Amiga 500/2000 – 1 MB Chip RAM erklärt

Auf http://amiga.resource.cx bin ich auf eine PDF gestoßen die euch erklärt wie ihr 1 MB Chip Ram auch bei älteren Amiga500/2000 Revisionen realisieren könnt.

So können zum Beispiel ältere Modelle wie der Amiga 500 mit der Revision 3-5, oder der Amiga 2000 mit der Revision 4.1 bis 4.5 nur 512 KByte Chip Ram ansprechen. Der Grund ist der verbaute Fat Agnus Chip Version 8371.

Wie ihr dies ändern könnt und was zu tun ist erfahrt ihr ausführlich in der PDF.

Hier gehts zur pdf: http://amiga.resource.cx/manual/1MBChip_A500_A2000.pdf

Alternativ (falls die Quelle nicht mehr verfügbar sein sollte) habe ich die Datei auch hier auf amiga68k.de hochgeladen:

1 MByte Chip Ram für ältere Amiga Revisionen
Download Now!124 Downloads

Amiga Shell Befehle

Frage: Was sind SHELL Befehle und welche gibt es?

Mit der Maus lässt sich ein Betriebssystem wie das robuste

Amiga OS recht gut steuern. Doch oft ist man mit Kommandozeilenbefehle in der Konsole viel schneller am Ziel. Einziges Problem: Man muss die Befehle kennen.

Zur Übersicht: Alphabetische Auflistung der Amiga DOS Konsolen-Kommandos. Als Befehl gilt jedes Programm, was in der über path definierten Ordnerliste liegt. Üblicherweise werden die Progamme aber in den Ordner “C:” auf der Amiga Start-Festplatte verschoben, um als Amiga Shell-Programme benutzbar zu sein.
Amiga Befehl: Beschreibung

- addbuffers:Erweitert den System-Cache für ein Laufwerk
- adddatatypes: Erzeugt ein Liste der Data Types
- alias: Setzt Verknüpfungen auf andere Dateien
- ask: Erzeugt eine ja/nein Abfrage während eines Shell-Skripts
- assign: Erstellt Verknüpfungen auf vorhanden Laufwerke und Ordner. So ist C: einAssign auf Sys:C/
- avail: Zeigt den verfügbaren RAM an
- binddrivers: Aktiviert Device-Treiber aus dem Expansion Ordner
- break: Bricht den genannten Prozess ab
- cd: Wechselt den aktuellen Ordner. cd bilder wechselt in den Bilder-Ordner
- changetaskpri: Ändert die Priorität des Shell-Tasks
- conclip: Tauscht Daten zwischen Konsole und Clipboard (Zwischenablage) aus.
- copy: Kopiert Dateien und Ordner
- cpu: zeigt den in den Amiga eingebautenProzessor (CPU) und schaltet den Cache an oder aus.
- date: Zeigt oder ändert das Amiga-Systemdatum
- delete: löscht Dateien oder Verzeichnisse
- dir: Zeigt den Datei-Inhalt eines Ordners an. Mehr Informationen zeigt der Befehl “list”
- diskchange: Meldet dem Amiga-System einen Diskwechsel
- echo: Gibt eine Zeichenkette aus: echo test

- ed: Startet den Editor ED
- edit: Editiert Text zeilenweise
- else: Alternative Anweisung einer IF-Abfrage in einem Shell-Script
- endcli: Beendet den aktuellen Shell-Prozess und schliesst das Konsolenfenster
- endif: Beendet eine IF-Abfrage in einem Shell-Script
- endshell: Beendet einen Shell-Prozess
- endskip: Beendet einen SKIP-Block in einem Shell-Script
- eval: Vergleicht Integer- oder Boolesche Ausdrücke
- execute: Startet ein Skript mit optional mit übergebenen Argumenten
- failat: Definiert, ab welchem Fehlerwert eine Befehlssequenz abbricht
- fault: Zeigt die Fehlerbeschreibung einer DOS-Fehlernummer an
- filenote: Fügt einen Kommentar an eine Datei an
- get: Holt den Wert einer lokalen Variable
- getenv: Holt den Wert einer globalen Variable
- iconx: Startet ein Shell-Script über ein Icon
- if: Mit IF-Abfragen lassen sich Entscheidungen in einem Shell-Script definieren
- info: Zeigt Informationen über die gemounteten Laufwerke (RAM-Disk, Festplatten,Disketten) an
- install: Schreibt oder überprüft einen Disk-Bootblock
- iprefs: Zeigt die Systemeinstellungen an
- join: Fügt mehrere Dateien in eine neue Datei zusammen
- lab: Definiert ein Label für eine Skript-Datei
- list: Zeigt eine genau Datei-Übersicht eines Ordners, ähnlich wie der Befehl “ls” unterLinux oder “dir” unter MS-DOS
- loadresource: Lädt eine System – Resource in den RAM
- loadwb: Startet die Amiga Workbench
- lock: Macht ein Laufwerk schreibgeschützt
- magtape: Spult SCSI Streamerkasseten vor oder zurück
- makedir: Erstellt einen neuen Ordner
- makelink: Erzeugt Verknüpfungen zeischen Dateinamen

- mount: Macht ein Device verfügbar. CD-Laufwerke müssen z.B. einmal gemountetwerden
- newcli: Öffnet ein neues Amiga Shell-Fenster
- newshell: Öffnet ein neues Amiga Shell-Fenster
- path: Kontrolliert die Liste an Verzeichnissen, die von der Shell nach Befehlen durchsucht werden
- prompt: Ändert das Aussehen der Eingabeaufforderung des aktuellen Shell-Fensters
- protect: Ändert das Protection Bit von Dateien oder Ordnern
- quit: Beendet ein Shell-Script
- relabel: Benennt das Medium im aktuellen Laufwerk um
- remrad: Entfernt die geschützte RAM Disk RAD. Der Inhalt von RAD geht bei einemNeustart nicht verloren
- rename: Ändert den Namen von Dateien und Verzeichnissen, kann auch zum Verschieben genutzt werden: rename C:prog s:prog
- requestchoice: Erlaubt die Benutzung von Standard – Systemabfragefenstern unter dem Amiga DOS
- requestfile: Erlaubt die Benutzung der Standard-Dateiauswahl
- resident: Zeigt und ändert die Liste der Systembefehle im Speicher
- run: Startet ein Programm als Hintergrund – Prozess, so dass in der aktuellen Shell weitergearbeitet werden kann
- search: Durchsucht Dateien nach Textstücken
- set: Füllt eine lokale Variable
- setclock: Zeigt oder ändert die Systemuhr des Amigas
- setdate: Ändert das Datum einer Datei
- setenv: Füllt eine globale Variable
- setfont: Ändert den Zeichensatz der Shell
- setkeyboard: Ändert das Keyboardlayout (deutsch, englisch…) der Shell
- setpatch: Startet einen Patch für das installierte ROM
- skip: Springt zu einem Label innerhalb eines Shell-Scripts
- sort: Sortiert die Zeilen einer Datei alphabetisch

- stack: Zeigt oder ändert die Stackgröße der Shell
- status: Listet Informationen über die aktuellen Shell-Prozesse
- type: Zeigt den Datei-Inhalt an
- unalias: Entfernt eine Verknüpfung
- unset: Löscht eine lokale Variable
- unsetenv: Löscht eine globale Variable
- version: Zeigt die aktuelle Versionsnummer der gewählten Datei. Ohne Dateiangabewird die Version des Betriebssystems gezeigt
- wait: Hält ein Shell-Script oder die Shell für eine bestimmte Zeit an
- which: Sucht den Commandopfad eines Objekts
- why: Zeigt die ausführliche Fehlermeldung, mit der der vorherige Befehl abgebrochen wurde

---

Amiga ohne Startup Sequence starten für max. ChipRam

Tales of Gorluth ist erschienen. Und genau wie ihr, hab ich mich auf das Spiel sehr gefreut.

Nachdem ich nun etwas Zeit gefunden habe, um mich dem Spiel zu widmen hatte ich Schwierigkeiten das Spiel zu starten.

Das Spiel startete zwar von der Workbench aus, doch nach kurzem Spielen, landete ich wieder mit einer Fehlermeldung (“Out of Memory”) auf dem Desktop.

Ein Nachlesen in deutschen und englischen Foren bestätigt, das ich mit dem Problem nicht alleine bin.

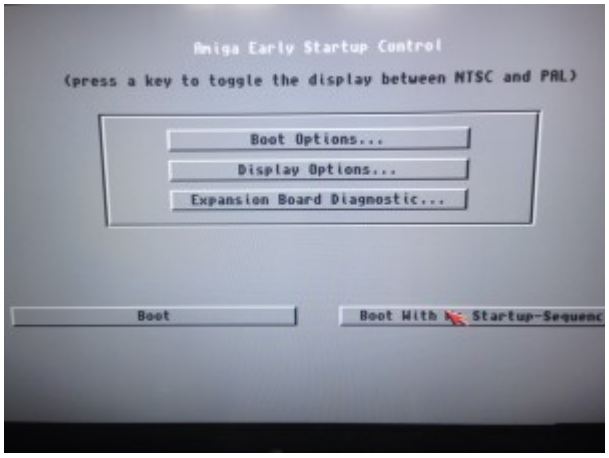
Das Problem ist, Tales of Gorluth verlangt volle 2 MByte

ChipRam.

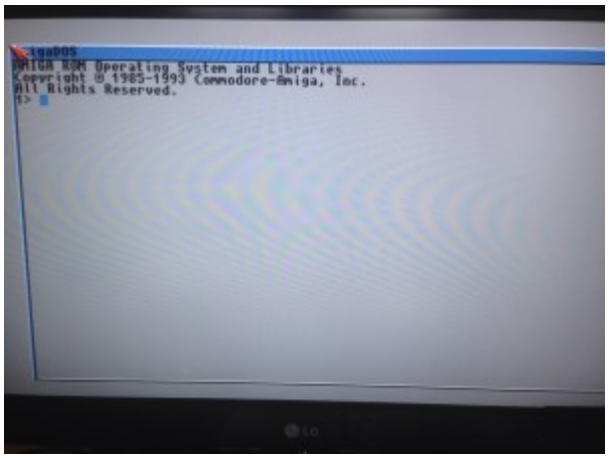
Ein Trick den ich versuchte, war die Auflösung auf LowRes , 2 Farben und weiterhin alle Speicherfresser abzustellen. So bekam ich 1.9 MByte frei, dies reichte jedoch auch noch nicht.

Doch wie bekommt man nun volle 2 MByte Frei und kann die Demo ohne Hindernisse spielen?

Dazu müssen wir den Amiga ohne StartupSequence starten (beide Maustasten beim Start des Amigas gedrückt halten)



Anschliessend erscheint das AmigaDOS



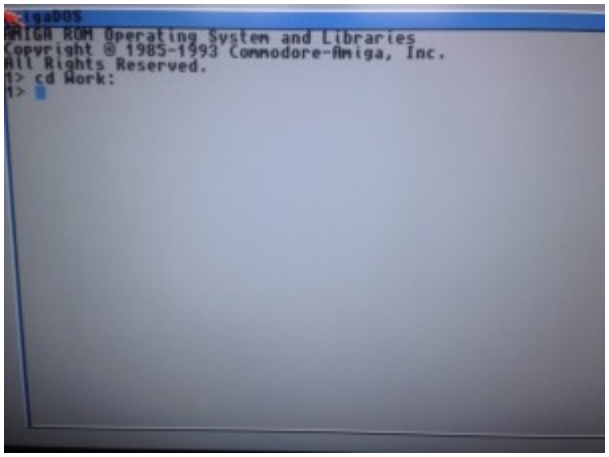
Dort müsst ihr nun zu der Partition vordringen auf der ihr das Demo abgelegt habt.

Bei mir ist es die Partition “Work”

Um auf die Partition zu gelangen nutzt man das Kommando “cd”

In meinem Fall also cd work:

(Der Doppelpunkt kann sich unter Umständen unter dem “Ö” verstecken, also “Shift + Ö” dafür drücken.



Nun befinden wir uns auf der Partition Work im Hauptverzeichnis.

mit dem Befehl “dir” werden alle Dateien angezeigt die sich auf Work befinden.

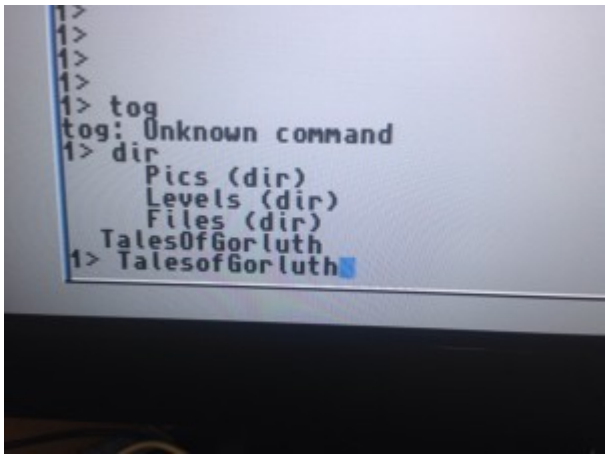
Meine Tales of Gorluth Demo befindet sich im Verzeichniss “tog”

Um in dieses Verzeichnis zu wechseln geben wir jetzt einfach “tog” ein.

Nun lassen wir uns mit dem Befehl “dir” wieder alle Dateien anzeigen.

Es müsste dann so aussehen wie auf dem Bild hier drunter.

Um Tales of Gorluth nun zu starten, gebt ihr einfach “TalesofGorluth” (ohne Anführungszeichen natürlich) ein.



Nun solltet ihr diesen Bildschirm sehen.

Glückwunsch, die Demo wird nun ohne Probleme funktionieren.

Weiteres Problem!

Wer anstelle des Startbildschirms eine Fehlermeldung bekommt “**exec.library**” oder “**xec.library**” oder “**Failed Returncode 20**” hat wahrscheinlich ein gepatchtes System so wie ich es hatte.

Bei mir befand sich die exec.library im FastRam und die Demo lies sich nicht starten.

Ob eure exec.library auch im FastRam liegt, könnt ihr ganz einfach mitSysInfo überprüfen.

Dort muss hinter der exec.library auf jeden Fall “ChipRam” stehen.

Sollte dort “32Bit” stehen, geht bitte in die StartupSequence

und ändert dort den entsprechenden Eintrag der die exec.library in das FastRam legte.

Amiga Startup Sequence editieren

Startup Sequence Editieren

StartUpSequence:
Die Startup und die die User-Startup sind die 2 Scripte (Batch-Dateien) über die der Amiga gestartet wird (z.B. von HD). Vielleicht etwas vergleichbar mit der autoexec.bat unter DOS, nur das das OS schöner ist.
Wie editieren?
Shell öffnen und da dann “ed s:startup-sequence” oder “ed s:user-startup ” eintippen. Speichern kann man mit ESC dann q und return.
Was eintragen?
Fast alle Befehle und deren Wirkung hab ich in folgender Liste zusammengetragen
Was beachten?
Wichtig, macht euch IMMER eine Kopie der startup sequence bevor ihr ans experimentieren geht!

LHA Dateien entpacken am

Amiga

LHA entpacken:

Im Aminet erhältliche Files sind vorzugsweise LHA-gepackt. Das bringt das berühmte (Henne Ei) Problem wie man LHA auf einen nackten Amiga bekommt. Das ist ab OS 2.1 und grösser recht einfach:

- 1. Diesen Schritt zuerst ausführen: Wie bekomme ich Daten vom XP PC zum Amiga? (min. WB 2.1)
- 2. Diese Datei vom PC zum Amiga transferieren: http://it.aminet.net/pub/aminet/util/arc/lha.run Das ist ein selbstextrahierendes Archiv.
- 3. Wenn sich diese Datei in der RAM Disk befindet einfach Doppelklick darauf, dann entpackt sich das Archiv an der Stelle wo es sich befindet.
- 4. Wir finden danach einen Guide und 2 Versionen des LHA im Ordner, LHA_68k und LHA_68020. Je nach Amiga muss eine Datei in LHA umbenannt werden und im Workbench Verzeichnis in C: kopiert werden.
- 5. Da entpacken über die Shell für unerfahrene nicht ganz einfach ist, holen wir uns gleich noch

ein Front-End für verschiedene Packer, z.B. Packmaster das für eine Vielzahl von Packern brauchbar ist. Zu finden zum Beispiel hier: http://de4.aminet.net/util/arc/Packmaster128.lha

- 6. Mit Diskette das Archiv auf den Amiga bringen.

- 7. Wenn wir das Archiv zum Beispiel in Work kopiert haben, müssen wir es mit folgendem Befehl entpacken:
- LHA e Work:Packmaster128.lha
- Das Archiv entpackt sich dann in Work. Es gibt eine Vielzahl von Optionen zum LHA – Befehl, die aber hier nur verwirren würden, weshalb ich hier besonders einfach vorgehe.

- 8. Mit dem Install-Script können wir nun Packmaster installieren und besitzen nun eine grafische Oberfläche, die in den Einstellungen auf unsere Bedürfnisse angepasst werden kann.

Unter Windows (zb. XP) Amiga Disketten formatieren und Daten übertragen

Jeder kennt das Problem, im Aminet tummeln sich die guten und nützlichen Tools, aber man kommt nicht ran.

Abhilfe schafft da eine unter Windows formatierte Amiga Diskette.

Also um unter Windows auf DD (HD Disketten funktionieren auch mit zugeklebtem zweitem Loch) zu formatieren musst man folgendes eingeben, wenn du auf “Ausführen” klickst:

format a: /t:80 /n:9

Damit wird die Amiga Disketten in ein Windows kompatibles Format umgewandelt.
Nun könnt ihr eure Daten auf die Diskette packen und diese in den Amiga stecken.

Dort wird nun **Workbench 2.05 und grösser vorausgesetzt**.

Im **Storage/Dosdrivers** habt ihr ein Symbol “**PC0**” ein doppelklick darauf bei eingelegter Diskette sollte diese nun auch am Amiga sichtbar machen.

Um “PC0″ dauerhaft zu mounten sollten ihr das Symbol nach **Devs/DosDrivers**kopieren. Damit steht die Diskette auch nach dem Neustart ohne umständliches “Neu”mounten sofort zur Verfügung.

Das stellt eine einfache Lösung für kleinere Programme dar, weitere Tutorials (um grössere Datenmengen vom PC auf den Amiga zu bringen) werden folgen, also bleibt dran

---

WBStartup – Die Workbench Startup erklärt…

….mit einigen wichtigen Funktionen

Der WBStartup Ordner ist ähnlich dem “Autostart” von Windows.

Programme die sich in diesem Ordner befinden werden durch den

Befehl c:mount automatisch nach dem Start der Workbench geladen. So ist es zum Beispiel möglich kleine Tools aber auch Scripte dort abzulegen die von dort aus gestartet werden.

Nun ist die WBStartup dem Autostart von Windows in einigem voraus. Unter anderem darin, die Prioritäten und Wartezyklen selber festzulegen.

Ich bin über diese Möglichkeit gestoßen, auf der Suche nach einer Möglichkeit ein Program im Startup Ordner solange warten zu lassen bis mein USB Stack geladen wurde.

Folgende Befehle können in den Tooltypes (Icon/Tool markieren > Rechte Maustaste > Informationen) eingegeben werden:

DONOTWAIT	Fügt diesen Parameter in jedes WBStartup Programm ein.Anderfalls wird der Amiga warten bis das Programm zu Ende geladen hat, was in manchen Fällen nicht passiert.
STARTPRI	Setzt die Startpriorität fest. Alles von -128 bis 127 kann eingegeben werden. Um so höher die Nummer, umso höher die Priorität. Höhere Priorität bedeutet, das Programm wird zuerst gestartet.
WAIT	Bestimmt, wie lange das System vor dem Ausführen des nächsten Programms warten soll

Um das ganze an einem Beispiel zu verdeutlichen:

Ich habe 2 Programme in meiner WBStartup. 1xclock und 1xbarclock

Nun möchte ich das clock vor der barclock geladen wird und das barclock erst 10 Sekunden nach der clock geladen wird. Also trage ich in den Tooltypes des Programs “clock” folgende 3 Zeilen ein:

DONOTWAIT

STARTPRI=1
WAIT=10

Bei dem Program “barclock” trage ich nur 2 Zeilen ein:

DONOTWAIT
STARTPRI=0

Nun läd das System zuerst das Program “clock” und wartet 10 Sekunden und läd dann das Program “barclock”.

Diese Einstellungen können jederzeit wieder gelöscht werden und das System verwaltet dann den Start wieder automatisch. Es kann aber in bestimmten Fällen wichtig sein es per Hand zu konfigurieren.

Nun viel Spass bei Testen.