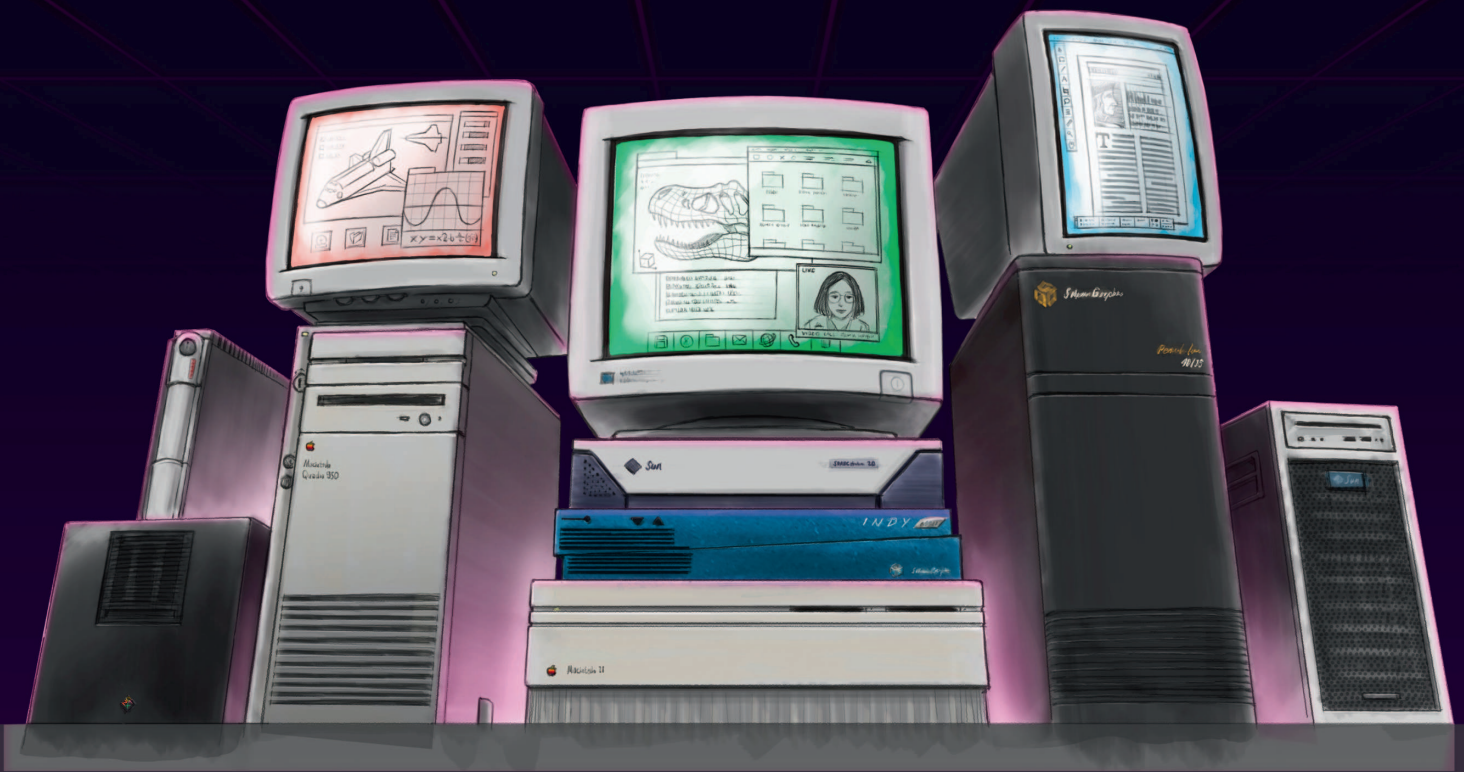


WORKSTATIONS



Workstations

**Sun, SGI, HP – Tipps
und Übersichten**

Atari

**Atari ATW 8000
Cross Entwicklung**

Texas Instruments

TI Taschenrechner

Apple

**Messen mit Apple II
Mac PC-Emulatoren**

Commodore

**UNIX auf AMIGA
Micro-PET
CBM Max Machine
Universal Cartridge**

Netze

**Sinclair QNET
DIY SuperNOS**

Projekte

**Einstieg in dBASE
Fluxcopy
PI1541 Floppy**

CLASSIC



COMPUTING

10. - 11. September 2022

Samstag 09.00 - 18.00 Uhr

Sonntag 09.00 - 17.00 Uhr

Halle IV

Kaiserstraße 10A

49809 Lingen



Technikhistorische Ausstellung



**Retrocomputer der
70er, 80er und 90er Jahre**



**Viele Geräte zum Ausprobieren
und „zurückerinnern“**



Reparaturecke

EINTRITT FREI



Die Veranstaltung wird unter den gültigen Corona-Regelungen durchgeführt.

Veranstaltet vom „Verein zum Erhalt klassischer Computer e.V.“

Aussteller melden sich bitte an unter <https://www.classic-computing.org/cc2022-anmeldung/>

readme.txt Inhalt

UNIX-Workstations sind als Sammlerobjekte in der Retrocomputer-Szene nicht so beliebt wie die klassischen Homecomputer von Atari oder Commodore. Trotzdem gibt es nicht wenige Freunde alter Rechner, die sich dieser Geräteklasse angenommen haben. Kein Wunder – jeder Hersteller hat eigene technische Innovationen in den Maschinen verbaut. Mehr noch: Die heute dominante Technik hat ihre Wurzeln in den RISC-Workstations der 1990er-Jahre. Viele der damals tonangebenden Ingenieure und Manager sind noch heute aktiv und mischen bei aktueller Technik mit. Wir stellen in dieser Ausgabe die wichtigsten Hersteller und Systeme vor, geben Tipps zur Instandsetzung und Erweiterung und schauen uns auch an, was Atari und AMIGA in Bezug auf UNIX zu bieten haben.

Das ist natürlich nicht alles. Wir stellen selten beschriebene Taschenrechner von Texas Instruments vor, beschreiben aktuelle Nachbauten alter Rechner und werfen einen Blick auf den ersten Commodore-Rechner mit dem SID-Chip. Auch die Praxis kommt nicht zu kurz, denn wir zeigen, wie Atari-Programme auf einem PC kompiliert und assembliert werden können, nutzen den Apple II zur Temperaturmessung und konfigurieren einen universellen Netzwerkserver. Besonders stolz sind wir, dass uns ein englischsprachiger Artikel zum wenig bekannten Sinclair QLAN erreicht hat, den wir in einer deutschen Übersetzung bringen.

Zu dieser Ausgabe gibt es auch wieder eine Heft-CD als Image zum Download, die viele Programme und zusätzliche Infos enthält.

Haben wir etwas vergessen? Ja! Mit dieser Ausgabe feiert die LOAD ihr 10-jähriges Bestehen, auch wenn es erst die 8. Ausgabe ist. Ein wenig Geschichte der LOAD finden Sie auf den Seite 79 und 83, ausführlich feiern werden wir uns aber erst in der 10. Ausgabe im Jahr 2024.

Viel Spaß also mit dieser Ausgabe der LOAD wünscht Ihnen

Ihr Georg Basse

readme.txt.....	3	Allgemein
Veranstaltungskalender.....	4	
Die CDROM zum Heft.....	5	
Kurz berichtet.....	6	
Sir Clive Sinclair verstorben.....	45	
Geschichte der Workstations.....	8	Workstations
Workstations: Hersteller und Modelle.....	12	
Leere Uhrenchips umschiffen.....	16	
Netzkonfiguration bei Sun Solaris.....	17	
lomega Laufwerke nutzen.....	18	
Editor vi benutzen.....	19	
UNIX für AMIGA.....	20	
DIE HP PA RISC Story.....	25	
Sun Ultra-1 neu installieren.....	28	
Linux auf Sun Hardware.....	31	
Atari Transputer Workstation.....	32	
Texas Instruments Voyage 200.....	35	Hardware
Texas Instruments TI-84plus.....	36	
Sinclair QNET entschlüsselt.....	38	
Der Micro-PET.....	46	
Die Commodore Max Machine.....	50	
Commodore Max Machine wiederbeleben....	52	
PC-Emulatoren für Apple Macintosh.....	54	
Atari ST Cross-Entwicklungsumgebung.....	64	Software
Atari ST Cross Assembler.....	66	
Einstieg in dBASE.....	67	
SuperNOS im Eigenbau.....	58	Projekt
Temperaturfühler am Apple][.....	63	
Universal Cartridge für Commodore 64.....	70	
Commodore 1541 Laufwerk emulieren.....	74	
Das Fluxcopy-Projekt.....	76	
Neues aus dem Verein.....	7	Verein
Foren-Highlights.....	34	
Classic Computing 2021.....	80	
10 Jahre LOAD Magazin.....	79	LOAD
Vorschau.....	82	
Impressum.....	82	

Alle Links der Artikel in dieser Ausgabe finden Sie unter <https://www.load-magazin.de/load8>

Veranstaltungen

Gern hätten wir an dieser Stelle einen randvollen Kalender mit Terminen der Retrocomputer-Treffen 2022 veröffentlicht. Doch leider sind aufgrund der gesetzlichen Auflagen zur Pandemiebekämpfung Veranstaltungen noch immer nicht verbindlich zu planen. Vielleicht hat sich die Lage mit dem Erscheinen dieser Ausgabe bereits verbessert. Darum Augen auf – viele Veranstalter warten gespannt darauf, dass es wieder losgehen kann. Hier ein paar Beispiele:

Das **Alternative Computermeeting in Flechtorf** zwischen Wolfsburg und Braunschweig heißt im Mai oder Juni alle historischen 8-Bit Rechner, ältere Konsolen, Amigas, Ataris, MorphOS-Rechner und PPC-Macs sowie X86er mit UAE und AROS willkommen. Wer einen Vortrag halten will, ist herzlich dazu eingeladen und kann sich deswegen beim Veranstalter melden.

Auf dem **Amiga-Meeting Neumünster** treffen sich Besitzer von Classic-Amiga, Amiga OS4, MorphOS, Emulatoren oder AROS. Die Veranstaltung ist auch offen für andere Classic-Systeme. Das nächste Treffen ist vom 03. bis 05.11.2022 geplant.

Die **Classic Computing 2022**, das große Treffen des Vereins zum Erhalt klassischer Computer und eines der größten Treffen in Deutschland soll vom 08. bis 11. September in Lingen an der Ems stattfinden.

Die **DoReCo** findet mehrmals im Jahr in Dortmund statt und ist ein Treffen für Freunde alter und neuer Video- und Computerspiele. Für 2022 waren zum Redaktionsschluss noch keine Termine bekannt. Zusammen mit dem Heinz Nixdorf Forum in Paderborn ist außerdem ein zweitägiges Retrocomputer-Festival geplant, der Termin steht noch nicht fest.

Auf der **Fujijama** treffen sich die Liebhaber der 8-Bit-Atarirechner. Die Veranstaltung wird vom User Club ABBUC veranstaltet.

Der Vintage Computer Club e.V. veranstaltet mit der **[connected]** und der **Interface Kiel** zwei Treffen im Norden der Republik. Dort wird klassische Computertechnik genutzt, gepflegt und präsentiert und Erfahrungen werden ausgetauscht.

Die **Lange Nacht der Computerspiele** findet in der Universität Leipzig statt und präsentiert auf 4 Etagen neue und alte Computerspiele.

Die **Luhecon** findet mehrmals jährlich in Winsen (Luhe) statt und ist ein offenes Treffen für alle Freunde klassischer Rechnertechnik.

Der **Marburger Stammtisch** ist ein unkommerzielles und familiäres Treffen retrobegeisterter Technikfreaks aller Couleur. Der Spaß an der Arbeit mit alter Computerhard- und software steht im Mittelpunkt.

Die **Retropulsiv** ist eine halbjährlich an der Hochschule Augsburg stattfindende Veranstaltung zum Thema antiquierte Computer aus den Anfängen der PCs in den 1970ern bis hin zu Spielekonsolen aus den 1990ern.

Der **Retrocomputer-Treff Niedersachsen** veranstaltet drei- bis viermal im Jahr ein ganztägiges Treffen in der Region Hannover. Das Treffen ist nicht auf bestimmte Hersteller festgelegt.

Das **Vintage Computer Festival Berlin** findet Anfang Oktober im Technikmuseum Berlin statt und präsentiert in einer großen Ausstellung historische Computer. Die Veranstaltung bietet dazu eine Reihe interessanter Vorträge.

Es lohnt sich also, hin und wieder auf die Webseiten der Veranstalter zu schauen. Termine werden dort kurzfristig bekanntgegeben.

Adressen und Links

Alternatives Computer Meeting

Dorfgemeinschaftshaus Flechtorf, Alte Braunschweiger Str. 21, 38165 Lehre OT Flechtorf

<http://acm.members.vzekc.de>

Amiga-Meeting Neumünster

Kiek in!, Gartenstraße 32, 24534 Neumünster

<http://www.amigameeting.de/>

Classic Computing 2022

Halle IV, Kaiserstraße 10a, 49809 Lingen

<https://www.classic-computing.org/cc2022-hauptseite/>

DoReCo

AWO Dortmund, Syburger Strasse 75, 44265 Dortmund

<http://www.doreco.de/>

DoReCo Party

Schützenhalle Anröchte, Altenmellrich, Alter Kirchweg 2, 59609 Anröchte

<http://www.doreco.de/>

HomeCon

Alte Schule – Eingang Haggasse – Großer Saal, EG Taubengasse 3, 63457 Hanau (Großauheim)

Fujijama

Schützenhausweg 11, 08485 Lengenfeld

<http://abbuc.de/~atarixle/fuji/2020/>

Interface Kiel

Jugendhaus Klausdorf (1. + 2. OG), Dorfstraße 101, 24222 Schwentinental/Klausdorf

<http://vccev.de/events>

Lange Nacht der Computerspiele

LIPSIUS-Bau, Karl-Liebkecht-Str. 145 in 04277 Leipzig

LuheCon

Marshall, Schloßplatz 11, 21423 Winsen an der Luhe

Marburger Stammtisch

Ortenberg-Gemeinde e.V., Rudolf-Bultmann-Straße 7

<http://www.marburger-stammtisch.de>

Retro Daddel Day Wuppertal

Blumenstr. 16, 42119 Wuppertal

<http://www.radio-paralax.de/>

RETROLUTION

Kulturhalle Steinheim, Ludwigstraße 67, 63456 Hanau (Steinheim)

RETROthek Karlsruhe

Aussteller-Voranmeldung bitte unter retrothek@kultur.karlsruhe.de

Stadtbibliothek im Neuen Ständehaus, Ständehausstraße 2, 76133 Karlsruhe

RETROPulsiv

Hochschule Augsburg, An der Hochschule 1, 86161 Augsburg

www.hs-augsburg.de/Service/Impressum*anmeldung@retropulsiv.de

Retrotreff Niedersachsen

Freizeitheim Döhren, An der Wollebahn 1, 30519 Hannover

<https://www.classic-computing.org/tag/hannover/>

Uni Mainz

Computersammlung der Universität

<https://www.classic-computing.org/veranstaltungskalender>

Vintage Computer Festival Berlin

Deutsches Technikmuseum (Historische Ladestraße), Möckernstr. 26, 10963 Berlin

www.vcfb.de

VCFe

Kulturzentrum Trudering, Wasserburger Landstraße 32, 81825 München

<https://www.vcfe.org>



Die CD-ROM zum Heft

Wir berichten in der LOAD oft über Software und Tools für klassische Computer. Dazu liefern wir Links zu den Download-Quellen. Doch das Internet ist vergänglich – manchmal sind wichtige Dateien nach einigen Jahren nicht mehr auffindbar.

Darum gibt es zu dieser Ausgabe der LOAD wieder eine CD-ROM, randvoll mit Zusatzinfos zu den Artikeln, Bildern, Software und Vielem anderen. Außerdem finden sich auf der CD-ROM die PDF Versionen der bisher erschienenen Ausgaben der LOAD. PC-Nutzer werden sich außerdem über das BIOS Kompendium und seine Tools freuen, ein unentbehrliches Nachschlagewerk für die Feinheiten der unterschiedlichen BIOS-Versionen.

Die CD-ROM zum Heft gibt es als ISO-Image zum Selbstbrennen und in limitierter Auflage auf unseren Veranstaltungen.

Das Passwort zum Download lautet: CD2022

Download: <https://www.load-magazin.de/load8>

Inhalt

1-Artikel
AMIX
CrossCompiling
MicroPET
PiZero1541
SinclairQL_Netzwerk
SuperNOS
TI-84plus
UniversalCartridge
2-LOAD_PDF
5-VzEkC_Materialien

GENERVT VON IHRER ALTEN?

Wir kümmern uns.
Rund- oder Teilerneuerung,
Modernisierung, Straffung,
Schönheitsreparaturen oder,
falls nötig, diskrete Entsorgung.

WIR reden von Ihrer Website.
Woran dachten Sie denn?

Wir sind eine Werkwerkstatt
und machen Websites.
Und Onlineshops.
Und Visitenkarten. Und Logos.
Und Broschüren. Und Flyer.
Und Plakate. Und, und, und...

SIE HABEN EINE IDEE?

Wir setzen sie um.

SIE HABEN KEINE IDEE?

Wir schon.



Werbeagentur für Frauen. Und Männer mit Mut.

Internet (zum Schnuppern):
pritti-wummen.de

Mail (für Entschlossene):
kontakt@pritti-wummen.de

Telefon (für Wildentschlossene):
0177 - 538 36 86 oder
0171 - 500 42 62

PC-Serial-Loader

Ursprünglich auf Disketten verteilte Software für den IBM PC sind heute als Diskimages im Internet in Hülle und Fülle zu bekommen. Um sie zu nutzen, müssen sie oft wieder auf reale Disketten geschrieben werden. Das kann Probleme mit sich bringen, wenn der Zielrechner zwar passende Floppies, aber keinen Netzanschluss hat, der Internet-PC aber keine passenden Diskettenlaufwerke. Der PC-Serial Loader löst dieses Problem. Es handelt sich um eine zweiteilige Software für IBM PCs, die Diskimages vom Internet-PC über ein serielles Nullmodem-Kabel auf den Zielrechner überträgt und dort auf Disketten schreibt. Zuerst ist dabei aber ein Henne-Ei Problem zu lösen: Wie kommt das Clientprogramm auf den Zielrechner, wenn dieser nur Diskettenlaufwerke besitzt? Der PC-Serial Loader macht das ähnlich der ADTPro Software für den Apple II oder dem Amiga Explorer. Ein in Assembler geschriebener Initial Loader wird über die Tastatur am Zielrechner eingegeben – kein Problem, denn er umfasst nur 22 Instruktionen. Die Eingabe passiert mittels Nummernblock und gedrückter ALT-Taste direkt als Bytefolge. Ist der Initial Loader erst einmal eingetippt, lädt er den eigentlichen Client (die Toolbox) auf den Zielrechner. Nun können Diskimages übertragen werden. Der Internetrechner erhält dafür ein Windows-Programm mit grafischer Benutzeroberfläche, das die Win32-API verwendet. Das nur etwa 60 KB große GUI-Programm läuft unverändert unter allen Windows-Versionen seit Windows 95. Unter Linux ist WINE als Zwischenschicht erforderlich. Das Programm ermöglicht ein Lesen und Schreiben von Diskimages mit den Floppy-Laufwerken des Zielrechners. Auch Dateien lassen sich in beide Richtungen transferieren.

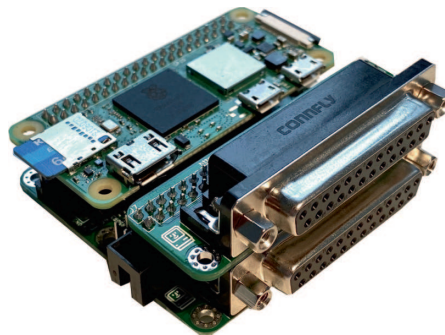
Links:

<http://www.zotteljedi.de/pc-serial-loader/index.html>

RaSCSI

RaSCSI ist ein virtueller SCSI-Emulator, der auf einem Raspberry Pi läuft. Er besteht aus einer Software unter dem Debian-Derivat Raspian und einer Hardwareerweiterung, die mit den GPIO-Pins des Pi verbunden wird. RaSCSI kann mehrere SCSI-Geräte gleichzeitig emulieren, bietet eine Steuerungsschnittstelle zum Anschließen und Entfernen von Laufwerken sowie zum Einlegen und Auswerfen von Wechselmedien. Mittels Software lassen sich virtuelle Laufwerke auf einer SD-Karte anlegen, SCSI-IDs und LUNs zuweisen und diese wie echte SCSI-Festplatten durch

einen Macintosh oder einen Atari TT, eine Sun oder DEC Alpha Workstation oder andere Computer mit SCSI-Anschluss nutzen. Besonderes Schmankerl: Die Software emuliert auch einen DaynaPort-SCSI-Ethernetadapter und bringt so klassische Macintosh- und Atari-Rechner ans Netz. Auch ein spezieller Webproxy zur Filterung von Webinhalten ist implementiert – dadurch verdauen auch alte Webbrowser die heute meist völlig überladenen Webseiten. Ein ausführlicher Erfahrungsbericht zum RaSCSI-Adapter erscheint in der nächsten Ausgabe der LOAD.



Links:

<https://github.com/akuker/RASCSI/wiki>

Geburtstagskinder

Apple Quicktime



About QuickTime...

Im Dezember 2021 war es drei Jahrzehnte her, dass Apple mit Quicktime den Pionier unter den Programmen und Formaten für Computervideos auf den Markt brachte. Voller Begeisterung saßen die Anwender vor ihren Macintosh-Rechnern und konnten echte Videos auf dem Monitor bewundern. Nur briefmarkengroß und meilenweit von heutigen 4K-Auflösungen entfernt, drangen Videoclips auf Rechner vor, die nie zuvor Bewegtbilder gesehen hatten. In den 1990er-Jahren und den ersten Jahren des neuen Jahrtausends wurde Quicktime praktisch überall eingesetzt. Apple setzte den Quicktime Player für Windows um, auch für andere Plattformen existieren entsprechende Player, die Quicktime-Codecs unterstützen.



Commodore 64

Im Januar 2022 ist der meist verkaufte Heimcomputer weltweit, der Commodore 64 tatsächlich 40 Jahre alt geworden. Seit seiner Vorstellung auf der Winter Consumer Electronics Show 1982 hat er sich millionenfach verkauft, die Schätzungen schwanken zwischen 12,5 und 30 Millionen Exemplaren. Nicht nur seine für die damalige Zeit üppige Speicherausstattung mit 84 KBytes RAM und seine Videodarstellung mit dem VID und Soundfähigkeiten mit dem dreistimmig polyphonen Soundchip MOS Technology SID 6581 machten ihn so populär. Vor allem die Erweiterbarkeit durch den herausgeführten Adress- und Datenbus machen ihn bis heute zur Plattform für Hardware-Ergänzungen aller Art. Eine ausführliche Würdigung des Brotkastens erscheint in der nächsten LOAD.

Sinclair ZX81



Etwas verspätete Glückwünsche gehen auch an den Sinclair ZX81. Er erblickte im März 1981 das Licht der Welt oder besser den Nebel der britischen Insel, denn zunächst war er nur in England über den Versandhandel verfügbar. Als Weiterentwicklung des Sinclair ZX80 wurde er als günstiger Lerncomputer beworben. Ende 1981 verkaufte Sinclair dann den ZX81 auch im übrigen Europa und in anderen Ländern der Welt. Bis Ende 1984 setzte Sinclair insgesamt etwa 2 Millionen der kleinen Geräte ab, darunter auch lizenzierte Nachbauten wie dem Timex Sinclair 1000, ein besonders in den USA beliebtes Gerät.

Commodore AMIGA 1200 und 4000

Im Oktober 2022 können alle AMIGA Fans auf den 30. Geburtstag der beiden letzten AMIGA-Modelle anstoßen, den A1200 und den A4000. Mit diesen beiden Modellen führte Commodore den leistungsfähigen AGA-Chipsatz ein und löste den glücklosen AMIGA 600 ab. Mit den drei Chips Lisa, Alice (Grafik) und Paula (Sound) konnte sich die Leistung des Motorola 68EC020 (14 Mhz) basierten Rechners durchaus sehen lassen. Die Produktion des A1200 durch Commodore dauerte nur etwa 2 Jahre und wurde nach der Insolvenz des Unternehmens zunächst eingestellt. Ab Mai 1995 nahm die ESCOM-Tochter Amiga Technologies dann die Produktion wieder auf. Doch nach einem Jahr war auch damit Schluss, als ESCOM und Amiga Technologies ebenfalls Konkurs anmelden mussten.



Atari-Home.DE übernommen

Knapp 23 Jahre ist es her, dass atari-home.de das Licht der Welt erblickte. Der bisherige Eigentümer und Betreiber der Seiten (Johannes) hat zum Februar 2022 nun Webpräsenz und Forum an den Verein zum Erhalt klassischer Computer e.V. übergeben. Johannes hat fast die Hälfte seines Lebens mit atari-home.de verbracht und hat dafür gesorgt, dass diese Domain auch weiterhin ein wichtiger Anlaufpunkt für alle Atari-Freunde in deutschsprachigen Raum bleibt. Der VzEkC e.V. pflegt die bestehenden Angebote in gewohnter Form unverändert weiter.

Links:

<http://www.atari-home.de>

Verein zum Erhalt klassischer Computer e.V.



Sehr geehrte Vereinsmitglieder,
sehr geehrte Freunde der Bits & Bytes vergangener Tage

Mit dem Jahr 2021 blickt der Verein zum zweiten Mal auf ein Jahr zurück, das anders verlief als geplant. Das Corona-Virus hat vieles erschwert und manches unmöglich gemacht und unseren Alltag länger dominiert, als wir es im Jahr zuvor für möglich gehalten haben.

Ein wichtiger Teil unseres Vereinslebens sind Veranstaltungen zum Thema Retrocomputing, sei es als Veranstalter regionaler und überregionaler Treffen oder als Teilnehmer an diesen Meetings. Unser jährliches Highlight, die Classic Computing stand 2021 lange auf der Kippe. Die Pandemielage war schwer abzuschätzen und auch die Hygienevorgaben der Landesregierungen veränderten sich in kurzen Abständen. Umso glücklicher waren wir, dass die Classic Computing 2021 vom 17. bis 19. September im Kulturzentrum Vöhringen stattfinden konnte wie geplant. Dort zeigten 38 Aussteller an insgesamt 65 Tischen ihre Schätze. Die Ausstellung war am Samstag und Sonntag für die Öffentlichkeit zugänglich, etwa 250 Besucher fanden ihren Weg zu uns. Erstmals hat der Verein eine Security-Firma beauftragt, die sich um die Einhaltung der "3G-Regeln" zum Schutz vor der Corona-Pandemie gekümmert hat. Der besondere Dank gilt Michael Vogt, der die Veranstaltung vor Ort organisiert hat und für einen reibungslosen Ablauf sorgte.

Aber nicht nur die Classic Computing war ein Erfolg, auch regionale Treffen konnten stattfinden. Das Münchberger Bastel- und Retrotreffen hat das Vereinsleben ebenso bereichert wie die Retrocomputing-Abende im Stuttgarter Shack oder der Retrocomputer-Treff Niedersachsen in der Nähe von Hannover. Weitere Kontaktmöglichkeiten bestanden bei Online-Stammtischen wie dem von Axel und Reinhard. Für das laufende Jahr 2022 sind wir optimistisch, dass unsere Treffen wieder weitgehend unter gewohnten Bedingungen stattfinden können. Die Planung der Classic Computing in Lingen ist weit fortgeschritten und auch regionale Treffen finden wieder statt.

Anders wird es mit unserer jährlichen Hauptversammlung sein. Nachdem diese in den Jahren 2020 und 2021 als Online-Veranstaltung abgehalten wurde und wir sehr gute Erfahrungen mit dieser Form machen konnten, werden wir dies auch in den nächsten Jahren beibehalten. Die Erfahrung hat gezeigt, dass auch Wahlen auf diesem Wege erfolgreich durchgeführt werden können. Auf der JHV 2021 wurde der 1. Vorsitzende Stephan Kraus in seinem Amt bestätigt. Der langjährige Kassenwart Thomas Linke scheidet aus beruflichen Gründen aus, Nachfolger wurde der bisherige 1. stellvertretende Vorsitzende Florian Stassen. Norbert Kötting rückt für ihn in den Vorstand nach. Das Amt des Schriftführers übernimmt Georg Basse von Matthias Krambeck und als neuer 1. Beisitzer des Schiedsgerichts wurde Bernd Dohr gewählt. Volker Mohr wurde als Vorsitzender des Schiedsgerichts bestätigt, ebenso Christa Schuldes als 2. Beisitzerin.

Der Verein und seine Aktivitäten erfreuen sich auch an anderen Stellen großer Beliebtheit. Im Jahr 2021 konnten wir 46 neue Vollmitglieder und 4 neue Fördermitglieder gewinnen. Damit zählt der Verein zum Februar 2022 bereits 315 Mitglieder. Auch das Vereinsforum wächst stetig – derzeit sind dort 4.250 Teilnehmer registriert. Seit Bestehen des Forums haben sich 350.000 Beiträge in 26.000 Themen rund um Retrocomputing und Vereinsleben dort gesammelt. Wir schätzen uns außerdem glücklich, mit der Übernahme des Forums Atari-Home.DE eine weitere Community mit 780 Mitgliedern unterstützen zu können.

Wir können also optimistisch in die Zukunft schauen. In diesem Sinne – viel Spaß bei der Lektüre der LOAD!

Der Vorstand des VzEkC
S. Kraus, N. Kötting, C. Dirks, F. Stassen, G. Basse

Eine kleine Geschichte der RISC Workstations

Weniger ist mehr

Oft wird gesagt, die University of California in Berkeley gegenüber von San Francisco sei vor allem durch zwei Dinge bekannt geworden: LSD und BSD. Die Behauptung ignoriert zumindest eine Entwicklung aus dieser Stadt, nämlich das Konzept der RISC-Prozessoren. Ohne diese wäre der Boom der Workstations in den letzten beiden Jahrzehnten des vorigen Jahrhunderts wohl ausgeblieben.

Versetzen wir uns einmal in die Computerwelt der 1960er-Jahre. Einige, wenige Großrechner (Mainframes) liefen in Banken, Versicherungen, Industrie oder Forschungseinrichtungen. Der direkte Kontakt zu diesen großen Eisen war dem Operator vorbehalten, der gekleidet im weißen Laborkittel die Maschine am Laufen hielt. Programmierer und Anwender hatten keinen direkten Kontakt – sie gaben ihre Arbeitsaufträge in Form von Lochstreifen oder Lochkarten ab und erhielten die Ergebnisse in Ausdrucken zurück. Die Programmierung erfolgte oft in Maschinensprache, wenngleich auch höhere Programmiersprachen wie Fortran durchaus verbreitet waren. Eine direkte Programmierung in Maschinensprache erlaubt dem Computer eine effiziente Abarbeitung eines Algorithmus, verlangt aber den Programmierer einiges ab. Die Prozessoren der Großrechner versuchten den Programmieren das Leben zu erleichtern, indem sie komplexe Befehle für den jeweiligen Anwendungszweck wie Buchhaltung oder wissenschaftliche Berechnungen boten. IBM beschritt mit seinem legendären System/360 schließlich einen neuen Weg und implementierte diese Befehle mittels Microcode. Das sind in einem CPU-eigenen Speicher abgelegte Routinen, die hart verdrahtete Befehle der CPU verwenden und sozusagen die Firmware der CPU bilden. Dieser Komfort für den Assemblerprogrammierer wird mit einem Nachteil an anderer Stelle erkaufte: Die Ausführung der komplexen Befehle ist zeitintensiv und nicht in einem CPU-Takt zu bewältigen.

Die Anfänge

Schon in den 1960er Jahren zeigte Seymour Cray, dass es auch anders geht. Cray war damals bei der Firma Control Data Corporation (CDC) angestellt. Mit den Erfahrungen, die er bei der Entwicklung von Supercomputern bei seinem vorherigen Arbeitgeber Engineering Research Associates (ERA) gesammelt hatte, stellte CDC im Jahr 1965 die CDC 6600 vor. Trotz durchschnittlicher Hardware besaß die Maschine eine herausragende Performance. Es handelte sich um eine 6-Bit Maschine mit Ringkernspeichern und 18 Bit Adressraum. Das Besondere: Der Rechner kannte nur 64 Maschinenbefehle – genau so viele, wie in ein 6 Bit Word passen. Komplexe Kommandos wie beim System /360 gab es nicht. Zwar mussten die Programmierer nun selbst implementieren, was der Rechner nicht konnte, dafür liefen die Programme deutlich schneller ab.

Der erste RISC Prozessor

Auch an anderer Stelle existierten Systeme und Prozessoren mit eingeschränktem Befehlssatz, beispielsweise die PDP-5 von DEC mit nur 3 Bit breiten Befehlssatz. Allerdings war hier mehr die Systemarchitektur an sich und weniger der Geschwindigkeits-

aspekt der Grund. Erst 1974 entwickelte John Cocke am Thomas J. Watson Research Center das RISC-Konzept für den Minicomputer IBM 801. In den 1970er-Jahren waren Hochsprachen schwer in Mode gekommen. Compiler übersetzten den für Menschen leichter verständlichen Quellcode in Maschinencode. Cocke machte die Beobachtung, dass Compiler meist nur wenige, elementare Befehle des Prozessors nutzten. Folgerichtig ersann er ein Designkonzept, bei dem der Prozessor nur diejenigen Maschinenbefehle kannte, die Compiler vorrangig verwenden. Nicht mehr der CPU, sondern dem Compiler kam die Aufgabe zu, komplexe Befehle in kompakten, vom Prozessor abzuarbeitenden Code zu übersetzen. John Cocke prägte auch den Begriff RISC für dieses Konzept und grenzte es vom "Complex Instruction Set" oder kurz CISC des bisherigen Designs ab.

Das Berkeley RISC Projekt

Zwischen 1980 und 1984 führten Wissenschaftler in Berkeley im Auftrag der Defense Advanced Research Projects Agency

COMPUTERSYSTEMSLABORATORY

DEPARTMENTS OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
STANFORD UNIVERSITY · STANFORD, CA 94305-2192



The SUN Workstation Architecture

Andreas Bechtolsheim
Forest Baskett
Vaughan Pratt

Technical Report No. 229

March 1982

Die Vernetzung der Bibliotheksrechner in Stanford schuf das Stanford University Network SUN

Bild: Peg Skorpinski (CC-BY-SA 3.0)



David A. Patterson gilt als geistiger Vater des heutigen RISC-Designs

(DARPA) das Forschungsprojekt Berkeley RISC durch. Unter der Leitung von David A. Patterson und Carlo H. Séquin sollte eine Architektur für VLSI-Chips (Very Large Scale Integration) entwickelt werden. So entstanden zwei Chipdesigns, RISC I und RISC II. 1981 erschien das Ergebnispapier mit dem Titel „Design and Implementation of RISC I“, das die Grundzüge der RISC Prozessoren beschreibt. Auf Basis dieses Dokuments begann sehr bald die Entwicklung von RISC-Prozessoren für den kommerziellen Gebrauch. Vor allem in Großbritannien, wo 1983 Sophie Wilson mit der „Acorn Risc Machine“ (ARM) einen Prozessor für die Acorn-eigenen Microcomputer vorstellte. Auch die SPARC-Architektur von SUN hat ihre Wurzeln in diesem Projekt, ebenso der DEC Alpha Prozessor. David A. Patterson hat die RISC Entwicklung in den folgenden Jahrzehnten weiter vorangetrieben und ist heute zusammen mit Krste Asanovic der Entwickler RISC-V-Architektur.

Was macht die RISC-Architektur nun so besonders? In klassischen RISC-CPU's erfolgt die Abarbeitung von Befehlen immer in vier Stufen: Ein Befehl wird aus dem Speicher geholt (Fetch), dekodiert (Decode), in die Ausführungseinheit des Prozessors geladen (Load) und dann ausgeführt (Execute). Bei dieser als Superskalarität bezeichneten Arbeitsweise befindet sich in jeder Stufe immer ein Befehl, die CPU arbeitet diese Stufen wie am Fließband ab. Neben der Superskalarität zeichnet eine RISC-CPU auch eine große Anzahl frei benutzbarer Register aus. So erleichtert die CPU eine effiziente Codeoptimierung. Mehr dazu findet sich in den Artikeln zu HP PA-RISC ab Seite 25 und zu SUN SPARC ab Seite 28.

Von der CPU zur Workstation

Das kompakte Design der RISC-Prozessoren und die damit verbundene geringe Stromaufnahme sowie die hohe Leistungsfähigkeit rückte einen langgehegten Traum in greifbare Nähe. Forscher und Entwickler wollten vor dem Eindruck der ersten Mikrocomputer von Apple oder Commodore nicht mehr auf Rechenzeit der Großrechner warten, sondern verlangten nach eigenen Supercomputern auf dem Schreibtisch. RISC versprach die Möglichkeit, dies kostengünstig zu realisieren. Die damals bereits mit CISC-Prozessoren (oft aus der Motorola 68000er Reihe) ausgestattete Klasse der Workstations schien die richtige Basis zu sein. Workstations sind Computer, die zur Benutzung durch einen Anwender gedacht sind. Ihre Domäne sind technische und wissenschaftliche Anwendungen. In lokalen Netzen miteinander verbunden und mit Mehrbenutzer-Betriebssystem ausgestattet, waren sie sehr attraktiv

Bild: Wikipedia, Jitae Couperus (CC-BY)



Die CDC 6600 von Seymour Cray aus dem Jahr 1965

Bild: <https://commons.wikimedia.org>, Public Domain



Eine typische RISC-CPU

für Nutzer in Universitäten und Forschungseinrichtungen, aber auch in der Verwaltung oder der Medizin. Als erste Modelle sind hier sicher der Xerox Alto zu nennen, aber auch Apollo Domain und frühe Sun Systeme

Workstations eroberten nicht nur neue Anwendungsgebiete, sie veränderten auch das soziale Gefüge im Computerbetrieb. Anders als in der Mainframe-Welt mit ihrer Trennung und Verteilung von Arbeitsabläufen auf verschiedene Rollen und Personen im Rechenzentrum hatte der Benutzer sein System fast unter alleiniger Kontrolle. Die Hierarchie beim Workstation-Betrieb konnte oft nur die Trennung in Systemadministratoren und Benutzer. Dies passte besser in die Stimmung des Umbruchs der 1970er- und 1980er-Jahre als der Mainframe-Betrieb. Workstations wurden von unterschiedlichen Herstellern produziert, nicht wenige davon angesiedelt im aufstrebenden Silicon Valley in Kalifornien. Systeme von Apollo Computer, DEC, Hewlett Packard, IBM, NeXT, Sun Microsystems oder Silicon Graphics teilten sich zwischen 1980 und der Mitte der 2000er-Jahre den Markt.

UNIX und GUI

Das Betriebssystem der RISC-Workstations ist fast immer ein UNIX-Derivat. Das kommt nicht von ungefähr, war doch die Universität Berkeley früh Unix System V-Lizenznehmer der Bell Labs (also AT&T). Die Universität hatte mit der Berkeley Systems Distribution (BSD) seine eigene, erweiterte Unix-Version im Angebot. Von BSD flossen viele Entwicklungen zurück in System V.

Ein weiteres Merkmal von RISC-Workstations ist die besonders starke Grafikleistung. Auch ist eine grafische Benutzeroberfläche fast immer vorhanden. Kein Wunder – Workstations eigneten sich besser als Mainframes dazu, neue Bedienkonzepte zu entwickeln und zu etablieren. Dieser technikhistorische Prozess des Entstehens und Optimierens von GUIs wird erlebbar beim Vergleich der Oberflächen des Xerox Alto, einer Apollo Domain Maschine



Die Apollo Domain DN4000 setzt auf einen Motorola 68030 Prozessor



Eine SUN-3/80 Workstation



Die SGI Iris Indigo, die erste Maschine gemäß der ACE Spezifikation

(also der CISC-Ära) mit den verschiedenen Ansätzen von SUN, DEC oder SGI bis hin zur Standardoberfläche CDE (common desktop environment). Genauso bezeichnend für Workstations sind aber auch die innovative und herstellerspezifische System- und Prozessorarchitektur und die Netzwerkfähigkeit mit TCP/IP.

Die großen Hersteller wurden nicht müde, sich gegenseitig mit besseren und schnelleren CPUs übertrumpfen zu wollen. Folgende sicher unvollständige Auswahl gibt hiervon einen Eindruck:

- DEC fuhr zweigleisig: Einerseits brachte der Hersteller CISC-Maschinen mit den hauseigenen 32-Bit-VAX-Hauptprozessoren für das Betriebssystem VMS heraus, andererseits erschienen MIPS-basierte RISC Workstations mit MIPS-CPU und DEC Ultrix (später OSF/1).
- SUN brachte die 32-Bit-SPARC und der 64-Bit-UltraSPARC Linie auf den Markt.
- Hewlett Packard nutzte ab 1986 die 32-Bit-PA-RISC-CPU (Precision Architecture), 1994 folgten 64-Bit-Modelle.
- Silicon Graphics setzte auf die "Microprocessor without interlocked pipeline stages" (MIPS) Architektur der MIPS Computer Systems Inc. aus Stanford.
- IBM brachte mit den RS/6000 Maschinen leistungsfähige Workstations mit PowerPC ("Performance optimization with enhanced RISC") Prozessoren heraus.
- Apollo Domain erlebte den Wandel von CISC zu RISC nicht mehr, in den Apollo-Workstations werkeln fast ausschließlich Motorola 68k-Prozessoren.

Diese Entwicklungen ließen sich die Hersteller entsprechend bezahlen, RISC Workstations waren niemals ein billiges Vergnügen.

Client/Server vs. Mainframes

Fließend ist die Grenze zwischen reinen Workstations zur Nutzung durch einen Anwender und dem Client-/Server-Computing. Ursprünglich sollte jede Workstation in einem Netz ihre speziellen Funktionen allen anderen Maschinen anbieten können. So konnte ein Drucker, ein Scanner oder auch ein Modem alle Maschinen im Netz entsprechend versorgen. Entsprechend wurde auch die Software aufgebaut: Ein Serverprozess auf der einen Maschine bedient die angeschlossene Hardware, ein Clientprozess kommuniziert über das Netz mit diesem Server. Client-/Serverarchitek-

turen erlauben auch andere Arten verteilter Software und eine Trennung von Anwendungen in eine Applikationsebene auf einem Server und einer Präsentationsebene auf einem Client. Eine weitere Maschine mag bei Bedarf eine Datenbank als Backend bereitstellen. Dieses Design unterscheidet sich von monolithischen Anwendungen auf einem Mainframe, die über Terminals bedient werden. Die Hersteller platzierten diese Ansätze als Alternative zu Mainframe-gestützten Anwendungen. Aber schnell war klar: Die zentralen Teile dieser Softwarearchitektur brauchen viel Power. Die Workstation-Hersteller brachten immer größere Systeme auf den Markt, abgeleitet aus der Architektur der ursprünglich kleinen Einzelplatzsysteme. Diese befriedigten den Leistungsbedarf von Datenbanken und Serveranwendungen und brachen in die Domäne der Mainframes ein. Mit Erfolg: Letztlich nahmen diese großen UNIX-Maschinen so schließlich die Rolle des Mainframe ein und verdrängten diese immer mehr. Heute bedienen Mainframes selbst meist Client-/Serversoftware und punkten mit hoher Skalierbarkeit.

Die Suche nach Standards

Der Markt der Workstations war in den letzten beiden Jahrzehnten des letzten Jahrhunderts heiß umkämpft. Viele Hersteller und beinahe genauso viele Prozessor- und Speicherarchitekturen wetteiferten um die Gunst des Kunden. Die Konsequenz: Um Interoperabilität der Systeme oder gar den Austausch von Software war es schlecht bestellt. Drittanbieter mussten ihre Produkte für eine Vielzahl von Plattformen anbieten, um den heterogenen Markt zu bedienen. Auch der Endanwender hatte dadurch nicht immer nur Freude – wer die Beschaffung von Systemen herstellerneutral ausschreiben musste, hatte über die Jahre bald einen Park der unterschiedlichsten Systeme im Hause. Großkunden wie Versicherungen entwickelten durchaus ihre eigene Middleware, um diese Sammlung bändigen zu können.

Das erkannten auch die großen Hersteller und so mangelte es nicht an Bemühungen, gemeinsame Standards zu etablieren. Die unterschiedlichen UNIX-Derivate wurden dank System V und dem POSIX-Standard miteinander kompatibel, zumindest in Bezug auf das Kompilieren des Quellcodes eines Programms. Auch das X Window System lieferte eine gemeinsam nutzbare Schnittstelle für grafische Anwendungen, die Motif-Bibliothek tat ihr übriges dafür. Der Anwender musste dank CDE beim Wechsel zwischen SUN-, HP- oder IBM-Maschinen nicht mehr umlernen, wenn es um die Nutzung der grafischen Oberfläche ging.

Beispiel ACE

Anfang der 1990er-Jahre trieb Compaq gemeinsam mit DEC und anderen die Gründung der Herstellervereinigung ACE (Advanced Computing Environment) voran. ACE sollte einen Standard für MIPS Rx000-basierte Maschinen etablieren und stellte sich gegen SUN auf. Sowohl SUNs eigene Maschinen als auch eine Reihe von SPARC-basierten Maschinen von Drittherstellern hatten eine starke Marktpräsenz. ACE definierte eine modulare Architektur, gab mindestens MIPS R3000-Prozessoren, 8 MByte RAM, Ethernet, SCSI-II und viele weitere Leistungsmerkmale vor. Als Betriebssystem war das UNIX System V Release 3-Derivat Irix 4.0 vorgegeben, die GUI stützte sich auf X11R4, Display-PostScript und OSF/Motif-Laufzeitbibliotheken. Die Referenzimplementierung von ACE ist wohl die SGI Indigo, eine Maschine in einem kompakten Towergehäuse, in frechem Lila gehalten. Was sich mit der Indigo machen lässt, zeigen Filme wie Terminator 2 und Jurassic Park. Lange währte die Freude aber nicht. Zuerst zog sich Compaq zurück, bald darauf auch DEC, um einen Alleingang mit dem Alpha-Prozessor zu unternehmen. Während also der Architekturstandard ACE Schiffbruch erlitt, segelte

SUN weiter mit gutem Wind voraus, dicht gefolgt von IBM und Apple, die mit dem PowerPC-Prozessor und der PReP-Plattform ein ganz eigenes Boot zu Wasser ließen. CPU-Hersteller MIPS wurde schließlich von Silicon Graphics übernommen. SGI selbst wurde 2016 dann von Hewlett Packard geschluckt.

Workstations heute

Letztlich machte die Heterogenität in Hardware und Software zusammen mit der drastischen Leistungssteigerung der PC-Hardware und ihrem niedrigen Preis den klassischen Workstations schließlich den Garaus. Auch *BSD und Linux als frei verfügbare UNIX-Betriebssysteme taten ihren Teil dazu. Nicht von ungefähr verstehen wir heute unter Workstation einen gut ausgestatteten Windows- oder Linux-PC oder einen fetten Apple Macintosh. Wer eine Maschine von SUN, SGI oder den anderen klassischen Herstellern erwirbt, sollte diese Geschichte im Blick haben. Zwar ist nicht jede Hardware gleichermaßen geeignet, dem Retrocomputer-Fan Freude zu bereiten, den meisten Spaß versprechen aber Systeme, die als Grafik-Workstations konzipiert sind. Hier sind die Parallelen in der Anwendung zu Windows-PCs oder Micro-

computern und Homecomputern am größten. Insbesondere 3D-Bildverarbeitung, die Verarbeitung analoger Videosignale oder auch die Sound-Erzeugung sind Domänen der entsprechenden Maschinen. Auf den nächsten Seiten beleuchten wir die Hersteller solcher Workstations kurz und nennen besonders interessante Modelle und Technologien.(gb)

Links

<https://www.bernd-leitenberger.de/cisc-risc.shtml>
https://de.wikipedia.org/wiki/Seymour_Cray
<https://www.digisaurier.de/computerhelden-19-david-patterson-der-erfinder-von-risc-und-raid/>
<https://de.wikipedia.org/wiki/SPARC-Architektur>
https://de.wikipedia.org/wiki/Sun_Microsystems
https://en.wikipedia.org/wiki/Sun_Enterprise
<https://de.wikipedia.org/wiki/PA-RISC>
https://de.wikipedia.org/wiki/Virtual_Address_eXtension

Quellen

iX Heft 11/2001, Heise Verlag

Die Menschen dahinter

Hinter den technischen Innovationen und den Produkten, die diese Innovationen zu den Nutzern tragen, stehen immer Menschen mit dem Antrieb, etwas Neues schaffen zu wollen. Wir stellen einige davon vor:



John Cocke (Mai 1925 - Juli 2002) wechselte 1956 von der Duke University als promovierter Mathematiker als Forscher zu IBM, wo er bis 1992 blieb. Seine Arbeiten zur Compileroptimierung sind essentiell für den Einsatz von RISC-Computern.

Sophie Mary Wilson (geboren 1957 in England) arbeitete nach ihrem Studium ab 1979 in der Firma Acorn, gegründet von ihrem früheren Lehrer Hermann Hauser. Zusammen mit Steve Furber entwarf sie den ARM-32-Bit-RISC-Prozessor und seinen Befehlssatz.



ARM-Prozessoren wurden in den Acorn-Computern Archimedes und RiscPC eingesetzt, Weiterentwicklungen treiben heute Smartphones und die Apple Macintosh M1 Serien an.



Andreas von Bechtolsheim (geboren September 1955) studierte zunächst an der TU München und wechselte dann als Stipendiat an die Carnegie Mellon University in Pittsburgh, USA. Anfang der

1980er Jahre entwickelte er mit Unterstützung der DARPA auf Basis des Motorola 68000-Prozessors einen Arbeitsplatzrechner für das Computernetzwerk der Universität. 1982 gründete er die Firma Sun, um seine Entwicklung für den kommerziellen Markt herstellen zu können. Später wechselte er zum Netzerkausstatter Cisco und schließlich zu Arista Networks. Seine Tätigkeit als Investor half unter anderem auch den Stanford-Studenten Larry Page und Sergei Brin bei der Gründung des Unternehmens Google.



Bill Joy (geboren 8. November 1954) studierte Elektrotechnik und Informatik an der Universität von Kalifornien in Berkeley und wirkte dort ab 1977 maßgeblich an der Berkeley Software Distribution (BSD) mit. Er trieb die Weiterentwicklung von TCP/IP voran und gründete zusammen mit Vinod Khosla, Scott McNealy und Andy Bechtolsheim die Firma Sun Microsystems. Dort war er maßgeblich an der Entwicklung der SPARC-Prozessoren, aber auch an Solaris (SunOS), Java und Jini beteiligt.



James Henry Clark (geboren März 1944) gründete mit Absolventen der Stanford University die Firma Silicon Graphics, die Workstations mit sehr leistungsfähiger Grafikhardware herstellte. 1993 gründete Clark zusammen mit Marc Andreessen die Firma Netscape Communications Corporation.



John William Poduska Sr. (geboren 1937) studierte Elektrotechnik am Massachusetts Institute of Technology. 1980 gründete er zusammen mit anderen die Firma Apollo Computer Inc. Dort entstand auch der RISC-Prozessor PRISM (Parallel Reduced Instruction Set Multiprocessor) für die Apollo Domain DN10000 Workstation.

Fast vergessen – Apollo Domain



Apollo Computer Inc. wurde 1980 von William Poduska und anderen in Chelmsford (Massachusetts) gegründet. Neben Symbolics und Sun Microsystems war Apollo einer der Pioniere bei grafischen Workstations. Ein Jahr später kam die DN100-Workstation auf den Markt. Die Motorola 68000-basierte Maschine läuft unter dem proprietären Betriebssystem Aegis. Die Modellpalette erweiterte sich nach oben mit zahlreichen Modellen mit Motorola 68030/50 MHz (9000/400DL) und 68040/33 MHz (9000/433S). In Deutschland wurden die Apollo Domain Maschinen von Siemens über der Bezeichnung WS vertrieben. Die Maschinen sind mit unterschiedlichen Grafikkarten ausgestattet, je nach Modell für die Nutzung an monochromen ECL-Monitoren oder an RGB-Farbbildschirmen. Massenspeicher sind bei älteren Modellen über ESDI angeschlossen, neuere verwenden SCSI. Das proprietäre Bussystem wich bald dem 16-Bit ISA-Bus (AT-Bus), für den es auch PC Emulatorkarten mit Intel 80386-Prozessor gibt. Zusammen mit der entsprechenden Software wird die Apollo Domain so MS-DOS-kompatibel.

Aegis besitzt eine POSIX-kompatible Unix-Shell, ist aber eine Eigenentwicklung von Apollo. Sein Konzept basiert auf Multics, es ist jedoch in einer proprietären Version von Pascal geschrieben. Seine Stärke liegt in der Netzwerktransparenz: Kommandos können sowohl für die lokale Maschine als auch für entfernte Maschinen aus-

geführt werden und Dateien (memory mapped files) können sich über das gesamte Netz erstrecken. Möglich ist dies durch einen eigenen, auf Token Ring aufbauenden Protokollstack. Erst später lernte Aegis mit dem darüber liegenden Domain/OS (Distributed On-line Multi-access Interactive Network/Operating System) auch Ethernet und TCP/IP. Domain/OS 10, enthält viele Elemente von Unix und kennt bereits das X Window System. Apollo Domain war sehr erfolgreich am Markt, 1987 lag das Unternehmen beim Anteil am Workstation-Markt an dritter Stelle hinter Digital Equipment Corporation und Sun, aber vor Hewlett-Packard und IBM. Die Erfolgsstory endete um 1988 rasch infolge von Fehlern und Börsenspekulationen des Managements. 1989 übernahm schließlich Hewlett-Packard für 476 Mio. US-Dollar das Unternehmen. Die Apollo-Maschinen wurden zunächst unverändert neben den eigenen Workstations verkauft, bis 1997 war Apollo aber vollständig abgewickelt. Workstations wie die HP Apollo 715 sind PA-RISC basiert und mit HP-UX als Betriebssystem ausgestattet.

Stanford University Network wird Sun

Als Ausgründung der Stanford University im Jahr 1982 entwickelte sich SUN (eine Abkürzung für Stanford University Network) zum führenden Hersteller von UNIX-Workstations. Die Gründer Andreas von Bechtolsheim, Bill Joy, Vinod Khosla und Scott McNealy verständeten sich darauf, den Bedarf an leistungsfähigen, vernetzten UNIX-Maschinen zu decken. Mit den Motorola 68010-basierten Maschinen SUN-1 und SUN-2 aus dem Jahr 1983 brachte SUN gleich sein eigenes UNIX-Derivat heraus. SunOS 1.0 basiert auf 4.1BSD.

Schon 1985 begann SUN damit, eigene RISC Prozessoren zu entwickeln, die SPARC-Architektur. Als 1986 die SUN-4 Systeme mit diesen Prozessoren vom Band liefen, fand sich SunOS 3.2 auf den Platten. Es zeigte bereits Einflüsse von System V und unterstützte die neuen Prozessoren nur teilweise. Schließlich wechselte SUN im Jahr 1989 bei SunOS von BSD zunächst teilweise auf System V IPC. Diese Version unterstützt auch erstmals vollständig die 32-Bit SPARC-Architektur. 1989 ist auch das Gründungsjahr von SPARC International, einer Non-Profit Organisation zur Weiterentwicklung der SPARC Prozessoren. Hießen die Versionen bis 4.0 noch SunOS, führte SUN mit der Version 4.1.1 die Bezeichnung Solaris für seine UNIX-Variante ein. Ab Solaris 2.0 für SPARC war der Wechsel auf System V Release 4 vollzogen. SUN nummeriert ab Solaris 2.0 übrigens doppelt und übernimmt die Solaris-Version-

nummer als Minor-Version für SunOS. Solaris 2.0 ist SunOS 5.0, Solaris 7 entspricht SunOS 5.7 – etwas Verwirrung erhält ja die Aufmerksamkeit. 1995 präsentierte SUN dann die ersten Maschinen mit dem UltraSPARC-Prozessor, bereits eine 64-Bit CPU.

Die Modellpalette der UltraSPARC-basierten Systeme reichte bis weit in die Domänen herkömmlicher Mainframes. 1997 erschien die SUN Enterprise 10000 (Starfire), die mit bis zu 64 Ultra SPARC II-Prozessoren auf 16 CPU Boards bestückt werden konnte. Die Maschine ist das Ergebnis der Arbeit der Business Systems Division von Cray Research. SUN hatte diese Abteilung übernommen, nachdem Konkurrent Silicon Graphics 1996 Cray Research geschluckt hatte, aber kein Interesse an der Business Systems Division zeigte. Die Starfire ist in Domänen aufteilbar und kann insgesamt mit bis zu 64 GB RAM bestückt werden. Um die Jahrtausendwende herum war diese Maschine bei Unternehmen des Internet-Hype sehr beliebt. Am günstigen Preis lag das nicht, für eine voll bestückte Starfire verlangte SUN etwa 1 Million US-Dollar. SUN nutzte den Markterfolg zur Übernahme verschiedener Firmen, unter anderem Tape-Roboterhersteller StorageTek im Jahr 2006. Doch der Boom währte nicht ewig – 2009 wurde SUN schließlich selbst geschluckt. Datenbank-Spezialist Oracle ließ sich die Übernahme immerhin 7,4 Milliarden US-Dollar kosten.



Pionier Hewlett Packard

Hewlett Packard hatte zunächst UNIX-Workstations auf Basis der Motorola 68000-Prozessoren im Angebot, also CISC-basierte Maschinen. Die Serie 200 erschien erstmals 1981, 1985 folgte die Serie HP 9000 / 300. Die von Apollo Computer übernommenen DN-Maschinen gliederte HP als HP Apollo 9000 Serie 400 in das eigene Portfolio ein. Die Entwicklung der eigenen PA-RISC-Prozessoren lief parallel und 1986 lieferte Hewlett Packard mit den Modellen 930 (identisch mit HP 9000 840s) die ersten RISC-Workstations aus. Sie nutzen den PA-RISC-7000 Prozessor mit 32-Bit Adressraum und einem großen, per Bussystem angebundnen L1-Cache, aber ohne L2-Cache. Die Prozessoren sind mit 35 bis 66 MHz getaktet.

Die Liste der PA-RISC basierten Systeme ist lang, aber durch die Trennung in verschiedene Serien noch halbwegs zu überblicken (siehe <https://www.openpa.net/systems/>). Die HP 9000-Maschinen sind Workstations verschiedener Leistungsklassen. Die Anfang der 1990er Jahre vorgestellte HP 9000 715-Serie stellt kompakte Desktops mit PA-RISC-Prozessoren zwischen 33 und 100 MHz und moderater Ausbaubarkeit hinsichtlich Speicher und Festplatten bereit. Ganz anders konzipiert ist der HP 9000 755 Tower, hier passen bis zu 768 MB RAM und zwei 5,25 Zoll SCSI-Platten in voller Bauhöhe hinein.

Die HP Visualize Workstations kamen Mitte der 1990er Jahre auf den Markt und sind in die B-, C- und J-Serien zu unterscheiden. In allen Serien kommen PA-RISC Prozessoren unterschiedlichem Entwicklungsgrad und unterschiedlicher Taktfrequenz vor. Jüngster Spross dieser Linie ist wohl die C8000 aus dem Jahr 2004, die von zwei PA-8800 Prozessoren mit je zwei Kernen und einer Taktfrequenz von bis zu 1 GHz angetrieben wird. Daneben existieren mit den HP 9000 800-Maschinen auch Server mit unterschiedlicher Ausstattung und Umbauten zu tragbaren Maschinen.

Es lohnt sich vor dem Kauf einer HP-Workstation daher genau zu schauen, in welche Serie die Maschine genau gehört und welche Betriebssysteme sie fahren kann. Neben HP-UX als hauseigenem UNIX lassen sich auch OpenBSD und NetBSD, Linux und NeXT OpenStep installieren – nur eben nicht gleichermaßen auf jeder Maschine. Tückisch auch der Anschluss von Tastatur und Maus – ältere Modelle verwenden hierfür das HP HIL-Bussystem oder einen Konnektor zum Anschluss von HIL-Komponenten und PS/2-Geräten. Auch hier muss der geneigte Sammler genau hinschauen, denn HP HIL-Geräte sind selten zu bekommen und entsprechend teuer. Adapter von HP HIL auf PS/2 sind bisher nicht gesichtet worden.



Silicon Graphics – der Grafikspezialist

SGI begann mit den IRIS Modellen 1000 und 1200 mit Motorola 68000-Prozessor im Jahr 1984. Die Maschinen waren als festplattenlose Grafikterminals für Maschinen wie die DEC VAX gedacht. Erst spätere Serien einen eigenen Massenspeicher mit. Je nach Modell musste der Kunde zwischen 45.000 und 100.000 US-Dollar für die Geräte locker machen, was trotz der geringen Zahl verkaufter Einheiten dem Unternehmen einen ansehnlichen Umsatz bescherte. 1991 orientierte sich Silicon Graphics um und stellte mit der IRIS Crimson (ausgestattet mit einem 64-Bit MIPS-R4000-Mikroprozessor) seine erste RISC-Workstation vor. Der Prozessor stammte von der 1984 gegründeten MIPS Computer Systems Inc, gegründet vom späteren Stanford-Präsidenten John L. Hennessy. SGI kaufte gleich danach den Prozessorhersteller auf, und benannte ihn ab 1992 in MIPS Technologies Inc. um.

Danach folgten eine Reihe sehr erfolgreicher Grafik-Workstations. Die IRIS Indigo mit Prozessoren der R4000-Serie machte als erstes Modell der ACE Referenzimplementierung Furore und gilt unter Sammlern oft als die schönste der SGI-Workstations. Nachfolgemodelle waren die Indigo2- und Indy-Workstations, letztere aus dem Jahr 1993 und in ein flaches Pizzabox-Gehäuse verpackt. Während die Indigo noch ein SGI-eigenes Bussystem für Tastatur und Maus besitzt, nutzen die Nachfolgemodelle die PS/2-Anschlüsse. Direkter Nachfolger der Indy ist die O2-Workstation aus dem Jahr 1996

mit R5000- oder R10000-Prozessoren und maximal 256 MB RAM. Die Octane-Workstation kam 1997 auf den Markt und ist mit zwei R10000 Prozessoren bestückt. Die 2002 erschienene Fuel fasst bereits maximal 1 GB RAM und ist mit Prozessoren bis hinauf zur R16000 mit 900 MHz bestückt. Ähnliches gilt für den Octane-Nachfolger Tezro, nur vermag die Rack-Version dieser Maschine schon bis zu 16 GByte RAM zu nutzen.

Neben diesen Workstations für den Schreibtisch brachte SGI auch eine Reihe großer Servermaschinen auf den Markt. Unter den Onyx- und Origin-Maschinen finden sich kühlstrangkroße Varianten der Workstations ebenso wie Rack-montierte Supercomputer für Grafikanwendungen. Aber SGI hat mit Kooperationen auch andere Märkte bedient. Die 1996 auf den Markt gekommene 64-Bit-Spielkonsole Nintendo 64 enthält einen Grafikprozessor von MIPS Technologies.

Von Haus aus verwenden SGI-Systeme das UNIX Derivat IRIX. IRIX 3.x basiert auf UNIX System V Release 3 und enthält Komponenten von 4.3BSD. Die 1993 erschienene Version 5.x hat Eigenschaften von UNIX System V Release 4 übernommen, unter anderem das ELF-Binärformat. In späteren Versionen fügte SGI das XFS Journaling Dateisystem hinzu. Erst mit Version 6 unterstützte IRIX dann alle 64-Bit-Features der MIPS Prozessoren. Letzte freigegebene Version ist IRIX 6.5 aus dem Jahr 2006.



Big Blue mischt mit – IBM RS/6000



IBM hatte schon früh kleine Systeme für den Einzelanwender im Programm. In die Riege der Anbieter von RISC-Workstations stieg das Unternehmen 1986 mit dem IBM RT ein. Als Nachfolger des IBM 801 gehandelt, lieferte IBM sowohl eine Towerversion (Modell 6150) als auch eine Desktopversion aus (Modell 6151). Beide nutzen den 32-Bit-ROMP-Prozessor, an dem IBM seit 1977 entwickelte und der seit 1981 kommerziell verfügbar war. Er ist auf der Standard-Prozessorkarte des IBM RT mit 5,88 MHz getaktet, die Advanced Processor Card (ab 1987) taktet mit 10 MHz, die Enhanced Advanced Processor Card (ab 1988) mit 12,5 MHz. Die Geräte nutzen IBM AIX als UNIX-Derivat, außerdem kann das Pick Operating System und das Academic Operating System (AOS) als IBM-Port von 4.3BSD gefahren werden.

Im Februar 1990 stellte IBM dann das RISC System 6000 (kurz RS/6000) vor, angetrieben von der POWER1 CPU. Diese besitzt eine Multi-Chip-Architektur. Die RIOS-1 Konfiguration besteht aus 10 einzelnen Chips, nämlich einem Instruction Cache Chip, je einem Festkomma- und einem Fließkomma-Chip, vier Data Cache Chips, einem Storage Control-Chip, zwei Ein- und Ausgabe-Chips und einem Clock Chip.

Die parallel erschienene RIOS.9 CPU ist weitgehend identisch, besitzt aber nur zwei Data Caches. 1992 erschien mit der RSC (RISC Single Chip) eine Version dieser Architektur auf einem einzigen Chip. Die Nachfolgeserie POWER2 baute diese Architektur durch je einen weiteren Festkomma- und Fließkomma-Chip aus. RS/6000 Systeme erschienen bis 1995 in verschiedenen Ausbaustufen.

Als der PowerPC-Prozessor, ein gemeinsames Werk mit Apple und Motorola im Jahr 1992 erschien, wechselte IBM auch für die RS/6000-Maschinen auf diesen Prozessor. Der PowerPC setzt auf der RISC-CPU auf und hat über die Jahre viele Weiterentwicklungen erfahren. Dementsprechend vielfältig ist die Palette an PowerPC-basierten RS/6000-Maschinen. Sie unterscheiden sich in der Auslegung als Workstation oder Server, im CPU Modell und der Anzahl von CPUs und dem maximalen Arbeitsspeicherausbau. Kam anfangs noch der MCA-Bus zum Einsatz, benutzten die späteren Modelle den PCI-Bus.

Auch die RS/6000-Maschinen liefen mit IBM AIX als Betriebssystem. Daneben existieren Versionen von Microsoft Windows NT 3.51 und Sun Solaris, auch Novell Netware, OS/2 und Taligent waren geplant. Aber nicht einmal das IBM-eigene OS/2 ist in einer endgültigen Fassung für RS/6000 erschienen.

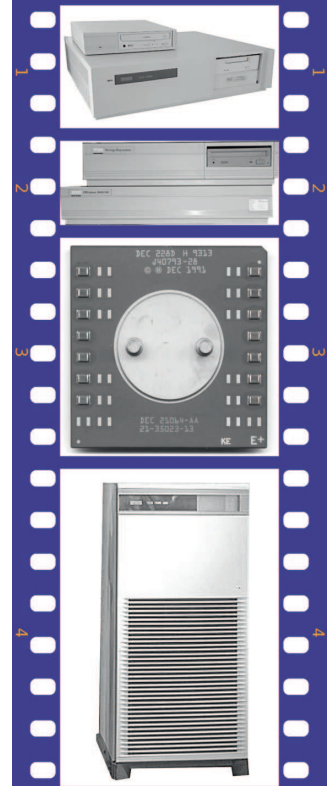
Digital Equipment goes RISC

DEC war im Workstation-Bereich spätestens seit 1984 aktiv, als mit der Vaxstation I das erste schreibtischtaugliche Modell erschien. Angetrieben von einem KD32-CPU-Modul (4 MHz) mit dem MicroVAX-I-Prozessor war der Rechner die erste in VLSI Technik gebaute VAX (Virtual Address eXtension). Dieser Prozessor ist ebenso wie seine Nachfolger eine CISC-CPU. DEC bot neben dem hauseigenen VMS mit Ultrix auch ein 4.2BSD-basiertes UNIX Betriebssystem an. Es unterstützt neben den Vax-Maschinen auch die DEC PDP-11.

Ab 1989 sprang DEC auf den RISC-Zug auf und brachte die DECstation-Serie auf den Markt. Die Systeme nutzen die MIPS-Prozessoren der Silicon Graphics Tochter MIPS Technologies, Inc. Die Modelle DECstation 3100 und 2000 nutzen den R2000 Prozessor und können mit Monochrom- und Farbgrafikkarten (8 Bit Farbtiefe) ausgestattet werden. Ethernet und SCSI runden die Systeme ab. Daneben bot DEC die Personal DECstation 5000 Serie als Einstiegsmodelle an. Sie kamen mit MIPS-R3000A oder R3010-CPU auf den Markt. Mit einem Preis von unter 1.000 US-Dollar für die kleinste Ausstattung, zielten die Modelle auf den Markt, den Sun bis dahin besetzt hielt. In anderen Regionen bewegten sich die DECstation 5000 Modell 100 und 200. Modell 100 ist mit R3000A- oder R4000-Prozessor bestückt zu finden, das Modell 200 nutzt eine R3000-, R3400- oder R4400-CPU. Die Geräte schlucken verschiedene Farbgrafik-

karten sowie 2D- und 3D-Beschleuniger, immer angebunden über den DEC-eigenen Turbo Channel. Mainboard, CPU-Modul, RAM, Netzteil und Grafikkarten nehmen soviel Platz im Desktop-Gehäuse ein, dass Massenspeicher in ein eigenes Gehäuse ausgelagert werden müssen, die per SCSI angebundene Expansion Box. Die Modell 200-Maschinen können beeindruckend viel Arbeitsspeicher nutzen: Mit bis zu 15 SIMM-Modulen in DEC-eigener 128-Pin-Ausführung sind bei Verwendung von 32-MB-Modulen insgesamt 480 MByte RAM möglich – nicht gerade wenig für eine Workstation von 1994.

DEC glaubte indes nicht an eine große Zukunft der MIPS-Prozessoren. Stattdessen machte sich das Unternehmen daran, aufbauend auf den Konzepten von PRISM, einem Forschungsprojekt von DEC aus Mitte der 1980er Jahre, eine eigene CPU zu entwickeln, den DEC Alpha AXP. Ungebremst von Kompatibilitätsvorgaben setzte die CPU von Anfang an auf eine 64-Bit-Architektur. Mit Taktraten von anfangs 200 MHz (Modell 21064) im Erscheinungsjahr 1992 bis zu 1,3 GHz im Jahr 2004 (Modell 21364) hatte DEC wirklich schnelle CPUs im Portfolio. 1994 setzte DEC im Workstation-Bereich ganz auf den Alpha-Prozessor. Die DEC 3000 AXP-Modelle waren noch nahe an den MIPS-basierten Workstations, die DEC 4000 AXP zielten hingegen auf den Servermarkt. Noch größer sind die Maschinen der DEC 7000 AXP und DEC 10000



NeXT Computer Inc. – Leistung ohne RISC

NeXT Computer, Inc. war ein Unternehmen des Apple Gründers Steve Jobs und wurde gegründet, nachdem Jobs die Pionierfirma aus Cupertino verlassen hatte. Bei seiner Gründung 1986 war der Öffentlichkeit noch unklar, was NeXT eigentlich produzieren und verkaufen wollte. Erst 1988 wurde der Schleier gelüftet und Jobs stellte in einer typisch pathetischen Weise den NeXT Computer vor. Positioniert als Computer für die höhere Bildungsschicht, beeindruckte der schwarze Würfel mit einer Kantenlänge von 301 mm – also 1 Fuß nach klassischem Maß – schon optisch, ebenso wie der Graustufenmonitor mit seinem U-förmig geschwungenen Ständer. Es handelte sich nicht um eine RISC-Workstation: Anders als Sun, aber vergleichbar mit Apollo Computer werkelte hinter der Hülle aus einer Magnesiumlegierung ein Motorola 68030, unterstützt von einem 68882-Coprozessor und einem 56001 digitalen Signalprozessor (DSP). Diese Kombination erhöhte die Performance bei manchen Anwendungen deutlich und glückte die Schwächen des Hauptprozessors aus. Jobs verglich die Architektur des NeXT Computers in seiner Präsentation mit der klassischer Mainframes, was die Aufgabenteilung innerhalb der Maschine betrifft. 1990 brachte NeXT dann den leicht veränderten NeXTcube heraus,

der über einen 68040-Prozessor mit einem Takt von 25 MHz und einer Festplatte anstelle des im NeXT Computer verwendeten MO-Laufwerks verfügte. Für höhere Ansprüche in der Grafikdarstellung konnten die Würfel mit einer NeXTdimension genannten 32-Bit-Grafikkarte basierend auf dem Intel i860-Prozessor aufgerüstet werden. Ebenfalls 1990 lieferte NeXT die NeXTStation aus, ebenfalls mit einem 68040-Prozessor mit 25 oder 33 MHz (NeXTStation Turbo) und wahlweise mit Farbgrafik (NeXTStation Color).

Viel interessanter und bis in die heutige Zeit hineinwirkend sind das Betriebssystem, die Entwicklungsumgebung und die Programmiersprache der NeXT-Maschinen. Eng verbunden mit dem NeXT-Computer ist die grafische Entwicklungsumgebung Objects. Das ursprünglich NeXT Step genannte Betriebssystem setzt auf einem Mach Microkernel auf. Dieser verwaltet und abstrahiert die Hardware, alle weiteren Funktionen laufen als Prozesse dieses Kernels. Der Mach Kernel führt dann eine auf 4.3BSD-basierende UNIX-Umgebung aus. Diese Architektur machte es später leicht, das Betriebssystem auch auf andere Architekturen zu portieren. Um 1993 konnte NeXT-Step so neben auf Motorola-CPU's auch auf Sun SPARC, HP PA-RISC und Intel 80486-Prozessoren laufen.

Von Anfang an setzte NeXT bei der Programmentwicklung neue Maßstäbe. Der Interface Builder erlaubt es, komplexe GUI-Anwendungen wie in einem Grafikprogramm zu designen und mit Programmcode zu hinterlegen. Als Programmiersprache wird dabei Objective C genutzt. Gemeinsam mit Sun definierte NeXT mit Openstep eine objektorientierte API und erleichterte so die Softwareentwicklung entscheidend. Programme wurden daher oft als NIHS-Versionen publiziert, taugten also für NeXT, Intel-, HP- und Sun-Maschinen gleichermaßen. 1993 gab NeXT schließlich die eigene Hardware auf und konzentrierte sich ab 1995 als NeXT Software, Inc. ganz auf API und Betriebssystem, nunmehr als OPENSTEP vermarktet. Als Steve Jobs im Dezember 1996 NeXT an Apple verkaufte und schließlich zu Apple zurückkehrte, wurde OPENSTEP und der Mach Kernel zur Basis von MacOS X. Auch die Entwicklungsumgebung und Objective C leben bei Apple in Xcode weiter.

AXP Serien, sie sollten im Rechenzentrum gleichauf mit Mainframes rangieren.

Mit Windows NT, DEC OSF/1 und Tru64 UNIX als Ultrix-Nachfolger sowie OpenVMS unterstützen gleich drei Betriebssysteme die DEC-eigene Architektur. Und weil ohne Anwendungen nichts geht, rühmte sich DEC zu dieser Zeit mit 2430 Programmen, die unter OpenVMS laufen, 2506 unter DEC OSF/1 und 793 unter Windows NT. Nach unten hin rundeten die Alpha PCs der Handelsketten Vobis und Escom die Workstation-Palette ab, diese laufen unter Windows NT 4.0. Für Server und Workstations hingegen platzierte DEC dann doch lieber OSF/1 oder eben OpenVMS – je nach benötigter Anwendung. Der Alpha AXP-Prozessor mit 320 MHz wurde dennoch auf der CeBIT 1994 als schnellster PC angepriesen.

Dennoch genügten die Erfolge von DEC nicht, um ein langfristiges Überleben des Konzerns zu garantieren. Die Ausrichtung auf den Alpha AXP-Prozessor, ein eher nebenbei betriebenes Geschäft mit Intel-basierten PCs und Notebooks und das zunächst verschobene und schließlich ausgefallene Release eines 64-Bit-Kernels von Windows 2000 besiegelten das Ende des Unternehmens. 1998 übernahm Compaq die Firma und mit der Übernahme eben dieser Compaq durch Hewlett Packard verschwand einer der Pioniere der Computerindustrie.

Homecomputer und Konsolen

Wenn es um RISC-Prozessoren und RISC-Computer geht, darf eine Architektur nicht unerwähnt bleiben: ARM. Von den Acorn-Mitarbeitern Sophie Wilson und Steve Furber schon 1983 begonnen, lieferten die ARM2-Prozessoren in den Acorn Archimedes von 1986 und später im Acorn RiscPC eine erstaunliche Leistung. Seit dem Jahr 1990 als Advanced RISC Machines Ltd. ausgegründet, sind die Nachfolgemodelle heute mehr denn je Vorreiter für leistungsfähige Computer und stromsparenden Mobilgeräte. Smartphones und Tablets mit iOS und TabletOS von Apple oder solche mit Android von Google sind heute zahlenmäßig wohl die am meisten genutzten Geräte mit ARM-Prozessoren. Aber schon davor

machten ARM-Prozessoren mobil, wie der Apple Newton einem der ersten Personal Digital Assistants (PDA) zeigt.

Auch MIPS-Prozessoren sind außerhalb von Workstations anzutreffen. Die Nintendo64-Konsole, die Sony Playstation 1 und 2 und die Playstation Portable nutzen MIPS-Prozessorkerne. Als hoch integriertes "System on a stick" finden sie sich beispielsweise ebenfalls in PDAs, aber auch in Netzkomponenten oder Network Attached Storage (NAS)-Systemen.

Auch PowerPC-Prozessorkerne hat es auf Abwege verschlagen – sie treiben den Nintendo GameCube ebenso an wie die wenig erfolgreiche und selten anzutreffene Spielekonsole von Apple, den Pippin.



Leere Uhrenchips umschiffen

Starthilfe

Manche Workstation vergisst die eingestellte Uhrzeit nach dem Ausschalten und zeigt ein merkwürdiges Verhalten beim Hochfahren der Netzfunktionen. Vermutlich ist die Pufferbatterie leer – was nun?

Die allermeisten Workstations speichern grundlegende Systeminformationen in einem batteriegepufferten RAM (NVRAM, non volatile RAM) ab. Beliebt sind für diese Aufgabe die Uhrenchips der Hersteller SGS Thomson oder Dallas Semiconductor. So liegen Echtzeituhr und NVRAM zusammen mit einer Batterie in einem kompakten DIL-Gehäuse vor. Das Problem dabei: Auch wenn diese Batterie nicht viel zu tun hat, nach zwei bis drei Jahrzehnten ist der Strom verbraucht. Anders als bei Mikrocomputern wie AMIGA oder Apple laufen die Batterien in den DIL Gehäusen nicht ohne weiteres aus, da sie zusammen mit dem Speicher- und Uhrenchip in Epoxydharz vergossen sind. Das ist die gute Nachricht, aber die schlechte folgt sogleich: Die Betriebssysteme der meisten Workstations erwarten essentielle Angaben im NVRAM. Fehlt der Batteriestrom, geht der Inhalt des NVRAM verloren und die Maschine funktioniert nicht mehr richtig.

Es fehlt nicht an Anleitungen, wie mittels Teppichmesser oder Mini-Schleifmaschine ein Dallas-Chip so geöffnet werden kann, das die Batteriekontakte frei liegen und eine Knopfzelle als Ersatz eingebaut werden kann. Es bleibt aber die Aufgabe, das NVRAM dann neu zu programmieren. Wie das geht und welche Fehler ein leeres NVRAM verursacht, zeigen die nächsten Abschnitte. Wer den Aufwand des Batterieumbaus scheut, hat auch etwas von den folgenden Angaben. Nur sind dann diese Schritte bei jedem Boot der Workstation erneut erforderlich.

Sun Solaris

Sun Solaris fragt beim Systemstart unter anderem die System-ID und die MAC-Adresse des Ethernetanschlusses ab. Ohne ein entsprechend programmiertes NVRAM funktioniert der Autoboot der Workstation nicht. Auch eine Netzverbindung kommt nicht zustande, obwohl der Ethernetanschluss einen Link signalisiert. Der Befehl `test net` im OpenBoot zeigt einen Fehler an.



Enthält eine Batterie – der Uhrenchip vieler Workstations

SUN NVRAM programmieren

Solange der NVRAM-Chip nicht getauscht wurde, kennt sich die SUN sozusagen selbst nicht. Es fehlen aufrufbare Angaben für die Maschinen-ID und die MAC-Adresse. Diese lassen sich aber am Eingabeprompt des OpenBoot-PROM auch manuell setzen. Sie bleiben solange erhalten, wie die Maschine nicht ausgeschaltet wird. Die Kommandoabfolge zum Setzen der Bytes im NVRAM für eine Sun Ultra-1 ist etwas länger:

```
set-defaults
1 0 mkp
80 1 mkp
8 2 mkp
0 3 mkp
20 4 mkp
xx 5 mkp
yy 6 mkp
zz 7 mkp
0 8 mkp
0 9 mkp
0 a mkp
0 b mkp
xx c mkp
yy d mkp
zz e mkp
0 f 0 do i idprom@ xor loop f mkp
```

Die mkp-Befehle setzen jeweils ein Byte des NVRAM. An Position 0 muss immer der Wert 1 stehen (1 0 mkp). Position 1 ist Byte 1 der Host-ID; hier ist das "80" für die Ultra-1. Die Positionen 2 bis 7 enthalten die MAC-Adresse. Hier sind die ersten drei Bytes die Hersteller-ID des Ethernet-Chips, bei SUN Ultra1 sind dies die Werte "80", "8" und "0". Die Positionen 8 bis 11 (hexadezimal also 0xB) sind das Release-Datum des Systems; das lässt sich gefahrlos auf "0" setzen. Die drei letzten Positionen sind Bytes 2,3 und 4 der Host-ID. Diese muss identisch mit den drei letzten Bytes der MAC-Adresse sein. In der letzten Zeile folgt ein Forth-Befehl zur Erzeugung einer Prüfsumme über die eingegebene Bytefolge.

Nach Eingabe der Bytefolge startet der Befehl "reset" das System neu. Nun sollte neben dem Sun-Logo beim Boot die eingegebene MAC-Adresse stehen. Stimmt diese nicht, gab es einen Tippfehler in der Zeichenfolge.

Eine genaue Beschreibung zur Programmierung des NVRAM zeigt der Kasten oben.. Wichtig zu wissen ist hier, dass jedes Sun-Modell eine passende Angabe braucht. Insbesondere die Maschinen-IDs weichen verständlicherweise zwischen den Modellen entsprechend ab.

SGI Irix

Eine SGI Indy startet ohne Angaben im NVRAM meist problemlos durch, hat aber Schwierigkeiten im Betrieb. IRIX zieht die IP-Adresse des Hosts nämlich aus dem NVRAM. Ohne diese Angabe klappt die Netzkonfiguration nicht; spürbar wird das an langen Reaktionszeiten beim Anmelden und beim Aufruf von Programmen.

Die manuelle Anpassung ist recht bequem: Beim Systemstart genügt der Aufruf des „Configuration mode“ und dort aus dem Menü der Wechsel in den „Command Monitor“. Mit dem Befehl

```
setenv -f eaddr 08:00:69:nn:nn:nn
```

lässt sich die MAC Adresse des Systems setzen. Dabei steht nn:nn:nn für die Seriennummer auf der Rückseite der Maschine. Die IP-Adresse wird dann mit dem Befehl

```
setenv netaddr a.b.c.d
```

gesetzt, zum Beispiel mit

```
setenv netaddr 192.168.178.42.
```


Netzkonfiguration bei Sun

Netzhilfe

Eine UNIX-Workstation macht erst dann richtig Spaß, wenn sie auch vernetzt ist. Die allermeisten Workstations besitzen schon von Haus aus mindestens einen Ethernetport, sei es als BNC-Anschluss, RJ45-Buchse oder AUI-Anschluss. Geräte, die nur über einen BNC-Anschluss für ein Koax-Kabel (Cheapernet) verfügen, sind zwar untereinander leicht zu verbinden. Da heute aber Vernetzung mit Twisted Pair Kabeln und Switches die Regel ist, braucht eine Cheapernet-Verkabelung einen passenden Übergang. Glücklicherweise ist, wer einen großen Switch oder Hub mit einer BNC-Buchse besitzt. Aber auch kleine 2-Port Hubs mit BNC-Anschluss auf der einen und RJ45-Buchse auf der anderen Seite sind noch zu finden. Heute verfügen die allermeisten Switches über eine automatische Erkennung der Netzgeschwindigkeit (Autosensing), so dass sich der Nutzer keine Gedanken machen muss, ob sein Ethernet mit 10-, 100- oder 1000 MBit/s läuft. Bei Problemen lohnt es sich dennoch, entweder an der jeweiligen Workstation oder am Switch das Autosensing für den jeweiligen Port zu deaktivieren. Suchen beide Seiten gleichermaßen nach der richtigen Geschwindigkeit, kann mitunter überhaupt kein Ergebnis herauskommen.

Etwas mehr Kopfzerbrechen bereitet hingegen die Netzkonfiguration, denn hier gehen die unterschiedlichen UNIX-Derivate recht unterschiedliche Wege.

Als Beispiel wollen wir eine neue Sun mit dem Namen mysun01 konfigurieren und ihr die IPv4-Adresse 192.168.178.42 geben. Das Netz hat die Netzmaske 255.255.255.0, der Router zum Internet hat die Adresse 192.168.178.1 und ist auch DNS-Server.

Beginnen wir mit der Vergabe eines Namens für den Rechner. Er muss in der Datei `/etc/nodename` stehen, sie besteht also nur aus einem Eintrag:

```
mysun01
```

Für die Vergabe der IP-Adresse der Netzinterfaces sind unter Solaris die Dateien `/etc/hostname.interface` zuständig, also beispielsweise `/etc/hostname.hme0` für den Ethernetanschluss `hme0`. Die Datei kann entweder den Rechnernamen oder eine IP-Adresse nebst Netzmaske enthalten. Gültige Einträge sind also

```
192.168.178.42 netmask 255.255.255.0
```

für die Angabe von IP-Adresse und Netzmaske oder

```
mysun01
```

für die Angabe eines Rechnernamens. In beiden Fällen muss `/etc/hosts` dem Namen eine Adresse zuordnen:

```
mysun01 192.168.178.42
```

In der Datei `/etc/netmasks` müssen außerdem die Netzmasken der angeschlossenen Netze stehen. Das ist in unserem Beispiel eine Zeile:

```
192.168.178.0 255.255.255.0
```

Unter Solaris 10 und neueren Versionen werden IP-Adressen

in der Datei `/etc/inet/ipnodes` hinterlegt. Die erste Zeile setzt dabei IP-Adresse und Name für das Default-Interface, die weiteren für alle anderen Ethernetports. Manche Sun-Workstations verfügen über eine oder mehrere Quadport-Ethernetkarten und haben so eine größere Zahl an Interfaces. Ein Beispiel für `/etc/inet/ipnodes` mit drei Netzinterfaces:

```
192.168.178.42 mysun01
192.168.0.42 eri1
192.168.1.42 eri2
```

Damit wären die Interfaces konfiguriert. Nun kann es an die Routingkonfiguration gehen. In unserem Beispiel besteht sie aus der Angabe des Routers zum Internet, also des Defaultrouters. Das geschieht in der Datei `/etc/defaultrouter`:

```
192.168.178.1
```

Zu guter Letzt muss nun noch die DNS-Namensauflösung konfiguriert werden. Hierzu dient wie bei vielen anderen UNIX Derivaten die Datei `/etc/resolv.conf`:

```
nameserver 192.168.178.1
nameserver 8.8.8.8
```

Mit diesen beiden ersten Einträgen sind der Internetrouter als DNS-Server und ein weiterer Nameserver als Fallback definiert. Solaris nutzt nicht zwangsläufig DNS zur Namensauflösung, sondern kann beispielsweise auch NIS (network information service) hierzu heranziehen. Diese Entscheidung wird in der Datei `/etc/nsswitch` getroffen. Solaris bringt vorkonfigurierte Beispiele dafür mit. Es reicht also, die richtige Beispieldatei zu kopieren:

```
cp /etc/nsswitch.conf /etc/nsswitch.old1
cp /etc/nsswitch.dns /etc/nsswitch.conf
```

Der erste `cp`-Befehl sichert eine möglicherweise vorhandene Datei; er wird also fehlschlagen, wenn keine `/etc/nsswitch.conf` vorhanden ist.



Iomega Laufwerke nutzen

Anschluss- hilfe



Ursprünglich als Floppydisk-Ersatz gedacht, eignen sich die ZIP-Laufwerke von Iomega auch zum Datenaustausch an Workstations.

Allerdings muss es schon ein Modell mit einer SCSI Schnittstelle sein – Parallelport-Laufwerke sind nicht geeignet und IDE- oder USB-Modelle finden an klassischen Workstations eine passende Schnittstelle. Was erforderlich ist, um ein angeschlossenes Laufwerk auch nutzen zu können, unterscheidet sich je nach Hersteller und Betriebssystem.

Sun Solaris

Exemplarisch sei im folgenden beschrieben, wie ein ZIP-Laufwerk an einer Sun SparcStation 5 mit Solaris 7 einzurichten ist. Alles beginnt mit einer Definition in */etc/format.dat*:

```
disk_type = "Zip" \  
: ctrl = SCSI\  
: ncyl = 2046 : acyl = 2 : pcyl = 2048 : nhead  
= 2\  
: nsect = 40 : rpm : bpt = 20480  
partition = "Zip" \  
: disk = "Zip" : ctrl = SCSI \  
: 2 = 0, 192480 : 2 = 0, 1159168
```

Hierbei beschreiben die einzelnen Parameter des ersten Blocks den Anschluss und die Größen des Laufwerks:

- _____ disk_type: Symbolischer Name des Laufwerks, hier "Zip"
- _____ ctrl: Typ des Controllers, hier SCSI
- _____ ncyl: Anzahl der erlaubten Datenzylinder auf dem Laufwerk, hier 2046
- _____ acyl: Anzahl der Zylinder für Informationen wie der Liste defekter Zylinder, hier 2
- _____ pcyl: Anzahl der physikalischen Zylinder, hier 2048 (entspricht ncyl + acyl)
- _____ nhead: Anzahl der Schreib-/Leseköpfe, hier 2
- _____ nsect: Anzahl der Datensektoren pro Track, hier 40
- _____ rpm : bpt: Umdrehungen pro Minute für das Laufwerk, hier 20480

Der zweite Block legt fest, wie Partitionen aufgebaut sind.

- _____ partition: Ein symbolischer Name, hier "Zip"
- _____ disk: Name des zugeordneten disk_type, muss dem oben vergebenen Namen entsprechen
- _____ ctrl: wie oben
- _____ <n> = x,y: Angabe von Beginn und Ende der Partitionen. Beim ZIP-Laufwerk wird immer Partition 2 benutzt und es sind zwei überlappende Angaben erforderlich, hier 2 = 0, 192480 : 2 = 0, 1159168

Damit kennt Solaris 7 schon einmal das Laufwerk an sich. Nun müssen nacheinander die Befehle *drvconfig* und *disks* aufgerufen werden, um die Gerätedateien zu erzeugen. Den Rest erledigt das Dienstprogramm, das menügesteuert Plattenlaufwerke formatiert, partitioniert und benennt und fehlerhafte Sektoren verwaltet. Mit dem Aufruf *format* fragt die Workstation auf Plattenlaufwerke ab, liest ihre Bezeichnungen und zeigt eine Liste der Ergebnisse. Das ZIP-Laufwerk taucht hiermit der Angabe *<drive type unknown>* auf. In unserem Beispiel ist diesem Eintrag das Gerätedevice */dev/dsk/c0t5d0s0* zugeordnet. Je nach Ausstattung der Sun-Workstation kann die Bezeichnung abweichen. Im Dialog wird mit der Option "Label" diesem Gerät der Typ "ZIP" zugewiesen. Nun ist noch eine Partition anzulegen, dabei genügt es, die Vorgaben der entsprechenden Option zu übernehmen. Das Laufwerk ist nun partitioniert, es fehlt nur noch ein Dateisystem. Das erledigt der Befehl *newfs*:

```
newfs /dev/dsk/c0t5d0s0
```

Das dauert etwa 10 min, das ZIP-Laufwerk zeigt währenddessen kräftige Aktivitäten. Um es zu benutzen, braucht es nun noch in den Verzeichnisbaum eingehängt werden:

```
mount -F ufs /dev/dsk/c0t5d0s0 /mnt
```

Ein *umount /mnt* entfernt es dann wieder und mit

```
eject /dev/dsk/c0t5d0s0
```

wird das Laufwerk die eingelegte Disk aus.

Das gleiche Prinzip gilt auch für Iomega JAZ Laufwerke, die im Gegensatz zu ZIP-Laufwerken echte Wechselplatten sind. Für die 1 GByte- und 2 GByte-Modelle braucht es nur passende Einträge in die */etc/format.dat*:


```

disk_type="Jaz 1GB"\
:ctrl=SCSI\
:ncyl=1018:acyl=2:pcyl=1020:nhead=64\
:nsect=32:rpm=3600:bpt=16384
partition="Jaz 1GB"\
:disk="Jaz 1GB":ctrl=SCSI\
:2=0,2084864
disk_type = "JAZ 2GB"\
: ctrl = SCSI \
: ncy l = 952 : acyl = 2 : pcyl = 954 : nhead =
64 : nsect = 64 \
: rpm = 5400 : bpt = 32768
partition = "JAZ 2GB"\
: disk="JAZ 2GB" : ctrl = SCSI \
: 2=0,3899392

```

Alles übrige funktioniert wie oben für die ZIP-Laufwerke beschrieben.

NeXT-Workstation mit NeXTSTEP

Iomega JAZ-Laufwerke funktionieren auch wunderbar zusammen mit der schwarzen Hardware von NeXT Computers, Inc. Das Vorgehen ähnelt stark dem unter Solaris, was wenig verwundert, denn beide Betriebssysteme haben ihre Wurzeln in 4.3BSD.

Hier braucht die Datei `/private/etc/disktab` einen passenden Eintrag, im folgenden für ein 1 GB JAZ Laufwerk gezeigt:

```

iomega jaz|iomega jaz 1GB|iomega jaz 1GB
G.60|iomega jaz 1GB G.6002/1:\
:ty=removable_rw_sc-
:si:nc#1021:nt#64:ns#32:ss#512:rm#5394:\
:fp#320:bp#0:ng#0:gs#0:ga#0:ao#0:\
:os=sdmach:z0#64:z1#192:\
:pa#0:sa#2045952:ba#8192:fa#1024:ca#8:da#4096:r
a#10:oa=time:\
:ia:ta=4.3BSD:aa:

```

Nach einem Neustart des Rechners genügt es, ein Medium in das Laufwerk zu schieben. Das Betriebssystem meldet sich dann und schlägt vor, das Medium zu formatieren, entweder mit einem NeXT- oder einem Macintosh-Dateisystem.



Anders als das Iomega ZIP Drive handelt es sich beim Jaz um ein Wechselplattenlaufwerk. Es wurde sowohl als Einzellaufwerk im kompakten Gehäuse angeboten als auch als Einbaumodell für 2,5" Schächte. Die Laufwerke und Medien sind als 1- oder 2 GByte Modelle verfügbar.

Editor vi benutzen

Tipphilfe

Vi ist als Texteditor wohl auf jedem UNIX System zu finden. Die ursprüngliche Implementierung wurde von Bill Joy im Jahr 1976 für eine frühe BSD-Version geschrieben und von POSIX standardisiert. Diese Version wurde niemals im Quelltext veröffentlicht, jedoch existieren eine Anzahl von Klonen mit zum Teil wesentlichen Erweiterungen. Viele verbindet mit dem vi eine Hassliebe, denn seine Unterscheidung Befehlsmodus, Einfügemodus und Kommandozeilenmodus wirkt zunächst befremdlich. Dafür ist der Editor auf fast allen Systemen verfügbar, die Einsen und Nullen unterscheiden können und alle diese Reinkarnationen bedienen sich zumindest in den Grundzügen gleich. Darum gibt es hier eine kurze Liste mit den wichtigsten Funktionen. Zu allererst gilt es, sich den Wechsel zwischen den Modi einzuprägen.

- _____ ESC
Beendet den Einfügemodus und wechselt in den Befehlsmodus
- _____ Doppelpunkt
Wechselt aus dem Befehlsmodus in den Kommandozeilenmodus

Danach ist alles ganz einfach. Im folgenden sind Funktionen im Kommandozeilenmodus mit einem vorangestellten Doppelpunkt gekennzeichnet, alle anderen sind Funktionen des Befehlsmodus.

Eingabe vor dem Cursor	i
Eingabe hinter dem Cursor	a
Eingabe am Zeilenende	A
Überschreiben	R
Zeile unterhalb einfügen	o
Zeile oberhalb einfügen	O
Zeile löschen	dd
Zeile in Puffer kopieren	yy
Zeile aus Puffer einfügen	p
Zeichen unter Cursor löschen	x
Bis Zeilenende löschen	D
Datei schreiben	:w
vi verlassen	:q
vi verlassen ohne speichern	:q!
Nicht druckende Zeichen anzeigen / nicht anzeigen	:set list
	:set nolist
	:set number
	:set nonumber
Von Dateianfang bis Dateieinde Zeichen-folge ersetzen	:1,\$ s/alt/neu/g
Zeilenanfang	^
Zeilenende	\$
Dateianfang	gg
Dateieinde	G

Commodore, Unix und viele Fehlentscheidungen

UNIX für AMIGA

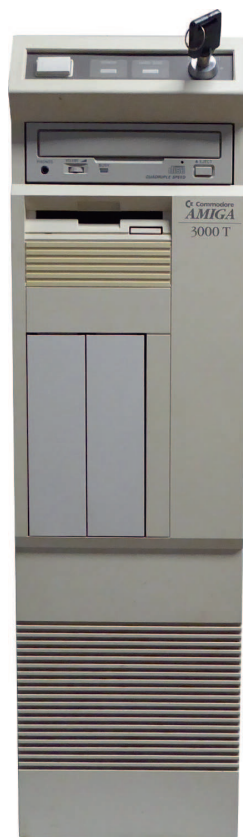
Die Führung von Commodore setzte in den letzten Jahren ihres Bestehens verschiedene Unix-Versionen in den Sand. Unerschrockene Aktivisten konnten jedoch zumindest AMIX als UNIX für den AMIGA retten.

Die Geschichte der Entwicklung der drei Unix-Versionen bei Commodore ist verworren und weitgehend unbekannt, obwohl sie durch Intrigen, Unfähigkeit und zwei Insolvenzen geprägt ist und so einen guten Stoff für einen Thriller abgeben würde. In diesem Artikel wage ich den Versuch, das Knäuel aufzudröseln und die einzelnen Handlungsstränge zusammenzuführen. Zum Schluss gibt es dann doch noch einen versöhnlichen Ausblick.

Beginn der Unix-Aktivitäten – der C900

1983 waren die Aktivitäten in Entwicklung und Verkauf bei Commodore vom C64 geprägt. Die PET-Serie („Personal Electronic Transactor“) für die Geschäftswelt erhielt in diesem Jahr mit dem Commodore 8296-D das letzte Familienmitglied und war technisch sichtlich ausgereizt. Auf der Suche nach einem zukunftsfähigen Produkt nahm Lloyd Taylor, Vizepräsident für Projektentwicklung, neben den PC-kompatiblen Computern auch das in Entwicklung und Wissenschaft bereits etablierte Unix-System in den Fokus (siehe das Buch von Brian Bagnall, „The Amiga Years“). Auch Commodore selbst hatte verschiedene Unix-Systeme im Einsatz, vor allem in der Buchhaltung und beim Design von Chips und Platinen.

Wegen aktueller Rechtsstreitigkeiten nahm man aber nicht den neuen MC-68000er-Prozessor von Motorola in den Blick, sondern den Z8001 von Zilog. Diesen durfte Commodore in der firmeneigenen Chip-Schmiede MOS Technology sogar in Lizenz herstellen, was die Entwicklung des neuen Rechners vereinfachte. Der intern „Z-Machine“ und später offiziell



Beinahe eine UNIX-Workstation –
Der AMIGA 3000T

„C900“ getaufte Rechner sollte 256kB an Arbeitsspeicher (bis 2 MByte erweiterbar) und zwei Floppy-Laufwerke, jedoch keine Festplatte erhalten. Später wurde der minimale Arbeitsspeicher auf 512 kByte festgelegt und eine 20 MByte-Festplatte in den Prototypen verbaut. Der Verkaufspreis war auf unter 2.000 US-Dollar anvisiert. Die Ingenieure bei Commodore waren von diesen Plänen aber nicht sehr angetan. Sie hätten lieber einen Business-Computer in

einer Art weiterentwickeltem C64 gebaut. Zumindest hätten sie als Prozessor des C900 statt des Z8001 lieber den MC68000 gewählt, weil dieser leistungsfähiger und bei den Softwareentwicklern weiter verbreitet war. Derartige technische Fragen wurden aber damals eben vom Management selbst entschieden.

1984 verließen einige der wichtigsten am C900 arbeitenden Entwickler Commodore vor allem in Richtung Atari. Dies warf das ganze Projekt zurück.

Software für den C900

In den 80er-Jahren gab es nicht das „eine“ UNIX, sondern verschiedene Firmen brachten jeweils ihr eigenes UNIX auf den Markt. Dabei durfte man den Quellcode des ursprünglichen AT&T Unix ab den frühen 1980er Jahren nicht mehr verwenden, sondern musste völlig eigenständige Fassungen der Software entwickeln. Commodore hatte damals keine Entwickler mit Unix-Kenntnissen, so dass man auf das bereits veraltete Coherent OS zurückgriff, das neben anderen Einschränkungen keinerlei Netzwerkfähigkeiten hatte. Coherent OS war ein sogenanntes „Version 7 Unix“, das auf dem technischen Stand von 1979 war. Da bei Coherent ebenfalls viele wichtige Entwickler die Firma verlassen hatten wurde kaum mehr an der Software weitergearbeitet. Eine Firma, der gerade die wichtigsten Hardware-Entwickler weggegangen waren nutzte also ein veraltetes Betriebssystem einer Firma, der die wichtigsten Software-Entwickler weggelaufen waren.

Auch bei der Entwicklung einer grafischen Benutzeroberfläche lief nicht alles rund. Erst durch die Unterstützung externer Entwickler konnten erste erfolgreiche Versuche zur Nutzung von Grafik unternommen werden. Wie auch bei Unix selbst waren auch die Benutzeroberflächen zu dieser Zeit individuelle Einzellösungen. Das später als Quasi-Standard weit verbreitete X Window System („X11“) kam erst im September 1987 auf den Markt, also zu einer Zeit, als das C900-Projekt schon fast fünf Jahre alt war – und bereits zwei Jahre tot.

Die erste Commodore Unix-Workstation steht kurz vor dem Verkauf

Im April 1985 sollte der Prototyp des C900 erstmals auf der Hannover-Messe vorgestellt werden, zusammen mit dem C128. Das Gehäuse ähnelte dem des späteren Amiga 2000, war aber etwas größer. Der Amiga 1000, obwohl ebenfalls weitgehend fertig, wurde dort erstaunlicherweise nicht gezeigt. Das Betriebssystem des C900 lief und rief zusammen mit der hochauflösenden Grafikoberfläche intern schon vorab Begeisterung hervor. Auch die Konkurrenz war beeindruckt. Commodore-Chefentwickler Bob Welland berichtete, dass mehrere Vertreter von SUN Microsystems lange staunend vor dem Rechner gestanden hätten. Kein Wunder, der Rechner konnte neben einer beeindruckenden Grafik auch mit Timesharing, Multitasking, einer funktionierenden Memory Management Unit (MMU), virtuellem Speicher und – über zwei bzw. vier RS232-Schnittstellen zum Anschluss von Terminals – auch mit einer Multi-User-Fähigkeit aufwarten. Neben den Fähigkeiten einer Grafik-Workstation war der C900 also auch als Mainframe mit bis zu fünf gleichzeitig arbeitenden Nutzern einsetzbar.

Es wurden in der zweiten Jahreshälfte 1985 etwa 400 Exemplare des C900 in zwei Konfigurationen mit unterschiedlicher monochromer Auflösung gebaut und an Softwareentwickler und Computermagazine geschickt. Schon am 3. Dezember wurde dann aber in der Zentrale von Commodore das Ende des C900 beschlossen. Im Ressourcen- und Aufmerksamkeits-Dreikampf mit dem Amiga und den PC-Nachbauten zog der C900 zum Schluss den Kürzeren. Nach dem Einstampfen der Serie mussten die verliehenen Computer zur Vernichtung zurückgeschickt werden. Andere Quellen sprechen davon, es habe sowie so nur 50 gebaute C900 gegeben, und diese seien 1986 für 2.500-4.000 US-\$ pro Stück verkauft worden. Jedenfalls haben nur wenige Exemplare überlebt.

Parallele Unix-Aktivitäten bei den PC-Kompatiblen

Schon früh in der Entwicklung des C900 hatte sich herausgestellt, dass das Management bei Commodore Unix eine große Zukunft vorhersagte – nicht nur in der Buchhaltung, sondern in Banken und Firmen insgesamt. Der Stern des Z8001-Rechners sank trotzdem immer schneller, aber ein eigener Unix-Rechner sollte unbedingt auf den Markt kommen. Da passte es gut ins Bild, dass Commodore seit Anfang der 80er-Jahre eigene PC-kompatible Systeme auf den Markt gebracht hatte. In etwa

Mein Amiga 3000T

1994 war ich in der Schweiz mit meiner Diplomarbeit beschäftigt, als mich der Anruf eines Freundes erreichte: Ob ich ihm bitte aus der Nähe von Zürich einen Computer, den er gerade dort gekauft hatte, mitbringen könne? Es handelte sich um einen Amiga 3000T. Zu dieser Zeit hatte ich immer noch meinen Amiga 1000, den ich im Juni 1986 erworben hatte, als Arbeitsrechner. Ich hatte ihn zwar um 2MB Hauptspeicher, eine 45MB SCSI-Festplatte am Parallel-Port und um eine Uhr erweitert, die Geschwindigkeit war aber dieselbe geblieben. Zu dieser Zeit überlegte ich, den Rechner mit einer Turbokarte, mehr Speicher, einer größeren Festplatte sowie Kickstart auf ROM statt von Diskette aufzurüsten. Die zu erwartenden Kosten von fast 3.000 DM hielten mich davon ab, hier tätig zu werden. Gerade der Einsatz von LaTeX und der notwendigen Generierung von Zeichensatzdateien, die Wochen an Rechenzeit benötigten, ließen mich aber auf eine schnelle Lösung hoffen.

Beim ersten Blick auf den neuen Rechner beim Auspacken ließ mich den schnellen Entschluss fassen, den gleichen Computer kaufen zu wollen. Alleine die Möglichkeit fehlte: Angeblich war der Züricher 3000T der letzte 3000er Amiga, der in Europa neu verkauft wurde. Und tatsächlich, nirgends mehr war noch ein Amiga 3000 zu bekommen, weder als Desktop noch als Tower.

Zur Computer Messe '94 in Köln hatte ich dank Presseausweis eine Stunde früher Zutritt. Das kam mir sehr zu pass, denn einer der Händler hatte den Amiga 3000T neu für 2.500 DM angeboten, also 75% unter dem Listenpreis von 10.000 DM. Leider erfuhr ich vor Ort, dass er von seinem Großhändler kein einziges Exemplar geliefert bekommen hatte. Bei einem zweiten Händler erfuhr ich hingegen, wo noch sechs neue Amiga 3000T stehen würden – nämlich im Büro des Konkursverwalters, der aus im Lager herumliegenden Einzelteilen noch insgesamt

sechs Rechner zusammenbauen ließ. Dieser zweite Händler wollte tatsächlich den vollen Listenpreis dafür haben. Ich kam aber auf eine andere Idee. Ich brachte den ersten Händler und den Konkursverwalter unter der Bedingung zusammen, dass ich einen der sechs Rechner für den beworbenen Preis von 2.500 DM kaufen könne. Der Händler sah bei diesem Preis seine Felle davon schwimmen, aber ich konnte ihn überzeugen, dass fünf mal eine gute Marge besser wäre als gar nichts. Der Händler hielt sein Wort, und zwei Wochen später brachte mir die Post die Kartons mit dem Rechner und dem Zubehör.

Eine Seriennummer hatte er keine, die Anleitung von Scala war in Italienisch und die Tastatur auf Französisch, aber das konnte ich zum Glück in deutsche Versionen tauschen. Dazu dann noch 16MB Hauptspeicher, ein CD-ROM-Laufwerk, eine Picasso II-Grafikkarte, ein A3070 Streamer und ein hochwertiger Idek Iiyama 17-Zoll-Monitor nachgerüstet. Meine persönliche Workstation war fertig! Mit NetBSD 0.9, Amiga DOS 3.1 und Macintosh System 7.5 fühlte ich mich gut für die Laborarbeit gerüstet. Ich brachte den Rechner mit in die Firma, bei der ich meine Diplomarbeit anfertigte. Die Kollegen konnten sich vor Lachen nicht mehr halten, brachte „der Deutsche“ doch einen vermeintlichen Spielcomputer ins Labor! Als ich den Rechner aufgebaut hatte und vorführen konnte verschlug es ihnen angesichts ihres eigenen Macintosh II mit einer Auflösung von 640x480 ersteinmal die Sprache. Ich konnte tatsächlich mit dem Amiga Textverarbeitung machen und gleichzeitig mit Hilfe der Software „Shapeshifter“ auf dem gleichen Computer unter dem Macintosh System mit Excel arbeiten. Mit MS-DOS, dem dritten gleichzeitig laufenden System konnte ich unsere Messgeräte kalibrieren oder auslesen.

gleichzeitig mit dem Amiga 3000 UX und dem C900 präsentierte Commodore deshalb in Partnerschaft mit Garmhausen & Partner noch eine weitere Familie von Unix-Maschinen auf Basis der „Commodore Profi Line“. Dabei gab es jeweils zwei Workstations (WSX I und WSX II) sowie zwei Server (UXS I und UXS II).

UXS II ...

Wer schon den C900 für einen unbekannteren Computer hält, der sieht sich auf der Suche nach Informationen zur Commodore Profi Line noch weit größeren Herausforderungen gegenüber. Zum Server UXS II sind wenigstens noch die Rahmenbedingungen zu finden: Als Basis wurde mit dem TW 486-25C mit einer 380 MByte SCSI-Festplatte, einem Streamer, zwei Diskettenlaufwerken, 16 MByte Haupt- und 32 kByte Cache-Speicher sowie sage und schreibe 16 serielle Schnittstellen auf einer einzigen Karte eine leistungsfähige Grundlage geschaffen. Als Betriebssystem kam SCO-UNIX System V Release 3.2 zum Einsatz, laut Magazin "Comm" das „meist installierte UNIX-Betriebssystem auf Rechnern mit 80386- und 80486-Prozessoren“. Gedacht war der Rechner als „Abteilungs-

rechner, Datenbank- und Bürokommunikationsserver“ ("Comm"). Vertrieben wurde der Rechner nur von solchen Commodore-Fachhändlern, die Unix-Kompetenz hatten – oder dies zumindest der Zentrale gegenüber glaubhaft machen konnten.

Schon eine Ausgabe vorher hatte das Magazin „Comm“ über den Einsatz des WSX II bei der Steuerung von LKW-Flotten aus dem All berichtet, damals allerdings mit etwas anderen Rahmendaten. So sollte der WSX II nun auf dem Desktop-Rechner DT 486-25C beruhen und über 64 kByte Cache-Speicher, dafür aber nur 8 MByte Hauptspeicher und eine 200MB Festplatte verfügen. Immerhin setzte die Commodore Büromaschinen GmbH in Frankfurt im Dezember 1991 selbst einen WSX II als UUCP-Server für zwei Telefonanschlüsse ein.

...und WSX I, WSX II, UXS I

Zum WSX I und den beiden UXS I und UXS II finden sich hingegen in den Weiten des Internets keine weiteren Daten. Selbst in den Prospekten der Profi Line wird der TW 486-25C-Rechner selbst zwar genannt, allerdings nur mit MS-DOS als Betriebssystem. Unix wird hier an keiner Stelle erwähnt.

RUNNING COMMENTARY ON 3000UX PERFORMANCE

If you really want to know how the Amiga 3000UX brings UNIX System V Release 4 to the workstation, you should ask the people who use it. So we did. Listen to what Amiga 3000UX users at some of the nation's leading institutions, and a leading publication, have to say.

"The Amiga just far outperformed the others."

Last year, after years of using Apple Macintosh® systems, Virginia Tech's Computer Science Department decided to change suppliers for their student computers. The school's computer science majors are required to buy UNIX-based systems, and from now on, the Department says, they'll be buying Amigas.

The decision was not a hasty one. The department carried out an exhaustive review of systems from all of the major UNIX players — nearly 20 in all — before making a choice. And the winner was the Amiga 3000UX.

Why? Ask James Arthur, Associate Professor of Computer Sciences at Virginia Tech and Chairman of the department's PC selection committee.

The Amiga 3000UX, Arthur says, "was head and shoulders above the other machines. The Amiga just far outperformed the others. We could have been conservative and gone with the IBM® or the Epson®, or we could go out on a limb and get a lot more machine for the money."

"We have been using Amiga almost from day one."

At Westinghouse Science and Technology Center's Human Sciences Department in Pittsburgh,

the Amiga is an indispensable part of the company's advanced research efforts. Lab Manager Gary Sherwin is crystal-clear about the reasons.

"We have been using the Amiga for advanced scientific research almost from day one," says Sherwin. "I like the solidity of UNIX and the usefulness of being able to run multiple operating systems on the same machine."

"The most complete implementation of UNIX SVR4."

You may have seen Ben Smith's review of the Amiga 3000UX in the December, 1990 issue of *Byte* magazine. In his article, titled "A UNIX Graphics Workstation for the Rest of the World," Smith had this to say about the Amiga 3000UX:

"This workstation is the most



complete implementation of the new AT&T UNIX System V Release 4. ...[I]t is not a clone, nor a work-alike, nor a toy. It is a no-nonsense workstation that is impressive and compact. ...The Amiga 3000UX greatly outperforms the equivalent NeXT® and Macintosh with A/UX®... it is an obvious choice as a low-end workstation."

"As slick an implementation as any I've seen."

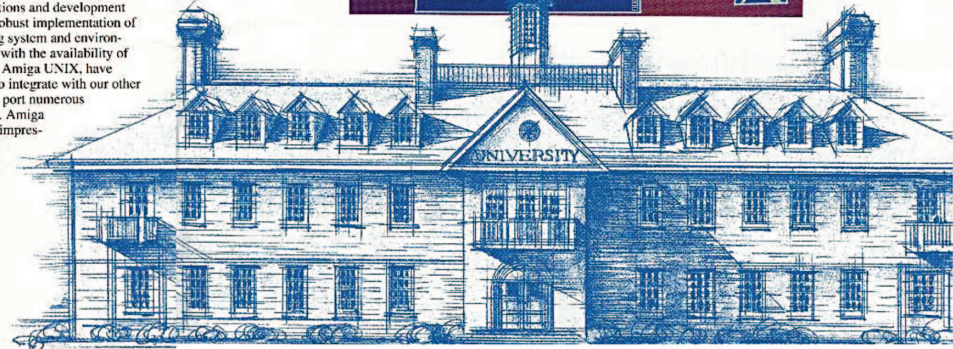
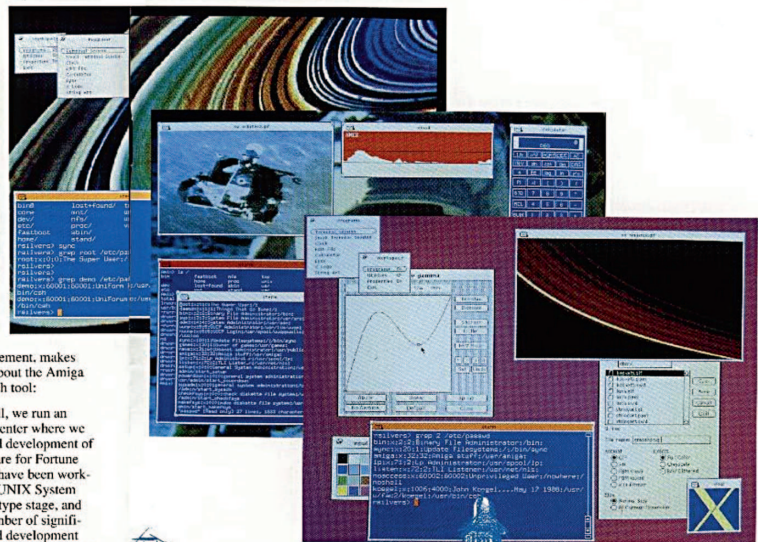
At the University of Southern California, Amiga 3000UXs are up and running after being connected with the school's existing network. How does USC Director of Systems Mark Brown rate the Amiga's connectivity? Listen:

"[It is] as slick an implementation as any I've seen," Brown says. "We took it out of the box, switched it on and connected it to our network. It worked fine."

"Amiga UNIX is an impressive implementation of System V Release 4."

Richard A. Minor, Research Manager at the University of Lowell's Center for Productivity Enhancement, makes these observations about the Amiga 3000UX as a research tool:

"Here at Lowell, we run an industry-research center where we conduct research and development of hardware and software for Fortune 500 companies. We have been working with the Amiga UNIX System since the early prototype stage, and have produced a number of significant applications and development tools. The robust implementation of the operating system and environment, along with the availability of standards in Amiga UNIX, have enabled us to integrate with our other systems and port numerous applications. Amiga UNIX is an impressive implementation of System V Release 4."



Commodore preist in dieser Broschüre vollmundig sein UNIX für AMIGA an

Andere Unixen

Ich arbeitete etwa bis Ende des Jahrtausends mit meinem Amiga und NetBSD, bis ich dann auf einen PC-Kompatiblen Rechner mit K6-II-Prozessor und Linux im Big Tower umstieg. AMIX war mir damals nur prinzipiell bekannt.

Im Rechenzentrum der Universität Karlsruhe gab es im Keller einige Tische, auf denen immer wieder Computer und Zubehör standen, die „zum Mitnehmen“ waren. So kam ich Anfang des Jahrtausends an einige Unix-Workstations und Server mit unterschiedlichen Unix-Versionen: von SUN so ziemlich alles von der SUN 3/60 über zwei Sparcstation 1 bis zur Ultra 10, von HP eine 9000/715, eine 9000/735 und eine 8000/825, dazu von DEC eine Vax, eine DECstation und eine Alpha. Ein guter Freund schenkte mir seine NeXTstation, eine SUN Javastation erwarb ich bei einer Online-Auktion. Sprich: Es hatte sich ein ganzer Zoo angesammelt, aber ausgerechnet AMIX fehlte.

AMIX, eine Beschreibung

Anfang 1988 verriet der CEO von Commodore, Irving Gould, der Presse die Pläne zum A2500UX, was die Entwickler unter

Druck setzte (siehe Brian Bagnall, „The Final Years“). Vorher gab es nur eine Machbarkeitsstudie, die erst ab diesem Zeitpunkt auf Amiga-Hardware lief, nämlich auf der neuen A2620-Beschleunigerkarte. Diese enthielt einen MC68020-Prozessor und eine MC 68851 Memory Management Unit (MMU). Als Listenpreis waren für den Rechner 4.699 \$ geplant, dazu kamen die Kosten für AMIX selbst. Eine der ersten öffentlichen Vorführungen von AMIX erfolgte im gleichen Jahr auf der Uniform Conference in Dallas.

Auf der Business Computing Show 1991 wurde das System in fast fertigem Zustand vorgeführt. Als Hardware verwendet wurde dabei ein Prototyp eines Amiga 3500, der ein Towergehäuse des PC-kompatiblen Rechners PC60-III mit einem neuen Mainboard des Amiga 3000 mit mehr Slots verband. Der Amiga 3500 wurde 1991 als Amiga 3000T (im Towergehäuse) auf den Markt gebracht, erhielt aber eine andere Front, die sich optisch mehr an das Desktopgehäuse des A3000 anlehnte. Der A3000UX – ein Amiga 3000 mit AMIX, dem A307-Streamer, einer 2410-TIGA-Grafikkarte, der A2065-Ethernetkarte, und einer Maus mit drei Tasten – wurde dann für 6.998 US-\$ angeboten, aber nur in sehr kleinen Stückzahlen verkauft. Schon am

25. Juni des gleichen Jahres stoppte Mehdi Ali, der Präsident von Commodore, das „Projekt Unix“.

Eines der Probleme war, dass AMIX ausschließlich mit der MC68851-MMU zusammenarbeitete. Diese war aber nur für den MC68020- und den MC68030-Prozessor verfügbar. Schnellere Prozessoren wie der MC68040 waren also nicht kompatibel, und eine Anpassung von AMIX hätte viel Zeit und Ressourcen gebraucht. Doch die stand Commodore zu diesem Zeitpunkt nicht mehr zur Verfügung, denn es hatte einfach zu lange gedauert, das System auf den Markt zu bringen. Ein oder zwei Jahre früher wäre der geforderte Preis noch konkurrenzfähig gewesen.

Endlich: Auftritt AMIX, erster Akt

Ich machte mich deshalb auf die Suche nach den AMIX-Installationsmedien. An das Original-Streamertape zu kommen war schon damals aussichtslos. Aber immerhin wurde ich nach langer Suche in der Newsgroup comp.unix.amiga auf eine offenbar vollständige Datensammlung von AMIX 2.1 hingewiesen. Ich konnte sie am 02. April 2002 von der Homepage www.mmhart.com/amix.htm herunterladen. Leider wollte die Installationsdiskette partout nicht

mit meinem Amiga und meinem A3070 Streamer zusammenarbeiten. Lange suchte ich erfolglos nach einer Lösung. Erst viele Jahre später stellte sich heraus, dass der veraltete SCSI-Chip in meinem Amiga das Problem war. Es gibt mehrere Revisionen des Chips von mehreren Firmen. Die spätere Version von Western Digital scheint zu funktionieren, während die an sich baugleichen Chips von AMD aufgrund fehlender Angaben zur Revision auf dem Gehäuse ein Glücksspiel sind. Da mein Streamer sich aber zwischenzeitlich mit einem zumindest für mich nicht reparablen Hardwareschaden verabschiedet hatte stellte ich meine Bemühungen im Jahr 2002 ohne Erfolg ein. Die Installationsdateien lagerte ich frustriert auf eine alte Festplatte aus, die ich im Keller lagerte.

Zwei Jahre später wollte ich endlich AMIX auf meinem Amiga installieren. Frischen Mutes machte ich mich ans Werk und suchte im Internet neue Hinweise. Da ich zuvor ja an der Installation gescheitert war, hoffte ich auf Erfolgsberichte anderer Nutzer, die ich dann nur noch nachzuvollziehen brauchte. Das einzige, was ich fand, waren scharenweise Hinweise darauf, dass die Homepage, von der ich die Installationsdateien geholt hatte, vom Netz gegangen war und die Installationsdateien nirgendwo verfügbar waren. Niemand schien sich Kopien der Daten gezogen zu haben, und so war AMIX nach landläufiger Meinung seit zwei Jahren verschollen. Es müssen noch Original-Bänder existieren, aber deren Besitzer haben wohl das Geschehen

nicht verfolgt, so dass diese Daten nicht verfügbar waren. Aber Moment, hatte ich nicht diese Festplatte in den Keller gelegt?

Auftritt AMIX, zweiter Akt

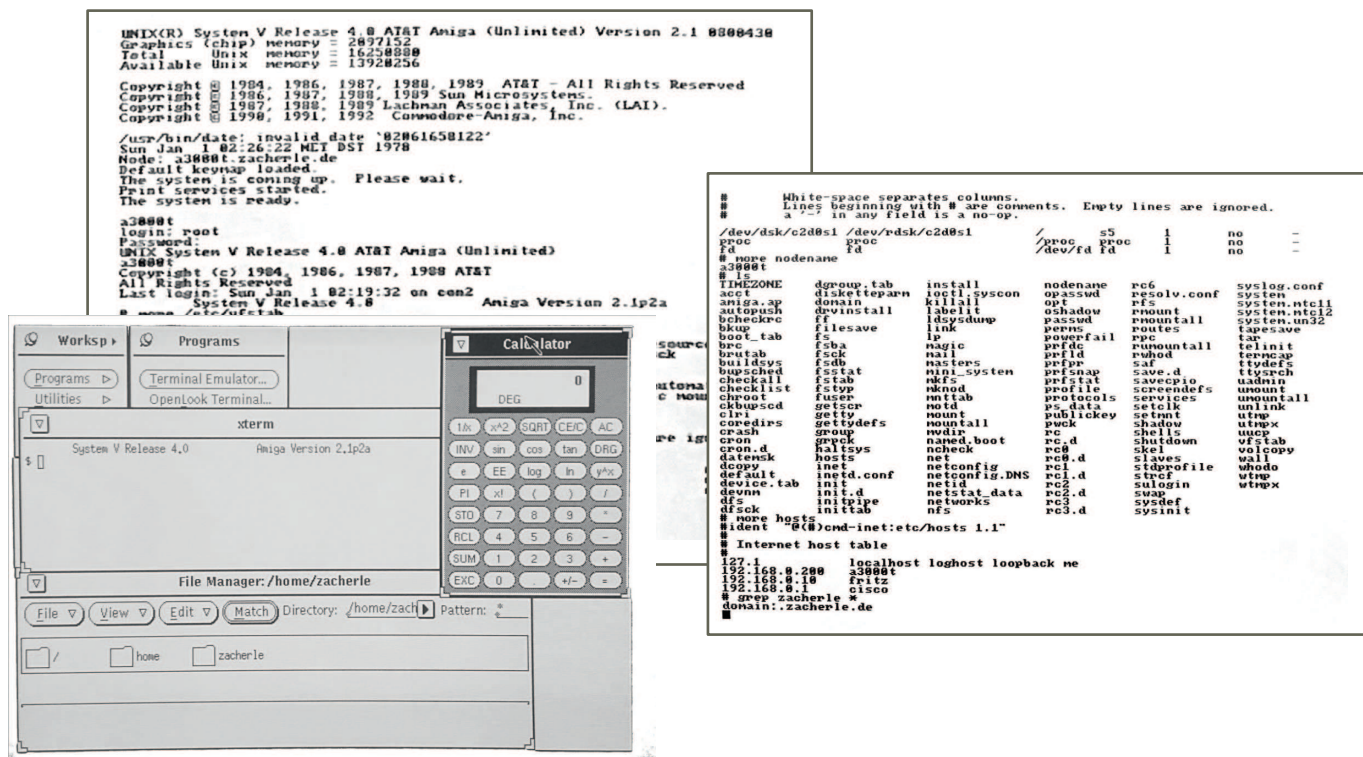
Tatsächlich konnte ich alle zur Installation benötigten Dateien von meiner Festplatte sichern. Der Kernel auf der ersten Installationsdiskette konnte booten und auch das Installationsskript auf der zweiten Diskette schien zu funktionieren – bis es auf den nicht mehr vorhandenen Streamer zugreifen wollte. Leider war es mir dann nicht möglich, einen Streamer auszuliehen. Ich musste also eine andere Lösung finden. Zuerst begann ich zum Datenaustausch SCSI-Festplatten zwischen meinem Linuxrechner und dem Amiga hin- und herzuwechseln, was mir irgendwann aber zu aufwändig wurde. Es war mir nämlich gelungen, Daten mit Hilfe des Programms dd unter Linux direkt und unter Umgehung von Partitionen auf eine externe Festplatte zu sichern. Auf dem Amiga konnte ich nach dem Abbruch des Installationsskriptes darauf zugreifen. Nur musste für jede Datei der Umbareigen ein Mal vollständig durchlaufen werden – für jeden Versuch, immer und immer wieder, bis der Datenaustausch funktionierte – und dann für jede Datei erneut.

Durch den günstigen Kauf eines zweiten ZIP-Laufwerks von Iomega hatte ich dann eine gute Möglichkeit gefunden, Daten ohne jeweiliges Herunterfahren und neu Booten zwischen beiden Systemen auszutau-

schen. So konnte ich eine Menge Zeit und Aufwand sparen. Und es gelang mir durch Inspektion des Installationskriptes und durch Ausgabe von Variablen auf dem Bildschirm herauszufinden, wann das Skript welche der 29 Dateien auf dem Tape erwartete. Auch konnte ich so ermitteln, was es dann mit der Datei tat, und auch, was es mit dem Amiga und seinen Festplatten machte. Der Rest lief dann fast wie am Schnürchen. Aber eben nur fast, denn ein etwas kurioses Problem hielt mich noch eine Weile auf. So war es notwendig, zur Installation zwei Programme auf der Konsole mit Hilfe des sogenannten Pipe-Zeichens „|“ zu verknüpfen. Dies ist unter Unix üblich und die Installationsdiskette war ja eine erste kleine Unix-Umgebung auf dem Amiga. Um Platz zu sparen, hatte ich in meinem Arbeitszimmer mehrere Systeme mit Hilfe eines KVM-Switches an nur einen Monitor, eine Tastatur und eine Maus angeschlossen. Aber ausgerechnet mein Amiga-Tastaturadapter konnte das Pipe-Zeichen nicht erzeugen. Eine Einkaufstour später war auch dieses Problem gelöst. Allerdings bekam ich AMIX nicht installiert, weil ich das Installationsskript nicht nachvollziehen konnte.

Finale

Es hatte sich ein weiterer Amiga-Nutzer in den USA gefunden, der auch unbedingt auf seinem Amiga AMIX installieren wollte. Ich schickte ihm nachts per Mail alle Fortschritte, die ich erreicht hatte, und er



AMIGA Unix in Aktion – ein Ergebnis langer Arbeit

Wikipedia zu AMIX

AMIX (Amiga UNIX) ist ein kommerzielles Unix-Betriebssystem der Firma Commodore und speziell auf die Amiga-Rechner der Firma ausgerichtet. In den 1990er-Jahren entwickelt, galt das 68k-basierende Unix-Derivat als eine der besten Implementierungen von System V R4, die seinerzeit verfügbar waren.

Das Betriebssystem gab es für den Amiga 2500 (ein Amiga 2000 mit 68020-Karte mit PMMU bzw. später 68030-CPU und A2091-SCSI-Karte) und in Folge auch für den Amiga 3000. Es wurde von Commodore als 150 MByte-Bandkassette für ein externes SCSI-Bandlaufwerk A3070 ausgeliefert.

Eingesetzt wurde das System lediglich kurze Zeit auf den 68030-basierenden Amiga 2500-UX- und Amiga-3000UX-Modellen – danach wurde die Entwicklungsmannschaft aufgelöst und die Vermarktung eingestellt. Es gab davor die Chance, über einen namhaften Workstation-Hersteller eine größere Zahl Maschinen als OEM-Ware zu verkaufen – was jedoch nie eintrat: Sun bot Commodore an, die A3000- und A4000-Reihe zusammen mit einem UNIX-Betriebssystem als Low-End-Ergänzung zu Suns Workstations zu vermarkten. Commodore lehnte dankend ab und verpasste damit nicht nur ein potentiell gutes Geschäft, sondern nach Ansicht vieler Beobachter auch die Gelegenheit, das kommerzielle Image seiner Amiga-Reihe zu heben.

schickte mir einige Stunden später seine. Am 20. Februar 2004 war es dann so weit: ich konnte meine erfolgreiche AMIX-Installation in der Newsgroup comp.unix.amiga verkünden und erste Hinweise für Nachahmer geben.

Kleinere Nachwehen

Als letzte Aufgabe blieb nur noch die Installation des Kernels und des Bootblocks. Ich löste dies, indem ich /etc/profile (das ist das Installationsskript) wiederum per Zip-Diskette in meinen Linux-Rechner kopiert und dort bearbeitete. Dies bedeutete, alles zu entfernen, was Pakete entpackt, Partitionen formatiert, prüft und installiert. Das bearbeitete Script wurde zurück kopiert und unter AMIX ausgeführt.

AMIX 2.1 lief nun, aber mit zwei Wermutstropfen: Zum einen war es nicht die letzte erhältliche Version 2.1c. Zum anderen war noch kein Treiber für meine Grafikkarte Picasso II installiert. Das erste Problem war schnell gelöst, denn es gibt eine Patchdiskette, mit deren Hilfe AMIX 2.1a auf 2.1c aktualisiert werden kann. Das zweite Problem löste ein X Server, den Klaus Burkert einige Jahre zuvor auf der Gateway Vol. 2-CD veröffentlicht hatte. Burkert arbeitete bei Village Tronic und war verantwortlich für das Hardware Design der Grafikkarte "Picasso II". AMIX war sein Hobby, und so schrieb er ein halbes Dutzend X Server für einige Grafikkarten.

Rückblick im Unmut

Welches Resümee bleibt in der Bewertung der Handelnden bei Commodore übrig? Mit zwei Unix-Varianten im Verkauf und einer kurz vor der Veröffentlichung stehenden Vorläufervariante hatte das Management alle Fäden in der Hand, um endlich wieder in die Business-Liga vorstoßen zu können. Nach dem Erfolg von VC20, C64 und dem leider weitgehend als Spielekonsole vermarkteten Amiga wäre dies eine große Chance gewesen. Dort hätten der Umsatz und die Margen für eine Stabilisierung der stark schwankenden Geschäftszahlen sorgen können. Nachträglich alles besser zu wissen ist natürlich einfach, aber drei Teams offenbar weitgehend unabhängig voneinander an einer jeweils eigenen Unix-Implementierung arbeiten zu lassen und dabei auf unterschiedliche Unix-Varianten zu setzen, erscheint nicht als kluge Geschäftsidee. Eines der Teams war so gut, dass die Firma SUN Microsystems gerne Amiga 3000UX unter dem Label „SUN“ verkaufen wollte. Doch Commodore lehnte dies sogar zwei mal ab, was rückblickend nur fassungslos machen kann.

Zum Entwicklungssystem, mit dessen Hilfe AMIX programmiert wurde, kursiert eine Geschichte ohne Quellenangabe: Der Konkursverwalter soll von Commodore nachvollziehbarerweise die Anweisung gegeben haben, aus allen im Lager herumliegenden Teilen so viele verkaufsfähige Computer zusammenzubauen wie möglich. Ein Techniker soll daraufhin die einzige Festplatte, auf der der gesamte Quellcode für AMIX 2.1 noch verfügbar war, in einen IBM-kompatiblen Computer verbaut und Windows 3.1 installiert haben. Dadurch gingen die ursprünglichen Daten natürlich für immer verloren. An Symbolkraft ist diese Geschichte natürlich kaum zu überbieten.

Leider wurden also über viele Jahre alle Chancen zu Unix bei Commodore vertan, und bis auf AMIX sind diesbezügliche Aktivitäten nur noch einem kleinen Kreis von Retro-Begeisterten bekannt. Schade, denn die spannende Geschichte hätte es eigentlich verdient, öfter erzählt zu werden.

AMIX – es gibt Hoffnung

Heute ist die Installation von AMIX zum Glück dank einiger Aktivisten viel leichter geworden, und es ist trotzdem nicht weniger faszinierend als vor zwei Jahrzehnten. Installationsdateien und die Anleitung gibt es ebenso wie gepatchte Installationsdisketten, die es ermöglichen, auch ohne den Streamer in den Genuss von AMIX zu kommen. Natürlich lässt sich AMIX auch im Emulator WinUAE installieren, Hinweise und vorbereitete Installationsdateien gibt es unter den Links am Ende des Artikels.

Aber ein echter Amiga bleibt natürlich die Königsklasse.

Für den Amiga gibt es auch Linux und NetBSD in aktuellen Versionen. Aber bei beiden Betriebssystemen ist der Amiga nur eine von vielen Möglichkeiten, es zu installieren. AMIX hingegen ist ausschließlich unter Amiga oder Amiga-Emulation lauffähig und deshalb technisch wie historisch spannender. Ich würde mich deshalb sehr freuen, wenn dieser Artikel den einen oder anderen anregen würde, sich mit AMIX zu beschäftigen und dieses spannende Betriebssystem durch die Installation auf einem Amiga ein weiteres Mal aus seinem Dornröschenschlaf zu erwecken.

Quellen

Beschreibung UXS II: Comm, das Magazin für Commodore Hardware Anwender, Ausgabe 2/1991, Seite 18

LKW-Logistik: Comm, das Magazin für Commodore Hardware Anwender, Ausgabe 1/1991, Seite 15

Links

<https://de.wikipedia.org/wiki/AMIX>

<http://www.amigaunix.de/>

<https://www.amigaunix.com/>

<https://www.amigaunix.com/doku.php/history>

<https://forum.amiga.org>

Installationsanleitung für AMIX:

<https://www.amigaunix.de/installation.html>

AMIX im Emulator:

<https://www.amigaunix.com/doku.php/installation>

<http://www.vintagebytes.de/downloads.html>

WSX II als UUCP Server:

<https://groups.google.com/g/sub.lists/c/Dq1GKndVKT4>



**AMIX und weitere Treiber
finden sich auf der Heft-CD**

Über den Autor

Michael Zacherle ist Diplomphysiker und arbeitet als Projektleiter am Karlsruher Institut fuer Technologie (KIT). Er sammelt Unix-Workstations und begeistert sich fuer rare Unix-Betriebssysteme.

HP Apollo 9000 Serie 715

Die PA RISC Story

1986 war aus der Sammlerperspektive gesehen ein bedeutsames Jahr für die Firma Hewlett Packard. Gegründet an der Westküste der USA, hatte das Unternehmen damals mit High End Geräten und dem wissenschaftlichen Gerätebau eine guten Stand. Aber HP entwickelte auch Ambitionen, in das ganz große Computergeschäft einzusteigen.



Im Rahmen des Projektes "Spectrum" setzte HP alles auf eine Karte und war zum Erfolg verdammt. 1986 erschienen die ersten vorzeigbaren Rechner – der Business Rechner HP 3000 930 und die technische Rechenmaschine HP 9000 Model 840. Beide basieren auf der neuen eigenen RISC Architektur namens PA-RISC.

Buntes Durcheinander

Bis wirklich größere Mengen bei den Kunden landen, dauerte es aber dann doch noch ein Weilchen. Aber HP hatte den Sprung geschafft. Ziel der Geräte aus dem Projekt Spectrum war nichts weniger als die Vereinheitlichung der kompletten Computerlinie bei HP. Dies sollte unter einer komplett neuen CPU passieren, aufgebaut nach dem neuen RISC Architekturkonzept, dessen Praxistauglichkeit noch nicht wirklich erwiesen war. Vorher gab es bei HP jede Menge unterschiedliche Geräte und Produktlinien mit jeweils eigenen und zueinander inkompatiblen Prozessoren. Neben den CPUs der Maschinen der HP 3000 Business Reihe gab es vor allem auf dem Motorola 68000-Chip basierte Geräte (die 200 Series) und die neue HP Vectra-Rei-

he auf Intel 80286 CPUs. Daneben existierte auch die 500 Series mit dem FOCUS Chip, ein an sich genialer Chip, aber mit schwierigen Wärmeproblemen und daher nicht (wie ursprünglich gedacht) überall nutzbar.

Dabei reicht die Idee der Entwicklung einer performanten und großen CPU bei HP bis ins Jahr 1969 zurück. Dieses Jahr setzte für sich wichtige Wegmarken: Menschen landeten erstmals auf dem Mond und dafür war Technik von HP durchaus bedeutsam. Auch die UNIX Entwicklung begann 1969 und ohne UNIX hätte der RISC Boom vermutlich nicht stattgefunden, zumindest nicht so explosiv.

1969 hatte HP mit dem Projekt Omega begonnen, eine ambitionierte 32 Bit-CPU zu bauen, diese Entwicklung aber wieder fallenlassen. Verschiedene Entwickler waren damit gar nicht einverstanden. In der Folge entstand das Projekt Alpha und als dessen Ergebnis eine 16 Bit-CPU. Im späteren Projekt FOCUS entstand bis 1982 die weltweit erste 32 Bit Single-Chip-CPU. Zu deren Entwicklung gehörte ein kompletter Chipsatz mit Memorycontroller, I/O Controller und 128 kBit großem RAM Chip.

Das PA RISC Projekt

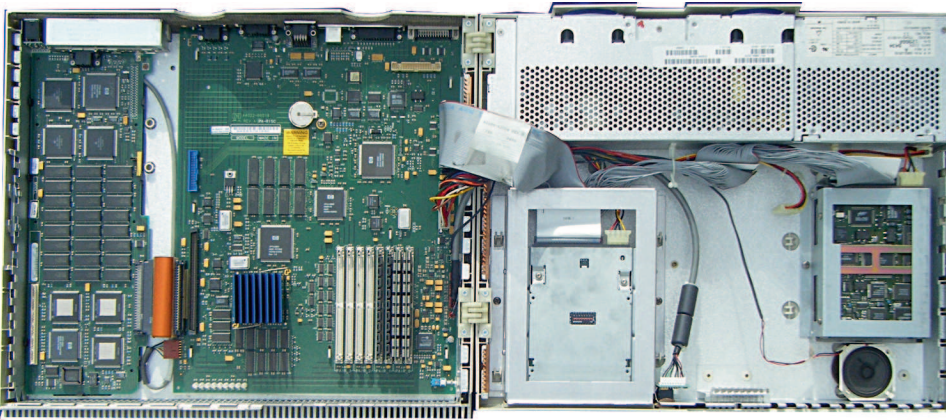
Insbesondere Joel Samuel Birnbaum, der 1980 zu HP kam und dort 1984 zum Vice President und Chef der HP Labs wurde, fördert die Idee, RISC als neue Variante überall bei HP zu benutzen. Er selbst wechselte vom IBM Forschungszentrum, wo im Projekt 801 (benannt nach dem Gebäude 801) die Idee eines RISC Computers ursprünglich entstanden war. Birnbaum beteiligte sich in seiner Zeit bei IBM auch an der Entwicklung der ersten RISC Computer des Konzerns.

Bei HP wurde zwischen 1982 und 1986 das Projekt Spectrum aufgesetzt, dessen Ziel die Vereinheitlichung sämtlicher nicht PC kompatibler Geräte auf einer gemeinsamen Basis war – der Precision Architecture, kurz PA-RISC. Dazu trugen alle Abteilungen der Firma HP bei, indem sie eine Art firmeninterne Abschlagszahlungen komplett an das RISC-Projekt abgaben. Sollte das Projekt Spectrum nicht zu einem brauchbaren Ergebnis führen, wäre HP ohne eigene CPU dagestanden und wäre finanziell ausgeblutet gewesen – ein riskantes Spiel.

Aber 1986 gelingt der Quantensprung, die erste PA-RISC-CPU ist serienreif. Ab 1987 werden zunächst die wichtigen Business Computer HP 3000 umgestellt und die HP 9000 840 mit PA-RISC-Chips gefertigt. Ab 1991 folgen dann endlich auch die neuen HP Apollo 9000 Desktop Computer mit der Bezeichnung Series 700.

Die PA RISC CPU im Detail

Die ersten Computer der Series 700 basieren auf dem PA-RISC-32 Bit Prozessoren mit 32 Registern (R0 bis R31) für Integer Zahlenwerte. Die Register sind zwar prinzipiell frei verwendbar, aber es gibt seitens HP bestimmte Zuordnungen, wofür welche Register zu benutzen sind. So werden etwa die Register R26, R25, R24, R23 zur Übergabe von Werten an Subroutinen benutzt. Ergebnisse von Funktionen werden per Definition in R28 zurückgegeben oder bei 64 Bit Ergebniswerten in R28 und R29. Für die Beachtung solcher Regelungen ist der zum System passende Compiler zuständig. Allen Geräten ist nämlich gemein, dass sie nicht mehr per Hand und in As-



Die praktische Klappschatulle – das Gehäuse der Serie 715 wird nach rechts aufgeklappt

sembler programmiert werden, sondern auf Hochsprachen und Compiler setzen. Dort werden auch Dinge optimiert, die woanders als Demo-Tricks durchgehen würden. So werden etwa Multiplikationen vom Compiler in geschickt angeordnete Folgen von Bitschiebeoperationen und Additionen aufgelöst, was schneller geht als das Multiplizieren durch die CPU.

Bei den PA-7000 sind auch noch die CPU und die FPU (mathematischer Coprozessor) zwei getrennte Chips. Die FPU ist also extern, was nach sich zieht, dass die ersten Desktop Rechner der Series 700 keine sogenannte Superskalarität beherrschen. Das betrifft die beiden kleineren 720- (50MHz) und 730-Systeme (66MHz) sowie die großen 750-Systeme (66MHz) und mit bis zu 192 MByte RAM, statt nur 64 MB). Sie können also nicht in einem Taktzyklus eine Integer-Berechnung und eine Fließkomma-Berechnung gleichzeitig ausführen. Spätere technische Desktops wie die 715-Serie lernen das aber bald.

Ein wesentliches Merkmal von PA-RISC und eine Grundidee der Architektur ist der Verzicht auf gestaffelte Caches, weil ein sehr großer Level-1 (L1) Cache zwischen CPU und RAM benutzt wird. Daher hat bereits die PA-7000 CPU einen L1 Daten-Cache von 256 KByte und einen davon getrennten Instruktioncache für die eigentlichen Befehle von ebenfalls 256 KByte. Zum Vergleich: Eine 68030 CPU hat einen L1 Cache von lediglich 256 Bytes für Daten und 256 Bytes für Instruktionen. Eine ARM3 CPU hat 4 kByte Cache ohne eine Trennung von Daten und Befehlen. Eine 80486 CPU besitzt üblicherweise 8 KByte. Dafür sind diese bei allen diesen Chips auf dem CPU integriert.

Bei HP wächst der verwaltbare externe L1 Cache bereits in der Nachfolgenera-tion auf gewaltige 2 MByte für Daten und 1 MByte für Befehle an, wenngleich dieses Maximum in den Desktop Rechnern nicht installiert wird. Bei der PA-7100-CPU wird die FPU in die CPU integriert, womit sich

die Möglichkeit ergibt, "2-Wege superskalar" zu rechnen. Im besten Fall kann die CPU also immer eine Integerberechnung und eine Floating Point Berechnung gleichzeitig abarbeiten. Außerdem steigt die Takt-rate an, 100 MHz werden möglich. Diese CPU findet sich in dem 1992 erschienenen Topmodell "HP 9000 Model 735/99" und wird später als PA-7150 noch auf 125MHz beschleunigt, dann als Modell 735/125. Der Einstandspreis lag bei schlappen 30.0000 US-Dollar, was dem Gegenwert eines ziemlich großen Luxuswagens dieser Zeit entspricht.

Die etwas langsameren CPUs finden sich dann im Model 725 wieder und in der Modellreihe 715. Dabei ist die 725 eigentlich eine 715 im Gehäuse der HP Vectra PCs, wodurch Platz für mehr Erweiterungen vorhanden ist.

Die Serie HP Apollo 9000 715

Der eigentliche Star dieser Zeit ist aber wohl die HP 9000 Model 715. Gewissermaßen die "Brot-und-Butter" Workstation mit 5.700 US-Dollar für 33 MHz, 14.000 US-Dollar für 50MHz und 17.300 US-Dollar für 75MHz. Die Geräte dieser ersten Reihe sind auch genau danach benannt: 715/33, 715/50 und 715/75. In alle Modelle der Serie 715 lassen sich 256 MB RAM einbauen, nur das ganze kleine Modell 715/33 muss mit 192 MByte auskommen. Dabei haben im Jahr 1992 sicher die wenigsten Anwender soviel Speicher installiert. Üblich waren eher 64 MByte oder 96 MByte – und das war bereits sehr viel, denn ein Standard-PC aus der Zeit hatte 4MB oder 8MB.

Die 715er sind richtig echte Desktop Computer mit ein wenig mehr Tiefe, aber eben keine Ungetüme in Maß und Gewicht und finden daher schön Platz auf dem Schreibtisch. Sie haben ein interessant gebautes Gehäuse, das sich seitlich entriegeln und dann nach oben aufklappen lässt. Ein bisschen erinnert das an einem Commodore PET, aber das HP Gehäuse öffnet

sich von links nach rechts und oben und nicht von vorne nach hinten und oben. In geöffnetem Zustand lässt sich eine weitere Festplatte nachrüsten oder eine Erweiterungskarte einbauen – etwa eine Grafikkarte, mit der dann zusammen mit der in-board Grafik zwei große Monitore angesteuert werden können. Das ist perfekt für CAD und große Datenmengen.

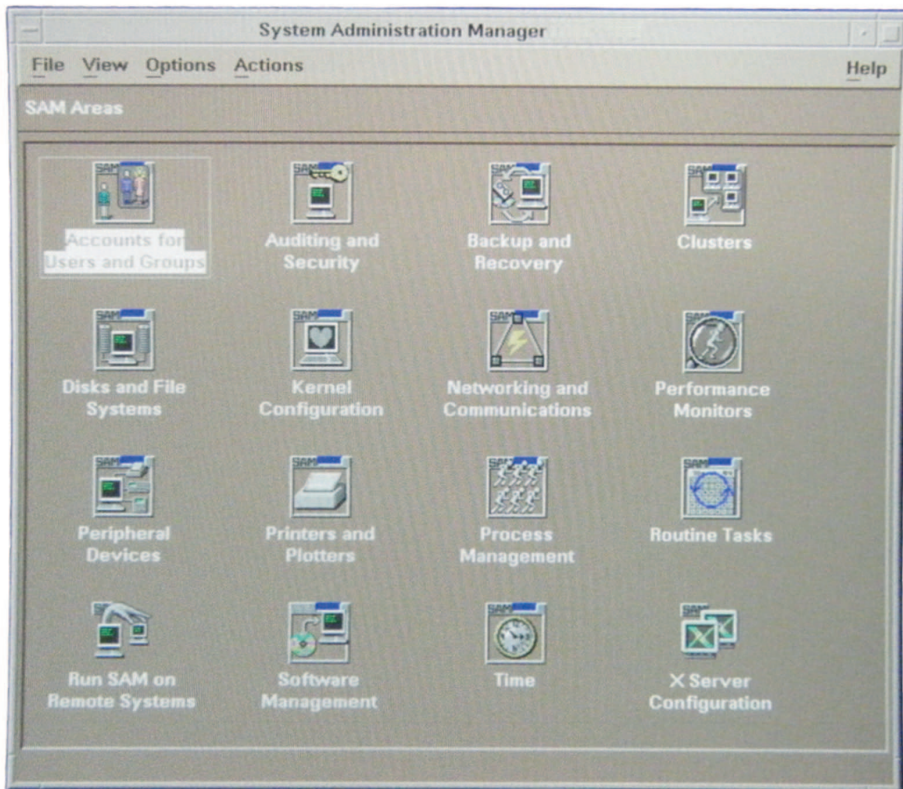
An der Frontseite findet sich ein Laufwerkslot, in den etwa ein Streamer, eine Floppy oder ein CD-ROM Laufwerk passen. Eventuell braucht es dafür ein passendes Plastikteil mit Öffnung, ein heute schwierig zu beschaffendes Teil. Ohne Laufwerk ergibt sich eine extrem elegante Gehäusefront. Da die 715 ja schließlich eine Workstation ist, fehlt auch ein externer SCSI Anschluss für externe Laufwerke nicht.

Die Rückseite hat neben einem Stromanschluss und dem SCSI Port einen parallelen und zwei serielle Anschlüsse, einen AUI Anschluss für einen Ethernet Transceiver und einen VGA Monitoranschluss für die in-board Grafik. Dankenswerterweise sind daran direkt VGA Monitore nutzbar, da die Sync Signale den üblichen Standards entsprechen. Die einzige echte Besonderheit ist der Keyboard- und Mausanschluss, denn hier kommt Hewlett Packards HIL (Human Interface Link) Schnittstelle zum Einsatz. HP HIL ähnelt dem Apple Desktop Bus (ADB) oder auch USB, ist aber natürlich zu diesen nicht kompatibel. Es braucht also ein spezielles Keyboard und eine passende Maus. Die Modelle 33, 55 und 75 können nur HP HIL, die späteren Baureihen 64, 80 und 100 besitzen anstelle der HP HIL Buchse einen SMB-10 Konnektor und führen damit PS/2 und HP HIL Protokoll heraus. Eine externe Zusatzbox dient dann als Splitter und ermöglicht es, sowohl PS/2 Tastatur und Maus als auch HP HIL Eingabegeräte wie Grafiktablets zu verwenden.

Das Betriebssystem HP-UX

Auf der Serie 715 wird üblicherweise UNIX in der HP-Variante installiert, es muss HP-UX ab Version 9.00 sein. Die erste Baureihe 715 wird von HP-UX bis zur Version 10.30 unterstützt. Alternativ geht aber auch Linux für PA-RISC, allerdings wird das heute ebenso wenig weiterentwickelt wie ein HP-UX von 1992. Außerdem lässt sich NetBSD als NetBSD/hppa verwenden und NeXTSTEP in der Version 3.3 ist für HP und Sun SPARC Maschinen benutzbar.

HP-UX ist ein bemerkenswert nutzerfreundliches System. Es kommt HP VUE und in späteren Versionen mit dem CDE Desktop daher und nutzt das Programm SAM als Oberfläche für die Administration. SAM erlaubt es, sämtliche wichtigen Din-



Mit SAM erledigt der HP-UX Administrator fast alle Aufgaben.

ge per Maus einzustellen: ob nun neue User angelegt werden müssen, ein wöchentliches Backup einzurichten ist, die Bildschirmauflösung geändert werden soll oder am Netzwerk geschraubt wird. Das geht sogar soweit, dass sich veränderbare Kernel Parameter einstellen und per Mausklick ein neuer Kernel generieren lässt. Dieser wird dann beim nächsten Booten geladen – bei Fehlern ist es gut, eine Kopie vom Original erstellt zu haben. Das System fühlt sich insgesamt auch heute noch schnell an und insbesondere auch die Textdarstellung im Editor ist bemerkenswert flott.

Die späteren Modelle

Besonders gilt das aber natürlich für die zweite Baureihe der 715 aus dem Jahr 1994. Die Modelle werden auch in der 1994er Reihe nach den Taktfrequenzen benannt, nämlich als 715/64, 715/80 und 715/100. Hierbei ändert sich am Gesamtaufbau des Gerätes nichts wesentliches, allerdings ist eine Weiterentwicklung der CPU verbaut. Das direkte kleinere Schwestermodell zur 715 ist die 712 - die "Gecko". Ihr Designmotto lautet "More Style, Less Space", denn das Gehäuse ist deutlich kleiner. Sie kennt nur PS/2 Anschlüsse für Maus und Tastatur.

Die PA-7100LC CPU ist hauptsächlich kostenoptimiert (LC steht für "low cost"), hat aber trotzdem Vorteile. Die Kostensenkung wurde nicht durch einen geringeren CPU Takt erreicht; hier sind bis zu 100 MHz

möglich. Vielmehr ist HP hier vom teuren, großen externen L1 Cache abgerückt. Stattdessen ist ein kleiner, 1 KByte großer L1 Cache nur für Instruktionen auf der CPU eingerichtet, dafür ist ein externer, mit bis zu 2 MByte auch recht großer Level-2 Cache vorhanden. Dieser ist auch noch "unified", wird also für Daten und Befehle gleichzeitig genutzt. Das spart die teuren L1 RAMs, gleichzeitig ist so die komplette Memoryverwaltung und auch der I/O Controller in die CPU verlagert. Die CPU hat außerdem ein zusätzliches Rechenwerk für Integerzahlen, also zwei Integereinheiten und ein Floating Point Rechenwerk. Immer zwei davon lassen sich parallel benutzen, also entweder zweimal Integer- oder einmal Integer plus Floating Point Rechenwerk. Die CPU arbeitet also Superskalar. Außerdem kennt die PA-7100LC-CPU eine rudimentäre Sprungvorhersage. Mit MAX-1 bietet sie zudem eine Multimedia Erweiterung, mit der SIMD und damit das Abspielen von MPEG-Videoclips möglich wird.

Fazit

Hewlett Packard hat mit den PA-RISC-CPU's und den Serien 700 sein Ziel erreicht, die eigenen Computer auf eine gemeinsame CPU Basis zu stellen. Die Geräte waren in den 1990er Jahren in Forschung und Lehre, aber auch in der IT generell weit verbreitet. HP OpenView als beliebtes Netzmanagementsystem lief ebenso auf den Maschinen wie medizinische und techni-

sche Anwendungen. Die PA-7150 als eine verbesserte, höher zu taktende PA-7100 sollte die Basis des Commodore „Hombre“ Chipsatzes werden und die wichtige Grundlage für einen AMIGA Nachfolger. Daraus ist bekanntermaßen nichts mehr geworden.

2008 endete auch die PA RISC Geschichte bei Hewlett Packard und gemeinsam mit Intel startete die Itanium-Reihe. Doch das ist eine andere Geschichte.

Links

<http://www.hpmuseum.net/>
<https://www.openpa.net/>
<http://www.robelle.com/library/smugbook/>
<http://www.parisc-linux.org/>
https://parisc.wiki.kernel.org/index.php/Main_Page
<https://www.youtube.com/watch?v=DIccm7H3OA0&t=296s>

PA RISC CPU

<http://www.hpl.hp.com/hpjournal/pdfs/IssuePDFs/1986-08.pdf>
<https://www.hpl.hp.com/hpjournal/pdfs/IssuePDFs/1987-03.pdf>
<https://www.youtube.com/watch?v=C53tGHzp1PI>

Modelle:

https://archive.org/details/hp_journal_1992-08/mode/2up
https://archive.org/details/hp_journal_1992-06/mode/2up
https://archive.org/details/hp_journal_1995-04/mode/2up
<https://www.youtube.com/watch?v=ERJPUDCBGZg>
<https://www.youtube.com/watch?v=hBn5U5xAd3Y&t=68s>

PowerShift Werbung

<https://www.youtube.com/watch?v=VLTh4uVJdul>

Ueber den Autor

Sebastian Barthel erlitt nach einer gluecklichen analogen Kindheit der digitale Fluch in Form eines Plus/4. BASIC, Assembler, Pascal und andere folgten. Das Thema blieb immer irgendwie interessant. Die Rechner aber sind momentan eher Workstations und Raspberries.

Schritt-für-Schritt Anleitung

Sun Ultra-1 neu installieren

Noch ist das Angebot an SUN-Workstations recht groß. An eine eigene Maschine zu gelangen, ist also nicht unmöglich. Was aber, wenn die Festplatte fehlt oder der Vorbesitzer sie gelöscht hat? Dann ist eine Neuinstallation des Betriebssystems unumgänglich. Wie das geht, zeigen wir am Beispiel einer SUN Ultra-1



Die Ultra-1 ist zusammen mit der Ultra-2 der Startpunkt der UltraSPARC Reihe. Sun hat diese Reihe 1995 gestartet und bis zum Firmenende im Jahre 2010 weiter gebaut. Die UltraSPARC-Prozessoren haben eine 64-Bit Architektur, sind aber abwärtskompatibel zu ihren 32-Bit Vorgängern. Sie sollten Sun wieder auf Augenhöhe mit der Konkurrenz von HP, SGI oder DEC bringen, insbesondere in Bezug auf die Rechenleistung. Die Prozessoren sind auf Multiprocessing ausgelegt. Sie ermöglichen große Server, wie das anbrechende Internet-Zeitalter sie damals forderte.

Auch die eingebaute Unterstützung für Mediendaten (VIS) geht in diese Richtung: VIS (Visual Instruction Set) ist eine Instruktionssatz-Erweiterung und wurde mit dem UltraSPARC-I-Prozessor eingeführt. Die Fließkommaeinheit (FPU) und die Grafikeinheit (GRU) sind die Einheiten (Unit) auf dem Siliziumplättchen des Prozessors

(Die), welche die Verarbeitung der jeweiligen VIS Instruktionen vornehmen. Die Prozessoren sind volle 64-Bit-CPU's und besitzen FPUs mit Parallelrechenfunktionalität (Superskalar). Sie können also mehrere Befehle aus einem Befehlsstrom gleichzeitig mit mehreren parallel arbeitenden Funktionseinheiten abarbeiten. Dank neuer Fertigung und großen externen CPU Caches (512 KByte) sind hohe Taktfrequenzen möglich. Die Sun Ultra-Maschinen können dank 8 RAM-Steckplätzen bis zu 1 GByte RAM aufnehmen, für das Jahr 1995 ein sehr großer Wert. Übrigens konnten erst mit Solaris 7 (SunOS 5.7 mit ONC und CDE) die V9 Instruction Sets der UltraSPARC's voll ausgenutzt werden. Bis dahin musste mit dem unter Solaris bekannten v8plusa Binärformat für UltraSPARC-Programme ein auf 32 Bit beschränkter Instruktionssatz der V9 Instruction Sets mit VIS 1.0 reichen.

Es existieren verschiedene Versionen der Ultra-1. Wichtige Hauptunterschiede sind

die Art der Anbindung der Grafikkarte an das Gesamtsystem und damit einhergehend unterschiedliche Taktfrequenzen von CPU und Bustakt. Bei der Grafik hat das recht deutliche Auswirkungen. Der normale SBus taktet dort mit 25 MHz, der UPABus aber 83MHz. Die beiden anderen wichtigen Unterschiede sind Fast-SCSI-2 bei den kleineren U1-Maschinen und Wide-SCSI-2 bei den größeren. Die kleinen U1 verfügen nur über Ethernetschnittstellen mit 10 MBit/s, die größeren schaffen 100 MBit/s. Die Taktfrequenzen der UltraSPARC-CPU's können 143 MHz oder 167 MHz oder 200 MHz betragen.

Aufbau der Ultra-1

Die Maschinen wirken recht spartanisch. Auf der rechten Seite finden sich CD-Laufwerk und ein normalerweise unbestückter Platz für ein Floppy Disk-Laufwerk. Es gibt aber nicht einmal eine Status-LED für die Festplatten. Zwei 80 Pin SCA-Laufwerke lassen sich einbauen, und zwar direkt in einem Einschubschacht, ähnlich eines grossen Servers. Sie werden von einem Lüfter gekühlt, der vorne links sitzt und seine Frischluft über die Frontleiste holt. Noch wichtiger ist die Lüftung quer durch das Gehäuse über zwei große Netzteil-Lüfter. Die Maschine muss also entsprechend frei aufgestellt werden, um keinen Wärmestau zu verursachen.

Auf der Rückseite sind oben zwei markierte Schrauben zu sehen. Herausgedreht ist der Deckel leicht nach oben aufzuklappen, ähnlich einer Motorhaube. Doch Vorsicht – die Plastikführungen hinten sind empfindlich, ebenso die kleinen Plastiknasen in der Front. Die Rückseite zeigt ansonsten die Anschlüsse für Parallelkabel, Tastatur, Ethernet (AUI und RJ45), einen 50-poligen SCSI-Anschluss und zwei serielle Schnittstellen. Ist Serial-A per Jumper auf RS-232 eingestellt, sind alle Boot- und Fehlermeldungen auf einem angeschlossenen Terminal zu sehen. Ebenfalls auf der Rückseite findet sich der 13W3-Anschluss der Grafikkarte.

In Betrieb nehmen

Wenn man die Geschichte nach Jahren der Nichtbenutzung anschaltet, ist im Normalfall die Batterie im NVRAM-Chip leer. Dieser führt bei Sun den eher unangenehmen Namen TOD für "Time Of Day". Abhilfe schafft ein Aufbohren des Chips und der Anschluss eines Batteriehalters. Mit etwas Glück findet sich aber auch ein neuer Chip bei einem Händler, bei dem die Batterie noch Strom hat. Sicher ist das nicht – oft sind die angebotenen Chips fast genauso alt wie der in der Sun.

Nach dem Einschalten der Maschine passiert etwa eine Minute lang scheinbar nichts. In Wirklichkeit arbeitet die Sun die POST-Tests ab. Nur sind die Ausgaben nicht auf der Grafikkarte zu sehen, sondern nur über die serielle Schnittstelle. Hängt die Maschine also nach dem Einschalten, können die Infos dort hilfreich für die Fehlersuche sein. War der Boot erfolgreich, meldet sich die Sun mit einem Logo, dem Namen der Maschine und Zufallswerten für Maschinen-ID und MAC-Adresse. Diese Angaben stehen normalerweise im NVRAM, das ja aber mangels Strom aus der Batterie seinen Inhalt verloren hat. Dann helfen unsere Tipps auf Seite 16, um die Maschine zu booten.

Dann meldet sich das OpenBoot-PROM der Sun mit einem lapidaren "OK". Von hier helfen einige Befehle weiter:

- _____ "**help**" listet die möglichen Kommandos auf
- _____ "**devalias**" zeigt die vorbelegten Namen für Geräte an, die von der Maschine vorzufinden erwartet werden. Das sind hilfreiche Abkürzungen und ersparen die Eingabe langer Devicenamen.
- _____ "**show-devs**" zeigt alle bekannte Geräte an.
- _____ "**printenv**" würde alle eingestellten Werte im NVRAM anzeigen, kann aber bei leerer Batterie nur Default-Werte zeigen. Interessant sind dabei die Einstellungen für `ttya` und `ttyb` und das eingestellte Boot-Device.
- _____ "**probe-scsi**" fragt den SCSI-Bus nach Laufwerken ab. Hier muss zumindest die vorhandene Festplatte zu sehen sein, ansonsten liegt ein Fehler bei Platte oder SCSI-Anschluss vor.

Installation

Die Installation von Solaris erfolgt von einer CD-ROM. Dies ist entweder eine Original-CD oder ein selbst gebranntes ISO-Image. Da mit dem Kauf einer originalen Sun Maschine zwischen 1982 und 2004 auch immer eine Lizenz für Solaris einherging, ist dies rechtlich unbedenklich. Die Codebasis von Solaris bis zur Version 9 ist

außerdem in weiten Teilen unter einer Open Source-Lizenz (CDDL) veröffentlicht. Vom OpenBoot-PROM initiiert das Kommando `boot cdrom` den Boot von dieser CD. Die Installation erfolgt im Grafikmodus mit einer OpenWindows-Oberfläche.

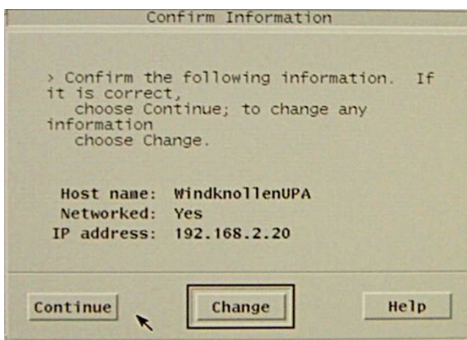
Schritt 1

Solaris nennt kurz die anstehenden Schritte und verlangt dann die Eingabe eines Hostnamens. Dieser sollte sinnvoll gewählt sein und darf im Netz kein zweites Mal vorkommen. Dann fragt der Installer, ob eine Netzwerkverbindung erfolgen soll. Die Frage wird natürlich bejaht, worauf das Programm eine IPv4-Adresse abfragt. Der Schritt wird mit einer Zusammenfassung der Einstellungen beendet.

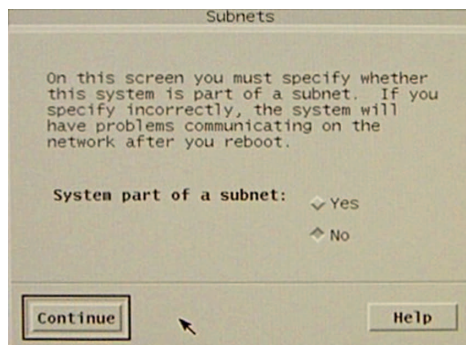
Schritt 2

Nun will der Installer wissen, welchen Namensdienst Solaris verwenden soll. NIS und NIS+ wären in einem großen Netz mit einem zentralen NIS-Authentifizierungs-server sinnvoll. Dieser wird sich Zuhause kaum finden. Damit ist "Other" die bessere Wahl; so können wir später einen DNS-Server eintragen. Die folgende Frage nach Aufstellung in einem Subnetz ist ebenfalls in einem Hausnetz ohne weitere Router zu verneinen.

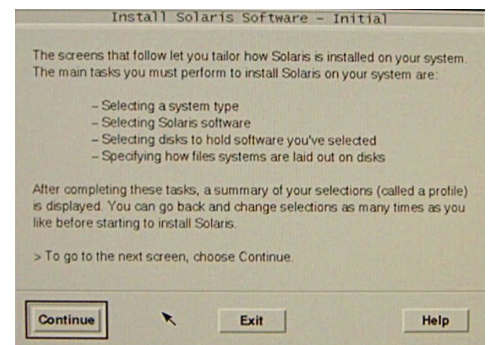
Schritt 1
Zusammenfassung der Eingaben



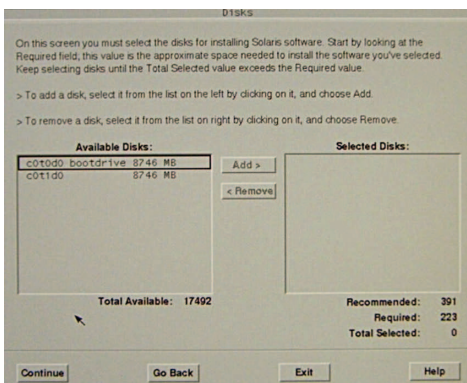
Schritt 2
Netzwerkeinstellung



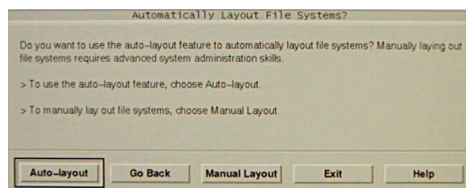
Schritt 3:
Zusammenfassung der Installationsschritte



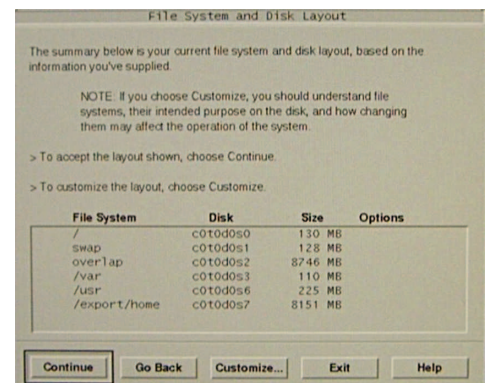
Schritt 4:
Auswahl der Festplatte für die Installation



Schritt 5
Automatisches Partitionslayout



Schritt 6:
Ergebnis der automatischen Partitionierung



Schritt 3

Wichtig ist die Auswahl der richtigen Zeitzone. Hier führt der Weg über "Geographic region" zu einer Auswahlliste, in der "Europa" und "Middle Europe" einzustellen ist. Mit der Einstellung von Uhrzeit und Datum ist auch dieser Schritt beendet.

Schritt 4

Nun geht es an die eigentliche Installation. Der Installer erfragt die Art des Systems, hier haben sich "Stand alone" und im nächsten Fenster "End User System Support" bewährt. Später lässt sich fehlende Software noch nachinstallieren. Zur Installation ist hier in der Abbildung die erste Platte im System ausgewählt, das ist das Device c0t0d0. Die Aufteilung der Platte überlassen wir dem Installer, das Auto Layout ist in den meisten Fällen passend und bedarf nur bei unverhältnismäßig großen Platten einer Anpassung. Hier beschränken wir uns darauf, drei Partitionen (hier Slices genannt) einzurichten, nämlich swap (virtuelles RAM), Root (/) und /usr (unix special resources). Der Installer erzeugt dafür die entsprechenden Partitionen. Wer nur das Partitionsschema der MSDOS- und Windows-Welt kennt, ist hier wahrscheinlich von den Bezeichnungen erst einmal verwirrt. Unter Solaris ist Slice 0 die Root-Partition und Slice 1 ist swap. Slice 2 ist ein Overlap-Slice, der die gesamte Platte umfasst und nicht direkt angesprochen wird. Slice 3, 4 und 5 sind schließlich /usr (Programme), /var (Daten) und /export/home (Benutzerverzeichnisse). Jeder Slice hat einen eindeutigen Namen, zum Beispiel c0t0d0s3 für die /usr-Partition der ersten Platte. Es ist auch möglich, die Einstellungen manuell zu verändern, worauf wir hier der Einfachheit halber aber verzichten.

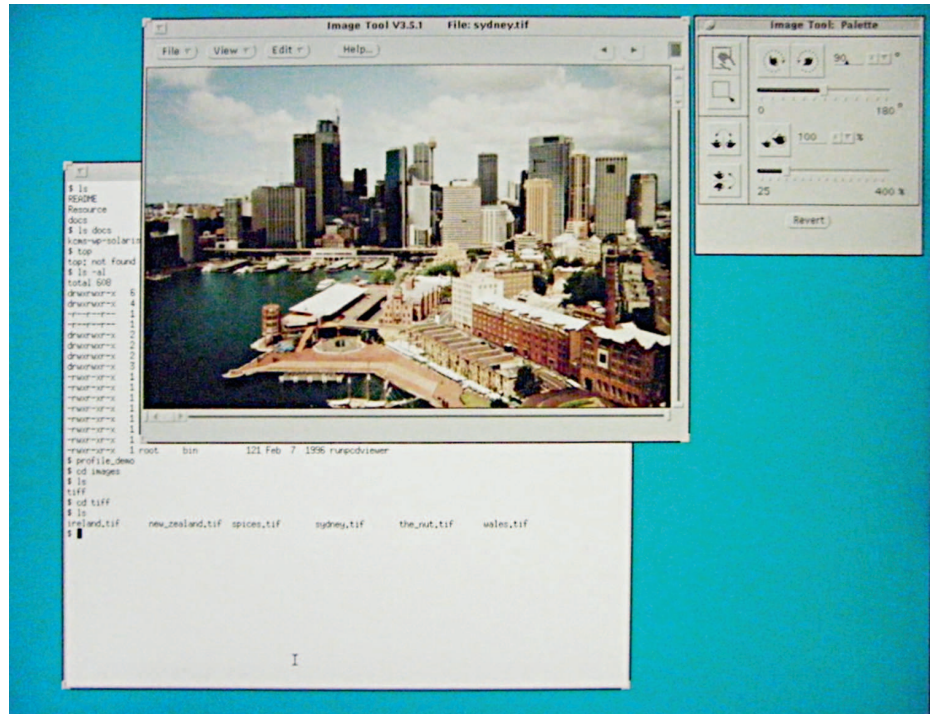
Nach einer Rückfrage, ob das System nach der Installation rebootet werden soll, startet der Installationsprozess. Die Installation dauert eine ganze Zeit. Der abschließende Reboot wird wieder auf das OpenBoot-PROM führen.

Das neue System booten

Mit dem Befehl `boot disk0` ist die Sun zu bewegen, in das neu installierte System zu booten. Die Bootmeldungen sind unspektakulär, der Solaris Kernel beschwert sich



Grafisches Login mit XDM



Der Lohn der Mühen– Solaris mit OpenWindows

lediglich (wie aufgrund der leeren Batterie zu erwarten) über einen ungültigen Code im IDprom. Am Ende des Bootvorgangs will Solaris dann ein Passwort für die Systemverwalterin "root" erfahren. Hier ist der Hinweis wichtig, dass nur die ersten 8 Zeichen des Passworts signifikant sind.

Normalerweise arbeitet der Benutzer unter Solaris mit einer nicht privilegierten Benutzererkennung und nicht als Systemverwalter. Daher gilt es nun, einen solchen Benutzer anzulegen. Mit dem Befehl `useradd -d /export/home/friedrich -k /etc/skel friedrich` werden ein Benutzer namens „friedrich“ erzeugt, sein Heimatverzeichnis angelegt und einige Konfigurationsdateien für ihn dort abgelegt. Ein Passwort besitzt der neue Benutzer nicht. Das will mit dem Befehl `passwd friedrich` angelegt werden.

Das System bootet in der hier gewählten Konfiguration immer in eine Textkonsole. Das bedeutet aber nicht, Solaris würde keine grafische Oberfläche mitbringen. Der Systemverwalter "root" muss diese nur mit dem Kommando `/usr/openwin/bin/xdm` starten. Dies ruft den XWindow Display Manager auf und dieser präsentiert ein grafisches Login-Fenster. Meldet sich der Benutzer "friedrich" dort an, landet er auf dem OpenWindows-Desktop von Solaris. Die Oberfläche bietet Dinge wie xclock, das audiotool oder das calctool. Das Starten gelingt am besten aus dem Menü heraus, das sich nach einem Linksklick auf den Desktop öffnet.

Damit ist die Sun Ultra-1 also wieder einsatzbereit. Wie Programme installiert oder

das System auf ein grafisches Login gleich nach dem Boot umgestellt wird, lässt sich leicht aus den unten angegebenen Quellen recherchieren.

Links

Allgemeine Infos:

https://en.wikipedia.org/wiki/Ultra_1

Weitere Infos:

https://en.wikipedia.org/wiki/Ultra_Port_Architecture

Die alten SunStuff Seite bei Archive.org:

<https://web.archive.org/web/20190513182807/http://sunstuff.org/hardware/systems/sun4/sun4u/ULTRA1/>

Restbestände an Doku zur Ultra1 bei Oracle:

<https://docs.oracle.com/cd/E19127-01/ultra1.ws/index.html>

Anleitung zum PROM Upgrade:

<https://docs.oracle.com/cd/E19...0-10/6jfrcj1q7/index.html>

NVRAM programmieren:

<https://www.rup12.net/posts/2021/adventures-with-Sun-ultra-1-workstation/>

Ueber den Autor

Sebastian Barthel erlitt nach einer gluecklichen analogen Kindheit der digitale Fluch in Form eines Plus/4. BASIC, Assembler, Pascal und andere folgten. Das Thema blieb immer irgendwie interessant. Die Rechner aber sind momentan eher Workstations und Raspberries.

Linux auf Sun

Das aktuelle Solaris 11 läuft nicht mehr auf alten SPARC- und UltraSPARC-Maschinen. Also begnügt sich der stolze Workstation-Besitzer mit Solaris 10 und findet sich mit dem spärlichen Angebot an legal herunterladbarer Software ab. Oder die Sun bekommt ein zeitgemäßes Betriebssystem, für das es Software in großen Mengen gibt. Schließlich sind Linux und BSD auch für Sun Maschinen verfügbar.

Das dachte sich auch der Autor und machte sich daran, seine Sun Ultra 10 neu zu installieren. Mit einer Taktfrequenz von 333MHz, 384 MByte RAM, zwei IDE-Festplatten mit je 40 GByte IDE-Festplatte, einem LSI SCSI-Controller im PCI-Slot und der Creator 3D Karte im UPA-Slot macht die Maschine auch heute noch durchaus etwas her.

Hinsichtlich einer brauchbaren Distribution ist zwar Auswahl vorhanden, aber es gibt auch Fallstricke. Sparc64 hat bei Debian GNU/Linux nur den Status "unofficial" und die Installations-CD bootet nicht ohne weiteres. Besser ist da schon Gentoo: Die Live-CD läuft auch mit 128 MByte RAM. Gentoo aber eine Source Distribution, alle Pakete kommen im Quellcode und werden erst auf der Maschine selbst kompiliert. Das bedeutet Zeitaufwand und kostet aufgrund des Leistungshungers der Sun auch einiges an Strom. Ubuntu hat die Unterstützung für Sun seit der Version 10.04 (Lucid Lynx) aufgegeben, was nunmehr schon 12 Jahre her ist.

Neben Linux unterstützen aber auch die verschiedenen BSD-Distributionen die SPARC-Architektur. OpenBSD läuft auf Anhieb auf der Ultra 10. Leider funktionierte der Treiber für die Creator 3D-Karte im Test nicht und die Smartmon Tools kamen mit den IDE-Laufwerken nicht zurecht. FreeBSD unterstützt in der Version 12 UltraSPARC als Tier 2 Architektur nicht vollständig und wird den Support in der Version 13 vollständig aufgeben. Zu guter Letzt ging ein Test mit NetBSD völlig schief, die Version lies sich nicht zum Booten bewegen.

Wirklich glücklich machen die Linux- und BSD-Distributionen also nicht. Letztlich fiel die Wahl dann doch auf Debian, denn hier wird eine komplette 64-Bit-Portierung der aktuellen Debian-Version (sparc64) vom Debian Ports Team zur Verfügung gestellt.

Die Installation beginnt mit dem Herunterladen eines ISO-Image des Debian sparc64-Installers und dem Brennen auf eine CD-R. Leider funktionierte im ersten Anlauf auch damit die Installation nicht. Das Problem ließ sich auf mangelnde Unterstützung der Grafikkarte während der Installation zurückführen. Hier hilft ein Kniff weiter: Wird die Grafik ausgeschaltet und ein Terminal an die serielle Schnittstelle angeschlossen, läuft die Installation doch. Voraussetzung sind aber mehr als 128 MByte RAM. Damit der Zugriff vom Terminal oder einem PC mit einer Terminalemulation funktioniert, sind ein paar Einstellungen im OpenBoot-PROM der Sun erforderlich:

```
setenv auto-boot? false
setenv input-device ttya
setenv output-device ttya
setenv ttya-mode 38400,8,n,1,-
reset-all
```

Jetzt sollte die an Port A angeschlossene Konsole ein Bild zeigen.

Mit *boot cdrom* lässt sich die Sun dann von der Installer-CD booten. Nach einer Weile meldet sich der normale Debian-Installer. Der Rest läuft wie bei jeder Debian-Installation ab und soll hier nicht weiter vertieft werden, da es dazu genügend Dokumentation auf den Debian-Servern gibt. Aber auf ein paar Besonderheiten gilt es zu achten:

Aus dem OpenBoot-PROM ist jede Disk als Boot Device auswählbar. Damit sind also ohne Probleme mehrere Betriebssysteme möglich, immer eines pro Festplatte. Bei der Installation darf jede vorhandene Platte ausgewählt werden. Die erste Partition der Platte wird dann zwingend als Bootpartition benötigt.

Das Partitionslayout der Platte ist unter Linux anders als bei Solaris. Die erste Partition ist die Bootpartition, sie wird mit ext2 formatiert und startet bei Block 0. Im ersten Block (512-Byte) liegt die Sun Partitionstabelle, im zweiten der GRUB-Bootblock und im dritten dann schon der Superblock des Dateisystems. Der restliche Teil von GRUB wird dann im Dateisystem eingebettet. Das hat ein paar Konsequenzen. Zunächst sollte *fdisk* die Partitionstabelle nur mit

```
fdisk -w never /dev/sdX
```

anfassern. Ansonsten löscht *fdisk* den ext2-Superblock. Das wird fairerweise auch angekündigt – stimmt der Anwender zu, ist die Partition nicht mehr zu gebrauchen und eine Neuinstallation ist fällig.

Wer GRUB neu installieren will, muss den Boot-Loader in die erste Partition installieren:

```
grub-install --skip-fs-probe --\
force /dev/sdX1
```

Außerdem sollte dazu die Partition */dev/sdX1* auf */boot* gemountet sein. Wird das nicht beachtet, schreibt GRUB in die falsche Partition und das System bootet nicht mehr. Es empfiehlt sich daher, von der Bootpartition ein Image zu machen,

```
dd if=/dev/sdX1 |
gzip > bootpartition.img.gz
```

das sich bei Bedarf wieder zurückschreiben lässt.

```
gzip -cd bootpartition.img.gz |
dd of=/dev/sdX1
```

Dafür ist es praktisch, wenn die Bootpartition wie vom Installer vorgeschlagen nur um die 500 MByte groß ist.

Den Treiber für die Creator 3D-Karte gibt es nur als Quellpaket. Von der offiziellen Debian Seite heruntergeladen, müssen die Abhängigkeiten mit apt installiert und mit

```
dpkg-buildpackage -b
```

das Paket kompiliert werden. Ein *dpkg -i* installiert das fertig kompilierte Paket, danach läuft die Creator 3D-Karte ohne weitere Konfiguration. Als Window Manager und Desktop für das X-Window System empfiehlt sich Openbox und XDM, diese laufen auch mit 384 MByte RAM ohne auf die Swap-Datei auszulagern. (r4m/gb)

Links

<https://forum.classic-computing.de/forum/index.php?thread/21778-howto-aktuelles-debian-auf-sun-ultra-10/&postID=259896#post259896>
<https://www.debian.org/ports/sparc/>

Eine Seltenheit unter den Workstations

Atari Transputer Workstation

Es gibt seltene Computer, sehr seltene und solche, die absolute Raritäten sind. Eines dieser extrem seltenen Stücke ist die Atari Transputer Workstation, kurz ATW genannt. Groß ist die Freude, wenn es gelingt, dieses seltene Stück zu erwerben.

Die Atari Transputer Workstation, auch bekannt als ABAQ, ATW-800 oder einfach ATW war ein Computer der Workstation-Klasse. Sie wurde 1989 von Atari herausgebracht und basierte auf dem INMOS-Transputer. Einige mögen sich vielleicht erinnern, dass Transputer in den späten 1980er Jahren als das nächste große Ding galten. Sie sollten das Problem lösen, wie sich die Leistung eines Computersystems steigern lässt, ohne schnellere CPUs entwickeln zu müssen. Denn das galt schon damals nur bis zu einer bestimmten Grenze als wirtschaftlich machbar. Diese Grenze schien im Jahr 2001 erreicht. Anstelle einer großen CPU sollten eine beliebige Anzahl von kleinen, billigen, aber vollständigen CPUs zusammenarbeiten, um die benötigte Leistung zu erbringen.

Das Prinzip klingt vertraut – es handelt sich im Grunde um dasselbe Konzept wie bei den heutigen Multicore-Rechnern. Im Unterschied zu diesen sind Transputer aber separate Chips ohne gemeinsamen Cache. Das macht die Zusammenarbeit der CPUs sehr wichtig. Daher erdachten die Entwickler schnelle Verbindungen zwischen den Transputern. Sie vermochten sogar über ein einzelnes Computersystem hinausreichen und verbanden somit mehrere Transputer zu einem kombinierten System. Die Transputer enthielten einen eingebauten RAM-Controller, wodurch RAM leicht hinzugefügt werden konnte.

Transputer waren das Produkt eines einzigen britischen Unternehmens: Die Firma Inmos brachte 1985 den ersten Transputer heraus. Die Transputersysteme konnten sich aber nicht gegen ihre traditionelleren Konkurrenten durchsetzen. 1989 wurde Inmos an SGS Thomson verkauft. Danach wurden die Transputer im Grunde genommen nicht mehr hergestellt. Inmos hat während seines Bestehens eine Reihe von



Transputer-CPU-Modellen entwickelt. Eine Übersicht ist der Tabelle zu entnehmen.

Atari goes Workstation

Der ATW und sein Betriebssystem HeliOS wurden von Perihelion entwickelt. Diese Firma war eine Gründung ehemaliger Mitarbeiter von MetaComCo Ltd. aus Bristol, England. MetaComCo entwickelte unter anderem mit TRIPOS ein Betriebssystem, das später die Basis für das AmigaOS werden sollte. MetaComCo hatte gute Verbindungen sowohl zu Atari als auch zu Commodore. Perihelion versuch-

te daher, beide Unternehmen für die Herausgabe einer Transputer-Workstation mit HeliOS zu interessieren. Commodore hatte ein gewisses Interesse an ihrem neuen System bekundet. Das Unternehmen zeigte Demos davon auf einer Zusatzkarte, die in einem Amiga 2000 lief. Später hat Commodore anscheinend das Interesse daran verloren. Zu diesem Zeitpunkt traf sich Atari mit Perihelion und die Arbeit begann an dem, was schließlich der ATW werden sollte.

Der Rechner wurde erstmals auf der COMDEX im November 1987 unter dem Namen ABAQ vorgestellt. Damals wurden zwei Versionen gezeigt: Eine war eine Karte, die an den MegaST Bus angeschlossen wurde. Die zweite Version war ein eigenständiges Tower-System, das einen miniaturisierten MegaST enthielt. Die Version externe Karte wurde irgendwann während der Entwicklung fallen gelassen. Perihelion erfuhr später, dass der Name "ABAQ" in Europa bereits für andere Produkte gebräuchlich war, und so wurde der Produktname in ATW-800 geändert.

Die Hardware

Das ATW-System steckt in einem großen Tower-Gehäuse und besteht aus drei Hauptbestandteilen:

- _____ der Hauptplatine mit einem T800-20 Transputer und 4 MByte RAM, erweiterbar auf 16 MB, einem kompletten miniaturisierten MegaST ST als E/A-Prozessor mit 512 kByte RAM
- _____ das Blossom-Videosystem mit 1 MByte Dual-Ported-RAM.
- _____ Alle diese Teile werden über die 20 Mbit/s-Prozessorverbindungen des Transputers verbunden.

Die Hauptplatine enthält auch drei Steckplätze für zusätzliche sogenannte Farmkarten mit jeweils vier Transputern. Damit besteht ein voll ausgebautes ATW aus 13 Transputern. Der Bus ist auch extern verfügbar, mehrere ATWs lassen sich zu einer großen Farm verbinden. Die Hauptplatine besitzt einen separaten Steckplatz für ei-

nen der INMOS-Crossbar-Switches, um die Leistung des Netzwerks zwischen den Chips zu verbessern.

Das Betriebssystem

Ein Computer ohne Betriebssystem ist gänzlich nutzlos. Die ATW-800 nutzt als Betriebssystem eine besondere Entwicklung von Perihelion namens HeliOS. HeliOS ist Unix-ähnlich, aber nicht Unix – so fehlt unter anderem ein Speicherschutz. Dies ist eine Konsequenz aus dem Fehlen einer MMU auf dem Transputer selbst. Die Konsequenzen sind nicht ganz so problematisch wie zu vermuten wäre. Anwendungen unter HeliOS laufen durchaus stabil. Die Stack-basierte Architektur des Transputers macht eine MMU nämlich weniger wichtig. Andererseits ist HeliOS Unix-ähnlich genug, um Standard-Unix-Dienstprogramme auszuführen, einschließlich des X Window Systems für eine grafische Benutzeroberfläche (GUI). Außerdem lief HeliOS auf allen Transputern in einer Farm "zur gleichen Zeit", so dass alle Rechenaufgaben vollständig verteilt werden konnten. Das Ausschalten eines ATW hatte keine Auswirkungen auf die gesamte Farm, die Aufgaben wurden einfach auf andere Prozessoren anderer Systemen verlagert. Später wurde HeliOS auf andere Prozessoren portiert, einschließlich der ARM-Architektur.

Video

Das Blossom-Videosystem wurde speziell für den ATW entwickelt. Es bietet 4 verschiedene Videomodi mit bis zu 1280 x 960 Pixeln bei 16 von 4096 Farben. Das Blossom umfasst auch eine Reihe von Hochgeschwindigkeitseffekten (128 Megapixel/s Füllrate) und Blitter-Funktionalität. Dies schließt auch die Möglichkeit ein, bis zu vier Masken auf eine Bit-Blit-Operation anzuwenden. Dies ähnelt der Fähigkeit eines modernen Grafikprozessors, mehrere Texturen auf ein 3D-Objekt anzuwenden. Das für die Blossom zuständige Team arbeitete später an einem anderen Atari-Projekt, der Videospielekonsole Atari Jaguar.

Preis und Leistung

5.000 GBP im Jahr 1990 entsprachen damals etwa 13.700 DM oder 8000 US-\$. Das entspricht heute etwa 9.900 GBP oder 14.000 US-\$. Zum Vergleich: Ein Atari TT kostete 1990 auch bereits 3.000 US-\$. Wie schneidet die ATW-800 nun im Vergleich zu anderen Atari-Computern in Bezug auf die Geschwindigkeit ab? Die Tabelle hierzu vergleicht die MIPS-Leistung mit anderen Atari-Computern. Es ließe sich argumentieren, dass der DSP im Falcon starke 16 MIPS Leistung bringt und somit CPU und DSP zusammen mit 20 MIPS mehr Power bringen als die kombinierten 11 MIPS des ATW. Aber erstens ist ein DSP kein Allzweckprozessor, diese Leistung steht nicht für jedes Programm zur Verfügung. Zweitens lassen sich bis zu 12 weitere T800-20 in einen ATW einbauen – die Performance läge also deutlich über allen anderen Atari-Systemen. Das macht die ATW-800 zum schnellsten Computer von Atari, wenngleich nicht zum schnellsten TOS-Rechner.

Kleinserie

Ein besonderer Markterfolg war die ATW-800-Workstation dennoch nicht. Ein Gerücht besagt, dass nur zwischen 200 und 350 ATWs gebaut wurden. Davon sollen 50 bis 100 Geräte als Prototypen bereits im Mai 1988 auf den Markt gekommen sein. Die Produktionsserie wurde erst im Mai 1989 freigegeben. Ein anderes Gerücht besagt, dass 200 ATWs an Kodak verkauft wurden. Auch an der Technischen Universität Braunschweig sowie bei Volkswagen in Wolfsburg soll die ATW genutzt worden sein, unter anderem für Fahrwerksimulationen. Auf dem Etikett an der Rückseite eines ATW steht etwas wie Serial Number: AB84A 90XXXX. Die Seriennummern, die dem Autor bekannt sind, lauten 909131 und 909215. Das Etikett bezeichnet die ATW auch als „Made in Gemany“. Das klingt ungewöhnlich, hat aber einen Grund: Die Prototypen der ATW wurden noch in England bei Perihelion gefertigt. Die Fertigung der Serienmodelle sollte dann im Atari Compu-

Die Preise der ATW in Großbritannien

Produkt	Preis in GBP	
	Liste	für Bildungseinrichtungen
ATW	5000,-	2500,-
Farm Card	2000,-	1500,-
+4MB RAM	750,-	562,-
Expansion	500,-	375,-

ter GmbH Technologiezentrum in der Julius-Konegen-Straße 24 in Braunschweig erfolgen.

Was hat nun dem ATW-800 den Markterfolg verwehrt? Für das Scheitern dieser Maschine sind mehrere Aspekte verantwortlich:

- _____ Das Gerät war viel zu teuer für den Massenmarkt. Atari scheint nicht viel Zeit und Mühe in die Unterstützung dieses Modells oder in die Entwicklung von Nachfolgern investiert zu haben. Leicht vorstellbar ist auch, dass Atari mit jedem Gerät Verluste eingefahren hat
- _____ HeliOS war eine zu exotische Umgebung
- _____ Perihelion blieb der exklusive Vertriebspartner in England und war immer ein kleines Unternehmen mit geringem Werbe- und Entwicklungsetat. Vielleicht waren auch Schwierigkeiten zwischen Perihelion und Atari nicht unschuldig an der frühen Einstellung der ATW.
- _____ Transputer als Technologie scheiterten schließlich, weil sie Probleme bei der Preisgestaltung und später bei der Leistung im Vergleich zur (traditionellen) Konkurrenz hatten
- _____ Inmos als einziger Hersteller dieser CPUs war ein zu kleines Unternehmen. Im Jahr der Veröffentlichung der ATW musste Inmos Insolvenz anmelden.

Die verschiedenen Transputer von Inmos

Modell	Taktfrequenz	Wordbreite	Hinweise
T212	17.5, 20 MHz	16 bit	
M212	17.5, 20 MHz	16 bit	Mit on-board Disk Controller
T222	20 MHz	16 bit	
T225	20 MHz	16 bit	
T414	15, 20 MHz	32 bit	
T425	20, 25, 30 Mhz	32 bit	
T400	20 Mhz	32 bit	stripped-down T425
T800	20, 25 MHz	32 bit	64 bit Fließkommaeinheit
T801	20, 25 MHz	32 bit	64 bit Fließkommaeinheit
T805	20, 25, 30 Mhz	32 bit	64 bit Fließkommaeinheit

Geschwindigkeiten im Vergleich

Modell	Takt	CPU	MIPS	Jahr
ST	8MHz	68000	1 MIPS	1985
MegaSTE	16MHz	68000	2 MIPS	1991
Falcon	16MHz	68030	3.84 MIPS (Motorola DSP: 16 MIPS)	1992
TT	32MHz	68030	8 MIPS (wahrscheinlich wegen Taktverdopplung auf 2 x 16MHz)	1990
ATW	20MHz	T800-20	10 MIPS (je T800, also 130 MIPS für 13 x T800-20)	1989

Was bleibt

Trotz des Scheiterns der Maschine für die breite Masse ist die ATW-800 ein guter Computer. Er hatte das Potenzial, in einigen Nischen wie dem wissenschaftlichen Rechnen vorteilhaft eingesetzt zu werden. Die Software dazu musste selbst geschrieben werden, für Forschung und Entwicklung kein ungewöhnlicher Vorgang. Nutzen Sie die Gelegenheit, sollte Ihnen jemals eine solche Rarität über den Weg laufen. (gb)

Links

http://en.wikipedia.org/wiki/Atari_Transputer_Workstation
<http://www.old-computers.com/museum/computer.asp?st=1&c=33>
<http://en.wikipedia.org/wiki/HeliOS>
http://www.classiccmp.org/transputer/rtu_atw800.htm
<http://www.classiccmp.org/transputer/atw800.htm>
<http://www.atarimuseum.com/computers/16bits/Transputer.html>
<http://www.atari-computermuseum.de/atw800.htm>

Danke an Fritz Hohl für die Überlassung seines Blogbeitrags zur ATW-800, der Vorlage dieses Artikel war.

Technische Daten

CPU:
Inmos T800-20 @20 MHz (10 MIPS)

RAM:
4 MByte (expandable to 16MB)

HDD:
44 MByte

Betriebssystem:
HeliOS

Grafik:
Blossom Video System mit 1 MByte Dual-ported-RAM, mit folgenden Modi

- _____ Modus 0: 1280 x 960 Pixel, 16 Farben aus einer Palette von 4096 (entspricht 16 echte Graustufen auf einem Monochrommonitor)
- _____ Modus 1: 1024 x 768 Pixel, 256 Farben aus einer Palette von 16,7 Millionen
- _____ Modus 2: 640 x 480 Pixel (2 virtuelle Bildschirme), 256 Farben aus einer Palette von 16,7 Millionen
- _____ Modus 3: 512 x 480 Pixel, 16,7 Millionen

Interfaces:
RGB Component Display Interface
Enthält einen miniaturisierten MegaST mit 512 kByte RAM und allen seinen Interfaces

Homepage Forum **134** Wissensdatenbank Kalender Mitglieder Karte Chat



Verein zum Erhalt klassischer Computer e.V.

Forum

Foren-Highlights

Besonders interessante Diskussionen auf <https://forum.classic-computing.de>

UNIX Geschichte

Foren-User ThoralAsmusen hat sich die Mühe gemacht, viele Fakten und noch mehr Links zur Entwicklungsgeschichte von UNIX zusammenzutragen. Neben einer Vorstellung der Hardware, auf der die erste UNIX Version entwickelt wurde, stellt der Beitrag auch wichtige Entwickler, Meilensteine der Entwicklung und einige Legenden rund um das Betriebssystem vor.
<https://forum.classic-computing.de/forum/index.php?thread/26082-unix-was-ist-das-woher-kommt-das/>

Junior Computer

Der Junior-Computer war ein erweiterungsfähiger Einplatinencomputer-Bausatz auf Basis des 6502-Prozessors. Entwickelt wurde der Rechner von Loys Nachtmann für die Zeitschrift Elektor, die den Bausatz in einer mehrteiligen Artikelserie ab Mai 1980 vorstellte. Forenuser 2ee hat nach 40 Jahren die Kopien dieser Artikel wiedergefunden und sich daran gemacht, den Einplatinen-Computer neu aufzubauen. Dabei hat er gleich mehrere Verbesserungen gegenüber dem Originalentwurf eingebaut.
<https://forum.classic-computing.de/forum/index.php?thread/26311-junior-computer>

Beliebte Programmiersprachen

Ausgehend von einem Videoclip bei Youtube entwickelte sich eine umfangreiche Diskussion über verschiedene Programmiersprachen, ihre Beliebtheit und die Vor- und Nachteile.
<https://forum.classic-computing.de/forum/index.php?thread/26338-beliebteste-programmiersprachen-1965-2019>

Forth

An der Programmiersprache Forth scheiden sich die Geister. Die einen lieben den kurzen und effizienten Code, die anderen lehnen Forth gerade darum ab. Eine Diskussion zu Forth auf verschiedenen Rechnern geht ausführlich auf diese Fragen ein.
<https://forum.classic-computing.de/forum/index.php?thread/26167-forth-diskussion-und-infos>

OS-9

OS-9 ist ein Unix-ähnliches Echtzeit-, Multiuser-, Multitasking-Betriebssystem, das ursprünglich in den 1980er Jahren von Microware Software für den 8-Bit-Prozessor 6809 von Motorola entwickelt wurde. Weitere Verbreitung erfuhr das System durch die Portierung auf die Motorola 68k-Architektur. Es gibt eine OS-9 Portierung auf den Atari ST, dessen Installation einige Tücken hat. Dazu hat sich eine sehr informative Diskussion entsponnen, die besonders auf das Kopieren der Diskettenimages auf echte Disketten eingeht.
<https://forum.classic-computing.de/forum/index.php?thread/26335-atari-st-os-9-68k-boot-failure>

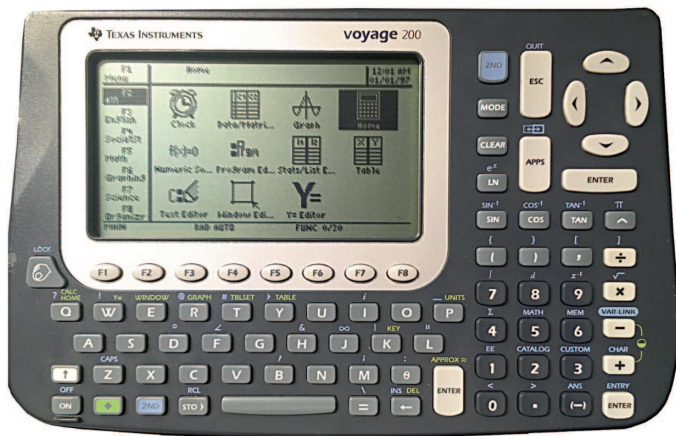
Apple I Seriennummern

Der Apple-1 war der erste Computer, den die junge Firma 1976 herstellte. Insgesamt wurden nur 300 dieser Einplatinenrechner gebaut, alle liebevoll in Handarbeit gefertigt. Auf einigen Apple-1 der ersten Serie findet sich eine handgeschriebene Seriennummer. 45 Jahre lang war nicht bekannt, wer der beiden Gründer und ihrer Mitarbeiter sie geschrieben hat – bis jetzt. Nach jahrelangem Sammeln von Schriftproben und unzähligen Befragungen brachte unser Vereinsmitglied Achim Baqué schließlich mit zwei forensische Untersuchungen durch die renommierten Authentisierungsexperen des Unternehmens Professional Sports Authenticator (PSA) endlich Licht in das Dunkel. Die Nummern zeigen eindeutig die Handschrift von Steve Jobs!
<https://forum.classic-computing.de/forum/index.php?thread/26706-autor-der-handgeschriebene-seriennummern-auf-apple-1-eindeutig-gefunden/>
<https://www.apple1registry.com/en/serial.html>



Taschenrechner von Texas Instruments

TI voyage 200



Angeregt durch die Beschäftigung mit einem Texas Instruments TI-84 Plus schaute ich mich nach weiteren günstigen Angeboten zu TI-Taschenrechner um.

Wie auf den nächsten Seiten noch ausführlich dargestellt wird, hatte der TI-84plus einige Überraschungen parat: Er beinhaltet eine Z80-CPU, lässt sich programmieren und wird dazu noch günstig auf Kleinanzeigen-Seiten angeboten. Auch der TI voyage 200 ist ein solches Schnäppchen. Er ist oft schon für etwa 15€ zu bekommen. Solange er einwandfrei funktioniert und das USB Datenkabel beiliegt, lohnt sich auch hier der Kauf.

Im TI voyage 200 werkelt eine alte bekannte CPU, nämlich die gute alte 32-Bit-CPU Motorola 68000. Sie treibt in anderer Bauform auch den Commodore AMIGA, den Atari ST und viele weitere klassische Computer an. Der TI voyage 200 gehört damit zur sogenannten TI-68k-Gruppe. Dazu gehören auch der TI-92(II)(Plus) und der TI-89(Titanium).

Spannend wird das kleine Gerät durch seine Programmierbarkeit in der Hochsprache C und durch eine Portierung der GNU Compiler Collection. TiGCC ist eine vollständige Entwicklungsumgebung für die Programmierung des kleinen Rechners. Damit nicht genug – mit dem TIEMU steht ein Emulator bereit, der unter Microsoft Windows läuft und das Testen während der Programmentwicklung deutlich erleichtert. Die Installation und Nutzung von TiGCC und TIEmu unter Windows werden auf den Webseiten von Andre Gassen gut beschrieben. Es muss übrigens kein natives Windows sein: Eine virtuelle Maschine mit Windows XP beispielsweise auf einem Apple Macintosh mit OS X Catalina tut es auch. Der Transfer von Programmen auf den Rechner passiert wie beim TI-84 plus mit TI-Connect und ist auf der nächsten Seite beschrieben. Als Ausgangspunkt für eigene Experimente eignen sich die

Beispielprogramme von der TiGCC-Seite. In der Sprache C zu programmieren ist komfortabel und es gibt viele kleinere C-Quellen, die sich relativ einfach portieren lassen. Dabei hilft das Tutorial von den Technoplaza-Seiten. Wem C als Programmiersprache zu langweilig ist, kann übrigens auch in Assembler arbeiten. Auch hierzu liefert Technoplaza das passende Tutorial. Und natürlich gibt es viele, fertige Programme in Assembler oder BASIC.

Und sollte einmal das Display defekt sein, so finden sich bei Youtube passende Reparaturanleitungen. Die nötigen Kenntnisse und Fertigkeiten vorausgesetzt, lässt sich auch ein kaputter TI voyage 200 wieder zum Leben erwecken.

Links:

Übersicht zur TI-68k-Gruppe:

<http://www.datamath.org/Architecture.htm#M68k-Mask>

TiGCC:

<http://tigcc.ticalc.org/>

Beispielprogramme:

<http://tigcc.ticalc.org/examples.html>

TIEMU:

http://lpg.ticalc.org/prj_tiemu/index.html

Seiten von Andre Gasser

<https://blog.andregasser.net/how-to-develop-ti-voyage-200-programs-on-windows-7-64-bit-using-tigcc/>

C-Tutorial:

<http://www.technoplaza.net/programming/>

Assembler-Tutorial:

<http://www.technoplaza.net/assembly/lesson1.php>

Basic Programme:

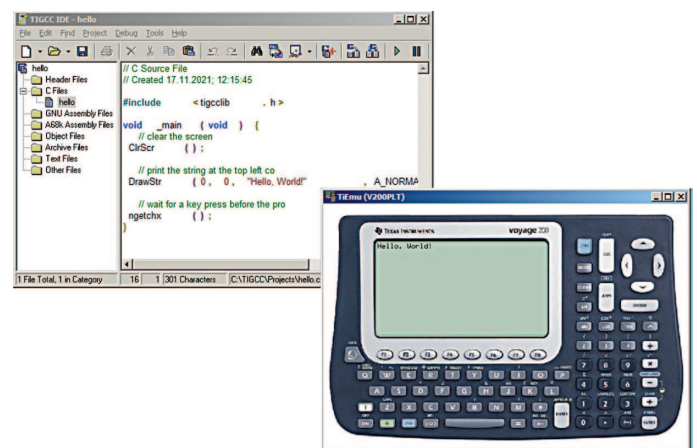
<https://www.ticalc.org/pub/v200/basic/programs/>

Assembler Programme:

<https://www.ticalc.org/pub/v200/asm/>

Display reparieren:

<https://www.youtube.com/watch?v=u74etmyXra0>



Windows XP VM mit TiGCC IDE und „Hello World“ plus TIEMU

Texas Instruments TI-84 plus

Als Sammler lohnt es sich, hier und da Gebrauchtmärkte zu besuchen. Mein TI-84 plus fand sich in einem sozialen Kaufhaus zum Spottpreis von 2 EUR. Eigentlich wollte ich keinen Taschenrechner kaufen – aber angesichts des Preises konnte ich nicht widerstehen. Die Funktionstüchtigkeit ließ sich vor Ort nicht testen, da die Batterien fehlten. So verschwand er erst einmal für einige Wochen in einer Schublade. Als dann endlich einmal vier AAA-Batterien zur Hand waren, zeigte sich das Gerät in einem ersten Test als voll funktionsfähig.

Es folgte meine Lieblingsbeschäftigung – die Suche von Informationen zum Gerät im Internet. Das Handbuch war schnell gefunden, ebenso einige andere nützliche Informationen. Zu diesen zählt der Hinweis, dass sich das Gerät mit TI-Connect mit einem PC oder Macintosh verbinden kann. Also wurde flugs TI-Connect für Macintosh installiert und siehe da: Das Gerät wurde sofort gefunden und war über ein Standard-USB-Kabel angebunden.

Die Infos aus dem Internet zeigten weitere sehr interessante Dinge. Der Taschenrechner ist als mathematischer Rechner konzipiert und verfügt über ein grafikfähiges Display mit 8 Zeilen und je 16 Zeichen oder 64 x 96 Pixeln. Dem Arbeitsspeicher von 24 KByte sind 480 KByte Flash-Speicher beigelegt, dort ist Platz für bis zu 30 Apps für Spezialfunktionen. Im Inneren des TI-84 plus werkelt die gute, alte Zilog Z80-CPU. Sie lässt sich selbst programmieren in TI-Basic und Z80 Assembler. Es gibt bereits eine Vielzahl von fertigen Programmen zum Download. Das Gerät wird heute noch als Neuware verkauft und kommt in Schulen zur Anwendung. Ein bekannter Internethändler verlangt für das Neugerät immerhin 133 EUR, in Online-Kleinanzeigen finden sich eine Vielzahl von Angeboten schon ab 15 €. Ein Schnäppchen war das kleine Ding damit aber immer noch.

Das TI-BASIC ist ein sehr spezielles BASIC, das Texas Instruments für neuere Taschenrechner verwendet und für jede

Modellreihe anpasst. Die Version des TI-84 plus kennt beispielsweise komplexe Zahlen und Matrizen als Variablentypen. Das eigentlich Wertvolle am BASIC sind die vielen, im Internet zu findenden Programme. Neben Programmen für den Mathematik-, Physik- und Biologieunterricht gibt es auch Spiele und anderes.

Ein einfaches Beispiel (smile.8xp):

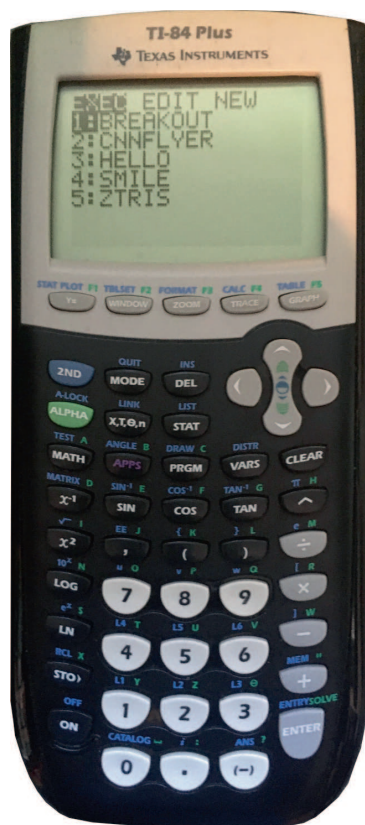
```

AxesOff
Lbl A
Circle(0,0,5)
Line(-2,-1,2,-1)
Line(-2,-1,-2.5,0)
Line(2,-1,2.5,0)
Pt-On(-2,2.5)
Pt-On(2,2.5)
Line(0,5,-1,7)
Line(0,5,2,6)
Circle(-2,2.5,1)
Circle(2,2.5,1)
Circle(0,100,1)
ClrDraw
Goto A
    
```

Dieser BASIC-Code zaubert ein Smiley auf das Display (siehe nächste Seite).

Spannender als BASIC ist natürlich die Programmierung in Z80-Assembler. Der Taschenrechner selbst kann Programme in Maschinensprache starten, aber Assembler-Quellcode nicht selbst übersetzen. Dazu braucht es einen Z80-Assembler – ich nutze *tasm* in einer virtuellen Maschine mit Windows XP und dort in der DOS Box. Die fertig übersetzten Programme wandern dann mit TI-Connect auf den angeschlossenen Taschenrechner.

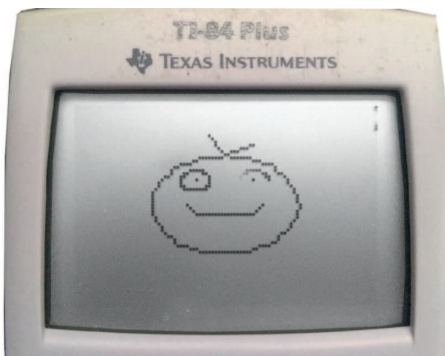
Auch für die Assemblerprogrammierung existieren zahlreiche Beispielprogramme. Um zu lernen, wie mit den spezifischen Funktionen und Includes in TI-Assembler gearbeitet wird, eignet sich das Tutorial von Sean McLaughlin. Das Tutorial richtet sich eigentlich an Nutzer des Vorgängermodells TI-83, da aber der TI-84 plus zu seinem Vater voll kompatibel ist, lassen sich die Lerninhalte fast 1:1 übertragen. Manchmal tauchen kleinere Unstimmigkeiten auf; so definiert die woanders heruntergeladene Include-Datei *ti83plus.inc* für das Beispielprogramm *Hello.Z80* den Funktionsnamen *bcall* und nicht wie im Tutorial beschrieben den Namen *b_call*.



Technische Daten

- Anzeige mit 8 Zeilen mit je 16 Zeichen, 64 x 96 Pixel monochrom
- 24 KByte RAM
- 1 MByte Flash ROM, 480 KByte verfügbar
- Z80-CPU mit 15 MHz
- Echtzeitzuhr
- USB-Schnittstelle
- Stromversorgung mit 4 x AA und Knopfzelle SR44

Die TI-84 SilverPlus Edition besitzt 1 MB Flash-ROM und ein Farbdisplay mit 65.535 Farben.



Ausgabe des BASIC-Programms smile.8xp



Ausgabe des Assembler-Programms hello.z80



MirageOS wird von manchen Spielen benötigt

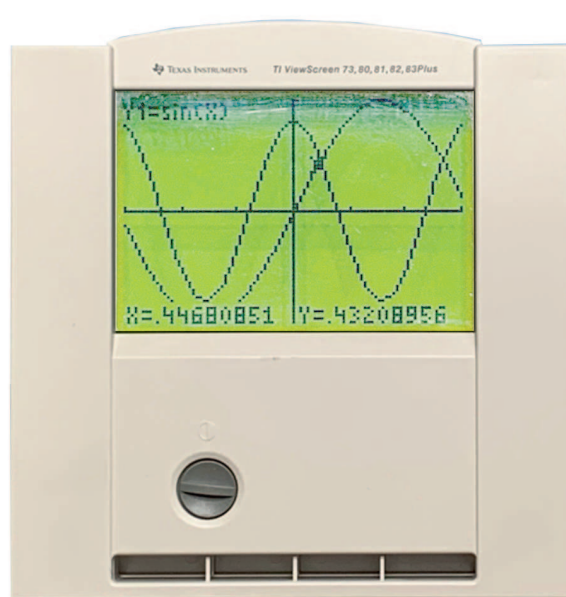
Auch hierzu ein Beispiel (hello.z80):

```
.nolist
#include "ti83plus.inc"
#define ProgStart $9D95
.list
.org ProgStart - 2
.db t2ByteTok, tAsmCmp
bcall(_ClrLCDFull)
ld hl, 0
ld (CurRow), hl
ld hl, msg
bcall(_PutS) ;Display the text
bcall(_NewLine)
bcall(_GetKey) ; Wait for a key
ret
msg:
.db "Hello world! ", 0
.end
.end
```

Assembliert wird das Programm mit der asm.bat Datei unter DOS:

```
@echo off
echo ==== Now assembling %1.z80
for the TI-83 Plus ====
tasm -80 -i -b
c:\asm\source\%1.z80
c:\asm\exec\%1.bin
if errorlevel 1 goto ERRORS
rem This is necessary because of a
DevPac8x bug
cd c:\asm\exec
c:\asm\tasm\devpac8x %1
cd c:\asm\tasm
echo ==== Job finished. Program
saved as %1.8xp ====
goto DONE
:ERRORS
echo ==== Errors!!! ====
:DONE
del c:\asm\source\%1.lst > NUL
del c:\asm\exec\%1.bin > NUL
echo ==== Done ====
```

Der Start von ASM-Programmen klappt mit der Tastenfolge 2nd – Catalog – Auswahl asm(- Enter - PRGM – Auswahl ASM – Enter. Die Ausgabe bringt wie erwartet ein "Hello world!" auf das Display.



Mit dem TI View Screen und dem TI Presenter können die Ausgaben des Taschenrechners auf einen Overhead-Projektor gebracht werden.

Es lohnt sich, die zahlreichen Beispielprogramme aus dem Internet auszuprobieren. Einige Spiele setzen ein Betriebssystem-Addon voraus, zum Beispiel das MirageOS. Es lässt sich wie andere Apps installieren und ist danach unter APPS zu finden und zu starten.

Insgesamt bietet das Gerät deutlich mehr, als am Anfang zu erwarten war. Es bietet viel Spaß und eignet sich sowohl als Taschenrechner, als kleine Spielekonsole und auch als Programmierplattform – und das alles für 2 EUR (ps).

Links

<http://www.datamath.org/Architecture.htm#Z80-Flash>
 TI-BASIC
<https://de.wikipedia.org/wiki/TI-Basic>
 TI-Basic Beispiel
https://thorben.voss.art/articles/gtr/gtr_ti-84.php
 TI-Assembler
<https://www.ticalc.org/archives/files/fileinfo/15/1504.html>

Sean McLaughlins Assembler Tutorial <https://tutorials.eeems.ca/ASMin28Days/welcome.html>

Weitere Assembler-Programme:

<https://www.ticalc.org/pub/83plus/asm/programs/>
<http://www.benryves.com/ti83plus.htm>

MirageOS:
<https://www.ticalc.org/pub/83plus/asm/games/mirageos/rate.html>



Der TI-Assembler findet sich auf der Heft-CD

Ueber den Autor

Peter Sieg ist seit 2006 Wiedereinsteiger im Retrocomputer-Hobby. Er ist Autor der Buecher "Commodore Hardware Retrocomputing" und "Simulation-Emulation Exotic Flavour" und hat zahlreiche Beitrage fuer die LOAD geliefert.

Das serielle Netzwerk von Sinclair Spectrum und QL

QLAN entschlüsselt



Sinclair Spectrum und Sinclair QL verfügen über eine einfache Möglichkeit der Vernetzung. Ein einfaches Audiokabel zwischen den Geräten genügt als Hardware. Alles andere bringen die piffigen Briten bereits von Haus aus mit.

Dieser Artikel richtet sich an alle QL-Benutzer, die ihre QLs oder kompatiblen Rechner über die eingebaute Netzwerkfunktion (QNET) miteinander verbinden möchten. Die Verbindung kann zum Austausch von Dateien dienen oder dazu, das Potenzial für Multi-User-Spiele zu erforschen. Vielleicht fasziniert aber auch schlicht nur die einfache, aber überraschend effektive Netzwerktechnologie. Der Artikel liefert auch Material, das denjenigen helfen soll, die ihren ZX Spectrum an das QL-Netzwerk anschließen wollen.

Während meiner mehrjährigen Beschäftigung mit QNET und dem zugrundeliegenden Protokoll – im Folgenden als "QLAN" bezeichnet – habe ich die Einfachheit und Eleganz des Designs schätzen gelernt. Ich möchte den Einfallsreichtum der ursprünglichen Entwickler, Sinclair und Tony Tebby, hervorheben. Ich möchte auch QView und insbesondere Lau Reeves für die akribische Kommentierung des Minerva-Quellcodes erwähnen. Diese haben mein frühes Verständnis des Netzwerktreibers unterstützt, bevor ich mich dann daran machte, den erweiterten TK2-QLAN-Quellcode zu verstehen.

Es gibt bereits einige ausgezeichnete Ressourcen zur Beschreibung von QNET im Internet, insbesondere:

- _____ Roy Wood/Q-Branch's Super-BASIC/SBASIC Reference Manual Online – Kapitel 17, ein Großteil davon wurde von Rich Mellor verfasst.
- _____ ein Artikel mit dem Titel Network von David Denham wurde ursprünglich in QL Today veröffentlicht und ist auf der Website von Dilwyn Jones verfügbar
- _____ das originale QL Benutzerhandbuch, verfügbar auf Dilwyn's Seite

Es ist unvermeidlich, dass sich in diesem Artikel bereits vorhandene Informationen teilweise wiederholen. Aber das Ziel war es, sich auf das zu konzentrieren, was noch nicht dokumentiert ist. Außerdem sollten Informationen verdeutlicht werden, die an anderer Stelle gefunden wurden. In einigen Fällen wurden Fehler in diesen Informationen auch korrigiert. Die Hoffnung ist es, etwas von meiner eigenen Faszination in diesem Bereich mitzuteilen. Mit etwas Glück wird hier etwas Nützliches präsentiert, das Sie noch nicht gelesen haben. Vielleicht fördert der Artikel die Entwicklung neuer Ideen für das QL-Netzwerk.

Der Stand der QLAN Entwicklung

Verbesserte Versionen der QLAN-Gerätetreiber wurden für verschiedene QL-Plattformen entwickelt. Sie haben auch einige bisher nicht dokumentierte Fehler im Netzwerkcode des Spectrum/Interface-1 Shadow-ROM behoben. Mit diesen verbesserten Treibern war es möglich, das Netzwerk mit dem 4,5- bis 8-fachen der ursprünglichen Bitrate mit entsprechend schneller QL-ähnlicher Hardware (QL/SGC zu Q68 bzw. Q68 zu Q68) zu betreiben. Der Autor ist gerade dabei, den grundlegenden QL-Treiber zu optimieren. Dies wird ihn auf etwa das 1,5-fache seiner ursprünglichen Geschwindigkeit bringen. Mit geeigneten Änderungen an den Interface-1-ROM-Routinen auf der Spectrum-Seite ist es endlich möglich, eine zuverlässige bidirektionale Kommunikation zwischen dem 7,5-MHz-QL und einem Spectrum zu erreichen, wenn auch mit der ursprünglichen Netzwerk-Bitrate.

Dank des klaren Designs und der umfangreichen Quellcode-Kommentare war es relativ einfach, den ursprünglichen TK2-Netzwerktreiber (für den S/GC) aus dem SMSQ/E-Quellbaum zu nehmen. Er ließ für den Q68 und seinen Hardware-Timer/Counter neu entwickeln. Dadurch wurden die meisten Abhängigkeiten von den m68k-Befehlszyklus-Timings beseitigt. Diese bestimmen das Design des ursprünglichen QLAN-Gerätetreibers und sind die Hauptursache für Kommunikationsprobleme zwischen verschiedenen QL-Maschinen.

Der ND-Q68-Treiber wurde im letzten Jahr über das Sinclair QL Forum und die Website von Dilwyn Jones der Community zur Verfügung gestellt. Es ist erfreulich zu sehen, wie die Anwender ihre Q68s im Netzwerk einsetzen. Eine aktualisierte Version des ND-Q68-Treibers wird Anfang nächsten Jahres zur Verfügung stehen. Ebenso werden die verbesserten Treiber für die Basisgeräte QL und SGC bereitgestellt werden, sobald sie für die Öffentlichkeit verfügbar sind.

Falls jemand an den Verbesserungen der Spectrum Interface-1-Netzwerkrouninen interessiert ist, könnten diese ebenfalls zur Verfügung gestellt werden. Es ist aber für die meisten Benutzer kein trivialer Prozess, das Interface-1 ShadowROM zu aktualisieren.

Mein aktuelles und langjähriges Projekt ist der QLAN-zu-USB-Bridge-'QLUB'-Adapter. Mit diesem kann ein emulierter QL auf einem PC an das systemeigene QL-Netzwerk angeschlossen werden. Dazu wird eine eigene Firmware verwendet, die auf einem preiswerten Mikrocontroller läuft. Sie wird über den USB-Anschluss angeschlossen. In Verbindung mit einer neuen QDOS-Software und einem leicht modifizierten QLAN-Gerätetreiber erleichtert sie den ansonsten mühsamen Datentransfer zwischen den beiden Plattformen deutlich.

Einführung in das QL Netzwerk

Es wird empfohlen, diesen Abschnitt in Verbindung mit dem SuperBasic/SBASIC-Referenzhandbuch zu lesen. Ich habe bewusst Details weggelassen, die dort bereits gut dokumentiert sind. Nur diejenigen Passagen wurden übernommen, die für den Kontext von Ergänzungen zum Thema erforderlich sind. Es wurden auch einige kleine Fehler korrigiert, die in dieser ansonsten sehr umfassenden Quelle gefunden wurden.

QNET kompatible Systeme

Zusätzlich zum Basis-QL, dem Aurora-QL-Mainboard-Ersatz, der QXL-PC-ISA-Karte und der neueren Q68-FPGA-basierten Maschine enthalten alle die meiste oder die gesamte QNET-Hardware, die für die Verbindung benötigt wird. Die Q68 erfordert einige zusätzliche Komponenten, die nachgerüstet werden müssen. Ein QL, der mit einer Gold- oder SuperGoldCard ausgestattet ist, läuft natürlich bereits sehr gut mit QNET. Alle erforderlichen Protokoll-Timings werden automatisch in der Software angepasst, um ihren schnelleren CPUs gerecht zu werden. Die Thor-Maschinen verfügen auch über die erforderliche QNET-

Hardware und -Software.

Zusätzlich zu QL-kompatiblen Rechnern wird funktionsgleiche Hardware und das gleiche Basisprotokoll vom ZX Spectrum verwendet, wenn das Interface-1 eingebaut ist. Aber aufgrund einiger subtiler Fehler im Interface-1 Shadow ROM kann ein zuverlässiger Empfang am QL vom Spectrum aus nur mit einer aktualisierten Interface-1-Firmware in Verbindung mit einem leicht optimierten QL NET-Treiber erfolgen. Das ist wirklich schade, denn viele Benutzer sind vom Spectrum auf den QL umgestiegen. Sie hätten von der Verbindung der beiden ansonsten inkompatiblen Geräte profitieren können, ohne auf die langsameren seriellen Schnittstellen zurückgreifen zu müssen.

Andere potenzielle Kandidaten für eine Verbindung mit QNET sind die Q40/Q60 und SAM Coupe. Aber diese Maschinen würden zusätzliche, wenngleich relativ einfache Hardware-Schnittstellen und die Entwicklung einer geeigneten Software erfordern.

Typische Anwendungen für QNET

Einfache Dateiübertragungen zwischen zwei oder mehreren Stationen sind mit dem QL-Basisnetz ohne die Erweiterungen von TK2 problemlos möglich. Auch ohne TK2 kann QNET sinnvoll eingesetzt werden, beispielsweise für

- _____ Boot-strapping": Übertragung von S/Basic-Erweiterungen und -Anwendungen auf einen nicht erweiterten QL beim Start.
- _____ Rettung laufender Arbeiten: Speichern von Dateien im Arbeitsspeicher auf einem anderen funktionierenden QL, wenn Sie die lokalen Mikrolaufwerke nicht zur Mitarbeit bewegen können.
- _____ "Spooling" von Druckdateien: Verwendung von COPY auf einem Gerät mit angeschlossenem Drucker, um eine Hardcopy zu erhalten.
- _____ Multiplayer-Gaming: Streaming von Spieldaten in Echtzeit zwischen zwei oder mehr Rechnern und Spielern.
- _____ Verknüpfung mit dem ZX Spectrum: Während Code- und BASIC-Dateien nicht direkt zwischen QL und Spectrum kompatibel sind, kann sich die Erhaltung von Spectrum-Dateien auf der zuverlässigeren QL-Hardware als sehr nützlich erweisen.
- _____ Projekte mit elektronischen Schnittstellen: Mit einem entsprechend programmierten Mikrocontroller, der QLAN "spricht", ist es möglich,



3,5mm Klinenstecker Mono



3,5mm Klinenstecker Stereo

Beide Stecker sind als Netzverbinder für QLAN geeignet.

den QL mit Ihren elektronischen Steckbrett-Projekten zu verbinden.

Sobald ein funktionierendes QNET eingerichtet ist, kann die Geschwindigkeit und die Zuverlässigkeit des Netzwerks unsere alternden MDVs tatsächlich übertreffen. Dies gilt, obwohl die rohe Bitrate des Netzwerks (ca. 89kbps) viel niedriger ist als die für Mikrolaufwerke (200kbps nom.). Das Ausführen großer Programme über das Netzwerk kann sich oft als zuverlässiger und schneller erweisen als das Laden von einer fehlerhaften Kassette. Dort sind oft mehrere Durchläufe des Bandes erforderlich, um die Dateiblöcke korrekt zu lesen.

In Anbetracht des interaktiven Charakters des NET-Geräts ist die Verwendung der NET-Basisfunktionalität nicht besonders praktisch, aber nichtsdestotrotz effektiv. Es ist erforderlich, dass Befehle auf beiden Peer-Stationen gleichzeitig erteilt werden, bevor die Dateiübertragung beginnen kann. Wie wir später lesen werden, löst TK2 dieses Problem sehr effektiv mit seiner FSERVE-Dateiserver-Funktion.

Stationen verbinden

Jeder QL und mehrere seiner kompatiblen Hardware-Ersatzgeräte enthalten die grundlegende Hardware und Software, um die Verbindung untereinander zu erlauben. Dies ist mit einem einfachen zweiadrigen Kabel möglich, wie es für Mono-Audio-An-

wendungen verwendet wird. An beiden Enden müssen die Kabel mit einem 3,5-mm-Klinkenstecker besitzen. Es ist auch durchaus möglich, eine 3-adrige Stereoverkabelung mit dem TRS-Stecker (Tip-Ring-Sleeve) zu verwenden.

Jede Station ist mit der nächsten in einer Reihenschaltung verbunden. Dabei werden eine oder beide der beiden (identischen) NET-Buchsen verwendet, die unauffällig hinten rechts am QL liegen. Die beiden Stationen an den äußersten Enden der Kette haben nur eine Buchse, während bei jeder weiteren Station beide belegt sind.

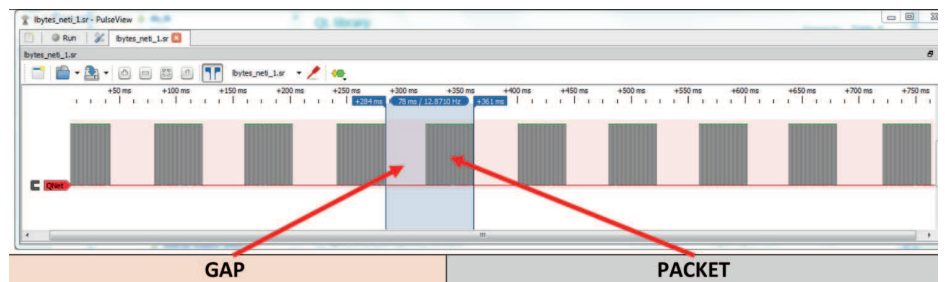
In der häufigsten Situation mit nur zwei angeschlossenen Stationen wird daher nur ein einziges 2-adriges Kabel benötigt. An jeder Station eine NET-Buchse bleibt frei. Es spielt keine Rolle, welche der beiden verfügbaren Buchsen an beiden Enden verwendet wird. Es ist Teil des QNET-Hardwaredesigns, die beiden Endstationen mit einer einzigen unbelegten Buchse zu belassen. Die Buchsen selbst sind geschaltet: Jede nicht angeschlossene Buchse besitzt einen Pull-Down Widerstand als Terminator für die NET-Leitung. Sobald ein Klinkenstecker eingesteckt ist, wird der Terminator an dieser Buchse effektiv deaktiviert. Die Bildung einer "Schleife" mit zwei miteinander verbundenen Endstationen führt zu schlechtem Empfang. Aufgrund von Signalreflexionen wird das Netz höchstwahrscheinlich unbrauchbar.

Die Einfachheit, leichte Verfügbarkeit, Flexibilität und Robustheit der erforderlichen Verkabelung ist nach Ansicht des Autors ein wesentlicher Vorteil des QNET-Konzepts gegenüber z.B. einer herkömmlichen RS-232/COM-Port-Lösung. Die maximale Gesamtlänge der Kabelstrecke wird mit 100 m angegeben (QL-Benutzerhandbuch, S.34). Der Autor hat in der Praxis mit handelsüblichen und preiswerten Mono- und Stereo-Audiokabeln erfolgreich Kabelstrecken von 30 m und mehr verwendet.

Grundlegende Netzwerknutzung

Der in QDOS eingebaute grundlegende NET-Gerätetreiber ermöglicht eine einfache Peer-to-Peer-Übertragung von Dateien. Dies passiert unter Verwendung bekannter S/Basic-Befehle wie COPY und LOAD, LBYTES, EXEC (mit ihren SAVE-Gegenstücken). Außerdem ist das Senden und Empfangen beliebiger Byteströme unter Verwendung expliziter OPEN/PRINT- und OPEN/INPUT/INKEY\$-Befehlsfolgen an den jeweiligen Enden der Verbindung möglich.

Jeder Station wird mit Hilfe des S/Basic NET-Befehls eine eindeutige "Stations-ID" zwischen 1 und 63 (oder 64, je nach Ver-



Die Signalverfolgung zeigt die Lücken in der Paketübertragung.

sion/ Alter des Treibers) zugewiesen. Eine Dateiübertragung wird durch die Eingabe gegenseitiger Befehle (Laden/Speichern) an jeder Station eingeleitet. Wenn nur zwei Maschinen miteinander verbunden sind, ist es möglich, erfolgreich zu kommunizieren, ohne die Stations-ID von ihrem Standardwert "1" zu ändern. Eine Mehrdeutigkeit wird dadurch vermieden, dass auf jeder Station zu einem bestimmten Zeitpunkt nur eine Aktivität (Senden oder Empfangen) stattfinden kann. Dennoch wird auch in dieser Situation empfohlen, eindeutige IDs zu vergeben. Neben der expliziten Angabe der Stations-ID der Gegenstelle beim Öffnen eines Kanals erkennt das NET-Gerät auch eine Stations-ID von 0 (Null) sowohl für den Eingang als auch für den Ausgang. Dies wird als "Netzwerk-Broadcast" bezeichnet. Außerdem kennt das NET Gerät die Möglichkeit "receive from any station", wobei die empfangende Station NET-Kanäle mit ihrer eigenen Stations-ID öffnet.

Das grundlegende QLAN-Protokoll erkennt End-of-File (EOF) Bedingung. Sie bestimmt, wann durch einen expliziten OPEN erstellte Bytestream-Verbindungen geschlossen werden sollen. Der NET-Treiber kennzeichnet das letzte Paket als solches, wenn der Ausgangskanal CLOSED ist. Dies kommt der "byte-seriellen" Kommunikation entgegen. Diese Methode ist jedoch unzureichend für Dateitransferverbindungen, die mit SAVE/SBYTES/SEXEC eingeleitet und mit LOAD usw. empfangen werden. Bei diesen muss die empfangende Station von vornherein die erwartete Dateilänge und den Dateityp kennen. Um dem Rechnung zu tragen, stellen die Dateiübertragungsbefehle (Speichern usw.) dem allerersten gesendeten Paket einen 15 Byte langen "einfachen Dateikopf" voran. Er gibt die Anzahl der folgenden Bytes sowie andere Metadaten an, die von den LOAD/LBYTES/EXEC-Verfahren auf der Empfängerseite erwartet und interpretiert werden. Leser, die mit dem Tape-File-Header des ZX Spectrum und dem Äquivalent für das Interface-1 vertraut sind, werden dies erkennen.

Die NET-Verknüpfung kann auch jederzeit manuell abgebrochen werden, indem

der Benutzer die Tastenkombination Strg+Leertaste drückt. Dadurch werden alle impliziten Dateiübertragungskanäle in diesem Prozess geschlossen. Eine "EOF"-Bedingung wird gekennzeichnet und kann auf der Empfangsseite erkannt werden, wenn sie auf der Absenderstation abgebrochen wird. Andererseits kann der Absender nicht wissen, ob der Benutzer an der Empfangsstation die Übertragung abgebrochen hat. Er wird stattdessen die Übertragung des letzten Pakets so lange wiederholen, bis sie selbst manuell abgebrochen wird.

Paketaufbau

Es ist interessant zu verstehen, wie das QLAN-Protokoll dafür sorgt, dass die Rohdaten über die Verbindung fließen. Ebenso ist es interessant zu wissen, wie es die Zeit für andere Aktivitäten in der Maschine verwaltet und wie es die Netzzugangszeit mit anderen angeschlossenen Stationen teilt.

Alle übertragenen Daten werden in "Blöcke" oder Pakete von bis zu 255 Byte aufgeteilt, denen ein Paketkopf vorangestellt wird, gefolgt von dem eigentlichen Datenpaket. Bei einer Datei, die aus mehreren Blöcken besteht, haben alle Blöcke bis auf den letzten die volle Länge von 255 Bytes. Der letzte Block enthält die restlichen Bytes. Der Paketkopf enthält sowohl die IDs der Quell- und Zielstation als auch andere Metadaten, die vom Protokoll verwendet werden, einschließlich der aktuellen "Blocknummer" und Prüfsummen.

Das Beispiel einer Signalverfolgung mit mehreren Paketen einer laufenden Übertragung (Bild oben) zeigt die allgemeine Nutzung der Verbindung.

Die sendende Station "lauscht" zunächst auf eine entsprechend lange Stille oder "Lücke". Sie zeigt an, dass die Verbindung frei ist, bevor sie ein "Scout"-Bitmuster auf die Leitung legt. Gleichzeitig liest sie den Verbindungsstatus zurück, um sicherzustellen, dass keine andere Station ebenfalls versucht, die Verbindung zu beanspruchen. Das Bitmuster, das für den Scout verwendet wird, wird sorgfältig berechnet, um sicherzustellen, dass verschiedene

Sendestationen ein unterschiedliches Muster erzeugen. Dies geschieht hauptsächlich auf der Grundlage ihrer eigenen, eindeutigen Stations-ID. Dies gewährleistet auch ein vorhersehbares Ergebnis – nämlich, dass alle Stationen außer der mit der niedrigsten Stations-ID den Konflikt erkennen und sich zurückziehen. Die Sendestation mit der niedrigsten Stations-ID bemerkt den Konflikt nicht einmal und fährt mit ihrer Übertragung fort.

Auf der Empfängerseite ignoriert die Empfangsstation den Verbindungsstatus vollständig, sobald der Beginn des Scouts erkannt wurde. Sie beginnt erst wieder, darauf zu achten, nachdem ein bestimmtes Zeitfenster verstrichen ist. Es dient dazu, die Scout-Phase zu überspringen, aber rechtzeitig auch, um den Beginn des darauf folgenden Paketkopfes zu erfassen. Nach der Übertragung des Headers wartet die sendende Station auf ein aktives "Acknowledge"-Signal (ein einzelnes Byte: 0x01), bevor sie mit der Übertragung des eigentlichen Datenpakets beginnt. Dann wartet sie wiederum auf eine abschließende Bestätigung, um festzustellen, ob das Datenpaket erfolgreich empfangen wurde. Bei Erfolg wird die "Blocknummer" an jedem Ende inkrementiert, so dass das nächste zu übertragende Paket bereit ist. Ein vertiefender Blick in den Signalverlauf während der Paketübertragung zeigt das Bild unten.

Wenn die Lücke nicht erkannt wird, weil die Verbindung bereits in Gebrauch ist, bricht die sendende Station den aktuellen Übertragungszyklus ab. Sie verlässt sich dann darauf, dass der QDOS IOSS-Scheduler die Übertragung später wieder aufnimmt. Das passiert auch, wenn beim Senden des Scouts ein Konflikt festgestellt wird, weil eine andere Station gleichzeitig versucht, die Verbindung zu beanspruchen.

Ebenso verursacht ein nicht empfangenes Antwort-Byte – weil der Empfänger nicht zuhört oder das Paket aus irgendeinem Grund abgelehnt – den Abbruch des aktuellen Übertragungszyklus. In diesem Fall wird die aktuelle Blocknummer beibehalten.

Die empfangende Station fragt die Verbindung regelmäßig ab. Sie bricht den aktuellen Abfragezyklus stillschweigend ab und sendet somit kein Antwortbyte, wenn _____ sie den Scout nicht innerhalb eines definierten Zeitfensters erkennt _____ sie feststellt, dass das Paket nicht für diese Station bestimmt ist oder von ihr erwartet wird _____ oder sie eine beschädigte Prüfsumme im Header oder im Datenpaket berechnet.

Dann verlässt sich wie bei der Übertragung auf den QDOS IOSS-Scheduler, um den Paketempfang zu einem späteren Zeitpunkt erneut zu aktivieren.

Der Empfänger passt seine "erwartete" Blocknummer in ähnlicher Weise wie der Sender an, je nach Erfolg oder Misserfolg. Auf diese Weise bleiben Sender und Empfänger synchron, indem sie sich auf die inkrementelle Blocknummer (0-65535) beziehen. Dadurch können alle wiederholten Pakete (Block N-1) erkannt und beim Empfänger verworfen werden, während er weiterhin auf das erwartete Paket (Block N) wartet. Eine solche Situation kann entstehen, wenn die sendende Station das als Antwort auf das letzte gesendete Paket gesendete Acknowledge-Byte verpasst. Damit ein Wiederholungspaket auf diese Weise verworfen werden kann, muss der Empfänger es noch bestätigen. Damit weiß der Sender, dass er mit dem nächsten Block weitermachen kann. Beim Spectrum und Interface-1 ist genau dies fehlerhaft implementiert – doch dazu später mehr.

Netzwerk-"Broadcast"-Übertragungen sind im Wesentlichen gleich, abgesehen von der Art und Weise, wie Bestätigungen gehandhabt werden und wie lang die Scout-Phase ist. Bei einem QL ohne TK2 wird kein Quittungs-Handshaking verwendet, was zu unzuverlässigen Übertragungen führt. Mit TK2 wird ein aktives ACK/negatives ACK zum Protokoll hinzugefügt. Es wird erst nach dem Ende des Datenpakets aktiv und verbessert die Zuverlässigkeit der Übertragung erheblich. Beachten Sie jedoch, dass dieser Aspekt nicht vollständig mit Nicht-TK2- oder Spectrum Broadcasts kompatibel ist.

Die Lücken beachten

Zwischen jedem Versuch, ein Paket zu übertragen, sind Lücken zu sehen. Sie erscheinen sowohl zwischen erfolgreichen als auch zwischen erfolglosen Übertragungen. Dabei ist die Länge der Lücke zwischen fehlgeschlagenen Versuchen normalerweise kürzer. Diese Übertragungslücken sind integraler Bestandteil des QLAN-Protokolls. Sie geben dem Empfänger Zeit, das letzte Paket zu verarbeiten, Auch geben sie dem Sender die Zeit, das nächste Paket vorzubereiten. Außerdem wird so die Leitung lange genug freigegeben, dass ein anderes Stationspaar die Verbindung während der "Pause" in Anspruch nehmen kann. Die Verbindung können sich also mehrere Stationspaare in Form eines "Timesharings" teilen. Dieser Ansatz zur zeitlichen Aufteilung einer Verbindung ähnelt dem, was in der herkömmlichen Netzterminologie als "Carrier Sense Multiple Access with Collision Detection" oder "CSMA/CD" bezeichnet wird.

Um einen Hinweis auf die Auswirkungen dieser Lücken auf den effektiven Durchsatz zu erhalten, haben wir Messungen vorgenommen. Zwischen zwei 7,5 MHz-QLs ohne andere laufende Aufträge und ohne beschädigte Pakete wird eine 32 kByte QL-Anzeigedatei in ca. 10,5 Sekunden übertragen. Dabei entfällt etwa 45 % dieser Zeit auf die Lücken zwischen den eigentlichen Datenpaketen. Zwischen zwei Q68-Rechnern hingegen wird dieselbe Datei in etwa 6,3 Sekunden übertragen. Dabei entfallen nur 15 % der Gesamtzeit auf die Lücken. Das liegt zum Teil auf die schnellere Verarbeitung zwischen den Paketen. Dadurch ist jede Station schneller für das nächste Paket bereit. Dies führt zu Übertragungsraten zwischen 3 kByte und 5 kByte pro Sekunde, je nach der Leistung der beteiligten Rechner und der Mindestlänge der Lücke, die jeder von ihnen aushalten kann. Auf diese Weise wird ein höherer effektiver Durchsatz mit einer verbesserten Link-Auslastung und Effizienz bei gleicher Bit-Rate erreicht.



Kommunikation zwischen Sender und Empfänger (Erklärung im Text)

Fehlersuche im QNET

Viele QL-Benutzer haben mit dem Anschluss ihrer QLS herum gespielt, um dann frustriert aufzugeben, ohne jemals die Vorteile der Vernetzung zu genießen. Hierfür gibt es mehrere mögliche Gründe. Die häufigsten sollen im Folgenden entweder hinsichtlich ihrer Ursache oder des beobachteten Verhaltens beschrieben werden. In jedem Fall werden mögliche Abhilfemaßnahmen genannt.

Unzuverlässige QL-Hardware

Bei der früheren Ausgabe-5-QL-Hauptplatine ist die ZX8302 ULA für das Netzwerk und die Mikroaufwerke an den "umkämpfen" Datenbus der Haupt-ULA des QL angeschlossen. Das führt zu unregelmäßigen Zugriffszeiten, die QNET lahmlegen können. Manchmal funktionieren sie, an anderen Tagen nicht. Das Ersetzen durch ein Issue-6/7-Motherboard oder eine krude Modifikation des Iss-5-Boards, um die späteren Versionen nachzubilden, löst dieses Problem. Der Autor hat eine solche invasive Modifikation erfolgreich durchgeführt, würde sie aber nicht empfehlen!

Verklemmte NET-Anschlüsse

Es wird gelegentlich beobachtet, dass die Spannung am NET-Anschluss auf dem einen oder anderen logischen Pegel hängen bleibt. Das weist entweder auf einen defekten PNP-Transistor im NET-Ausgangsschaltkreis oder auf einen defekten oder schlechten ZX8302 ULA hin. Ein Ein- und Ausschalten des zuständigen QL kann den feststehenden NET-Anschluss oft "befreien".

Auswechseln des Transistors

Der Austausch des TR2/ZTX-510 oder eines gleichwertig verbauten Transistors kann ebenfalls helfen. Dies gilt ebenso für eine obskure Modifikation, die einmal in einem QL-Magazin beschrieben wurde, nämlich das Einsetzen einer Diode in die Gnd-Leitung zum 5V-Linearspannungsregler. Der Autor hat diese Modifikation NICHT ausprobiert! Zum Zeitpunkt der Erstellung dieses Artikels sind Ersatzgeräte für den ZX8302 noch bei RWAPServices / Sell-MyRetro erhältlich. Der Autor empfiehlt, auf jeden Fall ein Ersatzgerät bereitzuhalten.

Korrodierte NET-Buchsen

Da sie in der Regel nicht benutzt werden, können die inneren Kontakte der NET-Buchsen angelautet oder sogar verrostet sein, was zu einer schlechten oder fehlenden Verbindung führt. Ein Versuch, die inneren Kontakte mit Isopropylalkohol und einem fusselfreien Tupfer o.ä. zu reinigen, kann helfen. In besonders schlimmen Fällen kann es nötig sein, die Buchsen abzulöten und durch eine ähnliche Sorte zu ersetzen – wenn Sie sie noch finden können. Denken Sie daran, dass die beiden Steckdosen miteinander verbunden sind, so dass sich der Zustand der einen auf die andere auswirkt; behandeln Sie sie also gemeinsam.

Die Zeit, die in Lücken verbracht wird, mag verschwenderisch erscheinen. Es ist aber zu bedenken, dass diese Lücken für ein inhärentes Multitasking-Betriebssystem wie QDOS auch Zeitscheiben darstellen, die der CPU nun zur Verfügung stehen, um sich anderen Aufgaben und Arbeiten zu widmen. Diese wollen ausgeführt werden und halten ein höheres Maß an Reaktionsfähigkeit für den Benutzer aufrecht. Das Spectrum ROM hingegen, das von Natur aus Single Tasking ist, verwendet in der

Fehlerhafte Kabel

Wenn Sie ein 2-adriges Kabel mit Klinkensteckern selbst hergestellt oder ein altes Kabel wiederverwendet haben, kann der Fehler in der Verbindung im gelöteten Stecker oder im Kabel selbst liegen. Prüfen Sie die Durchgängigkeit und stellen Sie sicher, dass die jeweiligen Pole des Klinkensteckers mit ihren Gegenstücken an beiden Enden verbunden sind (Spitze an Spitze, Hülse an Hülse).

Falsche Verkabelung

Bei mehr als ein paar miteinander verbundenen Maschinen kann man die Verkabelung erstaunlich leicht durcheinander bringen. Dadurch ist die eine oder andere Station nicht mehr angeschlossen, zu sich selbst zurückgeführt oder es ist eine unerwünschte "Schleife" zwischen den beiden äußersten Enden der Verbindung entstanden. Dadurch wird der NET-Leitungsabschluss deaktiviert. Es ist ratsam, eine laufende Übertragung an der Sendestation abzubrechen, bevor man die Kabel abzieht oder umsteckt. Der Autor hat allerdings noch keine bösen Überraschungen beim "hot-plugging" der Netzkabel erlebt!

Fehlkonfigurierte Stations-IDs

Gern wird der NET-Befehl vergessen, um jedem Gerät eine eindeutige Stations-ID zuzuweisen. Dies passiert besonders gern dann, wenn mehr als zwei Geräte angeschlossen sind. Genauso gern wird bei der Eingabe des jeweiligen NETI_x/NETO_x-Gerätenamens die Stations-ID der Maschine vergessen, an der Sie gerade tippen. Überprüfen Sie den NET-Befehl und geben Sie ihn erneut ein oder geben Sie den entsprechenden Gerätenamen erneut ein.

Broadcast Fehler

Die Netzwerk-Broadcast-Funktion der Station 0 (unter Verwendung von NETI_0/NETO_0) ist aufgrund des Fehlens einer aktiven "Bestätigung" im Broadcast-Protokoll nicht besonders zuverlässig. Testen Sie erneut eine direkte Peer-to-Peer-Dateiübertragung, um die Verbindung zu überprüfen, und wiederholen Sie dann die Übertragung. Das Aufrüsten eines alten QDOS-ROMs auf das wunderbare Minerva-Betriebssystem kann auch die Netzwerkübertragung zuverlässiger machen, da diese Funktion im NET-Treiber von Minerva neu kodiert wurde. Oder Sie fügen TK2 hinzu, das die Übertragungsfunktion vollständig durch eine wesentlich verbesserte Version ersetzt.

Der QL scheint zu hängen

Dies ist kein Fehler, sondern ein normaler Teil der Nutzung des Netzes. Interrupts innerhalb des "physischen" NET-Gerätetreibers werden während der Übertragung jedes Pakets abgeschaltet, um ein einheitliches Timing zu gewährleisten. Wenn Sie eine laufende Übertragung abbrechen, versucht die sendende Station, das aktuelle Paket bis zu 2.000-mal erneut zu übertragen. Dabei wird die Tastatur kaum beachtet, auch wenn der Benutzer zwis-

chen den einzelnen Versuchen Strg+Leertaste drückt. Die Gegenstelle reagiert in der Regel besser auf die Tastenkombination Strg+Leertaste.

Regel kürzere Pausen zwischen seinen Übertragungsversuchen. Es tastet die Tastatur zwischen den einzelnen Paketen nur nach einem "Break" ab, ohne dass weitere Benutzeraktivitäten möglich sind. Das passiert solange, bis das erwartete Paket erfolgreich empfangen und verarbeitet wurde.

Zusammenfassend lässt sich sagen, dass die QLAN- und ZXNet-Protokolle auf einem Mechanismus zur erneuten Übertragung von Paketen basieren (ähnlich wie

bei Ethernet). Dabei wird durchaus akzeptiert, dass einzelne Pakete "verpasst" werden, wenn der Empfänger gerade nicht zuhört. Das Paket wird dann immer wieder neu gesendet, bis eine aktive Bestätigung eingeht und das nächste Paket zur Übertragung in die Warteschlange gestellt werden kann. Der IOSS-Wiederholungsmechanismus in QDOS ist daher ein wesentliches Standbein für das QLAN-Protokoll. Der effektive Durchsatz hängt also in hohem Maße von der "Aufmerksamkeit"

Beim 'Streaming' serieller Daten wird nichts empfangen

Auch hier gilt, dass dies kein Fehler ist, wenn ein explizites OPEN oder sogar ein COPY von einem anderen "seriellen" Gerät verwendet wird. Das Paket-/Blockdesign des NET-Treibers bedeutet, dass sich ein volles Paket von 255 Bytes im Puffer des Absenders ansammeln muss. Erst dann wird der physische Treiber zur Übertragung über die Leitung aufgerufen. Da der aktuelle NET-Treiber keine 'FLUSH'-Funktion besitzt, muss entweder die Ausgabe aufgefüllt werden, um den 255-Byte-Puffer zu füllen. Oder es muss ein CLOSE an der sendenden Station ausgegeben werden. Wenn der NET-Treiber geschlossen wird, versucht er, das im Puffer befindliche Paket zu senden, unabhängig davon, wie voll es ist, und markiert dies als "letztes" Paket/Block. Dies wiederum führt dazu, dass die Empfangsstation eine EOF-Bedingung erkennt.

Bad Parameter (BP)-Fehler

Beim Empfang einer Datei mit LBYTES/EXEC usw. erwartet die Empfangsstation einen einfachen "seriellen Dateikopf". Dies sind die ersten 15 Bytes des allerersten Pakets, den sie erkennt und dessen erstes Byte "0xFF" ist. Die verbleibenden 14 Bytes des seriellen Headers enthalten die wichtige Dateilänge, das Byte "Datentyp" und – falls es sich um eine Datei vom Typ "Job" handelt – den Datenbereich. Wenn das erste empfangene Byte nicht 0xFF ist, schlägt der Befehl mit dem QDOS-Fehler 'Bad Parameter' fehl. Ein BP-Fehler tritt auch auf, wenn das Dateityp-Byte im Header nicht mit den Erwartungen übereinstimmt. Einige ältere Versionen von TK2 scheinen auch zu erwarten, dass LOAD mit einem seriellen Header arbeitet. Dieser wird aber nicht immer von dem entsprechenden SAVE gesendet. Aktualisieren Sie daher besser auf eine neuere Version von TK2

Nach dem Laden von TK2 von der Disk funktioniert nichts mehr

Aufgrund der sensiblen Software-Timing-Schleifen, die vom NET-Treiber verwendet werden, ist bei der Ausführung von TK2 aus dem RAM auf einem einfachen QL keine erfolgreiche Kommunikation möglich. Die RAM-Zugriffsbeschränkung führt zu inkonsistenten Verzögerungen bei der Ausführung von Software oder beim Lesen und Schreiben ins RAM. TK2, das im inhärent unbeaufsichtigten ROM oder im RAM auf einer (Super)GoldCard läuft, vermeidet dieses Problem. Nachdem der Autor nun mehrere Jahre lang mit QL-Netzwerken und verschiedenen QLS und kompatiblen Maschinen experimentiert hat, stößt er heute nur noch selten auf Probleme bei der Verwendung von QNET - außer denen, die er selbst verursacht!

des Empfängers ab. Er ist weniger von der rohen Netzwerk-Bitrate abhängig und das ist ein Grund, warum der Q68 als Beispiel einen besseren Empfangsdurchsatz erreicht als der Basis-QL.

Alternativen zu QLAN

Dieser Artikel hat bewusst die spannenden IPNet/Ethernet-Projekte von Martin Head und die SERNET-Lösung ausgeklammert. Letztere wurde ursprünglich von Bernd Reinhardt entwickelt und basiert auf Phil Bormans MidiNet-Software. Es gibt noch weitere interessante und verwandte Projekte. So stellt das jüngste Retro-WiFi-Projekt eine drahtlose IP-Fähigkeit über die serielle Schnittstelle des QL her. Wir empfehlen Ihnen, bei Interesse selbst nachzuforschen. Sie können in jedem Fall reichlich Informationen über diese anderen großartigen Projekte finden.

Die TK2-Erweiterungen von QNET

Während das NET-Basisgerät einige sehr nützliche Funktionen bietet, werden fast alle diese Funktionen mit TK2 verbessert oder anderweitig durch weiterentwickelte Funktionen ergänzt. Um ehrlich zu sein, wird TK2 für alles gebraucht, was über die triviale Nutzung des QL hinausgeht. Es holt das Beste aus dem Gerät heraus, und das ist bei der Vernetzung sicherlich der Fall. TK2 ersetzt den gesamten einmal installierten QDOS NET-Treiber. Die QLAN-Protokollspezifikation wird beibehalten, um eine grundlegende Kommunikation auch mit QLs ohne TK2 zu ermöglichen. Es verbessert die Zuverlässigkeit einiger Funktionen (z.B. Netzwerk-Broadcast) und fügt den wichtigen FSERVE-Server Job mit seinem entsprechenden 'Nx'-Gerätetreiber auf der Clientseite hinzu. Wie bereits erwähnt, benötigt TK2 einen unbelegten RAM- oder ROM-Speicher, um das QL-eigene RAM-Zugriffsverhalten zu vermeiden. Andernfalls würde das empfindliche Timing des Bitlevel-Treiber-codes durcheinander gebracht.

Die grundlegende NET-Funktionalität bleibt wie zuvor beschrieben. Änderungen ergeben sich, sobald der FSERVE-Job auf einer oder mehreren der angeschlossenen Stationen aufgerufen wird. Dann ist es möglich, auf die Datei- und Geräteressourcen auf der Server-Station zuzugreifen, als ob sie lokal installiert wären. Dies passiert durch die Verwendung des "Nx"-Client-Pseudogeräts. FSERVE ist flexibel genug, um der Client-Station fast alle Funktionen des entfernten Geräts zur Verfügung zu stellen. Auf dem Client muss TK2 installiert sein, FSERVE muss aber nicht laufen. Au-

ßerdem kann vom Client aus auf jeden Gerätetreiber zugegriffen werden, der mit der bedienenden Station verbunden ist. Dies ist nicht beschränkt auf die offensichtlichen Dateisystemgeräte wie MDV, FLP oder WIN. So können SCR, CON, SER, PAR und andere auf der bedienenden Station installierte Geräte vom Client aus mit der gleichen allgemeinen Form `Nx_<remote_device_spec>` angesprochen werden. Einige wirklich clevere Netzwerk-Programmiertechniken eröffnen sich insbesondere mit dem MEM-Speichergerät (dank des DIYTK-Treiber von Simon N. Goodwins).

Der größte Nutzen dieser Client-Server-Fähigkeit ergibt sich jedoch in der Praxis beim Laden oder Speichern von Dateien auf der Server-Station. Hier sind einige Beispiele; Nx steht dabei für `N<server_station-ID>`. Zu beachten ist, dass nach dem Aufruf von FSERVE an der Server-Station nichts mehr eingegeben werden muss.

_____ Sicherung und Archivierung ganzer MDV-Kassetten über das Netzwerk in einen Ordner auf dem WIN-Gerät des Servers, mit einem einzigen Befehl auf der Client-Station:

```
WCOPY 'mdv1_' TO  
'Nx_win1_backup_'
```

_____ Senden von Druckdateien an einen Qualitätsdrucker, der an den SER-Port des Servers angeschlossen ist. Besser noch, verwenden Sie den Befehl SPOOL von TK2 anstelle von COPY:

```
COPY 'mdv1_print_txt' TO  
'Nx_ser1hr'
```

_____ Senden von Nachrichten an die Bildschirmanzeige des Benutzers, der an der Bedienstation sitzt:

```
OPEN  
#3, 'Nx_scr_100x20a206x118':  
PRINT #3, "Hello!": PAUSE:  
CLOSE #3
```

_____ Laden von ausführbaren Programmen direkt aus dem Dateisystem des Servers:

```
EX  
'Nx_win1_APPS_QED_qed'; "mdv1_text_file"
```

_____ Direkte Manipulation des Speichers (z.B. Video-Speicher) des Servers mit MEM:

```
OPEN #3, 'Nx_mem': PUT  
#3\131072: PRINT #3,  
FILL$ (CHR$ (170) & CHR$ (0), 32767);: CLOSE #3
```

_____ Synchronisierung der lokalen Echtzeituhr (aus den Minerva-Dokumenten) – wer braucht schon NTP?

```
f$="Nx_ram1_date_tmp":  
d%=FOP_NEW(f$)  
IF d%>0 THEN CLOSE #d%:  
SDATE FUPDT(\f$): DELETE f$
```

FSERVEs verborgener Schatz

Es gibt eine verborgene oder zumindest wenig dokumentierte Funktion, die bei der Ausführung von FSERVE zur Verfügung steht. Sie ist von Vorteil beim "Bridging" zwischen einem QL und einem QL-Emulator auf einem PC oder anderen QL-Plattform, die keine NET-Ports haben. Ein auf einer Zwischenstation laufendes FSERVE öffnet gerne im Namen des Clients ein Netzwerkdevice. Es ist auf eine andere Zielstation ausgerichtet. Haben Zwischenserver (Nx) und Zielstation (Sy) ebenfalls den SERNET-Treiber geladen und sind diese über ein geeignetes serielles Kabel miteinander verbunden, führt ein Befehl wie

```
COPY 'mdv1_file' TO  
'Nx_Sy_win1_file'
```

dazu, dass `mdv1_file` auf das WIN1-Gerät des Emulators kopiert wird. Dies passiert, obwohl der emulierte QL nur indirekt mit der Client-Station verbunden ist und selbst nur einen SERIAL-Port und keine QNET-Verbindung hat. Damit wird die derzeitige Einschränkung überwunden, dass sich QL-Emulatoren nicht direkt mit dem QL-Netzwerk verbinden können.

In der Praxis erweist sich diese Überbrückung manchmal als schwerfällig. Dies liegt daran, dass jedes Paket, aus dem `mdv1_file` besteht, über die Zwischenstation "hüpfen" muss. FSERVE auf Nx verpackt es neu und gibt es an den SERNET-Treiber weiter. Dieser liefert es an den endgültigen Speicherort auf dem WIN-Gerät des Emulators auf Station Sy aus. Eine weitere Erklärung für diesen Leistungsabfall ergibt sich, wenn man bedenkt, dass der Client-Server-Treiber des TK2-Netzes das übliche "Befehl/Antwort"-Paradigma verwendet. Das führt zu einer zusätzlichen Latenz zwischen den einzelnen Datenpaketen führt. Für jeden gesendeten Datenblock sind zwei oder mehr "Befehls"-Pakete erforderlich. Obwohl dies bei der direkten Kommunikation von Station zu Station nur geringfügig auffällt, werden die Verzögerungen durch den zusätzlichen Sprung bei dieser Brückentopologie viel deutlicher. Der TK2-Client-Server verwendet erweiterte Paketgrößen bis zu 1020 Byte (obwohl Dateisystempakete typischerweise 524 Byte lang sind). Glücklicherweise wird ein Teil der Bandbreitenineffizienz des SimpleNET-Geräts durch das Vorhandensein von weniger Paketlücken für einen bestimmten Payload gemildert.

Alles in allem ist dies eine wirklich nützliche Funktion von FSERVE in Kombination mit seinem SERNET-Dateiserver-Äquivalent. Es ist wirklich ein Verdienst der Entwickler der einzelnen Treiber. Nur

aus akademischem Interesse oder zum Spaß können Sie diese Funktion auch in Aktion zwischen drei QLs beobachten, die nur das QLAN benutzen und auf denen TK2 läuft. Die Client-Station Pakete lässt von einer Zwischenstation Nx bouncen, um das Ziel Ny zu erreichen, Versuchen Sie es mit einer 32KB QL-Anzeigedatei und dem Befehl:

```
LBYTES 'Nx_Ny_win1_ql-dis-
play_scr',131072
```

Es wird funktionieren, aber nicht gerade atemberaubend sein. Abschließend ist zu

bemerken, dass wider Erwarten jeder Versuch, die Kette durch die Einführung einer weiteren Zwischenstation zu verlängern, völlig fehl zuschlagen scheint.

```
COPY 'mdv1_file' TO
'Nx_Ny_Nz_win1_file'
```

Es ist noch nicht klar, warum diese Konfiguration scheitern sollte. Einen praktischen Nutzen hat sie besitzt sie allerdings nicht.

Verbinden mit dem ZX Spectrum

Ungeachtet der bekannten Probleme bei der Kommunikation zwischen dem QL und

dem Spectrum über QNET/ZXNet gibt es einige nützliche Dinge, die möglich werden, wenn diese beiden Geräte miteinander verbunden sind. Sie sind es wert, hier genauer betrachtet zu werden. Während der Nachforschungen über den Anschluss des ZX Spectrum entdeckte der Autor, dass Sinclair möglicherweise ZXNet – oder den "Sinclair Network Standard" – als Kernbestandteil des ursprünglichen ZX82/Spectrum-Designs ins Auge gefasst hatte. Zunächst wurde seine Übernahme als "offenen Standard" gefördert. Später jedoch wurde er von der Spectrum-Hardware und

QLAN to USB Bridge (QLUB) Adapter

Der QLAN to USB Bridge (QLUB) Adapter ist eine Open-Source-Lösung. Sie ermöglicht die Verbindung und Kommunikation zwischen echter Sinclair QL-Hardware und einem QL-Emulator, der auf einem Host-PC läuft. Sie nutzt Sinclair's QLAN/QL-Net Protokoll und Verkabelung, mit Unterstützung für die TK2 QLAN Erweiterungen, einschließlich FSERVE.

Es gibt heute mehrere Möglichkeiten, Programme und Daten zwischen emuliertem und QL-Hardware zu übertragen. Dies schließt SERNET und herausnehmbare SD-Kartenmedien ein. Diese erfordern je-

Sie schafft eine Schnittstelle zwischen QDOS/SMSQE-Client-Anwendungen oder Gerätetreibern und dem Mikrocontroller, der über einen freien USB-Anschluss mit dem Host-PC verbunden ist. Der Adapter erfordert einige zusätzliche Komponenten, die leicht erhältlich sind. Sie sind für die meisten Enthusiasten einfach genug, um auf einer Lochrasterplatine mit minimalem Löt Aufwand zusammengebaut zu werden.

Es wurde auch ein SBASIC-Programm entwickelt, um das Senden und Empfangen von Dateien zwischen dem QL Emulator und einem anderen QL (oder einem

neuesten Mikrocontroller-Firmware und einer Anleitung sollen so Anwendertests ermöglicht sein. Die die Arbeit an der Software wird währenddessen fortgesetzt. Sie wird die letztendlich vollständig in in QDOS/SMSQE im Emulator integriert.

Der QL-Emulator QPC2 wurde während des gesamten Entwicklungszyklus verwendet. Die endgültige Lösung sollte jedoch mit jedem QL-Emulator kompatibel sein. Er muss dazu nur die COM/seriellen Schnittstellen des Host-PCs zugänglich machen und es braucht einen geeigneten USB Virtual COM Port Treiber für den Atmel Mikrocontroller.

Da der ZX Spectrum mit dem Interface-1 auch kompatible Netzwerk-Hardware und -Software bietet, funktioniert der QLUB ebenso gut mit einem Spectrum. Sogar ein Spectrum Next mit Int-1 ließe sich verbinden. Das gleiche gilt für jede QL-ähnliche Lösung, die die QLAN NET-Port-Hardware enthält, wie z.B. der Q68, QXL und Aurora andere.

Seit März 2021 funktioniert die Mikrocontroller-Firmware und der größte Teil der Software-API-Schnittstelle und der Entwurf der Datenstruktur wurde abgeschlossen. Auch die Realisierung der "Message-Queue"-Server, die den Adapter mit QDOS verbindet, war weit fortgeschritten. Nach der Beseitigung einer Reihe von Fehlern wurde eine vollständige Fassung für den Herbst 2021 angekündigt.

Die aktuelle Fassung der Bauanleitung ist auf der Heft-CD zu finden.

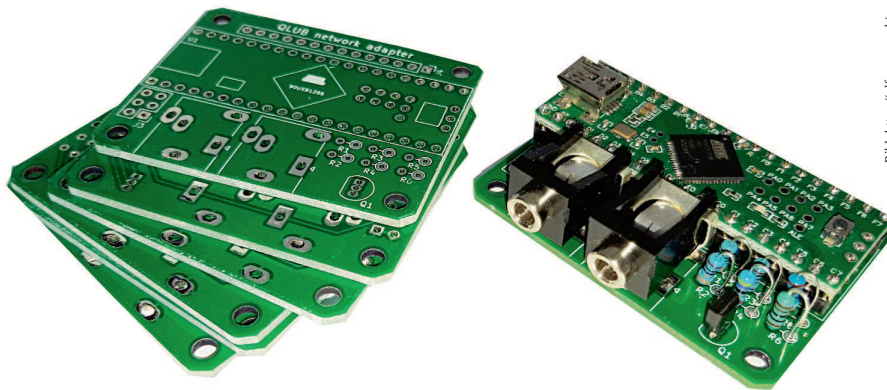


Bild: <https://qlforum.co.uk/>

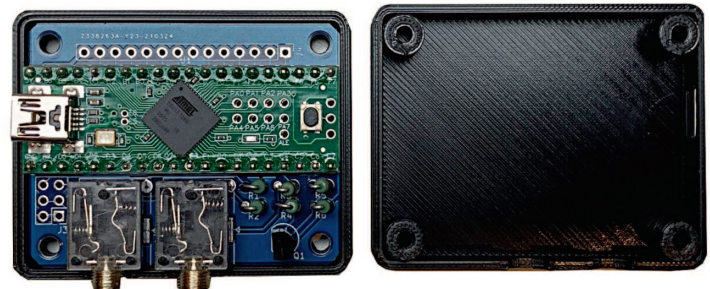
PCB der Prototypen des QKUB Adapters

doch entweder zusätzliche Hardware oder Software auf der QL-Seite. Sie schränken daher ihre Zugänglichkeit für "nicht erweiterte QL"-Benutzer ein. QL-Net hingegen ist in alle QLs eingebaut und ist daher selbst mit einer minimalen QL-Konfiguration kompatibel.

Der QLUB-Adapter besteht sowohl aus Hardware- als auch aus Softwarekomponenten. Die Hardware basiert auf einer Mikrocontroller-Entwicklungsplatine (Teensy ++2.0) mit kundenspezifischer Firmware und einigen zusätzlichen Komponenten. Außerdem gibt es eine Software, die in einem QL-Emulator auf dem Host-PC läuft.

kompatiblen Gerät) zu testen. Es funktioniert bereits bei der Übertragung von Dateien zwischen diesen Plattformen.

Diese "Proof-of-Concept"-Software wurde der QL-Community in diesem Stadium der Vorabveröffentlichung zur Verfügung gestellt (Dezember 2020). Zusammen mit der



Gedrucktes Gehäuse eines Prototypen

Bild: <https://qlforum.co.uk/>

ihrem 16K-ROM-Basisgerät gestrichen. Schließlich tauchte er in Form des Interface-1 Add-ons wieder auf.

Einer der vielen Spectrum-ROM-Neuentwickler, Geoff Wearmouth, ging sogar so weit, die ZXNet-Routinen in sein Ersatz-16K-ROM (The Sea Change ROM) zu implementieren. Es verwendet die Interface-1-Hardware, aber nicht deren Firmware. In jedem Fall haben der Basis-QL und Spectrum/Interface-1 ein völlig kompatibles Protokoll für ihre jeweiligen Netzimplementierungen.

Wie kommt es dann, dass die meisten Benutzer nicht in der Lage waren, die Spectrum/QL-Verbindung für sich nutzbar zu machen? Die Netzwerkkommunikation von Spectrum zu Spectrum erscheint ja relativ brauchbar. Während der Untersuchung dieser interessanten Tangente zum QL-Netzwerk durch den Autor wurden mehrere Fehler im Interface-1 Netzwerkcode identifiziert. Der wichtigste von diesen betrifft das inkonsistente Timing beim Senden bestimmter Bitmuster vom Spectrum. Die Ursache wurde beim Schreiben des nächsten Übertragungsbits in das Interface-1 ULA-Register als "IO Contention" festgestellt. Diese Feststellung deckt sich mit der allgemeinen Beobachtung, dass einerseits der Übergang von QL zu Spectrum recht gut funktioniert. Andererseits kommt nur selten etwas vom Spectrum zurück, was seinen Nutzen stark einschränkt. Dieses Verhalten wurde inzwischen diagnostiziert. Es wurden Ersatz-Z80-Routinen entwickelt und getestet. Sie beseitigen neben anderen subtilen Fehlern auch das ULA-Contention-Problem.

Der Austausch des Interface-1-ROM ist nicht trivial, um es milde auszudrücken. Dies liegt zum Teil an den außergewöhnlich engen Abständen innerhalb des eleganten Interface-1 Gehäuses. Nichtsdestotrotz hat der Autor eine zuverlässige Kommunikation zwischen diesen Maschinen mit den modifizierten Interface-1-ROM-Routinen erreicht. Er würde den Code gerne mit interessierten Benutzern teilen, die bereit sind, ihre wertvolle Interface-1-Hardware zu modifizieren.

Alternativ dazu kann der QL mit Hilfe eines leicht überarbeiteten QNET/TK2-Treibers auf dem QL und der Anpassung einiger Timing-Konstanten besser an die wackeligen Bit-Timings des Spectrum angepasst werden. Dies vermeidet eine Modifikation des Interface-1. Das Ergebnis ist nicht 100%ig perfekt, aber ein guter Schritt.

Glücklicherweise können der mit SGC ausgestattete QL sowie der QXL die Zeitanomalie des Spectrum bereits ohne weitere Anpassungen bewältigen, wenn auch nicht immer zu 100% zuverlässig. Dies gilt möglicherweise auch für den GC. Der Q68,

der mit dem ND-Q68-Treiber betrieben wird, scheint dagegen mit den Wacklern des Spectrum gut zurechtzukommen.

Wie auch immer es erreicht wird, sobald eine zuverlässige Verbindung hergestellt werden kann, eröffnen sich viele Möglichkeiten:

- _____ Sicherung der ZX-Microdrive-Kassetten auf einem zuverlässigen QL-Speicher
- _____ Bequeme Entwicklung von Spectrum-Software in einem Emulator auf dem QL (z.B. Speculator oder dem brillanten ZM/x) und Rückübertragung auf einen "echten" Spectrum
- _____ Übertragen von Bildschirmkopien Ihrer Lieblings-Spectrum-Spiele rendern Sie auf dem QL mit zusätzlicher QL-Software
- _____ Multiplayer-/Multiplattform-Netzwerkspiele

QLAN kann also auch hier einen echten Mehrwert bieten.

*Übersetzung und redaktionelle Bearbeitung:
Georg Basse (mit der freien Version von
www.deepl.com)*



Das Original dieses Artikels mit Programmbeispielen und weiteren Informationen findet sich auf der Heft-CD



Sir Clive Sinclair, Erfinder und Gründer von Sinclair Research, hat sehr früh als Erfinder von sich Reden gemacht, Schon 1958 hatte er als Schüler ein Miniaturradio entworfen. Er

schrieb in Fachjournalen über HiFi Technik und gründete im Alter von 21 Jahren seine erste Firma, Sinclair Radionics. Mit Taschenradios, Stereoverstärkern und Lautsprechern hatte er durchaus Erfolg am Markt. Etwa 2 Mio. Pfund spülte dann der Executive in die Kassen, ein schicker, auf die Grundrechenarten reduzierter Taschenrechner. Nachfolgeprodukte wie die erste Digitaluhr und ein Taschenfernseher hingegen flopten.

Zusammen mit Tim Curry hatte Sir Clive mit seinem neuen Unternehmen Sinclair Research mehr Erfolg. Die kleinen, preiswerten Modelle Sinclair

ZX80, ZX81 und ZX Spectrum brachten zu Beginn der 1980er Jahre den Computer in viele Haushalte. Trotz offensichtlicher Qualitätsmängel und Designfehler machte Sinclair 14 Mio. Pfund Gewinn pro Jahr und Sir Clive zu einem reichen Mann. Sein Erfolg brachte ihm 1983 den Ritterschlag ein. Sein letzter Rechner, der Sinclair QL, sollte das Profisegment erobern. Doch eine übereilte Markteinführung und halb fertige Software ließ das Gerät zum Misserfolg werden. Endgültig in die roten Zahlen wurde Sinclair Research durch den C5 katastrophiert, ein Elektro-Dreirad für eine Person. Sicherheits- und Qualitätsmängel ließen das Projekt nach 17.000 produzierten Fahrzeugen sterben. Sir Clive verkaufte schließlich 1986 sein Unternehmen an den Konkurrenten Amstrad.

In der Folgezeit konzentrierte sich Sir Clive auf seine Arbeit in einer Stiftung für Hochbegabte und machte Karriere als Pokerspieler. Er erlag am 16. September 2021 in London seiner langen Krankheit.

Links

SuperBASIC/SBASIC Reference Manual Online:

<https://superbasic-manual.readthedocs.io/en/latest/index.html>

QL Benutzerhandbuch:

<http://www.dilwyn.me.uk/docs/ebooks/olqlug/index.htm>

QL68 Nachrüstung:

<https://www.qlforum.co.uk/viewtopic.php?f=3&t=2881>

Paketaufbau ohne TK2:

<https://superbasicmanual.readthedocs.io/en/latest/Appendices/Appendix17.html#a17-1-4-qnet-without-toolkit-ii>

Reinigen der NET Buchsen:

<https://qlforum.co.uk/viewtopic.php?f=12&t=3279>

SeaChange ROM für Spectrum:

<http://zxspectrum.it.omegahg.com/rom/seachange/seachange.pdf>

Ueber den Autor

Martyn Hill arbeit als Technical Customer Success Manager fuer ein globales Softwareunternehmen mit Sitz in Grossbritannien in dessen Cyber Thread Abteilung. Heterogene Netzwerke haben ihn schon immer fasziniert. Mit 11 Jahren bekam er von seinem Bruder den ersten Sinclair Spectrum und spaeter einen Sinclair QL.

Eine PET Replika mit Extras

Der Micro-PET

Der Micro-PET ist eine "Neuaufgabe" des Commodore-PET von 1977 und seinen Nachfolgemodellen. Sie umfassen die Modelle bis zum 8296 aus den Jahren 1984 bis 1986. Der Commodore PET zählt zusammen mit dem Apple-II und dem TRS-80 zu der "1977 Trinity".

Der Commodore PET nutzt – wie auch einige andere Rechner dieser Zeit – den MOS 6502 als Prozessor und lässt ihn mit 1 MHz laufen. Die monochrome Bildschirmausgabe – meist Phosphorgrün auf Schwarz – wird am Anfang komplett mit Logik-Chips aufgebaut. Erst später kommt mit dem CRTC6545 ein programmierbarer Video-Chip hinzu. Der Bildschirm ist fest eingebaut.

Der Speicher läuft nur mit 1 MHz, deshalb muss die CPU auf den Bildschirm (Vertical Retrace) warten, um auf den Bildschirmspeicher zuzugreifen. Dies wurde später behoben. Apropos Speicher: Die ersten Modelle hatten 4 KByte RAM, wurden aber schnell auf 8 KByte aufgerüstet. Der "normale" PET hat dann 32 KByte und nur die späten 8296-Modelle haben 128 kByte RAM. Als Peripherie ist in den ersten Modellen ein Tape-Deck neben dem Keyboard eingebaut. Der Begriff "Datasette" war damals noch nicht erfunden. Spätere Modelle haben das nicht mehr, dafür aber ein größeres Keyboard. Eine Besonderheit ist die Nutzung des IEEE488 Bus zum Anschluss von Peripheriegeräten. Dieser auch als "HP-IB" oder "GP-IB" bekannte und von Hewlett-Packard entwickelte Bus erlaubt es, mehrere Peripheriegeräte anzubinden und über eigene Adressen zu verwenden. Dieser Bus wird noch heute in der Forschung und Entwicklung für die Anbindung von intelligenten Mess- und Steuergeräten verwendet.

Commodore baute dafür eine Serie von Floppy-Disk-Laufwerken. Diese konnten



mit dem ersten Set von ROMs aber noch gar nicht angesprochen werden. Die Laufwerke waren noch nicht fertig, der Code ungetestet und außerdem fehlerhaft. All das wurde in späteren Versionen bereinigt. Mit ausgeliefert wurde das Microsoft-BASIC. Commodore hatte das BASIC von Microsoft gekauft und nicht etwa lizenziert, wie es später bei Microsoft üblich wurde. Commodore hat tatsächlich dafür eine Einmalzahlung geleistet – Bill Gates hat hier wohl in Commodore CEO Jack Tramiel einen würdigen Gegner gefunden. Beim

Einschalten des Commodore PET erscheint ebenso wie beim C64 die Meldung mit dem typischen READY Prompt auf dem Bildschirm. Insgesamt ist der Commodore PET aus vielen Standard-Logikbausteinen aufgebaut, ergänzt durch ein paar der typischen 65er-I/O Chips. Insbesondere werden neben dem 6502 Prozessor die 6522 VIA, zwei 6521 PIA und der 6545 CRTC verwendet. Dies wird im folgenden noch wichtig sein.

Wie aus dieser Beschreibung zu sehen ist, war der Commodore PET in gewissem Sinne ein Vorläufer des VC20 und des C64. Mit diesen teilt sich der PET sich einige Dinge wie das BASIC, den Kernel, das Tape Interface und den IEC-Bus als eine hardwareseitig vereinfachte Variante des IEEE488.

Durch seinen Erfolg selbst, aber auch durch seine "Vaterschaft" am C64 hat sich der PET seinen Platz in den Geschichtsbüchern verdient.

Ziele des Micro-PET

Der Micro-PET sollte eine Open Source Replika des Commodore PET werden, der zum einen mit aktuell noch verfügbaren ICs gebaut werden kann. Zum anderen sollte er zusätzlich eben auch moderne Features besitzen: mehr Speicher, höhere Taktfrequenz und modernere Peripheriegeräte. Ganz besonders wichtig war das Ziel, den Micro-PET auch mit einem VGA Monitor nutzen zu können. Damit wird es einfach, ihn an moderne Monitore anzuschließen.

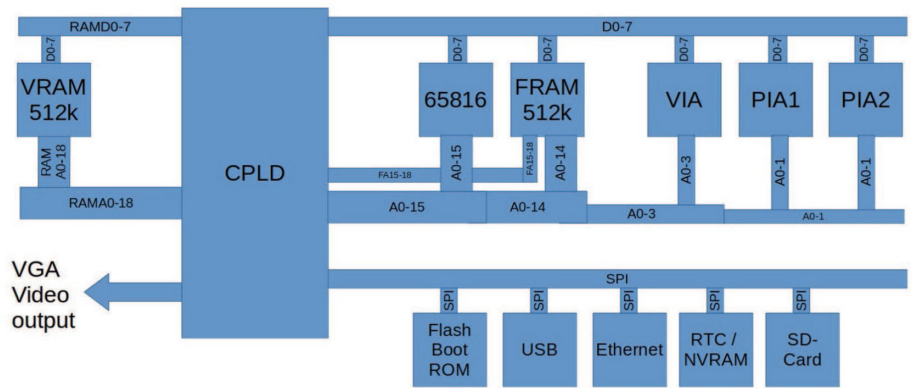
Zwar braucht man inzwischen meist einen HDMI-Adapter, doch erlaubt das VGA-basierte Timing damit die Nutzung von HDMI-Monitoren. In dem Zusammenhang sollte es auch möglich sein, verschiedene Modelle des PET abzubilden – von den 40-Zeichenmaschinen mit 32k bis zu den 80-Zeichenmaschinen mit 128k. Dies sollte auch mit Software umschaltbar sein, um Dip-Switches oder ähnliches zu vermeiden. Des Weiteren sollten die Erfahrungen, die der Autor mit SPI-basierten Ethernet-, USB- und SD-Card Schnittstellen gesammelt hatte genutzt werden, um diese Schnittstellen dem PET beizubringen. Und zu guter Letzt sollte das ganze mit einer „vernünftigen“ Tastatur in einem ansprechenden Gehäuse unterzubringen sein.

Systemübersicht

So wie der PET im Grunde genommen ein relativ einfaches System ist, so wurde auch der Micro-PET ein recht einfaches System. Das Diagramm zur Systemarchitektur zeigt das deutlich. Auf der oberen rechten Seite ist ein einfaches 6502-System zu sehen, mit der 65816 CPU, RAM und den drei aus dem PET stammenden I/O-Chips VIA und PIA1/2. Auffällig ist das fehlende ROM – dazu gleich mehr. Links davon ist der große CPLD – ein „Complex Programmable Logic Device“. Dieser übernimmt die Aufgabe des CRTC Video-Chips aus dem PET und hat dazu – eigentlich ganz analog zum Original PET – ein eigenes Video RAM. Nur sind die RAM-Bausteine nicht in 1 KByte zu bemessen, sondern eher 512kB. Dieser CPLD übernimmt damit praktisch die gesamte Logik der Adressselektion und Videogenerierung. Rechts unten im Diagramm ist noch der SPI-Bus (Serial Peripheral Interface Bus) zu sehen. An diesem hängen verschiedene Geräte wie ein Flash ROM, ein USB-Interface Chip, Ethernet, batteriegepuffertes RAM sowie eine SD-Card Schnittstelle.

CPU und Speed

Die verwendete CPU ist eine 65816 vom Western Design Center. Diese hat einen Nachteil und zwei Vorteile: Der Nachteil ist die fehlende Unterstützung der sogenannten „illegal Opcodes“ des 6502. Diese werden insbesondere häufig auf dem C64 zur Code-Optimierung in Spielen und Demos verwendet. Auf dem PET sind solche Programme aber nicht bekannt. Der erste Vorteil ist, dass die CPU von Haus aus 16 bit verarbeiten kann und auch einen Adressbereich von 16 MByte verwaltet. Das ist insbesondere wichtig bei der Adressierung des Micro-PET Speichers von 1 MByte RAM. Der zweite Vorteil liegt in der Geschwindigkeit. Die CPU ist deutlich schneller als die originale 6502 CPU. Eine 65816



Systemaufbau des Micro-PET

CPU ist für einen Takt von bis zu 14 MHz spezifiziert. Verschiedenen Berichten zufolge kann sie aber deutlich schneller betrieben werden. Die C64 „SuperCPU“ hat diesen Prozessor ebenfalls verwendet – sogar mit eigentlich nicht spezifizierten 20MHz.

Im Micro-PET wird die CPU grundsätzlich mit 12.5MHz betrieben. Dies dient dazu, die Videoausgabe mit der CPU synchron zu halten und genug Bandbreite für VGA-basiertes Videotiming zu bekommen (siehe unten). Über ein Konfigurationsregister kann die Geschwindigkeit aber auf PET-kompatible 1MHz, 2MHz, oder 4MHz gedrosselt werden.

Boot, ROM und Memory Map

Wie oben bereits erwähnt, fehlt auf der 65816-Seite der Maschine ein ROM. Daher lädt das CPLD vor dem Starten der CPU eine Speicherseite (256 Byte) aus

dem SPI Flash ROM und speichert es im RAM in der obersten CPU Speicherseite (\$FFxx). Damit bootet die 65816 CPU und kann allen weiteren Code vom Flash ROM selbst nachladen. Das vermeidet die Nutzung von parallel angesprochenen EPROMs, die in der Regel langsamer sind als das RAM und auch auf der Platine deutlich mehr Platz beanspruchen. Alle Bereiche, die im normalen PET als ROM ausgeführt sind, sind damit hier als RAM ansprechbar. Über ein Konfigurationsregister können diese Bereiche schreibgeschützt werden, um ein ROM zu simulieren. So ist es auch einfach, beispielsweise geänderte „ROMs“ einzuspielen und zu testen oder zu nutzen.

Da der Micro-PET 1MByte RAM hat, ist der Speicher so aufgeteilt wie im Diagramm gezeigt. Die ersten 512kByte sind das sogenannte Fast RAM – hier gibt es keine Konflikte im Speicherzugriff während der Videoausgabe. Die zweiten 512k dienen als Video RAM – hier kann der CPLD die CPU anhalten, wenn der Zugriff auf den Speicher durch die Videoausgabe erfolgt. Damit der PET emuliert werden kann, sind in Bank 0 die Speicherbereiche \$8xxx und \$E8xx gesondert gehandhabt:

Video	+----+	\$100000	
RAM			VRAM
			bank 15
	+----+	\$0f0000	
	...		
	+----+	\$090000	
			VRAM
			bank 8 (video)
	+----+	\$080000	
Fast			FRAM
RAM			bank 7
	+----+	\$070000	
	...		
	+----+	\$020000	
			FRAM
			bank 1 (8296 mapped memory)
	+----+	\$010000	
			FRAM (PET ROM / I/O / 4k Video mapped from VRAM)
			FRAM (lower 32k)
	+----+	\$000000	

Speichernutzung des Micro-PET Memory Map)

— \$8xxx – Dies ist ein Fenster in den Videospeicher. Statt des in Bank 0 normal genutzten Fast RAM wird hier das Video RAM verwendet, um den Inhalt wie beim PET auf den Bildschirm auszugeben.

— \$E8xx – Dies ist der I/O Bereich des PET. Hier werden die auf der Platine befindlichen VIA und PIA angesprochen, um die Tastatur oder den IEEE488 Bus zu bedienen.

Zusätzlich besteht die Möglichkeit, die oberen 32 KByte der Bank 0 in zwei Blöcken zu 16k aus der Bank 1 zu mappen. Dies entspricht der Speicherverwaltung des 8296, so dass auch dieses Modell korrekt nachgebildet wird. Anders als beim echten PET können auch die unteren 32 KByte der Bank 0 in eines von 16 Pages mit je 32 KByte des Fast RAM gemappt werden. Damit können beispielsweise unabhängige Programme im "time sharing" der CPU parallel laufen, ohne dass aufwändig Zeropa-ge oder Stack gesichert werden müssten.

Video

Die Video-Ausgabe findet grundsätzlich im Modus 640x480 bei 60 Hz statt. Dies erlaubt eine Darstellung von bis zu 80 Zeichen pro Zeile bei 50 Zeilen (also 8 oder 9 Pixelzeilen pro Zeichen). Ein Konfigurationsregister kann die Bildschirmauflösung aber auch an den Original-PET mit nur 40 oder 80 Zeichen pro Zeile bei 25 Zeilen anpassen. Ebenso kann die Videoausgabe in einen Hires Modus umgeschaltet werden. Die Ausgabe erfolgt dabei immer monochrom. Aufgrund von Platzproblemen im CPLD ist der in späteren PETs verbaute CRTC Video-Chip nur sehr begrenzt nachzubilden. So ist nur die Startadresse des

Speichers, sowie die Höhe eines einzelnen Zeichens (8 oder 9 Rasterlinien) einstellbar. Der CPLD ist in einer zweiten Variante aber auch als "4032" zu programmieren – es sind dann nur 40 Zeichen und keine der weiteren Funktionen für das Speicher-mapping vorhanden. Allerdings werden wesentlich mehr CRTC- Register simuliert.

Kompatibilität

Zugegebenermaßen wurde noch nicht alle PET Software getestet. Allerdings läuft das "8296 burn-in" Programm erfolgreich durch und bestätigt damit die Grundfunktionen. Bisher ließ sich auch kein Programm finden, das auf dem Micro-PET nicht läuft – solange es nur die einfache CRTC-Emulation benötigt. Die "No PETs Allowed" Demo läuft zwar mit der oben erwähnten 4032 Simulation, aber noch nicht "rund" – wobei das natürlich schon die höchsten Ansprüche stellt.

Zum Bauen

Wer einen Micro-PET haben will, muss ihn im Moment noch selbst bauen. Alle Informationen, die dazu nötig sind, liegen im Github Repository des Micro-PET.



Der Micro-PET im C64 Gehäuse

Dazu gehören:

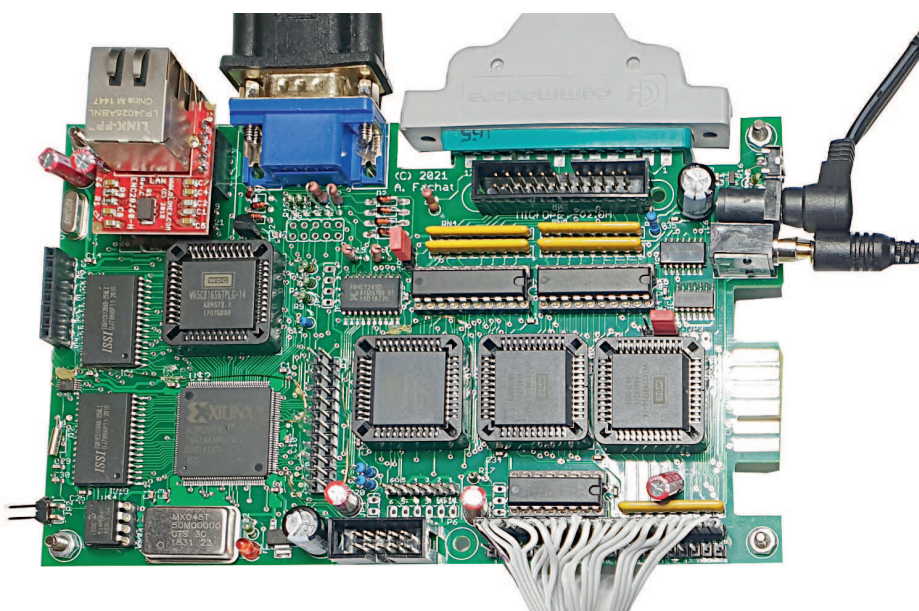
- Schaltplan des Boards und Eagle Layout, mit dem man sich die Platinen herstellen lassen kann, sowie eine Stückliste (inkl. Mouser Bestellnummern)
- VHDL-Code für den CPLD, mit entsprechenden Build- und Programmier-Anweisungen für das Xilinx Web-ISE IDE
- Scripts, um das Image für das Flash-ROM zu bauen (basierend auf Steve Gray's Editor ROM Projekt)
- Designs für 3D-druckbare Teile, mit denen die Platine mit einem PET Keyboard in ein C64c Gehäuse eingebaut werden kann

Fazit

Mit dem Micro-PET steht eine neue Fassung des Commodore PET mit hoher Kompatibilität, aktuell noch lieferbaren Bausteinen und modernen Schnittstellen zur Verfügung. Durch den Einbau in ein C64c-Gehäuse und VGA Video-Anschluss lässt sich das "PET Feeling" auch heute noch einfach auf echter Hardware erleben! Durch das offene Design können sich alle daran beteiligen und das System weiterentwickeln. So ließe sich auch eine Platine entwerfen, die statt dem bisherigen Euro-PCB Format auf den Original-Formfaktor des C64 oder gar des PET hat. So wäre verwaisten Gehäusen ein neues Leben einzuhauchen.

Links

- <http://www.6502.org/users/andre/upet/index.html>
- <https://github.com/fachat/MicroPET>
- <http://www.6502.org/users/andre/csa/index.html>
- https://en.wikipedia.org/wiki/Commodore_PET



Die fertig aufgebaute Platine

Interview mit Autor und Micro-PET Entwickler André Fachat

LOAD: Wie bist du dazu gekommen, dich mit Commodore Computern zu befassen?

André: Ich habe das Programmieren in der Schule auf einem Commodore PET 3032 gelernt, bevor ich dann meinen eigenen C64 bekam. Da habe ich dann schnell gemerkt, dass meine Lieblingsprogramme und Spiele nicht so einfach auf dem C64 liefen. Da ich zur Schulzeit bereits mit einem Freund an 6502-basierten Systemen gearbeitet habe, nahm ich mir zu Beginn des Studiums dann vor, eine PET-Replika zu bauen. Das war 1989.

LOAD: War das bereits der Anfang des MicroPET?

André: Nein, eigentlich nicht. Die erste PET-Replika ist der CaSpAer (aka CS/ A65), ein Multiboard-Computer auf Eurokartenbasis. In den letzten Jahren habe ich sogar eine Karte mit Ethernet, USB, und SD-Card Unterstützung gebaut.

LOAD: Das klingt doch nach einem großen Erfolg. Was hat dich motiviert, ein neues Projekt anzugehen?

André: Bei der Weiterentwicklung des CaSpAer habe ich festgestellt, dass die Technologie seit 1989 um einiges weiter gezogen ist. Diese Karte war schon nur 10x10cm groß, aber die Umsetzung auf 3.3V und SPI (Serial Peripheral Interface)-basierte Anbindung hat allein schon die halbe Karte ausgemacht. Die eigentliche Peripherie war noch viel kleiner. Daher war es schon lange auf meiner TODO-Liste, eine moderne PET-Replika zu bauen, die dann auch VGA-Bildschirme nutzen kann.

LOAD: Jetzt existieren ja bereits andere Projekte, die einen PET nachempfinden. Was unterscheidet den Micro-PET von diesen Geräten, zum Beispiel vom Mini-PET?

André: Nachdem der Mini-PET herauskam und ich ehrlich

erweise nicht von dessen Features überzeugt war, habe ich mich entschlossen, meinen Plan endlich in die Tat umzusetzen und den Micro-PET zu entwerfen. Der Micro-PET hat ein quell-offenes Design und kann von allen Interessierten weiterentwickelt werden.

LOAD: Hast du den Micro-PET allein entworfen oder hattest du Mitstreiter?

André: Der Micro-PET ist eine komplette Eigenentwicklung und komplett auf meinem Mist gewachsen. Mitstreiter hatte ich keine. Nicht etwa, weil ich nicht das wollte, ich habe es nicht als nötig angesehen. Ich hatte eine Vision davon, was ich bauen wollte und im Prinzip alle Ideen schon mal anderweitig ausprobiert, die musste ich nur kombinieren. Nur den VGA-Ausgang und das SPI-Boot habe ich in Revision 1 des Micro-PET getestet, mit parallelem ROM und Composite Video als Fallback für den Fall, dass es nicht geklappt hätte. Außerdem ist bei einem größeren Team die Gefahr vorhanden, dass man sich zu sehr in Diskussionen verzettelt. Andere Projekte haben darunter schon gelitten.

LOAD: Manche Projekte scheuen sich ja, ihre Entwicklungen quelloffen anzubieten. Sie befürchten, dass sich dann Dritte darüber hermachen und mit Bausätzen oder Fertiggeräten auf den Markt kommen. Wie stehst du dazu?

André: Ich würde mich sogar freuen, wenn jemand das aufgreifen würde und den Micro-PET fertig oder als (SMD-fertiges) Kit vertreiben würde, da es das Commodore PET "Feeling" weiter verbreiten würde.



Ein Clone des Git Repository ist auf der Heft-CD

Verein zum Erhalt klassischer Computer e.V.



Einer für Alle!

Ein Computer-Verein für alle klassischen Computer-Systeme? Na klar!

Egal ob Großrechner der 70er, Home-Computer der 80er oder PCs der 90er. Wir haben sie alle. Komm, mach mit und entdecke die faszinierende Welt der klassischen Computer bei uns im Verein!

Auszug aus den Computersystemen



Alte Computersysteme abzugeben? Wir sammeln und erhalten klassische Computer!



Anfassen, Ausprobieren, Spielen, Erinnern, Erhalten ...

Auf ins Forum!

In Vereinsforum diskutieren wir über dies und das, helfen bei Rechner-Problemen und haben eine gute Zeit!

Commodores unbekannter Rechner

Die Commodore MAX Machine

Seit vielen Jahren gilt sie als Rarität, denn nur eine geringe fünfstellige Anzahl wurde von ihr verkauft. Heute existieren daher entsprechend wenig Exemplare.

Für gut erhaltene Stücke, am besten mit vollständigem Zubehör und in Originalverpackung, zahlen Sammler derzeit zwischen 500 und 800 €. Die Rede ist von der Commodore Max Machine, die dieses Jahr ihren 40. Geburtstag feiert. Wer noch nie etwas von diesem geheimnisvollen Kleinod aus Commodores Produktportfolio gehört hat, braucht sich nicht wundern, denn dieses Gerät wurde nicht in Europa oder den USA verkauft. Einzig in Japan, wo der Rechner vom Commodore-Ingenieur Yashi Terakura entwickelt wurde, war das Gerät für ein knappes Jahr erhältlich. Die Markteinführung erfolgte Anfang 1982. Trotz des vergleichsweise niedrigen Verkaufspreises von 34.800 Yen (1982 ca. 340 DM oder 140 US-Dollar) hatte der Rechner leider keinen kommerziellen Erfolg. Er floppte grandios, denn der japanische Heimcomputer-Markt war 1982 stark von japanischen Spielekonsolen dominiert. Und obwohl die Max Machine eine vollwertige Konsole ist, hatte Commodore als US-Hersteller hier von Anfang an einen schweren Stand. Die Geräte blieben in den Regalen liegen und die spätere Markteinführung in den USA und in Europa wurde abgeblasen. Als Zeugnisse dieser Zeit sind noch Werbeanzeigen erhalten, die den VC10, wie die Max Machine in Deutschland heißen sollte, bewarben. Nur erreichten die angekündigten Geräte niemals die deutschen Geschäfte. Stattdessen startete bei uns das Weihnachtsgeschäft 1982 gleich mit dem Commodore 64 als neuem Heimcomputer-Flaggschiff.

Welche Geheimnisse stecken also in diesem Rechner, den Commodore Anfang 1982 in Japan im Spielekonsolen-Markt sowie als preisgünstige Alternative zum Commodore VC20 und in Konkurrenz zu Billig-Rechnern wie dem Sinclair ZX81 positionierte? Erscheinen sie heutzutage na-

hezu lächerlich, so waren im Erscheinungsjahr 1982 die technischen Werte der Max Machine für das angepeilte Niedrigpreissegment durchaus zeitgemäß: 2,5 kB RAM und dazu eine farbige Grafikausgabe durch den Grafikchip MOS6566. Dieser ist bis auf das Memory-Interface identisch zum kurz danach im Commodore 64 verbauten VIC-II Grafikchip. Ebenso feierte der legendäre Synthesizer Chip MOS6581, auch als SID bekannt, in der Max Machine seine Premiere und nicht etwa im etwas später erschienenen Commodore 64. Diese Tatsache ist heute selbst



eingefleischtesten SID-Fans kaum bewusst. Hinzu gesellt sich ein MOS6526 für die Abfrage von Joystick und Tastatur und natürlich der Prozessor MOS6510, der mit 1 MHz getaktet ist. Das RAM besteht aus einem 2 kByte SRAM und 0,5 kByte Farb-RAM. Eine PLA für die Adressdekodierung trägt die Bezeichnung MOS6703. Ein ROM ist nicht vorhanden. Aus technischer Sicht kann die Max Machine als Vorgänger des später erschienenen Commodore 64 gesehen werden. Sie ähnelt diesem in vielen technischen Details, wenn auch einige Dinge fehlen, wie beispielsweise der Anschluss für ein Diskettenlaufwerk.

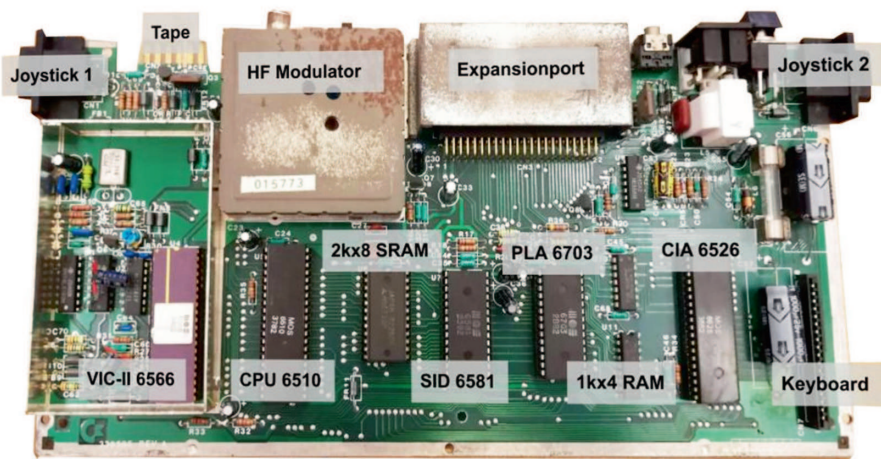
An Anschlüssen bietet die Max Machine

zwei Joystick-Ports und einen Antennenanschluss für den Anschluss an einen Fernseher sowie einen separaten Audio-Ausgang in Form einer 3,5 mm Klinkenbuchse. Im Gegensatz zur Konsolen-Konkurrenz ist aber auch ein Kassettenport vorhanden. Eine dort angeschlossene Datasette dient aber eher als Datenspeicher für die wenigen ernsthaften Anwendungen wie beispielsweise dem "Music Composer" denn als Quelle für Software. Spiele und andere Software wurden ausschließlich über Module ausgeführt, wie bei seinerzeit gängigen Konsolen auch. Dennoch macht der Datasetten-Anschluss und die eingebaute Folientastatur die Max Machine zu einem für damalige Verhältnisse vollwertigen Heimcomputer. Das hierfür unabdingbare Basic gibt es gleich in zwei Versionen, natürlich jeweils als Modul.

Das Mini-BASIC Modul bietet, im Vergleich zur Basic 2.0 Version des PET oder des VC20, einen reduzierten Funktionsumfang. So fehlen beispielsweise sämtliche Befehle zum Laden und Speichern auf der Datasette. Auch ist der verfügbare Programmspeicherplatz mit nur 510 Bytes sehr klein geraten. Daher ist diese BASIC-

Version mehr ein Spielzeug als eine wirklich sinnvolle Anwendung, zumal eingegebene Programme sich nicht einmal abspeichern lassen und jedes Mal frisch eingetippt werden müssen.

Die zweite BASIC-Version erschien auf dem Max-BASIC-Modul. Diese Version enthält den kompletten BASIC 2.0 Befehlsumfang, kann also auch die Datasette ansprechen. Programme können geschrieben, mit der Datasette abgespeichert und später wieder geladen werden. Dank einer im Modul eingebauten 2k RAM-Erweiterung, ist bei diesem Basic immerhin 2047 Byte Programmspeicher verfügbar. Dies ist in etwa die Hälfte dessen, was der Vorgänger VC20 bietet und somit immer noch nicht besonders großzügig bemessen. Der knappe Speicherausbau war offensichtlich vorwiegend dem Wunsch nach einem nie-



Oben: Die Hauptplatine der Max Machine
Unten: Die Anschlüsse auf der Rückseite

drigen Verkaufspreis geschuldet: Der MOS6566 VIC-II-Chip ist nicht in der Lage, den Refresh für kostengünstige dynamische DRAM-Bausteine zu erledigen. Dadurch war man, wie auch beim VC20, auf teures statisches SRAM angewiesen. Also wurde aus Kostengründen wohl so wenig wie möglich davon verbaut. Es soll dem Vernehmen nach bei Commodore heftige Diskussionen darüber gegeben haben, denn mit dem niedrigen Speicherausbau war der neue VIC-II-Chip nicht einmal mehr in der Lage, ein Bitmap-Bild im Speicher anzuzeigen. Doch am Ende setzte sich Commodore-

re-Chef Jack Tramiel durch und verlangte, den Speicher und damit die Kosten zu reduzieren. Erst mit der Weiterentwicklung des VIC-II-Chips zum MOS6567 konnte dann DRAM verwendet werden, was erstmals im Nachfolger Commodore 64 zum Einsatz kam. Von der Max Machine erbt dieser auch einen speziellen "Ultimax"-Modus, der sich über eine besondere Signalkombination am Expansionsport freischalten lässt. In diesem Modus ist der Commodore 64 weitgehend kompatibel zur Max Machine, sodass sich sämtliche existierenden Module für die Max Machine auch am Commodore 64 verwenden lassen. Und das sind, gemessen an der kurzen Marktpräsenz des Geräts, doch einige.

Etwa 20 Spiele und einige Anwendungen, vor allem aus dem Bildungsbereich, sorgten für Abwechslung. Viele der Module sind später auch fast unverändert für den Commodore 64 verkauft worden. Einige Spiele gelten bis heute als Klassiker: Commodores "Jupiter Lander", "Radar Rat Race", "Wizard of Wor" oder auch "Pinball" gab es für die Max Machine – diese Spiele sind bis heute unvergessen!

Heute finden sich sämtliche für die Max Machine erschienenen Module versammelt in

Ein Rechner, viele Namen

Die "Max Machine" wurde streng genommen nur in Japan so genannt. Für die USA war zunächst der Name "Ultimax" vorgesehen. Gerüchten zufolge soll dies aber zu sehr nach Hygieneprodukten geklungen haben, so dass man sich später auf "VIC-10" festlegte. Analog sollte das Gerät in Europa dann "VC10" heißen. Andere Namen sind "Commodore Max" und das eher liebevolle "Vicky", wie der Rechner bei seiner ersten Vorstellung in Tokyo genannt wurde.

dem 2014 erschienenen Multimax-Modul. Von einigen Spielen sind sogar mehrere Versionen abrufbar, die unter anderem zeigen, wie viel beziehungsweise wenig Änderungen Commodore bei einer späteren Commodore 64-Version vorgenommen hat. Da dieses Modul auch mit einem Commodore 64 funktioniert, bietet es einen interessanten Einblick in die Welt der Max Machine – auch für diejenigen, die nicht das Glück haben, eine solche zu besitzen (ab).

Links

- C64-Wiki:
https://www.c64-wiki.de/wiki/Max_Machine
- Wikipedia:
https://de.wikipedia.org/wiki/Commodore_Max,
Wikipedia (japanisch):
<https://ja.wikipedia.org/wiki/%E3%83%9E%E3%83%83%E3%82%AF%E3%82%B9%E3%83%9E%E3%82%B7%E3%83%BC%E3%83%B3>
- Secret Weapons of Commodore:
<http://www.floodgap.com/retrobits/ckb/secret/ultimax.html>
- Japanese vintage Computer Collection:
<https://monochromeffect.org/JVCC/2019/07/31/commodore-max-machine/>
- Michael Steil:
<https://www.pagetable.com/?p=1158>
- The Future was 8 Bit:
<https://www.thefuturewas8bit.com/maxmachine>
- Multimax Modul:
<http://www.multimax.co/>
- Technische Infos:
<http://www.zimmers.net/anonftp/pub/cbm/c64/html/ultimax.html>



Commodore VC 10 Der Kleine mit der großen Leistung !

Der neue Commodore VC 10 ist ein Video-Spiel-Computer „mit Intelligenz“ -für Heim und Hobby, Unterricht und Ausbildung, an jedes (Farb-) Fernsehgerät anschließbar. Der Commodore VC 10 ist der erste Video-Computer, der nicht nur zum Spielen geeignet ist, sondern frei programmierbar (in Mini-Basic) auch anderen Anwendungen offen steht.

Technische Daten: Speicherkapazität: 2,5 kByte RAM frei verfügbar · Programmiersprache: Mini-Basic · Zeichen, Zeilen: 40 Zeichen/Zeile, 25 Zeilen · Tastatur: großes Membran-Tastenfeld · Grafik: hochauflösende Farbgrafik · Sonstiges: Musiksynthesizer, an Lichtschreiber und Joystick anschließbar, Steckmodultechnik, Audio-Ausgang, an jedes Fernsehgerät anschließbar.

Best.-Nr. 14 09 53 Stück 498.-

Ankündigung des VC10 von 1982

Ueber den Autor

Andreas Beermann, Jahrgang 1969, ist Diplom-Ingenieur in der Halbleiter-Industrie und seit 2011 Mitglied im VzEkC e.V. Seinen ersten Commodore 64 kaufte er mit 14 Jahren. Schnell wurde daran kraeftig programmiert und geloetet, um die verborgenen Geheimnisse zu entdecken.

Commodore Max Machine wiederbeleben

Right to Repair

Nicht immer funktioniert ein Sammlerstück gleich auf Anhieb. Wenn es sich dabei noch um ein seltenes Gerät handelt, kann die Reparatur voll Tücke sein. Vereinsmitglied Andreas Beermann berichtet von seinen Erlebnissen mit einer Commodore Max Machine.

Einsam, von ihrem Original-Zubehör verlassen, ein bisschen schmutzig und leider auch defekt, so lag sie eines Tages vor mir: eine Commodore Max Machine. Ein befreundeter Sammler japanischer Heimcomputer hatte sie an mich weitergegeben. Er fand, dass dieses Gerät eines US-Herstellers nicht so recht zu seiner Sammlung passen würde. Er hatte es mehr oder weniger zufällig ergattert, als er die Webseite buyee.jp testen wollte. Über den Service dieser Seite ist die Teilnahme von Europa aus an japanischen Online-Auktionen möglich, beispielsweise an den in Japan sehr beliebten Yahoo-Auctions. Als ersten Test gab er das Mindestgebot für eine zufällig gerade zur Auktion gelistete Max Machine ab. Sie wurde ohne Netzteil, Verpackung und weiteres Zubehör sowie als defekt angeboten. Solche unvollständigen und defekten Geräte haben in japanischen Auktionen kaum eine Chance. So geschah es, dass er wider Erwarten mit seinem Erstgebot sogleich den Zuschlag erhielt. Der Versand nach Europa dauerte sehr lange, auch weil buyee.jp mehrere Käufe bündelt und erst bei einem ausreichend großen

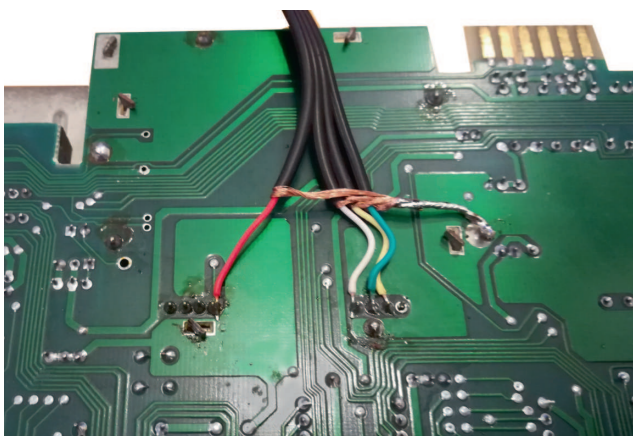
Versandvolumen die Ware verschickt. Doch was sind ein paar Monate Lieferzeit bei einem Gerät, das seit 40 Jahren darauf hofft, eines Tages in gute Sammlerhände zu geraten, um so der Verschrottung zu entgehen?

Natürlich konnte ich diesen aufregenden Neuzugang für meine Sammlung nicht in seinem bemitleidenswerten Zustand belassen. Also ging es darum, das Gerät wiederzubeleben. Zunächst zerlegte ich die Max Machine so weit, dass ich alle nicht-elektronischen Teile in warmem Seifenwasser spülen konnte. Die Tastatur wollte ich zunächst auch nass reinigen. Allerdings merkte ich schnell, dass die Folientasten offensichtlich auf einer saugfähigen Pappschicht angebracht sind. So wischte ich die Tastatur nur feucht ab, um sie gleich danach wieder gut zu trocknen. Bei der Reinigungsaktion fiel auch auf, dass einer der Gummifüße am Gerät fehlte. Zum Glück hatte ich noch nahezu identischen Ersatz in Form von selbstklebenden Gehäusefüßen aus dem Elektronikhandel vorrätig.

Die Stromversorgung des Gerätes stellte, trotz des fehlenden Netzteils, kein großes Hindernis dar: Die Max Machine benötigt dasselbe Netzgerät wie auch der Commodore 64. So konnte ich eines der zahlreich vorhandenen Geräte verwenden. Das hatte zugleich den Vorteil, dass es sich mit der in Deutschland üblichen Netzspannung betreiben lässt. Japanische Geräte sind für eine Netzspannung von nur 100 V ausgelegt, was am europäischen 230V-Stromnetz zwingend einen entsprechenden Vorschalttransformator voraussetzt.

Nachdem die Stromversorgung gesichert war, musste als nächstes der Videoausgang enträtselt werden. Die Max Machine verfügt nur über einen HF-Modulator-Ausgang. Aufgrund der japanischen Herkunft darf vermutet werden, dass hier ein japanisches NTSC-J Antennensignal anliegt. Ein Schalter am Modulator erlaubt die Auswahl von Kanal 1 oder 2. Im japanischen Kanalraster liegen diese Kanäle im Frequenzbereich von 90 MHz bis 102 MHz. Auf diesen Frequenzen liegt hierzulande aber der Bereich für das UKW-Radio. So hatte ich keine Hoffnung, mit meinem europäischen Fernsehgerät dieses Signal zu empfangen. Also musste eine andere Lösung her.

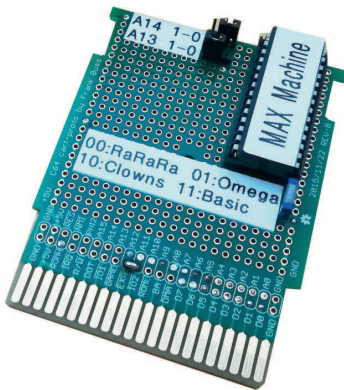
Im Netz kursiert der sogenannte One-Wire-Mod für die Max Machine. Bei dieser Modifikation wird eine Drahtverbindung vom Modulator auf die eigentlich für das Audiosignal gedachte 3,5mm Klinken-



Die S-Video-Modifikation (grün/gelb), hier noch ergänzt um Composite Video (weiß) und Audio (rot)



Das Netzteil eines Commodore 64 passt auch an die Max Machine.



Das selbst gebaute Steckmodul, mit Lötbrücke zwischen den Kontakten "EXR" und "IO2", liefert erstes Futter für die Max Machine.

buchse gelegt. Auf diese Weise liegt an der Buchse zusätzlich zum Audiosignal auch ein Composite-Video-Signal an. Mit einem entsprechenden Adapterkabel (3,5 mm Stereo-Klinke auf 2x Cinch) lässt sich so bequem das Videosignal von außen abgreifen, ohne dass zusätzliche Buchsen am Gehäuse angebracht werden müssen. So elegant diese Modifikation auch ist, so bescheiden ist leider die damit zu erzielende Bildqualität. Auch die Tonqualität leidet unter dem Übersprechen des Videosignals in das Audiosignal.

Also entschied ich mich für eine bessere Lösung: Ein S-Video-Kabel wird auf einer Seite von seinem Stecker befreit und direkt an die entsprechenden Pins des Modulators auf der Unterseite der Hauptplatine gelötet. Das Kabel muss dann durch eine Gehäuseöffnung, also z.B. durch den Kassettenport, nach außen geführt werden. Der Anschluss erfolgt schließlich an der S-Video-Buchse des Fernsehers. Dieser sollte NTSC-tauglich sein, damit das Bild farbig dargestellt wird. Die so erzielte Bildqualität ist hervorragend, deutlich besser als über den HF-Modulator-Ausgang, was ich nach Fertigstellung dieser Modifikation feststellen durfte: Viel zu spät entdeckte ich, dass sich mein Fernsehgerät per Ländereinstellung auf "Japan" stellen lässt. Damit ist es dann eben doch in der Lage, das HF-Signal der Max Machine zu empfangen. Dieser Fehler wäre einem erfahrenen Sammler japanischer Rechner wohl nicht passiert.

Wie dem auch sei, leider lieferte meine Max Machine auch mit diesem Kabel zunächst noch kein Bild. Bei der Fehlersuche wurde mir erst nach einem Blick in den im Netz verfügbaren Schaltplan klar, dass das Gerät über keinerlei eingebautes ROM verfügt. Ohne Steckmodul würde die Max Machine also folglich – wie zu erwarten – keinen Mucks von sich geben. Also musste ein geeignetes Steckmodul her. Auf die Schnelle ging das nur, indem ich auf Basis einer Prototypen-Platine für den mechanisch kompatiblen Expansionsport des

Commodore 64 ein ROM-Modul selbst baute. Das auf dem Modul verwendete EPROM hatte Platz für vier der einfacheren Multimax-Spiele und ich entschied mich für "Radar Rat Race", "Omega Race", "Clowns" und aus Gründen des leichteren Debuggens noch "Mini-Basic". Die Programme ließen sich jeweils über Steckjumper auswählen. Mit diesem Modul gelang es mir dann zum ersten Mal, der Max Machine ein Bild zu entlocken.

Doch die Freude währte nicht lange. Weder Tastatur noch Joysticks schienen zu funktionieren. Der Verdacht fiel sofort auf den CIA MOS6526, der in der Max Machine für die Tastatur- und Joystick-Abfrage zuständig ist. Tauschen gegen einen funktionierenden CIA brachte aber keinen Erfolg. Der Test des CIA aus der Max Machine in einem Commodore 64 ergab, dass der Baustein noch funktionierte. Erst nach längerem Suchen im Schaltplan bemerkte ich, dass das Chip-Select-Signal des CIA bei der Max Machine über zwei Pins am Expansionsport geschleift wird: Eine Pinbelegung, die sich vom Commodore 64 unterscheidet und daher von mir zunächst nicht bedacht wurde. Auf meinem selbst gebauten Modul half nun eine Lötbrücke zwischen den Pins EXROM und IO2, um das Signal entsprechend zu verbinden. Kaum war diese vorhanden, nahm der CIA seine Funktion auf und die Joysticks funktionierten nun. Mit der Tastatur gab es allerdings immer noch ein Problem, denn einige Tasten streikten weiterhin. Ein Verfolgen der Leiterbahnen auf dem Folienleiter der Tastatur ergab eine Unterbrechung in einer Leiterbahn. Diese war leicht mit etwas Leitsilber zu reparieren. Danach war nun auch die Tastatur funktional.

Doch das war immer noch nicht das Ende des Reparaturmarathons. Als ich die Spiele auf meinem Selfmade-Modul ausprobierte, war kein Ton zu hören. Weder

über die 3,5 mm Klinkebuchse, noch über das von mir gleichzeitig mit dem S-Video-Kabel direkt angelötete Audiokabel kam ein Signal. Durch Tausch wurde schnell klar, dass der MOS6581 SID defekt war. Ersatz wurde zunächst durch einen ohnehin vorhandenen FPGASID bereitgestellt. Aus Gründen der Originalität kam jedoch später dann ein originaler SID zum Einsatz, mit einem Datums-Code passend zum defekten SID.

Nach dieser doch etwas längeren Reparatur ist die Max Machine nun endlich wieder vollständig funktionsfähig. Einzig das aus dem Gehäuse baumelnde Video-Kabel stört ein bisschen die Optik. Womöglich sollte ich dies aus Gründen der Originalität doch wieder entfernen und das Gerät ausschließlich über den HF-Modulator betreiben. Auch sind einige Blechteile im Inneren des Gerätes etwas korrodiert, was sich ebenfalls noch verbessern ließe. Aber ein bisschen sollte man dem Gerät sein Alter ja auch ansehen und so werde ich die Korrosion zwar weiter beobachten, aber im Moment noch nicht beheben (ab).

Links

Scan des Schaltplans (kaum lesbar):
<http://www.zimmers.net/anonftp/pub/cbm/schematics/computers/c64/326100.png>
 Schaltplan reverse engineered von Donato Travaglini (leider in Teilen fehlerhaft):
<http://www.zimmers.net/anonftp/pub/cbm/schematics/computers/c64/maxschematic.jpg>
 Schaltplan von Ruud Baltissen (entspricht nicht meiner Platine-Revision)
<http://www.zimmers.net/anonftp/pub/cbm/schematics/computers/c64/ultimax.gif>
 Der One-Wire-Mod:
<http://www.6502.org/users/sjgray/projects/maxvid/index.html>
 Einkaufen in Japan:
<https://buyee.jp>
 Online-Auktionen in Japan:
<https://auctions.yahoo.co.jp>



Der Lohn der Mühen – die Max Machine in Aktion!

PC Emulatoren für Apple Macintosh

Einen DOS-Mac, bitte!

Wie bei allen anderen Hardware-Emulatoren für Computer wie dem Atari ST oder Commodore Amiga steht auch beim Macintosh am Anfang die Frage: „Warum muss der Mac PC kompatibel werden?“. Wir versuchen, eine Antwort zu geben.

Mehr noch als bei den anderen beiden beschriebenen Plattformen haben sich die Rechner aus Cupertino seit dem Ende der 1970er Jahre sehr gut verkauft. Sie haben mit ihrer Softwareausstattung wie beispielsweise Visicalc und AppleWorks selbst Maßstäbe gesetzt. Allerdings entstand beim Wechsel auf die 68k-Plattform eine immer größer werdende Lücke zur IBM-kompatiblen Welt, die irgendwie gefüllt werden musste.

MacCharlie

Bereits im Jahr 1985 entwickelte die Firma Dayna Communications einen PC-Emulator, den MacCharlie. Ähnlich wie das Commodore A1060 Pendant wird dieser seitlich an den Macintosh 128k, 512k oder Plus angeflanscht. Die externe Box enthält eine komplette PC-Hardware inklusive der Intel 8088 CPU mit einer Taktfrequenz von 4,77 MHz, 256 KByte bis 640 KByte RAM und ein 5,25"-DD-Laufwerk mit einer Speicherkapazität von 360 KByte. Die Kommunikation zwischen beiden Plattformen erfolgt über zwei 9-polige Anschlüsse des MacCharlie an den Drucker- und Modemanschluss des Mac. Der Mac dient so als Konsole für den PC-Teil, lauffähig sind also nur textbasierte DOS-Programme. Darüber hinaus konnten auch Dateien zwischen den beiden Welten ausgetauscht und die beiden getrennten Echtzeituhren synchronisiert werden. Da das Keyboard der 128k-, 512k- und Plus-Macs nicht alle Tasten zur Verfügung stellt, lieferte Dayna einen Keyboard-Extender mit, der die Funktionstasten sowie den Ziffernblock bereitstellt. Das System ist zum einen recht langsam, zum anderen war es aber auch recht teuer und fand folglich keine große Verbreitung.

Mac SE und Mac II mit DOS-Füllung

1987 stellte Apple die ersten intern erweiterbaren Modelle Macintosh SE und Macintosh II vor. In Zusammenarbeit mit Apple entwickelte Phoenix Technologies in der Folgezeit zwei PC-Emulatoren, die von der Firma AST Research ab 1988 vermarktet wurden. Für den Macintosh SE entstand so die Mac86-Erweiterung für den Processor Direct Slot (PDS). Die Karte hat einen 8086-Prozessor mit 7,86 MHz, bringt aber keinen eigenen Speicher mit. Die RAM-Größe wurde über die Application Size im Finder eingestellt. Die Maximalkonfiguration von 640 KByte für MS-DOS erfordert dort einen Wert von 1.300 KByte.

Das interne Diskettenlaufwerk des Mac ist nicht direkt zu nutzen, deshalb wird zwingend ein passendes externes Diskettenlaufwerk für den 37-poligen Anschluss benötigt. Bei Installation der mitgelieferten Software wird eine 10 MByte große Datei als Festplattendatei (Hardfile) mit MS-DOS 3.2 angelegt. Wie auch der MacCharlie hat auch die Mac86 keine Grafikausgabe, ist also auf textbasierte DOS-Programme beschränkt. Die Konsole wird direkt im Mac OS als Fenster eingeblendet und bietet 80x25 Zeichen.

Unter MS-DOS funktioniert die Maus und auch ein Zugriff auf den Mac-Lautsprecher klappt. Um serielle Geräte unter MS-DOS nutzen zu können, gibt es zwei emulierte serielle Schnittstellen (COM1 und COM2), die entweder auf den Drucker- oder den Modemanschluss umgeleitet werden können.

Anders gestrickt war die Mac286 des gleichen Herstellers. Hierbei handelt es sich um eine Steckkarte für den Nubus-Erweiterungssteckplatz, der ab dem Macintosh II über viele Jahre von Apple verwendet wurde. Auch bei dieser Karte finden sich erstaunliche Ähnlichkeiten zur Amiga-



MacCharlie am Macintosh

Parallelwelt. Auch die Mac286 ist wie die Commodore A2286 eine Karte mit zwei Platinen, diese sind allerdings mit zwei Kabeln verbunden. Auch die AST nutzt einen Intel 80286 mit 8 MHz Takt, hat 1 MByte RAM auf der Karte und kennt die Möglichkeit, eine FPU einzusetzen. Obwohl durch SIMMs realisiert, lässt sich der Arbeitsspeicher bei der originalen Karte von AST nicht auf 4 MByte erweitern. Als weitere Verbesserung kann entweder das Laufwerk des Host-Macs oder aber (wie bei der Mac86) ein externes Laufwerk über den 37-poligen Anschluss als Laufwerk A: konfiguriert werden. Dabei kommt ein NEC 765 basierter Floppy-Controller zum Einsatz. Im Gegensatz zu den meisten PCs akzeptiert das Laufwerk mit entsprechenden Tools auch Single Density-Disketten aus der CP/M-Welt. Beim Installationsvorgang wird ein 20 MByte großes Hardfile als Festplatte für die PC-Karte angelegt, MS-DOS 3.30 ist darauf vorinstalliert. Diese Karte macht es auch möglich, endlich graphische Anwendungen auf der Emulation zu nutzen, denn sie bietet neben dem Textmodus auch eine CGA-Emulation an. Die weiteren Einstellungsmöglichkeiten sind mit der Mac86 vergleichbar. Auch die Bildschirmdarstellung ist als reiner Bild-im-Bild-Modus realisiert. AST Research zog sich 1989 aus dem Apple-Geschäft zurück und verkaufte



Der AST Mac286-Emulator

Orange Micro Karten

Serie 200

Orange Micro fertigte vier verschiedene Modelle dieser Baureihe. Die einfachste Variante ist die OrangePC 210, die mit einer Standard-VGA-Grafik sowie serieller und paralleler Schnittstelle am Expansionskabel ausgeliefert wurde. Die OrangePC 220 bietet darüber hinaus auch SVGA-Grafik, ebenso wie die folgenden Modelle. Auf dem Modell OrangePC 250 ist ein PCMCIA Slot (CL-PD6710) verbaut, um so zum Beispiel Netzwerk- oder Soundkarten nutzen zu können. Dafür wurde aber auf die serielle und parallele Schnittstelle verzichtet. Das Top-Modell OrangePC 290 beinhaltet neben dem PCMCIA Interface auch wieder diese beiden Schnittstellen und zusätzlich 128 KByte 20-ns-Cache für die CPU.

Serie 300

Während die Nubus-Karte 340 die finale Entwicklungsstufe der OrangePC-Karten für die Nubus-Macs darstellt, waren die beiden PCI-Karten 420 und 440 die ersten PC-Emulatoren von Orange für den neuen Erweiterungsbus in den Power Macs. Soweit es in

Erfahrung zu bringen war, gibt es kein Nubus-Pendant zur OrangePC 420. Diese Version bietet neben einer 1 MByte Grafiklösung (S3 Trio64) einen Speichersteckplatz für PS/2-DIMMs mit maximal 32 MByte Arbeitsspeicher. Eine CPU-Cache ist bei der „kleinen“ neuen PCI-Karte nicht verbaut. Die Nubus 340 und die PCI 440 ermöglichen es dem Nutzer, den Grafikspeicher auf 2 MByte aufzurüsten. Sie stellen zwei PS/2-Slots für bis zu 64 MByte RAM zur Verfügung. Beide Karten sind mit 256 KByte Cache (20ns) für die CPU ausgestattet. Das Modell 340 kommt mit 3,3 V und 5 V 80486 CPUs wie den DX2- und DX4-CPU's, aber auch dem Cyrix 5x86 zurecht. Für die beiden 400er Boards findet sich auf der Liste der unterstützten CPUs nur der Cyrix DX2-80 sowie der Cyrix 5x86 mit 120 MHz. Dies mag daran liegen, dass die 400er Boards den FSB der Karte mit 40 MHz takten. Für die nun endlich implementierte SoundBlaster-kompatible Tonausgabe sorgt ein Chip von ESS.

Serie 500

Bei der Orange PC 520 ist eine 1 MByte Grafikkarte (S3 Trio64V+), erweiterbar auf

2 MByte verbaut, sie wurde häufig mit einer Cyrix 686+ CPU mit 166 MHz angeboten. Die 530 hat bereits 2 MByte Grafikspeicher verlötet und benötigt nicht den vorgenannten Zwischensteckplatz für MMX-CPU's. Der Arbeitsspeicher ist bei beiden 7"-Karten durch einen EDO-DIMM (5V) mit bis zu 128 MByte realisiert, ein Cache-RAM für die CPU ist nicht vorgesehen.

Die OrangePC 540 als erstes 12"-Modell dieser Reihe hat eine 2 MByte Grafikkarte (S3Trio 64V+), zwei DIMM-Steckplätze für bis zu 256 MByte EDO-RAM sowie 256 KByte Cache-RAM, um die CPU zu beschleunigen. Bei den drei bisher genannten Modellen ist eine 16-bit SoundBlaster-kompatible Audiolösung (ESS-Chip) verbaut. Bei der 550, der Top-Karte aus der 500er Serie, änderte Orange Micro so einiges: Der Hersteller stieg auf den S3-Virge Grafikchip um und spendierte diesem 4 MByte SG-RAM. Darüber hinaus wurde der Cache-RAM auf 512 KByte erweitert und der verwendete ESS-Audiochip bot 3D-Effekte und Wavetable.

Serie 600

Die 620 hat einen SIS 5598-Chipsatz mit integrierter Grafik, die neben 2D-Beschleunigung zumindest eine 3D-Emulation anbietet. Da der Chipsatz nur mit einem FSB bis 75 MHz (mit Übertaktung 83 MHz) arbeitet, sind bei diesen Karten alle CPUs mit 100 MHz FSB nicht verwendbar. Auch auf einen 2nd-Level-Cache wurde verzichtet. Laut Handbuch von Orange Micro war keine Installation von Windows NT möglich, ganz im Gegensatz zu den anderen PCI-Karten.

Das mit Abstand leistungsfähigste Modell ist die Ende 1998 eingeführte OrangePC 660. Mit ihrem VIA VT82C598-Chipsatz ermöglicht sie den Einsatz quasi aller Sockel7-CPU's bis zu 450 MHz und die Verwendung von zwei SD-RAM Modulen für einen maximalen Speicherausbau von 256 MB. Das große Highlight dieser Karte ist aber der Grafikchip nVidia Riva 128 mit 4 MByte SG-RAM. Kein Wunder, dass sie als Spiel-Lösung für den Mac vermarktet wurde. Die PCfx, auch 650 genannt, ist eine vereinfachte 660 mit einem RAM-Slot und wurde mit einer nicht austauschbaren Pentium-kompatiblen CPU (möglicherweise IDT Win-Chip C6 mit 200 MHz) angeboten.

te seine Rechte und technischen Unterlagen über die beiden PC-Karten an Orange Micro.

DOS-Mac mit Orangenaroma

Nach der Übernahme entwickelte Orange Micro die Mac286 weiter und ermöglichte als erstes die Verwendung von 1 MByte SIMMs, so dass ein Speicherausbau von insgesamt 4 MByte möglich wurde. In der finalen Weiterentwicklung wurde die Mac-286 auf eine Steckkarte reduziert. Die Karten laufen mit Taktfrequenzen von 12 und 16 MHz. Inzwischen vollzog sich in der IBM PC-Welt der Wechsel von den 80286er PCs hin zu den Maschinen mit Intel 80386. Auch Orange Micro entwickelte daraufhin seine PC-Karte weiter und präsentierte die Orange386. Bereits die erste Version hat zwei ISA-Slots (einmal 8 Bit, einmal 16 Bit) und ermöglicht einen Speicherausbau mit vier 30-poligen SIMM-Modulen auf maximal 16 MByte. Die 386SX-CPU ist mit 16 oder 25 MHz getaktet. Ohne Grafikkarte in einem der ISA-Slots bietet die Karte nur die bekannte langsame CGA-Emulation. Über den optionalen externen Anschluss stellt die erste Version des sogenannten Oktopus-Kabels einen Anschluss an die serielle und parallele Schnittstelle sowie den Floppy-Port bereit.

Auch Orange Micro verstand, dass diese Karte nur mit einer VGA-Karte wirklich nutzbar ist. Daher wurde bei späteren Modellen ein WDC WD90C30 SVGA-Chip integriert, im Gegenzug bei einer beiden ISA-Slots gestrichen. Die letzte Ausbaustufe

dieses Modells ist mit dem TI486SLC Prozessor ausgestattet, der mit 25 MHz läuft und Pin-kompatibel zur 386SX CPU ist.

Nächste Evolutionsstufe für den Nubus-Steckplatz sind die OrangePC-Karten der 200er Serie. Diese bieten nun zusammen mit einem OPTI-Chipsatz Platz für leistungsfähigere 486er CPUs mit 3,3 oder 5,0 V Spannungsversorgung. Angefangen beim 80486 SX-16 bis hin zum 80486 DX4-100 läuft auf dieser Karte beinahe jede CPU. Die 80486 DX4-100 CPU arbeitet allerdings nur mit den Kartenmodellen OrangePC 250 und OrangePC 290 zusammen, bei letzterer auch nur nach Modifikation. Als Arbeitsspeicher kann ein PS/2-Modul mit bis zu 32 MByte verbaut werden, laut vorliegendem Handbuch sogar bis 64 MByte.

Bei allen vorgestellten Modellen muss das Videosignal der PC-Karte immer über den Nubus Steckplatz an den Macintosh weitergegeben werden. Dieser kann es dann am Monitor ausgegeben. Es liegt in der Natur der Sache, dass dadurch die Bildausgabe recht langsam ist und zumindest zum Spielen nicht ausreicht.

Den Wechsel von der Nubus- hin zur PCI-Plattform vollzog Orange Micro mit den ungleichen 12"-Zwillingen der Baureihen 300 und 400 mit VIA 82C496-Chipsatz. Als weitere Neuerung gegenüber den Nubus-Karten der 200er Serie verfügt diese Zwillingsserie zum ersten Mal über das sog. Hydra- bzw. Oktopus-Kabel, welches zwei serielle und eine parallele Schnittstelle, einen Game- beziehungsweise Midi-Port sowie je einen Stereo-Klinkenstecker für Tonein-

und -ausgang sowie die Durchschleifmöglichkeit für das Bildsignal zur Verfügung stellt. Alle PCI Versionen der OrangePC-Karten können keine weiteren PCI-Erweiterungskarten nutzen.

Die nachfolgende 500er Serie wurde für den Intel Pentium und kompatible CPUs entworfen und repräsentiert so die erste Sockel7-Baureihe. Die externen Schnittstellen sind wie bei der 300er und 400er über das sogenannte Hydra- oder Oktopus-Kabel nach außen geführt. Der Hersteller unterscheidet hier zwischen einer 7"- und einer 12"-Version. Wie gewohnt ist die CPU gesockelt und kann daher leicht gegen andere unterstützte CPUs getauscht werden. Für die 2,8 V Stromversorgung für MMX-CPU's ist bei den Kartenmodelle 520 und 540 ein Zwischensteckplatz notwendig.

Die Spitze der PC-Emulatoren im Mac bilden definitiv die Karten der 600er Serie von Orange Micro für Sockel7-CPU's, die im Jahre 1998 vorgestellt wurden. Wie bei den vorher genannten 500ern gibt es ein 7"-Modell (620/625) und zwei 12"-Modelle (PCfx/PCfx 655 bzw. 660/665) wobei die Karten mit der 5 am Ende der Modellnummer für die neuen G3-Power Macs im bunten Blue-and-White-Tower modifiziert sind und ansonsten den Modellen mit der 0 am Ende entsprechen. Alle Karten schlucken SD-RAM als Arbeitsspeicher und durch die umfangreichen Einstellungsmöglichkeiten laufen eine große Bandbreite von CPU's, angefangen beim Pentium mit 100 MHz bis hin zum AMD K6-III mit 450 MHz. Orange Micro verabschiedete sich bei diesen Modellen auch vom Oktopus-Kabel, da die an-

Übersicht zu den wichtigsten PC-Emulatorkarten für den Macintosh

Karte	CPU	Taktfrequenz	RAM	Grafik	L2-Cache	Dhrystones/s	MIPS
AST Mac86	Intel 8086	7,86 MHz	640 kB	MDA	-	379,7	0,5
AST Mac286	Intel 80286	8 MHz	1 MB	MDA/CGA	-	1 551,0	2,0
Orange Micro OrangePC 386	AMD 80386SX	25 MHz	4 MB	WDC WD90C30	-	4 257,6	5,5
Orange Micro OrangePC 220	AMD 5x86	120 MHz	16 MB	WDC WD90C30	-	19 785,2	41,1
Orange Micro OrangePC 520	Intel Pentium	200 MHz	64 MB	S3 Trio64V+	-	113 660,0	88,6
Orange Micro OrangePC 620	AMD K6-III	400 MHz	128 MB	SIS 5598	-	314 516,5	164,3
Apple Houdini I	Intel 486SX	33 MHz	16 MB	C&T	-	14 095,3	11,5
Apple Houdini II	Intel 486DX2	66 MHz	32 MB	C&T	-	25 176,2	23,0
Apple 7" 100 MHz	Cyrix 5x86	100 MHz	64 MB	ATI Mach64	128 kB	37 849,5	40,0
Apple 12" 100 MHz	Intel Pentium	100 MHz	72 MB	ATI Mach64	256 kB	61 998,8	4,4
Apple 12" 166-P	Intel Pentium	166 MHz	80 MB	ATI Mach64VT	256 kB	97 338,6	73,9

Alle Geschwindigkeitstests wurden mit PC-Doktor 1.70 durchgeführt

gebotenen Schnittstellen und die Funktion der Soundkarte nun per Software-Emulation im Mac gelöst wurden. Die seriellen und parallelen Ports des Emulators können wie schon bei der Mac286 den Modem- oder Printer-Port des Macs nutzen. Somit bleibt die Möglichkeit, das Videosignal des Macs über einen VGA-Ein- und -Ausgang durchzuschleifen.

Bei den letzten OrangePC Nubus-Karten von sowie bei den PCI-Karten können neben dem Diskettenlaufwerk (als Image oder physikalisch) auch bis zu zwei Dateien („Hardfiles“), als Festplatte eingebunden werden. Auf- und Umsteigern ist es möglich, vorhandene Hardfiles von den Software-Emulatoren SoftPC, SoftWindows und Virtual PC oder von den Hardware-Vorgängern Mac286 und Orange386 zumindest als zweite Festplatte D: zu verwenden. Darüber hinaus sind bis zu vier Shared Volumes (Macintosh Ordner oder andere Geräte wie ganze Festplatten oder optische Laufwerke) möglich. Ebenso ist beispielsweise ein ZIP-Drive als ATAPI- oder SCSI-Device einzubinden. Auch schaffte es Orange Micro, 32 Bit-Treiber für Festplatte, CD-ROM und Diskettenlaufwerk für die Windows 9X-Familie bereitzustellen. Zumindest ab der 200er-Serie kann auch die Netzwerkschnittstelle des Mac genutzt werden. Der Mac selbst sollte bis einschließlich der 500er Serie über mindestens 2 MByte RAM, für die Modelle der 600er Serie besser 16 MByte RAM verbaut haben.

Ein DOS-Mac direkt vom Hersteller

Und was machte der Mac-Hersteller selbst? Commodore war auch Hersteller von PC-Emulatoren für den Amiga und hat quasi den Startschuss selbst gegeben. Ap-

ple hat zumindest AST als auch Orange weitreichend über die Interna der Macintosh Computer informiert, sodass eine gute Integration gelang.

Die Konfiguration der Apple-eigenen PC-Emulationskarten erfolgt über die PC-Setup-Software. Die seriellen Schnittstellen werden emuliert und auf den Modem- oder Printer-Port des Macs umgeleitet. Als Festplatte für die PC-Welt dient auch hier ein sogenanntes Hardfile, es sind bis zu zwei als Laufwerke C: und D: möglich. Diese können auch von den Software-Emulatoren SoftPC und SoftWindows stammen. Darüber hinaus ist noch ein Austauschordner mit dem Mac möglich, der dann beiden Systemen zur Verfügung steht. Auch ein im Mac vorhandenes CD-ROM-Laufwerk wurde eingebunden. Der Zugriff auf ein Netzwerk klappt ebenfalls, allerdings nicht gleichzeitig – nur ein System, also der Mac oder der PC, vermag eine Netzwerkverbindung herzustellen.

Im Oktober 1993 war es dann endlich so weit – Apple brachte mit dem Quadra 610 seinen ersten Computer auf den Markt, der vom Werk aus PC-kompatibel ausgestattet war. Realisiert wurde das mit Hilfe der Houdini I genannten Erweiterungskarte für den PDS-Erweiterungslot des Rechners mit 68040 CPU. Auf der Houdini I ist ein gesockelter Intel 486SX-Prozessor mit 25/33 MHz verbaut, zusammen mit einem PC-Chipsatz und einer VGA-Karte von Chips and Technologies mit 512 KByte Speicher. Der notwendige Arbeitsspeicher wird entweder vom Mac selbst zur Verfügung gestellt oder besser durch einen zusätzlich verbauten PS/2-Riegel mit bis zu 32 MByte. Eine SoundBlaster-Karte lässt sich über einen kleinen Steckplatz einbauen. Als Betriebssystem war DOS 6.22/Win3.1 vorgesehen, aber auch Windows 95 läuft mit der Karte. Das Video-Signal

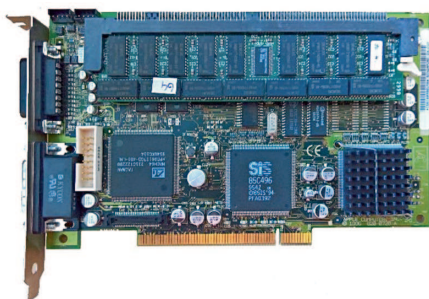
des Mac wird über ein spezielles Kabel durch die PC-Karte durchgeschleift. So klappt ein Umschaltung zwischen Mac und PC. Wichtig war Apple auch die Anschlussmöglichkeit für einen PC-Joystick mit Hilfe dieses Kabels – das zeigt durchaus den propagierten Verwendungszweck.

Die leicht modifizierte Houdini II Karte wurde mit einem anderen L-Adapter (angepasst für den PowerPC-601-PDS-Slot im Power Mac 6100) ab März 1994 mit einer DX2-66 CPU verkauft. Sie kann auch, wenngleich nicht offiziell von Apple unterstützt, im Power Mac 8100 und Quadra oder Centris 610 eingebaut werden. Im Juli folgte dann mit dem Performa 630 bzw. LC 630 eine LC-PDS-Slot-Variante, die ebenfalls einen 486DX2-66 mitbrachte.

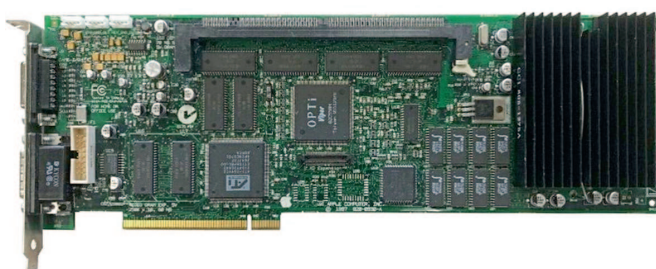
Mit dem Umstieg auf die PCI-Architektur, der ab dem Power Mac 7200 vollzogen wurde, wechselte Apple für seine PC-Karten ebenfalls auf diesen Erweiterungssteckplatz. Im April 1996 kam die 7“-Karte mit einem 100 MHz Cyrix 5x86-Prozessor, 128 KByte L2-Cache und SIS-Chipsatz, sowie eine 12“-Karte mit Intel Pentium 100 CPU, 256 KByte L2-Cache und OPTI Viper-Chipsatz auf den Markt. Die kleinere Variante wurde mit einem 8-MByte-DIMM ausgeliefert, auf der größeren sind 8 MByte onboard verlötet. Beide Karten haben einen ATI Mach64- Grafikkchip mit 1 MByte (7“-Karte) oder bis zu 2 MByte (12“-Karte) VRAM sowie einen 16-Bit SoundBlaster Vibra 16S Soundchip. Apple behielt die Lösung mit einem Durchschleifkabel bei. Allerdings wurde der Joystick-Port des Kabels entfernt, da die Karten selbst einen solchen Anschluss vorsehen.

Die PCI-PC-Karten waren für den Betrieb mit MS-DOS 6.22 und Windows 3.1 oder Windows 95 vorgesehen. Letztlich läuft auf ihnen aber auch Windows 98 problemlos, wenngleich es für Windows 9X-Systeme keine 32 Bit-Treiber für Festplatte, CD-ROM und Disklaufwerk gibt. Windows NT und auch OS/2 wurde zu keiner Zeit von den Apple-PC-Karten unterstützt. Auch können die PC-Emulatoren bis auf eine Ausnahme keine PCI-Erweiterungskarten ansprechen. Von Apple selbst gab es eine Karte mit einer seriellen und einer parallelen Schnittstelle, die aber mit einem zusätzlichen Kabel an den PC-Karten angeschlossen werden musste.

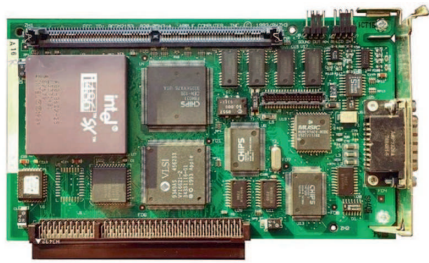
Für den Power Mac 4400 (bzw. 7220) und Power Mac 7300 brachte Apple dann ein Jahr später zwei überarbeitete Versionen der 12“-Karte heraus. So ist als Grafikkchip ein ATI Mach64VT mit 2 MByte Speicher sowie 16 MByte Onboard-RAM



Apple 7" PC-Compatible Card



Apple 12" PC-Compatible Card



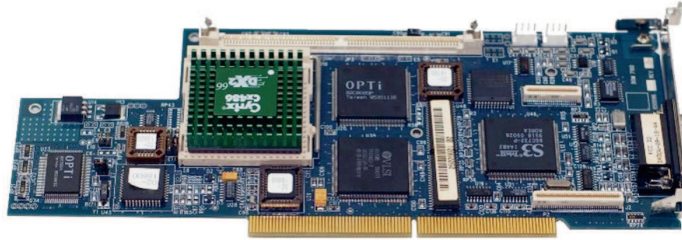
Apple Houdini-II Card

verbaut. Für den 4400er kommt ein Cyrix PR166 Prozessor mit 133 MHz zum Einsatz. Diese Karte ist speziell nur für dieses Modell designed, da laut Apple das Netzteil der anderen Macs keine ausreichenden Reserven bieten. Die maximale Ausbaustufe, quasi die Krönung der herstellereigenen PC-Karten ist die Variante für den 7300er mit 166 MHz Intel Pentium CPU. Die 166er Pentium-Karte ist das finale Meisterstück, denn Apple ließ mit dieser Erweiterung die Hardware-PC-Kompatibilität auslaufen. Zumindest die PCI-PC-Karten wurden auch als sogenanntes „PC compatible upgrade kit“ verkauft.

Aufgewärmtes von Reply und Radius

Der Hersteller Reply lizenzierte nach dem Erscheinen der Houdini II-Karten die Technologie von Apple, die den PC-Karten zugrunde lag. Dies zeigt sich unter anderem auch daran, dass zum Durchschleifen des MAC- Videosignals die gleichen Kabel wie bei den entsprechenden Apple-Karten verwendet werden. Die Einstellungen, die die Setup-Software zulässt, sind ähnlich zu den Apple-Karten. Allerdings ist es beiden Welten (dem Mac-Host und dem PC-Gast) möglich, durch das extra zu erwerbende NetWork Pack gleichzeitig auf die Netzwerkschnittstelle zuzugreifen. Bis auf den Houdini II Klon können alle Karten auch eine PCI-Schnittstellenkarte nutzen, die ebenfalls mit den Emulatoren per Kabel verbunden werden muss. Im Gegensatz zu den originalen Apple-Karten können die LC-PDS und PCI-Karten neben MS-DOS 6.22 und Windows 3.1 und Windows 95/98 (ebenfalls ohne 32 Bit-Treiber für Festplatte, CD-ROM und Diskettenlaufwerk) auch Windows NT als Betriebssystem nutzen.

Im Dezember 1995 brachte Reply eine sog. „DOS on Mac“-Karte für den PDS-Slot des Power Mac 7100 und 8100 heraus. Diese Karten bieten einen Sockel3 für 486DX2 und DX4-Prozessoren sowie den Cyrix 5x86 in Verbindung mit einem Opti-Chipsatz, wobei kein L2-Cache vorgesehen war. Es können bis zu 64 MByte Arbeitsspeicher in Form eines PS/2-SIMMs verbaut werden. Reply verwendet den S3 Trio32 mit 1 MByte RAM als Grafikchip. SoundBlaster-kompatiblen Sound gibt es



Reply DOM 7100 Card

nur mit einem Zusatzmodul, das direkt auf der PC-Karte Platz findet. Beide Karten bauen letztlich auf dem gleichen Grunddesign auf, wobei sie in ihrer jeweiligen Form auf die Vorgaben des entsprechenden Macs zugeschnitten sind. Zusätzlich war auch die Houdini II-Karte als reiner Klon im Sortiment, wobei zusätzlich eine kleine Zwischenplatine zu erwerben war, die den Anschluss für das Kabel beim Mac 7100/8100 nach hinten bis an ein Slotblech verlängert.

1996 brachte Reply eigene Interpretationen der PCI-PC-Karten von Apple auf den Markt. Die 7“-Karte kann bis zu 128 MByte Arbeitsspeicher aufnehmen und die CPU ist gegen eine 133 MHz AMD-5x86-CPU getauscht. Größere Veränderungen gab es bei der 12“-Variante der letzten Apple-Entwicklungsstufe. Neben der Möglichkeit, bis zu 128 MByte Arbeitsspeicher zu verwenden (auf Kosten des Onboard-Speichers), wurde die CPU gesockelt und so gibt es Versionen mit Pentium 133, Pentium 166-MMX, 200 MMX und auch 233 MMX. Darüber hinaus kann auch der L2-Cache mit einem entsprechenden SIMM-Modul von 256 KByte auf 512 KByte und der Videospeicher auf bis zu 4 MByte aufgerüstet werden.

Der Staffelpack der PC-kompatiblen Karten wurde im April 1997 durch den Verkauf der DOS on Mac Technologie an Radius weitergegeben. Diese verkauften die obigen Karten weiter und es gab zusätzlich noch DOS-on-Mac Karten für den LC-PDS Steckplatz wie der Macintosh 6200er Serie. Diese Karte ist von der Grundkonzeption offensichtlich denen für den PCI-Steckplatz ähnlich und nutzt ebenfalls einen AMD 5x86 Prozessor mit 133 MHz. Weitere Informationen dazu waren leider nicht auffindbar.

Letztlich zog sich Radius schleichend ab 1997 vom Apple-Hardware-Markt zurück. Somit gab es auch von dieser Seite keine Weiterentwicklungen der Apple PC Compatible Cards mehr.

Aufgegessen - DOSen Macs sind aus

Und plötzlich war es vorbei – nachdem von den Apple-Karten ab 1997 keine Weiterentwicklungen mehr herausgebracht wurden, kam, wie bereits erwähnt, Ende 1998 das Finale für die OrangePC-Linie.

Warum wurden beide vielversprechenden Produktlinien nicht mehr gepflegt? Zum einen war da sicher der Preis, denn schließlich kostete eine PC-kompatible Karte für den Mac, vor allem die leistungsfähigen wie die OrangePC 660 fast

so viel wie ein adäquater PC. Der bot dann aber noch die normalen Erweiterungsmöglichkeiten mit AGP-Grafik und anderem. Viel entscheidender war aber, dass es für diesen Aufwand eigentlich keine Rechtfertigung mehr gab. Die PC-Karten wurden zum einen zum DOS und Windows-kompatiblen Arbeiten benutzt. Inzwischen ließen sich die Dateien von Windows-Office und Mac-Office problemlos austauschen und es gab von den meisten Business-Programmen Lösungen für beide Welten. Zum anderen fiel auch der zweite, vor allem für die letzten Karten besonders wichtige Grund weg: Die Karten wurden oft als die Möglichkeit angepriesen, endlich am Mac richtig spielen zu können. Das zeigt sich im Anschluss für einen Joystick, den sogar schon die Houdini I Karte bereitstellte. Mit den leistungsfähigen PowerPC Prozessoren ab der G3- Serie und vor allem dem Erscheinen von Grafikkarten mit flotten ATI-, 3Dfx und nVidia-Chips für den PCI-Bus im Mac wurde auch dieser selbst zu einer Spieleplattform. Die wichtigsten Titel wurden für MacOS veröffentlicht. Und wer, aus welchen Gründen auch immer, noch ein DOS- oder Windows-System am Mac laufen lassen wollte, für den hatten diese PowerPC-Prozessoren genug Leistung, um das ausreichend flüssig mit einer Software-Emulation wie Virtual PC zu bewerkstelligen.

Den letzten Rechtfertigungsgrund verlor diese „Brückentechnologie“ letztlich einige Jahre später. 2006 wechselte Apple von der PowerPC-Architektur hin zur Intel-Welt. Damit lief Microsoft Windows sogar nativ auf der Macintosh Hardware und Boot Camp ermöglichte den Dual Boot zwischen MacOS X und der Windowswelt.

Ueber den Autor

Herwig Solf ist ein computer-begeisterter Realschulelehrer aus Niederbayern und seit November 2013 Mitglied im VzEkC e. V. Seine Leidenschaft gilt den PCs als Parasiten in anderen Wirtssystemen, also den Hardware-PC-Emulatoren in Amiga, Atari und Macintosh.

SuSE Linux als Server für viele Plattformen

SuperNOS im Eigenbau

Einst hatte Novell Inc. große Pläne: Netware 4.1 sollte mit Unixware Version 2 zu einem Betriebssystem verschmelzen, dem SuperNOS. Wäre es nach CEO Ray Noorda gegangen, hätte eine Codebasis sowohl Unix- als auch PC-Netze versorgt. 1995 wurden diese Pläne still zu Grabe getragen. Doch nichts hindert den beherzten Retrocomputing Fan, sich ein entsprechendes Betriebssystem selbst zu bauen.

1993 übernahm Novell die Unix System Laboratories inklusive des Gemeinschaftsprodukts Univel, einem Unixderivat von AT&T. Unter dem Namen Unixware vermarktet, war dieses Produkt parallel zur sehr erfolgreichen Novell Netware aufgestellt. 1994 kündigte Robert Frankenberg, Vorstandsvorsitzender von Novell, diese beiden Produkte würden verschmelzen. Das neue, intern als SuperNOS bezeichnete Produkt, sollte einen modularen, verteilten Mikrokern besitzen und sowohl UNIX- als auch PC Netze mit Netware-Clients bedienen können. Doch aus der für 1997 geplanten Veröffentlichung wurde nichts -- der Markt hatte sich nicht zuletzt durch Microsoft Windows NT zu Ungunsten von Novell verschoben. 1995 wurde Unixware an die Santa Cruz Operation (SCO) verkauft, welche im Anschluss vor allem durch unbelegte Behauptungen bezüglich Copyright-Verletzungen durch Linux von sich Reden machte und nicht durch ein SuperNOS.

Doch ein Betriebssystem, das sowohl Unix Rechner als auch MSDOS Maschinen im Netz mit Datei- und Druckdiensten versorgt, muss kein Traum bleiben. Mehr noch -- mit freier Software und der richtigen Konfiguration wird ein Server zur Realität, der neben Clients unter Unix oder MSDOS auch Windows- und Macintosh Clients versorgt. Dieser Artikel beschreibt, wie sich ein derartiger Netzwerkserver leicht selbst aufbauen lässt. Ziel ist es, einen Server aufzusetzen, der Verzeichnis- und Druckerfreigaben für MSDOS- und Windowsclients ebenso bereitstellt wie für Unix-Systeme und klassische Apple Macintosh Rechner. Der Server soll also Netware Laufwerke,

Windows Shares, Macintosh Laufwerke und NFS Verzeichnisse bereitstellen.

Das ist in vielerlei Hinsicht eine große Aufgabe. Jede dieser Serverfunktionen hat ihre eigene Architektur und funktioniert nach eigenen Regeln. Die Hintergründe und Technologien im Detail zu beschreiben, würde ganze Bücher füllen. Dieser Artikel streift diese daher nur am Rande und nur soweit, wie es für das Verständnis der beschriebenen Serveremulationen nötig ist. Wer sich in die Hintergründe einarbeiten möchte, kommt um ein Studium der zahlreichen Quellen im Internet nicht herum. Dies gilt auch für den Umgang mit Linux als Basis für unseren SuperNOS Server.

SuSE Linux 9 als Basis

Als Ausgangspunkt für das eigene SuperNOS ist eine ältere Linux-Distribution vonnöten. Neben der Serversoftware selbst auf der Anwendungsebene braucht es auch die passenden Netzwerkprotokolle. Novell Netware benötigt IPX, Windows will NetBIOS over TCP/IP beziehungsweise SMB (Server Message Block), ein Macintosh möchte AppleTalk oder genauer EtherTalk sehen und Unix das NFS Protokoll. Aktuellen Linux Kernel fehlt aber der IPX- und AppleTalk Protokollstack. Also muss also passend zur Aufgabe ein angestaubter Karton aus dem Regal gezogen werden. Die beste Wahl ist SuSE Linux in Version 9.1. Vom Bedienkomfort schon recht weit entwickelt, bringt diese Version alle erforderlichen Module der Netzprotokolle im Kernel mit und auch die Serversoftware selbst ist



im Grundumfang enthalten. SuSE Linux 9.1 lässt sich kostenfrei herunterladen und installieren, die erforderlichen Internet Links finden sich am Ende des Artikels.

Für diesen Artikel wurde ein Pentium III Rechner mit 512 MByte Arbeitsspeicher und einer 4 GB PATA Festplatte genutzt. Ein solcher Rechner sollte sich leicht auf-treiben lassen. Zu neu darf die Maschine aber nicht sein, denn SuSE Linux 9.1 unterstützt neue Hardware natürlich nicht so gut wie aktuelle Linux Distributionen. Prinzipiell ist auch die Installation als virtuelle Maschine denkbar. Jedoch haben die vorangehenden Experimente gezeigt, dass viele Virtualisierungsplattformen die Netzprotokolle jenseits von TCP/IP oft nicht korrekt durch reichen. Hier gilt es also, echte Hardware einzusetzen, um sich Probleme schon am Anfang zu ersparen.

Die Installation passiert menügeführt und im Grafikmodus, sie ist für einen geübten Retrocomputer-Fan keine Herausforderung. Die übrige Konfiguration und erfolgt dann mit dem Tool YaST (yet another setup tool) der SuSE Distribution. Dies sollte auch konsequent genutzt werden, wo immer es möglich ist. Das Tool ist sowohl mit einer grafischen Oberfläche verfügbar als auch in einem Konsolenfenster im Textmodus. Das ist praktisch, um einen einmal aufgesetzten Server über eine ssh-Verbindung zu administrieren. YaST ist auch dafür zuständig, nachträglich fehlende Softwarepakete zu installieren und Software-Updates einzuspielen.

Für die übrige Konfiguration brauchen wir nur noch ein Verzeichnis anzulegen, das später die im Netz bereitzustellenden Dateien aufnimmt. Am besten eignet sich hierfür ein Verzeichnis unterhalb von /var, in den folgenden Beispielen ist /var/share gewählt. Alle Konfigurationsarbeiten in diesem Artikel werden mit den Rechten des Systemverwalters ausgeführt, also vom Benutzer "root". Die Beispielkonfigurationen sind übrigens auf der CDROM zum Heft vorhanden. Wer also die Installation nachvollziehen möchte, kann diese einfach übernehmen.

NFS Server aufsetzen

UNIX Rechner im Netz nehmen am einfachsten den Kontakt über NFS mit unserem SuperNOS auf. NFS ist ein Netzwerkprotokoll für den Export von Verzeichnisknoten auf andere Rechner. Es wurde ursprünglich von Sun Microsystems Ende der 1980er Jahre entwickelt. Heute ist es als Internet Standard in mehreren RFC's standardisiert. Um als NFS Server funktionieren zu können, sind mittels YaST die Pakete „nfs-server“ und „nfs-utils“ zu installieren. Erforderliche Abhängigkeiten zu anderen Paketen löst YaST automatisch auf und installiert diese nach Rückfrage gleich mit. Seine Konfiguration erledigt YaST mit der Funktion „Netzwerkdienste / NFS-Server“. Als zu exportierendes Verzeichnis wählen wir `/var/share` mit den Optionen `rw,root_squash,sync`. Dadurch wird das Verzeichnis beschreibbar exportiert (`rw`), Clientzugriffe durch den Systemverwalter `root` werden auf den User `nobody` umgebogen (`root_squash`) und der NFS Server meldet einen Schreibvorgang erst dann als abgeschlossen zurück, wenn die Daten wirklich auf die Platte geschrieben wurden (`sync`). Die Zugriffsberechtigung ließe sich auf bestimmte Rechner oder Netze eingrenzen, indem diese entsprechend aufgelistet werden. Für unseren Server reicht hier ein Wildcard-Zeichen (*), dadurch erhalten alle Clients den Zugriff. Die Konfiguration wird in der Datei `/etc/exports` abgelegt und dort vom NFS Serverdaemon gelesen.

Samba für Windows Clients

Windows Rechner kennen ab der Version Windows für Workgroups 3.11 (WfW) eigene Netzfunktionen (Peer-to-Peer Netz). Spätere Versionen haben zusätzliche Funktionen erhalten und Microsoft zum führenden Anbieter für Serverbetriebssysteme im LAN gemacht. Um unser SuperNOS auch in diesen Netzen heimisch werden zu lassen, braucht es den Server Samba. Samba ist eine Entwicklung, die ursprüng-

lich von dem Australier Andrew Tridgell im Jahr 1992 begonnen wurde. Sie sollte den Datenaustausch zwischen SunOS und DOS auch ohne NFS ermöglichen. Samba implementiert dazu die Netzfunktionen von Windows Netzen (Server Message Block, SMB) auf UNIX und ermöglicht die Kommunikation mit SMB über TCP/IP. Mit der Portierung auf Linux begann der Siegeszug der Software. Sie wird bis heute weiterentwickelt und ist Bestandteil so ziemlich jedes Consumer NAS. War die ursprüngliche Samba Version auf Peer-to-Peer Ebene angesiedelt, so kann Samba heute selbst die Rolle des Domänencontrollers übernehmen und die Active Directory Domain Services auf einem Linuxsystem bereitstellen. Dementsprechend komplex ist auch die Konfiguration von Samba geworden. An dieser Stelle begnügen wir uns aber mit einer einfachen Grundkonfiguration. Sie ist auch auf der Heft-CDROM zu finden.

YaST konfiguriert Samba über das Menü unter „Netzwerkdienste / Samba-Server“. Aufgrund der vielen Einstellungsmöglichkeiten empfiehlt es sich aber, die Konfiguration direkt in der Datei `/etc/samba/smb.conf` vorzunehmen. Sie richtet den Server mit dem Namen `SMBSUPERNOS` ein und definiert eine Arbeitsgruppe `SUPERNOSGROUP`. Für jeden Benutzer, der Zugriff auf diesen Server haben soll, braucht es zweierlei: Es ist ein Eintrag in die Linux Benutzerdatenbank nötig und Samba braucht einen Eintrag in der eigenen Benutzerliste. Der Linux Benutzer ist mit YaST unter Sicherheit und „Benutzer / Benutzer“ bearbeiten und anlegen schnell erstellt. Der Name des Benutzers in diesem Beispiel ist einfach „user“. Anschließend wird dieser Benutzer auch für Samba eingerichtet, und zwar mit dem Befehl `smbpasswd` von der Kommandozeile aus:

```
smbpasswd -a user
```

Das Passwort sollte mit dem des Linux Benutzers identisch sein.

Samba Konfiguration

```
[global]
; ----- Netzwerk
interfaces = 192.168.0.100
bind interfaces only = Yes
; ----- Servername
netbios name = SMBSUPERNOS
workgroup = SUPERNOSGROUP
server string = SuperNOS Samba
Vers %v
socket options = TCP_NODELAY
SO_SNDBUF=12288 SO_RCVBUF=12288
; ----- Zugriffssteuerung
security = user
passwd backend = tdbsam
lanman auth = Yes
client lanman auth = Yes
client plaintext auth = Yes
ntlm auth = yes
; ----- WINS
; dieser Samba Server ist WINS
Server
wins support = yes
; ----- Browsing
domain master = yes
local master = yes
preferred master = yes
os level = 200
guest account = nobody
; ----- Drucken
load printers = yes
; ----- Logging
log file =
/var/log/samba/log.%m
max log size = 50
log level = 3
[share]
writeable = yes
path = /var/share
force group = users
comment = Offenes Verzeichnis
auf SuperNOS mit SMB
valid users = @users
create mode = 0750
public = yes
[user]
comment = User's Netzlaufwerk
valid users = user
path = /home/user
wide links = no
```

Netatalk für Macintosh Clients

Apple Macintosh Systeme haben von Anfang an eigene Netzfunktionen mitgebracht, nämlich AppleTalk. Ursprünglich funktionierte die Netzkommunikation über serielle RS422 Busverkabelung als LocalTalk. Die Netzfunktionen sind aber in mehreren Protokollschichten realisiert und lassen sich als Ethertalk auch über Ethernet betreiben. Mit Netatalk stehen die Datei- und Druckserverfunktionen auch unter Linux zur Verfügung. Damit können wir von einem Macintosh aus auf Daten auf einem Linux-Rechner zugreifen oder auf einem angeschlossenen Drucker drucken. Netatalk ist eine Reihe von Unix-Programmen, die auf dem kernelbasierten DDP (Data-gram Delivery Protocol) laufen und die AppleTalk-Protokollfamilie (ADSP, ATP, ASP,

- Samba-Server deaktivieren
- Samba-Server aktivieren

Freigabe-Typ

- Freigabe von Dateien und Druckern
- Backup Domain Controller
- Primary Domain Controller

Domain oder Arbeitsgruppe:

SUPERNOSGROUP [Durchsuchen]

Server-Beschreibung:

SuperNOS Samba Vers %v

Name des Server-NetBIOS:

SMBSUPERNOS

[Authentifikationsdetails]

Netatalk Konfiguration

Die Konfiguration des MAR.S NWE erfolgt in einer zentralen Konfigurationsdatei, die als `/etc/nwserve.conf` abgelegt ist. Sie enthält eine sehr ausführliche Beschreibung der möglichen Einstellungen in Form von Kommentaren. Die hier gezeigte Beispielkonfiguration zeigt nur die Einträge, die bei einem typischen Einsatzfall verändert werden müssen. Die Einträge erzeugen beim ersten Start die emulierte Bindery. Spätere Änderungen erfordern daher den Aufruf von `nwserve -h`, um wirksam zu werden.

Jede Einstellung wird innerhalb der Datei durch eine Sektionsnummer gekennzeichnet, es können mehrere Zeilen mit der gleichen Sektionsnummer vorkommen. Zeilen, die mit einem Hash-Mark (#) beginnen, sind Kommentare und werden als solche von MAR.S NWE ignoriert. Der erste Eintrag gibt eine Zuordnung von Linux-Verzeichnissen zu NetWare-Volumen an:

```
1 SYS
  /usr/local/nwe/SYS/
  kt 711 600
1 SHARE /var/share
  kt 777 666
```

Es muss immer ein SYS: Volume existieren, es können weitere Definitionen folgen. Zu beachten ist, dass der jeweilige Linux-Pfad (hier `/var/local/nwe/SYS`) vor dem Start des Servertasks auch tatsächlich besteht, sonst wird der Eintrag dauerhaft ignoriert. Im Falle des SYS: Volumes müssen auch Unterverzeichnisse `mail`, `public`, `login` und `system` existieren. Das Flag "k" legt fest, dass alle Dateien und Verzeichnisnamen in Kleinschrift erwartet werden, was bei der Bezeichnung von Dateien unter Linux meist die Regel ist. Das "t" Flag ermöglicht die Verwendung von Trustees für die Vergabe von Rechten an den Dateien, also einer differenzierten Art der Rechtevergabe und einer Spezialisierung von Novell NetWare. Durch den Eintrag

```
1 CDROM /mnt/cdrom
  irm
```

würde ein Volume CDROM: definiert, das auf dem Mountpoint für CDROMs unter Linux verweist. Dabei sind die Flags besonders wichtig: "i" bewirkt ein Ignorieren von Groß- und Kleinschrift, "r" weist das Volume als "read only" aus und "m" ermöglicht ein erneutes Mounten des Datenträgers, während der Emulator noch läuft, also ein Austauschen der Silberlinge im laufenden Betrieb. Möglich ist auch ein Eintrag in der Form

```
1 HOME ~
  kio
```

denn dieser macht für jeden Benutzer getrennt das Heimatverzeichnis unter Linux mit dem Volumenamen HOME: erreichbar.

Um den NetWare-Serverdienst im Netzwerk eindeutig zu identifizieren, wird normalerweise der Hostname des Linux-Rechners verwendet. Bei Clients, die auch TCP/IP verwenden und einem Linux-Rechner (hier SUPERNOS), der parallel zu MAR.S NWE auch Samba verwendet, kann dies zu Verwirrungen führen. In der Netzwerkumgebung von Windows 95/98 wird der Rechner doppelt auftauchen, einmal als SMB- und einmal als NetWare-Server. Besser ist es, dem NetWare-Server einen eigenen Namen zuzuweisen:

```
2 NWSUPERNOS
```

Die beiden nächsten Einträge legen die Konfiguration des IPX Protokolls fest. Es ist zunächst notwendig, für den Server ein internes Netz (Sektion 3) festzulegen. Dieses virtuelle Netz dient dem Routing von Daten vom realen Netz zum Serverprozess und muss im Netzwerk eindeutig sein. Eine Konvention bei der Einrichtung von NetWare-Servern schreibt hier die Verwendung der MAC Adresse der Ethernetkarte für das "IPX internal net" vor. Man kann hier dem Server die Wahl auch selbst überlassen, der dann allerdings die IP Adresse des Linux-Rechners zur Erzeugung einer eindeutigen Adresse bemüht. Dies setzt eine funktionierende TCP/IP Konfiguration voraus. In Sektion 4 findet eine Konfiguration der Netzwerk-Interfaces statt, als erster Parameter steht die Netzwerknummer, die ebenfalls eindeutig für den Server sein muss. Dann folgt die Angabe des Netzwerk-Devices und des gewünschten Ethernet-Frames. Hier ist "ethernet_ii" die richtige Wahl für Netzwerke, in denen auch TCP/IP Datenverkehr abläuft. Wichtig ist hierbei, dass auch die Clients diesen Rahmentyp verwenden, da sonst niemals eine Verbindung zum Server zustande kommt. Sektion 5 legt fest, wie mit den erzeugten IPX Einstellungen nach dem Stop des Serverprozesses umzugehen ist; der Eintrag `0x1` verhindert ein Rücksetzen der Parameter nach dem Server-Shutdown.

```
3 auto 1
4 0x10 eth0 ethernet_ii
5 0x04
```

Gibt man in der Sektion 4 mehrere Interfaces an, so routet der MAR.S NWE automatisch zwischen diesen.

Die Sektionen 6 (NetWare-Version), 7 und 8 (Passwort-Handling) können getrost auf den Voreinstellungen belassen werden. In-

teressanter ist da schon der folgende Eintrag: Er gibt die Rechteumsetzung für das Anlegen von Verzeichnissen und Dateien auf Linux-Ebene an. In der Regel lautet der Eintrag hier

```
9 0751 0640
```

wodurch neue Verzeichnisse mit den Rechten `drwxr-x-x` und Dateien mit `rw-r---` angelegt werden. Dies beschränkt den Vollzugriff also auf den Eigentümer der Dateien.

Unter Novell NetWare besitzt jeder Client bereits eine Zugriffsmöglichkeit auf den Server, ohne dass eine Anmeldung an diesem erforderlich ist. Dieses Server-Attachment dient der Bereitstellung einer Login-Möglichkeit und birgt für die Linux-Seite ein Sicherheitsrisiko: Es besteht ein offener Systemzugang ohne Passwort-Absicherung. Durch die Sektionen 10 und 11 ist daher die Festlegung einer Benutzer- und Gruppenidentität möglich, Zugriffe über eine "attach" Verbindung sind dann diesem Benutzer zugeordnet. Hier sollte der Benutzer "nobody" aus der Gruppe "nogroup" eingetragen werden, dessen ID sich leicht durch ein `grep nobody /etc/passwd` unter Linux ermitteln lässt; ein Ergebnis könnte lauten:

```
nobody:x:65534:65534::/tmp:/bin/false
```

Das erste Auftreten des Wertes "-2" stellt die Benutzer-ID (UID) dar, das zweite die Gruppen-ID (GID). Die UID gehört in Sektion 10, die GID in Sektion 11:

```
10 65534
11 65534
```

Zu beachten ist, daß "nobody" kein Schreibrecht auf das Verzeichnis besitzt, da sonst ein Attentat auf das System durch das Vollschieben der /var Partition möglich wäre. Ist diese Partition erst einmal zu 100% ausgelastet, bringt das alle Prozesse zum Stehen, die hier Daten ablegen.

Die Sektionen 12 und 13 legen die Benutzer des NetWare-Emulators und die Zuordnung dieser zu bestehenden Linux-Benutzern fest. Dabei bestimmt der Eintrag 12 den Namen und das Passwort des Systemadministrators, also zumeist SUPERVISOR. In der folgenden Sektion können nun alle Benutzer explizit aufgeführt werden, diese werden nach dem ersten Starten des Emulators in die Bindery übernommen. Danach sollte man aus Sicherheitsgründen zumindest das Initial-Passwort des Supervisors wieder aus der Datei löschen, es bleibt in verschlüsselter Form in der Bindery enthalten.

```
12 SUPERVISOR root
```

```
supernos
13 GUEST nobody
- 0x1
13 USER
  user SUPERNOS
  0x1
```

Alternativ dazu lassen sich auch alle schon auf dem Linux-Rechner vorhandenen Benutzer mit einem identischen Initialpasswort in die Bindery übernehmen. Dieses Verfahren erspart bei hoher Benutzeranzahl einige Arbeit und reduziert die Gefahr von falschen Einträgen :

```
15 0 supernos
  Dadurch übernimmt der Emulator alle lokalen Benutzer in die Bindery und setzt ihr NetWare-Passwort auf "supernos".
```

Die folgenden Sektionen können gefahrlos auf den Default-Einstellungen verbleiben, interessant wird es erst wieder in der Sektion 22. Hier findet nämlich die Einstellung der Druckdienste statt:

```
21 LP - lpr
```

Der Eintrag leitet Druckdaten über einen Druckereintrag (lp) in der `/etc/printcap` direkt an einen angeschlossenen Laserdrucker. Wichtig ist hier, dass die entsprechenden Druckbefehle auch wirklich funktionieren. Positive Druckergebnisse erhalten die NetWare-Clients nur, wenn auch Linux drucken kann.

Von den folgenden Einträgen sind nur wenige für den Anfang interessant. Der Ordnung halber sollte man die Pfade in den Sektionen 40-47 für die Bindery- und Trusteespeicherung auf ein entsprechendes Verzeichnis richten (z.B. `/var/nwserve`) richten. Außerdem ist es ratsam, in Sektion 201 den Eintrag "syslog" zu ergänzen, was eine Übergabe von Meldungen des Emulators an den Systemprotokollschreiber `syslogd` des Linux-Rechners bewirkt. Von dort lassen sich die Einträge zentral verwalten, was das Leben des Systemadministrators deutlich erleichtert.

Nach der Erstellung und dem Test der Konfiguration braucht man nicht mehr Hand an diese Datei zu legen. Die weitere Administration des Systems geschieht bequemer von einem Client-PC aus mit Tools wie "syscon" oder dem Netware Assistant. Viele Einstellungen wie das Anlegen von Benutzergruppen, der Einstellung von Login-Scripts oder von Zeitrestriktionen sind nur auf diesem Wege möglich. Die Speicherung erfolgt in dann wie es sich gehört in der Bindery des Emulators.

RTMP, NBP, ZIP, AEP und PAP) implementieren. DDP wird vom Kernelmodul „apple-talk" bereitgestellt. Das ist aber auch gleichzeitig der Pferdefuß: Neuere Linux Distributionen enthalten dieses Modul nicht mehr, weil es für Mac OS X nicht mehr benötigt wird. Das ist einer der Gründe, wie-so in diesem Artikel mit SuSE Linux 9.1 gearbeitet wird, denn hier ist das Modul von

Anfang an vorhanden. Netatalk besteht normalerweise aus drei Servern:

```
_____ atalkd ("AppleTalk Network Manager") für das Netzprotokoll selbst
_____ afpd ("AppleTalk Filing Protocol daemon") für Verzeichnisfreigaben
_____ papd ("Printer Access Protocol daemon") für Druckerfreigaben
```

Wichtig ist es, die Einschränkungen von

Netatalk zu kennen. Zunächst dürfen Passwörter für Benutzer nicht länger als 8 Zeichen sein. Macintosh Clients können ausserdem keine Dateien nutzen, deren Namen länger als 31 Zeichen sind oder den Doppelpunkt enthalten. MacOS nutzt den Doppelpunkt als Trennzeichen in Pfaden.

Netatalk nutzt zur Benutzerauthentifizierung die Linux Benutzerdatenbank. Unser oben eingerichteter Benutzer "user" wird also auch hier genutzt.

Die Konfiguration erfolgt über die Dateien unterhalb des Verzeichnisses `/etc/netatalk`. Hier liegen folgende Dateien:

netatalk.conf

bestimmt, welche Netatalk Daemons gestartet werden sollen.

atalkd.conf

gibt an, über welche Schnittstellen Dienste bereitgestellt werden. Wenn der Server gestartet wird, sucht er das Netzwerk nach vorhandenen Zonen und Servern ab und ändert die entsprechenden Zeilen, indem er die eingestellten AppleTalk-Netzwerkadressen einträgt.

afpd.conf

enthält Definitionen dafür, wie der Server auf MacOS-Rechnern als Element im Dialogfeld „Auswahl“ angezeigt wird.

papd.conf

definiert die Drucker für die MacOS Clients.

AppleVolumes.default

definiert die zu exportierenden Verzeichnisse. Die Zugriffsrechte definieren die üblichen Unix-Benutzer- und Gruppenrechte. Dies wird in der Datei `AppleVolumes.default` konfiguriert. Neben der Datei `AppleVolumes.default` können weitere Dateien existieren, z.B. `AppleVolumes.guest`, die von einigen Servern verwendet wird.

AppleVolumes.system

bestimmt, welche MacOS-üblichen Typ- und Erzeugerangaben bestimmten Dateiendungen zugeordnet werden. Meist genügen die Standardwerte. Erscheint eine Datei auf dem MacOS Client mit einem allgemeinen weißen Symbol, so gibt es für sie hier keinen Eintrag. Dann ist hier eine Ergänzung nötig.

Für diese Dateien auf der CDROM zum Heft passende Beispiele zu finden. Sie definieren einen Server ASUPERNOS mit einer Verzeichnisfreigabe „share“ für `/var/share`. Die Dokumentation zu SuSE Linux liefert in Kap. 17 weitere Informationen.

MAR.S NWE emuliert Novell Netware

Novell Netware war lange der unangefochtene Marktführer, wenn es um Netzwerke für IBM PCs ging. Ein durchdachtes System für Rechtevergaben auf der Serverseite und eine recht einfache Konfiguration der Clientseite machten Netware so beliebt. Auch die Nachbildung der Datei- und Druckdienste ist unter Linux möglich, und zwar mit dem Emulator MAR.S NWE

von Martin Stover. Er wurde bis zur Version 0.99p11 weiterentwickelt und ist in der Version auch Bestandteil der Paketquellen von SuSE Linux 9.1. Neben der Funktion eines RIP/SAP Routers vermag MAR.S NWE sowohl Datei- als auch Druckdienste anzubieten. Auch hier lassen sich beliebige Teile des Linux-Verzeichnisbaumes zu Volumes erklären und für Clients freigeben.

Voraussetzung für den Betrieb von MAR.S NWE ist die Unterstützung des IPX Protokolls durch den Linux-Kernel. IPX ist im Standardumfang eines jeden Kernels der Version 2.0.x oder höher enthalten und bereits als Modul in der SuSE Distribution als Modul vorhanden. Der Emulator implementiert den Wortschatz des 'netware core protocols' (NCP) und kommuniziert über IPX. Hierbei handelt es sich um Prozeduren, die der NetWare Server ausführt, um Anfragen von Clients über das Netz entgegen zu nehmen und zu beantworten. Sie erlauben es auch, NetWare-Server über das Netz zu administrieren. Novell stellt hierfür Programme (wie `syscon` oder `nwadmin`) auf dem Server bereit. Diese funktionieren auch mit MAR.S NWE, wenngleich nicht in vollem Umfang. Informationen zur Verwaltung von Ressourcen wie Benutzer- und Gruppendefinitionen, Drucker, Laufwerke, aber auch von Eigenschaften wie Zugriffsrechten und Kontoinformationen emuliert MAR.S NWE sogenannte Bindery, also eine an einen Server gebundene Datenbank. Jeder Benutzer von MAR.S NWE muss getreu der Netware 3- Vorgaben in der Bindery des Emulators eingetragen sein, was entweder durch die Übernahme aller Benutzer des Linux-Rechners oder durch einzelne Einträge in der zentralen Konfigurationsdatei geschehen kann. Beim ersten Start des `nw-serv` Programms wird anhand dieser Konfiguration eine Bindery generiert und auf einem voreingestellten Pfad abgelegt.

Zur weiteren Administration des Systems lohnt es sich, den Inhalt des `SYS:PUBLIC` Verzeichnisses eines NetWare 3.x Servers auf das Pendant von MAR.S NWE kopieren. Hierzu reicht die Version einer 2-User

Netatalk Konfiguration

AppleVolumes.default:

```
~ mswindows,codepage:maccode.iso8859-1
:DEFAULT:limitsize:500000000 allow:@users
rwlist:@users
/var/share "share" volsizelimit:63
```

netatalk.conf:

```
AFPD_MAX_CLIENTS=20
ATALK_NAME=supernos
ATALK_ZONE=""
AFPD_GUEST=nobody
ATALKD_RUN=yes
PAPD_RUN=yes
AFPD_RUN=yes
TIMELORD_RUN=no
ATALK_BGROUND=yes
```

atalkd.conf :

```
eth0 -phase 2 -net 0-65534 -addr 65280.42
```

afpd.conf :

```
"asupernos" -ipaddr 192.168.0.100 -transall -uamlist
uams_dhx.so,uams_guest.so,uams_clrtxt.so
```

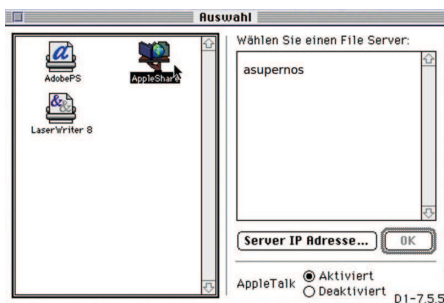
Demo- oder Schulungs-CD aus. Viele Einstellungen lassen sich nämlich sehr gut mit den Original-Tools von Novell ändern, beispielsweise mit `SYSCON`, dem liebsten Haustier jedes Netware-Supervisors. Benutzer und Gruppen lassen sich in bekannter Manier bearbeiten, auch die Serverüberwachung oder ein Shutdown funktionieren wie gewohnt. Hier zeigt sich, dass die Unterstützung der NCP Calls schon recht weit gediehen ist – unter anderem sind bereits Rechtevergaben innerhalb von Volumes in Form von Trustees möglich.

NFS nutzen

UNIX-Betriebssysteme haben in der Regel keine Schwierigkeiten, auf die Exports des NFS Servers zuzugreifen. Das exportierte Verzeichnis lässt sich einfach auf einen beliebigen Mountpoint montieren, beispielsweise nach `/mnt`:

```
mount -tnfs supernos:/var/share
/mnt
```

Der Name "supernos" muss dabei entweder lokal in der `/etc/hosts` Tabelle eingetragen sein oder sich über den DNS Server des heimischen Netzes auflösen lassen. Notfalls funktioniert aber auch die IP Adresse anstelle des Namens. Sollte dennoch ein Fehler auftreten, so fehlen vielleicht die NFS Funktionen. Bei vielen Linux-Distributionen sind diese Bestandteil der Pakete „nfs-client“ oder „nfs-util“ und müssen explizit installiert werden. Auch Sun Solaris, Hewlett Packards HP-UX oder NeXTSTEP greifen so auf unser SuperNOS zu. Probleme bereiteten dem Autor bisher nur Clients mit IBM AIX, die partout keine



Das Feld Auswahl zeigt den Server

Exports nutzen wollten. Die Ursache konnte bisher nicht geklärt werden.

SMB Clients unter Windows

Der Zugriff auf Samba sollte unter Windows die kleinsten Probleme bereiten, weil die Netzfunktionen spätestens seit Windows 95 tief in das System integriert sind. Der Client muss TCP/IP kennen und das Protokoll muss richtig konfiguriert sein. Außerdem muss er der Arbeitsgruppe SUPERNOSGROUP angehören und den SuperNOS Server als WINS Server nutzen. In der Netzwerkumgebung sollte unter den Windows Netzwerken dann diese Gruppe mit dem Server SMBSUPERNOS auftauchen. Wenn dies manchmal nicht der Fall ist, klappt die Namensauflösung über DNS und WINS nicht richtig. Die pragmatische Lösung besteht in einem statischen Eintrag der IP Adresse und des Namens des SuperNOS Servers in der Datei `windows\lmhosts`. Bei Windows NT ist dies `windows\system32\etc\lmhosts`. Wenn alle Stricke reißen, hilft meist eine Zuweisung eines Laufwerksbuchstabens zum Share in UNC Notation, hier also `\\smbsuper-nos\share`.

OS/2 und SMB

Am besten funktioniert der Zugriff auf den Server aus einem OS/2 Fenster heraus. Der Befehl `net` steht hier in ähnlicher Weise zur Verfügung wie unter MSDOS. Um Zugriff auf das freigegebene Verzeichnis auf dem Server zu erhalten, genügt also das folgende Kommando:

```
net use g: \\smbsuper-nos\share
```

Die Workplace Shell hat allerdings oft Schwierigkeiten mit derart zugeordneten Laufwerken. Besser klappt es mit einem alternativen Dateimanager wie File Freedom (siehe Links). Wer häufig zwischen Betriebssystemen hin- und her wechselt, wird das zweigeteilte Fenster mit Baumansicht und Dateibereich sowieso hakeligen Arbeit mit der Workplace Shell vorziehen.

Macintosh Clients

Vom Macintosh aus ist es recht einfach, den SuperNOS Server zu nutzen. Apple-



Netatalk fragt die Benutzerdaten ab

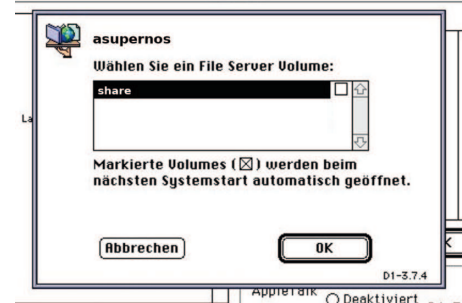
talk muss dazu über die Ethernet-Schnittstelle laufen, die Einstellung erfolgt im Kontrollfeld „Appletalk“. Rechner ohne eingebautes Ethernet haben es hier natürlich schwer. Sie sind nur ins Netz zu bekommen, wenn es entweder einen Macintosh mit laufender LocalTalk Bridge- Software im Netz gibt oder ein externer Ethernet Adapter angeschlossen wird. Entsprechende Adapter der Hersteller Farallon oder Asanté gibt es für die SCSI Schnittstelle oder für die serielle Schnittstelle. Allerdings sind diese Teile mittlerweile rar geworden. Eine Alternative ist der RaSCSI Adapter, der zusammen mit einem Raspberry Pi sowohl eine SCSI Festplatte als auch einen entsprechenden Adapter emulieren kann.

Hängt der Mac erst einmal am Netz, so ist der Rest schnell gemacht. Im Kontrollfeld „Auswahl“ (Chooser) findet sich unser Server unter dem Icon "AppleShare". Nach erfolgreicher Anmeldung als Benutzer „user“ bietet der Server die definierten Volumes an und packt sie nach Anklicken auf den Desktop.

Bei Macintosh System 6.x können Probleme auftauchen, wenn das exportierte Volume sich als zu groß meldet. Daher ist in der Beispielkonfiguration ein Limit von 63 MB eingetragen. Diesen Wert meldet Netatalk zwar an den Mac, er hat ansonsten aber keine größenbeschränkende Wirkung. Das Volume fasst soviel Daten, wie die Festplatte hergibt.

MSDOS mit Netware Workstation

Damit der Zugriff von MSDOS aus funktioniert, braucht es einen Protokollstack mit LSL und das IPX Protokoll aus dem Client Kit (siehe Links) sowie den passenden Netzkartentreiber. Dieser sollte sich auf der Treiberdiskette zur jeweiligen Netzwerkkarte finden. Für den Zugang von einem Client aus sind einige Tools erforderlich. Um nicht einen vorhandenen Novell NetWare Server plündern zu müssen, hat Martin Stover ein getrennt erhältliches Paket `mars_dosutils.tgz` mit einem unter DOS lauffähigen NET.EXE Programm erstellt. Dieses erlaubt die Anmeldung am Server und das Mapping von Ressourcen wie Laufwerken und Drucker. Die Dateien aus



Zugriff auf das eingerichtete Verzeichnis

diesem Archiv gehören in das Verzeichnis login unterhalb des definierten SYS Volumes und stehen so bereits zur Anmeldung zur Verfügung. Einige Hinweise zur Clientkonfiguration finden sich auch in der Beschreibung der MAR.S NWE Konfigurationsdatei.

Fazit

SuSE Linux auf diese Weise als SuperNOS für ein Retrocomputer Netz zu konfigurieren, ist vielschichtig, aber zu bewältigen. Sofern ein Rechner mit einer Ethernet Schnittstelle ausgestattet ist, lässt er sich fast immer dazu bewegen, mit dem Server in Kontakt zu treten. Dabei sind Clients mit aktuellen Betriebssystemen oft schwieriger zu beherrschen als ältere Systeme. Je neuer das System, desto mehr Sicherheitsfeatures sind implementiert und desto schlechter kommen die betagten Programme von SuSE 9.1 mit diesen Mechanismen klar. Doch für Freunde alter Maschinen kann das selbst gebaute SuperNOS vieles leisten, da es den Datenaustausch deutlich erleichtert. Es lohnt sich also, ausgehend von diesem Artikel in eigene Experimente einzusteigen und das eigene System zu optimieren. (gb)

Links

SuSE Linux 9.1
<ftp://ftp5.gwdg.de/pub/linux/suse/disccontinued/i386/9.1>

Dokumentation zu SuSE Linux
<https://www.novell.com/de-de/documentation/suse/pdfdoc/SuSE-Linux-Adminguide-9.0.0.0x86.pdf>
<https://www-ux-sup.csx.cam.ac.uk/pub/doc/suse/suse9.1/adminguide9.1/>

Novell Netware Clients
<https://www.systemhaus-brandenburg.de/download/treiber/novell/novell.html>
<http://files.mpoli.fi/hardware/NET/NOVELL/CLIENTKT.ZIP>

OS/2 File Freedom
https://hobbes.nmsu.edu/download/pub/os2/util/browser/FileFreedom_2-02.zip



Konfigurationsbeispiele und Software finden sich auf der Heft-CDROM

Temperaturfühler am Apple][

Der Apple II eignet sich nicht nur aufgrund seiner Slots –die sich der IBM PC übri-gens vom Apple II abge-schaut hat– zum Basteln eigener Erweiterungen. Auch sein Paddle- oder Joy-stick-Anschluss taugt für Basteleien.

Er besitzt einige 1 Bit Ein- und Aus-gänge, die wir hier aber nicht be-nötigen und deshalb nicht weiter betrachten. Interessanter ist etwas ande-res: Es stehen an diesem Anschluss sage und schreibe 4 Analog-Anschlüsse zur Ver-fügung. Jeder dieser Analogports hat eine Auflösung von 8 Bit, kann also 256 ver-schiedene Werte liefern. Einen kleinen Ha-ken hat die Sache aber schon – Spannung lässt sich nicht direkt messen, sondern nur der Widerstand. Übrigens besitzt auch der Commodore 64 diese Fähigkeiten, bietet jedoch im Gegensatz zum Apple keine BA-SIC-Befehle zum Auslesen. Dort muss der Nutzer mit PEEK und POKE arbeiten.

Um eine Temperatur messen zu können, braucht es ein Bauteil, das mit steigender Temperatur seine Eigenschaften ändert. Das ist idealerweise der Widerstand, denn den kann der Apple II messen. Dazu gibt es zwei Sorten: Heiß- und Kaltleiter, auch als NTC und PTC bezeichnet. Heißleiter (NTC) leiten mit steigender Temperatur immer besser, Kaltleiter (PTC) leiten im kalten Zustand besser als bei hohen Temperaturen. Prinzipiell sind beide Sor-ten zu verwenden – ich hatte gerade ein paar NTC verfügbar und habe deswegen Heißleiter verwendet. Egal, ob NTC oder PTC, der Widerstandsbereich ist in beiden Fällen zu beachten. Der Apple II erkennt Widerstände von Null bis 150 kOhm. Bei 8 Bit sind das 256 Möglichkeiten und das er-gibt wiederum eine Schrittweite von etwa



590 Ohm. Als Kennwert wird im Handel der R25 angegeben, das ist der Widerstand bei 25 Grad Celsius. Mein NTC hat den Wert R25 = 50 kOhm und eine flache Kennlinie. Vorsicht bei NTCs, die als Begrenzung für Einschaltströme eingesetzt werden: Sie ha-ben eine steile Kennlinie und eignen sich nicht für dieses Projekt. Mein NTC-Sensor hat bei -18 Grad in der Gefriertruhe einen Widerstand von 170 kOhm, was der Apple als Maximalwert erkennt. Bei steigender Temperatur sinkt der Widerstand, der Ap-ple liefert also kleinere Messwerte bei stei-gender Temperatur.

Wie wird der NTC nun am Apple ange-schlossen? Dazu betrachten wir die Pinbe-legung des GAME I/O Ports:

Apple II Joystick Pinout

+5V	1	16	NC
PB0	2	15	AN0
PB1	3	14	AN1
PB2	4	13	AN2
STB	5	12	AN3
GC0	6	11	GC3
GC2	7	10	GC1
GND	8	9	NC

Von Interesse sind hier die Anschlüsse GC0 bis GC3, besonders GC0 und GC1. Wir verbinden den einen Pol des NTC mit GC0 (Pin 6), den anderen mit + 5 Volt (Pin 1). Und damit sind wir schon fertig. Wer mag, kann noch einen zweiten Sensor zwi-schen GC1 und +5V anschließen. Das kann beispielsweise ein zweiter Tempera-tursensor oder ein lichtempfindlicher Wi-derstand (LDR) sein.

Ich habe zwei Temperatursensoren und relativ lange Kabel verwendet, damit ich In-nen- und Außentemperatur messen kann. Die Kabellänge ist unkritisch, bei 150 KOhm Maximalwert machen 2 oder 3 Ohm Kabel-widerstand keinen beachtenswerten Unter-schied. Wir erinnern uns: Die Schrittweite beträgt 590 Ohm.

Nun geht es an die Softwareseite: PRINT PDL(0) zeigt den Wert von GC0 an (PDL(1) entsprechend GC1). Ein erstes kleines Testprogramm ist ein Zweizeiler:

```
10 Print PDL(0), PDL(1)
20 GOTO 10
```

Nun schreibt der Apple nach einem be-herzten RUN zwei Zahlenreihen fortlaufend auf den Schirm. Wer nur einen Sensor verbaut hat sieht als zweite Zahl immer die 255. Bei zwei identischen Temperat-ursensoren zeigen beide Sensoren die gleichen Werte an. Liegt ein Sensor in der warmen Hand, sinken die Werte. Der Wert 255 bedeutet sehr kalt, der Wert 0 sehr warm. Mögliche Erweiterungen des Test-programms wären beispielweise ein Tem-peraturlog und eine Speicherung der Werte auf Diskette. Dann empfiehlt sich aber eine Warteschleife zwischen Zeile 10 und 20, um nur etwa jede Sekunde einen Wert zu lesen. Zur Auswertung ließe sich dann ein extra Programm schreiben, um den Verlauf in der HiRes-Grafik darzustellen.

Allerdings kann das Programm bis jetzt nur relative Temperaturänderungen mes-sen. Um von Widerstandswerten auf Tem-peraturen zu kommen, braucht es eine Kalibrierung des Sensors und eine Um-rechnung der Werte in konkrete Tempera-turen. Hier lohnen sich eigene Experi-mente. In Teil 2 dieses Artikels stellen wir hierzu einige interessante Ansätze vor.

Über den Autor

Jochen Emmes ist Diplom-Physiker und seit 1995 hauptberuflich als Anwendungsentwickler und Systemadministrator angestellt.

Plattformübergreifende Programmentwicklung

Atari ST Cross-Entwicklungsumgebung

Ein Cross-Compiler läuft auf einem bestimmten System, kompiliert aber Objektdateien oder ausführbare Programme für andere Systeme. Das können andere Betriebssysteme, andere Prozessoren oder eine Kombination der beiden sein. Wir zeigen hier, wie ein Cross-Compiler unter Linux, Windows oder Mac OS X helfen kann, um Programme auf den Atari ST zu portieren.

Für eine plattformübergreifende Entwicklung braucht es im wesentlichen zwei Dinge. Zunächst ist natürlich ein Compiler erforderlich, der die benötigte Programmiersprache verarbeitet und an einen Assembler übergibt, der Binärcode für die Zielplattform erzeugen kann. Hier bedeutet das, Compiler und Assembler müssen auf einer Intel-Plattform Binärcode für den Motorola 68000 erstellen können. Das allein reicht aber nicht aus: Programmiersprachen wie C und C++ stützen sich auf Standardbibliotheken, in denen die Funktionen der jeweiligen Programmiersprache realisiert sind. Im Falle einer Entwicklung in C unter Linux ist es die GNU glibc. Außerdem braucht es weitere Bibliotheken für ergänzende Funktionen wie Grafikdarstellung oder Netzzugriff. Sie werden zum eigentlichen Ergebnis eines Compilerlaufs hinzugefügt; dies übernimmt ein spezielles Programm, der Linker. Bei einer Cross-Entwicklung sind auch diese Bibliotheken in einer 68000er Version erforderlich und der Linker muss mit diesen umgehen können.

GNU GCC

Die Suche nach einem passenden Compiler für eine Cross-Entwicklung führt meist zur GNU Compiler Collection, GNU GCC. Hierbei handelt es sich um freie Software aus dem GNU Projekt. Seine Geschichte reicht weit zurück. Begonnen hat sie mit dem GNU C Compiler, vom Urvater der Idee freier Software, Richard M. Stallmann 1987 selbst freigegeben. 1997 spaltete sich aus der Entwicklung dieses Compilers das

EGCS Projekt ab. Es verfolgte das Ziel, bestehende und eher experimentelle Abspaltungen aus dem GCC Projekt wieder zu vereinen und eine Compiler Collection zu schaffen, die neben C auch C++, Fortran und andere Programmiersprachen unterstützt. Im Jahr 1999 vereinigten sich EGCS und das ursprüngliche GCC Projekt wieder und seitdem wird die GNU Compiler Collection in der heute erhältlichen Form gepflegt. GCC unterstützt heute die Programmiersprachen C, C++, Objective-C, D, Fortran, Ada und Go und kann Code für verschiedenste Plattformen erzeugen, neben Intel-Prozessoren auch für Motorola-, ARM und diverse Workstation-CPU's und deren Betriebssysteme, aber auch für eingebettete Systeme. Auf vielen Plattformen ist der GCC "self hosted", kann also seinen eigenen Quellcode für die Zielplattform kompilieren.

Vincent Rivière's m68k-atari-mint cross-tools

Etwas mehr Aufwand braucht es bei der Suche nach der zweiten Komponente für die Cross-Entwicklung, nämlich die erforderlichen Bibliotheken. Hier empfiehlt sich ein Blick auf die Webseite von Vincent Rivière, in der Atari ST Szene unter anderem durch seine Arbeit an EmuTOS und der Firebee bekannt. Dort finden sich Compiler und Build-Umgebungen für Microsoft Windows, verschiedene Linux Distributionen und für Mac OS X. Normalerweise arbeitet der Autor unter Linux Mint 20.1 mit Hatari 2.3.1 als AtariST Emulator. Die Umgebung wurde aber auch erfolgreich mit Mi-

crosoft Windows 10 und mit Mac OS X Mojave und Catalina getestet. Neben den erforderlichen Bibliotheken bringen die Cross Tools auch den kompletten Compiler und Assembler mit. m68k-atari-mint-as oder kurz GAS ist dabei der von Vincent Rivière bereitgestellte bereit gestellte Assembler. Er wandelt eine Assembler-Quelldatei des Compilers in eine binäre Objektdatei um. Dieser erzeugt Binaries in einem Format für TOS und nicht im Binärformat, das für den TOS Nachfolger MiNT (nicht zu verwechseln mit dem Ubuntu-Abkömmling Linux Mint) spezifiziert wurde.

Umgebung unter Linux installieren

Die Installation unter Linux Mint 20.1 verläuft recht einfach, denn Vincent Rivière stellt ein "personal package archive" (PPA) bereit. Ein PPA ist ein Service der Softwareplattform Launchpad, die von Canonical als Unternehmen hinter Ubuntu betrieben wird. Es bietet die Möglichkeit, Debian-Pakete für unterschiedliche Architekturen und für unterschiedliche Ubuntu-Versionen zu bauen.

Zunächst muss unter Linux Mint 20.1 das PPA registriert werden:

```
sudo add-apt-repository
ppa:vriviere/ppa
```

Anschließend wird die Paketverwaltung aktualisiert:

```
sudo apt update
```

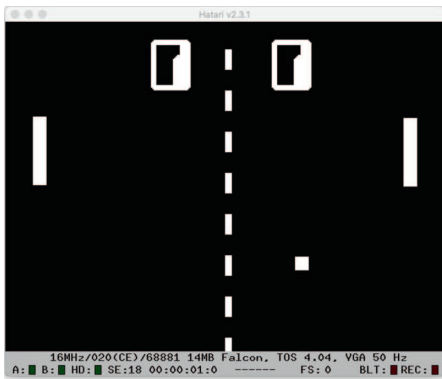
und die eigentliche Cross-Entwicklungsumgebung installiert:

```
sudo apt install cross-mint-essential
```

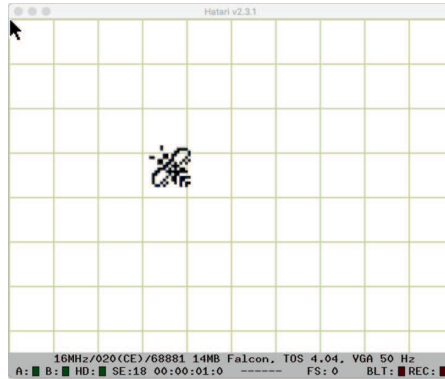
Der Erfolg lässt sich wie üblich mit einem kleinen "Hello World" Programm prüfen:

```
m68k-atari-mint-gcc hello.c
-o hello.tos
```

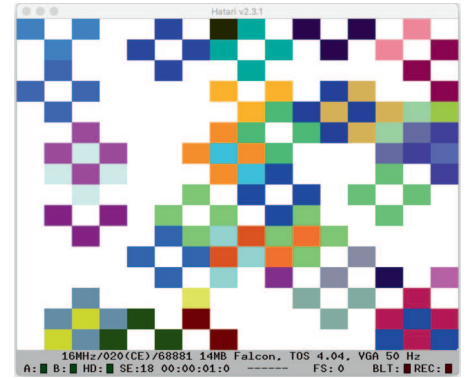
Die so übersetzten Programme werden allerdings ziemlich groß (Script 1), das "Hello World" Kompilat nimmt bereits 174.580 Bytes. Das hat einen Grund: Standardmäßig linken gcc-kompilierte Programme auf der Atari ST-Plattform auf mintlib. Mintlib zielt darauf ab, alle POSIX-Funktionen zu implementieren, um die Portierung von Unix/Linux-Programmen zu erleichtern Für kleine TOS und GEM-basierte



Pong läuft im Atari-Emulator Hatari



Ein einfaches SDL Programm



Farb-Demo in SDL

Programme hat dies jedoch den Nachteil einer entsprechenden Programmgröße. Deutlich kleiner werden die Binaries mit der libcmini von Thorsten Otto (Script 2). libcmini zielt darauf ab, eine C-Bibliothek mit geringem Platzbedarf für die m68k-atari-mint (cross)-Werkzeugkette zu sein, ähnlich der C-Bibliothek, mit der Pure-C geliefert wurde. Viele TOS und GEM-Programme benötigen nur eine Handvoll C-Bibliotheksfunktionen und nicht die volle POSIX Unterstützung der mintlib. Gegen diese Bibliothek kompiliert nimmt das "Hello World" Kompatil nur noch 2.722 Bytes ein.

Um die erzeugten Programme zu testen, braucht es nicht zwangsläufig einen vollständig installierten Atari ST Emulator. Vielmehr bietet TOSEMU von Paul Wratt einen einfachen Weg, ein TOS Binary auf Linux laufen zu lassen. TOSEMU fängt Betriebssystemaufrufe ab und emuliert diese auf der Host-Plattform. Das Prinzip ist Linux-anwendern sicher bereits von WINE bekannt – diese Software erlaubt es, Windows Binaries direkt auf Linux auszuführen.

Simple DirectMedia Layer

Programme im Textmodus lassen sich so recht zuverlässig kompilieren, sofern die passenden Bibliotheken bereitstehen. Spannend wird es, sollen auch grafikbasierte Programme wie Spiele portiert und in der Cross-Entwicklungsumgebung kom-

piliert werden. Bisherige Versuche des Autors haben sich auf SDL- basierte Programme beschränkt. SDL (Simple Direct Media Layer) ist eine plattformübergreifende Entwicklungsbibliothek, die den Low-Level-Zugriff auf Audio-, Tastatur-, Maus-, Joystick- und Grafikhardware ermöglicht. Einerseits macht der plattformübergreifende Ansatz SDL besonders reizvoll, für den Atari ST zeigen sich jedoch einige Nachteile:

- _____ Nutzbar ist für die Cross Entwicklung nur SDL1 und nicht SDL2.
- _____ SDL Programme sind sehr langsam und daher nur für schnelle Rechner wie den Falcon und wenig bewegte Grafik geeignet.
- _____ SDL Programme sind sehr groß, da die komplette Lib dazu gelinkt wird.
- _____ Ohne die Zusatzbibliotheken SDL_Image können nur BMP Grafikdateien verarbeitet werden. Der Versuch, die SDL_Image Lib und weitere benötigte Bibliotheken von Thorsten Otto einzubinden, misslang allerdings bisher.

Zwei Spiele ließen sich jedoch gut übersetzen, zum einen Snake und zum anderen Pong. Script 3 zeigt den Compileraufruf. Bei Pong gab es einen nicht untypischen Fehler. Das Binary lief auf dem emulierten AtariST unter Hatari aber nicht auf einem echten Falcon. Die Ursache: Eine Bilddatei mit dem Namen *numbermap.bmp*. Hatari unterstützt lange Dateinamen, dadurch konnte die Datei unter Linux gefunden werden. Auf dem Atari Falcon ist aber nacktes TOS vorhanden, das nur 8.3 Dateinamen kennt. Bei Portierungen lohnt es sich also, genau in den Quellcode zu schauen.

Gegen die vorkompilierte SDL Bibliothek gelinkte Programme ließen sich jedoch nicht auf einem 1040ST zum Laufen bringen. Das legt den Verdacht nahe, dass die Bibliothek Code für den Motorola 68020 enthält. Eine genaue Klärung steht noch aus.

Aber auch mit nur der reinen SDL lib ist viel zu machen. Texte lassen sich mit Bitmap Fonts ausgeben und selbst Sound geht ohne SDL_mixer. Ein kleines Beispiel findet sich im GitHub Repository des Autors. Da es sich um reines C und SDL handelt und nichts Atari Falcon spezifisches darin vorhanden ist, lässt es sich auf und für jedes System kompilieren, welches den (Cross)GCC und die LibSDL zur Verfügung stellt (ps).

Links:

- <http://vincent.riviere.free.fr/soft/m68k-atari-mint/>
- <https://tho-otto.de/crossmint.php?<-win32>
- <https://github.com/freemint/libcmini>
- <https://github.com/paulwratt/tosemu>
- <https://www.parallelrealities.co.uk/tutorials/sdl1/>
- https://lazyfoo.net/SDL_tutorials/
- <http://gamedevgeek.com/tutorials/moving-sprites-with-sdl/>
- <https://members.loria.fr/PGaudry/static/Methodo/animating-sprites.html>
- <https://gist.github.com/armornick/3447121>
- https://github.com/driedfruit/SDL_inprint
- Snake:
- <https://gist.github.com/cellularmitosis/6e54ea6d0965bff16ac947c19bc06d3c>
- Pong:
- <https://github.com/zotherstupidguy/pong>
- Beispiel:
- <https://github.com/petersieg/Simple-SDL-Demo>



Programmbeispiele finden sich auf der Heft-CD

```

Script 1:
#!/bin/sh
m68k-atari-mint-gcc $1.c -o
$1.tos
m68k-atari-mint-strip $1.tos

Script 2:
#!/bin/sh
m68k-atari-mint-gcc -nostdlib
./libcmini/lib/crt0.o $1.c -o
$1.tos -s -Llibcmini/lib/ -lcmini
-lgcc

Script 3:
#!/bin/sh
m68k-atari-mint-gcc $1.c -o
$1.prg -L /usr/m68k-atari-
mint/lib -lSDL -lgem

```

Ueber den Autor

Peter Sieg ist seit 2006 Wiedereinsteiger im Retrocomputer-Hobby. Er ist Autor der Buecher "Commodore Hardware Retrocomputing" und "Simulation-Emulation Exotic Flavour" und hat zahlreiche Beitrage fuer die LOAD geliefert.

Programme unter Linux assemblieren

Atari ST Cross Assembler

Github hat sich als Austauschplattform für verschiedenste Projekte etabliert und auch nach der Übernahme durch den Branchenriesen Microsoft nicht an Attraktivität eingebüßt. In den Repositories von Github finden sich aber beileibe nicht nur Quelltexte zu aktueller Software.

Die Plattform wird auch genutzt, um Quellen der Software für Retrocomputer zu archivieren. Eines dieser interessanten Projekte ist das Atari ST source code repository von Github-User "ggnkua". Dort sind Quellcodes in Assembler, C, GFA Basic, Omikron Basic und Pascal ebenso zu finden wie Dokumentationen und TOS Quellcodes.

Der Zweig der Assemblerquellen eignet sich hervorragend für eigene Experimente. Recht einfach wäre es, die Quellcodes mit einem Assembler auf dem Atari selbst zu assemblieren. Doch nachdem der Aufbau einer Crossentwicklungsumgebung für C-Code auf dem PC bereits viel Spaß gemacht hat, war es naheliegend, auch die Assemblerprogramme auf diesem Weg zu übersetzen. Als Assembler lässt sich dafür der VASM unter Linux Mint 20.1 nutzen. Auch der Test der Programme erfolgte aus Bequemlichkeit gleich auf dem Linuxrechner, hier diente Hatari 2.3.1 mit den jeweiligen Einstellungen für ST-RGB, Mono und STE-RGB als Testumgebung. Auch ein echter Atari ST kam zum Einsatz, wenn sich das Binary als prinzipiell lauffähig erwiesen hatte.

Um geeignete Programme zu finden, wurde der Assembler-Zweig des Repository ausführlich durchforstet und Spreu vom Weizen getrennt. Schnell wurde klar, dass es sich lohnt, für geeignete Programme einen eigenen Verzeichnisbaum anzulegen. Dieser sollte nur Quelltexte enthalten, die sich auch assemblieren ließen und das Programm danach auch lief. Das klappte nicht ohne Veränderungen an den Quellen. So mussten fast überall absolute Pfade bei INCBIN angepasst werden. Einige Unterverzeichnisse im Repository waren außerdem leer. Bei einigen Programmen ließen sich die Quellen nicht Cross-assemblieren,

manchmal lief ein scheinbar fehlerlos assembliertes Programm nicht.

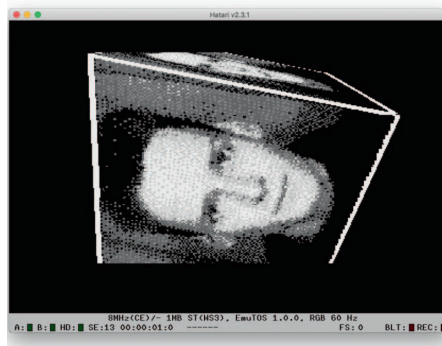
In den Unterverzeichnissen der auf diese Weise ausgesiebten Quellcode-Sammlung ist die Hauptquelldatei jeweils auf main.s kopiert und gegebenenfalls angepasst. Der Assemblerlauf wird mit

```
./vasm -Ftos -tos-flags=1
main.s -o main.tos
```

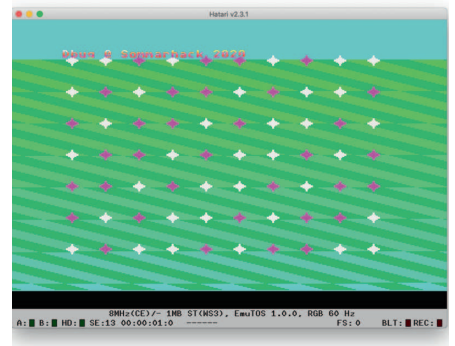
gestartet. Das fertige Binary main.tos findet sich dann in diesem Unterverzeichnis.

Einige besonders interessante Programme seien hier kurz erwähnt:

- _____ Cubemap zeigt einen drehenden Würfel mit einem Bild
- _____ Life ist eine Implementierung der bekannten "Life" Simulation für ST-Mono
- _____ Im Verzeichnis 0 finden sich einige kurze Demos sowohl für VASM als auch für GenST Assembler. Die dort assemblierten Binaries haben die Endung *.PRG



Cubemap



32 kByte Demo

In den Links am Ende des Artikels wird auf weitere Seiten verwiesen, die Tipps oder Tools und auch weitere Quellen für kurze Demos liefern. Insbesondere die DHS Demosysteme für ST und Falcon sind mehr als einen Blick wert.

Das Ausprobieren macht jedenfalls viel Spaß und vielleicht fühlt sich der eine oder andere Leser ja motiviert, in die Programmierung von Intros oder Demos in Assembler für den Atari ST oder Falcon einzusteigen. Wenn dieser kleine Artikel darauf Appetit gemacht hat, so ist sein Zweck erfüllt. (ps)

Links

Das Atari ST source code repository
https://github.com/ggnkua/Atari_ST_Sources
 VASM
<http://sun.hasenbraten.de/vasm/>
 Tools und Tipps
<https://dhs.nu/files.php?t=single&ID=138>
<https://nguillaumin.github.io/perihelion-m68k-tutorials/>
 Winzige Demos
http://www.sizecoding.org/wiki/Motorola_68k_based_CPUS
 DHS Demosysteme
<https://www.dhs.nu/>



Das Verzeichnis mit den ausgewählten Quellen findet sich auf der Heft-CD.

Der Klassiker der Datenbanken

Einstieg in dBASE



ASHTON·TATE

dBASE war das erste weithin genutzte datenbasierte Datenbankmanagementsystem (DBMS) für Mikrocomputer. Es wurde von dem Unternehmen Ashton-Tate ursprünglich für das Betriebssystem CP/M entwickelt und vertrieben. Später wurde die Datenbankapplikation auf den IBM-PC unter DOS portiert.

Seinen Ursprung hat dBASE in den 1960er Jahren. Im Jet Propulsion Laboratory der NASA wurde damals ein System namens RETRIEVE verwendet. Dieses wurde durch mehrere Personen im Laufe der Zeit angepasst, so letztendlich auch von einem Mann namens Wayne Ratliff. Dieser nannte seine Fortentwicklung VULCAN, Name des Heimatplaneten von Mr. Spock aus Star Trek.

Wechselvolle Geschichte

Ashton-Tate kaufte die Rechte an VULCAN und stellte auch die Entwickler dazu ein. Die Entwickler portierten VULCAN auf das Betriebssystem CP/M und Ashton-Tate nannte es dBASE II. Es gab keine „Version I“.

dBASE II war sehr erfolgreich, also wurde es auch auf andere 8-Bit System weiter portiert, z.B. Apple II (CP/M mit Z80-Prozessor-Karte). Später erschien auch eine Version für Atari ST. Im August 1982 veröffentlichte Ashton Tate eine 16 Bit Version für IBM PC unter dem Namen dBASE II 2.3. dBASE wurde zum erfolgreichsten PC-Datenbanksystem der 1980er Jahre.

Die Weiterentwicklungen waren in der zweiten Hälfte der 1980er Jahre die Releases dBASE III und dBASE III (plus). Hierbei wurden auch die neuesten Trends der PC-Technik berücksichtigt, wie Netzwerktechnologie mit Mehrplatznutzung oder die Integration von Schnittstellen zu Lotus 1-2-3 und MS-Word.

Im Jahre 1988 erschien dann dBASE IV. Ashton-Tate hatte hier viel investiert und versuchte, weitere moderne Konzepte einzubauen. Letztlich scheiterte diese Version jedoch an technischen Hürden wie der 640 KB Grenze. dBASE IV erwies sich als sehr langsam und einige Komponenten als schwer zu bedienen. Der Erfolg blieb aus und 1991 kaufte schließlich Borland das angeschlagene Unternehmen Ashton-Tate. Borland hatte bereits ein eigenes PC-Datenbank System namens PARADOX, doch das konnte nicht mit dem Erfolg von dBASE mithalten. Borland führte dBASE daher unter angepasstem Namen weiter und verkaufte es als Borland dBASE, dBASE for Windows und Visual dBASE.

Die Besitzverhältnisse von dBASE änderten sich im Laufe der Jahre weiter. Anfang der 2000er Jahre wurden die Rechte an ei-

ne Firma dBASE Inc. verkauft und im Jahr 2005 wanderten die Rechte an DataBased Intelligence Inc. weiter. Die Firma vermarktet es unter dem Namen dBASE plus. Der letzte Verkauf erfolgte im Jahre 2012 an die neu gegründete Firma dBASE LLC, die bis heute dBASE weiterentwickelt und verkauft.

Erste Schritte mit dBASE

Die Grundidee von dBASE ist es, eine relationale Datenbank gepaart mit einer Programmiersprache der vierten Generation in einer speziellen strukturierten Datei zu halten. Die dort entwickelte Syntax und das verwendete Datenbankformat (DBF) haben über Jahrzehnte den Bereich der desktopbasierten PC-Datenbank Systeme beeinflusst und bestimmt. Dieser Artikel zeigt exemplarisch die Schritte beim Entwerfen und Implementieren einer Datenbank. Die Schritte wurden auf einem IBM PC mit MS-DOS durchgeführt und lassen sich mit den Versionen dBASE II und III plus leicht nachvollziehen.

Die Arbeit beginnt in der Regel mit dem Entwurf eines Datenbankmodells der gewünschten Applikation unter Berücksichtigung der passenden Normalform. Zuerst legt der Benutzer also die Tabellen an, definiert dann die Indizes, Views und Reports und kann schließlich Programmcode integrieren, um weitere Funktionen und eine Oberfläche zur Verfügung zu stellen.

Das Programm dBASE wird unter MS-DOS einfach durch den Aufruf von dBASE gestartet.

```
C:\DBASE3P>dbase
```

DBASE begrüßt den Anwender und zeigt dann den dBASE Assistenten, der ab dBASE III vorhanden ist und eine Oberfläche im MS-DOS Menü Stil besitzt.

```

dBASE III PLUS Version 1.1
This Software is Licensed to:
=====
=====
Copyright (c) Ashton-Tate 1985, 1986. All Rights Reserved.
dBASE, dBASE III PLUS and Ashton-Tate are trademarks of Ashton-Tate

You may use the software and printed materials in the dBASE III
PLUS package under the terms of the Software License Agreement;
please read it. In summary, Ashton-Tate grants you a paid-up,
non-transferable, personal license to use dBASE III PLUS on one
computer work station. You do not become the owner of the
package nor do you have the right to copy (except permitted
backups of the software) or alter the software or printed
materials. You are legally accountable for any violation of the
License Agreement and copyright, trademark, or trade secret law.
```

```
Command Line | <C>: | 
Press ← to assent to the License Agreement and begin dBASE III PLUS._
```

Zuerst wird eine Datenbank mit einer entsprechenden Struktur auf einem passenden Laufwerk erstellt. Der Name der Datenbank ist gleichzeitig auch der Name der Datenbank-Datei mit der En-

dung DBF, also z.B. Datenbank Name ist TRAVEL, der Name der Datenbank-Datei ist TRAVEL.DBF:

Set Up **Create** Update Position Retrieve Organize Modify Tools 12:58:00 pm

Database file	A:
Format	B:
View	C:
Query	D:
Report	E:
Label	

```

.list all
00001 ALBERT EINSTEIN          LITTLE TOWN USA          (201)555-12
2 (201)555-1313 W
00002 ROBERT REDFORD          LOS ANGELES CA.          (213)555-05
5 (213)555-1000 H
00003 KILGORE TROUT          NOWHERE                   (312)555-00
0 (312)555-9999 W
00004 BOB HOPE                TOLUCA LAKE CA.         (213)555-77
7 (213)555-8888 H
00005 RONALD REAGAN          WASHINGTON DC             (212)555-50
0 CLASSIFIED H
    
```

Wenn die Datenbankstruktur dann geschaffen ist und die ersten Daten eingegeben wurden, sieht das Ganze im dBASE Assistenten so aus, hier der EDIT Mode (ein RECORD Anzeige):

CURSOR <-- -->	UP DOWN	DELETE	Insert Mode: Ins
Char: + +	Field: ↑ ↓	Char: Del	Exit/Save: ^End
Word: Home End	Page: PgUp PgDn	Field: ^Y	Abort: Esc
Pan: ^* ^~	Help: F1	Record: ^U	Memo: ^Home

Command: CREATE
ASSIST <C:> | Opt: 3/5 | Num

```

FIRSTNAME Claire
LASTNAME Buckman
PHONE (555)456-9059
TRAVELCODE C110
TRAVELPLAN 10-night Caribbean Island Cruise
DEPARTURE 10/24/85
COST 1199.00
PAID 1
AGENT MM
RESERDATE 07/15/85
NOTES memo
    
```

Dann wird die Struktur mit folgenden Feldern definiert, hier unten im Beispiel die Tabelle TRAVEL für die Verwaltung von Reisen:

Bytes remaining: 3864

CURSOR <-- -->	INSERT	DELETE	Up a field: ↑
Char: + +	Char: Ins	Char: Del	Down a field: ↓
Word: Home End	Field: ^N	Word: ^Y	Exit/Save: ^End
Pan: ^* ^~	Help: F1	Field: ^U	Abort: Esc

Field Name	Type	Width	Dec	Field Name	Type	Width	Dec
1 FIRSTNAME	Character	20		9 AGENT	Character	2	
2 LASTNAME	Character	20		10 RESERDATE	Date	8	
3 PHONE	Character	13		11 NOTES	Memo	10	
4 TRAVELCODE	Character	4					
5 TRAVELPLAN	Character	40					
6 DEPARTURE	Date	8					
7 COST	Numeric	10	2				
8 PAID	Logical	1					

Hier die Anzeige im BROWSE Mode des Assistenten:

CURSOR <-- -->	UP DOWN	DELETE	Insert Mode: Ins
Char: + +	Record: ↑ ↓	Char: Del	Exit: ^End
Field: Home End	Page: PgUp PgDn	Field: ^Y	Abort: Esc
Pan: ^* ^~	Help: F1	Record: ^U	Set Options: ^Home

MODIFY STRUCTURE <C:> | TRAVEL | Field: 1/11

```

FIRSTNAME----- LASTNAME----- PHONE----- TRAVELCODE
Claire Buckman (555)456-9059 C110
Rick Lisbom (555)456-3344 AV10
Iank Bicksby (555)966-1278 KS19
Lena Garnett (555)766-9121 GE23
Lisa Kafmanan (555)860-0300 H114
Jay Johnson (555)860-6784 KS19
Sara Collins (555)568-0953 H114
Rick Zambini (555)345-1945 C110
Kim Arlich (555)624-8773 JH10
John Montovan (555)566-3403 KS19
Jicky Gorenan (555)846-8801 WAB
    
```

dBASE lässt sich sowohl durch Menüs als auch durch Befehle auf der Kommandozeile bedienen. Mit ESC kommt der Benutzer vom Menü-basierten Assistenten zur Kommandozeile und mit dem Kommando ASSIST wieder zurück zum Assistenten. Über die Kommandozeile wird die Datenbank mit dem Befehl USE ausgewählt, die Anzeige der Datenbankstruktur erfolgt mit dem Befehl LIST STRUCTURE:

Copyright (C) 1982 RSP Inc.

```

*** dBASE II/86 Ver 2.4 1 July 1983
. use client
. list structure
STRUCTURE FOR FILE: C:CLIENT .DBF
NUMBER OF RECORDS: 00005
DATE OF LAST UPDATE: 00/00/00
PRIMARY USE DATABASE
FLD NAME TYPE WIDTH DEC
001 NAME C 030
002 ADDRESS C 030
003 WPHONE C 013
004 HPHONE C 013
005 PHPREF C 001
** TOTAL ** 00088
    
```

BROWSE <C:> | TRAVEL | Rec: 1/49

View and edit fields.

Dann beginnt im nächsten Schritt die Entwicklung von dBASE Code zur Erstellung von ausführbaren Programmen unter dBASE. Diese haben immer die Endung PRG.

Im Wesentlichen beinhaltet die Sprache viele Elemente einer damals modernen Hochsprache der vierten Generation. Hierbei können sowohl Programme entwickelt werden, die etwas mit den Datensätzen an sich tun, aber es kann auch eine dialogorientierte Anwendung entwickelt werden, welche den User über eine Dialogoberfläche durch die Nutzung der Datenbank führt.

Die Kommandos zur Pflege und Änderung von Datenbanken und Attributen sind im Code ebenso nutzbar wie von der Kommandozeile in dBASE.

. modify command checks.prg

Ueber den Autor

Bernd Dohr arbeitet als Softwareentwickler und interessiert sich seit den 90er Jahren fuer mobile Computer, DOS Portables, Laptops sowie PC-Betriebs- und -Datenbank Systeme.

Zur Anzeige aller Daten wird der Befehl LIST ALL aufgerufen, das Ergebnis ist dann die Anzeige aller Datensätze.


```

STORE 0 TO OK
DO WHILE OK = 0
  IF '&PROMPT' # ' '
    ACCEPT '&PROMPT' TO COMMAND
  ELSE
    ACCEPT TO COMMAND
  ENDIF
  STORE ! (COMMAND) TO COMMAND
  IF COMMAND = 'QUIT'
    QUIT
  ENDIF
  IF COMMAND # CHECK
    DO CLEAR@
    ?"You didn't enter"
    ?" &CHECK"
    ?"Please try again."
    LOOP
  ELSE
    STORE 1 TO OK
    &COMMAND
  ENDIF
ENDDO
STORE ' ' TO PROMPT

```

If this is the first time you've used dBASE II LESSONS, select the 1 option from the menu below. This introduction contains background material on computers, operating systems, and dBASE II.

If this is not the first time you've used dBASE II LESSONS, select the appropriate lesson number (or QUIT) from the menu below.

- I - Introduction, definitions of basic computer terms (reference, not interactive)
- 1 - Create files, generate Reports, and correct syntax errors
- 2 - Add, change, and delete data in files, and record positioning
- 3 - Selecting sub-sets of data on files, and computations
- 4 - Record keys, indexing, random processing, and Boolean operators
- 5 - File and system utilities, and intro to command procedures
- 6 - Sorting, updating, and use of memory
- 7 - Relational data base construction
- 8 - Other forms of accepting and displaying data, dBASE II functions, and SET options
- 9 - Building custom reports and screens
- 10 - Command procedures (programming features in dBASE II)
- QUIT - In response to any prompt, to return to the operating system

ENTER THE LESSON NUMBER: _

Fazit

Ein Programm wird gestartet mit dem Kommando DO. Als Muster kann hier die mitgelieferte Anwendung lessons.prg dienen.

C:\DBASEII>dbase

Copyright (C) 1982 RSP Inc.

*** dBASE II/86 Ver 2.4 1 July 1983
. do lessons.prg

DBASE Datenbanken sind auch heute noch nutzbar, da viele moderne PC-Datenbanken wie beispielsweise Microsoft Access das DBF Format der dBASE Datenbankdateien unterstützen. Heutzutage werden jedoch moderne Benutzeroberflächen im Webbrowser oder unter Microsoft Windows genutzt, um mit dBASE Dateien zu arbeiten.

Links

<https://de.wikipedia.org/wiki/DBASE>

Das Unternehmen hinter dBASE Ashton-Tate

George Tate und Hal Lashlee gründeten das Unternehmen Ashton-Tate im Jahre 1980. Die erste Entwicklung war ein einfaches Programm für den Betrieb eines Fußball-Tippspiels im Büro. Mit der Übernahme der Vulcan-Datenbank und der Weiterentwicklung zu dBASE wurde aus dem kleinen Softwarehaus ein multinationales Unternehmen, das viele Millionen US-Dollar umsetzte. Tate und Lashlee hatten vor der Gründung von Ashton-Tate bereits drei erfolgreiche Start-Up-Unternehmen gegründet, nämlich Discount Software, Software Distributors und Software Center International. Discount Software war eines der ersten, die PC-Software direkt über den Postweg an den Endanwender verkaufte. Software Center International war eine Einzelhandelskette für Software in den USA mit Geschäften in 32 Staaten.

dBASE II

Mit dieser Erfahrung im Verkauf von Software stieg Ashton-Tate mit einer ungewöhnlichen Strategie in den Verkauf von dBASE II ein. Die Käufer erhielten eine eingeschränkte Demoversion sowie die Vollversion auf einer separaten, versiegelten Diskette. Innerhalb von 30 Tagen konnten sie die ungeöffnete Diskette zurückgeben und erhielten den Kaufpreis zurück. Dies senkte die Hemmschwelle der Käufer, den doch stattlichen Betrag zu investieren und sorgte für einen guten Absatz des Produkts.

Im November 1983 ging Ashton-Tate an die Börse, was 14 Millionen US-Dollar in die Kassen des mittlerweile auf 226 Mitarbeiter angewachsenen Unternehmens spülte. Der Erfolg setzte sich fort: Am Ende des Geschäftsjahres hatten sich die Einnahmen auf 43 Mio. US-Dollar mehr als verdoppelt, der Nettogewinn war von 1,1 auf 5,3 Mio. US-Dollar gestiegen. Das meiste Geld machte Ashton-Tate dabei mit dBASE II und zusätzlich angebotenen Tools.

dBASE III

War dBASE II noch eine Assemblerentwicklung, schwenkten die Entwickler mit der Nachfolgeversion dBASE III Version 1.0 im Juli 1985 auf C als Programmiersprache um. Mit der Version 1.1 musste dann im November 1985 rasch ein Bugfix-Release hinterher geschoben werden. Die beste Stabilität wies dann die Version 1.2 auf, die Ashton-Tate als dBASE III Developer's Edition verkaufte. Bald danach folgte dBASE III+ (Version 2.0) mit vielen Optimierungen. Ende 1987 verbuchte Ashton-Tate 318 Mio. US-Dollar. Der Erfolg zog natürlich auch Nachahmer und Dritthersteller an. Insbesondere Nantucket Software mit dem dBASE Compiler Clipper machte von sich reden. Clipper erlaubte es, eine dBASE Anwendung in ein selbstständig lauffähiges Programm zu übersetzen. Das Programm benötigte selbst dann keine dBASE-Software mehr. Das war ein großes Ding für die Anwender, aber ein Dorn im Auge von Ashton-Tate. Die eigene, 395 US-Dollar teure Runtime-Lizenz für den Einzelplatz zum Betrieb von dBASE Anwendungen wurde nicht mehr gebraucht.

dBASE IV

Mit dBASE IV konnte Ashton-Tate seine Erfolge allerdings nicht wiederholen. Das im Juli 1988 ausgelieferte Produkt sollte netzwerkfähig sein, SQL unterstützen und eine schnelle Anwendungsentwicklung erlauben. Aber die erste Version war instabil und fehlerbehaftet. Der mitgelieferte Compiler erzeugte keinen nativen Code für die Zielplattform, sondern nur einen Object Code, der die dBASE IV Laufzeitumgebung brauchte. Aber anstatt schnell ein Bugfix Release nachzuschicken, konzentrierte sich Ashton-Tate auf andere Entwicklungen und vernachlässigte sein Flaggschiff. Als dann im Juli 1990 die Version 1.1 auf den Markt kam, hatte Ashton-Tate bereits deutlich Federn lassen müssen. Im August 1989 hatte das Unternehmen schon 400 seiner 1.800 Mitarbeiter entlassen müssen, der Marktanteil bei Datenbanken war auf 43% geschrumpft. Schließlich übernahm im Jahr 1991 dann Borland, was

vom ursprünglichen Primus der Datenbanken übrig war.

Weitere Software

Ashton-Tate allein auf dBASE zu reduzieren, tut den Entwicklern aber unrecht.

— Schon 1983 kam mit Friday! eine dBASE-basierte Anwendung für die Kontakt- und Aufgabenverwaltung auf den Markt.

— Ashton-Tate vertrieb seit 1986 auch die Kalkulationssoftware Javelin auf dem internationalen Markt.

— Framework ist ein integriertes Programm mit einer Textverarbeitung, einer Tabellenkalkulation, Grafikfunktionen, einer Datenbank und einem Terminalprogramm. Framework läuft unter MSDOS und bringt seine eigene GUI mit.

— Multimate ist eine Textverarbeitung aus dem Jahr 1985, die Funktionen ähnlich einer Wang Workstation für Textverarbeitung auf den PC bringen sollte. Gegenüber dem Konkurrenten WordPerfect machte Multimate aber einen arg angestaubten Eindruck, der Markterfolg blieb aus.

— Byline, ein von SkiSoft entwickeltes und ab 1987 von Ashton-Tate vertriebenes Desktop Publishing Programm, hatte durchaus begeisterte Anwender. Anders als andere DTP Programme besitzt aber Byline keine GUI, sondern verlangt eine genaue Beschreibung des Seitenlayouts.

(gb)

Links

<https://en.wikipedia.org/wiki/Ashton-Tate>

Programmspeicher nicht nur für Commodore 64

Die Universal Cartridge

Typisch für einen Homecomputer der 1980er Jahre besitzt auch der Commodore 64 einen Steckplatz für Erweiterungsmodule. Sie bringen Software auf den Brotkasten, ohne das hierfür ein teures Floppy Disk Laufwerk nötig wäre. Hier stellen wir ein Modul vor, das sich einfach selbst bauen und beladen lässt.

Für den C64 gibt es eine große Anzahl an Modulerweiterungen. Grundsätzlich sind Software-Module von Hardware-Modulen zu unterscheiden. Software-Module bringen -- wie der Name es erwarten lässt -- neue Software auf den Rechner. Die Auswahl reicht von einfachen Spielmodulen bis zu komplexen Anwendungen. Hardware-Module enthalten hingegen Erweiterungen wie IEEE-488 und RS-232-Schnittstellen, Hardware-Speeder oder neue CPUs. In diesem Artikel geht es um Software-Module und darum, wie sich eigene Software zusammensetzen und auf ein Modul bringen lässt.

C64 Module allgemein

Software für den C64 gibt es bekanntermaßen entweder auf Compact Cassetten, Floppy Disks oder eben auf Modulen. Letztere zeichnen sich durch einfache Handhabung aus: Einfach Modul in den C64 stecken, einschalten fertig. Oberflächlich scheinen Software-Module identisch zu sein, doch ein genauer Blick offenbart Unterschiede. Der Commodore 64 unterscheidet ursprünglich drei Modi, nämlich den 8K Modus, den 16K Modus und den 16K UltiMax Modus. Später wurde aus Gründen der Kapazitätssteigerung bei Modulen das sogenannte "Banking" eingeführt. Es umgeht die Größenbeschränkung von 16KB. Aber auch Module mit Banking sind zu jedem Zeitpunkt auf 16K und die drei grundsätzlichen Modi beschränkt. Nur sind hier die 16K Modulspeicher aus einem größeren Speicherbereich zu wählen. Der Banking Mechanismus ist aber nicht

standardisiert, weswegen Software auf dem Modul auf genau einen bestimmten Modultyp festgelegt ist.

Die Universal Cartridge

Praktisch sind sie also, die Module für den Commodore 64. Deshalb werden auch heute noch neue Programme für den Rechner als Module veröffentlicht, so zum Beispiel das beliebte Spiel "Sams Journey" von Protovision. Noch praktischer wäre es, würde es eine Möglichkeit geben, selbst nahezu beliebige Programme unkompliziert auf ein Modul zu bringen. Der Autor hat sich dieser Aufgabe angenommen und mit der Universal Cartridge eine solche Möglichkeit realisiert.

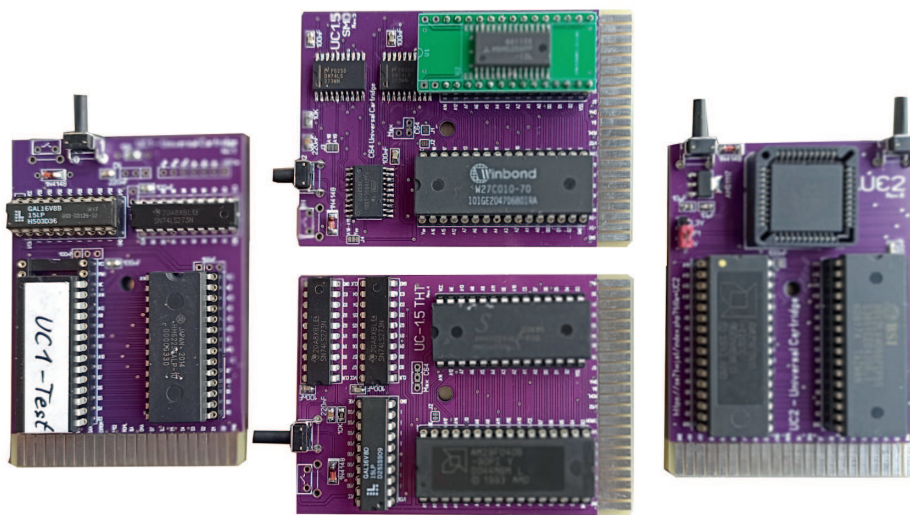
Die Universal Cartridge oder kurz "UC" ist ein Software-Modul für den C64, auf dem sich ein oder mehrere Programme und Spiele speichern und am C64 ausführen lassen. Das UC Modul hat einen eigenen Banking Mechanismus und kann alle drei Modul-Modi per Software emulieren. UC Module können folgende Programme starten:

- _____ Software für 8K Module
- _____ Software für 16K Module
- _____ Software für Ultimax Module
- _____ Software, die als eine einzige Datei geladen und mit "RUN" gestartet werden (One-File)
- _____ Software, die als eine einzige Datei geladen und mit RESET oder SYS gestartet werden

Es gibt drei verschiedene Versionen der Universal Cartridge und alle drei sind für den Selbstbau gedacht. Dementsprechend wurde auf einen möglichst schlichtes und leicht nachzubauendes Design geachtet. Die Unterschiede in Bezug auf Bauteile und Bauweise sind in der Tabelle zusammengefasst. Genaue Anleitungen für den Aufbau finden sich auf der Homepage des Autors.

Vergleich der verschiedenen Universal Cartridge Modelle

	UC1	UC2	UC1.5
Bauweise	TFT	TFT	SMD oder TFT
EPROM	64 oder 128K (zB. W27C512 oder W27C010)		128, 256 oder 512K (zB. W27C010 oder W27E040)
Flashspeicher		512KB (zB. AM29F040)	AM29F040 (nur TFT Version)
SRAM	32KB	512KB	32KB
Logikbaustein		CPLD ATF-1504	
TTL Baustein(e)	74LS273		2 x 74LS273
GAL	ein GAL 16v8		ein GAL 16v8
Weitere Bauteile	4 Stück Kondensatoren (100nF)		6 x Kondensatoren (100nF) 1 Widerstand (10K Ohm)
Optionen	LED Board (6 x LED)	Reset Taster A und B	Reset Taster A und B, eine Diode



Die fertigen Universal Cartridges

UC1

Das kleinste UC Modul ist die UC1, das seit Mitte Juli 2021 verfügbar ist. Es unterstützt ein EPROM mit einer Größe von 64KB oder 128KB. Zusätzlich sind 32kByte SRAM vorhanden, mit dem die C64 Standardmodule (8K, 16K, Ultimax) zu emulieren sind. Alle Bauteile sind in THT Technik aufgelötet, daher kann dieses Modul auch von einem unerfahrenen Bastler selbst hergestellt werden.

UC2

Das UC2 ist der große Bruder des UC1 Moduls und wurde Anfang Oktober 2021 veröffentlicht. Es unterstützt Flashspeicher bis zu 512KB (128K, 256K und 512K Flash Speicher). Das SRAM hat ebenfalls eine Größe von 512KB. Auch dieses Modul verzichtet des einfachen Nachbaus zuliebe auf SMD Bauteile.

UC1.5

Das UC1.5 ist ein Modul zwischen UC1 und UC2. Es hat wie die UC2 zwei Register und kann daher ebenfalls 512 KByte große EPROM Speicher adressieren. Der

SRAM Speicher ist wie bei der UC1 32K groß. Die Bauteile sind in SMD Technik aufgelötet, damit sie auf der kleinen Platine Platz finden.

Beladen der Cartridges

Um eine persönliche Programmsammlung für ein UC Modul zu erstellen, gibt es das Programm UC-Builder für den Windows-PC. Der UC-Builder nimmt die gewünschten C64 Programme und verpackt sie in eine einzelne Datei, die UC Image Datei. Diese Datei wird in den Speicher (EPROM, Flash) des UC Moduls geschrieben. Damit steht die Programmsammlung dem C64 zur Verfügung. Die C64 Programmsammlung auf dem UC Modul kann frei zusammengestellt werden, solange die Programme in eine der oben genannten Kategorien fallen. Die Größe der UC Image Datei darf den verfügbaren Platz im EPROM beziehungsweise dem FLASH Speicher natürlich nicht überschreiten. Wird ein W27C040 verwendet, darf die Image Datei eine maximale Größe von 512KB haben. Die Größe setzt sich zusam-

men aus der Summe der einzelnen C64 Programme plus der Größe des UC-Loaders. Der UC-Loader ist ein C64 Programm, das zur Bedienung des Moduls notwendig ist und das UC-Menü bereitstellt.

Der UC-Builder

Der UC-Builder ist nur erforderlich, wenn eigene Zusammenstellungen von Programmen auf eine UC geladen werden sollen. Fertige Imagedateien lassen sich mit einem EPROM Programmiergerät direkt aufbringen.

Der UC-Builder erstellt eine UC Image Datei aus einem oder mehreren C64 Programmen. Welche Programme das sind, beschreibt der Anwender in einer CSV Datei; deren Aufbau ist im Kasten am Ende des Artikels beschrieben. Die Image Datei enthält den UC-Loader, das UC-Menü für die Programmauswahl und optional einen File Browser (UC-FB) und natürlich alle ausgewählten C64 Programme. Der UC-Loader ist in dem UC-Builder bereits enthalten und wird immer automatisch in die Imagedatei geschrieben, das UC-Menü wird vom UC-Builder jeweils passend erzeugt.

Um ein UC-Menü zu erstellen, braucht der UC-Builder folgende Informationen zu jedem C64 Programm:

- _____ Name des Programms
- _____ Art des Programms
- _____ Dateinamen des Programms

Der Name des Programms wird beim Start vom UC-Loader angezeigt. Die Art des Programms benötigt der UC-Loader, damit das Programm korrekt gestartet werden kann. Die Dateiname des Programms benötigt der UC-Builder, damit er es in die UC Image Datei integrieren kann.

Der UC-Builder ist ein Programm für die Windows Kommandozeile. Es erstellt Image Dateien für das UC1, UC1.5 und das UC2 Modul. Wird es ohne Argumente aufgerufen, dann sucht das UC-Builder nach einer CSV Datei namens menu.csv. Wenn beim Bau des UC-Image keine Fehler auftreten, dann erfolgt nur die Ausgabe der Größe der Imagedatei sowie des freien Speicherplatzes. Die Berechnung des freien Speicher beruht auf der Annahme, dass ein 128K EPROM (W27C010) verwendet wird. Bei Verwendung eines 64K EPROM ist also darauf zu achten, dass die Image Größe 0x10000 nicht überschreitet.

Am Ende wirft der UC-Builder zwei Dateien aus: Die BIN Datei (EPROM Abbild) ist die UC Image Datei, die direkt in das EPROM für das Modul gebrannt werden kann. Die CRT Datei ist ein 8K CRT für den VICE, damit lässt sich das Menü im Emulator kontrollieren. Allerdings kann der VICE die Programme nicht richtig starten, da

Der UC-Builder

Der UC-Builder kennt folgende Optionen:

- h oder --help: Ausgabe aller verfügbaren Optionen zum UC-Builder.
- v oder --version: Version des UC-Builder .
- 1 oder --UC1: Image Datei für ein UC1 Modul erstellen.
- 2 oder --UC2: Image Datei für ein UC2 Modul erstellen. Diese Image Dateien laufen auch in der UC1.5
- 5 oder --UC1.5: Image Datei für ein UC1.5 Modul zu erstellen. Diese Image Dateien laufen auch in der UC2
- 8 oder --MD: Image Datei für ein Magic-Desk Modul zu erstellen. Die maximale Größe ist 1MB.
- 9 oder --EF: Image Datei für ein EasyFlash Modul zu erstellen.
- m oder --menu-csv: Angabe des Namens der zu benutzenden CSV Datei
- i oder --image-file: Angabe des Namens der Ausgabedatei (UC Image Datei)
- c oder --crt-file : Angabe des Namens der Ausgabedatei (CRT Datei).
- v ; erhöht man die Anzahl der Meldungen, das UC-Builder wird gesprächiger. Die Option kann mehrfach verwendet werden. Ab drei -vv erhält man detaillierte Informationen über den Aufbau der Image Datei, also die Adresse, Bank und Länge jedes Programmes in dem Image. Es wird auch angezeigt wie das Programm geladen und gestartet wird und ob das IO Register verborgen wird.

der VICE die UC Hardware derzeit nicht emulieren kann.

Die Image Dateien für Magic-Desk oder EasyFlash bestehen wie bei den UC-Images aus einer BIN Datei (EPROM Abbild) und einer CRT Datei für den VICE Emulator. Neuere Versionen des VICE Emulator können beide Arten (EasyFlash und Magic-Desk) perfekt emulieren.

Der UC-Loader

Nach dem Einschalten des C64 startet der UC-Loader. Der UC-Loader schaut nach, welche Programme auf dem Modul gespeichert sind. Ist es nur ein Programm, dann wird dieses direkt gestartet. Sind mehrere Programme auf dem Modul, dann zeigt der UC-Loader die Namen aller Programme im UC-Menü an. Der Benutzer vermag dann das gewünschte Programm auszu-

wählen und der UC-Loader startet es. Mit den Tasten <F7> oder <F8> geschieht der Wechsel in den normalen C64 Eingabemodus (C64 BASIC). Die Taste <F7> schaltet dabei das UC Modul aus. Der Zugriff auf die UC-Register an der Adresse \$DE0x funktioniert dann aber weiterhin. Mit der Taste <F8> werden zusätzlich auch die UC Register ausgeschaltet. Das Modul ist dadurch vollkommen unsichtbar und kann nur durch einen Modul Reset wieder aktiviert werden.

Der UC File Browser

Der UC File Browser wird vom UC-Menü mit der Taste <F1> gestartet. Der UC File Browser ist nur verfügbar, wenn im CSV File ein Dummy Eintrag gemacht wird. Der Menütext (erste Spalte) muss 'F1' lauten und die zweite Spalte enthält drei Bin-

destriche ('---'). Die dritte Spalte wird nicht ausgewertet. Optional kann man statt des internen UC-FB auch einen eigenen Browser einbinden, dazu muss nur die erste Spalte exakt 'F1' enthalten. Der UC-FB läuft als 8KB ROM und ist ansonsten wie der normale FB v2 zu verwenden. Er startet Programme, wechselt in Verzeichnisse oder Diskimage-Dateien und verlässt diese auch wieder. So ist eine sehr komfortable Navigation auf einem SD basierten Laufwerk (wie das SD2IEC) möglich. Der UC-FB kann auch CRT Dateien vom Typ 0 starten. Es wird automatisch der Modul Typ (8K, 16K oder Ultimax) aus dem CRT File gelesen und entsprechend in den UC RAM geladen. Der Start erfolgt über einen Soft Reset.

CSV Datei für den UC-Builder

Der UC-Builder erwartet eine CSV Datei mit einer Beschreibung derjenigen Programme, aus denen die UC Image Datei gebaut werden soll. Die Datei kann aus mehreren Zeilen bestehen, jede Zeile muss in folgender Form aufgebaut sein:

Programmname ; Programmtyp ; Dateiname ; Startoptionen

Die ersten drei Felder sind verpflichtend, das vierte Feld ist optional und erst seit Version 1.05 des UC-Builders vorhanden.

Feld 1: Programmname

Der Programmname wird nur für die Darstellung im UC-Menü benutzt. Sofern möglich, wird der Name automatisch in PETSCII konvertiert. Die Länge wird automatisch auf 17 Zeichen reduziert, damit der Name am C64 problemlos darstellbar ist.

Hierbei ist F1 ein reservierter Name. Der UC-Builder interpretiert diesen Eintrag als Aufforderung, einen File Browser einzubinden. Dieser erscheint nicht als Eintrag im UC Menü, sondern ganz unten links als Auswahl 'F1-File Browser'. Mit der Taste 'F1' wird dieser dann gestartet, und zwar immer mit aktiviertem IO Register. So ist später eine Konfiguration des UC Modul möglich. Der File Browser kann entweder ein externes Programm oder der interne UC-File Browser sein. Soll dieser eingebunden werden, so muss als Programmtyp '---' angegeben werden, ein Dateiname ist nicht erforderlich.

Feld 2: Programmtyp

Der Programmtyp steuert, wie das Programm geladen wird und bestimmt die Konfiguration der UC Hardware vor dem Programmstart. Zudem wirkt sich der Programmtyp auf die Startoption aus, sofern diese nicht manuell in Feld 4 eingestellt wird.

Folgende Kürzel sind in der Spalte Typ erlaubt:

- ___ PRG: normales BASIC Programm, das an die Adresse \$0801 geladen und mit RUN gestartet wird.
- ___ BIN: Ein Programm mit Ladeadresse, das an eine beliebige Adresse geladen und mit SYS oder RESET gestartet wird.
- ___ BLK nnn und ! nnn: Eine Datei OHNE Ladeadresse, die an eine beliebige Adresse geladen und mit SYS oder RESET gestartet wird. Es kann entweder 'BLK' oder '!' verwendet werden, das Ergebnis ist identisch. Die Datei wird an die angegebene Adresse 'nnn' geladen. Die Adresse kann dezimal, hexadezimal (beginnt mit '\$' oder '0x') oder Oktal (beginnt mit '0') angegeben werden.
- ___ 8KB: Das Programm wird als 8KB Modul gestartet.
- ___ 16KB: Das Programm wird als 16KB Modul gestartet,
- ___ UltiMax: Das Programm wird als Ultimax Modul gestartet,
- ___ CRT: Der Modultyp wird selbstständig erkannt (nur bei CRT Dateien)
- ___ "---": Sonderfall für die Verwendung des internen UC-FB, wenn der Programm Name exakt 'F1' ist

Feld 3: Dateiname

Dies ist der Name der einzubindenden Datei auf dem PC; er wird benötigt, damit der UC-Builder die Datei lesen und in die Image Datei einbinden kann. Es sind alle Arten von Pfaden erlaubt, die dem Betriebssystem bekannt sind, also Unterverzeichnisse, andere Laufwerke, absolute Pfade, Netzlaufwerke und so weiter. Dateien werden normalerweise

einfach als Byte Stream gesehen. Es gibt aber Ausnahmen, die inhaltlich interpretiert werden, nämlich die reservierten Namen P00 und CRT:

___ P00: Diese Dateien werden besonders behandelt, der P00 Header wird überlesen und ab Byte [0x1A] als Inhalt verwendet

___ CRT: Diese Dateien sind 'Modul Dateien' für den Emulator. Sofern es CRT Dateien vom Typ 0 (8K, 16K, Ultimax) sind, werden sie vom UC Builder und vom UC File Browser richtig interpretiert.

Feld 4: Startoptionen

Dieses Feld ist optional und dient dazu, bei Bedarf eine bestimmte Startoption zu erzwingen. Ohne eine Angabe in diesem Feld startet der UC-Loader alle PRG-Dateien mit einem 'RUN' Befehl und alle Moduldateien mit einem Soft-RESET. Bei allen springt das Modul nach dem LOAD ins BASIC. Ist eine Startoption definiert, so führt der UC-Loader diese nach dem LOAD aus, und zwar unabhängig davon, ob sie einen Sinn ergibt oder nicht.

Folgende Startoptionen sind erlaubt:

- ___ RESET: Nach dem Laden des Programmes wird ein Soft RESET ausgeführt. Das ist die Vorgabe bei Programmen des Typs 8KB, 16KB, UltiMax und CRT.
- ___ SYS nnn: Nach dem Laden des Programmes wird ein SYS auf die Adresse 'nnn' ausgeführt. Die Startadresse kann dezimal, hexadezimal (beginnt mit '\$' oder '0x') oder Oktal (beginnt mit '0') angegeben werden.
- ___ RUN: Nach dem Laden des Programmes wird ein RUN ausgeführt. Sofern es sich bei dem Programm um ein BASIC Programm handelt, wird es automatisch nach dem LOAD ausgeführt. Das ist die Vorgabe bei Programmen des Typs PRG.
- ___ READY: Nach dem Laden des Programmes springt der UC-Loader direkt nach dem LOAD in den Direktmodus des BASIC (Interpreter Loop). Das ist die Vorgabe bei Programmen des Typs BIN und BLK.
- ___ ROM: Das Programm wird direkt im ROM (EPROM/FLASH) der Universal Cartridge ausgeführt, ohne zuerst ins RAM geladen zu werden. Bei der Startoption ROM selektiert der RESET: Nach dem Laden des Programmes wird ein Soft RESET ausgeführt. Das ist die Vorgabe bei Programmen des Typs 8KB, 16KB, UltiMax und CRT.

Bei der Startoption ROM UC Loader selektiert die richtige Speicherbank und startet dann per Reset. Module vom Typ 8K und 16K werden direkt an der unteren Grenze einer 16K Bank angeordnet. Ultimax Module mit 8K werden automatisch in dem oberen 8K Block einer Bank angeordnet, das ist Standard für Ultimax Module wegen dem Reset Vektor. Diese Methode hat den Nachteil, dass ein Programm genau an den Blockgrenzen einer Bank angeordnet werden muss. Dadurch entstehen unter Umständen Lücken zum vorher geladenen Programm, der Speicher kann nicht mehr vollständig benutzt werden. Aber diese Option eignet sich gut, wenn hauptsächlich Modulprogramme (CRT Dateien) mit 8K und 16K Größe benutzt werden, beispielsweise das Multi-MAX Modul. Bei vielen 8K Modulen wird der Speicher ideal genutzt, wenn sich eine 8K und eine Ultimax immer abwechseln.



fertiges UC Menü

Fazit

Wer nun Lust bekommen hat, sich selbst an den Bau einer Universal Cartridge zu wagen, dem seien die ausführlichen Anleitungen auf der Homepage des Autors ans Herz gelegt. Viel Spaß mit dem Nachbau und der Verwendung der Universal Cartridge.

Links

https://oe7twj.at/index.php?title=Universal_Cartridge
https://oe7twj.at/index.php?title=Universal_Cartridge_1
https://oe7twj.at/index.php?title=Universal_Cartridge_2
https://oe7twj.at/index.php?title=Universal_Cartridge_1.5
<http://www.harries.dk/files/C64MemoryMaps.pdf>

Commodore MaxMachine

Die UC Module sind dafür konzipiert, in einem C64 Computer zu arbeiten. Neuere UC Module können aber auch in einer Commodore MAX Machine laufen. Die Module UC-1 und UC-1.5 haben dazu einen optionalen Jumper, mit dem sie in den MAX Machine Modus geschaltet werden können. Das UC-2 Modul kann per Software Einstellung in den MAX Machine Modus wechseln. Ein UC Modul im Max Machine Modus läuft auch in einem C64.

Der MAX Machine Modus unterliegt den Beschränkungen dieser Zielplattform:

- _____ es funktioniert nur noch der Ultimix Modus
- _____ der UC Loader funktioniert in diesem Modus nicht
- _____ ein C64 hat nur noch 4K RAM

- _____ an einer MAX Machine sind 2K RAM (intern) und 2K extern (UC-RAM) von \$0800 bis \$0x0FFF
- _____ das Signal IO-1 ist direkt mit /EX-RAM verbunden
- _____ das Signal IO-2 ist direkt mit /EX-ROM verbunden
- _____ die UC Register sind in einer MAX Machine in dem 2K Adressraum \$0800-\$0FFF ansprechbar (statt wie gewohnt ab \$DExx)
- _____ an einem C64 hat man 256 Byte UC RAM im IO-2 Bereich (\$DFxx)

Da der UC Loader im Max Machine Modus nicht funktioniert, sind auch UC Images nicht mehr nutzbar. Es braucht eine Software, die speziell für diesen Modus entwickelt wurde. Dies ist zum Beispiel die MultiMax Software für das UC Modul.

Ueber den Autor

Thomas Winkler ist Ingenieur und beschaeftigt sich in seiner Freizeit mit dem 6502, Mikrocontrollern, CPLD und FPGA. Er lebt in Oberhofen im oesterreichischen Inntal.

...und wann setzen Sie sich hin und schreiben für die LOAD?

Machen Sie mit und werden Sie Autor für das Retrocomputer-Magazin aus dem Verein zum Erhalt klassischer Computer e.V.

Wir suche ständig interessante Themen rund um klassische Computer, Homecomputer, Minicomputer, Programmierung, klassische Spiele und... natürlich über das Thema, das sie gerade am allermeisten interessiert.

Melden Sie sich noch heute bei uns:
redaktion@load-magazin.de



Mit dem Raspberry Pi-Zero

Commodore 1541 Laufwerk emulieren

Das Commodore 1541 Laufwerk ist selbst ein Computer. Er besteht aus einer CPU, ROM, RAM, I/O-Geräten und der Laufwerksmechanik. Es existieren einige Projekte, die dieses Gerät durch moderne Hardware emulieren und anstelle von Disketten auf SD Karten und Floppy Images setzen. Ein kostengünstiges Gerät zum Selbstbau stellt dieser Artikel vor.

Die Idee, mit einem Raspberry Pi eine Laufwerks-Emulation für den Commodore 64 und 128 zu bauen, ist nichts neues mehr, Das wurde bereits von Steve White vor einigen Jahren mit dem Pi1541 verwirklicht. Im Gegensatz zum SD2IEC emuliert der Pi1541 einen 6502 und die beiden 6522er. Da der Pi1541 Code auf seinem emulierten 6502-Kern ausführt, unterstützt er eine große Anzahl von Schnellladern (Spiele und Demoszenen), sogar kopiergeschützte Originale. Ursprünglich diente ein Raspberry Pi3 (Modell 3B, 3B+ oder 3A+) als Steuer-Rech-

ner. Dessen ARM-Prozessor enthält mehrere Kerne und so vermag etliches gleichzeitig intern abzulaufen.

Pi Zero als Basis

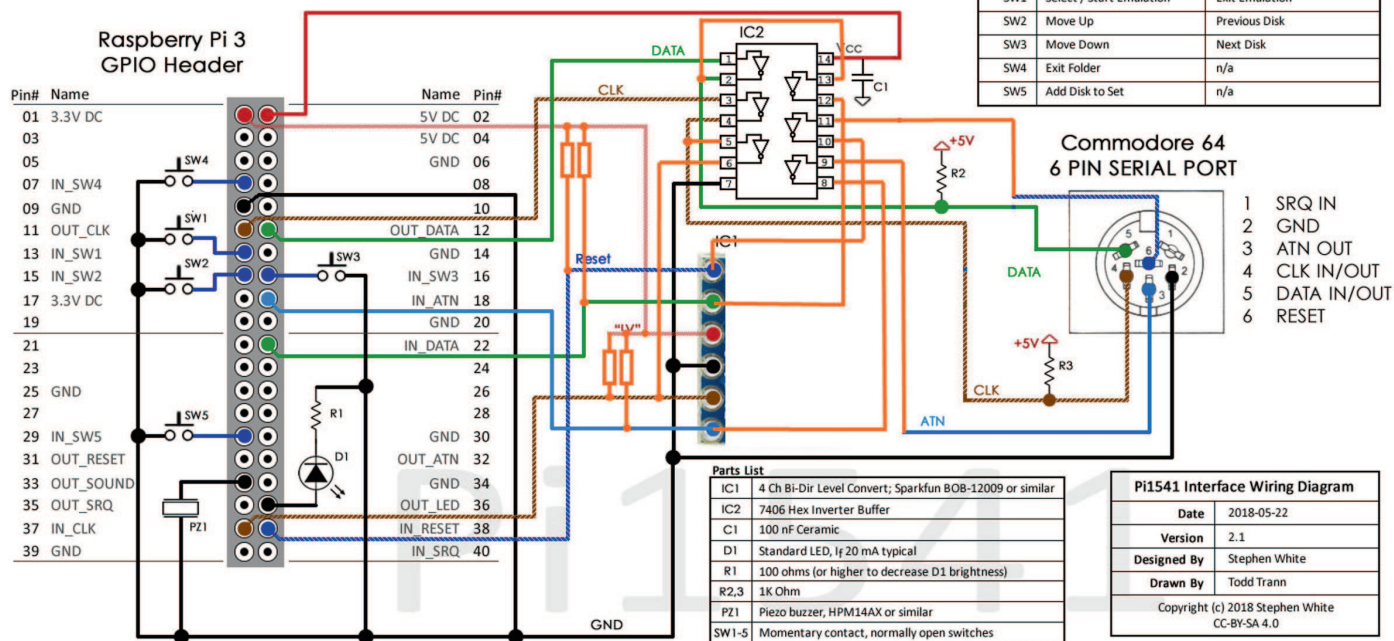
Allerdings sind diese Mini-Computer im Vergleich zu anderen Lösungen nicht billig. Daher kam schon früh der Wunsch auf, ob das ganze nicht vielleicht auch mit einem sehr viel günstiger zu beschaffendem Pi Zero zu realisieren wäre. Steve White setzte sich also daran, den Kernal seiner Emulation umzuschreiben, damit auch ein Pi Zero mit nur einem Kern so eine Aufga-

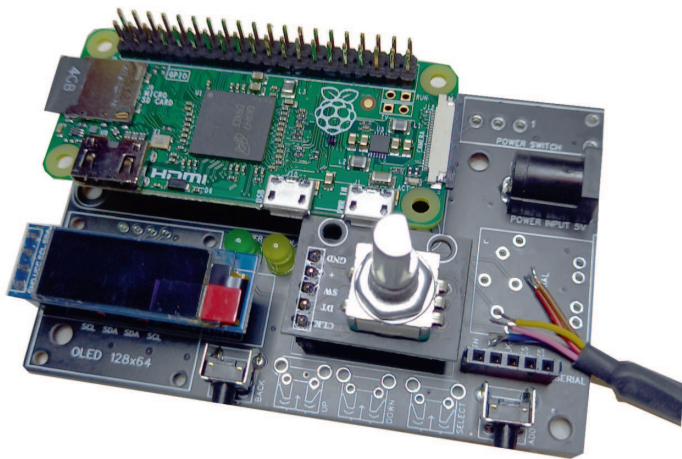
be übernehmen kann. Zwar sind mit dem Pi Zero als Steuer-Rechner gewisse Abstriche nötig. Zum einen gibt es keine Bildschirm-Ausgabe auf HDMI wie bei Pi 3. Zum anderen wurde anfangs auch kein Piezo-Speaker unterstützt wie bei den "großen Brüdern". Dort dient dieser dazu, Laufwerksgeräusche nachzuahmen und damit die Emulation retromäßiger wirken zu lassen. Aber all das ist zu verschmerzen, wenn man den wesentlich günstigeren Preis zu Grunde legt, denn der Pi Zero kostet gerade mal ein Fünftel eines Pi 3.

Am Anfang gab es noch große Probleme mit dem Timing. Vor allem Programme, die sich auf mehrere Dateien aufteilen wurden eher schlecht unterstützt. Inzwischen funktioniert das alles mit der aktuellen Softwareversion 1.24 schon wesentlich besser, auch der Piezo Speaker tut es jetzt.

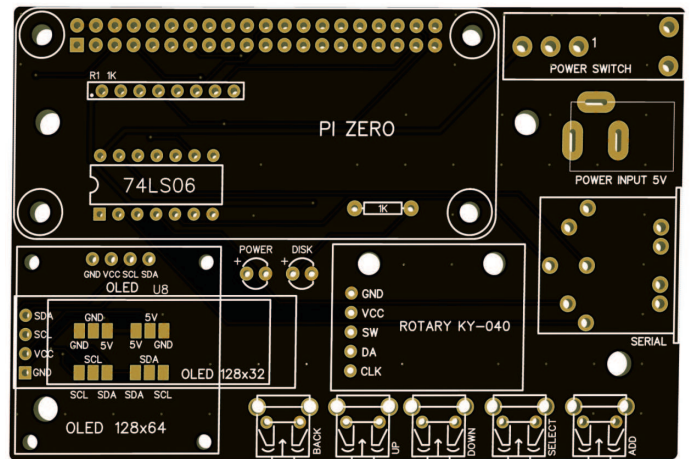
Das Grundprinzip der Schaltung hat sich hingegen kaum verändert. Für die neueste Platine wurde lediglich der unzuverlässigere und immer wieder für Probleme sorgende Level-Shifter (das kleine Platinchen mit den vielen Transistoren) gegen einen recht simplen 7406 (Hex-Inverter-

Pi1541 Interface Wiring Diagram





Die fertig aufgebaute Platine



Das Layout der Platine

Buffer) getauscht. Das erzeugt noch etwas mehr Stabilität.

Optimiertes Layout

Die Platine wurde wieder einmal von Matthias Lorenz gezeichnet, jedoch nach Vorgaben des Autors. Diese waren vorher mit einem kleinen Team in seinem damaligen Forum Loop64 besprochen worden. Ziel dabei war es, ein einfaches Design zu schaffen, das auch individuell anpassbar ist. So gibt es sowohl die Möglichkeit, das Laufwerk mittels 5 Tastern zu steuern, oder aber auch mit nur 2 Tastern und einem Rotary-Device (Drehgeber). Dieser übernimmt die Funktion der Taster (Up, Down und Select) und erlaubt es so, bequem durch das Menü zu steuern. Ein Druck auf die Achse wählt eine Datei aus und mounted sie direkt. Diese Datei liegt meist im Format *.d64 vor und wird dann wie eine echte 1541-Diskette behandelt. Sie kann ganz normal mittels LOAD "\$",8 gefolgt von LIST eingelesen und dargestellt werden.

Verbesserungen

Zusätzlich ist es möglich, sowohl ein OLED-Display mit 128x32 Pixeln oder 128x64 Pixel anzuschließen und somit entweder 2 oder 4 Zeilen des Inhaltsverzeichnis-

ses darzustellen. Im Options.txt auf der SD-Karte wird definiert, welches Display vorhanden ist. Weiterhin wurde durch eine einfache Pinreihe die Möglichkeit geschaffen, ein Kabel direkt an die Platine anzulöten oder auch mittels Pin-Header zu stecken. Dies spart sowohl eine zusätzliche IEC-Buchse (DIN- 6H), als auch die lästige Suche nach einem IEC-Kabel. Ist das Kabel bereits fest mit der Platine verbunden, kann es der Nutzer auch nicht vergessen, transportiert er sein Gerät. Die Platine erhält ihren Strom ganz normal wieder über einen Hohlstecker, der am besten von einem USB-Netzteil oder -Port aus gespeist wird. Ist das Netzteil des C64 stark genug, so lässt sich das Laufwerk auch direkt von dort aus versorgen.

Gehäuse

Wer ein Pi Zero 1541 Laufwerk nachgebaut hat, braucht auch ein passendes Gehäuse. Hier hat der User "bernd-7" im Forum.Classic-Computing.DE einen Entwurf für ein Gehäuse aus dem 3D-Drucker vorgestellt und feilt gegenwärtig noch an Details. Wer einen passenden Drucker besitzt, kann also selbst ein Gehäuse herstellen.

Fazit

Alles in allem ist das Pi Zero 1541 Laufwerk ein sehr günstiges Gerät, das im Selbstbau mit etwa 30€ an Materialkosten zu Buche schlägt. Es kommt schon ziemlich nahe an vergleichbare Emulationen heran. Noch geht nicht alles damit, aber die Software wird stetig verbessert. Es lohnt sich also, die Schaltung nachzubauen und dem Pi Zero eine Aufgabe am Commodore 64 zu geben.

Links

Pi1541 Homepage
<https://cbm-pi1541.firebaseio.com>
 Forums-Thread zum Bau des Pi Zero 1541:
<https://forum.classic-computing.de/forum/index.php?thread/23234-neues-pi1541-zero-platinen-layout-von-matthias-ausm-f64-kompakt/>



Die erforderlichen Dateien zum Selbstbau finden sich auf der Heft-CD

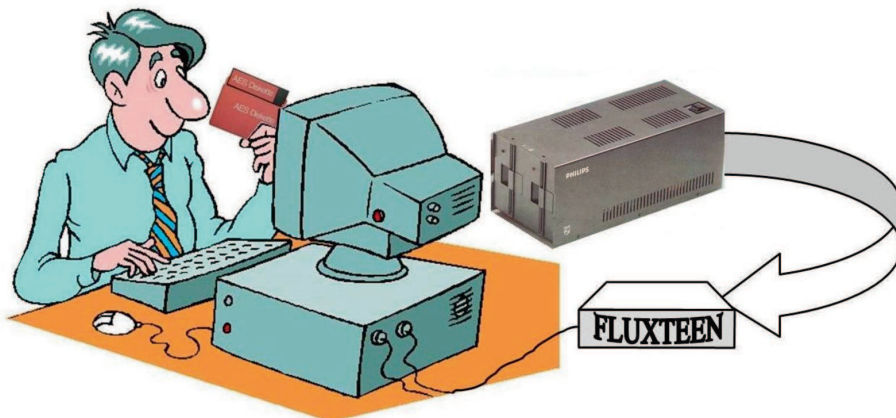


Ueber den Autor

Andreas Schabesberger bastelt gern an Retro-Rechnern herum, musste in seiner Jugend aber ohne C64 auskommen. Das holt er jetzt mit wachsender Begeisterung nach.

Kopieren hartsektorierte Disketten

Das Fluxcopy Projekt



Auf dem originalen IBM PC, aber auch auf vielen anderen klassischen Computern kommen softsektorierte 5 1/4 Zoll Disketten zum Einsatz. Um diese heutzutage auszu-lesen und die Inhalte zu archivieren, existieren Lösungen wie Kryoflux. Leider versagen diese, geht es um hartsektorierte Disketten. Dafür braucht es eine Speziallösung, bestehend aus einer Hardware am USB Port und passender Software.

Im Zuge seiner Arbeiten für die Konvertierung von CP/M und UCSD Disketten-Images hat der Autor eine Menge Know How gesammelt (siehe LOAD Ausgabe 6, Seite). Für das Erstellen und Zurückschreiben der Images wird dabei SAMdisk (Tool von Simon Owen) am PC verwendet. Das hat jedoch seine Grenzen: Am PC können nur Disketten gelesen oder geschrieben werden, für die der Diskettencontroller ausgelegt ist. Das ist meist ein μ PD765 oder ein ähnliches Produkt. Der verarbeitet aber nur soft-sektorierte Disketten im FM- oder MFM-Format. Andere Formate sind nicht lesbar, da der Controller dies ablehnt. Das gilt auch für hartsektorierte Disketten, die anstelle eines einzelnen Indexlochs beispielsweise 17 Indexlöcher besitzen und so die Lage der Sektoren je Spur fest vorgeben.

Projektziele

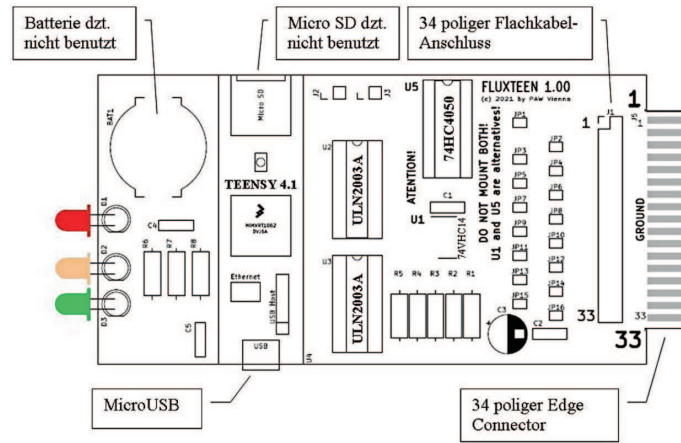
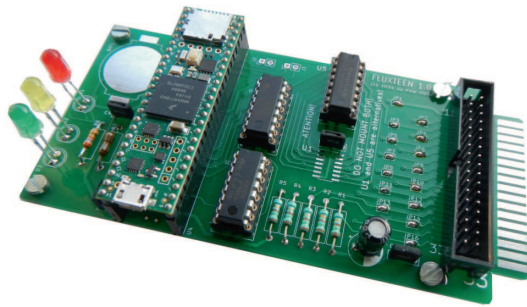
Getrieben durch die Erfahrungen mit den Disketten einer AES-Lanier Textverarbeitungsmaschine entstand daher ein Projekt, das zum Ziel hat, diese Lücke zu schließen. Projektziel war es, eine Lösung zum Einlesen von hartsektorierten Disketten via USB-Schnittstelle am Windows-PC zu entwickeln. Hierzu sollten das Programm "FLUXCOPY" und die Zusatzhardware "FLUXTEEN" neu entwickelt werden. Die Images der einzelnen Tracks sollen sich als FLX-Dateien (eigenes Format für FLUXCOPY) am PC speichern lassen. Ebenso sollte es möglich sein, diese Images später auf andere, unformatierte Disketten gleichen Formats – also mit gleicher Anzahl an Sektoren – trackweise zurückzuschreiben. Als Ziellaufwerk sollte dabei dasselbe Laufwerk wie beim Einlesen verwendbar sein, aber auch ein anderes Laufwerk. Insofern sollten Unterschiede bei den Drehzahlen

berücksichtigt werden. Als Schnittstelle zu den Diskettenlaufwerken wurde ein Shugart Bus (Drive Interface created by Shugart Associates) für ein 34-poliges Flachkabel vorgesehen. Für 8 Zoll Disketten ist dann ein 34-zu-50 Pin-Adapter erforderlich.

Überlegungen zu FLUXTEEN

Um ein Floppylaufwerk via USB steuern zu können, braucht es einen Prozessor. Die reinen Steuersignale wie Motor On, Step Impulse, Read Only und andere ließen sich ohne weiteres mit einem Arduino UNO bewerkstelligen. Dieser kann auch den Anschluss per USB meistern. Anders sieht es mit den Schreib- und Lesedaten aus. Die Fluxwechsel benötigen eine wesentlich höhere Datenrate als ein Arduino UNO mit seinem Takt von 16 MHz zu leisten in der Lage wäre. Der bereits erwähnte KryoFlux arbeitet mit einer Samplingrate von rund 24MHz, andere Systeme zumindest mit 16 MHz. Aus diesem und anderen Gründen entschied sich das Projekt für einen TEENSY 4.1. Die CPU des TEENSY arbeitet mit 600MHz und verfügt auch über ein superschnelles, 512 KByte großes RAM. Damit lässt sich eine Samplingrate von 25MHz leicht erreichen.

Da der TEENSY 4.1 jede Menge an Timern, Countern und anderem enthält, sind außer den Levelshiftern (TEENSY verträgt nur 3,3 Volt) keine ICs nötig. Er erfordert lediglich einige Widerstände, Leuchtdioden, ein paar Kondensatoren, sowie den 34-poligen Anschlussstecker für das Flachkabel zum Floppylaufwerk. Die Stromversorgung von FLUXTEEN erfolgt ausschließlich über den Micro-USB Anschluss (5Volt/max. 500mA). Der erste Testaufbau von FLUXTEEN erfolgte auf einem Steckboard. In der derzeitigen Print-Version der Platine ist zusätzlich zum 34-poligen Flachkabelanschluss noch ein 34-poliger Edge-Connector vorgesehen. Ein solcher Connector wird auch an 5,25 Zoll Laufwerken verwendet, zum Beispiel den externen Laufwerken des Philips P2500 Rechners. Die Anschlüsse sind wahlweise zu verwenden.



Die fertig aufgebaute FLUXTEEN Platine und ihr Layout

Die Firmware programmieren

Der TEENSY ist in C++ mittels der Teensyduino IDE programmierbar. Diese ist gratis im Netz verfügbar und auch relativ einfach zu bedienen. Auf Windows XP kann als letzte für dieses Betriebssystem erschienene Version das Release 1.8.5 installiert werden. Für Windows 10 gibt auch neuere Versionen, der Autor nutzt derzeit die Version 1.8.13. Abgesehen von kleinen Unterschieden im Handling sind beide Versionen identisch benutzbar. Um auch TEENSY-Modelle programmieren zu können, bedarf es noch der Zusatzsoftware Teensyduinoinstall.exe, die es ebenfalls gratis im Netz gibt. Sie wird auf die Arduino-IDE aufgespielt und ab diesem Zeitpunkt sind auch sämtliche TEENSY-Modelle in der IDE verfügbar, inklusive diverser Beispielprogramme.

Aufgrund der wesentlich schnelleren CPU mit 600MHz braucht es praktisch kei-

nen Assemblercode, die Programmierung kann vollständig in C++ erfolgen. Die Leistung entspricht zwischen 600 bis 1000 MIPS, weil ein oder zwei Befehle pro Takt ausgeführt werden können.

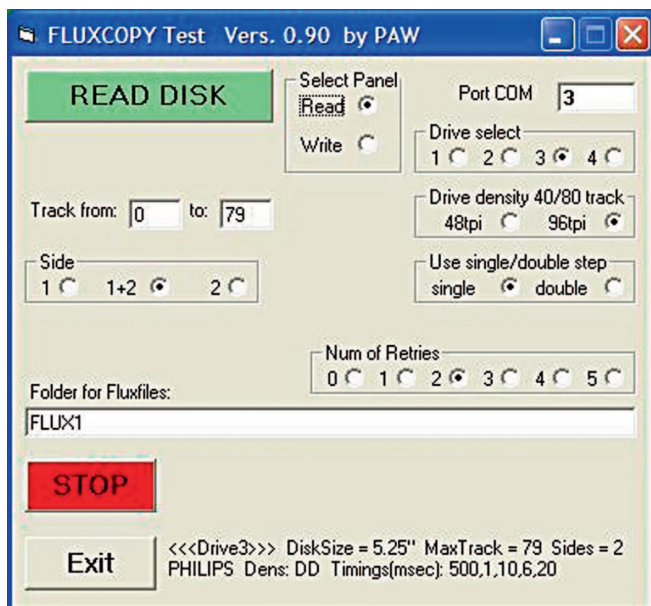
Ist die Firmware erst einmal erstellt, muss sie noch auf den TEENSY kommen. Dazu muss das TEENSY Ladeprogramm gestartet werden, es findet sich üblicherweise unter \hardware\tools\teensy.exe, wenn Teensyduino IDE installiert ist

Die PC-Software FLUXCOPY

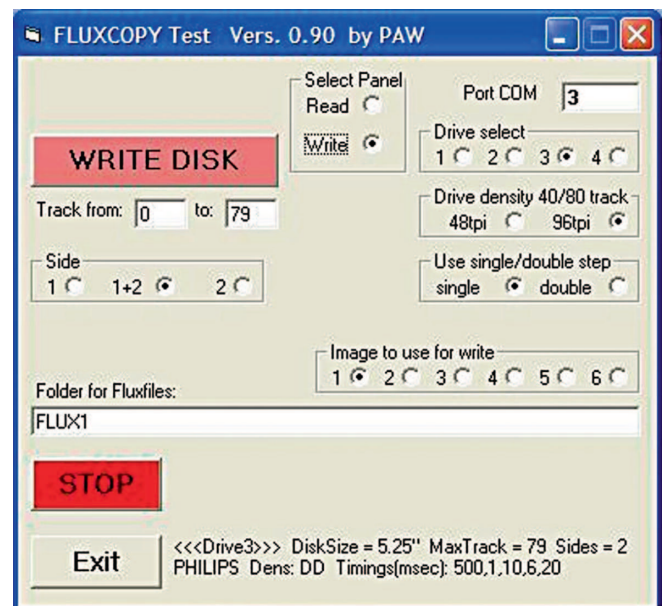
Um FLUXTEEN benutzen und steuern zu können, braucht es eine passende Software auf dem PC. Auch diese entstand im Projekt, und zwar als GUI Anwendung für Windows. Hier sind in einer Eingabemaske diverse Parameter einzustellen, bevor sich ein Image einer Diskette erstellen lässt. Ein Großteil davon sind auch über die Kommandozeile zu initialisieren. Im Feld "Port COM" muss die entsprechende

Schnittstelle angegeben werden. Es kann aus bis zu vier Drives ausgewählt werden. Je nach Laufwerkstyp (48tpi oder 96tpi) und Datenträgerbeschaffenheit muss eine Auswahl in den Feldern "Drive density" und "Use single/double step" getroffen werden. Wichtig ist die Angabe der Anzahl von Kopien (Num of Retries): Bei 0 wird jeder Track nur einmal eingelesen. Das ist natürlich am Schnellsten und braucht auch den geringsten Platz. Da aber die Richtigkeit der Daten während der Aufzeichnung nicht überprüft werden kann, ist es empfehlenswert, mehrere Durchgänge aufzuzeichnen (Vorgabewert ist 2). Damit das Ergebnis auch überprüft werden kann, gibt es das Tool FLUXDUMP. Damit lassen sich die Checksummen aller Sektoren für ein bestimmtes Format überprüfen.

Das Verzeichnis für die FLX-Dateien muss immer ein direktes Unterverzeichnis zum Verzeichnis von FLUXCOPY.exe sein. Das Image einer Diskette besteht aus je ei-



FLUXCOPY Eingabemaske für das Lesen von Disketten



...und die Ansicht für das Zurückschreiben.

```

03=== CKS-OK          Track:01  Sec:03  (0)
05 20 2D 20 5A 65 69 6C 65 3A 20 32 0D E1 ED 09 |. - Zeile: 2....|
E9 44 61 73 20 69 73 74 20 76 69 65 6C 20 54 65 |.Das ist viel Te|
78 74 20 2D 20 5A 65 69 6C 65 3A 20 33 0D E1 ED |xt - Zeile: 3...|
09 E9 44 61 73 20 69 73 74 20 76 69 65 6C 20 54 |..Das ist viel T|
65 78 74 20 2D 20 5A 65 69 6C 65 3A 20 34 0D E1 |ext - Zeile: 4..|
ED 09 E9 44 61 73 20 69 73 74 20 76 69 65 6C 20 |...Das ist viel |
54 65 78 74 20 2D 20 5A 65 69 6C 65 3A 20 35 0D |Text - Zeile: 5.|
E1 ED 09 E9 44 61 73 20 69 73 74 20 76 69 65 6C |...Das ist viel|
20 54 65 78 74 20 2D 20 5A 65 69 6C 65 3A 20 36 | Text - Zeile: 6|
0D E1 ED 09 E9 44 61 DE 00 00 00 00 00 00 00 00 |.....Da.....|
    
```

Beispiel für einen eingelesenen Sektor

ner FLX-Datei pro Spur und Seite, also üblicherweise aus 40, 80 oder 160 Dateien.

Das Anklicken von "Select Panel Write" innerhalb von FLUXCOPY zeigt ein leicht geändertes Formular, in dem sich auswählen lässt, welche der zuvor gespeicherten Kopien zurückgeschrieben werden sollen. Zum Redaktionsschluss waren bereits neuere Versionen in Arbeit. Diese werden ein ähnliches Layout haben und auch eine Möglichkeit zum Lesen und Schreiben von C64- Disketten und anderen anbieten.

Erfolgreiche Tests

Mit dem Testaufbau und lauffähigen Versionen von FLUXCOPY ging es anschließend daran, Disketten von einer AES-Lanier Textverarbeitungsmaschine auszulesen. Dies klappte mit dem angeschlossenen Laufwerk nach ein paar Anläufen

ohne Probleme. Spannend wurde es beim Zurückschreiben. FLUXCOPY meldete keine Fehler und die Textverarbeitungsmaschine akzeptierte die Diskette tatsächlich. Damit war das gesteckte Ziel erreicht, nämlich die hartsektorierten Disketten von AES-Lanier kopieren zu können.

Im Zuge der Analysen der AES Disketten fanden sich viele interessante Besonderheiten:

- _____ Im AES 7100 werden doppelseitige 80 Track Laufwerke verbaut, allerdings wird nur jede zweite Spur auf der ersten Seite auch tatsächlich genutzt. Die Disketten sind also nur einseitig mit 37 Tracks in doppeltem Abstand beschrieben.
- _____ Das logische Diskettenformat verwendet ein komprimiertes Inhaltsverzeichnis, bei dem 3 Zeichen in 2

Bytes untergebracht werden, also 5 1/3 Bits pro Zeichen. Die Verzeichniswerte berechnen sich dabei als Doppelbytwert = (1. Zeichen * 1600) + (2. Zeichen * 40) + (3. Zeichen). Wer sich für die Details dieses Formats interessiert, sei auf den Thread "XY - Software Rätsel ungelöst" im Forum.Classic-Computing.DE verwiesen (siehe Links).

Zukunftsaussichten

Da in diesem Projekt noch jede Menge Potential steckt, geht die Entwicklung natürlich weiter. Das Programm FLUXKRYO-CONV kann RAW-Dateien, die mit KryoFlux erstellt wurden, in FLX-Dateien für FLUXCOPY umwandeln. Die Fluxdaten müssen dafür je Track und Seite in einer eigenen Datei gespeichert sein. So konvertierte Images kann FLUXCOPY dann wieder auf Disketten schreiben.

Die am 22. August 2021 veröffentlichte Version 050 von FLUXDUMP bietet nun auch die Möglichkeit, viele andere Formate zu dumpen:

- _____ FM und MFM-DD-Disketten (Softsektor, µPD765-kompatibel)
- _____ MFM-HD-Disketten (Softsektor, µPD765-kompatibel)
- _____ WANG-FM-Disketten (Hartsektor)
- _____ C64-Disketten (GCR Softsektor)

Die Entstehung des Projekts

Natürlich gibt es diverse Tools wie KryoFlux und andere, mit denen auch andere Formate gelesen und geschrieben werden können. 2020 klagte mir Günther Pospischil sein Leid, dass er für seine AES-Lanier Textverarbeitungsmaschine keine neuen Disketten erstellen kann, da es kein Kopier- und Formatierungsprogramm dafür gibt. Es handelt sich dabei um 5.25 Zoll hartsektorierte Disketten mit 16 FM-formatierten Sektoren, es gibt also 17 Indexlöcher in der Diskette. Bei einem Versuch in meinem PC wurde die Diskette nicht einmal erkannt. Viele PC Laufwerke können mit so kurzen Intervallen beim Index nichts anfangen, sie interpretierten das als "keine Diskette im Laufwerk".

Günther konnte mit KryoFlux wohl Images aufzeichnen, aber beim Rückschreiben auf Diskette gab es Probleme. Laut KryoFlux Doku ist nur das Lesen möglich, nicht aber das Schreiben. Günther konnte KryoFlux mit einem Trick überlisten, indem er 16 Indexlöcher abdeckte und damit eine Softsektor-Disk vortäuschte. Allerdings gab es dann beim Einlesen auf dem Originalsystem (nachdem wieder alle Löcher freigemacht wurden) jede Menge Lese-

fehler, die wohl auf die Drehzahlschwankungen der Laufwerke zurückzuführen sind. Vermutlich verwendet dieses System keine PLL (phased locked loop) um die Schwankungen auszugleichen, sondern liest mit fixem Timing die Daten ein. Damit ist dieser Weg nicht gangbar. Dieses Problem wurde schon an anderer Stelle diskutiert:

"Ich habe ein AES 7100 samt Typenrad-Drucker und Beschreibung der Software (Textverarbeitung AES Office System 4.40.22). Angeblich soll es auch CP/M geben, das habe ich aber nicht. Die beiden Systemdisketten mussten ursprünglich teuer gekauft werden und es gibt dafür kein Kopierprogramm am AES selbst. Ich wollte sie auch schon länger irgendwie sichern bzw. kopieren. Mit KryoFlux scheint die Sicherung funktioniert zu haben, nur bringe ich das Image eben nicht mehr zurück auf Disketten (leere hartsektorierte Disks habe ich zur Verfügung). Wenn ich etwas Zeit habe, werde ich mit den KryoFlux Herstellern Kontakt aufnehmen und die Disk-Images übermitteln. Dazu muss ich aber erst Texte am AES schreiben und auf Disketten speichern (mit Textfiles ist die Formatanalyse viel einfacher, als mit den Systemdisketten). Vielleicht können sie das Format analysieren und finden einen Weg, es wieder auf Disketten zu schreiben."

(Zitat von Günther am 24. Sept. 2020). Damit war der Start eines neuen Projektes geboten. Gemeinsam mit Günther wollte ich ein Kopiersystem für die hartsektorierten AES-Disketten (16 Sektor FM) bauen. Günther und ich haben das Projekt 2020 gemeinsam begonnen, mit dem Ziel die AES-LANIER-Disketten kopieren zu können. Das ist 2021 gelungen. Im Projekt hat Günther die Logistik, also die Beschaf-

fung und Verteilung von FLUXTEEN-Platinen, die Organisation und die Tests auf dem Zielsystem übernommen. Mein Part ist das Design der Hardware, Programmierung der benötigten Tools, sowie grundlegende Tests. Da wir nun den Leistungsumfang erweitert haben, war es auch nötig, andere User mit einzubinden. Ich bedanke mich an dieser Stelle bei allen Mitwirkenden herzlich und hoffe auch in Zukunft auf tatkräftige Unterstützung.



Die AES Textverarbeitungsanlage

Für das Jahr 2022 sind weitere Entwicklungen und Verbesserungen geplant:

- Ausweitung der Diskettenformate auf 3.5", 5.25" und 8", DD und HD, sowie FM, MFM, MMFM und GCR, Hard- und Softsektor
- Erstellung von D80- und D82-Imagefiles für CBM Disketten
- Erweiterung von FLUXKRYO-CONV in Richtung FLX-Files zu KRYO-Files. Das würde eine Anbindung auch an HxC ermöglichen.
- Sobald die FLUXCOPY Versionen sich nicht mehr dauernd aufgrund der Erweiterungen ändern, wird es eine ausführliche Dokumentation dafür geben.

Weitere Informationen zum FLUXCOPY Projekt, sowie die Downloads der letzten Versionen finden sich im Forum (siehe Links). Es lohnt sich also, regelmäßig vorbeizuschauen und die Entwicklung zu verfolgen.

Links

<https://forum.classic-computing.de/forum/index.php?thread/23170-fluxcopy-projekt-f%C3%BCr-hard-und-software-zum-kopieren-von-hard-sektorierten-disket/>
<https://forum.classic-computing.de/forum/index.php?thread/23989-xy-software-r%C3%A4tselungel%C3%B6st/&postID=295213#post295213>

TEENSY is a trademark of PJRC.COM LLC
KryoFlux is a registered trademark ® of KryoFlux GmbH.

Ueber die Autoren

Peter aus Wien war in den 1980iger Jahren bei Philips im Bereich der P2000 Rechner taetig und stellt seit Mitte 2019 sein Philips Know-How im Forum zur Verfuegung. Guenther Pospischil hat vor 35 Jahren als Schueler erste EDV Erfahrungen mit dem Commodore 128 erworben. Seit 15 Jahren sammelt und repariert er Retro-Computer, aktueller Schwerpunkt sind die Philips P2000 Systeme sowie Triumph Adler Alphantronic Rechner.

Fundstücke aus der
LOAD Fernschreibmaschine

+++ Nachricht von Bender +++ 31.12.2007
+++ Ein Glück, das der Mensch lesen kann, wie wäre es mit einem Vereinsmagazin? Nichts monatliches, aber etwas was 4x im Jahr dem Rest der Welt zeigt, um was es hier geht. Berichte über Treffen, neue Techniken und Produkte, Interviews usw. Wer mag kann sich sowas mal am 19ten anschauen, das ist nichts professionelles, aber mit Freude und Begeisterung für die Sache gemacht. +++ STOP+++

+++ Nachricht von Kangaroo MusiQue +++ 20.01.2012 +++ Das VzEkC-Magazin - Gäbe es für sowas Autoren der unterschiedlichen Systeme, damit man Inhalt bekommt? Partybericht von der einen oder anderen Party. Spieletests, Vorstellung einzelner Mitglieder und deren Sammlung oder besondere Aktivitäten, Terminkalender für Retro-Börsen, Parties (Partys) etc. +++ STOP +++

+++ Nachricht von Blader +++ 01.03.2012
+++ Ich will nicht zu viel verraten, aber ich kann sagen, dass die erste Ausgabe eine sehr interessante Auswahl an Themen beherbergen und überraschend umfangreich sein wird. Ideen sind mehr als genug vorhanden - auch für die zukünftigen Ausgaben. (..) In gerade mal einer Woche haben wir die Struktur des Magazins ausgearbeitet und es sind schon fast 3/4 des Magazins mit Inhalt gefüllt. +++STOP +++

+++ Nachricht von Blader +++ 06.03.2012
+++ Der Launch unseres neuen Magazins steht kurz bevor. Es wird den Namen "LOAD" tragen. "LOAD", das ist der Name, der bei jedem klassischen Computer ein gängiger Begriff ist, steht er doch für das Laden eines Programms oder eines Spiels. +++STOP+++

+++ Nachricht von Hexagon +++ 18. Juli 2013 +++ Hab 2 Paletten voll Magazinen hier. Jetzt geht's los. +++STOP+++

+++ Nachricht von yalsi +++ 18. April 2017 +++ Die 3.Ausgabe unseres Vereinsmagazins LOAD ist erschienen. Es handelt sich um eine Online-Ausgabe, die ihr auf der Homepage downloaden könnt. Vielen Dank von mir an die fleissigen Autoren. Euch allen viel Spaß beim Lesen. +++STOP+++

+++ Nachricht von yalsi +++ 2. Februar 2018 +++ In den letzten Monaten wurde fleissig an der LOAD#4 gearbeitet. Autoren aus dem Verein haben Artikel geschrieben, das Layout wurde entworfen, ein Titelthema festgelegt, über eine Druckausgabe nachgedacht usw. Zwar steht die Finanzierung noch nicht endgültig, aber es besteht durchaus Hoffnung, in diesem Jahr durchaus wieder eine Druckausgabe herausgeben zu können. +++STOP+++

Übersicht zu allen bisher erschienenen Ausgaben unter
<https://www.classic-computing.org/load-online/>

...und um Ulm herum

2021 fand die Classic Computing in Vöhringen statt, einem Städtchen südlich von Neu-Ulm. Unter Pandemiebedingungen war einiges anders als in den vergangenen Jahren, aber der Spaß war so groß wie immer.

Leider konnte im Jahr 2020 aufgrund der Pandemielage die Classic Computing nicht durchgeführt werden. Umso größer war die Spannung im Jahr 2021. Als der Termin für das große Jahrestreffen des Vereins zum Erhalt klassischer Computer e.V. festgesetzt wurde, war die Zahl von Neuinfektionen mit dem SARS/CoV2 Virus in Deutschland so hoch, dass die Durchführung mehr als fraglich erschien. Dennoch wurden die Planungen fortgesetzt, besonders von unserem engagierten Mitglied vor Ort.

Je näher der Termin rückte, desto gespannter waren alle Beteiligten. Im Sommer standen die Zeichen noch auf Entspannung, doch ab August schien sich die Lage wieder zu verschlechtern. Gleichzeitig veränderten sich die Vorgaben zur Infektionsprävention mehrfach. Um auf der sicheren Seite zu sein, entwickelte der Verein ausgehend von den Vorgaben der bayerischen Landesregierung ein umfangreiches Hygienekonzept. Seine Einhaltung wurde glücklicherweise durch die Vorarbeiten des Kulturzentrums „Wolfgang-Eychmüller-Haus“ in Vöhringen deutlich erleichtert.

Dann war es schließlich soweit – die Classic Computing konnte wie geplant vom 17. bis 19. September 2021 stattfinden. Über 50 Teilnehmer aus dem ganzen Bundesgebiet hatten sich angesagt und bauten ab Donnerstag ihre Exponate auf. Die großzügigen Räumlichkeiten machten es möglich, die Tische lockerer und offener als sonst aufzubauen und so genügend Sicherheitsabstand zwischen den Teilnehmern zu ermöglichen. Zum Schutz aller musste beim Flanieren durch die Räume eine Schutzmaske getragen werden. Besucher durften die Ausstellung besuchen, sofern sie sich am Eingang entsprechend registrieren ließen. Erstmals hatte der Verein hierfür einen professionellen Sicherheitsdienst beauftragt. Die Gäste schreckte das nicht ab – mit etwa 270 Besuchern am Wochenende war die CC 2021 eine erfolgreiche Veranstaltung.

Besondere Highlights der CC 2021 waren neben vielen Mikrocomputern und Homecomputern der 1970er und 1980er Jahre sicher der neu entwickelte AMIGA-Rechner Vampire 2, eine funktionstüchtige Atari Transputer Workstation und eine große Sammlung von Atari ST Computern. Vielen dürfte auch das durchdringende Rattern des Lochstreifenstanzers am Wang Computer in Erinnerung bleiben. (gb)



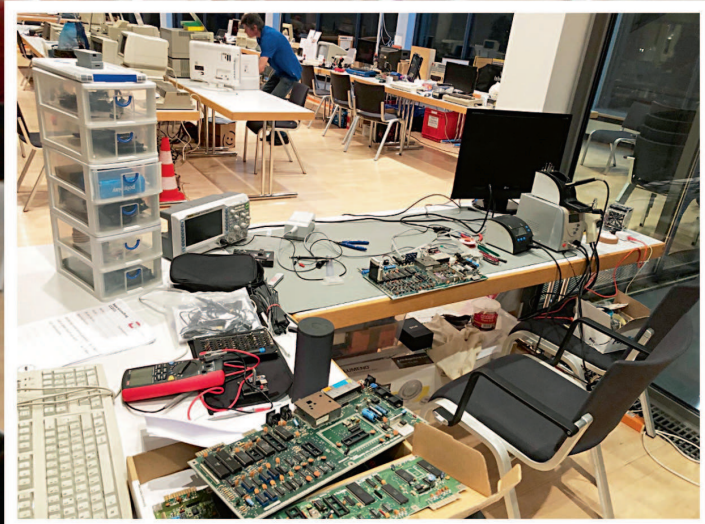


Bild oben:
Bild Mitte:
Bild unten:

CBM PET im Einsatz
Die Reparaturrecke hatte viel zu tun
Das Lokalfernsehen war ebenfalls zu Gast

Das bringt LOAD#9



Betriebssysteme

AmigaOS, TOS, MacOS und Windows kennt jeder – aber wir stellen Systeme wie BeOS, Coherent UNIX, SymbOS oder OS-9 vor und geben Tipps für die Installation auf verschiedenen Plattformen.

20 Jahre VzEK e.V.

Ende November 2003 fanden sich im Gasthof eines kleinen Kurorts im Landkreis Calw in Baden-Württemberg zwölf Retro-computer-Begeisterte zusammen, um ihr liebstes Hobby besser zu organisieren. Daraus ist heute ein Verein mit über 300 Mitgliedern geworden, in dessen Diskussionsforen sich über 5.000 Fans alter Rechner tummeln. Das 20-jährige Jubiläum bietet Gelegenheit für einen Rückblick und hält einige Überraschungen bereit.

Jubiläumsjahr 2023

Der Commodore 64 feiert 2023 seinen 40. Geburtstag, der PET wird 45 Jahre alt und WANG Computer feiern ihr 50. Jubiläum. Anlass genug, die Geburtstagskinder ausführlich zu würdigen.

RaSCSI

Mit einer Zusatzplatine wird der Raspberry Pi zu einer SCSI Festplatte mit vielen Goodies.

Die erste Maus

Wir erinnern an die erste Rollkugel-Maus, eine Entwicklung aus Deutschland.

...und außerdem:

MFM-Emulator, Apple-1 Mysterien, Neues aus der Retro-computing-Szene, Hintergründe, Portraits, Interviews, Berichte und Spieletipps.

LOAD#9
erscheint im Frühjahr 2023



Herausgeber:

Verein zum Erhalt klassischer Computer e.V.
c/o Stephan Kraus (1. Vorsitzender) (V.i.S.d.P.)
Am Schloßgarten 25/1, 74743 Seckach-Großeicholzheim

ISSN für die Printausgabe: 2194-3567
ISSN für die PDF-Ausgabe: 2194-3575

Redaktion

Leitung:
Georg Basse (gb)
Telefon: +49 5723 9865 700
redaktion@load-magazin.de
Lektoren: Sabine Cortnumme, Axel Rutzen, Michael Wegner



Autoren dieser Ausgabe

Georg Basse, Andreas Beermann, Sebastian Barthel, Bernd Dohr, Jochen Emmes, André Fachat, Fritz Hohl, Günther Pospischil, Peter Sieg, Herwig Solf, Thomas Winkler, Peter aus Wien, Michael Zacherle

Layout und Druck

Cover-Bild: User "globus" aus dem forum.classic-computing.de

Gestaltung:
Verein zum Erhalt klassischer Computer e.V.

Druck:
Wir-Machen-Druck.de, 1.Auflage 2022 (750 Exemplare) [20220324]

Wichtige Hinweise

Wir freuen uns über eingesandte Beiträge, behalten uns aber Veröffentlichungen, Kürzungen und Änderungen vor. Für unverlangt eingesandtes Bild- und Textmaterial können wir keine Haftung übernehmen. Namentlich gekennzeichnete Artikel geben nicht unbedingt die Meinung der Redaktion wieder. Die Beiträge der namentlich genannten Autoren und der Redaktion stehen nach Veröffentlichung im Heft unter einer Creative Commons-Lizenz (CC-BY-NC-SA) und dürfen für nichtkommerzielle Zwecke und unter Namensnennung des Autors verwendet und für abgeleitete Werke unter der gleichen Lizenz benutzt werden. Autoren können ihre Artikel bis zum Redaktionsschluss zurückziehen, wodurch alle Rechte an den Autor zurückfallen. Nach Redaktionsschluss ist dies nicht mehr möglich. Autoren akzeptieren mit ihrer Einsendung diese ehrenhaften Bedingungen.

Logos, Warenzeichen und Produktabbildungen werden redaktionell ohne Nennung des Eigentümers benutzt. Das Fehlen einer Kennzeichnung impliziert nicht die freie Verwendbarkeit dieser Elemente. Trotz sorgfältiger Prüfung ist es uns nicht gelungen, alle Rechteinhaber zweifelsfrei zu identifizieren und anzuschreiben. Bitte wenden Sie sich gegebenenfalls an die Redaktion.

Preis

Das Magazin LOAD wird in gedruckter und elektronischer Form grundsätzlich kostenlos abgegeben. Um einem Missbrauch vorzubeugen, kann die ausgebende Stelle für gedruckte Hefte eine Schutzgebühr in Höhe von 3,- EUR erheben.

Alte Computer verständlich gemacht

LOAD MAGAZIN

DAS VEREINSMAGAZIN

LOAD ist das Magazin des Vereins zum Erhalt klassischer Computer. LOAD ist eine Zeitschrift für Retro-Liebhaber, die sich für Homecomputer, Workstations oder klassische Spielekonsolen und deren Software interessieren. Egal ob Hardware, Spiele, Anwendungsprogramme, Artikel über längst vergessene Computer oder News und Berichte aus der Retro-Szene, all dies macht das Magazin interessant für jeden Leser, der sich mit den Themen Retro-Computing und Retro-Gaming beschäftigt. Das Heft wird vollständig von den Vereinsmitgliedern gestaltet – wir schreiben Artikel, machen das Layout, erstellen die Druckvorstufe und übernehmen die Verteilung des fertigen Hefts.



Vergriffen

LOAD#1

Unsere Erstausgabe. Viele Berichte über Homecomputer, die im Jahr 2012 ihr Jubiläum gefeiert haben.



Vergriffen

LOAD#2

Viele Details und Hintergrundberichte über Computer in der DDR, außerdem „Die Acorn Story“ und vieles andere mehr.



nur 3,-
Schutzgebühr

LOAD#4

Neue und klassische Hardware für Homecomputer, Selbstbauprojekte und Vorstellung klassischer Computer.



nur 3,-
Schutzgebühr

LOAD#5

Emulation auf alter Hardware wie sich mit Atari, Acorn, Amiga, CBM und Co. neue Welten eröffnen.



nur 3,-
Schutzgebühr

LOAD#6

Grafische Benutzeroberflächen erobern den Alltag: Macintosh, AMIGA, Atari – und ihre Gegenentwürfe wie Canon Cat oder Sinclair QL



nur 3,-
Schutzgebühr

LOAD#7

IBM PC Geschichte, Betriebssysteme, Modelle, Olivetti M24, Netzwerke, HP-Interface Bus, Sinclair Spectrum, TheVC20, Assembler, Spielen auf dem Würfel-Mac

Und LOAD #3 ?

Ausgabe #3 ist bisher nur online erschienen. Eine Druckausgabe ist für den Herbst 2022 geplant.

VzEkC e.V.
Verein zum Erhalt
klassischer Computer

Online
Jede LOAD erscheint 6 Monate nach der Druckausgabe als PDF zum kostenlosen Download.
<https://www.load-magazin.de>

Der VzEkC e.V. ist Deutschlands großer Retrocomputer-Verein

LOAD

Das Retrocomputer-Magazin

Verein zum Erhalt klassischer Computer e.V.
Stephan Kraus
(1. Vorsitzender)

Am Schloßgarten
25/1
74743 Seckach-
Großeicholzheim

Einzelhefte gibt es online vorbehaltlich der Verfügbarkeit gegen eine Spende von EUR 3,- zzgl. EUR 2,- für Porto und Verpackung unter <https://www.classic-computing.org/get-load5/>

Folgende Shops haben die LOAD ebenfalls im Angebot:

Poly.Play:
<https://www.polyplay.xyz/Magazine>

ACP&TCP
Andreas Magerl
<https://www.amigashop.org/index.php?cPath=44>

Außerdem ist die LOAD auf vielen Retrocomputing-Veranstaltungen erhältlich, zum Beispiel hier:

Lange Nacht der Computerspiele Leipzig

Alternatives Computer-meeting

Retro Computer Treff Niedersachsen

Classic Computing 2022

Vintage Computer Festival Berlin

Amiga Future Mai/Juni 2012 • € 6,50 • CD-Edition € 9,50 • www.amigafuture.de

**A M I G A
FUTUR**



A M I G A FUTURÉ

DAS FACHMAGAZIN RUND UM DEN AMIGA

Geboten werden die aktuellsten Nachrichten aus der Amiga-Szene!

Das Magazin erscheint komplett in Farbe mit einer optionalen Leser-CD.

Alle zwei Monate kann man die brandneue Ausgabe direkt bei uns im Online-Shop oder im Amiga-Fachhandel erhalten.

WWW.AMIGAFUTURE.DE