

Christian Aschauer

Softwareentwicklung für die Feldbussysteme - INTERBUS, PROFIBUS, CAN und ETHERNET

Am PC unter MS-Windows mit proprietären
Softwarebibliotheken und dem Industriestandard OLE for
Process Control (OPC) unter Verwendung der
Programmiersprachen Visual Basic und Visual C++

Diplomarbeit

Bibliografische Information der Deutschen Nationalbibliothek:

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Copyright © 2004 Diplomica Verlag GmbH
ISBN: 9783832486686

Christian Aschauer

Softwareentwicklung für die Feldbussysteme - INTER-BUS, PROFIBUS, CAN und ETHERNET

Am PC unter MS-Windows mit proprietären Softwarebibliotheken und dem Industriestandard OLE for Process Control (OPC) unter Verwendung der Programmiersprachen Visual Basic und Visual C++

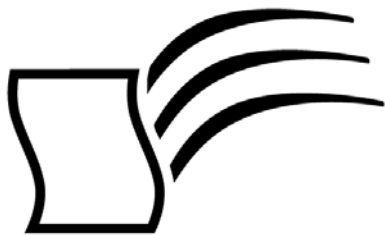
Christian Aschauer

**Softwareentwicklung für die Feldbussysteme -
INTERBUS, PROFIBUS, CAN und ETHERNET -
am PC unter MS-Windows mit proprietären
Softwarebibliotheken und dem Industriestandard
OLE for Process Control (OPC)**

*Unter Verwendung der Programmiersprachen Visual Basic
und Visual C++*

Diplomarbeit
Fachhochschul-Studiengänge der Wiener Wirtschaft GmbH

Abgabe Juni 2004



Diplom.de

Diplomica GmbH _____
Hermannstal 119k _____
22119 Hamburg _____

Fon: 040 / 655 99 20 _____
Fax: 040 / 655 99 222 _____

agentur@diplom.de _____
www.diplom.de _____

ID 8668

Aschauer, Christian: Softwareentwicklung für die Feldbussysteme - INTERBUS, PROFIBUS, CAN und ETHERNET - am PC unter MS-Windows mit proprietären Softwarebibliotheken und dem Industriestandard OLE for Process Control (OPC) - Unter Verwendung der Programmiersprachen Visual Basic und Visual C++
Hamburg: Diplomica GmbH, 2005
Zugl.: Fachhochschul-Studiengänge der Wiener Wirtschaft GmbH, Diplomarbeit, 2004

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Die Informationen in diesem Werk wurden mit Sorgfalt erarbeitet. Dennoch können Fehler nicht vollständig ausgeschlossen werden, und die Diplomarbeiten Agentur, die Autoren oder Übersetzer übernehmen keine juristische Verantwortung oder irgendeine Haftung für evtl. verbliebene fehlerhafte Angaben und deren Folgen.

Diplomica GmbH
<http://www.diplom.de>, Hamburg 2005
Printed in Germany

ABSTRACT

The basis of this thesis is the development of software to integrate personal computer based test equipment into existing production lines.

This thesis describes the different possibilities to establish communication between a Microsoft Windows driven personal computer and other field devices like programmable logic controllers. Software development for the communication over INTERBUS, PROFIBUS, CAN and ETHERNET with Visual C++ and Visual Basic is described in detail.

Chapter 1 portrays the basics of field-bus-systems, where and why such systems make sense. The basics and problems of the vertical integration are also explained in this chapter.

Chapter 2 is about the fundamentals of software development for personal computers. It deals with the problems of timing in software using threads, polling and timer mechanisms.

The development of software to get access to above mentioned field bus systems using proprietary software libraries is explained in chapter 3, 4, 5 and 6. The most important things are explained with pieces of software code.

Chapter 7 works on the international standard for process automation also called OPC. The development of OPC-clients with Visual C++ and Visual Basic is described in detail.

A summary and recommendations for selecting the right technique to gain access to the desired information are given in chapter 8.

KURZFASSUNG

Um PC-basierende Prüfsysteme in bestehende Produktionsanlagen zu integrieren ist eine Einbindung in bestehende Feldbussysteme erforderlich. Für diese PC-basierenden Systeme ist es notwendig Software zu entwickeln welche dies ermöglicht. Die vorliegende Diplomarbeit beschreibt in Theorie und Praxis wie Software für Feldbussysteme am PC unter MS-Windows mit den Programmiersprachen Visual Basic und Visual C++ entwickelt werden kann. Diese Arbeit bietet Softwareentwicklern theoretisches und praktisches Wissen, welches zur Softwareentwicklung für diese Technologien notwendig ist.

Kapitel 1 gibt einen allgemein gehaltenen Überblick über den Bereich Bussysteme. Warum diese Technologien immer mehr zum Einsatz kommen und welche Rolle sie in Unternehmen spielen. Weiters werden Kosten, Nachteile und Vorteile behandelt.

Kapitel 2 befasst sich mit dem Thema Softwareentwicklung im industriellen Umfeld. Techniken wie Multithreading und die Verwendung von Zeitgebern werden erklärt und verglichen.

Kapitel 3 erläutert die Grundlagen des INTERBUS. Weiters wird die Theorie erläutert welche erforderlich ist, um Software für die Realisierung einer PC-Interbus-PC Verbindung mit einer PC-Karte der Firma Phoenix Contact zu entwickeln. Im dritten Teil des Kapitels Interbus wird die Entwicklung von Software in C++ anhand von Programmcode beschrieben.

Kapitel 4 fasst im ersten Teil die Grundlagen des PROFIBUS zusammen. Weiters wird die Theorie erklärt welche notwendig ist um Software für die Realisierung einer PC-PROFIBUS-SPS Anbindung mit einer PC-Karte und einer speicherprogrammierbaren Steuerung der Firma Siemens zu entwickeln. Im dritten Teil des Kapitels Profibus wird die Entwicklung der Software für Visual Basic und C++ anhand von Programmcode gezeigt.

Kapitel 5 verschafft einen Einblick in die Grundlagen des Controller Area Networks (CAN). Weiters wird die Theorie beschrieben welche notwendig ist um Software für die Realisierung einer PC-CAN-SPS Verbindung mit dem Parallelport Interface BU104 der Firma Sigmatek zu entwickeln. Im dritten Teil des Kapitels CAN wird die Entwicklung der

Software mit Visual Basic und Visual C++ anhand von Programmcode gezeigt.

Kapitel 6 verschafft Einblicke in die Grundlagen von ETHERNET. Weiters wird die Theorie behandelt welche notwendig ist um Software für die Realisierung einer PC-Ethernet-PC Verbindung mit Standard Ethernetkarten zu realisieren. Im dritten Teil des Kapitels Ethernet wird detailliert die Entwicklung von Software für VB und C++ anhand von Programmcode gezeigt.

Kapitel 7 umfasst im ersten Teil die Grundlagen von OPC und dem Component Object Modell (COM) und zeigt, welche Vorteile OPC bringt. Die Theorie, welche für die Realisierung von OPC-Clients notwendig ist, wird erläutert. Auch die Entwicklung von Client-Software in VB und C++ wird anhand von Programmcode gezeigt. Den Abschluss dieses Kapitels bildet eine Schritt für Schritt Anleitung um Daten aus der Feldebene von einer SPS über den OPC-Server der Firma Sigmatek mit einem Visual Basic-Client zu visualisieren.

Kapitel 8 fasst die Realisierungen zusammen, analysiert die Ergebnisse, Vorteile, Nachteile, Erfahrungen und ungelöste Probleme.

Inhaltsverzeichnis

ABSTRACT.....	3
KURZFASSUNG.....	4
1 BUSSYSTEME.....	9
1.1 WAS SIND BUSSYSTEME ?.....	9
1.2 WARUM WERDEN BUSSYSTEME VERWENDET ?.....	9
1.2.1 Von der seriellen zur parallelen Verdrahtung.....	9
1.2.2 Kostenersparnis durch Feldbussysteme ?.....	11
1.2.3 Anforderungen an Feldbussysteme.....	12
1.2.4 Die vertikale Integration.....	13
1.3 AUFBAU VON FELDBUSSYSTEMEN.....	16
1.3.1 Busstrukturen.....	16
1.3.1.1 Linien- oder Busstruktur.....	16
1.3.1.2 Baumstruktur.....	16
1.3.1.3 Ringstruktur.....	17
1.3.1.4 Kombinationen von Baum- und Ringstruktur.....	17
1.3.2 Modelle.....	18
1.3.2.1 Producer-Consumer / Client-Server.....	18
1.3.2.2 Nachrichtenorientiert vs. E/A-orientiert	20
1.3.3 Physikalische Medien.....	22
1.3.4 Buszugriffsverfahren.....	22
1.3.4.1 Master-Slave Verfahren.....	22
1.3.4.2 Token-Passing Verfahren.....	23
1.3.4.3 Summenrahmen-Verfahren.....	24
1.3.4.4 CSMA/CD-Verfahren.....	26
1.3.4.5 CSMA/CA-Verfahren.....	26
1.3.5 Datensicherung.....	26
1.3.5.1 Paritybit.....	27
1.3.5.2 Prüfsumme.....	28
1.3.5.3 CRC-Cyclic Redundancy Check.....	28
2 SOFTWAREENTWICKLUNG AM PC	29
2.1 DIE ROLLE DES WINDOWS PC's IN DER FELDEBENE.....	29
2.2 DIE VERSCHIEDENEN AUFGABEN DER SOFTWARE.....	29
2.2.1 Polling.....	31
2.2.2 Multithreading.....	32
2.2.3 Timer.....	33
3 INTERBUS.....	36
3.1 THEORIE.....	36
3.1.1 Buszugriffsverfahren.....	36
3.1.2 Topologie.....	36
3.1.3 Das INTERBUS Übertragungsprotokoll.....	37
3.1.4 Prozessdaten / Parameterdaten.....	37
3.1.5 INTERBUS und das ISO/OSI -Modell.....	39
3.1.6 Systemdaten.....	39
3.2 PRAXIS.....	40
3.2.1 Die Software.....	40
3.2.2 Die Hardware.....	40
3.2.2.1 Der Aufbau zu Softwareentwicklung.....	40
3.2.2.2 Die INTERBUS Systemkoppleranschaltbaugruppe	41
3.2.2.3 Die CMD-Software von Phoenix Contact.....	43

3.2.3	Projektierung.....	43
3.2.4	Die realisierte Software.....	44
3.2.4.1	Zugriff mit C++ über das Device Driver Interface	44
4	PROFIBUS DP.....	55
4.1	THEORIE.....	55
4.1.1	Datenübertragungsverfahren & Topologie.....	55
4.1.2	Leistungsdaten.....	56
4.1.3	GSD-Datei.....	57
4.2	PRAXIS.....	57
4.2.1	Die Software.....	57
4.2.1.1	Funktionsübersicht der Firmware.....	57
4.2.2	Die Hardware.....	58
4.2.2.1	Die PROFIBUS PC-Einsteckkarte CP5611 für den PC	58
4.2.2.2	Die Simatic S7-314 DP.....	58
4.2.2.3	Der Aufbau zur Softwareentwicklung.....	59
4.2.3	Die realisierte Software	59
4.2.3.1	Zugriff mit C++.....	59
5	CAN-CONTROLLER AREA NETWORK.....	64
5.1	THEORIE.....	64
5.1.1	Datenübertragungsverfahren & Topologie.....	64
5.1.2	Leistungsdaten.....	64
5.1.3	Datenorganisation in der SPS.....	64
5.2	PRAXIS.....	65
5.2.1	Der Aufbau zur Softwareentwicklung.....	65
5.2.2	Die Software.....	65
5.2.2.1	Die Funktionen der Firmware.....	65
5.2.2.2	C++ Beispielapplikation.....	68
5.2.2.3	Visual Basic Beispielapplikation.....	70
5.2.2.4	Das Programm in der SPS DCC080	71
6	ETHERNET.....	72
6.1	THEORIE.....	72
6.1.1	Datenübertragungsverfahren & Topologie.....	72
6.1.2	Echtzeitverhalten.....	74
6.1.3	Windows Sockets.....	76
6.1.3.1	Das Winsock 2 Control.....	76
6.2	PRAXIS.....	77
6.2.1	Der Aufbau zur Softwareentwicklung.....	77
6.2.2	Die realisierte Software	78
6.2.3	Ein Client-Server Anwendung in VBasic mit dem Winsock2 Control.....	78
6.2.3.1	Der Ethernet UDP-Server.....	78
6.2.3.2	Der Ethernet UDP-Client.....	80
7	OPC-OLE FOR PROCESS CONTROL.....	83
7.1	WAS IST OPC?.....	83
7.2	WIE & WO HILFT OLE FOR PROCESS CONTROL?.....	83
7.3	THEORIE.....	87
7.3.1	COM - Component Object Modell.....	87
7.3.3	Der OPC-Server.....	91
7.3.4	Die Schnittstellen des OPC-Servers.....	93
7.3.5	OPC-DA.....	93
7.4	PRAXIS.....	94
7.4.1	Cliententwicklung mit C++.....	94

7.4.2 Cliententwicklung mit Visual Basic.....	100
7.4.2.1 Schritt 1 - Programmierung der SPS.....	103
7.4.2.2 Schritt 2 - Generieren der Konfigurationsdatei für den OPC-Server.....	103
7.4.2.3 Schritt 3 - Programmierung des OPC-Clients	104
8 ZUSAMMENFASSUNG.....	115
8.1 ALLGEMEINES.....	115
8.2 EMPFEHLUNGEN FÜR SOFTWAREENTWICKLER.....	115
8.3 ÜBERSICHT DER ENTWICKELTEN APPLIKATIONEN.....	117
8.4 ETHERNET - DER FELDBUS DER ZUKUNFT ?.....	117
Literatur und Datenblätter im Internet.....	118
Literaturverzeichnis.....	119
Abbildungsverzeichnis.....	120
Tabellenverzeichnis.....	121

1 BUSSYSTEME

1.1 WAS SIND BUSSYSTEME ?

Definition:

Ein Feldbus lässt sich wie folgt beschreiben, adaptiert von [1]

Ein Feldbus ist ein Netzwerk zur seriellen und digitalen Datenübertragung. Es gibt keine direkte Verkabelung sondern alle Aktoren und Sensoren sind über ein Bussystem miteinander verbunden. Feldbussysteme ersetzen die analogen und parallelen Verkabelungen durch eine seriell arbeitende Busleitung.

1.2 WARUM WERDEN BUSSYSTEME VERWENDET ?

Folgende Kapitel beantworten die Frage warum Bussysteme zur Anwendung kommen, welche Kostenersparnisse sich ergeben und welche Anforderungen an Feldbussysteme gestellt werden. Auch auf die Problematik der vertikalen Integration wird eingegangen.

1.2.1 Von der seriellen zur parallelen Verdrahtung

Als in den sechziger Jahren erstmals Digitalrechner zur Führung von technischen Prozessen zum Einsatz kamen, wurden zunächst streng zentrale Strukturen verwendet. Ein einzelner Prozessrechner wurde mit einem Prozess verbunden. Über digitale Ein/Ausgänge wurden binäre Sensoren wie Schalter und Relais mit dem Prozessrechner verbunden.

Über analoge Eingänge war es möglich Sensoren mit analogem Ausgangssignal (z.B.:Temperatursensor) einzulesen und zu verarbeiten. Mit Hilfe analoger Ausgänge konnten Aktoren angesteuert werden (Bsp.: Drehzahl eines Motors). Wie aus Abbildung 1 ersichtlich, ist der Verdrahtungsaufwand direkt proportional zur Anzahl der Sensoren und Aktoren. Die Übertragung von analogen Signalen über lange Leitungen in elektromagnetisch belasteter Umgebung stellt ein Problem dar. Es kann zur Induktion von Strömen kommen welche das Messsignal überlagern und verfälschen.

In modernen Prozessleitsystemen werden die einzelnen Aufgaben der Prozessführung auf hardwaremäßig getrennte Geräte aufgeteilt. Die Kommunikation zwischen diesen Geräten erfolgt über ein **seriell** arbeitendes Bussystem. Diese Struktur ist in Abbildung 2 dargestellt.