INHALT

Kaufen Sie günstiger mit CCC-Card

- **Ing. Herbert Friedl** Roboter für PC, Seite 65
- **CM-Computer** Seiten 3, 124-125
- **Karl Krauskopf** Siehe CCC, Seite 25
- **excon** Seite 5
- **ISDN-Technik** Seiten 18-19

Neue Inserenten

- **Digital Communications** Seite 119-123
- **Fran-Com** Seite 13
- **Semrad-Elektronik** Seite 75
- **Wien Schall** Seite 61

Windows '95 Schuledition

- **Ein ganz außergewöhnliches Angebot für Schulen, Lehrer und Clubmitglieder für den Umstieg auf Windows '95. Siehe Seite 16 und Bestellkarte.**

Inhalt

LIESMICH			
CCC-Card, Neue Inserenten	1		
Windows '95 Schuledition	1, 16		
Inhalt	1		
Autoren	4,6		
Paragrafen fürs Internet?	7	*Martin Weissenböck*	
Liebe Leser!	10	*Franz Fiala, Werner Krause*	
Berichte über uns	12		
Gewinnspiele	12		
Vorschau	14		
Nachträge	14		
PCNEWS-Disketten, CD	16		PCN-DSK-520..535
Schuledition Windows '95	16		⮜ Bestellmöglichkeit
Bestellkarten	16a-d		⮜ Bestellmöglichkeit
Impressum	128		
ADIM WIEN GRAZ · Computer Communications Club · HYPERBOX · MCCA · PCC-S · PCCTGM			
Termine-PCNEWS	17		
Termine-PCNEWS-Familie	20		
Liebe Freunde des MCCA	22	*Josef Sabor*	
Neues von der ADIM	21	*Martin Weissenböck*	
Liebe Mitglieder des CCC!	23	*Werner Illsinger*	
Liebe Freunde der HYPERBOX	22	*Martin Reinsprecht*	
Liebe Mitglieder des PCCTGM	21	*Robert Syrovatka*	
Schule			
Die Vertreibung der Frauen aus der Wissenschafts- und Technikgeschichte	26	*Margarete Maurer*	
Spezielle Technik für Berufstätige	30	*A. Weichselberger*	
Ausgewählte Projekte	30	*Norbert Bartos*	
Computer- und Systemtechnik	31	*Norbert Bartos*	
EUREKA Multimedia Umbrella	32	*OCG*	⮜ **Teilnahme erwünscht**
Gewinner des Mathematik-Wettbewerbs	32	*Danja Stiegler*	
Österreichische Schulen: Ihre WWW-Seiten und Provider	48	*Franz Fiala*	
PCNEWS			
Lernsoftware für die Kleinsten	35	*Bärbel Pöch*	Programmbeschreibung
Der kleine Tiger braucht ein Fahrrad	36	*Markus Dietinger*	Programmbeschreibung ❷
Neandertal	36	*Katharina Simon*	Programmbeschreibung ❸❹
Paulie Python	36	*Florian Simon*	Programmbeschreibung ❸❹
Valley	36	*Stefan Simon*	Programmbeschreibung ❸❹
Solitär	36	*Gerd Simon*	Programmbeschreibung ❸❹
Die Siedler2 (Veni, vidi, vici)	37	*Ali Eghdamian*	Programmbeschreibung ❶
Die Fugger2	38	*Ali Eghdamian*	Programmbeschreibung ❶
Command & Conquer	40	*Florian Eichelberger*	Programmbeschreibung ❺

INHALT

Netze			
Windows NT im Netzwerk	41	*Fritz Schmöllebeck*	🕮
SystemPerformace Tuning	41	*Fritz Schmöllebeck*	🕮
Internet-Server, Einrichten und Verwalten	42	*Peter Balog*	🕮
Internet Für Jedermann	42	*Werner Cyrmon*	🕮
HTML	43	*Gerwald Oberleitner*	
Gut Ding braucht Weile!	43	*Fritz Eller*	🕮
Internet Firewalls	44	*Hans Blocher*	🕮
Freie Bilder	46		
Highway 194 - Eine Sackgasse?	46	*Werner Illsinger*	
Von PAN/BTX zum Internet	47	*Walter Riemer*	
Österreichische Schulen: Ihre WWW-Seiten und Provider	48	*Franz Fiala*	
Ärger mit den Laufwerksbuchstaben	54	*Sepp Melchart*	
Programmierung			
MSDN	53	*Franz Fiala*	⮜ Schulpreise
Rekursion	55	*Karel Štípek*	PCN-DSK-520:\rekurs
CDROM „Delphi Programmiertools“	56	*Hans Lohninger*	🕮
Laut und leise am PC-Lautsprecher	57	*David Solonyna*	PCN-DSK-520:\toene
Embedded Systems			
Tools für Embedded Systems	16	*NOWATRON*	PCN-DSK-521..535
Was sind Embedded systems?	59	*Peter Balog*	
electronic only	60	*Peter Ullrich*	⮜ low-cost Elektronik-Programme
Handbuch des 80C166	67	*Christoph Baumgartner, Martin Weissenböck*	🕮
Mikroprozessortechnik mit LOCAD	68	*Werner Holler*	Programmbeschreibung
ANALOGSIMULATION MIT PSPICE	69	*Klaus P. Eckl*	PCN-DSK 139, 149; PCN-LIT 12
Analoge Integration	74	*Othmar Fischer und Markus Seidl*	
KLEIN, aber PICfein	76	*Robert Schwager*	PCN-DSK-520:\pic
Mikrocontrollerboard für 80C537 oder 80C517A	86	*Hermann Schönbauer*	PCN-DSK-520:\uc537 ⮜ Nachbaumöglichkeit
Neues von KEIL	91	*Hermann Sailer*	Markt
Eingebettetes MS-DOS und Windows	91	*Franz Fiala*	Markt
Die richtige Wahl	92	*Andreas Pfeiffer*	
Mehr als nur ein Cross-Debugger	95	*Andreas Pfeiffer*	
Informatik+Mikroelektronik = binäre Bäume+Mikrocontroller	99	*Wilhelm Brezovits*	PCN-DSK-520:\baeume
UNISTEB	110	*Stefan Huber, Ondrej Gavura*	Betreuer: Robert Seufert
HYBRIDES NEURONALES NETZ	112	*Kurt Bohuslav, Thomas Hinterstoisser, Harald Steinmetz, Gregor Strnad*	Betreuer: Norbert Bartos
Audiomischpult	115	*Oliver Singer*	Betreuer: Gottfried Patels
BODENKONTAKTSOHLE	116	*Günther Gmeindl, Norbert Winkler*	Betreuer: Christian Kollmitzer
FREISPRECHANLAGE	117	*Christian Hofer, Stefan Lampert*	Betreuer: Gottfried Patels
Datenflußrechner	118	*Aleksandar Radovanovic, Theobald Seyffertitz*	Betreuer: Norbert Bartos
Beihefter SIEMENS	**96/97**		
CAN-Bus bringt Bordnetz auf Hochtouren		*Thomas Kattwinkel*	
Systemkosten-Controller			
Aufbruch bei Mikrocontrollern		*Burghardt Schallenberger, Anton Müller*	
C166/C500 Family			
Lustiges			
Super DAU	14		
Glühbirne kaputt	16		
Wie ruiniert man einen Betrieb	22		
Sofwareentwickler hat eine Panne	29		
The Difference Between Intel Chips	54		
Wie die verschiedenen Programmierer ihre Fahrräder bauen	56		
Wenn Betriebssysteme eine Fluggesellschaft betrieben	66		
Real programmers don't	90		
Computer-Sprache	96		
Real computer scientists don't	111		
Mathematiker, Informatiker und Physiker in Irland	114		
Real software engineers don't	115		

Autoren

Balog Peter Dipl.-Ing.

Lehrer für Nachrichtentechnik und Elektronik, Kustos für Novell-Net und Internet

Schule TGM-N
Club PCC-TGM
Absolvent TGM, TU-Wien
Werdegang TU-Assistent, TGM-L
InterNet balog@ping.at

Bartos Norbert Dipl.-Ing. Jg.1954

Professor für Computer- und Systemtechnik, Leiter der Speziallehrgänge für Elektronik

Schule TGM-N, FhE-Wien 20
Club PCC-TGM
InterNet bartos@email.tgm.ac.at

Baumgartner Christoph Jg.1974

Student der Mechatronik

Schule Johannes Kepler Uni Linz
Club PCC-TGM
Absolvent HTL Braunau, Elektrotechnik
Interessen Computer, Hardware, Elektronik, Videofilmen, uC
Hobbies Wasserski, Tischtennis, Radfahren
InterNet Christoph.Baumgartner@jk.uni-linz.ac.at

Blocher Hans Mag. Jg.1956

Lehrer für EDV, Netzwerkbetreuer, Leiter der ARGE-EDV an HTL/OÖ

Schule HTBLA Braunau
Club CCC
InterNet 74074.2763@compuserve.com
CIS 74074,2763

Bohuslav Kurt

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Brezovits Wilhelm Ing. Jg.1968

Produktspezialist für Mikrocontroller und UNIX-Administrator

Firma Siemens AG
Absolvent HTL-Mödling, E5b, 1987
Interessen C, C++ und µC-C/C++
Privates Verheiratet, 3 Kinder
InterNet Wilhelm.Brezovits@siemens.at

Cyrmon Werner Mag. Jg.190

Lehrer für Informatik

Schule HTBLVA Wiener Neustadt
InterNet Cyrmon@msn.com
FidoNet 2:313/20.99

Dietinger Markus Jg.1965

Kundendiensttechniker für Nebenstellenanlagen

Firma Philips-Telekom

Eckl Klaus Peter Dipl.-Ing. Jg.1943

Lehrer für Nachrichtentechnik und Elektronik im TGM, Kustos für das Labor Nachrichtentechnik

Schule TGM-N
Club PCC-TGM
InterNet eckl@email.tgm.ac.at

Eghdamian Ali Jg.1980

Schüler der 7. Klasse AHS Realgymnasium

Schule GRG Wien 23
Club CCC
Interessen moderne Kunst, Streitgespräche über Hermann Nitsch
Hobbies Billardspielen, Computerspiele, 3D Game Creation

Eichelberger Florian

Schüler

Schule Kollegium Kalksburg
InterNet florian.eichelberger@hmv.ccc.or.at

Eller Fritz Mag. Jg.1962

Lehrer für Informatik, Deutsch und Geschichte, Lehrbeauftragter des WIFI

Schule Höhere Bundeslehranstalt für Tourismus St. Johann in Tirol
InterNet eller@netwing.at

Fiala Franz Dipl.-Ing. Jg.1948

Lehrer für Nachrichtentechnik und Elektronik, Leitung der Redaktion und des Verlags der PCNEWSedu

Schule TGM-N
Club CCC MCCA PCC-TGM
Absolvent TU-Wien, Nachrichtentechnik
Werdegang BFPZ-Arsenal
Privates verheiratet, 1 Kind
Hobbies Schwimmen
InterNet franz.fiala@atnet.at
FidoNet 2:310/1.36
SchulNetz 22:102/1.136
CIS 100024,1325

Fischer Othmar Dipl.-Ing.Ing. Jg.1932

Emeritierter Professor für Nachrichtentechnik und Elektronik, Technisch-wissenschaftlicher Berater, Schulbuchautor

Firma TGM-N

Gavura Ondrej jun.

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Gmeindl Günther

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Hinterstoisser Thomas

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Hofer Christian

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Holler Werner Mag. Jg.1957

Lehrer für Mathematik und Informatik

Schule BG & BRG Leibnitz
Club CCC
InterNet wholler@borg-graz.ac.at
FidoNet 2:316/88.148
SchulNetz 22:100/1.148

Huber Stefan

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Illsinger Werner Ing. Jg.1968

Sysop der Mailbox His Master's Voice, Präsident des CCC

Firma EDVg debis
Club CCC
Absolvent TGM-N87D
InterNet illsin@ccc.or.at
FidoNet 2:310/1.0
SchulNetz 22:102/1.0
WWW http://www.ccc.or.at/ccc/mitgl/illsin.html

Kollmitzer Christian Dipl.-Ing.

Lehrer für Nachrichtentechnik und Elektronik, Stammprofessor in der Fachhochschule für Elektronik

Schule TGM, FhE Wien 20

Krause Werner Mag. Jg.1955

Lehrer für Bildnerische Erziehung

Schule GRG Wien 23
Interessen Grafik: CorelDraw, PhotoShop, Picture Publisher

Lampert Stephan

Absolvent des Speziallehrgangs für Mikroelektronik

Absolvent TGM-SLME-96

Lohninger Hans Dr.

Schule TU-Wien, Inst.f. allgemeine Chemie
Club PCC-TGM
InterNet hlohning@email.tuwien.ac.at
WWW http://echm10.tuwien.ac.at/inst/lo/lo.html

Maurer Margarete Dr.

Hochschullehrerin, Universitätslektorin für Wissenschaftstheorie der Biologie und Publizistin

Schule Uni-Wien, Lektorin am Institut für Humanbiologie sowie RLI
Werdegang 1991-92 Gastprofessorin an der Gh Universität Kassel
InterNet margarete.maurer@univie.ac.at
WWW http://www.iguwnext.tuwien.ac.at

AUTOREN

Melchart Josef Dipl.-Ing. Jg.1957

Lehrer für Nachrichtentechnik und Elektronik, Lehrbeauftragter an der Fh
Schule TGM-N, FhE Wien 20
Club PCC-TGM
InterNet melchart@email.tgm.ac.at

Oberleitner Gerwald Jg.1976

Student der Wirtschaftsinformatik
Schule Uni-Wien
Club CCC
Absolvent BORG Wien 22
InterNet oberleit@ccc.or.at
FidoNet 2:310/1.63
WWW http://www.ccc.or.at/ccc/mitgl/oberleit.html

Patels Gottfried Dipl.-Ing.

Lehrer für Nachrichtentechnik und Elektronik
Schule TGM-N

Pfeiffer Andreas Dipl.-Ing. Jg.1966

European Marketing Manager
Firma Take Five Software, Salzburg
Absolvent HTBLA Braunau/EL&NT, TU-Wien Industrielle El.
Werdegang Entwicklertätigkeit in der Luftfahrtindustrie, Marketing-Engineer bei Microtec Research
InterNet pfeiffer@taekfive.co.at

Pöch Bärbel Mag.

Psychologin, Ansprechpartnerin für den Bereich "Selected Games" bei Pablitos
Firma Pablitos
InterNet office@pablitos.co.at

Radovanovic Aleksandar

Absolvent des Speziallehrgangs für Mikroelektronik
Absolvent TGM-SLME-96

Reinsprecht Martin Dipl.-Ing.Dr. Jg.1964

Lehrer für technische und mathematische Fächer, Vorsitzender der Hyperbox
Schule HTL-Leonding
Club HYPERBOX
Interessen Computer, Grenzwissenschaften, Philosophie
InterNet martin.reinsprecht@hyperbox.org
FidoNet 2:314/50
WWW http://www.hyperbox.org/

Riemer Walter Dipl.-Ing. Jg.1940

Lehrer für Informatik, Leiter des Rechenzentrums der Abteilung, Autor mehrerer Lehrbücher für den Unterricht, Ingenieurkonsulent für Elektrotechnik
Schule TGM-EN/NA/BW
Club PCC-TGM
InterNet riemer@email.tgm.ac.at

Sabor Josef Ing. Jg.1950

Leiter der Anbietergruppe des MCCA; Beamter; Inhaber des PAN Advertising Service Sabor
Firma Wiener Linien
Club MCCA PCC-TGM
InterNet joesabor@mcca.ping.at
FidoNet 2:313/9.31
CIS 100342,2312

Sailer Hermann Ing. Jg.1959

Verkaufsleiter Meßtechnik bei Firma Walter Rekirsch
Firma Rekirsch Elektronik
InterNet 100565.3231@compuserve.com
CIS 100565,3231

Schmöllebeck Friedrich Dipl.-Ing.Dr.

Lehrer für Nachrichtentechnik und Elektronik
Schule TGM-N, FhE Wien 20
Club PCC-TGM
Absolvent TU, Nachrichtentechnik
InterNet fschmoe@pan.at

Schönbauer Hermann Jg.1943

Ausbilder für Nachrichtenelektronikerlehlinge
Firma Siemens AG Österreich, Bregenz
Interessen MC-Steuerungen, Pädagogik
Hobbies Höhlenforschen, Orchideen
InterNet hermann.schoenbauer@siemens.at

Schwager Robert Jg.1972

Student für Elektro-(Computer)technik
Schule TU-Wien
Club PCC-TGM
Absolvent HTL-Elektrotechnik
Interessen Mikroelektronik (FPGAs, GALs, uC)
Hobbies Entwickeln elektronischer Schaltungen

Seidl Markus Dipl.-Ing. Jg.1968

Software-Entwickler
Firma PDTS-Wien
Club PCC-TGM
Absolvent TGM-N

Seufert Robert Dipl.-Ing.

Lehrer für Nachrichtentechnik und Elektronik
Firma TGM-N
Club PCC-TGM

Seyffertitz Theobald

Absolvent des Speziallehrgangs für Mikroelektronik
Absolvent TGM-SLME-96

Simon Gerd Dipl.-Ing.

Schule HTL-Graz-Gösting
InterNet gsimon@gewi.kfunigraz.ac.at
FidoNet 316/3.12
SchulNetz 22:100/1.17

und

Stefan (8 Jahre),
Florian (10 Jahre),
Katharina (12 Jahre)
die jüngsten Autoren der PCNEWSedu

Singer Oliver

Absolvent des Speziallehrgangs für Mikroelektronik
Absolvent TGM-SLME-96

Solonyna David Jg.1978

Schüler der Abteilung Nachrichtentechnik
Schule TGM-N
Absolvent N98A
InterNet sol@email.tgm.ac.at

Steinmetz Harald Jg.1974

Assistent an der Höheren Lehranstalt für Elektronik, Vortragender bei PCC-TGM Seminaren
Firma TGM-N, BA
Club PCC-TGM
Absolvent TGM-N94b, TGM-SLME-96
InterNet steinmh@email.tgm.ac.at

Stiegler Danja

Ansprechpartnerin für den Bereich Wissenschaft und Technik bei Pablitos
Firma Pablitos
InterNet office@pablitos.co.at

Štípek Karel Dipl.-Ing. Jg.1953

Programmierer für Pascal, Clipper, Excel
Firma Metropolitan
Absolvent CVUT Praha, Starkstromtechnik
Interessen Elektronik
Privates 2 Kinder
Hobbies Wandern
InterNet 106076.701@compuserve.com
CIS 106076,701

Strnad Gregor

Absolvent des Speziallehrgangs für Mikroelektronik
Absolvent TGM-SLME-96

Syrovatka Robert Ing.

Lehrer für Nachrichtentechnik und Elektronik
Schule TGM-N
Club PCC-TGM
InterNet syro@email.tgm.ac.at

Ullrich Peter Ing. Jg.1967

Club PCC-TGM
Absolvent TGM-N86B,0SL87
InterNet Peter.Ullrich@kapsch.co.at

Weichselberger A. Dir.

Direktor der HTL Wien 1
Schule HTL Wien 1

Weissenböck Martin Dir.Dr. Jg.1950

Direktor der HTL Wien 4, Leiter der ADIM und Autor von ADIM-Skripten, Leiter der ARGE Telekommunikation
Schule HTL Wien 4, ADIM
Club ADIM CCC PCC-TGM
InterNet mweissen@ping.at
FidoNet 2:310/1.35
SchulNetz 22:102/1.135
CIS 100016,172

Winkler Norbert

Absolvent des Speziallehrgangs für Mikroelektronik
Absolvent TGM-SLME-96

Vorbemerkung: Ich habe diesen Beitrag versuchsweise nach den Regeln der „neuen Rechtschreibung“ überarbeitet. Das Mühevollste waren die neuen Regeln für das „ß“. Es ist mir völlig unverständlich, warum nicht die Schweizer Version gewählt wurde, die jedes „ß“ durch „ss“ ersetzt. Nicht nur die EDV-Spezialisten hätten ihre Freude daran!

Daher in Abwandlung eines bekanntes Satzes: stellt euch vor, es ist Rechtschreibreform, und keiner hält sich daran - was wäre, wenn alle das „ß“ ignorierten? Wie das aussieht, können Sie hier lesen.

Alle Buchstaben, die durch die Rechtschreibreform eine Änderung erfahren, sind unterstrichen.

„ss“ *Derzeit würde man diese Wörter mit „ß“ schreiben. Nach der Rechtschreibreform wird man sie mit „ss“ schreiben können.*

„ss“ *Diese Wörter wird man auch nach der Rechtschreibreform mit „ß“ schreiben müssen.*

Paragrafen fürs Internet?

Martin Weissenböck

Pressestimmen

Rechtsradikale Propaganda, Kinderpornos, Anleitungen zum Bombenbasteln - all diese Schlagzeilen hat das Internet in letzter Zeit gemacht. Zeitungsberichten zufolge sollen negative Auswirkungen des Internet durch rechtliche Vorschriften bekämpft werden. Vor ein paar Wochen[1]wurden Ideen dazu von der Bundesgeschäftsführerin der SPÖ, Brigitte Ederer, im Kurier präzisiert:

- Gewünscht wird eine klare urheberrechtliche Regelung.
- Medieninhaber soll derjenige sein, der die Informationen gestaltet.
- Jede Internetseite soll mit einem Impressum gekennzeichnet sein.
- Die medienrechtliche Zuständigkeit der Gerichte soll am Sitz des Medieninhabers (im Impressum ersichtlich) angesiedelt sein. Nur so ist es möglich, auch jene zu belangen, die vom Ausland aus Meldungen verbreiten.

Soweit ein kurzes Zitat aus der Zeitung. Ich meine, dass einige Gedanken dazu angebracht sind...

Die alte Zeitungsweisheit - „Only bad news are good news“ - bestätigt sich auch beim Thema Internet. Das renommierte Time-Magazin titelt die Ausgabe vom 3. Juli 1995 mit *„Cyberporn - A new study shows how pervasive and wild it really is. Can we protect our kids - and free speech?“* und deutet damit schon den entstehenden Widerspruch zwischen Zensur und freier Meinungsäusserung an. Allerdings sind das Time-Magazin und alle auf dem Artikel aufbauenden Synoptiker einem suspekten Datenmaterial auf den Leim gegangen. Die Zeitschrift Focus[2] schreibt dazu:

> Die „Studie der Carnegie Mellon University“, auf die sich „Time“ und die politischen Zensurbefürworter berufen, entpuppt sich als die unausgegorene Arbeit eines Elektrotechnikstudenten von zweifelhaftem Ruf: Marty Rimm. Die als Co-Autoren genannten Professoren können sich nicht erinnern, die Endfassung des Papiers gesehen zu haben, einer bestreitet gar jede Mitarbeit.
>
> Die Datensammlung von Rimm verquirlt Statistiken von privaten Porno-Mailboxen mit Benutzerzahlen der Internet-Diskussionsforen, sogenannten Newsgroups, und kommt so zu seinen haarsträubenden Schlüssen: Von 917 410 untersuchten Dateien seien 83,5 Prozent pornografisch. Doch die Rechnungen von Donna Hoffman und Thomas Novak von der Vanderbilt University zeigen ein ganz anderes Bild: Im Untersuchungszeitraum liefen nicht einmal 3000 digitalisierte Pornos durch die einschlägigen Internet-Gruppen. Gemessen am Nachrichtenaufkommen aller Gruppen nicht einmal ein halbes Prozent.
>
> Mit diesen Darstellungen werden zudem nicht etwa schutz- und ahnungslose Benutzer konfrontiert: wer Foren mit eindeutigen Namen wie „alt.sex“ anwählt, tut dies selten aus Versehen. Im Übrigen kann das archaische Datenformat der Newsgroups Bilder nur als codierten Buchstabensalat übertragen. Wer sich daran ergötzen will, muss sie mit umständlichen Hilfsprogrammen in Pixel zurückverwandeln.
>
> Im populären grafikorientierten World-Wide-Web muss man die Pornografie gar mit der Lupe suchen: Unter den 11 576 Web-Seiten, die Rimm im Dezember letzten Jahres untersuchte, listet seine Studie nur neun seiner Meinung nach anstössige (0,08 Prozent) auf.
>
> Einige der Daten sind nicht nachprüfbar. So gibt es technisch keinerlei Möglichkeit, die Benutzerprofile von Internet-Usern beim Datenverkehr zu registrieren. Unter Beschuss gerät Rimm inzwischen auch an der Carnegie Melon University: Für seine Studien hat er heimlich die Nutzungsstatistiken von Arbeitsplatzrechnern angezapft.
>
> Obwohl sich Rimm über seinen Lebenslauf ausschweigt, hat die zornige Internet-Gemeinde inzwischen einige schmutzige Geheimnisse des selbsternannten Enthüllers aufgedeckt: So trat er bereits als Autor eines Handbuches für Porno-Mailbox-Betreiber in Erscheinung. Titel: *„Wie man Frauen ausbeutet, Männer reinlegt und jede Menge Geld verdient“.*

Soweit die Zeitschrift Focus. Es scheint auch bei unseren Presseberichten sinnvoll zu sein, den Anteil der unerwünschten Botschaften im Netz einmal zu hinterfragen. Interessant wäre eine Gegenüberstellung der Verbreitung (und Förderung?) von Presseerzeugnissen, wie „Kontaktmagazin“, „Deutsche Militär- und Soldatenzeitung“, „Tatblatt“ zum restlichen Zeitungsmarkt auf der einen Seite und von kritisierten Internet-Angeboten zum Gesamt-Internetangebot auf der anderen Seite.

Wir sollten uns des schmalen Grates zwischen schutzwürdigen Interessen und staatlicher Bevormundung, zwischen Zensur und freier Meinungsäusserung bewusst sein. Ferner müssen derartige Diskussionen von einzelnen Anlässen gelöst werden. Als Lehrer ist uns eine solche „Anlassgesetzgebung“ ja nicht fremd. Beispiel: da an *einer Schule* die Abrechnung einer Schulsportveranstaltung nicht korrekt durchgeführt worden ist, wurde *alle Schulen* per Erlass zu Massnahmen („Abrechnung über das Schulkonto“) aufgefordert, die mit erheblichem Verwaltungsmehraufwand verbunden sind, daher viel mehr Arbeitszeit und damit Geld kosten, realitätsfern und in einem Punkt sogar undurchführbar sind. Wohin eine Anlassgesetzgebung führt, wurde dem staunenden Staatsbürger vor kurzem vorgeführt: um das beliebte *Höchteln* abzustellen, bekommen Abgeordnete jetzt sogar eine Art vor Fahrzeitentschädigung - wo darf ich mich melden, um auch für täglich zweimal 40 Minuten U-Bahn- und Strassenbahnfahrt einen Obolus zu bekommen?

Analyse

Damit soll das Problem natürlich nicht verniedlicht werden. Ich schlage vielmehr eine Analyse der erstellten Vorschläge vor.

Wohltuend fällt - im Gegensatz zu früheren Diskussionen bei uns, vor allem aber in Deutschland[3] - auf, dass die Netzwerk-Betreiber für die verbreiteten Meldungen nicht haftbar gemacht werden können. Zitat: *„Die Internet-Anbieter für den Inhalt verantwortlich zu machen wäre so, als würde man die Post für Inhalte der Telefongespräche verantwortlich machen“*, meint Ederer.

Ich behaupte, dass jede (rechtliche) Norm nur dann einen Sinn hat, wenn zumindest theoretisch die Möglichkeit besteht, die Einhaltung dieser Norm zu überwachen. Im elektronischen Universum hat es die Polizei schwer...

Impressum im Internet

„Jede Internetseite soll mit einem Impressum gekennzeichnet sein“. Aber was ist eine Internetseite? Zu BTX- oder PAN-Zeiten war die heile Welt angesagt: jede Seite ist 40 mal 24 Zeichen gross und hat eine Nummer. Die hierarchische Nummerierung sichert, dass von einer Leitseite ausgehend auch nachfolgende Seiten *einem* Besitzer zugeordnet werden können. Hyperlinks gibt es nicht und die Hardware - von der

[1] Kurier vom 23. Juli 1996, Seite 3.

[2] Heft 30/1995, p.108

[3] Die Firma CompuServe wurde in Deutschland für pornografische Internetseiten verantwortlich gemacht

Post bereitgestellt - steht in Österreich. Verträge werden mit der Post abgeschlossen und alles ist schön unter Kontrolle[4].

Noch einmal: was ist eine Internetseite? Je eine HTML[5]-Datei? Soll dann auf jeder ladbaren Seite das Impressum stehen? Und was ist mit Grafikseiten? Soll bei gescannten Bildern das Impressum pixelweise dazueditiert werden? Ferner wird die multimediale Landschaft auch - sobald es die Bandbreite erlaubt - Videosequenzen anbieten. Ein Impressum pro Kader oder doch nur am Anfang? Ich weiss auch nicht, ob der Badenweiler-Marsch[6] oder das Horst-Wessel-Lied[7] gegen das Wiederbetätigungsgesetz verstossen; was, wenn nun diese Melodien in MIDI[8]-Form als File übertragen werden?

Trotzdem meine ich, dass es emsigen Gesetzesmachern gelingen wird, eine schöne Definition für eine Internet-Seite zu produzieren. Legistische Kreativität können wir ja derzeit bei der *Sozialversicherungspflicht für Werkverträge* bewundern: es gibt nun *freie Dienstverträge* und *dienstnehmerähnliche Werkverträge*. Zwar weiss niemand, wie es gemeint ist, das Gesetz ist auch rasch novelliert[9] worden. Das Bundesgesetzblatt ist dafür wieder um ein paar Seiten reicher geworden.

Und was macht das Ausland?

Das Problem liegt sicher wo anders: *„Die medienrechtliche Zuständigkeit der Gerichte soll am Sitz des Medieninhabers (im Impressum ersichtlich) angesiedelt sein. Nur so ist es möglich, auch jene zu belangen, die vom Ausland aus Meldungen verbreiten."* Bei allem Respekt: das verstehe ich nicht. Damit müssten alle Seitengestalter - auf der *ganzen Welt* - entsprechend den österreichischen Wünschen sich mit einem Impressum vorstellen. Hofft Frau Ederer hier darauf, dass die gesamte Welt - ausnahmslos - ähnliche Regelungen beschliesst? Das kann wohl nicht der Fall sein. Die Regierungen schaffen es nicht einmal bei Problemen, die für das Überleben einzelner Völker oder der gesamten Menschheit von wesentlicher Bedeutung sind, zu einem gemeinsamen Vorgehen zu kommen. Beispiele gefällig: Verhindern des Völkermordes in Ex-Jugoslawien (Handke zum Trotz); Ende der Stammeskriege in Afrika; Beschränkung der CO_2- oder FCKW[10]-Produktion. Da hat ein UN-Impressumsbeschluss wohl wenig Chancen!

Wer also einen Server mit in Österreich verbotenen Inhalten betreiben will, stellt ihn demnach irgendwo - in einem impressumsfreien Land - auf; via World Wide Web hat dann jeder darauf Zugriff.

Wer kontrolliert die IP-Adressen?

Was schreibt Dürrenmatt in seinen *Physikern*:[11] *„Alles Denkbare wird einmal gedacht!"* Spinnen wir daher den Gedanken weiter: vielleicht könnte man den Datentransfer nach Österreich „an der Grenze" kontrollieren und alle Aufrufe von IP-Adressen[12], die als verboten bekannt sind, sperren.

- Dazu müsste man wissen, welche IP-Adressen denn verbotene Inhalte vermitteln. Vielleicht über Bundes-IP-Adressen-Überwachungsamt? Aber Spass beiseite - allein die Erhebung wäre unmöglich.
- Die Listen müssten ausserdem ständig auf den letzten Stand gehalten werden - täglich kommen neue Server dazu und andere Server werden wieder abgeschaltet.
- Da Verbotenes besonders interessant ist, würden findige Österreicher vielleicht über Telefonwählleitungen oder gar über Funk mit dem Ausland kommunizieren und sich dort einen Service-Provider (ohne Zugangsbeschränkungen) suchen. Alle Telefonleitungen zu überwachen ist auch 12 Jahre nach Orwells *Grossem Bruder* nicht recht vorstellbar.

[4] Auch nicht alles; auch im BTX gab es zweideutig-eindeutige Angebote; in deutschen System noch wesentlich mehr als im österreichischen System.

[5] Hyper Text Makup Language - Beschreibungssprache für WWW-Seiten

[6] Lieblingsmarsch von Adolf Hitler

[7] Marschlied der SA, im Dritten Reich eine Art von „zweiter Nationalhymne"

[8] MIDI=Mucical Instrument Digital Interface, eine Norm für die Datenübertragung zwischen elektronischen Musikinstrumenten und Computern

[9] 53. ASVG-Novelle, beschlossen am 10. Juli 1996

[10] Flourierte Chlor-Kohlen-Wasserstoff-Verbindungen, hauptverantwortlich u.a. für das Ozon-Loch

[11] Friedrich Dürrenmatt: Die Physiker, 1962

[12] Für Uneingeweihte: die IP-Adresse ist die weltweit eindeutige Nummer des Rechners, vergleichbar mit einer Telefonnummer.

Index und Zensur

Ab 1559 hat die katholische Kirche mit ihrem Index librorum prohibitorum[13] versucht, unliebsames Schriftgut von Gläubigen fernzuhalten. Da die Sache im 20. Jahrhundert nicht mehr sehr erfolgreich war, wurde die Idee fallen gelassen[14]. Ich bin überrascht, dass gerade von sozialdemokratischer Seite her diese Idee auf elektronischer Basis reanimiert werden soll.

Wer bestimmt aber, was verboten ist? Hier rückt die gesamte Diskussion, die sich bisher mehr mit technischen Problemen beschäftigt hat, in die gefährliche Nähe der Zensur. Die freie Rede, der freie Meinungsaustausch - auch über elektronische Medien - gehört zu den unverzichtbaren Eckpfeilern jeder modernen Demokratie. Vergessen wir nicht, dass das unblutige Ende des Kommunismus in Osteuropa auch durch einen internationalen Informationsaustausch beschleunigt worden ist. Totalitäre Regime werden versuchen, den internationalen Gedankenaustausch zu unterbinden. Ein passendes „Ermächtigungsgesetz", das in demokratischen Zeiten zum Schutz und Wohl des Volkes vor Schmutz und Schund beschlossen worden ist, gäbe dann den pseudolegalen Rahmen dafür.

Was tun?

Die Analyse der Pläne, das Internet zu verrechtlichen, führt zu wenig ermutigenden Ergebnissen: die Massnahmen sind undurchführbar, unsinnig, undemokratisch - jeder kann es sich aussuchen. Ich meine daher, dass bereits der *Ansatz* falsch ist.

Der Ruf nach einem neuen Gesetz ist ziemlich phantasielos und vor allem nur eine *defensive* Strategie. Wer im Umgang mit den neuen Medien unliebsame Auswüchse verhindern will, ist gut beraten, über *offensive* Strategie nachzudenken. Hier ein paar Anregungen:

- Zu den *ewiggestrigen* Themen: Errichten eines WWW-Servers, der die Ergebnisse der Geschichtsforschung in einer ansprechenden Weise aufbereitet. Hyper-Links zu jenen Servern, die ihre Sicht darstellen. Das Ziel muss sein, sie nicht totzuschweigen, sondern sich mit ihnen aktiv auseinander zu setzen.
- Teams: Zusammenstellen eines kleinen Teams (Zeitgeschichtler, Lehrer, Psychologe, Internet-Guru) zur Gestaltung von WWW-Seiten.
- Zugang zu Jugendlichen: Besondere Betreuung, Aufbereitung der Inhalte nicht in „Schulfunk"-Manier, sondern eher nach dem Muster von CNN.
- Wettbewerb für Schulen ausschreiben: mehrere Schulen gestalten ihre Sicht der Zeitgeschichte im WWW. Die besten Darstellungen werden prämiert, die WWW-Seiten über Medien bekanntgemacht.
- Betroffenheit erzeugen: Berichte über Bombenopfer sammeln und im WWW aufbereiten.
- Ebenso: Berichte über die Opfer von Kinderpornografie sammeln und im WWW aufbereiten.

Eine kritische Auseinandersetzung ist notwendig, auch um das Internet und seine Nutzung vor ungerechtfertigten Pauschalangriffen zu schützen. Neue Paragrafen lösen selten Probleme, neue Ideen schon eher. Wer macht mit? □

[13] lat.: Verzeichnis der verbotenen Bücher

[14] Der Index librorum prohibitorum ist mit Wirkung vom 29. März 1967 ausser Kraft gesetzt worden.

Liebe Leser!

Einleitend konnten Sie eine Stellungnahme zu der drohenden Reglementierung im Internet lesen, eine Neuheit im Aufbau der PCNEWSedu, ein Editorial des Herausgebers.

Wir können nicht versprechen, in jeder Ausgabe Meinungen zu einem aktuellen Thema abdrucken zu können, aber **wir laden hiermit unsere Leser - und besonders die Autoren - ein, Diktafon, Bleistift oder Tastatur für die anderen Mitleser zu ergreifen und abseits vom eigentlichen technischen Inhalt der PCNEWSedu Themen aufzugreifen, die unser Leben mit der EDV betreffen und über das Sie schon immer berichten wollten.**

Wir gestalten eine Welt mit EDV, die in ihrer Eigenart unser Denken und Handeln schon so stark beeinflußt, daß wir als die zuerst Betroffenen auch die Spielregeln mitgestalten oder zumindest mitdiskutieren sollten.

Embedded Systems

Einmal im Jahr gibt es für die große Zahl der Elektronik-Interessierten eine Schwerpunktausgabe, die wir heuer mit „Embedded Systems“ benannt haben. Dieses Schlagwort hat sich in den letzten Jahren als Überbegriff für das Arbeiten an mikroelektronischen Subsystemen herausgebildet, und wir stellen in dieser Ausgabe ein weites Spektrum typischer Anwendungen für mikroelektronische Systeme vor, sodaß die Bedeutung dieses traditionellen Gebiets mit neuem Namen unterstrichen wird.

Besonders freut es mich, Ihnen ausgewählte Arbeiten der Teilnehmer des Speziallehrganges für Mikroelektronik vorstellen zu dürfen, ein Ausbildungszweig des TGM, der unter der Leitung von Norbert Bartos zu einem beachtenswerten Niveau fand. Einerseits dient der Speziallehrgang Mikroelektronik seit 15 Jahren als projektorientiertes Unterrichtsgebiet, in dem sowohl Schüler erste Projekte fertigten, auch Lehrer Unterrichtsmethoden erarbeiteten und in den Unterricht an der Tages- und Abendschule einfließen ließen.

Unter dem Eindruck der allgemeinen Sparmaßnahmen und des Konkurrenten Fachhochschule im eigenen Haus wird dieser Speziallehrgang ab dem nächsten Jahr unter stark veränderten Bedingungen geführt werden.

Embedded Systems

Die Prägnanz und Verbreitung dieses Begiffs läßt kaum Platz für eine eingedeutschte Alternative wie etwa „eingebettete Systeme“. Erstaunlich ist auch seine Bandbreite, denn was schon ist nicht in einem übergeordneten System eingebettet. Werner Krause hat solche Subsysteme am Beispiel eines Cockpits am Titelbild dargestellt.

Eigentlich verstehen wir darunter aber prozessorgesteuerte Systeme, die mit dem übergeordneten System in Wechselwirkung treten können. Genauere Betrachtungen über den Begriff finden Sie im einleitenden Artikel von Peter Balog „Was sind Embedded Systems?“.

Die besondere Problematik bei der Entwicklung dieser Systeme ist die Tatsache, daß im allgemeinen Prozessor und Bauweise des Subsystems nicht mit Prozessor und Bauform des Entwicklungsrechners übereinstimmen werden. Daher stellt uns die Programmentwicklung mit diesen Cross-Compilern vor besondere Probleme. Beispiele für Prozessoren und Prozessorplatinen finden Sie in den Beiträgen von Hermann Schönbauer und Robert Schwager; Beispiele für die Programmentwicklung in den Beiträgen von Andreas Pfeiffer.

Die Entwicklung eines „Embedded System“ besteht aber nicht nur aus der Programmentwicklung selbst. Der Entwickler muß alle Verfahren kennen, die zur Herstellung mikroelektronischer Baugruppen zählen. Klaus Eckl hat für Schüler einen Arbeitsbehelf für die Benutzung von Pspice erstellt, einem Programm zur Simulation analoger Schaltkreise und auch den Lesern der PCNEWSedu im gleichnamigen Beitrag zur Verfügung gestellt.

Für die Programmentwicklung verwenden wir spezialisierte Tools, deren Bedienungskomfort in der letzten Zeit stark verbessert wurde und der sich auch schon mit vergleichbaren Tools für den PC messen kann.

Der Beitrag von Andreas Pfeiffer stellt besondere Tools von Microtec vor, von Firma NOWATRON wurde den Lesern der PCNEWSedu eine ganze Reihe von funktionsfähigen Tools zur Verfügung gestellt, die über den Diskettendienst bezogen werden können. Besteller der PCNEWS-CD-1 finden diese Tools bereits auf ihrer CD. Ein didaktisch komplett aufbereitetes Beispiel finden Sie im Beitrag von Wilhelm Brezovits.

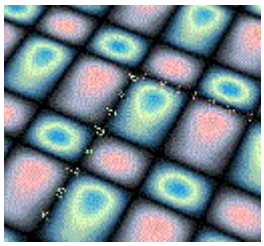
Bevor man aber ein Problem auf einem Tool für Mikrocontroller programmiert, ist es von Vorteil, die grundlegende Programmstruktur auf einem Compiler für den PC zu testen. David Solonyna zeigt, wie man über das eine Bit des PC-Lautsprechers mehr als nur eine Rechteckschwingung abgeben kann. Die Übersetzung in eine Mikrocontroller-C-Variante ist damit nicht mehr sehr schwierig.

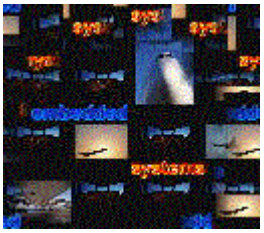
Elektronische Grundkenntnisse sind natürlich Voraussetzung für ein Gesamtverständnis. Ein Beispiel für PC-unterstütztes Messen lesen Sie im Beitrag von Othmar Fischer.

Franz Fiala

Zum Titelbild „Embedded Systems“

Die Vorgänger des Titelbildes





Der Schriftzug „EMBEDDED SYSTEMS“ wurde zunächst mit dem Textwerkzeug im Photoshop mehrmals mit unterschiedlicher Fontgröße erzeugt und auf 3 Ebenen übereinandergelegt. Den entstandenen Textmasken wurden weiche Kanten zugewiesen, um mit dem KPT-Explorer „Verläufe auf Pfaden“ behandelt zu werden. Zwischenergebnis waren 3 Ebenen mit neonröhren-ähnlichen Leuchtschriften vor dunklem Grund, die einzeln dem radialen Weichzeichner unterzogen (um den Verwischungseffekt zu erreichen) und anschließend mit der Montagefunktion „Aufhellen“ auf eine Ebene reduziert wurden.

Auf Wunsch der Redaktion sollten „EMBEDDED SYSTEMS“ durch ein anschauliches Bild unterstützt werden: die Abbildung eines Cockpits mit seinen Instrumenten (aus einer Photo-CD) bot sich an - die stark weichgezeichnet, als Ebene hinter das Schriftbild gelegt und hineinkopiert wurde.

Werner Krause

Berichte über uns

PCNEWS in Salzburg

In der Zeitschrift „htl up to date“ wird über die PCNEWSedu als Computerzeitschrift für den Unterricht und über die Möglichkeit berichtet, Probeexemplare anfordern zu können.

Hans Blocher

Ins Internet! Aber Wie?

Auf Einladung von Oskar Wagner, HTL Wien 1, habe ich mir erlaubt, die Leser der Zeitschrift „Schwerpunkt Bildung“ 3/96, die vom VdPÖ (Verband der Professoren Österreichs) an Lehrer an AHS und BHS verteilt wird, im Beitrag „Ins Internet! Aber wie?“ mit den Zugangsmöglichkeiten zum Internet bekannt zu machen. Es wurde versucht, die öffentlichen Projekte (KEM, ÖSRZ und BlackBoard) den Eigeninitiativen (ADIM, AGTK, CCC, CLUB AT, HYPERBOX, MCCA, PCC-S und PCC-TGM) gegenüberzustellen, sowie auf die PCNEWSedu als gemeinsames Sprachrohr dieser Vereine hinzuweisen.

Franz Fiala

PCNEWS in Neapel

DIE PCNEWS AUF DER AUSSTELLUNG „MULTIMEDIA IM UNTERRICHT“ IN NEAPEL

Von 20. bis 24. Mai fand in Neapel ein internationaler Kongreß zum Thema „Multimedia supports to Training“ statt. Experten aus ganz Europa diskutierten den Einsatz der neuen Medien im Bildungswesen. Gleichzeitig wurden auf einer Ausstellung multimediale Produkte für die Ausbildung sowie nationale und regionale Initiativen zu den Themenbereichen Multimedia - Netzwerke - Computer im Unterricht gezeigt. Der Österreichstand wurde von Wirtschaftsministerium, Unterrichtsministerium und Bundeswirtschaftskammer gestaltet und präsentierte das österreichische Schulnetz, unser Ausbildungssystem im allgemeinen sowie einige österreichische Multimediaprodukte. Auf Anregung von OR. Dr. Dorninger im Unterrichtsministerium wurde auch die PCNEWS als Vorzeigeprojekt der österreichischen (Schul-)Medienlandschaft präsentiert.

Wolfgang Scharl

Gewinnspiele

Media-Umfrage

Bei der Media-Umfrage sollte jeder 5-te Einsender eine PCNEWS-CD erhalten. Die Gewinner wurden auch in Ausgabe 46 bekanntgegeben. Die Versendung wird vorbereitet. Bitte die Verzögerung zu entschuldigen.

Buchgeschenke für Autoren

Das „Handbuch des 80C515 und 80C515A“ erhalten: David Solonyna, Klaus Eckl, Othmar Fischer, Schwager.

Spiele-Gewinnspiel

Die Gewinner des Spiele-Gewinnspiels von PABLITOS werden im Abschnitt PeCiNEWS vorgestellt.

Plakataktion

Gemeinsam mit der Plakataktion an allen österreichischen Schulen wurde ein Gewinnspiel angekündigt. *Ein Danke allen Teilnehmern fürs Mitmachen.* Jeder Einsender erhielt einen Gewinn. Mehrfacheinsendungen wurden nur einmal gezählt. Die Gewinne wurden im Juni zugeschickt; die PCNEWS-CD1 folgt; die CCC-Mitgliedschaften laufen bis Ende 1997. Hier die Gewinner:

1	MS-Visual-Basic 4.0 Enterprise	Prof.Mag. Clemens Gottfried
2	MS-Visual-Basic 4.0 Enterprise	Ing. Wolfgang Zelinka
3	MS-Visual-Basic 4.0 Professional	Ing. Peter Ullrich
4	MS-Visual-Basic 4.0 Professional	Ing. Franz Frühwirth
5	MS-Visual-Basic 4.0 Professional	Gert Gansfuss
6	MS-Visual-Basic 4.0 Professional	Dipl.-Ing. August Hörandl
7	MS-Visual-Basic 4.0 Professional	Robert Schwager
8	MS-Visual-C++ 4.0 Standard	Gerwald Oberleitner
9	MS-Visual-Basic 4.0 Standard	Mag. Gerald Kurz
10	MS-Visual-Basic 4.0 Standard	Prof.Dkfm. Helmut Hamminger
11	MS-Visual-Basic 4.0 Standard	Dr. Georg Czedik-Eysenberg
12	MS-Visual-Basic 4.0 Standard	Gebhard Löhnert
13	MS-Visual-Basic 4.0 Standard	Mag. Alois Vilim
14	MS-Visual-Basic 4.0 Standard	Dipl.-Ing. Andreas Pfeiffer
15	MS-Visual-Basic 4.0 Standard	Christian Hofer
16	MS-Visual-Basic 4.0 Standard	Michael Werner
17	Inside Visual C++ (Buch+CD)	Mag. Martin Wertjanz
18	MS Solutions Development Kit (CD)	Mag. Fritz Eller
19	MS Solutions Development Kit (CD)	Dipl.-Ing. Josef Felkel
20	MS Solutions Development Kit (CD)	Dipl.-Ing. Bertram Geiger
21	MSJ April-96 + PCNEWS-Leiberl	Peter Bildstein
22	MSJ April-96 + PCNEWS-Leiberl	Ing. Wolfgang Portugaller
23	MSJ April-96 + PCNEWS-Leiberl	Mag. Wolfgang Kleinrath
24	MSJ April-96 + PCNEWS-Leiberl	Günther Faulhuber
25	MSJ April-96 + PCNEWS-Leiberl	Reinhold Wenzl
26	MSJ April-96 + PCNEWS-Leiberl	Thomas A. Terényi
27	MSJ April-96 + PCNEWS-Leiberl	Günter Reise
28	MSJ April-96 + PCNEWS-Leiberl	Richard Seidl
29	MSJ April-96 + PCNEWS-Leiberl	Markus Eggenbauer
30	MSJ April-96 + PCNEWS-Leiberl	Günter Wallner
31	PCNEWS-CD1	Mag. Klaus Scheiber
32	PCNEWS-CD1	Dir.Dr. Martin Weissenböck
33	PCNEWS-CD1	Mag. Werner Krause
34	PCNEWS-CD1	Prof. Werner Kuntschik
35	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Peter Ziehensberger
36	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Christian Kollmann
37	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Hauptschule
38	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Mag. Werner Benedikt
39	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Mag. Peter Mazohl
40	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Mag. Leopold Mechtler
41	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Mag.Prof. Alfred Nußbaumer
42	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Dr. Herwig Reidlinger
43	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Konrad Neuwirth
44	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Doz.Dr. Erich Neuwirth
45	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Hannes Schiefer
46	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Ali Eghdamian
47	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Dr. Manfred Pfeifer
48	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Georg Kirchner
49	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Helmut Oberländer
50	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Martin Matuska
51	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Gunnar Haslinger
52	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Johann-Peter Eisbacher
53	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Anita Kail
54	PCNEWS 48..52 + CCC-Mitgliedschaft 96/97	Jürgen Wagner
55	PCNEWS 48..52	Emanuela Schwab
56	PCNEWS 48..52	Mag. Michael Dobes
57	PCNEWS 48..52	Peter Gassner
58	PCNEWS 48..52	Richard Kapongia
59	PCNEWS 48..52	Andrea Deutsch
60	PCNEWS 48..52	Franz Stoiber
61	PCNEWS 48..52	Daniel Csiky
62	PCNEWS 48..52	Gottfried Amtmann
63	PCNEWS 48..52	Andreas Marschall
64	PCNEWS 48..52	Alexander Jr. Schramm
65	PCNEWS 48..52	Georg Horvath
66	PCNEWS 48..52	Thomas Lustig
67	PCNEWS 48..52	Gerald Horvath
68	PCNEWS 48..52	Gregor Göstl
69	PCNEWS 48..52	Peter Winkler
70	PCNEWS 48..52	Thomas Obermayer
71	PCNEWS 48..52	Werner Süss
72	PCNEWS 48..52	Horst Schmid
73	PCNEWS 48..52	Manfred Bergmann
74	PCNEWS 48..52	Peter Ferchländer
75	PCNEWS 48..52	Thomas Schwartz
76	PCNEWS 48..52	Dipl.-Ing. Karel Štípek
77	PCNEWS 48..52	Peter Brandl
78	PCNEWS 48..52	Mag. Edith Reiter-Rödhammer
79	PCNEWS 48..52	Peter Burger
80	PCNEWS 48..52	Karl Falschlehner
81	PCNEWS 48..52	Christoph Chiu
82	PCNEWS 48..52	Dr. Rainer Jelinek
83	PCNEWS 48..52	Mag. Ulrike Ferstl
84	PCNEWS 48..52	Karl Höbartner
85	PCNEWS-CD1	Hans Lohninger
86	PCNEWS-CD1	Dipl.-Ing. Stefan Sedlitz
87	PCNEWS-CD1	Ing. Hubert Steiner

Vorschau

PCNEWSedu 50 „Grafik“

Zum Thema „Grafik“ sind sehr viele Beiträge eingelangt, sodaß die Reihenfolge der Themen der nächsten Nummern geändert wurde. (siehe auch Rubrik Termine). Die nächste Ausgabe ist daher statt dem geplanten Themen „ISDN“ dem Thema „Grafik“ gewidmet. Hier bereits eingelangte Themen von

Werner Krause:
- Kai's Power Tools 3
- KPT Convolver, Version 1.0 für Windows 3.1 / 95 / NT
- Insiderbuch Photoshop
- Das thermische Entwicklungsverfahren
- GEFÜHLSECHT - Gafikdesign der 90er Jahre
- Plug-In Filter von Andromeda
- Fractal Design Painter 4
- 3D Studio MAX
- Free Objects und 3D Fonts
- Do-It-Yourself Szenerien für den Flightsimulator 5
- Kidpix Studio

Siegfried Pflegerl:
- *„Sieht Ihr Film rot auf rot“* Der Riese Polyphem in der Bewußtseinstheorie

Wolfgang Scharl:
- Computergestützte Audio- und Videotechnik

Martin Weissenböck:
- Vergleichstabelle für die GSM-Tarife (A1 und max.mobil) und Features
- Grundlagen über das GPS (Global Positioning System)

Margarete Maurer:
- Weibliche Wissenschaft – Frauen als Mütter der Bombe?
- „Keiner Bombe Mütter“ (Filmbesprechung)

Franz Fiala, Werner Krause, Helfried Kurzmann:
- Grauwertbilder richtig drucken

Franz Fiala:
- Zeichensatzkonversionen mit WinWord-Makros
- BIOS-Interrupts in C

Die weitere Reihenfolge der Themen des nächsten Jahres kann sich verändern. Ankündigungen erfolgen über den Listserver des ccc.

PCNEWSedu 51: „Teleworking, Telelearning“

Josef Felkel:
- Teleworking

PCNEWSedu 52 „Objektorientierte Programmierung“

Peter Speckmayer:
- GRAVITATION

Andreas Pfeiffer:
- Speicherhunger und Leistung von C++ Applikationen

PCNEWSedu 53 „Datenbanken Java“

Franz Fiala:
- WWW-Seiten per Programm aus ACCESS-Datenbank erstellen

PCNEWSedu 50a „PCNEWS-transparent“

Gleichzeitig mit Ausgabe 50 („Grafik“, Auflage 5000) wird auch Ausgabe 50a („PCNEWS-transparent“, Auflage 15000) verteilt. Es ist derzeit noch unklar, ob Ausgabe 50a fester Bestandteil von Ausgabe 50 sein wird oder, ob die Abonnenten 2 getrennte Hefte bekommen. PCNEWS-transparent dient

- zur Information der Leser, da ein Gesamtverzeichnis aller bisher erschienenen Informationen enthalten sein wird;
- für Interessenten für eine Clubmitgliedschaft, da sich alle Herausgeber auf einer Seite präsentieren;
- für Inserenten, da Kenndaten über die PCNEWS und Inseratenpreise enthalten sein werden.

Nachträge

PCNEWS edu -46

Lexikon der Datenkommunikation, S. 52

Das in Ausgabe 46 beschriebene Buch „Lexikon der Datenkommunikation“ wird bei den neueren Auflagen ohne die CD geliefert. Die aktuelle Ausgabe ist die vierte (ISBN 3-89238-147-X). Die CD-ROM kann man im Rahmen eines regelmäßigen Update-Service um DM 30,- bestellen.

Erich Ganspöck

PCNEWS edu -47

Kolleg Multimedia-Designer und -Producer

Das gemeinsam von Graphischer Bundes-Lehr- und Versuchsanstalt und vom TGM angebotene Kolleg ist während der Drucklegung der letzten Ausgabe von einer Abend- in eine Tagesform umgewandelt worden. Das Kolleg startet bereits in diesem Herbst.

Schulen im Internet

Es erreichten uns viele Nachrichten über neue WWW-Seiten von Schulen. Statt eines Nachtrags finden Sie in diesem Heft einen eigenen Beitrag über österreichische Schulen im Internet, in dem alle diese Nachträge eingearbeitet wurden.

Alle Internet-Adressen wurden auch in den WWW-Seiten der PCNEWSedu eingetragen.

➢Einladung⮜ ➢Einladung⮜ ➢Einladung⮜

Die PCNEWSedu bietet im Internet ein Verzeichnis aller Schulen an. Angegeben sind der Name der Schule, Ansprechpartner, Adresse, Telefon- und Faxnummern, E-Mail-Adresse und WWW-Homepage. Wenn eine Homepage existiert, wurde ein Link auf diese Homepage gelegt.

Es besteht die Möglichkeit, ohne großen Aufwand

- ein Bild und/oder Logo und
- eine genauere Charakterisierung

bereits in diesem Verzeichnis zu zeigen. Eine Probe sehen Sie bei der Grafischen Versuchsanstalt und beim TGM. Beide unter „Berufsbildendes Schulwesen“, Wien 14 und Wien 20. **Senden Sie Ihr Logo oder ein Bild an die Redaktion der PCNEWSedu.**

Diese Daten werden bei Bekanntwerden von Änderungen auf den letzten Stand gebracht. Möglicherweise sind Telefonnummern nicht mehr aktuell. **Bitte kontrollieren Sie die Daten Ihrer Schule und schic??ken Sie ein Mail, wenn etwas nicht in Ordnung ist.**

Der Super-DAU! ([D]ümmster [A]nzunehmender [U]ser)

... eines Tages bei der PC-Hotline:

Telefon: Klingel, klingel.
Hotline: Hotline, guten Tag.
DAU: Guten Tag, mein Name ist Epel. Ich habe da ein Problem mit meinem Computer.
Hotline: Welches denn, Herr Epel?
DAU: Auf meiner Tastatur fehlt eine Taste.
Hotline: Welche denn?
DAU: Die Eniki-Taste!
Hotline: Wofür brauchen Sie denn diese Taste?
DAU: Das Programm verlangt diese Taste.
Hotline: Was ist das denn für ein Programm?
DAU: Das kenne ich gar nicht. Aber es will, daß ich die Eniki-Taste drücke. Ich habe schon die STRG-, die ALT- und die Groß-Mach-Taste ausprobiert, aber da tut sich nichts.
Hotline: Was steht denn auf Ihrem Monitor?
DAU: Eine Blumenvase.
Hotline: Nein, ich meine, lesen Sie mal vor, was auf Ihrem Monitor steht.
DAU: Haischkrien!
Hotline: Nein, Herr Epel, was auf Ihrem Schirm steht, meine ich.
DAU: Moment, der hängt an der Garderobe ...
Hotline: Herr Epel ...!
DAU: So, jetzt habe ich ihn aufgespannt ... da steht aber nichts drauf!
Hotline: Herr Epel, schauen Sie mal auf Ihren Bildschirm und lesen Sie mir mal genau vor, was da geschrieben steht.
DAU: Ach so, Sie meinen ... oh, Entschuldigung. Da steht: "Please press any key to continue!"
Hotline: Ach, Sie meinen die "any-key-Taste". Ihr Computer meldet sich auf Englisch.
DAU: Nein, wenn der was sagt, piepst er nur.
Hotline: ... Drücken Sie mal die Enter-Taste.
DAU: Isses wahr! Jetzt geht's. Das ist also die Eniki-Taste! Das können die aber auch draufschreiben! Vielen Dank und auf Wiederhören!
Hotline: ...??????... Bitte, bitte, keine Ursache.

PCNEWS-Disketten, CD

Wenn eine PCNEWS-CD (siehe Ausgabe 47) bestellt wird, enthält diese CD auch automatisch alle bis zu diesem Zeitpunkt erschienenen Disketten. Beispiel: Alle Besteller, die die Bestellkarte im letzten Heft ausgefüllt haben, erhalten nicht nur die Disketten 1..500 sondern auch alle Nummern, die in der heutigen Ausgabe beschrieben werden.

Durch eine Störung am CD-Writer konnten die CDs nicht wie hier beschrieben versendet werden. Nach Schadensbehebung wird der Versand wieder aufgenommen.

PCN-DSK-520: PCNEWSedu-49

\baume	Informatik+Mikroelektronik	Wilhelm Brezovits
\biosint	Software-Interrupts in C	Franz Fiala
\pic	Klein aber PIC-fein	Robert Schwager
\Rekurs	Rekursion	Karel Štípek
\toene	Laut und leise am PC-Lautsprecher	David Solonyna
\uC537	Mikrocontrollerboard für 80C537 oder 80C517A	Hermann Schönbauer
\wordia	Word Internet Assistant 2.0z	

Tools für Embedded Systems

Die folgenden Disketten wurden den PCNEWS von Herrn Ing. Muttenthaler von Firma NOWATRON zur Verfügung gestellt. Für weitere Fragen steht Ihnen Herr Ing. Muttenthaler gerne zur Verfügung: NOWATRON, Eitnergasse 7, 1230 Wien, 0222-865 85 43-0, Fax:DW-7.

PCN-DSK-521..526: 80C166-Tool Chain Demo-V5.0

BSO/TASKING-V5.0: C-Compiler, Assembler, Linker, Locator. Im Code eingeschränkte Demo. Und XVW-Debugger, der sich alle 5 Minuten mit einer Demomeldung meldet.

PCN-DSK-527..529: 8051-C-Demo V4.0

BSO/TASKING: C-Compiler, Assembler, Linker, Locator. Im Code eingeschränkte Version.

PCN-DSK-530..532: 8051 XVW-DEMO

BSO/TASKING: XVW-Debugger, meldet sich alle 5 Minuten mit einer Demomeldung.

PCN-DSK-533..534: EDE-DEMO

BSO/TASKING: Embedded Development Environment, Oberfläche für so ziemlich alle am Markt befindlichen Compiler. Ist auch unter dem Namen CodeWright bekannt. Im Zusammenhang mit Tasking besondere Möglichkeiten. Zeitlich beschränkte (1.1.96) aber sonst vollständig lauffähige Demo.

PCN-DSK-535: SIEFUZZY-DEMO V2.0

BSO/TASKING: Voll lauffähige Demo. Es lassen sich jedoch keine eigenen Files abspeichern. Auch ist On-Line-Debugging nicht möglich.

Glühbirne kaputt

Frage: *Wieviele Programierer braucht man, um eine Glühbirne einzuschrauben?*

Antwort: *Keine, das ist ein Hardwareproblem!!*

Schuledition

Windows '95

für Schulen und Lehrer, Sonderkonditionen für Clubmitglieder

Microsoft, Pablitos, Pcnews

In einer gemeinsamen Aktion von Hersteller, Distributor und Clubs wird versucht, die besonders günstigen Bedingungen des SELECT-Programms von Microsoft direkt an Schulen, Lehrer (Clubmitglieder) weiterzugeben. ***Der Hersteller*** stellt Windows '95 und alle bisher erschienenen Service-Packs sowie freie Software im Rahmen des SELECT-Programms zur Verfügung, ***die Clubs*** stellen eine Sharewaresammlung zum Betriebssystem zusammen, ***der Distributor*** kopiert die Software auf eine gemeinsame CD und stellt die Rechnung, die PCNEWS-Redaktion liefert Installationshinweise.

Schulen und Lehrer können zu einem außergewöhnlich günstigen Preis eine einzigartige Betriebssystem-CD mit SELECT-Lizenz des Herstellers erwerben. Durch die im Preis enthaltenen Service-Packs I und II und zahlreiche frei erhältliche Zusatzprogramme erhalten Sie ein Komplettpaket, das sonst nur mit Mühe in dieser geschlossenen Form verfügbar ist. Installationshinweise werden mitgeliefert. *(Schulbestätigung erforderlich)*

Lehrer, die Clubmitglieder sind, erhalten diese Sonderausgabe der Windows´95 CD zu einem zusätzlich vergünstigten Preis. *(Mitgliedsnummer angeben)*

Clubmitglieder außerhalb von Schulen (oder Lehrer, die Windows-95 bereits besitzen) können die lizenzfreien Teile der CD auch separat auf CD bestellen. *(keine Schulbestätigung nötig)*

Unterstützen Sie diese Aktion und bestellen Sie - falls Sie einen Umstieg auf Windows-'95 planen - diese CD. *Bestellkarte beigebunden. Lieferung ab Seütember.*

FRIC inter AKTIV

FRIC Technische Fachbuchhandlung
Anton FRIC GmbH
Wiedner Hauptstraße 13
A-1040 Wien
Tel.: 0222/505 64 52
FAX: 505 64 52/22

FRIC im Internet:
E-Mail: fric@ping.at
Homepage: http://www.fric.co.at/fric/

Bei uns finden Sie alle Infos über:

Mathematik, Physik, Chemie, Kunststofftechnik, Maschinenbau, Produktion/Automation, Bauingenieurwesen, Wörterbücher, Technische Lexika, Umweltschutz

Computertechnik:

Grundlagen, Hardware, Software, Datenkommunikation

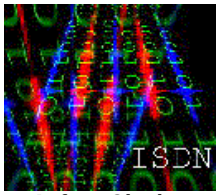
Sie erhalten bei uns auch Zeitschriften und Software!

Termine-PCNEWS edit

1996

Ausgabe	46	47	48	49	50	50a
Schwerpunkt				***Embedded Systems***	***Grafik***	***PCNEWS transparent***
Auflage	***5000***	***5000***	***9000***	***6500***	***5000***	***15000***
Titelbild						
Redaktionsschluß	**Mitte Dezember**	**Anfang März**	**Anfang Mai**	**Ende Juni**	**Anfang Oktober**	**Anfang September**
Druckbeginn Beim Leser	Mitte Jänner Anfang Februar	Mitte März Anfang April	Mitte Mai Anfang Juni	Mitte August Anfang September	Mitte Oktober Anfang November	Ende September Anfang November
Sonderverteilung			Alle Schulen, Plakataktion	Elektronik-HTLs, Entwickler von Programmen für Mikrocontroller	Grafische Lehranstalten, Betriebe Bildnerische Erziehung	Darstellung von 11 Jahren PCNEWS, Beilage zu Nummer 50

1997

Ausgabe	51	52	53	54	55
Schwerpunkt	***Telelearning Teleworking***	***OOP***	***Datenbanken Java***	***LAN, WAN***	***ISDN Telefontechnik***
Auflage	***ca. 5000***	***ca. 5000***	***ca. 5000***	***ca. 5000***	***ca. 5000***
Titelbild					
Redaktionsschluß	**Mitte Dezember**	**Anfang März**	**Anfang Mai**	**Ende Juni**	**Anfang Oktober**
Druckbeginn Beim Leser	Mitte Jänner Anfang Februar	Mitte März Anfang April	Mitte Mai Anfang Juni	Mitte August Anfang September	Mitte Oktober Anfang November
Sonderverteilung	Schulen, Fernlehrinstitute, EDV-Firmen				Anwender von Telekommunikationssystemen

Anmerkungen

- **Termine und Themen** können sich verschieben
- Die **Schwerpunktthemen** sind lediglich Anhaltspunkte.
- **Genaue Termine** werden über den Listserver des CCC bekanntgegeben. Subskribieren mit einer E-Mail an: listserv@ccc.or.at, kein Betreff, Text: SUBSCRIBE PCNINFO.
- **Beleg- und Werbeexemplare:** Inserenten erhalten von der Ausgabe, die das Inserat enthält, 3 Belegexemplare. Solange der Vorrat reicht, erhalten Sie weitere 10 Hefte für die Verteilung an Kunden kostenlos. Wenn Sie Bedarf an einer größeren Stückzahl haben, bitte bis zum Redaktionsschluß bekanntgeben, Sie erhalten die angeforderte Anzahl kostenlos zugeschickt.
- **Preise 1996 (Ausgaben 49, 50):** unverändert A4-SW/Folgeseite: S 2500,-/S 1250,-, A4-2-farb: S 3125,- A4-4farb S 4000,-, Dauerinserat für ein Jahr-20%. Alle Preise jeweils +10% + 20%.

Ausgabe 50a PCNEWS edit-*transparent*

- In dieser Ausgabe 50a, PCNEWS edit-***transparent*** wird ein Gesamtverzeichnis aller Inserenten zu finden sein (und bitte senden Sie dazu unbedingt die Tabelle am Begleitbrief korrigiert und ergänzt zurück). Gleichzeitig laden wir Sie zu einem Inserat in dieser Ausgabe ein.
- Die Ausgabe 50a wird einerseits zur Ausgabe 50 beigelegt sein (5000 Stück), zusätzlich aber auch in hoher Stückzahl an alle Interessenten der PCNEWS edit und an alle Schulen in Österreich versendet (10000 Stück).
- Diese Ausgabe würde sich daher besonders für „langlebige“ Imagewerbung eignen, da für die Stammleser der PCNEWS edit diese Ausgabe ein Nachschlagewerk über alle bisherigen Beiträge und Disketten und daher ein immer wieder benutztes Werkzeug sein wird.
- **Preise (Ausgabe 50a):** A4-SW/Folgeseite: S 5000,-/S 2500,-, A4-2-farb: S 6250,- A4-4farb S 8000,-. Alle Preise jeweils +10% + 20% Teilung der Seite möglich.

➢➢

Termine-PCNEWS edit-Familie

Sep.	02	Mo	18:30	CCC	Treffen	**Point-Treffen**		wird bekanntgegeben
Sep.	02-07	Mo-Sa				**Ars Electronica „memesis - die Zukunft der Evolution“**		Linz, http://www.aec.at
Sep.	04-06	Mi-Fr			Konferenz	**1:1 - Architektur-Modelle im Zeitalter der Virtuellen Realität**		Univ.-Doz. Dr. Bob Martens, Institut für Raumgestaltung der TU Wien, Tel. 0222- 58801 - 3382 DW (3288 DW)
Sep.	05	Do	20:00	Lehrer	Treffen	**Informatiker-Stammtisch Graz**	Mag. Klaus Scheiber	Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz
Sep.	10	Di	19:00	Lehrer	Treffen	**Informatiker-Stammtisch Wien**	Mag. Theresia Oudin	Kleiner Rathauskeller, Rathausstr. 11, 1010 Wien
Sep.	11-13	Mi-Fr		TU-Wien	Konferenz	**Lebenslanges Lernen für Techniker**		Dr. Franz Reichl, Außeninstitut der TU Wien, Tel. 0222- 58801 - 4029 DW (4031 DW)
Sep.	12-15	Do-So		Wiener Messen	Messe	**PCmultimediaHIT**		TEL: 0222-72 72 00, FAX: 0222-72 72 04 13
Sep.	19	Do	18:00	MCCA	Clubabend			1030 Wien, Ungargasse 69
Sep.	25-26	Mi-Do		Temmel und Seywald	Messe	**Internationale Messe für Internet & Electronic Commerce**		TEL: 0222-402 48 51-0, FAX: 0222-402 48 51-18
Sep.	25-27	Mi-Fr		OCG	Konferenz	**Informatik ’96** Beherrschung von Informationssystemen - Weichenstellungen für die Zukunft	http://www.ifi.uni-klu.ac.at/Conferences/informatik96/	Uni-Klagenfurt, kostenpflichtig; Institut für Informatik, Universitätsstraße 65-67, 9020 Klagenfurt FAX: 0463-2700 505
Okt.	01	Di	18:30	CCC	Treffen	**Point-Treffen**		wird bekanntgegeben
Okt.	03	Do	20:00	Lehrer	Treffen	**Informatiker-Stammtisch Graz**	Mag. Klaus Scheiber	Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz
Okt.	04	Fr			Tag der Begegnung	**Technik und Medizin**		Univ.-Doz. DDr. Frank Rattay, Institut für Analysis, technische Mathematik und Versicherungsmathematik der TU Wien, Tel. 0222- 58801 - 5379 DW
Okt.	07	Mo		PCNEWS	Red.schluß	**Grafik**		
Okt.	08	Di	19:00	Lehrer	Treffen	**Informatiker-Stammtisch Wien**	Mag. Theresia Oudin	Kleiner Rathauskeller, Rathausstr. 11, 1010 Wien
Okt.	08	Di		ADV	Vortrag	**Multimedia Internet**	Computer Associates	ADV, Trattnerhof 2, 1010 Wien, 0222-533 09 13, FAX: DW 77
Okt.	09	Mi		PCCTGM	Clubabend	**Clubabend des PCCTGM**		1200 Wien, Wexstraße 21
Okt.	16	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 1**		1200 Wien, Wexstraße 21, 1400
Okt.	15-17	Di-Do		Network	Ausstellung	**Open Systems**		TEL: 050 33-70 57, FAX: 050 33-79 44
Okt.	17	Do	18:00	MCCA	Clubabend			1030 Wien, Ungargasse 69
Okt.	20-21	So-Mo		FID	Seminar	**Working and Learning in a Networked Environment**	iw@db.dk	TEL: +45-31-586 666, FAX: +45-32-840 201
Okt.	23	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 2**		1200 Wien, Wexstraße 21, 1400
Okt.	24	Do		ADV	Tagung	**Computer Integrierte Telefonie**		ADV, Trattnerhof 2, 1010 Wien, 0222-533 09 13, FAX: DW 77
Okt.	30	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 3**		1200 Wien, Wexstraße 21, 1400
Nov.	04-06	Mo-Mi		ECTF	Konferenz	**TELEWORK ‘96**	http://www.bco.co.at/tw96	Wien-Rathaus, kostenpflichtig; bco, Lenaugasse 11/36, 1080 Wien, TEL: 0222-541 72 64, FAX: 0222-559 92 74
Nov.	06	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 4**		1200 Wien, Wexstraße 21, 1400
Nov.	06	Mi	18:30	CCC	Treffen	**Point-Treffen**		wird bekanntgegeben
Nov.	07	Do	20:00	Lehrer	Treffen	**Informatiker-Stammtisch Graz**	Mag. Klaus Scheiber	Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz
Nov.	12	Di	19:00	Lehrer	Treffen	**Informatiker-Stammtisch Wien**	Mag. Theresia Oudin	Kleiner Rathauskeller, Rathausstr. 11, 1010 Wien
Nov.	13	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 5**		1200 Wien, Wexstraße 21, 1400
Nov.	20	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 6**		1200 Wien, Wexstraße 21, 1400
Nov.	21	Do	18:00	MCCA	Clubabend			1030 Wien, Ungargasse 69
Nov.	21-25	Do-Mo		Messe München	Messe	**Systems**, Internationale Fachmesse für Informationstechnologie und Kommunikation		TEL:+49-89-510 70, FAX: +49-89-521 20 86
Nov.	25	Mo		Technikum Kärnten	Konferenz	**Technologie im Lebensraum** - Gebäudeautomation & Elektronik für Lebensqualität	Info@FH-Spittal.ac.at http://www.FH-Spittal.ac.at/cont/htech.htm	Technikum Kärnten, Ortenburgerstraße 27, 9800 Spittal, TEL: 04762-5111-121
Nov.	27	Mi	17:45-20:55	PCCTGM	Seminar	**Einführungsseminar 7**		1200 Wien, Wexstraße 21, 1400
Nov.	27	Mi		PCCTGM	Clubabend	**Clubabend des PCCTGM**		1200 Wien, Wexstraße 21
Dez.	05	Do	18:30	CCC	Treffen	**Point-Treffen**		wird bekanntgegeben
Dez.	05	Do	20:00	Lehrer	Treffen	**Informatiker-Stammtisch Graz**	Mag. Klaus Scheiber	Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz
Dez.	10	Di	19:00	Lehrer	Treffen	**Informatiker-Stammtisch Wien**	Mag. Theresia Oudin	Kleiner Rathauskeller, Rathausstr. 11, 1010 Wien
Dez.	19	Do	18:00	MCCA	Clubabend			1030 Wien, Ungargasse 69
Jan.	09	Do	20:00	Lehrer	Treffen	**Informatiker-Stammtisch Graz**	Mag. Klaus Scheiber	Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz
Feb.	06	Do	20:00	Lehrer	Treffen	**Informatiker-Stammtisch Graz**	Mag. Klaus Scheiber	Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz

➢

Wie ruiniert man einen Betrieb ??

Mit Alkohol Das ist die ungesündeste Methode

Mit Frauen Das ist die schönste Methode

Mit einem Computer Das ist die sicherste Methode

Neues von der ADIM

Martin Weissenböck

Windows

Christian Zahler hat einen Band über Windows (Windows 3.1x, Windows 95) vorbereitet. Außer den wichtigsten Befehlen von der Windows-Varianten werden auch die Produkte der Office-Reihe (Word, Excel, Power-Point) kurz vorgestellt.

Das Skriptum ist als Unterlage für die erste Einführung im Unterricht gedacht. Es soll keine umfangreichen Handbücher ersetzen; vielmehr soll nach der Einführung mit den umfangreichen Hilfedateien weitergearbeitet werden.

Der Band mit der Nummer 63 sollte spätestens ab Oktober verfügbar sein.

Delphi

Ein Delphi-Band wird schon von vielen Interessenten erwartet; leider steht noch kein genauer Termin für die Fertigstellung fest.

Java

Java ist der Sprache C++ sehr ähnlich. Ein der nächsten Ideen: ein Java-Band soll, im Aufbau gleich dem Band 50, C++, entstehen. Derzeit werden noch passende Entwicklungsumgebungen erprobt. Borlands C++ enthält in der Version 5.0 auch einen Java-Compiler.

Martin Weissenböck

Liebe Mitglieder des PCCTGM

Robert Syrovatka

Für das Schuljahr 1996/97 kündigen wir Ihnen einige Termine bereits jetzt an, obwohl eine exakte Planung zum Redaktionsschluß leider noch nicht möglich ist. Die Termine sind vorerst noch unverbindlich und solle nur eine Vorinformation sein.

Clubabende

Mittwoch 9. Oktober 1996

Mittwoch 27. November 1996

Die Themen, sowie allfällige Änderungen erfahren Sie über unseren Anrufbeantworter (Tel.: 0222-332 23 98).

Seminare

Einführungs-Seminar für Anfänger

Betriebssystem DOS, WINDOWS 3.x, Installation von Programmen, Datensicherung, Viren und Virenschutz. Übersicht über Windows 95.

Beginn: Mittwoch 16. Oktober 17:45, Saal 1400

Allfällige Änderungen erfahren Sie über unseren Anrufbeantworter (Tel.: 0222-332 23 98).

Robert Syrovatka

Allen Mitgliedern des PCC-TGM können ab sofort den Internet-Zugang des CCC benutzen. Die Bedingungen dazu finden Sie bei den CCC-Ankündigungen weiter hinten.

Franz Fiala

Regelmäßige Termine

Graz

- **Treffen des CLUB AT** finden jeden zweiten Freitag im Clublokal Wickenburggasse 8, 8010 Graz ab 19:00 statt.
- **Informatiker-Stammtisch Graz** Gasthaus „Zum Goldenen Hirschen“, Kahngasse 22, 8045 Graz-Andritz (1. Termin: 5. September 1996)

Wien

- **FIDO-Wien**: FIDO-User und FIDO-Interessierte treffen einander jeden Freitag um 19:00 am im Café Heine, 1020 Wien.
- **FIDO-Point-Treffen** der Mailbox „*His Master's Voice*“ treffen einander am Monatsanfang mit wechselnden Wochentagen. (Ort wird im FIDO-Echo CCC.AUS oder über die Mailing-List CCCINFO bekanntgegeben.)
- **Clubabende des MCCA** finden jeden 3. Donnerstag im Monat, Am Heumarkt 4, 1030 Wien ab 18:00 statt.
- **INTERNET-Schnupperseminare von PING** jeden Donnerstag,. 19 Uhr Thurngasse 8/3, Dauer 90 Minuten, 500 ATS, inkl. MWSt.
- **Magnet Wien Stammtisch:** Jeden letzten Mittwoch im Monat im Café Stein, Ecke Kolingasse/Währingerstraße 6-8, 1090 Wien ab 19:00 oben links im Lokal
- **Magnet Stammtische in den Bundesländern:** unregelmäßige Termine zu erfragen bei Magnet unter 0222-585 1771 oder im magnet CITY Online Service unter magnet/magnetTreff
- **Informatiker-Stammtisch** jeden zweiten Dienstag im Monat. 19 Uhr Kleiner Rathauskeller, Rathausstr. 11, 1010 Wien, 19.00 Uhr, Kontakt: `Theresia_Oudin@bboard.blackbox.or.at` oder Mag. Gerald Kurz - `Schulnetz: 22:102/1.100`.

Veranstaltungen der TUWIEN

4. - 6. 9. 1:1 - Architekturmodelle im Zeitalter der Virtuellen Realität

Die Konferenz versucht einen kritischen Blick auf den Bereich der Simulation in der Architektur. Dabei geht es vor allem um 1:1-Modelle und andere Designwerkzeuge, einschließlich der psychologischen Grundlagen dieser Techniken. Kontakt: Univ.-Doz. Dr. Bob Martens, Institut für Raumgestaltung der TU Wien, Tel. (0222) - 58801 - 3382 DW (3288 DW)

11. - 13. 9. Lebenslanges Lernen für Techniker

Die Konferenz „Educating the Engineer for Lifelong Learning“ ist die diesjährige Konferenz der SEFI (European Society for Engineering Education) und wird vom Außeninstitut der TU Wien veranstaltet, das traditionell sehr aktiv im Bereich der Weiterbildung ist (seit Jahren ist die TU Wien größter Anbieter von Hochschulkursen und Hochschullehrgängen). Die Konferenz steht im Zeichen des „Europäischen Jahres des lebens-begleitenden Lernens“. Der thematische Bogen spannt sich von inhaltlichen Fragen (was soll gelehrt werden, welche Fähigkeiten soll ein Techniker haben) bis zu formalen und technischen Problemen (Fernlehre, technische Hilfen, Computer, aber auch pädagogische Fragen). Kontakt: Dr. Franz Reichl, Außeninstitut der TU Wien, Tel. (0222) - 58801 - 4029 DW (4031 DW)

4. 10. (Hörsaalzentrum Allgemeines Krankenhaus Wien) Tag der Begegnung: „Technik und Medizin“

Vor knapp zwei Jahren wurde die Arbeitsgruppe „TU-BioMed“ gegründet, ein informeller Zusammenschluß von TU-Wissenschaftern, die im Bereich biomedizischer Technik arbeiten. Der „Tag der Begegnung“, der gemeinsam mit der medizinischen Fakultät der Universität Wien veranstaltet wird, soll jüngste Forschungen und Entwicklungen einem breiteren Publikum präsentieren. In einer Reihe von Referaten werden sowohl allgemeine Fragen behandelt als auch vier konkrete Projekte vorgestellt. Begleitend dazu wird von 4. bis 31. Oktober eine Posterausstellung zu sehen sein. Kontakt: Univ.-Doz. DDr. Frank Rattay, Institut für Analysis, technische Mathematik und Versicherungsmathematik der TU Wien, Tel. (0222) - 58801 - 5379 DW

Liebe Freunde des MCCA

Josef Sabor

Die drei U bestimmen den Herbst des MCCA

Umstellung • **U**morganisation • **U**mzug

Umstellung

Die Umstellung auf das neue System A-Online hat nicht nur Ihnen, sondern auch dem MCCA einige Schwierigkeiten bereitet. Das gesamte Mitteilungs- und Mitgliedersystem, das unser Ehrenobmann Helmuth Schlögl in mühevoller Arbeit aufgebaut hatte, mußte umgestellt werden.

Wir hoffen aber, daß Sie nun bereits eine eigene E-Mail-Kennung haben, sodaß Sie unsere *941# des Mitteilungsdienstes wieder erreichen können.

Umorganisation

Die Umstellung des Systems und die damit verbundenen Terminschwierigkeiten der datakom-austria haben den MCCA zu einer Änderung seiner Zielvorstellungen veranlaßt:

Während in den letzten Jahren die Förderung des PAN/BTX im Vordergrund stand, und wir eigentlich schon in dieser Zeit die Anliegen der User (Benutzer und Anbieter) gegenüber dem Netzbetreiber vertraten, stellt die Umstellung auf A-Online eine neue und geänderte Aufgabenstellung für den MCCA dar.

Die Förderung des Systems - und damit die Unterstützung einer privaten Firma - kann und soll nicht mehr unbedingt unser vordringlichstes Ziel sein. Sicherlich hängt unser Herz noch am CEPT-Teil des A-ONLINE, aber ungeachtet dessen, wird unser neuer Vereinszweck als **A-ONLINE Usergroup** verstärkt auf **Ihre** Forderungen eingehen.

Mit Ihrer Unterstützung haben wir es gemeinsam geschafft, sinnlose und unbedachte Vergebührungsideen der datakom austria zu unterbinden, bevor diese noch zur Verrechnung kamen.

Mit Ihrer Unterstützung wird die **A-ONLINE Usergroup** auch weiterhin dem Netzbetreiber genau auf die Finger schauen.

Umzug

Leider mußten wir im Juni schweren Herzens kurzfristig unser Clublokal „Am Heumarkt 4“ in den Räumen der GiroCredit (bei der wir seit Anbeginn - in wechselnden Standorten - kostenlos Gast sein durften) räumen, da die GiroCredit diese Räumlichkeiten nur angemietet hatte, und diese Räume nun an den Vermieter zurückgingen.

Dankenswerterweise konnten wir aber rasch durch das Entgegenkommen des Schulzentrums Ungargasse eine neue Bleibe finden!

Sie finden uns daher ab September in unseren neuen Räumlichkeiten in 1030 Wien, Ungargasse 69. Unser neuer Clubabend-Tag wird dann jeder 3. DIENSTAG im Monat sein (Verschiebungen gibt es, wenn dieser Tag auf Ferientage oder anschließend an Ferien fällt).

Sie sehen also, daß das eigentlich eingeplante vierte U (nämlich Urlaub) mit verschiedenen administrativen Arbeiten (neue Telefonleitung, Adreßänderungen usw.) verbunden sein wird. Da der Vorstand des MCCA ausschließlich ehrenamtlich und in seiner Freizeit tätig ist, wird für das persönliche U daher weniger Zeit bleiben.

Wir hoffen aber, daß Sie die Möglichkeit haben (hatten), sich zu erholen und hoffen, Sie zahlreich am 17.September 1996 in unserer Mitte begrüßen zu dürfen.

Ing. Josef Sabor

PS: Unsere Halbpreisaktion für neue Mitglieder läuft bereits wieder (*255010#)!
PPS.: Da wir nunmehr in den neuen Räumlichkeiten Miete und Nebenkosten zahlen müssen, würden wir uns über zahlreiche neue Mitglieder sehr freuen, da wir ja außer den Mitgliedsbeiträgen keine Einkünfte haben!

Liebe Freunde der HYPERBOX

Martin Reinsprecht

Die permanente Internetpräsenz der HYPERBOX bietet einige Möglichkeiten der Veränderungen und Erneuerungen. Nicht nur, daß der in der letzten Ausgabe angekündigte URL durch einen einfacheren ersetzt wurde, nämlich:

`http://www.hyperbox.org/`

Zu jenem Zeitpunkt, zu dem Sie hiesige Ausgabe in Ihren Händen halten, müßte die redaktionelle Umgestaltung der Web-Seiten bereits abgeschlossen sein. Die Maturaaktivitäten einiger dafür zuständiger Mitglieder verhinderten hierbei leider einen rascheren Fortschritt.

HYPERBOX zum Ortstarif!

Der Idee eines global verbreiteten Computernetzes, nämlich mit konstanten Kosten jeden beliebigen Standort zu erreichen, folgend, gibt es jetzt die HYPERBOX zum Ortstarif.

Kurz: Der User wählt bei seinem lokalen Internet-Provider ein, eröffnet eine TELNET-Sitzung auf die HYPERBOX und weiter wie gewohnt...

`telnet bbs.hyperbox.org`

FTP & HYPERBOX

Momentan existieren 3 Möglichkeiten Dateien von der HYPERBOX zu erlangen:

- Im Rahmen der konventionellen FIDO-Mailboxbenutzung unter der Telefonnummer 0732/677567.
- Zugriff via WWW (WorldWideWeb), der URL lautet: `http://www.hyperbox.org/ftp/`
- FTP-Sitzung, URL: `ftp://bbs.hyperbox.org`

Internetzugang

Die Hyperbox wird auch Mitgliedschaftspakete anbieten, die den Zugang zum Internet ermöglichen. Inkludiert ist daher auch eine E-Mail Adresse `vorname.nachname@hyperbox.org`.

Die Abwicklung erfolgt über den Netscape Mail Server, der einige Features bietet:

Nach einer Authentifikation kann der User via WWW

- sein Paßwort ändern
- eine „Finger Reply Information“ eingeben, welche u.a. dem UNIX-Befehl „finger“ folgt
- - wenn er nicht erreichbar ist - eine „Vacation Message“ bekanntgeben
- eine automatische Weiterleitung von E-Mails auf einen beliebigen anderen Mail-Account („Forwarding“) einrichten
- ... und noch einiges mehr

Nähere Infos erhältlich bei: `bernhard.jungwirth@hyperbox.org`

Bleibt noch zu erwähnen, daß sich die HYPERBOX mit dem in Oberösterreich längst überfälligen FIDO/Internet-Gateway beschäftigt und sich darauf freut, in Kürze die Eröffnung bekanntzugeben.

Martin Reinsprecht

Liebe Mitglieder des CCC!

CCC-Card - Ermäßigungen !

Unter `http://www.ccc.or.at/partner/kk/index.html` finden Sie einen neuen Partner für unser Clubkarten-Programm. Firma

Karl Krauskopf
Büromaschinen und Copyshop
Margaretenstraße 46
1040 Wien

bietet Inhabern der Computer Competence Card - für Selbstkopien den Preis für Studenten an.

Gratis WWW - Werbeplatz für Unternehmen

Falls auch Sie mit Ihrer Firma das Angebot nutzen wollen, gratis Werbung für Ihr Unternehmen (bis max. 0,5 MB Speicherplatz auf unserem Server) im Tausch gegen Ermäßigungen für unsere Mitglieder (gegen Vorweis der Computer Competence Card) zu machen, sind Sie gerne herzlich dazu eingeladen. Kontaktieren Sie uns unter office@ccc.or.at.

Wir können Ihnen auch gerne jemanden Vermitteln, der professionell Seiten erstellt und falls Ihnen die 0,5 MB zu wenig sein sollten - natürlich über unseren Internet-Partner auch mehr WWW Speicherplatz zur Verfügung stellen.

His Master's Voice via Internet - Telnet

Unsere Clubmailbox His Master's Voice ist seit kurzem auch Telnet (Protokoll zur Übertragung von Terminalsessions über TCP/IP) aus dem Internet erreichbar.

Sie können die Mailbox mit Telnet unter der Adresse "hmv.ccc.or.at" erreichen. Falls Ihr Telnet es erlaubt, die Terminalemulation der Telnet - Session zu verändern, so stellen Sie als Emulation "ANSI-BBS" ein. Beim Telnet von OS/2 funktioniert das beispielsweise mit dem Befehl

telnet -t ansi hmv.ccc.or.at

unter Windows'95 gibt es ein Shareware-Programm namens CRT. Dieses habe ich auf www.shareware.com gefunden. Es wird sehr empfohlen, das Programm zu registrieren. Erstens ist das Programm das beste telnet, daß ich unter Windows'95 gefunden habe und zweitens läuft das Programm Ende September ab.

Falls Ihr Telnet es nicht erlauben sollte, die Terminalemulation auf ANSI umzustellen, so müssen Sie, um keine zerstörten Bildinhalte zu sehen in der Mailbox im Setup Menü die Terminalemulation auf "TTY" stellen. Stellen Sie dann auch die IBM Grafikzeichen ab. Leider ist hmv in diesem Fall dann auch wieder schwarz/weiss. Der Bildschirm ist dann jedoch hoffentlich wieder lesbar. Die zweite alternative wäre natürlich sich ein brauchbares "Telnet" für das jeweilige Betriebssystem zu besorgen.

Was bringt es nun, daß die Mailbox über Telnet erreichbar ist ?

1) Benutzer, die Zugriff auf eine Internetverbindung (via Standleitung) haben, können diese nutzen, um in HMV einzuloggen.

2) Benutzer, die nicht im Nahbereich Wien wohnen, können bei Ihrem lokalen Internet-Provider (zum Ortstarif) anrufen und eine Verbindung zu HMV über Internet aufbauen. HMV ist dadurch sozusagen aus der ganzen Welt zum Telefon-Ortstarif erreichbar.

Technische Realisierung

Die Mailbox lauft unter dem Betriebssystem IBM OS/2 3.0 (Warp). Es wurde zusätzlich ein Treiber (SIO+VMODEM) installiert, der auf das TCP/IP Protokoll (in Warp Connect enthalten) aufsetzt und nach oben hin ein Modem simuliert. Auf dieses "Virtuelle Modem" setzt nun wieder unser Mailboxprogramm "Maximus" auf, das denkt es redet mit einem Modem. Maximus "weiss" also sozusagen gar nicht, dass eine Verbindung über TCP/IP geschaltet wird.

Weitere Pläne

Wir arbeiten daran, die Dateien von HMV im Internet via FTP (File Transfer Protocol) zur Verfügung zu stellen. Wenn wir das geschafft haben, wird es möglich sein, alle Dateien von HMV aus dem Internet via FTP zu übertragen.

Schon längere Zeit, werden wir von unseren Mitgliedern danach gefragt - längere Zeit arbeiten wir schon darauf hin. In den vorigen PCNEWS haben wir es bereits angekündigt. Wir möchten unseren Clubmitgliedern einen günstigen

Internet - Vollzugang

bieten. Ich glaube mit dem folgenden Angebot ist uns dies gelungen. Die folgenden Angebote gelten nur für Clubmitglieder - Nutzung nur für private Zwecke. Einwahlpunkt derzeit in Wien (V.34+). St. Pölten für Herbst in Vorbereitung.

CCCat ClubLink

Voller, **zeitmäßig und mengenmäßig unlimitierter Internet - Zugang** inklusive weltweit gültiger E-Mail Adresse (POP3 Server). Private Homepage (einmalige Einspielung einer Homepage inklusive Bild), **um nur 250,-- (incl.) pro Monat.** Bei Bezahlung von 12 Monaten im voraus werden nur 2500,-- (incl.) pro Jahr, das sind

208,33 pro Monat

CCCat ClubNight

Voller, mengenmäßig unlimitierter Internet - Zugang (Zugangszeit beschränkt zwischen 18h und 9h, Samstag, Sonntag ganztägig) inklusive weltweit gültiger E-Mail Adresse (POP3 Server) Private Homepage (einmalige Einspielung einer Homepage inklusive Bild), **um nur 138,-- (incl.) pro Monat.** Bei Bezahlung von 12 Monaten im voraus werden nur 1380,-- (incl.) pro Jahr, das sind

115,00 pro Monat

CCCat ClubDay

Voller, mengenmäßig unlimitierter Internet - Zugang (Zugangszeit beschränkt zwischen 8h und 18h, Samstag, Sonntag ganztägig) inklusive weltweit gültiger E-Mail Adresse (POP3 Server) Private Homepage (einmalige Einspielung einer Homepage inklusive Bild), **um nur 174,-- (incl.) pro Monat.** Bei Bezahlung von 12 Monaten im voraus werden nur 1740,-- (incl.) pro Jahr, das sind

145,00 pro Monat

Die Vertreibung der Frauen aus der Wissenschafts- und Technikgeschichte

Margarete Maurer

Ort des Geschehens: Ein internationales Meeting von Biochemikern in England. Unter den Teilnehmern, wie es bei solchen eher lockeren Treffen üblich ist, einige der Gemahlinnen. Manche von ihnen haben ihre Männer schon öfter begleitet und sind daher mehr oder weniger über die verhandelten Themen informiert, so daß sie sich wenigstens ab und zu mit einigen Bemerkungen ins Gespräch mengen können. Gerade erläutert eine Dame in scharfsinniger Form eine der angeschnittenen Fragen. Sagt ein Herr zu ihrem Ehemann: *„Sie haben ihre Frau aber gut trainiert“* worauf ihn ein weiterer Gast aufklärt: *„She's not a lady, she is a scientist! Sie ist keine Dame, sie ist Wissenschaftlerin.“* Dies geschah nicht etwa im Mittelalter, sondern im dunklen 20. Jahrhundert und macht deutlich, daß sich längst noch nicht alle Männer an die Existenz von Frauen in Naturwissenschaft und Technik gewöhnt haben. Eine Kollegin erzählte mir: Wenn eine Ingenieurstudentin zu spät in die Vorlesung kommt, ist es an vielen Technischen Hochschulen so, daß sie ein Pfeifkonzert erlebt oder freche bis anzügliche Bemerkungen zu hören bekommt, und zwar nicht nur von seiten der Studenten, sondern auch der Dozenten. Immer noch müssen Frauen, die es wagen, in das von Männern beherrschte Feld der Technik und Naturwissenschaft einzudringen, mit Schwierigkeiten rechnen und sich mühsam durchsetzen.

Dabei haben Frauen sich längst einen Platz in der Wissenschaftsgeschichte erkämpft, wenn auch der Frauenanteil in den naturwissenschaftlich-technischen Studienfächern und Berufsfeldern bei uns noch keineswegs ihrem Anteil an der Bevölkerung entspricht. In den USA und in der UdSSR sehen die Zahlen etwas besser aus, doch ist der Kampf der Frauen um angemessene naturwissenschaftlich-technische Berufsmöglichkeiten auch dort noch keineswegs zu Ende.

Im Laufe der abendländischen Wissenschaftsgeschichte hat es nämlich durchaus immer wieder Frauen gegeben, denen es gelang, die Schranken des herrschenden patriarchalischen Systems zu durchbrechen und ihren mathematischen naturwissenschaftlichen und technischen Interessen nachzugehen. Ihre zum Teil hohen wissenschaftlichen Leistungen werden in den entsprechenden Nachschlagewerken, Handbüchern und biographischen Lexika indes häufig noch nicht einmal erwähnt. Und wenn die Autoren wissenschaftsgeschichtlicher Werke hier und da doch nicht umhin kommen, einige bedeutende Frauen ihrer außerordentlichen fachlichen Leistungen wegen wenigstens zu nennen, da wird ihre Bedeutung oft unterschätzt, zu gering bewertet oder gar bewußt herabgewürdigt.

Es ist also nicht verwunderlich, wenn der gebildeten Öffentlichkeit im allgemeinen und den Wissenschaftler(inne)n im besonderen nur allzu wenig über die wissenschaftlichen Leistungen von Frauen bekannt ist. Anhand der im folgenden in den Text eingestreuten Biographien kann jede(r) Leser(in) an sich selbst den diesbezüglichen Kenntnisstand testen.

Wer sich allerdings die Mühe macht, in Bibliotheken und Archiven gründlich nachzuforschen, wird nach längerem Suchen doch einige Informationen über die Rolle der Frauen In Naturwissenschaft und Technik finden. Vieles läßt sich dem sehr materialreichen „Woman in Science“ von H.J. Mozans entnehmen, das 1913 zum ersten Mal erschien und 1970 vom Massachusetts Institute of Technology neu herausgegeben worden ist. Wie dieses Buch ist die Mehrheit der wenigen Schriften zu diesem Thema in Englisch erschienen. Dies dürfte zum Teil der in den angelsächsischen Ländern stärker fortgeschrittenen „women's lib“-Bewegung zu danken sein, zum Teil aber auch dem Wunsch der US-Regierungen, im Interesse der Rüstungspolitik auch die weiblichen naturwissenschaftlich-technischen Begabungsreserven zu mobilisieren.

Wie die herrschende deutschsprachige Wissenschaftsgeschichtsschreibung im allgemeinen mit den Frauen umgeht, will ich im folgenden an einigen Beispielen demonstrieren.

Die einfachste Möglichkeit, die Geschichte von Frauen zu unterdrüc??ken, besteht darin, ihre Namen oder ihre Leistungen oder gleich beides schlicht totzuschweigen. So macht es zum Beispiel H.J. Störig in seiner „Kleinen Weltgeschichte der Philosophie“. In seinem 550 Seiten starken Werk kommen Frau praktisch überhaupt nicht vor und die wenigen Ausnahmen höchstens in irgendeiner Beziehung zu einem Mann, nicht aber selbständig, als Wissenschaftlerinnen. Die einzige im Register aufgeführte Frau ist Königin Christin von Schweden. Sie wird im Text allerdings nur deshalb und nur insofern erwähnt, als sie Descartes nach Schweden eingeladen hatte. Die weiteren spärlichen Frauen des Textes erscheinen ebenfalls nur als bloßes Attribut eines männlichen Philosophen oder Wissenschaftlers. Dies gilt selbst für die Übersetzerin und Kommentatorin von Newtons Hauptwerk, Emile de Breteuil. Sie wird (als „Marquise du Chatelet") lediglich in ihrer Funktion als „die Geliebte des Voltaire“ für erwähnenswert gehalten; dies widerfährt ihr auch in anderen Werken.

TAPPUTI-BELATEKALLIN, THEOSEBEIA, PAPHNUTIA, KLEOPATRA: Anfänge der Chemie

Der Name von Tapputi-Belatekallin, der Parfümherstellerin, findet sich auf einer Keilschrifttafel des alten Mesopotamien, die sich auf die Zelt zwischen 1256 und 1209 vor unserer Zeitrechnung datieren läßt. Sie wurde zusammen mit sechs anderen Tafeln gefunden, die allesamt die technologischen Aspekte der Parfümherstellung behandeln. Aus dem zweiten Teil ihres Namens geht hervor, daß Tapputi, die Parfümherstellerin, der Vorstand eines Haushalts war, eine Hausfrau also. Offensichtlich unterstand die Herstellung von Parfümprodukten Hausfrauen, die für die einzelnen Produktionsschritte Ihre eigenen Methoden entwickelt hatten. Dabei müssen sie durchweg empirisch vorgegangen sein, und zwar so, daß sie von der Kombination normaler Küchenoperationen über die weitere Entwicklung von Haushaltsgeräten zu immer spezialisierteren Verfahren der Destillation Extraktion und Sublimation fortschritten. Für die Extraktion beispielsweise schreiben sie vor, den Vorgang je nach Fall 20 bis 40 Mal zu wiederholen – ein Gedanke, der sich durch die gesamte Geschichte der späteren Alchimie hindurchzieht. Zur Destillation und Sublimation wurde von den Babylonierinnen ein sogenannter „Diquaru“ benutzt – das ist ein Topf aus Metall, der mit einem Deckel versehen ist und über lange Zeit erhitzt werden kann. Er stellt als Fortentwicklung eines normalen Küchentopfes die Urform der Destillationsgeräte dar, wie sie später von den Ägyptern, den Alchimisten der ersten nachchristlichen Jahrhunderte und von den islamischen Chemikern des 8. bis 11. Jahrhunderts benutzt worden sind. Die frühe Geschichte der abendländischen Chemie ist also babylonischen Hausfrauen zu danken, die vor über 3000 Jahren in Mesopotamien Parfüm herstellten. (Levey 1956, 1961).

Eine ähnliche, etwas abgeschwächte, Form des Totschweigens besteht darin, daß man nicht prinzipiell alle Frauen (als Wissenschaftlerinnen/Technikerinnen/Medizinerinnen) totschweigt, sondern – gewissermaßen als Feigenblatt – nur einige wenige – nennt, die schon zu berühmt waren, als daß man ihre Namen und ihre Leistungen aus dem historischen Gedächtnis der Wissenschaft einfach hätte verdrängen können. So enthält beispielsweise jede Geschichte der Chemie zumindest eine Erwähnung von Marie Curie, die 1903, zusammen mit ihrem Mann Pierre Curie und Henry Becquerel, den Nobelpreis für Physik und 1911 zusätzlich noch alleine den Nobelpreis für Chemie erhielt. Weitere Frauen, die aus solchen Gründen in wissenschaftsgeschichtlichen Büchern relativ häufig vorkommen, sind Hypatia von Alexandria, Caroline Herschel, Irene Joliot-Curie, Lise Meitner und Maria Göppert-Meyer. Doch damit hört die weibliche Geschichte dann im allgemeinen auch schon auf. Alle weiteren, wie fast alle in den nebenstehenden Kästen erwähnten Wissenschaftlerinnen, fallen in den meisten Werken unter den Tisch. Das gilt für die große „Biographische Enzyklopädie der Naturwissenschaften und Technik“ von Isaac Asimov, die 1973 bei Herder erschienen ist, genauso wie für Armin Hermanns Buch „Große Physiker – Vom Werden des neuen Weltbildes“ (Stuttgart 1959). Selbst das ansonsten so gründliche und umfassende vierbändige Werk „Wissenschaft. Science in History“ des Physikers J.D. Bernal (Rowohlt, 1970) entspricht diesem Schema. Die beiden letztgenannten Autoren halten noch nicht einmal Maria Göppert-Meyer und Irene Jollot-Curie für erwähnenswert.

Hingewiesen werden soll allerdings auch auf eine Ausnahme aus dem angloamerikanische Raum: Das sehr umfangreiche mathematikgeschichtliche Nachschlagewerk „Bibliography and Research Manual on the History of Mathematics“, 1973 von V. Kenneth und O. May herausgegeben, führt neben Hypatia auch Maria Gaetana Agnese, Sonya Kovalevskaya, Sophie Germain u.a. auf.

MARIA DIE JÜDIN (1. Jh.n.Chr.)

Arbeiten der babylonischen Parfümherstellerinnen wurden in den ersten nachchristlichen Jahrhunderten von bedeutenden Alchimistinnen weitergeführt, unter ihnen Theosebia, die Schwester des Zosimos, Paphnutia die Jungfrau und Kleopatra die Alchimistin. Eine unter ihnen ragte besonders heraus: Maria die Jüdin. Maria war eine reiche Frau und von Priestern, die ihre Begabung erkannt hatten, in die geheime Kunst der Alchimie eingeweiht worden. Sie wird in der späteren alchimistischen Literatur sehr häufig zitiert. Ihr wird die Entwicklung zahlreicher technischer Apparate zugeschrieben, ohne die auch die heutige Chemie nicht auskommt; so die Entwicklung verschiedener Öfen und von Koch- und Destilliergefäßen aus Metall, Ton und Glas. Als Verbindungsmittel benutzte Maria einen sogenannten „Tonerdekitt der Philosophen“. Einige ihrer Erfindungen tragen ihren Namen und sind in der Alchimie vielfach zitiert. Sie ist Erfinderin des Wasser-, des Sand- und des Ölbades, also von Techniken, die heute feste Bestandteile des Chemiestudiums sind. (Wininger o.J., Alic 1981).

Eine subtilere Form der Verdrängung von Frauen aus der Wissenschaftsdokumentation geht so vor sich: Man nennt zwar ihre Namen, unterbewertet aber ihre Leistungen. Dies geschieht häufig bei Frauen, die mit ihrem Mann oder einem anderen männlichen Verwandten zusammengearbeitet haben, was jedoch oft die einzige Möglichkeit war, überhaupt wissenschaftlich zu arbeiten. Die Leistungen der Frau werden dabei in der Regel ab-, die des Mannes aufgewertet. So geschieht es oft mit Caroline Herschel, die mit ihrem Bruder zusammenarbeitete, aber selbständig acht Kometen entdeckte und mehrere astronomische Schriften veröffentlicht hat.

HYPATIA (370-415 n.Chr.)

Sie war die berühmteste Gelehrte der Antike, ein Universalgenie von großem Wissen, und lehrte im 4./5. Jahrhundert in Alexandria Philosophie, Algebra, Geometrie, Mechanik und Astronomie. Sie verfaßte mehrere mathematische Schriften und einen Kommentar zum „astronomischen Kanon“ des Ptolemäus, der Tafeln über die Bewegung der Himmelskörper enthielt. Außerdem entwickelte sie zwei wichtige astronomische Instrumente sowie ein Areometer. Wie Plotin berichtet, galt Hypatia bald als die fähigste und klügste Repräsentantin der platonischen und aristotelischen Philosophie ihrer Zeit, so daß aus allen Gegenden der hellenistischen Welt Schüler zu ihren Vorlesungen strömten. Als Platonikerin wurde sie 415 von fanatischen christlichen Mönchen auf bestialische Weise ermordet. (Mozans 1981, Alic 1981).

Eine weitere Form der Leistungsabwertung findet sich in Asimovs „Biographischer Enzyklopädie der Naturwissenschaften und Technik“. Dieses Werk enthält folgende Passage über Hypatia: *„Berichte rühmen ihre Schönheit und Tugend, ihr Wissen und die Beliebtheit ihrer Vorlesungen, so daß spätere Zeiten ihre Gestalt in einem verklärten Lichte sahen. Sie brachte nichts Eigenständiges in die Wissenschaft ein, verfaßte aber sehr nützliche Kommentare über so frühe Gelehrte wie Ptolemäus und Diophantes.“*

HILDEGARD VON BINGEN (1098-1179)

Sie entwickelte bereits Im 12. Jahrhundert Ansätze eines heliozentrischen Weltbildes. Die Sonne, so war ihre Ansicht, stehe im Zentrum des Firmaments, und sie halte die um sie kreisenden Sterne in derselben Weise fest, wie die Erde die Lebewesen anzieht, die auf ihr leben. Dies ist eine sehr bemerkenswerte Ansicht, denn im 12. Jahrhundert galt allgemein das geozentrische Weltbild, und die universelle Gravitation war noch unbekannt. Hildegard von Bingen verfaßte auf der Grundlage ihres enzyklopädischen Wissens eine ganze Reihe wichtiger medizinischer und naturgeschichtlicher Werke. Aber sie war nicht nur eine prominente Naturforscherin, Ärztin und Philosophin; als Äbtissin des Benediktiner-Konvikts von Bingen am Rhein verfaßte sie drei große theologische Schriften und hatte bedeutenden Einfluß auf die Gelehrten und Politiker ihrer Zeit. Außerdem wurde sie als Dichterin und Musikerin berühmt. (Mozans 1981, S.233-235, Jonas 1979).

Dieses vorschnelle Urteil Asimovs muß schon angesichts des Wenigen erstaunen, das derzeit über Hypatias Schriften bekannt ist. Warum Asimov die „sehr nützlichen“ Kommentare nicht als eigenständige Leistung anerkennen will, an anderer Stelle hingegen die Enzyklopädie eines männlichen Philosophen (des Pappus) als „meisterhafte“ Leistung bewertet, bleibt unerfindlich. Und im übrigen: welcher überragende Wissenschaftler wird zu späterer Zelt nicht in „verklärtem Licht“ gesehen? Des weiteren mußte auch Asimov bekannt sein, daß Hypatia auch durch die Entwicklung wissenschaftlicher Instrumente Wichtiges – und „Eigenständiges“ – geschaffen hat. Trotzdem wirkt die Darstellung Hypatias bei Asimov noch etwas freundlicher als bei J.D. Bernal: Dieser schreibt gar nichts über sie – außer über ihren tragischen Tod. Muß eine Frau erst gefoltert und ermordet werden, um in ein wissenschaftsgeschichtliches Werk aufgenommen zu werden?

BARONESSE DE BEAUSOLEIL (17. Jh.)

Sie war nicht nur in Chemie, Mineralogie, Geometrie, Mechanik und Hydraulik versiert, sondern verfügte auch über einen großen politischen Weitblick. In ihrer ersten, 1632 erschienenen Schrift bewies sie dem König, wie er durch die Nutzung der reichen Bodenschätze seines Landes sich und Frankreich vom Ausland unabhängig machen könnte. Ihre zweite Schrift mit dem Titel „La Restitution de Pluton“, erschien 1640, war an Kardinal Richelieu gerichtet und sollte zeigen, wie der König mittels der Nutzung der Bodenschätze der reichste aller christlichen Fürsten werden und sein Volk glücklich machen könnte. Die Baronesse de Beausoleil behandelt in ihren Schriften die Wissenschaft des Bergbaus, die verschiedenen Arten von Minen, den Metallgehalt von Erzen, die Methoden des Schmelzens sowie die allgemeinen Prinzipien der Metallurgie auf dem damaligen Stand der Wissenschaft. (Mozans 1981, S.238-240).

Nicht viel besser als Hypatia werden jene Frauen behandelt, mit denen die Geschichte der abendländischen Chemie begann: die babylonischen Parfümherstellerinnen. Ich habe nur in einem der vielen wissenschaftsgeschichtlichen Nachschlagewerke, Bücher und Lexika, die ich durchgesehen habe, etwas über sie gefunden, und zwar bei Martin Levey, in dem englischen Buch „Great Chemists“ von 1961 – also wiederum *nicht* in einem deutschsprachigen Werk. In ähnlicher Weise unterschlagen die einschlägigen Chemiegeschichten auch die Leistungen Marias der Jüdin, einer der großen klassischen Alchimistinnen, deren Erfindungen immerhin bis in die heutige Zelt nachwirken.

Statt den Babylonierinnen und Maria der Jüdin wird in den meisten wissenschaftsgeschichtlichen Werken die Herausbildung der Destillation und der Alchimie hingegen einem Mann, nämlich Zosimos aus Panopolis, zugeschrieben, der etwas später als Maria ebenfalls in Alexandria lehrte, nämlich gegen Ende des 4. und Anfang des 5. Jahrhunderts. Dabei hat Zosimos – dem deutschsprachigen „Buch der großen Chemiker“ zufolge – nicht einmal „viele eigene Erfahrungen gehabt“, sondern hat „mehr durch die Sammlung und Erklärung älterer Schriften als durch die Entdeckung neuer chemischer Tatsachen" auf die Alchimie gewirkt.

Würde man also, wie Asimov, die Bedeutung eines Wissenschaftlers oder einer Wissenschaftlerin danach einschätzen, ob er oder sie etwas „Eigenständiges“ in die Wissenschaft eingebracht hat, dann wären die Verfasser(innen) der besagten „älteren Schriften“, z.B. die Alchimistinnen der ersten nachchristlichen Jahrhunderte, zweifellos als bedeutender zu bewerten als Zosimos, ihr männlicher Nachfolger, dem dennoch die wesentlichen Erfolge zugeschrieben werden.

EMILIE-GABRIELLE DE BRETEUIL (1706-1749)

Sie übersetzte Newtons Hauptwerk „Philosophiae naturalis principia mathematica“ ins Französische und versah diese noch heute einzige französische Übersetzung mit einem zusätzlichen Kommentar. In ihrem Schloß in Cirey hatte sie sich ein eigenes physikalisches Labor eingerichtet. Ihre erste naturwissenschaftliche Arbeit, eine Abhandlung über die Natur der Wärme, wurde von ihr als Lösung einer Preisaufgabe der französischen Akademie der Wissenschaften eingereicht. Emilie de Breteuil gewann zwar nicht den ausgeschriebenen Preis, doch wurde ihre Arbeit sehr gelobt und von der Akademie veröffentlicht. Für ihren Sohn verfaßte sie ein Lehrbuch der Physik mit dem Titel „Institution de Physique“, dem sie Leibniz' Theorie der lebendigen Kräfte zugrunde legte. Über dieses Buch sagte Voltaire, daß darin „eine Methode und eine Klarheit“ entwikkelt sei, „die Leibniz selbst nie besaß und deren seine Gedanken bedürfen, sei es, daß man sie verstehen will, sei es, daß man sie widerlegen will“. In diesem Lehrbuch hat Emilie de Breteuil offenbar einige spätere Schlußfolgerungen über die Natur der Energie vorweggenommen. (Mozans 1981, 5.151-153).

Hier liegt der Gedanke nahe, daß die wissenschaftlichen Leistungen der frühen Alchimistinnen - oder auch die Ihrer babylonischen Vorgängerinnen - deshalb nicht angemessen herausgestellt wurden, weil sie eben von Frauen erbracht worden sind. Diesen Verdacht, daß derart patriarchalische Überheblichkeit und Ignoranz die wissenschaftsgeschichtliche Ehrlichkeit reduziert haben könnten, wird erhärtet durch die einleitenden Sätze des Zosimos-Artikels im „Buch der großen Chemiker“. In ihnen geht der Autor sehr direkt davon aus, daß nur Männer zu technischen und wissenschaftlichen Neuerungen fähig sind. Denn mit Bezug auf die Erfindungen der Steinzelt, die Entwicklung der Keramik, die Entwicklung der Pflanzenkultivierung, des Hausbaus und weiterer technischer Errungenschaften stellt er fest: „Keine Inschrift nennt den Namen eines Mannes, dem ein Fortschritt in technischen Dingen zu verdanken ist“, und: „Kein Denkmal kündet uns den Ruhm des Mannes, der das erste Metall erschmolzen hat, den ersten Krug im Feuer gebrannt hat. „Woher weiß der Schreiber eigentlich, daß dies nicht Frauen waren?

LAURA BASSI (1711-1778)

Im Alter von 21 Jahren hat Laura Bassi in Bologna vor sieben Professoren und zwei Kardinälen mit größtem Erfolg und in elegantestem Latein ihre philosophische Dissertation verteidigt. Im selben Jahr erhielt sie an der Universität Bologna einen Lehrstuhl für Philosophie. Sie lehrte Algebra, Geometrie und Experimentalphysik und zog außerdem noch zwölf Kinder groß. Durch Ihre Vermittlung konnte Voltaire Mitglied der Akademie werden. (Mozans 1981, S.203-212, Lexikon der Frau, S.716)

Es gibt nämlich einige Hinweise darauf, daß an den ersten Entdeckungen und Erfindungen der Menschengeschichte, die allgemein als wichtige kulturelle Fortschritte bezeichnet werden, Frauen einen entscheidenden Anteil hatten, wenn sie nicht sogar deren alleinige Schöpferinnen waren. Dies z.B. bei der Entwicklung der Töpferkunst, des Hausbaus, der Landwirtschaft und der Textilproduktion. Zumindest Ernest Bornemann stellt in seinem Buch „Das Patriarchat“ diese These auf, und er führt dafür neuere Erkenntnisse der Archäologie, Geschichtswissenschaft und Anthropologie an. So geht man heute davon aus, daß die Stämme und Sippen der neueren Steinzeit (7000 bis 6000 Jahre vor Christus), die die Pflanzenkultivierung entwickelten, mutterrechtlich organisiert waren, daß sie Gemeinschaftsbesitz hatten, daß die Geschlechter im Prinzip gleichberechtigt waren, und daß sie keine Kriege führten, weshalb sie auch keine Befestigungsanlagen um ihre Siedlungen bauten.

ANNA MORANDI MANZOLINI (1716-1774)

Der Inhaberin des Lehrstuhls für Anatomie an der Universität zu Bologna Anna Morandi Manzolini sind eine Reihe wichtiger Entdeckungen dieser damals noch jungen Wissenschaft zu danken. Ihr besonderer Ruhm: Sie entwickelte die anatomischen Wachsmodelle, die den Anfang der heute in jeder Schule und Hochschule gebräuchlichen biologischen und anatomischen Modelle darstellen. Aus ganz Italien erreichten sie Bitten um Exemplare solcher Wachsmodelle, und aus vielen Städten Europas, darunter London und Petersburg, erhielt sie ehrenvolle Einladungen. (Mozans 1981, 5.35, Universitá 1981, 5.35).

Aufgrund der vorhandenen Ansätze einer Arbeitsteilung zwischen Männern und Frauen ist möglicherweise die Entwicklung der Pflanzenkultivierung das alleinige Werk der Frauen gewesen. Entsprechendes gilt für die Keramik, zu deren Entwicklung Ernest Bornemann schreibt: „Andererseits ist Keramik von äußerster Wichtigkeit, weil man erst in letzter Zeit entdeckt hat, daß es Töpferinnen gewesen sein müssen... Diese Entdeckung führte zu der Vermutung, die sich mittlerweile bestätigt hat, daß ein großer Teil der vorgeschichtlichen Werkzeuge von Frauen erfunden worden ist. Im Widerspruch zu der unter heutigen Männern herrschenden Ansicht, daß der Mann „von Natur“ größere technische Begabung als die Frau besitze und die technische Entwicklung der Menschheit deshalb allein das Werk der Männer sei, stellt es sich immer mehr heraus, daß gerade diejenigen Werkzeuge der Vorzeit, auf denen sich Jahrtausende später das industrielle Zeitalter aufbauen sollte, Erfindungen der Frau waren. Wir haben bereits vom Grabstock und der Hacke gesprochen... aber auch Spinnwirtel, Spinnrad und Webstuhl, auf denen sich die ganze spätere Textilindustrie aufbaut, sind weibliche Erfindungen, genau wie die Rohstoffe der Textilproduktion, vor allem Flachs und Baumwolle, zuerst von Frauen kultiviert und angepflanzt worden sind. Der Mann hat später vieles verbessert, aber der intellektuelle Durchbruch, die schöpferische Leistung, die eigentliche Kreativität kam niemals vom Mann, sondern von der Urproduzentin: der Frau.“

MARIA GAETANA AGNESI (1718-1799)

Sie fand bereits als Vierzehnjährige für einige Probleme der Ballistik und analytischen Geometrie neue originelle Lösungen und korrespondierte darüber mit angesehenen Mathematikern ihrer Zeit. Durch ein 1748 erschienenes, umfangreiches Lehrbuch der Analysis, das die französische Academie des Sciences ins Französische übersetzen ließ, wurde Maria Gaetana Agnesi in ganz Europa berühmt. Sie erntete Anerkennung und Begeisterung sowohl von berühmten Mathematikern und wissenschaftlichen Gesellschaften als auch von Fürsten und Regenten. Die größte Ehre wurde ihr durch Papst Benedikt XIV. zuteil, der ihr (selbst mathematisch gebildet) einen Lehrstuhl für höhere Mathematik an der Universität zu Bologna anbot. Erschüttert durch den Tod ihres Vaters schlug sie dieses Angebot jedoch aus und widmete ihr weiteres Leben den unterdrückten Schichten ihres Volkes. (Mozans 1981, S.78 u. S.143150, Kenneth/May 1975, S.716).

Wenngleich m.E. nichts dagegen spricht, daß in dieser Zeit auch Männer produktive Gedanken gehabt haben könnten, so bleibt in diesem Zitat doch ein wichtiges Ergebnis bestehen: Offenbar hatten die Frauen in der Frühzeit der Menschheitsgeschichte einen wesentlich größeren Anteil an der allgemein technischen Entwicklung als allgemein bekannt ist - ja, noch mehr: daß Frauen wichtige Erfindungen gemacht haben, die heute ganz selbstverständlich Männern zugeschrieben werden. Vielleicht hat die seitherige Situation blind gemacht für eine in diesem Sinne angemessene Geschichtsforschung? Dann gilt es, durch die Wiederentdeckung der weiblichen Technikgeschichte und der weiblichen Wissenschaftsgeschichte die Frau zu rehabilitieren.

Mme. LAVOISIER (18. Jh.)

Die Ehefrau des berühmten Chemikers Lavoisier war an der Entstehung seines für die moderne Chemie grundlegenden Werkes „Traité de Chemie“ wesentlich beteiligt. Sie machte die erforderlichen Illustrationen, sie arbeitete praktisch im Labor mit, und sie protokollierte die Ergebnisse der Experimente. Sie war es auch, die Englisch und Latein lernte, um die für die gemeinsame Arbeit wichtige fremdsprachige Literatur ins Französische zu übersetzen. Nach dem Tode ihres Mannes publizierte sie die noch von ihm projektierten „Memoirs on Chemistry“ und in ihrem Hause trafen sich auch weiterhin die angesehensten Wissenschaftler ihrer Zeit. (Mozans 1913, S.214-216).

Haben Frauen die frühe Geschichte des technischen und wissenschaftlichen Fortschritts entscheidend mitbestimmt, so ist der heute wieder

zunehmende Anteil von Frauen in Technik und Wissenschaft nicht als Moment eines „langen Marsches“ der Frauen in die Wissenschaft zu interpretieren (bzw. in die Technik), sondern als langsames Zurückgewinnen eines Terrains zu deuten, aus dem die Frauen einst vertrieben worden sind. Dabei müssen sie sich allerdings die Frage stellen, ob sie sämtliche der inzwischen stattgefundenen Veränderungen dieses Terrains akzeptieren wollen oder nicht, denn die menschenfeindlichen und umweltzerstörenden Folgen dieser Veränderungen sind für alle spürbar.

ELEONOR ORMEROD (2. Hälfte des 19. Jahrhunderts)

Sie gewann als Insektenforscherin und erster großer „economic entomologist“ internationales Ansehen. Ihre gesamte Arbeit hat sie der wirtschaftlich so wichtigen Schädlingsbekämpfung gewidmet. Als Spezialistin auf diesem Gebiet führte sie viele Forschungen durch und gab Landwirten und Gärtnern aus ihrer Gegend praktische Hinweise. Von 1877 bis 1898 verfaßte sie jährlich einen Bericht über ihre neuesten Forschungsergebnisse. Darüber hinaus erschienen mehrere Handbücher und Textbücher über Getreide-, Wald und Obstschädlinge. Führende Insektenforscher aus aller Welt, Landwirte und Regierungsbeamte ersuchten sie in täglich dutzenden von Briefen um Rat oder spezielle Informationen. Huxley sagte von Eleonor Ormerod bezüglich einer Arbeit, die sie als Mitglied eines speziellen Komitees verfaßt hatte, sie wisse mehr übers Geschäft als alle anderen zusammen. Sie war Mitglied mehrerer wissenschaftlicher Gesellschaften, doch wurde sie als Frau nie geadelt, welche Ehre einem männlichen Wissenschaftler ihrer Leistung und Bedeutung wahrscheinlich zuteil geworden wäre. Doch wurde ihr in Anerkennung ihrer Leistungen von der Universität Edinburgh der Doktortitel verliehen - womit diese konservativ-altehrwürdige Institution sich zum ersten Mal dazu durchrang, eine Frau in solcher Weise zu ehren. (Mozans 1981, S.246-252).

Verwendete Quellen

Alic, Margaret: Woman and Technology in Ancient Alexandria: Maria and Hypatia, in: Women's Studies, 1st Quart., Vol.4, Nr.3, 1981, S.305-312 (in der ORF-Sendung noch nicht berücksichtigt; siehe auch das in der Zwischenzeit erschienene Buch: Alic, Margaret: Hypatias Töchter. Der verleugnete Anteil der Frauen an der Naturwissenschaft, Zürich, Unionsverlag 1987).

Kenneth, V./May, O.: Bibliography and Research Manual of the History of Mathematics, University of Toronto Press 1975.

Jonas, Emil: Hildegard von Bingen. Deutschlands erste Naturforscherin und Ärztin, in: Illustrierte Wochenzeitung (iwz), Nr.37, 15. September 1979 (Beilage zu Schwäbisches Tagblatt/SWP).

Lexikon der Frau, 2 Bände, Zürich, Encyclios-Verlag, 1953.

Levey, Martin: [Artikel ohne Titel], in: Farber, Eduard (Hg.): Great Chemists, New York-London 1961, S.2-6.

Levey, Martin: Babylonian Chemistry, in: Osiris, Nr.12/1956, S.376-389.

Maurer, Margarete: Frauen in Naturwissenschaften und Technik, in: Wechselwirkung, Nullnummer, Januar 1979, S.35-38.

Mozans, H.J. [= John Augustine Zahm]: Woman in Science, MIT Press Edition, Cambridge/Mass.-London 1974 (Neuauflage des Werkes von 1913).

Università degli Studi di Bologna: Le cere anatomiche bolognesi del Settecento. Catalogo della mostra organizzata dall' Università degli Studi di Bologna nell' Accademia delle Scienze, Settembre-Novembre 1981, Bologna 1981.

Vetter, Betty M.: Woman Scientists and Engineers: Trends in Participation, in: Science, Vol.214, 18. Dez. 1981, S.1212ff.

Wininger, S .: Große Jüdische National-Biographie, 4 Bände.

Wisniewski, Roswitha: Frauen als Hochschullehrer, in: Mitteilungen des Hochschulverbandes, Heft 3, Bonn, Juni 1979, S.143.

Hier kritisch untersuchte wissenschaftsgeschichtliche Werke

Asimov, Isaac: Geschichte der Biologie, Frankfurt/M. 1968.

Asimov, Isaac: Biographische Enzyklopädie der Naturwissenschaften und Technik, 1973.

Bernal, John D.: Wissenschaft. Science in History, 4 Bände, Reinbek bei Hamburg 1970.

Das Buch der Großen Chemiker, 1930.

Hermann, Armin: Große Physiker. Vom Werden des neuen Weltbildes, Stuttgart 1959.

Kenneth, V./May, O.: Bibliography and Research Manual of the History of Mathematics, 1973.

Störig, Hans-Joachim: Kleine Weltgeschichte der Philosophie, Wien 1965.

Anmerkung 1983: Es mag eingewendet werden, diese Bücher seien nur zum Teil Standardwerke, insbesondere Asimov und Störig gälten als wenig fundierte Autoren; die von mir nachgewiesenen Mängel seien daher nicht der Wissenschaft anzulasten, sondern nur der schlampigen Arbeitsweise einiger ihrer Vertreter, die wirklich den wissenschaftlichen Standards entsprechenden Werke seien nicht so borniert. Dem ist zweierlei entgegenzuhalten: Zum einen prägen auch minderwertige Bücher das öffentliche Bewußtsein (um das es hier auch geht), zum anderen drücken sie durchaus Haltungen aus, die auch für die angeseheneren Wissenschaftler - und damit für unsere Wissenschaft überhaupt - typisch zu sein scheinen: Die erste Durchsicht von dreizehn weiteren (mir von Fach-Historikern empfohlenen) Werken erbringt im wesentlichen dieselben Resultate; der Nachweis dieser These im einzelnen würde den Rahmen dieser Veröffentlichung sprengen. Die oben genannten - hier untersuchten - Werke stellen m.E. eine charakteristische Stichprobe dar (eine Musterauswahl einer Universitätsbibliothek).

Anmerkung 1996: Die wissenschaftlichen und gesellschaftlichen Strukturen und Machtverhältnisse, die zur oben dargelegten Situation geführt haben, haben sich zwischenzeitlich nicht grundlegend, sondern nur graduell ein klein wenig verändert. Die Kategorie "Geschlecht" wird zumindest von einigen Wissenschaftshistoriker/innen thematisiert (siehe Meinel, Christoph/Renneberg, Monika (Hg.): Geschlecht als Thema der Naturwissenschafts-, Medizin- und Technikgeschichte [vorläufiger Titel], Stuttgart, GNT, 1996, in Druck).

Weitere Hinweise zu Publikationen und Quellen, welche das eigene Weiterforschen ermöglichen, gibt die umfangreiche 650-seitige Bibliographie der Autorin dieses Beitrages: Margarete Maurer: Frauenforschung in Naturwissenschaft, Technik und Medizin. Dokumentation und Bibliographie. Literatur, Zeitschriften, Adressen, Wien (WFV, Reihe Dokumentation, Band 6) 1993.

Drucknachweis: Die ursprüngliche erste Fassung dieses Beitrages wurde am 22.11.1982 unter dem Titel "Naturwissenschaften und Technik. Die Vertreibung der Frauen aus ihrer Geschichte" in der Radio-Sendereihe „Der Mensch und die Welt“ im ORF (I. Programm, Wien-Regional) ausgestrahlt und für den Druck umgeschrieben. Die Original-Druckfassung erschien in: Soznat. Blätter für soz.* Aspekte der Naturwissenschaften und des naturwissenschaftlichen Unterrichts, 6.Jg., Heft 3, Juni 1983, S.119-127. Der Abdruck erfolgt mit freundlicher Genehmigung der Autorin; die Zeitschrift Soznat gibt es nicht mehr. ☐

Softwareentwickler hat eine Panne

Was macht ein Software-Entwickler, wenn er vorne links einen Platten fährt?

Er wechselt den Reifen nach hinten rechts und prüft, ob der Fehler mitwandert!

zunehmende Anteil von Frauen in Technik und Wissenschaft nicht als Moment eines „langen Marsches“ der Frauen in die Wissenschaft zu interpretieren (bzw. in die Technik), sondern als langsames Zurückgewinnen eines Terrains zu deuten, aus dem die Frauen einst vertrieben worden sind. Dabei müssen sie sich allerdings die Frage stellen, ob sie sämtliche der inzwischen stattgefundenen Veränderungen dieses Terrains akzeptieren wollen oder nicht, denn die menschenfeindlichen und umweltzerstörenden Folgen dieser Veränderungen sind für alle spürbar.

ELEONOR ORMEROD (2. Hälfte des 19. Jahrhunderts)

Sie gewann als Insektenforscherin und erster großer „economic entomologist“ internationales Ansehen. Ihre gesamte Arbeit hat sie der wirtschaftlich so wichtigen Schädlingsbekämpfung gewidmet. Als Spezialistin auf diesem Gebiet führte sie viele Forschungen durch und gab Landwirten und Gärtnern aus ihrer Gegend praktische Hinweise. Von 1877 bis 1898 verfaßte sie jährlich einen Bericht über ihre neuesten Forschungsergebnisse. Darüber hinaus erschienen mehrere Handbücher und Textbücher über Getreide-, Wald und Obstschädlinge. Führende Insektenforscher aus aller Welt, Landwirte und Regierungsbeamte ersuchten sie in täglich dutzenden von Briefen um Rat oder spezielle Informationen. Huxley sagte von Eleonor Ormerod bezüglich einer Arbeit, die sie als Mitglied eines speziellen Komitees verfaßt hatte, sie wisse mehr übers Geschäft als alle anderen zusammen. Sie war Mitglied mehrerer wissenschaftlicher Gesellschaften, doch wurde sie als Frau nie geadelt, welche Ehre einem männlichen Wissenschaftler ihrer Leistung und Bedeutung wahrscheinlich zuteil geworden wäre. Doch wurde ihr in Anerkennung ihrer Leistungen von der Universität Edinburgh der Doktortitel verliehen - womit diese konservativ-altehrwürdige Institution sich zum ersten Mal dazu durchrang, eine Frau in solcher Weise zu ehren. (Mozans 1981, S.246-252).

Verwendete Quellen

Alic, Margaret: Woman and Technology in Ancient Alexandria: Maria and Hypatia, in: Women's Studies, 1st Quart., Vol.4, Nr.3, 1981, S.305-312 (in der ORF-Sendung noch nicht berücksichtigt; siehe auch das in der Zwischenzeit erschienene Buch: Alic, Margaret: Hypatias Töchter. Der verleugnete Anteil der Frauen an der Naturwissenschaft, Zürich, Unionsverlag 1987).

Kenneth, V./May, O.: Bibliography and Research Manual of the History of Mathematics, University of Toronto Press 1975.

Jonas, Emil: Hildegard von Bingen. Deutschlands erste Naturforscherin und Ärztin, in: Illustrierte Wochenzeitung (iwz), Nr.37, 15. September 1979 (Beilage zu Schwäbisches Tagblatt/SWP).

Lexikon der Frau, 2 Bände, Zürich, Encyclios-Verlag, 1953.

Levey, Martin: [Artikel ohne Titel], in: Farber, Eduard (Hg.): Great Chemists, New York-London 1961, S.2-6.

Levey, Martin: Babylonian Chemistry, in: Osiris, Nr.12/1956, S.376-389.

Maurer, Margarete: Frauen in Naturwissenschaften und Technik, in: Wechselwirkung, Nullnummer, Januar 1979, S.35-38.

Mozans, H.J. [= John Augustine Zahm]: Woman in Science, MIT Press Edition, Cambridge/Mass.-London 1974 (Neuauflage des Werkes von 1913).

Università degli Studi di Bologna: Le cere anatomiche bolognesi del Settecento. Catalogo della mostra organizzata dall' Università degli Studi di Bologna nell' Accademia delle Scienze, Settembre-Novembre 1981, Bologna 1981.

Vetter, Betty M.: Woman Scientists and Engineers: Trends in Participation, in: Science, Vol.214, 18. Dez. 1981, S.1212ff.

Wininger, S .: Große Jüdische National-Biographie, 4 Bände.

Wisniewski, Roswitha: Frauen als Hochschullehrer, in: Mitteilungen des Hochschulverbandes, Heft 3, Bonn, Juni 1979, S.143.

Hier kritisch untersuchte wissenschaftsgeschichtliche Werke

Asimov, Isaac: Geschichte der Biologie, Frankfurt/M. 1968.

Asimov, Isaac: Biographische Enzyklopädie der Naturwissenschaften und Technik, 1973.

Bernal, John D.: Wissenschaft. Science in History, 4 Bände, Reinbek bei Hamburg 1970.

Das Buch der Großen Chemiker, 1930.

Hermann, Armin: Große Physiker. Vom Werden des neuen Weltbildes, Stuttgart 1959.

Kenneth, V./May, O.: Bibliography and Research Manual of the History of Mathematics, 1973.

Störig, Hans-Joachim: Kleine Weltgeschichte der Philosophie, Wien 1965.

Anmerkung 1983: Es mag eingewendet werden, diese Bücher seien nur zum Teil Standardwerke, insbesondere Asimov und Störig gälten als wenig fundierte Autoren; die von mir nachgewiesenen Mängel seien daher nicht der Wissenschaft anzulasten, sondern nur der schlampigen Arbeitsweise einiger ihrer Vertreter, die wirklich den wissenschaftlichen Standards entsprechenden Werke seien nicht so borniert. Dem ist zweierlei entgegenzuhalten: Zum einen prägen auch minderwertige Bücher das öffentliche Bewußtsein (um das es hier auch geht), zum anderen drücken sie durchaus Haltungen aus, die auch für die angeseheneren Wissenschaftler - und damit für unsere Wissenschaft überhaupt - typisch zu sein scheinen: Die erste Durchsicht von dreizehn weiteren (mir von Fach-Historikern empfohlenen) Werken erbringt im wesentlichen dieselben Resultate; der Nachweis dieser These im einzelnen würde den Rahmen dieser Veröffentlichung sprengen. Die oben genannten - hier untersuchten - Werke stellen m.E. eine charakteristische Stichprobe dar (eine Musterauswahl einer Universitätsbibliothek).

Anmerkung 1996: Die wissenschaftlichen und gesellschaftlichen Strukturen und Machtverhältnisse, die zur oben dargelegten Situation geführt haben, haben sich zwischenzeitlich nicht grundlegend, sondern nur graduell ein klein wenig verändert. Die Kategorie "Geschlecht" wird zumindest von einigen Wissenschaftshistoriker/innen thematisiert (siehe Meinel, Christoph/Renneberg, Monika (Hg.): Geschlecht als Thema der Naturwissenschafts-, Medizin- und Technikgeschichte [vorläufiger Titel], Stuttgart, GNT, 1996, in Druck).

Weitere Hinweise zu Publikationen und Quellen, welche das eigene Weiterforschen ermöglichen, gibt die umfangreiche 650-seitige Bibliographie der Autorin dieses Beitrages: Margarete Maurer: Frauenforschung in Naturwissenschaft, Technik und Medizin. Dokumentation und Bibliographie. Literatur, Zeitschriften, Adressen, Wien (WFV, Reihe Dokumentation, Band 6) 1993.

Drucknachweis: Die ursprüngliche erste Fassung dieses Beitrages wurde am 22.11.1982 unter dem Titel "Naturwissenschaften und Technik. Die Vertreibung der Frauen aus ihrer Geschichte" in der Radio-Sendereihe „Der Mensch und die Welt“ im ORF (I. Programm, Wien-Regional) ausgestrahlt und für den Druck umgeschrieben. Die Original-Druckfassung erschien in: Soznat. Blätter für soz.* Aspekte der Naturwissenschaften und des naturwissenschaftlichen Unterrichts, 6.Jg., Heft 3, Juni 1983, S.119-127. Der Abdruck erfolgt mit freundlicher Genehmigung der Autorin; die Zeitschrift Soznat gibt es nicht mehr. ☐

Softwareentwickler hat eine Panne

Was macht ein Software-Entwickler, wenn er vorne links einen Platten fährt?

Er wechselt den Reifen nach hinten rechts und prüft, ob der Fehler mitwandert!

Spezielle Technik für Berufstätige

Seit dem Jahr 1980 bietet die HTL Wien I, Schellinggasse 13, die Speziallehrgänge „INFORMATIONS- UND AUTOMATISIERUNGSTECHNIK“ an. In dieser 6 Semester dauernden Ausbildung wird HTL-Absolventen und Personen mit technischer Vorbildung die Möglichkeit geboten, wichtige EDV-Kenntnisse zu erwerben bzw. zu erweitern.

A. Weichselberger

Nach der Schule hört das Lernen nicht auf. Spätestens im Beruf kommt man darauf, was man alles verabsäumt hat. Die höhere Technische Bundeslehranstalt (HTBLA) Wien 1 bietet daher spezialisiertes technisches Wissen in Weiterbildungskursen an. In den Lehrgängen Informationstechnik und Automatisierungstechnik haben HTL-Absolventen und Personen mit vergleichbarer technischer Vorbildung die Möglichkeit, in eine stark computerdurchsetzte Ausbildung einzusteigen. Es geht um moderne Entwicklungen in der Prozess- und Fertigungsautomatisierung bzw. um Software-Technologien, Mikroprozessor-Technik sowie Meß-, Steuer- und Regeltechnik. Englisch und Projektmanagement runden die Ausbildung ab.

- **modulares Lehrgangskonzept:** 6 einsemestrige Ausbildungsmodule mit verschiedenen Schwerpunkten, die je nach Interesse und Bedarf kombinierbar sind

ein breit gefächertes Angebot an Lehrinhalten:

- **Grundlagen:** Programmieren, Betriebssysteme, Programmieren, Grundlagen der Elektrotechnik und Elektronik, Computer Hardware, Netzwerke, Steuerungs- und Regelungstechnik, Fertigungs- und Prozessautomatisierung, Projektmanagement, Technisches Englisch, Internet
- **Software:** C++ Programmieren, DOS, Winword, Excel, Autocad,
- **Hardware:** Mikroprozessor-Technik PC Interna
- nur 3 Abende pro Woche durch Fernunterrichtsanteile
- **Aufnahmebedingungen:** erfolgreicher Abschluß einer höheren Schule; bei Abschluß einer HTL kann der Grundkurs entfallen
- die Ausbildung ist kostenlos, nur für Skripten wird ein Unkostenbeitrag eingehoben

Kursbeginn

- jeweils mit Semesterbeginn;
- nächster Termin: 4. September 1996, 18.15 Uhr

Anmeldung und Auskunft

ab sofort bis zur Einschreibung am 4. September 18.15 Uhr
HTL W I, Schellinggasse 13, 1015 Wien
Tel 51579/113
E-Mail: `lehrgang@elina.htlw1.ac.at`

□

Ausgewählte Projekte

aus den Speziallehrgängen der HLA für Elektronik am TGM im Schuljahr 1995/96

Norbert Bartos

Wie auch in den vergangenen Jahren möchte ich den interessierten Lesern dieser Zeitschrift einige Projekte durch die folgenden kurzen Artikel, welche von den ausführenden Studenten geschrieben wurden, präsentieren lassen. Die Entwicklungen fanden im Rahmen der folgenden Ausbildungsformen am TGM statt:

- Speziallehrgang für EDV und angewandte Mikroelektronik,
- Speziallehrgang für Robotiksysteme und Automation.

Das Erscheinen der Lehrgänge in der Öffentlichkeit ist zur Zeit besonders wichtig, da aufgrund der Sparmaßnahmen diese Ausbildungsform im Mai 1996 bereits als eingestellt galt. Dank verschiedener Bemühungen, unter anderem auch durch die massiven Rettungsversuche von Seiten des Direktors des TGM gegenüber dem BMUK und diverser lehrgangsinterner Umstrukturierungsmaßnahmen, sowie die Ausarbeitung und Einreichung eines den Lehrgang mitverwendenden EU-Projektes im Rahmen des LEONARDO-Programmes durch den Autor dieser Zeilen, konnte der Lehrgang gerettet werden. Leider hat sich die Personalvertretung am TGM wiederholt **gegen eine Fortführung des Lehrganges** ausgesprochen. Die Begründungen dafür sind leider weder in pädagogischer Hinsicht stichhaltig, noch dem Ansehen des TGM förderlich. Ich möchte behaupten, daß in dieser Frage die Mehrheit der vertretenen Lehrkräfte anderer Meinung ist. In Zeiten des Sparens dürfen natürlich die Kernbereiche der technischen Ausbildung nicht angetastet werden. Vielmehr sollten diejenigen zusätzlichen Bildungsangebote aufgelassen werden, welche für die Erhaltung der Qualität der fachlichen Ausbildung keinen wesentlichen Beitrag liefern.

Nach diesen kritischen Bemerkungen wenden wir uns, versehen mit einem gehörigen Maß an Optimismus für die Zukunft, den in den nachfolgenden Artikeln detailliert vorgestellten Bereichen zu. Es sind dies die Projekte

- Hybrides neuronales Netz zur Aussprache handschriftlicher Worte
- UNISTEB - Ein universelles Steuer- und Bediengerät für die Fernsehtechnik
- Audiomischpult
- Bodenkontaktsohle
- Freisprechanlage für Autotelephon
- Datenflußrechnersystem

Sollten Sie Interesse an einer Zusammenarbeit mit uns oder Fragen zu den Projekten haben, so wenden Sie sich bitte an diejenige Person, welche unter den Autoren zuletzt genannt wird. Es ist dies der zuständige Projektleiter am TGM, dessen Daten Sie bitte der Autorenseite der Zeitschrift entnehmen wollen.

Wir sind auch für Rückmeldungen per Brief oder einfach formlos per E-Mail sehr dankbar, welche die Bedeutung der Speziallehrgänge für die Industrie bestätigen. Sie können als Argumentationshilfe dienen, wenn in der nächsten Zeit der Sparstift wiedereinmal in Aktion tritt.

Abschließend wünsche ich Ihnen eine kurzweilige Lektüre der Artikel. im Schwerpunktteil in diesem Heft. □

Computer- und Systemtechnik

an der FhE am TGM

Norbert Bartos

Der Fachhochschul-Studiengang Elektronik am TGM

Seit dem Schuljahr 1994/95 befindet sich im Gebäude des TGM der 4jährige *„Fachhochschul-Studiengang Elektronik"*. Er ermöglicht den Absolventen durch eine kompakte, effiziente und praxisorientierte Ausbildung die Erlangung des akademischen Titels *„Diplomingenieur/in für Elektronik FH (Dipl. Ing. FH)"*. Diese Ausbildung ist durch das Fachhochschulstudiengesetz als gleichwertig, aber nicht gleichartig zur universitären Ausbildung festgelegt.

Die Unterstufe des Studienganges (1. bis 4. Semester) vermittelt ein solides technisches und betriebliches Grundwissen. In der Oberstufe (5. bis 8. Semester) spezialisiert sich der Absolvent auf einen bestimmten Fachbereich. Dabei stehen folgende Fachbereiche zur Auswahl:

- Bio-Medizintechnik
- Meß-, Steuer- und Regeltechnik
- Telekommunikationstechnik
- Leistungselektronik und Energietechnik
- Computer- und Systemtechnik
- Mikroelektronik
- Consumer Electronics

Das 5. und 6. Semester dient im Wesentlichen zur Aneignung des Wissens aus dem gewählten Bereich. Im 7. Semester arbeitet der Absolvent an einem Projekt seines Fachbereiches in einer Partnerfirma der FhE. Dieses Projekt wird im 8. Semester als Diplomarbeit aufbereitet. Danach wird durch die kommissionelle Diplomprüfung die Ausbildung abgeschlossen.

Dieser Artikel soll keine detaillierte Beschreibung der gesamten Fachhochschulausbildung sein, sondern eine der erwähnten Fachbereiche, nämlich die *„Computer- und Systemtechnik"*, im Folgenden kurz CST genannt, näher vorstellen.

Der Fachbereich „Computer- und Systemtechnik"

Derzeit besteht der Fachbereich CST aus einem Stammprofessor (dem Autor dieses Artikels) und vier Lehrbeauftragten. Der Fachbereich setzt sich zur Aufgabe, die Absolventen mit den wesentlichen Aspekten der technischen Informatik vertraut zu machen. Durch die hohe Innovationsrate dieses Fachgebietes und die intensive Verwendung englischer Fachbegriffe ist es für den Techniker notwendig, die dahinter stehenden Grundprinzipien zu kennen. Liest man heute in einer beliebigen technischen Zeitschrift beispielsweise einen Artikel über einen neuen Prozessor, so tauchen unweigerlich Begriffe wie „Pipelining", „Harvard-Architektur", „Superskalarer RISC" und ähnliches auf, ohne daß diese näher erklärt werden. Für einen durchschnittlichen HTL-Absolventen ist ein derartiger Artikel nicht mehr auf Anhieb verständlich, da solche gehobenen Techniken im Rahmen der Ausbildung meist keinen Platz finden. Wesentliches Ziel der CST ist daher, zu erreichen, daß die Absolventen in der Computertechnik „mitreden" können, wenn es um aktuelle Technologien und zukünftige Entwicklungstrends geht. Nach diesen Kriterien wurde auch der Lehrplan vom Autor dieses Artikels erstellt.

Der Inhalt des Fachbereiches

Der Fachbereich ist dermaßen umfangreich, so daß nur zwei Möglichkeiten bleiben. Die erste Möglichkeit besteht in der detaillierten Behandlung einiger ausgewählter Bereiche der Informatik. Die zweite Möglichkeit ist eine enzyklopädisch gehaltene Vorlesung, wobei bei aktuellem Bedarf, durch die Praxissituation oder die Absolventen, natürlich eine punktuelle Vertiefung durch Seminare möglich ist. Da als Ziel die Vermittlung umfassenden Wissens gesetzt wurde, was auch den Absolventen größere Freiheiten bei der Berufswahl erlaubt, wurde die letztere Variante gewählt. Es werden daher die folgenden Themen behandelt:

a) Mikroarchitektur (Architektur einer CPU; Design moderner Prozessoren)

- **Von Neumann-Systeme und deren Modifikationen:**
  Standardarchitekturen, Koprozessoren, Signalprozessoren, Pipeline-Systeme, Harvard-Architektur, Superskalar-Prozessoren, Vektorprozessoren, Very Long Instruction Word-Systeme
- **Non-Von Neumann-Systeme:**
  Datenflußrechner, Systolic Arrays, Wavefront Arrays, Neurocomputer, Optische Computer

b) Makroarchitektur (Architektur von Computersystemen)

- **Statische Multiprozessorsysteme:**
  einfaches Multiprocessing, massiv parallele Systeme; Arrays, Trees, Hypercubes und andere Architekturen
- **Dynamische Multiprozessorsysteme:**
  Crossbar-Systeme, Multistage Systeme; Routing im System

c) Algorithmen

- **Sequentielle Algorithmen:**
  sequentielle Algorithmen für technische Anwendungen; Aufwandsschätzung; Optimierung von Algorithmen
- **Parallele Algorithmen:**
  parallele Algorithmen für technische Anwendungen; Parallelisierung sequentieller Algorithmen; Synchronisationsprobleme; Load Balancing; Genetische Algorithmen

d) Artificial Intelligence

- **Regelbasierte Systeme**
  Aussagenlogik, Prädikatenlogik und ihre Erweiterungen, Wissensrepräsentation, Lernen, Expertensysteme und Anwendungen, Programmiersprachen für Expertensysteme (PROLOG, LISP), Fuzzy Logic, Fuzzy Prozessoren

- **Konnektionistische Systeme**
  Medizinische Grundlagen neuronaler Netze, Modellierung, Neuronen- und Netzarchitekturen, Lernalgorithmen, Software-Implementation, digitale und analoge Hardware-Implementation, Neuroprozessoren

Die Praxisarbeitsplätze

Zumal der Computer heute in alle Bereiche der industriellen Anwendung eingedrungen ist, ermöglicht ein detaillierteres Wissen aus diesem Bereich, in jeder Sparte der Elektronik ein besseres Bewältigen der praktischen Anforderungen im täglichen Berufsleben. Das Wissen aus dem Modul CST kann daher universell eingesetzt werden. Das spiegelt auch die Struktur der Praxisarbeitsplatzangebote für die Diplomarbeit wider. Reges Interesse zeigen hierbei nicht nur die Großkonzerne der Elektro- und Elektronikindustrie, sondern auch Klein- und Mittelbetriebe, sowie Banken und Energieversorgungsunternehmen. Die Arbeitsbereiche reichen von der Realisation kaufmännischer Software und Datenbanken, über Bildverarbeitung und Sprachverarbeitung (Hardware und Software), sowie Expertensysteme bis hin zum Neurocomputing. Den Absolventen steht somit ein reichhaltiges, vielfältiges und zukunftsträchtiges Tätigkeitsgebiet offen.

Sollten Sie technische oder organisatorische Fragen zum Fachbereich haben, so wenden Sie sich bitte an den Fachbereichsleiter:

Prof. Norbert Bartos
TGM/FhE Zimmer 1426
Tel.: (0222)-33 126-341
Fax.: (0222)-33 126-204
E-Mail: bartos@email.tgm.ac.at

Gewinner des Mathematik- Wettbewerbs

Danja Stiegler

Unsere Firma, Pablitos Software, veranstaltete vor einiger Zeit (siehe PCNEWSedu-47, Seite 47) erstmals einen Mathematikwettbewerb an allen höheren Schulen. Unsere Hauptintentionen waren einerseits, das Interesse für Mathematik in den Schülern zu wecken und andererseits, Ihnen Mathcad, ein Programm der amerikanischen Firma Mathsoft, das auf übersichtliche Weise anspruchsvolle mathematische Probleme löst, vorzustellen.

Der Wettbewerb stieß auf großes Interesse bei den Schülern. Aus über 530 Einsendungen wurden 28 Gewinner gezogen. Der Hauptqewinn, eine Reise nach London für 2 Personen ging an die 16jährige ***Sabine Leiner***, Schülerin des Gymnasiums Schwechat.

Wir sprachen mit Sabine vor ihrem Abflug vom Flughafen Graz Thalerhof, wo auch das Photo aufgenommen wurde. Sie freute sich schon sehr auf die Reise: *„Das ist mein erster Flug! Ich war auch noch nie in London! Dieser Gewinn ist eine ganz neue Erfahrung für mich!“* Bei einem Telephongespräch nach ihrer Rückkehr verriet sie uns, daß sie London, vor allem die im Programm inkludierte Themse-Rundfahrt, faszinierend fand.

Neben der Reise wurden auch 2 Vollversionen von Mathcad PLUS 6.0 und 25 Mathcad 6.0 Studentenversionen verlost.

Die beiden Vollversionen gingen an ***Florian Pichler***, 4. Klasse BN der HTL-Braunau, und an ***Daniela Karl***, ehemals 8.B-Klasse des Gymnasiums Dachsberg. Daniela, die im Juni ihre Matura erfolgreich bestanden hat, versicherte uns, sie könne Mathcad auch weiterhin verwenden, und Florian möchte Mathcad in der Schule für technische Berechnungen einsetzen.

Die 25 Gewinner der Mathcad-Studentenversionen sind:

- ***Margarete Rebhan***, HBLA Elmberg
- ***Karin Spitzbart***, Mädchenpensionat Gmunden
- ***Ingo Türr***, Borg Lauterach
- ***Sonja Bajic***, Borg Lauterach
- ***Hildegard Zehentner***, HIB Saalfelden
- ***Susanne Brandstätter***, HIB Saalfelden
- ***Barbara Hinterhoelzl***, HBLA Elmberg
- ***Philipp Stampfl***, HTBLA Braunau
- ***Karina Forsich***, Borg Oberpullendorf
- ***Sabine Dollmann***, Borg Radstadt
- ***Barbara Welsch***, Bundesbildungsanstalt f. Kindergartenpäd.
- ***Claudia Perlega***, HBLVA f. chem. Ind., Wien, Rosensteingasse
- ***Sabine Wagner***, Studienheim Mariahilf, Unterwaltersdorf
- ***Claudia Pölderl***, HBLA Linz Auhof
- ***Reingard Peyrl***, BG/BRG Freistadt
- ***Michael Kreilmeier***, BRG Linz Landwiedstraße
- ***Barbara Jäger***, Borg Radstadt
- ***Christina Grubich***, BG/BRG Oberpullendorf
- ***Christian Unger***, BG/BRG Neusiedl/See
- ***Gertraud Pirklbauer***, HAK Rohrbach
- ***Bettina Fuchs***, HAK Auhof
- ***Anita Sillipp***, BHAK Zwettl
- ***Johann Bulfone***, TGM Wien Wexstraße
- ***Edith Jauk***, BG Rein
- ***Silvija Marinovic***, BG/BRG Schwechat

Für weitere Informationen stehen wir Ihnen unter der Nummer 03117-3251 gerne zur Verfügung. □

EUREKA Multimedia Umbrella

Offizieller Start und Aktionen.

OCG

EUREKA hat unlängst ein neues „Umbrella Programm“ im Bereich Multimedia genehmigt, den *„EUREKA Multimedia Umbrella“*.

Österreich ist in Brüssel bei dieser Aktion administrativ durch Dkfm. Herwig Renner vom BIT in Wien (Büro f. Internationale Forschungs- und Technologiekooperation) vertreten, und wissenschaftlich durch Professor Dr.Dr.h.c. Hermann Maurer vom IICM in Graz (Institut für Informationsverarbeitung und Computergestützte neue Medien).

Umbrella Programme dienen zur Förderung der Kooperation zwischen Organisationen aus den EUREKA Mitgliedsländern (d.s. die Staaten der EU, der ehemaligen EFTA assoziierte Staaten wie Israel und Kanada). Projekte von mindestens zwei Organisationen in zwei verschiedenen EUREKA Ländern können bei einer nationalen EUREKA Vertretung eingereicht werden, in Österreich z.B. beim BIT, wobei eine vorherige Abstimmung mit den nationalen Vertretern oft sinnvoll sein wird.

Projekte werden dann von EUREKA begutachtet und können, falls befürwortet, mit der EUREKA-Empfehlung (die also eine Art „Qualitätsstempel“ ist) bei nationalen Forschungsorganisationen eingereicht werden.

Als für Österreich zuständiger wissenschaftlicher Vertreter des EUREKA Multimedia Umbrella ist es u.a. Maurer's Aufgabe, eine möglichst vollständige Liste aller jener Organisationen in Österreich zu erstellen, die sich mit multimedialen Produktionen beschäftigen oder sich dafür in Zukunft interessieren werden, damit diese Organisationen von wichtigen EUREKA Ereignissen informiert werden können. Die Aktivität wird in den Arbeitskreis „Hypermediasysteme“ der OCG (Österreichische Computer Gesellschaft) eingebunden.

Insbesondere wird im Herbst 96 in Graz eine erste Veranstaltung „EUREKA Multimedia Umbrella“ stattfinden, die alle Interessenten zusammenbringt, und wo prominente Fachleute Chancen, Entwicklungen und Probleme bei multimedialen Entwicklungen erläutern werden.

Alle Organisationen, die sich im weitesten Sinne mit multimedialen Entwicklungen beschäftigen bzw. beschäftigen wollen sind hiermit herzlichst eingeladen eine Anlaufperson mit allen notwendigen Parametern (Name, Adresse, Telefon, Fax, Email) zu benennen und zu senden an:

Prof. Hermann Maurer, IICM, Schießstattgasse 4a, 8010 Graz , Tel.: (0316) 873-5612 Fax: (0316) 82 43 94, Email: hmaurer@iicm.edu, WWW: http://www.iicm.edu.

Weitere Informationen zum EUREKA Multimedia Umbrella allgemein von Dkfm. H. Renner, BIT, Wiedner Hauptstr.76, A-1040 Wien, Tel.: (01) 581-1616-0, Fax: 581-1616-16.

Lernsoftware für die Kleinsten

SCHUBI Learnsoft - eine systematisch aufgebaute Lernmedien-Serie

Bärbel Pöch

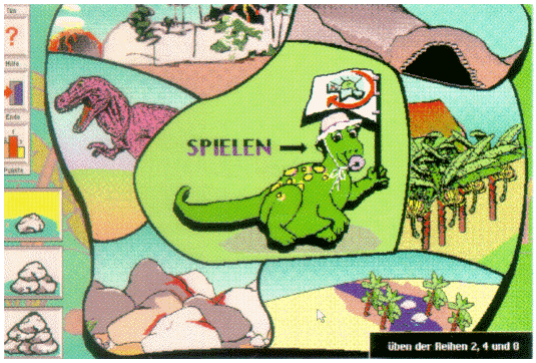
In der letzten Ausgabe der PCNEWSedu habe ich einige der Spiele vorgestellt, die die Firma PABLITOS in ihr Angebot ausgewählter Spiele und Lernsoftware aufgenommen hat. Die Liste unserer Selected Games wird ständig erweitert! Heute möchte ich Ihnen einige Lernprogramme für die Kleinsten kurz beschreiben: SCHUBI Learnsoft! SCHUBI Learnsoft ist eine systematisch aufgebaute Lernmedien-Serie für die Förderung von kleinen Schülerinnen und Schülern, vom 2. bis zum 6. Schuljahr. Die aufwendig und sehr liebevoll und ansprechend gestalteten Programme beinhalten Rechenaufgaben wie Zu- und Wegzählen, Ergänzen, Vermindern, Malrechnen, Teilen, Kettenaufgaben, oder, für die Größeren: Rechnen mit Geldsorten, Zeitmaßen, Gewichten, Längenmaßen...

Der Schwierigkeitsgrad der Aufgaben läßt sich einstellen, und es ist möglich, die Lösungen mit oder ohne Zeitvorgabe zu suchen. Erfolgskontrolle durch einen mitlaufenden Report ist gegeben.

Folgende Programme sind bei PABLITOS erhältlich:

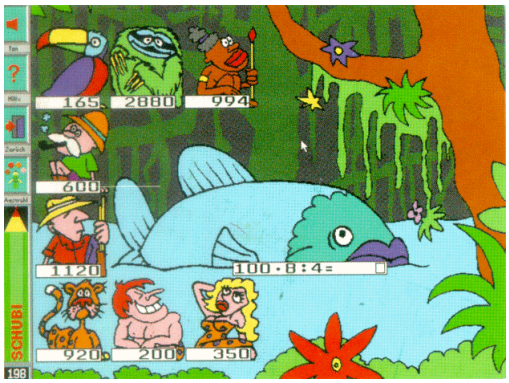
Abenteuerreise

Rechnen mit Maßen (2.-6. Schuljahr)



Im Urwald

Rechnen bis 10 000 (4. Schuljahr)

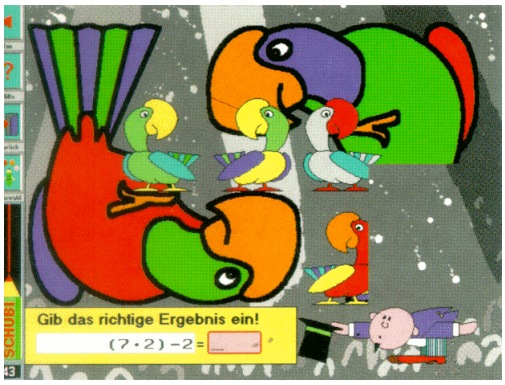


Multiplo

1x1 Übungsprogramm (2. Schuljahr) (keine Abbildung)

Im Zirkus

Rechnen bis 100 (2. Schuljahr)

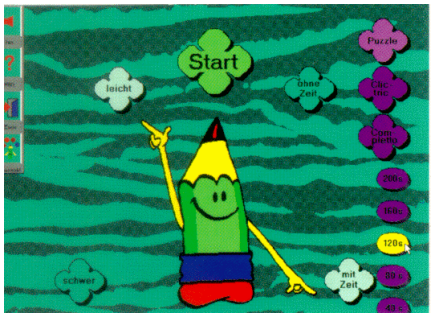


Auf dem Rummelplatz

Rechnen bis 1000 (3. Schuljahr)

Im Weltraum

Rechnen mit Brüchen und Dezimalzahlen (5./6. Schuljahr)



Zu jedem Programm gibt es auch die Möglichkeit, eine Klassenraumlizenz zu erwerben. Für nähere Informationen rufen Sie uns einfach an! Wir stehen Ihnen gerne für Fragen zur Verfügung:

PABLITOS Software GesmbH.
A-8063 Eggersdorf bei Graz, Edelsbachstr. 52,
Tel. 03117-3251, Fax 03117-3251-90
E-Mail: `office@pablitos.co.at`
WWW: `http://www.pablitos.co.at/pablitos/`
Ihre Ansprechpartnerin für den Bereich „Selected Games“: Bärbel Pöch, Psychologin. □

Die Beiträge auf den nächsten Seiten wurden im Rahmen des PABLITOS-Spiele-Gewinnspiels (PCNEWSedu-49, Seite 33) eingesendet. Die Zahlen im Titel sind Hinweise auf den jeweiligen Gewinn und sind keine Wertung.

NEANDERTAL ❸+❹

Katharina Simon

1. Es handelt sich um einen Steinzeitmenschen, der mit seiner Keule die Höhlenbären und Dinosaurier auf seinem Weg zum Ziel zusammenschlagen muß. Dafür bekommt man Punkte.
2. Es gibt auch Rieseneier. Wenn man auf die Rieseneier 3x klopft, zerspringen sie und bringen Punkte. Man muß über Schluchten springen und vor einem riesigen See, aus dem Fische springen, den Kokosnuß werfenden Affen ausweichen. Über den See gelangt man mit einem Luftballon. Hin und wieder kommen auch Vögel entgegen, denen man ausweichen muß. Manchmal liegen auch spitze Steine im Weg. Denen darf man nicht zu nahe kommen, weil man sonst eines der drei Leben verliert, die man von Anfang an bekommt.
3. Das Spiel Neandertal gefällt mir so gut, weil darin nicht geschossen wird, sondern nur die Keule geschwungen wird. Außerdem ist es so lustig gezeichnet, daß der Bär und der Dino nach ein paar Schlägen am Boden liegen und Sternchen um den Kopf flattern.

Paulie Python

Erfinder des Spieles: Craig P. Thompson

Florian Simon

1. Worum geht es?
Es geht um eine Schlange, die möglichst viele Mäuse fressen muß.

2. Wie wird es gespielt?
Man hat am Anfang drei Leben. Wenn man an einem Stein anstoßt, hat man ein Leben weniger.

Die zweite Welt ist noch gemeiner! Dort gibt es weiße Kugeln. Wenn sie einem am Körper treffen, dann passiert nichts. Wenn sie einem am Kopf treffen, dann bekommt man auch ein Leben weniger.

3. Und warum gefällt es mir so gut?
Weil ich es so gut kann und es macht auch sehr, sehr großen Spaß.

Valley

Stefan Simon

Es handelt sich um einen Goldgräber, der möglichst viele Goldstücke sammeln muß. Es gibt auch Spinnen und Steine. Wenn man an denen anstößt, ist man kaputt und man muß mit dieser Welt von vorn anfangen. Man hat auch Bomben und Dynamitstangen.

SOLITÄR

Ich habe bereits eine Menge an Spielprogrammen installiert, durchgesehen und beschrieben, oft auch kategorisiert. Der Einfachheit halber ist jenes Spiel, daß ich mir als einziges zur Entspannung immer wieder gebe.

Gerd Simon

1. Es geht darum, Karten aus verschiedenen Kartenpaketen so umzuschlichten, daß letztendlich alle Karten aufsteigend sortiert abgelegt sind.
2. siehe Microsoft-Hilfe Windows 3.11
3. Nachdem der Algorithmus nicht immer terminiert, wenn man in Dreierpaketen aufschlägt, ist dadurch ein gewisser Reiz gegeben. Geht sich das Spiel aus und alle Karten werden über den Bildschirm verstreut, freut es mich. Im anderen Fall kommt eine Zeitschranke, d.h. nach etwa 15 min ruft wieder die Arbeit.

Der kleine Tiger braucht ein Fahrrad ❷

Markus Dietinger

Der kleine Tiger braucht ein Fahrrad ist die CD-ROM-Umsetzung des gleichnamigen Buches (ISBN 325 70 7337). Das Programm besteht aus den Teilen

- Interaktives Buch und
- Lernspiele rund um die Verkehrssicherheit.

Dieses Programm hat einen sehr hohen Lehrwert bei Kindern im Alter zwischen fünf und neun, kann aber auch so manchen Vater in den Bann ziehen.

Im Laufe der Geschichte lernt der Spieler alle Bestandteile eines „Verkehrssicheren Fahrrads“ kennen und bekommt grundlegende Verhaltensweisen im Straßenverkehr vermittelt.

In der Geschichte lernt der kleine Tiger den Hund Bonzo Schmidtchen kennen, welcher ohne Helm und in Unkenntnis der Verkehrsregeln auch zwei Unfälle hat, die aber - Gott sei Dank - alle glimpflich ausgehen. Ich glaube, daß diese Begebenheit den Kindern vor Augen führt, was passieren kann, wenn man sich im Straßenverkehr falsch verhält oder keinen Helm verwendet.

Im Laufe der Geschichte hat man immer wieder die Möglichkeit, kleine Spielchen zu spielen:

- Bilder anmalen
- Puzzle
- Verkehrsschilder lernen
- Memory
- Ampelfarben zuordnen usw.

Die Bedienung des Spieles ist SEHR EINFACH und wird am Anfang erklärt. Es werden alle Symbole und ihre Auswirkungen vom Erzähler erklärt. Am Programmbeginn hat man die Möglichkeit, entweder de Spielteil oder die Erzählung (auch mit kleinen Spielchen) auszuwählen.

Positiv hervorzuheben ist die hervorragende Grafik und die originellen Stimmen der Erzähler.

Das Programm benötigt nur 2MB Festplattenspeicher und läuft von der CD. Auch werden keine allzu großen Anforderungen an die Hardware gestellt. Das Spiel läuft wunderbar auf einem 486/66.

Einziges Manko des Programms ist die fehlende Speichermöglichkeit, um wieder an den zuletzt gehörten Teil des Spieles zurückzukommen. ☐

Die Siedler2 (Veni, vidi, vici)

1

Blue Byte konnte mit 200000 verkauften Exemplaren von „Die Siedler“ einen großen Erfolg feiern. Ist „Die Siedler2“ ein würdiger Nachfolger und kann es an den großen Erfolg seines Vorgängers anknüpfen? Diese und viele andere Fragen werde ich versuchen zu beantworten.

Ali Eghdamian

Das Spiel

Die Geschichte beginnt damit, daß man nahe einem unbekannten Land Schiffbruch erleidet. Die Vorgeschichte wird in einem Intro gezeigt. Mit Hilfe von treuen Gehilfen müssen Sie versuchen, auf der Insel zu überleben und sich gegen die anderen Völker zu behaupten. Sie können entweder eine Kampagne starten oder eines der 10 vorgefertigten Szenarien spielen. Im freien Spiel kann man aber auch die Bündnisse, den Lagerbestand und die Völker selbst einstellen. Es gibt insgesamt drei gegnerische Völker: Asiaten, Afrikaner und Wikinger.

Im Kampagne-Modus spielen sie als Römer, aber im freien Spiel können sie mit dem Volk, welches ihnen am sympathischsten scheint, spielen. Man muß versuchen, mit den Vorräten im Lager eine möglichst große und effektive Wirtschaft zu bauen. Hat man dies einmal geschafft, kann man beginnen, falls man genügend Rohstoffe hat, Schwerter und Münzen zu produzieren. (Mit Schwertern kann erst ein Gehilfe zu einem Soldaten rekrutiert werden, Münzen sind für die Ausbildung sehr wichtig). Man sollte aber die Schwerter-Produktion am Anfang nicht zu sehr vernachlässigen, da man später keine Soldaten zum Besetzen von Militärgebäuden hat.

Dies alles klingt sehr einfach, aber wenn man „Die Siedler2“ genauer betrachtet, merkt man, wie gut dieses Spiel durchdacht ist. Zum Beispiel: Wenn man Schwerter produzieren möchte, braucht man eine Schmiede. Um diese wiederum zu bauen, braucht man Bretter und Steine. Die Bretter müssen erst angefertigt werden (sprich: Holz muß gefällt werden), also braucht man einen Holzfäller. Falls es in der Nähe keinen Waldbestand gibt, muß man ein Forsthaus bauen (wieder werden Bretter benötigt). Hat man endlich das Holz gefällt, braucht man ein Sägewerk (benötigt Bretter und Steine), um es zu verarbeiten. Das allein muß berücksichtigt werden, wenn man eine Schmiede baut. Die Schmiede verwendet aber auch Rohstoffe, um Schwerter herstellen zu können. Die Rohstoffe bekommt man aus den Bergwerken. Hat man ein Kohlebergwerk und ein Eisenbergwerk gebaut, kann man beginnen, die Rohstoffe abzubauen. Natürlich ist ein Eisenklumpen allein für einen Schmied wertlos, deshalb muß man eine Eisenschmelze bauen, um den Klumpen in Stahl zu verarbeiten. Die Arbeiter in den Bergwerken arbeiten natürlich nicht ohne Nahrung, welche von Schweinehöfen, Fischerhütten, Jagdhäusern, Bäckereien produziert wird. Um Brote zu backen benötigt man Mühlen, Brunnen, Bauernhöfe, usw. (Die Aufzählung vieler weiterer Werkzeuge möchte ich mir ersparen. Meine Finger können bald nicht mehr, Gnade!)

Das Spiel klingt nach der ausführlichen Beschreibung vielleicht etwas verwirrend, doch das Setzen von Gebäuden dauert nicht länger als drei Sekunden. Hat man das Gebäude durch Wege mit dem Hauptquartier verbunden, setzen sich kleine Siedler in Bewegung und beginnen zu bauen. Zuerst wird das Holzgerüst von dem Gebäude aufgebaut, welches dann vervollständigt wird. Ihre Gehilfen sind (fast) immer in Bewegung. Sie können ihre Siedler entweder im Beobachtungsfenster (sehr hilfreich, wenn man woanders gebraucht wird) oder im Vollbild beobachten: zum Beispiel den Bauer, wie er Samen auf einem Feld aussät, wie der Holzfäller die Bäume fällt, der Förster die Bäume pflanzt, oder wie einfache Gehilfen die Waren von Fähnchen zu Fähnchen transportieren. Apropos Waren: Sie sollten versuchen, einen Weg mit möglichst vielen Fähnchen auszustatten, da diese den Transport erheblich beschleunigen. Die Wegebau-Option wurde beibehalten, doch die Programmierer haben hier sich auch etwas Neues einfallen lassen. Man kann entweder die Wege manuell bauen, oder sich vom Computer automatisch bauen lassen (erspart viel Zeit und Nerven). Je öfter eine Straße von Siedlern benutzt wird, desto höher ist die Wahrscheinlichkeit, daß diese asphaltiert wird. Auf asphaltierten Straßen können sie auch Esel benutzen, welche den Transport einer Ware erheblich erleichtern. Mit insgesamt 9 neuen Gebäuden kann „Siedler2“ auftrumpfen. Unter anderem: Jagdhaus (für Nahrung), Katapult (extrem wichtig, dazu später), Eselszucht (für Transport), Brauerei (für Soldaten)... Das Militärsystem wurde ausgebaut. Man muß jetzt nicht mehr das Gebäude erobern (wie in der Vorgängerversion), sondern kann es mit Hilfe des Katapults zerstören. Pro Treffer wird von dem Militärgebäude ein Soldat eliminiert. Es wird das Militärgebäude solange beschossen, bis keine Soldaten mehr stationiert sind und das Gebäude in Flammen steht. ACHTUNG! Nicht-besetzte Militärgebäude werden nicht beschossen! Eine beliebte und erfolgreiche Taktik besteht darin, nahe den gegnerischen Gebäuden Katapulte zu bauen, und sie so, ohne militärische Verluste, zu zerstören.

Wie beim ersten Teil kann man „Die Siedler2“ auch zu zweit an einem Computer (mit zwei Mäusen) mittels Splitscreen spielen.

Eigene Meinung

Würde ich sagen, mir hätte dieses Spiel sehr gut gefallen, wäre es eine Untertreibung. Ich bin ganz begeistert von diesem unglaublich schönen und fesselnden Spiel. Obwohl sich gegenüber dem Vorgänger nicht sehr viel geändert hat, haben die Programmierer von Blue Byte die Grafik (die damals schon ein Hit war) verbessert, welche leicht mit dem heutigen Standard mithalten kann. Blue Byte hat bewiesen, daß man auch ohne stundenlange Videosequenzen und speicherfressenden 3D Kurzfilmen auskommen kann. Der Sound ist gut, die Grafik ist super, das Spiel macht unglaublich Spaß, und noch dazu um nur öS 490.- bei Software Dschungel zu bekommen - was will man mehr? Für mich ist „Siedler2“ eines der besten Strategie/Wirtschaftsimulationen auf dem Markt.

Also mein Rat: Geld schnappen und sofort kaufen! Dieses Spiel sollte jeder probieren, ob Strategie-Fan oder nicht. Ich bin sicher, „Die Siedler2“ kann jeden begeistern.

Bewertung

Grafik: 91% Wie beim Vorgänger ist die Grafik hervorragend. Man kann zwischen drei Auflösungen im VGA oder SVGA Modus wählen, und selbst bei einem 486 DX/2 66 MHz, 8 MB RAM, läuft das Spiel ruckelfrei. Jede einzelne Grafik und Animation ist liebevoll gezeichnet und bereitet eine große Freude beim zusehen: zum Beispiel: flatternde Fahnen im Wind, die brennenden Häuser, Non-Player-Characters (Rotwild), die eigenen Siedler (bis zu 28 verschiedene Charaktere). Ein weiteres Plus sind die eigenen Gebäude für jedes Volk. Die japanischen Gebäude sind verziert mit verspielten Türmchen, im Gegensatz zu afrikanischen Gebäuden, die mit Stroh eher robust gebaut sind.

Sound: 73% Der Sound ist gut, obwohl man nach einigen Stunden den Sound abstellen will. Hat man dies getan, vermißt man das Hämmern und das Klopfen, und schaltet den Sound wieder ein.

Spielsteuerung: 81% Die Spielsteuerung ist sehr gut. Das Einzige, was mich störte war, daß wenn man den Mausknopf betätigte und die Maus zu schnell wegbewegte, die Aktion nicht ausgeführt wurde. Das Handbuch selber ist sehr informativ, da dort alle Gebäude genauer erklärt werden. Für Siedler-Veteranen wirft „Die Siedler2“ keine Probleme auf, bei Siedler-Neulingen treten kleine Unklarheiten auf, die schnell im Laufe des Spiels selber beseitigt werden.

Spielspaß: 90% Das Spiel macht unglaublich viel Spaß. Selbst nach mehreren Stunden Spielzeit möchte man nicht aufhören. Die Nacht, bzw. die Stunden, vergehen schneller als man denkt!

Bugs: 13% Das Spiel ist erstaunlich bugfrei, aber da das Spiel 8 MB RAM braucht, muß man oft die *autoexec.bat* und *config.sys* umschreiben. Hat man endlich das Spiel zum Laufen gebracht, steht vielen ereignisreichen und aufregenden Stunden nichts mehr im Wege. Das einzige Negative ist, daß, wenn „nur“ 8MB RAM zur Verfügung stehen, man nicht ein großes Szenario mit vielen verschiedenen Gegnern wählen kann. Bei einem kleinen Szenario kann man gegen jedes Volk spielen. Sofern es bei einem größeren Szenario mit dem freien Speicher knapp wird, muß man mit mehreren Gegnern vom gleichen Volk spielen.

Gesamt: 88% „Die Siedler2“ ist ein Spiel, daß jeder kennen sollte. Es ist eines der besten Strategiespiele überhaupt. Stundenlang sitzt man vor dem Computer und spielt „Die Siedler2“. Wenn man anfängt, kann man damit nicht mehr aufhören. Mit diesem Spiel kann man sich stundenlang beschäftigen ohne daß es an Spielspaß verliert oder Langeweile auftritt. Blue Byte ist mit „Siedler2“, wie beim Vorgänger, ein Volltreffer gelungen!

PS: Ich möchte mich bei diesen Personen besonders bedanken, die mir das Testen durch ihre Hilfe ermöglichten: Herrn Dipl. Ing. Franz Fiala, der sich mit Software-Dschungel in Verbindung setzte, Herrn Mag. Werner Krause für Diverses, meinem Freund Andreas Fritsch für seine hilfreiche Unterstützung in technischen Belangen, meinem Vater, der mir, nachdem mein altes 2Speed-CD-Rom-Laufwerk kaputt ging, ein neues 6Speed-CD-Rom-Laufwerk kaufte, Herrn Günther Goll (SOFTWARE-DSCHUNGEL), der mir dieses Spiel (Siedler2) für zwei Wochen zur Verfügung stellen konnte. □

Die Fugger2

1

Fugger2 ist neben „Die Siedler2“, eines der neuesten/besten Wirtschaftssimulationen auf den Markt. Für alle, die sich ein Spiel kaufen wollen und noch unschlüssig sind, hier ein ausführlicher Test.

Ali Eghdamian

Das Spiel

Das Spiel beginnt im 16.Jahrhundert während des dreißigjährigen Krieges. Nach einer unerwarteten Erbschaft muß man versuchen, seinen Reichtum mit dem Handel von Gütern zu vergrößern. Dies erweist sich schwieriger als gedacht, da die Gegenspieler (die vom Computer übernommen oder von menschlichen Spielern gesteuert werden können) nichts unversucht lassen, um dem Spieler Stolpersteine in den Weg zu stellen. Dies erschwert die Aufgabe erheblich. Doch nach dem Motto: „Wie du mir, so ich dir“ kann man den Gegenspieler ausspionieren und sich so Beweise verschaffen. Mit diesen Beweisen kann man dann den Gegner anklagen, oder bei anderen Personen anschwärzen, bei welchen dann das Ansehen vom Gegenspieler erheblich angekratzt wird. (Bei Ernennungen von Ämtern eine durchaus wirksame Waffe). Natürlich reicht es nicht, nur Taler anzuhäufen, ein anderes Ziel ist es auch, ein möglichst hohes Amt zu bekleiden. Apropos Ziel: Die Programmierer von „Fugger2“ haben sich etwas neues einfallen lassen: In Fugger2 ist kein Ziel vorgegeben. Der Spieler darf selbst aussuchen, was sein Ziel sein soll. Mit Auftragskarten, welche in drei Kategorien aufgeteilt sind (Leicht, Mittel und Schwer), kann man sich für einen der vielen Aufträge entscheiden. Falls keine der Auftragskarten zusagt, kann man auch ohne einen Auftrag spielen.

Bemerkung: Man kann, nachdem man den Auftrag erfüllt hat, weiterspielen. Das Spiel ist nicht beendet. Deshalb mein Rat: Man sollte am Anfang irgendeine Aufgabe nehmen, die mehr Zeit in Anspruch nimmt: z.B.: Feiert fröhliche Feste, oder Belagert sieben Städte erfolgreich. Aufgaben wie: werdet Bürgermeister in einer Stadt, sind schnell erledigt, und daher keine Herausforderung.

Bei „Fugger2“ spielen Bestechungen eine große Rolle. Ohne Bestechungen wird man kaum gewählt, da es in Fugger2 genügend Gegenspieler gibt, die einen Platz selbst beanspruchen. Bei Bestechungen sollte man aber aufpassen, daß man nicht die falschen Personen besticht. Zum Beispiel: Man will Bürgermeister werden und besticht den Richter, damit dieser einen wählt. Doch bei der Ernennung kommt man drauf, daß der Bürgermeister durch die Ratsherren gewählt wird. Die Folge: Man hat unnötig Geld und Zeit verschwendet. Ein Blick im Handbuch wäre sehr empfehlenswert, da dort alles ganz genau erklärt wird. Es gibt aber auch andere Möglichkeiten einen Gegenspieler eins auszuwischen. Neben Erpressung, stehen dem Spieler noch Saboteure, oder Räuber bzw. Söldner zur Verfügung. Man sollte am Anfang dieses Spieles auf diese Sachen (Saboteure, Spione) verzichten, da diese meist sehr viel Geld kosten und man dadurch schnell im Schuldenturm landet. Also mein Rat: Am Anfang des Spieles sparen, sparen und nochmals sparen. Man sollte immer versuchen, die meistgefragte (und teuerste) Ware zu produzieren, da diese am meisten Geld einbringt. Es hat zum Beispiel keinen Sinn, Fische zu fangen und um sechs Taler zu verkaufen, wenn man für Glas zum Beispiel zwanzig Taler bekommen kann. Man sollte aber versuchen, bei den Ämtern so schnell wie möglich aufzusteigen, da man schon als Richter, Baumeister oder Kämmerer Gehalt bekommt. (Deshalb sind Bestechungen vom Anfang bis zum Ende des Spieles sehr wichtig und unentbehrlich). Hat man endlich genügend Taler angehäuft, kann man ein zweites Lager eröffnen, um noch mehr Geld zu verdienen. Wenn man das getan hat und genügend Geld hat, kann man ein Räuberlager bzw. eine Zollburg erwerben. Ich empfehle eher, als erstes ein Räuberlager, da diese Warentransporte überfallen können. Wenn man viel Glück und Ausdauer hat, entdecken die Räuber einige Warentransporte, welche man überfallen kann. Meistens halten sich in der Umgebung weitere Truppen auf, welche man bestechen kann, damit diese auf der eigenen Seite mitkämpfen. Falls der Transport keine Eskorte hat, sollte man sein Geld lieber im Sparstrumpf lassen, da die paar Fuhrknechte selten eine Gefahr darstellen. Bei einer Eskorte sieht die ganze Sache anders aus. Manchmal geschieht es bei einem wertvollen Warentransport, daß der Händler versucht, den Angriff mit größeren Geldbeträgen abzuwehren. Natürlich kann man das Geld einstreichen und trotzdem den Angriffsbefehl geben. Bei einem Überfall nicht vergessen, die Waren zu erobern! Es reicht nicht, jede gegnerische Einheit zu eliminieren! Man sollte, falls man andere Truppen bestochen hat, versuchen, die Waren mit den eigenen Truppen zu erobern und die bestochenen Truppen angreifen zu lassen. Danach kann man die erbeuteten Waren in ein Lager bringen und verkaufen lassen. Man kann neben Plünderungen auch andere Räuberlager/Zollburgen überfallen und einnehmen.

Irgendwann im Spiel sollte man auch an das Heiraten denken. Bevor man stirbt, sollte man sein Testament machen. Dies ist allerdings nur möglich, wenn man Kinder hat. Bei der Lesung des Testaments durch den Notar hört man verschiedene rührende Bemerkungen der Erben. (Zum Beispiel: „Darauf einen Humpen!“, „Wurde aber auch endlich Zeit!“ oder „Endlich bin ich reich!“) Danach kann man mit dem Haupterben weiterspielen. Es ist empfehlenswert immer das **letzt**geborene Kind zum Haupterben zu machen, da zwischen dem Erstgeborenen und dem Letztgeborenen meist viele Jahre liegen und man in Fugger2 ungefähr nur 30 Jahre lebt. Zum Beispiel: Das erste Kind ist im Jahre 1648 geboren, das zweite Kind im Jahre 1655. Die beiden Kinder haben einen Altersunterschied von sieben Jahren. Das heißt: Das erste ist schon sieben Jahre alt während das letzte Kind gerade ein Jahr alt ist. Je später das Kind der Erbe wird, desto höher ist die Wahrscheinlichkeit, daß die Gesundheit des Kindes nicht mehr so optimal ist. Es hat also keinen Sinn, wenn man anfängt zu spielen, wenn der Arzt sagt: „Es sieht nicht gut aus“. Man kann also, wenn man mit dem Letztgeborenen spielt, viel länger spielen, als mit dem Erstgeborenen.

Eigene Meinung

„Die Fugger2“ ist ein tolles und unterhaltsames Spiel, welches jeden Wirtschaftssimulationsfan begeistern wird. Da man mit seinem Nachfolger weiterspielen kann, wird man immer motiviert, sich noch einmal von unten bis zu den höchsten Ämtern hochzuarbeiten. Die Programmierer von „Fugger2“ haben in dem Spiel an alles gedacht. Neben Bestechungen, Kämpfen, Prozessen will ich noch die vielen Ereignisse am Ende des Jahres erwähnen, in die der Spieler selbst eingreifen kann. Zum Beispiel bekommt man die Aufgabe, Musiker zu engagieren. Man kann, wenn man will, den Auftrag ablehnen oder annehmen. Im nächsten Jahr kann man dann zwischen drei Musikergruppen auswählen. „Die Fugger2“ macht Spaß, da sich die Handlungen und die Ereignisse selten wiederholen. Wer dies nicht glaubt, kann sich selbst überzeugen. Vor einer Runde speichern und nachdem man die Ereignisse gelesen hat, wieder laden. Danach wieder die Runde beenden und siehe da: andere Ereignisse.(ausgenommen: Prozesse, Geburten, Versteigerungen).

Grafik: 80% Die Grafik ist im allgemeinen gut. Nette Filme (bei Katastrophen) und schöne Grafiken bereiten dem Spieler viel Spaß.

Sound: 86% Der Sound ist sehr gut und abwechslungsreich. Für (fast) jede Handlung gibt es eine eigene Musik. Zum Beispiel: Geburten, Arzt, Katastrophen, Prozesse... Die Sprachausgabe ist ein weiteres Schmankerl, welches sich Sunflowers einfallen ließ. Bei der Sprachausgabe gibt es viele verschiedene Stimmen, sodaß man sich nicht immer die gleiche Stimme hören muß.

Spielsteuerung: 85% Die Spielsteuerung ist gut, da fast ausschließlich die Maus verwendet wird (außer bei der Eingabe von Namen). Beim Kampfmodus ist die Steuerung etwas gewöhnungsbedürftig, da man die Einheiten genau anklicken muß und z.B.: bei einem riesigen Heer auf einem Fleck etwas länger braucht. Das Handbuch ist sehr ausführlich und übersichtlich. Wer sich für den historischen Hintergrund interessiert, kann im Zusatzheft schmökern.

Spielspaß: 89% Wie macht das Sunflowers nur? Obwohl ich (zu meinem Kummer) meist nach der dritten Generation aussterbe, fange ich wieder und wieder an, ein neues Imperium aufzubauen. Obwohl einem schon die Augen vor Müdigkeit (4 Uhr in der Früh) zufallen oder man zu einem Treffpunkt gehen muß, kann man sich von diesem Spiel einfach nicht losreißen.

Bugs: 15% Den einzigen Bug, den ich bisher entdeckt habe, ist der: daß manchmal dieselbe Zollburg bei einer Versteigerung zweimal versteigert wird. **Vorsicht:** Man sollte, falls dieser Bug auftritt, immer beim zweiten Angebot mitsteigern, da nur dieses gilt.

Gesamt: 82% Ich glaube, daß jeder mit diesem Programm gut beraten ist. Vom Einsteiger bis zum Profi, „Fugger2“ wird sicher jeden begeistern! Vor allem mit der Multiplayer-Funktion wird „Die Fugger2“ seine Liebhaber finden.

Dieses Spiel wurde auf einem 486 DX4 100 MHz, 8MB RAM und CD-ROM getestet. ☐

Command & Conquer

5

Der Tiberiumkonflikt

Florian Eichelberger

Allgemeines:

Command&Conquer (später nur noch CC genannt) ist meiner Meinung nach eines der besten Strategiespiele, die es gibt. Es handelt in der Welt des 21 Jahrhunderts. Die Menschheit ist in zwei Lager gespalten:

- Die **Bruderschaft von Nod** unter ihrem charismatischen Anführer *Kane*. In diesem Spiel verkörpert sie die Bösen.
- Die **Globale Defensiv Initiative** (GDI). Sie steht unter dem Kommando von *General Sheppard*. Sie verkörpert das Gute in diesem Spiel.

Der Grund des Konfliktes ist Tiberium, eine Pflanze, die durch einen Meteoriteneinschlag auf die Erde kam und überall zu wuchern begann. Diese Pflanze kann von speziellen Sammlern aufgelesen und dann in einer Tiberium-Raffinerie weiterverarbeitet werden. Ein weiterer Taktikaspekt ist der, daß Tiberium für Soldaten von GDI und Nod gleichermaßen gefährlich ist, was weitere Aktionen erlaubt. Als Beispiel, das unter CC-Spielern beliebte „Wer rennt zuerst ins Tiberiumfeld". Ein Wort möchte ich vorwegnehmen: ich werde das Spiel nur indirekt mit Prozentangaben benoten, da sich das Spielgefühl nicht rational darstellen läßt, wie es manche andere Autoren versuchen (nach dem Motto „Oft kopiert und nie erreicht").

Grafik:

CC hat eine unglaubliche Detailvielfalt und eine Fülle von verschiedenen Einheiten.

Dem Spieler steht vom etwa 8 Bots zur Verfügung; vom einfachen Infanteriebot bis zum stark gepanzerten Commandobot. Dazwischen liegen noch ca. 6 Mittelbots (Flammenwerfer, Raketenwerfer).

An Fahrzeugen hat man praktisch alles, vom einfachen M1A1 Panzer bis zum Mammut-Riesenpanzer zur Verfügung. Dazu muß man sagen, daß nicht jeder (GDI, Nod) alles bauen kann. Ein Beispiel dafür: der Mammut-Riesenpanzer kann nur von GDI, der Tarnpanzer nur von Nod gebaut werden. Bemerkenswert ist auch die Namengebung. Auch hierzu einige Beispiele:

- Hammer der Gnade (Panzer)
- Schwingen des Lebens (Helikopter).
- Flamme der Läuterung (Flammenwerferbot)
- Stahlhagel (Panzer)

Zu jeder Einheit gibt es ein mehr oder weniger passendes Gegenstück. So ist Fairneß fast immer gewährleistet.

Die Gebäudevielfalt ist auch beeindruckend. Es gibt von der Kaserne über Waffenfabriken und Tiberiumraffinerien samt Sammler bis hin zur Radar- bzw. Kommunikationszentrale fast alles, was sich der Feldherr wünscht. Zwei Sonderwaffen stehen nur im letzten Level zur Verfügung und heben den Spielspaß meiner Meinung nach erheblich, obwohl der Vorteil eindeutig bei Nod liegt. Es stehen zur Verfügung: GDI: Ionenwerfer, Nod: Atomschlag.

Beide Parteien haben spezielle Hauptgebäude, die es zu schützen gilt.

Die Grafik ist durchwegs in 256 Farben VGA, wird aber bald in SVGA erscheinen.

In der Grafikpalette sind auch natürliche Gegebenheiten wie Flüsse und Berge berücksichtigt.

Sound/Animation:

Es gibt in diesem Spiel eine relativ gute und gut verständliche Sprachausgabe, besonders vor jedem Auftrag. Weiters wird nach jedem erfolgreichen Auftrag eine der unzähligen, gerenderten Animationen gespielt und am Anfang jedes Levels darf man sich das Missionsziel als Video ansehen. In diesen Videos kommt natürlich auch Musik vor.

Die Videos (60 min gesamt) können beliebig wiederholt und mit Untertiteln versehen werden.

Bugs:

Der einzige Bug, den ich fand, war der Fehler „Überlauf bei Division" bei manchen Rechnern. Danach steht der PC. In diesem Falle muß man die Dateien händisch kopieren und sich von einem Freund die INI-Datei holen und modifizieren.

Sonst ist das Spiel fast nicht zum Abstürzen zu bringen und ist sehr robust.

Spielspaß:

Das Spiel hat mich an einem Sonntag 8(!!) Stunden an den Bildschirm gefesselt. Die Missionen sind ausgewogen und es gibt bereits eine Zusatz-CD mit weiteren 30 Levels, die ca. doppelt so schwer sind wie das „Grundspiel". Da es keine Cheats gibt und Veränderungen im Programm sofort und unnachgiebig zurückgewiesen werden, ist man nicht versucht - wie z.B. bei dem Klassiker „Doom" - die Cheats zu verwenden. Mir ist bisher nur ein brauchbarer Editor bekannt.

Sonstiges:

Das Spiel ist über Netzwerk (IPX) und Modem spielbar, wobei das Spiel selbst schlechteste Leitungen akzeptiert und Fehler korrigiert. Wenn eine Leitung „abreißt", versucht das Programm sie selbst wieder herzustellen.

Levelvielfalt:

Die Levels sind sehr unterschiedlich gestaltet und haben alle verschieden Primärziele (z.B. Zerstörung aller Nod-Einheiten).

Technische Daten/Aspekte:

Das Spiel läßt sich fast gleich schnell unter Windows wie unter DOS spielen, nur sollte man unter Windows alle anderen Applikationen schließen.

Das Spiel benötigt (darunter geht es nicht !) 486-33, 8 Mb RAM, 573.440 Byte konventionellen Speicher Für Mehspielermodus: 486-66.

Sound: Es werden unterstützt: Alle 100 % kompatiblen Soundblaster-Clones, Sb16, Awe 32 Gravis Ultrasound und viele mehr.

Bildschirm: Farbmonitor 256 Farben (VGA oder MCGA).

CD-ROM: Doublespeed mit mind. 300 kb/s.

DOS/Windows: Benötigt DOS 5.0 oder neuer oder Windows 3.1 oder Win 95.

Festplatte: Das Spiel benötigt 30 Mb auf der Festplatte (bei z.B. Stakker 69 Mb auf der Festplatte.) .

Schlußbemerkung:

Jeder, der ein abwechslungsreiches Strategiespiel sucht, wird begeistert sein, und auch Genreneulinge werden ihren Spaß haben. Wer allerdings nur seichtes Abballern sucht, wäre mit anderen Spielen besser beraten. □

Windows NT im Netzwerk

Planung, Installation und Management von Netzwerken mit Windows NT 3.51 Server und Workstation • Martin Kuppinger, Microsoft Press, 676 Seiten, 1 CD, S 694,-, ISBN 3-86063-351-1

Fritz Schmöllebeck

Nach einer grundlegenden Einführung zu Netzwerken wird die Windows NT Architektur besprochen. Diese beiden Kapitel beschränken sich auf die wesentliche Funktionalität sind jedoch ausgesprochen leicht lesbar und bieten einen guten Überblick über derzeit verwendete Netzwerk und Systemkonzepte. Auch unabhängig von einer geplanten NT Installation ist das Netzwerk-Kapitel durchaus empfehlenswert.

In Kapitel 2 wird die Installation von Windows NT beschrieben. Es empfiehlt sich dieses Kapitel (ca. 35 Seiten) vor Beginn der Installation zu studieren. Hier gibt der Autor nützliche Tips zu den Installationsmethoden von Servern und Arbeitsstationen. Das Domänenkonzept von Windows NT gemeinsam mit der Namenvergabe kommt in Kapitel 3 zur Sprache. Die Struktur eines Servers (Verzeichnisse, Namen, ...) kann nach erfolgter Installation nur zum Teil wieder verändert werden. Deshalb empfiehlt sich ein genaues Studium der Kapitel 2 und 3 besonders. Die Zugriffsrechte sowie die Methoden der Resourcenfreigabe werden hier ebenfalls beschrieben.

Dem Systemmanagement widmet der Autor ca. 100 Seiten im Kapitel 4. Die Überwachung eines laufenden Produktionssystems gehört ebenso zu den Themen wie Druckmanagement und Backup-Strategien.

Kapitel 5 geht auf heterogene Netzwerkstrukturen ein. Das Arbeiten im Microsoft-Umfeld wird ebenso wie Netware-Connectivity, TCP/IP-Anbindung und die Kommunikation mit der Apple-Welt behandelt.

Kapitel 7 bechäftigt sich mit den Anwendungen für Windows NT. EMAIL und Terminplanung, das Einrichten eines MS-Postoffice sowie der Betrieb von DOS und 16-Bit Windows Anwendungen wird erklärt. Ein Abschnitt über Kompatibilität rundet dieses Kapitel ab.

Anhänge zu den Befehlen von Windows NT sowie ein Glossar sind nützliche „Helferlein“. Der Abschnitt zu Tips und Tricks vom Microsoft Support ist bei vielen Problemen ausgesprochen hilfreich, dürfte aber aufgrund des schnellebigen Themas leider rasch veralten.

Auf der mitgelieferten CD finden sich die deutsche Version des NT-Service Pack 1, Shell Technology Preview (Windows-95-Oberfläche für Windows NT) sowie Kostproben des Microsoft Technet-Service.

Anläßlich einer Erstinstallation von Windows-NT an der Abteilung Elektronik des TGM konnten dem besprochenen Werk wertvolle Hinweise zur Installation über ein Novell-Netzwerk sowie zu diversen Hardwareinkompatibilitäten mit CD-ROM Drives entnommen werden.

Es handelt sich um ein recht gelungenes und abgerundetes Werk welches sowohl dem Einsteiger wie auch dem alten „Netzwerk-Hasen“ etwas bieten kann.

□

System Performace Tuning

Mike Loukides, O’REILLY-Thomson Publishing, ISBN 3-930673-26-6, öS 538,-

Fritz Schmöllebeck

Dieses Buch richtet sich an Personen, die UNIX-basierte Rechner, egal ob für Einzelplätze oder im Netzwerksverbund, für mehrere Benutzer mit unterschiedlichsten Anwendungen verwalten müssen. Es wird angenommen, daß der Leser mit den elementaren Aufgaben der Systemverwaltung (Benutzer und Verzeichnisse anlegen und verwalten, Archivierung, Shells, etc.) vertraut ist. Ist dies nicht der Fall, so empfehle ich das Buch „Essential System Administration“ aus der selben Serie von Handbüchern wie das vorliegende.

Ein UNIX-System auf die gegebene Benutzer- und Anwendungssituation abzustimmen kann nicht wie viele meinen nur von „Gurus“, die im Kernel des Betriebssystems herumhacken, gelöst werden. Vielmehr stellen heutige UNIX-Implementierungen eine Vielzahl von Dienstprogrammen zur Verfügung um dem System bei seiner Arbeit auf die Finger sehen zu können. Und genau auf diese Werkzeuge wird in diesem Buch eingegangen. Aber nicht nur die Aufrufsyntax mit den oft zahllosen Optionen wird enzyklopädisch gelistet, sondern wann wird welches Programm eingesetzt und wie sind die meist textuell in Tabellen präsentierten Daten zu interpretieren, um daraus die notwendigen Maßnahmen abzuleiten, damit möglichst viel der vorhanden Hardwareleistung ausgenutzt werden kann.

Das Buch ist folgendermaßen aufgebaut:

- Einführendes zum Systemdurchsatz
- Überwachung des Systemaktivität
- Handhaben des Arbeitslast
- Der Hauptspeicher
- Das Dateisystem
- Die Netzwerke
- Die Terminals
- Die Kernel-Konfiguration
- Echtzeitprozesse unter System VR4
- Strategien zur Leistungsoptimierung

Bei der Beschreibung der Werkzeuge wurde darauf geachtet, daß vor allem jene Dinge beleuchtet werden, die für die meisten UNIX-Implementierungen und -derivate gelten. Implementierungsspezifische Eigenheiten werden überdies getrennt betrachtet. Als Basis wird eine klassisches BSD4.3-UNIX-Implementierung herangezogen, weiters wird dann auf SunOS 4.1, auf AT&T System V und dessen Derivate, wie die Release 2 (SVR2), besser bekannt als XENIX, die Release 3 (SVR3), besonders für die x86-Architektur (PC) die Implementierungen von Interactive und SCO, und schließlich die Release 4 (SVR4) mit dem vernetzten Dateisystem von SunOS (NFS = Network File System), eingegangen.

Abschließend möchte ich sagen, daß für jemanden der UNIX-Systeme zu betreuen hat, und auf eine optimale Ausnützung der zur Verfügung stehenden Hardwareressourcen bedacht ist, dieses Buch eine sehr wertvolle Unterstützung darstellt.

□

Internet-Server, Einrichten und Verwalten

Cricket Lin, Jerry Peek, Russ Jones, Bryan Buns, Adrian Nye
O'REILLY-Thomson Publishing, ISBN 3-930673-17-7, öS 538,-

Peter Balog

Das Anbieten von Informationen im Internet ist in letzter Zeit für viele Unternehmen zu einem wichtigen Thema geworden. Parallel dazu entstand der Begriff „Intranet"; also das interne Netz einer Firma, wo natürlich in einer typischen vernetzten Umgebung auch die Informationen den Mitarbeitern auf elektronischem Wege entsprechend freundlich aufbereitet angeboten werden sollen. Die für die EDV Verantwortlichen werden also eher früher als später ein neues Aufgabengebiet bearbeiten müssen, nämlich den Aufbau und die Wartung von Internet-Server.

Das vorliegende Buch mit dem selben Titel versucht dieses weite Gebiet möglichst kompakt und leicht zugänglich zu erschließen. Es bietet Systemverwaltern von UNIX-Rechnern und -Netzen das notwendige Rüstzeug um diverse Server zu installieren, mit Information zu bestükken und zu warten. Obwohl es heute schon mehrere Betriebssystemplattformen gibt, für die Internet-Server verfügbar sind, so war und ist dies (noch) die Domäne der UNIX-Rechner.

Der Basisserver ist der INETD (Internet-Dämon), der aus einem UNIX-Rechner einer Internetknoten macht. Auf diesen setzen dann die anderen Server auf. Dem INETD und dem XINETD (für erhöhte Sicherheitsanforderungen) sind zwei sehr wesentliche Kapitel gewidmet. Nach einer Einführung in die Philosophie von Informationsdiensten werden dann die einzelnen Internetserver im Detail beschrieben:

- Finger, Telnet
- Aufbau eines FTP-Archives, der WU-FTP-Dämon, Pflege von FTP-Archiven
- WAIS-Datenbankserver, Generierung von WAIS-Sourcen
- Gopher-Server, Datenaufbereitung für Gopher, Gopher und andere Dienste
- Gopher-Datenbanken
- Veronica und Jughead
- Gopher+
- Das World Wide Web (WWW), der NCSA-WWW-Server
- Erstellen von WWW-Dokumenten in HTML, Gateways
- Zugangskontrolle und Sicherheit im Web
- Electronic Mail, Sendmail, Mailinglists
- FTP via Email
- Firewalls, Datenschutz, Urheberschutz

Über das reine Einrichten der Server-Software hinaus wird ganz wesentlich auf die Anwendung und auf die Datenaufbereitung eingegangen. Die Sicherheitsaspekte werden nur angerissen; hier muß sicher auf Spezialliteratur verwiesen werden. Leider wird beim WWW-Server die NCSA-Implementierung erklärt, und nicht die populärere Version vom CERN. Hat man jedoch einmal einen NCSA-Server aufgesetzt und wechselt auf den CERN-Server (wegen der Vorteile des Caching mit einer Proxy-Unterstützung), so treten dabei meiner praktischen Erfahrung nach keine besonderen Probleme auf.

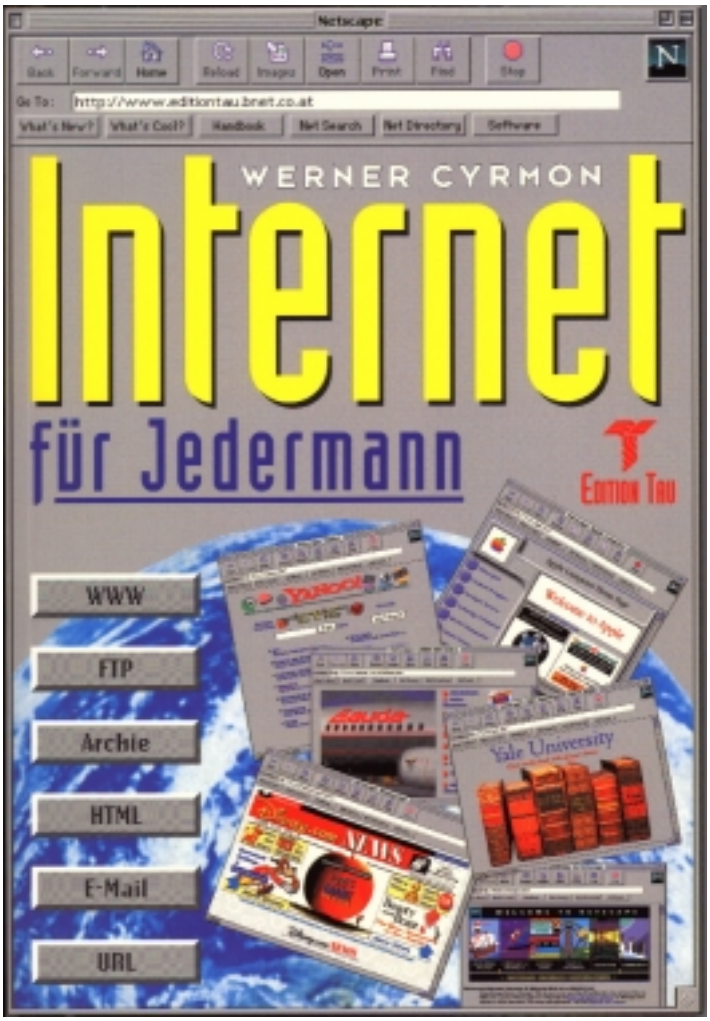
Das Buch „Internet-Server" stellt den gelungenen Versuch dar, die Vielzahl unterschiedlicher Internet-Informationsdienste sowohl von ihrer Anwendungs- als auch von der Installationsseite zu beleuchten.

□

Internet für Jedermann

Edition Tau, ISBN 3-900977-76-3, S 297,-

Werner Cyrmon



Das Internet stellt ein neues Medium zur Kommunikation, zur Informationsbeschaffung, aber auch zur Unterhaltung dar. Die Möglichkeiten, die das Internet bietet, sind heute schon überwältigend und wachsen ständig. So kann man bereits viele Zeitungen, nationale wie internationale (Standard, Stern, America Today...), über das Internet lesen, sehr schnell und billig Informationen weltweit austauschen, bekommt praktisch laufend aktuelle Daten z. B. aus der Forschung (etwa aktuelle Satellitenbilder) oder neueste Softwareupdates, kann ohne Schwierigkeiten in z. B. amerikanischen Buchläden Online-Einkaufen und vieles mehr. Die Benutzung des Internets wird bereits in kurzer Zeit genauso selbstverständlich sein wie heute noch das Telefonieren oder Lesen einer Zeitung. Nachdem immer mehr Haushalte über PCs und die meisten davon auch über das Telefon verfügen, welches zum Anschluß an das Internet notwendig ist, steigt momentan die Zahl der Internet-User auch im EU-Raum explosionsartig. Momentan gibt es weltweit ca. 56 Millionen User des Internets, und man schätzt, daß bis ins Jahr 2000 noch weitere 200 Millionen dazukommen werden. Wie vor ca. 100 Jahren mit der Erfindung des Telefons wird sich mit und durch die neue Technik des Internets vieles ändern bzw. ändern müssen (man denke nur an die Möglichkeit ca. 250 Mill. Menschen praktisch sofort via Computer kontaktieren zu können!). Für die Vielzahl an PC-Usern, die zwar mit ihrem PC gut umgehen können, aber doch keine Computer- und schon gar keine Internetexperten sind, wird in diesem Buch in leicht verständlicher Form - durch Graphiken unterstützt - auf alle Themen, die den Einsteiger betreffen, eingegangen. □

HTML

Teil 5: Gutes HTML-Design

Gerwald Oberleitner

Auch beim Entwerfen von HTML-Seiten sollte man sich gewissen Stilvorgaben, ähnlich wie beim Programmieren, unterwerfen. Auch sind einige andere Faktoren im Internet für einen guten Gesamteindruck seiner Präsentation im WorldWideWeb verantwortlich.

Einer der bedeutendsten Faktoren ist die Größe der HTML-Datei und der darin enthaltenen Bilder, da dies wesentliche Faktoren dafür sind, wie schnell eine Seite geladen wird. Es empfiehlt sich aber auch „Text-only"-Versionen der Homepage anzubieten, da es sicher einige Benutzergruppen gibt, die auf Grafiken wenig wert legen und damit auch meistens schneller und günstiger zu Ihren Informationen kommen. Reine Textseiten können jedoch auch zu lange sein und überladen wirken. Inhaltlich sollte eine WWW-Seite hinsichtlich einiger Dinge geprüft werden:

- Links prüfen: Auch Server im Internet werden umbenannt oder nicht mehr betrieben. Daher müssen auch die Links gewartet werden. Es ist sicher nicht gut, wenn ein Server, auf den ein Link gelegt wurde und auf den man sich im Text inhaltlich bezieht, nicht mehr erreichbar ist. Gründe dafür können zum Beispiel Firmenzusammenschlüsse und Konkurse sein, wovon mitunter auch ganz bekannte Adressen betroffen sind.
- Um den Lesern der Seiten Zeit und Nerven zu sparen, sollte auch auf Mirrorsites geachtet werden. (Mirrorsites sind orginalgetreue Spiegelungen des Dateninhaltes eines Webservers, die in anderen Ländern oder Kontinenten betrieben werden. Dadurch kann der Übertragungsweg optimiert und eventuell wesentlich verkürzt werden.)
- „Füttern" Sie Ihre Webpage mit neuen Informationen oder berichten Sie über kürzlich stattgefundene Ereignisse. Eine rein statische Information (z.B.: ein Firmenprofil das sich nie oder fast nie verändert) veranlaßt die Websurfer sicher nicht, sich Ihre Seiten mehr als zweimal anzusehen. Kündigen Sie zum Beispiel Veranstaltungen an oder versuchen Sie anders „Bewegung" in Ihre HTML-Seiten zu bringen.
- Für kommerzielle Webpages: Publizieren Sie Kontaktadressen, Telefon- und Faxnummer, damit es dem Leser möglich ist, Sie im Bezug auf die publizierte Information anzusprechen. Antwortformulare im Web können zum Beispiel dazu dienen, dem Leser zu ermöglichen, weiteres Informationsmaterial anzufordern oder Produkte zu bestellen.

Auch im HTML-Code gibt es die Möglichkeit, Kommentarzeilen zu verwenden (Tag: *<!--Dies ist ein Kommentar-->*) wovon eigentlich jeder gute Programmierer Gebrauch macht.

HTML-Tags

Zum Abschluß nun noch einige Hinweise zu den HTML-Tags: In einer gut geschriebenen HTML-Seite sollte darauf geachtet werden, daß nur genormte Tags verwendet werden. Dies ist auch für den Gestalter der sicherste Weg, daß jeder Browser diese Information darstellen kann und daß sie überall annähernd gleich aussieht, was bei einer Darstellung nach der „corporate identity" eines Unternehmens vielleicht von Bedeutung ist.

Overlapping Tags

Weiters sollte auch noch das Problem der „Overlapping Tags" besprochen werden. Overlapping Tags können ebenfalls zu Problemen bei der Darstellung führen und das Erscheinungsbild der Seite beeinträchtigen. Ein Overlapping Tag wäre zum Beispiel: *Dies ist ein Beispiel f&uumlr; ein overlapping Tag in HTML.* Die Tags und überschneiden sich mit den öffnenden und schließenden Tags. Dieses Beispiel richtiggestellt: *Dies ist (k)ein Beispiel f&uumlr; ein overlapping Tag in HTML.*

Im letzten Teil des HTML-Kurses möchte ich noch einige Tools, die das Gestalten von Webseiten erleichtern sollen, vorstellen. ☐

Gut Ding braucht Weile!

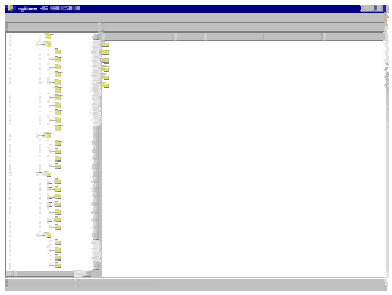
Aber was zu lange dauert, ist out !

Fritz Eller

Es ist eine heikle Sache, wenn man versucht, optimale Software zu einem bestimmten Thema zusammenzustellen. Wenn man sich dann entscheidet, daß man nun zu pressen beginnt, die frische CD unter die Leute bringen will und hofft, doch den einen oder anderen Freak zu finden, der ausgerechnet auf diesen - frischen, tollen, mit viel Liebe und Know-how zusammengestellten Software-Sampler gewartet hat, dann ist meist schon einige Zeit vergangen! Besonders bitter ist es bei Software für den Internet-Freak oder auch -Anfänger. Da nützt es nichts, wenn ich die CD ganz frisch vor mir habe, wenn sie noch quasi unberührt in meinem CD-Laufwerk abgetastet wird. Da nützt es nichts, wenn die Dateiverzeichnisse auf dieser CD noch ganz jung sind, wenige Tage alt erst, wenn sich die Software darin schon im Greisenalter befindet und schon ein Jahr auf dem Buckel hat.

Was ist schon, möchte man meinen, was ist denn schon ein Jahr bei Software? Richtig!, gebe ich zurück, aber bei Software für das Internet? Wenn ich die Programme herunterladen will, hab ich schon das Update vom Update zur gepreßten Version.... Und da ist das Minus an der NETSURF.

Und das Plus?
Für beides (+ und -) spricht der dargestellte Verzeichnisbaum!



Die Zusammenstellung ist ganz passabel, wenn auch nicht umwerfend! Packer sind ebenso zu finden, wie Verschlüsselungsutilities. Gemischte Net-Software für Win 3.1x findet man ebenso, wie fürs W95, mit der Einschränkung siehe oben.

Und im Detail findet man das Chaos:

Erfreulicherweise kann sich der Unwissende informieren: Über Internet allgemein, über Suchen im Net, über Newsgruppen. „Und das einheitlich im Format ..."- hätt' ich gern geschrieben: Die eine Info als Hilfedatei, die andere als Write-Dokument, die dritte im Word-Format und die vierte als TXT-Datei.

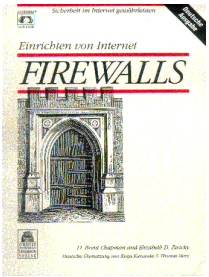
Und genau das ist es, was irgendwie an der Professionalität zweifeln läßt. Schade !

Denn eigentlich wollen viele NETSURFER sein....☐

Internet Firewalls

Einrichten von Internet Firewalls von D. Brent Chapman und Elizabeth D. Zwicky; Verlag O´Reilly, International Thomson-Verlag, 1996, ISBN 3-930673-31-2, 576 Seiten, öS 538,-

Hans Blocher



In jeder Gesellschaft gibt es einen gewissen Prozentsatz von Personen, die anderen mutwillig schaden. Das Internet umfaßt derzeit schätzungsweise 40 Millionen Benutzer. Auch wenn der Anteil böswilliger Benutzer weniger als ein Prozent der Gesellschaft ausmacht, ist er doch groß genug, um sich damit auseinandersetzen zu müssen. Manche Sicherheitslücken (z.B. Schwachstellen in Betriebssystemen) sind hinlänglich bekannt, es gibt inzwischen aber auch höchst raffinierte Angriffsmethoden. Wie kann man sein eigenes System vor solchen Bedrohungen schützen, ohne daß der eigene Zugang ins Internet unnötig erschwert wird? Derzeit stellen Internet-Firewalls die effektivste Verteidigung dar.

Was ist ein Firewall? In einem Gebäude dient eine Brandmauer (firewall) dazu, das Übergreifen eines Feuers von einem Gebäudeteil auf einen anderen zu verhindern. Im Prinzip verfolgt ein Internet-Firewall einen ähnlichen Zweck: Er gewährt nur an einem streng kontrollierten Punkt Zutritt; er hält Angreifer davon ab, anderen Schutzvorrichtungen zu nahe zu kommen; er sorgt dafür, daß das System nur an einem einzigen Punkt verlassen werden kann.

Ein Firewall soll daher trennen, einschränken und analysieren. Die physikalische Implementation sieht bei jedem Standort anders aus. Meistens besteht er jedoch aus verschiedenen Hardware-Komponenten (Router, Computer, ...) mit geeigneter Software.

Das Buch „Einrichten von Internet-Firewalls“ gibt eine praktische Anleitung zur Konzeption, Einrichtung und Verwaltung von Firewalls. Es enthält unter anderem:

- Ausführliche Beschreibungen zum Aufbau von Firewalls mit Paketfilterung oder Proxy-Diensten sowie Anleitungen zur Konfiguration von Internet-Diensten (E-Mail, FTP, DNS, Telnet, WWW) für die Zusammenarbeit mit Firewalls.
- Allgemeine Kapitel über die Gefahren im Internet, Firewall-Architekturen, Sicherheitsstrategieen, Methoden zur Benutzerauthentifizierung, Betreuung von Firewalls und Reaktionen auf Einbrüche.
- Eine Übersicht über Informationsquellen und frei verfügbare Tools (hauptsächlich für Unix-Systeme), die man für den Aufbau eines effektiven Firewalls nutzen kann.

Für wen ist dieses Buch gedacht? Obwohl es sich in erster Linie an Personen richtet, die Firewalls einrichten müssen, ist es in weiten Teilen auch für Leute geeignet, die sich allgemein mit der Sicherheit im Internet befassen.

Frei gehaltene Auszüge aus dem Buch:

Was können Firewalls?

Ein Firewall ist das Zentrum für Sicherheitsmaßnahmen

Die Bündelung von Sicherheitsmaßnahmen ist wesentlich effizienter, als die sicherheitsrelevanten Entscheidungen und Technologien über die ganze Organisation zu verteilen und alle Schwachstellen stückweise abzudecken. Die Kosten für den Aufbau eines Firewalls können in die Zehntausende (DM) gehen.

Ein Firewall kann die Sicherheitspolitik durchsetzen

Viele der Dienste, die die Anwender vom Internet verlangen, sind von Haus aus unsicher. Ein Firewall läßt nur anerkannte Dienste passieren, und auch diese nur innerhalb der dafür festgelegten Regeln. Er kann aber auch die Aufgabe haben, kompliziertere Richtlinien durchzusetzen. Zum Beispiel dürfen nur bestimmte Systeme innerhalb des Firewalls Dateien vom und zum Internet übertragen. Zu diesem Zweck werden oft Paketfilter-Systeme verwendet. Sie routen Pakete zwischen internen und externen Rechnern, gehen dabei aber selektiv vor. Sie lassen bestimmte Pakettypen passieren oder blockieren sie auf eine Art, die die Sicherheitspolitik eines Standortes widerspiegelt. Der in einem Paketfilter-Firewall verwendete Routertyp wird Überwachungsrouter genannt.

Ein normaler Router sieht sich einfach die Zieladresse eines jeden Pakets an und wählt den besten ihm bekannten Weg aus, um das Paket in Richtung seines Bestimmungsortes zu senden. Wie das Paket zu behandeln ist, wird nur vom Zielort bestimmt. Ein Überwachungsrouter dagegen sieht sich die Pakete genauer an. Er entscheidet nicht nur, ob er ein Paket in Richtung des Zielortes routen kann, sondern auch, ob er es überhaupt soll.

Ein Beispiel für die Art, wie ein Überwachungsrouter programmiert sein könnte, um Pakete selektiv von und zum Standort zu routen:

> *Blockiere alle Verbindungen von Systemen außerhalb des internen Netzes. Lasse lediglich eintreffende SMTP-Verbindungen passieren, damit E-Mail empfangen werden kann.*

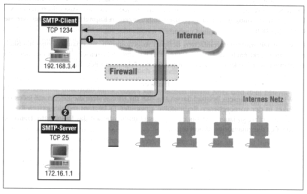
Im folgenden Faksimile-Auszug wird beschrieben, wie so ein Regelsatz entsteht. Die Filter werden dazu schrittweise entwickelt und nicht gleich in ihrer endgültigen Fassung angeführt.

Regel	Richtung	Quell-Adresse	Ziel-Adresse	Proto-koll	Ziel-Port	Aktion
A	ein	extern	intern	TCP	25	zulassen
B	aus	intern	extern	TCP	>1023	zulassen
C	aus	intern	extern	TCP	25	zulassen
D	ein	extern	intern	TCP	>1023	zulassen
E	beliebig	beliebig	beliebig	beliebig	beliebig	verbieten

- Die Regeln A und B erlauben eingehende SMTP-Verbindungen (eintreffende E-Mail).
- Die Regeln C und D lassen hinausgehende SMTP-Verbindungen zu (hinausgehende E-Mail).
- Regel E ist die Standardregel, die angewandt wird, wenn keine der anderen Regeln greift.

Betrachten wir nun anhand einiger Beispiele, wie Pakete behandelt werden. Angenommen, Ihr Rechner besitzt die IP-Adresse 172.16.1.1, und jemand möchte Ihnen Post von einem fernen Rechner mit der IP-Adresse 192.168.3.4 senden. Ferner benutzt der SMTP-Client des Absenders Port 1234, um mit dem SMTP-Server an Port 25 zu kommunizieren. Wie in der Beschreibung von SMTP in Kapitel 8 erwähnt, benutzen SMTP-Server die Standardportnummer 25.

Abb. 6-10: Paketfilterung: nach innen gerichtetes SMTP (Beispielpakete 1 und 2).



Paket	Richtung	Quell-Adresse	Ziel-Adresse	Proto-koll	Ziel-Port	Aktion (Regel)
1	ein	192.168.3.4	172.16.1.1	TCP	25	zulassen (A)
2	aus	172.16.1.1	192.168.3.4	TCP	1234	zulassen (B)

Abbildung 6-10 zeigt diesen Fall.

Die Paketfilterregeln erlauben bei Ihnen eingehende E-Mail:

- Regel A läßt zu, daß Pakete vom SMTP-Client des Absenders bei Ihrem SMTP-Server eintreffen (siehe Paket Nummer 1).
- Regel B erlaubt, daß Ihr Server dem Client des Absenders antwortet (siehe Paket Nummer 2).

Aus dem Inhaltsverzeichnis des Buches:

Teil I: Sicherheit im Netz 1 Wozu braucht man Internet-Firewalls? 2 Internet-Dienste 3 Sicherheitsstrategien **Teil II: Einrichten von Firewalls** 4 Entwurf von Firewall-Systemen 5 Bastion-Hosts 6 Paketfilterung 7 Proxy-Systeme 8 Konfiguration von Internet-Diensten 9 Zwei Beispiele für Firewalls 10 Authentifizierung und eingehende Dienste **Teil III: Kontinuierlicher Schutz Ihres Standortes** 11 Sicherheitspolitik 12 Betreuung von Firewalls 13 Reagieren auf Zwischenfälle **Teil IV: Anhänge** A Ressourcen B Werkzeuge C Grundlagen von TCP/IP

Zusammenfassung:

Wer sich von diesem Buch konkrete, leicht umsetzbare Beispiele erwartet, wird enttäuscht sein. Das Gebiet der Schutzeinrichtungen für das Internet ist zu komplex, als daß es an Hand von einigen Musterinstallationen erklärt werden könnte. Aufgrund dieser Tatsache können die Redundanzen, die bei dem Versuch auftraten, eine möglichst erschöpfende Abhandlung aller Aspekte der einzelnen Internet-Dienste bezüglich Paketfilterung, Proxy-Diensten, Konfiguration etc. anzubieten, doch teilweise als störend empfunden werden.

Ein Hinweis dieses Buches erscheint mir jedoch besonders wichtig: auch ein gutes Firewall-System kann keinen 100%igen Schutz gewährleisten. Man wird Kompromisse schließen müssen zwischen dem gewünschten Maß an Sicherheit, den dadurch entstehenden Kosten für Installation und Betreuung, und den Einschränkungen, den gewisse Sicherheitsstufen für den alltäglichen Betrieb mit sich bringen. ☐

Highway 194 - Eine Sackgasse ?

Werner Illsinger

Schon bisher war es möglich, Datendienste von Post und Radio Austria zu bundesweit konstanten Zugangstarifen zu nützen. Durch den neuen Dienst HIGHWAY 194 öffnet sich diese Option auch für alle Serviceprovider im Lande, nicht nur für den ehemals Monopolisten Post.

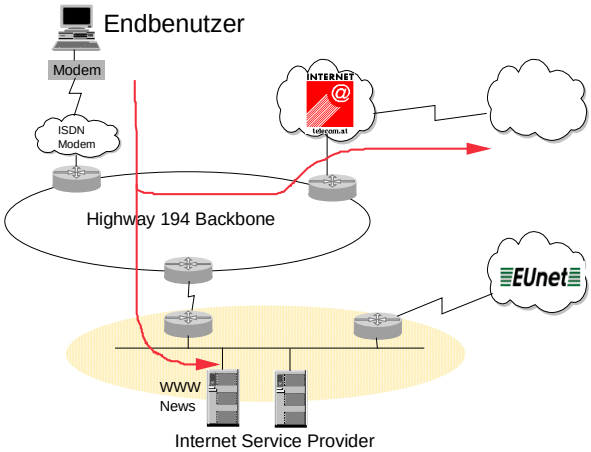
Zitat aus der WWW-Seite der Datakom:

> Durch den neuen HIGHWAY 194 können Rechner- und Servicebetreiber diese Möglichkeit ebenfalls für ihre Kunden nützen. Der Kunde wählt mit seinen Telekommunikationseinrichtungen die bekanntgegebene Nummer. Diese beginnt immer mit den Ziffern 1 - 9 - 4 - daraus wurde der Name des Dienstes abgeleitet. Die Telefonrechnung des Kunden wird nur mit dem entsprechenden Telefon-Ortstarif belastet.
>
> Anschaltungen sind mit den meisten am Markt erhältlichen Modems möglich, wobei die Geschwindigkeiten von 1.200 bis 28.800 bit/s (V.34 Standard) unterstützt werden. Weitere Zugangsmöglichkeiten sind aus dem Datex-P (bis 64 kbit/s) und dem ISDN sowohl mit V.110 Norm (Bitratenadaption 9.600 bit/s, 19.200bit/s und 38.400 bit/s) als auch mit X.75 (64.000 bit/s).

Dieser Dienst wäre für österreichische Internet Provider natürlich ideal, um österreichweit zum Ortstarif erreichbar zu sein, wenn - ja wenn ....

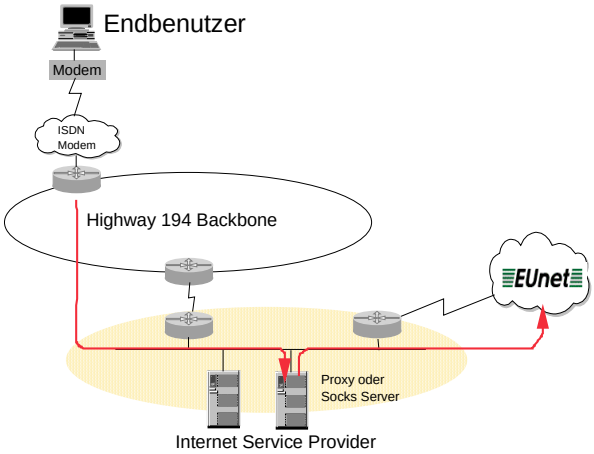
Technische Realisierung

Der Highway-194 besteht aus einem MAN-Backbone und Einwählknoten in vielen österreichischen Städten. An diesen Einwählknoten stehen sowohl analoge (V.34) als auch digitale (ISDN) Einwählknoten zur Verfügung. Ein Highway-194-Kunde wählt sich also an so einen Knoten an, bekommt eine TCP/IP-Adresse des Highway-194 zugewiesen und hat somit eine Verbindung zum Backbone. Der Anbieter kann entscheiden, ob dieser Benutzer nur auf seine Infrastruktur oder auf das Internet zugreifen darf. Entscheidet sich der Anbieter für zweitere Möglichkeit, so wird der Verkehr ins Internet (ausgenommen Highway-194-Teilnehmer) über den Internet Provider ***telecom.at*** geroutet (Konsortium aus Post, Radio Austria, Spardat). Der Internet Provider übernimmt in diesem Fall also nur noch die Aquirierung der Kunden, die Berechtigungspüfung und eventuell die Verrechnung. Damit entsteht also sozusagen ein großer Provider mit vielen kleinen Subprovidern, die zur eigentlichen Internetanbindung nichts beitragen können.



Einzige Möglichkeit dieses Verhalten zu umgehen, ist es, dem Benutzer nur Zugriff auf die Rechner des Internet-Service-Providers zu gestatten. und vom Highway-194 aus den Zugriff auf das Internet sperren zu lassen. Um dann dem Benutzer zu ermöglichen, auf das Internet über den Provider des Internet Service Providers (ISP) zuzugreifen (in unserem oben gezeigten Beispiel die Firma EUnet), muß der ISP einen s.g. Proxy Server oder Socks Server installieren. Mit Hilfe dieser beiden Protokolle wäre (und ist) es möglich, die ***telecom.at*** als Internet-Provider zu umgehen.

Diese Vorgehensweise bringt jedoch Nachteile für den Benutzer. Falls der ISP, der am Highway 194 angeschaltet ist, nur Proxy anbietet, so können - grob gesagt - nur jede Dienste genutzt werden, die im Browser integriert sind (HTTP, FTP, GOPHER). Will der Benutzer andere Dienste nutzen, so wird er scheitern. (Beispielsweise Clientsoftware für FirstClass-Mailboxen, Telnet, Iphone, etc.) Dies stellt aus unserer Sicht eine drastische Einschränkung der Dienste für den Benutzer dar.



Falls der Provider zusätzlich noch einen Socks Server anbietet, so kann - falls der Benutzer Client-Software hat, die das Socks-Protokoll spricht - (nicht zu verwechseln mit WinSock) mit dieser auch auf das Internet zugreifen. Diese Software ist jedoch auch nicht besonders zahlreich vertreten.

Im großen und ganzen zwingt man den Internet Provider dazu ***telekom.at*** als Provider zu verwenden oder eine merkliche Serviceeinschränkung für die Benutzer hinzunehmen. Was also wie ein besonderer Vorteil für die österreichischen Internet-Surfer aussieht, könnte sich in einiger Zeit als Bummerang für den Konsumenten herausstellen - nämlich dann wenn alle Internetprovider nur noch Anhängsel der datakom und telekom.at sind. ☐

FREIE BILDER

Mit Stand vom 5.7.96 finden Sie in der Konferenz FREIE BILDER auf dem Highway Nr. 1 241 Bilder vor, die Sie alle KOSTENFREI an Ihren Rechner downloaden und in jeglicher gewünschten Art weiterverwenden können.

Viel Spaß, Ihr Highway-Team

Highway Nr. 1 - die Mailbox für das grafische Gewerbe
`Highway-Nr1@highway.atnet.at`
Modem-Line (28.800 V.34): 0222/60803-33
ISDN-Line: 0222/600 15 35 und 600 15 36

Von PAN/BTX zum Internet ! ?

Walter Riemer

Im Mai 1996 wurden PAN-Kunden per Rundbrief in helle Vorfreude gestürzt: Die Post, pardon „datakom austria“, kündigt ein bezüglich Leistungsumfang und Kosten konkurrenzloses Internet-Paket an. Etwas verspätet, aber doch immerhin am 1. Juli langte ein ausführliches ONLINE-Heft samt CD-ROM beim Schreiber dieser Zeilen ein, der, wie ja die meisten wissen, OS/2 Warp-User ist. Mit angemessenem Optimismus wurde unter OS/2 in WINDOWS gegangen und eine Installation versucht.

Überraschenderweise kam die Meldung (sinngemäß):

Version 2.0 des Netscape Navigators verwendet OLE 2.03, die nicht mit dieser Version von WIN32S kompatibel ist. Vorher auf WIN32S Version 1.2 aufrüsten. Näheres kann README.TXT entnommen werden.

Nun habe ich WIN32S mit Mathcad ohne besonderes Zutun mitbekommen und mich dann nicht mehr darum gekümmert: ich weiß gar nicht, welche Version ich habe! Guter Rat scheint teuer, also drücke ich voll Hoffnung auf den OK-Button. Ein informativer Hinweis auf die Möglichkeit, die 16-Bit-Version zu installieren, hätte sehr beruhigend gewirkt! `README.TXT` existiert nämlich auf der CD-ROM gar nicht (oder nur in komprimierter Form) und auf meiner Festplatte erst nach geglückter Installation!

Irgendwie ist es mir später geglückt, die 16-Bit-Version von Netscape zu installieren.

Natürlich will ich auch eine E-Mail-Adresse aufmachen, also markiere ich den diesbezüglichen Radio-Button im Zuge des Installationsprozesses. Tatsächlich wird mir nach „erfolgreicher“ Beendigung der Installation mitgeteilt, daß ich im Setup festgelegt habe, daß ich mich zur Internet-Mail anmelden will: *„Sie werden jetzt automatisch mit den Systemservices verbunden“*. Da mich jeder automatische und daher unproblematische Vorgang sehr freut, drücke ich begeistert den OK-Button und gelange tatsächlich in das sich öffnende Fenster „System Services“. Daß laut einem Hinweis auf einem offenbar in letzter Minute der ONLINE-Zeitschrift beigelegtem Ergänzungsblatt stand *„Klicken Sie auf den Punkt Systemservice und füllen Sie dort das Mail-Formular aus“* habe ich in meiner Freude schon fast vergessen und insbesondere, daß ich in den AOnLine-Tastern und -Menüpunkten diesen Punkt nicht entdeckt habe. Was ich nicht vergessen kann, ist, daß trotz Automatik die auf dem Ergänzungsblatt abgedruckten Einstellungsfenster einfach nicht erscheinen wollen.

Vielleicht klappt die Verbindung nicht?

Ein Anruf bei der Hot-Line bringt die Information, daß die auf der Rückseite des besagten Ergänzungsblatts angekündigte, anscheinend überaus vorteilhafte Neuerung Shiva Winsock nach Meinung des Hot-Line-Helfers keineswegs empfehlenswert ist und Trumpet Winsock „besser“ funktioniert.

Also Neuinstallation mit Auswahl *Trumpet Winsock*. Zu meiner E-Mail-Adresse komme ich trotzdem nicht, aber um die Verbindung als solche zu testen, versuche ich, ins ETB (Elektronisches Telefonbuch) zu kommen. Oh Wunder über Wunder: es funktioniert tatsächlich! Allerdings dieses erste und einzige Mal: ich habe es seitdem im Laufe von etlichen Stunden kein zweites Mal fertiggebracht.

Um meine Leser nicht zu sehr zu langweilen, führe ich jetzt nur kurz etliche „Erlebnisse“ an, die ich im Laufe des 1. Juli 1996 beim Versuch hatte, den Anleitungen des ONLINE-Zeitschrift, insbesondere Seiten 12 und 13 samt Ergänzungsblatt, zu folgen. Ich möchte erwähnen, daß dort auch der ungemein beruhigende Satz steht: Ein Ingenieursdiplom der Computerwissenschaften ist gänzlich überflüssig. Das stimmt auch: selbst ein TU-Professor für Computerwissenschaften kann sich wahrscheinlich aus der lapidaren Meldung „Winsocket kann nicht geladen werden. Ausführung abgebrochen“ keinen Reim machen, insbesondere wenn er unmittelbar nach pflichtgemäßem Betätigen des OK-Buttons einen zweiten Versuch macht, ins Netz zu gelangen und das Moden jetzt plötzlich pflichtschuldigst wählt (dann allerdings tat sich trotz blauer Meldung *„Verbindung erfolgreich hergestellt“* gar nichts mehr).

Oder was fängt man beim Beenden des Programms mit der Meldung an *„Achten Sie auf bestehende TCP/IP-Verbindungen“* (ja wie denn, zum Teufel?) und nach Betätigen des OK-Buttons mit einem GPF (Allgemeine Schutzverletzung, General Protection Fault) in `TCPMAN.EXE` konfrontiert wird?

GPFs waren überhaupt häufige Höhepunkte. Nach dem resultierenden Absturz der WINDOWS-Session kam beim nächsten Start der AOnLine-Programms meistens die Meldung *„Die Datenbank kann nicht geöffnet werden. Überprüfen Sie den Pfad in AONLINE.INI: [Database] Path= ...“*. Nach reiflichem Studium eines unversehrten `AONLINE.INI` erkennt man, daß die dritte Zeile im Abschnitt `[Database]` bei mir den Inhalt `D:\AONLINE\TRUMPET` hatte, allerdings in der Fehlersituation diese Zeile einfach nicht mehr da war! Wie sollte man sie daher überprüfen?

Abschließend möchte ich nicht unerwähnt lassen, daß ordentlich geschriebene WINDOWS-Programme unter OS/2 sowohl problemlos laufen als auch problemlos installierbar sind. Da aber der Teufel bekanntlich nie schläft, machte ich auch etliche Versuche mit Original MSDOS 6.2/WINDOWS 3.1 (was ja dank Boot-Manager eines OS/2-Rechners auch ganz leicht geht), aber der Erfolg war im wesentlichen ähnlich (gleich war auch nicht zu erwarten, da das Programm hinsichtlich der Aktivierung möglicher Fehler einen Zufallsgenerator eingebaut zu haben scheint).

Abschließend kam ich zu dem Ergebnis, daß zumindest die zentrale Aussage der ONLINE-Zeitschrift ihre Richtigkeit hat: Ein Ingenieursdiplom der Computerwissenschaften ist gänzlich überflüssig.

Gesucht werden Fachleute, welche das A-OnLine-Programm zum Laufen gebracht haben, und das möglichst noch in der ersten Juli-Woche, weil die nächsten drei Juliwochen bin ich auf Urlaub und das gebührenfreie Internet-Surfen ist mit Ende Juli terminiert. Immerhin habe ich dahingehend profitiert, daß ich jetzt meine PAN-Teilnehmerkennung auswendig weiß!

PS: Abschließende Bemerkung: Stand 1. August: Unter DOS/WIN 3.1 läuft die Software nunmehr doch; unter OS/2 gibt es noch immer Probleme, insbesondere auch mit der Parametrisierung. ☐

Österreichische Schulen: Ihre WWW-Seiten und ihre Provider?

Vor etwa einem Jahr wurde das ASN, das Austrian School Network zum ersten Mal öffentlich bei der viet-95 präsentiert. (siehe PCNEWSedu-45). Die PCNEWS-Redaktion nahm 1-Jahr-ASN zum Anlaß, die Anbindungsformen der Schulen darzustellen.

Franz Fiala

Schulen im Internet

Quellen

Es sind derzeit einige Verzeichnisse bekannt, die einen Überblick über die Schulen bieten (siehe PCNEWSedu 47, S.32), die sich in WWW-Seiten oder mit eigenen Servern präsentieren:

- http://ourworld.compuserve.com/homepages/ag_hollabrunn/wwwschul.htm
- http://hyperg.tu-graz.ac.at/Info-AT.sch;sk=7E040067
- http://borg-6.borg-graz.ac.at/bmproj.htm
- http://www.padl.ac.at/schule/schule.htm
- http://www.tgm.ac.at/tgm/pcnews/familie/schulen/schulen.htm
- http://www.styria.co.at/Schulen;sk=7AA36368
- http://www.ccc.or.at/schulen/schulen.html
- http://www.vip.at/GelbeSeiten/Oesterreich/Schulen.html
- http://www.educat.hu-berlin.de/schulen/austria.html
- http://www.kem.ac.at/

Auch der Styria-Verlag bietet ein Verzeichnis von Schulen im Internet an, ein Zeichen, daß diesem Sektor von kommerzieller Seite durchaus Bedeutung beigemessen wird.

Aus diesen Quellen wurde das Verzeichnis auf der nächsten Seite erstellt.

Zu beachten

- Es gibt sicher eine Anzahl von Schulen, die in den Verzeichnissen nicht enthalten sind.
- Es gibt in Österreich etwa 2300 Schulen (ohne Volksschulen). Die hier untersuchten 118 Schulen sind daher erst 5 % des Gesamtvolumens.
- Die Fachhochschulen mit bekanntem Internet-Zugang sind im Verzeichnis ebenfalls enthalten (Joanneum-Graz, Technikum Vorarlberg, Technikum Kärnten, FH-Salzburg) aber keine Universitäten.
- Möglicherweise wurden einige Zuordnungen in der folgenden Aufstellung nicht korrekt getroffen (Beispiel: Anbindung der HTL-Salzburg).
- Die „Szene“ ist stark in Bewegung. Angegebene Adressen können sich kurzfristig ändern, Zahl der Schulen nimmt rasch zu. Viele Schule warten mit ihrem Eintrag in Verzeichnisse bis wirklich „alles läuft“.
- Viele Web-Server und Web-Seiten sind an einem Standort mit mehreren Schulen, daher ist die Auswertung der Providerübersicht geringfügig anders als die Gesamtübersicht.

Internet-Verbindungswege

Wie wurde geprüft?

Ausgehend von der Liste der Schulen mit WWW-Seiten im Internet wurde mit Hilfe des Programm TRACERT (Bestandteil von Windows 95) jede Schuladresse getestet. TRACERT gibt Hinweise, wie die einzelnen Schulen mit dem Internet verbunden sind. Der Aufruf

```
TRACERT hostname
```

liefert eine Liste aller Stationen, die das Nachrichtenpaket durchläuft inklusive Verzögerungszeiten.

Der Ausgangspunkt war ein Account bei at-net, der laut Provider-Verzeichnis der PCNEWS (siehe PCNEWSedu-46, Seite 60) mit EBONE und mit EUNET verbunden ist.

Der PCNEWS-Rechner war mit Wählverbindung mit ***atnet*** verbunden (in der Grafik oben). Aus den TRACERT-Tabellen wurde ein Netzplan gezeichnet „Verbindungswege im Internet in Österreich“. Jedes Oval ist ein Rechner. In der oberen Zeile steht die Provider-Kurzbezeichnung, in der unteren Zeile steht der Name des Rechners. Manchmal steht aus Platzgrünen ein Oval für mehrere Rechner (Beispiel: ***cc.univie*** cerberus/doppler oder ***bbnplanet*** ...(4)=4 Rechner).

WWW-Präsenz

Die Papierrolle mit dem Aufdruck „WWW“ zeigt an, daß sich Schulpräsentationen auf dem links davon befindlichen Server befinden. Bei welcher Organisation dieser Server steht, kann am Providernamen abgelesen werden. Diese Selbstdarstellung der Schule im WWW ist mehr oder weniger statisch. Jede Veränderung der WWW-Informationen wird durch den Kustos am jeweiligen Rechner mit ftp durchgeführt. Die Schüler können an der Seitengestaltung teilnehmen, indem sie HTL-Seiten off-line erstellen, die Übertragung ins System oder das Arbeiten im WWW können sie nur mit einer Wählverbindung zu einem Provider, wenn die Schule (oder der Lehrer) einen solchen Account besitzt.

Wählzugänge

Da temporären Wählzugängen keine festen Adressen zugeordnet werden, kann man solche Anschlüsse als Internet-User nicht feststellen. Soweit bekannt ist, verfügen das KEM, das borg-graz und die Pädak Linz über solche Wählzugangsmöglichkeiten im Schulbereich. Alle anderen Zugangsmöglichkeiten werden über private Anbieter geschaffen. Es wäre wünschenswert, die E-Mail-Adressen dieser Schulen zu kennen, um mit ihnen Kontakt aufnehmen zu können. Die Verzeichnisse der Schulen sollten daher unbedingt ein Feld für eine E-Mail-Adresse enthalten.

Schulen und ihre Service-Provider

Aus den „Verbindungswegen“ wurde die Tabelle „Schulen und ihre Service-Provider“ erarbeitet. Die Statistik am Ende des Beitrags rundet den ASN-Report ab.

Zusammenfassung

Etwa 5 % der österreichischen Schulen und Bildungseinrichtungen sind durch Web-Seiten im Internet vertreten. Diese Verbindungsart ist für „Unterricht am Netz“ weniger geeignet, vielmehr kann sich die Schule im Internet präsentieren.

17 Schulen (4 Fachhochschulen mitgerechnet) betreiben einen eigenen Internet-Server, sind daher in der Lage, das schuleigene LAN mit dem Internet zu verbinden, sodaß alle Lehrer und Schüler das Informationsangebot des Internet auch im Unterricht nutzen können.

Ein Anfang.

	Schulen mit eigenem Internet-Server
Oberösterreich	padl, htl-leonding
Niederösterreich	minic, htlwrn
Salzburg	htl
Steiermark	borg-graz monsbergergasse
Tirol	kbib2
Wien	brg22, bg19, htlw1, htl-tex,tgm,htlvie22
FH	Joanneum, FH-Spittal, FH-Sbg, Technikum Vorarlberg

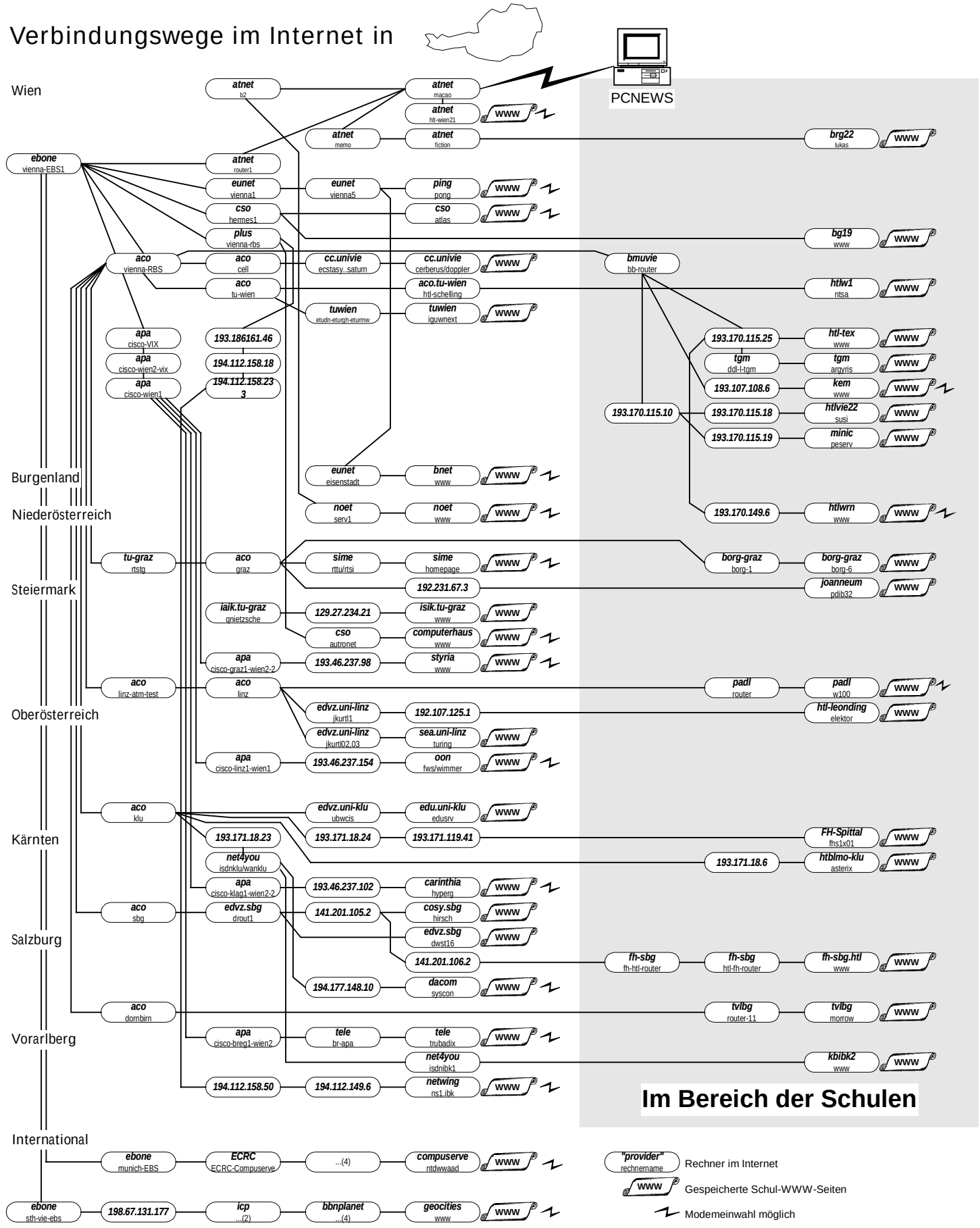
Die Service-Provider können in folgende Kategorien unterteilt werden:

	Anbieter von Leitungen für Schulen
Schule	bmuvie, fh-sbg
Universität	tu-wien, uni-linz, uni-sbg, uni-klu
Kommerziell	derzeit: atnet, cso; prinzipiell alle
	Anbieter von Wählzugängen für Schulen
Schule	kem, htlwrn
Universität	-keine-
Kommerziell	-alle-
	Anbieter von WEB-Space für Schulen
Schule	kem, borg-graz, padl
Universität	tu-graz, tu-wien, uni-klu, uni-linz, uni-sbg, univie
Kommerziell	derzeit: atnet, bnet, carinthia, carrier, compuserve, computerhaus, cso, dascom, geocities, ivnet, netwing, noet, oon, ping, sime, styria, vol; prinzipiell: alle

Ob die allgemeinen Zuwachsraten der Internetzugänge auch im Schulwesen anzuwenden sein werden, können wir durch eine Wiederholung dieser Auswertung nach einiger Distanz sagen. Sie wird auch davon abhängen, wie es dem ASN gelingt, seine Dienste an die Schulen heranzutragen. Immerhin bieten auch viele Provider ihre Dienste an.

Wir laden alle Schulen ein, ihre WWW- und E-Mail bei den angegebenen Verzeichnissen zu hinterlassen, damit wir in einem Jahr diese begonnene Darstellung mit geänderten Daten fortsetzen können. Auch mit Ihrer Schule! □

TYP	NAME	PLZ	ORT	Straße	E-Mail	WWW
AHS	BR G	1010	Wien	Schottenbastei 7-9		http://iguwnext.tuwien.ac.at/~brg1/
BMHS	HTBLA	1015	Wien	Schellingaasse 13		http://www.htlw1.ac.at
BMHSP	Schule für Informatik-Kaufleute	1030	Wien	Salmaasse 6		http://www.sdv.ac.at/wien3/
BMHS	HBLVA Textilindustrie	1050	Wien	Spengergasse 20		http://www.htl-tex.ac.at/spg/spg_overview.html
AHSP	Lycée Francais de Vienne	1090	Wien	Liechtensteinstraße 37a	lfwien@atnet.at	
AHS	BG	1100	Wien	Ettenreichgasse 41-43		http://www.borg-graz.ac.at/grg10/grg10.htm
BMHS	BHAK & BHASCH	1100	Wien	Wendstattgasse 3	aon.912212422@aon.at	
AHS	BG	1150	Wien	Diefenbachgasse 19		http://www.cso.co.at/diefenbachgasse/
FS	BS für Bürokaufleute	1150	Wien	Meiselstraße 19		http://www.kem.ac.at/bsbk/bsbkmain.htm
BMHS	HBLA für Mode und Kunst	1160	Wien	Herbststraße 104		http://www.kem.ac.at/hblm/hblm.htm
AHS	BG	1190	Wien	Billrothstraße 26-30		http://www.bg19.ac.at/bg19/
AHS	BG	1190	Wien	Gymnasiumstraße 83		http://www.cso.co.at/bg19/bg19-home.html
BMHS	TGM	1200	Wien	Wexstraße 19-23		http://www.tgm.ac.at/
BMHS	HBLA f.Wirt.Berufe	1210	Wien	Wassermanngasse 12		http://www.hlt-wien21.ac.at/hlt21/
AHS	BG	1220	Wien	Bernoullistraße 3		http://www.brg22.ac.at/
AHS	BRG	1220	Wien	Polgarstraße 24		http://www.borg-graz.ac.at/~brg22vie/index.htm
BMHS	HTBLA	1220	Wien	Donaustadtstraße 45		http://www.htlvie22.ac.at/
AHSP	AG Erzdiözese Wien	2020	Hollabrunn	Kirchenplatz 2		http://ourworld.compuserve.com/homepages/ag_hollabrunn/homepage.htm
BMHS	HTBLA	2020	Hollabrunn	Dechant-Pfeifer-Str. 1-3		http://ourworld.compuserve.com/homepages/dangl/htl-hl.htm
BMHSP	Schule für Informatik-Kaufleute	2100	Korneuburg	Wiener Straße 42		http://www.sdv.ac.at/korneuburg/
BMHSP	HAK & HASCH	2230	Gänserndorf	Hans-Kudlich-Gasse 30		http://www.kem.ac.at/gaensern/gaensern.htm
AHS	BG & BRG	2320	Schwechat	Ehrenbrunngasse 6		http://doppler.thp.univie.ac.at/~bgschwe/
BMHS	HTBLVA	2340	Mödling	Technikerstraße 1-5		http://www.minic.ac.at/
FS	Wein-Fachschule	2352	Gumpoldskirchen	Kajetan Schellmanng. 27	wein@atnet.at	
BMHS	BHAK & BHASCH	2501	Baden	Mühlgasse 65	hakbaden@xpoint.at	
AHS	BG	2700	Wiener Neustadt	Zehnergasse 15		http://www.ping.at/members/bgzehnwn/
BMHS	HTBLVA	2700	Wiener Neustadt	Dr. Eckenergasse 2		http://www.htlwrn.ac.at/
BMHS	HTBLVA	3100	St.Pölten	Waldstraße 3		http://www.noet.at/htlstp/
HS	Hauptschule	3193	St.Aegyd/NW	Pfarrsiedlung 1		http://www.borg-graz.ac.at/aegyd/aegyd0.htm
HS	Hauptschule	3542	Göhl	Jaidhofergasse 18	hs.gfoehl@wynet.at	
HS	Hauptschule	3701	Grossweikersdorf	Präsident Listplatz 2		http://ourworld.compuserve.com/homepages/ag_hollabrunn/hsgw.htm
AHSP	ORG Linz / Musikgymnasium	4014	Linz	Stifterstraße 27	orgdioez@padl.ac.at	http://www.geocities.com/Athens/4817/index.html
AHS	BRG	4020	Linz	Hamerlingstraße 18		http://www.sea.uni-linz.ac.at/pub/hamerling/home.html
AHS	BORG	4020	Linz	Honauerstraße 24		http://193.170.67.100/schule/borg-honauerstr/
BMHS	HBLA f.Wirt.Berufe	4020	Linz	Landwiedstraße 80		http://www.oon.at/hbla/hblahome.htm
BMHS	HTBLA	4020	Linz	Paul-Hahn-Straße 4		http://www.padl.ac.at/schule/htl-linz/
AHSP	Bischöfl.Gymnasium	4040	Linz	Petrinumstraße 12		http://www.padl.ac.at/petrinum/petrinum.htm
BMHS	HTBLA	4060	Leonding	Limesstraße 12-14		http://elektor.htl-leonding.ac.at/
BMHS	BHAK & BHASCH	4400	Steyr	Leopold-Werndl-Straße 7		http://www.padl.ac.at/schule/hak_steyr/
HS	Hauptschule	4400	Steyr	Annaberg 4		http://w100.padl.ac.at/schule/port/st-anna.htm
SVS	Volksschule	4400	Steyr	Annaberg 4		http://w100.padl.ac.at/schule/port/st-anna.htm
BMHS	Tourismusschulen Bad Ischl	4820	Bad Ischl	Katrinstraße 2	office@hlt-ischl.co.at	
HS	Hauptschule	4971	Aurolzmünster	Schloßstraße 10		http://www.ivnet.co.at/poly2000/index.htm
AHSP	Erzbischöfl.Gymnasium	5020	Salzburg	Gaisbergstraße 7		http://www.sbg.ac.at/geo/gw-unter/borromae/b_home.htm
AHSP	Gymn.Herz Jesu Missionare	5020	Salzburg-Liefering	Schönleitenstraße 1		http://info.cosy.sbg.ac.at/pgl/index.html
BMHS	HTBLA	5020	Salzburg	Itzlinger Hauptstraße 30		http://www.htl.fh-sbg.ac.at/
BMHS	BHAK II	5020	Salzburg	Johann-Brunauer-Str. 2		http://www.dascom.or.at/intswitch/www/wino.htm
FH	FHS Telekommunikationst.	5020	Salzburg	Jakob Haringerstraße 5		http://www.tks.fh-sbg.ac.at/
AHS	BG	5400	Hallein	Schützengasse 3		http://www.borg-graz.ac.at/~hallein/1bg-brg1.htm
FS	Kaufmännische Berufsschule	6020	Innsbruck	Lohbachufer 6c		http://www.kbibk2.ac.at/kbibk2
AHSP	ORG	6111	Volders	Volderwaldstraße 3		http://www.borg-graz.ac.at/~porgvold/
BMHS	BHAK & BHASCH	6370	Kitzbühel	Traunsteiner Weg 11	aon.915310056@aon.at	
BMHS	BLA f.Tourismus	6380	St.Johann/Tirol	Neubauweg 9		http://www.netwing.at/hblatour/
AHSP	Gymn.& Arg	6422	Stams	Stift Stams		http://www.stams.ac.at/meinhardinum/
AHS	BRG	6460	Imst	Füllsackstraße 13	kurt.leitl@telecom.at	
HS	Hauptschule	6710	Nenzing	Landstraße 20		http://www.vol.at/shsnenzing/welcome.html
AHS	BG	6800	Feldkirch	Rebbergasse 25-27		http://www.geocities.com/CapeCanaveral/2186/
FH	FHS Fertigungsautomatisierg.	6850	Dornbirn	Höchsterstraße 73		http://www.tvlbg.ac.at/
FH	FHS Projektmanagement	6850	Dornbirn	Zollgasse 10		http://www.tvlbg.ac.at/
AHS	BORG	6863	Egg	Pfister 925		http://www.vol.at/Borg/Egg/
BMHS	BHAK & BHASCH	6900	Bregenz	Hinterfeldgasse 19		http://www.vol.at/hak-bregenz/index.html
AHS	BORG	6923	Lauterach	Montfortplatz 16a	m.huber@computerhaus.at	http://www.computerhaus.at/vorarlb/borgl/
HS	Hauptschule	6971	Hard-Markt			http://www.vol.at/HS_Markt/
HS	Hauptschule	6971	Hard-Mittelweiherburg			http://www.vol.at/HS_Mittelweiherburg
AHSP	ORG D.Schw.V.Göttl.Erl.	7000	Eisenstadt	Kalvarienbergplatz 8		http://www.bnet.co.at/THERES/index.htm
FS	LWFS	7000	Eisenstadt	Neusiedler Straße 6		http://www.bnet.co.at/lbs_ei/index.htm
BMHS	BHASCH	7132	Frauenkirchen	Kirchenplatz 1 b		http://www.bnet.co.at/hak_fr2/schulpr.htm
BMHS	BHAK & BHASCH	7210	Mattersburg	Michael-Koch-Straße 44		http://www.bnet.co.at/bhak_ma/schule.htm
AHS	BG & BRG & BORG	7350	Oberpullendorf	Gymnasiumstraße 21		http://www.bnet.co.at/brg_op/schule.htm
BMHS	HBLA f.Wirt.Berufe	7400	Oberwart	Badgasse 7		http://www.bnet.co.at/hbla_ow/index.htm
BMHS	HTBLA	7423	Pinkafeld	Meierhofplatz 1		http://www.borg-graz.ac.at/htlpinka/htbl.htm
AHS	Akadem.Gymnasium	8010	Graz	Bürgergasse 15	akg-post@iaik.tu-graz.ac.at	http://www.iaik.tu-graz.ac.at/AKG/
AHS	BORG	8010	Graz	Monsbergergasse 16	adam@borg-6.borg-graz.ac.at	http://www.borg-graz.ac.at/
AHS	BG	8010	Graz	Pestalozzistraße 5		http://www.borg-graz.ac.at/~pestalo/start.htm
AHS	Wirt.BRG f.Mädchen	8010	Graz	Sandgasse 40		http://www.borg-graz.ac.at/~wiku/index.html
AHS	BG & BRG	8010	Graz	Seebachergasse 11		http://www.borg-graz.ac.at/seebach/start.htm
BMHS	BHAK & BHASCH	8010	Graz	Grazbachgasse 71		http://www.borg-graz.ac.at/~bhakgrb/hak.htm
BMHS	BHAK & BHASCH	8010	Graz	Monsbergergasse 16		http://www.borg-graz.ac.at/hakmons/hakmons.htm
BMHS	Expositur Der HBLA	8010	Graz	Ortweinplatz 1		http://www.borg-graz.ac.at/~ortwein/htl_home.htm
FH	FHS Industrial Economics	8010	Graz	Steyrergasse 17		http://www.joanneum.ac.at
FH	FHS Industrial Design	8010	Graz	Steyrergasse 17		http://www.joanneum.ac.at
FH	FHS Industrielle Elektronik	8010	Graz	Steyrergasse 17		http://www.joanneum.ac.at
AHS	BRG	8020	Graz	Keplerstraße 1		http://www.borg-graz.ac.at/kepler/kep_home.htm
AHS	BG	8020	Graz	Oeverseegasse 28		http://www.borg-graz.ac.at/~oeversee/
SAKAD	Pädagogische Akademie	8026	Graz-Eggenberg	Georgigasse 85-87	pae-post@iaik-tu-grak.ac.at	http://www.iaik.tu-graz.ac.at/PAE/index.html
SLEBP	Religionspädak	8026	Graz-Eggenberg	Georgigasse 85-89		http://homepage.sime.com/porg/
BMHS	HTBLVA	8051	Graz-Gösting	Ibererstraße 15-21		http://borg-6.borg-graz.ac.at/bulme.htm
HS	Realschule-Webling	8053	Graz-Neuhart	Unt.Bründlweg 19	rs-graz@cut.big.ac.at	http://www.borg-graz.ac.at/~rs-graz/
AHS	BG	8103	Rein	Stift Rein		http://www.borg-graz.ac.at/~bgrein/
AHS	BG & BRG	8160	Weiz	Offenburgergasse 23		http://www.borg-graz.ac.at/weiz/wz_start.htm
BMHS	HTBLA	8160	Weiz	Dr.Karl-Widdmann-Str.40		http://www.borg-graz.ac.at/htlweiz/htlwhome.htm
AHS	BORG	8190	Birkfeld	Birkfeld 286		http://www.borg-graz.ac.at/birkfeld/borgbir1.htm
BMHS	BFS f.Wirt.Berufe	8230	Hartberg	Edelseegasse 13		http://www.computerhaus.at/steierm/hartberg/home.htm
HS	Hauptschule	8380	Jennersdorf	Schulstraße 2	hs-jennersdorf@bnet.co.at	
HS	Hauptschule	8413	St.Georgen/St.			http://wwwsec.iaik.tu-graz.ac.at/HSG/index.html
AHS	BG & BRG	8430	Leibnitz	Wagnastraße 6		http://www.borg-graz.ac.at/bgleibn/bg&brg.htm
HS	Hauptschule	8510	Stainz	Sauerbrunnstraße 173		http://www.styria.co.at/hs_stainz
BMHS	HTL	8530	Kaindorf	Grazer-Straße 202		http://www.borg-graz.ac.at/~kaindorf/german.htm
AHS	BG & BRG	8580	Köflach	Piberstraße 15		http://www.computerhaus.at/steierm/wizard/bg.htm
AHS	BG & BRG	8600	Bruck	Keplerstraße 2	Andrea.Mayer@iic.wifi.at	http://www.borg-graz.ac.at/~bgbruck/bruck1.htm
FH	FHS Industriewirtschaft	8605	Kapfenberg	Werk VI Straße		http://www.joanneum.ac.at
FH	FHS Industrielle Elektronik	8605	Kapfenberg	Werk VI Straße		http://www.joanneum.ac.at
FS	Waldbauernschule	8662	Mitterdorf			http://www.computerhaus.at/steierm/mitterdorf/home.htm
FS	LWFS & BS f.Burschen	8720	Knittelfeld	Kobenz 4		http://www.zeltweg.computerhaus.at/regional/LBSK/
AHSP	Abteigymn.Benediktiner	8732	Seckau	Abtei Seckau		http://www.iaik.tu-graz.ac.at/PAE/index.html
AHS	BG & BRG	8750	Judenburg	Lindfeldgasse 10	jdbrg@borg-6.borg-graz.ac.at	http://www.borg-graz.ac.at/~jdbrg/jd_start.htm
AHSP	Stiftsgymn.Benedikt.	8911	Admont	Stift Admont 1		http://www.liezen.computerhaus.at/schulen/Stifsgym/a_start1.htm
HS	Hauptschule	8940	Liezen	Hauptstraße 20		http://www.liezen.computerhaus.at/schulen/hsliez/hsliez.htm
BMHS	BHAK & BHASCH I	9020	Klagenfurt	Kumpfgasse 21		http://www.borg-graz.ac.at/~khak/hak1.htm
BMHS	BHAK & BHASCH II	9020	Klagenfurt	Kumpfgasse 21		http://www.borg-graz.ac.at/~khak/hak2.htm
BMHS	HTBLA	9020	Klagenfurt	Mössingerstraße 25		http://www.htblmo-klu.ac.at/
BMHS	Zweispr.BHA	9020	Klagenfurt	Professor-Janezic-Platz 1		http://www.borg-graz.ac.at/~khak/hak3.htm
HS	Hauptschule	9020	Klagenfurt	Weichselhofstraße 6	hs11-klu@edu.uni-klu.ac.at	http://www.edu.uni-klu.ac.at/~hs11-klu/
AHS	BG	9063	Maria Saal	Tanzenberg		http://edusrv.edu.uni-klu.ac.at/~bgtanzen/
BMHS	BHAK & BHASCH	9330	Treibach	Friesacherstraße 4		http://www.borg-graz.ac.at/~khak/althofen.htm
AHS	BG & BRG	9501	Villach	St.Martiner Straße 7		http://www.borg-graz.ac.at/~bgvill/
BMHS	HBLA f.Wirt.Berufe	9501	Villach	Richard-Wagner-Straße 8	hbla-villach@mail.carinthia.co.at	http://www.carinthia.co.at/hbla_villach
BMHS	HTBLVA	9501	Villach	Tschinowitscherweg 5		http://www.kem.ac.at/villach/htl.htm
BMHS	BHAK & BHASCH	9560	Feldkirchen	Flurweg 3		http://www.borg-graz.ac.at/~khak/feldk.htm
FH	Technikum Kärnten	9800	Spittal an der Drau	Ortenburgerstraße 27	info@FH-Spittal.ac.at	http://www.FH-Spittal.ac.at


Verbindungswege im Internet in
PCNEWS
Wien
atnet b2
atnet macao
atnet hit-wien21
atnet memo
atnet fiction
brg22 lukas
ebone vienna-EBS1
atnet router1
eunet vienna1
eunet vienna5
ping pong
cso hermes1
cso atlas
plus vienna-rbs
bg19 www
aco vienna-RBS
aco cell
cc.univie ecstasy..saturn
cc.univie cerberus/doppler
bmuvie bb-router
aco tu-wien
aco.tu-wien htl-schelling
htlw1 ntsa
tuwien etudn-eturgh-eturmw
tuwien iguwnext
apa cisco-VIX
193.186161.46
apa cisco-wien2-vix
194.112.158.18
apa cisco-wien1
194.112.158.233
193.170.115.25
htl-tex www
tgm ddl-l-tgm
tgm argyris
193.107.108.6
kem www
193.170.115.10
193.170.115.18
htlvie22 susi
193.170.115.19
minic peserv
Burgenland
eunet eisenstadt
bnet www
Niederösterreich
noet serv1
noet www
193.170.149.6
htlwrn www
tu-graz rtstg
aco graz
sime rttu/rtsi
sime homepage
borg-graz borg-1
borg-graz borg-6
Steiermark
192.231.67.3
joanneum pdib32
iaik.tu-graz gnietzsche
129.27.234.21
isik.tu-graz www
cso autronet
computerhaus www
apa cisco-graz1-wien2-2
193.46.237.98
styria www
aco linz-atm-test
aco linz
padl router
padl w100
Oberösterreich
edvz.uni-linz jkurtl1
192.107.125.1
htl-leonding elektor
edvz.uni-linz jkurtl02,03
sea.uni-linz turing
apa cisco-linz1-wien1
193.46.237.154
oon fws/wimmer
aco klu
edvz.uni-klu ubwcis
edu.uni-klu edusrv
Kärnten
193.171.18.23
193.171.18.24
193.171.119.41
FH-Spittal fhs1x01
net4you isdnklu/wanklu
193.171.18.6
htblmo-klu asterix
apa cisco-klag1-wien2-2
193.46.237.102
carinthia hyperg
aco sbg
edvz.sbg drout1
141.201.105.2
cosy.sbg hirsch
edvz.sbg dwst16
Salzburg
141.201.106.2
fh-sbg fh-htl-router
fh-sbg htl-fh-router
fh-sbg.htl www
194.177.148.10
dacom syscon
aco dornbirn
tvlbg router-11
tvlbg morrow
Vorarlberg
apa cisco-breg1-wien2
tele br-apa
tele trubadix
net4you isdnibk1
kbibk2 www
194.112.158.50
194.112.149.6
netwing ns1.ibk
Im Bereich der Schulen
International
ebone munich-EBS
ECRC ECRC-Compuserve
...(4)
compuserve ntdwwaad
ebone sth-vie-ebs
198.67.131.177
icp ...(2)
bbnplanet ...(4)
geocities www
"provider" rechnername
Rechner im Internet
Gespeicherte Schul-WWW-Seiten
Modemeinwahl möglich

Österreichische Schulen und ihre Service-Provider

Providerart	Provider	• Schule; **fett** eigener Rechner **fett** Support für andere Schulen
FH	**fh-sbg**	• **S HTBLA Salzburg Itzlinger Hauptstraße**
Schule	**bmuvie**	• **W HBLVA Textil Wien 5 Spengergasse** • **W HTBLA Wien 22 Donaustadtstraße** • **W TGM Wien 20 Wexstraße** • **N HTBLVA Mödling** • **N HTBLVA Wr.Neustadt** • **W KEM Wien 13 Feldmühlgasse**
	borg-graz	• **St** HTBLVA Graz-Gösting • **St** BR & BRG Bruck/Mur • **St** BG Rein • **St** BHAK & BASCH Graz • **St** BR & BRG Judenburg • **St** HTL Kaindorf • **St** BG Graz Oeverseegasse • **St** HBLA Graz Ortweinplatz • **St** BG Graz Pestalozzistraße • **St** BG & BRG Leibnitz • **St** BORG Birkfeld • **St** BHAK & BHASCH Graz Monsb. • **St** HTBLA Weiz • **St** BRG Graz Keplerstraße • **St** BG & BRG Graz Seebachergasse • **St** BG & BRG Weiz • **St** Wirtschaftsk. BRG Graz Sandgasse • **St** BHAK Althofen/Treibach • **St** Graz Weibling • **N** Hauptschule St-Ägyd/Neuwald • **W** BG Wien 10 Ettenreichgasse • **W** BG Wien 13 • **W** GRG Wien 16 • **W** BRG Wien 22 Polgarstraße • **B** HTBLA Pinkafeld • **S** BG & BRG Hallein • **T** ORG Volders • **K** HAK I Klagenfurt Kumpfgasse • **K** HAK II Klagenfurt Kumpfgasse • **K** 2-spr HAK III Klagenfurt • **K** BHAK Feldkirchen • **K** BG Villach
	kem	• **W** BS für Bürokaufleute Wien 15 • **W** HBLA für Mode und Kunst Wien 16 • **N** HAK & HASCH Gänserndorf • **K** HTBLVA Villach
	padl	• **O** BORG-Linz Hohenauerstraße • **O** Hauptschule Steyr • **O** Bischöfl. Gymnasium Linz Petrinumstraße • **O** BHAK & BHASCH Steyr • **O** HTL Linz Paul-Hahn-Straße

Anbindung über das ASN *(oben)*

Die Tabelle zeigt, daß das Schulnetz **6 Standleitungsanbindungen** (5 ausgehend vom Schulrechenzentrum (bmuvie) und eine von der Fachhochschule Salzburg (fh-sbg) (?)) sowie **41 Schulpräsentationen** (als WWW-Seiten) anbietet.

Anbindung über andere Provider *(rechts)*

Kommerzielle und universitäre Provider bieten **13 Standleitungsanbindungen** und **48 Schulpräsentationen**.

Das heißt, österreichs Schulen präsentieren sich derzeit mehrheitlich durch Eigeninitiative im Internet. Wäre da nicht die Grazer Monsbergergasse, wäre es um das ASN schlecht bestellt. Es wäre den engagierten Steirern zu wünschen, daß sich die vorhandenen Geldmittel proportional zur Leistung aufteilen mögen.

Providerart	Provider	• Schule; **fett** eigener Rechner **fett** Support für andere Schulen
Kommerziell international	**compuserve**	• **N** AG Erzdiözese Wien- Hollabrunn • **N** Hauptschule Großweikersdorf • **N** HTBLA Hollabrunn
	geocities	• **O** ORG Linz Stifterstraße • **V** BG Feldkirch
Kommerziell	**atnet**	• **W** HLT Wien 21, Tourismusschulen • **W** Informatik-Schule Wien 3 Salmgasse • **N** Informatik-Schule Korneuburg • **W BRG Wien 22 Bernoullistraße**
	bnet	• **B** BHAK & BHASCH Mattersburg • **B** BG & BRG & BORG Oberpullendorf • **B** BHASCH Frauenkirchen • **B** HBLA f. Wirt.Berufe Oberwart • **B** LWFS Eisenstadt • **B** ORG d. Schw.v.Göttl.Erl. Eisenstadt
	carinthia	• **K** HBLA f. wirt. Berufe Villach
	carrier	• **T** Kaufm. Berufsschule Innsbruck
	computerhaus	• **St** (Stmk) BFS f. wirt. Berufe Hartberg • **St** Waldbauernschule Mitterndorf • **St** BG & BRG Köflach • **St** Hauptschule Liezen • **St** Landesberufschule Knittelfeld • **St** Stiftsgymnasium Admont • **V** BORG Lauterach
	cso	• **W BG Wien 19, Billrothstraße 26-30** • **W** BG Wien 19 Gymnasiumstraße 83 • **W** BG Wien 15 Diefenbachgasse
	dascom	• **S** BHAK II Salzburg
	ivnet	• **O** HS Aurolzmünster
	netwing	• **S** HBLA in St.Johann
	noet	• **N** HTBLVA St.Pölten
	oon	• **O** HBLA Linz Landwiedstraße
	ping	• **N** BG Wiener Neustadt Zehnergasse • **T** Gymnasium Stams
	sime	• **St** Religionspädak Graz-Eggenberg
	styria	• **St** Hauptschule Stainz
	vol	• **V** Hauptschule Hard-Markt • **V** Hauptschule Hard-Mittelweiherburg • **V** Sporthauptschule Nenzing • **V** HAK-Bregenz • **V** BORG Egg
Universität	**aco**	• **St BORG Graz Monsbergergasse** • **K Technikum Kärnten** • **K HTBLA Klagenfurt Mössingerstraße** • **St Fachhochschulst. Graz/Kapfenberg** • **V Technikum Vorarlberg** • **O padl** • **W bmuvie**
	tu-graz	• **St** Akadem. Gymnasium Graz Bürgergasse • **St** Abteigymnasium Benediktiner Seckau • **St** Pädagogische Akademie Graz-Eggenberg • **St** Hauptschule St.Georgen/Stiefling
	tu-wien	• **W** BRG Wien Schottenbastei • **W HTBLA Wien 1 Schellinggasse**
	uni-klu	• **K** BG Maria Saal • **K** Hauptschule Klagenfurt Weichselhofstraße
	uni-linz	• **O HTBLA Leonding** • **O** BRG Linz Hamerlingstraße
	uni-sbg	• **S Gymn.Herz Jesu Salzburg Liefering** • **S** Erzbischöfl. Gymnasium Salzburg • **S Fachhochschule Salzburg**
	univie	• **W** BG & BRG Schwechat

Statistik nach Bundesländern und Schultypen

Zahl der Schulen	Schulen Gesamt	Schulen Internet	Ö	Vbg	B	Stm	Knt	W	Sbg	OÖ	NÖ	Ti
			2218	82	75	390	174	321	166	378	432	201
			110	8	7	36	12	15	6	11	11	5
Österreich	2126	117	**5,4%**	**10%**	**9,3%**	**9,2%**	**6,9%**	**4,7%**	**3,6%**	**2,9%**	**2,5%**	**2,5%**
FH	24	9	**38%**	2		5	1		1			
AHS	315	42	**13%**	3	2	16	2	7	3	4	3	2
BMHS	500	40	**8%**	1	4	7	8	7	2	4	6	1
PÄD	37	2	**5%**			2						
FS	184	5	**3%**		1	2		1				1
HS	1066	12	**1%**	2		4	1			2	2	1
VS	...	1	**...**							1		

- Bei den Fachhochschulen ist der Anteil zwar hoch, aber da oft von einem Server mehrere Fachhoschulstudiengänge versorgt werden, ist der Prozentsatz möglicherweise zu nieder.
- Die Zahl der Schulen pro Bundesland (2218) und pro Schulart (2126) ist verschieden, da bei der Gesamtzahl der Schulen die Universitäten und einige Institute sowie Berufsschulen mitgezählt wurden.
- Die Auswertung beruht auf einer noch kleinen Anzahl erfaßter Schulen, daher sind die Prozentzahlen großzügig zu bewerten. Man sollte daher statt der Zahlen besser davon sprechen, daß in Vorarlberg, in der Steiermark, im Burgenland und in Kärnten eine ***stärkere***, in Wien und Salzburg eine ***mittlere***, in Oberösterreich, Tirol und Niederösterreich eine eher ***geringere*** Internetpräsenz vorhanden ist.

5,4 % der österreichischen Schulen und Bildungseinrichtungen können im WWW des Internet erreicht werden. Das Vorarlberger Konzept macht sich bemerkbar, Vorarlberg läßt das benachbarte Tirol weit hinter sich. Bemerkenswert die Spitzenposition des Burgenlandes, gefestigt durch den rührigen Provider bnet, und der Steiermark, dank der Aktivitäten von Hans Adam am BORG Graz Monsbergergasse. Enttäuschend die Position von Wien, keine Spur davon, daß es in Wien vergleichsweise preiswert ist, Standleitungsverbindungen zu unterhalten, keine Spur davon, daß die Zentrale des ASN in Wien beheimatet ist.

MSDN

Microsoft Developer Network: Neue Namen, neue Vertriebswege, neue Verteilform, neuer Level, neuer Preis (für Lehrer/Schulen).

Franz Fiala

Was ist das MSDN?

In 3-monatigen Abständen erscheint das 4-stufige Microsoft Developer Network und wird nur an Abonnenten abgegeben. Ein Abonnement dauert ein Jahr. Die PCNEWS berichteten darüber in den Heften 31, S.26 und 37, Seite 22. Die aktuelle Version ist vom Juli 1996. Das Microsoft Developer Network dient der Unterstützung der Entwickler, die

- ein Produkt zeitgleich mit einem neuen Betriebssystem auf den Markt bringen wollen,
- ein Produkt in verschiedenen Sprachversionen testen wollen,
- die Treiber für eigene Hardware entwickeln,
- die Betriebssysteme testen wollen, ohne sie gleich endgültig zu kaufen.

Bisher konnte man die CDs des Developer Networks in 3 Stufen beziehen. Dia Anzahl der CDs in der folgenden Tabelle beziehen sich auf die April-Ausgabe 96:

	Anzahl der CDs	Inhalt
Level 1	2	**Informationen, Handbücher, Spezifikationen für Entwickler**
Level 2		**Betriebssysteme, SDKs und DDKs**
	7	*Teil 1:* US-Versionen der aktuellen Betriebssysteme (DOS, WIN31, WIN95, WINNT351, WIN40 Beta2) sowie der nicht-lokalisierten SDKs;
	12	*Teil 2:* Internationale Versionen soweit übersetzt;
	4	*Teil 3:* DDKs
Level 3	8	**Back-Office:** NT-Server, SQL-Server, System Management, Server, Mail Server, SNA Server, Exchange Server

Ein Level enthält jeweils alle angegebenen CDs und auch jene des kleineren Level. Mit Level 3 hat man daher alle CDs.

Weiters erhalten alle Abonnenten die englischsprachige Ausgabe der „Developer Network News“, sowie den deutschsprachigen „Windows Monitor“.

Daneben gibt es auch ein Abonnement für die jeweils letztgültige Version von Visual C++ (1-2 CDs) pro Quartal sowie TechNet, eine Sammlung aktueller Treiber und sonstiger Informationen für Systembetreuer.

Mit Juli-96 wurde das Verteilungssystem reorganisiert. Die Neuerungen sind vielversprechend und machen das MSDN auch für Lehrer und Schulen interessant.

Neue Namen

Die Namen des MSDN wurden an die Bezeichnungen der anderen Softwareprodukte angepaßt. Hier die neuen Namen:

Alte Bezeichnung	***Neuer Name***
Level 1	**Library** Subscription
Level 2	**Professional** Subscription
Level 3	**Enterprise** Subscription

Neue Vertriebswege

Bisher konnte man das MSDN nur auf dem direkten Bezugsweg aus den Niederlanden beziehen.

Microsoft Subscription Centre
P.O.Box 5147
NL-2900 EC Capelle a/d Ijssel
Niederlande
Tel.: +31-10-258 88 64
Fax.: +31-10-258 88 63
Tel.: (deutsch) +43-130-8102 11

Diese Adresse gilt nach wie vor. Ab sofort ist es aber möglich, das MSDN auch über Händler zu beziehen. Ein Preisbeispiel finden Sie am Ende. Weiters kann man die Library-Ausgabe auch unabhängig von einem Dauerbezug um etwa S 1000,- im Handel kaufen.

Neue Verteilform

Bisher erhielten die Abonnenten jedes Quartal einen neuen Satz von CDs. Was vor vier Jahren, als MSDN gestartet wurde, noch praktikabel war, wurde unhandlich: 36 CDs pro Lieferung! Vieles hatte man mehrfach, weil sich einfach nichts änderte; vieles fand man nicht, weil es auf einer früheren Version war. Jetzt gibt es einen massiven Ordner, in dem alle CDs Platz finden. In der nächsten Sendung werden nur mehr CDs ausgeliefert, die sich ändern.

Einen kompletten Index des Inhaltes finden die Abonnenten und Interessenten auf `http://www.microsoft.com/msdn/index/`.

Neu: „Universal“ Subscription“

Alte Bezeichnung	***Neuer Name***
Level4 << neu >>	**Universal** Subscription

Völlig neu ist diese umfassendste (und teuerste) Stufe der Information, bei der vor allem MS-Office hervorzuheben ist. Sie enthält zusätzlich zur Enterprise-Subscription:

- Visual C++ 4.1 Professional Edition
- Visual Basic 4.0a, Enterprise Edition
- Visual SourceSafe
- Visual FoxPro 3.0b, Professional Edition
- Office Test Platform (Microsoft Office Professional für Windows 95 Ver.7.0b und MS-Project für Windows 95 Version 4.1)
- Access Developer’s Toolkit für Windows 95
- Internationale Versionen für MS-Office Professional für Windows 95
- Alle Upgrades für diese Produkte innerhalb eines Jahres
- Ausgewählte Internet Entwicklungs-Tools, die im Laufe des Jahres erscheinen.

Neue Preise

Bisher gab es lediglich einen Preis: den des Direktbezugs aus den Niederlanden, Zahlung nur per Kreditkarte. Ab Juli 1996 kann man möglicherweise über lokale Händler günstiger kaufen. Derzeit (bis 16.9.96) gibt es besondere Angebote für Abonnenten, die auch C++ beziehen. Wegen der neuen Bezugsmöglichkeit, ist es ratsam, die genauen Preise bei Ihrem Händler (vielleicht bei einem PCNEWS-Inserenten?) zu erfragen.

Neuer Name	***Normal***	***Schule***
Library Subscription	ca. 2700,-	ca. 1200,-
Professional Subscription	ca. 7000,-	ca. 4800,-
Enterprise Subscription	ca. 20000,-	ca. 13000,-
Universal Subscription	ca. 32000,-	ca. 20000,-

□

Ärger mit den Laufwerksbuchstaben

Sepp Melchart

Wer in seinem Rechner schon einmal eine 2. physische Festplatte unter DOS installiert hat, der hat sich möglicherweise über die neuen Laufwerksbuchstaben geärgert. Die neue Platte drängt sich nämlich mit dem Buchstaben D: mitten in die vorhandenen logischen Laufwerksbuchstaben.

Beispiel

1 Festplatte, auf 3 logische Laufwerke C:, D:, E: aufgeteilt (landläufig unpräzise als „Partitions“ bezeichnet), und 1 CD-ROM-Laufwerk:

	C:	D:	E:	F:
1. Festplatte:	1.1	1.2	1.3	
CD-ROM-Laufwerk:				CD

Nach dem Einbau der 2. Festplatte (mit 2 logischen Laufwerken):

	C:	D:	E:	F:	G:	H:
1. Festplatte:	1.1		1.2	1.3		
2. Festplatte:		2.1			2.2	
CD-ROM-Laufwerk:						CD

Die erste "Partition" der 2. Festplatte hat sich mit D: hineingedrängt, dadurch verschieben sich die Laufwerksbuchstaben der 1. Platte von D: auf E: und von E: auf F:, was z.B. für installierte Windows-Programme gravierende Folgen hat: Die alten Pfade stimmen nicht mehr, die Programme laufen nicht mehr und müssen im allgemeinen neu installiert werden (sofern man kein Freak ist, der gern in INI-Dateien und noch sonstwo herumhackt). Oder man kopiert sie auf die neue Platte 2.1.

Das CD-ROM-Laufwerk wird „hintendran“ installiert und bekommt ebenfalls einen neuen Buchstaben.

Abhilfe kann der DOS-Befehl `SUBST` bieten (DOS 6.2). Mit ihm kann man einen neuen logischen Laufwerksbuchstaben für ein vorhandenes Verzeichnis vergeben.

In unserem Beispiel:

```
SUBST I: G:\   d.h. Verzeichnis G:\ wird mit log. Laufwerk I: angesprochen.
SUBST G: D:\
SUBST D: E:\
SUBST E: F:\
```

Die richtige Reihenfolge dieser Befehle muß eingehalten werden! Man kann diese Befehle in eine Batch-Datei schreiben oder gleich in die `AUTOEXEC.BAT`.

Der Vorrat an Laufwerksbuchstaben muß vorher in der `CONFIG.SYS` mit dem Befehl `LASTDRIVE=I` (irgendein Buchstabe von I bis Z) entsprechend erhöht werden (Neustart nach Änderung der `CONFIG.SYS` nicht vergessen!).

Es ergeben sich dann folgende neue Laufwerksbuchstaben:

	C:	D:	E:	F:	G:	H:	I:
1. Festplatte:	1.1	1.2	1.3	1.3			
2. Festplatte:					2.1		2.2
CD-ROM-Laufwerk:						CD	

Somit sind die alten Laufwerksbuchstaben der 1. Festplatte wiederhergestellt, und die installierten Programme laufen weiterhin.

Nicht behebbarer Schönheitsfehler: Die logischen Laufwerksbuchstaben `E:` und `F:` sind dann identisch = 3. Partition der 1. Festplatte (1.3).

Eine weitere Zuordnung, z.B. `SUBST F: I:\` wird nicht akzeptiert, weil `I:` bereits ein substituiertes logisches Laufwerk ist. Eine echte Vertauschung von Laufwerksbuchstaben ist mit diesem DOS-Befehl somit leider nicht möglich (typisch DOS!).

Auch eine Verlegung des CD-ROM-Laufwerks mißlingt generell, z.B. von `H:` auf `F:` mit `SUBST F: H:\`, weil `H:` bereits eine logische Zuordnung darstellt. Daher müssen installierte Programme, die auf CD zugreifen (z.B. Herold-Telefonbuch, Lexika, etc.), jedenfalls pfadangepaßt werden.

Ein Tip: Den logischen Laufwerken mit `LABEL` sinnvolle Namen geben, sodaß man auch später noch weiß, welche physische Platte dahintersteckt. Weiters die `AUTOEXEC.BAT`-Datei mit den `SUBST`-Befehlen sichern (Notfall-Startdiskette!) oder zumindest notieren.

Wozu überhaupt Partitionierung unter DOS?

Die Verwaltung großer Partitionen unter DOS ist ziemlich ineffizient. Es kann nämlich nur eine begrenzte Anzahl von Clustern (= Blockgröße, Zuordnungseinheit in kB) verwaltet werden. Größere Partitionen haben daher größere Cluster. Eine 100 MB-Partition hat z.B. eine Clustergröße von 2 kB (2048 Byte), bei 240 MB sind es 4 kB, bei 1,6 GB bereits 32 kB! Die Clustergröße kann mit dem Befehl `CHKDSK` angezeigt werden (Anzahl der Bytes pro Zuordnungseinheit).

Eine Datei belegt immer eine ganzzahlige Anzahl von Clustern auf der Platte. Jede noch so kleine Datei belegt somit mindestens 1 Cluster. Bei kleinen Dateien (Daten, INI-, Batch-Dateien) wird auf diese Art ziemlich viel Platz verschwendet. Ein Kollege mit einer vollen 1 GB-Partition rechnete einmal nach (`DIR` zeigt ja die tatsächlichen Dateigrößen und ihre Summe an, `CHKDSK` den belegten Speicherplatz) und kam auf einen „Verlust“ von über 200 MB! Hat man z.B. viele kleine Datendateien, so ist eine eigene "kleine" Partition dafür sinnvoll (Das erleichtert u.U. auch die Datensicherung).

Leider kann die Partitionierung nachträglich mit `FDISK` nicht ohne Datenverlust geändert werden. Es lohnt sich daher, beim Einbau einer großen Festplatte genau über eine sinnvolle Partitionierung nachzudenken.

□

The Differences Between Intel Chips

By Dave Smith

For those who are confused by the various processor offering by Intel, here's a quick guide to clear you up.

8086 *A spacious closet. Has four walls and plenty of shelf space. It's a lot better than keeping your stuff in stacked cardboard boxes like you had to with the 8080*

8088 *A spacious clost like the 8086, but lacks the mirrored doors and instead uses a narrower door made of standard plywood. Not as pretty, but a less expensive.*

80186 *Still a closet with the mirrored doors, but now includes a new railing to hang your clothes and this one is bolted on, not that wooden thing that always falls when you bump it off. Also the doors run on a better track. Not the greatest, but the builder was a little slow.*

80286 *A closet as with the 8086, but includes a small trap door on the ceiling that you can climb up with great effort to reach a larger attic to store stuff in. The door is sufficiently small that you can only shove small things in or out of it.*

80386 *A 1930s victorian house somewhat rundown, but acceptable. Has a closet, but also has a large garage. You can be in the garage or the closet, but you can't have a closet or a workbench in the garage. An option in building allows you to build out the house into several small closet size apartments. Each occupant of an apartment is unaware they have any neighbors. (Unless they give a note to you to post on the board in front hall for other neighbors to see.)*

80386SX *Another 1930s house, but this one doesn't have the double doors in the entry.*

80486 *A small condo with closets.*

80486SX *The same condo, but with strong owners association. You can buy the condo for less money, but you can't get the keys unless you buy them from owners association for a large fee.*

Rekursion

in Pascal, C und BASIC

Karel Štípek

Stellen Sie sich vor, Sie sollten folgende Aufgabe lösen:

```
  SEND
+ MORE
------
 MONEY
```

Die Buchstaben sollen durch Ziffern so ersetzt werden, daß das Beispiel mathematisch stimmt.

Als ich diese Aufgabe bekommen habe, hatte ich Lust ein allgemeines Programm zu schreiben, mit dem man auch andere ähnliche Aufgaben lösen kann. Prinzipiell geht es darum, alle möglichen Ziffernkombinationen zu testen und die richtige Lösung anzuzeigen. Gut, nehmen wir Turbo-Pascal zu Hilfe und beginnen wir. Anfang ist nicht schwer:

PASCAL

```
program algebgr; uses dos, crt;
```

In unserem Beispiel gibt es 8 verschiedene Buchstaben, diese Zahl definieren wir als Konstante `MAXB`. Die 8 möglichen Ziffern fassen wir zu einem Array `B` zusammen. Alle Buchstaben können Werte von 0 bis 9 haben, nur `S` und `M` (Anfang des Wortes) sollten größer Null sein. Die Grenzwerte werden in den Arrays `VON` und `BIS` definiert. Jede Ziffer darf in der Kombination nur einmal auftreten, welche verwendet wird, speichern wir im Array `USED`.

```
const maxb = 8;                      { S E N D M O R Y }
      von:  array[1..maxb] of byte = (1,0,0,0,1,0,0,0);
      bis:  array[1..maxb] of byte = (9,9,9,9,9,9,9,9);
      used: array[0..9] of boolean = (false, false, false,
            false,false,false,false,false,false,false);
      var b:   array[1..maxb] of longint;
```

Kern des Programms stellt die Funktion `TEST` dar. Die wird alle möglichen Ziffernkombinationen erstellen und testen. Parameter `I` bezeichnet das Index im Array `B`, also Reihenfolge des Buchstabens. `J` und `K` sind lokale Hilfsvariablen.

```
procedure test(i:integer);
var j,k: byte;
begin
```

Der Wert des gerade bearbeiteten Buchstabens wird sich im vordefinierten Intervall ändern.

```
  for j:= von[i] to bis[i] do
```

Der Wert ist nur dann zulässig, wenn die Ziffer in der Kombination noch nicht auftritt, in dem Fall wird er im Arrayelement `B` gespeichert und die Ziffer wird in `USED` „besetzt“.

```
    if not used[j] then begin
      b[i]:= j; used[j]:= true;
```

Jetzt kommt der wichtigste Trick, der uns viel Code erspart. Das gleiche, was wir mit dem ersten Buchstaben gemacht haben, machen wir auch mit dem nächsten, solange wir nicht beim letzten sind. Die Prozedur `TEST` ruft sich selbst neu auf - eine sogenannte Rekursion.

```
      if i < maxb then TEST(i+1)
```

Wenn wir den letzten Buchstaben bearbeiten, zeigen wir einfach die erstellte Kombination an (immer in der gleichen Zeile)

```
        else begin write(#13);
        for k:= 1 to maxb do write(b[k]:2);
```

und testen, ob die aktuellen Buchstabenwerte unsere Bedingung erfüllen.

```
        if ((b[1]*10 + b[2])*10 + b[3])*10 + b[4] +
           ((b[5]*10 + b[6])*10 + b[7])*10 + b[2] =
           (((b[5]*10 + b[6])*10 + b[3])*10 + b[2])*10 + b[8]
then
```

Wenn ja, dann schreiben wir neben der Ziffern „OK“ und setzen auf der nächsten Zeile fort.

```
            begin write('  OK'); writeln; end;
        end;
```

Nachdem eine Ziffer in allen Kombinationen mit den weiter rechts liegenden Ziffern verwendet wurde, muß sie wieder „freigesetzt“ werden.

```
      used[b[i]]:= false; end;
end; {test}
```

Das Hauptprogramm besteht aus einer einzigen Zeile - dem Aufruf der Funktion `TEST` ab dem ersten Buchstaben.

```
begin
  test(1);
end.
```

Auf dem Bildschirm sehen Sie eine Zeile von Ziffern, die sich schnell ändern. Immer, wenn eine gültige Lösung gefunden wird, erscheint die Kombination mit „OK“ und die Anzeige wird auf der nächsten Zeile fortgesetzt.

C

Bei der Übersetzung dieses Programms in C gibt es keine nennenswerten Besonderheiten.

```
/* REKURS.CPP */
/* VISUAL-C++, Vers. 4.1 */

#include <stdio.h>

#define TRUE -1
#define FALSE 0
const int MAXB = 9;
char von[] = {0, 1, 0, 0, 0, 1, 0, 0, 0, 0}; // S E N D M O R Y
char bis[] = {9, 9, 9, 9, 9, 9, 9, 9, 9, 9};
int used[10] = {FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE };
long b[MAXB];

void test(int i)
{
  char j, k;
  for (j=von[i]; j<=bis[i]; j++)
  {
    if (!used[j])
    {
      b[i] = j;
      used[j] = TRUE;
      if (i<MAXB)
      {
        test (i + 1);  // Rekursion
      }
      else
      {
        printf("\x0d");
        for (k=1; k<=MAXB; k++)
        {
          printf("%i ",b[k]);
        }
        if (((b[1] * 10 + b[2]) * 10 + b[3]) * 10 + b[4] + // SEND
            ((b[5] * 10 + b[6]) * 10 + b[7]) * 10 + b[2] == // MORE
            (((b[5] * 10 + b[6]) * 10 + b[3]) * 10 + b[2]) * 10 + b[8]) // MONEY
        {
          printf(" OK\n");
        }
      }
      used[b[i]] = FALSE;
    }
  }
}

void main(void)
{
  test(1);
}
```

BASIC

Die Übersetzung des Programms in BASIC war bezüglich der Initialisierung der Arrays etwas abweichend. Uninitialisierte Arrays wie etwa `used` oder `b` werden mit dem richtigen Typ deklariert. Bei den Arrays, die einen Anfangswert erfordern (`von, bis`) müßte man die Anfangswerte im Rahmen einer `For-Next`-Schleife setzen, was nicht besonders elegant ausschaut. In diesem Beispiel wurden daher die Variablen `von` und `bis` als Typ Variant gewählt, der die Initialisierung mit der Funktion `Array` erlaubt. Für optimale Laufgeschwingigkeit sollte man aber den Typ Variant vermeiden.

Als Testausgabe wurde das Objekt `Debug` mit der Methode `Print` verwendet. Zur Beschleunigung des Ablaufs wurde aber die laufende Ausgabe jedes Versuchs ausgeklammert. Es werden nur die gültigen Lösungen ausgegeben.

```
'VISUAL BASIC for APPLICATIONS (ACCESS)
Option Explicit
Const maxb As Integer = 9                    ' S E N D M O R Y
Dim von As Variant
Dim bis As Variant
Dim used(0 To 9) As Boolean
Dim b(1 To maxb) As Long
Dim i As Byte
```

```
Public Sub test(i As Integer)
  Dim j, k As Byte
  For j = von(i) To bis(i)
    If Not used(j) Then
      b(i) = j
      used(j) = True
      If i < maxb Then test (i + 1)
      Else
        'For k = 1 To maxb
        '  Debug.Print (b(k));
        'Next k
        'Debug.Print
        If ((b(1) * 10 + b(2)) * 10 + b(3)) * 10 + b(4) + _
           ((b(5) * 10 + b(6)) * 10 + b(7)) * 10 + b(2) = _
           (((b(5) * 10 + b(6)) * 10 + b(3)) * 10 + b(2)) * 10 + b(8) Then
          For k = 1 To maxb
            Debug.Print (b(k));
          Next k
          Debug.Print "  OK"
        End If
      End If
      used(b(i)) = False
  Next j
End Sub

Public Sub testtest()
  von = Array(0, 1, 0, 0, 0, 1, 0, 0, 0, 0)
  bis = Array(9, 9, 9, 9, 9, 9, 9, 9, 9, 9)
  test (1)
End Sub
```

□

CDROM „Delphi Programmiertools“

International Thomson Publishing, ISBN 3-8266-0230-7, öS 233,-

Hans Lohninger

Das CDROM „Delphi Programmiertools“ enthält eine bunte Mischung aus Free- und Shareware und einigen wenigen Demos („Crippleware“), insgesamt 80 Produkte. Die enthaltene Software kann man grob in zwei Kategorien einteilen: zum einen Delphi-Komponenten und -Erweiterungen, und zum anderen allgemeine Zubehör-Software für Programmierer.

Mit den 80 Produkten (die übrigens das CDROM nur zu 8 % belegen, der Rest ist leeres Plastik) wird ein 'CD-Manager' mitgeliefert, der helfen soll, einen besseren Überblick zu den angebotenen Produkten zu bekommen. Man kann sich mit Hilfe des CD-Managers die jeweiligen (deutschen) Hilfetexte ansehen und bei Interesse das jeweilige Software-Produkt auf die Harddisk kopieren.

Dieser CD-Manager ist im Prinzip eine gute Sache. Allerdings ist er wohl etwas zu mager geraten, sodaß der genervte Benutzer sich schnell ein kleines Batch-Programm bastelt, das alle Hilfetexte in ein File kopiert - womit der Überblick wieder da ist. Störend ist auch, daß der CD-Manager anscheinend nicht auf Systemen mit veschiedenen Grafik-Auflösungen getestet worden ist: Bei einer Auflösung von 1024x768 passen einige Elemente der Oberfläche des CD-Managers nicht mehr zusammen.

Die Auswahl der auf der CD enthaltenen Produkte erfolgte meiner Ansicht nach eher nachlässig. So gibt es insgesamt zwar 4 verschiedene Texteditoren im Angebot, allerdings ist zum Beispiel der sehr gute PFE nur in der 32-Bit Version enthalten. Verglichen mit dem Platz, der noch auf der CD frei ist (immerhin runde 600 MB), hätten die Produzenten durchaus noch das eine oder andere Delphi-Juwel mit auf das CDROM nehmen können.

Der Wert des CDROMs wird auch insofern relativiert, als man sich für dessen stolzen Preis von 233,- öS bei entsprechender Hardware-Ausstattung ca. 60 MB aus dem Internet holen kann - also um circa 10 MB mehr als auf der CD vorhanden ist. □

Wie die verschiedenen Programmierer ihre Fahrräder bauen

ADA-Programmierer bauen ein viereckiges Rad und passen alle Straßen an.

ALGOL-Programmierer weigern sich, Räder zu bauen, weil es sie irgendwohin bringen könnte.

APL-Programmierer schweben in höheren Sphären, sie brauchen keine Räder.

ASSEMBLER-Programmierer bauen tausende von Rädern, keine 2 passen auf eine Achse.

BASIC-Programmierer bauen nur ein Rad, aber finden keine Achse dazu.

COBOL-Programmierer bauen TAUSEND-RÄDRIGE-TRANSPORT-MODULE und verbieten das Gehen.

FUNKTIONAL-Programmierer rufen eine Funktion HOLZ auf und hoffen, ein Rad zu bekommen.

FORTH-Programmierer bauen Räder und vergessen, wo sie sie gestapelt haben.

FORTRAN-Programmierer werden wahnsinnig bei der Suche nach Rädern, die mit "I" beginnen.

LOGO-Programmierer bauen kleine rote Autos.

LISP-Programmierer (bauen Räder mit (Rädern mit/Rädern mit (Rädern mit (dem was LISP-Programmierer bauen)))).

MICRO-Programmierer wissen nicht, daß Räder existieren.

PASCAL-Programmierer erklären das Gehen zur Tugend.

PL/I-Programmierer setzen ein Team ein, um eine Räderfabrik zu entwerfen, die Räder der falschen Größe bauen wird.

RPG-Programmierer haben ein Rad, schade, daß es viereckig ist.

SYSTEM-Analytiker sind viel zu beschäftigt, Räder zu suchen, um eins zu bauen.

TEXTVERARBEITUNGS-Benutzer bauen Räder.

TOS-Benutzer bleiben wo sie sind, sie sind das Warten gewohnt.

UNIX hat unter irgendeiner Schale sicher irgendwo ein Verzeichnis von Rädern.

BASIC

Die Übersetzung des Programms in BASIC war bezüglich der Initialisierung der Arrays etwas abweichend. Uninitialisierte Arrays wie etwa `used` oder `b` werden mit dem richtigen Typ deklariert. Bei den Arrays, die einen Anfangswert erfordern (`von, bis`) müßte man die Anfangswerte im Rahmen einer `For-Next`-Schleife setzen, was nicht besonders elegant ausschaut. In diesem Beispiel wurden daher die Variablen `von` und `bis` als Typ Variant gewählt, der die Initialisierung mit der Funktion `Array` erlaubt. Für optimale Laufgeschwingigkeit sollte man aber den Typ Variant vermeiden.

Als Testausgabe wurde das Objekt `Debug` mit der Methode `Print` verwendet. Zur Beschleunigung des Ablaufs wurde aber die laufende Ausgabe jedes Versuchs ausgeklammert. Es werden nur die gültigen Lösungen ausgegeben.

```
'VISUAL BASIC for APPLICATIONS (ACCESS)
Option Explicit
Const maxb As Integer = 9                    ' S E N D M O R Y
Dim von As Variant
Dim bis As Variant
Dim used(0 To 9) As Boolean
Dim b(1 To maxb) As Long
Dim i As Byte
```

```
Public Sub test(i As Integer)
  Dim j, k As Byte
  For j = von(i) To bis(i)
    If Not used(j) Then
      b(i) = j
      used(j) = True
      If i < maxb Then test (i + 1)
      Else
        'For k = 1 To maxb
        '  Debug.Print (b(k));
        'Next k
        'Debug.Print
        If ((b(1) * 10 + b(2)) * 10 + b(3)) * 10 + b(4) + _
           ((b(5) * 10 + b(6)) * 10 + b(7)) * 10 + b(2) = _
           (((b(5) * 10 + b(6)) * 10 + b(3)) * 10 + b(2)) * 10 + b(8) Then
          For k = 1 To maxb
            Debug.Print (b(k));
          Next k
          Debug.Print "  OK"
        End If
      End If
      used(b(i)) = False
  Next j
End Sub

Public Sub testtest()
  von = Array(0, 1, 0, 0, 0, 1, 0, 0, 0, 0)
  bis = Array(9, 9, 9, 9, 9, 9, 9, 9, 9, 9)
  test (1)
End Sub
```

□

CDROM „Delphi Programmiertools“

International Thomson Publishing, ISBN 3-8266-0230-7, öS 233,-

Hans Lohninger

Das CDROM „Delphi Programmiertools“ enthält eine bunte Mischung aus Free- und Shareware und einigen wenigen Demos („Crippleware“), insgesamt 80 Produkte. Die enthaltene Software kann man grob in zwei Kategorien einteilen: zum einen Delphi-Komponenten und -Erweiterungen, und zum anderen allgemeine Zubehör-Software für Programmierer.

Mit den 80 Produkten (die übrigens das CDROM nur zu 8 % belegen, der Rest ist leeres Plastik) wird ein 'CD-Manager' mitgeliefert, der helfen soll, einen besseren Überblick zu den angebotenen Produkten zu bekommen. Man kann sich mit Hilfe des CD-Managers die jeweiligen (deutschen) Hilfetexte ansehen und bei Interesse das jeweilige Software-Produkt auf die Harddisk kopieren.

Dieser CD-Manager ist im Prinzip eine gute Sache. Allerdings ist er wohl etwas zu mager geraten, sodaß der genervte Benutzer sich schnell ein kleines Batch-Programm bastelt, das alle Hilfetexte in ein File kopiert - womit der Überblick wieder da ist. Störend ist auch, daß der CD-Manager anscheinend nicht auf Systemen mit veschiedenen Grafik-Auflösungen getestet worden ist: Bei einer Auflösung von 1024x768 passen einige Elemente der Oberfläche des CD-Managers nicht mehr zusammen.

Die Auswahl der auf der CD enthaltenen Produkte erfolgte meiner Ansicht nach eher nachlässig. So gibt es insgesamt zwar 4 verschiedene Texteditoren im Angebot, allerdings ist zum Beispiel der sehr gute PFE nur in der 32-Bit Version enthalten. Verglichen mit dem Platz, der noch auf der CD frei ist (immerhin runde 600 MB), hätten die Produzenten durchaus noch das eine oder andere Delphi-Juwel mit auf das CDROM nehmen können.

Der Wert des CDROMs wird auch insofern relativiert, als man sich für dessen stolzen Preis von 233,- öS bei entsprechender Hardware-Ausstattung ca. 60 MB aus dem Internet holen kann - also um circa 10 MB mehr als auf der CD vorhanden ist. □

Wie die verschiedenen Programmierer ihre Fahrräder bauen

ADA-Programmierer bauen ein viereckiges Rad und passen alle Straßen an.

ALGOL-Programmierer weigern sich, Räder zu bauen, weil es sie irgendwohin bringen könnte.

APL-Programmierer schweben in höheren Sphären, sie brauchen keine Räder.

ASSEMBLER-Programmierer bauen tausende von Rädern, keine 2 passen auf eine Achse.

BASIC-Programmierer bauen nur ein Rad, aber finden keine Achse dazu.

COBOL-Programmierer bauen TAUSEND-RÄDRIGE-TRANSPORT-MODULE und verbieten das Gehen.

FUNKTIONAL-Programmierer rufen eine Funktion HOLZ auf und hoffen, ein Rad zu bekommen.

FORTH-Programmierer bauen Räder und vergessen, wo sie sie gestapelt haben.

FORTRAN-Programmierer werden wahnsinnig bei der Suche nach Rädern, die mit "I" beginnen.

LOGO-Programmierer bauen kleine rote Autos.

LISP-Programmierer (bauen Räder mit (Rädern mit/Rädern mit (Rädern mit (dem was LISP-Programmierer bauen)))).

MICRO-Programmierer wissen nicht, daß Räder existieren.

PASCAL-Programmierer erklären das Gehen zur Tugend.

PL/I-Programmierer setzen ein Team ein, um eine Räderfabrik zu entwerfen, die Räder der falschen Größe bauen wird.

RPG-Programmierer haben ein Rad, schade, daß es viereckig ist.

SYSTEM-Analytiker sind viel zu beschäftigt, Räder zu suchen, um eins zu bauen.

TEXTVERARBEITUNGS-Benutzer bauen Räder.

TOS-Benutzer bleiben wo sie sind, sie sind das Warten gewohnt.

UNIX hat unter irgendeiner Schale sicher irgendwo ein Verzeichnis von Rädern.

Laut und Leise am PC-Lautsprecher

Die Ausgabe akustischer Signale am PC ohne Soundkarte beschränkt sich auf mehr oder weniger lange Rechtecksignale konstanter Lautstärke. Nachfolgend beschreibe ich ein Verfahren, bei dem die Töne in ihrer Lautstärke und Kurvenform variiert werden können.

David Solonyna PCND-DSK-520:\toene

Prinzip der Tonerzeugung am PC

Eine Tonquelle sendet Schwingungen aus, die sich durch ein Medium (meistens Luft, aber auch Wasser) als Welle fortsetzen. Im PC ist die Tonquelle ein Lautsprecher, dessen Membran als Schwingungserzeuger dient. Der Lautsprecher ist mit eine Kanal des eingebauten Timerbausteins verbunden. Diese Verbindung kann ein- oder ausgeschaltet werden. Die voreingestellte Tonhöhe ist 1 kHz. Da der Timer Rechteckschwingungen abgibt, ist auch die Tonform eine Rechteckschwingung.

Klänge am PC

Windows mit Sound-Unterstützung bietet dem Benutzer die Möglichkeit, vielfältige Klangeffekte über Soundkarten auszugeben. Die Sounds sind in WAV-Dateien gespeichert. WAV-Dateien enthalten den Sound in gesampelter (=abgetasteter Form).

Abtasten (Samplen)

Die Sample-Daten bezeichnen jeweils einen Punkt auf der Schwingungskurve. Je mehr Punkte pro Zeiteinheit zur Verfügung stehen, desto genauer läßt sich die Kurve zeichnen beziehungsweise ausgeben.

Beim Abtasten wird das analoge Tonsignal von einem AD-Wandler umgesetzt. Der Wandler tastet das Signal mit einer bestimmten Frequenz ab und wandelt die Ergebnisse in Zahlen um. Normalerweise reichen 8 Bit breite Zahlen aus.

Shannon-Theorem

Das Shannon-Theorem besagt, daß die Abtastfrequenz mindestens doppelt so groß sein muß, wie die höchste Tonfrequenz, die man aufnehmen will.

PIT

Um Klänge mit verschiedenen Lautstärken auszugeben, wird der *programmierbare Intervall-Timer* (PIT = 8253/8254) verwendet.

Der PIT weist drei voneinander unabhängige und getrennt programmierbare Zähler mit den Nummern 0 bis 2 auf, die jeweils 16 Bit breit sind. Jedem der drei Zähler wird ein eigenes Takt-signal zugeführt, das als Zeitbasis für den jeweiligen Zähler dient. Im XT sorgte ein 14.317.180Hz-Quarz für das Basistaktsignal, das durch drei geteilt wird, um das Taktsignal des Prozessors von 4,77 MHz zu erzeugen. Durch eine weitere Vierteilung entsteht ein Signal mit etwa 1,193180 MHz, das den drei Eingängen des 8253/54 zugeführt wird.

Dieser Baustein kümmert sich um verschiedene Aufgaben, so ist er für die Echtzeituhr, die Tonerzeugung mit dem Lautsprecher und für den RAM-Refresh notwendig. Der 8253/54 stellt dafür drei Kanäle zur Verfügung. Der erste wird benutzt, um die Systemuhr zu steuern, der zweite ist für den DMA-Controller zuständig (RAM-Refresh), der dritte steuert den Lautsprecher.

Der PIT zählt intern Zähler herab, bis diese den Wert 0 aufweisen. Sodann liefert der Baustein das Ausgangssignal und setzt die Zähler auf ihren Anfangswert. Das Vermindern der Zählerstände startet daraufhin von neuem.

Angesprochen wird der PIT über die Port-Adressen 40h bis 43h. Die Adressen 40h bis 42h dienen dabei für die Initialisierungswerte, während 43h für die Steuerung zuständig ist.

Um einen Zähler zu aktivieren oder zu triggern, wird ihm ein Gattersignal GATEx zugeführt. Je nach Zählmodus wird der zugehörige Zähler durch einen Übergang niedrig-hoch oder einen hohen Pegel des GATEx-Signals aktiviert. Die Form des Ausgangssignals hängt vom Modus des jeweiligen Zählers ab.

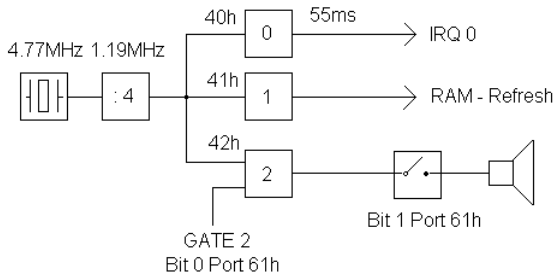
Beschreibung des Steuerwortbytes

Bit	Bedeutung	
0	Zählerart	0 = Binärzähler, 1 = BCD-Zähler
1-3	Modusnummer	000 = Interrupt bei Ende der Zählperiode
		001 = programmierbares Monoflop
		010 = Taktgenerator
		011 = Rechtecksignalgenerator
		100 = Triggerung des Ausgangs durch Software
		101 = Triggerung des Ausgangs durch Hardware
4-5	Leseformat	00 = Zählstand speichern (8254)
		01 = nur Lo-Byte schreiben
		10 = nur Hi-Byte schreiben
		11 = zuerst Lo-Byte, dann Hi-Byte schreiben
6-7	Zeitgeber-Nummer	00 = Timer 0 (Systemuhr)
		01 = Timer 1 (RAM-Refresh)
		10 = Timer 2 (Lautsprecher)

Kanal 0 ist normalerweise im Modus 2 programmiert, mit einer Frequenz von 18.2Hz. Sobald bis auf 0 hinuntergezählt ist, wird am Ausgang (Eingang des Interrupt - Controllers) ein Nadelimpuls ausgelöst.

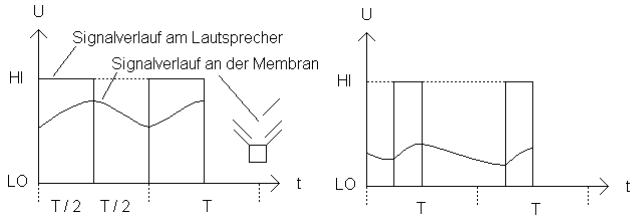
Die erforderliche Abtastfrequenz kann mit dem PIT erzeugt werden, indem man die Frequenz des Interrupt 8 neu einstellt. Der Startwert für den Kanal 0 errechnet sich nach der Formel „1193180Hz/Frequenz“. Bis ca. 20kHz läßt sich der Modus 2 (Taktgenerator) verwenden. Für höhere Frequenzen muß Modus 3 (Rechtecksignal) verwendet werden, da sonst der Computer abstürzt.

Um den Lautsprecher zu aktivieren müssen die Bits 0 (GATE2) und 1 (Ein/Aus-Schalter) des "Port B" 61h gesetzt werden.



Einen Piep-Ton mit einer bestimmten Frequenz erzeugt man, indem man den Kanal 2 (Lautsprecher) des PIT im Modus 3 (symmetrisches Rechtecksignal) programmiert. Ist der Ausgang des PIT auf "HI" bzw. "LO", so bewegt sich die Membran nach vor, bzw. zurück. Bei genügend schnellen Wechsel entsteht ein Ton. Die Amplitude läßt sich jedoch nicht beeinflussen, deshalb ist die Lautstärke nahezu bei jeder Frequenz konstant. Würde man aber ein Rechtecksignal erzeugen, bei dem das Signal z.B. nur sehr kurz auf HI ansteigt und gleich darauf wieder abfällt, wird die Membran wegen der Massenträgheit diesem Verlauf nicht mehr folgen können. Dasselbe gilt, wenn das Signal nur kurz abfällt und gleich darauf wieder ansteigt (Nadelimpuls). Die Dauer dieser Änderung kann man variieren. Gibt man ein solches Signal sehr oft aus, pendelt sich die Membran auf eine bestimmte Position ein. Einen derartigen Signalverlauf kann man mit dem Modus 0 (Interrupt bei Ende der Zählperiode) des PIT erzeugen. Mit den verschiedenen

Startwerten kann wie beim PCM-Format eine Kurve beschrieben werden. Um einen Ton mit variabler Lautstärke zu erzeugen, speichert man eine Periode z.B. eines Rechtecksignals mit z.B. 20 Startwerten. Die Frequenz des Timers ändert man auf 8800Hz. Bei jedem Aufruf des Interrupt 8 gibt man einen Wert der Kurve am Kanal 2 des PITs aus. Bei einer Abtastfrequenz, die kleiner als die obere Hörschwelle (ca. 19kHz) ist, hört man ein Pfeifen im Hintergrund. In diesem Fall muß die Kurve durch mehr Punkte beschrieben und die Abtastfrequenz erhöht werden.



Das verbreitetste Verfahren zur Speicherung digitalisierter Töne, Geräusche, Klänge ist das PCM-Format (Puls Code Modulation). Dabei werden den einzelnen Abtastwerten des analogen Signals bestimmte Zahlen zugeordnet. Normalerweise werden 8 Bit breite Daten verwendet. Würde man ein PCM-kodiertes Signal nach dem oben beschriebenen Verfahren ausgeben, wird man mit an Sicherheit grenzender Wahrscheinlichkeit gar nichts hören. Begründung: Auf einer Soundkarte erzeugt ein DA-Wandler aus den Werten ein analoges Signal. Bei Ausgabe über den PC-Lautsprecher geben die einzelnen Zahlen die Dauer zwischen der ersten und zweiten Flanke des Rechtecksignals an. Da der PIT jedoch auf allen Kanälen mit derselben Frequenz von 1193180 Hz dekrementiert, würde bei einem Zählwert von z.B. FFh das Ausgangssignal des Kanal 2 nicht mehr ansteigen, da der nächste Interrupt bereits ausgelöst wird, bevor noch das Signal wieder abgefallen ist. Deshalb benötigt man eine Übersetzungstabelle. Kommt jeder Wert zwischen 0 und FFh gleich oft vor, hört man nur einen leisen Ton. Durch das konvertieren werden die Abstände zwischen den einzelnen Punkten der Kurve verringert.

Vorschlag für ein Unterprogramm, das eine Tabelle erzeugt, bei der die Lautstärke kaum verringert wird:

```
void GenerateTab(char *tab,WORD freq)
{
  float Step=float((1193180/freq & 0xf0)-0x1)/64;

  for (BYTE i=0;i<=128;i++)
    tab[i]=0x1;
  for (i=128;i<=254;i++)
    tab[i]=1193180/freq & 0xf0;
  for (i=0;i<=64;i++)
    tab[i+96]=i*Step+1;
}
```

Während des Abspielens empfiehlt es sich alle IRQs außer dem Timer abzuschalten. Dies geschieht am besten durch die Ausgabe eines OCW1 (Operation Command Word #1) auf die Ports 21h und A1h.

Aufbau des OCW1 (1 Byte)

```
Bit:   7  6  5  4  3  2  1  0
   M7 M6 M5 M4 M3 M2 M1 M0
```

M7 .. M0 0 ... zugehörige IRQ - Leitung ist nicht maskiert;

1 ... zugehörige Leitung ist maskiert (gesperrt)

Bsp.:

```
   outportb(0x21,0xfe);
   outportb(0xa1,0xff);
   ...
   outportb(0x21,0);
   outportb(0xa1,0);
```

Das Sperren aller IRQs verhindert, daß durch Tastatur, Maus, etc. der Abspielvorgang unterbrochen wird, was zu einem Absturz führen kann. Vor dem Beenden des Programmes müssen alle IRQs wieder freigegeben werden, da sonst kein weiters Arbeiten möglich ist.

Das WAVE - Format

Microsoft hat für die meisten unter Windows benützten Dateitypen eine Grundstruktur erstellt, die das Arbeiten mit den verschiedenen Formaten erleichtert. Diese Struktur nennt sich RIFF (Ressource Interchange Format), und stellt eine Weiterentwicklung des IFF - Standards dar. RIFF - Dateien werden dabei in Chunks aufgeteilt.

RIFF - Header
Format - Chunk
Data - Chunks
...

Jeder Chunk wird durch eine 4-Byte-Kennung (FOURCC, Four Character Code) eingeleitet. Anschließend folgt ein 4-Byte-Feld mit der Länge des Chunks. Danach folgt eine variable Anzahl von Parametern. Da die Daten immer wortweise abrufbar sein müssen, wird bei einer ungeraden Anzahl von Bytes am Ende ein Pad-Byte eingefügt. Dieses Füllbyte enthält den Wert 0 und wird in der Längenangabe nicht berücksichtigt. Die Informationen werden in RIFF-Dateien im Intel-Format (little endian) gespeichert.

Der WAV - Header

Der Header beginnt mit einem RIFF-Chunk, einem Container. Nur Containerchunks können innerhalb des Daten-Teils weitere Chunks enthalten.

Offset	Bytes	Bemerkungen
00h	4	Chunk - Name 'RIFF'
04h	4	Chunk - Länge
08h	4	RIFF-Typ 'WAVE'

Die ersten 4 Byte enthalten die Buchstaben 'RIFF'. Nach der Längenangabe wird der Header mit einem 4-Byte-Feld abgeschlossen, welches den Typ der Datei enthält. Es gibt zur Zeit fünf verschiedene RIFF-Formate:

PAL	Palettendatei
RDIB	geräteunabhängige Bitmap
RMID	MIDI-Datei
RMMP	Multimedia-Movie-Datei
WAVE	Sound-Datei

Der FMT - Chunk

Direkt nach dem Header folgt der Format-Chunk.

Aufbau:

Offset	Bytes	Bemerkungen
00h	4	Chunk-Name 'FMT '
04h	4	Chunk-Länge (10h)
08h	2	Format Typ: 0: Mono 1: Stereo
0Ah	2	Kanalzahl
0Ch	4	Abtastfrequenz in Hertz
10h	4	Byte pro Sekunde
14h	2	Bytes pro Sample: 1: 8 Bit Mono; 2: 8 Bit Stereo; oder 16 Bit Mono; 4: 16 Bit Stereo
16h	2	Bits pro Sample

Auf den FMT-Chunk folgen die eigentlichen Sounddaten in Form eines DATA-Chunks.

Aufbau:

Offset	Bytes	Bemerkungen
00h	4	Chunk-Name 'data'
04h	4	Chunk-Länge
08h	n	Datenbereich

Die Interpretation der Daten erfolgt über den vorherigen FMT-Chunk. Für jedes Sample sind n Bits zu lesen und auszugeben. Bei Stereo-Daten sind die Werte für den linken und rechten Kanal wechselweise gespeichert. Normalerweise befindet sich nur ein DATA-Chunk in einer WAVE-Datei. Es sind jedoch auch mehrere DATA-Chunks möglich. ☐

Was sind Embedded Systems?

Der Versuch eines FAQ

Peter Balog

Was sind Embedded Systems ?

Embedded Systems (eingebettete Systeme), sind applikationsspezifische Mikrocomputerschaltungen mit applikationsspezifischer Software innerhalb einer zu steuernden (zu beeinflussenden) Umgebung.

Wo findet man *Embedded Systems* ?

In jeder „intelligenten“ Anlage,- vom modernen Telefonapparat, Faxgerät, Videorekorder, Laserprinter über die Systeme zum Motormanagement in modernen Autos bis hin zu Hochleistungsanlagen im Bereich der Telekommunikation und Automatisierungstechnik. Sowohl die Geräte der modernen Meß-technik, als auch die Apparaturen im Medizinbereich wären ohne den Einsatz von leistungsfähigen *Embedded Systems* nicht denkbar.

Welche Hardware wird verwendet ?

Die Komplexität der Hardware von *Embedded Systems* richtet sich nach der spezifischen Anwendung. Von der einfachen Mikrocontrollerlösung (Singlechip) über Singleboardcomputer, bestückt mit Standardprozessoren, Standardperipheriebausteinen und spezifischen ASICs bis hin zu Multiboardsystemen basierend auf lose und eng gekoppelten Multimikroprozessormoduln, deren Rechnerleistung oft über denen eines Workstation-Clusters oder Mainframes liegt.

Nach welchen Gesichtspunkten erfolgt der Hardwareentwurf ?

Der Hardware-Designer von *Embedded Systems* hat die Aufgabe aus den definierten zeitlichen, „räumlichen“ und sonstigen (firmenspezifischen) Randbedingungen die geeignete Mikrocomputerplattform zu finden. Das Zusammenspiel zwischen Prozessor(en), Standardperipherie, Speicher und anwendungsspezifischen (meist hochintegrierten *Standard-Cell* oder *Full-Custom*)-ASICs, die sogenannte *Glue-Logic*, erfolgt nach den Richtlinien für einen konstruktiven, streng synchronen Digitalentwurfs und wird (meist) in FPL (*Field Programmable Logic*) implementiert. Formale Spezifikation, Verifikation und Simulation unterstützen den Designer. Die „*Design for Testability*“-Strategien, gepaart mit der modernen Meßtechnik nehmen der Inbetriebnahme des Komplettsystems den Schrecken. Neben all der Digitaltechnik dürfen die analogen Aspekte, wie EMV, Stromversorgung und *Ground-Bounce*-Probleme, um nur einige zu nennen, nicht außer Acht gelassen werden.

Welche Software wird verwendet ?

Zufolge der Applikationsabhängigkeit steht i.A. keine „Standardsoftware“ zur Verfügung,- sehr wohl gibt es jedoch Standardansätze im Softwaredesign.

Wie erfolgt die Softwareentwicklung für *Embedded Systems* ?

Die Softwareentwicklung läßt sich in die beiden Bereiche Systemsoftware und Applikationssoftware gliedern. Fast die gesamte Zeit der Softwareentwicklung wird zur Erstellung der Applikationssoftware benötigt,- diese wird plattform-unabhängig in einer höheren Programmiersprache (C, C++), basierend auf einem auf einem *Application Programming Interface* mit *Realtime*-Funktionalität (*preemptive multitasking with asynchronous IO-capability*), welches meist in Form einer Hochsprachenbibliothek vorliegt, entwickelt.

Lediglich 5-10% der Softwareentwicklungszeit entfallen auf die Systemprogrammierung, deren Aufgabe es ist, einen standard *Real-Time-Kernel* an die Hardwareplattform anzupassen und ein entsprechendes Laufzeit- und IO-System zu implementieren. Obwohl hier sehr hardwarenahe programmiert werden muß (*Device-Driver, Interrupt-Handler*), wird doch ein Großteil in einer höheren Programmiersprache (C) erledigt.

Was ist ein Real-Time-Kernel ?

Ein RTK ist der Kern eines Multitaskingbetriebssystems speziell für eingebettete Systeme (*preemptive multitasking with asynchronous IO-capability*). Der RTK stellt, plattformunabhängig, folgende Funktionen zur Verfügung:

- Task-Management
- Critical Ressource Management
- Task-Synchronization (Semaphores, Signals)
- Intertask-Communication (Mailboxes, Message-Passing, Pipes)
- IO-Management (IO-Driverinterface)
- Interrupt-Management (Interrupt-Handler-Interface, Interrupt-Tasks)
- Memory-Management (static, dynamic)
- Error-Handling (optional)
- Monitoring, Debugging (optional)

Für fast alle Mikroprozessoren und Mikrocontroller gibt es RTKs, die in kürzester Zeit an die gegebene Hardware angepaßt werden können. Zudem bieten RTK-Hersteller für gängige Peripheriebausteine *IO-Driver* und *Interrupt-Handler* (meist in C-Source) an,- der Systemprogrammierer muß meist nur die aktuelle Hardwareadresse des Chips in eine C-Header-Datei eintragen.

Wann sollte ein RTK verwendet werden ?

Immer dann, wenn im System mehr als eine Hardwarekomponente einen Interrupt generieren kann, oder wenn im System mehr als ein Eingang und ein Ausgang angesteuert werden muß.

Warum soll die Applikationssoftware nicht „direkt auf der Hardware“ implementiert werden ?

Für Kleinstsysteme (1 Eingang, 1 Ausgang, kein Interrupt) ist die Anwendung eines RTKs sicher ein „Overkill“. In allen anderen Systemen würde die Software-entwicklung mindestens doppelt so lange dauern, wobei die Verifikation des Systems ungleich schwieriger ist, wenn keine „ordnende Instanz“ (RTK) verwendet wird.

Wird diese „ordnende Instanz“, welche das „Interrupt-Monster“ zähmt, innerhalb der Applikation implementiert, so hat man seinen eigenen RTK entwickelt ! Und es gibt eigentlich keinen Grund einen RTK selbst zu entwickeln, oder doch ?

Gibt es Gründe, die für die Eigenentwicklung eines RTKs sprechen ?

Die beste Antwort auf diese Frage gibt eine Overheadfolie der Probitas Corporation, die im Rahmen des „1995 Embedded System Design Symposiums“ von HP gezeigt wurde:

Reasons to Write a Real-Time-Kernel

- You enjoy re-inventing the wheel
- You like chasing difficult bugs
- There’s no rush to get your product out
- Porting software to new platforms is your idea of really living it up
- Your firm has the NIH[1] syndrome
- You want to be in the business of sellings RTKs yourself

[1] Not Invented Here

© Probitas Corporation, 1994

Warum gewinnt heute objekt-orientierter Softwareentwurf zunehmend an Bedeutung ?

Der objektorientierte Ansatz (Kapselung, Hierarchien, Vererbung) zwingt zu einer noch abstrakteren Spezifikation des Problems. Dadurch wird die „ordnende Instanz“ im System mächtiger.

Wie sieht die Zeitaufteilung für den Entwurf des Gesamtsystems aus ?

Unter der Annahme, daß die Hardware auf Standard-Prozessoren und Peripherie-einheiten basiert und daß die Entwicklung etwaig verwendeter, hochintegrierter *Standard-Cell*- bzw. *Fullcustom*-ASICs (speziell im Hi-Speed-Telekommunikationsbereich) in die Hardwareentwicklungszeit nicht miteingerechnet wird, und daß außerdem die notwendige *Glue-Logic* zeitgemäß entworfen (synchron, HDL, Verifikation, Simulation) und implemetiert (FPL) wurde, teilt sich die Gesamtzeit für die Entwicklung von Embedded Systems (typisch) wie folgt auf: 25% Hardware, 5% Systemsoftware, 70% Applikationssoftware.□

electronic only

Peter Ullrich

Mit der bereits zweiten Ausgabe der CDROM „electronic only“ der deutschen Firma „RUESS electronic“ werden dem Elektronik-Interessierten 230 Programme geliefert, die ihm bei der Entwicklung oder Simulation elektronischer Schaltungen, Erstellung von Layouts, Programmierung von speicherprogrammierbaren Steuerungen und Mikrocontrollern oder bei Amateurfunkaufgaben helfen. Bei den Programmen handelt es sich einerseits um Voll-, Freeware- bzw. Shareware-versionen, andererseits um interaktive Demos oder Slideshows.

Inhalt

230 Programme unterteilt in folgende 13 Kategorien:

- Electronic CAD
- Schaltungssimulation
- Elektronische Datenbücher
- Digitaltechnik
- Regelungstechnik
- SPS
- Mikrocontroller
  Allgemein
  8051er Familie
  68HC11
  PIC-Controller
  8048er Familie
- Programmierbare Logik
- Meßwertverarbeitung
- CAM
- Audiotechnik
- Amateurfunk
- Dies & Das

Hinter der Rubrik **Electronic CAD** verbergen sich viele Demopakete (Slideshows und interaktive Demos, Vollversionen ohne Speichermöglichkeit) von einigen bekannten CAD-Paketen. Unter anderem findet man aber auch die uneingeschränkte Vollversion des Layoutprogrammes EASYTRAX mit dem man ohne spezielle Schulungen recht flott zu einem Leiterplattenlayout kommt.

Bei den Kategorien **Schaltungssimulation** und **Digitaltechnik** sind Demos und Programme (teils von bekannten Herstellern) für Analog- und Digitalsimulatoren vertreten.

Einige äußerst nützliche **elektronische Datenbücher** helfen beim Analog- und Digital-Schaltungsdesign.

Regelschaltungen kann man mit den diversen Programmen der **Regelungstechnik** generieren und simulieren.

Für Interessierte an speicherprogrammierbaren Steuerungen ist die **SPS**-Rubrik gerade richtig: viele SPS-Programmierumgebungen und Simulatoren.

Das größte Angebot an Programmen ohne Einschränkungen (Vollversionen, Freeware, Shareware mit Einschaltmeldung) befindet sich unter der Rubrik **Mikrocontroller**. Hier werden zahlreiche Compiler, Assembler, Disassembler, Simulatoren zu den verschiedensten Mikrocontrollern angeboten.

Für die rund um Mikrocontroller eingesetzten programmierbaren Logik-Bausteine (GALs, EPLDs) findet man viele bekannte und zum Teil sehr gute Entwicklungsumgebungen mit Logikcompiler in der Kategorie **Programmierbare Logik**.

Zum Thema **Meßwertverarbeitung** werden viele Demos von Softwarepaketen der großen Softwareanbieter angeboten.

Zum Umwandeln von CAD in CNC-Daten befinden sich einige Programme im Verzeichnis **CAM**.

Für Lautsprecherboxen-Designer werden in der Rubrik **Audiotechnik** viele Berechnungs- und Simulationsprogramme, aber auch ein Meßprogramm zum Boxen messen, angeboten.

Bei den **Amateurfunk**-Programmen erstreckt sich das Angebot über die üblichen Packet-Radio- und über Satelliten-Tracking-Programme bis hin zu Antennen-Designprogrammen.

Schließlich findet man in der Kategorie **Dies & Das** verschiedene Programme, die sich nicht in eine der oben angeführten Kategorien einordnen ließen.

Bedienung

Die zahlreichen Programme sind über ein komfortables Windows-Menüprogramm zu installieren beziehungsweise auf Diskette zu kopieren. Die Programme sind thematisch in dreizehn Kategorien untergliedert und sind mittels Mausklick anzuwählen. Nach der Anwahl eines Programmes erfährt man in einem Textfenster die ungefähre Aufgabe des Programmes und die Art der Software (Vollversion, Freeware, Public Domain, Shareware, Demo, Slideshow). Bei den meisten Programmen kann man sich auch gleich die zum jeweiligen Programmpaket gehörenden Textdateien (Beschreibungen, Registrierungshinweise, Bestellformular) vor dem Installieren ansehen (Schaltfläche : DETAILS). Dies bietet einen guten Überblick über die Funktion des Programmes, und informiert auch gleich über eventuelle Einschränkungen der einzelnen Programme.

Angenehm ist die Tatsache, daß jedes Programm in einem eigenen Unterverzeichnis mit weiteren zwei Unterverzeichnissen gespeichert ist. Aus dem INST-Verzeichnis wird das Programm auf die Festplatte installiert, während im jeweiligen DISK-Verzeichnis die Disketten-Images abgelegt sind. Muß ein Programm nicht speziell installiert werden, kann man es gleich vom INST-Verzeichnis aus starten. Leider wird diese Funktion vom Menüprogramm nicht angeboten.

Die Installation beziehungsweise die Erstellung eines Diskettensatzes geht normalerweise problemlos vor sich. Sollte es bei einem Programm (z.B. unter Windows 95) Probleme geben, wird man darauf durch ein Dialogfenster aufmerksam gemacht und kann diese Meldung beziehungsweise auch Installationstips auch gleich ausdrucken.

Herausgeber: Ruess electronic, Kirchstraße 19, D-89291 Holzheim

Inhalt: Vollversionen, Public Domain, Shareware, Demos und Informationen speziell für den Elektronik-Bereich.

Voraussetzungen: Das Menüprogramm benötigt Windows. Bei den diversen Programmen gelten unterschiedliche Voraussetzungen.

Installation: Zur Installation des Menü-Programms rufen Sie unter Windows das Programm \SETUP auf. Es erstellt eine Programm-Gruppe mit dem Menü-Icon und kopiert auf Wunsch das Menüprogramm auf die Festplatte.

Menü-Programm: Die Programme sind zwecks Übersichtlichkeit in 13 Kategorien eingeteilt. Jede Kategorie enthält mehrere Programme, die durch Mausklick auszuwählen sind. Sie können nun Detail-Informationen über das aktuelle Paket abrufen, es auf Ihrer Festplatte installieren oder davon einen Disketten-Satz erstellen, um es z.B. auf einem anderen Computer zu installieren.

Bezugsquellen: Diese CD erhalten Sie bei allen 20 Radiobastler-Filialen in Österreich. Der Preis war bei Redaktionsschluß noch nicht bekannt. In Deutschland ist die CDROM bei Conrad-Elektronik und Reichelt-Elektronik um 39.90 Mark erhältlich.

Electronic CAD	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
Geddy CAD & Turbo Router		X		
Turbo Router & Flashlight		X		
Platin			X	
ELEC		X		
Target V3			X	
Ariadne		X		
Windraft		X		
ULTIboard			X	
PADS TestDrive for Windows		X		
PADS TestDrive for DOS		X		
CADdy EDS			X	
VeriBest Design Capture (ACE Plus)			X	
VeriBest PCB Interactive Demo			X	
VeriBest Synthesis Guided Demo				X
Veribest Self-running Demo				X
PROTEL Advanced Schematic			X	
PROTEL Advanced PCB			X	
PROTEL Schematic & Autotrax (DOS-Pack)			X	
PROTEL EasyTrax	X			
MaxEDA				X
MaxSYS				X
Easyplan		X		
Elektro CAD		X		
Specctra				X
Autosketch			X	
AutoCAD LT				X
Timing Designer			X	
Timing Diagrams		X		
Time Crafter		X		

Schaltungssimulation	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
DesignWorks			X	
Simplorer			X	
Microcap V			X	
Proseries				X
Design Center (PSPICE)			X	X
Design Center AMD			X	
Electina			X	

Elektronische Datenbücher	Vollversion Freeware PD	Share-ware	Demo Interak-tiv	Demo / Show
WinTIDIG	X			
TI-Digital	X			
TI-Linear	X			
Transistor Datenblatt		X		
BB Curcuits Data	X			
TRD		X		
SCRL		X		

Digitaltechnik	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
WinLog			X	
LoKon		X		
Digital Logic Analyser		X		
Logik		X		
Logic	X			
DISJ		X		
LA		X		
LogiSym		X		
4-Kanal-Logik-Analysator		X		

Regelungstechnik	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
WinFact			X	
WinReg-51			X	
SimRegW		X		
SimRegW (neue Beta-Version)		X		
Julia		X		
SEAL-S57901	X			
Regeln		X		
Regler		X		
FuzzGen		X		
Fuzzy Control Manager			X	

Mikrocontroller - Allgemein	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
AS DOS	X			
AS DPM	X			
AS OS/2	X			
Cross Assembler TASM		X		
Cross Assembler XASM		X		
Macro Assembler 1		X		
Macro Assembler 2		X		
HASM	X			
PseudoASM	X			
Cross-Assembler AS	X			
ICE Emulator				X
HTC ANSI-C Compiler			X	
Pacific C		X		
Cross 65		X		
SIM 85		X		
TS 6805		X		
PC MAC		X		
AS8 / EU8		X		
ECO_C			X	
C-Compiler/Debugger 680X0			X	
Compass / Z8			X	
C-Compiler ICC Z80			X	
C-Compiler ICC 7700			X	
C-Compiler ICC H8300			X	
C-Compiler ICC H8500			X	
C-Compiler ICC 6816			X	
C-Compiler ICC 740			X	
C-Compiler ICC 8096			X	
C-Compiler ICC T900			X	
Serieller Master		X		
CSIDE			X	
SEPROG	X			

Mikrocontroller - 8051er Familie	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
NiliBASIC		X		
MPE		X		
PSIM 8051/8052	X			
Compass/51			X	
Compass/251			X	
CESY-Compiler 80X51			X	
IST8051 /Windows			X	
HTC-51			X	
HTC-XA			X	
EMILY 51	X			
NOICE 51		X		
SIMUL51			X	
SIM 51		X		
BASIC 51	X			
PFX51		X		
IAR-Cross-Compiler ITS 8051			X	
WinASS 8051			X	
ASS51		X		
AW 80535		X		
Disassembler 51	X			
TS 8051		X		

Mikrocontroller - 68HC11	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
AS 11	X			
SIM 68HC11			X	
MIMO 11		X		
NOICE 11		X		
Motorola 68HC11	X			
CESY-Compiler 68HC11			X	
EW6811/WINDOWS			X	
IAR-Cross-Compiler ICC 6811 DOS			X	
SBASIC 68HC11	X			

Mikrocontroller - PIC-Controller	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
PIC C-Compiler		X		
IL-Sim16			X	
ASPIC		X		
EUPIC		X		
PicGraph		X		
IDE-ASM for Windows		X		

Mikrocontroller - 8048er Familie	Voll-version / Free-ware / PD	Share-ware	Demo Interak-tiv	Demo / Show
SIM 8048		X		
ASSI 8048		X		
MC 48	X			
TS 48		X		
DIS 48		X		
EPASM		X		

SPS	Voll-version / Free-ware	Share-ware	Demo Interak-tiv	Demo / Show
WinSPS		X		
ACCON-PS			X	
S5 Win			X	
ACCON-PG			X	
DOS-AG		X		
SPS-Sim		X		
SPS1		X		
HL-SPS		X		
AEG 020	X			
S5	X			
SPSSIM		X		
SPS-Pilot		X		
PC-Panal		X		

Programmierbare Logik	Voll-version / Free-ware / PD	Share-ware	Demo Interak-tiv	Demo / Show
First Step	X			
CUPL (Demo)			X	
CUPL (Slide Show)				X
MaxPlus II			X	
Synario			X	
State CAD			X	X
Easy Abel		X		
ALLMAX-MEGAMAX			X	
PLDshell Plus	X			
XEPLD XEPLD 5.0 (Vollversion)	X			
Update von 5.0 auf 5.0.2	X			
GAL Development			X	
GALASM		X		
TANNER Tools			X	

Meßwertverarbeitung	Voll-version / Free-ware / PD	Share-ware	Demo Interak-tiv	Demo / Show
DASY Lab			X	
Test Point			X	
WINLAB			X	
Visual Designer			X	
Labtech Notebook			X	
Genie			X	
VisuaLab			X	
Personal Line			X	
Visual Test Extensions				X
Visual DAS				X
Famos				X
Frame			X	
Search			X	
Unistat			X	
Pmess			X	
View DAC			X	
Testware			X	
WorkBench			X	
DO-IT			X	
Zellabor			X	
ProMi		X		
PicoScope			X	

Audiotechnik	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
Speaker		X		
Hobbybox		X		
BoxPlot		X		
LSP		X		
AudioCAD		X		
Chassis		X		
Boxen		X		
WinATB			X	
Analysis			X	

Amateurfunk	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
Packy for Windows		X		
PacketPeT Lite		X		
PktWin		X		
UltraPak		X		
Mlog / MPkt	X			
WinLink	X			
WinGT / CB			X	
WinLog für Windows	X			
WriteLog		X		
Antenna Systems Analyser		X		
NEX-WIN Basic		X		
Electromagnetic Antenna Modelling			X	
LogSat		X		
WinOrbit	X			
DOVE	X			
Cwave		X		
NuMorse		X		
MRP-1			X	
GMT Clock		X		
Locator Calculator	X			
WinHDG		X		
E-C-Selector		X		

CAM	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
ECAM			X	
CNC / PC			X	
CAD-2-CNC		X		

Dies & Das	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
ECW und PECW		X		
ETG		X		
Elektronik-Manager		X		
ECalc		X		
WidCode	X			
Widerstand für Windows		X		
Resist	X			
CCCR		X		
Win-Elektronik		X		
ECap		X		
Omega		X		
F-CAP		X		
Filter		X		
Elektrische Felder		X		
Fouriersynthese		X		
Schaltzentrale für Windows		X		
LWL		X		

□

Wenn Betriebssysteme eine Fluggesellschaft betrieben

DOS Airlines: Alle schieben das Flugzeug an bis es abhebt. Dann springen alls auf und lassen das Flugzeug trudeln, bis es wieder auf den Boden schlägt. Dann schieben wieder an, springen auf ...

DOS mit QEMM Airlines: Genau dasselbe, nur mit mehr Platz für die Füße beim Anschieben.

Mac Airlines: Alle Stewards, Stewardessen, Piloten, Gepäckträger und Ticketverkäufer sehen gleich aus, bewegen sich gleich und sagen das Gleiche. Wenn man nach Details fragt, bekommt man immer die gleich Antwort: Das müsse man nicht wissen, wolle es auch nicht wissen, und alles laufe schon richtig. Man solle also lieber gleich still sein.

Windows Airlines: Das Flughafenterminal ist schön bunt, die Stewards und Stewardessen freundlich. Man gelangt ohne Probleme an Bord, ein reibungsloser Start ... Plötzlich stürzt das Flugzeug ohne jegliche Vorwarnung ab.

OS/2 Airlines: Um an Bord des Flugzeugs zu kommen, muß man sein Ticket zehnmal stempeln lassen und in zehn unterschiedlichen Schlangen anstehen. Dann füllt man ein Formular aus, in dem man angeben muß, wo man sitzen möchte und ob der Sitzplatz wie in einem Bus aussehen soll. Wenn es einem gelingt, an Bord zu kommen und wenn das Flugzeug tatsächlich vom Boden abhebt. hat man einen wunderbaren Flug ... - außer, wenn die Höhen- und Seitenruder einfrieren. In diesem Fall hat man jedoch immer noch genügend Zeit, sich auf den Absturz vorzubereiten.

UNIX Airlines: Jeder Passagier bringt ein Stück des Flugzeugs zum Flughafen mit. Alle gehen auf die Startbahn und setzen das Flugzeug Stück für Stück zusammen. Dabei diskutieren sie fortwährend, welche Art von Flugzeug sie gerade zusammenbauen.

NT Airlines: Alle gehen auf die Startbahn, sagen im Chor das Paßwort und bilden die Umrisse eines Flugzeugs. Dann setzen sich alle auf den Boden und geben Geräusche von sich, als würden sie wirklich fliegen.

Gesammelt von Alexander Hirt, zugeschrieben Paul Giangarra, Chefentwickler von OS/2.

Audiotechnik	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
Speaker		X		
Hobbybox		X		
BoxPlot		X		
LSP		X		
AudioCAD		X		
Chassis		X		
Boxen		X		
WinATB			X	
Analysis			X	

Amateurfunk	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
Packy for Windows		X		
PacketPeT Lite		X		
PktWin		X		
UltraPak		X		
Mlog / MPkt	X			
WinLink	X			
WinGT / CB			X	
WinLog für Windows	X			
WriteLog		X		
Antenna Systems Analyser		X		
NEX-WIN Basic		X		
Electromagnetic Antenna Modelling			X	
LogSat		X		
WinOrbit	X			
DOVE	X			
Cwave		X		
NuMorse		X		
MRP-1			X	
GMT Clock		X		
Locator Calculator	X			
WinHDG		X		
E-C-Selector		X		

CAM	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
ECAM			X	
CNC / PC			X	
CAD-2-CNC		X		

Dies & Das	Vollversion / Freeware / PD	Shareware	Demo Interaktiv	Demo / Show
ECW und PECW		X		
ETG		X		
Elektronik-Manager		X		
ECalc		X		
WidCode	X			
Widerstand für Windows		X		
Resist	X			
CCCR		X		
Win-Elektronik		X		
ECap		X		
Omega		X		
F-CAP		X		
Filter		X		
Elektrische Felder		X		
Fouriersynthese		X		
Schaltzentrale für Windows		X		
LWL		X		

□

Wenn Betriebssysteme eine Fluggesellschaft betrieben

DOS Airlines: Alle schieben das Flugzeug an bis es abhebt. Dann springen alls auf und lassen das Flugzeug trudeln, bis es wieder auf den Boden schlägt. Dann schieben wieder an, springen auf ...

DOS mit QEMM Airlines: Genau dasselbe, nur mit mehr Platz für die Füße beim Anschieben.

Mac Airlines: Alle Stewards, Stewardessen, Piloten, Gepäckträger und Ticketverkäufer sehen gleich aus, bewegen sich gleich und sagen das Gleiche. Wenn man nach Details fragt, bekommt man immer die gleich Antwort: Das müsse man nicht wissen, wolle es auch nicht wissen, und alles laufe schon richtig. Man solle also lieber gleich still sein.

Windows Airlines: Das Flughafenterminal ist schön bunt, die Stewards und Stewardessen freundlich. Man gelangt ohne Probleme an Bord, ein reibungsloser Start ... Plötzlich stürzt das Flugzeug ohne jegliche Vorwarnung ab.

OS/2 Airlines: Um an Bord des Flugzeugs zu kommen, muß man sein Ticket zehnmal stempeln lassen und in zehn unterschiedlichen Schlangen anstehen. Dann füllt man ein Formular aus, in dem man angeben muß, wo man sitzen möchte und ob der Sitzplatz wie in einem Bus aussehen soll. Wenn es einem gelingt, an Bord zu kommen und wenn das Flugzeug tatsächlich vom Boden abhebt. hat man einen wunderbaren Flug ... - außer, wenn die Höhen- und Seitenruder einfrieren. In diesem Fall hat man jedoch immer noch genügend Zeit, sich auf den Absturz vorzubereiten.

UNIX Airlines: Jeder Passagier bringt ein Stück des Flugzeugs zum Flughafen mit. Alle gehen auf die Startbahn und setzen das Flugzeug Stück für Stück zusammen. Dabei diskutieren sie fortwährend, welche Art von Flugzeug sie gerade zusammenbauen.

NT Airlines: Alle gehen auf die Startbahn, sagen im Chor das Paßwort und bilden die Umrisse eines Flugzeugs. Dann setzen sich alle auf den Boden und geben Geräusche von sich, als würden sie wirklich fliegen.

Gesammelt von Alexander Hirt, zugeschrieben Paul Giangarra, Chefentwickler von OS/2.

Mikroprozessortechnik mit LOCAD

LOCAD ist ein für den Unterricht in Digitalelektronik entwickeltes CAD-Programm, das auf fast spielerische Weise in die grundlegenden Prinzipien und Arbeitsweisen einer Datenverarbeitungsanlage auf Hardware-Ebene einführen soll. Es ermöglicht auch dem Nicht-Techniker, ohne Lötkolben und ohne damit nicht selten verbundene Enttäuschungen und Kosten logische Schaltungen einfach aufzubauen und zu testen.

Werner Holler

LOCAD bietet zusammen mit den Handbüchern die Möglichkeit, die Funktionsweise wesentlicher elektronischer Baugruppen von Computern zu erlernen und auf einfache Art zu verstehen. Im direkten Umgang mit digitalen Schaltungen erfährt man, daß die Digitalelektronik relativ leicht zu erlernen ist. Grundkenntnisse der Elektrizitätslehre und der Mathematik sind nützlich, aber nicht unbedingt erforderlich.

Die Arbeit mit LOCAD erleichtert nicht nur das Begreifen der zugrundeliegenden Schaltung, sondern ermutigt auch zu eigenen Schaltungsentwürfen.

Zum Lieferumfang der Vollversion gehören auch 3 Handbücher:

- Technische Information mit LOCAD (Erläuterung der Grundfunktionen; Normalformen; Technische Realisierung der Grundfunktionen; Schaltnetze; Halb-, Voll-, Paralleladdierer, Parallelsubtrahierer; Code-Umsetzer, Koppeldioden; ROM, PROM; Multiplexer; Digitale Speicherelemente; Rechenwerke; Struktur einer DVA; Steuersignalerzeugung; Lösungen zu Musterbeispielen)
- Messen, Steuern und Regeln mit LOCAD 2 (Entwicklung von Schaltungen etwa zur digitalen Spannungsmessung, zum Steuern eines Modellplotters oder zum Untersuchen einer Zweipunktregelung, die sich über ein Interface einsetzen lassen)
- Mikroprozessortechnik mit LOCAD 3 (Handhabung des Programmes; Architektur und Arbeitsweise eines Von-Neumann-Rechners; Aufbau eines Modellrechners mit mikroprogrammiertem Prozessor, Arbeitsspeicher und Ein/Ausgabeeinheit; Programmierung des Modellrechners, Grundzüge der Assemblerprogrammierung)
- LOCAD für Windows (Begleitheft zur neuen WINDOWS-Version von LOCAD)

Systemvoraussetzungen:

DOS-Version: IBM-kompatibler Computer mit mindestens 512 kByte Arbeitsspeicher; DOS ab 3.11; Standard-Videokarten; Festplatte von Vorteil, aber nicht notwendig; Maus zur Bedienung möglich

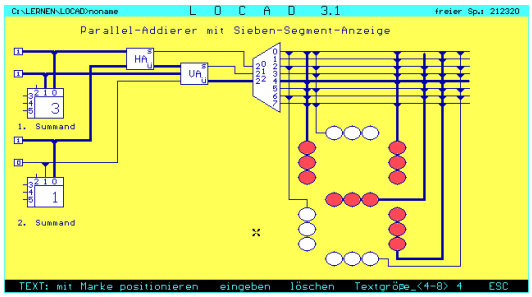
WINDOWS-Version: 386er oder höherer Prozessor; Grafik-Karte mit Mindestauflösung von 640 x 480 Pixeln (VGA-Karte); WINDOWS 3.1 oder höher; WINDOWS-kompatible Maus; ca. 2 MB Festplattenspeicher.

Mit dem mitgelieferten Installationsprogramm kann man LOCAD nach eigenen Wünschen konfigurieren. Man hat die Möglichkeit,

- zwischen alten und neuen DIN-Symbolen der Schaltsymbole zu wählen,
- die Farbeinstellungen für Rahmen, Arbeitsblatt, Linien, Bauteile, Texte und Marken zu verändern,
- gewünschte Druckersteuerzeichen einzustellen,
- die Bauelemente anzuwählen, die im Bauteilemenü zur Verfügung stehen sollen; beim Einsatz im Unterricht kann es didaktisch durchaus sinnvoll sein, bei den einfachen Bauelementen nur die Grundverknüpfungen *UND, ODER* und *NICHT* zuzulassen und bei den komplexeren nur die selbstdefinierten ICs („geschrumpfte“ Schaltungen),
- den Warnton ein- oder auszuschalten,
- die Mausgeschwindigkeit zu ändern,
- die Werte für die Simulationsverzögerung zu ändern und
- die Parallelschnittstelle für das Interface festzulegen.

Die soeben erst auf den Markt gekommene WINDOWS-Version von LOCAD bietet zusätzlich zu den oben angegebenen Möglichkeiten noch folgende:

- vereinfachte Menüführung durch den Einsatz von Schaltflächen,
- vergrößerte Arbeitsfläche (ca. 35% bei Super-VGA-Karten),
- Wegfall der 640 KB-Grenze,
- alle Bauelemente (Ausnahme: Ziffernanzeige) in allen Richtungen positionierbar,
- verbesserte Druckmöglichkeiten (Farbdruck),
- Farben und Symbole während der Laufzeit änderbar,
- manuelle Symbole jederzeit (also auch während bereits andere Signale laufen) auslösbar,
- vereinfachtes IC-Definitionsverfahren und
- kontextsensitive Online-Hilfe.



Beispiel einer einfachen Schaltung mit LOCAD 3 (DOS-Version): Parallel-Addierer mit Sieben-Segment-Anzeige. Die aktivierten Leitungen, die zum Ergebnis der „schwierigen“ Rechnung 011 + 001 = 100 (im Dualsystem) bzw. 3 + 1 = 4 (im Dezimalsystem) führen, werden vom Programm verstärkt gezeichnet.

Um eigene Schaltungen aufzubauen, wählt man die benötigten Bauteile aus dem Angebot, setzt sie mit Maus- oder Pfeiltasten an die gewünschten Stellen und verbindet sie mit Leitungen.

Das Verhalten der entworfenen Schaltungen läßt sich untersuchen, indem man (manuell oder mit Taktgeber) ihre Eingänge mit 1- oder 0-Signalen belegt. Die (auf Wunsch) zeitverzögerte Ausbreitung der 1-Signale wird durch eine Verbreiterung der Leitungsstriche veranschaulicht; auf diese Weise kann die Wirkung der Eingangsbelegung auf die Ausgänge gut beobachtet werden.

Das Programm und die dazugehörigen Handbücher sind didaktisch sehr geschickt gestaltet. Alle Themenbereiche beginnen mit einer theoretischen Einführung und einer Begriffsklärung, bevor sehr ausführlich auf die Schaltungen eingegangen wird. Anschließend werden zur persönlichen Kontrolle noch einige Aufgaben gestellt und Hinweise auf weitere Problemstellungen gegeben. Die Gliederung der Themen ist für den Lehrer eine ausgezeichnete Unterstützung bei der Unterrichtsvorbereitung. Sämtliche Schaltungen im Buch werden übrigens auf Diskette mitgeliefert. Man kann sie testen oder verändern und neu abspeichern.

Die untersuchten Schaltungen sollen die Vorgänge im Prozessor eines Computers auf Hardware-Ebene verständlich machen. So werden Rechenschaltungen vom Halbaddierer bis hin zum JK-Master-Slave-Flipflop aus Gattern aufgebaut, getestet und eingesetzt. Diese Schaltungen stellt das Programm aber zum weiteren Ausbau auch als eigene Bauteile zur Verfügung. Mit ihnen lassen sich dann Register und Elemente vom Rechenwerk und vom Steuerwerk aufbauen und untersuchen. Sehr von Vorteil ist auch die Möglichkeit, umfangreiche Schaltungen zu einer integrierten Schaltung (IC) „schrumpfen“ zu lassen; sie beseitigt die Beschränkung der Arbeitsfläche auf eine Bildschirmseite und ermöglicht durch stufenweise Integration von Schaltungen den Aufbau komplexer Systeme wie z.B. den Aufbau eines kompletten mikroprogrammierten Rechners.

Ich selbst verwende das Programm im Mathematik-Unterricht 5. Klasse AHS zum Lehrstoff Schaltalgebra (Grundschaltungen und ihre Verknüpfungen, Entwerfen von Schaltungen, Äquivalente Schaltungen), im Wahlpflichtgegenstand Mathematik (Normalformen, Anwendungen), Informatik 5. Klasse sowie Wahlpflicht Informatik (Schaltfunktionen, Schaltnetze, Schaltwerke; Funktionsweise von Computerelementen, Speichern oder Verarbeiten von Daten). Einsatzmöglichkeiten im physikalisch-technischen Bereich sind ebenfalls leicht vorstellbar (Messen - Steuern - Regeln, ...).

Preise LOCAD 3 (DOS oder WINDOWS):

DOS-Einzellizenz:	DM 189.-	WIN-Upd-Einzellizenz:	DM 39.-
DOS_Schullizenz:	DM 389.-	WIN-Upd-Schullizenz:	DM 59.-

Weitere Informationen, Demoversionen (eingeschränkte Vollversion - Schaltungen nur bis 9 Bauteile) oder Bestellungen:

PÄDAGOGIK & HOCHSCHULVERLAG
Raderfeld 30
D-47807 Krefeld ☐

ANALOGSIMULATION MIT PSPICE

Klaus P. Eckl PCN-DSK 139, 149; PCN-LIT 12

1. Allgemeines

Simulieren von Schaltungen heißt messen. Messen bedeutet bekannte Gegebenheiten überprüfen. Um richtig messen zu können, muß man die Eigenschaften des Prüflings gut kennen, auf Grund des erwarteten Analyseergebnisses die richtigen Methoden zu wählen. Wahlloses Probieren führt in beiden Disziplinen zu Mißerfolg. Die Interpretation des Ergebnisses auf Basis der gewählten Methoden erfordert großes Wissen.

SPICE (Simulation Program with Integrated Circuit Emphasis) ist eine Public Domain Software, weil sie mit Unterstützung öffentlicher Mittel an der University of California ursprünglich für Forschungszwecke entwickelt, aber sehr bald zu einem Industriestandard wurde. Mit Spice lassen sich beliebige Modelle wie auch z.B. mechanische Aufbauten simulieren. Von Softwareherstellern wurde das Kernprogramm nun durch Benutzeroberflächen und Modellbibliotheken erweitert. Die zwei bekanntesten Simulationsprogramme sind PSpice von MicroSim und Micro-Cap von Spectrum Software. Neben den Kosten für Schnittstellen zu CAD Programmen oder graphischen Eingabemöglichkeiten oder Postprozessoren zur graphischen Ergebnisdarstellung, schlagen sich der große Aufwand für Bauteilbibliotheken im hohen Anschaffungspreis der Software nieder.

Spice berechnet die Netzwerke mit der Methode der Knotenpotentialanalyse. Ein Netzwerk mit m Knoten (nodes) wird auf m-1 Knoten reduziert, weil einer zum Bezugspotential (Masse, ground) erklärt wird. Die restlichen Knoten werden numeriert (z.B. von 1 - n) und von diesen führen zum Bezugspotential die Potentialdifferenzen (Knotenspannungen) V1 - Vn. Für die entstehenden Maschen lassen sich mit Hilfe der Kirchoffschen Regeln das folgende Gleichungssystem aufstellen.

$$\begin{pmatrix} Y11 & Y12 & .. & Y1n \\ Y21 & Y22 & .. & Y2n \\ .. & .. & .. & .. \\ Yn1 & Yn2 & .. & Ynn \end{pmatrix} * \begin{pmatrix} V1 \\ V2 \\ .. \\ Vn \end{pmatrix} = \begin{pmatrix} I1 \\ I2 \\ \\ In \end{pmatrix}$$

Dabei sind die Ströme I1 - In jeweils die Summe aller Ströme, die in die Knoten 1 - n fließen. (Ströme, die aus dem Knoten fließen werden negativ gezählt.) V1 - Vn sind die oben genannten Knotenspannungen. Die Yik sind die Leitwerte der Bauelemente zwischen den Knoten. Dabei sind die Yii gleich der Summe aller Leitwerte, welche an dem jeweiligen Knoten i angeschlossen sind. Yik ist der Leitwert zwischen dem Knoten i und k und er ist mit negativem Vorzeichen zu versehen, wenn man beim Durchlaufen der Masche von i nach k dieselbe Richtung eingeschlagen hat, wie die Zählpfeilrichtung von Vk zeigt.

Für das im ***Bild 1*** gezeigte Beispiel können die Gleichungen wie anschließend gezeigt lauten.

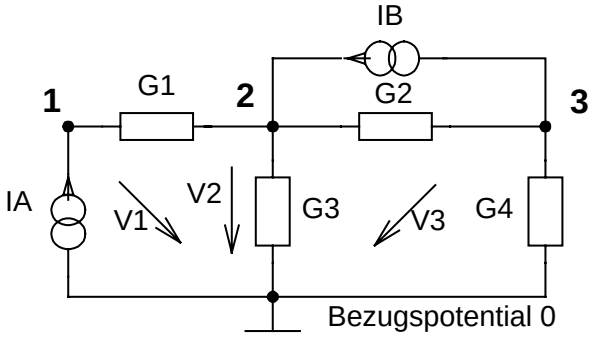


Abbildung 1: **Netzwerk mit 4 Knoten**

$$\begin{pmatrix} G1 & -G1 & 0 \\ -G1 & G1+G2+G3 & -G2 \\ 0 & -G2 & G2+G4 \end{pmatrix} * \begin{pmatrix} V1 \\ V2 \\ V3 \end{pmatrix} = \begin{pmatrix} IA \\ IB \\ -IB \end{pmatrix}$$

Dabei ist zu beachten, daß Spice nur mit idealen Bauteilen rechnen kann und daher folgende Einschränkungen gelten. Spice führt im ersten Schritt eine DC Analyse durch und sucht für jeden Knoten einen Gleichstromweg zum Bezugspotential. Kondensatoren riegeln Gleichstrom jedoch ab und die Serienschaltung von zwei idealen Kondensatoren führt zu einem undefinierten Potential im Verbindungspunkt (floating node). Bei realen Kondensatoren fließt ein sehr geringer Leckstrom und diesen kann man durch Parallelschalten von großen Widerständen (GΩ) nachbilden. Ein ähnliches Problem ergibt sich bei der Parallelschaltung von zwei idealen Spulen. Flußänderungen erzeugen Induktionsspannungen und diese werden als Spannungsquelle nachgebildet. Zwei Spannungsquellen mit Innenwiderstand 0 darf man jedoch nicht parallel schalten. In diesem Fall sind zu den idealen Spulen Serienwiderstände zu setzen. Für Gleichstrom wird die ideale Spule entsprechend dem Innenwiderstand einer Spannungsquelle ein Kurzschluß.

2. Programmversionen

Am TGM stehen zwei Versionen zur Verfügung. Eine Evaluationversion von PSpice 6 für Windows, welche den Vorteil der graphischen Schaltungseingabe, aber nur eine sehr eingeschränkte Bauteilbibliothek und geringe Anzahl an Knoten zuläßt, und eine Vollversion von PSpice 5 für DOS, bei welcher man die Schaltung textuell mit Hilfe eines Editors eingeben muß. Die Vollversion besitzt auch eine Shell, kann aber wegen des benötigten DOS-Extenders nicht unter Windows gestartet werden. Für die Eingabe von Stimuli (Eingangstestspannungen unterschiedlichster Kurvenform) stehen ebenso wie für die Einstellung der Simulationsparameter Menüs bez. Programme in beiden Versionen zur Verfügung. Die Kenntnis der Bauteile, Funktionen oder Parameter ist bei beiden Versionen erforderlich.

3. Komponenten, Parameter und Funktionen

Im xxx.CIR File wird die Schaltung, welche man simulieren will, beschrieben. Jedes Element steht mit seinem Namen in einer neuen Zeile und der erste Buchstabe ist signifikant für das Bauteil. (Siehe ***Tabelle 1***.) .Durch Leerzeichen getrennt folgen nun die Knotennamen (Nummern)(für einen Widerstand zwei, für Transistoren drei usw.) und wenn erforderlich, als Zusatzangabe der Bauteilwert. Eingegeben kann mit Kommastellen oder den Zeichen der technischen Notation werden. (siehe ***Tabelle 2***).

Tabelle 1: Bauteilnamen in PSpice

Name:	type	Bauteil:
Vxxx	VS	Spannungsquelle
Ixxx	CS	Stromquelle
Exxx	VCVS	spannungsgesteuerte Spannungsquelle
Fxxx	CCCS	stromgesteuerte Stromquelle
Gxxx	VCCS	spannungsgesteuerte Stromquelle
Hxxx	CCVS	stromgesteuerte Spannungsquelle
Sxxx	VSWITCH	spannungsgesteuerter Schalter
Wxxx	ISWITCH	stromgesteuerter Schalter
Rxxx	RES	Ohmscher Widerstand
Cxxx	CAP	Kondensator
Lxxx	IND	Induktivität
Dxxx	D	Diode
Qxxx	NPN	NPN Transistor
Qxxx	PNP	PNP Transistor
Qxxx	LPNP	lateral PNP Transistor
Jxxx	NJF	n Kanal J-FET
Jxxx	PJF	p Kanal J-FET
Mxxx	NMOS	N Kanal MOS FET

Mxxx	PMOS	P Kanal MOS FET
Kxxx	CORE	nichtlinearer magnetischer Kreis
Bxxx	GASFET	N Kanal GaAs MESFET
Nxxx	DINPUT	digitaler Eingang (von digitalem Ausgang gesteuert)
Oxxx	DOUTPUT	digitaler Ausgang (von digitalem Eingang gesteuert)
Uxxx	UIO	digitales I / O Modell
Uxxx	UGATE	Standard Gatter
Uxxx	UTGATE	Tri state Gatter
Uxxx	UEFF	Flankengesteuertes Flip Flop
Uxxx	UGFF	Zustandsgesteuertes Flip Flop
Uxxx	UWDTH	Pulsbreitentester
Uxxx	USUHD	setup und hold Prüfer
Uxxx	UDLY	digitale Verzögerungsleitung
Uxxx	UADC	n bit DAC
Uxxx	UDAC	n Bit DAC
Xxxx		Subcircuits wie OP, 555 timer etc.

Wie in den folgenden Beispielen ersichtlich muß jedes Design eine Titelzeile erhalten. Weitere Kommentarzeilen müssen mit einem * beginnen bez. in einer Eingabezeile trennt der „ ; “.

Tabelle 2:
Abkürzungen der technischen Notation (Scale Suffixes)

Symbol		Faktor	Symbol		Faktor
f	F	10-15	m	M	10-3
p	P	10-12	k	K	103
n	N	10-9	meg	MEG	106
u	U	10-6	g	G	109
mil	MIL	25.4*10-6	t	T	1012

In Spice 6 erfolgt die Schaltungseingabe graphisch. Dabei wird automatisch ein Referenzdesignator (z.B. R1) und für Bauteile ein Defaultwert vergeben. Ebenso werden die Knoten mit Nummern versehen. Mit einem Doppelklick der Maus auf solche Werte wird ein Fenster geöffnet und man kann die Werte verändern. Um an bestimmten Knoten messen zu können, ist es vorteilhaft diese zu benennen (einen Label zu vergeben). Mit einem Doppelklick auf die Leitung und benennt man diese (z.B. IN oder OUT). Dies erleichtert beim Arbeiten mit dem Postprozessor zur graphischen Darstellung das Auffinden von Meßpunkten.

Spice 6 ist ein Programm, das voll die Oberfläche von Windows benutzt und über Pull Down Menüs ausgeführt werden kann.

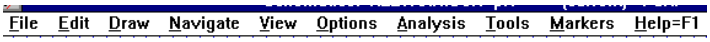


Abbildung 2: **Menüleiste von PSpice 6**

Abbildung 2 zeigt die Hauptgruppen der Befehle. Im Rollbalken File finden wir das für Windows übliche File handling, Ausdrucken etc., sowie auch die Möglichkeiten zum Editieren von Bauteilbibliotheken. Das Edit Menü erlaubt Ausschneiden, Kopieren, Rotieren sowie auch das Editieren von Stimuli und Modellen. Im Draw Block finden sich alle Befehle zum Erstellen eines Schaltbildes (Schematic). Navigate wird von der Evaluationsversion kaum unterstützt und würde das Erstellen von Multisheet und Hierarchischen Schaltungen erlauben. Die Befehle von View enthalten alle wichtigen Zoomfunktionen. Im Balken Options kann man die Zeichenhilfen wie Gridgröße, Fangbereich, Autonumerierung der Bauteile, den Bibliotheksseditor etc. einstellen bez. aktivieren. Der Rollbalken Analysis enthält die wichtigsten Konfigurationsdaten und Steuerbefehle für den Simulator und wird noch gesondert ausführlich behandelt. In der Box Tools finden sich einige Zusatzwerkzeuge wie Netzliste erstellen, Netzlistenbrowser oder Erstellen von Subcircuits.

Beim Erstellen eines Schematic setzt man zuerst alle Bauelemente. Dazu wählt man aus dem Pull Down Menü Draw den Button Get New Part. Über einen Browser (Zusatzfenster mit Suchfunktion) kann man aus den Bauteilbibliotheken das gewünschte Symbol finden und mit der Maus plazieren. Mit Hilfe der weiteren Befehle von Draw und Edit kann man verdrahten, beschriften, verändern usw. . Bauteilwerte, Referenzdesignatoren oder Labels kann man durch Doppelklick der Maus in ein Editierfenster holen und dort die meist gegebenen Defaultwert ausbessern. Die meisten Befehle des Schematiceditors lassen sich auch über Hot Keys aufrufen. Die wichtigsten sind in der folgenden Tabelle zusammengestellt.

Ctrl S	File Save	Ctrl P	Place Part
Ctrl C	Kopierenanwahl	Ctrl R	Rotieren 90°
Ctrl F	Spiegeln	Ctrl S	File Save
Ctrl G	Get New Part	Ctrl T	Draw Text
Ctrl I	Zoom In	Ctrl U	Undelete
Ctrl L	Redraw	Ctrl V	Einfügen
Ctrl M	Spannungs-Level	Ctrl W	Verdrahten
Ctrl N	Fit	Ctrl X	Ausschneiden
Ctrl O	Zoom Out	Space	Wiederholfunktion

Wenn man als Eingangssignal einen Stimulus (Strom oder Spannung mit bestimmter Kurvenform) als Symbol wählt und zur Eingabe von Werten auf das Symbol den Doppelklick setzt, wird automatisch ein Stimuluseditor aufgerufen, welcher nach Eingabe der Werte das Signal auch graphisch in einem Window darstellt. Stimuli werden in separaten Files abgelegt und beim Aufruf des Simulators in das *.CIR File automatisch Inkludiert.

Mit der Funktionstaste F11 oder Button Analysis/Simulate kann der Simulator gestartet werden. Dabei werden die beiden Programme Netlist (Netzliste aus dem Schematic erstellen) und Rule Check (Suche nach Verdrahtungsfehlern wie offene Knoten etc.) automatisch aufgerufen. Bevor man den Simulator aufruft, sollte man jedoch den Button Setup anwählen, um die Methoden und Parameter zu setzen. Es öffnet sich das Window von ***Abbildung 3***:

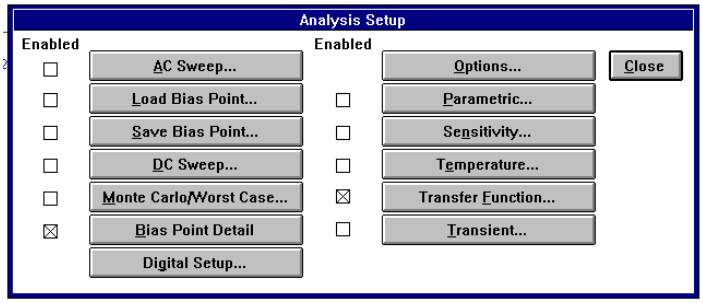


Abbildung 3: **Analysis Setup Window**

Für jeden aktivierten (Enabled) Button wird in das Simulationsfile (*.CIR) ein Statement (Anweisung) als eine neue Zeile eingefügt. Den Statements muß ein „ . “ vorgesetzt werden. So muß das `*.CIR` File mit `.END` abgeschlossen werden (macht Spice 6 automatisch). Das Statement .OP setzt Spice 6 auch immer und weist das System an, eine Gleichstromanalyse zu erstellen. Dabei werden Kondensatoren wie Unterbrechungen und Spulen wie Kurzschlüsse behandelt. Die errechneten Potentiale stellen die Arbeitspunkte des Netzwerkes für allfällige Simulationen mit Kleinsignalparametern dar. Die errechneten Simulationswerte werden im `*.OUT` File ausgegeben und können mit jedem Texteditor gelesen werden. Mit Bias Point Detail Enabled werden diese Werte im `*.OUT` File aufgenommen. So kann man z.B. die Arbeitspunkte von Transistoren kontrollieren. In der Folge wollen wir Simulationen mit der Schaltung von ***Abbildung 4*** durchführen. Die Simulationsergebnisse sind teilweise im File xxx.OUT zu finden oder können mit Hilfe des Postprozessors PROBE graphisch dargestellt werden. Auf Seite 7 ist die Ausgabe der DC Werte im `*.OUT` File wiedergegeben.

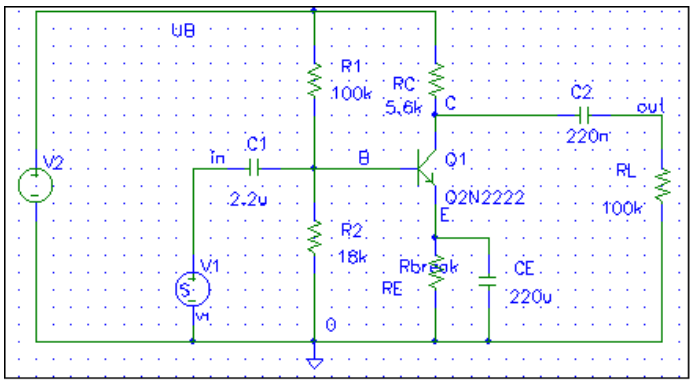


Abbildung 4: **Schematic eines einstufigen Transistorverstärkers**

Einige Leitungen wurden mit Labels (anklicken der Leitung mit der Maus) versehen und neben den in Klammern gesetzten Namen stehen die Gleichspannungswerte gegen Masse gemessen.

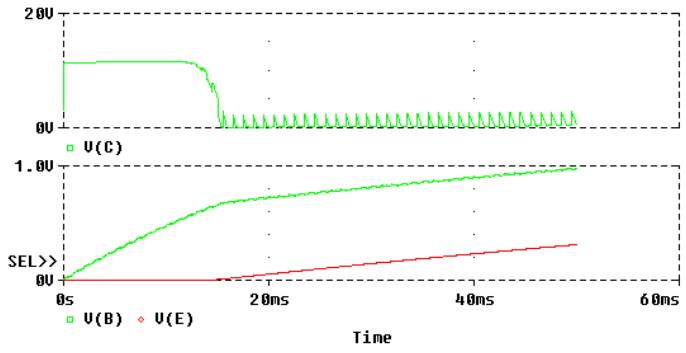


Abbildung 5: **Transientenanalyse des einstufigen Transistorverstärkers mit BIAS Berechnung**

Als Analyseart wurde im Setup Window der Button „Transient“ angewählt. In den freigegebenen Datenfenstern wurden Simulationszeit, die Simulationsschritte und Printschritte auf ca. 1/500 der Zeit gestellt. Weiters wurde die Option „Use init. Conditions“ gewählt.

Nach der Simulation wird der Postprozessor automatisch gestartet. Das leere Plotfenster kann mit Hilfe der Befehle der Rollbalken von „Trace“ und „Plot“ mit Diagrammen mit linearen und logarithmischen Achsen mit einem oder mehreren Kurvenzügen gefüllt werden. Zur Vorbereitung wurde die Quelle VStim durch Doppelklick der Maus zum Leben erweckt. Automatisch wird ein Stimuluseditor aufgerufen, der die Konstruktion von beliebigen Stimuli erlaubt. Eingegeben wurde eine Rechteckspannung. Das Simulationsergebnis des gut dimensionierten Verstärkers, zeigt keine vertrautes Ausgangssignal. Bei der Transientenanalyse mit Übernahme der Initialwerte sind alle Kondensatoren ungeladen und die Spulen stromlos. Spice errechnet wieder schrittweise den Arbeitspunkt und den Einschwingvorgang. Man sieht wie sich in den ersten 50 ms der Emitterkondensator langsam auflädt und der Transistor am Ausgang noch voll übersteuert ist.

```
 ****     SMALL SIGNAL BIAS SOLUTION       TEMPERATURE =   27.000 DEG C
******************************************************************************
 NODE   VOLTAGE     NODE   VOLTAGE     NODE   VOLTAGE     NODE   VOLTAGE
(    B)    1.7259  (    C)    5.9924  (    E)    1.0796  (   in)    0.0000
(   UB)   12.0000  (  out)    0.0000
```

Abbildung 6: **Spannungen gegen O der Schaltung 4 (xx.OUT)**

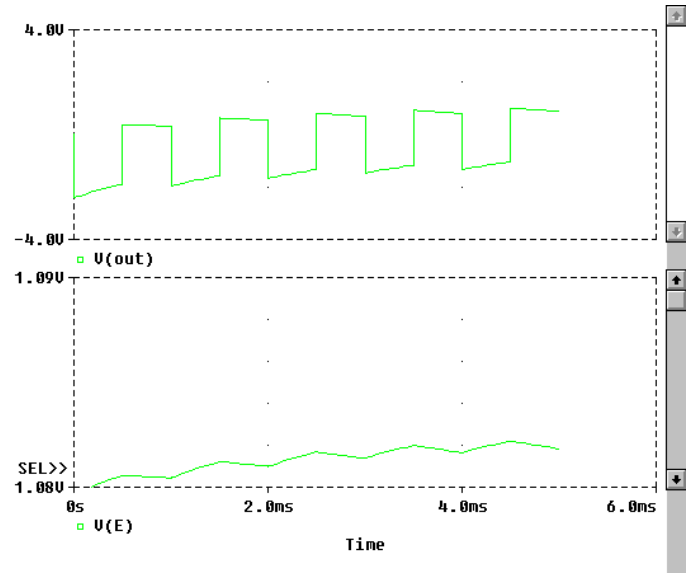


Abbildung 7: **Transientenanalyse von Schaltung 4**

Beim zweiten Versucht schalten wir die Initial conditions aus. Jetzt beginnt Spice von den bereits errechneten Gleichstromknotenspannungen zu simulieren an. Durch die relativ große Eingangsspannung muß, verursacht durch die nichtlineare Steuerkennlinie des Transistors, der Emitterkondensator noch geringfügig nachgeladen werden. Man sieht auch, daß die Spannung am Emitterkondensator leicht wellig bleibt (einige mV). Das Rechtecksignal am Ausgang weist wegen der, für die Meßfrequenz von 1 kHz, zu klein gewählten Kondensatoren eine Dachschräge auf.

Bei der Fourieranalyse kann man noch optional zusätzlich eine Fourieranalyse durchführen lassen. (Button Transientenanalyse). Dabei ist die Frequenz der Grundschwingung und die Anzahl der gewünschten Oberwellen im Setup Window einzugeben. Um das Spektrum graphisch anzuzeigen, ist im Programm Probe der Rollbalken Plot anzuwählen. Der Befehl X Axis Settings erlaubt das Umschalten des Zeitverlaufes in eine Spektralabbildung. Weiters werden die Spektralanteile mit Betrag und Phasenwinkel im xx:OUT File festgehalten.

Will man den Einschwingvorgang gänzlich unterdrücken, dann wählt man den Button „Save BIAS“. Damit werden am Ende der Transientenanalyse die Knotenspannungen in ein File abgespeichert und können beim folgenden Start der Analyse durch setzen von „Load BIAS“ wieder gesetzt werden.

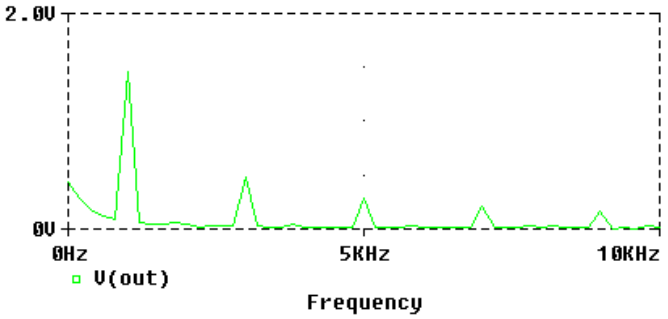


Abbildung 8: **Spektrum des Ausgangssignals vom Transient Bild 7**

```
FOURIER COMPONENTS OF TRANSIENT RESPONSE V(out)

 DC COMPONENT =  -1.071329E-01

 HARMONIC   FREQUENCY    FOURIER    NORMALIZED    PHASE        NORMALIZED
    NO        (HZ)      COMPONENT   COMPONENT    (DEG)        PHASE (DEG)
     1     1.000E+03    1.355E+00   1.000E+00   -1.774E+02     0.000E+00
     2     2.000E+03    2.940E-02   2.170E-02    1.392E+02     3.166E+02
     3     3.000E+03    4.524E-01   3.339E-01    1.786E+02     3.560E+02
     4     4.000E+03    2.248E-02   1.659E-02    1.168E+02     2.942E+02
     5     5.000E+03    2.716E-01   2.004E-01    1.764E+02     3.538E+02
     6     6.000E+03    2.099E-02   1.549E-02    1.056E+02     2.830E+02
     7     7.000E+03    1.941E-01   1.432E-01    1.745E+02     3.519E+02
     8     8.000E+03    2.050E-02   1.513E-02    9.861E+01     2.760E+02
     9     9.000E+03    1.511E-01   1.115E-01    1.727E+02     3.501E+02

    TOTAL HARMONIC DISTORTION =   4.310681E+01 PERCENT
```

Abbildung 9: **Spektralanteile des Ausgangsrechtecksignals von** *Bild 7*

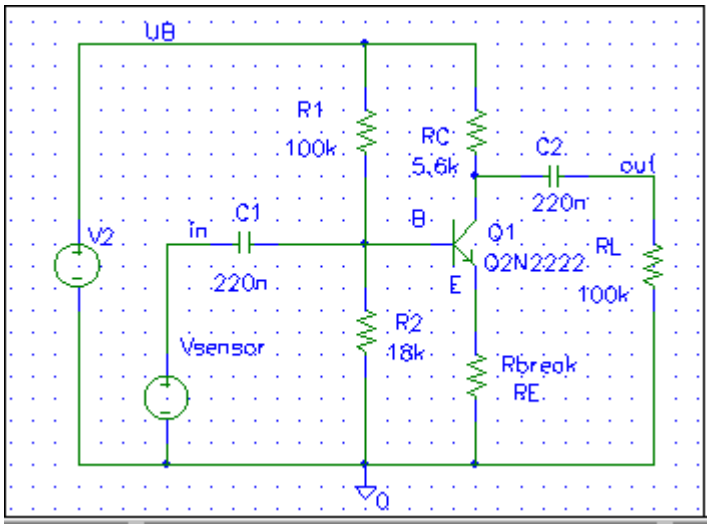


Abbildung 10: **Gegengekoppelter Transistorvorstärker**

Nun entfernen wir den Emitterkondensator und wollen das Wechselstromverhalten des gegengekoppelten Verstärkers untersuchen. Die Quelle VStim wird durch den Typ VSRC ersetzt. Damit erhalten wir eine Sinusspannung am Eingang. Der Button „AC Analyse“ wird aktiv gesetzt und durch anklicken wird ein Fenster zur Eingabe der Simulationsparameter geöffnet. Um die Veränderung der Verstärkung bei Variation des Emitterwiderstandes RE studieren zu können, wählen wir ein Widerstandssymbol Rbreak, welches stufenweise seinen Nennwert ändern läßt. Der Botton Parametric wird aktiv gesetzt und das Editierfenster durch Anklicken des Button geöffnet. In Abbildung 11 sind die

gewählten Werte zu sehen. Der Startwert 0.5 bedeutet, daß die Simulation mit einem Emitterwiderstand von 0.5*Nennwert beginnt und der Emitterwiderstand dann stufenweise um 25 % seines Nennwertes erhöht wird. Abbildung 12 zeigt als Ergebnis die 5 errechneten Meßreihen. Das Ergebnis kann jedoch nicht der Wirklichkeit entsprechen, weil wir bei einer Batteriespannungen von 12 V ein Ausgangssignal von 10 V erreichen. Spice rechnet bei der AC Analyse mit einem linearen Transistormodell, welches seine Kleinsignalparameter sehr wohl vom DC Arbeitspunkt abhängig macht, aber dieses lineare Modell gilt eben nur für kleine Signale und berücksichtigt keine Übersteuerungen.

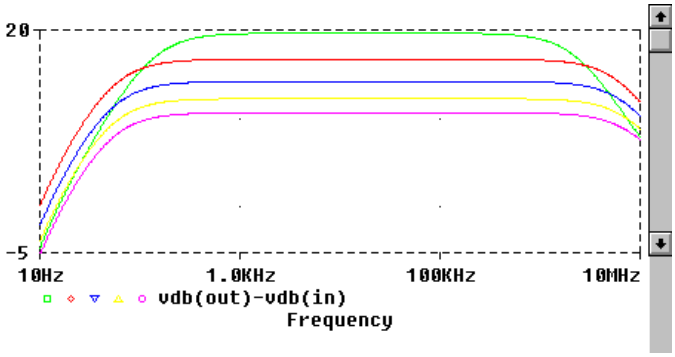


Abbildung 11: **Amplitudengang des gegengekoppelten Verstärkers**

Die Y Achse kann im Rollbalkenmenü „Plot“ auch mit dem Befehl Y Axis auf logarithmisch gestellt werden. Günstiger ist es jedoch wenn man einen dB-Maßstab wählt. Das Programm Probe bietet auch eine Reihe von Rechenfunktionen wie VdB(xx) für die Spannung des Knotens xx, oder VP (xx) für das Phasenmaß oder VG(xx) für die Gruppenlaufzeit. Weiters beherrscht das Programm auch alle Grundrechenarten.

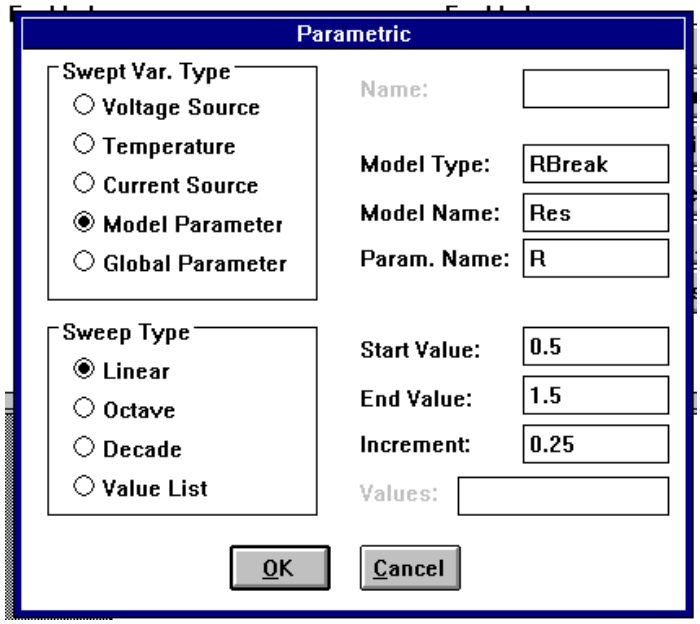


Abbildung 12: **Eingabefenster des Button Parametric**

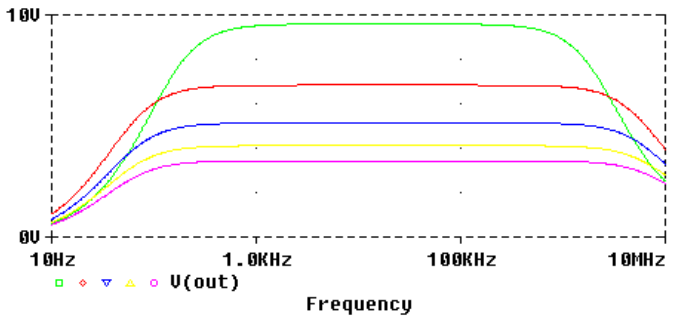


Abbildung 13: **AC Analyse des Verstärkers von Bild 10 mit linearer Y Achse**

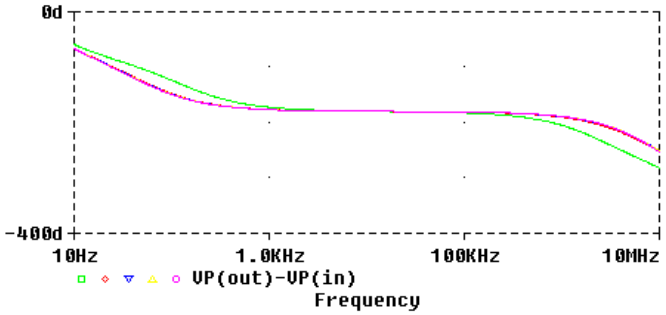


Abbildung 14: **Phasengang des gegengekoppelten Verstärkers**

Auf Seite 10 sind zwei von den drei genannten Diagrammen zu sehen. Gleichzeitig mit der AC Analyse kann auch eine Rauschanalyse durchgeführt werden. Zu diesem Zweck haben wir unsere VSRC Quelle mit dem Referenzdesignator VSensor versehen. Die Einstellungen im mit Button „AC Analyse“ aktivierten Einstellfenster zeigt Abbildung 16. Je nachdem welche X - Achse man im Diagramm plant, wird man eine lineare oder dekadische (logarithmische) Analyse mit gegebener Anzahl von Rechenpunkten wählen.

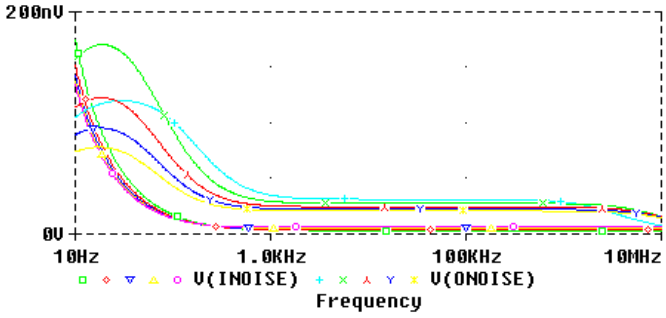


Abbildung 15: Rauschanalyse des gegengekoppelten Verstärkers

Die Rauschanalyse ermittelt den Effektivwert der äquivalenten Rauscheingangsspannung am Ort des Sensors und auch an einem beliebigen Knotenpunkt. In Abbildung 15 ist das Ergebnis zu sehen. Die an den Eingang versetzte äquivalente Rauschspannung zeigt das typische 1/f Rauschen des Transistors und ein sehr geringes Weißes Rauschen der Widerstände. Am Ausgang nach den Koppelkondensatoren hat das Rauschen entsprechend der Filterung bei etwa 100 Hz ein Maximum.

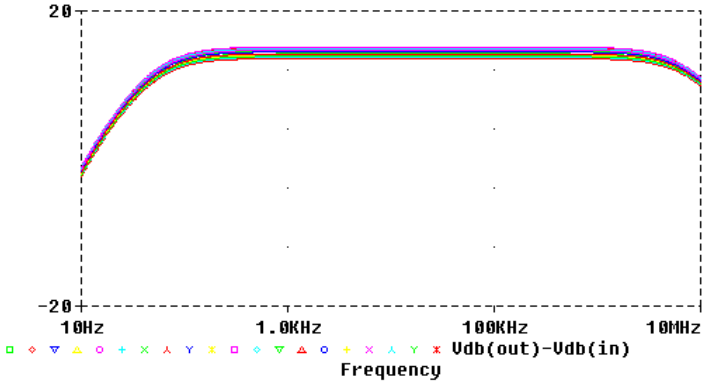


Abbildung 16: **Monte Carlo Analyse (RE) des einstufigen Verstärkers**

Mit der Option Parametric lassen sich eine Vielzahl von Werten, wie etwa die Temperatur oder die Stromverstärkung eines Transistors, stufenweise ändern. Eine weitere Möglichkeiten Schaltungen auf Bauteiltoleranzen zu untersuchen ist die Worst-Case und Monte-Carlo-Analyse. Beide setzen voraus, daß das Modell Bauteiltoleranzen enthält. In Spice für Windows läßt es z.B. das Modell Rbreak zu, Toleranzgrenzen einzugeben. Mit dem Rollbalkenbefehl Edit/Modell wird der Modelleditor aufgerufen. Durch einfügen der Zeile:

```
R=1 DEV=10%
```

wird mitgeteilt, daß der Nennwert um bis zu 10 % sich verändern kann. Mit der Worst-Case-Analyse kann man nun feststellen, um wieviel sich maximal die Eigenschaften der Schaltung bei Variation aller toleranzbehafteten Modellen ändert. Bei der Monte Carlo Analyse muß man eine Anzahl von Durchläufen wählen. Für jede Berechnung wählt das Programm eine Zufallszahl

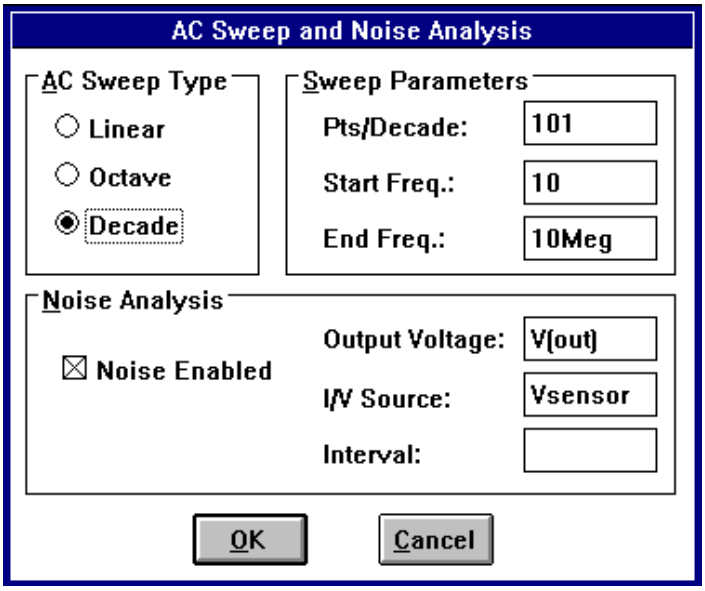


Abbildung 16: **Konfigurationsfenster der AC-Analyse**

und multipliziert im Rahmen der Toleranzen alle Modelle mit Variationsmöglichkeit. Das Ergebnis einer solchen Analyse für einen toleranzbehafteten Emitterwiderstand zeigt die ***Abbildung 17***.

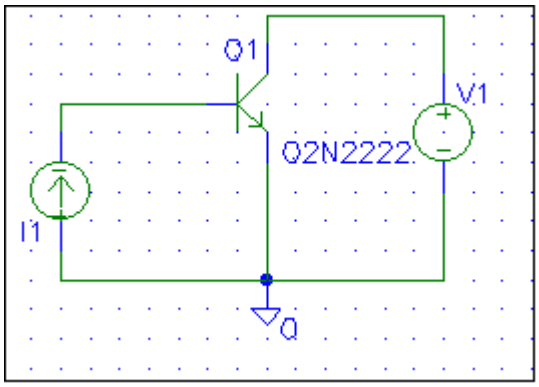


Abbildung 17: **Schaltung zur Aufnahme der Transistorkennlinie**

Zur Demonstration der DC Analyse wollen wir das Ausgangskennlinienfeld eines Transistors aufnehmen. Die Meßschaltung ist in *Abbildung 17* zu sehen. Die DC Analyse erlaubt das verschachtelte Ändern von Bauteilwerten, besonders jedoch das stufenweise variieren von Strom- und Spannungsquellen. Als Main Sweep wurde die Stromquellen in 10 µA Stufen beginnend von 10 µA bis 100 µA geändert und verschachtelt dazu die Spannungsquelle von

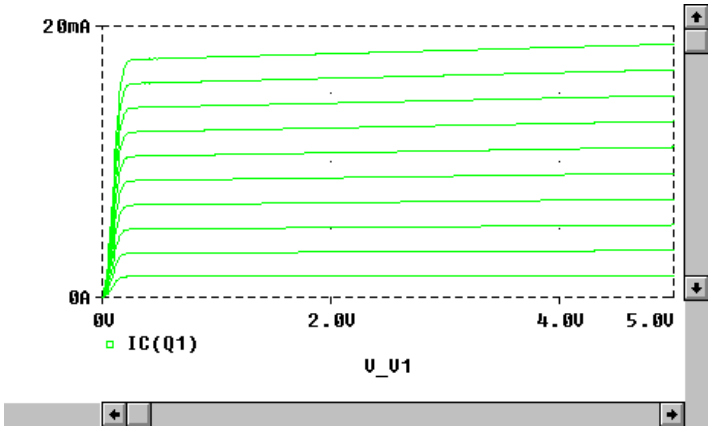


Abbildung 18: **Transistorausgangskennlinienfeld**

0 - 10 V in Schritten von 0.01 V. Das Ergebnis sieht man in Abbildung 18.

```
** Analysis setup **
.ac DEC 101 10 10Meg
.noise  V([out]) V_Vsensor
.STEP LIN Res Rbreak (R) 0.5 1.5 0.25
.OP
.STMLIB TRTRAN.stl
```

□

WinKon

Mit WinKon werden klassische Konstruktionsverfahren mit moderner Bedienerführung so kombiniert, daß Lernen und Lehren geometrischer Zusammenhänge gleichermaßen zum Vergnügen wird.

Michael Kugler

WinKon, ein Programm zum Konstruieren, wie mit Zirkel und Lineal, so steht es in der Einleitung zu der Hilfe zu diesem Programm. Um zu testen, ob das Programm hält, was es verspricht, werde ich eine Zeichnung mit WinKon konstruieren. Die letzte „schöne“ Zeichnung aus dem Gebiete der Darstellenden Geometrie habe ich während meines Studiums gemacht, Grund genug, mein altes DG Buch hervorzuholen und schon längst Vergessenes wieder zu zeichnen. Als Aufgabe habe ich mir folgendes Beispiel ausgewählt:

Eine Kugel ist durch ihren Mittelpunkt M und ihren Radius r gegeben. Diese Kugel soll mit der drittprojizierenden Ebene ε, die durch die Punkte P und Q gegeben ist, geschnitten werden.

Die speziellen Werte sind:

```
M(5|-4|4), r=4, P(2|0|8), q(10|0|5).
```

Das DG-Buch (Lichtensteiner; Darstellende Geometrie Band 2) erläutert mir den Lösungsweg:

1. Zeichne den Kreuzriß. Die wahren Umrisse für die drei Projektionsrichtungen sind die Großkreise k1, k2, und k3. Der Kreuzriß des Schnittkreises ist die auf ε‴ liegende Sehne des scheinbaren Umrisses u3‴.
2. Im Aufriß und im Grundriß erscheint der Schnittkreis als Ellipse. Durch Ermittlung der großen Halbachse und der Brennpunkte ist die Ellipse zu zeichnen.

Nun zur Durchführung des ersten Schrittes in WinKon:

```
.Die Angaben in Variablennamen ablegen
mx=5
my=-4
mz=4
r=4
px=2
py=0
pz=8
qx=10
qy=0
qz=5

.Zeichnung der Großkreise
M1(my,-mx)
M2(my,mz)
M3(mx,mz)

k1=krs(M1,r)
k2=krs(M2,r)
k3=krs(M3,r)

.Die Punkte P und Q zeichnen
P1(py,-px)
P2(py,pz)
P3(px,pz)
Q1(qy,-qx)
Q2(qy,qz)
Q3(qx,qz)

.Die Ebene im Kreuzriß einzeichnen
ep3=ger(P3,Q3)
```

Die Eingabe und die gleichzeitig entstehende Konstruktion in der WinKon-Oberfläche zeigt ***Bild 1***. In den Bildern 2..8 wurde der umgebende Bildschirm aus Gründen der Übersichtlichkeit nicht mehr dargestellt.

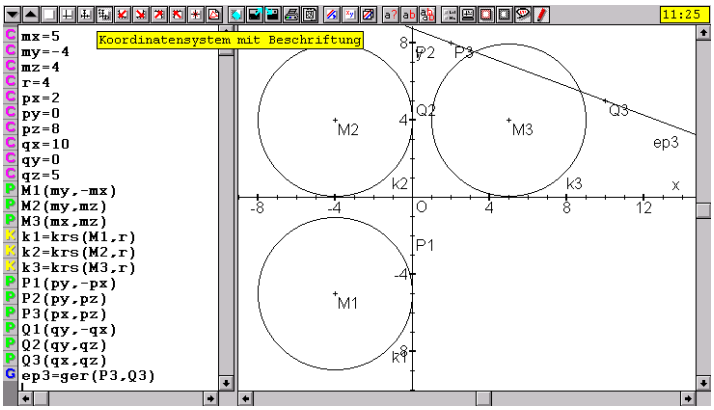


Bild 1: Eingabe und gleichzeitige Darstellung in WinKon. Sprechblasentechnik erleichtert das Arbeiten mit den Symbolen

```
.einige Hilfslinien
n3=nor(M3,ep3)
pi3=nor(M3,x)
u2=nor(M2,x)
u1=nor(M1,y)

.Schnittpunkte  im Kreuzriß
C3=pkt (ep3,k3)
V3=pkt (pi3,ep3)
N3=pkt (n3,ep3)

.nun die Ordner zum Kreis k1 im Aufriß ziehen
c3=ger(C3,x)
c3'=ger(C3',x)
v3=ger(V3,x)
n3'=ger(N3,x)
```

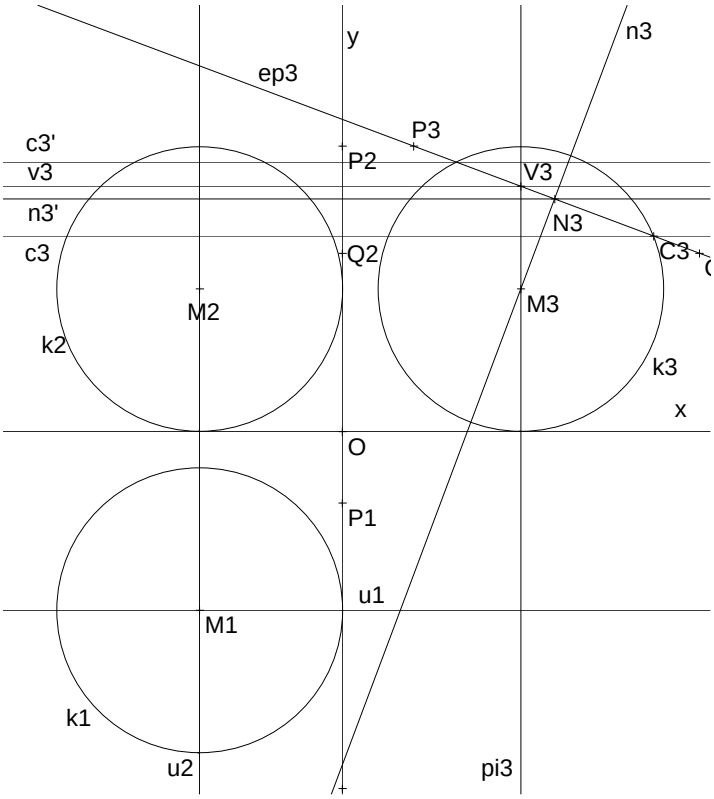


Bild 2:

```
.Nun die Schnittpunkte der Ellipse mit dem Kreis
C2=pkt(c3',u2)
D2=pkt(c3 ,u2)
V2=pkt(v3,k2)
N2=pkt(n3',u2)

. mit diesen Punkten ist nun die Ellipse zu zeichnen
. der Radius des Schnittkreises ist die Länge von C3 nach N3
r'=|str(N3,C3)|
.Die Punkte A und B der Ellipse sind r' vom Mittelpunkt N2
entfernt, und werden mit einer Hilfskonstruktion konstruiert.
kh2=krs(N2,r')
A2=pkt(kh2,n3')

.Die Bestimmung der Brennpunkte
kh3=krs(C2,r')
F2=pkt(kh3,n3')

.Das Zeichnen der Ellipse
el2=ell(F2,F2',r')
```

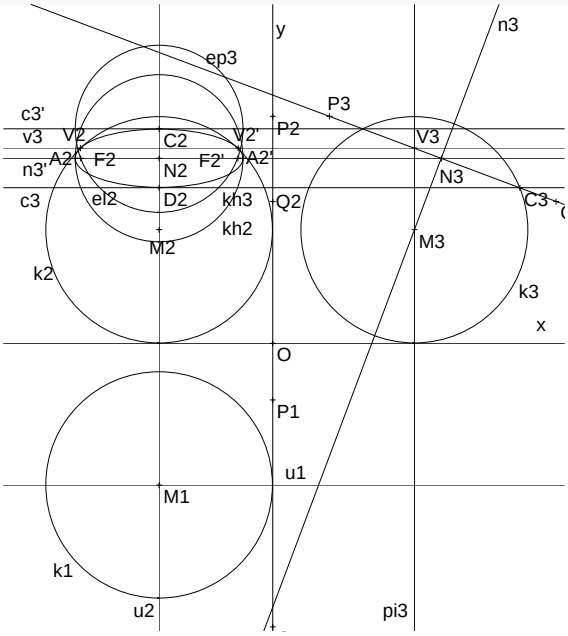


Bild 3:

Diese unübersichtliche Zeichnung wird nun durch das Unsichtbarmachen der Hilfslinien verbessert. Nach dieser Kosmetik sieht die Zeichnung wie folgt aus:

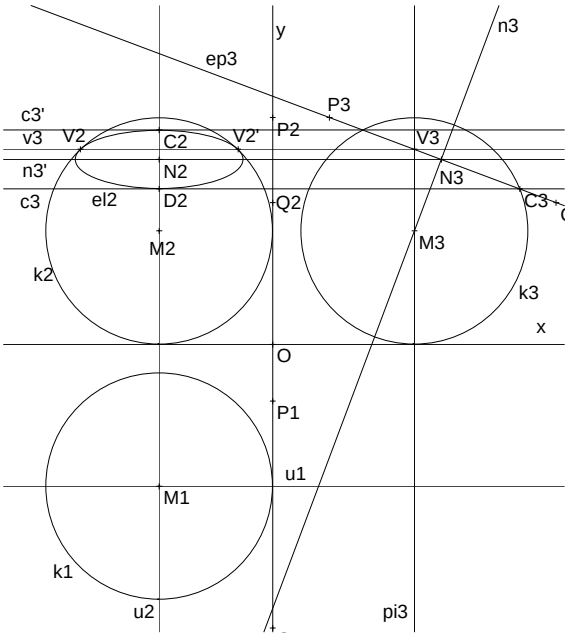


Bild 4:

Im Prinzip folgt die gleiche Konstruktion für der Grundriß

```
.wieder Hilfslinien und Schnittpunkte
o1=nor(A2,x)
o2=nor(V2,x)
o3=nor(V2',x)
o4=nor(A2',x)
V1=pkt(o2,u1)
V1'=pkt(o3,u1)

.Die Punkte A1 und B1 der Grundrißellipse
H3=pkt(n3',y)
h3=|H3,N3|

.Ermittlung des Mittelpunktes der Ellipse
kh4=krs((my,0),h3)
N1=pkt(kh4,u2)

.die große Halbachse
u3=nor(N1,y)
A1=pkt(u3,o1)
B1=pkt(u3,o4)

.Konstruktion für C1
H4=pkt(c3',y)
h4=|H4,C3'|

.die kleine Halbachse
kh5=krs((my,0),h4)
C1=pkt(kh5,u2)

.Bestimmung der Brennpunkte
kh6=krs(C1,r')
F1=pkt(kh6,u3)
.die Ellipse
el1=ell(F1,F1',r')
```

Und nun wieder das Ergebnis der Arbeit ohne Kosmetik

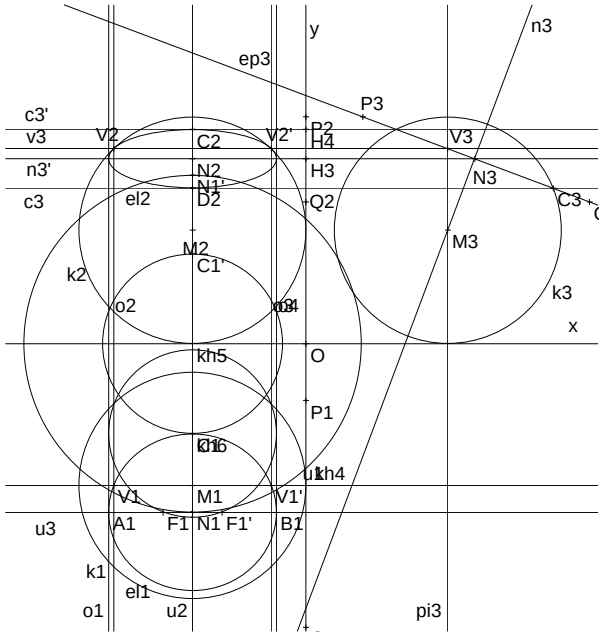


Bild 5:

Durch Weglassen der Hilfskonstruktionen und etwas Kosmetik in der Anordnung der Bezeichnungen für die Grafikobjekte sieht die Zeichnung wieder feiner aus.

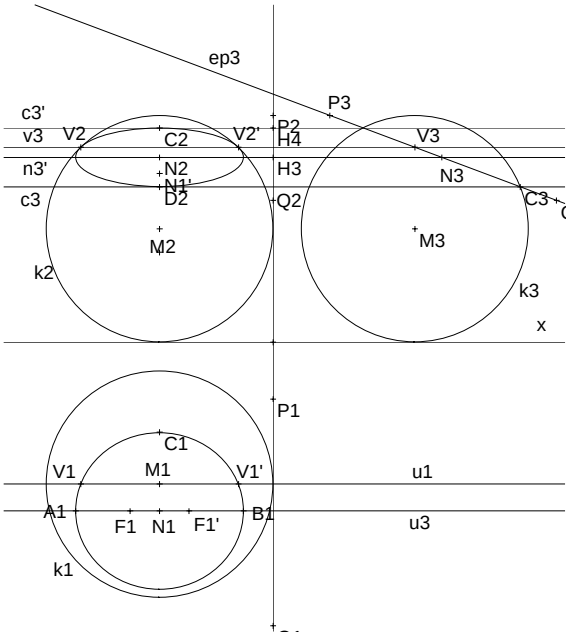


Bild 6:

Und nun nur mehr die „schönen“ Linien:

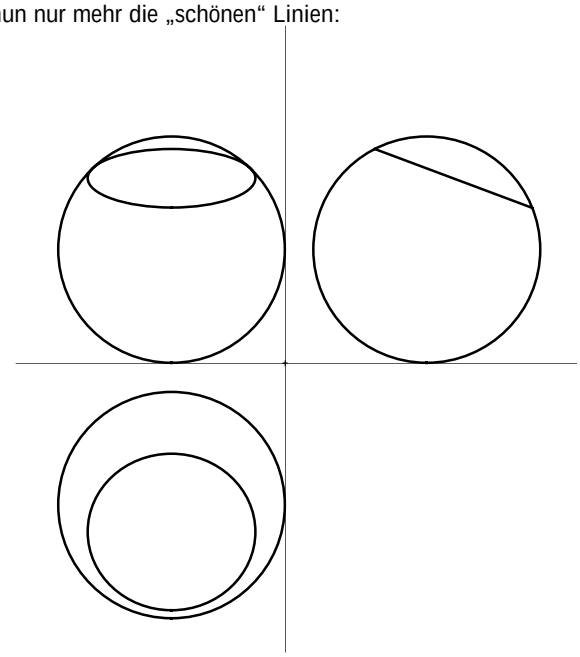


Bild 7:

Jetzt der Teil, der früher in dieser Art gar nicht denkbar war. Durch einfaches Verändern der Punkte in der Angabe wird automatisch die gesamte Konstruktion neu gezeichnet.

```
mx=5
my=-4
mz=4
r=4

px=3
py=4
pz=8
qx=0
qy=4
qz=0
```

liefert

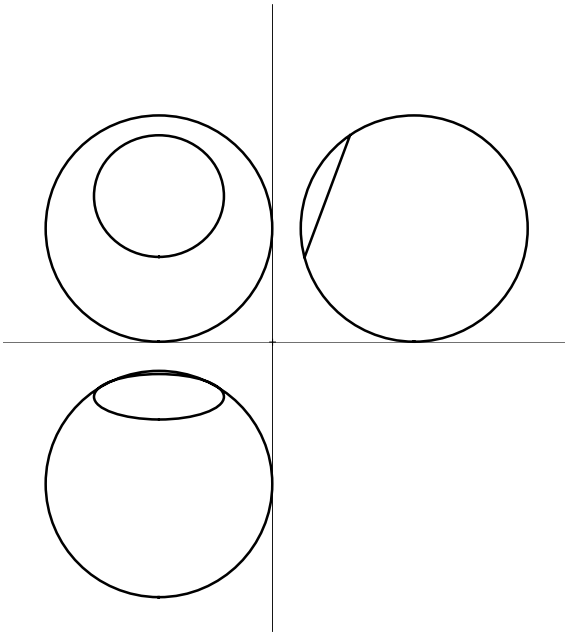


Bild 8:

Bemerkungen

Die Einschränkung auf nur drei Zeichen für die Bennenung eines Grafikobjekts, ist für größere Konstruktionen sicher eine Erschwernis. (Das Haushalten mit bereits vergebenen Namen wird zwar durch Winkon erleichtert, da es doppelt vergebene Namen durch Nichtdurchführen des Befehles signalisiert; doch ist bald eine zu Beginn geplante Systematik der verwendeten Symbole verloren.)

Die Verwendung der eingebauten Hilfe ist in der Regel ausreichend, die mitgelieferte Beschreibung muß nur in seltenen Fällen gelesen werden. Das Einbinden der Grafiken in einem Winwordtext geht nach dem Lesen der mitgelieferten Dokumentation ohne Probleme.

Was mir noch abgeht, ist das Prinzip der Layer, wie es aus den technischen Zeichenprogrammen, wie ACAD oder PCAD üblich ist. Hier muß jedes einzelne Grafikobjekt einzeln ein- beziehungsweise ausgeschaltet werden, was bei größeren Konstruktionen lästig werden kann.

Schlußbemerkung

Anders als die technischen Zeichenprogramme sind hier die traditionellen Konstruktionsschritte (das sind jene, die mit Zirkel und Lineal ausgeführt werden) im Vordergrund. Meiner Ansicht nach ist dieses Programm daher sehr gut geeignet, im Gegenstand Darstellende Geometrie eingesetzt zu werden, da dort diese Tradition (und damit auch das Wissen um die Konstruktionstechniken!) gepflegt wird. ☐

Ganz deutlich wurde mir wieder eines vor Augen geführt: "Gutgemeinte" Einschränkungen (um vielleicht gar den Anwender zu leiten) soll man beim Programmdesign möglichst lassen. Der Grund dafür, die Namen von Graphikobjekten auf drei Zeichen zu beschränken, war nur der, die Graphik nicht durch lange Bezeichner zu verunzieren - eine Entscheidung, die ganz sicher dem Anwender vorbehalten bleiben sollte.

Zwei Lehren ziehe ich daraus:

- *Gut gemeint ist sicher schlimm.*
- *Du sollst Deinen Anwender nicht bevormunden.*

Die Einschränkung ist im nächsten Update sicher gefallen.

Die von Herrn Kugler angesprochenen Layer halte ich für sehr interessant und bedanke mich für die Anregung!

Robert P. Michelic

Analoge Integration

Othmar Fischer, Markus Seidl

Die Laborübung „Analoge Integration“ wird am TGM, Höhere Abteilung für Elektronik, durchgeführt und vermittelt als Lehrinhalt das Zusammenwirken einer Analog-Schaltung (Integrator) mit einem Personalcomputer, verbunden mit einer Einführung in die Datenanalyse, mit der die erfaßten Meßdaten einer linearen Regression (Kurvenanpassung) unterzogen und der prozentuelle Fehler zwischen den gemessenen und den errechneten Werten bestimmt und grafisch dargestellt wird, um die Genauigkeit des Analog-Integrators zu erfassen. Dazu wird eine Gleichspannung während eines festgesetzten Zeitintervalls integriert; als Resultat entsteht eine zeitproportionale Ausgangsspannung. Diese Vorgangsweise erlaubt es auch, die Qualität eines Analogwert-Speichers zu überprüfen.

Ein Analog-Integrator besteht aus einem dazu geeigneten Operationsverstärker in Inverter-Grundschaltung, der mit einem RC-Glied beschaltet ist, ergänzt mit Einrichtungen zum Setzen der Anfangsbedingung und zum Festlegen des Integrations-Zeitintervalls. Unter der Voraussetzung eines idealen Operationsverstärkers gilt für den Analog-Integrator die Übertragungsgleichung

$$u_q(t) = U_0 - \frac{1}{R.C}\int_0^t u_i(t)dt \quad \text{für}$$

$$u_i(t) = U_i \quad \text{folgt} \quad u_q(t) = U_0 - U_i.t$$

Die Ausgangsspannung $u_q(t)$ des Analog-Integrators entspricht dem Integral der Eingangsspannung $u_i(t)$ in den Grenzen von 0 bis t, wobei der Integrations-Zeitkonstanten $\tau = R.C$ die Bedeutung einer Skalierung zukommt, vermehrt um die Anfangsbedingung U_0; die negativen Vorzeichen entstehen durch die Inverter-Grundschaltung.

Hardware

Ein Integrator arbeitet grundsätzlich statisch instabil und dynamisch stabil; daher stellt die Integration einer Gleichspannung technisch das schwierigste, mathematisch aber das einfachste Problem dar, denn eine Konstante („Gleichspannung“) integriert, liefert als Resultat eine lineare Funktion („zeitproportionale Spannung“).

Ein Analog-Integrator benötigt grundsätzlich die drei Steuerbefehle

1. SET (Setzen der Anfangsbedingung),
2. HOLD (Speichern des Analogwertes) und
3. RUN (Integration).

Der Personalcomputer gibt diese Steuerbefehle als TTL-kompatible Signale programmgesteuert ab, übernimmt die Ausgangsspannung des Analog-Integrators, digitalisiert sie und führt diese Werte der Verarbeitung durch die Software zu. Das ***Bild 1*** veranschaulicht den bei dieser Laborübung verwendeten grundsätzlichen Versuchsaufbau.

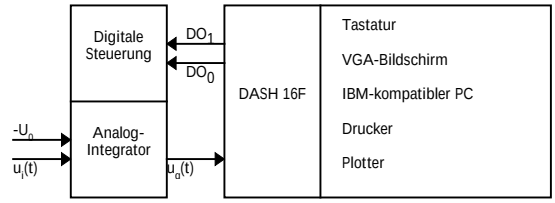


Bild 1: **Grundsätzlicher Aufbau der Laborübung „Analoge Integration“**

Software

Im Hinblick auf schulische Laborübungen wurde diese Software nur für die Integration einer Gleichspannung ausgelegt und beinhaltet innerhalb der gewählten Meßzeit drei Betriebsarten:

1. „manuell“: Steuerung von SET, HOLD und RUN über Funktionstasten
2. „periodisch“: abwechselnd eine Sekunde lang RUN und dann HOLD
3. „kontinuierlich“: Integration (RUN) während der gesamten Meßzeit

Die gewählte Betriebsart und der aktive Steuerbefehl werden am Bildschirm festgehalten; wenn ein Steuerbefehl wirksam ist, wird die Ausgangsspannung des Analog-Integrators numerisch ausgegeben. Nach erfolgter Messung läßt sich in allen Betriebsarten ein Cursor über den gewählten Zeitbereich bewegen; dabei werden die der Cursor-Position entsprechenden Zahlenwerte am Bildschirm dargestellt.

Das Programm setzt einen IBM-kompatiblen AT-Personalcomputer mit DOS 3.3, einen VGA-Bildschirm und die Meßwerterfassungskarte DASH 16F (MetraByte, Keithley), eingestellt auf 8 Differenzeingänge für eine bipolare Eingangsspannung von ±10 V, voraus. Nach dem Programmstart erscheint das Programmlogo mit dem Hauptmenue, aus dem eine Hilfe aufgerufen, die Meßumgebung konfiguriert, die Meßzeit eingestellt und die manuelle, periodische oder kontinuierliche Steuerung des Analog-Integrators gewählt werden kann. Das Hauptmenue und die Untermenues „manuell“, „periodisch“ und „kontinuierlich“ mit den Folgemenues für „Messen“ zeigt das ***Bild 2***.

Hauptmenue

F1	F2	F3	F4	F5	F6	F7	ESC
Hilfe	Konfig.	Manuell	period.	kontin.			Quit

Untermenue „manuell“

F1	F2	F3	F4	F5	F6	F7	ESC
Messen	Speich.	Laden		Cursor	Text	Drucken	Quit
SET	HOLD	RUN					Quit

Untermenue „periodisch“

F1	F2	F3	F4	F5	F6	F7	ESC
Messen	Speich.	Laden		Cursor	Text	Drucken	Quit
SET	START						Quit

Untermenue „kontinuierlich“

F1	F2	F3	F4	F5	F6	F7	ESC
Messen	Speich.	Laden	Regres.	Cursor	Text	Drucken	Quit
SET	START						Quit

Bild 2: **Menuestruktur der Software „Analoge Integration“**

In der Konfiguration wird festgelegt, welchen Analog-Eingangskanal (CH0 bis CH7) der Integrator-Ausgang belegt und welche Digital-Ausgänge (OP0 bis OP3) die Steuersignale für den Analog-Integrator liefern; außerdem kann eine Meßzeit von 10 s, 20 s, 30 oder 40 s gewählt werden, innerhalb der der Personalcomputer den Analog-Eingang periodisch 500mal abtastet.

Die Betriebsart „manuell“ gestattet es, die Anfangsbedingung zu setzen und den Analog-Integrator nach dem Start der Messung während der gewählten Meßzeit mit den zugeordneten Funktionstasten (SET, HOLD und RUN) zu steuern.

In der Betriebsart „periodisch“ wird der Analog-Integrator nach dem Setzen der Anfangsbedingung und folgendem Start abwechselnd während einer Sekunde auf RUN und dann während einer Sekunde auf HOLD geschaltet, bis die konfigurierte Meßzeit abgelaufen ist.

Bei der Betriebsart „kontinuierlich“ läuft der Analog-Integrator nach dem Setzen der Anfangsbedingung und dem Start während der festgelegten Meßzeit auf RUN; anschließend kann die lineare Regression (mathematisch: y = a + b.x) - die ermittelte Übertragungsfunktion wird angegeben - mit der Berechnung des prozentuellen Fehlers durchgeführt werden, wobei die Ergebnisse grafisch aufscheinen. Überschreitet die Ausgangsspannung des Analog-Integrators den Arbeitsbereich des Analog-Digital-Konverters (9,995 V), so wird die laufende Messung abgebrochen und die bis dahin erfaßten Daten ausgewertet.

Die aufgenommenen Dateien können mit einem erklärenden Text bis zu 20 Bildschirmzeilen versehen, gespeichert und geladen sowie vierfarbig über einen 24-Nadel-Farb-Matrixdrucker (Fujitsu DL 2600, NEC P60) oder Plotter (HP 7475 A) ausgegeben werden.

Messung

Beispielsweise vermittelt das ***Bild 3*** einen Eindruck von der Qualität eines Analogwert-Speichers. Dazu wurde in der Betriebsart „kontinuierlich“ als Anfangsbedingung die Ausgangsspannung auf den Wert -5 V gesetzt, die Eingangsspannung 0 V mit der Zeitkonstante 1 s über eine Zeit von 10 s integriert, anschließend eine lineare Regression durchgeführt und der prozentuelle Fehler berechnet.

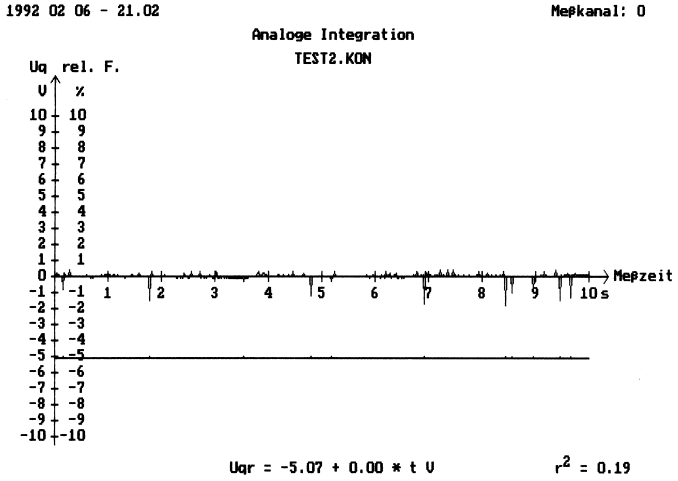


Bild 3: **Integration der Eingangsspannung 0 V mit der Anfangsbedingung U_0 = -5 V; prozentueller Fehler nach Regression**

Als weiteres Beispiel zeigt das ***Bild 4*** die analoge Integration einer Gleichspannung mit der Zeitkonstanten $\tau = 1$ s über eine Zeit von 10 Sekunden bei der Anfangsbedingung U_0 = -5 V in der Betriebsart „kontinuierlich“. Aus den aufgenommenen Meßwerten wurde die Übertragungsgleichung durch lineare Regression bestimmt; dann wurde der prozentuelle Fehler zwischen den gemessenen und den berechneten Werten ermittelt und dieser grafisch dargestellt.

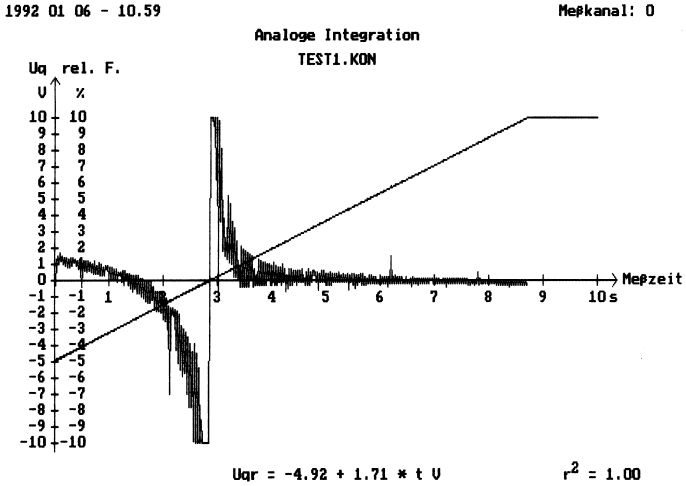


Bild 4: **Integration einer Gleichspannung, lineare Regression und Fehlerberechnung**

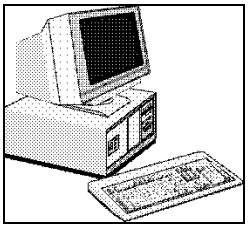
□

Unverzichtbar bei Protokollierungen

Die neue low-cost Meßdatenerfassung ADC-11 von Megalab ist ein komfortabler Bedien- und Automationstool am PC unter Windows:

11 Analogkanäle werden bei direktem Anschluß an die parallele Schnittstelle mit ATEWIN-Software mit hoher Samplingrate betrieben. Der Eingangsspannungsbereich ist mit 2,5 V und 10 Bit Auflösung angegeben.

Der Baustein in Größe einer Streichholzschachtel ist hervorragend für die Meßdatenerfassung mit Notebooks, z.B. als Datenlogger, für zahlreiche Routineaufgaben konzipiert, da keine zusätzliche Stromversorgung benötigt wird.



MESSEN AUTOMATISIEREN und PROTOKOLLIEREN

einfach und schnell
mit dem PC und LAPTOP

INDUSTRIEMESSTECHNIK
1170 Wien, Clemens-Hofbauer-Platz 10
Telefon und Fax 0222-486 51 25

KLEIN, aber PICfein

Microchip's PIC-Mikrocontroller

Robert SCHWAGER PCN-DSK-520:\pic

Der Mikrocontroller von Microchip - PIC - ist eine 'neue' Art von „Mikroprozessörchen“ mit wenigen Befehlen (, was eine kurze Einarbeitszeit ermöglichen sollte,) mit integrierten Speichereinheiten für Programm und Daten, wobei die ‘breiten’ Daten-, Adreß- und Steuerbusleitungen auf die äußere Prozessorperipherie entfallen. Die integrierte Peripherie des PICs ist ideal für Kleinsteuerungen, Überwachungen von Prozessen, Motorüberwachungen und PAL-Ersatz (Emulation) in RISC-Architektur.

Der Name PIC soll von *Periphery Intergrated Controller* abgeleitet sein.

PIC16/17-Mikrocontroller vom Hersteller Microchip ist eine Kombination von „Hochleistungsprozessor“, niedrigem Preis und kleinem Gehäuse, und bietet ein gutes Preis-Leistungsverhältnis für Industrieanwendungen. Über 200 Millionen PICs werden in kostengünstigen Konsumgütern, Computerperipherien, Büroautomatisations-, Sicherheits- und Telekommunikationsanwendungen verwendet.

Durch die von Microchip mitangebotene Entwicklungsumgebung ist die kostengünstige Softwareentwicklung und Softwareprüfung auch für Hobbyelektroniker möglich.

Überblick

Die folgende Überblickstabelle zeigt, welche PIC-Prozessoren mit welcher integrierter Peripherie- und Speicheroption erhältlich sind.

		Clock	Memory					Peripherals								Features‡		
		Maximum Frequency of Operation (MHz)	Program Memory: EPROM	Program Memory: ROM	Program Memory: EEPROM	Data Memory (Bytes)	Data EEPROM (Bytes)	Timer Module(s) †	Capture/Compare/PWM Module	Serial Port (s) (SPI/I²C, SCI)	Parallel Slave Port	Analog to Digital Converter (8-bit)	External Interrupts	Interrupt Sources	I/O	Voltage Range (Volts)	Number of Instructions	Packages
High-end	PIC17C42§	25	2K	—	—	128	—	TMR0, TMR1, TMR2, TMR3	—	SCI	—	—	Yes	11	33	4.5 - 5.5	56	40-pin DIP, 44-pin PLCC 44 pin QFP
Mid-range	PIC16C84	20	2K	—	—	128	—	TMR0, TMR0, TMR2	1	SPI/I²C	Yes	—	Yes	8	33	2.5 - 6.0	35	40-pin DIP, 44-pin PLCC 44 pin QFP
	PIC16C71	16	1K	—	—	36	—	TMR0	—	—	—	4 ch	Yes	4	13	3.0 - 6.0	35	18-pin DIP, 18-pin SOIC 44 pin QFP
	PIC16C74	20	4K	—	—	192	—	TMR0, TMR1, TMR2	2	SPI/I²C, SCI	Yes	8 ch	Yes	12	33	2.5 - 6.0	35	40-pin DIP, 44-pin PLCC 44 pin QFP
	PIC16C84	10	—	—	1K	36	64	TMR0	—	—	—	—	Yes	4	13	2.0 - 6.0	35	18 pin DIP, 18 pin SOIC 20-pin SSOP
Base Line	PIC16C54	20	512	—	—	25	—	RTCC	—	—	—	—	—	0	12	2.5 - 6.25	33	18-pin DIP, 18-pin SOIC 20-pin SSOP
	PIC16C54A	20	512	—	—	25	—	RTCC	—	—	—	—	—	0	12	2.5 - 6.25	33	18-pin DIP, 18-pin SOIC 20-pin SSOP
	PIC16CR54	20	—	512	—	25	—	RTCC	—	—	—	—	—	0	12	2.5 - 6.25	33	18-pin DIP, 18-pin SOIC 20-pin SSOP
	PIC16C55	20	1K	—	—	25	—	RTCC	—	—	—	—	—	0	20	2.5 - 6.25	33	28-pin DIP, 28-pin SOIC 28-pin SSOP
	PIC16C56	20	1K	—	—	25	—	RTCC	—	—	—	—	—	0	12	2.5 - 6.25	33	18-pin DIP, 18-pin SOIC 20-pin SSOP
	PIC16C57	20	2K	—	—	72	—	RTCC	—	—	—	—	—	0	20	2.5 - 6.25	33	28-pin DIP, 28-pin SOIC 28-pin SSOP
	PIC16CR57A	20	—	2K	—	72	—	RTCC	—	—	—	—	—	0	20	2.5 - 6.25	33	28-pin DIP, 28-pin SOIC 28-pin SSOP
	PIC16C58A	20	2K	—	—	72	—	RTCC	—	—	—	—	—	0	12	2.5 - 6.25	33	18-pin DIP, 18-pin SOIC 20-pin SSOP

Tabelle 1: **Microchip - PIC - Family**

Erläuterung

TMRx	Timer
RTCC	Real Timer Clock Counter
SCI	Serial Communication Interface (USART)
SPI	Synchron Serial Port
I²C	Inter-IC-Bus

Natürlich kann jeder digitale I/O-Pin des PIC als Ein- bzw. Ausgabeport geschaltet werden (siehe Befehl TRIS).

Als Grundtype der PIC-Familie kann man den PIC16C5x ansehen. Daher wird in dieser Mikrocontrollerbeschreibung zunächst die PIC16C5x-Familie beschrieben, welcher als Grundlage für weitere Prozessortypen dienen soll.

Die Tabelle 1 ist in 3 Gruppen geteilt.
Die **‘Base-Line-Familiy’** ist ein guter „Grundprozessor“, welcher die grundlegendste und kostengünstige Peripherie-Einheit beinhaltet. Die Spannungsversorgung kann bis zu 2,5 Volt absinken, was ideal für batteriebetriebene Steuerungen ist. Den PIC16C5x kann man sozusagen als Grundbaustein für die zwei weiteren Einteilungen ansehen.

Die **‘Mid-Range-Family’** bietet eine größere Vielzahl an intern vorhandener Peripherie (serielle Schnittstelle, I²C-Bus, interner Analog-Digital-Converter, größerer Speicher und Timermodule und Interrupteingänge).

Die **‘High-End-Family’** übertrifft dieses noch mit mehr I/O-Pins und etwas mehr interner Peripherie.

Anmerkung: Im Originalaufsatz werden in 3 Abschnitten die Familien PIC16C5x, PIC16C71 und PIC16C84 besprochen. Der vorliegende Beitrag beschränkt sich auf die Familie PIC16C5x. Interessierte Leser finden sowohl die Beschreibungen (Datei PIC.DOC) als auch die Programme der beiden anderen Familien auf der Begleitdiskette. Der abschließende Teil „Entwicklungsumgebung“ ist wieder vollständig abgedruckt.

PIC16C5x

Aufbau

Der grundsätzliche Aufbau des Mikrocontrollers kann man mit einem Blockschaltbild am einfachsten erklären.

Wie man in ***Abbildung 1*** erkennt, ist das EPROM (auch als One Time Programable-Ausführung [d.h. nur einmal programmierbar] ausgestattet), welches das Programm beinhaltet, welches gleich im Baustein implementiert ist.

Als Register dient der Block mit der Beschriftung GENERAL PURPOSE REGISTER FILE, welches je nach Prozessortype verscheiden groß ist, die Anzahl der General Purpose Register File beträgt minimal 24 und bei manchen Typen maximal 256. Hingegen kann man das ‘Arbeitsregister’ oder WORK-Register - W - als den Akkumulator ansehen, die dazugehörenden obligaten Flagbits befinden sich im STATUS-Register. Der interne Daten-’Arbeits’-Bus beinhaltet 8-Bits, hingegen ist die Programmdatenbreite 12 Bits, und der Adressierungsraum benötigt maximal 11 Bits. Die Breite des Programmadressierungsraumes ist vom Prozessortyp abhängig. Einen Stackbereich von 2 Rücksprungadressen (nur bei der PIC16C5x-Familie) bis zu 16 (abhängig vom Prozessortyp) dienen für strukturierte Programmabläufe. Die Ein/Ausgabeports ein 4 Bit und ein bzw. zwei (nur bei den Prozessoren PIC16C55/C57) 8 Bit (bis zu 33 I/O-Pins - abhängig vom Prozessortyp), welche als Ein- und Ausgabeport dienen können, sind für direkte Steuerung an die Umwelt verantwortlich. Die ALU (Arithmetic Logic Unit) ist eine 8-Bit-breite Operationseinheit, wobei die zweite Operationsgröße fast immer das W-(Work)-Register ist. Das Ergebnis kann in das W-Register bzw. in ein General-Purpose-Register-File (allgemein gesagt in ein Register) abgelegt werden. Ein Real-Time-Clock-Counter bzw. ein Watch-Dog-Timer befindet sich ebenfalls standardmäßig in PICs. Die Oszillatorbeschaltung kann auf 3 Arten realisiert werden (Achtung: Abhängig von der Prozessorspezifikation) (1) RC-Glied, (2) TTL-Oszillator, (3) Quarz-Ozillator.

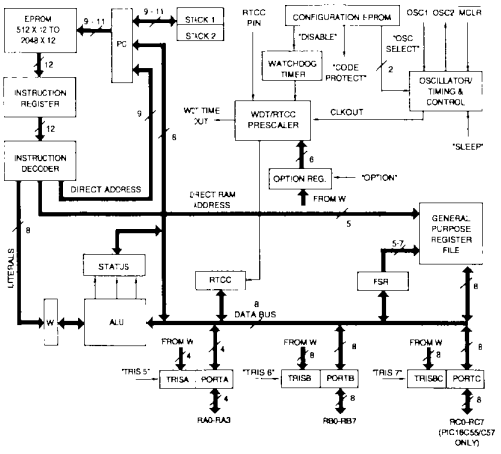


Abb. 1: **Blockschaltbild der PIC16C5x-Familie**

Der Aufbau der PIC-Prozessoren basiert daher auf Havard-Architektur, welche auf Registerfiles basiert mit separatem Bus und Speicher für Daten und Instruktionen und mit einer RISC-Prozessorarchitektur.

Die PIN-Belegung der PIC16C5x-Familie ist daher minimal und sehr übersichtlich, die Gehäuseform ist hauptsächlich in DIL in zwei Varianten erhältlich. Das CERDIP bietet ein Löschfenster für den Programmspeicher - EPROM - an, hingegen besitzt PDIP - die billigere Variante - kein Löschfester für den EPROM-Programmspeicher daher auch die OTP-EPROM-Ausführung. Natürlich ist der PIC auch in SOIC- und SSOP-Gehäuseform erhältlich.

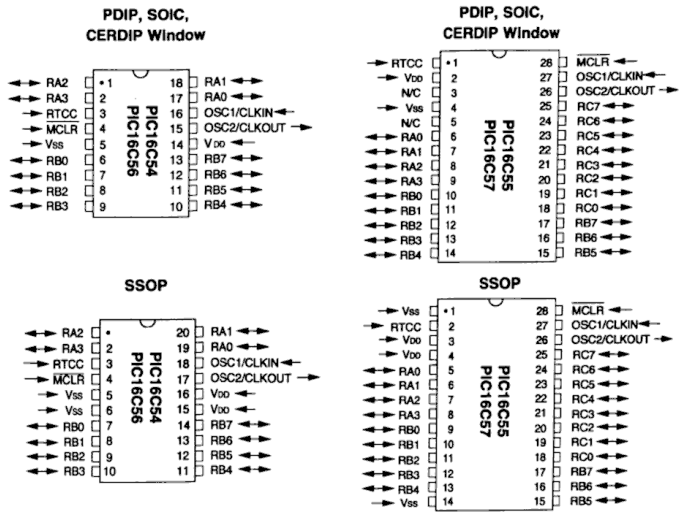


Abb. 2: **PIN-Belegung**

Buslos? - extern

Der PIC ist grob gesehen ein externer busloser Prozessor, d.h. daß der PIC keine Daten-, Adreß- und Steuerbusleitungen an die Umwelt ausgeführt hat, wie man es von der MSC51-Familie, Z80, etc. gewohnt ist (Ausnahme 87xx, o.ä.). Man kann jedoch mit einem gewissen Algorithmus der Ein-Ausgabe einen Bus 'emulieren' und somit einen seriellen oder parallelen Bus verwalten. Jedoch muß man sagen, wenn man mit einem PIC einen externen Bus verwalten möchte, ist dieses vielleicht eine Erweiterung der internen Peripherie-Möglichkeiten, aber die Arbeitsgeschwindigkeit nimmt bezüglich der 'Emulation' bei gleicher Taktrate ab.

Die Fileregister

Die Fileregister, auch Register genannt, sind direkt mit Befehlen ansteuerbar, Ausnahmen siehe im Text.

Die Register 00 bis 07 sind bei den meisten Typen gleich.

Überblick der Register mit Adresse und Bedeutung
00 indirect Addr. (8bits)
01 RTCC
02 PC (8bits schreib/lesbar, interne Verwaltung 12 Bits)
03 Status (PA2|PA1|PA0|/TO|Z|DC|C)
04 FSR
05 Port A
06 Port B
07 Port C nur bei PIC16C54/56
08 .. **0F** general purpose register files (8bits)
10 .. **1F** gen. purpose Reg. bei allen Typen (Seite 0)
30 .. **3F** gen. purpose Reg. nur bei PIC16C57 (Seite 1)
50 .. **5F** gen. purpose Reg. nur bei PIC16C57 (Seite 2)
70 .. **7F** gen. purpose Reg. nur bei PIC16C57 (Seite 3)

Indirekt Data Adressing (INDF)
INDF [00h]=(7) d7 d6 d5 d4 d3 d2 d1 d0 (0)
INDF ist kein physikalisch implementiertes Register. Es ist ideal als indirekte adressierter Zeiger. z.B.: ADD WF IND,W

Real Time Clock/Counter Register (RTCC)
RTCC [01h]=(7) d7 d6 d5 d4 d3 d2 d1 d0 (0)
Dieses Register kann wie ein gewöhnliches Register durch das Programm geladen und ausgelesen werden. Mit dem Setzen der OPTION-Registers kann es als Zähler mit externem Eingang - RTCC - oder mit dem Viertel des Oszillatortaktes angesteuert werden. (siehe Option-Register und Prescaler)

Program-Counter (PC)
PC [02h]=(7) d7 d6 d5 d4 d3 d2 d1 d0 (0)
Der PC generiert die Adresse für den maximal 2048x12-On-Chip-Eprom. Der PC wird nach einem RESET auf die höchstwertige Adresse gesetzt! Daher braucht man das eigentliche Programm nicht von der Adresse NULL starten, sondern von jeder beliebigen Adresse jedoch innerhalb der 9-Bit-Adressierung (nur bei PIC16C56/57, da der PIC16C54/55 nur 9-Bit PC hat.) also in der untersten Programmseite (siehe STATUS-Register). Diese Methode ist die einfachste.

Man kann natürlich den PC im Programm gezielt manipulieren, z.B. *ADDWF PC,1* , wobei PC ein EQU von 02 steht. Dieses kann verwenden um ein in Basic *ON..GOTO* bzw. *ON..GOSUB* bzw. in Borland Pascal ein eingeschränktes *CASE..OF* zu realisieren.

Das STATUS-Wort-Register
STATUS [3h]= (7) PA2 | PA1 | PA0 | /TO | /PD | Z | DC | C[RS1] (0)
Dieses Register beinhaltet arithmetische Statuse der ALU, Reset-Status und die Seitenadressierung des Programmspeichers.

Wird der Befehl *CLRF 02h* (= Löschen des Statusregisters) durchgeführt, so besitzt das Status-Register den Wert **000UU100**, wobei U als ungeändert steht.

C*: CARRY|/BORROW Bit & DIGIT CARRY|BORROW-Bit*
Das CARRY-Bit ist ein Übertrag bei Additionsoperationen (ADDWF) und ein negiertes 'Ausleih'-Bit bei Subtraktionsoperationen (SUBWF). Es wird für die Rotierbefehle RRF und RLF als weiteres Bit verwendet.

```
; Programm: PRG1.ASM

   LIST P=16C54

   ORG 0h
Start MOVLW 0FFh ; W:=0FFh
   MOVWF 8       ; GPFR-8 := W

   MOVLW 01h     ; W:=1
                 ; W  F8  Z DC C
                 ; 01 FF  0 0  0
   ADDWF 8,1     ; F8:=F8+W
   MOVLW 01h     ; 01 00  1 1  1
   ADDWF 8,1     ; F8:=F8+W
   MOVLW 01h     ; 01 01  0 0  0
   SUBWF 8,1     ; F8:=F8-W
   MOVLW 01h     ; 01 00  1 1  1
   SUBWF 8,1     ; F8:=F8-W
   NOP           ; 01 FF  0 0  0
   ORG 1FF
   GOTO Start
   END
```

Programmlisting 1: **Beispiel für STATUS-Register-Optionen**

DC*: DIGIT CARRY|/BORROW Bit*
Das Digit-Carry-Bit operiert auf die gleiche Weise wie das CARRY-Bit, jedoch nur zwischen den höheren und unteren Nibbles.

Z: *ZERO Bit*
Das Zero-Bit oder Null-Bit signalisiert, ob das aktuelle Ergebnis nach einer Operation gleich Null ist.

/PD*: POWER DOWN Bit* und ***/TO****: TIME OUT Bit*
Das /TO- und /PD-Bit kann als Reset-Überprüfung dienen.

Ereignis	/TO	/PD	Bemerkung
Versogungsspg. einschalten	1	1	
WDT Timeout	0	X	kein Effekt an /PD
SLEEP Befehl	1	0	
CLRWDT Befehl	1	1	

Tabelle 2: **/TO & /PD Ereignisse**

Nach einem Reset sind diese 2 Bits des Status-Register wie folgt gesetzt:

/TO	/PD	RESET-Ursache
0	0	WDT 'Aufgeweckt' vom SLEEP-Befehl
0	1	WDT-Time-Out (nicht durch SLEEP verursacht)
1	0	/MCLR 'Aufgeweckt' von SLEEP-Befehl
1	1	Versorgung eingeschaltet
X	X	=LOW-Pulse an /MCLR-Eingang

Tabelle 3: **/PD & /TO nach Reset**

File Reset Register *PA0/1*

PA0, PA1 PIC16C54/55 2 generelle Lese/Schreibbits
PIC16C56 PA0=0..Page 0 (000-1FF)
PA0=1..Page 1 (200-3FF)
PA1 .. generelles Lese/Schreibbit (siehe PA2)
PIC16C57 00 = Page 0 (000-1FF)
01 = Page 1 (200-3FF)
10 = Page 2 (400-5FF)
11 = Page 3 (600-7FF)

PA2..generelles Lese/SchreibBit, reserviert für zukünftige Anwendungen, d.h. der noch unbenutzte Bit kann als setz- und lesbarer Bit für diverse Programmaufgaben genutzt werden.

Generell kann man sagen, daß das STATUS-Byte die wichtigsten Flags beinhaltet.

File Select Register (FSR)
FSR [04h] = (7) d7|d6|d5|d4|d3|d2|d1|d0 (0)
Bei den PIC16C54/C55/C56 sind die Bits 0 bis 4 die 'Adresse' der Register, also 32 verfügbare Registeransteuerungen. Bits 5 bis 7 sind nur lesbar und beinhalten eine EINs. Wird das FSR-Register nicht für die indirekte Adressierung benötigt, so kann es als 5-Bit-langes generelles Lese/Schreibe-Register verwendet werden.

Nur beim PIC16C57 sind die Bits 5 und 6 als Auswahl der Seiten für die Datenbankseite zuständig, jedoch die unteren 16 Register sind auf allen Seiten der Bank physikalisch ident. Bit 7 des FSR ist nur lesbar und beinhaltet eine EINs.

EIN- und AUS-Gabeports PortA, PortB & PortC
PortA [05h] = (7) X|X|X|X|PA3|PA2|PA1|PA0 (0)
PortB [06h] = (7) PB7|PB6|PB5|PB4|PB3|PB2|PB1|PB0 (0)
PortC [07h] = (7) PC7|PC6|PC5|PC4|PC3|PC2|PC1|PC0 (0)
Die Ein- und Ausgaben des Prozessors an die Umgebung werden mittels Registern durchgeführt. Diese I/O- bzw. E/A-Register werden durch die MOV- bzw. Transportbefehle und registerarbeitenden Befehle (jene welche die E/A-Register manipulieren) angesprochen (z.B. MOVWF PortB, W; ANDWF 0Fh,f).

Bevor man jedoch Ein- oder Ausgaben vom Programm durchführen lassen kann, muß man dem Ein/Ausgabe-Interface mitteilen, ob es sich um einen Eingabe- oder Ausgabe-Pin (individuell) handelt. Hier wird im W-Register an derselbigen Position eine **0 für AUS**gabe und eine **1 für EIN**gabe positioniert. Dann mit dem Befehl TRIS und dessen Port-Adresse (5 bis 7) dem Interface die Richtungen mitgeteilen.

(zB. MOVLW 0h TRIS 6h)

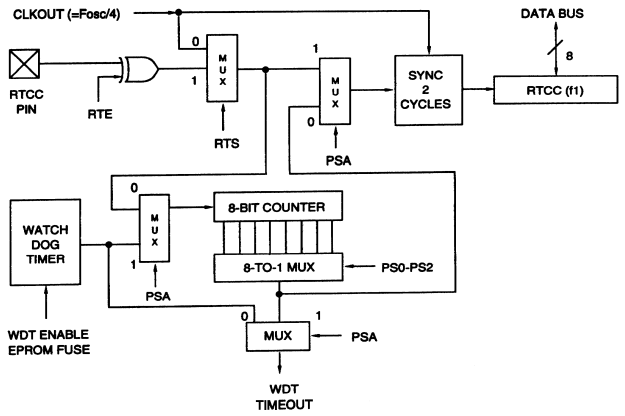
Port A: Das Port A besteht auf nur 4 Bits des unteren Bytes (bei PIC16C5x).
Port B: Ein-8-Bit Register
Port C: Ein 8-Bit-Register nur bei PIC16C55/C57
Bei PIC16C54/C56 als General Purpose Register verwendbar.

OPTION Register
OPTION = (7) - | - | RTS | RTE | PSA | PS2 |PS1 |PS0 (0)
Mit dem OPTION-Register kann man den Real-Time-Clock-Counter konfigurieren.

RTS - RTcc signal Source
Echtzeit-Zähl-Signal-Quelle
0 .. Zähleingang = CLKOUT (¼ f_{OSZ})
1 .. Zähleingang = RTCC-Eingang

RTE - RTcc Signal Edge
Echtzeit-Zähler-Signal-Flankenerkennung
0 .. Inkrement by LOW-HIGH-Flanke
1 .. Inkrement by HIGH-LOW-Flanke

PSA - PreScale Assigment bit
Vorteilerzuweisung
0 .. Anweisung von RTCC
1 .. Anweisung von WDT
PS2,PS1,PS0 - PreScal Value
Vorzählereinstellung
RTCC-Rate = $2^{PS[2:0]+1}$
WDT-Rate = $2^{PS[2:0]}$



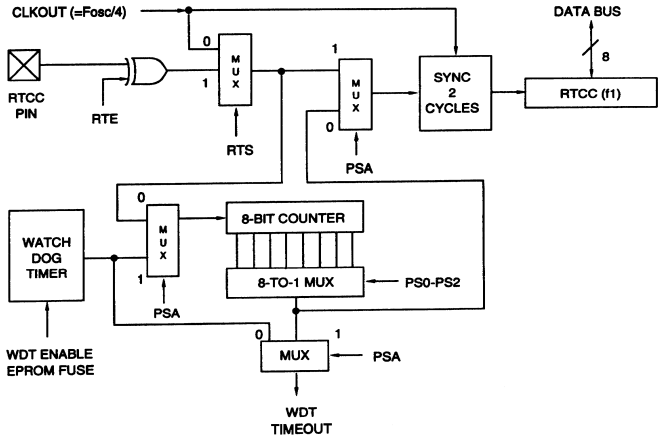


Abb. 3: **Blockdiagramm v. RTCC/WDT Prescaler**

In dem folgenden Programmbeispiel wurde der RTCC-Teiler und die Länge des Programmes so gewählt, daß das Zählregister (im General Purpose Register File Nr. 8) und das RTCC-Register gleichen Inhalt vor dem Befehl 'NOP' haben.

```
; Programm: PRG2.ASM
LIST P=16C55
Start  ORG 0
   CLRF 8    ; Lösche GPRF 8h
   CLRF 1    ; Lösche RTCC-Zähler
   MOVLW 1       ; Option: --000001
          ; RTS=0
          ; RTE=0
          ; PSA=0
          ; PS=001 => RTCC-Rate 1:4
   OPTION    ; W -> Option-Register
Marke  INCF 8,1  ; F8 := F8 +1 (1 Zyklus)
   NOP    ;             (1 Zyklus)
   GOTO Marke;             (2 Zyklen)
          ;             ===
          ; Programm:    4 Zyklen
   ORG 1FFh
   GOTO Start
   END
```

Programmlisting 2: Beispiel für RTCC

Die Befehle

(Kurzfassung)
Mit nur 33 Befehlen (beim PIC16C5x, sonst 35 bzw. 55) ist der PIC ein sogenannter „SMALL-COMMAND-PROCESSOR“. Jedoch mit so wenige Befehlen muß man manchmal einige Tricks anwenden, um gewisse Programmabläufe zu realisieren.

f Registerfile
d Zielregister (d=0 -> W (Workfile) d->1 selbige Registerfile)
d Bitstelle
k 8-Bit-Konstante
(x) Inhalt(e) von x
-> Zuordnung
Die Befehlslänge beträgt 1 Zeile.

Transportbefehle:

MOVF f,d
(f) -> d

Der Inhalt des Registers f wird in das in d angegebene Register geladen. Ist d=0, so ist das Zielregister W, bei d=1 ist es das ursprüngliche Register.

Statusbits: keine Zykluszeit: 1

MOVWF f
W -> f

Es werden die Daten, die im Register W stehen, in das Register f geladen.

Statusbits: keine Zykluszeit: 1

arithmetische Operationen:

ADDWF f,d
(W + f) -> d

Addiert den Inhalt von W mit dem Inhalt von Register f. Ist d gleich 0, so wird das Resultat in W und bei 1 in das ursprüngliche Register f gespeichert.

Statusbits: C, CD, Z Zykluszeit: 1

SUBWF f,d
(f - W) -> d

Subtrahiert den Inhalt von W von dem Inhalt von Register f. Ist d gleich 0, so wird das Resultat in W und bei 1 in das ursprüngliche Register f gespeichert.

Statusbits: C, CD, Z Zykluszeit: 1

DECF f,d
(f - 1) -> d

Dekrementieren von Register f. Ist d gleich 0, so wird das Resultat in W und bei 1 in das ursprüngliche Register f gespeichert.

Statusbit: Z Zykluszeit: 1

INCF f,d
(f + 1) -> d

Inkrementieren von Register f. Ist d gleich 0, so wird das Resultat in W und bei 1 in das ursprüngliche Register f gespeichert.

Statusbit: Z Zykluszeit: 1

logische Operationen:

ANDWF f,d
(W .AND. f) -> d

Logische UND-Verknüpfung zwischen den Inhalten von W und dem Register f. Ist d gleich 0, so wird das Resultat in W und bei 1 in das ursprüngliche Register f gespeichert.

Statusbits: Z Zykluszeit: 1

ANDLW k
(W .AND. k) -> W

Logische UND-Verknüpfung zwischen den Inhalten von W und der Konstanten k. Das Ergebnis wird in das W-Register gespeichert.

Statusbits: Z Zykluszeit: 1

IORWF f,d
(W .OR. f) -> d

Logische INKLUSIVE-ODER-Verknüpfung zwischen den Inhalten von W und dem Register f. Ist d gleich 0, so wird das Resultat in W und bei 1 in das ursprüngliche Register f gespeichert.

Statusbits: Z Zykluszeit: 1

IORLW k
(W .OR. k) -> W

Logische inklusive ODER-Verknüpfung zwischen den Inhalten von W und der Konstante k. Das Ergebnis wird in das W-Register gespeichert.

Statusbits: Z Zykluszeit: 1

XORWF f,d
(W .XOR. f) -> d

Logische EXKLUSIVE-ODER-Verknüpfung zwischen den Inhalten von W und dem Register f. Ist d gleich 0, so wird das Resultat in W, und bei 1 in das ursprüngliche Register f gespeichert.

Statusbits: Z Zykluszeit: 1

XORLW k
(W .XOR. k) -> W

Logische exklusive ODER-Verknüpfung zwischen den Inhalten von W und der Konstanten k. Das Ergebnis wird in das W-Register gespeichert.

Statusbits: Z Zykluszeit: 1

COMF f,d
/f -> d

Der Inhalt vom Register f wird negiert, d.h. es wird jedes einzelne Bit des f-Registers invertiert, und je nach Wertigkeit von d wird das Ergebnis in das W (d=0) bzw. in das Ursprungsregister f (d=1) gespeichert.

Statusbits: Z Zykluszeit: 1

Lösch- und Setzbefehle:

CLRF f
00h -> f

Löscht den Inhalt von Register f.

Statusbits: Z Zykluszeit: 1

CLRW
00h -> W

Löscht das W-Register.

Statusbits: Z Zykluszeit: 1

MOVLW k
k -> W

Die 8-bit-lange Konstante k wird in das W-Register geladen.

Statusbits: keine Zykluszeit: 1

BCF f,b
0 -> f(b)

Das b-te Bit (b=0..7) des Registers f wird gelöscht, d.h. auf 0 gesetzt.

Statusbits: keine Zykluszeit: 1

BSF f,b
1 -> f(b)

Das b-te Bit (b=0..7) des Registers f wird gesetzt, d.h. auf 1 gesetzt.

Statusbits: keine Zykluszeit: 1

Befehle mit Bedingungen:

BTFSC f,b
SKIP IF f(b)=0

Ist das b-te Bit (b=0..7) gleich 0, so wird der im Programm stehende nächste Befehl übersprungen.

Statusbits: keine Zykluszeit: 1 bzw. 2

BTFSS f,b
SKIP IF f(b)=1

Ist das b-te Bit (b=0..7) gleich 1, so wird der im Programm stehende nächste Befehl übersprungen.

Statusbits: keine Zykluszeit: 1 bzw. 2

DECFSZ f,d
(f - 1) -> d ; SKIP IF RESULT = 0

Der Inhalt des Registers f wird dekrementiert. Ist das Ergebnis der Dekrementierung bei der nächsten Abarbeitung gleich 0, so wird der nächste Befehl übersprungen.

Statusbits: keine Zykluszeit: 1 bzw. 2

INCFSZ f,d
(f + 1) -> d ; SKIP IF RESULT = 0

Der Inhalt des Registers f wird inkrementiert. Ist das Ergebnis der Dekrementierung bei der nächsten Abarbeitung gleich 0, so wird der nächste Befehl übersprungen.

Statusbits: keine Zykluszeit: 1 bzw. 2

Rotierbefehle

RLF f,d
f<n> -> d<n+1> {für n=0..6}, f<7> -> C, C -> d<0>

Der Inhalt des Registers f wird um ein Bit nach LINKS durch das CARRY verschoben. Ist d=0, so ist das Zielregister W, bei d=1 ist es das ursprüngliche Register f.

Statusbits: keine Zykluszeit: 1

RRF f,d
f<n> -> d<n-1> {für n=1..7}, f<0> -> C, C -> d<7>

Der Inhalt des Registers f wird um ein Bit nach RECHTS durch das CARRY verschoben. Ist d=0, so ist das Zielregister W, bei d=1 ist es das ursprüngliche Register f.

Statusbits: keine Zykluszeit: 1

Tauschbefehl:

SWAP f,d
f<0:3> -> d<4:7>, f<4:7> -> d<0:3>

Die untere und obere Nibbles des Registers werden getauscht. Das Endregister kann mit d=0 als W-Register oder mit d=1 als Ursprungsregister ausgewählt werden

Statusbits: keine Zykluszeit: 1

Sprungbefehle:

GOTO k
k -> PC<8:0>; PA2, PA1, PA0->PC<11:9>

Die unteren neun Bits werden von den direkt eingegeben Wert bestimmt. Die oberen drei Bits werden vom den Bits PA2, PA1 und PA0 des STATUS-Register-Bit bestimmt. Die Konstante k wird in den Programmzähler geladen und verursacht einen Sprung im Programm.

Statusbits: keine Zykluszeit: 2

CALL k
PC+1 -> TOS; k -> PC>7:0>; '0'->PC>8>;

PA2, PA1, PA0 -> PC<11:9>

Subroutinenaufruf. Als erstes die Rücksprungadresse (PC+1) wird in den Stack-Speicher gespeichert. Das Byte wird in den PC Bit (0 bis 7) geladen. Der achte Bit des PC wird gelöscht. PA<2:0> wird in den PC<11:9> geladen.

Statusbits: keine Zykluszeit: 2

RETLW k
k -> W; TOS -> PC

Das W-Register wird mit der 8-bit-langen Konstanten k geladen. Der Programm Counter wird mit dem oberen Stack-Inhalt, welcher die Return-Adresse beinhaltet, geladen.

Statusbits: keine Zykluszeit: 2

Steuerbefehle:

TRIS f
W -> TRIS register f

TRIS-Register f (f=5,6 oder 7) wird mit dem Inhalt des W-Registers geladen. Eine 0 bedeutet eine AUS-Gangsschaltung, eine 1 EIN-Gangsschaltung.

Statusbits: keine Zykluszeit: 1

CLRWDT
00h -> WDT; 0 -> WDT prescaler

CLRWDT-Befehl setzt den Watchdog-Timer zurück. Dieser Befehl setzt ebenfalls den Prescaler des WDT zurück. Die Statusbits /TO und /PD werden gesetzt.

Statusbits: 1 -> /TO; 1-> /PD Zykluszeit: 1

SLEEP
0 -> PD; 1 -> TO

Das POWER-DOWN-Status-Bit (/PD) wird gelöscht. TIME-OUT-Bit (/TO) wird gesetzt. Der Watchdog-Timer und dessen Prescaler werden gelöscht. Dieser Prozeß fällt in den SLEEP-Modus und schaltet den Oszillator aus.

Statusbits: /TO; /PD Zykluszeit: 1

OPTION
W -> OPTION

Der Inhalt des W-Registers wird in das OPTION-Register geladen.

Statusbits: keine Zykluszeit: 1

NOP
keine Operation

Statusbits: keine Zykluszeit: 1

Takt

Takt-Typen
Die PIC16CXX-Serie ist in vier Taktvarianten verfügbar.

Quarz-Oszillator
Die PIC16C5X-***XT***, -***HS*** oder -***LP*** benötigen einen Kristall oder Keramik-Schwingquarze

Die Typenbezeichnungserweiterung bedeuten:

XT = Standard (bis ca. 4MHz)
HS = Hochgeschwindig (bis ca. 20Mhz)
LP = Niedergeschwindigkeitsschwingquarz (bis ca. 32kHz)

Osc-Type	Quarz-Type	Frequenz	$C_{OSC1-GND}$	$C_{OSC2-GND}$
LP	Kristall	32kHz	15pF	15pF
XT	Kristall	100kHz	15-30pF	300-300pF
XT	Kristall	200kHz	15-30pF	100-200pF
XT	Kristall	455kHz	15-30pF	100-200pF
XT	Keramik	455kHz	150-330pF	150-300pF
XT	Kristall	1MHz	15-30pF	15-30pF
XT	Kristall	2MHz	15pF	15pF
XT	Keramik	2MHz	20-330pF	20-330pF
XT	Kristall	4MHz	15pF	15pF
XT	Keramik	4MHz	20-330pF	20-330pF
HS	Kristall	4MHz	15pF	15pF
HS	Kristall	8MHz	15pF	15pF
HS	Keramik	8MHz	20-200pF	20-200pF
HS	Kristall	20MHz	15pF	15pF

Tabelle 4: **Quarz-Oszillatoren**

RC-Oszillator
Für zeitunempfindliche Anwendungen ist der PIC16C5X mit der Typenerweiterungsbezeichnung ***RC*** ausreichend. Der Widerstand R_{ext}, welcher zwischen der Versorgungsleitung und dem Eingang OSC1 geschaltet ist, soll einen Wertebereich **über** 2,2kOhm haben, da sonst die

RC-Schaltung unstabil wird. Man sollte daher einen Widerstandswert zwischen 5kOhm und 100kOhm wählen. Der Kondensator C_{ext}, welcher zwischen OSC1 und Masse (0Volt) geschaltet ist, sollte aus Stabilitätsgründen mindestens 20pF haben.

C_{ext}	R_{ext}	F_{osz} (bei 5V, 25°C)	Toleranz
20pF	3,3kΩ	4,973MHz	±27%
20pF	5kΩ	3,82MHz	±21%
20pF	10kΩ	2,22MHz	±21%
20pF	100kΩ	262,15kHz	±31%
100pF	3,3kΩ	1,63MHz	±13%
100pF	5kΩ	1,19MHz	±13%
100pF	10kΩ	648,64kHz	±18%
100pF	100kΩ	71,56kHz	±25%
300pF	3,3kΩ	660,0kHz	±10%
300pF	5kΩ	484,1kHz	±14%
300pF	10kΩ	267,63kHz	±15%
300pF	100kΩ	29,44kHz	±19%

Tabelle 5: RC-Takt

Die durch vier geteilte Oszillatorfrequenz ist an dem Ausgang OSC2 bzw. CLKOUT verfügbar.

externe Takteingang
Die einfachste „Betaktung“ des PICs ist es einen externen Taktgenerator bzw. Taktschaltung an dem Eingang OSC1 zu legen, dieses ist bei den Typen HS, XT und LP möglich.

WATCHDOG TIMER (WDT)

Der Watchdog-Timer ist durch einen freilaufenden RC-Oszillator, welcher durch keine externe Komponente beeinflußbar ist, realisiert. Durch Ausführen eines SLEEP-Befehls wird die Taktoszillation gestoppt. Eine WDT-Auszeit generiert ein Bausteinresetzustand. Der WDT kann durch eine Null in einem speziellen Bit im EPROM bei der Programmierung ausgeschaltet werden.

WDT Periode
Der WDT hat eine Nenn-“Auszeit“-Periode von (typisch) 18ms, laut Datenbuch von 9ms bis 30ms (ohne Vorteiler). Wird eine längere „Auszeit“ benötigt, so kann ein maximales Verhältnis von 1:128 im OPTION-Register eingestellt werden. Dieses kann eine maximale Zeit typisch 2,304s jedoch zwischen 1,152s bis 3,84s bewirken.

Der CLRWDT- und der SLEEP-Befehle löschen den Watch-Dog-Timer (WDT) und den Zähler-Vorteiler (Prescaler). Ist der WDT zugewiesen, so wird eine Auszeit verhindert, und ein Reset wird ausgelöst.

Das /TO-Bit im STATUS-Register wird durch einen WDT-Auszeit gelöscht. Die WDT-Periode hat eine Zeitperiode von ca. 18ms (typischer Wert).

Ein WDT-Outtime bzw. eine WDT-Auszeit kann man als Interruptquelle bzw. Programmabbruch ansehen.

Abhängigkeit der Länge der WDT-Auszeit
Die WDT-Zeit ist abhängig von der Temperatur (-55..+125°C) und der angelegten Versorgungsspannung (V_{DD}=0..+7,5V).

WDT	V_{DD}					
T	3V	4V	4,5V	5V	5,5V	6V
-40°C	18,5ms	10ms	9,7ms	9ms	9,25ms	8,75ms
0°C	21,9ms	13,75ms	13,75ms	12,5ms	12,2ms	11,25ms
25°C	24,1ms	20,6ms	20ms	18ms	17,5ms	16,25ms
70°C	36ms	30,6ms	28,75ms	27ms	25ms	23,75ms
85°C	37ms	32,5ms	33,2ms	38ms	26,9ms	26,7ms

Tabelle 6: **WDT-Zeitperiode abhängig von Temperatur und Versorgungsspg.**

Bereitschaftsmodus (Power Down Mode -PDM [Sleep])

Der Bereitschaftsmodus (PDM) wird durch einen SLEEP-Befehl initialisiert.

Falls ein SLEEP-Befehl durchgeführt worden ist, wird der WDT gelöscht, jedoch bleibt erhalten. Das /PD-Bit im STATUS-Register wird gelöscht und das /TO-Register wird gesetzt. Der Oszillatortreiber wird ausgeschaltet, und die Pegel der Ein-Ausgabeports werden beibehalten, welche sie vor dem Ausführungsbefehl hatten (Low, High).

Die Ein/Ausgänge sollten in diesem Modus in den niedrigsten Stromverbrauch geschaltet werden. Ein hochohmiger Anschluß sollte mit einem Pull-Up- bzw. Pull-Down-Widerstand ausgestattet werden, um Schwingungen zu verhindern.

Der /MCLR-Eingang muß eine maximale Eingangsspannung von V_{IHMC} aufweisen.

Signal	Wertigkeit	Symbol	Min.	Max.	Einheit
$/M_{CLR}$	Low-Input	V_{ILMC}	V_{SS}	0,15 V_{DD}	V
	High-Input	V_{IHMC}	0,85 V_{DD}	V_{DD}	V

Tabelle 7: **Spannungspegel an /MCLR**

„Aufwecken“ (Wake Up)
Der Baustein kann durch einen Watchdog-Timer-Auszeit, wenn der WDT-Eprombit gesetzt wurde, oder durch einen externen Impuls an /MCLR-Eingang ausgelöst werden. In beiden Fällen geht der PIC16C5x in den RESET-Modus für eine Periode des Oscillator-Start-up-Timer - t_{OST} - bevor das ‘normale’ Programm abgearbeitet wird.

Das /PD-Bit im STATUS-Register, welches wieder gesetzt (/PD=1) wird, (nachdem es beim SLEEP-Befehl gelöscht (/PD=0) wurde), kann verwendet werden, um zu erkennen, daß der Prozessor eingeschaltet worden ist oder ‘aufgeweckt’ von einem Bereitschaftsmodus (PDM) worden ist.

Das /TO-Bit im STATUS-Register kann Auskunft geben, ob der Prozessor durch ein Signal an /MCLR (hier ist /TO=1) oder durch eine Watchdog-Auszeit (hier ist /TO=0) ‘aufgeweckt’ worden ist.

Oscillator Start-Up Timer (OST)

Taktoszillatoren, welche mit Kristall- oder Keramikquarze aufgebaut sind, benötigen eine gewisse Einschwingzeit (= Start-Up-Time) für eine stabile Taktschwingung. Ein im Chip eingebaute Einschaltverzögerung haltet den Baustein in einem RESET-Zustand für ca. 10ms nach dem erreichen eines Spannungswertes V_{IHMC} am /MCLR-Eingang. Eine Resetschaltung mit einem RC-Netzwerk ist daher in den meisten Fällen an dem Eingang /MCLR nicht notwendig. (siehe Power-On-Reset)

Der OST wird mit dem Watchdog-Auszeit getriggert, was für das verwenden des „Aufwecken“ vom SLEEP-Befehl wichtig ist.

Der OST ist für Niedrigfrequenzschwingquarze, die eine höhere Einschwingzeit als ca. 18ms besitzen nicht geeignet.

T_{OST} = 9ms(Minimum) bis 30ms(Maximum) typisch 18ms

automatischer Einschaltreset (Power-On-Reset [POR])
Der PIC16C5x besitzt auf dem Chip eine automatische Einschaltresetschaltung, welche einen Reset bei (den meisten) Einschaltsituationen der Versorgungsspannung auslöst, wenn man den /MCLR-Eingang auf V_{DD}-Potential (positive Versorgungsspannung) hält.

Der am Chip befindliche Einschaltresetmechanismus arbeitet garantiert, wenn die Anstiegsflanke der Versorgungsspannung kleiner als 0,05V/ms ist und von 0V beginnt. Der POR ist nicht für Niedrigfrequenzen mit höheren Einschwingzeit als etwa 18ms geeignet.

Aufbauend auf dieser Prozessorinformation kann nun auf die zwei weiteren Prozessoren der BASE-LINE (siehe Abb. 1) vorgehen.

Entwicklungsumgebung

Assembler

Der Assembler von Microchip (MPASM = Microchip Processor AsSeMbler) ist ein universaler Assembler für alle Microchip-Prozessoren. Die Handhabung des Assembler sowie die Programmierung der mnemonischen Quellprogramme ist sehr einfach. Neben den mnemonischen Befehlen stehen dem Programmierer noch (ca.) 39 Assemblerbefehle zur Verfügung.

Der Assembleraufruf

X:>*MPASM* [/<Option>[,/Option>...)) <file_name>

Einige Optionen kurz erklärt:

/h, **/?** zeigt die Optionserklärung an.
/l± erzeugt eine (+) / keine (-) Dokumentationsdatei (opt. +)
/e± erzeugt eine (+) / keine (-) Fehlerdatei (optional +)
/p{xx} Wählen der Prozessortype
PIC16C54, PIC16C55, PIC16C56, PIC16C57, PIC16C71, PIC16C84, PIC17C42, PIC16C58, PIC16C64
/x± erzeugen einer (+) / keiner (-) Cross-Referenz-Liste (optional -)

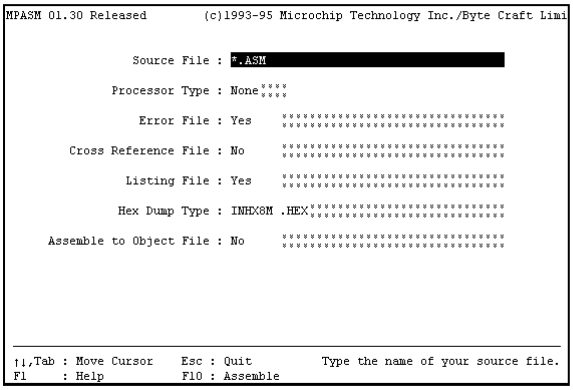


Abb.8: **Maske des MPASM**

Wenn man nur MPASM ohne Aufrufparameter eingibt, so kann in einer Maske der Dateiname und einige Optionen ausgewählt werden.

Die wichtigsten Dateitypen

Der Assembler erzeugt gewisse Dateien, die hier kurz erklärt werden sollen.

Die Quelldatei -**.ASM**
Die Quelldatei, welcher in einen Texteditor geschrieben ist, und eine Dateiendung mit ASM besitzt, beinhaltet das mnemonische Programm mit den Assemblerbefehlen.

Die Fehlerdatei -**.ERR**
Die Fehlerdatei (bzw. ERROR-Datei), welche vom Assembler erzeugt wurde (mit der Endung ERR) ist eine ASCII-Datei, welche die Fehler während Assemblierung mitprotokolliert. Es wird die Art der Fehler und Zeilen bekanntgegeben.

Beispiel: PRG4.ERR

```
Message PRG4.ASM 59 : Argument out of range.  Least significant bits used.
Message PRG4.ASM 68 : Argument out of range.  Least significant bits used.
Message PRG4.ASM 73 : Argument out of range.  Least significant bits used.
Message PRG4.ASM 78 : Argument out of range.  Least significant bits used.
```

Die Dokumentationsdatei -**.LST**
Die Dokumentationsdatei, welche vom Assembler erzeugt wurde (mit der Endung LST) beinhaltet das ganze Programm mit den Mnemonikcodes, deren Adresse und hexadezimalen Adressen.

Beispiel: PRG1.LST

```
MPASM 01.21 Released              PRG1.ASM   3-28-1996
22:34:07         PAGE  1

LOC  OBJECT CODE     LINE SOURCE TEXT
  VALUE

                    00001 ; Programm: PRG1.ASM
                    00002
                    00003         LIST P=16C54
                    00004
0000                00005         ORG 0h
0000 0CFF           00006 Start   MOVLW 0FFh ; W:=0FFh
0001 0028           00007         MOVWF 8    ; GPFR-8 := W
                    00008
0002 0C01           00009         MOVLW 01h  ; W:=1
                    00010                        ; W  F8  Z DC C
                    00011                        ; 01 FF  0 0  0
0003 01E8           00012         ADDWF 8,1  ; F8:=F8+W
0004 0C01           00013         MOVLW 01h      ; 01 00  1 1  1
0005 01E8           00014         ADDWF 8,1  ; F8:=F8+W
0006 0C01           00015         MOVLW 01h      ; 01 01  0 0  0
0007 00A8           00016         SUBWF 8,1  ; F8:=F8-W
0008 0C01           00017         MOVLW 01h      ; 01 00  1 1  1
0009 00A8           00018         SUBWF 8,1  ; F8:=F8-W
000A 0000           00019         NOP            ; 01 FF  0 0  0
01FF                00020         ORG 1FF
01FF 0A00           00021         GOTO Start
                    00022         END

MPASM 01.21 Released              PRG1.ASM   3-28-1996  22:34:07
PAGE  2

SYMBOL TABLE
  LABEL                              VALUE

Start                                00000000
__16C54                              00000001


MEMORY USAGE MAP ('X' = Used,  '-' = Unused)

0000 : XXXXXXXXXXX----- ---------------- ---------------- ----------------
0040 : ---------------- ---------------- ---------------- ----------------
0180 : ---------------- ---------------- ---------------- ----------------
01C0 : ---------------- ---------------- ---------------- ---------------X

All other memory blocks unused.


Errors   :    0
Warnings :    0
Messages :    0
```

Das codierte Programm -**.COD**
Das codierte Programm mit der Endung COD beinhaltet (unlesbar) die Symbole und Daten.

Das assemblierte Programm in Intel-HEX-Format -**.HEX**
Der vom Assembler erzeugte Code im Intel-HEX-Format, dient zur Programmierung des PIC-Microchip-Prozessors.

Beispiel: PRG1.HEX

```
:10000000FF0C2800010CE801010CE801010CA8001C
:06001000010CA800000035
:0203FE00000AF3
:00000001FF
```

Assembler-Befehle

Die 39 Befehle erlauben das Mnemonikprogramm komfortabel zu gestalten.

Überblick (ohne Erläuterung) der Assemblerbefehle
CBLOCK, CONSTANT, DATA, DB, #DEFINE, DW, ELSE, END, ENDC, ENDIF, ENDM, ENDW, EQU, ERROR, EXITM, EXPAND, FILL, IF, IFDEF, IFNDEF, INCLUDE, LIST, LOCAL, MACRO, MESSG, NOEXPAND, NOLIST, ORG, PAGE, PROCESSOR, RADIX, RES, SET, SPACE, SUBTITLE, TITLE, #UNDEFINE, VARIBALE, WHILE

Die wichtigsten Assemblerbefehle

PROCESSOR
Wählen der Prozessortype im Assemblerprogramm.

Syntax: PROCESSOR <Type>
Type = PIC16C54, PIC16C55, PIC16C56, PIC16C57, PIC16C71, PIC16C84, PIC17C42, PIC16C58, PIC16C64
Beispiel: PROCESSOR 16C54

TITLE
Der Text nach diesem Befehl, in zwischen zwei " " (double quotes) eingeschlossen (maximal 60 Zeichen), wird in der Dokumentationsdatei (LST) in den ersten Zeile gedruckt.

Syntax: TITLE "<Titel_Text>"
Beispiel: TITLE "Motorsteuerung Version 5.0"

SUBTITLE
siehe TITLE

Syntax: SUBTITLE "<Untertitel_Text>"
Beispiel: SUBTITLE "Diagnostikprogramm"

LIST
Der List-Befehl kann einige Parameter direkt im ASM-Programm einstellen.

Syntax: LIST [<list_option>, ... , <list_option>]

Die wichtigsten Optionen in Kürze:
C=nnn setzen der Zeichenbreite (optional 80)
N=nnn setzen der Zeile-pro-Seite (optional 59)
P=<Typ> Wählen der Prozessortype
PIC16C54, PIC16C55, PIC16C56, PIC16C57, PIC16C71,
PIC16C84, PIC17C42, PIC16C58, PIC16C64
R=nnn setzen des Zahlensystems
HEX (16), DEC (10), OCT (8) (optional HEX)

Beispiel: LIST p=16C54

EQU
Der Symbolname wird einer Konstanten, welche eine Zahl darstellt, im ganzen Programm zugewiesen.

Syntax: <Symbol-Name> EQU <Konstante>
Beispiel: Vier EQU 4

#DEFINE
Der Befehl DEFINE weist eine Zeichenkette einen Symbolnamen zu.

Syntax: #DEFINE <Symbol-Name> <Zeichenkette>
Beispiel: #DEFINE IntCon,3

ORG
Dieser Befehl setzt den Programmzähler auf die "neue" Adresse <Adresse>. Dieses kann gleich mit Setzen einer Sprungmarke <Label>, welche maximal 31 Zeichen lang sein darf, (GOTO, CALL) verbunden werden.

Syntax: <Label> ORG <Adresse>
Beispiel: Init ORG 0010h

END
Beendet das Assemblerprogramm (Mnemonikprogramm).

Darstellung von Zahlen im Assemblerprogramm
Wenn das Zahlensystem nicht explizit mit den Befehlen RADIX, dem Befehl LIST r=nnn oder der Option des MPASM-Aufrufes vereinbart wurde, so gilt das hexadezimale Zahlensystem im ganzen Mnemonikprogramm.

Dezimalzahlen [0,1,2,3,4,5,6,7,8,9]
Darstellung: **D'**<Zahlen>**'**
Beispiel: D'123'

Hexadezimalzahlen [0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F]
Darstellung: **H'**<Zahlen>**'**
Beispiel: H'0FF'

Oktalzahlen [0,1,2,3,4,5,6,7]
Darstellung: **O'**<Zahlen>**'**
Beispiel: O'176'

Binärzahlen [0,1]
Darstellung: **B'**<Zahlen>**'**
Beispiel: B'10101010'

Zeichen [0,..,9,A,..,Z,a,..,z]
Darstellung: **'**<Zeichenkette>**'** oder **A'**<Zeichenkette>**'**
Beispiel: **A**'Hallo'

Aufbau des Mnemonikprogramms mit Assemblerbefehlen
Der Aufbau eines Mnemonikprogramm sollte in der Art und Weise erfolgen, wie es in den Programmbeispielen im Text (1 bis 6, 6a, 6b) gezeigt worden ist.

Simulator
Der Microchip Prozessorsimulator (MPSIM=Microchip Prozessor SIMulator) ist ein guter Ersatz für ein Monitorprogramm. Dieser Simulator kann zwar keine Echtzeitsimulation, wie es bei dem Microchip-Emulator der Fall ist, durchführen, jedoch ist er eine gute Kontrolle für das ganze Programm oder Teilen davon.

Es können alle Register, der Stack und beim 16C84 sogar das EEPROM angezeigt und manipuliert werden.

Nur eine kurze Einarbeitungszeit ist notwendig, um mit Hilfe dieses Simulators alle PIC-Programme zu testen.

Der Programmaufruf
Der Programmaufruf ist sehr einfach.
X:>*MPSIM*

Der Umgang mit MPSIM
Die Bildschirmmaske des Simulationsprogramm wird in 3 Bereiche geteilt:

Die *Titelleiste*, welche den Programmnamen, das Zahlensystem, den Prozessortyp, die Laufzeit und die Zyklenanzahl angibt.

Der *Überwachungsmonitor*, welcher den Inhalt der veränderbaren Register anzeigt.

Der *Kommandobereich*, welcher als Schaltzentrale für den Simulator dient. Hier werden einzelne Befehle mit bzw. ohne Optionen für den Simulationsbetrieb eingegeben.

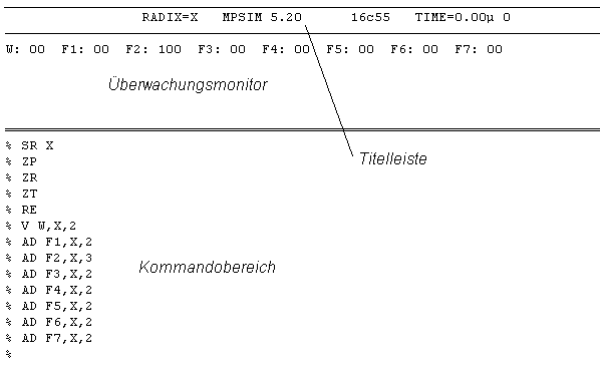


Abb.9: **Bildschirmmaske des MPSIM**

Die Simulatorbefehle
Dem Anwender stehen (ca.) 64 Befehle zur Verfügung.

Darstellung von Register und anderes
Register werden hier üblich mit F und anschließend der Registeradresse angegeben. Die Ports (PA, PB, PC) werden hier mit Register (RA, RB, RC) angeben. Das Eingabeformat (Zahlensystem) ist das Standardzahlensystem.

Die wichtigsten Befehle
Hilfe anzeigen: ***H***
Syntax: ***H***
Anzeige der einzelnen Befehle mit deren Optionen.

Laden eines Programmes: ***LO***
Syntax: ***LO*** *Dateiname*
Dieser Befehl lädt eine Datei (konkret eine Dokumentationsdatei (LST), wenn nicht anders angegeben) in den Simulator.

Bsp: *LO PRG6*

Anzeigen des Programmes: ***DI***
Syntax: ***DI*** *[Anfangsadresse [,Endadresse]]*
Zeigt das Programm im Kommandofenster (auch in Mnemonikcode) von der Anfangsadresse (falls angegeben), bis zur Endadresse (falls Anfangsadresse angegeben wurde). Ist diese Anfangsadresse nicht angegeben, so wird das Programm fortlaufend angezeigt.

Bsp: *DI* .. Zeigt Programmcodes 00h-0Ah, 0Bh-15h, 16h-20h,...
DI 3 .. Zeigt Programmcodes von 03h bis 0Dh
DI 4,6 .. Zeigt Programmcodes von 04h bis 06h

Dem Überwachungsmonitor ein Register zufügen: ***AD***
Syntax: ***AD*** *Register [,Zahlensystem[,Stellen]]*
Fügt dem Überwachungsmonitor ein Register in dem Zahlensystem (nur wenn angegeben *B*inär *H*exadezimal *D*ezimal *O*ktal) und deren maximale Stelle (nur wenn dieses und Zahlensystem angegeben) hinzu.

Bsp: *AD F9* .. Fügt dem Überwachungsmonitor das Register F9 in dem Standardzahelnsystem hinzu.
AD F9,B,4 .. Fügt dem Überwachungsmonitor das Register F9 im binären Zahlensystem der ersten 4 Bit an.

Aus dem Überwachungsmonitor ein Register löschen: ***DV***
Syntax: ***DV*** *Register*
Löscht das angegebene Register aus der Anzeige des Überwachungsmonitors.

Bsp: *DV F2*

Einen Breakpoint setzen: ***B***
Syntax: ***B*** *Adresse* bzw. ***B*** *Register [Bedingung Zahlenwert]*
Bricht die Simulation NACH Exekution einer bestimmten Adresse oder einem bestimmten Register mit einer Bedingung (=,>,>,>=,<=,!=) und einem Zahlenwert ab.

Bsp: *B 0100h* .. Abbruch nach Exekution des Befehles in 0100h
B Marke .. Abbruch nach Exekution des Befehles in Marke
B F2 > 80 .. Abbruch wenn Register 80 den Wert 80 überschreitet.

Einen Breakpoint löschen: ***BC***
Syntax: ***BC*** , ***BC*** *Adresse* bzw. ***BC*** *Register*
Löscht alle Breakpoints, wenn keine speziellen Adressen und Register angegeben wurden oder einen bestimmt Breakpoint, welcher durch den Registernamen oder die Adresse angegeben wurde.

Bsp: *BC* .. löscht alle Breakpoints
BC F2 .. Löscht Breakpoint mit Bedingung mit F2
BC 0010h .. Löscht Breakpoint in Zeile 0010h

Die Breakpoints anzeigen: ***DB***
Syntax: ***DB***
Der Simulator gibt eine Liste über alle Breakpoints aus.

Ein Programm starten: ***E***
Syntax: ***E*** bzw. ***E*** *Adresse*
Dieser Befehl exekutiert das Programm ab dem momentanen Standpunkt (Programmzähler), wenn keine Adresse angegeben wurde, oder startet das Programm bei einer bestimmten Adresse.

Bsp: *E* .. Startet das Programm ab momentanen Adresse (PC)
E 06 .. Startet das Programm bei der Adresse 06h.

Ein Programm vom Anfang starten: ***GO***
Syntax: ***GO***
Dieser Befehl realisiert einen POWER-ON-RESET und startet das Programm.

Einzelbefehlschrittfolge: ***SS***
Syntax: ***SS*** bzw. ***SS*** *Adresse*
Dieser Befehl exekutiert einen einzelnen Befehl, wenn angegebenen, an einer bestimmten Adresse.

Ein Registerinhalt ändern: ***F***
Syntax: ***F*** *Register*
Dieser Befehl erlaubt es, den Inhalt des angegebene Registers zu ändern. Der Inhalt wird nach Aufruf geändert.

Bsp: *F F2* ... Ändern des Inhaltes Register F2 (02h).

Einen Ein/Ausgabeport/pin ändern: ***SE***
Syntax: ***SE*** *[EinAusgabeport/pin]*
Ändern des physikalischen Ein/Ausgabeports (RA, RB, RC, ..) oder eines Pins.

Bsp: *SE RA* .. Ändern des ganzen PA-Ports
SE RA0 .. Ändern des Portes (Pin) PA0

Die Register anzeigen: ***DR***
Syntax: ***DR***
Zeigt in einer übersichtlichen Liste alle Register, Stacks, Portzustände (auch TRIS), Konfigurationsregister, etc. und sogar den Inhalt des EEPROM-Datenspeicher (bei PIC16C84) an.

Den EEPROM-Inhalt modifizieren: ***EE***
Syntax: ***EE*** *EEPROM-Adresse*
Mit diesem Befehl kann man den Inhalt einer einzelne EEPROM-Zelle ändern.

Bsp: *EE 1* .. Ändern der EEPROM-Zelle mit der Adresse 1

Die Systemzykluszeit ändern: ***SC***
Syntax: ***SC*** *[Zyklenzeit]*
Setzen der Zyklenzeit (in Mikrosekunde)
Bsp: *SC .2* .. Setzen der Zyklenzeit auf 200ns

Ein Taktsignal einem Eingang zuführen: ***CK***
Syntax: ***CK*** *[Pin, {#Hi,#Low|-}]*
Mit diesem Befehl wird ein Taktsignal an ein Pinport *Pin* des Prozessors angelegt (nur Simulation), welches eine Highzeit von *#Hi*-Takten und *#Low*-Takten besitzt. Statt der Low-High-Zeiten kann auch ein Bindestrich - eingegeben werden, welcher das Signal ausschaltet.

Bsp:*CK RC0,5,4* .. Takt (5:4) an PC0
CK RC0,- .. kein Takt an PC0

Eine Taste eine "hardwaremäßige" Eingabe zuweisen: ***DK***
Syntax: ***DK*** *[Funktionstaste#, [PortPin, Ereignis]|[-]]*
Mit diesem Befehl (ohne weitere Optionen) werden alle Tastenzuweisungen angezeigt. Bei einer Zuweisung gibt die Zahl (1-12) nach dem Code *DK* die Funktionstaste, die bei Betätigen der ALT-Funktionstaste-Nr.x an dem angegebenen Pin eines angegebenen Ports folgende angegebenes Ereignis (*H* - high, *L* - low, *T* - Toggle, *P* - Puls) simuliert, an. Ein Befehlsaufruf mit der Option Bindestrich löscht eine/alle ALT-Funktionstastendefinitionen.

Bsp: *DK* .. Zeigt alle Definitionen
DK 1,RB0,L .. Bei ALT-F1 ein LOW-Signal an PB0
DK 2,MCLR,P .. Bei ALT-F2 ein Impuls an MCLR
DK 2,- .. Löschen des Belegung von F2
DK - .. löschen aller Belegungen (F1..F12)

Programmende: ***Q***
Syntax: ***Q***
Quittiert das Simulationsprogramm.

Zum Exekutieren der eingegebenen Befehle muß die Taste RETURN bzw. ENTER betätigt werden.

Das Verhalten des Simulators bei speziellen Operationen

RESET-Verhalten
Es werden alle Reset-Typen (Power-On-Reset, /MCLR=0, WDT-Auszeit) simuliert.

SLEEP
Der MPSIM simuliert auch den SLEEP-Befehl; eine SLEEP-Funktion kann durch ein Aufwecken durch z.B. einen Watchdog-Auszeit durchgeführt werden.

Watch-Dog-Auszeit
Der WDT wird voll im Simulator unterstützt.

Die INItialisierungsdatei
Mit Hilfe einer in einem Editor geschriebenen INI-Datei, welche eine Liste von Simulatorbefehlen beinhaltet, kann mit Hilfe des Befehles GE eine Datei geladen und ausgeführt werden.

Programmierung

Neben Assembler und Simulator hat Microchip auch noch ein kleines Programmiergerät entwickelt. Es können nicht alle PIC-Prozessoren mit ein und demselben Gerät programmiert werden. Das Entwicklungstool PICSTART16B1 kann die PIC16C54, PIC16C54A, PIC16CR54, PIC16C55, PIC16C56, PIC16C57, PIC16C58A, PIC16C61, PIC16C620, PIC16C621, PIC16C622 PIC16C71, PIC16C71A und PIC16C84 programmieren. Die Steuerungssoftware ist einfach zu bedienen, da es Pull-Down-Menüs

besitzt, wie man in der folgenden Abbildung erkennen kann. Und das Programmiergerät (Netzgerät beiliegend) wird an die serielle Schnittstelle angesteckt.

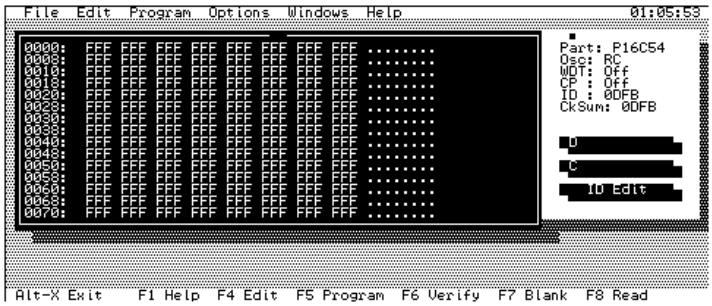


Abb.10: **Bildschirmmaske des MPS16B**

Entwicklungssysteme

Microchip bietet für wenig Geld ein gesamtes Entwicklungstool inkl. Assembler, Simulator, Programmiergerät, Datenbücher, Netzadapter und Musterprozessoren an.

Daher kann man sagen, das der PIC vom Microchip klein aber „oho“ in der Prozessortechnik und Entwicklungsumgebung und „PIC-fein“ in Sachen Service ist.

Quellenverzeichnis

Literaturquelle:

MICROCHIP Databook 1994
Microchip Technology Inc.
2355 West Chandler Blvd. Chandler, AZ 85224-6199

MPASM Assembler User's Guide
Microchip Technology Incorporation 1994

MPSIM Simulator User's Guide
Microchip Technology Incorporation 1994

Tabellen- und Bilderquellen

Abb. 1: **Blockschaltbild der PIC16C5x-Familie**
Microchip Databook 1994, „Figure 2.1.1 - PIC16C5X Series Block Diagram“, Page 2-6, DS30015K-page 4

Abb. 2: **PIN-Belegung**
Microchip Databook 1994, „Figure A - PIN Configuration“, Page 2-3, DS30015K-page 1

Abb. 3: **Blockdiagramm v. RTCC/WDT Prescaler**
Microchip Databook 1994, „Figure 9.0.1 - Block Diagram RTCC/WDT Prescaler“, Page 2-19, DS30015K-page 17

Abb. 4: **Blockschaltbild des PIC16C71**
Microchip Databook 1994, „Figure B - PIC16C71 Block Diagram“, Page 2-328, DS30150E-page 2

Abb. 5: **Pinbelegung des PIC16C71**
Microchip Databook 1994, „Figure A - Pin Configuration“, Page 2-327, DS30150E-page 1

Abb. 6: **Blockschaltbild des PIC16C84**
Microchip Databook 1994, „Figure B - PIC16C84 Block Diagram“, Page 2-536, DS30081C-page 2

Abb. 7: **Pinbelegung des PIC16C84**
Microchip Databook 1994, „Figure A - Pin Configuration“, Page 2-535, DS30081C-page 1

Abb.8: **Maske des MPASM**
Microchip Assembler MPASM Version 1.21 (DOS)

Abb.9: **Bildschirmmaske des MPSIM**
Microchip Simulator MPSIM Version 5.11 (DOS)

Abb.10: **Bildschirmmaske des MPS16B**
Microchip Programmiergerät PIC-Start 16 B 1 (DOS)

Tabelle 1: **Microchip - PIC - Family**
Microchip Databook 1994, „PIC16/17 Family of 8-Bit Mikrocontrollers Cross-Reference Guide“, Page 2-1, DS30352A-page1

Tabelle 2: **/TO & /PD-Ereignisse**
Microchip Databook 1994, „Table 4.5.2.1 - Event Affecting /PD|/TO Status Bits“, Page 2-14, DS30015K-page 12

Tabelle 3: **/PD & /TO nach Reset**
Microchip Databook 1994, „Table 4.5.2.2 - /PO|/TO Status after Reset“, Page 2-14, DS30015K-page 12

Tabelle 4: **Quarz-Oszillator**
Microchip Databook 1994,
„Table 12.2.1 - Capacitor Selection For Ceramic Resonant“ &
„Table 12.2.2 - Capacitor Selection For Crystal Oscillator“,
Page 2-26, DS30015K-page 24

Tabelle 5: **RC-Takt**
Microchip Databook 1994, „Table 18.0.1 - RC Oscillator Frequencies“, Page 2-40, DS30015K-page 38

Tabelle 6: **WDT-Zeitperiode abhängig von Temperatur und Versorgungsspg.**
Werte aus folgendem Diagramm abgelesen:
Microchip Databook 1994, „Figure18.0.15 - WDT Timeout Period vs. V_{DD}“, Page 2-44, DS30015K-page 42

Tabelle 7: **Spannungspegel an /MCLR**
Microchip Databook 1994, „16.6. DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (Commercial), PIC16C5XI-RC, XT, HS, LP (Industrial); 16.7. DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (Automotive)“, Page 2-35 & 36, DS30015K-page 33 & 34

Programmlisting 1: **Beispiel für STATUS-Register-Optionen**

Programmlisting 2: **Beispiel für RTCC**

Programmlisting 3: **Beispiel für INTERRUPT an PB0**

Programmlisting 4: **Beispiel für 16-Bit-Zähler mit INTERRUPT**

Programmlisting 5: **Beispiel für AD-Betrieb**

Programmlisting 6a: **Beispiel für EEPROM-LESEN**

Programmlisting 6b: **Beispiel für EEPROM-SCHREIBEN**

Programmlisting 6: **Beispiel für EEPROM**

weitere Literatur

„PIC Controller“, Michael Thieser, Franzis-Verlag

„Mikrocontroller mit RISC-Struktur; Die PIC-16C5X-Familie“, C.F. Urban, elektor

„PIC Anwendungssammlung; komplexe Schaltung einfach lösen mit PICs“, elektor

„Klein, aber PICfein“, Prof. Dr. Anne Frohn-König + Dipl. Ing. (FH) Manfred König, ELRAD Heft 5/93 und 6/93

„PIC16Cxx/PIC17Cxx Mikrocontroller; Architektur, Werkzeuge, Applikationen“, Johann Wiesböck + Martin Burghart, Design&Elektronik + Microchip

„Das Parallax Assembler Arbeitsbuch zu dem Mikrocontrollern PIC16Cxx“, Scott Edwards + Claus Kühnel, Electronic Media GmbH

☐

Mikrocontrollerboard für 80C537 oder 80C517A

und parallele Ein- Ausgabeerweiterung für diese Mikrocontroller (oder andere).

Hermann Schönbauer PCN-DSK-520:/uc537

Diese Applikation besteht aus zwei einseitigen Europakarten. Die erforderliche 2. Verdrahtungsebene ist mittels Drahtbrücken realisiert und dadurch auch im Schul- oder Ausbildungsbereich nachbaubar. Keine Durchkontaktierungen!!

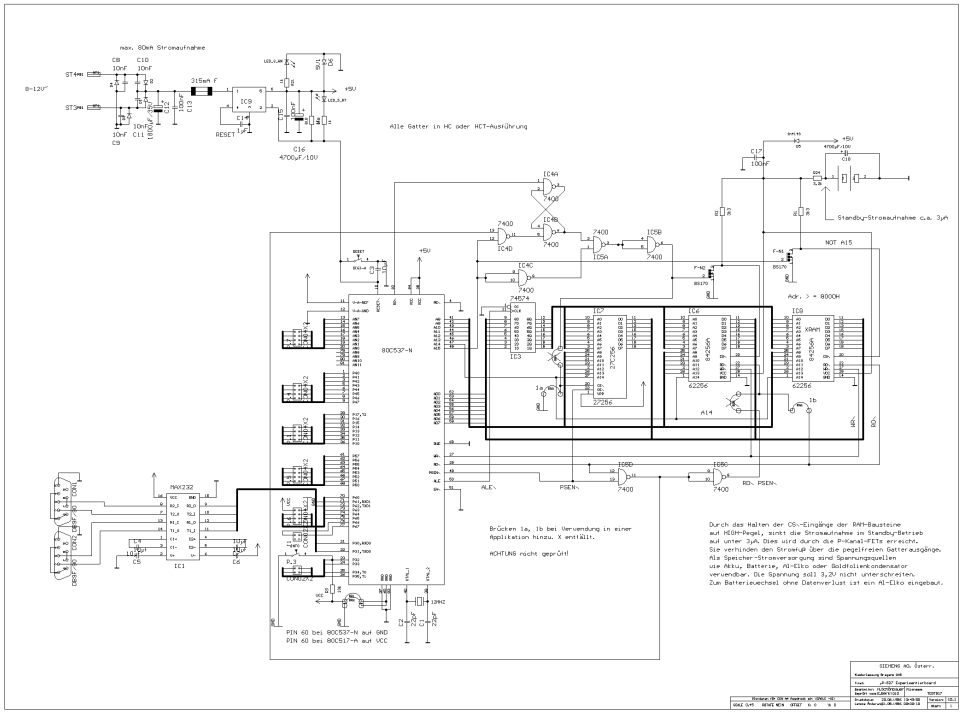
Mikrocontrollerboard

Das CMOS-RAM des Mikrocontrollerboards ist mit Batterie oder Akku bufferfähig (3µA). Es sind alle Ports ausgenommen Port 8 (4 A/D-Eingänge) herausgeführt und die zwei seriellen Schnittstellen auf V24 umgesetzt. Die Stromversorgung ist integriert, so daß nur eine Einspeisung von 8-12V Wechselspannung erforderlich ist. Bei Verwendung des 'KEIL-MONITORs' kann ein HEX-File über die S1-Schnittstelle downgeladen werden. Der Monitor (EPROM) schaltet sich dann auf die Adr. 8000H, wobei das downgeladene Programm im RAM steht.

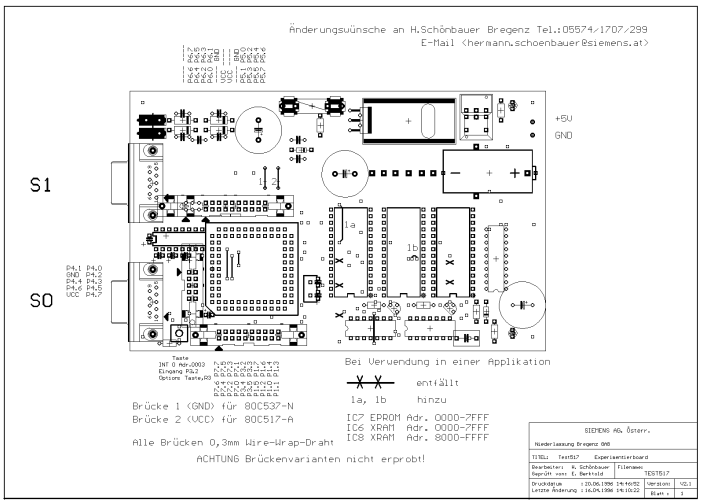
Das Handling mit einem Monitorprogramm ist der jeweiligen Monitorbeschreibung zu entnehmen. Es kann aus lizenzrechtlichen Gründen nicht weiter darauf eingegangen werden.

Alle Logikschaltungen in HC- oder HCT-Ausfürung.

Schaltbild



Layout



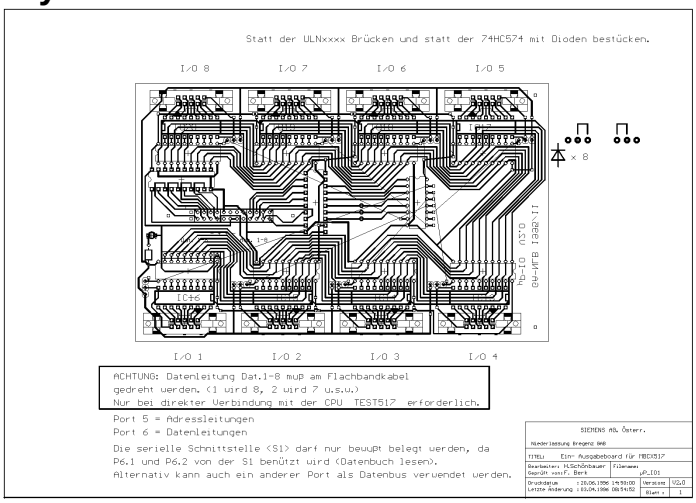
Ein- Ausgabeerweiterung

Sie ermöglicht die Ansteuerung bzw. Abfrage von bis zu 192 I/Os. Hierbei wird ein Port (8-Bit) für die Bausteinadressierung und ein zweites Board für die Datenvermittlung verwendet. Zur Adressierung werden 3 Bit pro Baugruppe (2'0 - 2'2) und weitere 3 Bit (2'3-2'5) zur Platinenselektion benötigt. 2'6 ist der Übernahmetakt für den gerade adressierten Baustein und 2'7 bestimmt die Datenrichtung (Lesen oder Schreiben).

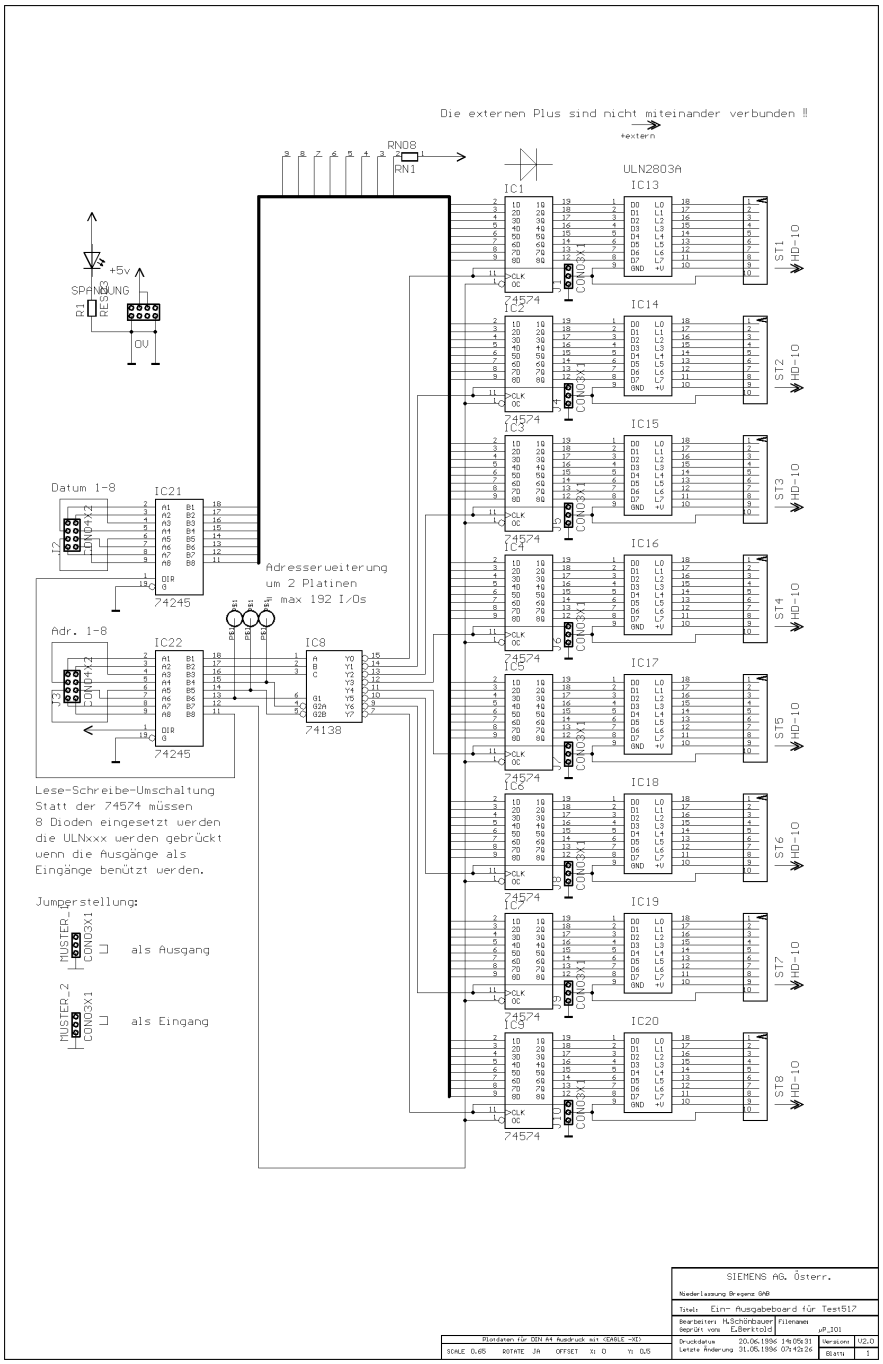
Eine höhere Dekodierung der Baugruppen wäre möglich (7x8 statt 3x8), aber nur bei sehr langsamen Applikationen sinnvoll (>100ms) und ist bei dieser Hardware nicht realisiert.

Alle Logikschaltungen in HC- oder HCT-Ausfürung.

Layout



Schaltbild



Wie mache ich meine Eingänge störungsunempfindlich?

Zur Philosophie:

Die Eingänge sollen möglichst oft abgefragt werden, um eine vernünftige Entprellung (Ausblenden von Störeinflüssen) zu gewährleisten.

Die Ausgänge sollen innerhalb von 10ms bedient werden.

Zur Entprellung:

1.) Eine Methode (hier nicht verwendet) ist, den Zustand 1 oder 0 als Addier- oder Subtrahierkriterium herzunehmen und die Auswertung über ein Trendbit und Grenzwerte vorzunehmen. Diese Art benötigt eine relativ hohe Abfragerate und ist recht aufwendig.

Sie ist nur für Extremfälle sinnvoll, da ein Abtastabstand von etwa 1ms erforderlich ist. Dafür ist sie FUZZY-ähnlich und kann mit entsprechenden Schwellwerten ausgestattet werden.

2.) Eine andere Methode, die hier zur Anwendung kommt, ist:

Man pollt die Eingänge ca. alle 10ms, wobei das Lesen (Portzugriff) nur eine Übernahmezeit von 1µs benötigt und akzeptiert den Wert nach etwa 8 gleichen Ergebnissen pro Bit.

Nach 8 x 10ms = 80ms wird dieses Ergebnis einer Auswerteroutine übergeben (ein für die meisten Eingänge annehmbarer Wert).

Allgemeines über Störungen innerhalb von Geräten oder Maschinen

Eine Störung kann durch Lichtbögen oder durch Wechselmagnetfelder entstehen. Den Einfluß von Funkstörungen schließe ich bei dementsprechendem Aufbau und Außenbeschaltung aus. Ein Lichtbogen ist außer mit der Methode 1 kaum auszuschließen (ähnlich Funkstöreinflüssen). Ein Wechselmagnetfeld induziert hingegen seine Störung relativ langsam 10ms in eine und 10ms in die andere Richtung (bei 50Hz), wobei das Nutzsignal einmal verstärkt und einmal geschwächt wird. Verbleiben also 10ms in denen sich die Störung auswirkt. Wenn man die Abtastrate asynchron zu den 50Hz setzt wird die Wahrscheinlichkeit einer Störung stark reduziert. Es sollten 8 Abfragen nach jeweils 10ms für eine sichere Eingangsbestimmung reichen.

Bei Anwendung der Methode 2 steht das Eingangsergebnis also nach 80ms fest.

Tritt eine Störung innerhalb dieser Zeit auf, so wird der Zyklus von 8 Abfragen wiederholt. Es tritt eine dementsprechende Verzögerung ein. Diese Verzögerung ist in den meisten Fällen tragbar. Wenn man noch bedenkt, daß die Abfragezeit jeweils nur 1µs beträgt und der Abfragezyklus 10ms, so entsteht ein Verhältnis von 1:10.000 und die Wahrscheinlichkeit einer Störung wird dementsprechend gering. Man kann auch pro Störung in der Abfragephase den Entprellwert inkrementieren und so eine schnellere Gültigkeit erreichen.

Es gibt natürlich Anwendungen wo dies nicht möglich ist und eine andere Lösung gesucht werden muß (Hardwareaufwand). Nach meiner langjährigen Erfahrung sind solche Fälle ausgesprochen selten und wenn, dann vorhersehbar.

Softwaremodul-Beschreibung

Datenfelder:

`def_feld[max_port]`

Bestimmt die Verwendung als Eingang, Ausgang oder nicht benützt.

1=Ausgang

0=Eingang

2=NICHT

`entprell[max_port][8]`

Vorgabe der Abfrageanzahl pro Eingang (Bit) - Entprellzahl

`entp_count[max_port][8]`

Temporäre Abfrageanzahl wird aus `entprell[][]` bei Eingangsänderung nachgeladen und, wenn größer `NULL`, bei Gleichheit dekrementiert. Zur Quittierung wird eine `80H` in dieses Byte geschrieben, um eine mehrfach Auswertung zu verhindern. Mit einer Zahl unter 80H kann eine "REPEAT-Funktion" erreicht werden.

`ein_aus[max_port]`

Speicher der Portzustände, daß heißt die Ausgänge werden in 'ein_aus' geschrieben und von dort an die Ports per Timerinterrupt weitergegeben. In der Leserichtung steht immer der letzt gelesene Eingangspegel in 'ein_aus'. Aber erst nach einer Abfrage von `'entp_count[][]'` auf "`IST=NULL`", ist diese Information auch gültig.

Um sich Schreibaufwand zu ersparen und die Lesbarkeit eines Programmes zu erhöhen, ist es von Vorteil, Abfrage- ,Quittierungs- und Repetitions-Makros zu schreiben. Diese kann man dann in einen eigenen Header (`xxx.h`) unterbringen.

z.B.:

```
// Taste 1 Abfragen auf LOW-aktiv
#define   t1_akt   (entp_count[x][y]&0x7f==0)&&(!ein_aus&z)
```

Im Programm sieht dies so aus:

```
   'if(t1_akt) {....;}'
       x ... Baustein  (0-7)
       y ... Bit       (0-7)
       z ... x*8+y
```

Programmodule:

`I_O()`	Selektion der Portrichtung (Schreiben/Lesen/Unbenutzt)
`ausg()`	Ausgabe des `'ein_aus[]'` Feldes auf den Port
`eing()`	Einlesen des Ports in das Feld `'ein_aus[]'`, Entprellen der einzelnen Eingänge eines Ports
`init_mpio()`	Löschen der Ausgänge im `'ein_aus[]'`-Feld und deren Ausgabe; Vorbesetzen der Entprellbytes

Alle weiteren Module dienen der Kontrolle der Ein- und Ausgänge und sind je nach Anwendung zu gestalten.

Programmaufbau in diesem Beispiel:

```
main()
{
  init_mpio()  // Ausgänge passiv setzen
  init_int0()  // EXTERN Interrupt 0 bei Bedarf (Ausstieg aus
               // dem KEIL-Monitor
  b9600()      // Serielle 0 mit 9600Baud initialisieren
  init_sp()    // V24-Spooler löschen und initialisieren
  init_t()     // Timer 2 mit 10ms autoreload setzen
               // und Interrupt vorbereiten
  forever      // Ab hier nur zur Demonstration
  disp_ios()   // Anzeige der Ein- und Ausgänge über V24 (S0)
  V24_ausw()   // Auswertung der seriellen Schnittstelle 0
               // wenn Datensatz vollständig
}

t2_int()       // Aufruf alle 10ms per Timer2-Interrupt
{
  v24_abf()    // Pollen der S0-Schnittstelle
               // und deren Bewertung
  I_O()        // Bedienung der I/Os je nach Verwendung
  {
    ausg()     // Ports alle 10ms beschreiben, unabhängig
               // von einer Änderung. So werden etwaige
               // Störungen der Ausgänge unterbunden.
    eing()     // Ports lesen und entprellen.
  }
}
```

Die Unterlagen zum Nachbau der hier verwendeten CPU-, I/O-Platine und eine Demosoftware kann über 'hermann.schoenbauer@siemens.at' bezogen werden

Die Hardware ist mit ‘EAGLE 3.02’, die Software mit ‘µVISION/51 für Windows’ Version 1.12 und C51-Compiler V5.0 von ‘KEIL’ erstellt worden.

Unter der Telefonnummer '05574 1707 299' stehe ich gerne für weitere Fragen zur Verfügung. ☐

```
/*****************************************************************
*                    Ein- Ausgabe-Programm                       *
*  fuer Platine "MPIO" ab Version V2.0 der GA-Bregenz            *
*  Benuetzte Entprellroutine (c) by H. Schoenbauer 1986          *
****************************************** 31-03-1996 **********/

#include        <reg517a.h>
#include        <stdio.h>
#include        <ctype.h>
              // 3 Platinen x 8 Bausteine x 8 Pin = 192 Pin
              // 3 Platinen x 8 Bausteine             =  24 Pin
#define         max_port        24
                   // I/O-Port
#define         daten_p     P6
              // Select und Clock
#define         adr_p       P5
              // verwendete Registerbank fuer Timer-Interrupt
#define         reg_bank    2
              // Eingangsrepetition z.B.:
              // fuer Tasteneingaenge
#define         repeat          8
#define         reload          -10000    //10ms TimerInterrupt
/* Eine praktikable Entprellzeit betraegt 80ms, dem entsprechend
   muss 'MPIO' per Timerinterrupt aufgerufen werden.
   Die Ausgaenge sind in einem Daten-Array jederzeit setz- und
   ruecksetzbar. Die Bedienung erfolgt per Timerinterrupt.
   Es werden max 1x8 Ein- oder Ausgaenge pro Interrupt bedient,
   um das Hauptprogramm nicht zu sehr zu bremsen.

Aufbau der Portadressen (adr_p):

b^0-b^2         Bausteinauswahl 0-7
b^3-b^5         Platinenauswahl 0-2       0 0 1   1.Platine
                                          0 1 0   2.Platine
                                          1 0 0   3.Platine
b^6             Ausgabetakt/Einlesetakt         LOW-aktiv
b^7             Schreibe/Leseumschaltung fuer 74HC245
                        gesteuert ueber 'def_feld'
                                          1       Schreiben
                                          0       Lesen
def_feld (Definitionsfeld)
        Bestimmung der Erweiterungsports als Ein- bzw. Ausgaenge
                                          0 = Eingang
                                          1 = Ausgang
                                          2 = nicht verwendet oder vorhanden
max_port (max. I/O-Portanzahl (Port=8bit))   1-24
*/
//
// Timerinitialisierung, T2 mit 'reload'-Wert
#include        "T2.c"
// V24 Schnittstelle 'S0' mit 9600 Baud initialisieren
#include      "b9600.c"
/*********************************************************
*                        V 24 - H A N D L E R              *
*                                                          *
*********************************************************
*/

#define                 max_s_len       10
        // Eingangs-Spooler
xdata unsigned char     feld[max_s_len];
        // Ein- Ausgabezeiger
xdata unsigned char     *ein_z;
xdata unsigned char     *aus_z;
xdata unsigned char     diff;       // Differenzzaehler fuer V24-Empfang
idata unsigned char     text[9];
idata unsigned char     temp[4];

        // Eingangsspooler loeschen, Zeiger setzen
void init_sp(void)
{
  for(diff=0;diff<max_s_len;diff++)
          { feld[diff]=0; }
  ein_z=&feld[0];
  aus_z=&feld[0];
  diff=0;
}

void v24_abf(void)         using reg_bank
{
  if(RI0)   {
              *ein_z=S0BUF;
              if((*ein_z>0x2f) && (*ein_z<0x3a) || (*ein_z=='+') || (*ein_z=='-'))
                {
                diff++;
                S0BUF=*ein_z; TI0=0;          // Echo
                if(++ein_z>=&feld[max_s_len])     ein_z=&feld[0];
                }
              RI0=0;
              }
}

code unsigned char def_feld[max_port] = {
                        0,0,2,2,1,1,1,1,  // 2xEingang,2xnicht,4xAusgang..
                        2,2,2,2,2,2,2,2,  // nicht verwendet.............
                        2,2,2,2,2,2,2,2   // ............................
                                        };
   // 'entprell' = Anzahl der Einlesezyklen pro Eingang, NICHT pro Port
   // normal 80ms. Wenn alle 10ms eine Portbehandlung erfolgt, ergeben
   // sich 8 x 10ms = 80ms. Damit besteht die Moeglichkeit stoeranfaellige
   // Eingaenge hoeher zu entprellen.
   // Bereich 1-127, 128 (0x80) wird zur Quittierung beim Abarbeiten
   // verwendet.
code unsigned char entprell[max_port][8] = {
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8,
                                8,8,8,8,8,8,8,8,  8,8,8,8,8,8,8,8
                                                        };

// Gegenstueck von 'entprell' im RAM, wird beim Generieren mit dem
// Wert aus 'entprell' beschrieben und bei jeder Eingangsaenderung
// nachgeladen.
xdata unsigned char entp_count[max_port][8];

//Shadow-Ports, Inhalt entspricht dem momentanen I/O-Zustand. Je nach Verwendung des
Erweiterungsportes befindet sich //der Eingangszustand IST unabhaengig vom Entprellstatus
im Speicher 'ein_aus[x]'.
//Oder der Ausgangszustand wird in 'ein_aus[x]' geschrieben und beim entsprechenden
Timerinterrupt ausgegeben.

xdata unsigned char ein_aus[max_port];

// Zaehler fuer Ein-/Ausgabe beim Timerinterrupt
idata unsigned char     zaehler;

// Fuer Aenderungskennung eines Einganges
bit                             ungleich;

void ausg(void)    using reg_bank
{
  idata unsigned char   n, x;
  n=zaehler&0x07;      // Bausteinauswahl:     Maske fuer Baustein
                 // Baugruppenauswahl:   Maske fuer Baugruppe
  if(zaehler<8)            n|=0x20;  // 1. Baugruppe
  else if(zaehler<16)   n|=0x10;   // 2. Baugruppe
  else                          n|=0x08;  // 3. Baugruppe
      n|=0x80;                // DIR auf Ausgabe, Ausgabedatum holen
  x=ein_aus[zaehler];
  n|=0x40;        // OC ON       OC==H     Daten ausgeben
  adr_p=n;    // OC OFF      OC==L
  daten_p=x;
      n&=0xbf;        // OC OFF      OC==L     und Trennen
  adr_p=n;
}

void eing()       using reg_bank     // Selbe Registerbank wie T2-Interrupt
{
  idata unsigned char   n,mex,x;
  adr_p=0xbf;
  adr_p=0x3f;                // OC==L , DIR auf Eingabe
  x=zaehler&0x18;            // Baugruppenauswahlmaske
  n=zaehler&0x07;    // Bausteinauswahl
                             // die Pin-Adressierung ist ohne Bedeutung,
                             // da alle 8 Bit
                             // gemeinsam eingelesen werden.
                             // Baugruppe bestimmen
  if(x<8)        n|=0x20;      // 1. Baugruppe
  else if(x<16)     n|=0x10;      // 2. Baugruppe
  else                  n|=0x08;      // 3. Baugruppe
  adr_p=n;                           // Adresse und CLK
      daten_p=0xff;               // Port auf Eingabe
      mex=daten_p;               // Port lesen
  adr_p=0x3f;                    // Baugruppe deselektieren
      //     Entprellvorgang Anfang
  x=mex;                                  // Neu-Daten merken
  mex^=ein_aus[zaehler];          // Unterschied zum letzten Lesen -> mex
                                          // 'mex' bedeutet Merker EXOR Alt<->Neu
  ein_aus[zaehler]=x;              // Neuen Wert merken
  x=0x01;                                 // Bit-Maske
  for(n=0;n<8;n++)                   // Bit-Zaehler
          {
        if(mex&x)                                     // Eingang veraendert?
           entp_count[zaehler][n]=entprell[zaehler][n];    // Counter nachladen
        else if(entp_count[zaehler][n] & 0x7F)         // Eingang gleich aber nicht
entprellt
            {entp_count[zaehler][n]--;                    // Entprellcounter -1
              if(entp_count[zaehler][n]==0)    ungleich=1;
            }                     //      else    ; Eingang ist gleich und schon entprellt -
nichts tun
        x=x<<1;                                  // naechstes Bit, - Masken-SHIFT
        }  }  //          Entprellvorgang Ende
/* Bei einer einzelnen Abfrage der Ports empfehle ich eine  switch(timer_zaehler) {
case(0).......case(x)  } Bedienung
Der Vorteil, bestimmte Programmteile koennen so in regelmaessigen kurzen Abstaenden bedient
werden, waehrend die I/Os zyklisch in groesseren Schritten ausgefuehrt werden. Dadurch
bleibt fuer das Hauptprogramm mehr Ausfuehrungszeit. */


void I_O(void)  using reg_bank
{
  zaehler=0;
  while(zaehler<max_port)
  {
                                          // Portverwendung selektieren
  if(def_feld[zaehler]==1)            ausg();
  else if(def_feld[zaehler]==0)       eing();
  zaehler++;
  }
}

/*******************************************************
        Initialisierung besetzt die Entprellbytes
          und setzt die Ausgaenge auf HIGH-Pegel
*******************************************************/
void init_mpio(void)
{
  idata unsigned char   j;
  for(zaehler=0;zaehler<max_port;zaehler++)
          {
          if(def_feld[zaehler]==1)
           {
                  ein_aus[zaehler]=0x00;       // Ausgang loeschen
                  ausg();                           // und ausgeben
           }
          else if(def_feld[zaehler]==0)
            {for(j=0;j<8;j++)        // Entprellcounter vorbesetzen
                  entp_count[zaehler][j]=entprell[zaehler][j];
            }
          }
}

init_int0(void)
{
  IT0=1;              // Initialisieren von Interrupt EX0
  EX0=1;              // Flankengesteuert
  IE0=0;              // Fuer Monitorabbruch bei Programmlauf im CMOS-RAM
}

t2_int(void)    interrupt 5             using reg_bank
{
  v24_abf();
  I_O();
  TF2=0;
}
```

```
// Bitmuster der Ausgaenge und Eingaenge darstellen

void disp_ios(void)       //      Darstellmodul der I/Os            ANFANG
{
  idata unsigned char     i,j,x1;
  for(i=0;i<max_port;i++)
         {
         if(def_feld[i]==1)      // Ausgabebaustein
           {
            j=0;
             for(x1=0x80;x1!=0x08;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='1'; }
                      else                  { text[j]='0'; }
                     j++;
       }
             text[j]='|'; j++;
             for(;x1>0x00;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='1'; }
                     else                  { text[j]='0'; }
                     j++;
         }
           printf("%c%c%c%c%c%c%c%c ",text[0],text[1],text[2],text[3],
                                      text[4],text[5],text[6],text[7],
                                      text[8]);
           }


    else if(def_feld[i]==0)// Eingabebaustein
           {
            j=0;
             for(x1=0x80;x1!=0x08;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='H'; }
                     else                  { text[j]='L'; }
                     j++;
       }
             text[j]='-'; j++;
             for(;x1>0x00;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='H'; }
                     else                  { text[j]='L'; }
                   j++;  }
           printf("%c%c%c%c%c%c%c%c ",text[0],text[1],text[2],text[3],
                                      text[4],text[5],text[6],text[7],
                                      text[8]);
           }

         else                   // Nicht benuetzt
           {
                 printf("----|---- ");
           }
         }
  printf("\n");
}        //        Darstellmodul der I/Os            ENDE

v24_ausw()      //  V24_Auswertung       ANFANG
{
  idata unsigned char     i, x1;
        temp[0]=*aus_z;
        diff--;
        temp[0]&=0x31;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
        temp[1]=*aus_z;
        diff--;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
        temp[2]=*aus_z;
        printf("\tAusgang %c%c%c ",temp[0],temp[1],temp[2]);
        diff--;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
        temp[3]=*aus_z;
        if(*aus_z=='+')
        { printf(" ON\n");}
        else
        { printf(" OFF\n");}
        diff--;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
  temp[0]=(temp[0]&0x01)*100;
  temp[0]=(temp[1]&0x0f)*10;
  temp[0]+=temp[2]&0x0f;
  temp[2]=temp[0]/8;            // Baustein bestimmen
  x1=temp[0]%8;                 // Bit-Position
  if(temp[2]<24)                // Baustein-Max begrenzen
  {
  if(def_feld[temp[2]]==1)      // Dann ist es eine Ausgabe
          {
    i=0x01; i=i<<x1;            // Bitposition einstellen
    if(temp[3]=='+')            // Ausgang ON
            {
            ein_aus[temp[2]] | =i;
            }
    else                          // Ausgang OFF, es ist eine Eingabe
            {
            i^=0xff;
            ein_aus[temp[2]] & =i;
            }
        }
  else
          { printf("\t\t\t\t U N G U E L T I G\n"); }
    printf("\n");
         }
  disp_ios();
  printf("Ihre Eingabe:  ");
}        //      V24_Auswertung          ENDE


void main(void)
{
  init_mpio();
  init_int0();
  b9600();
  RI0=0;
  init_sp();
  init_t();
  zaehler=0;
  printf("%c\n\t\t\tI/O-Karte mit 192 Ein- Ausgaengen",temp[0]);
  printf("\n\t\t\tgesteuert von MP-Karte 'TEST517'\n");
  printf("\n\t\tAusgang  ON   mit Ausgangnummer 0-191 und + , z.B.: 001+\n");
  printf("\t\tAusgang  OFF  mit Ausgangnummer 0-191 und - , z.B.: 001-\n\n");
  printf("Darstellung der Eingaenge mit H/L, der Ausgaenge mit 1/0\n\n");
  printf("Ihre Eingabe:  ");

  while(1) {
      P52=0;
      if(ungleich)
              {
               ungleich=0;
               printf("\n\n");
               disp_ios();
              }
      if(diff>=4)           // xxx = Ausgang   y=+  == ON  y!=+  == OFF
              {
              v24_ausw();
        }
      }
}
```

Real programmers don't.....

- *Real programmers don't write specs. Users should consider themselves lucky to get any programs at all and take what they get.*
- *Real programmers don't comment their code. If it was hard to write, it should be hard to read.*
- *Real programmers don't write application programs, they program right down on the bare metal. Application programming is for feebs who can't do systems programming.*
- *Real programmers don't eat quiche. Real programmers don't even know how to spell quiche. They eat Twinkies, Coke and palate-scorching Szechwan food.*
- *Real programmers don't draw flowcharts. Flowcharts are, after all, the illiterate's form of documentation. Cavemen drew flowcharts; look how much it did for them.*
- *Real programmers don't read manuals. Reliance on a reference is a hallmark of the novice and the coward.*
- *Real programmers programs never work right the first time. But if you throw them on the machine they can be patched into working in only a few 30-hours debugging sessions.*
- *Real programmers don't use Fortran. Fortran is for wimpy engineers who wear white socks, pipe stress freaks, and crystallography weenies. They get excited over finite state analysis and nuclear reactor simulation.*
- *Real programmers don't use COBOL. COBOL is for wimpy application programmers.*
- *Real programmers never work 9 to 5. If any real programmers are around at 9 am, it's because they were up all night.*
- *Real programmers don't write in BASIC. Actually, no programmers write in BASIC, after the age of 12.*
- *Real programmers don't document. Documentation is for simps who can't read the listings or the object deck.*
- *Real programmers don't write in Pascal, or Bliss, or Ada, or any of those pinko computer science languages. Strong typing is for people with weak memories.*
- *Real programmers know better than the users what they need.*
- *Real programmers think structured programming is a communist plot.*
- *Real programmers don't use schedules. Schedules are for manager's toadies. Real programmers like to keep their manager in suspense.*
- *Real programmers think better when playing adventure.*
- *Real programmers don't use PL/I. PL/I is for insecure momma's boys who can't choose between COBOL and Fortran.*
- *Real programmers don't use APL, unless the whole program can be written on one line.*
- *Real programmers don't use LISP. Only effeminate programmers use more parentheses than actual code.*
- *Real programmers disdain structured programming. Structured programming is for compulsive, prematurely toilet-trained neurotics who wear neckties and carefully line up sharpened pencils on an otherwise uncluttered desk.*
- *Real programmers don't like the team programming concept. Unless, of course, they are the Chief Programmer.*
- *Real programmers have no use for managers. Managers are a necessary evil. Managers are for dealing with personnel bozos, bean counters, senior planners and other mental defectives.*
- *Real programmers scorn floating point arithmetic. The decimal point was invented for pansy bedwetters who are unable to "think big."*
- *Real programmers don't drive clapped-out Mavericks. They prefer BMWs, Lincolns or pick-up trucks with floor shifts. Fast motorcycles are highly regarded.*
- *Real programmers don't believe in schedules. Planners make up schedules. Managers "firm up" schedules. Frightened coders strive to meet schedules. Real programmers ignore schedules.*
- *Real programmers like vending machine popcorn. Coders pop it in the microwave oven. Real programmers use the heat given off by the cpu. They can tell what job is running just by listening to the rate of popping.*
- *Real programmers know every nuance of every instruction and use them all in every real program. Puppy architects won't allow execute instructions to address another execute as the target instruction. Real programmers despise such petty restrictions.*
- *Real programmers don't bring brown bag lunches to work. If the vending machine sells it, they eat it. If the vending machine doesn't sell it, they don't eat it. Vending machines don't sell quiche.*
- *Real programmers know that the word is disk, not disc. Disc is a definite commie plot put forth by blubbering quiche eaters.*

```
// Bitmuster der Ausgaenge und Eingaenge darstellen

void disp_ios(void)       //      Darstellmodul der I/Os            ANFANG
{
  idata unsigned char     i,j,x1;
  for(i=0;i<max_port;i++)
         {
         if(def_feld[i]==1)      // Ausgabebaustein
           {
            j=0;
             for(x1=0x80;x1!=0x08;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='1'; }
                      else                  { text[j]='0'; }
                     j++;
       }
             text[j]='|'; j++;
             for(;x1>0x00;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='1'; }
                     else                  { text[j]='0'; }
                     j++;
         }
           printf("%c%c%c%c%c%c%c%c ",text[0],text[1],text[2],text[3],
                                      text[4],text[5],text[6],text[7],
                                      text[8]);
           }


    else if(def_feld[i]==0)// Eingabebaustein
           {
            j=0;
             for(x1=0x80;x1!=0x08;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='H'; }
                     else                  { text[j]='L'; }
                     j++;
       }
             text[j]='-'; j++;
             for(;x1>0x00;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='H'; }
                     else                  { text[j]='L'; }
                   j++;  }
           printf("%c%c%c%c%c%c%c%c ",text[0],text[1],text[2],text[3],
                                      text[4],text[5],text[6],text[7],
                                      text[8]);
           }

         else                   // Nicht benuetzt
           {
                 printf("----|---- ");
           }
         }
  printf("\n");
}        //        Darstellmodul der I/Os            ENDE

v24_ausw()      //  V24_Auswertung       ANFANG
{
  idata unsigned char     i, x1;
        temp[0]=*aus_z;
        diff--;
        temp[0]&=0x31;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
        temp[1]=*aus_z;
        diff--;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
        temp[2]=*aus_z;
        printf("\tAusgang %c%c%c ",temp[0],temp[1],temp[2]);
        diff--;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
        temp[3]=*aus_z;
        if(*aus_z=='+')
        { printf(" ON\n");}
        else
        { printf(" OFF\n");}
        diff--;
        if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
  temp[0]=(temp[0]&0x01)*100;
  temp[0]=(temp[1]&0x0f)*10;
  temp[0]+=temp[2]&0x0f;
  temp[2]=temp[0]/8;            // Baustein bestimmen
  x1=temp[0]%8;                 // Bit-Position
  if(temp[2]<24)                // Baustein-Max begrenzen
  {
  if(def_feld[temp[2]]==1)      // Dann ist es eine Ausgabe
          {
    i=0x01; i=i<<x1;            // Bitposition einstellen
    if(temp[3]=='+')            // Ausgang ON
            {
            ein_aus[temp[2]] | =i;
            }
    else                          // Ausgang OFF, es ist eine Eingabe
            {
            i^=0xff;
            ein_aus[temp[2]] & =i;
            }
        }
  else
          { printf("\t\t\t\t U N G U E L T I G\n"); }
    printf("\n");
         }
  disp_ios();
  printf("Ihre Eingabe:  ");
}        //      V24_Auswertung          ENDE


void main(void)
{
  init_mpio();
  init_int0();
  b9600();
  RI0=0;
  init_sp();
  init_t();
  zaehler=0;
  printf("%c\n\t\t\tI/O-Karte mit 192 Ein- Ausgaengen",temp[0]);
  printf("\n\t\t\tgesteuert von MP-Karte 'TEST517'\n");
  printf("\n\t\tAusgang  ON   mit Ausgangnummer 0-191 und + , z.B.: 001+\n");
  printf("\t\tAusgang  OFF  mit Ausgangnummer 0-191 und - , z.B.: 001-\n\n");
  printf("Darstellung der Eingaenge mit H/L, der Ausgaenge mit 1/0\n\n");
  printf("Ihre Eingabe:  ");

  while(1) {
      P52=0;
      if(ungleich)
              {
               ungleich=0;
               printf("\n\n");
               disp_ios();
              }
      if(diff>=4)           // xxx = Ausgang   y=+  == ON  y!=+  == OFF
              {
              v24_ausw();
        }
      }
}
```

Real programmers don't.....

- *Real programmers don't write specs. Users should consider themselves lucky to get any programs at all and take what they get.*
- *Real programmers don't comment their code. If it was hard to write, it should be hard to read.*
- *Real programmers don't write application programs, they program right down on the bare metal. Application programming is for feebs who can't do systems programming.*
- *Real programmers don't eat quiche. Real programmers don't even know how to spell quiche. They eat Twinkies, Coke and palate-scorching Szechwan food.*
- *Real programmers don't draw flowcharts. Flowcharts are, after all, the illiterate's form of documentation. Cavemen drew flowcharts; look how much it did for them.*
- *Real programmers don't read manuals. Reliance on a reference is a hallmark of the novice and the coward.*
- *Real programmers programs never work right the first time. But if you throw them on the machine they can be patched into working in only a few 30-hours debugging sessions.*
- *Real programmers don't use Fortran. Fortran is for wimpy engineers who wear white socks, pipe stress freaks, and crystallography weenies. They get excited over finite state analysis and nuclear reactor simulation.*
- *Real programmers don't use COBOL. COBOL is for wimpy application programmers.*
- *Real programmers never work 9 to 5. If any real programmers are around at 9 am, it's because they were up all night.*
- *Real programmers don't write in BASIC. Actually, no programmers write in BASIC, after the age of 12.*
- *Real programmers don't document. Documentation is for simps who can't read the listings or the object deck.*
- *Real programmers don't write in Pascal, or Bliss, or Ada, or any of those pinko computer science languages. Strong typing is for people with weak memories.*
- *Real programmers know better than the users what they need.*
- *Real programmers think structured programming is a communist plot.*
- *Real programmers don't use schedules. Schedules are for manager's toadies. Real programmers like to keep their manager in suspense.*
- *Real programmers think better when playing adventure.*
- *Real programmers don't use PL/I. PL/I is for insecure momma's boys who can't choose between COBOL and Fortran.*
- *Real programmers don't use APL, unless the whole program can be written on one line.*
- *Real programmers don't use LISP. Only effeminate programmers use more parentheses than actual code.*
- *Real programmers disdain structured programming. Structured programming is for compulsive, prematurely toilet-trained neurotics who wear neckties and carefully line up sharpened pencils on an otherwise uncluttered desk.*
- *Real programmers don't like the team programming concept. Unless, of course, they are the Chief Programmer.*
- *Real programmers have no use for managers. Managers are a necessary evil. Managers are for dealing with personnel bozos, bean counters, senior planners and other mental defectives.*
- *Real programmers scorn floating point arithmetic. The decimal point was invented for pansy bedwetters who are unable to "think big."*
- *Real programmers don't drive clapped-out Mavericks. They prefer BMWs, Lincolns or pick-up trucks with floor shifts. Fast motorcycles are highly regarded.*
- *Real programmers don't believe in schedules. Planners make up schedules. Managers "firm up" schedules. Frightened coders strive to meet schedules. Real programmers ignore schedules.*
- *Real programmers like vending machine popcorn. Coders pop it in the microwave oven. Real programmers use the heat given off by the cpu. They can tell what job is running just by listening to the rate of popping.*
- *Real programmers know every nuance of every instruction and use them all in every real program. Puppy architects won't allow execute instructions to address another execute as the target instruction. Real programmers despise such petty restrictions.*
- *Real programmers don't bring brown bag lunches to work. If the vending machine sells it, they eat it. If the vending machine doesn't sell it, they don't eat it. Vending machines don't sell quiche.*
- *Real programmers know that the word is disk, not disc. Disc is a definite commie plot put forth by blubbering quiche eaters.*

Neues von KEIL

Entwicklungs-Software, alle Preise excl. MWST

Hermann Sailer

C166 Compiler Version 3 - mit Windows IDE

Der C-Compiler ist komplett überarbeitet und verbessert worden bezügl. Optimierung, Warnungs-Meldungen und die Unterstützung der 167/165 Derivate. DScope für Windows simuliert vollständig die Peripherie des 166,167 und 165. Wie in den neuen 8015 und 251 Produkten ist auch µVision enthalten und eine integrierte Entwicklungsumgebung mit Projekt-Verwaltung für Windows 3.1 und Windows 95. Version 3 hat eine komplett neue Dokumentation in Buchform mit mehr als 900 Seiten sowie Windows On-line-Hilfe.

Demodisk	PK166	KOSTENLOS
PK166-Lite		öS 2.800,-
C166 Prof. Developers Kit DOS & Win, C166 Compiler, A166 Assembler, Utilities, dScope166 für Windows, µVision für Windows, 4 kByte Code Limit		
MBC167C	Prototypen Baugruppe mit C167	öS 4.200,--
MC167C-Starter Kit	PK166-Lite + MCB167C	öS 5.460,--

C251 Compiler Version 2 - mit Windows IDE

C251 Version 2 besitzt das Front End der C166 Version 3 und einen komplett überarbeiteten Code Generator.

Demodisk	DK251	KOSTENLOS
DK251-Eval		öS 1.750,--
C251-Compiler, A251 Assembler, dScope-251 für Windows, Utilities, µVision für Windows, 2 kByte Codelimit		
MCB251SB-Starter Kit		öS 4.200,--
MCB251 Baugruppe, Handbuch, DK251-Eval		

C51 Compiler Version 5.1

Die neue C51 Version 5.1 enthält Support für Intel 151 uns Siemens 515C, sowie Verbesserungen im Detail. Version 5.0 Benutzer können Version 5.1 kostenfrei vom BBS laden (Tel.: 089-4606-286)

Demodisk	PK51	KOSTENLOS
PK51-Lite		öS 2.800,--
C51-Compiler, dScope51 & µVision für Windows, 2 kByte Codelimit		
PK51-Eval		öS 1.750,--
wie PK51-Lite, jedoch mit reduziertem Handbuch		
MCB-517A	Prototypen-Baugruppe	öS 2.660,--
MCB-517A-Starter Kit		öS 4.200,--
PK51-Eval + MCB-517A Baugruppe		

Sie haben Internet-Zugang ?

Dann los - Surfen Sie durch die Keil Internet Home Page

http://www.keil.com

Von dort können Sie direkt die Demo-Versionen herunterladen. Weitere Informationen bei:

WALTER REKIRSCH

Elektronische Geräte Ges.m.b.H. & Co. Vertriebs-KG.

1220 Wien, Obachgasse 28

NEU ab März 1994: Tel. 259 72 70-0, Fax 259 72 75

Sitz: Wien - FN5699g - Handelsgericht Wien

Embedded MS-DOS & Windows

Lizenzen für eingebettetes MS-DOS

Franz Fiala

Andreas Pfeiffer beschreibt es in seinem Beitrag und die PCNEWS berichteten auch schon in früheren Elektronik-Ausgaben über die besonderen Probleme der Cross-Compiler, die man für Embedded Systems braucht.

Eine Möglichkeit, dieses Problem zu umgehen, ist ein guter Compiler, ein anderer, es gar nicht erst entstehen zu lassen, denn konstruiert man das System mit Standard-CPUs der 80x86-Serie und hält man sich dabei an grundlegende Kontruktionsmerkmale eines Industriestandard-PC (was durch Chip-Sätze praktisch immer gegeben ist,) liegt es nahe, ein Standard-BIOS zu verwenden. Nicht viel weiter ist es dann, auch gleich MSDOS oder gar Windows einzubetten.

Ein wesentliches Problem bei der Herstellung romfähiger Software sind Anpassungen an die jeweiligen Systeme, da in ROM-basierten Systemen keine selbstmodifizierenden Codeteile ablaufen können, sie sind ja im (im Gegensatz zum PC) im ROM. Daher bietet Microsoft spezialisierte Versionen der Betriebssysteme für Controlleranwendungen an.

Annabooks, ein Verlag aus den USA, bietet eine große Auswahl von Soft- und Hardware inklusive Microsoft-Lizenzen an.

Eingebettetes BIOS

Annabooks liefert sowohl XT- als auch AT-BIOS-Kits in Source-Code-Form an, die im Prinzip lauffähig sind. Adaptierung an aktuelle Hardware ist leicht möglich. Große Teile des BIOS sind in C geschrieben. Diese Kits habe schon eine mehrjährige Entwicklung hinter sich und heißen jetzt AnnaBios. Mit dem Zusatzprodukt PromKit ist man in der Lage, Diskimages in (Flash)-Rom-Speichern anzulegen. Dieses Betriebssystem im Speicher verhält sich wie Laufwerk A:.

Eingebettete Betriebssysteme

Je nach Komplexität der Applikation und verfügbarem Speicherplatz kann man zwischen MS-DOS ROM-Versionen wählen, die dem bekannten MS-DOS 3.3, 5.0 und DOS 6.22 entsprechen. Weiters gibt es auch Windows 3.1 ROM-fähige Version (Flash File System 2). Während man im PC-Sektor immer eher zur neueren Version schielt (damit ja nur alle Programme laufen), ist es bei Betriebssystemen eher umgekehrt: man nimmt das kleinste und damit meist das älteste System, mit dem die Applikation läuft, denn man spart damit eine Menge an Speicher. ROM-fähige Betriebssysteme inkludieren jeweils 10 Lizenzen.

Echtzeitkern

Den Anforderungen der Embedded Systems entsprechend, liefert Annabooks auch einen Rom-fähigen Echtzeitkern für MS-DOS-Taskswitching. Da der komplette Quellkode mitgeliefert wird, gibt es keine weiteren Lizenzprobleme.

Preisbeispiele

MS-DOS 6.22 ROM version	$ 695,-
MS-DOS 5.0 ROM version	$ 695,-
MS-DOS 3.22 ROM version	$ 595,-
Windows 3.1 ROM version	$ 895,-
Flash File System	$ 195,-
PromKit	$ 179,-
AnnaBios	$ 995,-

Lieferfirma

Annabooks
11838 Bernardo Piazza Court
San Diego
CA 92128-2414
USA

Tel	+1-619-673-0870
Fax	+1-619-673-1432
BBS	+1-619-673-1773 (CATALOG.EXE downloaden)
E-Mail	73204.3405@compuserve.com
URL	http://www.annabooks.com/ana/

Die richtige Wahl

Die Auswahl von Entwicklungswerkzeugen für Embedded Systeme

Andreas Pfeiffer

Die richtige Auswahl von Werkzeugen für die Entwicklung von Software für Embedded Systeme ist entscheidend für den Ablauf und die Qualität des Entwicklungsprozesses. Darüber hinaus müssen die gewählten Werkzeuge auf bestimmte Entwicklungsstufen optimal abgestimmt sein und können nicht immer universell eingesetzt werden. Dieser Artikel beleuchtet den Entwicklungsprozeß und versucht für die einzelnen Entwicklungsschritte herauszustreichen, welche Werkzeuge genau passen, wie man passende Werkzeuge findet, welche Parameter und Optionen berücksichtigt werden müssen und wie es um die Integration der Werkzeuge steht.

Stufenweiser Entwicklungsprozeß

Der Entwicklungsprozeß kann in einzelne Stufen eingeteilt werden. Man kann die einzelnen Werkzeuge als zusammenhängende Kette betrachten, wobei jedes Glied dieser Kette mit dem nächsten eng verbunden ist. Bild 1 zeigt diese Kette vom Hochsprachencompiler über Assembler, Linker bis zu den verschiedenen Debugger-Varianten.

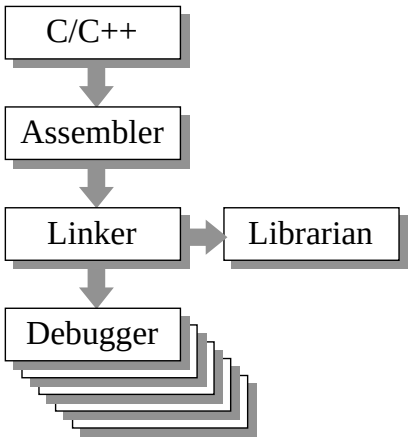


Bild 1: Entwicklungswerkzeuge

In dieser Kette gibt es zwei große Teile:

- Die **Build-Tools**: Werkzeuge zum Übersetzen von Quellkode in ausführbaren Code
- Die **Debug-Tools**: Werkzeuge zum Testen und zur Beseitigung von Fehlern

Die Build-Tools sollen unabhängig von der Entwicklungsumgebung sein und sämtliche Optionen und Parameter müssen zu gleichen Endergebnissen führen. Unterschiedliche Debugger-Varianten dürfen keinen Einfluß auf den Ablauf und das Erstellen der Applikation haben. Es ist daher nicht nötig je nach Umgebung unterschiedliche Bibliotheken ein zu müssen. Notwendige Änderungen am Quellkode um diesen für die verschiedenen Stufen des Entwicklungsprozesses anzupassen sind immer fehleranfällig und daher nicht erwünscht.

Wichtige Eigenschaften des Compilers

Die klassische Embedded-Programmierung erfolgt in C. Mit steigender Popularität von objektorientierter Programmierung kommt immer mehr die Sprache C++ zum Einsatz. Compiler für Cross-Entwicklungen können nicht nach den gleichen Kriterien ausgewählt werden wie Native-Compiler.

- Programmiersprache nach Standard: Um Portabilität zu gewährleisten muß ein Compiler nach ANSI-C-Standard arbeiten. Für ältere Programmteile kann auch Kompatibilität zu K&R-C gewünscht sein. Bei C++ stellt AT&T V2.1 einen weit verbreiteten Standard dar. Neue C++ Compiler werden konform zum Annotated Reference Manual (Stroustrupp) als Vorstufe zu ANSI-C++ entworfen.
- Bibliotheken: Ein Schwachpunkt der ANSI-C-Laufzeitbibliothek ist, daß viele der enthaltenen Funktionen nur bedingt echtzeitfähig sind. Dies kann bei der Entwicklung von Embedded Systemen zu erheblichen Einschränkungen führen. Nachdem an einzelne Bibliotheksfunktionen sehr hardwarenahe Anforderungen gestellt werden (z.B. printf), ist auch der Zugriff auf den Quellkode der Bibliotheksroutinen wünschenswert.
- Objektorientierte Programmierung ermöglicht einmal erstellten Code wiederzuverwenden. Ein guter C++ Compiler sollte durch eine leistungsfähige Standard-Klassenbibliothek ergänzt werden.
- Unterstützung von Prozessorfamilien: Man wird einen Cross-Compiler auswählen, der genau den gewählten Zielprozessor unterstützt. Zukünftige Erweiterungen basieren oft auf Prozessoren, die der gleichen Prozessorfamilie angehören. Ein Compiler für eine Prozessorfamilie stellt eine sichere Investition auch für die Zukunft dar. Es sollten jedoch nicht nur die Gemeinsamkeiten aller Prozessoren unterstützt werden sondern auch die ganz speziellen Eigenschaften der unterschiedlichen Prozessor-Derivate. Der erzeugte Code wird optimal an den gewählten CPU-Typ angepaßt.
- Support: Die Qualität eines Produktes soll auch an der Unterstützung des Herstellers nach dem Kauf gemessen werden.

Erweiterungen für Embedded Systeme

Ein Cross-Compiler ist wesentlich komplexer als ein native Compiler. Bei der Entwicklung des Compilers können sehr wenige Aussagen darüber getroffen werden, wie das endgültige Zielsystem aufgebaut sein wird. Der Hersteller muß zur Anpassung flexible Einstellmöglichkeiten implementieren.

Die Speicherkonfiguration unterscheidet sich bei Embedded-Hardware wesentlich von einer klassischen PC-Umgebung. Einfachste Systeme bestehen zumindest aus ROM für Programm-Code und RAM für Variablen. Ein Embedded-Compiler muß in jedem Fall Rom-fähigen Code generieren können, der streng vom Datenbereich getrennt ist. Anforderungen dieser Art liegen jedoch noch meist viel höher (ROM, RAM, FLASH, EEPROM, NOVRAM,...siehe Bild 2).

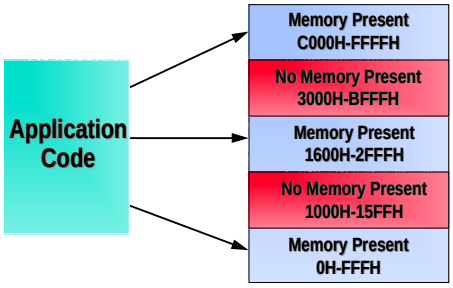


Bild 2: Aufteilung einer Applikation im Speicher

Ein wichtiger Punkt betrifft ein Sprachkonstrukt von C. In C vergibt der Programmierer einen Initialisierungswert an eine statische Variable. Dadurch vermeidet man komplexe Initialisierungsroutinen für jene Variablen, die bereits zur Übersetzungszeit auf bekannten Adressen liegen. In der Programmdatei werden diese Variablen auf ihren Startwert gesetzt und samt Programm in den Speicher des Systems geladen. In einem System, indem das Programm bereits fix im ROM steht, kann dieser Mechanismus nicht funktionieren. Man könnte diese Situation auf drei Arten umgehen:

- Statische Variablen werden nicht initialisiert
- Statische Variablen werden als Konstanten im ROM abgelegt.
- Beim Programmstart werden die Initialisierungswerte aus dem ROM ins RAM kopiert.

In C und C++ ist der direkte Zugriff auf den Speicher erlaubt. Darum sind diese Sprachen für hardwarenahe Entwicklungen sehr populär. Hier darf ein Compiler zu keinerlei Einschränkungen führen.

Speicher ist oft nur sehr begrenzt verfügbar. Der Cross-Compiler muß in diesem Bereich effizient beeinflußt werden können. Mit dem Schlüsselwort *packed* kann man z.B. Felder und Strukturen platzsparender definieren. Diese Einsparungen führen andererseits oft zu längeren Zugriffszeiten.

Das Schlüsselwort `volatile` ist eine Erweiterung zum ANSI-Standard. Eine derart definierte Variable kann sich während des Programmablaufes beliebig ändern. Der Compiler darf den Zugriff auf diese Variable daher nicht optimieren. Volatile Variablen können z.B. für den Zugriff auf I/O-Register definiert werden. Die Sprache C++ beinhaltet dieses Schlüsselwort nicht. Manche Compiler bieten diese Spracherweiterung trotzdem an und sind daher für Embedded-Entwicklungen besser geeignet.

Trotz Hochsprache sollte es auch möglich sein Programmteile in Assemblersprache zu implementieren. Assemblercode verwendet man meist um die letzten Nanosekunden aus einem Prozessor zu holen. Er bietet aber auch einfachen Zugriff auf Prozessor-Resourcen, die nicht direkt durch C-Konstrukte unterstützt werden (z.B.: Interrupt enable/disable). Aus Effizienz- und Portabilitätsgründen sollte natürlich soweit wie möglich auf den Einsatz von Assembler-Code verzichtet werden. Ein Compiler sollte aber so flexibel sein, auch diese Möglichkeit der Programmierung zu unterstützen und erlauben, beliebige Assemblerzeilen in das Hochsprachenprogramm einzufügen.

Echtzeitanforderungen

Viele Embedded Anwendungen müssen echtzeitfähig sein. Auch die Build-Tools müssen dies unterstützen. In Echtzeitsystemen werden zumeist Interrupt Service Routinen verwendet, die in C bzw. C++ z.B. über ein Schlüsselwort (*interrupt*) zu markieren sind. Die Funktion wird dann mit Codeteilen zum Sichern von relevanten Registern versehen und ein spezieller Rückkehrbefehl wird am Ende angefügt.

Codeteile in Echtzeitsystemen können mehrfach in verschiedenen Programmteilen verwendet werden. Dieser Code muß wiedereintrittsfähig (reentrant) geschrieben sein. C und C++ ermöglichen es derartigen Code zu generieren, da Daten- und Programmteile zusammengehörend definiert werden können. Manche Standard-Bibliotheksfunktionen sind jedoch nicht wiedereintrittsfähig definiert und müssen daher mit besonderer Vorsicht verwendet werden.

Optimierungstechnologie

Moderne Compiler bauen auf Optimierungstechnologien zum Erzeugen von qualitativ hochwertigem Code auf und können durchaus mit handoptimierten Assemblerprogrammen konkurrieren was Größe und Effizienz betrifft. Optimierungen können sich dabei lokal auf eine Funktion oder global auf ein Modul beziehen.

Viele Optimierungen können prozessorunabhängig eingesetzt werden. Interessanter sind jedoch jene Optimierungsarten, die Vorteile aus den ganz speziellen Eigenschaften eines Zielprozessors ziehen. Dies betrifft z.B. die Reihung von Instruktionen, Inline-Funtionen und die switch-Anweisung.

Die Reihenfolge, in der ein Prozessor Instruktionen abarbeitet kann großen Einfluß auf Ausführungszeit und Auslastung des Prozessors haben. Grundlegend ist dies bei RISC-Architekturen, es gilt aber auch bei CISC-Prozessoren.

Unter In-Lining versteht man das direkte Einfügen von kleineren Funktionsblöcken ohne diese über Unterprogrammaufrufe zu aktivieren. Die Ausführungszeiten werden dadurch verkürzt. Komfortable Compiler wählen Funktionen automatisch für In-Lining aus.

Bei der `switch`-Anweisung von C hängt es sehr von den Werten und der Abfolge der Verzweigungen ab, wie optimaler Code zu generieren ist. Explizite Ausführung von Testabfragen, Implementierung von Lookup-Tables oder indizierte Sprungtabellen können dabei zielführend sein. Auch Dummy-Werte für jene Verzweigungen die nicht explizit angegeben wurden können zur effizientesten Lösung führen. Der Compiler kann bei Änderungen den passenderen Mechanismus verwenden und sorgt so für höhere Effizienz als die direkte Programmierung in Assembler.

Die Möglichkeit der Feinabstimmung des Compilers auf die Zielplattform ist eine Grundvoraussetzung für brauchbare Entwicklungsergebnisse. Vom Benutzer muß auf jeden Fall die Zielrichtung (Optimale Speichereffizienz bzw. optimale Laufzeit) beeinflußbar sein.

Eigenschaften des Debuggers

Nachdem ein Programm fehlerfrei durch den Compiler übersetzt wurde, wird im nächsten Schritt überprüft, ob sich die Applikation auch wie geplant verhält. Dies gilt generell und nicht nur für die Entwicklung von Embedded Systemen. Hier gibt es jedoch besondere externe Einflüsse die zu speziellen Anforderungen bei der Auswahl der Werkzeuge führen.

Die wichtigste Forderung ist, daß ein Debugger voll optimierten Code verarbeiten können muß. Obwohl das naheliegend ist, sind noch nicht alle Debugger dafür geeignet. Es muß oft noch gewählt werden ob man optimierten Code oder vollüberprüften Code ausliefern will. Ein Mikroprozessor wird auf Grund seiner Leistungsfähigkeit ausgewählt und der Entwickler verläßt sich auf den Compiler, daß dieser die gewünschte Leistung auch gewährleistet. es ist jedoch nicht zu akzeptieren, wenn der Debugger in diesem Punkt zu Einschränkungen führt.

In der Praxis kann das Überprüfen von optimiertem Code zu einer ziemlichen Herausforderung für den Entwickler werden. Die Ergebnisse von so manchen Optimierungen (z.B.: Verschieben von Codeteilen oder Register Coloring) können es extrem erschweren dem Programmfluß zu folgen. Zur allgemeinen Überprüfung wird man deshalb noch nicht alle Optimierungsoptionen voll ausnützen. Für den endgültigen Test muß der Debugger jedoch vollständig optimierten Code verarbeiten können.

Der Debugger muß auf Hochsprachenebene arbeiten. Die Ausführung des Programms muß Befehl für Befehl und nicht nur zeilenweise erfolgen. Der Zugriff auf Datenbereiche muß angepaßt und bequem möglich sein (Berechnung von Ausdrücken, Dekodieren von Strukturen, Verfolgen von Listen). Andererseits muß auch wahlweise ein maschinennaher Zugriff auf Code und Daten möglich sein.

Auch bei Einsatz von C++ muß der Debugger auf Hochsprachenebene arbeiten können. Der vom Präprozessor erzeugte C-Zwischencode muß im Hintergrund verdeckt bleiben. Funktionsnamen müssen immer in der C++ Form erscheinen (unmangled).

Auch ein Debugger muß flexibel an die Zielumgebung anzupassen sein. Bei einem Debugger kann die Funktionalität über eine Makrosprache erweitert werden um I/O-Einheiten zu modellieren, Testläufe zu automatisieren oder Codebereiche zu patchen.

Die Benutzerschnittstelle des Debuggers beeinflußt in hohem Maße die Produktivität des Entwicklungsprozesses. Die Oberfläche muß leistungsfähig, durchgängig in der Bedienung und leicht erlernbar sein (siehe Bild 3). Wenn Debugger-Varianten in unterschiedlichen Umgebungen eingesetzt werden sollen, muß der Umstieg ohne Lernaufwand erreichbar sein. Eine Debugger-Familie für verschiedene Anforderungen bietet hier die beste Lösung.

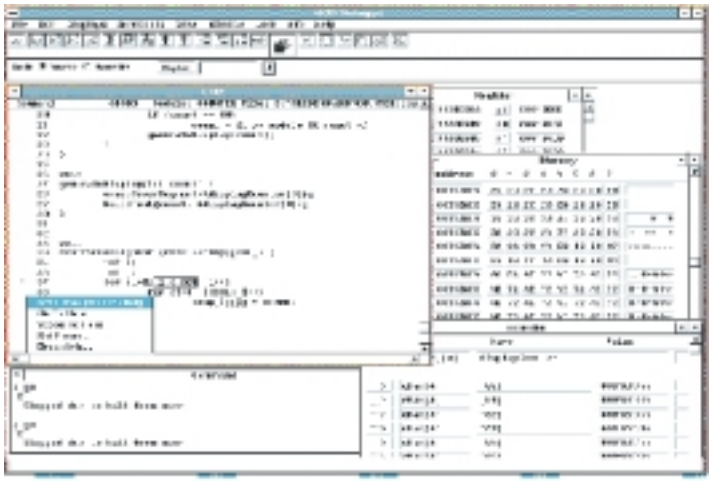


Bild 3: Oberfläche des XRAY-Debuggers von Microtec Research

Entwicklungsphasen

Den Entwicklungszyklus für ein Embedded System kann man in fünf Phasen einteilen (siehe Bild 4). Die Randbedingungen für die Entwicklung ändern sich von Phase zu Phase.

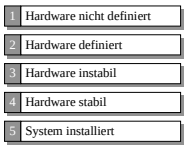


Bild 4: Entwicklungsphasen

In Phase 1 können auch ohne definierte Hardware Softwarealgorithmen und neue Ideen ausgearbeitet werden.

In Phase 2 ist die Hardwarekonfiguration bereits bekannt. Die Softwareentwickler können bereits einen großen Teil der Anwendung im Detail erstellen.

Die Integration von Hardware und Software kann in Phase 3 beginnen, obwohl die Hardware noch nicht 100%ig stabil läuft.

In Phase 4 gibt es stabil lauffähige Hardwareeinheiten zur endgültigen Integration und für Abschlußtests.

Manche Entwicklungsprojekte können bereits vor Phase 5 abgeschlossen werden. Manchmal muß jedoch in Phase 5 ein Feinabgleich bei Inbetriebnahme im Feld vorgenommen werden. Zu einem späteren Zeitpunkt werden ggf. noch Erweiterungen im Feld vorgenommen.

Für jede einzelne dieser Entwicklungsphasen sollte eine angepaßte Version des Debuggers existieren.

Native Debugger

Ein native Debugger (ausführen des Codes in der Host-Umgebung) für die Embedded Entwicklung erscheint zunächst unpassend. Während Phase 1 des Entwicklungsprozesses kann man aber bereits mittels native Debugger die Funktionalität der Algorithmen und Programmteile überprüfen. Bietet ein native Debugger darüber hinaus noch die gleiche Benutzerschnittstelle, wie später die Cross-Debugger-Versionen, so kann dieser Debugger eine optimale und stabile Trainingsplattform zum Erlernen der Werkzeuge sein.

Debugger mit Simulator

Die Prozessor-Simulation auf Instruktionsebene am Hostsystem ist in fast allen Phasen nützlich. Speziell in Phase 2 beschleunigt ein Simulator den Entwicklungsprozeß erheblich.

Mittels Simulator kann die Applikation bereits sehr detailliert getestet werden. Trotz reduzierter Geschwindigkeit kann der Simulator bereits genaue Angaben über die Laufzeiten geben. Kritische Programmbereiche können sehr früh optimiert werden. Mit dem Simulator werden aber auch Informationen generiert, die mit anderen Debugger-Varianten nicht verfügbar sind. Die Analyse der Laufzeiten erfolgt exakt und ohne Overhead. Informationen über die prozentuelle Ausnutzung verschiedener Programmbereiche können auch nur in einer Simulatorvariante generiert werden. Ein leistungsfähiger Simulator muß des weiteren in der Lage sein, über den Prozessorkern hinaus auch Interrupts und I/O-Einheiten nachzubilden.

Debugger mit Emulator-Anbindung

Incircuit-Emulatoren sind sehr leistungsfähige Werkzeuge um Software auf noch instabiler Hardware zu integrieren. Die Applikation läuft mit voller Geschwindigkeit und es können sehr komplexe Breakpoints gesetzt werden. Die Bedienoberfläche des Emulators sollte zu jener der anderen im Projekt eingesetzten Debugger-Varianten kompatibel sein. Ein wichtiger Faktor ist hier die Unterstützung namhafter Emulatorhersteller.

Debugger und Monitor

Ist die Hardware stabil lauffähig, braucht man nicht mehr unbedingt den Leistungsumfang eines teuren Emulators, der ja nebenbei nicht für jeden Entwickler im Team zur Verfügung steht. Hier führen Monitor-Debugger zum Ziel, wobei die Zielhardware über eine Schnittstelle (Ethernet, seriell,...) mit dem Hostsystem verbunden ist. Am Zielsystem läuft ein kleines (<10kByte) Monitorprogramm zur Kommunikation mit dem Host. Man erhält eine billige aber sehr leistungsfähige Debugger-Lösung bei voller Ablaufgeschwindigkeit.

Der Monitor am Zielsystem muß einfach an neue Hardwaregegebenheiten anpaßbar sein. Für Standard-Boards müssen Anpassungen ab Lager verfügbar sein.

Ein Monitor-Debugger wird meist in Phase 4 eines Projekts eingesetzt. Auch wenn die Applikation ausgeliefert wurde kann das Monitorprogramm am Zielsystem verbleiben und erlaubt einfache Erweiterungen und Anpassungen im Feld z.B. mittels Notebook-Computer.

Debugger für Echtzeitsysteme

Mit steigender Komplexität der Applikationen werden vermehrt Echtzeitbetriebssysteme eingesetzt. Dafür müssen auch im Debugger funktionelle Erweiterungen vorgesehen werden:

- Das Debuggen muß taskabhängig vorgenommen werden können. Ein Breakpoint muß abhängig von einer Task gesetzt werden können. Dies ist dann wichtig, wenn einzelne Programmteile in mehreren Tasks Verwendung finden. Auch Datenbereiche müssen als zu einer Task gehörend identifiziert werden können.
- Informationen über die Betriebssystemumgebung wie z.B. Task Status oder Kommunikationsmechanismen usw. müssen extrahiert und dargestellt werden können.

Wenn ein selbsterstelltes Betriebssystem verwendet wird, muß auch dieses in eine leistungsfähige Debugger-Umgebung einfach zu integrieren sein.

Debugger mit Hardwareunterstützung

Der Preis und die Komplexität von Hardware-Emulatoren steigt mit der Leistungsfähigkeit der heutigen Mikroprozessoren. Daher werden von den Halbleiterherstellern immer mehr Funktionsblöcke auf den Prozessorchips integriert, die den Zugriff eines Debuggers in die Interna des Prozessors erleichtern. Das Spektrum reicht hier von der direkten Unterstützung mittels Breakpoint-Register (Adress- und Datenkomparatoren) bis zu speziellen Betriebsmodi um den Prozessor in einen Debug-Mode zu setzen.

Ein gutes Beispiel für den letzten Punkt ist der sogenannte Background Debug Mode (BDM) von Motorola, welcher in den Prozessoren der Serie 683xx (CPU32) implementiert wurde. Wird der Prozessor in diesen Modus gesetzt, so kann der Debugger über eine eigene Schnittstelle Register und Speicherstellen lesen und schreiben. Die Anbindung an ein Host-System wurde einfach gehalten und bedeutet kaum Mehraufwand für die Systementwicklung. Die Hardwareunterstützung bietet emulatorähnliche Möglichkeiten zu sehr geringen Kosten jedoch ohne zusätzlich notwendigem Monitorprogramm am Zielsystem.

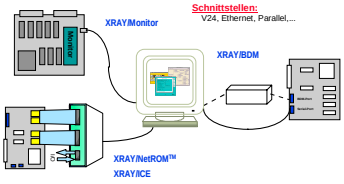


Bild 5: In-Circuit-Varianten des XRAY-Debuggers von Microtec Research

Einflüsse auf die Auswahl

Die Auswahl der Entwicklungswerkzeuge ist natürlich nur einer aus vielen Entscheidungsprozessen beim Entwurf neuer Systeme. Weitere Entscheidungen betreffen den Zielprozessor, den Entwicklungsrechner und das Echtzeitbetriebssystem. Diese Entscheidungen greifen ineinander und sollten nicht isoliert betrachtet werden.

Zielprozessoren

Es gibt viele gute Gründe sich für einen bestimmten Prozessor zu entscheiden. Natürlich sollte nicht außer Acht gelassen werden, wie es um die Unterstützung durch Entwicklungswerkzeuge steht. Selten wird nur ein Hersteller die vollständige Palette zur Entwicklung eines Prozessors anbieten. Sind für einen Prozessor nur sehr eingeschränkt gute Werkzeuge am Markt zu finden, so sollte man diesen Typ eher in Frage stellen.

Entwicklungsrechner

Eigentlich kann man zur Programmentwicklung jeden beliebigen Computer verwenden. Man kann sich auch für jene Werkzeuge entscheiden, die für einen bestimmten Computer portiert wurden. Der Einsatz von standardisierten Systemen bietet jedoch die größte Sicherheit auch in Zukunft mit Werkzeugen und Updates versorgt zu werden. Hersteller, die ihre Werkzeuge auf sehr vielen unterschiedlichen Plattformen anbieten sollten genau bezüglich Support überprüft werden. Oft leidet die Pflege der Portierung auf exotischen Computern unter den Versionen auf populären Plattformen.

Echtzeitbetriebssysteme

Hier wäre ein eigener Artikel angebracht. Der Blick sollte bei der Auswahl durchaus auch auf die Unterstützung durch Entwicklungswerkzeuge gelenkt werden. Ein Betriebssystem sollte in jedem Fall eine flexible offene Architektur bieten. Verschiedene Build-Tools sollen eingesetzt werden können. Passende Debug-Tools müssen natürlich auch zur Verfügung stehen. Oft wird daran gedacht, ein eigenes Betriebssystem zu entwickeln. Man muß jedoch berücksichtigen, daß dann auch für die Anbindung an die Entwicklungswerkzeuge gesorgt werden muß.

Zusammenfassung

Die Auswahl von Software-Entwicklungswerkzeugen für Embedded Systeme ist keine leichte Aufgabe. Viele Anbieter bieten Teillösungen an, manche verfügen über komplett abgestimmte Produktserien. Microtec Research hat eine umfassende Palette an Werkzeugen um diese Entscheidung zu erleichtern. Die Compiler und XRAY Debugger von Microtec Research erfüllen alle oben genannten Forderungen. Die VRTX-Echtzeitbetriebssysteme bieten eine offene, abgestimmte Architektur für härteste Echtzeitanforderungen. Die integrierte Entwicklungsoberfläche XRAY MasterWorks bietet eine Plattform zur Integration aller Werkzeuge, auch jener von Drittherstellern. Die Spectra-Software-Backplane ermöglicht größtmögliche Flexibilität zur Entwicklung von Echtzeitapplikationen durch eine durchgängige offene Architektur. □

Mehr als nur ein Cross-Debugger

Andreas Pfeiffer

Die steigende Komplexität von Embedded Applikationen erfordert immer leistungsfähigere Entwicklungswerkzeuge um neue Produkte rechtzeitig auf den Markt zu bringen. Microtec Research ist seit 20 Jahren auf die optimale Unterstützung von Softwareentwicklern in der Embedded Welt spezialisiert und bietet als einziger Hersteller eigene Technologie vom Compiler über unterschiedliche Debugger bis zu Echtzeit-Betriebssystemen. Der XRAY Debugger setzt seit Jahren den Standard bei Cross-Debuggern. Abgestimmt auf die aktuellen Bedürfnisse, ob hostbasierender Simulator oder Cross-Umgebung, sind alle Funktionen implementiert, die einen rascheren Entwicklungszyklus ermöglichen und noch vieles mehr...

Grundbedürfnisse gestreßter Entwickler

Leistungsfähige Standardfunktionen zeitgemäßer Cross-Debugger sind natürlich auch im XRAY-Debugger implementiert:

- Stack-Traceback: Überblick der Stack-Verschachtelung auf Hochsprachenebene und Einsicht auf die lokalen Variablen auf jeder beliebigen Stack-Ebene
- Single-Step: auf Hochsprachen- und/oder Assembler-Ebene
- Flexible Darstellung und Manipulation von Variablen, Objekten, Register und Codebereichen
- Symbolischer Zugriff auf sämtliche Bereiche der Applikation
- Einheitliche, grafische und intuitiv bedienbare Multi-Windows Bedienoberfläche
- Kompatibilität zu Vorgängerversionen
- Viele Varianten mit durchgängiger Bedienphilosophie (Simulator, Incircuit-Monitor, Incircuit-Emulator, BDM)

Embedded Systeme laufen fast immer stand-alone und müssen daher extrem robust konstruiert werden. Viele gängige Debugger haben den Nachteil, daß sie nur un- bzw. halboptimierten Objektcode des Compilers verarbeiten können, weil z.B. Codeverschiebungen oder Registerlebenszyklen nicht exakt nachverfolgt werden können. Mit dem XRAY Debugger kann immer volloptimierter Objektcode verarbeitet werden. Unliebsame Überraschungen gegen Ende des Entwicklungszyklus werden so von Anfang an vermieden.

Moderne Applikationen basieren auf vielen tausend Zeilen C bzw. C++ Source-Code. Microtec Research hat sehr viel Know-how investiert, um die Arbeit auf Hochsprachenebene zu vereinfachen. Ein Ergebnis daraus ist das Inspektor-Fenster zur übersichtlichen Darstellung von komplexen Datenstrukturen. Die automatische Verarbeitung von dynamischen Datenstrukturen (Listen, Bäume etc.) hilft fehlerhafte Verweise und Einträge sehr einfach zu finden.

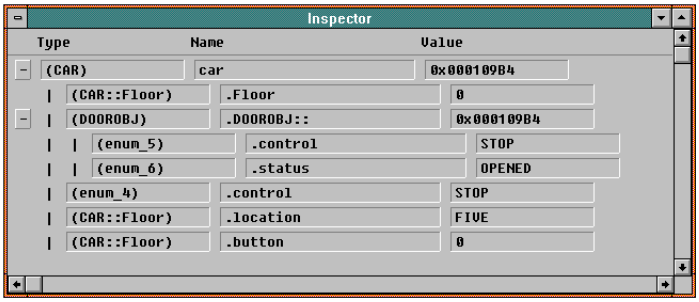


Bild 1: **Das XRAY Inspektor-Fenster erlaubt die einfache Darstellung und Manipulation von komplexen Datenstrukturen**

Eine sehr nützliche Eigenschaft von XRAY betrifft die Preprozessor-Unterstützung. Werte, die durch **#define** zugewiesen wurden, können wie Variablen dargestellt werden. Das lange Suchen nach der aktuellen Zuweisung in vielen Header-Dateien hat somit ein Ende.

Einzigartig bei XRAY ist, daß das schrittweise Durchlaufen der Applikation befehlsorientiert anstatt zeilenorientiert erfolgt. **if**, **while** und **do** Konstrukte werden exakt durchschritten und Breakpoints können in der entsprechenden Spalte auf Hochsprachenebene gesetzt werden. Bei vielen Debuggern muß noch auf Assembler-Ebene gewechselt werden, um Breakpoints z.B. auf den Bedingungs-Teil einer **for**-Schleife zu setzen.

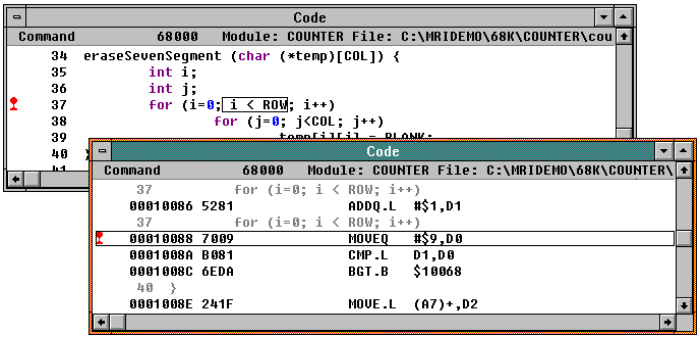


Bild 2: **Der XRAY-Debugger arbeitet befehlsorientiert. Breakpoints können auch innerhalb von for-Anweisungen gesetzt werden.**

Breakpoints: Wie hätten Sie's denn gerne?

Einfache Instruktions-Breakpoints sind die Grundvoraussetzung für die Arbeit mit einem Debugger. Breakpoints bei Speicherzugriffen sollten auch nach Schreib- oder Lesezyklus unterschieden werden können. Natürlich muß auch eine mögliche Hardwareunterstützung durch den Zielprozessors (BP-Register) Verwendung finden.

Die gefürchtetsten Bugs sind jene, die nur sehr selten auftreten. Die Triggerbedingungen in der Umgebung solcher Fehler sind meist sehr komplex. Der XRAY-Debugger kann über eine leistungsfähige Makrosprache Abbruchbedingungen quasi frei programmierbar und beliebig komplex qualifizieren.

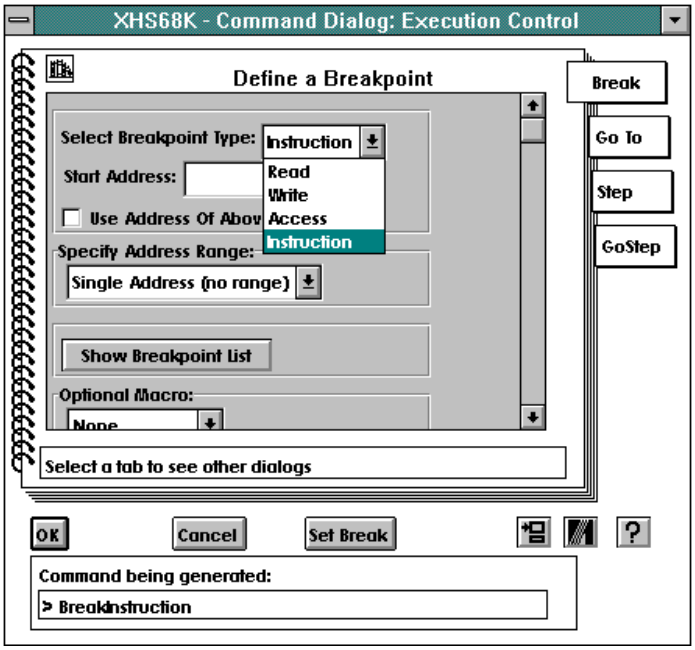


Bild 3: **Mit grafischer Unterstützung durch das Notebook-Interface können auch komplexe Breakpoints sehr einfach gesetzt werden**

Und oft möchte man die Applikation nur schnell bis zu einer bestimmten Zeile laufen lassen. In XRAY werden diese temporären Breakpoints per Point-and-Click angelaufen.

Objektorientierte Programmierung...

Die Popularität von objektorientierten Programmiermethoden macht auch vor Embedded Applikationen nicht halt. Natürlich muß ein leistungsfähiger Cross-Debugger auch in dieser Welt zu Hause sein. So kennt der XRAY Debugger sämtliche aktuellen Spracheigenschaften von C++, wie überladene Operatoren und Funktionen, Konstruktoren und Destruktoren und Vererbung. Soll ein Breakpoint auf eine überladene Funktion gesetzt werden, so bietet XRAY eine Liste der möglichen Funktionen zur Auswahl an. Die exakte oft komplizierte Parameter-Syntax braucht man jetzt nicht mehr im Kopf behalten.

...und wenn's sein muß, auch Assembler

Embedded Applikationen beinhalten nach wie vor auch Assembler-Anweisungen. Im XRAY Debugger kann Hochsprache gemischt mit der Assemblerebene dargestellt werden (Bild 2). Der Debugger sorgt für die genaue Zuordnung der beiden Welten. Anstatt Objektcode einfach nur zu disassemblieren, verwendet XRAY Debug-Informationen die vom Assembler generiert wurden. Die Arbeit auf Maschinenebene ist dadurch wesentlich vereinfacht, weil sie symbolisch unterstützt ist. Das StepOver-Kommando verhindert, daß immer wieder Makros oder Repeat-Blöcke schrittweise durchwandert werden müssen.

Hardware noch nicht komplett?

PC-Programme zu bearbeiten ist einfach, die Hardwareumgebung ist immer gleich. Bei Embedded Applikationen ist die Peripherie jedoch meist sehr unterschiedlich aufgebaut. Ein sehr guter Cross-Debugger muß dem Entwickler die Möglichkeit geben, schon vor Fertigstellung der Hardware Programme möglichst vollständig auszutesten. Mit einem Instruction-Set Simulator, wird der Zielprozessor auf der Host-Maschine nachgebildet. Natürlich muß hier periphere Hardware berücksichtigt werden. XRAY bietet eine Vielfalt an Simulationsmöglichkeiten für Input/Output-Ports oder Interrupts (Bild 4).

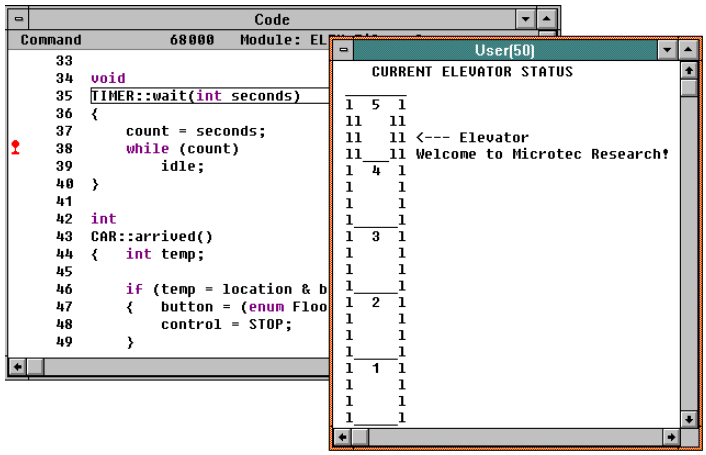


Bild 4: **Hardwarenachbildung im XRAY-Debugger: Die Funktionalität dieser C++ Fahrstuhlsteuerung läßt sich mit Semi-Grafik-Unterstützung noch einfacher überprüfen**

Hardwarenahe Programmteile, die zum aktuellen Zeitpunkt noch nicht so interessant sind, können in XRAY umgangen werden, ohne eine Zeile am Quellkode zu manipulieren.

Umgekehrt können Funktionen der Applikation wie ein Debugger-Kommando einzeln im Zielsystem zur Ausführung gebracht werden (Target-Function-Calls).

Makrosprache und noch viel mehr...

Der XRAY Debugger kann mittels Makrosprache funktionell beliebig erweitert werden. Die Sprache ist nicht kompliziert und neu, sondern basiert auf normaler C-Syntax und kann auf sämtliche Symbole der Applikation zugreifen (BILD5). Makro-Anwendungen reichen von automatisierten Debugging-Läufen, über Test-Scripts, bis zu neuen Debugger-Funktionen. Mittels Makros können Codebereiche einfach auf Debugger-Ebene gepatcht werden, ohne fehleranfällig im Source-Code manipulieren zu müssen.

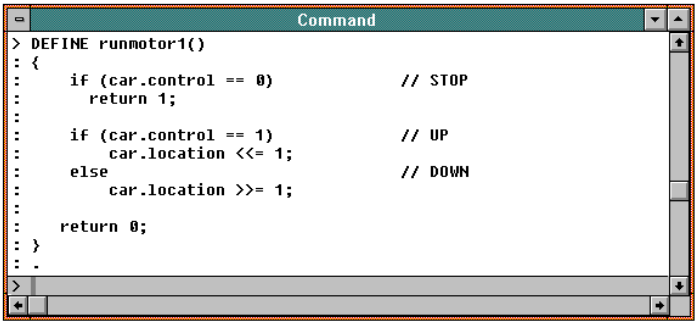


Bild 5: **Die Makro-Sprache ist in simpler C-Syntax gehalten**

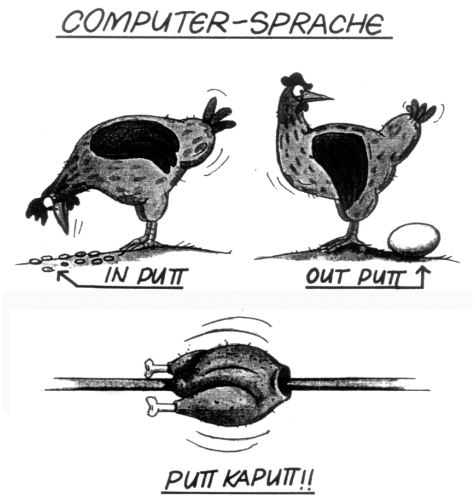
Sicher in die Zukunft

Über Debugger-Features hinaus bietet XRAY aber noch viel mehr Funktionalität, um ein Embedded-Software Design zu optimieren. Stichworte wie Performance- und Code-Coverage-Analyse oder RTOS-System-Level-Debugging wären Themen für eigene umfangreiche Abhandlungen. Der XRAY-Debugger kann auch im Batch-Mode betrieben werden und z.B. im Feld in der Schaltung über längere Zeiten Daten loggen, die dann im Labor z.B. am Simulator ausgewertet werden.

Erhältlich ist der XRAY-Debugger in zahlreichen Simulator- und In-Circuit-Varianten für unterschiedliche Prozessorfamilien, wie Motorola 68K, PowerPC, Intel 80x86-kompatible und läuft auf den Host-Plattformen SUNSparc, HP9000/700 und PC-386/486/Pentium.

Für weitere Auskünfte kontaktieren Sie bitte

Herrn Schuster
Microtec Research GmbH
Haidgraben 1c
D-85521 Ottobrunn/München
Tel.: +49-89-609 00 81



Objektorientierte Programmierung...

Die Popularität von objektorientierten Programmiermethoden macht auch vor Embedded Applikationen nicht halt. Natürlich muß ein leistungsfähiger Cross-Debugger auch in dieser Welt zu Hause sein. So kennt der XRAY Debugger sämtliche aktuellen Spracheigenschaften von C++, wie überladene Operatoren und Funktionen, Konstruktoren und Destruktoren und Vererbung. Soll ein Breakpoint auf eine überladene Funktion gesetzt werden, so bietet XRAY eine Liste der möglichen Funktionen zur Auswahl an. Die exakte oft komplizierte Parameter-Syntax braucht man jetzt nicht mehr im Kopf behalten.

...und wenn's sein muß, auch Assembler

Embedded Applikationen beinhalten nach wie vor auch Assembler-Anweisungen. Im XRAY Debugger kann Hochsprache gemischt mit der Assemblerebene dargestellt werden (Bild 2). Der Debugger sorgt für die genaue Zuordnung der beiden Welten. Anstatt Objektcode einfach nur zu disassemblieren, verwendet XRAY Debug-Informationen die vom Assembler generiert wurden. Die Arbeit auf Maschinenebene ist dadurch wesentlich vereinfacht, weil sie symbolisch unterstützt ist. Das StepOver-Kommando verhindert, daß immer wieder Makros oder Repeat-Blöcke schrittweise durchwandert werden müssen.

Hardware noch nicht komplett?

PC-Programme zu bearbeiten ist einfach, die Hardwareumgebung ist immer gleich. Bei Embedded Applikationen ist die Peripherie jedoch meist sehr unterschiedlich aufgebaut. Ein sehr guter Cross-Debugger muß dem Entwickler die Möglichkeit geben, schon vor Fertigstellung der Hardware Programme möglichst vollständig auszutesten. Mit einem Instruction-Set Simulator, wird der Zielprozessor auf der Host-Maschine nachgebildet. Natürlich muß hier periphere Hardware berücksichtigt werden. XRAY bietet eine Vielfalt an Simulationsmöglichkeiten für Input/Output-Ports oder Interrupts (Bild 4).

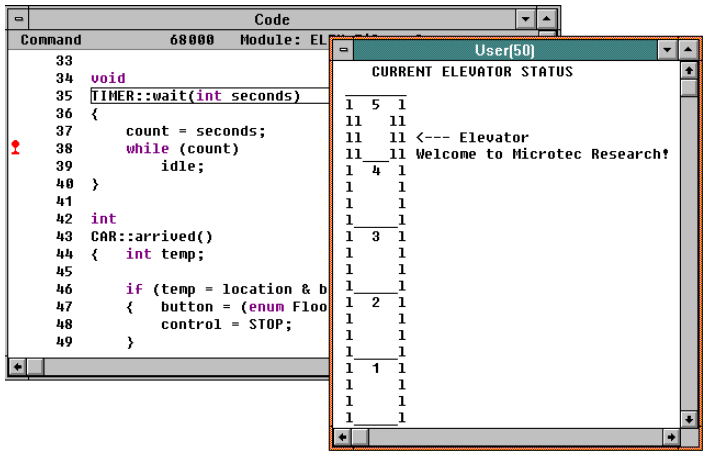


Bild 4: **Hardwarenachbildung im XRAY-Debugger: Die Funktionalität dieser C++ Fahrstuhlsteuerung läßt sich mit Semi-Grafik-Unterstützung noch einfacher überprüfen**

Hardwarenahe Programmteile, die zum aktuellen Zeitpunkt noch nicht so interessant sind, können in XRAY umgangen werden, ohne eine Zeile am Quellkode zu manipulieren.

Umgekehrt können Funktionen der Applikation wie ein Debugger-Kommando einzeln im Zielsystem zur Ausführung gebracht werden (Target-Function-Calls).

Makrosprache und noch viel mehr...

Der XRAY Debugger kann mittels Makrosprache funktionell beliebig erweitert werden. Die Sprache ist nicht kompliziert und neu, sondern basiert auf normaler C-Syntax und kann auf sämtliche Symbole der Applikation zugreifen (BILD5). Makro-Anwendungen reichen von automatisierten Debugging-Läufen, über Test-Scripts, bis zu neuen Debugger-Funktionen. Mittels Makros können Codebereiche einfach auf Debugger-Ebene gepatcht werden, ohne fehleranfällig im Source-Code manipulieren zu müssen.

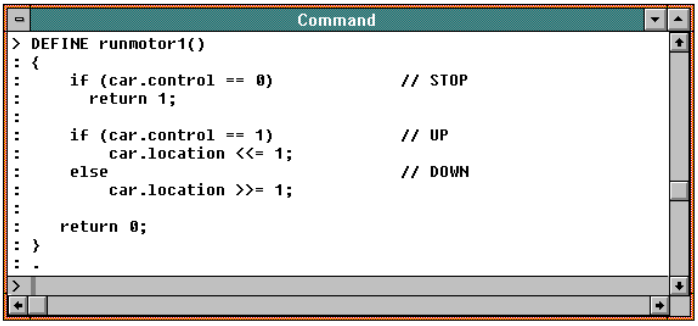


Bild 5: **Die Makro-Sprache ist in simpler C-Syntax gehalten**

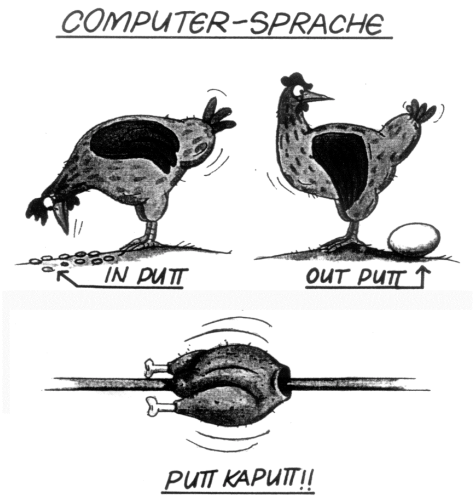
Sicher in die Zukunft

Über Debugger-Features hinaus bietet XRAY aber noch viel mehr Funktionalität, um ein Embedded-Software Design zu optimieren. Stichworte wie Performance- und Code-Coverage-Analyse oder RTOS-System-Level-Debugging wären Themen für eigene umfangreiche Abhandlungen. Der XRAY-Debugger kann auch im Batch-Mode betrieben werden und z.B. im Feld in der Schaltung über längere Zeiten Daten loggen, die dann im Labor z.B. am Simulator ausgewertet werden.

Erhältlich ist der XRAY-Debugger in zahlreichen Simulator- und In-Circuit-Varianten für unterschiedliche Prozessorfamilien, wie Motorola 68K, PowerPC, Intel 80x86-kompatible und läuft auf den Host-Plattformen SUNSparc, HP9000/700 und PC-386/486/Pentium.

Für weitere Auskünfte kontaktieren Sie bitte

Herrn Schuster
Microtec Research GmbH
Haidgraben 1c
D-85521 Ottobrunn/München
Tel.: +49-89-609 00 81



Support für 80C166

- *Handbuch des 80C166 - Reiner Johannis*
  Verlag Siemens, ISBN 3-8009-4203-8. 1. Auflage, Oktober 1993
- *MC-Tools 17, Arbeiten mit C166-Controllern- Karl-Heinz-Mattheis, Steffen Storandt*
  Verlag Otmar Feger, ISBN 3-928434-26-8. 1. Auflage, 1995.

Christoph Baumgartner, Martin Weissenböck

Christoph Baumgartner

1990 startete Siemens München den Versuch den Mikrocontrollerbereich aufs neue zu erobern und reformieren. Da die Familie der 8051 in letzter Zeit den Leistungsanforderungen von neuen komplexen Steuerungen oft nicht mehr gerecht werden konnten, wurde ein neues Konzept gefordert und so entstand eine komplett neue Familie von Mikrocontrollern auf 16 Bit Basis. Durch das absolut neue Design wurde auf eine Abwärtskompatibilität verzichtet, um ein Maximum an Leistung zu erzielen. Vor allem wurde bei der Entwicklung auf die Universalität geachtet, sodaß die Einsatzmöglichkeiten fast uneingeschränkt sind. Schließlich wurde eine Lösung gefunden, die einen Kompromiß zwischen Leistungsfähigkeit und Kosten darstellt. Denn meist ist bei der Entwicklung neuer Steuerungen nicht die exzellenteste Lösung gesucht sondern die, die mit geringstem Preis- und Leistungsaufwand auskommt.

Inzwischen gibt es schon die verschiedensten Derivate aus der 166-er Familie wie zum Beispiel den c167 dessen Adressraum von 256kB (80c166) auf 16 MB erweitert wurde, weiters verfügt er über 16 AD-Kanäle (80c166 10 Kanäle), eine PWM Einheit mehr IO-Ports und Interrupts, und besonders interessant ist auch der Bootstrap-Loader, der nach einem Reset das Programm über die serielle Schnittstelle in den Speicher lädt. Ein weiteres Exemplar ist der c165 der kleine Bruder des c167 der ohne AD-Wandler, PWM Einheit, Capture und Compare, und mit weniger IO-Ports auskommen muß.

Die CPU: Das Herz des 80c166 wurde mit einer beachtlichen Rechenleistung ausgestattet, da bei einer Oszillatorfrequenz von maximal 40Mhz alle 100ns ein Befehl in der vierstufigen Pipeline weitergerückt wird. Dadurch dauern die meisten Befehle genau einen Maschinenzyklus lang, das sind dann 100ns was zu eine sehr theoretischen Rechenleistung von bis zu 10 MIPS führt.

Der Speicher: Das interne RAM hat die Größe von 1 kB, in dem der Stack und die Registerbänke liegen (GPR, General Purpose Register) da es sich beim 166 um eine Registermaschine und nicht um einen Akkumulator CPU handelt, und weiters sind noch jede Menge SFR (Spezialfunktionsregister) untergebracht. Ein weiteres beachtenswertes Merkmal ist das flexible Bussystem des 166 zum Anschluß des externen Speichers. Es stehen 4 verschieden Arten von Speichermodellen zur Verfügung :

- 16 bit Datenbus, Multiplexverfahren
- 8 bit Datenbus, Multiplexverfahren (vom 8031 bekannt)
- 16 bit Datenbus, Demultiplexverfahren
- 8 bit Datenbus, Demultiplexverfahren

Beim Multiplexverfahren werden Daten von Adressen durch ein Latch und das ALE Steuersignal getrennt, beim Demultiplexverfahren gibt es für jede Daten und Adressleitung einen eigenen Pin. Diese Betriebsart muß natürlich unmittelbar nach einem Reset bekannt sein, ebenso muß man dem Bootstrap-Loader mitteilen ob er aktiviert werden soll, dies geschieht durch Pull-Down-Widerstände an den dafür vorgesehen Ports, die aber trotzdem für IO-Zwecke zur Verfügung stehen.

Das Interruptsystem: Der 166 hat einen 16-stufigen Prioritätssystem das 32 Interruptquellen verwaltet (der c167 noch um einiges mehr), weiters gibt es 8 Traps für interne Fehler (z.B.: Stacküberlauf, Division by Zero, usw.)

Parallele Ports: Zur Ein- und Ausgabe stehen 76 bitadressierbare Pins zur Verfügung

Serielle Ports: Zur Kommunikation mit anderen Einheiten besitzt der 166 2 identische serielle Schnittstellen die entweder synchron oder asynchron im Vollduplexbetrieb arbeiten können.

Die Timer-Einheit (GPT) und Capture-/Compare: Diese beiden Einheiten bieten eine solche Vielzahl von Möglichkeit das ihre Programmierung am Anfang sicher bei einigen leichte Verzweiflungserscheinungen hervorruft, doch mit ein paar guten Blockschaltbildern kommt man doch bald zu passablen Ergebnissen.

Natürlich wurde bei der Entwicklung auch auf einen programmierbaren **Watchdog-Timer** und auf **Stromsparmodi** nicht vergessen.

Wer genaueres zu diesem Prachtexemplar wissen möchte und wer sich mit der Entwicklung eines Systems mit einem Prozessor dieser Familie befaßt, sollte sich das Buch MC-Tools 17 - Arbeiten mit C166-Controllern von Karl Heinz Mattheis / Steffen Storandt-Verlag Otmar Feger zulegen, in diesem Buch gibt's genauere Details zu Hardwareentwicklung mit Berechnungsbeispielen, auf einer beigelegten Diskette sind einige brauchbare Excel-Files mit denen man diverse Ablaufzeiten für die verschiedenen Taktfrequenzen berechnen kann. Auch zur Programmierung gibt's einige Praktische Beispiele in C und Assembler.

Ein weiteres Buch zum selben Thema gibt's natürlich auch von Siemens, daß ein inhaltlicher Zusammenhang besteht ist schon am Cover ersichtlich. Auch das Handbuch des 80C166 von Reiner Johannis - Verlag Siemens bietet einen guten Überblick über die breite Funktionspalette jedoch gibt's in dieser Ausführung keine praktischen Beispiele, sondern nur Theorie.

Natürlich ist für jeden Entwickler auch das Original-Datenbuch unumgänglich, denn gelegentlich gibt es Produktspezifikationen und kleine Änderungen, die in den Büchern natürlich nicht angeführt werden können.

Alles in allem gibt das ca. 2,8 kg Papier (mit dem Buch siehe unten ca. 3,8kg) mit dem man für fast alle Anwendungen bestens gerüstet ist.

Speziell für die Derivate 165 und 167 gibt's aus dem Franzis-Verlag das Buch 80C166 Mikrocontroller -c167 c165 von Schultes/Pohle in dem auf die erweiterten Funktionen eingegangen wird, auf der beigelegten Demodiskette befinden sich einige nützliche Programme und Demoversionen von C und Assempler-Compilern für einen ersten Test.

Für diejenigen Entwickler die auf der sicheren Seite arbeiten gibt's zur Simulation auch VHDL-Modelle, mit denen man eine komplette Entwicklung auf einem entsprechend leistungsfähigen Rechner softwaremäßig durchtesten und simulieren kann. ☐

Martin Weissenböck

1990 startete Siemens München den Versuch, den Mikrocontrollerbereich aufs neue zu erobern und reformieren. Da die Familie der 8051 in letzter Zeit den Leistungsanforderungen von neuen komplexen Steuerungen oft nicht mehr gerecht werden konnten, wurde ein neues Konzept gefordert. Es entstand eine komplett neue Familie von Mikrocontrollern auf 16 Bit Basis.

Bei den Prozessoren 8048, 8049, 8051 usw. wurde noch Wert darauf gelegt, daß bestehende Programme weiterverwendet werden können (Abwärtskompatibilität). Durch das absolut neue Design wurde darauf verzichtet, um ein Maximum an Leistung zu erzielen. Vor allem wurde bei der Entwicklung auf die Universalität geachtet, sodaß die Einsatzmöglichkeiten fast uneingeschränkt sind. Damit kein Mißverständnis entsteht: der 80C166 ist immer noch ein Steuerungsprozessor und wird daher einem Pentium keine Konkurrenz machen. Eine „eierlegender Woll-Milch-Sau-Prozessor“ wäre aber auch nicht wirtschaftlich. Mit dem 80C166 wurde als Lösung ein guter Kompromiß zwischen Leistungsfähigkeit und Kosten gefunden. Denn meist ist bei der Entwicklung neuer Steuerungen nicht die exzellenteste Lösung gesucht, sondern die, die mit geringstem Preis- und Leistungsaufwand auskommt. Außerdem werden einige „Verwandte“ (80C167, 80C165) angeboten, die zum Teil mehr, zum Teil weniger Features aufweisen.

Die CPU: Das Herz des 80C166 wurde mit eines beachtlichen Rechenleistung ausgestattet, da bei einer Oszillatorfrequenz von 40MHz alle 100ns ein Befehl in der vierstufigen Pipeline weitergerückt wird. Dadurch dauern die meisten Befehle genau einen Maschinenzyklus lang, das sind dann 100ns was zu eine sehr theoretischen Rechenleistung von bis zu 10 MIPS führt.

Der Speicher: Das interne RAM hat die Größe von 1 KB, in dem der Stack und die Registerbänke liegen (GPR, General Purpose Register). Der 80C166 ist eine Registermaschine und keine Akkumulator-CPU. Weiters sind noch jede Menge an Spezialfunktionsregistern untergebracht. Ein weiteres beachtenswertes Merkmal ist das flexible Bussystem des 80C166 zum Anschluß des externen Speichers. Es stehen vier verschiedene Arten von Speichermodellen zur Verfügung :

- 16 bit Datenbus, Multiplexverfahren
- 8 bit Datenbus, Multiplexverfahren (vom 8031 bekannt)
- 16 bit Datenbus, Demultiplexverfahren
- 8 bit Datenbus, Demultiplexverfahren

Beim Multiplexverfahren werden Daten von Adressen durch ein Latch und das Address-Latch-Enable-(ALE-)Steuersignal getrennt, beim Demultiplexverfahren gibt es für jede Daten und Adreßleitung einen eigenen Pin. Diese Betriebsart muß natürlich unmittelbar nach einem Reset bekannt sein, ebenso muß man dem Bootstrap-Loader mitteilen, ob er aktiviert werden soll: dies geschieht durch Pull-Downwiderstände an den dafür vorgesehen Ports, die aber trotzdem für Input-Output-Zwecke zur Verfügung stehen.

Das Interruptsystem: Der 80C166 hat ein 16-stufiges Prioritätssystem, das 32 Interruptquellen verwaltet (der 80C167 noch um einiges mehr). Weiters gibt es 8 Traps für interne Fehler (z.B.: Stacküberlauf, Division by Zero, usw.). **Parallele Ports:** Zur Ein- und Ausgabe stehen 76 bitadressierbare Pins zur Verfügung. Einzeln adressierbare (bitadressierbare) Pins sind für Steuerungsaufgaben sehr wichtig. **Serielle Ports:** Zur Kommunikation mit anderen Einheiten besitzt der 80C166 zwei identische serielle Schnittstellen, die entweder synchron oder asynchron im Vollduplexbetrieb arbeiten können. **Die Timer-Einheit** (General Purpose Timer, GPT) und Capture-/Compare: Diese beiden Einheiten bieten eine solche Vielzahl von Möglichkeiten, sodaß ihre Programmierung am Anfang sicher bei einigen leichte Verzweiflung hervorruft. Jedoch mit ein paar guten Blockschaltbildern kommt man bald zu passablen Ergebnissen. Natürlich wurde bei der Entwicklung auch auf einen **programmierbaren Watchdog-Timer** und auf Stromsparmodi nicht vergessen.

Inzwischen gibt es schon die verschiedensten Derivate aus der 166-er Familie: zum Beispiel den C167, dessen Adreßraum von 256KB beim 80C166 auf 16 MB erweitert wurde. Der 80C167 verfügt außerdem über 16 Analog-Digital-Kanäle (10 Kanäle beim 80C166), eine Puls-Weiten-Modulations-Einheit, mehr Input-Output-Ports und Interrupts. Besonders interessant ist auch der Bootstraploader, der nach einem Reset das Programm über die serielle Schnittstelle in den Speicher lädt.

Ein weiteres Familienmitglied ist der 80C165 (der kleine Bruder des 80C167), der ohne Analog-Digital-Wandler, Puls-Weiten-Modulations-Einheit, Capture-/Compare, und mit weniger Input-Output-Ports auskommen muß.

Wer genaueres zu diesen Prachtexemplaren wissen möchte und wer sich mit der Entwicklung eines Systems mit einem Prozessor dieser Familie befaßt, sollte sich das Buch MC-Tools 17 - Arbeiten mit C166-Controllern von Karl Heinz Mattheis und Steffen Storandt (Verlag Otmar Feger) zulegen: in diesem Buch gibt’s genauere Details zu Hardwareentwicklung mit Berechnungsbeispielen, auf einer beigelegten Diskette sind einige brauchbare Excel-Files mit denen man diverse Ablaufzeiten für die verschiedenen Taktfrequenzen berechnen kann. Auch zur Programmierung gibt’s einige praktische Beispiele in C und Assembler.

Hier ein kurzer Inhaltsüberblick:

Einleitung. Reset und Systemeinstellung (Beschaltung des Reseteinganges, Abschalten der Versorgungsspannung, Reset und Power Down Mode). Taktversorgung (Schaltungen für Taktgeneratoren). Externer Bus (Timing, ALE, RD#, WR#..., Bus-Betriebsarten, Speicheranschluß [EPROM, RAM], Adreßraum). Eingangs- und Ausgangs-Pins. Befehle und Flags (Addition, Schieben und Rotieren, Compare, Division, MOV, Multiplikation, CPL und NEG, Call, PRIOR, Subtraktion, exklusives Oder). General Purpose Timer GPT1 und GPT2. CAPCOM (Timer, Capture-Funktionen, Pulsweitenmodulation, komplexe Pulsfolgen). Analog-Digital-Wandler. Asynchrone serielle Schnittstelle (Automatisches Erkennen der Baudrate). Synchrone serielle Schnittstelle (Anschluß eines seriellen EEPROMs, Emulation der asynchronen Übertragung mit der seriellen Schnittstelle). Interruptssystem. Softwarebeispiele (C-Programmierung, Laufzeit, Festkommawerte, Digitalfilter, lineare Interpolation). ADIS166 Disassembler mit One-Line-Assembler. Entwicklungsunterstützung. Anhang.

- Positiv: Übersichtliche Diagramme, viele Programmbeispiele, Tabellen, Diagramme, mitgelieferte Diskette.
- Negativ: Sehr dürftiges Stichwortverzeichnis (gerade eineinhalb Seiten), nicht gerade billig.

Ein weiteres Buch zum selben Thema gibt’s natürlich auch von Siemens. Der inhaltliche Zusammenhang ist schon am Cover ersichtlich. Auch das Handbuch des 80C166 von Reiner Johannis (Verlag Siemens) bietet einen guten Überblick über die breite Funktionspalette.

Auch hier eine kurze Inhaltsübersicht:

Vorwort. Inhaltsverzeichnis. Einleitung. Grundzüge der Architektur (Speicherorganisation, CPU und Befehlssatz, Interrupt-System, externer Bus, parallele Ports, serielle Ports, Timer-Einheit [GPT], Capture-Compare-Einheit [CAPCOM], A/D-Wandler, Watchdog-Timer, Stromsparmodi). Speicherorganisation (internes ROM, externer Speicher, internes RAM, Spezial-Funktions-Register). CPU (Pipeline, Timing, Spezial-Funktions-Register der CPU). Struktur des Befehlssatzes (Befehlsgruppen, Adressierungsarten, Verzweigungsbedingungen). Interruptssystem. Externer Bus. Reset. Parallele Ports. Serielle Schnittstellen. Timer-Einheit (GPT). Capture-Compare-Einheit (CAPCOM). Analog-Digital-Wandler. Watchdog-Timer. Stromsparmodi. Oszillator und Taktversorgung. Befehlssatz. Spezial-Funktionsregister. Versionen des 80C166. Datenblatt. Stichwortverzeichnis.

- Positiv: Klarer, systematischer Aufbau, deshalb gut als Nachschlagewerk geeignet. Umfangreicheres Stichwortverzeichnis (4 Seiten).
- Negativ: Keine praktischen Beispiele, sondern nur Theorie. Vorkenntnisse sind notwendig. Das Stichwortverzeichnis könnte noch um einiges umfangreicher sein.

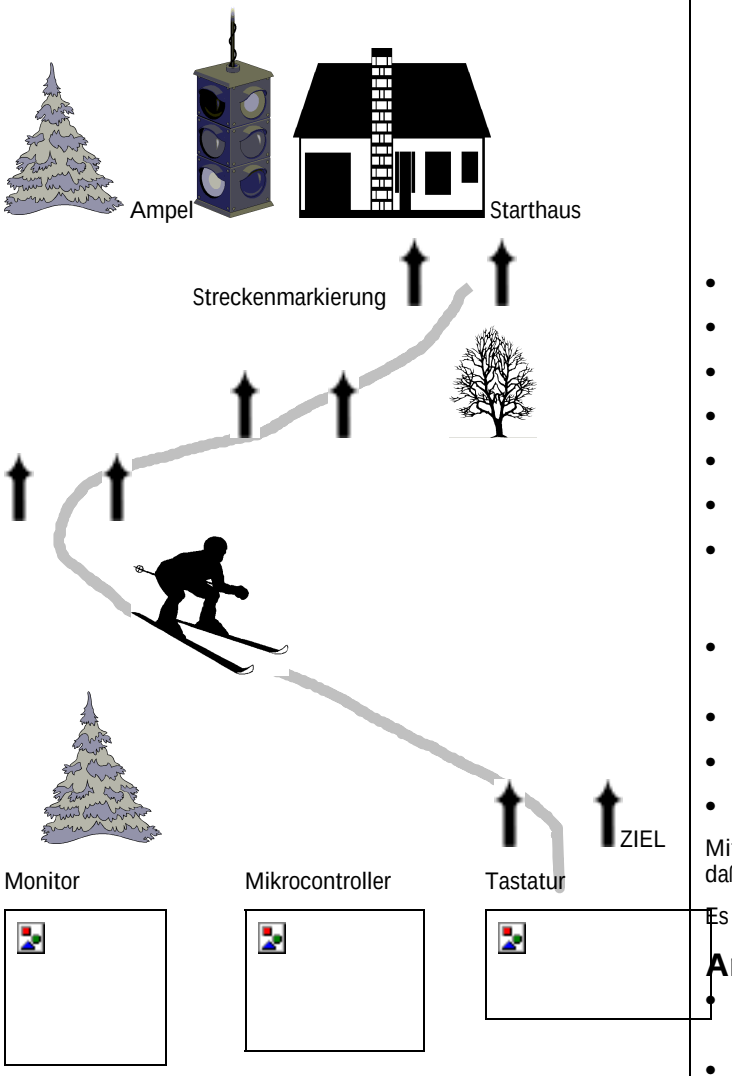
Beide Bücher zusammen wären zum Selbststudium und für weitere Entwicklungen gut geeignet. Natürlich ist für jeden Entwickler auch das Original-Datenbuch unumgänglich, denn gelegentlich gibt es Produktspezifikationen und kleine Änderungen, die in den Büchern natürlich nicht angeführt werden können. Alles in allem gibt das ca. 2,8 kg Papier, mit dem man für fast alle Anwendungen bestens gerüstet ist. ☐

Informatik + Mikroelektronik = binäre Bäume und Mikrocontroller

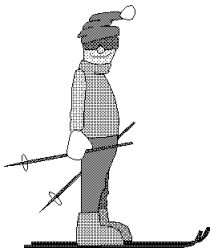
... eine sportliche Applikation rund um das Schirennen

Wilhelm Brezovits PCN-DSK-520:\baeume und DSK-521..526, 533..534

Applikation Schirennlauf



Kurzbeschreibung



- Die Ampel steht auf rot (Piste für den Schirennläufer gesperrt).
- Der Name des Schirennläufers wird erfaßt.
- Die Ampel wechselt auf grün (Piste frei).
- Der Schiläufer startet, die Zeitmessung beginnt.
- Die Ampel schaltet auf rot (Piste für den nächsten Läufer gesperrt).
- Der Schiläufer erreicht das Ziel, die Zeitmessung stoppt.
- Der Name und die Zeit des Schiläufers wird sortiert (1. nach der Zeit, bei gleicher Zeit alphabetisch nach Name) in einem binären Baum gespeichert.
- Ein Zwischenergebnis wird ausgegeben (1. Platz: Frau C, 2. Platz Herr B, 3. Platz Frau X, u.s.w.).
- Der Name des nächsten Schirennläufers wird erfaßt.
- Die Ampel wechselt auf grün (Piste frei).
- u.s.w.

Mittels Name nächster Schirennläufer = * wird dem Programm signalisiert, daß alle Läufer gestartet sind.

Es erfolgt der Ausdruck des Endergebnisses.

Anmerkung:

- Der binäre Baum hat den Vorteil, daß bei Programmstart die Anzahl der Schirennläufer nicht feststehen muß.
- Es wird nur bei Bedarf Speicherplatz angefordert!
- Weiters stehen die Daten jederzeit sortiert zur Verfügung!

Hinweis_1:

- *Demnächst ist die objektorientierte Version des C166-C-Compilers (C++) verfügbar.*
- *Diese Applikation ist extra für die PCNEWS edit geschrieben.*
- *Der binäre Baum in diesem Beispiel beschränkt sich auf das „sortierte Einfügen“ und „Ausgeben“, außerdem ist er „nur“ in ANSI-C codiert.*
- *Wenn sich mindestens eine Person bei der Redaktion meldet und Interesse bekundet (per Internet ?) bin ich gerne bereit, das ganze Beispiel zu erweitern (Speicherung in einer Liste / einem Baum; mit den Funktionen (Methoden) ausgeben, sortiert eintragen und löschen); natürlich in C++ codiert!*

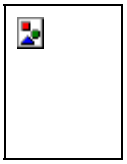
Hinweis_2:

- *Um mit der Beschreibung nicht den zur Verfügung gestellten Platz in der PCNEWS edit zu sprengen, wurden sämtliche Dateien mit sehr viel Kommentar versehen (selbsterklärend!).*
- *Weiters wurden aussagekräftige Namen (Variablen, Funktionen) verwendet.*
- *Das Studium nachfolgender Dateien müßte also ausreichen, den Inhalt zu verstehen und nachvollziehen zu können.*

Hinweis_3:

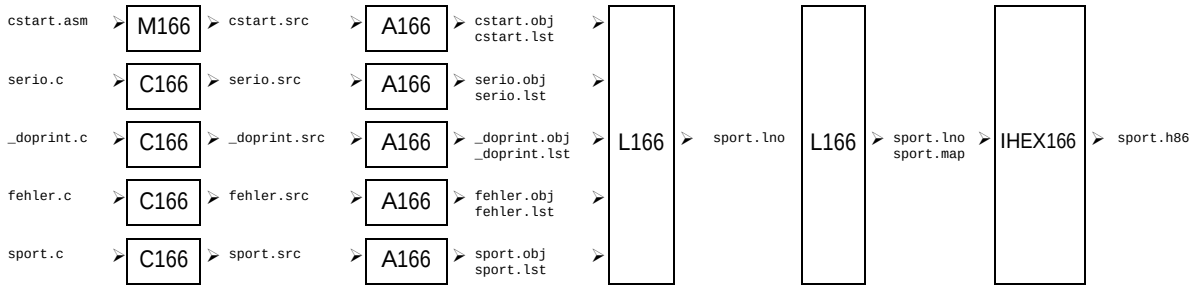
- *Um unsere Applikation so kurz und überschaubar wie möglich zu halten (aber funktionsfähig), werden Benutzereingabefehler programmtechnisch nicht abgefangen.*
- *Weiters erfolgt die Zeitmessung in Sekunden (obwohl diese 16 bit Mikrocontrollerfamilie eine Auflösung von !!! 40 ns !!! für externe Signale bei nur 25 MHz Taktfrequenz schafft.*

REALISIERUNG DER APPLIKATION



1. Schritt

Ein Schaubild der verwendeten Dateien wird entworfen. Die notwendigen Compiler, Makroassembler, Linker und Locater- Aufrufe werden eingetragen.



C166	C-Compiler
M166	Makroassembler
A166	Assembler
L166	Linker/Locater
IHEX166	Hex-Konverter

`cstart.asm` Vom Compilerhersteller mitgelieferte "Startup-Datei". Der Inhalt dieser Datei wird vor " void main (void) { } " abgearbeitet. cstart.asm initialisiert den Mikrocontroller wie zum Beispiel: externer Bus: gemultiplext, nicht gemultiplext, Waitstates, ... Stackgröße des Mikrocontrollers; Watchdog-Timer u.s.w.

`serio.c` Vom Compilerhersteller mitgelieferte Datei zur Unterstützung der asynchronen, seriellen Schnittstelle des Mikrocontrollers damit Funktionen wie "printf" oder "scanf" nicht ins Leere schreiben oder von der binären Wüste lesen.

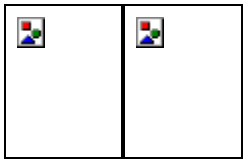
`_doprint.c` Vom Compilerhersteller mitgelieferte Datei. Zur Verbesserung des Laufzeitverhaltens stehen 3 Versionen von "printf" zur Verfügung.

`fehler.c` Bestimmte Fehler können während der Programmabarbeitung im Mikrocontroller auftreten (wie zum Beispiel: Stack-Überlauf, oder Stack-Unterlauf, auch das Bitmuster eines geschützten Befehles (Idle, Powerdown,...) kann vom externen Bus falsch eingelesen werden). Tritt so ein Fehler auf, springt der Mikrocontroller auf jedem Fall auf die dafür vorgesehene Interrupteinsprungadresse. Deshalb sollte auf dieser Interrupteinsprungadresse ein sinnvoller Befehl stehen (in unserem Beispiel ein Software-Reset).

`sport.c` Unsere Applikation (Schirennlauf) formuliert in ANSI-C mit einigen Erweiterungen die bei Mikrocontrollern notwendig sind.

Laden von sport.h86 — ERTEC-MONITOR

Bootstrap-Loader aktivieren		➢ Stellung „BTL ON“ ➢ RESET-Taste
Mon16x	RETURN	➢ Ertec-Monitor wird geladen
load „sport.h86“	RETURN	➢ Anwenderprogramm wird geladen
exit	RETURN	➢ Monitor wird verlassen
Bootstrap-Loader deaktivieren		➢ S2
	RESET	➢ Anwendung läuft



2. Schritt:

Die Dateien `cstart.asm`, `serio.c` und `_doprint.c` werden nach Hardware/Anforderung angepaßt.

Die Steuerdateien für den Linker und Locater L166 werden geschrieben:

Datei cmd_link

```
; datei: cmd_link
;
;  CMD_LINK
;  - steuerdatei für Linker l166
;

; Der heap wird benoetigt, wenn Speicherplatz waehrend
; der Laufzeit mittels
; malloc angefordert wird.
; 30000 Bytes reichen fuer ca. 1300 Schilauefer
; genau: HEAPSIZE / sizeof(KNOTEN)
HEAPSIZE (30000)

; Liste der Objektdateien die gebunden werden:
sport.obj fehler.obj  doprint.obj serio.obj cstart.obj

; Pfadangabe der Bibliotheken die mitgebunden werden:
LIBPATH(C:\bso_v5\lib\ext) C166S.LIB F166S.LIB

; gewuenschter Name der Output-Datei
TO sport.lno
```

Datei cmd_loc.e_e

```
; datei: cmd loc.e e
;
;  CMD LOC.E E
;  - steuerdatei für locater l166 fuer ertec-board und ertec-monitor
;

;zu bearbeitende Datei:
sport.lno

;Interrupt-Einsprungtabelle wird erzeugt:
VECTAB(0)
;NOVECINIT

;Hinweis zur Hardware:
;Standard Speicherkapazität = 256 KByte
;(2x1MBit static RAM = 2x128KByte devices: 0H - 3FFFFH)

;diverse Code/Datensegmente werden bestimmten Adressen zugewiesen:
CLASSES ('CUSTACK' (  4000H-  0BFFFH))
CLASSES ('CPROGRAM'( 20000H-  3FFFFH))

;belegte Speicheradressen vom ertec_167_monitor
RESERVE MEMORY (   200H to    1FFFH )
;belegte Registerbank vom ertec_167_monitor
RESERVE MEMORY ( 0FCC0H to   0FCDFH )

;freies RAM, nicht bestückt:
RESERVE MEMORY ( 40000H to   7FFFFH )
;mit standard devices nicht erreichbare Speicheradressen:
RESERVE MEMORY ( 80000H to 0FFFFFFH )

;PEC-Pointer Reservierung:
RESERVE PECPTR ( PECC0 to PECC7 )
```

Die Datei `fehler.c` wird geschrieben:

Datei fehler.c

```
        /* datei: fehler.c                                           Seite 1/1
        ╔══════════════════════════════════════════════════════════════════╗
        ║ TRAP PRIORITY III, RESET FUNCTIONS:                              ║
        ║  » Hardware Reset,             Trap-Nummer: 00H, Vector: 00'0000 H ║
        ║  » Software Reset,             Trap-Nummer: 00H, Vector: 00'0000 H ║
        ║  » Watchdog Timer Overflow,    Trap-Nummer: 00H, Vector: 00'0000 H ║
        ╠══════════════════════════════════════════════════════════════════╣
        ║ TRAP PRIORITY II,  CLASS A HARDWARE TRAPS:                       ║
        ║  » Non-Maskable Interrupt      Trap-Nummer: 02H, Vector: 00'0008 H ║
        ║  » Stack Overflow,             Trap-Nummer: 04H, Vector: 00'0010 H ║
        ║  » Stack Underflow,            Trap-Nummer: 06H, Vector: 00'0018 H ║
        ╠══════════════════════════════════════════════════════════════════╣
        ║ TRAP PRIORITY I,   CLASS B HARDWARE TRAPS:                       ║
        ║  » Undefined Opcode,           Trap-Nummer: 0AH, Vector: 00'0028 H ║
        ║  » Protected Instruction Fault, Trap-Nummer: 0AH, Vector: 00'0028 H ║
        ║  » Illegal Word Operand Access, Trap-Nummer: 0AH, Vector: 00'0028 H ║
        ║  » Illegal Instruction Access,  Trap-Nummer: 0AH, Vector: 00'0028 H ║
        ║  » Illegal External Bus Access, Trap-Nummer: 0AH, Vector: 00'0028 H ║
        ╚══════════════════════════════════════════════════════════════════*/

        interrupt ( 0x02 ) void laufzeit_fehler_trap_a_nmi   (void);
        interrupt ( 0x04 ) void laufzeit_fehler_trap_a_stkof (void);
        interrupt ( 0x06 ) void laufzeit_fehler_trap_a_stkuf (void);
        interrupt ( 0x0A ) void laufzeit_fehler_trap_b       (void);


    /**********************************************************************/
        interrupt ( 0x02 ) void laufzeit_fehler_trap_a_nmi (void)
        {
         _int166(0); // SRST...Software Reset
        } /* ende laufzeit_fehler_trap_a_nmi */

    /**********************************************************************/

    /**********************************************************************/
        interrupt ( 0x04 ) void laufzeit_fehler_trap_a_stkof (void)
        {
         _int166(0); // SRST...Software Reset
        } /* ende laufzeit_fehler_trap_a_stkof */

    /**********************************************************************/


    /**********************************************************************/
        interrupt ( 0x06 ) void laufzeit_fehler_trap_a_stkuf (void)
        {
         _int166(0); // SRST...Software Reset
        } /* ende laufzeit_fehler_trap_a_stkuf */

    /**********************************************************************/
/
        interrupt ( 0x0A ) void laufzeit_fehler_trap_b (void)
        {
         _int166(0); // SRST...Software Reset
        } /* ende laufzeit_fehler_trap_b */

    /**********************************************************************/
```

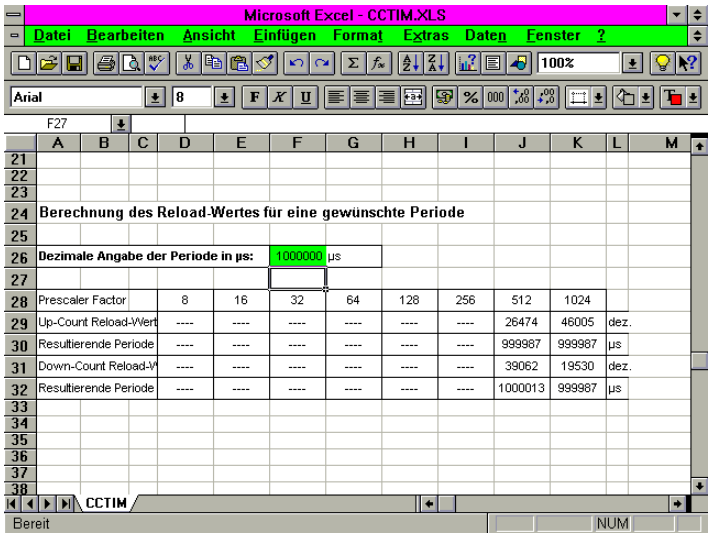
Die Datei `sport.c` unsere eigentliche Applikation wird geschrieben.

Datei sport.c

```
/*******************************************************************************
* Applikation Schirennlauf:                                                    *
*                                                                              *
********************************************************************************
* Beschreibung:                                                                *
*                                                                              *
* ...die Ampel steht auf Rot (die Piste ist fuer Schirennlaeufer gesperrt).    *
* Die rote Lampe der Ampel wird mittels P7.0 vom Mikrocontroller angesteuert.  *
* Die gruene Lampe der Ampel wird mittels P7.1 vom Mikrocontroller angesteuert.*
*                                                                              *
* Der Name des Schirennlaeufers wird erfasst.                                  *
* Der externe Interrupt fuer Port 2.0 wird freigegeben.                        *
* Die Ampel wechselt auf Gruen (Piste frei).                                   *
*                                                                              *
* Der Schilaeufer startet, die Zeitmessung beginnt:                            *
* Mittels positiver Flanke an Port 2.0 wird ein externer Interrupt ausgeloest. *
* = START-SIGNAL                                                               *
* Dieser Interrupt startet Timer 0 und schaltet die Ampel auf Rot.             *
*                                                                              *
* Der Schilaeufer erreicht das Ziel, die Zeitmessung stoppt:                   *
* Mittels positiver Flanke an Port 2.1 wird ein externer Interrupt ausgeloest  *
* = STOPP-SIGNAL                                                               *
* Dieser Interrupt stoppt Timer 0.                                             *
* Mittels globaler Variable schilaeufer_ziel_erreicht, wird dem Hauptprogramm  *
* mitgeteilt, dass der Schilaeufer das Ziel erreicht hat und die Zeitmessung   *
* abgeschlossen ist.                                                           *
*                                                                              *
* u.s.w. ...                                                                   *
*                                                                              *
*                                                                              *
*******************************************************************************/


#include <REG167.H>   /* Register Definitionen f r Mikrocontroller C167       */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include "serio.h"

#define MAXNAME 15    /* maximale Anzahl von Zeichen-1 f r Schi-Laeufer-Namen  */

#define TRUE 1
#define FALSE 0

// Bitname fuer das Port-Richtungsbit AUSGANG_ROT definieren:
// AUSGANG_ROT = 0 bedeutet: Port 7.0 ist Eingang (hochohmiger Zustand)
// AUSGANG_ROT = 1 bedeutet: Port 7.0 ist Ausgang (PUSH-PULL)
// PUSH-PULL bedeutet: es wird aktiv 0 Volt oder 5 Volt ausgegeben
// (je nach Inhalt vom Portlatch P7.0)
sfrbit AUSGANG_ROT _atbit(DP7,0);

// Bitname fuer das Port-Richtungsbit AUSGANG_GRUEN definieren:
sfrbit AUSGANG_GRUEN _atbit(DP7,1);

// das bit ROT ist der Zustand der aktiv ( 0 oder 5 Volt) am Pin
// AUSGANG_ROT = P7.0 ausgegeben wird:
sfrbit ROT _atbit(P7,0);

// das bit GRUEN ist der Zustand der aktiv ( 0 oder 5 Volt) am Pin
// AUSGANG_GRUEN = P7.1 ausgegeben wird:
sfrbit GRUEN _atbit(P7,1);


// Definition eines Knoten vom binaeren Baum:
typedef struct knoten
{
     char name[MAXNAME];    /* Name des Schi-Laeufers         */
     int  zeit;             /* gelaufene Zeit in [s]          */
     struct knoten *links;  /* linker Nachkomme               */
     struct knoten *rechts; /* rechter Nachkomme              */
} KNOTEN;


/*==================== Prototypen =========================================*/
void main (void); /* Hauptprogramm = Interruptservice-Routine mit            */
           /* Interruptvektor (Einsprungadresse) auf Adresse 0        */
           /* Ereignis welches zum Aufruf f hrt: RESET                */

void init_capcom(void);      /* Initialisierung der externen Interrupts       */
void init_timer_0(void);     /* Initialisierung von Timer 0                   */
void init_interrupt(void);   /* Initialisierung der Interrupt-Prioritaeten    */

KNOTEN *einordnen(KNOTEN *knot_zeig, char *name);
void drucke_binaeren_baum(KNOTEN *knot_zeig, int *nr);
KNOTEN *knoten_fuer_binaeren_baum_anlegen(void);

void externen_interrupt_fuer_start_freigeben(void);
/*========================================================================*/

/*==================== Prototypen Interruptservice-Routinen ================*/
interrupt (0x10) void externer_interrupt_START (void);  /* Signal an Port 2.0 */
interrupt (0x11) void externer_interrupt_STOPP (void);  /* Signal an Port 2.1 */
interrupt (0x20) void timer_0_SEKUNDENINTERRUPT (void);
/*========================================================================*/

/*== globale Variablen zur Kommunikation: Hauptprogramm <-> Interrupts =======*/
char schilaeufer_ziel_erreicht;
int zeit_in_sekunden;
bit start_signal_freigabe;
bit stopp_signal_freigabe;
/*========================================================================*/


/*========================================================================*/
void main (void)
{
 KNOTEN *wurzel;       // Pointer der auf den binaeren Baum zeigt anlegen
 char wer[MAXNAME];    // Platz fuer den Namen des Schirennlaeufers schaffen
 int i=1, platz;

 ROT = TRUE;           // Ampelinitialisierung bei Programmstart
 GRUEN = FALSE;        // Ampelinitialisierung bei Programmstart
 AUSGANG_ROT = TRUE;   // Port 7.0 = AUSGANG (PUSH-PULL), P7.0 ist jetzt high
 AUSGANG_GRUEN = TRUE; // Port 7.1 = AUSGANG (PUSH-PULL), P7.0 ist jetzt low

 start_signal_freigabe = FALSE;
 stopp_signal_freigabe = FALSE;

 wurzel=NULL;          // Anfang des binaeren Baums festlegen

 init_serio();         // Initialisierung der seriellen Schnittstelle aufrufen
 init_capcom();        // Initialisierung der externen Interrupts
 init_timer_0();       // Initialisierung von Timer 0
 init_interrupt();     // Initialisierung der Interrupt-Prioritaeten

 IEN=1;                // Interruptsystem global freigeben

 while (1)
 {
    do
    {
         printf("\n\nName des am START stehenden Schilaeufers = ? ");
         scanf("%s",wer); fflush(stdin);

         if (wer[0]!='*')
         {
              schilaeufer_ziel_erreicht='n'; // n...nein
              zeit_in_sekunden=0;

              externen_interrupt_fuer_start_freigeben();

              printf("\n");
              printf("\n%5u Sekunden, ",zeit_in_sekunden);
              printf(" ( aktueller Zwischenstand Laeufer: ");
              printf("%15s )",wer);
              while (schilaeufer_ziel_erreicht == 'n')   // warten, bis die
              {                                          // Zeitmessung
                                         // beendet ist !!!
                   printf("\r%5u",zeit_in_sekunden);

              } /* ende while */
              printf("\n");

              wurzel=einordnen(wurzel,wer);
              printf("\n\nZwischenstand nach %d Laeufer(n)\n",i++);

         } /* ende if */

         else
              printf("\n\nEndergebnis:\n");

         printf("\n%12s %-15s %12s [Sekunden]\n","Platz","Name","Zeit");
         platz=1;
         drucke_binaeren_baum(wurzel,&platz);

    } while(wer[0]!='*');

    _int166(0); // SRST...Software Reset
             // Aus der Sicht eines PC-Programmierers eine ungewoehnliche
             // Art und Weise mittels malloc reservierten Speicher freizugeben
 } /* ende while (1) */

} /* ende main */
/*========================================================================*/

/*========================================================================*/
void init_capcom(void)
{
 CCM0 = 0x0011;   /* 0000 0000 0001 0001 B                                */
           /* |||| |||| |||| 0001: CAPCOM1_Kanal_0                    */
           /* |||| |||| |||| ||01: pos. Flanke auf P2.0 = Signal START */
           /* |||| |||| |||| |0: capture mod = externer Interrupt      */
           /* |||| |||| |||| 0: zugeordneter Timer egal                */
           /* |||| |||| 0001: CAPCOM1_Kanal_1                          */
           /* |||| |||| ||01: positive Flanke auf P2.1 = Signal STOPP  */
```

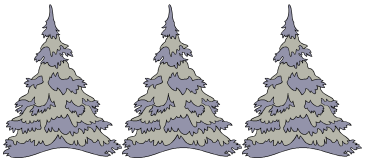
```
            /* |||| |||| |0: capture mod = externer Interrupt          */
            /* |||| |||| 0: zugeordneter Timer egal                    */
            /* |||| 0000: CAPCOM1_Kanal_2                              */
            /* 0000: CAPCOM1_Kanal_3                                   */

} /* ende init_capcom */
/*==========================================================================*/

/*==========================================================================*/
void init_timer_0(void)
{

 T01CON = 0x0006;  /* xxxx xxxx x0xx 0110 B                                 */
             /*            |   |110: T0I:Timer/Counter 0 Input Selection*/
             /*            |   |          Teiler fuer fCPU = 512        */
             /*            |   0: T0M: Timer/Counter-Mode: 0=Timer Mode */
             /*           0: Timer/Counter 0 Run Control: Timer steht  */

 T0REL  = 26474;  /* Reload-Wert fuer Timer 0 Hochlauf fuer Periode=1 Sekunde */
            /* PT0 = ( (65536-T0REL) * 2 hoch(T0I+3) ) / fCPU =
            /*     = ( (65536-26474) * 2 hoch(6+3)   ) / 20 000 000 = 1 */
 T0=T0REL;

} /* ende init_timer */
/*==========================================================================*/

/*==========================================================================*/
void init_interrupt(void)
{
  // Interrupt Prioritaet Timer 0 (Sekundensignal): ILVL=10,GLVL=0
 T0IC  = 0x0068;   /* xxxx xxxx 01 1010 00 B                                */
             /*           || |||| 00: Group Level GLVL=0                 */
             /*           || 1010: Interrupt Priority Level ILVL=10      */
             /*           |1: Interrupt Enable Control Bit setzen        */
             /*           0: Interrupt Request Flag loeschen             */

 // Interrupt Prioritaet Signal START: ILVL=14,GLVL=0
 CC0IC = 0x0078;   /* xxxx xxxx 01 1110 00 B, IR=0, IE=1, ILVL=14, GLVL=0     */

 // Interrupt Prioritaet Signal STOPP: ILVL=14,GLVL=1
 CC1IC = 0x0079;   /* xxxx xxxx 01 1110 01 B, IR=0, IE=1, ILVL=14, GLVL=1     */


} /* ende init_interrupt */
/*==========================================================================*/



/*==========================================================================*/
KNOTEN *einordnen(KNOTEN *knot_zeig, char *name)
{

 if (knot_zeig==NULL)
 {
     knot_zeig=knoten_fuer_binaeren_baum_anlegen();
     strcpy(knot_zeig->name,name);
     knot_zeig->zeit=zeit_in_sekunden;
     knot_zeig->links=knot_zeig->rechts=NULL;
 } /* ende if */

 else
     if (((zeit_in_sekunden-knot_zeig->zeit)<0) ||
     ((zeit_in_sekunden==knot_zeig->zeit) && (strcmp(name,knot_zeig->name)<0)))
          knot_zeig->links=einordnen(knot_zeig->links,name);
     else
          knot_zeig->rechts=einordnen(knot_zeig->rechts,name);

 return(knot_zeig);
} /* ende *einordnen */
/*==========================================================================*/

/*==========================================================================*/
void drucke_binaeren_baum(KNOTEN *knot_zeig, int *nr)
{
 if (knot_zeig!=NULL)
 {
     drucke_binaeren_baum(knot_zeig->links,nr);
     printf("%12d %-15s %12d\n",(*nr)++,knot_zeig->name,knot_zeig->zeit);
     drucke_binaeren_baum(knot_zeig->rechts,nr);
 } /* ende if */
} /* ende drucke_binaeren_baum */
/*==========================================================================*/

/*==========================================================================*/
KNOTEN *knoten_fuer_binaeren_baum_anlegen(void)
{
 KNOTEN *return_wert;

 return_wert=((KNOTEN *)malloc(sizeof(KNOTEN)));

 if (return_wert == NULL)
    printf("\n\n!!!ERROR: KEIN SPEICHERPLATZ VORHANDEN !!!"), _int166(0);

 return return_wert;

} /* ende KNOTEN *knoten_fuer_binaeren_baum_anlegen */
/*==========================================================================*/

/*==========================================================================*/
void externen_interrupt_fuer_start_freigeben(void)
{
 // Aufgabe: Start des Schilaufers vorbereiten: Ampel = Gruen

 start_signal_freigabe = TRUE;

 ROT   = FALSE;  /* Piste frei                                            */
 GRUEN = TRUE;   /* Piste frei                                            */

} /* ende externen_interrupt_fuer_start_freigeben */
/*==========================================================================*/

/*==========================================================================*/
interrupt (0x10) void externer_interrupt_START (void)
{
 // Aufgabe: Zeitmessung (Timer 0) starten und Ampel auf ROT schalten

 if (start_signal_freigabe)
 {
     T0R = 1; /* Timer 0 Run-Bit gesetzt -> Timer 0 laeuft (Zeitmessung laeuft)   */

     GRUEN = FALSE;  /* Piste fuer den naechsten Laeufer gesperrt                 */
     ROT   = TRUE;   /* Piste fuer den naechsten Laeufer gesperrt                 */

     start_signal_freigabe = FALSE;
     stopp_signal_freigabe = TRUE;
 } /* ende if */

} /* ende externer_interrupt_START */
/*==========================================================================*/

/*==========================================================================*/
interrupt (0x11) void externer_interrupt_STOPP (void)
{
 // Aufgabe: Timer_0 wird gestoppt.
 // mittels globaler Variable schilaeufer_ziel_erreicht wird dem Hauptprogramm
 // mitgeteilt, dass der Schilaeufer das Ziel erreicht hat, und die Zeitmessung
 // beendet ist.

 if (stopp_signal_freigabe)
 {
     T0R = 0; /* Timer 0 Run-bit geloescht -> Timer 0 steht (Zeitmessung beendet) */
     T0  = T0REL; /* Timer 0 Zaehlregister loeschen                                 */

     stopp_signal_freigabe = FALSE;

     schilaeufer_ziel_erreicht = 'j'; /* j...ja...Ziel erreicht !!!                */
 } /* ende if */

} /* ende externer_interrupt_STOPP */
/*==========================================================================*/

/*==========================================================================*/
interrupt (0x20) void timer_0_SEKUNDENINTERRUPT (void)
{
 zeit_in_sekunden++;
} /* ende timer_0_SEKUNDENINTERRUPT */
/*==========================================================================*/
```

Bei der Berechnung des Sekundeninterrupts (Timer 0 Nachladewert) für die Zeitmessung ziehen wir die Excel-Dateien aus dem Buch „Arbeiten mit C166 Controllern von Karl-Heinz Mattheis“ zu rate:



!!! Alle Source-Dateien und Steuerdateien sind jetzt vorhanden !!!

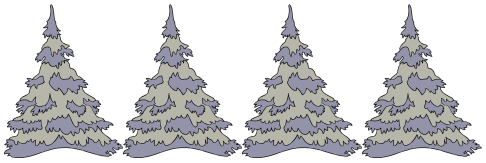
Der nächste Schritt ist die Generierung der Hex-Datei sport.h86 mittels make, welche per Bootstrap-Loader und Monitor-Programm in unsere Zielhardware geladen wird:



3. Schritt: Erstellung des makefile:

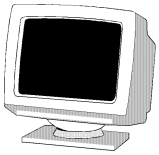
Datei makefile

```
# datei: makefile
#
# ╔═══════════════════════════════════════════════════════════════════════════════════════════════════════════╗
# ╟│ MAKEFILE                                                                                                 │
# ╟│                                                                                                          │
# ╟│ Aufruf: mk166        // Generierung aller Dateien laut Toolkette                                         │
# ╟│         mk166 clean  // Löschen aller erzeugten Dateien                                                  │
# ╟│                                                                                                          │
# ╚═══════════════════════════════════════════════════════════════════════════════════════════════════════════╝
#
# ╔═══════════════════════════════════════════════════════════════════════════════════════════════════════════╗
# ╟│ Hinweis_1:                                                                                               │
# ╟│                                                                                                          │
# ╟│ Wird das MAKEFILE mit "mk166 clean" von MS-DOS oder von MS-WINDOWS (Codewright) aufgerufen, dann werden alle unter dem Label │
# ╟│ clean angegebenen Dateien gelöscht.                                                                      │
# ╟│                                                                                                          │
# ╚═══════════════════════════════════════════════════════════════════════════════════════════════════════════╝
#
# ╔═══════════════════════════════════════════════════════════════════════════════════════════════════════════╗
# ╟│ Hinweis_2:                                                                                               │
# ╟│                                                                                                          │
# ╟│ Wird das MAKEFILE mit "mk166" von MS-DOS oder von MS-WINDOWS (Codewright) aufgerufen, dann werden alle Dateien laut Toolkette │
# ╟│ erzeugt.                                                                                                 │
# ╟│ Bereits übersetzte und noch aktuelle Dateien werden nicht nochmals übersetzt -> Zeitersparnis !!!       │
# ╟│                                                                                                          │
# ╚═══════════════════════════════════════════════════════════════════════════════════════════════════════════╝
#
# ╔═══════════════════════════════════════════════════════════════════════════════════════════════════════════╗
# ╟│ Hinweis_3:                                                                                               │
# ╟│                                                                                                          │
# ╟│ Ein MAKEFILE liest man am besten von unten nach oben.                                                    │
# ╟│ Wichtig für make ist die vom Betriebssystem für die zu bearbeitenden Dateien gesetzte Uhrzeit.          │
# ╟│                                                                                                          │
# ╟│ Es wird nur dann der Compiler, Assembler, Linker, Locater oder Objekt-Hex-Wandler aufgerufen, wenn die "Ergebnisdatei" │
# ╟│ älteren Datums als die "Ursprungsdatei(en)" ist (sind) oder die "Ergebnisdatei" nicht vorhanden ist (Vorteil: Zeitersparnis). │
# ╟│                                                                                                          │
# ╟│ In den ersten beiden Zeilen des Makefile (Leserichtung von unten nach oben) steht:                       │
# ╟│                                                                                                          │
# ╟│ serio.src: serio.c serio.h                                                                               │
# ╟│  C166 -x -s -g -Ms serio.c                                                                               │
# ╟│                                                                                                          │
# ╟│ Diese beiden Zeilen werden von mk166 folgendermaßen interpretiert:                                       │
# ╟│ serio.src ist die "Ergebnisdatei" der "Ursprungsdateien" serio.c und serio.h                             │
# ╟│ Ist serio.src nicht vorhanden oder hat serio.src ein älteres Datum als serio.c oder serio.h dann wird serio.src (neu) erzeugt. │
# ╟│ Wie das "Erzeugen" aussieht steht in der nächsten Zeile (mit TAB oder BLANK) eingerückt.                 │
# ╟│ In unserem Beispiel ist es ein Compileraufruf C166.                                                      │
# ╟│ -x bedeutet Mikrocontroller = C167                                                                       │
# ╟│ -s bedeutet Assemblercode in der Protokolldatei anführen                                                 │
# ╟│ -g erzeugt HLL-Debug-Information                                                                         │
# ╟│ -Ms selektiert das verwendete Speichermodell (s...small)                                                 │
# ╟│                                                                                                          │
# ╚═══════════════════════════════════════════════════════════════════════════════════════════════════════════╝


# OBJEKT-HEX-WANDLER:
# Verwandelt den OUTPUT vom Locater in ein Format, welches von EEPROM-Programmern oder in unserem Beispiel vom Monitor verstanden
# wird.
sport.h86: sport.out
 IHEX166 sport.out -i32 -o sport.h86

# LOCATER:
# Der Locater wandelt verschiebbare Programm/Datensegmente in ablauffaehige Segmente mit festen Adressen (er verteilt sie dabei
# auf den verfuegbaren Speicher.
sport.out: sport.lno cmd_loc.e_e
 L166 LOCATE @cmd_loc.e_e TO sport.out

# LINKER:
# Der Linker fuegt nach gewissen Regeln Programm-  und Daten-Segmente zusammen.
sport.lno: cstart.obj serio.obj _doprint.obj fehler.obj sport.obj cmd_link c:\bso_v5\lib\ext\c166s.lib c:\bso_v5\lib\ext\f166s.lib
 L166 LINK @cmd_link

# ASSEMBLER
# Der Assembler uebersetzt den vom Compiler/Makroassembler erzeugten mnemonischen Code in Maschinenkode
sport.obj: sport.src
 A166 sport.src TO sport.obj
fehler.obj: fehler.src
 A166 fehler.src TO fehler.obj
_doprint.obj:  _doprint.src
 A166 _doprint.src  TO _doprint.obj
serio.obj:  serio.src
 A166 serio.src  TO serio.obj
cstart.obj: cstart.src
 A166 cstart.src TO cstart.obj

# MAKROASSEMBLER
cstart.src: cstart.asm
 m166 cstart.asm

# COMPILER
sport.src: sport.c
 C166 -x -s -g -Ms sport.c
fehler.src: fehler.c
 C166 -x -s -g -Ms fehler.c
_doprint.src:  _doprint.c
 C166 -x -s -g -Ms _doprint.c
serio.src:  serio.c serio.h
 C166 -x -s -g -Ms serio.c

clean:
       $(exist sport.src del sport.src)
       $(exist sport.obj del sport.obj)
       $(exist sport.lst del sport.lst)
       $(exist sport.lno del sport.lno)
       $(exist sport.out del sport.out)
       $(exist sport.map del sport.map)
       $(exist sport.h86 del sport.h86)
       $(exist fehler.src del fehler.src)
       $(exist fehler.obj del fehler.obj)
       $(exist fehler.lst del fehler.lst)
       $(exist _doprint.src del _doprint.src)
       $(exist _doprint.obj del _doprint.obj)
       $(exist _doprint.lst del _doprint.lst)
       $(exist serio.src del serio.src)
       $(exist serio.obj del serio.obj)
       $(exist serio.lst del serio.lst)
       $(exist cstart.src del cstart.src)
       $(exist cstart.obj del cstart.obj)
       $(exist cstart.lst del cstart.lst)
```



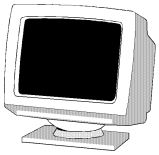
4. Schritt:

Es gibt jetzt 2 Möglichkeiten der Programmentwicklung:



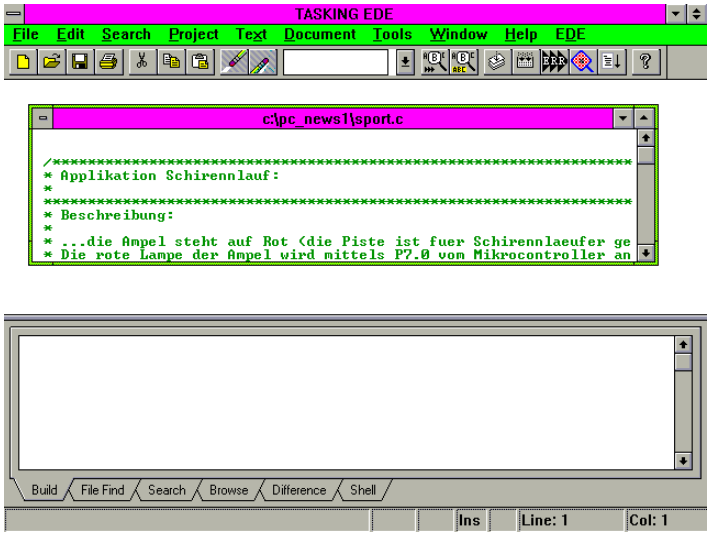
4.1) Programmentwicklung unter MS-DOS:

- Wir nehmen unseren MS-DOS Lieblings-Texteditor und bearbeiten sport.c.
- Anschließend starten wir das makefile mittels mk166.
- Danach laden wir `sport.h86` in unser Zielsystem.
- Befinden sich Syntaxfehler in `sport.c` bricht make (mk166) ab.
- Wir merken uns Fehler und Zeile die der Compiler meldet und beheben diese in unserem Source-File `sport.c`.

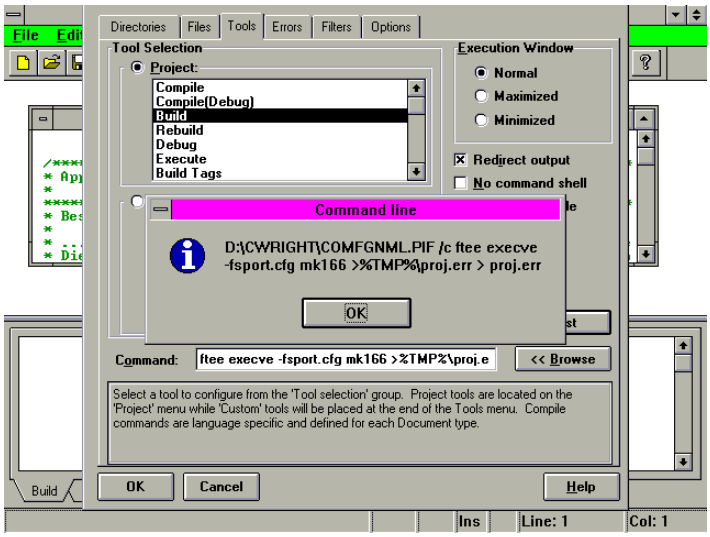


4.2) Programmentwicklung unter MS-WINDOWS und CodeWright:

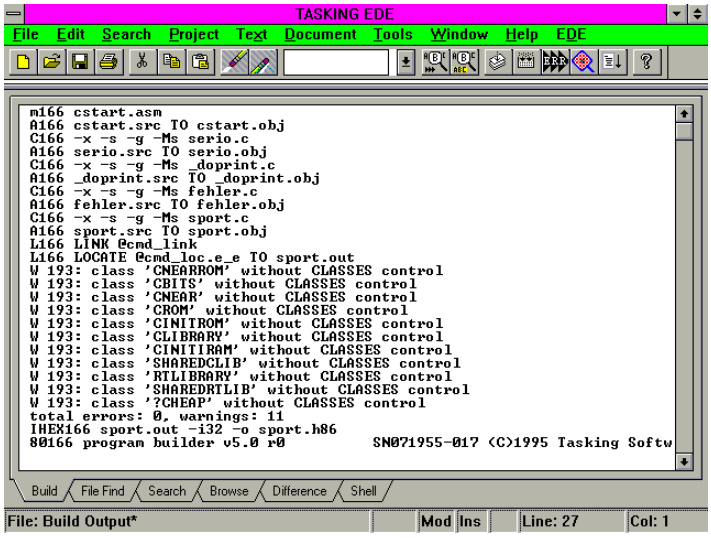
So sieht unsere Entwicklungsoberfläche aus:



Der Aufruf des makefile durch mk166 wird folgendermaßen in unsere graphische Benutzeroberfläche eingebunden:



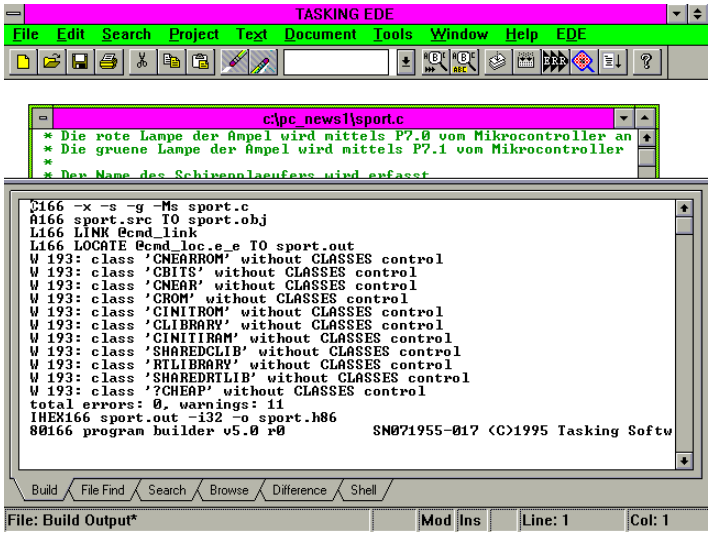
Wird jetzt Build aufgerufen dann sieht das so aus:



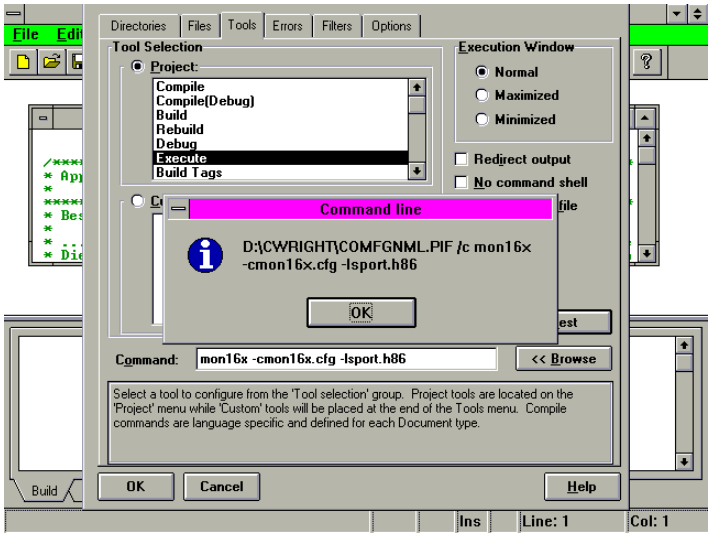
Das schöne an der Windows-Oberfläche ist, daß bei einem Syntaxfehler in `sport.c` mk166 abbricht, der Compiler die Fehler und Zeilennummern zurückmeldet, und der Texteditor automatisch im Source-File an der richtigen Stelle positioniert!

Das Schöne an mk166 ist, daß nur die Dateien neu generiert werden, die notwendig sind.

Ändern wir zum Beispiel nur `sport.c`, dann sieht der mk166-Lauf folgendermaßen aus:



Das Laden und Ausführen von `sport.h86` wird auch in unsere Entwicklungsoberfläche eingebunden:



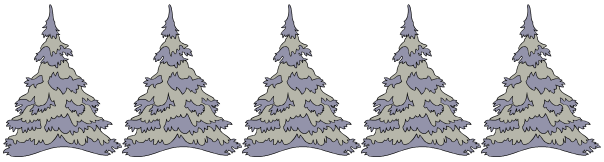
Das Laden des Monitors mittels Bootstrap-Loader und das Nachladen von `sport.h86` mittels Monitor sieht so aus:

```
MS-DOS Prompt
MON16x-User-Interface  V1.31        (c)ertec GmbH

booting
loading file 'sport.h86'... _
```

```
W 193: class 'SHAREDRTLIB' without CLASSES control
W 193: class '?CHEAP' without CLASSES control
total errors: 0, warnings: 11
IHEX166 sport.out -i32 -o sport.h86
80166 program builder v5.0 r0          SN071955-017 (C)1995 Tasking Softw
```

Build | File Find | Search | Browse | Difference | Shell

Ins | Line: 1 | Col: 1



5. Schritt: Benutzerschnittstelle

Egal ob wir unsere Applikation unter MS-DOS oder MS-WINDOWS entwickelt haben: jetzt ist es Zeit, die Benutzerschnittstelle als Hardcopy anzuführen. An den Mikrocontroller kann ein ASCII-Terminal oder einfach ein Terminalprogramm angeschlossen werden. Dies sieht dann so aus:

```
Terminal - 9600.TRM
Datei  Bearbeiten  Einstellungen  Telefon  Übertragung  Hilfe

Name des am START stehenden Schilaeufers = ? wilhelm

   42 Sekunden,  ( aktueller Zwischenstand Laeufer:         wilhelm )

Zwischenstand nach 1 Laeufer(n)

       Platz Name                    Zeit [Sekunden]
           1 wilhelm                   42

Name des am START stehenden Schilaeufers = ? martina
```

```
Terminal - 9600.TRM
Datei  Bearbeiten  Einstellungen  Telefon  Übertragung  Hilfe

   42 Sekunden,  ( aktueller Zwischenstand Laeufer:         wilhelm )

Zwischenstand nach 1 Laeufer(n)

       Platz Name                    Zeit [Sekunden]
           1 wilhelm                   42

Name des am START stehenden Schilaeufers = ? martina

   48 Sekunden,  ( aktueller Zwischenstand Laeufer:         martina )

Zwischenstand nach 2 Laeufer(n)

       Platz Name                    Zeit [Sekunden]
           1 wilhelm                   42
           2 martina                   48

Name des am START stehenden Schilaeufers = ? dominik
```

```
Terminal - 9600.TRM
Datei  Bearbeiten  Einstellungen  Telefon  Übertragung  Hilfe

Zwischenstand nach 2 Laeufer(n)

       Platz Name                    Zeit [Sekunden]
           1 wilhelm                   42
           2 martina                   48

Name des am START stehenden Schilaeufers = ? dominik

   24 Sekunden,  ( aktueller Zwischenstand Laeufer:         dominik )

Zwischenstand nach 3 Laeufer(n)

       Platz Name                    Zeit [Sekunden]
           1 dominik                   24
           2 wilhelm                   42
           3 martina                   48

Name des am START stehenden Schilaeufers = ? sebastian
```

und so weiter, bis zum Endergebnis:

```
Terminal - 9600.TRM
Datei  Bearbeiten  Einstellungen  Telefon  Übertragung  Hilfe

Zwischenstand nach 5 Laeufer(n)

       Platz Name                    Zeit [Sekunden]
           1 dominik                   24
           2 sebastian                 28
           3 matthias                  32
           4 wilhelm                   42
           5 martina                   48

Name des am START stehenden Schilaeufers = ? *

Endergebnis:

       Platz Name                    Zeit [Sekunden]
           1 dominik                   24
           2 sebastian                 28
           3 matthias                  32
           4 wilhelm                   42
           5 martina                   48

Name des am START stehenden Schilaeufers = ?
```

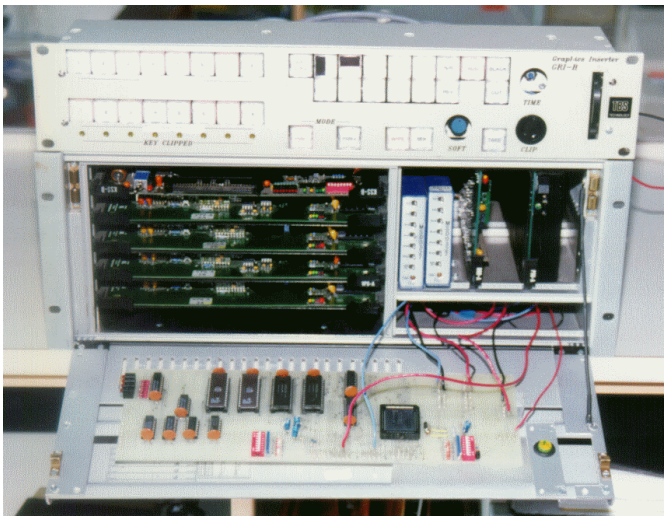
□

UNISTEB

Universelles Steuer- und Bediengerät für die Fernsehtechnik

Betreuer: Robert Seufert

Stephan Huber, Ondrej Gavura



1) Allgemeines

In der Fernsehtechnik werden zum Schnitt und für Live-Übertragungen speziell programmierbare Bedienpulte und Steuergeräte für den Regisseur benötigt. Diese Bedienpulte dienen dazu, die verschiedensten Quellen, wie Videosignale (Kamera, Magnetbandaufzeichnung,, ...), Tonsignale (Mikrofon, CD-Player, ...), Grafikinserts u. a. zu koordinieren und korrekt zur Sendeanlage zu leiten. Die zur Zeit im Einsatz befindlichen Geräte haben die Nachteile, daß sie entweder zu speziell, zu groß oder nicht flexibel genug zu handhaben sind.

2) Aufgabenstellung

Es sollte ein Redesign eines bestehenden Systems (System ARANEA von TBS) entworfen werden. Die Hauptmerkmale: flexibler, einfacher zu bedienen und zu programmieren (mittels PC); von einem kleinen Grundsystem erweiterbar..

3) Vorgangsweise

- Untersuchung des bestehenden Systems
- Schnittstellen- / Leistungsdefinition
- Steuersoftware
- Konfigurationssoftware für PC

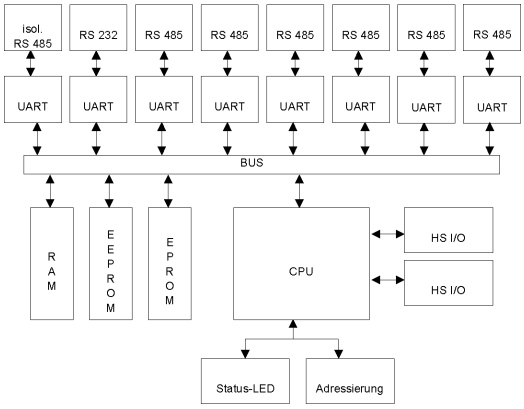


Bild1: **Blockschaltbild der Kreuzschienensteuerung (KSS)**

Das Blockschaltbild gliedert sich in mehrere Teile:

a) Controller 80C196KT

Der Controller 80C196KT übernimmt alle Steueraufgaben auf der KSS. Er programmiert die UARTs und empfängt von ihnen die über die serielle Schnittstelle einlaufende Signale. Er verwaltet den Speicher, in dem Konfigurationsdaten gespeichert werden. Die Steuerung der synchronen Schnittstellen für die Programmierung der Kreuzschienen o.ä. wird ebenfalls direkt vom Controller übernommen.

Der 80C196KT besitzt einen Adreßraum von 64 kByte, die über einen 16 bit breiten Daten- und Adreßbus angesprochen werden können.

b) Speicher: RAMs, EPROMs, Flash-EEPROMs

Es wurde mehr Speicher für Konfigurationsdaten vorgesehen. Das Flash-EEPROM dient zum dauerhaften Sichern der Konfigurationsdaten bei Netzausfall.

c) Serielle Schnittstellen

Das Senden und Empfangen der seriellen Daten übernehmen schnelle UART-Bausteine des Typs 16550. FIFO-Buffer verhindern, daß ankommende Daten nicht sofort von der CPU abgeholt werden müssen.

d) Burst-Signal-Decoder

Da ein Fernsehbildschnitt nur im Bildrücklauf erfolgen darf, müssen die Kreuzschienen mit dem Fernsehbild synchronisiert werden.

e) Relais-Steuerung

Da im Fehlerfall die komplette KSS-Platine vom übrigen System abgetrennt werden können sollte, sind alle Signale mittels Relais schaltbar.

4) Bestimmung der Größe

- Schnittstellen, davon 1x RS232 (PC), 1x RS485 isol.
- bis zu 10 Tastenpulte (RCPs) pro Schnittstellenbus
- RCP bis zu 128 Tasten
- 8 Über-Monitor-Displays (ÜMD's) mit je 16 Zeichen pro Schnittstellenbus
- max. 8 KSS-Platinen in einem Gerät

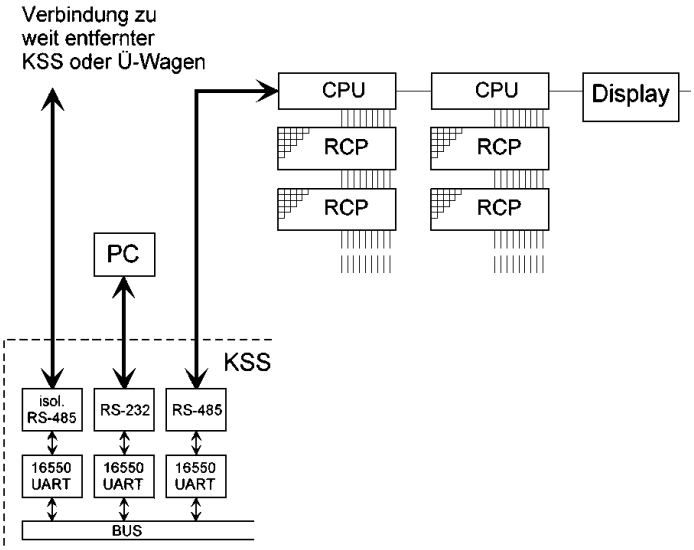


Bild 2: **Komponenten**

Bild 2 zeigt die verschieden Komponenten (Tastenpulte, PC, usw.) und wie sie mit den drei Schnittstellentypen der KSS verbunden werden

5) Sicherheitsmaßnahmen

Da ein Sendungsausfall in einem Studio unbedingt vermieden werden muß, werden aus Sicherheitsgründen folgende Maßnahmen getroffen:

- Mindestens doppelte Ausführung aller Systemkomponenten
- Wenn ein Teil ausfällt sollen andere übernehmen
- Selbständige Abschaltung, wenn Fehler erkannt werden
- Keine „Zentrale“ in der alle Konfigurationen gespeichert sind, damit das System weiterarbeiten kann, wenn genau diese ausfällt (jede Komponente enthält alle Daten)
- Ständiger Datenaustausch und -update zwischen allen „intelligenten“ Komponenten
- Rettung der Daten und des Zustandes des Systems bei Netzausfall

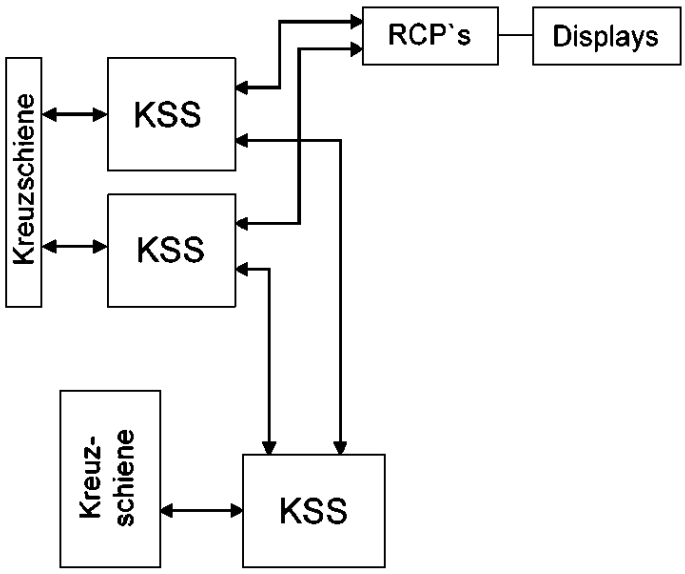


Bild 3: **Beispiel einer redundanten Verbindung der einzelnen Komponenten**

Die Einsatzgebiete sind:

- mobile Einsatzgebiete

Übertragungswagen

- stationäre Einsatzgebiete

  Ton- und Bildregieplätze

  Schneideräume ☐

Real computer scientists don't ...

* *Real Computer Scientists Don't Write Code*
* *Real Computer Scientists don't write code. They occasionally tinker with programming systems, but those are so high level that they hardly count (and rarely count accurately, precision is for applications).*
* *Real Computer Scientists don't comment their code. The identifiers are so long they can't afford the disk space.*
* *Real Computer Scientists don't write the user interface, they merely argue about what they should look like.*
* *Real Computer Scientists don't eat quiche. They shun Schezuan food since the hackers discovered it. Many Real Computer Scientists consider eating an implementation detail. (Others break down and eat with the hackers, but only if they can have ice cream for dessert).*
* *If it doesn't have a programming environment complete with interactive debugger, structure editor and extensive cross module type checking, Real Computer Scientists won't be seen tinkering with it. They may have to use it to balance their checkbooks, as their own systems can't.*
* *Real Computer Scientists don't program in assembler. They don't write in anything less portable than a number two pencil.*
* *Real Computer Scientists don't debug programs, they dynamically modify them. This is safer, since no one has invented a way to do anything dynamic to FORTRAN, COBOL or BASIC.*
* *Real Computer Scientists like C's structured constructs, but they are suspicious of it because it's compiled. (Only Batch Freaks and Efficiency Weirdos bother with compilers, they're soooo un- dynamic).*
* *Real Computer Scientists play Go. They have nothing against the concept of mountain climbing, but the actual climbing is an implementation detail best left to programmers.*
* *Real Computer Scientists admire ADA for its overwhelming esthetic value, but they find it difficult to actually program in, as it is much too large to implement. Most Computer Scientists don't notice this because they are still arguing over what else to add to ADA.*
* *Real Computer Scientists work from 5 pm to 9 am because that's the only time they can get the 8 megabytes of main memory they need to edit specs. (Real work starts around 2 am when enough MIPS are free for their dynamic systems). Real Computer Scientists find it hard to share 3081s when they are doing 'REAL' work.*
* *Real Computer Scientists only write specs for languages that might run on future hardware. Nobody trusts them to write specs for anything Homo Sapiens will ever be able to fit on a single planet.*
* *Real Computer Scientists like planning their own environments to use bit mapped graphics. Bit mapped graphics is great because no one can afford it, so their systems can be experimental.*
* *Real Computer Scientists regret the existence of PL/1, PASCAL and LISP. ADA is getting there, but it still allows people to make mistakes.*
* *Real Computer Scientists love the concept of users. Users are always real impressed by the stuff computer scientists are talking about; it sure sounds better than the stuff they are being forced to use now.*
* *Real Computer Scientists despise the idea of actual hardware. Hardware has limitations, software doesn't. It's a real shame that Turing machines are so poor at I/O.*
* *Real Computer Scientists love conventions. No one is expected to lug a 3081 attached to a bit map screen to a convention, so no one will ever know how slow their systems run.*

Bild 2 zeigt die verschieden Komponenten (Tastenpulte, PC, usw.) und wie sie mit den drei Schnittstellentypen der KSS verbunden werden

5) Sicherheitsmaßnahmen

Da ein Sendungsausfall in einem Studio unbedingt vermieden werden muß, werden aus Sicherheitsgründen folgende Maßnahmen getroffen:

- Mindestens doppelte Ausführung aller Systemkomponenten
- Wenn ein Teil ausfällt sollen andere übernehmen
- Selbständige Abschaltung, wenn Fehler erkannt werden
- Keine „Zentrale“ in der alle Konfigurationen gespeichert sind, damit das System weiterarbeiten kann, wenn genau diese ausfällt (jede Komponente enthält alle Daten)
- Ständiger Datenaustausch und -update zwischen allen „intelligenten“ Komponenten
- Rettung der Daten und des Zustandes des Systems bei Netzausfall

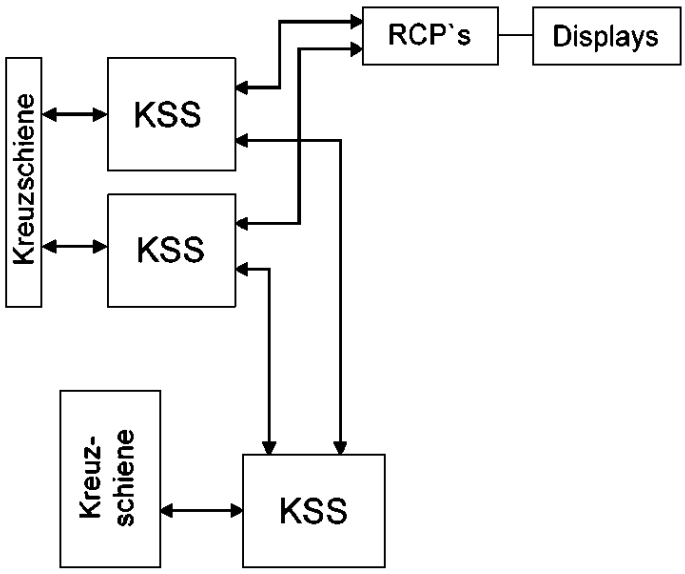


Bild 3: **Beispiel einer redundanten Verbindung der einzelnen Komponenten**

Die Einsatzgebiete sind:

- mobile Einsatzgebiete

Übertragungswagen

- stationäre Einsatzgebiete

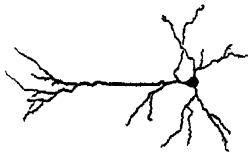
  Ton- und Bildregieplätze

  Schneideräume ☐

Real computer scientists don't ...

* *Real Computer Scientists Don't Write Code*
* *Real Computer Scientists don't write code. They occasionally tinker with programming systems, but those are so high level that they hardly count (and rarely count accurately, precision is for applications).*
* *Real Computer Scientists don't comment their code. The identifiers are so long they can't afford the disk space.*
* *Real Computer Scientists don't write the user interface, they merely argue about what they should look like.*
* *Real Computer Scientists don't eat quiche. They shun Schezuan food since the hackers discovered it. Many Real Computer Scientists consider eating an implementation detail. (Others break down and eat with the hackers, but only if they can have ice cream for dessert).*
* *If it doesn't have a programming environment complete with interactive debugger, structure editor and extensive cross module type checking, Real Computer Scientists won't be seen tinkering with it. They may have to use it to balance their checkbooks, as their own systems can't.*
* *Real Computer Scientists don't program in assembler. They don't write in anything less portable than a number two pencil.*
* *Real Computer Scientists don't debug programs, they dynamically modify them. This is safer, since no one has invented a way to do anything dynamic to FORTRAN, COBOL or BASIC.*
* *Real Computer Scientists like C's structured constructs, but they are suspicious of it because it's compiled. (Only Batch Freaks and Efficiency Weirdos bother with compilers, they're soooo un- dynamic).*
* *Real Computer Scientists play Go. They have nothing against the concept of mountain climbing, but the actual climbing is an implementation detail best left to programmers.*
* *Real Computer Scientists admire ADA for its overwhelming esthetic value, but they find it difficult to actually program in, as it is much too large to implement. Most Computer Scientists don't notice this because they are still arguing over what else to add to ADA.*
* *Real Computer Scientists work from 5 pm to 9 am because that's the only time they can get the 8 megabytes of main memory they need to edit specs. (Real work starts around 2 am when enough MIPS are free for their dynamic systems). Real Computer Scientists find it hard to share 3081s when they are doing 'REAL' work.*
* *Real Computer Scientists only write specs for languages that might run on future hardware. Nobody trusts them to write specs for anything Homo Sapiens will ever be able to fit on a single planet.*
* *Real Computer Scientists like planning their own environments to use bit mapped graphics. Bit mapped graphics is great because no one can afford it, so their systems can be experimental.*
* *Real Computer Scientists regret the existence of PL/1, PASCAL and LISP. ADA is getting there, but it still allows people to make mistakes.*
* *Real Computer Scientists love the concept of users. Users are always real impressed by the stuff computer scientists are talking about; it sure sounds better than the stuff they are being forced to use now.*
* *Real Computer Scientists despise the idea of actual hardware. Hardware has limitations, software doesn't. It's a real shame that Turing machines are so poor at I/O.*
* *Real Computer Scientists love conventions. No one is expected to lug a 3081 attached to a bit map screen to a convention, so no one will ever know how slow their systems run.*

HYBRIDES NEURONALES NETZ



Betreuer: Norbert BARTOS; Projektgruppe: Kurt BOHUSLAV, Thomas HINTERSTOISSER, Harald STEINMETZ, Gregor STRNAD

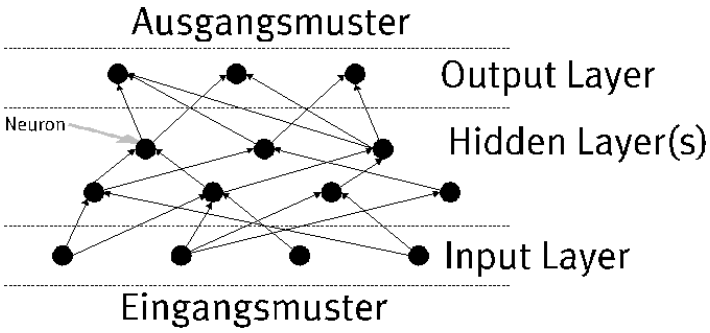
Harald STEINMETZ

1) Woher? Warum? Wieso?

Bereits in den 40er-Jahren wandte man sich der Technik der Neuronalen Netze zu. Diese weisen durch ihre parallele Datenverarbeitung eine hohe Geschwindigkeit auf. Durch ihre Spezialisierung auf ein bestimmtes Problem sind diese Architekturen wesentlich billiger als z.B. Parallelrechnersysteme. Neuronale Netze spiegeln in gewisser Weise die Gehirnstruktur von Lebewesen wider. Sie sind aus künstlich nachgebildeten Zellen (Neuronen) aufgebaut, die stark untereinander vernetzt sind. Durch Angabe der Gewichtung der einzelnen Verbindungen der Neuronen wird dem elektronischen Gehirn eine Struktur übermittelt. Es hat gelernt.

2) Was ist eigentlich ein Neuronales Netz?

Ein Neuronales Netz kann man sich als System vorstellen, das aus vielen Elementen besteht die miteinander verbunden sind. Diese Elemente verarbeiten die ihnen zur Verfügung gestellten Informationen. Die Elemente in einem Neuronalen Netz sind mathematische Modelle mit mehreren Ein- und Ausgängen. Bei den zu verarbeitenden Informationen kann es sich um Signale, Bitmuster oder Zahlenwerte handeln. Die bis jetzt erfolgten Erkenntnisse kann man sich wie in der folgenden Abbildung vorstellen:



Diese Abbildung soll die Vernetzung der Neuronen verdeutlichen. Die einzelnen Neuronen sind untereinander verbunden. Eine wichtige Erkenntnis ist, daß nicht alle Neuronen miteinander verbunden sind. Es sind nur so viele Verbindungen vorhanden, wie zur Erledigung der gestellten Aufgabe nötig sind. Diese Verbindungen erhalten sogenannte Gewichte, die aufgrund der Anforderungen, die vom Benutzer gestellt werden, festgelegt werden. Durch die Übermittlung dieser Gewichte erhält das Netz eine bestimmte Struktur. Diese Gewichte können verändert werden. Das Festlegen und Ändern von Gewichten in einem Neuronalen Netz bezeichnet man als den Lernvorgang, der sich folgendermaßen gestaltet:

Dem Neuronalen Netz werden Beispielmuster am Ein- und Ausgang präsentiert. Das Neuronale Netz wird nun dazu veranlaßt, seine Verbindungen zwischen den Neuronen und dessen Gewichte so zu setzen, daß das gewünschte Übertragungsverhalten angenommen wird. Eine besondere Eigenschaft eines Neuronalen Netzes ist, daß es in der Anwendungsphase weiterlernen kann. Man nennt dies Adaptives Verhalten.

Die Einteilung der Neuronen erfolgt in drei verschiedene Klassen. Die Neuronen des Input Layers sind direkt mit den Leitungen verbunden, die das Eingangsmuster an das Neuronale Netz weiterleiten. Analoges gilt für die Neuronen des Output Layers, die das Ausgangsmuster an die nachgeschalteten Geräte weiterleiten. Die Neuronen der Hidden Layers zeichnen sich durch die besondere Eigenschaft aus, daß ihre Verbindungen nur unter Neuronen existieren. Die Hidden Layers übernehmen daher die Hauptaufgabe der Neuronalen Netzes. Mit unseren bisherigen Erkenntnissen können wir somit sagen, daß ein Neuronales Netz ein System zur Informationsverarbeitung ist, das mit einfach vernetzten Elementen mit gerichteten Ein- und Ausgängen aus einem anliegenden Eingangsmuster ein Ausgangsmuster erzeugt.

3) Was sollte konstruiert werden?

Es war beabsichtigt, ein Gerät zu entwickeln, das in einem Laborübungsbetrieb die grundlegenden Funktionen des Lernens eines Neuronalen Netzes demonstriert.

Das Gerät besteht aus Einzelkomponenten, die je nach den gefragten Anforderungen während des Laborübungsbetriebes zusammengefügt werden können.

Bis wir jedoch zu diesen Angaben gelangten, vergingen einige Tage, denn anfangs war lediglich geplant, eine Sprachausgabe mit Hilfe eines Digitalen Neuronalen Netzes zu realisieren. Die Eingabe der Zeichen sollte über die PC-Tastatur erfolgen.

Es wurden Überlegungen angestellt, ob die sprachliche Ausgabe von eingelesenen Ziffern möglich wäre. Diese Erwägung wurde wegen zu hoher Komplexität der Aussprache von Ziffern, und Ziffernfolgen (Zahlen) verworfen, um sich der Aussprache von Worten der deutschen Sprache zu widmen.

Weiters wurde beschlossen, das zweite Projekt zum Thema „Neuronale Netze“ mit einzubeziehen. Es handelt sich dabei um ein Analoges Neuronales Netz, welches zur Zeichenerkennung eingesetzt wird. Anfangs bestand der Wunsch durch die Bewegung eines Stiftes, vergleichbar mit dem Schreiben von Buchstaben mit einem Bleistift, das Einlesen der Zeichen zu realisieren. Bei genauerer Überlegung dieses Problems stießen wir jedoch an die Grenzen des Möglichen.

Wie erwähnt, besteht ein Neuronalen Netz aus Neuronen, wobei jedes Neuron für sich selbst arbeitet, woraus zu schließen ist, daß alle am Eingang anliegenden Signale im Neuronalen Netz parallel weiterverarbeitet werden müssen.

Aufgrund dessen ist ein serielles Einlesen, wie es mit dem Stift erfolgen würde ein viel zu langsamer Vorgang. Das Neuronale Netz benötigt zur Erkennung von Buchstaben alle Informationen über das Aussehen des Buchstabens zur gleichen Zeit, also ein paralleles Anliegen der zu verarbeitenden Daten.

Die Lösung dieses Problems besteht darin, daß das Einlesen über eine beleuchtete Phototransistormatrix erfolgt, auf die der Anwender mit Buchstaben beschriebene Folien auflegen kann. Die aufgelegten Buchstaben werden vom Analogen Neuronalen Netz erkannt, die Auswertung der eingelesenen Buchstaben zu einem Wort erfolgt über das Digitale Neuronale Netz, das die Daten zur Sprachausgabe weiterleitet.

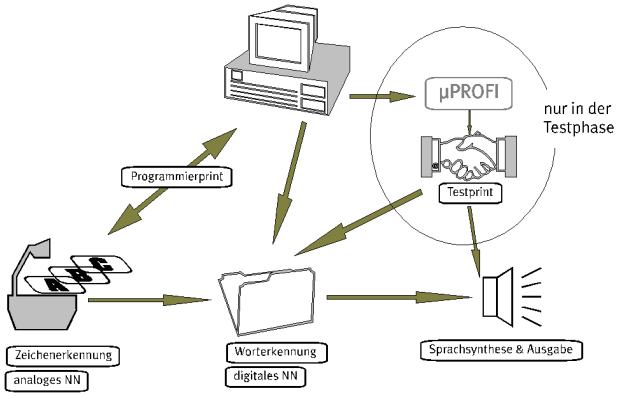
Weiters soll eine Testschaltung entwickelt werden, bevor die einzelnen Geräte gemeinsam in Betrieb genommen werden.

Somit konnten wir folgende Gerätespezifikationen festlegen:

- Es soll ein Laboraufbau zur Demonstration der Lernfähigkeit eines Neuronalen Netzes konstruiert werden.
- Das Gerät soll in Hardwareform aufgebaut werden.

- Auf Folien geschriebene Buchstaben werden über eine Phototransistormatrix eingelesen.
- Das angeschaltete Analoge Neuronale Netz bewerkstelligt die Buchstabenerkennung. Es wird ein Signal mit einer Länge von 8 Bit pro Buchstabe erzeugt.
- Dieses Signal wird in das Digitale Neuronale Netz weitergeführt. Hier erfolgt die Verarbeitung der eingelesenen Buchstaben zur Worterkennung.
- Die Sprachausgabe übernimmt die Signale des Digitalen Neuronalen Netzes, und bringt sie über den Sprachprozessor zum Lautsprecher.
- Zum Testen der Sprachausgabe und des Digitalen Neuronalen Netzes dient eine Testschaltung in Zusammenhang mit den µProfi-51.

Das folgende Bild veranschaulicht den Aufbau des Gerätes:

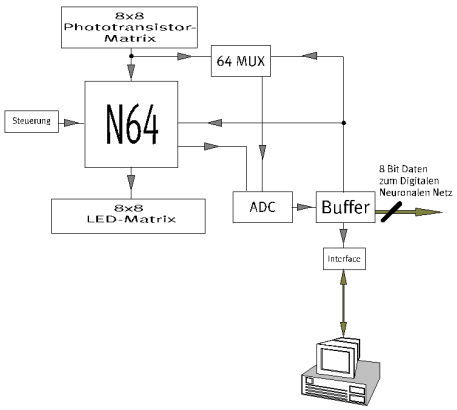


3.1) Analoges Neuronales Netz

Auf der Platine des Analogen Neuronalen Netzes befindet sich eine optische Eingabe- und Ausgabeschnittstelle. Eine Anordnung von 8x8 Phototransistoren ermöglicht die Erfassung eines Grauwertbildes, des Bildes, das auf die Matrix aufgelegt wurde. In unserem Fall sind dies die auf Folien geschriebenen Buchstaben. Weites befindet sich eine 8x8 Matrix von Low Current LEDs, die als direkte optische Ausgabe dienen. Das Herzstück dieser Platine bildet der analoge neuronale Prozessor N64 von Intel.

Die Ausgangswerte dieses Prozessors bewegen sich zwischen 0V und 5V. Damit kann ein Ausgang dieses Prozessors theoretisch unendlich viele Ausgangspegel annehmen. Aus diesem Grund kann die Zeichenerkennung durch die Zusammensetzung mehrerer Grauwerte viel genauer erfolgen als mit einem digitalen Baustein, der pro Ausgang nur zwei Ausgangswerte (in diesem Fall. Schwarz und Weiß) zur Verfügung stellen kann.

Über einen ADC und einen Buffer werden die erkannten Buchstaben in ein 8-Bit-Muster umgewandelt und dem Digitalen Neuronalen Netz zur Sprachauswertung zur Verfügung gestellt.



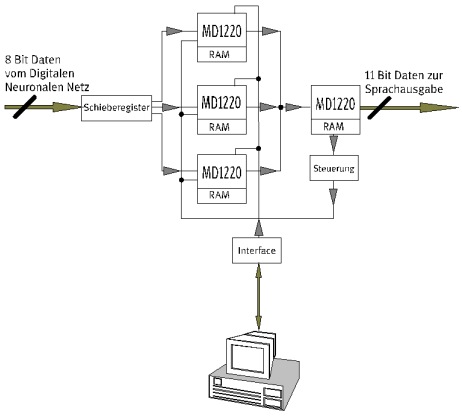
3.2) Digitales Neuronales Netz

Die Platine des Digitalen Neuronalen Netzes besteht aus vier neuronalen Prozessoren des Typs MD1220 der Firma Micro-Devices. Es handelt sich dabei um einen Non-Von-Neumann-Prozessor, der 8 Neuronen enthält. Jeder dieser Neuronen hat in der Grundkonfiguration 8 Neuroneneingänge, die mit einem 16 Bit Gewicht gewichtet werden. Die 16 Bit Gewichte für die Neuroneneingänge werden in einem externen RAM (256 Byte für einen Prozessor) gespeichert. Über eine der 3 Aktivierungsfunktionen des MD1220 wird nach den Berechnungen der entsprechende Neuronenausgang aktiviert bzw. deaktiviert. Wir erhalten eine digitale Ausgangsinformation.

Die Arbeitsfolgen des MD1220 lassen sich wie folgt beschreiben:

- Daten von den Eingängen einlesen
- Daten auf die 8 Neuronen verteilen
- Datenverteilung entsprechend gewichten
- Neuronen berechnen
- Neuronenausgänge aktivieren bzw. deaktivieren

Die Neuroprozessoren erhalten die eingelesenen Buchstaben vom Analogen Neuronalen Netz über 8 Leitungen. Jeder Buchstabe wird mit 8 Bit kodiert; das heißt pro Zyklus gelangen 8 Bit eines Buchstabens über die 8 Leitungen. Die Besonderheit an der Schaltung des Digitalen Neuronalen Netzes besteht darin, daß der eingelesene Buchstabe nicht sofort an den die Sprachausgabe weitergeleitet wird, denn dazu bräuchte man ja kein Neuronales Netz! Das Digitale Neuronale Netz analysiert jeweils 3 Buchstaben und wählt aus dieser Erkenntnis das richtige Phonem aus. Das folgende Bild illustriert die Arbeitsweise des Digitalen Neuronalen Netzes:



Jedes Bit der Buchstaben wird in ein 16 Bit Schieberegister geschoben, bis die Bits des ersten Buchstabens am Ende des Registers stehen. Die Schieberegister übernehmen in diesem Fall die Funktion des Input Layers, der ja nicht in jedem Fall aus Neuronen bestehen muß. Hat das Wort weniger als 16 Buchstaben, so werden die Schieberegister mit 0 aufgefüllt.

Der Hidden Layer des Digitalen Neuronalen Netzes hat 3 Prozessoren, wobei jeder Prozessor **einen** Buchstaben "betrachtet". Das heißt die jeweils letzten Stellen der Schieberegister sind mit den 8 Eingängen des 1. Prozessors verbunden, die vorletzten Stellen mit dem 2. Prozessor und analoges gilt für den 3. Buchstaben.

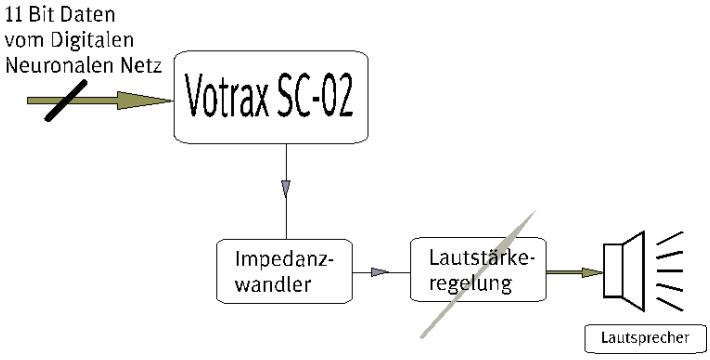
Die 3x8 Ausgänge des Hidden Layer werden auf 8 Leitungen zusammengefaßt und dem 4. Prozessor des Typs MD1220, der den Output Layer repräsentiert, zugeführt.

Die zur Steuerung benötigten Logikschaltungen, sowie die Schieberegister des Input Layers sind aus platzsparenden Gründen nicht diskret aufgebaut, sondern mittels eines FPGA-Bausteins der Firma XILINX realisiert.

3.3) Sprachausgabe:

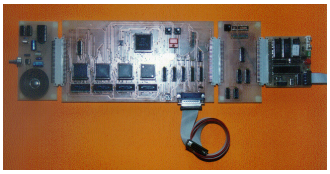
Zur Realisierung der Sprachausgabe wurde ein integrierter Schaltkreis der Firma Votrax gewählt. Der Votrax SC-02 ist ein Baustein, der auf 64 Adressen Phoneme der menschlichen Sprache gespeichert hat. Phoneme sind Bausteine der gesprochenen Sprache. Um die menschliche Sprache nachzubilden ist es nicht zielführend, Buchstaben aneinander zu reihen, und diese von einem Synthesizer aussprechen zu lassen. Zur Nachbildung der menschlichen Sprache sind die 26 Buchstaben des Alphabets noch lange nicht ausreichend. Wenn wir sprechen, so reihen wir Phoneme aneinander. So besteht das Wort „HALLO“ zwar aus fünf Buchstaben, jedoch nur aus drei Phonemen, nämlich *<ha>+</>+<oh>*. Nun gibt es eine Vielzahl dieser Phoneme. Der Votrax SC-02 kann 64 Phoneme der amerikanischen, britischen, deutschen, und französischen

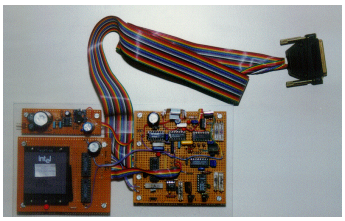
Sprache zur Verfügung stellen. Zugegeben ist das nicht sehr viel, aber für unseren Laboraufbau ist dieser Umstand mehr als zufriedenstellend. Der Votrax SC-02 erhält seine Daten vom Digitalen Neuronalen Netz über 11 Leitungen. Davon liegen 8 Bit an den Aktivierungseingängen zur Auswahl der Phoneme an. Über die verbleibenden 3 Leitungen werden die Register die zur Einstellung der Phonemlänge, Artikulation und Höhe angesteuert. Am Ausgang des SC-02 befindet sich ein Impedanzwandler, der das Ausgangssignal an den Lautsprecher über eine Lautstärkenregelung weiterleitet wie die folgende Abbildung demonstriert:

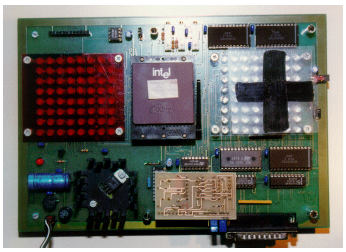


4) Testphase und Testschaltung

Zum Testen des SC-02 wurde der Baustein laut der beiliegenden Applikationsschaltung in den mitgelieferten Unterlagen beschalten. Die Auswahl der Adressen wurde vorerst mittels DIL-Schalter vorgenommen. Das ledigliche Einstellen der Phonemadressen führte zu keinem sichtlichen Erfolg. Aus den Datenblättern zu dem integrierten Schaltkreis der von der Distributorfirma nachgefordert wurde, war zu entnehmen, daß das Register mit mehreren Informationen zum gesprochenen Phonem geladen werden muß. Die Registerauswahl wurde vorerst ebenfalls mit DIL-Schaltern vorgenommen. Nachdem die Register und das gewünschte Phonem ausgewählt waren, konnte das gesprochene Phonem gehört werden. Um den Ausgang des Neuronalen Netzes zu simulieren, wurde der µProfi-51 zur Hilfe herangezogen. Über die Ausgangsleitungen und einen Adreßdecoder wurde die Testschaltung mit dem µProfi-51 in Betrieb genommen. Das Bestreben war, einige Phoneme hintereinander sprechen zu lassen. Dies wurde durch ein Assemblerprogramm verwirklicht. Genauso wie die 8 Bit Datenleitungen für die Sprachausgabe simuliert werden können, ist es auch möglich die 8 Bit Datenleitungen des Analogen Neuronalen Netzes zu simulieren. Zum Testen des Digitalen Neuronalen Netzes wurde diese Möglichkeit herangezogen.







Technische Daten:

Analoger Neuronaler Chip:	Intel N64
Digitaler Neuronaler Chip:	Micro Devices MD1220
Sprachausgabe Chip:	Votrax SC-02
Logikbaustein:	XILINX 8064
Schnittstellen:	RS232, Centronics
Betriebsspannung:	12 V DC

□

Mathematiker, Informatiker und Physiker in Irland

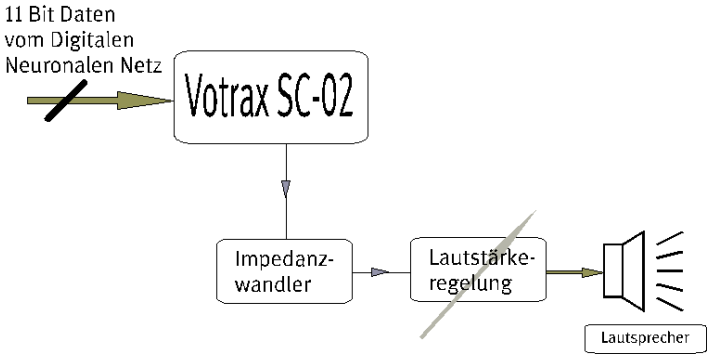
Ein Mathematiker, ein Informatiker und ein Physiker fahren nach Irland. Bei der Ankunft sehen sie eine Herde Schafe, darunter ein Schwarzes.

Physiker: „Ah, hier gibt es schwarze Schafe."

Informatiker: „Nein. Hier gibt es e i n schwarzes Schaf."

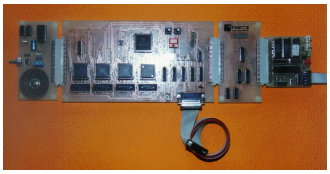
Mathematiker: „Alles Unsinn. Das einzige, was wir wissen, ist: Hier gibt es mindestens ein Schaf, das von mindestens einer Seite schwarz ist."

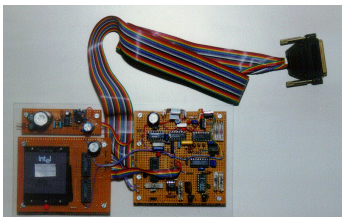
Sprache zur Verfügung stellen. Zugegeben ist das nicht sehr viel, aber für unseren Laboraufbau ist dieser Umstand mehr als zufriedenstellend. Der Votrax SC-02 erhält seine Daten vom Digitalen Neuronalen Netz über 11 Leitungen. Davon liegen 8 Bit an den Aktivierungseingängen zur Auswahl der Phoneme an. Über die verbleibenden 3 Leitungen werden die Register die zur Einstellung der Phonemlänge, Artikulation und Höhe angesteuert. Am Ausgang des SC-02 befindet sich ein Impedanzwandler, der das Ausgangssignal an den Lautsprecher über eine Lautstärkenregelung weiterleitet wie die folgende Abbildung demonstriert:

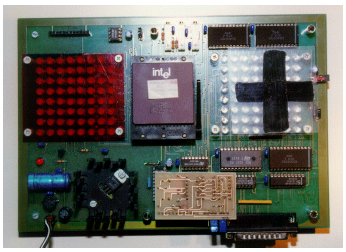


4) Testphase und Testschaltung

Zum Testen des SC-02 wurde der Baustein laut der beiliegenden Applikationsschaltung in den mitgelieferten Unterlagen beschalten. Die Auswahl der Adressen wurde vorerst mittels DIL-Schalter vorgenommen. Das ledigliche Einstellen der Phonemadressen führte zu keinem sichtlichen Erfolg. Aus den Datenblättern zu dem integrierten Schaltkreis der von der Distributorfirma nachgefordert wurde, war zu entnehmen, daß das Register mit mehreren Informationen zum gesprochenen Phonem geladen werden muß. Die Registerauswahl wurde vorerst ebenfalls mit DIL-Schaltern vorgenommen. Nachdem die Register und das gewünschte Phonem ausgewählt waren, konnte das gesprochene Phonem gehört werden. Um den Ausgang des Neuronalen Netzes zu simulieren, wurde der µProfi-51 zur Hilfe herangezogen. Über die Ausgangsleitungen und einen Adreßdecoder wurde die Testschaltung mit dem µProfi-51 in Betrieb genommen. Das Bestreben war, einige Phoneme hintereinander sprechen zu lassen. Dies wurde durch ein Assemblerprogramm verwirklicht. Genauso wie die 8 Bit Datenleitungen für die Sprachausgabe simuliert werden können, ist es auch möglich die 8 Bit Datenleitungen des Analogen Neuronalen Netzes zu simulieren. Zum Testen des Digitalen Neuronalen Netzes wurde diese Möglichkeit herangezogen.







Technische Daten:

Analoger Neuronaler Chip:	Intel N64
Digitaler Neuronaler Chip:	Micro Devices MD1220
Sprachausgabe Chip:	Votrax SC-02
Logikbaustein:	XILINX 8064
Schnittstellen:	RS232, Centronics
Betriebsspannung:	12 V DC

□

Mathematiker, Informatiker und Physiker in Irland

Ein Mathematiker, ein Informatiker und ein Physiker fahren nach Irland. Bei der Ankunft sehen sie eine Herde Schafe, darunter ein Schwarzes.

Physiker: „Ah, hier gibt es schwarze Schafe."

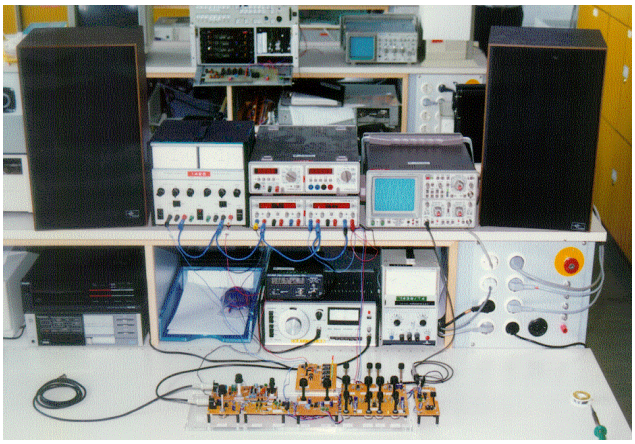
Informatiker: „Nein. Hier gibt es e i n schwarzes Schaf."

Mathematiker: „Alles Unsinn. Das einzige, was wir wissen, ist: Hier gibt es mindestens ein Schaf, das von mindestens einer Seite schwarz ist."

Audiomischpult

Betreuer: Gottfried Patels

Oliver Singer



ALLGEMEINES

Ein Mischpult ist die zentrale Einheit eines jeden Studios oder einer Bühnenanlage, und es hat verschiedene Aufgaben zu erfüllen. Die von den einzelnen Tonquellen (z.B. Mikrophon, Bandgerät, CD-Spieler, DAT-Recorder,...) kommenden Signale werden zuerst auf einen einheitlichen Arbeitspegel gebracht. Diese Tonsignale können dann klanglich beeinflußt, in ihrer Lautstärke aufeinander abgestimmt und rückwirkungsfrei zusammengemischt werden. Es ist auch eine Verteilung auf verschiedene Leitungswege und eine Zumischung von Effekten möglich. Neben der Beeinflussung des Pegels und der Klangfarbe ist auch die Beeinflussung der Richtungszuordnung von grundlegender Bedeutung. Schließlich werden die bearbeiteten Signale auf einem Tonband oder anderen Tonträgern aufgezeichnet, oder aber einer Endverstärkerstufe zugeführt. Damit alle gestalterischen Forderungen erfüllt werden können, sind zum Teil sehr umfangreiche Mischpulte erforderlich.

PROJEKTBESCHREIBUNG

Geplant ist ein analoger Tonregietisch, der qualitativ hochwertigen Studiogeräten entspricht.

Für die Erfüllung aller Aufgaben sind einzelne Funktionsgruppen zuständig, die wiederum aus mehreren Baugruppen bestehen.

Das Mischpult ist in Streifentechnik aufgebaut, d.h. die Funktionsgruppen sind mechanisch abgegrenzte, austauschbare Einheiten. Daher ist das Mischpult fast beliebig erweiterbar.

Die einzelnen Funktionsgruppen werden nun näher erklärt:

1) Eingangskanäle

Hier werden die von den einzelnen Tonquellen (z.B. Mikrophon, CD-Spieler, Bandgerät, DAT-Recorder) kommenden Signale auf einen einheitlichen Arbeitspegel gebracht. Der symmetrische Mikrophoneingang umfaßt einen genügend großen Regelbereich, um sowohl dynamische, als auch Kondensatormikrophone anschließen zu können. Eine Phantomspeisung für Kondensatormikrophone ist genauso vorgesehen, wie eine Phasenumkehr, mehrere Meßpunkte für die Peak-Detection, ein Insert-Punkt (um Zusatzgeräte einschleifen zu können) und vier AUX-Abzweige. Mir den eingebauten Filtern und dem Equalizer kann das Tonsignal klanglich, und mit dem Kanal-Fader in der Lautstärke beeinflußt werden. Das Panorama-Potentiometer sorgt für die räumliche Verteilung des Signals (zur Erzeugung eines Stereo-Effektes). Eine Zuordnung zu den verschiedenen Summenkanälen ist ebenfalls vorgesehen.

2) Mastermodul

In dieser Funktionsgruppe erfolgt eine rückwirkungsfreie Summierung der Eingangskanäle, und die Beeinflussung der Lautstärke des Gesamtsignals. Der Ausgang des Mischpultes liegt ebenfalls hier.

3) Subgruppen

Bei diesen Einheiten handelt es sich um einfache Zwischensummen. Man kann hier die Signale betreffender Eingangskanäle zusammenfassen und dem Mischpult entnehmen (z.B. für Mehrspuraufnahmen), oder aber an das Mastermodul weiterleiten.

4) Abhörmodul

Diese Funktionsgruppe enthält im wesentlichen einen Kopfhörerverstärker und einen Wahlschalter, mit dem man die verschiedenen Tonsignale an den unterschiedlichsten Stellen des Pultes abhören kann.

5) AUX-Mastermodule

Die abgezweigten Signale der einzelnen Eingangskanäle werden hier summiert, und können dem Mischpult entnommen werden (um z.B. Effektgeräte anzusteuern).

6) Meter-Brücke

Hier befinden sich die Aussteuerungsanzeigen zur Pegelüberwachung der Eingangskanäle, der Subgruppen und des Mastermoduls. □

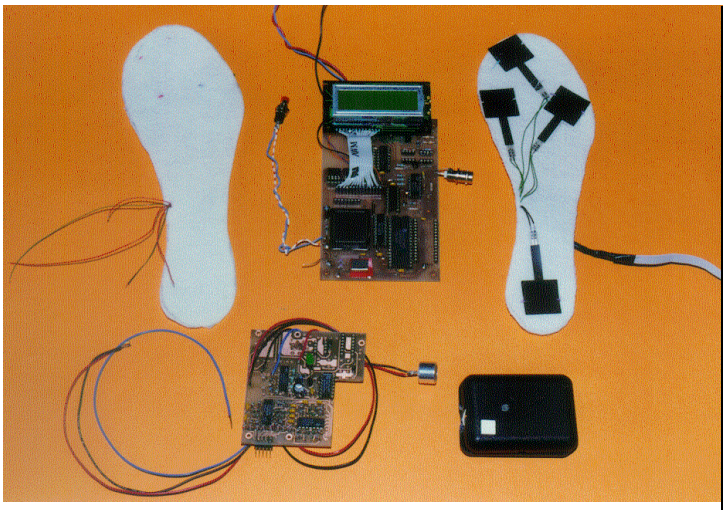
Real software engineers don't ...

* *Real Software Engineers Don't Read Dumps*
* *Real Software Engineers don't read dumps. They never generate them, and on the rare occasions that they come across them, they are vaguely amused.*
* *Real Software Engineers don't comment their code. The identifiers are so mnemonic they don't have to.*
* *Real Software Engineers don't write applications programs, they implement algorithms. If someone has an application that the algorithm might help with, that's nice. Don't ask them to write the user interface, though.*
* *Real Software Engineers eat quiche.*
* *If it doesn't have recursive function calls, Real Software Engineers don't program in it.*
* *Real Software Engineers don't program in assembler. They become queasy at the very thought.*
* *Real Software Engineers don't debug programs, they verify correctness. This process doesn't necessarily involve executing anything on a computer, except perhaps a Correctness Verification Aid package.*
* *Real Software Engineers like C's structured constructs, but they are suspicious of it because they have heard that it lets you get "close to the machine."*
* *Real Software Engineers play tennis. In general, they don't like any sport that involves getting hot and sweaty and gross when out of range of a shower. (Thus mountain climbing is Right Out). They will occasionally wear their tennis togs to work, but only on very sunny days.*
* *Real Software Engineers admire PASCAL for its discipline and Spartan purity, but they find it difficult to actually program in. They don't tell this to their friends, because they are afraid it means they are somehow Unworthy.*
* *Real Software Engineers work from 9 to 5, because that is the way the job is described in the formal spec. Working late would feel like using an undocumented external procedure.*
* *Real Software Engineers write in languages that have not actually been implemented for any machine, and for which only the formal spec (in BNF) is available. This keeps them from having to take any machine dependencies into account. Machine dependencies make Real Software Engineers very uneasy.*
* *Real Software Engineers don't write in ADA, because the standards bodies have not quite decided on a formal spec yet.*
* *Real Software Engineers like writing their own compilers, preferably in PROLOG (they also like writing them in unimplemented languages, but it turns out to be difficult to actually RUN these).*
* *Real Software Engineers regret the existence of COBOL, FORTRAN and BASIC; PL/1 is getting there, but it is not nearly disciplined enough; far too much built-in function.*
* *Real Software Engineers aren't too happy about the existence of users, either. Users always seem to have the wrong idea about what the implementation and verification of algorithms is all about.*
* *Real Software Engineers don't like the idea of some inexplicable and greasy hardware several aisles away that may stop working at any moment. They have a great distrust of hardware people, and wish that systems could be virtual at ALL levels. They would like personal computers (you know no one's going to trip over something and kill your DFA in mid-transit), except that they need 8 megabytes to run their Correctness Verification Aid packages.*
* *Real Software Engineers think better while playing WFF 'N' PROOF.*

BODENKONTAKTSOHLE

Betreuer: Christian Kollmitzer

Günther Gmeindl, Norbert Winkler



Einleitung

In der orthopädischen Diagnostik ist es oft notwendig die Standphasen, Schrittlänge, Geschwindigkeit oder Kadenz eines Patienten, der durch einen Unfall oder von Grund auf gehbehindert ist, zu messen und auszuwerten. Um ein Gerät zu entwickeln das genau diese Forderungen erfüllt, ist gemeinsam mit dem Ganglabor AKH-Wien, eine Spezifikation ausgearbeitet worden.

Der Prototyp der Bodenkontaktsohle ist bereits im Speziallehrgang 94/95 von Robert Berger und Markus Bickert entwickelt worden.

Im Speziallehrgang 95/96 soll nun ein Gerät mit den oben geforderten Erweiterungen bis hin zur Kleinserienreife entwickelt werden.

Problem-stellung

Einerseits soll das Gerät durch eine Taste am Gerät gesteuert werden können, andererseits soll die Messung synchron mit der verwendeten Lichtschranke starten und stoppen. D.h. es muß der Lichtschrankenimpuls über einen Sender an das Gerät gesendet werden.

Bei der Start-/Stopauslösung standen drei Möglichkeiten zur Auswahl:

1. Infrarotübertragung
2. Funkübertragung
3. Ultraschallübertragung

ad1) Die Schwierigkeit bei einer IR-Übertragung besteht darin, daß durch ungewollte oder unvorhersehbare Bewegungen (schwingender Arm) des Patienten der Lichtstrahl unterbrochen werden kann.

ad2) Da alle Geräte zur Funkübertragung postgenehmigt sein müssen, haben wir diese Möglichkeit von vornherein ausgelassen.

ad3) Es wird durch einen Lautsprecher ein 41kHz Signal abgegeben. Dieses Ultraschallsignal wird von einem Mikrofon am Gerät empfangen.

Spezifikation

- 2 Sensormatten mit je 4 Sensoren

Es handelt sich hier um handelsübliche Drucksensoren, mit einem Widerstandsbereich von einigen kOhm bis 2-3MOhm.

- Start/Stopimpuls durch Lichtschranke

Bei der Übertragung des Lichtschrankensignals wurde eine Ultraschallübertragung gewählt.

- Sync-Signal

Bei dem Sync- Signal handelt es sich um ein 200Hz Sinussignal. Mit diesem Signal sollen andere Geräte zum Meßvorgang synchron geschaltet werden können.

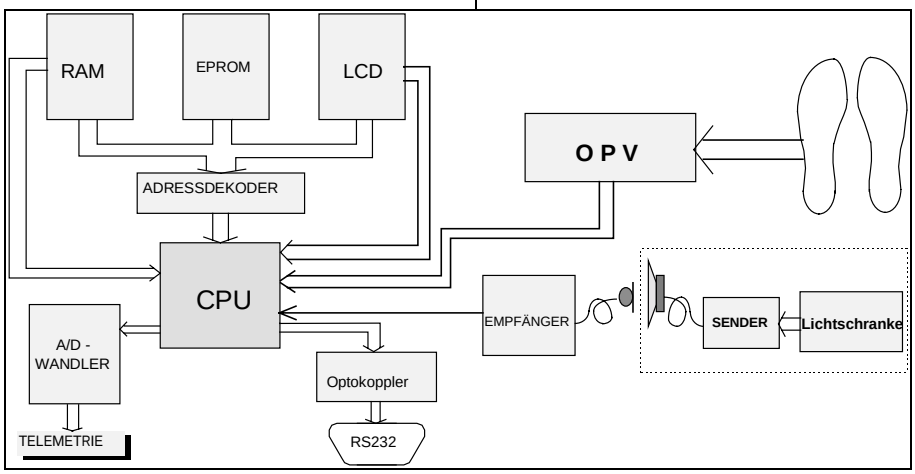
- Serielle Schnittstelle

Alle aufgenommen Daten sollen in binärer Form über eine serielle Schnittstelle an einen PC übertragen werden können.

Weiters soll mit der Schnittstelle ein serieller Drucker angesteuert werden können.

- Darstellung der Meßdaten am Display

Das Gerät verfügt über eine zweizeilige LCD- Matrixanzeige die folgende Daten anzeigen soll:



Blockschaltbild

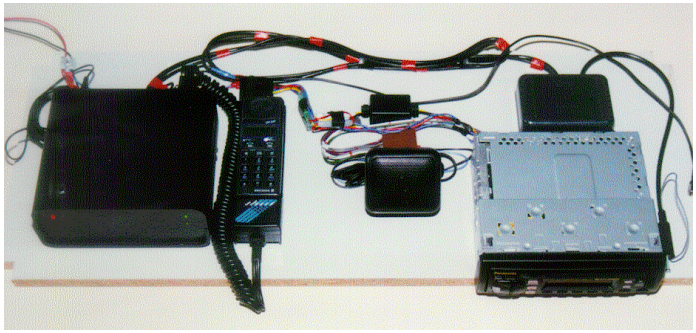
Geschwindigkeit	v = xxx cm/s oder m/min
Schrittlänge	sl = xxx cm oder m
Kadenz	c = xxx st/min oder st/s
Zyklusdauer	st = xxx s
Linke Standphase	Spl = in %
Rechte Standphase	Spr = in %

Mit einer Taste soll zwischen den Zeilen 1,2 3,4 und 5,6 umgeschaltet werden können. ☐

FREISPRECHANLAGE

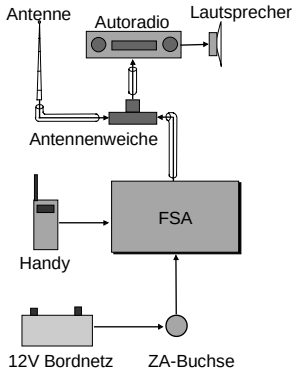
Betreuer: Gottfried Patels

Christian Hofer, Stefan Lampert



Die Freisprechanlage wurde im Auftrag einer österreichischen Firma entwickelt.

Installation



1. Merkmale der Freisprechanlage

- **Einfache Installation:** Anschluß an Autoradio über Antennenkabel und an die Bordspannung über Zigarettenanzünderbuchse.
- **Mit allen herkömmlichen Autoradios:** Die Freisprechanlage funktioniert mit allen herkömmlichen Autoradios.
- **Automatischer Sendersuchlauf:** Die Freisprechanlage sucht automatisch die eingestellte Empfangsfrequenz des Autoradios, unterdrückt bei einem Telefongespräch den momentan empfangenen Rundfunksender und macht das Gespräch über die Lautsprecherboxen des Radios hörbar.
- **Automatischer Sendernachlauf:** Sollte während des Gespräches die Empfangsfrequenz des Autoradios geändert werden, wird automatisch auf diese neue Frequenz nachgeregelt.
- **Kurze Suchzeit:** Der Sendersuchlauf ist äußerst schnell. Die maximale Suchzeit liegt bei 0.5 Sekunden.
- **Fixe Sendefrequenz:** Auf Wunsch ist auch eine fixer Sendeplatz einstellbar auf dem das Telefongespräch mit dem Radio empfangen werden kann.
- **Handy:** Jedes Handy mit der passenden Anschlußbuchse für die Handyhalterung ist geeignet.
- **Handyhalterung:** Das Handy kann in einer stabilen Handyhalterung befestigt werden.
- **Laden des Akkus:** Befindet sich das Handy in der Halterung, wird der Akku aufgeladen.
- **Rückkopplungsunterdrückung:** Die fix eingebaute Klangwaage des Handys schaltet automatisch zwischen Hören und Sprechen um und schließt somit eine Rückkopplung aus (Schaltplan Version 1).

2. Funktionsprinzip

Da die Local-Oszillator-Frequenz (LO) des Autoradios die Filter an der Antennenbuchse ausreichend stark durchdringt, kann diese Frequenz zur Erkennung der eingestellten Empfangsfrequenz verwendet werden. Die LO-Frequenz liegt dabei immer genau 10.7 MHz über der Empfangsfrequenz.

Zuerst muß daher mit dem **FM-DEMODULATOR** diese LO-Frequenz empfangen und demoduliert werden. Um zu starke Störsignale von regionalen Rundfunksendern zu vermeiden, und eine Abstrahlung der Frequenzen von der Freisprechanlage auszuschließen, wird die Autoantenne durch die **ANTENNENWEICHE** im Betrieb weggeschaltet.

Die Empfangsfrequenz des Tuners überstreicht mittels des **SWEEP** den Frequenzbereich, in dem sich der LO befinden kann. Aus der Feldstärke der empfangenen Frequenz und dem resultierenden demodulierten NF-Signal kann geschlossen werden, ob irgendein Rundfunksender oder der LO des Autoradios empfangen wird. Diese Aufgabe wird vom **LO-DETEKTOR** gelöst, der den Integrator stoppt, wenn die Feldstärke ausreichend groß und das demodulierte NF-Signal ausreichend klein ist. Wird also die LO-Frequenz empfangen, bleibt der Tuner auf dieser Frequenz stehen und liefert sie dem **MISCHER**.

Im **MODULATOR** wird ein 10.7 MHz-Träger mit der NF-Spannung vom Handy FM-moduliert. Diese 10.7 MHz werden nun mit der gefunden LO-Frequenz gemischt, wobei sich Summe und Differenz beider Frequenzen ergeben. Genau diese LO-Frequenz minus 10.7MHz ist die Empfangsfrequenz des Autoradios. Diese Frequenz wird im **VERSTÄRKER** verstärkt, damit das Signal stark genug ist, Reste von Rundfunksendern zu überdecken. Über die Antennenweiche gelangt das Signal zurück in das Autoradio.

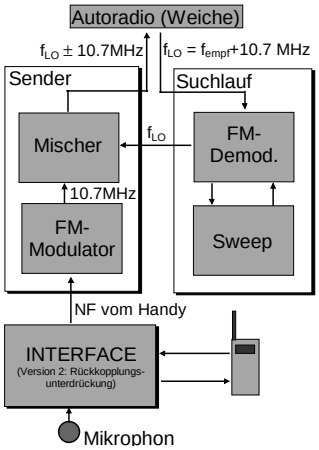
Für den störungsfreien Betrieb, auch bei sehr starken Schwankungen im Bordnetz, sorgt ein **DC-DC-KONVERTER**, der eine konstante und saubere Spannung bereitstellt. Ohne der konstanten Spannung würde der eingerastete Suchlauf wieder rausspringen (Schwankungen an Kapazitätsdiode). Außerdem werden diverse Betriebsspannungen bereitgestellt.

Das **INTERFACE** sorgt für eine Anpassung an die Signale vom Handy. Über eine Transistorstufe kann das Handy die Schaltung aktivieren. Ein Eingangs- und Ausgangsverstärker erlaubt die Anpassung der Signalpegel von und zu dem Handy.

Das Handy enthält eine fix eingebaute Klangwaage, die eine Rückkopplung verhindert.

Die Schaltung **Version 2** unterscheidet sich von **Version 1** nur durch eine zusätzliche Rückkopplungsunterdrückung. Möglicherweise ist die Klangwaage des Handys (mit einem Mikrocontroller) ausschaltbar, wobei nun unsere Rückkopplungsunterdrückung einen Duplex-Betrieb ermöglichen würde.

Funktionsprinzip



3. Momentaner Entwicklungsstand

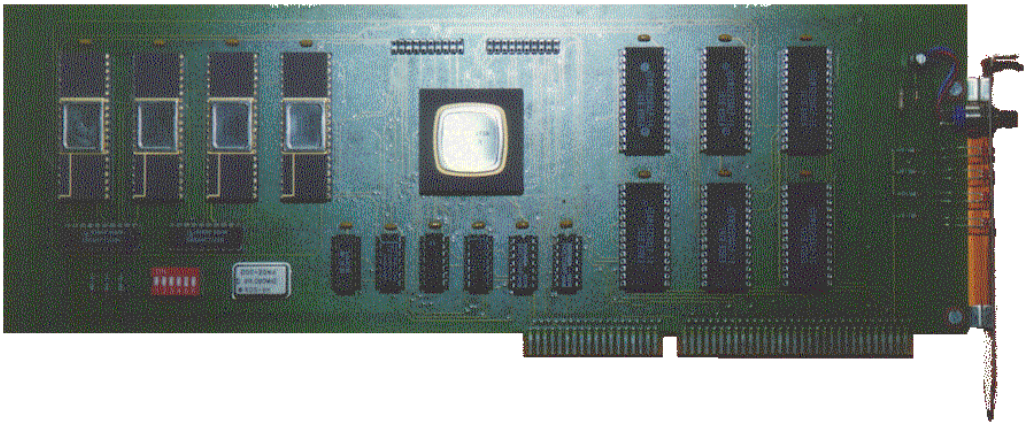
Die Schaltung ist fertig entwikkelt und wurde der Auftragsfirma übergeben. Die Firma übernimmt die Anpassung an SMD-Technik und die Layout-Entwicklung bis zu Serienreife.

Es wurde außerdem ein Prototyp fertig aufgebaut und steht als Brettaufbau zu Demonstrationszwecken zur Verfügung. □

Datenflußrechner

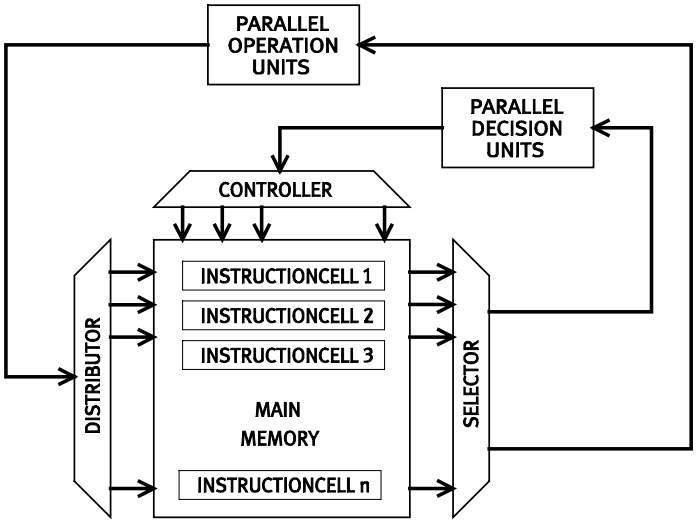
Betreuer: Norbert Bartos

Aleksandar Radovanovic, Theobald Seyffertitz



1) Einleitung

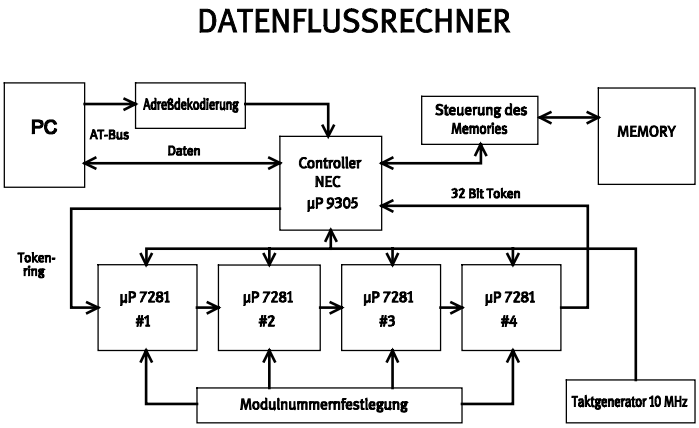
Unter den Namen Datenflußrechner, genauer gesagt Datenflußprozessor, ist eine eigene Prozessorarchitektur gemeint, die im Vergleich zur „Von Neumann"-Architektur keinen festen Programmablauf hat. Es ist eine Art parallelarbeitendes System. Vom Prinzip her heißt das, daß es keine sequentielle Abarbeitung der Daten gibt, sondern ein paralleles Abarbeiten der Eingangswerte. Jede Rechenoperation benötigt daher Eingangswerte und liefert Ausgangswerte. Eine Operation wird ausgeführt, wenn alle ihre Eingangswerte vorhanden sind. Der Datenflußprozessor verteilt dann die Ausgangswerte an diejenigen Rechenoperationen, welche diese Werte als Eingangsgrößen benötigen. Der Programmablauf ist somit datenabhängig, daher auch der Name Datenflußprozessor.



2) Funktionen

Die Einschubkarte wird vorwiegend im Labor verwendet werden. In Laborübungen könnte man dann zeigen, wie diese Prozessoren arbeiten und welche neuen Möglichkeiten mit dieser Prozessorarchitektur in der Programmierung entstehen. Die verwendete Programmiersprache ist Assembler. Es gibt aber auch eine graphische Programmieroberfläche.

Die einzelnen Prozessoren und der Controller bilden den Tokenring, d.h. die Datenpakete die diesen durchlaufen sind 32bit breit (Token) und werden zu je 2*16bit Datenworten gemultiplext über den Bus geschickt. Jedes Datenpaket enthält im höherwertigen Datenwort die Modulnummer. Diese Modulnummer bestimmt das Ziel des Datenpaketes (die Modulnummern der Prozessoren sind festgelegt: MN0:PC; MN1-6: Speicher; MN11-14: Prozessoren). Entspricht die Modulnummer des Prozessors nicht der des Datenpaketes, so wird das Datenpaket solange weitergeschickt bis es den richtigen Prozessor erreicht hat (Tokenring).



3) Spezifikationen

- als AT-Einschubkarte mit 4 Datenflußprozessoren ausgeführt
- 2 Ports für Messungen
- Controller (µP 9305) von der Firma NEC der die Kommunikation zwischen PC und den Datenflußprozessoren erleichtert
- Speicher von 64k*18 Bit, der für die Assemblerprogramme der Prozessoren verwendet wird
- Controller und Prozessoren sind mit 10 MHz getaktet
- die Adreßdekodierung für den Print ist über DIP-Schalter einstellbar, im Bereich von 300h-31Fh (Prototypenkarten) ☐

Impressum

PCNEWS edit-Herausgeber

ADIM-Graz	Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik
✉	Postfach 37, 8028 Graz (Klaus Scheiber)
E✉	InterNet adimgraz@borg-graz.ac.at FidoNet 2:316/88.105 SchulNetz 22:100/1.105
Online	Internet http://www.borg-graz.ac.at/~adimgraz/adimgraz.htm listserv@ccc.or.at SUBSCRIBE ADIMINFO
KONTO DVR	PSK, Blz. 60000, Kto. 7.224.353, ADIM 0547328
PCNEWS edit	empfohlen

ADIM-Wien	Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik
✉	Postfach 23, 1191 Wien (Martin Weissenböck)
☎	**TEL**: 369 88 59 **FAX**: 369 88 59-7 **ANR**: 369 88 59-8
🕒	Mo ab 20:00 tel. Sprechstunde, außer in der Zeit der Wiener Schulferien
E✉	InterNet mweissen@ping.at FidoNet 2:310/1.35 SchulNetz 22:102/1.135
🖷	**His Master's Voice (siehe ccc)**
Online	Fidonet *Files:* ADIM Internet listserv@ccc.or.at SUBSCRIBE ADIMINFO
KONTO DVR	PSK, Blz. 60000, Kto. 7.254.969, ADIM 0547328
PCNEWS edit	empfohlen

CCC	Computer Communications Club
✉	Flurschützstraße 36/12/5, 1120 Wien (Werner Illsinger)
☎	**TEL**: 810 17 09 **FAX**: 810 17 08 **ANR**: 810 17 09
Clublokal	Flurschützstraße 36, Stiege 9, Partyraum, 1120, Wien
🕒	Jän-Mo, Feb-Di, Mär-Mi..., ab 18:30
E✉	InterNet office@ccc.or.at CIS: 100024,1325 FidoNet 2:310/1.0 SchulNetz 22:102/1.0
🖷	**His Master's Voice** ☎ :(0222)-815 48 71[USR-V.34/V.42bis, 8N1] *Fido*: 2:310/1.0 *SchulNetz*: 22:102/1.0 ☎-2 (0222)-810 17 08[USR-V.34+/V.42bis, 8N1] ☎-ISDN (0222)-810 17 08 *Fido-Adresse*: 2:310/1000.0
Online	Fidonet *Echo:* CCC, MASTERS-ANNOUNCE, MASTERS-INFO Internet http://www.ccc.or.at/ listserv@ccc.or.at SUBSCRIBE CCC
KONTO DVR	PSK, Blz. 60000, Kto. 9.621.301, ccc 0771627
Jahresbeitrag	platin (400,-), gold (310,-), silber (240,-), bronze (150,-)
PCNEWS edit	bei Platin- und Silber-Mitgliedschaft inkludiert, 🖷bei Platin- und Gold-Mitgliedschaft inkludiert
Vorstand	*Präsident/Stv.* Werner ILLSINGER, Franz FIALA *Kassier/Stv.* Eva ILLSINGER, Gerwald OBERLEITNER *Schriftführer* Andreas HOFFMANN *Rechnungsprüfer* Lukas HEINDL, Helmut SCHLUDERBACHER

CLUB AT	CLUB für (A)utomatisierungs(T)echnik
✉	Wickenburggasse 8, 8010 Graz (James Steinbauer)
☎ 0316	**TEL**: 676 677-9 **FAX**: 676 677-19 **ANR**: 676 677-9
Clublokal	Wickenburggasse 8, 8010, Graz
🕒	Do,Fr,Sa,20:00..22:00, Treffen: jeden 2 Freitag im Monat
E✉	InterNet James.Steinbauer@2-316-11-0.ccc.or.at FidoNet 2:316/11.0, 2:316/111.0
🖷	**CLUB AT-Mailbox** ☎: (0316)-68 71 30[USR 28k8, V.34/V.42bis, 8N1] *Fido*: 2:316/11 ☎-ISDN: (0316)-68 01 80[68 01 80 *Fido-Adresse*: 2:316/111
KONTO DVR	PSK, Blz. 60000, Kto. 92.025.392, CLUB AT 0735485
Jahresbeitrag	S 1200,-
PCNEWS edit	liegt auf, 🖷inkludiert
Vorstand	*Präsident/Stv.* James M.STEINBAUER, Reinhold STACHL *Kassier* Andreas RAAB *Schriftführer* Michael SKOTNIK

Hyperbox	Verein zur Förderung und Erforschung moderner Kommunikationstechnologien
✉	Reitherstraße 19, 4060 Leonding (Martin Reinsprecht)
☎ 07229	**TEL**: 76472
E✉	InterNet root@elektor.htl-leonding.ac.at FidoNet 2:314/50.1
🖷	**Hyperbox** ☎: (0732)-67 75 67[V.34] *Fido-Adresse*: 2:314/50 ☎-2: (0732)-67 14 38-40[SLIP/PPP]
Online	Internet http://www.rema.co.at/hyperbox/
KONTO DVR	PSK, Blz. 60000, Kto. 78.337.657, Christian Olescher 0735485
Jahresbeitrag	Schüler: 300,- sonst 450,-
PCNEWS edit	S 150,-/5 Hefte, 🖷inkludiert
Vorstand	*Vorsitzender/Stv.* Martin REINSPRECHT, Christian KNEISSL *Kassier/Stv.* Christian OLESCHER, Martin POLAK *Schriftführer* Bernhard JUNGWIRTH *Rechnungsprüfer* Gerhard BÖHM, Alexander KOPPELHUBER

MCCA	Multi Computer Communications Austria
✉	Postfach 143, 1033 Wien (Josef Sabor)
☎	**FAX**: 811 17 - 484
Clublokal	Ungargasse 69, 1030, Wien
🕒	monatlich, meist dritter Dienstag, ab 18:00
E✉	InterNet info@mcca.ping.at CIS: 100775,1777 FidoNet 2:313/9.24
Online	Fidonet *Echo:* PAN.AUS Internet http://www.ping.at/mcca/ at.fido.btx
KONTO DVR	PSK, Blz. 60000, Kto. 93.001.133, MCCA 0536229
Jahresbeitrag	Firma: 1200,- Einzel: 600,- Ermäßigt: 300,-
PCNEWS edit	S 100,-/5 Hefte
Vorstand	*Obmann/Stv.* Josef SABOR, Peter MARSCHAT *Kassier/Stv.* Hans Jörg LOCHMANN, Marcus POLLAK *Schriftführer* Leopold SCHEIDL

Impressum, Offenlegung

Grundlegende Richtung	Auf Anwendungen im Unterricht bezogene Informationen über Personal-Computer-Systeme. Berichte über Veranstaltungen der Herausgeber.
Erscheint	5 mal pro Jahr, Feb, Apr, Jun, Sep, Nov
Medieninhaber	PCNEWS-*Eigenverlag*
Herausgeber	ADIM, CCC, CLUB-AT, HYPERBOX, MCCA, PCC-S, PCC-TGM
Belichtung	**PCG**, Prodat Computer Grafik Maroltingergasse 67, 1160 Wien
☎	**TEL**: 495 58 01, **FAX**: 495 58 01-22
Druck	**Zlínské tiskárny a.s.**, POBOX 79, CZ-76097 Zlín-Kudlov,
☎	**TEL**: 0042-67-721 33 06, **FAX**: 0042-67-721 13 85
Versand	**CONCEPT**, Baumgasse 52, 1030 Wien,
☎	**TEL**: 713-59-41, **FAX**: 713-87-72
Vertrieb	**MORAWA**, Pressevertrieb, Wollzeile 11, 1010 Wien Postfach 159
☎	**TEL**: 51 5 62, **FAX**: 512 57 78

PCNEWS-Eigenverlag & Redaktion

PCNEWSedu	PCNEWSedu-Eigenverlag
✉	Siccardsburggasse 4/1/22, 1100 Wien (Franz Fiala)
☎	**TEL**: 604 50 70 **FAX**: 604 50 70-2 **ANR**: 604 50 70
🖷	0664-101 50 70
🕒	Mi ab 20:00
E✉	InterNet pcnews@atnet.at CIS: 100024,1325 FidoNet 2:310/1.36 SchulNetz 22:102/1.136
🖷	**His Master's Voice (siehe ccc)**
Online	Fidonet *Echo:* PCNEWS.AUS*Files:* PCN Internet http://www.tgm.ac.at/tgm/pcnews/pcnews.html listserv@ccc.or.at SUBSCRIBE PCNINFO
KONTO DVR	PSK, Blz. 60000, Kto. 7.486.555, Franz Fiala - Eigenverlag 0735485 UID: ATU11721207
PCNEWS edit	S 50,-/1 Heft; S 200,-/5 Hefte; S 350,-/10 Hefte; S 450,-/15 Hefte

Bezugsbedingungen

Mitglieder des PCC-S, PCC-TGM, CCC(platin+silber), CLUB-AT:	S	0,-
Mitglieder des MCCA Jahresabo (5 Hefte)	S	100,-
Einzelbezugspreis	S	50,-
1-Jahresabo inkl. Versand (5 Hefte)	S	200,-
2-Jahresabo inkl. Versand (10 Hefte)	S	350,-
3-Jahresabo inkl. Versand (15 Hefte)	S	450,-
Auslandsabo + Versandanteil/Hefte	S	14,-
Diskettenabo f. 10 Disketten via CCC	S	350,-
Bei Abo-Abbruch werden für jedes nicht bezogene Heft vergütet	S	30,-

PCNEWS edit-on-line

FIDO	Echo PCNEWS . AUS		
	Wien	2:310/1	815 48 71 810 13 54 (ISDN)
		2:310/77	665 17 60
		2:310/110	401 04-472
		2:310/111	504 14 44-22
		2:310/1000	810 13 54 (ISDN)
	NÖ. Mödling	2:313/3	(022 36) 477 43
		2:313/5	(022 36) 274 18
		2:313/9	(022 36) 238 17
	Kirchstetten	2:313/24	(027 43) 88 57
	Sbg. Salzburg	2:315/3	(062 45) 81251
		2:315/30	(062 45) 855201 (ISDN)
	Golling	2:315/5	(045 66) 244 990
		2:315/50	(045 66) 244 990 (ISDN)
	Stmk. Graz	2:316/3	(0316) 481 080
	Hartberg	2:316/50	(033 32) 61 666
		2:316/51	(033 32) 66 907 (ISDN)
	Vlbg Gais	2:318/1	(055 25) 312 683

PCNEWS edit-49

Kennzeichnung	ISSN 1022-1611, EAN 9771022161000-09
Kopien	Für den **Unterricht** oder andere **nicht-kommerzielle Nutzung** für Beiträge, die mit ☐ oder ■ gekennzeichnet sind, gestattet und erwünscht. Für erwerbsmäßige Weiterverwendung ist der Autor zu befragen. Beiträge, die mit ○ oder ● gekennzeichnet sind. auf Anfrage.
Layout	WinWord 6.0
Belichtung	S/W: Folex-Folien, Farbseiten: PCG
Herstellung	Rollenoffset, 65g Recycle-Papier, Deckel: MAGNO
Erscheint	Wien, September 1996
Disk/Info	
Beitrags-kennzeichnung	Name, [Zusatzinformation/Programm] Nicht gekennzeichnete Beiträge von der Redaktion.
Schlußzeichen	Unkostenfrei / Bezahlt ☐ PCNEWS edit-Autoren / ■ vom Autor/Firma ○ andere Quellen/Firmen / ● vom Verlag

Werbung	
A4-Seite	**S/W**: S 2500,-, **2-Farb**: S 3125,- **4-Farb**: S 4000,-
Beilage	**bis 40g** S 1,-/Stück, mit Verpackung S 1,50/Stück
	Preise zuzügl. 10% Werbeabgabe und 20% MWSt.

Auflage 6000

Abo	450	Abonnenten
CCC	350	Abonnenten
CLUB-AT	30	Exemplare
Freiverkauf	1000	Exemplare
Hyperbox	10	Exemplare
MCCA	60	Abonnenten
PCC-S	50	Abonnenten
PCC-TGM	1460	Abonnenten
Belegexemplare	200	Exemplare
Elektronik-Firmen	1500	Exemplare

Inserenten

Firma	Seite	Zahl
at-net, Franz Penz	49	**1**
B&K-Verlag	U3	**1**
CM, Christian Metzger	3,124-125	**3**
CompDelphin	9,11	**2**
Digital Communications	119-123	**5**
EÖSPC	39	**1**
Etlinger	67	**1**
Excon	5	**1**
Fran-Com	13	**1**
Fric	16	**¼**
Friedl	65	**1**
Hummer	63	**1**
ISDN-Technik	18-19	**2**
Magnet	45	**1**
Microsoft	U4	**1**
Nowatron	101,109	**1**
PABLITOS	33-34	**1**
PC-Gourmet	15	**1**
PESACO	126-127	**2**
Radiobastler	61	**1**
Semrad Elektronik	75	**1/8**
Siemens	107,108	**2**
SoftwareDschungel	Beilage	
Xpoint	U2	**1**

Statistik

Quelle	Seiten
Deckel (U1..U4)	**4**
Antwortkarten	**4**
Textteil	**128**
	136
Autoren	**74**
Clubs	**7**
Eigen	**6**
Inserate	**33**
LIESMICH/Imp	**11**
Red/Tit/Ant	**5**
Gesamt	**136**

PCNEWS edit-49 Embedded Systems

PCC-S	Personal Computer Club-Salzburg
✉	Itzlinger Hauptstraße 30, 5022 Salzburg (Otto R.Mastny)
☎ 0662	**TEL**: 45 36 10-0 **FAX**: 45 36 10-52
🕒	Mo-Fr: 8.00 - 12.00 (über Direktion der HTBLA-Salzburg)
E✉	InterNet haiml@cosy.sbg.ac.at
KONTO DVR	Salzburger Sparkasse, Blz. 20404, Kto. 02300 330 720, PCC-S 0559610
Jahresbeitrag	Vollmitglieder 340.-, Schüler 150.-, Förderer 1000.-
PCNEWS edit	: für Vollmitglieder und Förderer inkludiert
Vorstand	*Obmann/Stv.* Otto R. MASTNY, BREITFUß *Kassier/Stv.* R. LACKNER, PICHLER *Schriftführer/Stv.* ECK, Gottfried HAIML

PCC-TGM	Personal Computer Club-Technologisches Gewerbemuseum
✉	Postfach 59, 1202 Wien (Robert Syrovatka)
☎	**TEL**: 332 23 98 **FAX**: 332 23 98-2 **ANR**: 332 23 98
Clublokal	Wexstraße 19-23, 1200, Wien
🕒	Mi: 19.00-20.30 (Frau Jelinek)
E✉	InterNet syro@email.tgm.ac.at FidoNet 2:310/1.36 SchulNetz 22:102/1.136
🖷	**His Master's Voice (siehe ccc)**
KONTO DVR	EÖSPC, Blz. 20111, Kto. 053-32338, PCC-TGM 0596299
Jahresbeitrag	Schüler: 200,- Student: 300,- sonst: 400,-
PCNEWS edit	: inkludiert, 🖷inkludiert
Vorstand	*Obmann/Stv.* Rudolf KÖNIG, Robert SyROVATKA *Kassier/Stv.* Leo ZEHETNER, Norbert PAY *Schriftführer/Stv.* Robert THUMFARTH, Martin WEISSENBÖCK *Rechnungsprüfer* Robert BERTHOLD, MÖDLER *Beiräte* Paul OSTERMAIER, Wolfgang ZELINKA, Martin REITER, Wolfgang NITSCHE

Druckfehler und Irrtümer vorbehalten.

Preisangaben in Inseraten sind wegen des Fertigungszeitraums der PCNEWSedu von 1 Monat nicht am letzten Stand. Wir bitte die Leser, die aktuellen Preise nachzufragen.

Alle erwähnten Produktnamen sind eingetragene Warenzeichen der entsprechenden Erzeuger.

Alle Fernsprechnummern ab Ortsnetz Wien.
Vorwahl Wien aus Österreich: 0222
Vorwahl Wien international: +431