

## Protected Mode

Die AirLand Battle 1991 hat es einmal mehr gezeigt: Unter den postmodernen Strategien des Scheins ist keine so wirksam wie die Simulation, daß es Software überhaupt gibt. Bis zum Gefechtsfeldbeweis des Gegenteils, als Computer unzweideutig zu erkennen gaben, Hardware zur Zerstörung irakischer Hardware (oder Eisenwaren, wie Hardware im Alltagsenglisch noch heißt) zu sein, verbreiten Werbefrospekte und Medienkonferenzen das Märchen von einer Softwareentwicklung, die schon immer sanfter und benutzerfreundlicher, spiritueller und intelligenter geworden wäre, bis sie eines unfernen Tages den Deutschen Idealismus effektiv heraufführen, also Mensch werden würde.

Weshalb Software, dieses Milliarden-Dollar-Geschäft mit einem der billigsten Elemente dieser Erde, nichts unversucht läßt, um besagte Menschen an die entsprechende Hardware gar nicht erst heranzulassen. Man kann mit WORD 5.0 auf einem No Name AT 386 und (wie es so schön heißt) unter Microsoft DOS 3.3 über eben diese drei Wesenheiten ganze Aufsätze schreiben, ohne die Strategie des Scheins auch nur zu ahnen. Denn man schreibt – das »Unter« sagt es schon – als Subjekt oder Untertan der Microsoft Corporation.

Diese Froschperspektive herrschte nicht immer. In der guten alten Zeit, als Mikroprozessorpins noch groß genug für schlichte LötKolben waren, konnten auch Literaturwissenschaftler mit dem Intel-Prozessor 8086 anstellen, was sie wollten. Durch Nichtunterscheidung zwischen RAM und ROM, Mißbrauch der beiden Stackregister als Allzweckregister, Vermeidung jeglicher Interruptvektoren, Zweckentfremdung des Wait-Eingangs undsoweiter waren selbst Standardchips, die für eine einzige Ganzzahlmultiplikation damals noch einhundertdreiunddreißig Takte brauchten, auf die Verarbeitungsgeschwindigkeit primitiver Signalprozessoren zu heben. Weil die Von-Neumann-Architektur keinen Unterschied zwischen Befehlen und Daten kennt, konnte es der Silizium-Chip an Dummheit

mit seinem Bastler und Benutzer aufnehmen. Denn dieser Benutzer, um ein Programm zum Laufen zu bringen, mußte erst einmal alles vergessen, was aus Schulzeiten noch als mathematische Eleganz oder geschlossene Lösung in seinem Kopf spukte. Er vergaß sogar seine zehn Finger und übersetzte alle Dezimalzahlen, die im Programm mitspielen sollten, in eintönige Binärzahlenkolonnen. Daraufhin vergaß er den Imperativ als solchen und wälzte Datenblätter, um auch die (selbstredend schon englisch formulierten) Befehle IN, OUT undsoweiter in ihren Opcode zu übersetzen. Eine Tätigkeit, die nur Alan Mathison Turing, als er seine Universale Diskrete Maschine von 1936 einen Weltkrieg später endlich zur technischen Verfügung hatte, allen Gedächtnisentlastungen und Hochsprachprogrammen vorgezogen haben soll.<sup>1</sup> Aber nachdem diese Austreibung von Geist und Sprache einmal vollbracht war, tat es die Maschine ihrem Benutzer an Dummheit gleich: Sie lief.

Sicher, diese sogenannte Maschinensprache lief millionenmal schneller als der Bleistift, mit dem der Benutzer ihre Nullen und Einsen aus Intels Datenblättern zusammengestückelt hatte. Sicher, sie nahm in den Flipflops, deren endlos wiederholte Muster den Siliziumchip überziehen, millionenmal weniger Raum als auf dem Papier ein. Aber damit waren die Unterschiede zwischen Computer und Papiermaschine, wie Turing den Menschen umgetauft hatte,<sup>2</sup> auch schon erschöpfend aufgezählt.

Diese guten alten Zeiten sind unwiderruflich vergangen. Unter Stichworten wie Benutzeroberfläche, Anwenderfreundlichkeit oder auch Datenschutz hat die Industrie den Menschen mittlerweile dazu verdammt, Mensch zu bleiben. Mögliche Mutationen dieses Menschen zur Papiermaschine sind mit vielfacher Tücke versperrt. Erstens gehen Microsofts Benutzerdatenblätter dazu über, Assemblerkür-

1 Vgl. Andrew Hodges, 1983, *Alan Turing: the enigma*. New York, S. 399.

2 Vgl. Alan M. Turing, 1937, *Über berechenbare Zahlen. Mit einer Anwendung auf das Entscheidungsproblem*. In: Turing, 1987, S. 20 f. und S. 40–43.

zel als maximale Zumutbarkeit oder Maschinennäherung zu unterstellen und das heißt überhaupt keinen Opcode mehr zu veröffentlichen.<sup>3</sup> Zweitens »versprechen uns« die einschlägigen Fachzeitschriften, »vom Programmieren in Maschinensprache bestenfalls, nach kurzer Zeit wahnsinnig zu werden«.<sup>4</sup> Drittens schließlich halten es dieselben Zeitschriften auch schon für sträflich, »eine Prozedur zur Berechnung des Sinus ausgerechnet in Assembler zu schreiben«.<sup>5</sup>

Auf die Gefahr hin, schon längst wahnsinnig geworden zu sein, kann man aus alledem nur folgern, daß die Software offenbar im selben Maß an Benutzerfreundlichkeit gewonnen hat, wie sie das kryptologische Ideal der Einwegfunktionen approximiert.<sup>6</sup> Je höher und komfortabler die Hochsprachen, desto unüberbrückbarer ihr Abstand zu einer Hardware, die nach wie vor alle Arbeit tut. Ein Trend, der wahrscheinlich weder durch den technischen Fortschritt noch durch Formalitäten einer Typentheorie ausreichend erklärt werden kann, sondern wie alle Kryptologie strategische Funktionen hat. Während es auf der einen Seite, in Kenntnis von Codes oder Algorithmen, prinzipiell machbar bleibt, Anwendersoftware oder Kryptogramme zu schreiben, wird es auf der anderen und benutzerfreundlich kaschierten Seite nachgerade unmöglich, vom Fertigprodukt auf seine Produktionsbedingungen zurückzuschließen oder diese Bedingungen gar zu verändern. Die

3 Vgl. etwa Microsoft Corporation, 1987, *Macro Assembler 5.1, Reference*, o. O., S. 115: »This section provides an alphabetic reference to instructions of the 8087, 80287, and 80387 coprocessors. The format is the same as for the processor instructions except that encodings are not provided.«

4 TOOL Praxis, 1989, *Assembler-Programmierung auf dem PC. Ausgabe 1*. Würzburg, S. 9.

5 TOOL Praxis, 1989, S. 39.

6 Über Einwegfunktionen in Mathematik und Kryptologie vgl. Patrick Horster, 1982/1985, *Kryptologie: eine Anwendung der Zahlentheorie und Komplexitätstheorie*. Mannheim – Wien – Zürich, S. 23 – 27.

Benutzer fallen einer mathematischen Tücke zum Opfer, die schon Hartley als einstigen Chef der Bell Labs in seine Altersdepression getrieben haben soll: dem Tatbestand, daß so vielen Operationen die Operanden nicht mehr anzusehen sind.<sup>7</sup> Die Summe versteckt die Summanden, das Produkt die Faktoren undsoweiter.

Für Software kommt diese mathematische Tücke wie gerufen. In einer Epoche, die die Phantome des Schöpfers oder Autors längst verabschiedet hat, aus guten finanziellen Gründen aber am Copyright als historischem Effekt solcher Geister mit Leidenschaft festhält, wird die Tücke zur Geldquelle. Die Untertanen von Microsoft sind jedenfalls nicht vom Himmel gefallen, sondern wie alle ihre medienhistorischen Vorläufer, die Bücherleser, Kinobesucher und TV-Zuschauer, erst einmal produziert worden. Das Problem ist nur, wie die Unterwerfung, um ihren weltweiten Siegeszug anzutreten, vor den Subjekten verborgen werden kann.

Die Antwort folgt, in ihrem wissenspolitischen Teil, einem bewährten Erfolgsrezept neuzeitlicher Demokratien, während ihr technischer Teil die Hardware von Mikroprozessoren selber verändert hat. Was die Wissenspolitik angeht, können vielleicht nur Siemens-Ingenieure die Dinge so schlichtweg beim Namen nennen, wie das im *80186-Handbuch* Klaus-Dieter Thies getan hat:

Moderne 16-Bit-Mikrocomputer übernehmen heute in zunehmendem Maße Aufgaben, die im typischen Anwendungsbereich klassischer Minicomputer liegen.

So ist es in Mehrbenutzersystemen erforderlich, daß Programme und Daten der einzelnen Anwender voneinander getrennt sind, ebenso wie auch das Betriebssystem vor Benutzerprogrammen geschützt sein muß. Um jedem einzelnen Benutzer die Möglichkeit zu geben, seine Software unabhängig von

7 Persönliche Mitteilung Hartleys an Friedrich-Wilhelm Hagemeyer/Berlin.

der Anzahl der anderen Benutzer zu implementieren und um ihm auch den Eindruck zu geben, daß der Rechner für ihn allein da ist, ist es notwendig, die CPU durch Multitasking auf die einzelnen Programme aufzuteilen, was dem Benutzer allerdings nur dann verborgen bleiben kann, wenn die CPU sehr leistungsstark ist.<sup>8</sup>

Nach dieser Siemens-Lesart, die aber auch bei der IBM Deutschland umgeht, hat Intel die Arbeitsfrequenzen des 80286 und des 80386 also nicht deshalb auf Höhen zwischen 12 und 33 Megahertz getrieben, weil sie erst so den Ansprüchen professioneller Benutzer oder gar den Pentagon-Spezifikationen für Electronic Warfare entsprachen,<sup>9</sup> sondern um zivile Anwender in eine undurchschaubare Simulation zu verwickeln. Multitasking soll wie der Igel im Märchen den Benutzern vorspiegeln, daß nur ein einziger Igel oder Prozeß läuft, vor allem aber, daß dieser Lauf oder Prozeß auch nur einem einzigen Hasen oder Benutzer zugute kommt. Das ist dieselbe Melodie, nach der Romane und Gedichte seit der Goethezeit ihren Lesern und vorab Leserinnen versprechen, nur für diese einzige unersetzliche Adresse da zu sein; dieselbe Melodie auch, nach der neuzeitliche Politik sich die Bevölkerungen als reines Gegenteil, nämlich als Individuen unterwirft.

Im Unterschied zu traditionellen Simulationen, die alle an der Macht oder Ohnmacht von Alltagssprachen eine absolute Grenze hatten, verfügt aber die elektronische Simulation, derzufolge jeder Mikroprozessor nur für einen einzigen Benutzer da sein soll, neuerdings auch über Hardware. Seit dem 80286 kennen Intels Prozessoren einen Protected Mode, der (in den Worten jenes Siemens-Ingenieurs) das Betriebssystem vor den Anwendern schützt und damit deren Illudierung überhaupt erst ermöglicht. Was im selbst-

8 Klaus-Dieter Thies, 1986, *Das 80186-Handbuch*. Düsseldorf – Berkeley – Paris, S. 319.

9 Die Einzelheiten des Pentagon-Pflichtenhefts siehe bei D. Curtis Schleher, 1986, *Introduction to Electronic Warfare*. Norwood/MA.

redend verschwiegenen Konkurrenzsystem, dem 68000 von Motorola, als bloße Umschaltmöglichkeit zwischen Supervisor-Stack und User-Stack begonnen hatte,<sup>10</sup> gelangt als Trennung zwischen Real Mode und Protected Mode zu systemweiter Durchsetzung: Unterschiedliche Befehlsätze, unterschiedliche Adressierungsmöglichkeiten, unterschiedliche Registersätze, ja sogar unterschiedliche Befehlsausführungszeiten trennen fortan die Spreu vom Weizen, die Anwender vom Systemdesign. In genau dem Silizium also, auf das die Propheten einer mikroprozessierten Zukunftsdemokratie ihre ganze Hoffnung gesetzt haben, kehrt die elementare Dichotomie moderner Medientechniken wieder. Einen deutschen Zivilrundfunk etwa durfte es ab dem Tag geben, als die Reichspost der Reichswehr glaubhaft versprechen konnte, daß die plombierten und um jede Sendemöglichkeit gebrachten Anwenderradios von 1923 den militärisch-industriellen Funkverkehr schon deshalb nie würden stören können, weil eine automatisierte Verschlüsselungsmaschine, die dann erst Turings Protocomputer im Zweiten Weltkrieg wieder unschädlich machten, soeben erfunden war.

Die Innovation von Intels Protected Mode besteht nur darin, diese Logik aus dem militärisch-industriellen Bereich in den der Informatik selber übertragen zu haben. Die Unterscheidung zwischen den beiden Betriebszuständen ist nicht bloß quantitativ, wie das etwa von den unterschiedlichen Arbeitstemperaturbereichen der kommerziellen, der industriellen und der militärischen Siliziumchips (in dieser bezeichnenden Stufung) gilt; die CPU selber arbeitet vielmehr mit Vorrechten und Verboten, Privilegien und Handicaps, über die sie fortlaufend, aber natürlich auch nur im Protected Mode Buch führt. Daß solche Kontrollen, die ja selber Zeit verbrauchen, dem allgemeinen Ziel der Datendurchsatzsteigerung nicht gerade förderlich sind, liegt auf der Hand. Im Protected Mode braucht der-

10 Vgl. Gerry Kane, 1981, *68000 Microprocessor Book*. Berkeley, S. 8f.

selbe Interrupt bis zu achtmal mehr Takte als im Real Mode. Aber offenbar darf eine Hochtechnologie erst dann an (wie Intel sie nennt) »nicht vertrauenswürdige« Programme und Endverbraucher weitergegeben werden, wenn auch und gerade die Signalverarbeitung, diese militärisch-industrielle Dimension von Computern,<sup>11</sup> von bürokratischer Datenverarbeitung abgebremst worden ist. Es sind zwar keine schriftlichen Verbotstafeln mehr, die ein Machtgefälle garantieren, sondern das Binärzahlensystem als solches codiert, was Befehle und was Daten sind, was dem System erlaubt und was umgekehrt Anwenderprogrammen verboten ist. John von Neumanns klassische Architektur, die ja zwischen Befehlen und Daten keinerlei Unterschied machte und in einer Epoche, als alle existierenden Computer noch Staatsgeheimnisse waren, wohl auch nicht zu machen brauchte, verschwindet unter vier durchnummerierten Privilegebenen. Mit allem ironischen Recht schrieb die einzig unbestechliche unter Deutschlands Computeranwendungszeitschriften: »Auch wenn nun ausgiebig von Privilegien, höher privilegierten Codesegmenten, Privilegverletzungen und dergleichen mehr die Rede ist, Sie lesen kein politisches Manifest eines früheren SED-Funktionärs, sondern die Erläuterung des Sicherheitskonzepts des 80386!«<sup>12</sup>

Politische Manifeste, wie der Name schon sagt, spielten im Herrschaftsbereich der Alltagssprache; deshalb sind die von ihnen eingeklagten Privilegien, spätestens seit November 1989, ja auch gegenstandslos. Die Privilege Levels von Intels sogenanntem Flaggschiff dagegen, diese ins innerste Binärzahlensystem verlagerte Cocom-Liste, dürften mehr als alle Fernsehberieselung Ostmitteleuropas dazu beigetragen haben, bloß politisch begründete Privilegien zu liquidieren. Eine schmale Schrift Carl Schmitts, das *Gespräch über die Macht und den Zugang zum Machthaber*, gipfelte

11 Vgl. Michael Löwe, 1985, *VHSIC: Ultraschnelle Schaltkreise frisch vom Band ins Pentagon*. In: *Militarisierte Informatik*. Hrsg. Joachim Bickenbach, Reinhard Keil-Slawik, Michael Löwe, Rudolf Wilhelm, Münster, S. 64.

12 Harald Albrecht, 1990b, *MSDOS in a box. Teil 1*. c't, S. 258.

einst in der These, daß sich Macht auf ihre Zugangsbedingungen reduzieren läßt: die Antichambre, das Büro oder neuerdings der Vorraum aus Schreibmaschine, Telefon und Sekretärin.<sup>13</sup> Mit solchen Instanzen und über solche Instanzen waren in der Tat noch Gespräche zu führen, während technisch implementierte Privilegebenen ihre Macht gerade aus stummer Wirksamkeit beziehen. Der 80386-Benutzer, um in einer Art posthistorischer Metaphysik endlich auf seine Speicherreserven jenseits von DOS zugreifen zu können, installiert eine der angebotenen benutzerfreundlichen Utilities, lädt den Debugger mit einem Eigenbauprogramm, das gestern noch problemlos lief, und muß feststellen, daß die Neuinstallation nicht nur wie versprochen Speicher verwaltet, sondern zugleich, aber ohne jede Vorwarnung, sämtliche privilegierten Befehle gesperrt hat.<sup>14</sup> Wie Mick Jagger es schon so unzweideutig formulierte, bekommt der Benutzer anstelle dessen, was er wünscht, immer nur, was er (und zwar nach Maßgabe des Industriestandards) braucht.

Aus alledem folgt für die Analyse von Machtsystemen, diese große von Foucault hinterlassene Aufgabe, ein Doppeltes. Erstens sollte versucht werden, Macht nicht mehr wie üblich als eine Funktion der sogenannten Gesellschaft zu denken, sondern eher umgekehrt die Soziologie von den Chiparchitekturen her aufzubauen. Zunächst einmal liegt es nahe, die Privilegebenen eines Mikroprozessors als Wahrheit genau derjenigen Bürokratien zu analysieren, die seinen Entwurf in Auftrag gegeben und seinen Masseneinsatz veranlaßt haben.<sup>15</sup> Nicht umsonst fiel die Trennung

13 Vgl. Carl Schmitt, 1954, *Gespräch über die Macht und den Zugang zum Machthaber*. Pfullingen.

14 So geschehen unter QEMM 386 von Quarterdeck beim Versuch, innerhalb von CODEVIEW die Befehle LGDT und SGDT auszuführen.

15 Als Beleg aus berufener Quelle, nämlich der Intel Corporation, vgl. Sharad Gandhi, *Die Intel Architektur & RISC*. In: *Pentium-Prozessor. Die nächste Generation der Intel Architektur*. Hrsg. Johann Wiesböck, Bernhard Wopperer, Gerold Wurthmann,

zwischen Supervisor Level und User Level bei Motorola, Protected Mode und Real Mode bei Intel in die Jahre, als auch US-Amerika an den Aufbau eines wasserdichten Zweiklassensystems ging. (Man kennt den Embedded Controller in jedem besseren Hoteltürschloß New Yorks.) Nicht umsonst sind beim 80386 gerade die Input- und Output-Befehle durch höchste Privilegstufe geschützt: In einem Imperium, dessen Bevölkerung den Rest der Welt nur durch die Mattscheibe von Fernsehnachrichten zu sehen bekommt, bleibt schon der Gedanke an Außenpolitik ein Regierungsprivileg. Das ist wahrscheinlich auch der Grund, warum die neuesten Spielarten der Systemtheorie den Befund, daß Informationssysteme über Input und Output verfügen, auf höchstem theoretischem Niveau einfach in Abrede stellen. Schließlich und endlich wäre das alles aber auch ein guter Grund für Informatikerinnen und Informatiker anderer Länder, irgendwo zwischen Japan und Europa also, der ins Silizium versenkten US-Bürokratie andere

Haar 1993, S. 80: »Der Hauptvorteil einer schnellen CPU, den die RISC-Prozessoren früher hatten, wurde so mit der Zeit zu einer eher sekundären Eigenschaft. Lassen Sie mich ein Analogon anführen. Der CPU-Kern verhält sich wie ein Unternehmer, der eine neue, innovative Firma gründet. Wenn die Firma klein ist, gedeiht sie mit den Ideen und Einfällen des Unternehmers. Mit zunehmender Größe der Firma wird ihr Erfolg mehr und mehr durch periphere Funktionen wie Vertrieb, Marketing, Öffentlichkeitsarbeit, Finanzen, Controlling usw. bestimmt, Funktionen, die meistens mit den Produkten der Firma überhaupt nicht in Zusammenhang stehen. Große Unternehmen konkurrieren aufgrund ihrer peripheren Funktionen miteinander. In Zukunft wird der Leistungsanteil des [mathematischen] Kerns weiter abnehmen.« Eine seltsam zweideutige Erfolgsbilanz, die auch ganz andere Schlüsse und den Gedanken an noch neuere, innovative Firmen erlauben würde. Wenn auf der Chipfläche des Pentium-Prozessors nurmehr 8 Prozent Rechenleistung 92 Prozent Firmen-Selbstabbildung gegenüberstehen (Ghandi, 1993, S. 79), kommt Hochtechnologie dem Verhältnis zwischen Gehirn und Körpergewicht bei Dinosauriern schon gefährlich nahe.

mögliche Bürokratien entgegenzusetzen. Ob es bessere wären, steht dahin, weil sie ja jedenfalls auch Bürokratien sein müßten; aber eine Konkurrenz zwischen unterschiedlichen Systemen und unterschiedlichen Bürokratien würde schon als solche die Untertanen von MSDOS aufatmen lassen.

Solange freilich die IBM-Kompatibilität ihre Triumphe feiert, ist weniger Soziologie als vielmehr Strategie gefragt. Die Macht hat mit ihrem Umzug aus Vorzimmern und Alltagssprachen in den Mikrometerbereich auch die Verfahren und Angriffsflächen verändert. Das schroffe Nein der Auskunftsverweigerung ist einem Binärcode nicht mehr gegeben, einfach weil die ganze Typenhierarchie selbstähnlicher Programmebenen – von der höchsten Programmiersprache bis zum elementaren Maschinencode<sup>16</sup> – vollkommen platt einer Materie aufruht. Im Silizium selber kann es, frei nach Lacan, keinen Anderen des Anderen geben,<sup>17</sup> also auch keine Protektion der Protektion. Noch die versteckten Segmentdeskriptoren, die über alle Zugangsrechte aller Programme eines Systems Buch führen, müssen zugänglich sein, um wirken zu können. Selbst daß die CPU diese Deskriptoren bei Privilegverletzungen allen möglichen und expliziten Befehlen zum Trotz auf einen

16 Vgl. Arne Schäpers, 1989, *Turbo Pascal 5.0*. Bonn – Reading/MA. – Menlo Park/CA. – New York – Don Mills – Wokingham – Amsterdam – Sydney – Singapore – Tokyo – Madrid – San Juan, S. 1: »Der Adreßraum des Prozessors ist durch die Architektur des Systems in RAM- und ROM-Bereiche gegliedert, wobei der RAM vom Betriebssystem in Speicherblöcke zur Aufnahme von Programmen unterteilt wird. Ein Programm besteht aus einzelnen Segmenten, von denen einige Prozeduren und Funktionen enthalten, die ihrerseits weitere Routinen lokal definieren können.« Fast fühlt man sich an ein Gemälde von Escher (oder auch zeitgeistiger: an Apfelmännchen) erinnert.«

17 Vgl. Jacques Lacan, 1973 – 1980, *Schriften*. Hrsg. Norbert Haas, Olten – Freiburg/Br., Bd. II, S. 194.

Nullpointer setzt,<sup>18</sup> hinterläßt lesbare Spuren. Auf der Maschinenebene geraten Schutzmechanismen also in die Verlegenheit, über kein absolut geschütztes Versteck zu verfügen. Im selben Maß, wie Mikroprozessoren denn doch für Anwender brauchbar bleiben und das heißt mit ihnen kommunizieren können sollen, gerät Intels Protected Mode in ein klassisches Dilemma der Macht.

Laut *Programmer's Reference Manual* dürfen auch Tasks des Betriebssystems nicht das Privileg genießen, umstandslos die Tasks einer niederen Privilegstufe aufzurufen. Einfach weil der Verkehr über den Stack gut symmetrisch oder basisdemokratisch läuft, der Aufrufer also genausoviele Bytes pushen muß, wie das aufgerufene Programm wieder poppt, könnte die minder privilegierte Task in Versuchung geraten, die Kontrolle am Ende nicht freiwillig zurückzugeben, sondern sich selber durch einen programmtechnisch simulierten Return auf die Ebene der höheren zu schmuggeln. Deshalb hielten es Intels Ingenieure für sicherer, das boolesche Basiskonzept der Gates oder Gatter der Schaltalgebra zu entwenden und in eine bürokratische Zugangskontrolle umzufunktionieren.

Was solche Verbote schlagend vorführen, ist aber nur die Unmöglichkeit perfekter Zugangskontrollen. In der guten alten Mikroprozessorenzeit, als der Unterschied zwischen System und Applikationen buchstäblich eingebrennt im Silizium selber hauste, das System im ROM, die Applikationen im RAM, war an ihm nicht zu rütteln. Ist der Unterschied dagegen erst einmal programmierbar gemacht, steht er auch schon allen möglichen Umgehungen offen.

18 Vgl. Intel Corporation, 1986, *80386 Programmer's Reference Manual*. Santa Clara/CA., Kap. 17, S. 145: »The DS, ES, FS, and GS segment registers can be set to 0 by the RET instruction during an interlevel transfer. If these registers refer to segments that cannot be used by the new privilege level, they are set to 0 to prevent unauthorized access from the new privilege level.« Beim Codesegment CS scheiden solche Holzhammerschutzmethoden allerdings aus, um nicht das Gesamtsystem zu blockieren.

Ungefähr einhundert-siebzimal, nämlich bei jedem einzelnen 80386-Befehl, wiederholt das *Programmer's Reference Manual* von Intel die Drohung, im Real Mode werde Interrupt 13 ausgelöst, sobald auch nur irgendein Teil der Befehlsoperanden außerhalb des effektiven 20-Bit-Adressraums zu liegen komme. Der 80386, mit anderen, aber immer noch firmeneigenen Worten, laufe also im Real Mode nur als schnellerer AT.<sup>19</sup> Bei Zuwiderhandlungen gelte der drakonische Satz, daß »alle Privilegverletzungen, die keine andere Ausnahme auslösen«, ein Wortungetüm namens »General Protection Exception« triggern.<sup>20</sup> Aber weder die einhundert-siebzig Wiederholungen im Handbuch noch ihre ungezählten Abschriften auf einem Computerbuchmarkt, der unter fiktiven Autorennamen meistens nur maschinelle Teilübersetzungen anzubieten scheint, machen diese Drohung wahrer. Ein einziger Nebensatz im Handbuch verrät nämlich, daß jene Adressiergrenzen des Real Mode nicht mehr und nicht weniger sind als beim Systemstart programmierte Voreinstellungen. Dieser Satz verschwindet natürlich aus allen Übersetzungen, Zusammenfassungen, Popularisierungen und Anwenderbüchern, einfach um den Untertanen von Microsoft seine logische Umkehrung verschweigen zu können: den Satz nämlich, daß Voreinstellungen problemlos zu verändern sind.<sup>21</sup> Anstelle der absichtlich niedrigen Defaultwerte, die die CPU bei je-

19 Vgl. Intel Corporation, 1986, Kap. 14, S. 1 u. ö.

20 Intel Corporation, 1986, Kap. 9, S. 16.

21 Vgl. Harald Albrecht, 1990 a, *Grenzenlos. Vier Gigabyte im Real Mode des 80386 adressieren*. c't, Heft 1, S. 212: »Die Segmentgrenze von 64 KByte ist aber keinesfalls beim 80386 im Real Mode so fest zementiert, wie es zum Beispiel in der Dokumentation Intels über den 386 DX den Anschein hat. Führt man die für die Rückkehr des 80386 aus dem Protected Mode zum Real Mode notwendigen Schritte konsequent weiter, so eröffnet sich plötzlich der gesamte Adressraum von 4 GByte dem Real Mode (womit das Grinsen der Motorola-Fans zusehends an Breite verlieren dürfte).« Das folgende ist weitgehend diesem nachgerade genialen Vorschlag verdankt.

dem Rücksprung zum Real Mode automatisch in die versteckten Teile ihrer Segmentregister lädt, können Programme auch ganz andere Werte setzen. Mit hundert Zeilen Code geht also jeder 386-AT in jede der vier möglichen Betriebsarten: in Protected Modes mit 32-Bit- oder mit 16-Bit-Segmentbreite, aber auch in Real Modes mit der entsprechenden Segmentbreite. Wobei der Real Mode mit 32-Bit-Segmenten den mit Abstand kompaktesten und damit schnellsten Code produzieren dürfte, in den Datenblättern und Handbüchern aber nicht einmal als Möglichkeit Erwähnung findet,<sup>22</sup> um von real existierenden Betriebssystemen des 80386 ganz zu schweigen.

Hundert Zeilen Assembler, aber auch nur Assembler lösen also das Problem einer postmodernen Metaphysik: Auf die Gefahr hin, wahnsinnig zu werden, führen sie unter MSDOS jenseits von MSDOS. Mit der berühmt-berüchtigten Schallmauer, daß der Arbeitsspeicher unter DOS auf ein lächerliches MegaByte beschränkt bleibt, zergehen auch alle WINDOWS nachgerühmten Vorzüge zu nichts. In drastischer Paradoxie ermöglicht gerade das rückständigste aller Betriebssysteme den Ausstieg aus ihm. Intels eingebaute Sperren, die unter komplexeren Betriebssystemen wie etwa UNIX sofort greifen, jene hundert Programmzeilen mithin gar als illegale Befehle sofort herauspicken und verweigern würden – : vor der Dummheit stehen sie ohnmächtig.

22 Eine Ausnahme macht, bezeichnenderweise allerdings ohne jeden Kommentar, Klaus-Dieter Thies, 1989, *PC XT AT Numerik Buch. Hochgenaue Gleitpunkt-Arithmetik mit 8087 .. 80287 .. 80387 ... Nutzung mathematischer Bibliotheksfunktionen in »Assembler« und »C«*. München, S. 638. Edmund Strauss dagegen, obwohl er (laut Vorwort des 80286-Architekten Robert Childs) »has seen the full range of system issues and devised many practical solutions during his work for Intel«, bringt das Kunststück fertig, ein ganzes autoritatives Handbuch lang über nichtdokumentierte Spielräume zu schweigen. Vgl. Edmund Strauss, 1987, *80386 Technical Reference. The guide for getting the most from Intel's 80386*. New York, passim.

Eine Maschine kann also zugleich weniger und mehr, als ihre Datenblätter zugeben. Nicht nur erweitert besagter Trick den adressierbaren Speicherbereich von einem MegaByte auf 4 GigaByte, also etwa viertausendfach. Der 80386 hat auch mindestens zwei »undokumentierte Befehle«, die das Datenblatt zielbewußt verheimlicht,<sup>23</sup> und im 32-Bit-Real Mode mindestens eine Betriebsart, die es gar nicht zielbewußt ignoriert. Solches Chaos herrscht nicht etwa auf den Höhen der Informatik, wo um Berechenbarkeit und Vorhersagbarkeit von Finite-State-Maschinen überhaupt gestritten wird, sondern in schlichter ingenieurmäßiger Empirie. Nur weil, frei nach Morgenstern, »nicht sein kann, was nicht sein darf«, werden den Benutzern bloße Voreinstellungen als Absolutheiten verkauft. Nicht viel anders sorgte einst die Reichspost dafür, daß an Radioendbenutzer der frühen zwanziger Jahre nur Detektor- und keine Röhrengeräte verkauft wurden: andernfalls hätten die Hörer eben auch senden und den militärisch-industriellen Funkverkehr stören können.

Die Informatik, mit anderen Worten, scheint mit internen Informationssperren konfrontiert. Im Raum der Codes, auf die sie faktisch zurückgreifen muß, auch wenn die Theorie ganz andere Modelle erzeugen könnte (und sollte), sind Entzifferungen wider Willen und Wissen der Codeentwickler ebenso möglich wie rar. Offenbar sorgt das Phantom des Menschen noch lange nach Ende von Buchmonopol und Autorschaft dafür, daß Ansichten oder gar Schutzbehauptungen weiter abgeschrieben werden, statt daß es zum Knacken von Codes käme. An genau dieser Stelle müßte ein Arbeitsprogramm stehen – zunächst für Programmierer und im Prinzip wohl auch für Maschinen. So wie es möglich und mittlerweile auch machbar ist, zu-

23 Vgl. etwa Andreas Stiller, 1990, *Bitter für 32-Bitter*. c't, Heft 8, S. 202. Näheres zum Befehl LOADALL (samt der zweifelhaften Behauptung, daß nur der 80286 ihn akzeptiert) siehe bei Norbert Juffa/Peter Siering, 1990, *Wege über die Mauer. Loadall – Extended Memory im Real Mode des 80286*. c't, Heft 11, S. 362 – 366.

fallsgenerierte Programme nach rein darwinistischen Regeln gegeneinander antreten zu lassen, wäre das empirische Schaltverhalten der Maschinen erst einerseits zu dechiffrieren und andererseits ihren Datenblättern entgegenzuhalten.

Zumindest dem Literaturwissenschaftler will scheinen, daß diese sozusagen kriegslistige Sparte der Informatik eine große Zukunft vor sich hat. Sie würde nämlich auf einem strikt technischen Feld nach ähnlichen Verfahren vorgehen können, wie sie die Diskursanalyse Foucaults für Reden und Texte vorgeschlagen hat. Statt wie die Interpretation nach der Bedeutung einer Zeichenkette oder wie die Grammatik nach den Regeln hinter einer Zeichenkette zu fragen, handelt die Diskursanalyse schlicht und einfach von Zeichenketten, sofern es sie gibt und nicht vielmehr nicht. Ob Bedeutungen nicht bloß eine pädagogisch-philosophische Fiktion sind, ob Grammatikregeln vollständig greifen und vollständig greifbar sind, steht alles dahin. Aber daß die zwei Wörter Grammatik und Regel eben in einem Redezusammenhang standen, ist und bleibt Tatsache.

Johannes Lohmann, der große Sprachwissenschaftler und Indogermanist, hat schon vor dreißig Jahren vorgeschlagen, den historischen Ermöglichungsgrund von Programmiersprachen einfach in der Tatsache zu suchen, daß es im Englischen und nur im Englischen Verben wie Read oder Write gibt, Verben also, die im Unterschied zum lateinischen *amo amas amat* undsoweiter alle Konjugationsformen abgestreift haben. Kontextlose Wortblöcke, die nach Lohmann auf das historisch einmalige Durcheinander von Normannisch und Sächsisch im alten England zurückgehen mögen, hindert aber nichts daran, in kontextlose Mnemonics und schließlich in Computer-Opcode übersetzt zu werden. Die endlose Litanei von Read und Write, Move und Load heißt bekanntlich Assembler.

Eine solche Diskursanalyse, deren Elemente ersichtlich nicht nur Wörter, sondern auch Codes sind, würde natürlich den heiligen Unterschied zwischen Alltagssprachen und formalen Sprachen einebnen. Angesichts der schönen

Orthogonalität, mit der zum Beispiel Motorolas Prozessorserie seit dem 68000 prunkt, wäre das wohl Häresie. Die Geschichte vom Protected Mode als einer halb kompatiblen, halb inkompatiblen Fortschreibung guter alter Standards könnte aber lehren, daß Codes auf dieselbe Opazität wie Alltagssprachen hinauslaufen. Beim 8086 gab es bekanntlich gar nicht wenige Befehle, die mit anderen Befehlen synonym waren und ihnen nur in der Ausführungsgeschwindigkeit zuvorkamen. Es machte erhebliche Zeitunterschiede, ob ein Allzweckregister oder aber der Akkumulator seinen Inhalt in den Speicher schrieb. Nachdem die neue Intel-Generation genau diesen Geschwindigkeitsvorteil wegoptimiert hat, die Synonyme unter den Befehlen aber aus Kompatibilitätsgründen überleben durften, hat der Code eine Redundanz erreicht, wie sie die Alltagssprache schon in Freges schönem Beispiel von »Abendstern« und »Morgenstern« aufwies.

Diese Redundanz aber kann historisch nur zunehmen, wenn Maschinencodes über die Generationen hinweg kompatibel bleiben sollen. Im Unterschied zu Alltagssprachen und zumal zum Deutschen, wo es (um sogleich zwei autonome Beispiele zu liefern) weder Wortlängenbegrenzungen noch Wortkombinationslängenbegrenzungen gibt, sind alle Elemente eines Befehlssatzes von endlicher Länge und deshalb auch von abzählbarer Menge. Mit dem Resultat, daß für die erweiterten Befehle etwa des 80386 gar kein Platz mehr da wäre, wenn für sie keine Überlängen zugelassen würden. Spätestens damit beginnen die Codes, wie ökonomisch oder orthogonal auch immer ihr erster Entwurf gewesen sein mag, in Wildwuchs überzugehen. Die silberne Chipoberfläche, zugleich Vorbild und Hauptanwendungsfeld aller topologischen Optimierungen, verliert diese ihre mathematische Transparenz: Sie wird zum babylonischen Turm, in den die Trümmer schon wieder abgerissener Türme eingebaut bleiben. Der Protected Mode als Feindschaft und Koexistenz mit einem technisch längst überholten Real Mode ist Computergeschichte *on chip*. Und David Hilberts traumhaftes Programm, die Opazität der

Alltagssprachen ein für allemal durch Formalisierung auszuräumen, scheidet nicht nur auf den lichten axiomatischen Höhen Gödels oder Turings, sondern in ingenieurmäßiger Empirie. Codes mit Kompatibilitätsproblemen beginnen zu wuchern und genau jene Opazität von Alltagssprachen anzunehmen, die die Leute seit Jahrtausenden zu Subjekten dieser Sprachen gemacht hat. Der schöne Begriff Quellcode wird buchstäbliche Wahrheit.

Sicher, eine Diskursanalyse kann solche Wucherungen weder zähmen noch debuggen. Aber es ist womöglich effizienter, mit ihnen einfach zu rechnen. Turings alter Gedanke, die Maschinen selber ihre Codes erwürfeln zu lassen, mag insgeheim schon wahr geworden sein. Gerade weil »die komplexe Funktion hochintegrierter Schaltkreise (von Speicher-ICs abgesehen) nicht mehr wie bei einer einfachen logischen Verknüpfung durch Testen aller möglichen Eingangssignalkombinationen überprüft werden kann«<sup>24</sup>, wären herstellerunabhängige Tests am Platz. Widerstände, wie das US-amerikanische Patentrecht sie nachgerade zum System erhoben hat, sollten nicht daran hindern, lauter Meßwerte, Patches und Umgehungstechniken, von denen in offizieller Paperware keine Rede ist, unter die Leute zu bringen. Das wäre, ob für den Frieden oder nicht, Information über Informatik.

Lesen zu können, »was nie geschrieben wurde«, dichtete einst Hugo von Hofmannsthal dem »wundervollen Wesen« namens Mensch an. Im Chaos der Codes, das mit der welt-historischen Abdankung von Alltagssprachen zugunsten einer Universalen Diskreten Maschine beginnt, müßte solche Kryptoanalyse universal und maschinell werden.

24 Lowe, 1985, S. 70.

---

Friedrich Kittler

**Draculas Vermächtnis**

---

Technische Schriften

RECLAM VERLAG LEIPZIG

ISBN 3-379-01476-1

© Reclam Verlag Leipzig 1993 (für diese Ausgabe)  
Quellen- und Rechtsnachweis am Schluß des Bandes

Reclam-Bibliothek Band 1476

1. Auflage, 1993

Reihengestaltung: Hans Peter Willberg

Umschlaggestaltung: Friederike Pondelik unter Verwendung  
der Computergrafik »Tanz der Silikone« von Werner Drescher

Printed in Germany

Satz: Schroth Fotosatz GmbH Limbach-Oberfrohna

Druck und Binden: Offizin Andersen Nexö Leipzig GmbH

Gesetzt aus Meridien

---

# Inhalt

Vorwort .....	8
I	
Draculas Vermachtnis .....	11
Die Welt des Symbolischen – eine Welt der Maschine .....	58
II	
Romantik – Psychoanalyse – Film: eine Doppelgängergeschichte .....	81
Benns Gedichte – »Schlager von Klasse« .....	105
Der Gott der Ohren .....	130
III	
Vom Take Off der Operatoren .....	149
Signal-Rausch-Abstand .....	161
Real Time Analysis, Time Axis Manipulation .....	182
Protected Mode .....	208
Es gibt keine Software .....	225
Literaturverzeichnis .....	243
Quellen- und Rechtsnachweis .....	258