

## Neue Hardware und modulares Echtzeitbetriebssystem

Deutschland und die EU brauchen eine eigene Hardware und Software, möglichst Open Hardware und Open Source. Offene Hardware/Software schafft Vertrauen und kann bei der Verbreitung helfen, auch international. Geld kann mit dem Verkauf von Hardware und Dienstleistungen verdient werden, vor allem auch indirekt, durch allgemeine Produkte (z.B. supersichere Smartphones, Computer, usw.). Das anzustrebende möglichst perfekte (sichere, zuverlässige, nutzerfreundliche, modulare) Echtzeitbetriebssystem (RTOS) wird dann überall Verwendung finden, wo man ein Betriebssystem (OS) benötigt, von Embedded bis zu Supercomputern, in der Infrastruktur, beim Militär, in Haushaltsgeräten, Autos, usw.

Die Bedeutung dessen ist gigantisch, das ist neben der KI-Entwicklung eine weitere, sehr entscheidende Revolution von potentiell weltweiter Bedeutung. Daher ist mit erheblichem Widerstand der Profiteure der derzeitigen Mißstände und ihrer Diener zu rechnen. Das ist nicht zu unterschätzen. Wichtige Maßnahmen sind daher Open Source, Open Hardware und Staatlichkeit, d.h. der Staat muss dies gezielt und mit hoher Priorität fördern, unterstützen, realisieren, absichern, usw. Deutschland kann da eine maßgebliche Rolle spielen aber sicher auch die EU. Nach Möglichkeit ist dabei mit guten Staaten zu kooperieren, zusammenzuarbeiten.

Das Vorgehen ist prinzipiell ganz einfach: Deutschland und EU müssen das Projekt mit hohem jährlichen Milliardenetat (whatever it takes aber vernünftig und möglichst ohne Geldverschwendung) realisieren und dabei das Projekt auch überwachen, kontrollieren und koordiniert gegen Fehlschläge und Bedrohungen schützen. Das Projekt hat ohne Wenn und Aber 100% staatlich zu sein - natürlich können auch Unternehmen mitarbeiten aber der Staat (bzw. die EU) soll die Leitung haben, die Sourcen hosten, die Server (für Updates, usw.) betreiben.

Das ist alternativlos. Ein Rumgemurkse nur mit Privatkapitalisten, Konzernen, usw. kommt da nicht infrage. Die Mächte des Bösen wollen das Projekt vermutlich verhindern und würden dafür vielleicht Hunderte Milliarden Dollar ausgeben wollen. Dabei geht es nicht nur um jährliche Milliardenprofite, sondern vor allem auch um Macht, Einfluß und quasi die Weltherrschaft.

### Inhaltsverzeichnis

Neue Hardware und modulares Echtzeitbetriebssystem.....	1
Überzeugung von Deutschland oder EU.....	1
Eine Chance für den Amiga?.....	1
Erhebliches Optimierungspotential.....	1
3D Chip Stack.....	1
Co-Prozessoren.....	1

## Überzeugung von Deutschland oder EU

Deutschland und EU müssen von dem Projekt (u.a. modulares Echtzeitbetriebssystem) überzeugt werden. Das ist wegen schlechter Menschen in Machtpositionen nicht so einfach aber sollte machbar sein. Politische Parteien und Organisationen können dabei eine Rolle spielen aber offensichtlich hat sich da in über 25 Jahren nicht genug getan - hoffentlich kriegt Deutschland bald mal eine supergute Rechtspartei.

Windows, Linux und macOS sind vermutlich überladen, quasi verfettet aber das zu entwickelnde RTOS muss auch möglichst schnell und effizient sein. Ein wichtiger Aspekt ist dabei die Modularität: für bestimmte Funktionalitäten kann es unterschiedlich aufwendige Module geben: von möglichst klein/schlank bis so umfangreich wie sinnvoll. Hierzu braucht es einen Neustart (eine Neuentwicklung), denn die verfetteten Betriebssysteme lassen sich schwerlich perfektionieren. Die Entwicklung des neuen RTOS ist eine Aufgabe für Experten über etliche Jahre, das ist nicht so einfach aber andererseits gibt es schon viel Erfahrung und funktionierende Beispiele, wo man eigentlich nur die besten Ideen/Konzepte neu kombinieren muss.

Das neue Echtzeitbetriebssystem sollte nach Möglichkeit Programme von Windows, Linux, macOS, usw. laufen lassen können - wie auch immer und was halt rechtlich geht. In Zukunft können Programme von KI automatisch neu kompiliert werden.

## Eine Chance für den Amiga?

Vielleicht kann man mit dem [AmigaOS](#) und einer neuen [APOLLO CORE 68080](#) zeigen, was mit nach derzeitigen Maßstäben leistungsschwacher Technik möglich ist.

Ein Schritt in Richtung neuer Hardware und neuem RTOS ist vielleicht auch die Förderung von Open Source Amiga (68000 oder darauf aufbauend) und einer neuen, verbesserten 68000 CPU. Die Hardware kann dabei ggf. erstmal proprietär sein und später gegen einen höheren Betrag Open Hardware werden, wobei das vielleicht nicht so entscheidend ist, weil eine moderne, neue Hardware (CPU, GPU, usw.) entwickelt werden muss und es nur eine überzeugende CISC (Complex Instruction Set Computer) Demonstration braucht.

Zu dieser Demonstration ist natürlich auch eine Abschätzung zu liefern, wie viel leistungsfähiger das mit einer professionellen, modernen Multicore-CPU wäre und vielleicht auch noch eine Demonstration, wie viel langsamer ein aktuelles Linux im Vergleich zu AmigaOS ist. Na klar kann Linux mehr aber man kann trotzdem einen Vergleich, eine Abschätzung versuchen. Wenn ein neues OS auf neuer Hardware bei vergleichbaren Fähigkeiten vielfach effizienter/permanter als Windows, Linux und macOS ist, dann ist das ein riesiger Wettbewerbsvorteil. Ein universelles RTOS sollte sich weltweit durchsetzen können und das ist nicht weniger als eine kleine Revolution.

Wohl bemerkt: das dann irgendwann professionell neu entwickelte RTOS muss mit AmigaOS nicht mehr so viel gemeinsam haben aber das hängt davon ab, was man zuvor noch aus AmigaOS machen kann.

Des Weiteren braucht AmigaOS eine Art Stiftung zur Finanzierung von Open Source für den Amiga. Diese Stiftung sollte gemeinnützig sein, zumindest sollte man auch hohe Geldbeträge ohne Schenkungssteuer spenden können - eine steuerliche Absetzbarkeit der Spenden für den Spender halte ich nicht für ganz so wichtig.

Wichtig ist, dass das Geld möglichst sinnvoll eingesetzt wird. Ich würde das Betriebssystem und wichtige Anwendungen (eMail, Browser, so was wie LibreOffice, usw.) priorisieren und nicht Spiele. Dabei sollte es möglichst viel Transparenz geben: die Verwendung eines jeden Euro ist zu veröffentlichen, auch Gelder, die an Entwickler gehen, mit deren Namen und Projekten. Wie viel Geld Entwickler bekommen sollten, kann man diskutieren aber Mindestlohn wäre vermutlich schon zu viel, das würde die (anfänglichen) Möglichkeiten der Stiftung übersteigen.

Ein bedeutender Aspekt kann Werbung für die Stiftung sein: möglichst kostenlos und natürlich angemessen in Foren, in der Szene und mit Versuchen, die Presse dafür zu begeistern. Dafür kann vielleicht dann doch eine Menge Geld zusammenkommen.

Ein konkretes Ziel können Entwicklung und Verkauf eines brauchbaren AmigaOS Notebooks sein, auf dem auch eine abgespeckte Linux-Version läuft. Das Notebook sollte wertig aber nicht zu teuer sein - vielleicht kann man ca. 1000 Euro anpeilen aber lieber für 1500 Euro als zu schlechte Qualität. Leistungsmäßig kann man da nicht so viel erwarten aber Internet, eMail, usw. sollten passabel funktionieren.

## Erhebliches Optimierungspotential

Derzeit wird vermutlich mittels verbesserter Fertigungstechnik (kleinere Strukturen) mehr Leistung bei der Chiptechnik rausgeholt aber was könnte man mit einem CPU-Redesign z.B. auf verbesserter PowerPC Basis erreichen, z.B. auch mit 48 statt 64 Bit für mehr Effizienz (siehe auch [Von der Überlegenheit der 48 Bit-CPU's](#))? Vielleicht mehr als Faktor 2? Mindestens einen weiteren Faktor 2 kann man mit verbesserter Software rausholen. Der 3. Faktor 2 ergibt sich durch optimierte Anwendungsfälle, also weniger Bullshit. Das wären dann zusammen Faktor 8 und vielleicht geht noch mehr, z.B. bis zu Faktor 16 (natürlich nicht bei bereits schon sehr optimierten Einzelanwendungen aber insgesamt gibt es vermutlich ein erstaunliches Optimierungspotential).

Das ist spektakulär und im weltweiten Wettbewerb können bereits wenige Prozent mehr bringen. Diese Optimierungsmöglichkeiten werden derzeit nicht genutzt, weil 386 Code so verbreitet ist ... aber China ist groß und schlau genug für neue Ansätze, Russland auch. Das kleine Deutschland muss sich sputen, da mitzuhalten - Deutschland soll endlich aufwachen. Wie viel würde ein modulares und supersicheres [RTOS](#) bringen, das überall genutzt werden kann?

Neuronale KI sind mangelhaft und ineffizient. Eine gezielt programmierte algorithmische KI (AKI) kann vielfach effizienter und sicherer sein und der Clou: AKI werden sich eines Tages selbst weiterentwickeln können. Das kann noch mal über Faktor 8 bringen, zusammen vielleicht um die Faktor 100 Effizienzgewinn. Okay, das ist eine etwas schräge Rechnung aber es gibt riesiges Optimierungspotential und im Wettbewerb zwischen den Staaten wird das schnell kommen, weil Russland und China nicht so wie die EU von den USA kontrolliert werden.

Die bisherige Computertechnik ist schon erstaunlich leistungsstark aber was, wenn mit bestehender Technologie Faktor 10-100 drin sind? Diese Zusatzleistung kann bei weiter verbesserter Hardwareleistung in einigen Jahrzehnten abrufbar sein und das wird kommen, von wegen China, Russland und hoffentlich auch Deutschland, das endlich aufwacht. Das bedeutet, dass dem Menschen weit überlegene KI dieses Jahrhundert so gut wie sicher sind, dass es bereits 2050 erstaunliche KI geben kann, die dann auch Software entwickeln können (eine Revolution und wichtiger Schritt bei der technologischen Singularität). KI können dann auch Chips designen und entwickeln.

Deutschland braucht dringend einen Masterplan für Open Source, eigene KI-Entwicklung und völlige Unabhängigkeit von den USA. Ein sehr wichtiger Aspekt ist auch die Systemoptimierung hin zum idealen [Sozialismus](#).

Das große China hat das schon kapiert - macht euch das mal klar. Das kleine Deutschland muss da mithalten, wenn es nicht zum pittoresken Tourismusgebiet für Chinesen werden will, um ein paar Devisen für Einkäufe in China zu sammeln. Deutschland könnte jährlich über 100 Milliarden Euro effizienter sein und das Geld in Open Source, Chipentwicklung und KI-Entwicklung stecken.

## 3D Chip Stack

Weiß nicht aber die Verkleinerung der Chipstrukturen stößt so langsam auf ihre Grenzen, wobei man natürlich bedenken muss, dass es um die Größe der kleinsten Chipstrukturen geht und nicht um die Auflösung der Fertigungstechnik.

Vielleicht kann die 3D Stapelung von Chipflächen eine Lösung sein, auch wenn das auch beträchtliche Nachteile hat: die Kühlung ist ein großes Problem, man braucht ein ausgeklügeltes Wärmemanagement auch für gleichmäßige Wärmeverteilung und auch der Flächenverbrauch durch Durchkontaktierungen, die bei Chipstapeln besser in der Fläche als am Rand vorgenommen werden, weil das einfacher und robuster ist. Dabei verbraucht man erstmal mehr Chipfläche und nicht weniger aber es gibt einen potentiell großen Vorteil: die Leitungswege/Signalpfade können kürzer werden.

Man kann auch RAM/Cache und CPU-Logik mischen, z.B. in abwechselnden Schichten. Das Kühlungsproblem muss ganz einfach mit geringerer Taktfrequenz und somit lokal stark verminderter Rechenleistung angegangen werden aber dadurch spart man automatisch Energie und ein Chipstapel kann womöglich trotzdem noch auf passable oder sogar erstaunliche Leistungen kommen: ggf. mehr Rechenleistung bei weniger Energieverbrauch - genial.

Zum Wärme- und Energiemanagement gehört, dass man u.a. überflüssige Rechenschritte, also sogenannte spekulative Codeausführung meiden muss. Des Weiteren kann man mit 40 oder 48 Bit anstatt 64 Bit Platz und Energieverbrauch sparen, siehe auch [Von der Überlegenheit der 48 Bit-CPUs](#).

Natürlich sind kleinere Chipstrukturen potentiell sparsamer, auch in der Herstellung (mehr Dies auf dem Wafer), zumindest, wenn es nicht zu viel Ausschuss gibt aber die 3D Chipstapel gehen natürlich auch mit größeren Strukturen, das ist vielleicht sogar besser, z.B. mindestens 15 nm.

Noch eine Möglichkeit: Mit Chipstapeln, bzw. 3D-Chips (3D Chip Stack) können weitgehend identische Strukturen von Speicher und Rechenlogik mehrfach in einem 3D-Chip vorkommen und das kann für optionale Redundanz genutzt werden, indem die gleiche Rechnung gleichzeitig mindestens zweifach ausgeführt und dann das Ergebnis verglichen wird - bei einem Fehler muss die Rechnung wiederholt werden oder man entscheidet sich automatisch für die 2 (X-1) gleichen Ergebnisse von 3 (X) Parallelrechnungen. Der Clou: das kann optional sein, man kann die gleichen Stapelchips in unterschiedlichen Modi nutzen.

Ein kleines Beispiel: anstatt eine z.B. 10 cm x 10 cm große flache Ein-Chip-Fläche kann man 150 x 1 cm x 1 cm Chipflächen übereinander zu einem 1 Kubikzentimeter Chipwürfel stapeln (150 statt 100 wegen Verluste durch Kontaktflächen aber andererseits spart man so ggf. auch viel Leiterbahnlänge, so dass 3D-Chips bei gleicher Fläche vielleicht sogar mehr Transistoren haben können).

Wichtig ist, dass das Chiplayout/-Design per Computer/Programm möglichst automatisch erstellt wird und nicht handgefrickelt ist, denn einen Programmfehler kann man einmal beheben und dann alles neu berechnen aber bei Handarbeit können immer wieder Fehler passieren. Per Computerprogramm kann man auch automatisch verschiedene Designs/Varianten realisieren, z.B. mit mehr oder weniger Cash, Floatingpointmodulen, usw.

Vielleicht können optimierte FPGA noch mal bei KI und insbesondere Robotern eine Rolle spielen - man bedenke, dass man bei FPGA intern auch mehrere Schichten stapeln können sollte. Vielleicht kann man z.B. die Steuerung grundlegender Körperfunktionen von Robotern mit FPGA realisieren.

[VLIW](#) (oder eher die goldene Mitte) kann eine gute Idee sein - [Sprungvorhersage](#) und rechnen auf Verdacht ist dagegen bescheuert, das kostet Energie und Transistoren.

Mehrere Chipwürfel können auf einem Mainboard kombiniert werden und es kann auch noch gemeinsames RAM geben. Das ist vermutlich keine so blöde Idee - gute Experten sollen das ausarbeiten.

## Co-Prozessoren

Man kann mal über Co-Prozessoren nachdenken: es kann auch unterschiedliche Co-Prozessoren geben (möglichst mit einem einheitlichen Sockel). Der Vorteil sind günstigere CPU und bessere Wärmeverteilung, sowie optional mehr Rechenleistung durch mehrere Co-Prozessoren. Nicht alles auf einen Chip packen, kann die Ausbeute (den Yield) verbessern. Die

CPU kann natürlich auch die Aufgaben eines (fehlenden) Co-Prozessors übernehmen. Ein Entwicklungsziel sollte (neben Sicherheit, Robustheit, Zuverlässigkeit, Effizienz) auch ein geringer Stromverbrauch sein (wenig Wärmeentwicklung) aber das nicht nur mit gefährlich kleinen Strukturen, sondern ggf. mit weniger Rechenleistung, die durch bessere Software und klügere Nutzung kompensiert werden muss. Wenn ein einzelner Chip nicht so viel Strom verbraucht, ist auch die Stromversorgung einfacher, nicht mehr so kritisch. Co-Prozessoren können auch eigenen Caches und natürlich eigene CPU-Cores haben. Zumindest kann man mal darüber nachdenken.

## **Hier steht dazu relativ wenig**

aber vielleicht fällt mir ja noch was ein.

Die Grundidee ist ganz einfach, da muss man eigentlich nicht viel dazu schreiben - man muss Deutschland und EU überzeugen. Vielleicht kann die Bundeswehr ein größeres Software-Team bekommen und auch am neuen RTOS mitprogrammieren.

Zu der Amiga-Idee (Stiftung) brauche ich eigentlich nur einen hohen Lottogewinn - man sollte meinen, dass sich dann einige für die Stiftung finden, wenn sie dafür z.B. 1 Million Euro gespendet bekommen.

Die Hardwareentwicklung kann auch eine höhere Spende bekommen, auch wenn sie proprietär sein sollte - die Hoffnung ist, dass das dann später noch Open Hardware werden kann, ggf. auch gegen weitere Geldzahlungen.

(alles imho)