität Magdeburg, Tätigkeitsbericht, 2003  
<http://www.ifak-md.de>
- [3.61] [www.bürkert.com](http://www.bürkert.com)
- [3.62] PROFINET Realtime Konzept © Profibusnutzerorganisation  
Kolloquium UNI S NOV 2003.PPT
- [3.63] [www.profibus.ch](http://www.profibus.ch) März/2006
- [3.64] [www.profibus.com](http://www.profibus.com) März/2006
- [3.65] [www.synqnet.org](http://www.synqnet.org) Dezember/2005
- [3.66] [www.motioneng.com](http://www.motioneng.com)
- [3.67] [www.iaona.org](http://www.iaona.org) März/2006
- [3.68] [www.standards.ieee.org](http://www.standards.ieee.org) Dezember/2005
- [3.69] [www.microconsult.de](http://www.microconsult.de) Januar/2006
- [3.70] [www.tttech.com](http://www.tttech.com) Juli/2005
- [3.71] [www.hirschmann.de](http://www.hirschmann.de) März/2006
- [3.72] [www.ethernetip.de](http://www.ethernetip.de) März/2006
- [3.73] Messerschmidt, R.; IAONA & Ethernet Echtzeitkonzepte, SEI-Tagung, Jühlich, 2003,  
Center Verteilte Systeme am IAF, Otto-vonGuericke Universität Magdeburg, IOANA  
[Ralf.messerschmidt@mb.uni-magdeburg.de](mailto:Ralf.messerschmidt@mb.uni-magdeburg.de)
- [3.74] Scheitlin, H.; Das „Industrial Ethernet“ und seine Anwendungsschichten, Professur für  
Automatisierungstechnik, FH Solothurn, 2000, [www.fhso.ch](http://www.fhso.ch)
- [3.75] <http://ieee1588.nist.gov/> März/2006

- [3.78] Färber, G.: Feldbus-Technik heute und morgen. Automatisierungstechnische Praxis (1994) H.11
- [3.79] <http://www.all-electronics.de/news/5/9916065f8a7.html>
- [4.01] Aspel, T.: Konzeption des Aufbaus eines universell einsetzbaren Simulationssystems, SCS – Society for Computer Simulation Int., Erlangen, Chent Istanbul San Diego, 1996
- [4.02] Rechenberg, P.; Pomberger, G.: Informatikhandbuch, Carl Hanser Verlag, München Wien, 1997
- [4.03] Liebl, F.: Simulation Problemorientierte Einführung, R.Oldenburg Verlag, München Wien, 1995
- [4.04] Siegert, H.-J.: Simulation zeitdiskreter Systeme, R.Oldenburg Verlag, München Wien, 1991
- [4.04] Sedláček, J.: Einführung in die Graphentheorie, Teubner Verlagsgesellschaft, Leipzig, 1968
- [4.05] Hopcroft, J.-E.; Ullman, J.-D.: Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, Addison-Wesley, Bonn, 1994
- [4.06] Biess, G.: Mathematik für Ingenieure, Naturwissenschaftler, Ökonomen und Landwirte, Band 21/2: Graphentheorie, Teubner Verlagsgesellschaft, Leipzig, 1988
- [4.07] Milke, B.: Integration serieller Feldbussysteme in hochdynamische Regelkreise, Shaker Verlag, Aachen, 1997
- [4.08] Zitterbart, M.: Flexible und effiziente Kommunikationssysteme für Hochleistungsnetze, Internat. Thomson Publ., Bonn, 1995
- [4.09] Uhlmann, H.: Grundlagen der elektrischen Modellierung und Simulationstechnik, Akademische Verlagsgesellschaft Geest & Portig K.-G., Leipzig, 1997
- [4.10] Kaderali, F.; Poguntke, W.: Graphen, Algorithmen, Netze, Grundlagen und Anwendungen in der Nachrichtentechnik. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig, Wiesbaden, 1995
- [4.11] Biala, J.: Mobilfunk und Intelligente Netze, Grundlagen und Realisierung mobiler Kommunikation, Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig, Wiesbaden, 996 - ISBN 3-528-15302-4

- [4.12] Seemann, J. von Gudenbeg, Wolff, J.: Software-Entwurf mit UML, Objektorientierte Modellierung mit Beispielen in Java. Springer Verlag, Berlin, Heidelberg, New York, 2000
- [4.13] Amossowa, N. N.; Gillert, H.; Küchler, U.; Maximow, J. D.: Bedienungstheorie, Eine Einführung, Teubner Verlagsgesellschaft, Leipzig, 1986
- [4.14] Böhm, F.; Weise G.: Graphen in der Datenverarbeitung, Anwendung des Markierungsmodells in Technik und Ökonomie. VEB Fachbuchverlag, Leipzig, 1981
- [4.15] König, D.; Sajkiewicz, J.; Stoyan, D.: Leistungsberechnung für Fördersysteme, VEB Deutscher Verlag für Grundstoffindustrie, Leipzig, 1985
- [4.16] Frank, M.; Lorenz, P.: Simulation diskreter Prozesse, Eine Einführung für den Anwender. VEB Fachbuchverlag, Leipzig, 1979
- [4.17] Gnedenko, B.W.; König, D.: Handbuch der Bedientheorie I,II, Akademie-Verlag, Berlin, 1984
- [4.18] Kluge, P.D.: Zufallsabhängige Fertigungsprozesse, Methoden zu ihrer Beherrschung, Verlag die Wirtschaft Berlin, 1984
- [4.19] Starke, P.H.: Analyse von Petri-Netz-Modellen, B.G. Teubner, Stuttgart, 1990
- [4.20] Fiedler, K.: Zeitverhaltensanalyse serieller Bussysteme mittels Lastmethode, Dissertation A, Magdeburg, 1990
- [4.21] Bossel, H.: Simulation dynamischer Systeme, Fried. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig, 1989, ISBN 3-528-04746-1
- [4.22] Lingenthal, T.: Echtzeitverhalten von Feldebussystemen. München, 1993
- [4.23] Wetzlar, E.C.J.: Eine Objekt-orientierte Speicherverwaltungseinheit für den Bereich der Prozessdatenverarbeitung. Verlag Shaker, Dissertation A, 1993
- [4.24] Drechsel S.: Graphenorientierte Modellierung eines PROFINET-gesteuerten Transportsystems, Diplomarbeit, TU Freiberg 2006
- [4.25] Bergmann, J.: Automatisierungs- und Prozessleittechnik. Fachbuchverlag Leipzig im Carl Hanser Verlag, München Wien, 1999

- [4.26] Krug, W.: Simulation für Ingenieure in CAD/CAM-Systemen. VEB Verlag Technik, Berlin, 1989
- [4.27] Aho, A.V.; Sethi, R.; Ullmann, D.J.: Compilerbau Teil1 Teil2, R.Oldenburg Verlag, München Wien, 1999
- [4.28] Stürz, H.; Cimander, W.: Automaten, VEB Verlag Technik, Berlin, 1972
- [4.29] Metz, J.; Merbeth, G.: Schaltalgebra, VEB Fachbuchverlag, Leipzig, 1972
- [4.30] Krapp, M.: Digitale Automaten, VEB Verlag Technik, Berlin, 1988
- [4.31] Bies, G.: Graphentheorie. B.G. Teubner Verlagsgesellschaft, Leipzig, 1988
- [4.32] Löber, P.: Modulare integrationsfähige Steuerungseinrichtungen für spurgeführte bedienarme innerbetriebliche Transportsysteme, Dissertation B, TU Chemnitz, 1988
- [4.33] Jensen, K.: Coloured Petri Nets, Basic Concepts Volume 1, Springer-Verlag, Berlin, 1997
- [4.34] Vossen, G.: Datenmodelle, Datenbanksprachen und Datenbankmanagementsysteme (4.Auflage), Wissenschaftsverlag GmbH, Oldenburg, 2000
- [4.35] Reinig, W.: Elements of Distributed Algorithms, Modelling and Analyses with Petri Nets. Springer Verlag, Berlin, 1997
- [4.36] Zuse, K.: Petri-Netze aus der Sicht des Ingenieurs, Braunschweig, 1998
- [4.37] Reisig, W.: Petri-Netze, Eine Einführung, Berlin Heidelberg – NewYork, 1982
- [4.38] Klaus, G.; Buhr, M.: Philosophisches Wörterbuch, VEB Bibliographisches Institut, Leipzig, 1974
- [4.39] Oesterreich, B.: Objektorientierte Softwareentwicklung: Analyse und Design mit der Unified modeling language, Oldenburgverlag, München Wien, 1998, ISBN 3-486-24787-5, [www.oose.de/uml](http://www.oose.de/uml)
- [4.40] Erler, T.: UML : Das Einsteigerseminar, verlag moderne industrie Buch AG & Co. Kg, Landsberg, 2004, ISBN3-8266-7363-8
- [4.41] Forbrig, P.: Objektorientierte Softwareentwicklung mit UML, Fachbuchverlag Leipzig im Carl Hanser Verlag, 2001, ISBN3-446-21572-7



- [4.42] Siegert, H.-J.: Simulation zeitdiskreter Systeme, R. Oldenbourg Verlag, München Wien, 1991
- [5.01] [www.blancomed.de](http://www.blancomed.de) März/2006
- [5.02] Verteilte Antriebssysteme in der Medizintechnik  
Anders, G.: TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Symposium, 10 Jahre Biokinetische Medizintechnik in Mittweida  
Band 3, 2000, ISSN 1437-7624
- [5.03] [www.trumpf-med.de](http://www.trumpf-med.de) März/2006
- [5.04] Prospekt Jupiter 1234229, 10/2002, Firmenschrift  
TRUMPF MEDIZIN SYSTEME GmbH, Postfach 1263 07303 Saalfeld
- [5.05] Innovation und Produkte der Trumpf Medizin Systeme, NewsLetter, Ausgabe 1-2003
- [5.06] Anders, G.: Entwicklung und Test von Softwaremodulen zur Ansteuerung linearer Schrittmotorachsen mit wählbaren Geschwindigkeits- und Beschleunigungsverhältnissen, Ingenieurhochschule Mittweida, Diplomarbeit, 1995
- [5.07] Bender, K.: Zukünftige Systemtechnik der Automatisierung, Elektronik, 1/2000
- [5.08] Schlüter, G.: Digitale Regelungstechnik interaktiv, Grundlagen zeitdiskreter Steuerungssysteme, Fachbuchverlag Leipzig im Carl Hanser Verlag, München Wien, 2000, ISBN 3-446-21477-1
- [5.09] Micescu, A.: Über die Beschreibung und Optimierung verteilter Automatisierungssysteme, Technische Universität Braunschweig, Dissertation, Shaker Verlag, 1999, ISBN 3-8265-3209-0
- [5.10] <http://www.lateinboard.de/lexikon/index.php/Microcontroller>
- [5.11] [www.berchthold.de](http://www.berchthold.de) Januar/2005
- [5.12] [www.schmitz-soehne.de](http://www.schmitz-soehne.de) Januar/2005
- [5.13] [www.agasan.com](http://www.agasan.com) Januar/2005
- [5.14] [www.admi.com](http://www.admi.com) Januar/2005
- [5.15] [www.akrus.de](http://www.akrus.de) Januar/2005

- [5.16] [www.angeloframeschini.it](http://www.angeloframeschini.it) Januar/2005
- [5.17] [www.baxter.com](http://www.baxter.com) Januar/2005
- [5.18] [www.borcal.cz](http://www.borcal.cz) Januar/2005
- [5.19] [www.brumaba.de](http://www.brumaba.de) Januar/2005
- [5.20] [www.copharn.nl](http://www.copharn.nl) Januar/2005
- [5.21] [www.delta-trade-company.com](http://www.delta-trade-company.com) Januar/2005
- [5.22] [www.fenix.cz](http://www.fenix.cz) Januar/2005
- [5.23] [www.lameris.com](http://www.lameris.com) Januar/2005
- [5.24] [www.jlipovac.hr](http://www.jlipovac.hr) Januar/2005
- [5.25] [www.mtf.hr](http://www.mtf.hr) Januar/2005
- [5.26] [www.opt-ita.com](http://www.opt-ita.com) Januar/2005
- [5.27] [www.panno-med.com](http://www.panno-med.com) Januar/2005
- [5.28] [www.ramed.cz](http://www.ramed.cz) Januar/2005
- [5.29] [www.schlumbohn-medizintechnik.de](http://www.schlumbohn-medizintechnik.de) Januar/2005
- [5.30] [www.spiritmed.cz](http://www.spiritmed.cz) Januar/2005
- [5.31] Heß, K.: Die Funktion ist wichtig – Über die Entwicklung im modernen Maschinenbau, motion world, 07/2004
- [5.32] Marketing und Entwicklung Trumpf-Med Saalfeld, Interne Informationen, 2006
- [6.01] Rudolph, R.: Kickoff, Vernetzte Hochschulen, Begleitunterlage zum Workshop, Siemens AG, Nürnberg, 1999
- [6.02] Buchwitz, M.: Jetter AG, Scope – Das moderne Industrie-Magazin, April 2002  
[www.scope-online.de](http://www.scope-online.de)
- [6.03] Kuschel. S.: AIS Automation GmbH Dresden, SPS-Magazin, März 2002

- [6.04] Sobottka, D.; Meier F.G.: Die dezentrale Automatisierungslinie von Siemens, 64. Mikroelektronik - Stammtisch, Chemnitz, 2003
- [6.05] Snell, J.; Tidwell, D.; Kulchenko, P.: Webservice-Programmierung mit SOAP, O'Reilly Verlag GmbH & Co. KG, 1. Auflage, Köln, 2002, ISBN 3-89721-159-9
- [6.06] SIMATIC NET Gerätehandbuch CP343-1IT, 6GK7-343-1GX00-0XE0, Siemens AG, Ausgabestand 2000
- [6.07] Felser, M.; Dokumentation Profibusanalysetool Amprolyzer, 2002  
[http://www.profibus.felser.ch/werkzeuge/Amprolyzer/Anleitung\\_Amprolyzer\\_D0.pdf](http://www.profibus.felser.ch/werkzeuge/Amprolyzer/Anleitung_Amprolyzer_D0.pdf)
- [6.08] SPH Kommunikation, Band 7, DP-Slave, Ausgabestand V1.0, © Siemens AG, 2000
- [6.09] SPH Ablauf der Firmwaregenerierung DP-Slaves, Ausgabestand V1.2, © Siemens AG 2001,
- [6.10] Anders, G.; Görsch, R.; Löber, P.; Brechter, D.:  
Internetbasierte Diagnose in verteilten Systemen der Automatisierungstechnik  
TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Internationale wissenschaftliche Konferenz in Mittweida, IKWM  
Band A/R 9, 2002, ISSN 1437-7624
- [6.11] Anders, G.; Görsch, R.; Strobelt, M.:  
Bedienen und Beobachten eines graphenbasierten Aufzugsmodells über das Internet durch Java-Applets  
TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Deutschland  
Schmeißer, S.: Hochschule für Technik und Wirtschaft Mittweida (FH)  
Internationale wissenschaftliche Konferenz in Mittweida, IKWM  
Band A/R 9, 2002, ISSN 1437-7624
- [6.12] Anders, G.; Arnold, H.; Göhler, R.; Görsch, R.; Hänel, U.; Löber, P.; Martin, M.:  
„Studie zum Thema: Offene verteilte Automation“, Forschungsbericht,  
TU Bergakademie Freiberg, Siemens AG Standort Chemnitz, 2000
- [6.13] Anders, G.; Brechter, D.; Görsch, R.; Löber, P.;  
„Internetbasierte Bedien- und Beobachtungskonzepte für ET200-basierte Systeme“,  
Forschungsbericht, TU Bergakademie Freiberg, Siemens AG Standort Chemnitz, 2001
- [6.14] Anders, G.; Görsch, R.; Löber, P.:  
„Internetmechanismen für ET200-basierte Systeme“, Forschungsbericht,  
TU Bergakademie Freiberg, Siemens AG Standort Chemnitz, 2002

- [6.15] Brechter, D.: Untersuchung der Möglichkeiten zur Anbindung dezentraler Peripheriegeräte an Internetbasierte Strukturen, Ingenieurpraktikum, TU-Freiberg, 2001
- [6.16] Brechter, D.: Internetbasierte Durchgriffsfunktion auf ein DP-Slave-System, Studienarbeit, TU Bergakademie Freiberg, 2001
- [6.17] Patentanmeldung „Verfahren zur Nutzung von Webservices in Systemen der Automatisierungstechnik“, Patentnummer 200E00164 DE / 200312785  
Siemens AG 2002, TU Bergakademie Freiberg 2002  
Anmelder: Anders, G.; Böhme, H.; Görsch, R.:
- [6.18] SIMATIC NET – Industrielle Kommunikation, CP343-1 IT und CP 443-1 IT, Siemens AG 2000, C79000-G8900-C120-02
- [6.19] Strobelt, M.: Entwicklung einer graphenbasierten internetfähigen Aufzugsteuerung als Laborversuch und Implementierung auf einer SPS SIMATIC-S7 mit STEP 7 und HiGraph, Bakkalaureusarbeit, Technische Universität Bergakademie Freiberg, 2002
- [6.20] TrendGuide, MicroConsult GmbH, Schule für MicroElektronik & Informationstechnologie, München, 2001
- [6.21] Arbeitsergebnis des OZA Team zur A&D Internetstrategie, Siemens AG, 2001
- [6.22] Klasen, F.: M2M-Kommunikation erfolgt zunehmend über Web-Services, Automation & Drives, Publish - Industrie Verlag GmbH, 3/05, [www.aud24.net](http://www.aud24.net)
- [6.23] [www.w3.org/soap](http://www.w3.org/soap)                      Dezember/2005
- [6.24] [www.w3.org/tr/wsdl](http://www.w3.org/tr/wsdl)                      Dezember/2005
- [6.25] Anders,G.; Görsch,R.; Löber,P.:  
Abschlussbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2003
- [6.26] Löber, P.; Gralla, R.; Henschel, M.: Portal „Prozessautomation“, TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Klaus, P.J.; Fähnrich,K.J.; Wittig, S.: Marktplatz Internet: Von e-Learning bis e-Payment, 13. Leipziger Informatik-Tage, 2005, ISBN 3-88579-401-2
- [6.27] Gäbel,M.: Beitrag zur Inbetriebnahme von Antrieben mechatronischer Produktionsmaschinen als internetbasierte Lösung, Dissertation, TU Chemnitz, 2006

## 10 Abkürzungsverzeichnis

4B5B	4Binär to 5Binär Codierung für Fast Ethernet
AMP	Arbitration on Message Priority
AR	Application Relationship (Applikationsbeziehung)
ARP	Address Resolution Protocol
ASIC	Application Specific Integrated Circuit
ASPC2	Advanced Siemens PROFIBUS Controller 2 (Mastercontroller)
ATM	Asynchron Transfer Modus
CAN	Controller Area Network
CI	ControlNet International
CIP	Control- & Information Protocol
CPU	Central Professor Unit
CR	Communication Relationship (Kommunikationsbeziehung)
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CST	Cyclic Start Telegram
DDT	Device Data Telegram
DECT	Digital Enhance Cordless Communication
DP	Dezentrale Peripherie
DPV1	Dezentrale Peripherie Profile Version 1
DPV2	Dezentrale Peripherie Profile Version 2
DXCHG	Zustand eines Slaves im zyklischen Datenaustausch
EPSPG	ETHERNET Powerlink Standardization Group
ERP	Enterprise Resource Planning
ERTEC	Enhanced Real-Time Ethernet Controller
ESC10	EtherCAT Slave Controller
ETG	EtherCAT Technology Group
FPGA	Field Programmable Gate Array
GPL	General Public Licence
GPS	Global Positioning System

HSE	High Speed Ethernet
IANA	Internet Assigned Numbers Authority
IAONA	Industrial Automation Open Networking Alliance
IDA	Interface for Distributed Automation
IEA	Industrial Ethernet Association
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGS	Interessengemeinschaft SERCOS interface e.V.
IP	Internet Protocol
IPC	Industrie PC
IRDA	Infrared Data Association
IRT	Isochrone - Realtime (PROFINET V3)
ISM	Industrial Science Medical
IWLAN	Industrial Wireless Area Network
JTWG	Joint Technical Working Group
LVDS	Low Voltage Differential Signalling
M2M	Machine - to - Machine
MDT	Master Data Telegram
MES	Manufacturing Execution System
MLT3	Multi Level Transmission 3
MTU	Maximum Transfer Unit
NIST	National Institute of Standards and Technology
ODVA	Open DeviceNet Vendor Association
OSI	Open System Interconnection Referenz Model
OZA	Ortsunabhängiger Zugriff auf Automatisierungssysteme
PCS7	Process Control System Version 7 © Siemens AG
PDA	Personal Digital Assistant
PNO	PROFIBUS Nutzerorganisation
PTCP	Precision Transparent Clock Protocol
PTP	Precision Time Protocol
QoS	Quality of Service

RFC	Request for Comment
RPC	Remote Procedure Call
RT	Realtime (PROFINET V1)
RTPS	Real-Time-Publish-Subscribe Protocol
SCNM	Slot Communication Network Management
SERCOS	Serial Realtime Communication System
SOAP	Simple Object Access Protocol
SoC	Start of Cyclic Frame
SPC3	Siemens PROFIBUS Controller 3 (Slavecontroller)
SPH	System Pflichtenheft
SPI	Synchron Serial Peripheral Interface (Motorola)
SPS	Speicherprogrammierbare Steuerung
SRT	Soft - Realtime (PROFINET V2)
STP	Spanning Tree Protocol
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TTA	Time Triggered Architecture
TTCAN	Time Triggered Controller Area Network
TTP	Time Triggered Protocol
UDDI	Universal Description, Discovery and Integration
UDP	User Datagram Protocol
UTC	Universal Time Coordinated
VHDL	Verry Hardware Description Language
WBAT	Web Based Automation Technology
WLAN	Wireless Local Area Network
WSDL	Web Services Description Language
ZVEI	Zentralverband Elektrotechnik- und Elektronikindustrie e.V.

## 11 Abbildungsverzeichnis

Abbildung 2.1:	Kommunikationsformen in informationsverarbeitenden Systemen [2.19].....	10
Abbildung 2.2:	Belegungszeiten in industriellen Kommunikationssystemen (Beispiel)..	15
Abbildung 3.1:	Einteilung nach den Übertragungssystemen (Auszug) .....	19
Abbildung 3.2:	Einteilung nach den Medienzugangsverfahren (Auszug) .....	20
Abbildung 3.3:	Klassifizierung paralleler Kommunikationssysteme.....	21
Abbildung 3.4:	Wichtige funkbasierte Systeme für den industriellen Einsatz [3.11] (Auszug) .....	27
Abbildung 3.5:	Klassisches Hierarchieprinzip der Automatisierungstechnik.....	30
Abbildung 3.6:	Klassifizierung der Echtzeitanforderungen bei Industrial Ethernet (IAONA) [3.67].....	32
Abbildung 3.7:	Relative Zeitverzögerung als Folge der Busbelastung bei Ethernet [3.07].....	36
Abbildung 3.8:	Wichtige Parameter von Zeitsynchronisationsprotokollen .....	39
Abbildung 3.9:	Klassifizierung leitungsgebundener Verfahren nach dem DOD-Modell.	39
Abbildung 3.10:	Beispiel einer MLT 3-Codierung .....	41
Abbildung 3.11:	Gegenüberstellung Architekturkonzepte für Industrial Ethernet .....	42
Abbildung 3.12:	Leistungsdaten bei PROFINET V2.....	45
Abbildung 3.13:	Telegrammformat für die Kommunikation bei PROFINET SRT [3.41].	46
Abbildung 3.14:	Telegrammformat für die Kommunikation mit EtherCAT [3.35][3.36]..	46
Abbildung 3.15:	Telegrammformat für die Kommunikation mit Ethernet/IP[.....	47
Abbildung 3.16:	Telegrammformat für die Kommunikation mit Powelink[3.26, 3.47].....	48
Abbildung 3.17:	Telegrammformat für die Kommunikation mit Modbus/TCP [ 3.51] .....	49
Abbildung 3.18:	Telegrammformat für die Kommunikation bei SERCOS-III [3.37][3.38].....	50
Abbildung 4.1:	Modellklassen [4.26].....	56
Abbildung 4.2:	Modellierungsstufen [4.21] .....	58
Abbildung 4.3:	Darstellung eines Zustandsgraphen.....	61
Abbildung 4.4:	Darstellung eines Phasenmodells .....	61
Abbildung 4.5:	Grundelemente eines algorithmischen Modells .....	62
Abbildung 4.6:	Netzabwicklung mit determinierten Verkehrsangebot.....	65
Abbildung 4.7:	Netzabwicklung mit kombiniertem Verkehrsangebot.....	66
Abbildung 4.8:	Netzabwicklung mit Broad- und Multicastnachrichten .....	68
Abbildung 4.9:	Netzabwicklung von Concast-Kommunikationsbeziehungen .....	69
Abbildung 4.10:	Abbildungsbeispiel eines Graphen als Adjazenzmatrix.....	71
Abbildung 4.11:	Stabilitätsfall des PROFIBUS .....	75
Abbildung 4.12:	Grenzlastfall des PROFIBUS .....	75
Abbildung 4.13:	Überlastfall des PROFIBUS.....	75
Abbildung 4.14:	Netzabwicklung ethernetbasierter Kommunikationssysteme .....	76
Abbildung 5.1:	Klassifikation von industriellen Steuerungskonzepten .....	80



---

Abbildung 5.2:	Zentralisiert organisiertes Steuerungskonzept für die Bewegungssynchronisation .....	82
Abbildung 5.3:	Dezentralisiert bzw. verteilt organisiertes Konzept für Bewegungssynchronisationen .....	83
Abbildung 5.4:	Client/Server Model für die Säulensteuerung als verteiltes System .....	84
Abbildung 5.5:	Algorithmus für die Bewegungssynchronisation (Auszug) .....	86
Abbildung 5.6:	IT-Struktur des Gesamtsystems .....	87
Abbildung 6.1:	Webbasierter Zugriff über S7300 und CP343-1 IT .....	96
Abbildung 6.2:	Webbasierter Zugriff durch direkte Kommunikation mit der dezentralen Peripherie .....	98
Abbildung 6.3:	Klasse-2-Master basierte Zugriffsstruktur auf ET200-Komponenten .....	99
Abbildung 6.4:	Webservicearchitektur für verteilte Automatisierungssysteme [6.17] ...	101
Abbildung 6.5:	Dynamisierung der Client-Server-Architektur für Automatisierungssysteme .....	102
Abbildung 6.6:	Schematischer Ablauf eines internetbasierten Firmwareupdates [6.17]	103
Abbildung 6.7:	Struktur der Versuchsanlage des Instituts für Analysezwecke .....	105
Abbildung 6.8:	Versuchsanlage für die web-basierte Zugriffsfunktionalität auf ET200-Komponenten .....	106
Abbildung 15.1:	Eingesetzte Komponenten und Standorte des Testaufbaus .....	138
Abbildung 15.2:	Programmablaufplan für den echtzeitfähigen Zugriff auf eine ET200IS als azyklischer Dienst .....	139
Abbildung 15.3:	Ansicht der technischen Realisierung des Gesamtsystems [5.04] .....	140
Abbildung 15.4:	Arbeitsräume des Gesamtsystems (Auszug) [5.04] .....	140
Abbildung 15.5:	Client-Server-Kommunikationsbeziehungen für eine Zykluszeit .....	141
Abbildung 15.6:	Prinzip der Offset-Kompensation zwischen PTP-Master und PTP-Slave .....	144
Abbildung 15.7:	Prinzip der Delay-Kompensation zwischen PTP-Master und PTP-Slave .....	145
Abbildung 15.8:	Beispiel: PTP-Slave geht gegenüber PTP-Master „nach“ (Offset-Kompensation) .....	146
Abbildung 15.9:	Beispiel: PTP-Slave geht gegenüber PTP-Master „nach“ (Delay-Kompensation) .....	147
Abbildung 15.10:	Beispiel: PTP-Slave geht gegenüber PTP-Master „vor“ (Offset-Kompensation) .....	148

---

## 12 Tabellenverzeichnis

Tabelle 3.1: Codetabelle für ein Halbbyte (Nibble) bei einer 4B5B Codierung.....	41
Tabelle 3.2: Klassifizierung ausgewählter Verfahren nach der IOANA (siehe Abbildung 3.6).....	43
Tabelle 3.3: Leistungsparameter von EtherCAT (Auszug).....	47
Tabelle 3.4: Leistungsparameter von Ethernet/IP (Auszug).....	47
Tabelle 3.5: Leistungsparameter von Powerlink (Auszug).....	48
Tabelle 3.6: Leistungsparameter von SERCOS-III (Auszug).....	51
Tabelle 3.7: Ansätze industrieller Ethernetkommunikation im asiatischen Raum .....	52
Tabelle 4.1: Schritte der Modellbildung [4.25].....	55
Tabelle 4.2: Simulationsstudie .....	58
Tabelle 5.1: Wichtige bereitgestellte Dienste (Auszug).....	84
Tabelle 5.2: Entwicklungsstände mit wesentlichen Merkmalen [5.32] .....	88
Tabelle 5.3: Produzierte Einheiten des „Jupiter“ seit Beginn der Markteinführung [5.32] .....	88
Tabelle 15.1: Messwerte der Prototypischen Realisierung des Updates einer ET200IS .....	137

### **13 Eidesstattliche Selbständigkeitserklärung**

Erklärung:

Ich erkläre, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Freiberg / Elsterwerda, den 08.05.2006

Dipl.-Ing. (TU)(FH) Gert Anders

## 14 Veröffentlichungen – Forschungsberichte

- [1] Anders, G.:  
REISS Zeichentechnik GmbH, Bad Liebenwerda, Ingenieurschule für Elektronik und Informationsverarbeitung Görlitz,  
Studie eines neuartigen Beschriftungsgerätes zur Elektronifizierung konventioneller Zeichentechnik, Ingenieurarbeit, 1991
- [2] Anders, G.:  
Entwicklung und Test von Softwaremodulen zur Ansteuerung linearer Schrittmotorachsen mit wählbaren Geschwindigkeits- und Beschleunigungsverhältnissen, Diplomarbeit, Hochschule für Technik und Wirtschaft Mittweida (FH), 1993
- [3] Anders, G.:  
Entwicklung und Test von Softwaremodulen zur Ansteuerung linearer Schrittmotorachsen mit wählbaren Geschwindigkeits- und Beschleunigungsverhältnissen, Diplomarbeit, Ingenieurhochschule Mittweida - Technische Universität Dresden, 1995
- [4] Anders, G.:  
TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Verteilte Antriebssysteme in der Medizintechnik, Symposium,  
10 Jahre Biokinetische Medizintechnik in Mittweida, Band 3, 2000, ISSN 1437-7624
- [5] Anders, G.; Löber, P.; Würker, R.:  
TU Bergakademie Freiberg, Deutschland  
CAN – PROFIBUS – Gateway  
14. Internationale wissenschaftliche Konferenz in Mittweida,  
Band A 5, 2000, ISSN 1437-7624
- [6] Anders, G.; Görsch, R.; Brechter, D.:  
Interaktives Lernkonzept für das Praktikum „Automatisierungstechnik“  
TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Vorträge zum 52. Berg- und Hüttenmännischen Tag 2001 in Freiberg,  
ISBN 3-86012-145-6
- [7] Römer, D.; Schmeißer, S.:  
Hochschule für Technik und Wirtschaft Mittweida (FH)  
Anders, G.: TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Interaktives Web-basiertes Labor Automatisierungstechnik  
Vorträge zum 52. Berg- und Hüttenmännischen Tag 2001 in Freiberg,  
ISBN 3-86012-145-6

- [8] Anders, G.; Görsch, R.:  
Internetmechanismen für SIMATIC/ET200-basierte Systeme  
5. Industrie Fachmesse in Chemnitz, Intec, 2002  
TU Bergakademie Freiberg, Siemens AG A&D Chemnitz
- [9] Anders, G.; Görsch, R.; Löber, P.; Brechter, D.:  
Internetbasierte Diagnose in verteilten Systemen der Automatisierungstechnik  
TU Bergakademie Freiberg, Institut für Automatisierungstechnik  
Internationale wissenschaftliche Konferenz in Mittweida, IKWM  
Band A/R 9, 2002, ISSN 1437-7624
- [10] Anders, G.; Görsch, R.; Strobel, M.:  
TU Bergakademie Freiberg, Deutschland  
Schmeißer, S.: Hochschule für Technik und Wirtschaft Mittweida (FH)  
Bedienen und Beobachten eines graphenbasierten Aufzugsmodells über das Internet  
durch Java-Applets, Internationale wissenschaftliche Konferenz in Mittweida, IKWM  
Band A/R 9, 2002, ISSN 1437-7624
- [11] Anders, G.; Görsch, R.; Löber, P.; Strobel, M.:  
Internetbasierte Diagnose in verteilten Systemen der Automatisierungstechnik  
6. Industrie Fachmesse in Chemnitz, Intec, 2003  
TU Bergakademie Freiberg, Siemens AG A&D Chemnitz
- [12] Anders, G.; Görsch, R.; Gralla, R.; Löber, P.; Strobel, M.:  
Portal „Automatisierungstechnik“  
38. Regelungstechnische Kolloquium, Boppard, 2004
- [13] Anders, G.; Göhler, R.; Löber, P.:  
Digitale Regelung in verteilten Systemen auf Basis isochron arbeitender SIMATIC /  
ET200 Komponenten, 7. Industrie Fachmesse in Chemnitz, Intec, 2004  
TU Bergakademie Freiberg, Siemens AG A&D Chemnitz
- [14] Anders, G.; Löber, P.; Strobel, M.:  
Digitale Regelung in verteilten Systemen auf Basis isochron arbeitender SIMATIC /  
ET200-Komponenten  
3. Workshop der Automatisierungstechnik in Mittweida, 2004, ISSN 1437-7624
- [15] Anders, G.; Fischer, R.:  
apromaceSYS – Ein MES für den Mittelstand  
3. Chemnitzer Tag der Automation, Tagungsband VDI/NIKA Sachsen e.V., 2004
- [16] 8. Industrie Fachmesse in Chemnitz, Intec, 2005  
TU Bergakademie Freiberg, Siemens AG A&D Chemnitz

- [17] Anders, G.; Göhler, R.; Löber, P.:  
Untersuchung des Einflusses der dezentralen Peripherie ET200iSP auf das dynamische Verhalten von digitalen Ein- und Mehrgrößenregelungen
- [18] Anders, G.; Henschel, M.; Löber, P.; Strobel, M.:  
TU Bergakademie Freiberg, Deutschland  
Ethernetkonzepte für den echtzeitfähigen Einsatz in verteilten Systemen der Automatisierungstechnik  
17. Internationale wissenschaftliche Konferenz in Mittweida, 2005, ISSN 1437-7624
- [19] Anders, G.; Arnold, H.; Görsch, R.; Göhler, R.; Görsch, R.; Hähnel, U.; Löber, P.; Martin, M.:  
Studie zum Thema „Offene verteilte Automatisierung“, Forschungsbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2000
- [20] Anders, G.; Jucht, F.; Strobel, M.; Würker, R.:  
Entwicklung eines CAN-PROFIBUS-Gateway Profibusimplementierung, Abschlussbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2000
- [21] Anders, G.; Brechter, D.; Görsch, R.; Löber, P.:  
Internetbasierte Bedien- und Beobachtungskomponente für ET200-basierte Systeme, Forschungsbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2001
- [22] Anders, G.; Görsch, R.; Löber, P.:  
Internetmechanismen für ET200-basierte Systeme, Forschungsbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2002
- [23] Anders, G.; Görsch, R.; Löber, P.:  
Abschlussbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2003
- [24] Anders, G.; Görsch, R.; Löber, P.; Strobel, M.:  
Dynamische Adressvergabe dezentraler Peripheriekomponenten in Profibussystemen, Abschlussbericht, TU Bergakademie Freiberg, Institut für Automatisierungstechnik, Siemens AG A&D, 2004
- [25] Patentanmeldung „Verfahren zur Nutzung von Webservices in Systemen der Automatisierungstechnik“, Patentnummer 200E00164 DE / 200312785  
Siemens AG 2002, TU Bergakademie Freiberg 2002  
Anmelder: Anders, G.; Böhme, H.; Görsch, R.:

## 15 Anhang

### Ermittelte Zugriffszeiten des webbasierten Fernzugriffs im Vergleich zum lokalen Zugriff über ein PG mit dem Softwaresystem PCS7

Am 1. Oktober 2002 wurden zum Meilenstein 3 des F/E Themas „Internetmechanismen für ET200-basierte Systeme (Referenten: G. Anders, R. Görsch, P. Löber und M. Strobel) in Chemnitz die wesentlichsten Ergebnisse der bestehenden Forschungskooperation vorgestellt und einer umfangreichen Evaluierung durch den Industriepartner unterzogen.

Dabei wurde die Funktionalität des webbasierten Zugriffskonzepts im Versuch praktisch nachgewiesen sowie die Messergebnisse wie in Tabelle 15.1 ersichtlich, vorgestellt und diskutiert.

Nr.	Lösung	PG -Typ	OS	CP-Typ	DP-State	Baudrate [kBaud]	Zeit *
1	Internet	PIII-450	NT4	CP5611	Clear	187,5	86m 01s
2	Internet	PIII-1000	NT4	CP5613	Clear	187,5	26m 43s
3	STEP7	PIII-1000	NT4	CP5613	Clear	187,5	3m 23s
4	Internet	PIII-450	NT4	CP5611	Clear	500,0	86m 20s
5	Internet	PIII-1000	NT4	CP5613	Clear	500,0	18m 09s
6	Internet	PIII-750	W 2k	CP5611	Clear	500,0	18m 27s
7	Internet	PIII-1000	NT4	CP5613	Unclear	500,0	18m 24s
8	STEP7	PIII-1000	NT4	CP5613	Unclear	500,0	2m 09s
9	Internet	PIII-1000	NT4	CP5613	Clear	1500,0	18m 24s
10	Internet	PIII-1000	NT4	CP5613	Unclear	1500,0	18m 29s
11	STEP7	PIII-1000	NT4	CP5613	Clear	1500,0	1m 31s
12	Internet	PIII-1000	NT4	CP5613	Unclear	3000,0	18m 46s
13	Internet	PIII-750	W 2k	CP5611	Clear	3000,0	20m 30s
14	STEP7	PIII-1000	NT4	CP5613	Clear	3000,0	1m 35s
15	Internet	PIII-1000	NT4	CP5613	Clear	6000,0	17m 59s
16	STEP7	PIII-1000	NT4	CP5613	Clear	6000,0	1m 32s
17	STEP7	PIII-1000	NT4	CP5613	Unclear	6000,0	1m 31s

\* angegebene Zeiten stellen nur Näherungswerte dar

Tabelle 15.1: Messwerte der Prototypischen Realisierung des Updates einer ET200IS

## Struktureller Aufbau des Testsystems zum Nachweis der Funktionalität des webbasierten Zugriffs auf ET200 basierte Steuerungssysteme

Für die Evaluierung der Ergebnisse des webbasierten Zugriffs auf dezentrale Peripheriekomponenten wurde ein struktureller Aufbau mit den Komponenten nach Abbildung 15.1 eingesetzt. Die gleiche Anordnung diente auf der Intec-Messe in Chemnitz 2003 dazu, Auszüge der gewonnenen Erkenntnisse und Ergebnisse dem Fachpublikum vorzustellen.

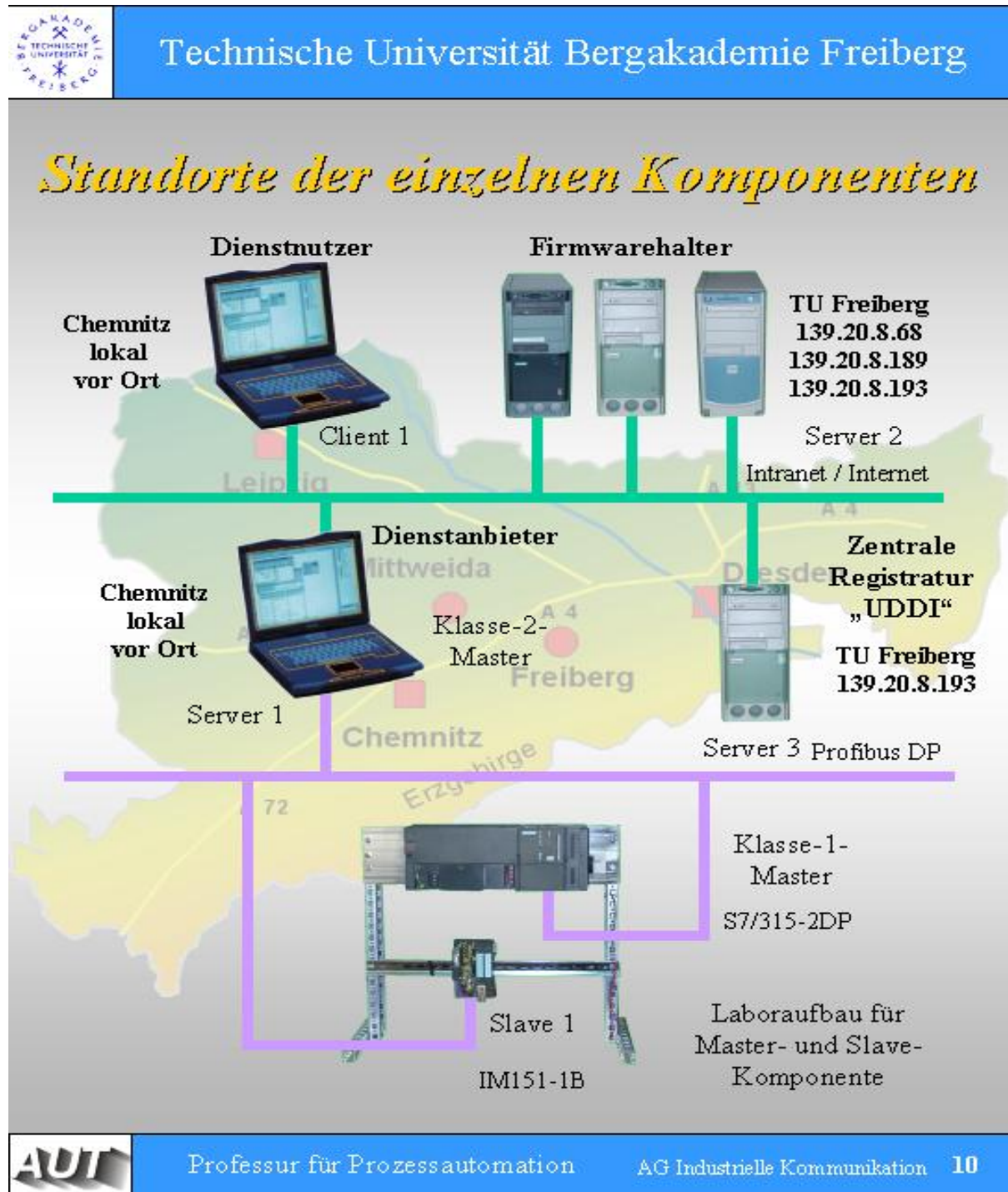


Abbildung 15.1: Eingesetzte Komponenten und Standorte des Testaufbaus<sup>72</sup>

<sup>72</sup> F/E Thema, Internetmechanismen für ET200-basierte Systeme, Meilenstein 3, Siemens Chemnitz, 2002



## Programmablaufplan für den Ablauf eines Firmwareupdates einer dezentralen Peripheriebaugruppe ET200S über PROFIBUS-DP

In Abbildung 15.2 ist der prinzipielle Ablauf eines Firmwareupdates einer dezentralen Peripheriebaugruppe ET200S dargestellt. Das Ablaufschema basiert auf den Vorgaben der Dienste und Dienstzugangspunkte für die Baugruppe, welche in den Systempflichtenheften [6.08][6.09] der Siemens AG festgelegt sind. Die Kommunikation erfolgt zum Zeitpunkt der Realisierung der Forschungsprojekte über PROFIBUS-DP (V0/V1).

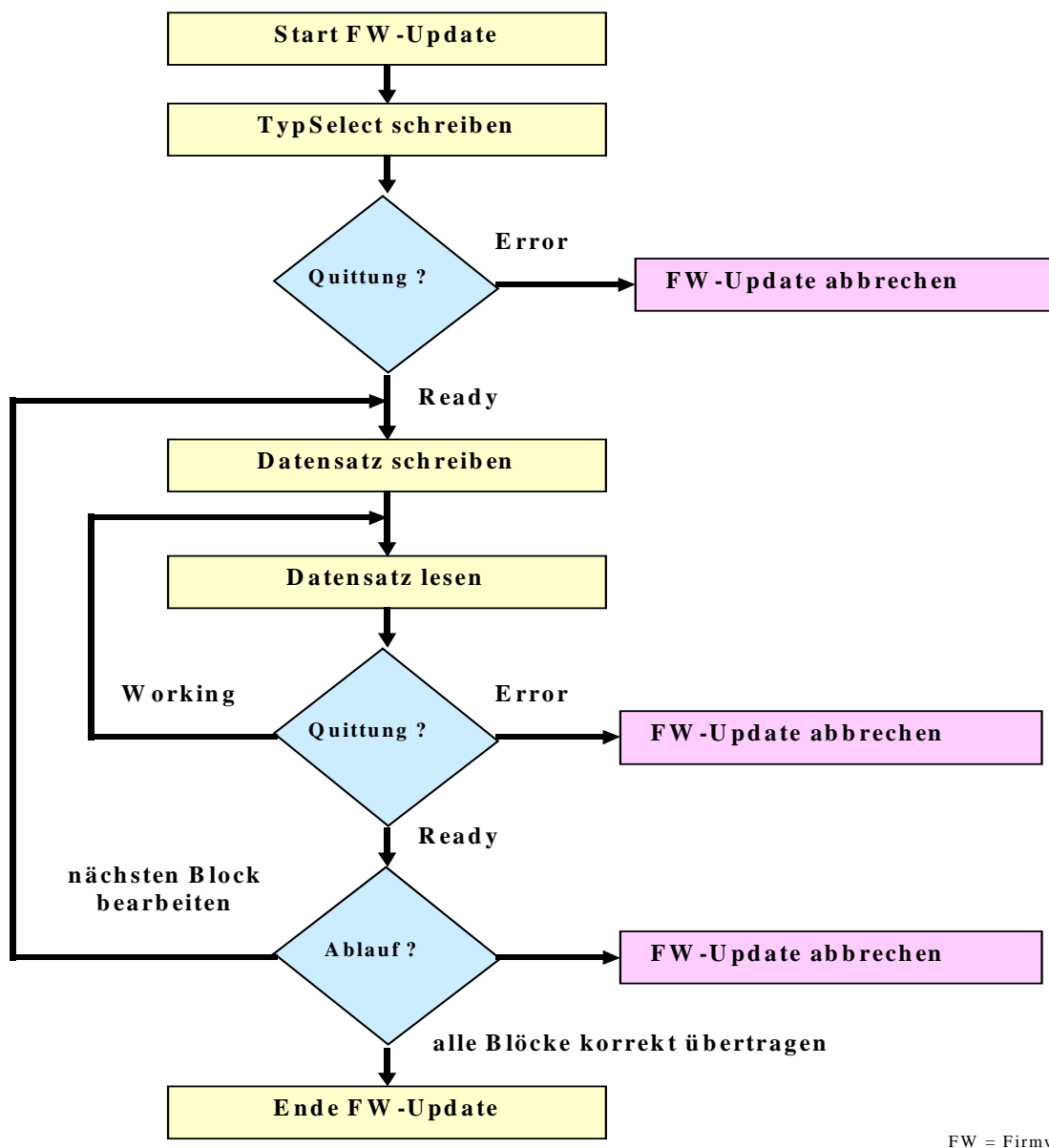


Abbildung 15.2: Programmablaufplan echtzeitfähigen Zugriff auf eine ET200IS als azyklischer Dienst

### Das modulare Patientenbewegungssystem „Jupiter“ mit seinen Positioniermöglichkeiten (Auszug)

Charakteristische Eigenschaften des Patientenbewegungssystems „Jupiter“ sind der modulare Aufbau, der Einsatz intelligenter Antriebstechnologie sowie das Konzept verteilter echtzeitfähiger Steuerungstechnologie. In Abbildung 15.3 ist das Gerätesystem in seiner maximalen Ausbaustufe dargestellt. Die Abbildung 15.4 zeigt einige Auszüge der maximal möglichen Positionen im Arbeitsbereich.



Abbildung 15.3: Ansicht der technischen Realisierung des Gesamtsystems [5.04]

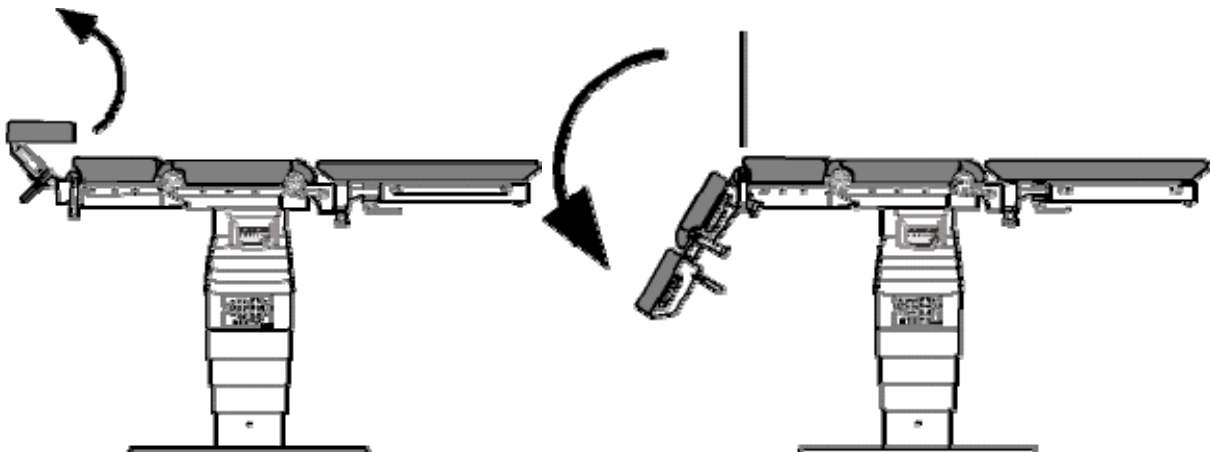


Abbildung 15.4: Arbeitsräume des Gesamtsystems (Auszug) [5.04]

### Kommunikationsbeziehungen zur Bewegungssynchronisation (Auszug)

In Abbildung 15.5 sind schematisch alle notwendigen Kommunikationsbeziehungen zwischen der Clientkomponente (Zeitmaster und Bewegungskoodinator) und den drei Serverkomponenten (intelligente Antriebssysteme) für Synchronisation eines Bewegungsablaufes in einem Zyklus darstellt. Der Zyklus wird mit einem Broadcast als Synchronotelegramm eingeleitet. Darauf folgend werden die aktuellen Positionswerte der Antriebssysteme abgefragt (im Beispiel werden alle drei Antriebserver bedient). Die eingesetzten Algorithmen zur Bewegungssynchronisation ermitteln unter Nutzung des Prinzips der virtuellen Achse die Abweichungen aus der virtuellen Sollposition und den Istpositionen der Antriebserver und berechnen die neuen Geschwindigkeitssollwerte für den folgenden Zyklus. Die Sollwerte werden an die Antriebserver übergeben und mit dem nächsten Synchronotelegramm übernommen.

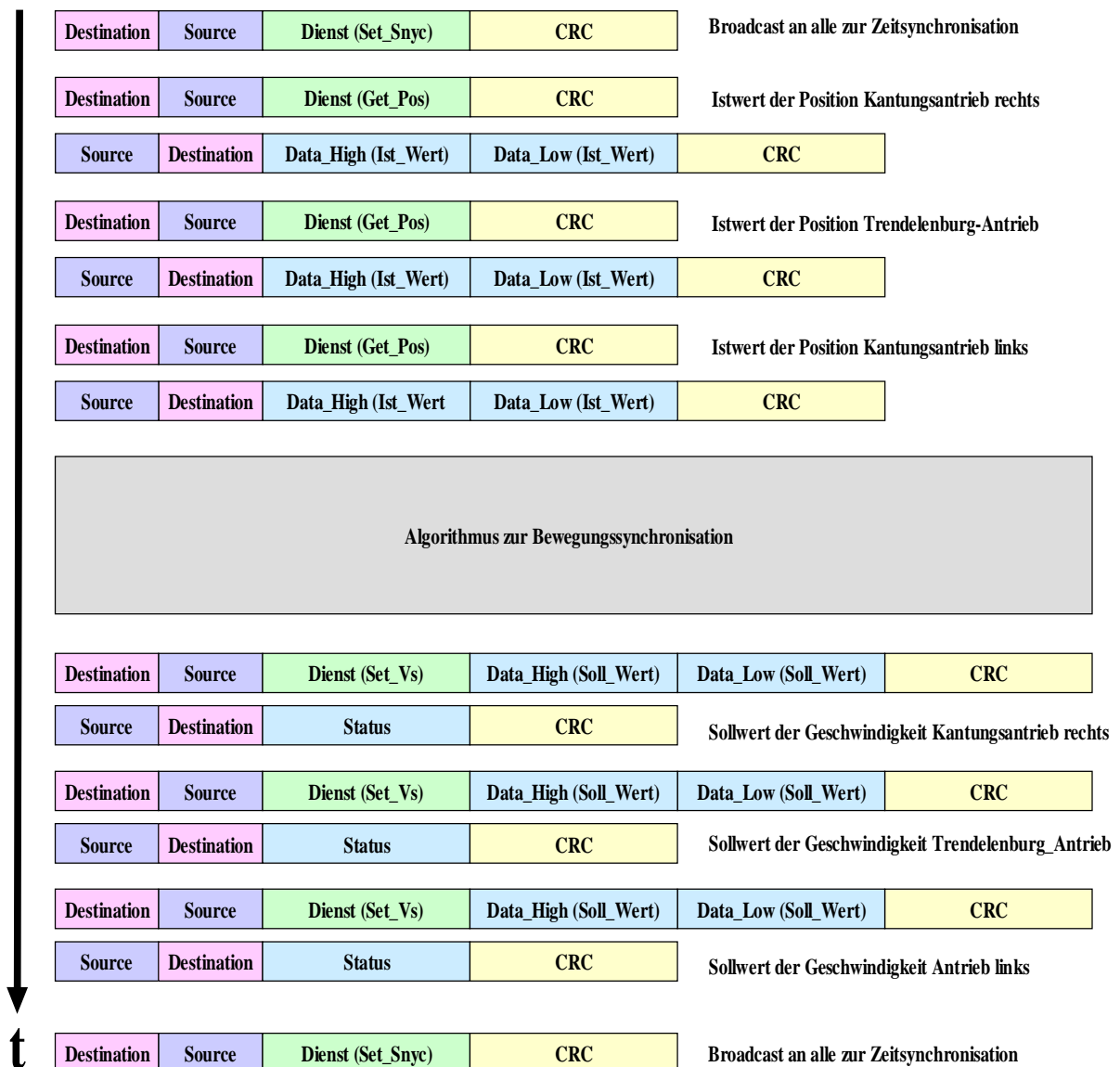


Abbildung 15.5 Client-Server-Kommunikationsbeziehungen für eine Zykluszeit

**Physikalische Grundlagen für die Synchronisation gleichförmiger Bewegungsvorgänge durch verteilte Steuerungssysteme am Beispiel von zwei Achsen beim Patientenbewegungssystem „Jupiter“**

$$v = \frac{ds}{dt} \quad (15.1)$$

In zeitdiskreten Systemen, charakteristisch ist die endliche Abtastzeit, können die Größen  $ds$  durch  $\Delta s$  sowie  $dt$  durch  $\Delta t$  ersetzt werden.

$$\Delta s = V_{soll} * \Delta t \quad (15.2)$$

Bei einem äquidistant betriebenen verteilten Kommunikationssystem wird die Zeit  $\Delta t$  durch  $T_{Zyklus}$  ersetzt. Die verfahrenre Wegstrecke im Kommunikationszyklus ergibt sich somit aus

$$\Delta S_{soll} = V_{soll} * T_{Zyklus} \quad (15.3)$$

Die zurückgelegte gesamte Wegstrecke seit Synchronisationsbeginn wird mit folgender Gleichung ermittelt:

$$S_{soll} = \Delta S_{soll} * n \quad (15.4)$$

Die Abweichungen der beiden Antriebssysteme von der aktuellen Sollposition ergeben sich jeweils aus der Differenz von Sollposition und Istposition.

$$\Delta S_{soll\_r} = S_{soll} - S_{ist\_r} \quad (15.5)$$

$$\Delta S_{soll\_l} = S_{soll} - S_{ist\_l} \quad (15.6)$$

Die erforderliche Korrektur der Sollgeschwindigkeiten für die beiden Antriebe für den folgenden Zyklus ergeben sich mit  $\Delta S_{soll} = V_{soll} * T_{Zyklus}$

$$\Delta V = \frac{\Delta S_{soll}}{\Delta T_{Zyklus}} \quad (15.7)$$

$$\Delta Vr = \frac{\Delta S_{soll\_r}}{\Delta T_{Zyklus}} \quad (15.8)$$

$$\Delta V_l = \frac{\Delta S_{soll\_l}}{\Delta T_{Zyklus}} \quad (15.9)$$

Die neuen absoluten Geschwindigkeitswerte für den folgenden Kommunikationszyklus lassen sich wie folgt berechnen:

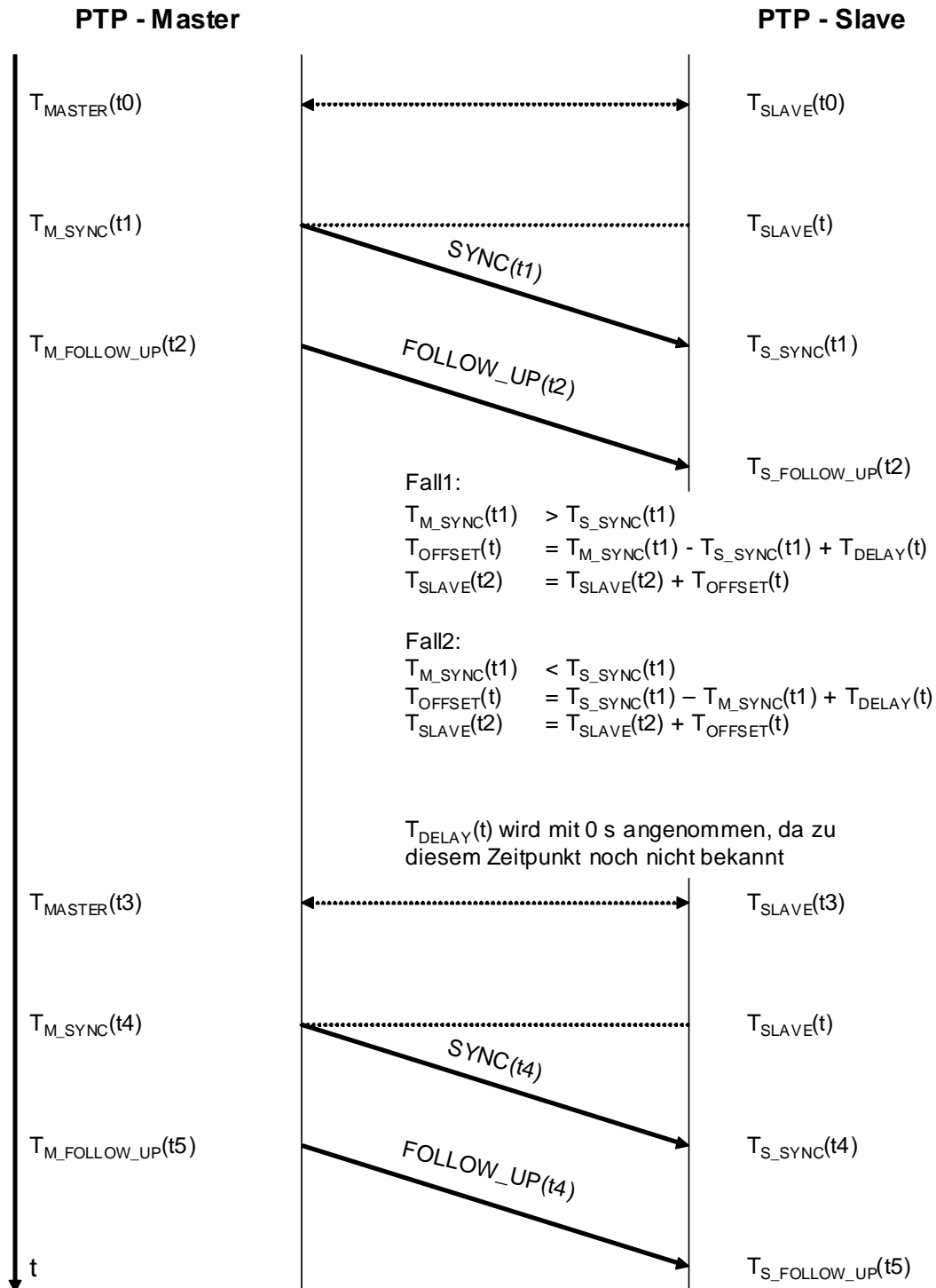
$$V_{stell\_r} = V_{soll} + \Delta V_r \quad (15.10)$$

$$V_{stell\_l} = V_{soll} + \Delta V_l \quad (15.11)$$

### Legende :

$S_{soll}$	Sollwert der aktuellen Position der resultierenden Bewegung
$V_{soll}$	Sollwert der Geschwindigkeit der resultierenden Bewegung
$\Delta S_{soll}$	Sollwert der Positionsänderung pro Zeitintervall (Kommunikationszyklus)
$T_{Zyklus}$	Zeitintervall des Kommunikationszyklus
$n$	Anzahl der abgearbeiteten Zyklen
$S_{ist\_l}$	Istwert der aktuellen Position des linken Antriebes
$S_{ist\_r}$	Istwert der aktuellen Position des rechten Antriebes
$\Delta S_{soll\_r}$	Sollwertabweichung rechter Antrieb
$\Delta S_{soll\_l}$	Sollwertabweichung linker Antrieb
$\Delta V_r$	Regelabweichung Antrieb rechts
$\Delta V_l$	Regelabweichung Antrieb links
$V_{stell\_r}$	Stellgröße Antrieb rechts
$V_{stell\_l}$	Stellgröße Antrieb links

## Zeitsynchronisation in ethernetbasierten verteilten Systemen nach der Norm IEEE 1588 Teil 1 : Prinzip der Offset-Kompensation zwischen PTP-Master und PTP-Slave



**Uhren laufen bis auf Delay-Error synchron! (Offset-Error ist korrigiert)**

Abbildung 15.6: Prinzip der Offset-Kompensation zwischen PTP-Master und PTP-Slave

## Zeitsynchronisation in ethernetbasierten verteilten Systemen nach der Norm IEEE 1588 Teil 2: Prinzip der Delay-Kompensation zwischen PTP-Master und PTP-Slave

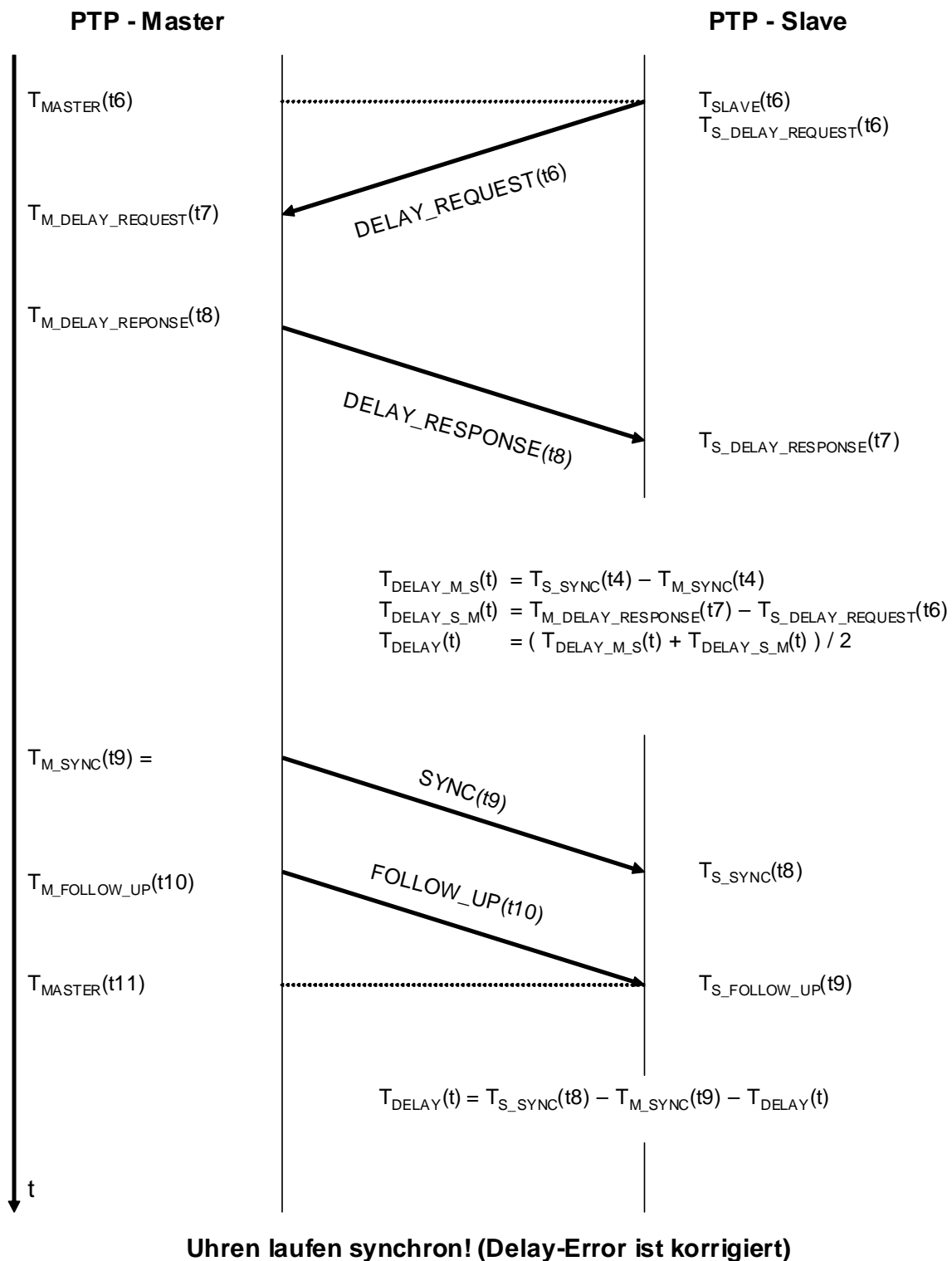
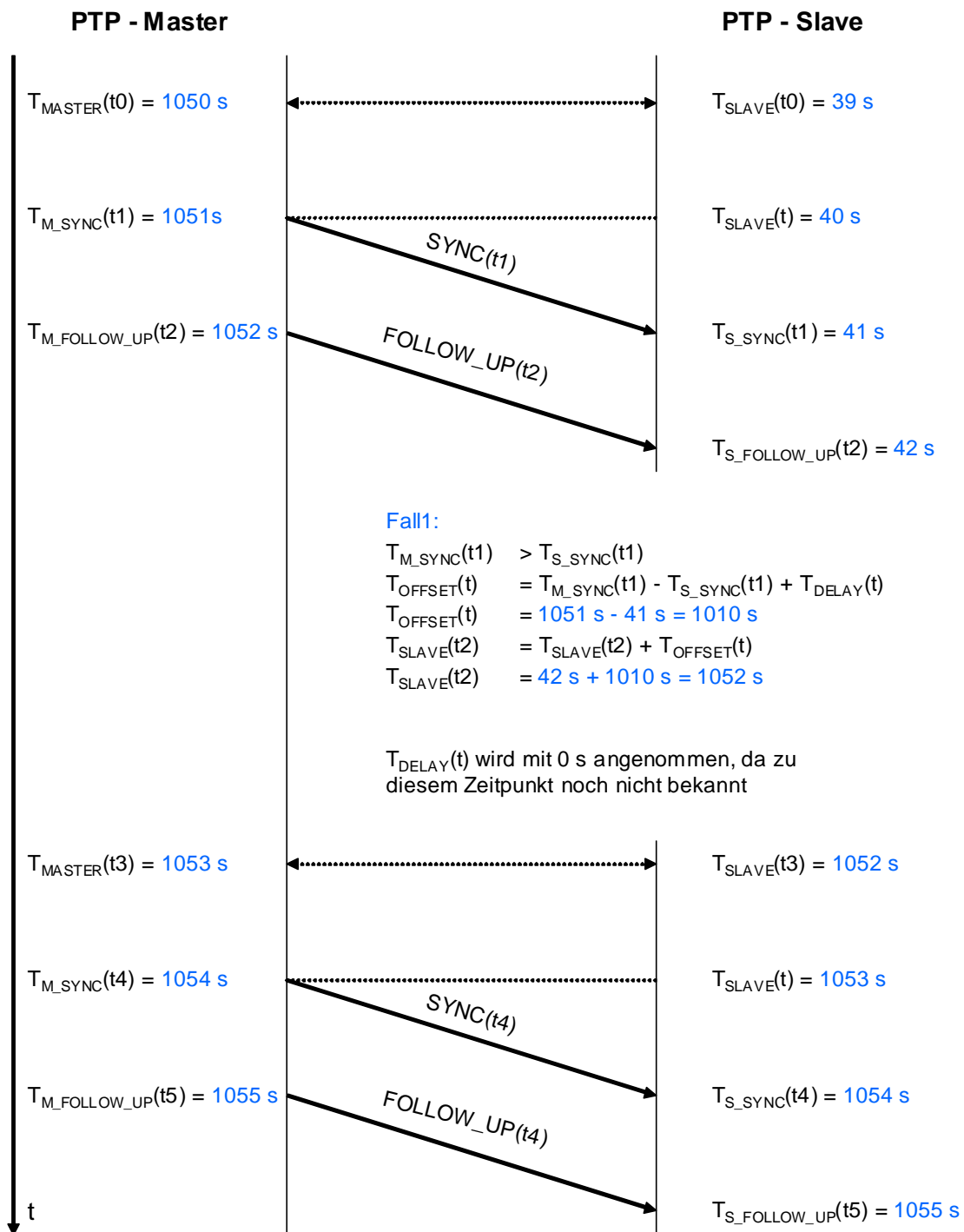


Abbildung 15.7: Prinzip der Delay-Kompensation zwischen PTP-Master und PTP-Slave

**Zeitsynchronisation in ethernetbasierten verteilten Systemen nach der Norm IEEE 1588**  
**Beispiel: Uhr des PTP-Slave geht gegenüber dem PTP-Master „nach“ (Teil 1: Offset)**

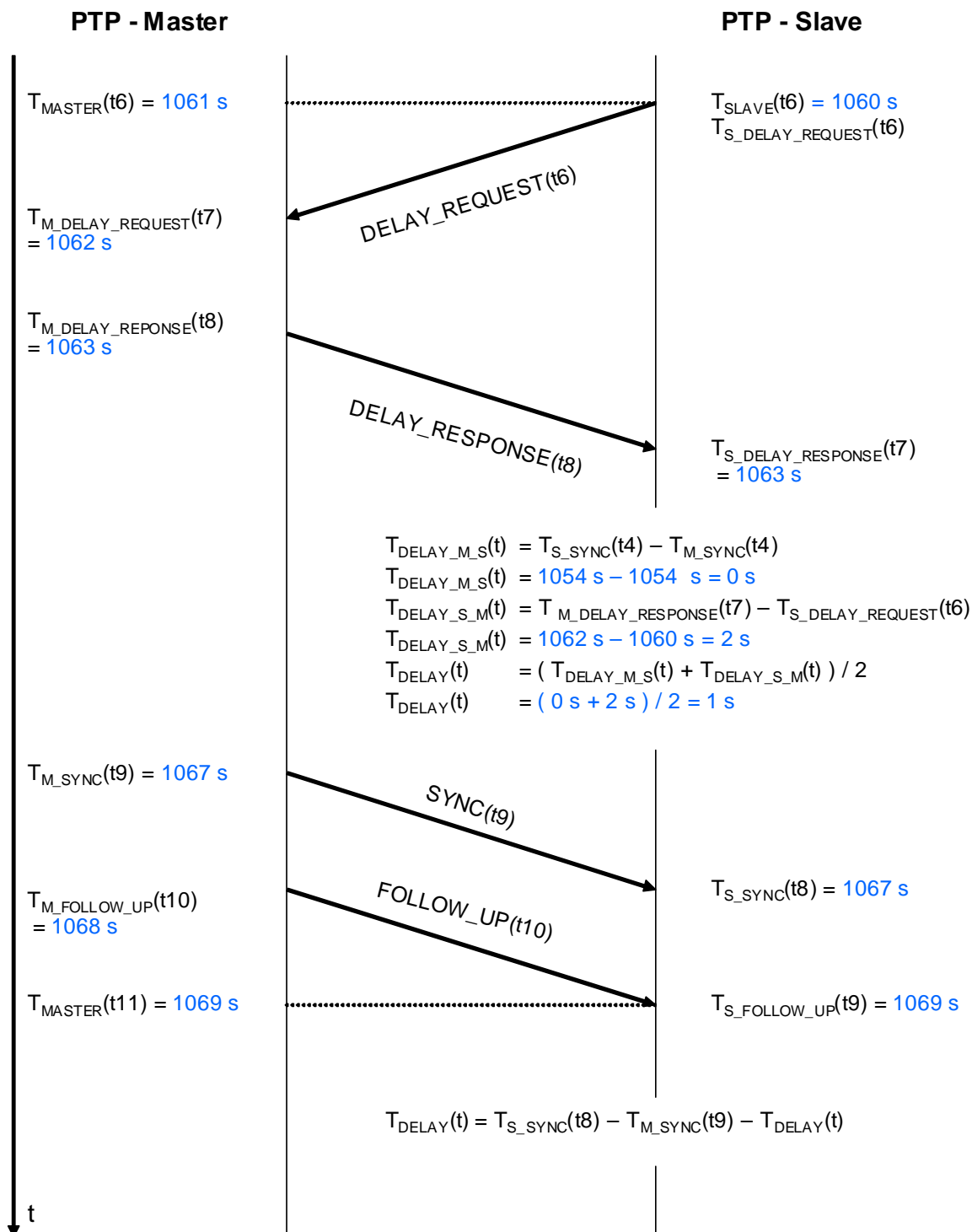


**Uhren laufen bis auf Delay-Error synchron! (Offset-Error ist korrigiert)**

**Abbildung 15.8: Beispiel: PTP-Slave geht gegenüber PTP-Master „nach“ (Offset-Kompensation)**



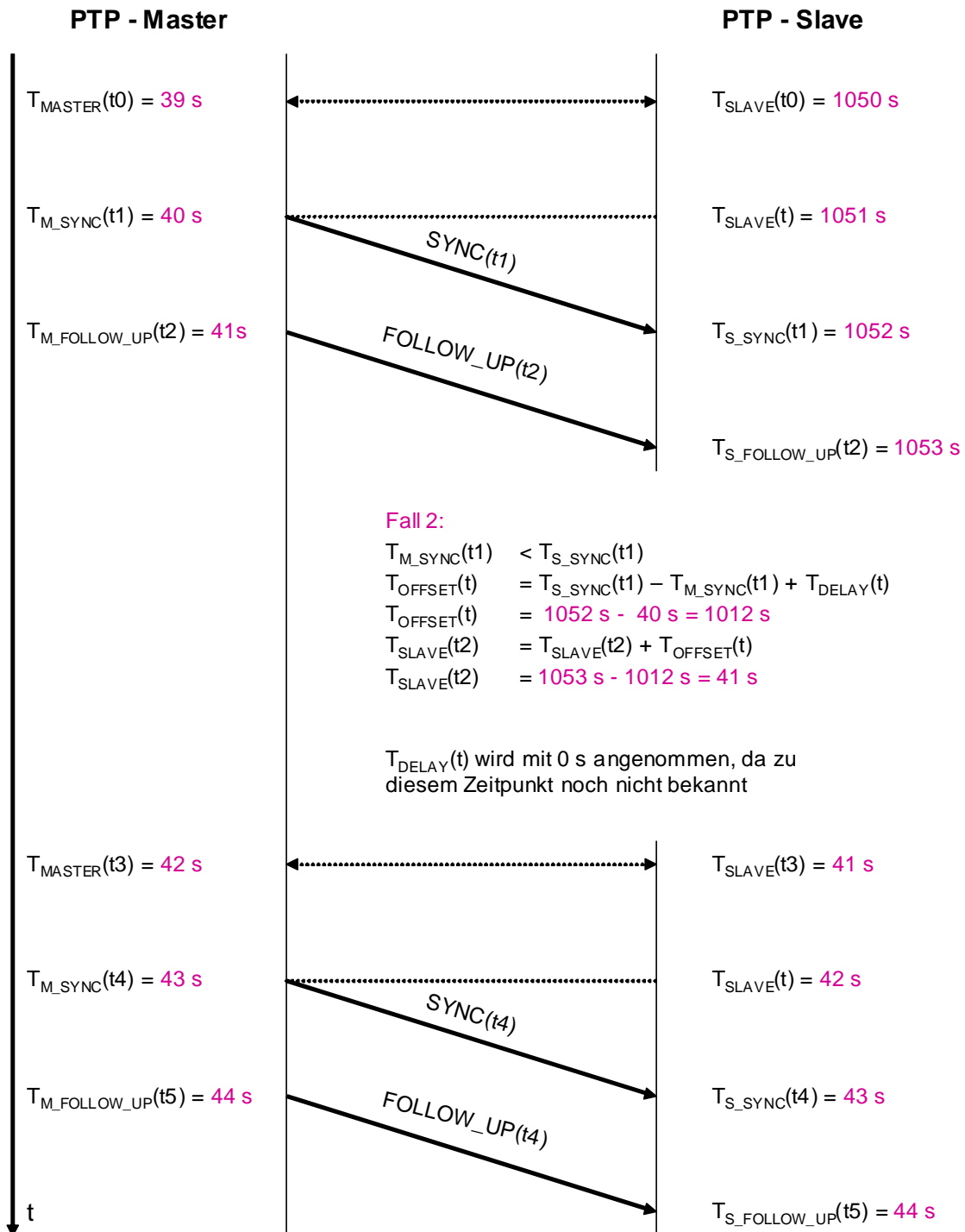
**Zeitsynchronisation in ethernetbasierten verteilten Systemen nach der Norm IEEE 1588**  
**Beispiel: Uhr des PTP-Slave geht gegenüber dem PTP-Master „nach“ (Teil 2: Delay)**



**Uhren laufen synchron! (Delay-Error ist korrigiert)**

**Abbildung 15.9: Beispiel: PTP-Slave geht gegenüber PTP-Master „nach“ (Delay-Kompensation)**

**Zeitsynchronisation in ethernetbasierten verteilten Systemen nach der Norm IEEE 1588**  
**Beispiel: Uhr des PTP-Slave geht gegenüber dem PTP-Master „vor“ (Teil 1: Offset)**



**Uhren laufen bis auf Delay-Error synchron! (Offset-Error ist korrigiert)**

**Abbildung 15.10 Beispiel: PTP-Slave geht gegenüber PTP-Master „vor“ (Offset-Kompensation)**