

Virtuelle Maschinen für Anwendungssoftware-Kompatibilität und zukunftsorientierte Lehre

Dieter Jungmann

Leiter Technologischer Vorlauf
Robotron Datenbank-Software GmbH
Stuttgarter Str. 29
01189 Dresden
dieter.jungmann@robotron.de

Abstract: Es wird die Entwicklung der Computer in der DDR vom ersten Kleinrechner bis hin zu Mainframes und Superminicomputern aus der Sicht der Kompatibilität ihrer Rechnerarchitekturen dargestellt. Sie war insbesondere bei den Kleinrechnern dadurch gekennzeichnet, dass jedes Nachfolgesystem nicht kompatibel zu seinem Vorgängersystem war. Am Rechenzentrum der Ingenieurhochschule Dresden wurde eine Vielzahl von Kompatibilitätslösungen auf der Grundlage der Implementierung virtueller Maschinen realisiert. Sie dienten zunächst der Gewährleistung einer experimentell gestützten Lehre für fortgeschrittene Rechnerarchitekturen und deren Sprachverarbeitungssysteme. Mit leichtem Zeitversatz wurden aber auch Simulatoren und Emulatoren entwickelt, die die uneingeschränkte und produktive Weiternutzung geschaffener Software auf nicht kompatiblen Nachfolgerechnern gewährleistet. Später lösten gezielte Forschungsarbeiten im Auftrag des Kombirates Robotron zu Rechnerarchitektur und Sprachverarbeitungssystemen die aus eigener Sicht betriebenen Arbeiten ab. Die fachliche Entwicklung des Autors war eng verbunden mit diesen Arbeiten von Anbeginn an.

O Vorbemerkungen

Meine fachliche Entwicklung stand in enger Verbindung zur Entwicklung von elektronischer Rechentechnik und Informatik in der DDR. Das 1963 begonnene 3-jährige Studium „Hochfrequenztechnik“ an der Ingenieurschule für Maschinenbau und Elektrotechnik wurde im 2. Studienjahr zu „Elektronischer Rechentechnik“ als Konsequenz aus dem „Ersten Datenverarbeitungsprogramm“ vom 3.7.1964 [Me64] umgestaltet. Fachrichtungsleiter Wilhelm Leupold spielte dabei eine Pionierrolle hinsichtlich der Profilierung des Ausbildungsprogramms, der Absicherung der rechentechnischen Basis für Praktika und der ersten Lehrveranstaltung „Elektronische Rechentechnik“. Die rechentechnische Ausstattung bestand aus einem Kabinett mit mechanischen Tischrechenmaschinen, einem programmgesteuerten Modellrechenautomaten auf Relaisbasis, einem Analogrechner des Institutes für Datenverarbeitung (IDV) und einem ersten digitalen Kleinrechner SER2a. Erste ingenieurmäßige Arbeiten führte ich in der Analog- und Hybridrechentechnik durch:

- Rekonstruktion des Analogrechners zum Modellregelkreis für die Simulation technischer Systeme

- Entwicklung eines Spezialrechners für die Simulation von Projektablaufen mittels CPM, den ich mit zwei weiteren Kommilitonen zu einem Wirtschaftspatent anmeldete.

Mein Fachrichtungsleiter überzeugte mich, an der Ingenieurschule als Laboringenieur für Elektronische Rechentechnik zu bleiben. Durch Wartungs- und Programmierarbeiten für den geschenkten D4a war mir die Gelegenheit gegeben, tiefer in die Materie einzudringen. Wenig später kam die industrielle Version C8205 hinzu. Die intensive Beschäftigung mit Methoden der Simulation führte sehr bald dazu, den C8205 als Basis für die CPM-Simulation, aber auch für die Simulation des wesentlich leistungsschwächeren SER2 einzusetzen.

1 Motivation

Basierend auf meinen ersten Simulationserfahrungen wurde dann am Rechenzentrum der Ingenieurhochschule Dresden eine lückenlose Folge von Simulatoren bzw. Emulatoren für Kleinrechner vom SER2 über den C 8205/C 8205Z zum KRS 4200/4201 und K1600 realisiert. Der KRS-Prozessor für die K 1630 wurde in Kooperation mit dem VEB ZFT Robotron in die Produktion überführt.

Die zur damaligen Zeit relativ hohe Operationsgeschwindigkeit des PRS 4000 bzw. KRS 4201 ermöglichte sogar die Architektur der EDVA R300 nahezu uneingeschränkt auf diesen Kleinrechnern zu simulieren und so mittels der R300 Hardware-Testprogramme zu verifizieren.

Zur Gewährleistung einer modernen Ausbildung wurden aber auch Assembler, Compiler und Simulatoren für Computer neuer Rechnergeneration auf Computern älterer Generationen implementiert und für Praktika in der Lehre eingesetzt. Im Vortrag werden die theoretischen Grundlagen, eine Systematik realisierter Anwendungen und einige Fallbeispiele dargestellt. Abschließend wird die Aktualität der damaligen Arbeiten am Beispiel des Produktes VMware für die Desktop- und Server-Virtualisierung gezeigt.

2 Theoretische Grundlagen

In historischen Abhandlungen ist es sinnvoll auf einheitliche Begriffe zu verzichten, weil zu bestimmten Zeitabschnitten und in verschiedenen Ländern unterschiedliche Begriffe für das gleiche Objekt verwendet wurden. Das trifft insbesondere für die DDR zu. Ähnlich wie heute noch in Frankreich, wurden international anerkannte Begriffe in englischer Sprache durch Begriffe der Landessprache ersetzt.

Einschränkend ist zu bemerken, dass die politische Doktrin der DDR dazu Wandlungen unterworfen war. In diesem Beitrag werden die Begriffe Rechner, Computer, Rechenmaschine, Rechananlage, EDVA für reale Objekte synonym verwendet. Der Begriff Maschine, eng. *machine*, wird dagegen für abstrakte Konstrukte verwendet.

Das Prinzip der Virtualität wurde in der elektronischen Rechentechnik für die Realisierung aller Komponenten eines Computers verwendet, um die physische Begrenztheit realer Komponenten zu überwinden:

- Hauptspeicher als virtueller Speicher
- Zentraleinheit (Prozessor) als virtuelle Maschine (Virtual Machine, VM)
- Periphere Geräte als Virtual Devices
- Computer als Computer Grid.

Zur Lösung des Problems der Lauffähigkeit von Programmen eines Computersystems auf einem anderen gibt es folgende Ansätze:

- Kompatibilität zwischen Generationen einer Computerfamilie (IBM 360, ...)
- Kompatibilität mittels expliziter virtueller Maschinen (Interpreter):

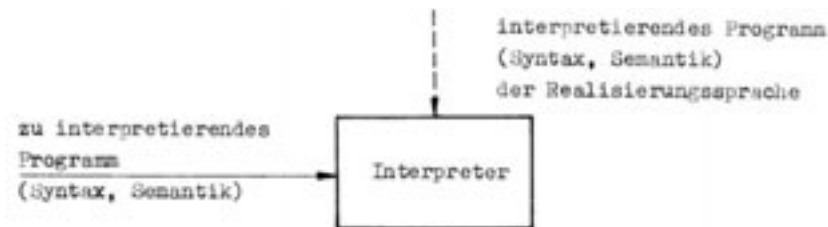


Abbildung 1: Blockschaltbild eines interpretierenden Sprachverarbeitungssystems

- Portabilität mittels impliziter virtueller Maschinen (Compiler):

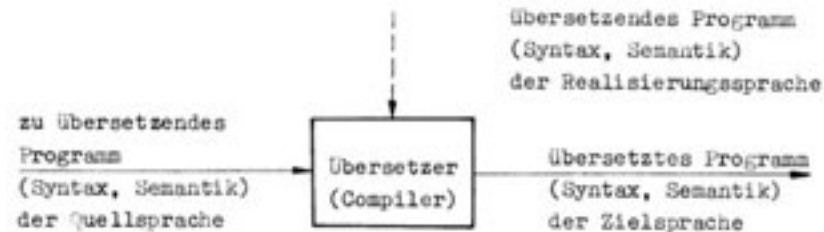


Abbildung 2: Blockschaltbild eines übersetzenden Sprachverarbeitungssystems

Der Klassiker auf dem Gebiet der Realisierung von Interpretern zum Zwecke Kompatibilität war V. Risak mit seinem Buch „Simulation von Digitalrechnern“ [Ri71].

Die darin enthaltenen Methoden mit Beispielanwendungen wurden in meiner Dissertation A mit dem Titel: „Simulation von Digitalrechnern – Technik und Technologie der programmtechnischen Implementierung von Digitalrechnern“ [Ju77] auf der Basis der DDR-Rechentechnik erweitert angewendet. Die nachfolgende Definition entspricht der experimentellen Basis dieser Arbeit und deckt auch die aus der Literatur bekannten Anwendungen.

In beiden Fällen ist Datenträgerkompatibilität eine notwendige Voraussetzung. Dieses Kriterium ist aus heutiger Sicht besonders kritisch zu bewerten, weil die meisten früheren Datenträger mit heutigen Computern nicht mehr lesbar sind.



Abbildung 3: Definition Simulation von Digitalrechnern

3 Anwendung virtueller Maschinen

3.1 Systematik der Anwendung

In der Zeit von 1970-2004 wurden zunächst im Rechenzentrum der Ingenieurhochschule Dresden, dann ab 1986 am Lehrstuhl Rechnerarchitektur des Informatikzentrums und ab 1993 am Lehrstuhl Entwurf von Rechnersystemen eine Vielzahl von F/E-Arbeiten zum Thema virtuelle Maschinen durchgeführt. Etwa 1975 waren es zunächst nur Überführungsleistungen an das Kombinat Robotron, die Ende der 70er Jahre zu Auftragsarbeiten geführt wurden.

Es wurden folgende Anwendungsgebiete verfolgt:

1. Praktikumssysteme für die **Informatiklehre**
2. Kompatibilität für Anwendungsprogramme auf nichtkompatiblen Nachfolgerechnern (**IT-Anwendung**)
3. Experimentalsysteme für die **Informatik**-, insbesondere aber Rechnerarchitekturforschung.

Eine Veröffentlichung aus dem Jahr 1982 [Ju82] zeigt eine Übersicht in Form einer Kompatibilitätsmatrix zum Stand der Kompatibilität zwischen den häufig genutzten Computern. In [Ju77] ist eine weit umfangreichere Darstellung enthalten, die weitere Computer beinhaltet, die ich z. B. während meines mehrfachen Studienaufenthaltes am LETI und am Informatik- und IT-College in Budapest kennenlernte. Abbildung 4 aus [Ju82] zeigt in jeder Zeile einen Wirtsrechner und in jeder Spalte einen Zielrechner. Die Kennzeichen Y haben folgende Bedeutung:

Y = A In Anwendung befindlich
 R In Realisierung befindlich

P Geplante Vorhaben
 S Sinnvolle Anwendung
 M Mögliche Anwendung
 (Y) Eingeschränkte Anwendbarkeit
 Y. Von IH Dresden realisiert

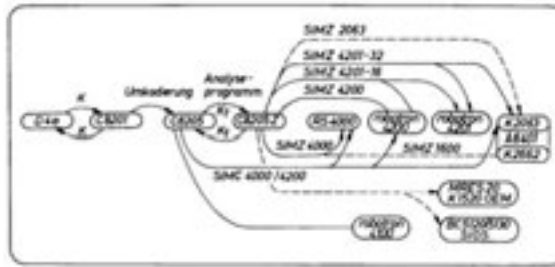
	SER 2	C 8205	C 8205 Z	d 1840	KSR 4100	KRS 4201	RS 4000	MRES 20	BC 5100	URS 5000	K 2063	SKR	R 300	ESER I	ESER II	BESM 6
SER 2																
C 8205	A.	A				A.										
C 8205 Z	A	A	A			A.										
d 1840	S															
KSR 4100		A														
KRS 4201	A.	A.	A.	S		A.	A.	A	S	(A)	R.	A.	A.	P	M	M
RS 4000	A.	A.	A	M		A.	A.	A	S	(A)	S	S	A.	(A)	M	M
MRES 20	M	M	S	S				A	S	A	S	S				
BC 5100	M	M	R.	(P)				A				(P)		(P)		
URS 5000						S		A			S	S				
K 2063	R.	R.	R.	S	S	R ₍₁₎					A.		R.	R.	S	S
K 2062			R					R.		R.	A.	A				
K 2063			R					R.			A.	A				
R 300		(A)	(A)			(A)	(A)						A.	(A)		
R 22						(A)		(A)				R	M	A		
R 40						(A)		(A)				R	A	A		
R 55 M						M		M.				R	M	A	A	S

Abbildung 4: Gesamtübersicht zum Stand der Kompatibilität im Jahr 1982

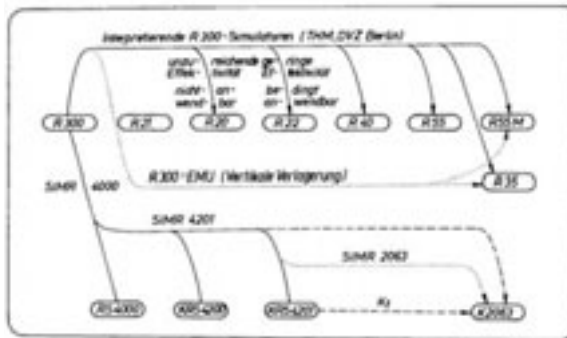
Diese Matrix veranschaulicht, dass die meisten Anwendungen an der IH Dresden entwickelt wurden (mit A. markiert). Es werden auch die verwendeten Wirtsrechner ersichtlich, das sind Zeilen mit Symbol Y. In den nachfolgenden Abbildungen werden die Kompatibilitätsbeziehungen zwischen Computern ersichtlich. Meistens sind es Computer einer Größenklasse. An den Bögen steht K für Kompatibilität, Kt für Teilkompatibilität oder eine Software-Produktbezeichnung.

Softwareprodukte sind Editoren, Cross-Assembler/Compiler, interpretierende Testsysteme, Konvertierungsprogramme, Interpreter (Simulatoren), Software-Support des KRS-Emulator-Prozessors für den K1630.

- **C 8205 (Z):** Das war der erste Rechner mit einem Prozessor, der Bitketten verarbeiten konnte und eine für die damalige Zeit akzeptable Operationsgeschwindigkeit hatte (etwa 10fach höher als SER 2). Ausführliche Informationen sind in [KJ04] enthalten.



- **R 300:** Das war der erste Computer mit einer leistungsfähigen Peripherie. Er wurde deshalb insbesondere als Host für Cross-Assembler/Compiler der Kleinrechner mit eingeschränkter Peripherie eingesetzt.



- **KRS 4201/RS 4000:** Diese Computer hatten ein günstiges Preis/Leistungs-Verhältnis, verfügten über einer relativ hohe Operationsgeschwindigkeit (≤ 1 MIPS) und einen Hauptspeicher von 64 KByte. Mit diesen Eigenschaften konnte die Rechnerarchitektur aller Vorgänger programmtechnisch implementiert werden.
- **K1630:** Die Hauptspeicherkapazität war zwar höher als die des KRS 4201, doch die Operationsgeschwindigkeit war wesentlich geringer durch die Verwendung von N-MOS-Prozessorschaltkreisen an Stelle von TTL-Schaltkreisen. Die Kompatibilität zum Vorgänger KRS 4201 war programmtechnisch nicht realisierbar.

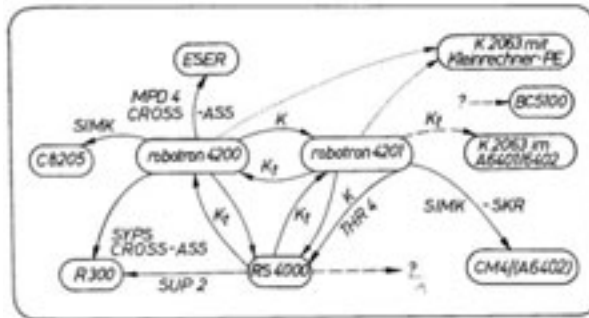


Abbildung 7: KRS 4201/RS 400-K1630-Kompatibilitätsbeziehungen

- **ESER I/II:** Bis auf das schlechtere Preis/Leistungsverhältnis, vergleichbar mit RS 4000.

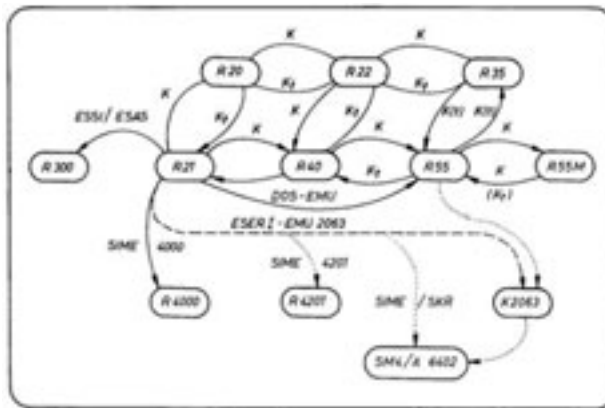


Abbildung 8: EDVA (Mainframe)-Kompatibilitätsbeziehungen

3.2 Informatiklehre

Das Rechenzentrum der IH Dresden wurde im Jahr 1969 primär als Ausbildungsrechenzentrum gegründet. Zunächst war ich Leiter des Einsatzstabes und später stellv. Leiter des RZ und Leiter des Softwarebereichs. Die Mitarbeiter waren bis auf Ausnahmen Absolventen: Am Anfang von der Ingenieurschule für Maschinenbau und Elektrotechnik, später von der Ingenieurhochschule. Ein großer Vorteil war, dass bereits 1966 mit der Ausbildung von Programmieringenieuren begonnen wurde. Maßgeblich beteiligt waren Wilhelm Leupold und Helmut Jenke.

Bedingt durch limitierte Bilanzen und fehlende finanzielle Mittel bekamen wir 1970 zunächst nur ausgesonderte Computer des Kombinales Robotron: Ein R 300 mit Seriennummer 1, der später an das Technische Museum Dresden übergeben wurde und ein Prozessrechner PR 2100, der später in das Museum folgte.

Das war in einer Zeit, in der in der Wirtschaft bereits IBM 360-kompatible Mainframes und Prozessrechner DDP 516 bzw. die Eigenentwicklung RS 4000 im Einsatz waren.

Mit viel Kreativität und Engagement wurde in einem erheblichen Umfang maschinenorientierte Software (damals Systemunterlagen genannt) für neue Computergenerationen entwickelt. Unser Entwicklungspotential konnte durch den Einsatz von Diplomanden und Kooperationspartner der Hochschulrechenzentren des Hochschulwesens bedeutend erweitert werden.

Folgende größere Softwarepakete wurden entwickelt und in Praktika eingesetzt:

- **PL/1-Compiler mit dem IBM/OS-Sprachumfang für den R 300:** Der Compiler wurde in der R300-MACRO-Assembler-Sprache als 8-Pass-Compiler im Wesentlichen von Peter Hoffmann und Walter Schiller entwickelt. Die Übersetzung des Compilers dauerte 24 Stunden. Es kam oft vor, dass die Maschine abstürzte und neu begonnen werden musste. So wurden Änderungen weitestgehend im Maschinencode ausgeführt, um Rechen- und Entwicklungszeit zu sparen. Als schließlich das erste minimale PL/1-Programm generiert war, stellten wir fest, dass für die Übersetzung des generierten MACRO-Assembler-Programms eine Stunde Rechenzeit erforderlich war. Peter Hoffmann entwickelte in nur 6 Monaten einen neuen MACRO-Assembler, der nur 1/3 der Anzahl der Befehle aufwies und 3mal schneller war. Damit war die Durchführung von PL/1-Praktika auf dem R 300 gesichert. Bei einem Hauptspeicher heutiger PCs von 1 GByte kann man sich nicht mehr vorstellen, solch komplexe Software mit einem Hauptspeicher von 40.000 Zeichen zu 6 Bit implementieren zu können.
- **ESER-Assembler/Simulator für den R300:** Ein R300-Assembler-Praktikum entsprach nicht mehr den Praxisanforderungen. Basierend auf umfangreichen Erfahrungen in der Realisierung von Sprachverarbeitungssystemen konnte in nur einem Jahr der volle ESER-Assembler-Sprachumfang implementiert werden. Die geringe Operationsgeschwindigkeit des ESER-Simulators war für den Testbetrieb völlig ausreichend. Eine besondere Herausforderung bestand darin, einen binär organisierten Prozessor auf einem dezimal organisierten Prozessor zu simulieren.
- **Crossassembler Klein- und Prozess- und Mikrorechner:** In diesem Fall wurden die Vorteile des R 3000 mit Magnetbandtechnik für die Quellcodeverwaltung und leistungsfähigem Paralleldrucker für das Drucken der Assemblerlisten genutzt. Ein schneller Lochbandstanzer stellte den Maschinencode für den Test auf Computern wie KRS 4201, Prozessrechner PR 2100 und Mikrorechner K1520 bereit. Mittels PDP 11-Simulator konnten mit dem KRS 4201 bereits SKR- bzw. K1600-Programme getestet werden.
- **BASIC-Timesharing-System für den R4000:** Mit dieser Entwicklung von Peter Hoffmann konnte erstmalig der Stapelverarbeitungsbetrieb eines geschlossenen Rechenzentrums verlassen werden. Die Studenten konnten interaktiv ihre Programme entwickeln und testen.

3.3. IT-Anwendung

Die Entwicklung der Kleinrechentechnik in den früher selbständigen Betrieben Karl-Marx-Stadt und Zella-Mehlis sowie später im Kombinat Robotron war dadurch gekennzeichnet, dass die Architektur jedes Nachfolgerechners nicht kompatibel zum Vorgänger war. Die Ursache dafür war zunächst von technischer Natur, weil bis zur Verfügbarkeit von TTL- und Halbleiterschaltkreisen ein Top-down-Entwurf ausgehend von einer Rechnerarchitektur hin zu einer Rechner-Implementierung und -Realisierung nicht möglich war. Das trifft zu für die Serien SER2 und C8205. So wird die Architektur des C8205 (Z) eindeutig durch die Struktur des Trommelspeichers bestimmt. Unter Verwendung eines Kern- oder Halbleiterspeichers wäre man niemals auf eine Wortlänge von 33 Bit gekommen.

Danach beim RS 4000 und damit auch beim KRS 4201 war die Orientierung an dem Industrie-Standardrechner DDP 516 bestimmend. Wegen der geringen Abweichung in der Befehlssatz-Architektur (*Instruction Set Processor, ISP*) und vollständig abweichenden E/A-Architektur war die Verwendung des Original-Betriebssystems verhindert. Eine umfangreiche Eigenentwicklung von Betriebssystemen und Systemsoftware war die Konsequenz.

Diese Vorgehensweise wurde von Kombinatzen wie Zeiss Jena nicht akzeptiert. Sie entwickelten und produzierten einen sogenannten Kleinststeuerrechner KSR 4100, der 100 %-kompatibel zu dem auf der Welt als anerkannter Standard geltenden PDP 8 war. Die entwickelte Anwendungssoftware war auf Originalrechnern lauffähig. Der KSR 4100 war aus vielen Gründen in das sogenannte NSW nicht exportierbar. Später musste Robotron diesen Rechner auch produzieren.

Die Beibehaltung der KRS/PRS-Architektur war aus vielen Gründen für die Familie K1600 nicht fortsetzbar. Wichtigster Grund war, dass im RGW auf die PDP 11-Architektur orientiert wurde. Diese Architektur ist unvereinbar mit der Vorgängerarchitektur. Wichtigste Anforderung war, dass das Originalbetriebssystem und damit auch die Anwendungssoftware lauffähig sind. Abweichungen gab es aber im Bussystem und in den metrisch ausgerichteten Steckverbindern.

Inzwischen wurde von führenden Kombinatzen mit dem Kombinat Mikroelektronik an der Spitze durchgesetzt, dass genau wie bei Schaltkreisen Steckerkompatibilität zu gewährleisten ist. Auch die bisher geringe Zuverlässigkeit von Computersystemen wurde nicht länger akzeptiert. So wurde auf höchster Ebene durchgesetzt, dass abweichend von der TGL das Zoll-System für den K1600-Nachfolgerechner K1840 zu Grunde gelegt wurde.

Auf höchster politischer Ebene musste entschieden werden, dass künftig Gold für Steckverbinder eingesetzt werden durfte. Ein Gesetz zur Rückgewinnung des eingesetzten Goldes wurde durch die Volkskammer beschlossen. Durch die strenge Orientierung am Vorbild VAX 11/780 mit einem PDP 11-Kompatibilitätsmodus war auch die Programmkompatibilität vom K1830 zum K1600 gesichert.

Nachfolgend werden Lösungen zu offenen Kompatibilitätsproblemen zwischen den Kleinrechnergenerationen dargestellt. Eine allgemeine Übersicht wurde dazu bereits im Abschnitt 3.1 vermittelt.

SER 2-Serie: Die Anwendungsprogramme waren durch die SER 2-Simulatoren für C 8205 und KRS 4201, insbesondere aber beim KRS 4201 mit sehr hohem Performancegewinn weiter nutzbar.

C 8205 (Z): In so manchem Unternehmen war die Anwendungsentwicklung für C 8205 Z noch nicht abgeschlossen, als die Produktion der Serie eingestellt wurde. Die Simulatoren SIMC für C 8205 und SIMZ für C8205 Z sicherten eine uneingeschränkte Kompatibilität für alle Anwendungsprogramme. Die Gewährleistung der Abwesenheit von Fehlern wurde zunächst durch Anwendung aller Hardwaretestprogramme in Angriff genommen. In der mindestens 800fachen praktischen Anwendung in der DDR und im SW traten noch 3 weitere sehr spezielle Fehler auf, die nach gründlicher Analyse behoben wurden. Danach war Fehlerfreiheit gesichert. Die Effektivität der Anwendung konnte bedeutend gesteigert werden: Mindestens 7fache Steigerung der Operationsgeschwindigkeit und wesentlich vereinfachte Bedienung durch die Simulation der Lochbandtechnik auf Magnetband (virtuelle Gerätetechnik).

R 300: Das Problem der weiteren Nutzung von R300-Anwendungsprogrammen stand erstmals auch für unser Rechenzentrum. Der riesengroße Rechnerraum des R 300 wurde für die geplante ESER-Anlage EC 1022 aus Minsk gebraucht. Es war klar, dass für eine Simulation der R 300-Architektur mindestens die Operationsgeschwindigkeit der EDVA R 40 aus der Robotron-Produktion erforderlich ist, der uns aber nicht zur Verfügung stand. Zu einem relativ frühen Zeitpunkt wurden uns Anlagen vom Typ RS 4000 durch Robotron geschenkt. Mit unseren Erfahrungen aus anderen Simulationsverfahren kamen wir zu der Einschätzung, dass der RS 4000 mit ESER-Peripherie als R 300-Simulator bestens geeignet ist. Die Entwicklung wurde unverzüglich begonnen und durch Lösung vieler Detailprobleme zu einem guten Ende geführt. Aus der Praxis kam die Forderung, diesen Simulator auch für einen ausgebauten KRS 4201 verfügbar zu machen. Beide Versionen wurden an Robotron zum Zwecke des Vertriebs übergeleitet. Später ergab es sich, dass sowohl das Rechenzentrum der Universität Magdeburg als auch das LfA Berlin einen R 300-Simulator für den R 40 entwickelten.

KRS 4200/4201: Die Situation beim Übergang zum System K1600 war vergleichbar: viele Entwicklungsvorhaben waren noch nicht abgeschlossen. Bei der Planung des Nachfolgesystems hatte sich Robotron darauf verlassen, dass die KRS-Assembler-Programme in K1600-Assemblerprogramme konvertiert werden können, und dass inzwischen auch Anwendungen in höheren Programmiersprachen entwickelt wurden, die durch einen äquivalenten Compiler nur neu zu übersetzen sind (Portabilität).

Beide Annahmen stellten sich sehr bald als Trugschluss heraus. Ich hatte vom Chefentwickler den Auftrag erhalten, zu diesen Annahmen eine detaillierte Studie zu verfassen. Nach kurzer Zeit stand fest, dass den Anwendern eine Alternative geboten werden musste: Einsatz eines Emulationsprozessors auf der Basis eines schnellen Bit-Slice-Microprocessor-Systems analog Intel 3000, hergestellt in der Sowjetunion.

Der Auftrag zur Entwicklung eines Funktionsmusters ließ nicht lange auf sich warten. Parallel zum Architekturentwurf wurden Simulatoren für die Ebenen Befehlssystem- und Mikroprogrammebene entwickelt, um Programme entwickeln zu können, noch bevor der Hardwareprozessor verfügbar war. Die Gesamtarchitektur entstand durch eine optimierende Verteilung auf die verschiedenen Ebenen: Elementare Funktionen mit größter Häufigkeit der Verwendung in die Hardware, Funktionen mit mittlerer Häufigkeit der Verwendung in die Firmware (Mikroprogramme) und Funktionen mit geringer Häufigkeit der Verwendung in die Software. Das Befehlssystem bestand aus den Originalbefehlen und selbst definierten Befehlen zur Optimierung der Performance der in Software implementierten Funktionen des Originalsystems. Früheren Erfahrungen folgend, wurden die Hardware-Testprogramme des KRS 4201 verwendet. Dazu musste die Architektur des KRS 4201 einschließlich Interrupt- und E/A-System originalgetreu nachgebildet werden. Das Ergebnis konnte sich sehen lassen: Die Performance war besser als die des K1630-Prozessors, und es war sogar das Betriebssystem für kommerzielle und wissenschaftlich-technische Anwendungen lauffähig [KS82]. Auch die früher für den KRS entwickelten Simulatoren konnten weiter genutzt werden. Der Prozessor wurde in einer größeren Stückzahl produziert.

3.4 Informatikforschung

Eigene experimentelle Arbeiten, unterstützt durch eine Vielzahl von Diplomarbeiten, wie z. B. die Simulation der ESER-Architektur auf dem RS 4000 sowie die Verfügbarkeit der weltweiten Rechnerarchitekturliteratur an der TU Dresden und bei Robotron, wurden zur Gestaltung von Rechner-, insbesondere Prozessorarchitekturen durchgeführt. Die Ergebnisse sind in [Ju84] zusammengefasst. Das finale Ergebnis waren eine Universalarchitektur UNIA und ein Virtual Instruction Set Prozessor (VISP).

Durch meine Tätigkeit bei Robotron auf dem Gebiet der Entwurfssoftware für Schaltkreise und komplette Rechnersysteme wurde an mein damaliges Team der Auftrag vergeben, ein universelles Simulationssystem für den Rechnersystementwurf zu entwickeln. Uwe Crenze erstellte im Rahmen seiner Dissertation dafür das Konzept und implementierte auch den größten Teil des ISPS-Simulationssystems. Verwendet wurde die Spezifikation der Hardwarebeschreibungssprache ISPS der CMU.

Ende der 80er Jahre war das Problem der Auswahl eines RISC-Prozessors zu lösen [SJTM94]. Mit RISC-Prozessoren waren höhere Taktfrequenzen und eine größere Anzahl von Befehlen pro Takt zu erzielen. Durch ergänzende Marktbetrachtungen waren die Prozessoren MIPS und SPARC in die engere Wahl gefallen.

In einem weiteren Forschungsauftrag war ein Simulator zu entwickeln, der Metriken für die Auswahl eines der beiden Alternativen lieferte. Die Ergebnisse wurden in der Dissertation von Wasja Suworow beschrieben. Eine spezielle Frage war, ob anstelle dieser Universalprozessoren der Ein-Chip-Computer „Transputer“ für massiv-parallele Systeme als nächster Chip in die Entwicklungspläne von Robotron und Mikroelektronik einzuordnen ist.

Aus Ergebnissen einer parallelen Forschungsgruppe des Informatikzentrums und internationalen Tagungsberichten war bekannt, dass noch keine praxiswirksamen Ergebnisse einer universellen Parallelisierbarkeit von Software vorlagen. Teilweise wurden diese Auseinandersetzungen ohne Berücksichtigung einer soliden fachlichen Basis auf politischer Ebene ausgefochten.

In der Diplomarbeit von Wolfgang Schwarz wurde untersucht, ob man durch Simulation eines Intel x86-Prozessors problemfrei schneller Viren erkennen kann. Aus Kapazitätsgründen konnte dieser Ansatz nicht weiter verfolgt werden.

Es war inzwischen klar, dass der Performance-Vorteil der RISC-Prozessoren mit dem Nachteil einer geringeren Codedichte erkaufte werden muss. Bei universellen Systemen ist dies bei dem schnell wachsenden Integrationsniveau von Speicherschaltkreisen kein Problem. Bei Embedded Systems und Ein-Chip-Computern sieht es aber anders aus. Im Rahmen eines Förderprojektes entstand dazu die Dissertation von Gerd Markwart.

Vor meinem Ausscheiden aus der Technischen Universität Dresden hatte ich grundlegende Ideen zu einem universellen Parallelprozessor entwickelt, der rekonfigurierbar ist. Der Compiler einer höheren Programmiersprache entscheidet in Abhängigkeit von dem analysierten Quellcode, in welchem Modus der Parallelprozessor betrieben wird. Patentstudien zeigten, dass Anteile dieser Ideen bereits patentiert wurden, aber Alleinstellungsmerkmale noch offen blieben.

4 Virtuelle Maschinen aus heutiger Sicht

Analysiert man heute Anwendungen virtueller Maschinen kommt man zu folgenden Ergebnissen:

- Unternehmen, die RISC-Prozessoren in ihren PCs und Workstation einsetzen, simulieren die CISC-Architektur Intel x86.
- IBM nutzt inzwischen seinen sehr leistungsfähigen PowerPC-Chip für die Emulation der Mainframearchitektur.
- AMD hat in Dresden ein Kompetenzzentrum gegründet, dass mit Mitteln der Simulation und Compiler-technik die Prozessorarchitektur optimiert. Aus der Sicht der Methodik entspricht die Vorgehensweise genau der unserer früheren Arbeiten.
- Die Technologie Virtueller Server hat durch die hohe Prozessorleistung eine enorme Bedeutung erlangt. Auf einer realen Maschine können mehrere virtuelle Maschinen mit völlig eigenständigen Softwarekonfigurationen laufen.

5 Schlussbemerkungen

Mit diesem Beitrag soll vermittelt werden, dass die begrenzte Verfügbarkeit moderner Computertechnik und Software für Lehre und Forschung in der DDR eine Herausforderung war, international fortgeschrittene Rechnerarchitekturen und Programmiersysteme auf veralteten Klein- und Prozessrechnern sowie Datenverarbeitungsanlagen für eine fortgeschrittene Lehre zu implementieren.

Dadurch war es im begrenzten Umfang möglich, den oft beklagten zeitlichen Abstand von 8-10 Jahren zu verkürzen.

Auf der Grundlage des dabei erworbenen Knowhows wurde das Problem der Praxis gelöst, die Investitionen in umfangreiche Anwendungssoftware beim Übergang zu nichtkompatiblen Nachfolgerechnern zu schützen. Die hohe Zahl von ca. 1.000 Lizenzabschlüssen für die SER2- und C8205 (Z)-Simulatoren für KRS 4200/4201 waren der Beweis dafür, dass die These: „Kompatibilitätslösungen zu neuen Rechnergenerationen verhindern die Ausnutzung neuer technischer Möglichkeiten“ nicht haltbar war. Genau das Gegenteil war der Fall: Erst durch diese Möglichkeit war der Zeitdruck der Einsatzvorbereitung genommen und der Entwurf sowie die Implementierung von Anwendungssoftware mit neuen funktionalen und nichtfunktionalen Eigenschaften möglich. Unter den Bedingungen einer freien Marktwirtschaft wäre die Ausgründung einer Firma die Konsequenz gewesen. So kamen einige weitere Kompatibilitätslösungen nicht über das Konzeptstadium hinaus. Produkte zur Server-Virtualisierung wären die logische Fortsetzung gewesen.

Eine gesicherte Finanzierung von F/E-Arbeiten durch das Ministerium für Hoch- und Fachschulwesen und die Absicherung der materiell-technischen Basis durch die Kooperation mit dem Kombinat Robotron war die Basis für eine langfristige Profilierung von Forschungsarbeiten zur Rechnerarchitektur und Sprachverarbeitungssystemen, die bis in die ersten Jahre nach der politischen Wende hinein reichten.

Literaturverzeichnis

- [Ju77] Jungmann, D.: Simulation von Digitalrechnern – Technik und Technologie der programmtechnischen Realisierung von Rechnerarchitekturen, Dissertation A, Ingenieurhochschule Dresden, 1977.
- [Ju83a] Jungmann, D.: Zur Programm- und Datenkompatibilität zwischen Rechnern, rechentechnik datenverarbeitung 10/1982, S. 5-9.
- [Ju84] Jungmann, D.: Tendenzen zur Gestaltung der Befehlssysteme von Rechnerarchitekturen, INFO84, Vorträge Rechnerarchitektur, 1984, S.23-32.
- [KJ04] Krause, Ch.; Jacobs, D.: Von der Schreibmaschine zu Mikrorechnersystemen, LNI – Informatik in der DDR, Symposium in Chemnitz, 2004, S. 116-139.
- [KS82] Kofer, C.; Schöne, S.: Entwicklung eines robotron-4200/4201-Emulators für K1600, rechentechnik datenverarbeitung 10/1982, S. 10-13.
- [Me64] Merkel, G.: Computerentwicklungen in der DDR – Rahmenbedingungen und Ergebnisse; LNI – Informatik in der DDR, Symposium in Chemnitz, 2004, S. 40-54.
- [SJTM94] Suworow, W., Jungmann, D., Tavangarian, D., Monjau, D.: Klassifikation von RISC-Prozessoren, Wissenschaftliche Beiträge zur Informatik 8(19994) H. 2, Technische Universität Dresden, Fakultät Informatik, 1994