# Commodore FREE

**Jack Tramiel, founder of Commodore International and former CEO of Atari International, died on Sunday 8th April at the age of 83.**



computers for
the masses,
not the classes!

Jack Tramiel
1928 – 2012

C= JA                    retrofan

# Ccommodore

# CONTENTS

# EDITORIAL

As you will no doubt be aware Jack passed away recently at the age of 83 (Sunday 8th April 2012) Jack was born in Poland in 1928 into a Jewish family, Jack and his family survived imprisonment in Auschwitz. The family emigrated to the U.S. in 1947 and Jack joined the army and learnt how to repair typewriters.

In 1955 he moved to Toronto and founded a typewriter business called "Commodore Business Machines" it's suggested Jack wanted a military name but Admiral and General were already taken. After various innovations Commodore would move again to Silicon Valley in the 1960's and moved into the digital market selling calculators and then finally moved into the home computer market. Commodore purchased MOS Technologies where the parts for the Commodore range of computers would readily be available and at rock bottom prices. In 1982 Commodore released the Commodore 64 home computer selling almost 17 million units. The Guinness Book of World Records lists the Commodore 64 as the best-selling computer model of all time.

Jack left Commodore in 1984 and purchased the consumer division of Atari. Jack over saw the Atari Jaguar development. Jack was a very keen business man and always drove the prices of machines down. One of Jacks many famous quotes was "Machines for the masses not the classes"

**Interview with Jack**

http://www.commodore.ca/history/people/1989_you_dont_know_jack.htm

**In this issue**

We have the usual news and some more cp/m tutorials we also continue or on off look at programming, and as a memorial to Jack we have my personal look at Commodore and what Commodore gave me as an individual. Of course you will have your own unique stories about Commodore and fights with owners of other, machines about how Commodore is better than XYZ. Heck the debates still go on as to why X machine is better than Y. To me it was Commodore all the way, sadly as jack left the company that direction seemed to falter somewhat, and commodore lost its direction. With Jack at the helm Commodore gave me personally many years of happy computing (and it still does)

Bye Jack
Let's hope you are in silicone heaven.

# Readers Comments

----- Original Message -----
**From:** *Brian Bagnall*
**To:** *nigel parker commodorefree@commodorefree.com*
**Sent:** *Tuesday, 1 May 2012, 21:06*
**Subject:** *COMMODORE FREE magazine comments on Jack's passing*

Hi Nigel,
My comment on Jack's passing:

I only knew him through research on "Commodore: a Company on the Edge" but I felt like I knew him better than some of my own family. He was a complex, driven man who would never let himself be defeated. He changed millions of lives for the better and helped usher in the era of computing to those who didn't have thousands of dollars to spend on IBM or Apple computers--teenagers, families on tight budgets, people on low incomes. Linus Torvald's first computer was a VIC-20, and there wouldn't be nearly as many silicon valley entrepreneurs today if he hadn't planted those seeds years ago. We all owe him a lot for the world we have today.

Best,
*Brian*
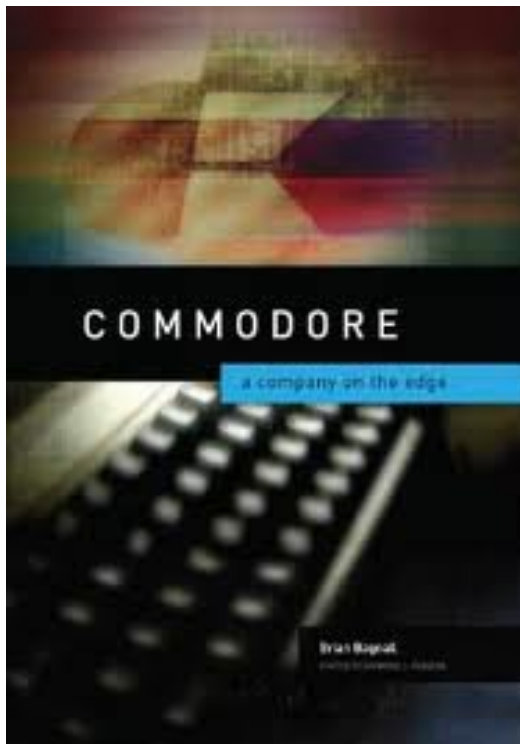


# NEWS

**Automated Amiga Disk Archiver**

OzzyDweller has 5,000 3.5" Amiga floppy disks that he needed to archive, so he built a floppy-disk bulk-importing device. It has a magazine filled with floppies that are auto-inserted into a drive, copied, then ejected out, these are then photographed (so their art can be captured), and then dropped on the floor. He figured out the clever intake and exhaust mechanism by reverse-engineering a bulk floppy writer from eBay.
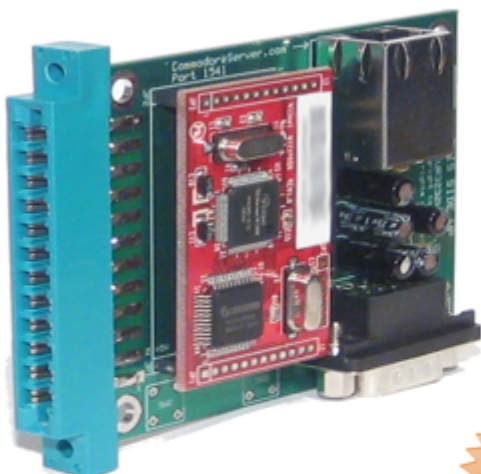
http://boingboing.net/2012/04/02/machine-for-bulk-archiving-tho.html

# NEWS

**Cloanto Releases Amiga Forever and C64 Forever 2012 "R2"**

CLOANTO®

On the April 27, 2012 - Cloanto released the long-awaited "R2" updates of its Amiga Forever 2012 (http://www.amigaforever.com) and C64 Forever 2012 (http://www.c64forever.com) preservation, emulation and support packages for Windows.

The new "R2" versions include hundreds of refinements that took more than eight months to develop, also taking into account user feedback received since the initial 2012 release last August. The new versions feature social and sharing functionality, improved content authoring, pervasive Unicode support and faster loading times.

On the social side, the software allows for manual posts and automatic status updates on Facebook, Twitter, Skype and Messenger. Facebook and Twitter further support screenshot sharing. The "now playing" functionality for Facebook uses the latest Open Graph features to aggregate posts without "spamming" a user's wall. Amiga Forever and C64 Forever were among the very first Facebook clients for Windows approved to use the latest Open Graph and Timeline functionality.

RP9 files can now not only be played back, but also be edited directly from the Windows desktop. This brings more freedom to organize and access content without going through the player interface, and new possibilities like comparing configurations side by side.

New Amiga platform features for content curators include better support for floppy and hard disk images, with media editor options to quickly embed blank disks and system disks into an RP9 package. Features include recognition of RDB disk images and automatic conversion of ADZ, DMS and HDZ images to the more popular and faster ADF and HDF formats (while the RP9 layer takes care of compression).

C64-specific improvements include better keyboard translation, mouse support and the addition of GEOS, the amazing graphical 8-bit desktop suite, to the list of preconfigured systems. Special thanks to Brian Dougherty and his team for this contribution.

By including officially licensed system ROM and OS content, Amiga Forever 2012 and C64 Forever 2012 embody a form of continuity with the original machines sold by Commodore, whose founder Jack Tramiel passed away earlier this month. As it has been doing since 1997, Cloanto keeps working full time on its RetroPlatform family of preservation software, also thanks to its generous supporters and contributor friends. Because of their ease of use, packages like Amiga Forever and C64 Forever have the potential to go beyond preservation and access continuity, by providing inspiration and a well-documented learning environment for future generations.
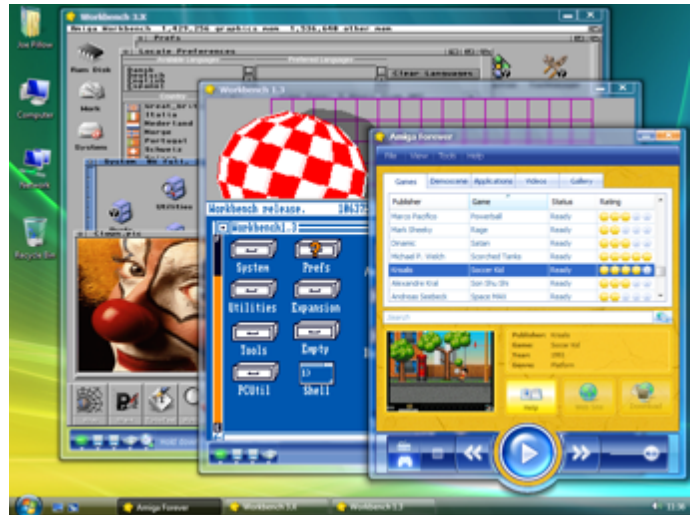
Amiga Forever is available now in three editions:

- Value Edition (downloadable installer for Windows systems)
- Plus Edition (downloadable ISO image with additional Windows and platform-neutral content)
- Premium Edition (physical Plus Edition content plus additional videos on 3 DVDs)

C64 Forever is available now in three editions:
- Free Express Edition (feature-limited version)
- Downloadable Plus Edition (downloadable installer plus CD ISO image)
- Boxed Plus Edition (downloadable Plus Edition content plus boxed CD)

Thanks to a new logistics platform and partners, since December 2011 physical packages ordered from amigaforever.com and c64forever.com have been shipping in almost real time from either California or Germany. This has resulted in reduced delivery times and increased customer satisfaction.

Amiga Forever and C64 Forever passed official Windows 7 logo tests on both x86 and x64 systems, and have also shown to run well on pre-release versions of Windows 8.

Prices for both Amiga Forever and C64 Forever start from $9.95 (special upgrade offers). The "R2" updates are free for existing 2012 version customers (select Check for Updates from the Help menu).

# NEWS

**BOH update15 released**

On the day it turns 3, BOH receives its 15th and final update, which brings even more contents and improvements!

**More specifically, this update:**
* adds 5 missions;
* adds the Dutch translation;
* optimizes even further the rendering engine;
* replaces the transition effect with a cooler one;
* centres the graphics vertically when playing in a screen bigger than the game area;
* allows providing just alphamask files for all the graphics of themes;
* replaces icons with bigger and crisper ones;
* allows to configure the URL for the submissions of records to the online world standings (so that, in future, standings will remain accessible also if the URL changes);
* updates the manuals;
* applies various other minor optimizations/fixes.

The update can be downloaded for free from the DOWNLOADS page of the official website at
www.bohthegame.com
The demos have been updated to reflect the changes.

**About BOH**
BOH is a unique retro-flavoured puzzle-riddled maze game spiced up with thrilling action.
You move in claustrophobic, mysterious, dangerous battlefields searching for the Evil Masters, who throw countless enemies at you until you discover and face them in the final battle. Although your quests are made slightly less hard by the power-ups scattered all around, carrying out the missions demands lots of concentration and quick reflexes.

With BOH the fun never ends: new missions can be added anytime and you can even create your own! And, as if that was not enough, you can also customize entirely its audio-visual aspect!
BOH has been created with lots of passion and care to offer, with its old-school style, a fascinatingly different video gaming experience.

Website: www.bohthegame.com
Facebook: www.facebook.com/pages/BOH/99598418583
Twitter: twitter.com/bohthegame
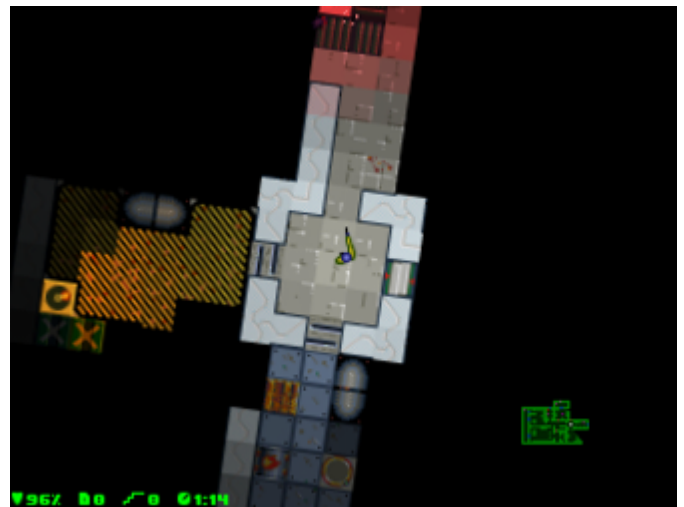


**Lotek64 #40 RELEASED**

Georg Fuchs and his team released issue #40 of the free magazine "Lotek64". The PDF version is absolutely free, but for a small fee you can even get Lotek64 in printed form delivered directly to you.

**Contents**
-Unknown things about the legendary successor of the C64
-Commodore C65
-Commodore 64 1982 - 2012
-The C64 is 30
-The most pointless C64 copy protection of all time?
-Leaderboard Dongle
-Edge Grinder, C64anabalt, Panic Analogue
-Four C64 games reviewed

Download from www.lotek64.com  or www.c64.at.

# NEWS

**QR Code Generator**

This little programme when run will prompt for a line of text to a maximum of 32 characters then convert the text into a QR CODE so you can scan it with a compatible hardware device. I don't have a hardware device to test it with but the code here should say "COMMODORE FREE MAGAZINE" let me know, it may just be a naughty word!

http://noname.c64.org/csdb/release/?id=106999&rss

**UPDATE ON THE P1565 PROJECT**

http://scacom.bplaced.net/Collection/p1565.php

The pictures show 3D-Render of the housing. I tried to reproduce the original housing as accurately as possible. The pictures show the unfinished project and some of the details are missing. Missing also are the LEDs at the top of the housing of the 1565.

The technology of the original 1565 is extensively unknown and only a small, number as known to exist.

**Chillin' with Porter**

**News from Viva Amiga Film**
A little while back I went to Jeff Porter's house and picked up some archival B-roll footage on VHS for Viva Amiga. All kinds of promotional video and some great print materials as well. Jeff was cleaning out his basement and gave me a call to come get it. And while we were there, we played with Jeff's Commodore laptop prototypes and his CDTVs. Check it out!!

http://vivaamigafilm.blogspot.com/2012/02/chillin-with-porter.html

**Galli - Plus/4**
Nino has created a new game for the Commodore Plus/4. Called "Galli" its a side-scrolling platform game. The game was developed between 1989 and 1995, but was never released. The game is now finished and can be downloaded. Because he game is only 16 Kbyte in size it can also be played on the Commodore C16.

**Moldi's Treasure #5**
The web page Plus/4 World has been searching through its large software archive of Moldi files. In this 5th edition there are many new items,

**This is the complete list for the 5th batch, enjoy**
Doki News 1, Doki News 2, Doki News 3, Doki News 4, Doki News 5, TFSS-nibble V7.4, TFSS-packer V8.5, Prosort+ (Hungarian), 2x2 Karakter Keszlet, A4 Printer, No News! 4, Mega-tool, BOB-packer 1.0, 64'er Masterbase Plus/4 (original),

Char-editor 2.0, Mega Writer I, Skoro's Coder V1.2, Sector Editor, Korak Times 9, Lamer News #008, Aircraft, Kikstart Plus, Legicsata (LEH), Teke Jatek, Teke, Big Casino, Aranyalmák, Auto Zone (LEH), Super Boulder (Hungarian), Cartoon 1, Cartoon 2, CAD Box Demo, F-C Trans Plus/4, Italo-mix, Radio Speech, Porno Show 2, The Camera, Picture Show 1, Metallica Demo, Sex Pause 48 Scenes, Demo Designer Info, Flipper (Basic), Flipper 2, Funny Girls (2 versions).

http://plus4world.powweb.com/forum/22554

# NEWS

## AmigaONE X1000 Now Shipping

A-EON Technology http://www.a-eon.com/ is now shipping AmigaONE X1000 systems to a select group of "First Contact" customers. Instead of waiting until AmigaOS 4.2 was completed, A-EON decided to do a limited production run of X1000 systems for the general public. These systems are shipping with a special version of AmigaOS 4.1 Update 5 only for X1000 users.

All "First Contact" customers will be provided with their own unique AmigaOS 4.2 serial number via email. This serial number entitles them to free support via Hyperion's support forum as well as access to any new drivers and updates via the main site.

Given the rushed nature of this AmigaOS 4.1 Update 5 release, there are still some bugs and a few rough edges. Remember, the X1000 was originally planned to be released only with



AmigaOS 4.2 installed. Also please keep in mind most of the X1000 system is still unoptimised. This is truly the most powerful Amiga Operating System hardware platform ever released and we plan to utilize this hardware to its full potential in due course.

The development and beta testing teams we will do their best to help manage any issues as we march towards the final 4.2 product release!

Steven Solie
AmigaOS Development Team Lead

## Return #10 RELEASED

Return magazine is a German print magazine about the 8-bit computers from A (Atari) to Z (ZX Spectrum). In this edition: Boulder Dash, C64anabalt, Commodore 16, ABBUC, Xain'd Sleena, Lt. Kernal Hard drive, Tom Snyder Productions, James Monkman, Racing the Beam, Exploits of Fingers Malone, The universe and 8 Bit, Future Looter (Sinclair Spectrum), Timeline: 35 Years Home-computer.

http://www.return-magazin.de/



## Sqrxz 3

Sqrxz 3 is now available for Amiga OS4 and AROS. Sqrxz and his girlfriend were on a romantic adventure journey in a jungle somewhere here on our planet. All of a sudden Yve gets kidnapped by an evil power and Sqrxz is forced to collect dozens of shiny little rings to free her. The old ruins he has to explore are anything else than safe. Traps are spread all over the place, and Sqrxz is not alone!
http://www.sqrxz.de/



## VFLI - VIC-20

Michael started a project in 2010 to enhance the graphics of the Commodore VIC-20. The new graphics mode requires a hardware modification to the VIC-20. The normal RAM is expanded from 5 kB to 8kB and the new colour RAM is 16 times bigger than the original machine. Recently Michael updated the conversion software and introduced gamma-correction which improves the quality of the converted pictures. You can read all about the VFLI system on the Denial forum.

http://sleepingelephant.com/ipw-web/bulletin/bb/viewtopic.php?t=4882

# NEWS

## PETvet

The PETvet is a RAM and ROM replacement board for the Commodore PET, and potentially other 6502 based microcomputers. The PETvet connects to the 6502 socket in your PET, and allows you to select your boot ROM via jumper settings. It also allows you to view the memory of a running PET using the replacement RAM by halting the CPU and sending the memory contents over a serial port.

http://www.bitfixer.com/bf/PETvet
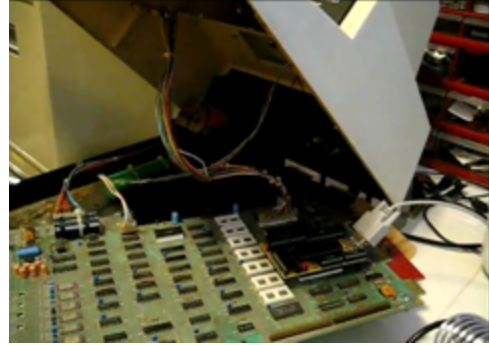


## DiscreetFX Seeking to Add Resources for AmigaOS 4.x & MorphOS Development

AmigaOS 4.x & MorphOS community we have listened to your feedback and are now seeking additional resources for DiscreetFX Labs to develop Aladdin 4D 6.0 for AmigaOS 4.x & MorphOS. You stated it was taking too long and we should not work on the OS X Lion version first so we are talking your feedback to heart. We are looking for one Amiga and MorphOS programmer proficient in C. No naughty Windows, Mac or Linux skills required, your time will be spent 100% working on coding what you love, AmigaOS 4.x & MorphOS!

A past resume on AmiNet, OS4Depot or MorphZone is a plus, preferably in the 3D animation/rendering software department. What do you get in return? Part time steady income from your Amiga development, don't quit your day job. This will be a long term commitment so we are looking for someone that loves to code 3D animation applications that wants to take Aladdin 4D to the next level. We don't need you just for version 6.0 but version 7,8,9 and beyond. Since you make royalties plus regular pay the more you do the more you make. We want someone excited to bite their programming teeth into a large exciting Amiga project. In the future you may have to work with the other developer working on the OS X Lion version of Aladdin 4D 6.0 but this will not be necessary in the beginning. AROS will not be left out in the cold, we will

also need assistance in getting the AROS version coded once the AmigaOS 4.x and MorphOS versions are complete.

Contact us via the link below please with your CV/Resume and reasons why you are right for job. You will also be required to submit your code on a daily or at least weekly bases to a CVS and Blog about your progress to our customers. The Wizard of Oz behind the curtain development cycle of Aladdin 4D 6.0 has not pleased us either and we want future development to be in the public eye so customers see rapid progress.

http://www.discreetfx.com/contact.html

Best regards

DiscreetFX Team



## HVTC High voltage TED Collection

The HVTC or High Voltage TED Collection is an collection of music for the Commodore C16, 116 and the Plus/4. The files that are available, are the original programs that produce the music with the TED music chip. At the moment there are 200 files available in the database. A few examples are: Auf Wiedershen Monty, Karate+, Lunar Blitz, Melon v1, Rainbow Trout, Emerald Mine 2, Summer events, Bandits at Zero, Power Ball, etc.

http://plus4world.powweb.com/features/High_Voltage_TED_Collection

# NEWS

**Scene World #18 is OUT NOW!**

Hello everybody!
Today we release Scene World #18! (Download at:
http://noname.c64.org/csdb/release/?id=106679 )

We are very proud of this release because for the first time we got 2 Skype Video interviews as well
(echeck on the issue's page right on our homepage):

http://www.sceneworld.c64.org

We have got ex-phreaker and legend Captain Crunch (John Draper) and ex-Commodore right-hand of Jack Tramiel - Michael Tomczyk to speak about the stuff they had their hands in!

(Of course you can read full transcriptions inside the magazine itself, so lean back and enjoy!).

Please also send us articles, logos, disk covers and exclusive tunes, we need them, thank you! :)
Nafcom
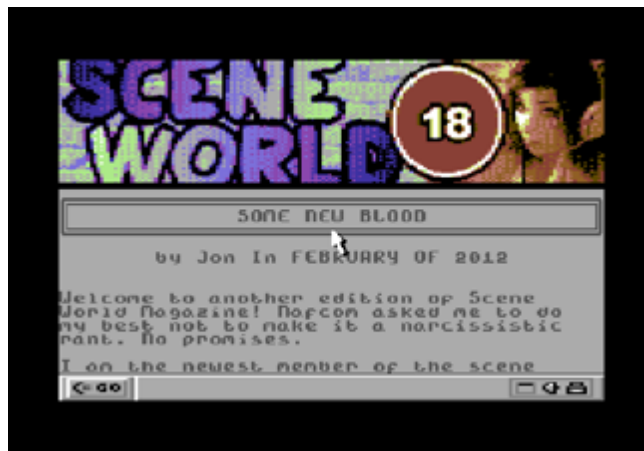


**Aladdin 4D New Name / New Ownership**

http://www.discreetfx.com/Aladeen4D.html

Aladdin 4D has been renamed Aladeen 4D

We were conquered and had no choice!

On February 26th, 2012 DiscreetFX surrendered all inventory, source code and intellectual property rights to his Holiness and Excellency Admiral General Aladeen. Aladeen 4D is one of his many conquests and he will be updating and releasing a new 6.0 version for OS X Lion, AmigaOS 4.x, MorphOS 2.x, AROS, and Linux. Admiral General Aladeen said the Windows version of Aladdin 4D is cancelled because Megan Fox does not like that OS. We hope it will not take another 1000 years for version 6.0 to be finished!

Oscars 48 Hour Sale!
http://www.discreetfx.com/Aladeen4D.html

When the Oscar's Start You Are Allowed to Buy it!

Unbest regards

DiscreetFX Team
(Captured and being held prisoner in the Mighty Nation of Wadiya)



**PC WORLD REVIEW WINUAE**

http://www.pcworld.com/downloads/file/fid,201995-order,4/description.html
PC World have reviewed the WINUAE software allowing user to run Amiga software on their PC systems

QUOTE from the site
*Also, acquainting (or reacquainting) yourself with the ins and outs of an old computer system can take a little time, even if you're no stranger to the platform: be prepared to take a few steps up the learning curve. For the Amiga faithful, however, none of this matters. WinUAE is nothing less than a time*

*machine back to the golden age of computing. It's a ride well worth taking.*
*--Jim Norris*
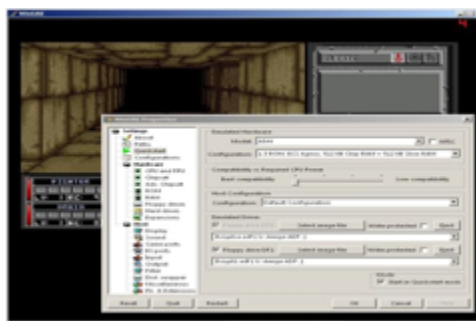
# NEWS

**Cinnamon Writer v. 0.80 is ready for Download**

**News from Desler**
http://desler.be/modules/wfchannel/

It is a pleasure to announce that version 0.80 of Cinnamon Writer is available for download at desler.be or at OS4Depot.net

**Get the new version here**
Cinnamon Writer is a compact and efficient word processor developed especially with Amiga and Amiga-like systems in mind. CW have all of the basic functionality expected of a modern word processor. This includes changeable layout style, support of anti-aliased true type fonts, infinite "intelligent" Undo/Redo history, user defined page setup and file compatibility with other word processors as Word and OpenOffice that can use RTF or DocX files.

**New in version 0.80:**
CW now supports spell checking while you type: I have been able to port a small and very efficient spell checker and integrate it into CW. As a result, you now have the possibility to have your document checked for spelling mistakes while you are working with it. You are able to work with as many different languages within the same document as you wish. When a miss-spelled word is detected it is highlighted and, if possible, suggestions for replacement are given. Please read the supplied manual for usage.

English and German dictionaries are supplied as standard with CW 0.80 and dictionaries of other languages will be made available from the download section of Desler.be in the near future. If you have access to a comprehensive list of words within your language of choice, please send me the list or a link to it at projects@desler.be. I will build a dictionary file and make it available as fast as possible. However, please make sure that the licensing of the file allows distribution.

If you are a software developer who would like to add spell-checking abilities to your program, feel free to contact me for the source codes for CW spell-checker. You are also welcome to use the CW dictionary files for your own projects as long as you abide to the licenses that may follow a dictionary. Please contact me at projects@desler.be.

Better integration with AOS: The focus of CW has so far to implement functionality and not necessarily implement the typical Amiga feel that people would expect from a project like this. As the feature list is now becoming more and more comprehensive I will start integrating the small things that make the program more "Amiga" like and ease the daily use.

Implemented in this version are a comprehensive collection of short-cut keys and the introduction of a right-click menu. This work will continue in future versions of CW

**Known Issues / bugs:**
The RTF file format is not being developed further. With the introduction of DOCX support I have decided to out phase RTF support. It is simply too much work to support different file formats and DOCX is the de facto standard within most areas. CW also supports load/save of unformatted text

The first time the spell checker needs to spell check a word in a specific language, it will have to load the corresponding dictionary. As a result there will be a small lag until the file has been loaded. This lag is dependent on the size of the dictionary and the speed of your hard drive / system. This is not a bug. If you find the spell checking to be slowing down CW, you can turn it off. CW is quite usable on my G4 @ 800. But I do not know how it runs on lower specced machines

Printer / PDF functionality of CW 0.80 has not been improved since the prior version. Any inconsistencies therefore still exist. This will be improved in a future version

A lot of work has been put into this version, however bugs may still persist. If you find a bug and it is not described in "Known problems" in the manual, please submit it to me at bugs@desler.be. You can also use the new bug tracker, which you can find here

Visit the homepage for more information here

**Donations:**
If you like this product and would like to see its development continue, please consider a donation
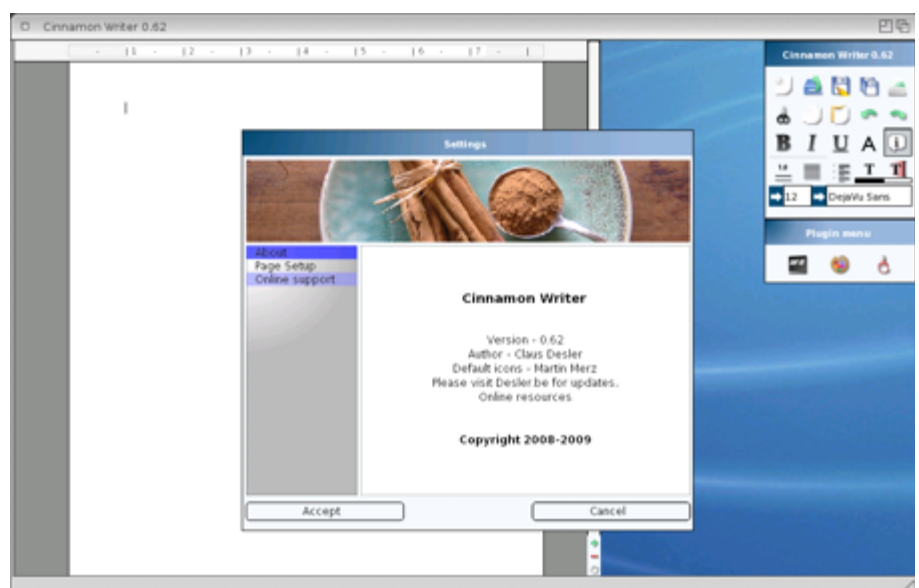
# NEWS

**"Boing Ball" re-visited - part 2**

**TrevorDick**

Following the recent release of the RadeonHD graphics driver for AmigaOS 4.1 Update 5, which made its official debut on the AmigaONE X1000 "First Contact" system, we are pleased to reveal that we have now concluded a further agreement with Hans de Ruiter for the continued development of his RadeonHD drivers for the AmigaOS. In a statement De Ruiter commented, "I am currently working to get the driver fully ready for 3D, and to release quality. Thanks to the support of Trevor Dickinson (A-EON Technology), I will be able to devote more time to RadeonHD driver development over the next few months, and achieve this goal much sooner than would otherwise have been the case. The features being worked on include: Vertical blanking interrupt support for RadeonHD cards (no more wasting CPU cycles when double-buffering); Video RAM (VRAM) management for the RadeonHD_RM.resource (intelligent VRAM buffer paging)." De Ruiter added, "The last feature is essential for stable 3D operation. I look forward to delivering RadeonHD.chip version 1.x and, of course, to be able to run modern 3D applications using RadeonHD cards and AmigaOS." Dickinson said in reply, "I am really pleased to help fund Hans' RadeonHD driver work. This is excellent news for all AmigaONE X1000 owners, including myself ;-) and hopefully should benefit all AmigaOS 4 users in the future."

About Hans de Ruiter & HDRLab: Hans de Ruiter is an electrical & electronic engineer and leading AmigaOS developer and Amiga enthusiast. His website - HDRLab - details various projects, many of which are AmigaOS related. The biggest of these is the RadeonHD driver for AmigaOS, an ambitious project that aims to deliver modern graphics capabilities for the AmigaOS platform.

About AmigaOS: The AmigaOS dates back to 1985 with the introduction of the landmark Amiga 1000. Today, under the guidance and control of Hyperion Entertainment, AmigaOS has evolved into a modern operating system without losing that intrinsic "Amiga" look and feel.

About the AmigaONE X1000: The AmigaONE X1000 is not like other computers. It is a culmination of efforts by real Amiga enthusiasts and developers to create powerful, modern desktop hardware for the AmigaOS. It is the natural evolution of the Amiga's PowerPC lineage and is based on the PA-Semi Dual-core PA6T-1682M CPU and includes Xena, a "Software Defined Silicon" co-processor. Above all it runs the latest version of the AmigaOS.

HDR Labs - http://hdrlab.org.nz/projects/amiga-...deonhd-driver/

A-EON Technology - http://www.a-eon.com

AmigaOS http://www.amigaos.net/

A-EON Technology

# NEWS
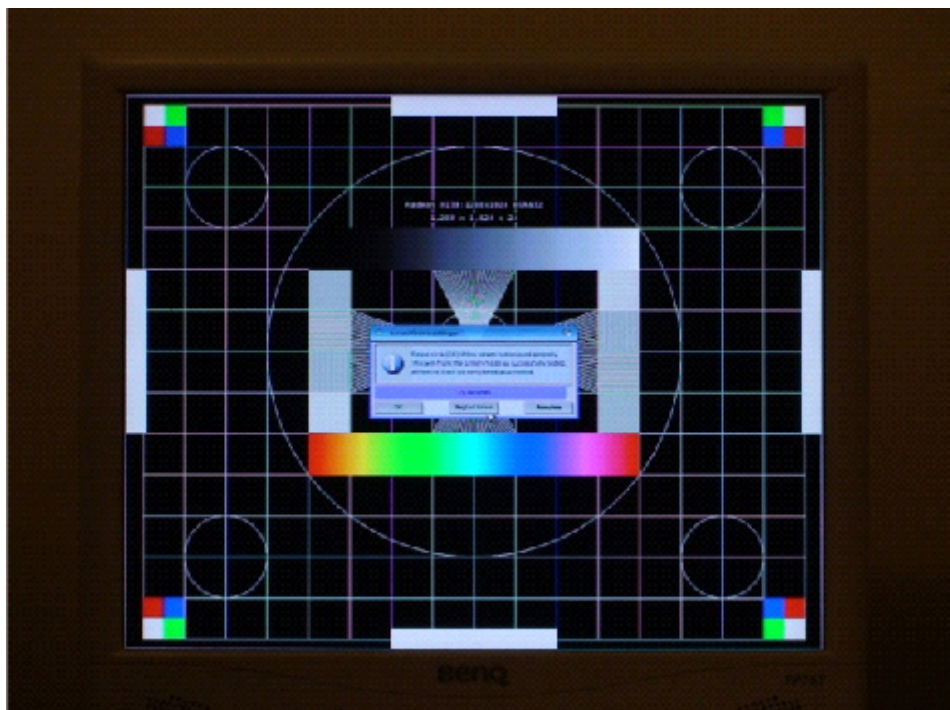
**EMULATOR**

http://fengestad.no/fs-uae/
**FS-UAE 0.9.13beta5 README**

FS-UAE is a multi-platform Amiga emulator for Windows, Linux and Mac OS X based on WinUAE/E-UAE/PUAE, with a focus on emulating floppy-disk and CD-ROM based games.

FS-UAE is fully controllable with a gamepad, with an on-screen GUI, which means that you can easily swap floppies and load save states with your gamepad. Because of this, FS-UAE is well suited to be started from an emulator frontend running on a HTPC, but will work equally great on your personal computer. The emulator uses the cycle-exact emulation code from WinUAE and requires a moderately fast computer with accelerated graphics (OpenGL) to work.

The official home page of FS-UAE is:
http://fengestad.no/fs-uae/
Thank you for your interest in FS-UAE :)

**IMPORTANT INFORMATION**
FS-UAE does not include a configuration UI, so you must create a configuration file before the emulator will be usable. This is fairly easy, please read on for more information.

Please report any errors found in the documentation to fs-uae@fengestad.no.

**CONFIGURATION FILE**
You will also need to create a configuration file for FS-UAE. The important things you need to configure is what floppy disks to use, and where the Kickstart ROMs are stored. Documentation (and examples) is available here:
http://fengestad.no/wp/fs-uae/documentation

You have several options for where to store the configuration file:

1. In the same directory as the FS-UAE executable, as Config.fs-uae. FS-UAE will always try to load this file first.

2. As a per-user configuration file: (My) Documents/FS-UAE/Configurations/Default.fs-uae

3. You can also store the file anywhere you want, and invoke FS-UAE with the config file as a parameter (fs-uae /path/to/config-file.fs-uae)

4. In graphical shells (Windows Explorer, Mac OS Finder), you can also select a configuration file, and then choose to open it in FS-UAE (you may need to browse to find the program), or you may

also drag and drop a configuration file in top of the FS-UAE application/executable.

(you can see in the log file exactly what path FS-UAE tries to load the default configuration from on your system)

**KICKSTART**
You will need Kickstart ROMs for the Amigas you want to emulate. One source for this is your real Amiga, but you can also purchase the Kickstart ROMs in file format from Cloanto - they are bundled in their Amiga Forever-product.

**JOYSTICKS**
If you have a recognized joystick connected, FS-UAE will automatically try to use this as the primary Amiga joystick. Otherwise, keyboard joystick emulation will be enabled, where the cursor keys and the right control (right alt on Mac) on your keyboard controls the joystick.

FS-UAE does not (currently) support two keyboard emulated joysticks. If you want to use two Amiga joysticks, you need to have at least one real joystick or gamepad. You can specify which joystick to use for each Amiga joystick port. Please see example.conf for more information. Information about supported controllers are here: http://fengestad.no/wp/fs-uae/controllers

If your joystick is not recognized by FS-UAE (you can see this in the log file), you can create a configuration file for it. Seehttp://fengestad.no/wp/fs-uae/custom-controller-configuration If you create a configuration file, I would appreciate it if you would send a copy to fs-uae@fengestad.no with the producer and model name of the joystick/gamepad (for inclusion in future versions of FS-UAE).

**EMULATOR MENU**
While running the emulator, you press F12 to enter and exit the emulator menu. From here, you can load and save states,

and swap floppy disks. On Mac OS X, you can use ALT + F12 instead (Possibly even FN + ALT + F12).

With gamepads, you enter the menu either by using the dedicated "menu" button, if the gamepad has one, or you press and hold "start" and "selected" at the same time (or equivalent buttons).

You use the same key/button to dismiss the menu.

In the menu, you use the following keys for navigation:
Cursor keys
Enter (choose item, enter sub-menu)
Back-space (leave sub-menu)

On the game pad, you can use either the hat or the primary analogue stick for navigation. You choose items and enter sub-menus with the "primary" button on the game pad. This is generally the "south" button on the right side of the controller. The back button is the "east" button (Just like you would navigate on the Xbox).

## SCREENSHOTS
Press the "Print Scrn" key on the keyboard to save a screenshot of the running game to the desktop.

## KEYBOARD SHORTCUTS
F12 - Toggle menu (On Mac: ALT + F12 or FN + ALT + F12)
F11 - Ungrab/grab mouse pointer (On Mac: ALT + F11 or FN + ALT + F11)

CTRL + F10 - Toggle audio/video performance overlay
CTRL + F11 - Toggle debug crop rectangle for auto scale algorithm

## PROGRAM ARGUMENTS
&lt;path&gt;          Use a custom configuration file
--video-sync=auto    Automatically enable appropriate sync mode
--video-sync=off    Do not enable vsync
--video-sync=vblank  Will sync rendering to vblank interval (avoid tearing)
--video-sync=full    Sync emulation to vblank interval
--fullscreen=1      Start in fullscreen mode instead of windowed mode
--stdout=1        Also log to stdout (not on Windows)

With recent version of FS-UAE, you can override any configuration option with --key=value (e.g. --floppy-drive-0=/path/to/adf)

## LOG FILE AND REPORTING PROBLEMS
A log file is stored as (My) Documents/FS-UAE/Logs/FS-UAE.log

When reporting a problem, you should include this log file with you error report.

## FLOPPY DISKS
New data is always saved to overlay ADF files (.sdf). As an example, if you have floppy.adf insert into df0 and the Amiga writes to df0, the data is written to floppy.sdf instead of modifying floppy.adf itself. You need both floppy.adf and floppy.sdf to represent the entire modified disk. The save files are created on demand (when data is written to a drive).

The floppy file overlays are by default save in the directory (My) Documents/FS-UAE/Floppy Overlays

The GUI does not allow you to browse for files on the file system. Disks can be inserted at startup time (specified in the configuration file), and FS-UAE maintains a list of available floppy images to insert (also specified in the configuration file). See example.conf for more information.

## SUPPORTED AMIGA MODELS
The following Amiga models are supported:
- Amiga 500
- Amiga 500+
- Amiga 600
- Amiga 1200
- Amiga 1200/020
- Amiga CD32
- Commodore CDTV

The default configurations corresponds to the quickstarts in WinUAE for these systems, on the most compatible setting (cycle-exact mode).

## DISPLAY REFRESH RATE AND RESOLUTION
For really smooth scrolling in games, your display MUST run at the same refresh rate as the Amiga: 50Hz (PAL). Many monitors and TVs with 1920x1080 resolution supports this refresh rate. If FS-UAE detects that you display is running at 50Hz, vsync will automatically be enabled for perfect scrolling.

FS-UAE will (not yet) on its own change your refresh rate to 50Hz –you must do this yourself before starting FS-UAE.

If you run your display with another refresh rate, the emulator will still work just fine, but scrolling will be somewhat jerky -it varies with games whether this is very noticeable or not.

Note: On Mac OS X, FS-UAE does not currently detect the refresh rate, which means that full vsync will not be enabled. If
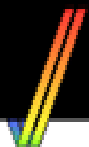
you are running at 50Hz, you can force vsync with --video-sync=full

FS-UAE will use whatever full-screen resolution you desktop is using, and will assume that you have square pixels. The Amiga image will be scaled to fit the display, and you can choose whether to stretch to fill the entire screen, or scale up while maintaining the original aspect ratio.

### SUPPORTED HOST OPERATING SYSTEM
- Linux (Other Unices will probably also work just fine)
- Windows
- Mac OS X 10.6 or newer

### DEBUGGING SUPPORT
FS-UAE does not include a debugger for debugging Amiga programs. For debugging FS-UAE itself, a native debugger can be used if you compile FS-UAE yourself. Other debug information can be retrieved from the log file.

There is debug overlay to debug video, audio and vsync issues. You can toggle the display of this with CTRL+F10.

FS-UAE automatically crops the Amiga video output. Sometimes this does not work perfectly. You can override this behaviour in the configuration file. Also, you can press CTRL+F11 when running to visually see the entire Amiga output with the crop rectangle.

### KNOWN ISSUES / TODO
CD32 gamepad (emulated by keyboard)

Amiga mouse must currently be emulated with a native mouse device. A nice feature would be to be able to emulate the mouse with gamepads – analogue controls for instance.

An on-screen keyboard, controllable with a game pad / joystick would also be nice, and is planned for a future version.

Picasso 96 is not supported.

Floppy list UI does not support scrolling, so if you have many floppy images in the floppy list, some may be obscured.

The CD32 does not always reset properly when choosing RESET from the
on-screen menu.

### COPYRIGHT AND CREDITS
FS-UAE is Copyright (c) 2011-2012, Frode Solheim Large portions are copyrighted by other individuals.

FS-UAE is based on the fantastic work of the original UAE authors, the authors of WinUAE, E-UAE, P-UAE, and contributions from external contributors.

UAE was created by Bernd Schmidt, with the help of a host of volunteer contributors (see the UAE distribution for full credits). WinUAE has been developed and maintained by Mathias Ortmann and Toni Wilen.

E-UAE was developed by Richard Drummond, and P-UAE is developed by
Mustafa 'GnoStiC' TUFAN.

http://www.amigaemulator.org/
http://www.winuae.net/
http://www.rcdrummond.net/uae/
https://github.com/GnoStiC/PUAE

libfsemu is Copyright (c) 2011-2012, Frode Solheim

IPF decoder library is Copyright (c) 2001-2011 by István Fábián with additional work by Christian Sauer.

# NEWS

**AmigaOS Beta Testers**
**Hyperion Entertainment**
FROM Steven Solie
AmigaOS beta testers are not often acknowledged for the important work they do.

A fresh list of AmigaOS beta testers has now been posted on the AmigaOS web site. http://www.amigaos.net/

This list is not 100% complete in that it only includes those beta testers that worked on version 4.0 of the Amiga's operating system and up.

Beta testers are not paid for their work and they are expected to pay for their own hardware as well as software licenses.

Beta testing is truly a labour of love so hats off to all the volunteers.

Again, special thanks to Olaf Barthel who is the keeper of this kind of historical information.

**Steven Solie**
AmigaOS Development Team Lead

**Credits**
The following is the list of developers who have worked directly on the Amiga Operating System (AmigaOS) over the years:

| | | | | |
|---|---|---|---|---|
| Adam Kowalczyk | Alexander Kneer | Alexander Lohrmann | Allan Havemose | Álmos Rajnai |
| Andre Dörffler | Andrea Vallinotto | Andrija Antonijevic | Andy Finkel | Barry Walshand |
| Bart Whitebook | Bill Toner | Bob Burns | Bob Pariseau | Brian Jackson |
| Bruce Barrett | Bruce Drake | Bryce Nesbitt | Carl Sassenrath | Carolyn Scheppner |
| Charles Warwick | Chris Green | Chris Morris | Christopher Aldi | Christopher Kossa |
| Colin Wenzel | Costel Mincea | Craig Makarowski | Csaba Simon | Dale Larson |
| | | | | |
| Dale Luck | Dan Baker | Darren Greenwald | Dave Berezowski | Dave Needle |
| David Haynie | David Joiner | David Junod | David Rey | Davy Wentzler |
| Deryk Robosson | Detlef Würkner | Doug Walker | Eike M. Lang | Eric Cotton |
| Frank Wille | Greg Miller | Gudrun Haage | Hans de Ruiter | Hans-Jörg Frieden |
| Kevin Klop | Leo Schwab | Leonard Poma | Markus Nerding | Markus Pöllmann |
| Martin Hunt | Martin McKenzie | Martin Merz | Martin Steigerwald | Martin Taillefer |
| | | | | |
| Mary-Ellen Toebes | Massimiliano Tretene | Massimo Tantignone | Matthew Kille | Michael Sinz |
| Mike Whitcher | Neil Katin | Olaf Barthel | Oliver Roberts | Palmyra Pawlik |
| Paul Heams | Paul Schifferer | Peter Annuss | Peter Cherna | Peter Gordon |
| Hartwig Haage | Hedley Davis | Heinz Wrobel | Henning Nielsen Lund | Henrik Isaksson |
| Ignatios Souvatzis | Jay Denebeim | Jay Miner | Jens Langner | Jerry Horanoff |
| Jim Barkley | Jim Cooper | Jim Mackraz | Joanne Dow | Jochen Becher |
| | | | | |
| Joe Pillow | John Szucs | John Toebes | Jon Prince | Jörg Strohmayer |
| Joshua B. Helm | Jürgen Haage | Kaori Kuwata | Karl Churchill | Kenneth Dyke |
| Philippe-Andre Bourdin | Phillippe Ferrucci | Randell Jesup | Raymond S. Brand | René W. Olsen |
| Richard Drummond | Robert "RJ" Mical | Ross Vumbaca | Rudolph Riedel | Sam Dicker |
| Sam Jordan | Scott Wegener | Sebastian Bauer | Sebastian Reinfeldt | Simon Archer |
| Spencer Shanson | Stan Shepard | Stefan Burström | Stefan Falke | Stefan Robl |
| | | | | |
| Steffen Häuser | Stephan Rupprecht | Stéphane Guillard | Stephen Fellner | Steve Beats |
| Steven Solie | Stuart Ferguson | Sven Ottemann | Thomas Frieden | Thomas Graff Thøger |
| Thomas Richter | Thore Böckelmann | Tim King | Tobias Abt | Tony Wyatt |
| Tracy D. McSheery | Valentin Pepelea | William H. M. Parker | William S. Hawes | |

**The following is the list of beta testers who have helped test AmigaOS since version 4.0:**

| | | | | |
|---|---|---|---|---|
| Achim Stegemann | Adam Kowalczyk | Al Stremming | Alan Buxey | Albert Jasinski |
| Aleksander Czarnowski | Alen Hadzihasanovic | Alex Carmona | Alexander Lohrmann | Alexander Weber |
| Alexandre Balaban | Alfredo Amendolagine | Álmos Rajnai | Andre Dörffler | Andrea Favini |
| Andrea Monni | Andrea Palmatè | Andrea Vallinotto | Andreas Loong | Andreas Weyrauch |
| Andrew Boyarintsev | Andrija Antonijevic | Andy Broad | Anthony Bandiera | Antti Korhola |
| Arkadiusz Hucko | Arturino Mazzei | Bill Borsari | Bill Eaves | Bill Toner |
| Björn Hagström | Borut Zivkovic | Carl Moppett | Carles Bernardez | Carlos Orois |
| Csaba Simon | Charles F. Warwick | Chris Dallimore | Chris Handley | Chris Morris |
| Christoph Gutjahr | Christopher Follett | Christopher Kossa | Colin Ward | Colin Wenzel |
| Costel Mincea | Damien Stewart | Damir Arh | Damir Dardagan | Daniel Allsopp |
| Dariusz Wisniewski | Darren Eveland | Darren Glenn | Darren Stevens | Dave Fisher |
| | | | | |
| Dave W. Pitcher | David Braconnier | David Burström | David J. Aiau | David Pitcher |
| David Rey | Davy Wentzler | Denis Troller | Dennis Zweedijk | Deron Kazmaier |
| Detlef Würkner | Dirk Baeyens | Dirk Stoecker | Doug Moir | Douglas McLaughlin |
| Dwayne Jarvis | Edgar Schwan | Emmanuel Rey | Enrico Vidale | Fabian Kreitner |
| Felix Schwarz | Ferran Garcia | Francis Labrie | Frank Wille | Frans van Egmond |
| Fredrik Nilsson | Gabriele Favrin | Gary Herring | Gaspar Marton Imre | George Sokianos |
| Gerd Frank | Gregory S. Donner | Guido Jonk | Guvenc Kaplan | Hans de Ruiter |
| Hans-Jörg Frieden | Heinz Wrobel | Henning Nielsen Lund | Henrik Jacobsson | Herve Dupont |
| Ignatios Souvatzis | Jacques Vanhove | Jaime Cagigal Bordonaba | James Carroll | Jamie Krueger |
| Janez Starc | Janne Peräaho | Javier de las Rivas | Jean-Francois Bachelet | Jens Langner |
| Jerome Senay | Joakim Nordström | Joanne B. Dow | Jochen Kuse | |
| Jens Dueholm Christensen | | | | |
| | | | | |
| Jody J. Tierney | Jörg Strohmayer | Johan Banis | Johan Engdahl | Johan Samuelsson |
| Johannes Geiss | John Scolieri | Joni Halme | Jose Manuel Menendez | Joshua B. Helm |
| Jürgen Schober | Jürgen Weiß | Juha Niemimäki | Julien Cervellera | Justin Hemmings |
| Justin Raug Veggerby | Kamil Niescioruk | Karl Churchill | Karsten Fischer | Karsten Limberg |
| Ken Wilde | Kendrick Hughes | Konrad Bielski | Kyle Caldock | Kyle Kivi |
| Kymon Zonias | Larry Keller | Lars Nelson | Louie Dituri | Lyle Hazelwood |
| Marc Jouault | Marco De Vitis | Marcus Comstedt | Mario Cattaneo | Mariusz Danilewicz |
| Marko Mihailovic | Martin Lafunte | Martin McKenzie | Martin Steigerwald | Martin Wolf |
| Marton Dosa | Massimiliano Tretene | Massimo Tantignone | Mathias Roslund | Matthew Kille |
| Matthew Leaman | Matthijs Jansen | Maurizio Lotauro | Michael Boehmer | Michael Carrillo |
| Michael Christoph | Michael Malyshev | Michael Merkel | Michel De Meerleer | Mick Sutton |
| | | | | |
| Mike Beecham | Ned Kelly | Nick Clover | Nicolas Mendoza | Niels Bache |
| Nikolaos Damilakis | Ole-Egil Hvitmyren | Oleg Sergeev | Oliver Roberts | Olrick Lefebvre |
| Ondrej Zima | Pål Ringkilen | Paolo Suriano | Pascal Rullier | Paul Heams |
| Paul Sadlik | Paul Sabourenkov | Pavel Svetlik | Pawel Stypula | Peter Gordon |
| Peter Jensen | Petrus Bergman | Philip Young | Philippe Bourdin | Philippe Ferrucci |
| Philippe Rouxhet | Piotr Zadora | Rafal Chyla | Ralf Gruner | Ralf Toenjes |
| Ralph Holzer | Raul Silva | Remco Komduur | René Thol | René W. Olsen |
| Ricco Clemens | Richard Drummond | Robert Kihl | Robert Mattin | Robert Mustin |
| Robert T. Dickinson | Robert Wahnsiedler | Robert Williams | Ronald Dean Hallman, Jr | Ronald Teune |
| Ross Vumbaca | Rudolph Riedel | Russ Norrby | Ryan C. Gordon | Sven Ottemann |
| Sascha Hoogen | Scott Wegener | Sebastian Bauer | Shane Ponting | Simon Archer |
| | | | | |
| Simon Neumann | Simon Tyrrell | Sinan Gurkan | Steen Lund Nielsen | Stefan Falke |
| Stefan Juhas | Stefan Kost | Stefan Nordlander | Stefan Robl | Stefano Guidetti |
| Steffen Häuser | Stephan Rupprecht | Stephan Scheele | Stéphane Guillard | Stephen Fellner |
| Stephen Jones | Stephen Robinson | Stephen Umney | Steve Bowman | Steve Harrison |
| Steven Fuller | Steven Solie | Stuart C. Paterson | Sven K. Harvey | Sylvio Kurze |
| Tejinder Brar | Terry Palfrey | Thomas Blatt | Thomas Frieden | Thomas Graff Thøger |
| Thomas Palestig | Thomas Richter | Thomas Wenzel | Thomas Würgler | Thore Böckelmann |
| Thorsten Freund | Timothy De Groote | Tobias Franke | Tom Crecelius | Tony Aksnes |
| Tony Wyatt | Tore B. Bjørnsen | Torfinn Ingolfsen | Tuomas Hokka | Uros Bogataj |
| Val Marti | Vasileios Kostopoulos | Vicente Gimeno | Viktor Soponyai | Vit Sindlar |
| Warren T. Katchmar | Wilfred Luesebrink | Xavier Bal | Yurdaer Donmez | |

# Commodore Free review Galli

Title: Galli
Category: Game/Platformer
Release Date: 2012-02-22
Language: English
Size: 16K
Machine: PAL Only
Code Type: Machine code
Distribution: Freeware
Game Ending Type: Has an end, game ends
Notes: Created by Antonino Scala. The main character has been named "Gallina", that means "chicken" in Italian.



http://plus4world.powweb.com/software/Galli
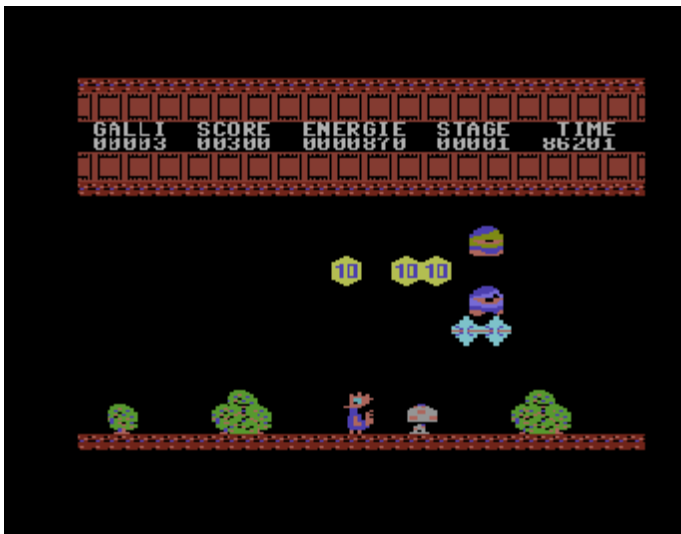
The game loads and they typing run show the screen GAME over, pressing the fire button takes you to the games start, however I don't seem to be able to find any instructions and as you will read I became confused as to what I was supposed to actually do.

The game looks rather basic and when jumping out character seems to soar into the air as if  he was lighter than something incredibly light indeed, maybe because he is a bird he can fly a little who knows, This gravity or lack of it can come in handy though as you will be able to jump to amazing heights to collect treasures that would normally be impossible to reach.

Here is a YouTube taster
http://www.youtube.com/watch?v=-12YtS6DUSw

I did see this line in a forum
*How do you play the game? Collect all diamonds and coins and don't touch the aliens.*

So that must be where I was going wrong I need to collect everything before I can move on so let's try again, with a joystick plugged into port 1 and loading the game then running and pressing Fire at the game over we are on the main screen.

Moving is just a case of pressing the joystick ion the right direction, as you move to the right the screen scrolls with you can move back but only to the screen edge the screen will no scroll back.
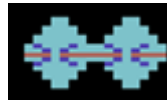
You control this guy



And need to avoid these aliens



Collect these



And these treasure items



Not too bad a game really, it's a bit glitchy but that could be my machine I only have 1 machine working (of sorts) to test the game on at the moment.

I seemed to have missed the games design, as I thought I had to collect all the treasure and the 10 coins were just a bonus but it seems to advance to the next level you have to collect everything possible, or if you pass something and it scrolls off screen you then have to replay the level. Sounds are minimal although the game was designed to play with a SID card the code has been removed. Animation is quite crude, it's a decent couple of hours play but not something you will continue to go back to over and over again.

| | |
|---|---|
| **Game play** | 4/10 |
| **Graphics** | 4/10 |
| **Sounds** | 3/10 |
| **Longevity** | 3/10 |
| **Overall** | 4/10 |

# AMIGA FOREVER AND COMMODORE 64 FOREVER

**Amiga Forever**
http://www.amigaforever.com
http://www.facebook.com/AmigaForever

**C64 Forever**
http://www.c64forever.com
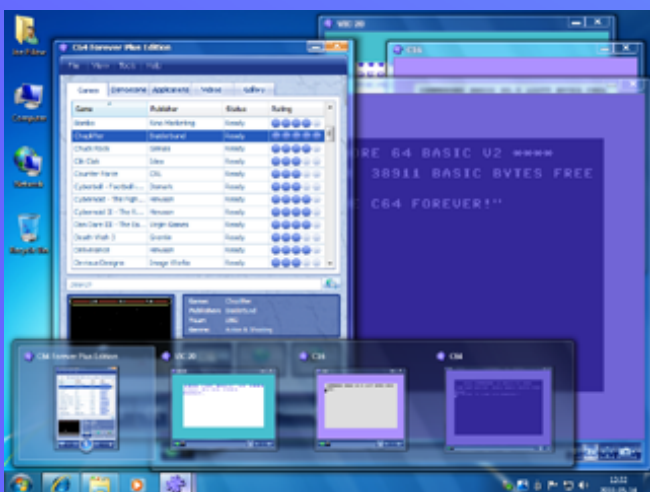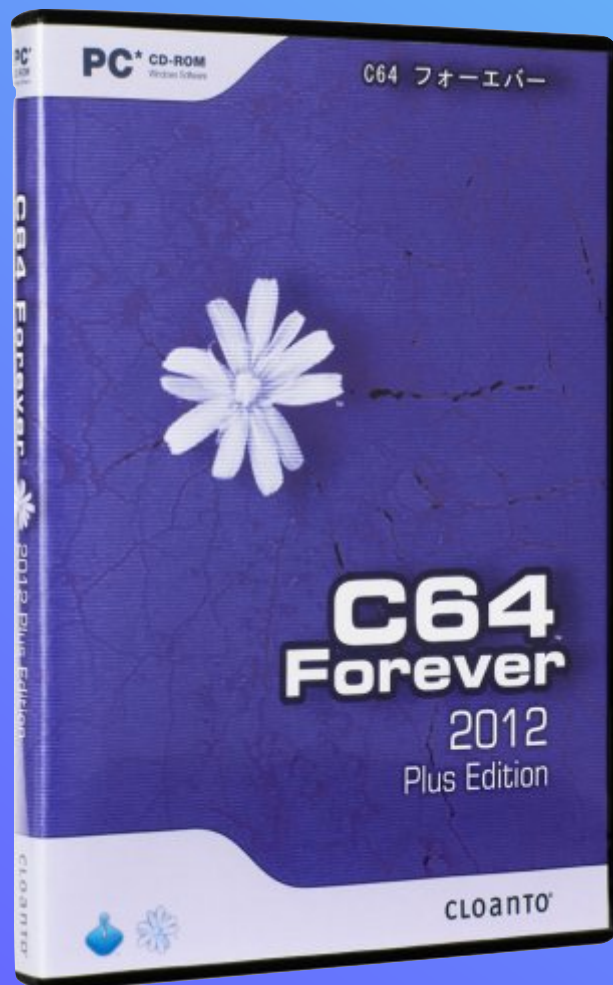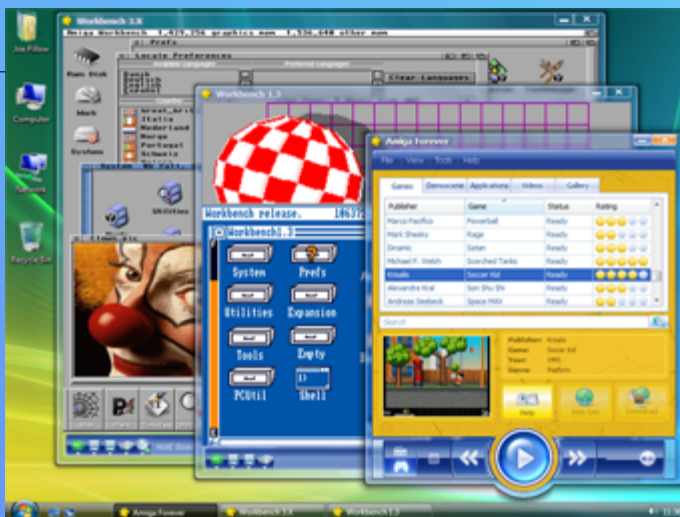http://www.facebook.com/C64Forever

**RetroPlatform Project**
http://www.retroplatform.com

# Rediscovering CP/M

## Part 3

### By Commodoreman ©

**PART 3**

Before beginning with this third part, I would like to refer you to the summary provided at the end of Part 2. This should bring everyone up to speed on where I am and will set the stage for this part. I have also included an updated Checklist of things to do and a List of Resources available to me. I have links to digital files that I can read on my other computers, but I prefer to flip through the actual paper books. I'm not intimidated by large manuals.

I took a directory of the CP/M System Disk in drive A by typing **DIR <RETURN>**. Here is what was on each side (the disk provided with the 128 was formatted so a 1541 could access it)...

Directory of CP/M System Disk:

**Side 1**
CPM+.SYS  CCP.COM  KEYFIG.HLP  FORMAT.COM  HELP.HLP  DIR.COM  HELP.COM  PIP.COM  KEYFIG.COM  COPYSIS.COM

**Side 2**
DATE.COM   DIRLBL.RSX  GET.COM   RENAME.COM  SUBMIT.COM  DATEC.ASM  DUMP.COM  INITDIR.COM  SAVE.COM   TYPE.COM  DATEC.RSX  ED.COM  PATCH.COM  SET.COM   DEVICE.COM  PIP.COM  SETDEF.COM   DIR.COM  GENCOM.COM  PUT.COM  SHOW.COM

While perusing through the reading material I have, I saw a lot of information describing what these files are and what they do. I will be researching these files in more detail.

Many different books about CP/M typically start by formatting and copying files. I am going to go a different way. Now, when I tackle a project like this, I prefer to be surrounded by manuals and books. My way of learning may be a little difficult to follow for others, but maybe someone somewhere might find something useful here.

I like to be able to look at the contents of disks in their drives. I remember, from back when I was learning about CP/M, that I would take a directory from a drive by entering the drive letter (as opposed to the numbering system used on the Commodore line of products), followed by the built-in command DIR. CP/M gives a simple directory listing of files on the selected drive.

Referencing the Digital Research User's Guide, DIR has quite a few parameters and there is a lot I can do with this command. Comparing this to the method used by the Commodore 128 DIRECTORY command, CP/M offers more options of displaying files - or how they can be displayed. DIRECTORY on the 128 used alone will give the directory listing of files on drive 8; showing the name of the disk, id number, file names, file size in blocks, file type, and number of blocks free. Certain files or file types can be selected using wildcard patterns. Other drives

can also be selected by adding D0,Ux (x can be any number from 8 on up – theoretically to high numbers less than 100). CP/M's DIR has many parameters. Here's a quick rundown of ways that DIR can be used..

**DIR <RETURN>**

Directory of contents of default drive (indicated by the letter at prompt).

**DIR B: <RETURN>**

Directory of disk in drive B.

**DIR FIL*.* <RETURN>**

Show all files on the disk in the default drive that start with FIL.

**DIR D: *.COM <RETURN>**

Show all files on the disk in drive D that end with the filetype .COM.

**DIR [FULL] <RETURN>**

Gives a complete listing of the file information on the disk in the default drive.

**DIR [USER=ALL] <RETURN>**

List all files on the disk in the default drive under all the user numbers.

This is interesting because files can be defined by the user that created them (providing the user specified his/her unique user number). So if I had a system with three users and they all signed on with their unique user number, I could take a directory listing of one, two, or all files created by each user. This is different from the Commodore filesystem, but I do see its unique advantage.

I can also list files by their size, attribute, or even tell CP/M to list all files on all connected drives! (This would be a separate step for each drive on the Commodore – if I didn't have a special program to do it)!

**DIR F***

This does not work as it does in Commodore mode. I found I need to use the previous method to find what I was looking for. In other words, this will *not* show all files beginning with the letter F.

Taking a directory has always been one of the first things I do because I want to find out what program files exist, or if there

are text files I can try to access to read. There's more, but I have found enough for now.

Next, referring to the directory listing of the CP/M System Disk, I see there are different types of files. To me, the files appear to be defined by the three letter description that follows the filename. Here I see .COM, .RSX, HLP, ASM, and .SYS. The Programmer's Manual describes these files as (in order listed above): Command file (program), Resident System Extension, HLP is a special text file (see Digital Research User's Guide p. 5-47 thru 5-49), Assembler Source, and System file. There are of course other extensions, and I found this list in the Programmer's Manual p. 2-11:

| | |
|---|---|
| ASM | Assembler Source |
| PRN | Printer Listing |
| HEX | Hex Machine Code |
| BAS | BASIC Source File |
| INT | Intermediate File |
| COM | Command File |
| PRL | Page Relocatable |
| SPR | Sys. Page Reloc. |
| PLI | PL/I Source File |
| REL | Relocatable Module |
| TEX | Tex Formatter Source |
| BAK | ED Source Backup |
| SYM | SID Symbol File |
| $$$ | Temporary File |
| DAT | Data File |
| SYS | System File |

After searching the manuals, I found I was correct. This naming convention is just a general rule.  If I wanted to, I could name a text file and use the .txt extension, which is the standardized extension for text files (typically ASCII). For the most part, I'll stick with the rest here until I learn more details later.

On to the program disks that I have. I wanted to see what this program was, and if I could get it to load.

I placed the genealogy program (disk 1 of 3) into drive A and took a directory. Here are the files:

CPM+ SYS  CCP COM   PIP COM   FR COM   STARTUP OVR
FINISH OVR    ENTRY OVR    MODIFY OVR    MODIND OVR
MODMAR OVR MODFAM OVR   ADDFAMCH OVR
ADDFAMPR OVR   DELFAMPR OVR   DELFAMCH OVR
PEDSRCH OVR   NOTES OVR   NTOPTION OVR   NTGETIND OVR
NOTEPRNT OVR NE OVR   UTILITY OVR   DELDATA OVR
DELIND OVR   DELFAM OVR DELETE OVR

I noticed there are three filetypes, .SYS, .COM, and .OVR. The last one strikes me as an overlay, but that is only a guess, but right now I am only interested in the program files. There is only one on this disk - FR.COM. So I ran it. The screen printed an error message on drive B. So, I thought, maybe it was looking for a disk in that drive. So I decided disconnect the 1541 and connect a 1571. I needed a disk for drive B and did not have one immediately available, so I remembered seeing a FORMAT command somewhere in the text I have been reading. Then I grabbed a blank disk, inserted the CP/M System Disk in

drive A and typed in **FORMAT <RETURN>**. The FORMAT program loaded and displayed the following:

C128 FORMAT PROGRAM
  15 May 1985
Drive A is a 1571 <— reversed characters


Please select disk type to format
C128 double sided <— reversed characters indicating this item is default selection
C128 single sided
C64 single sided

I removed the System Disk and placed a blank in the drive. Then I selected the first option (I'll research the 64 single sided option later) and then the program prompted:

Insert diskette TO BE FORMATTED
in drive A.  Type $ when ready,
any other key to abort

I typed **$** and the disk was formatted (once the $ key was pressed, the program immediately went to work – I did not have to press <RETURN>).

Then I put this disk in drive B, the genealogy program into drive A, and typed **FR <RETURN>**. The program loaded and displayed the title screen – and now I know what this program is (it did check for a disk in drive B). It is called Personal Ancestral File and carries version number two. I'll need to add an internet search to see if I can find an instruction manual (not too hopeful that I will find one, but it is worth a try).

By the way, I do have a 1581 drive connected, I wonder how (or if) CP/M uses it? I searched through the 1581 Disk Drive Manual, but I did not find anything specific (another item to add to my list).

O.k., since I do not have any word processing programs for CP/M, I decided to try out ED. As I was looking through the manuals, I did take some time to read a little about this program. According to the Digital Research User's Guide, ED is, "...a line oriented context editor. This means that you create and change character files line-by-line, or by referencing individual characters within a line (p. 5-30)".  This page referred me to a more detailed section for this utility (Section 6).

Here is the opening paragraph, "To do almost anything with a computer you need some way to enter data, a way to give the computer the information you want it to process. The programs most commonly used for this task are called editors. They transfer your keystrokes at the keyboard to a disk file. CP/M 3's editor is named ED. Using ED, you can easily create and alter CP/M 3 text files (p. 6-1)".

I checked to make sure side two of the CP/M System Disk was in drive A and I typed in **ED TEST1.TEX <RETURN>**.

The computer displayed:

NEW FILE

:*

I looked at this for a moment and decided that the disk I formatted earlier would be a better place to experiment with new files. So I placed this disk into drive B. I also thought I would copy PIP.COM and ED.COM to this disk (I wanted to make sure I would not corrupt my System Disk). I typed in PIP <RETURN> and the computer displayed this:

CPM/3 PIP VERSION 3
*

I could vaguely remember this utility, but went to the Digital Research User's Guide for more specifics. On page 5-70 I found what I needed. Here is what it says...

"This form of the PIP command starts the PIP utility and lets you type multiple command lines while PIP remains in user memory.

PIP writes an asterisk on your screen when ready to accept input command lines.

You can type any valid command line described under previous PIP formats following the asterisk prompt.

Terminate PIP by pressing only the RETURN key following the asterisk prompt. The empty command line tells PIP to discontinue operation and return to the CP/M 3 system prompt".

What I remember from before was that to copy a file from one drive to another, there was a specific way to type in the command. To make sure I had it right, I checked the syntax after I had entered this after the asterisk...

B: = A:PIP*.*

and the program copied the PIP.COM utility from drive A to drive B (the destination is typed in first along with any specifics followed by the equal character, then by the source drive and any specific parameters (here it is "PIP*.*"). I did the same thing with the ED utility. This way I have these utilities available when I create text files with ED. I might even add more later (e.g. SHOW and TYPE).

I noticed also that if I wanted to, I could copy a file designated for a certain user to any one of the other users. If it is on the same disk, it would be two copies of the same file (I will verify this in Part 4).

After this, I changed the default drive to B and entered the following at the System Prompt:

ED TEST1.TEX

This started the ED utility which showed the following (just like before):

NEW FILE

:*

It took me a little bit of reading to figure out how to start entering text because this is the ED utility prompt. There are a number of commands that can be entered from here that do certain things. One I found interesting was 0V (that's a zero followed by the letter 'v'). This command will show how much free space is left in the Transient Program Area (RAM) for the text file and the total amount of memory available.

Then I found the insert command "I", which places a user into "Insert Mode". Aha! The screen prompt changed to a number followed by the semi-colon. Each line is numbered for easier reference (this line numbering feature can be turned off with the "-V" command). I didn't see a "save to disk" command, well, one that made sense to me, so I had to sift through the manual to find out how to do it. I read that by pressing the ESC (escape) key, I would exit "insert" mode. Then I could type "E" to exit the ED utility, thereby saving the file to the default (or defined) disk (when starting ED, it is possible to tell it where to save the information when done).

Then I used the TYPE utility to display the text file on the screen.

I just had a thought, I'm almost sure that I can read and write to a CP/M disk from my old Windows 98 tower. I have a 5 1/4" drive installed. That could make file transfers easy. I will add that to my checklist of things to do later.

To summarize Part 3, I would do it like this...

*CP/M has many utilities to aid the user in managing disks. The DIR command will give a concise (specific info is accessed by using the SHOW utility) directory of files on whichever drive the user specifies. This is done by entering the following at the system prompt:*

***DIR <RETURN>***

*There are many filetypes used by CP/M of which program files have a .COM extension, system files have a .SYS extension, resident system extension has .RSX, and text files have .TEX. Even though these are quite common, there are many other filetypes for other purposes. To have CP/M list only those files ending in .COM, I enter the following at the system prompt:*
***DIR *.COM <RETURN>***

*To list only files starting with the letter K, I type in the following at the system prompt:*

***DIR K*.* <RETURN>***

*The FORMAT utility prepares a disk for use by CP/M. The FORMAT command on the CP/M System Disk for the Commodore 128 has specific options for that system enabling the user to format a disk in a 1571 for CP/M if the disk is a single sided or double sided disk. This is done by typing in the following at the system prompt:*

*FORMAT <RETURN>*

*Then selecting from the menu the option desired.*

*PIP is a utility with many features among of which I can copy files from drive to drive, or user to user (or both).*

*ED is a utility that can create text files. Although it may not be as easy to use as a word processor, it is much better than not having anything at all. It does seem cumbersome at first, but I anticipate more use will provide a more accurate analysis and critique of its abilities.*

*Accessing existing files (this example filename is LETTER.TEX) is a matter of entering in the following at the system prompt:*

*ED LETTER.TEX <RETURN>*

*Text can then be inserted at the ED prompt (*) by entering the letter I.*

*ED will save when the user presses the ESC key and enters E at the ED prompt (exit program and save file).*

*Creating a new text file is easy. Simply enter ED followed by the filename and the extension you wish to use. After pressing RETURN, ED will show a prompt. Entering the letter 'I' will put the user into 'insert mode' where text can be typed. The procedure for saving the file is the same as given above.*

*There are many features that ED offers and details are found in the Digital Research manuals.*

*TYPE is a fast way to read a text file. Simply enter TYPE followed by the filename and extension and the utility will display the file to the screen. Specifying a drive before TYPE enables the utility to access the file if it is on a different drive.*

End of Part 3
See you in the fourth instalment!

Commodoreman

**CHECKLIST**
-Hard drive (note: logical drive capacity cannot exceed 512 MB – CP/M Programmer's Guide, sec. 1.5, p. 1.11). Find out more about this feature (can I still get one?)
-REU CP/M disk
-BDOS
-What is the date displayed on startup? Is it the date of the OS or is it there to represent the current date? If so how do I change it?
-How do I use other peripheral devices (modem, printer, joystick, etc.)?
-What other high-level languages exist (BASIC, FORTRAN, COBOL, etc.)?
-Find a manual for Personal Ancestral File version 2
-How do I use a 1581 with CP/M?

-What does FORMAT mean with the 64 single sided format option?
-Can I read and write to a CP/M formatted disk using my Win98 tower?

**LIST OF RESOURCES**

**Hardware:**Commodore 128D
1541
1571 (2)
1581
1750 REU
1702 Monitor
1902 Monitor

**Software:**CP/M System Boot Disk
Genealogy Program
other CP/M formatted disks (copies of system disk)

**Printed Material:**
Commodore 128D User's Guide
Commodore 128 Programmer's Reference Manual
Digital Research User's Guide (3 books in one)
Commodore 128 CP/M User's Guide - ABACUS ISBN#0916439453
Commodore World Magazine

**Digital resources:**
Commodore 128D User's Guide - www.commodore.ca/manuals
Commodore 128 Programmer's Reference Manual - www.commodore.ca/manuals
Digital Research User's Guide - www.commodoreman.com/commodore/books/man/Digital_CPM_Manual.zip
C Hacking Magazine

**Other**
-An Internet connection

-Tower computer with Windows98 and a 5 1/4" floppy drive (I am almost positive I can read CP/M formatted disks with this computer!)



http://www.youtube.com/watch?v=-MbOnDQUMXU

# Happy Prog'ing
# By John Fielden

**Functions we haven't covered yet**

A bit of time here allows me to ponder the things that the prog'ing series hasn't covered from the c16 manual. Well, I haven't covered ~Assembly yet; dreading it!!! Even though the c16 was my very first computer. The monitor still remains much the mystery, and before the pc and thus internet became available to me, I'd never heard of "TedMon". In fact the only Mon I had heard of was Solomon from dear old Granny waffling on after her church commitments etc. -still bruised from all the bashing! But no, I find they are probably not related! -unless you know otherwise... Still Mon was an abbreviation for Monday. And the only Ted I knew of was from the comedy series HideHi. Even when Dad and I fumbled on a runstop & reset, to save a program that crashed, I don't think either of us knew the 'gibberish' collection of letters and numbers meant we'd entered the world of TedMon! Proceeding to blindly press letters and getting random results got boring after awhile with no explanation. (I think he left that book on the shelf opting for a games book that taught nothing much on its own. -enough of my depraved childhood let's get on with it!)

**DefFNA commands.**

The main thing that had been concerning me was whether the Basic statements were covered. And I realised DEF FN... has so far been over looked. So we'll start with that in a moment.

I'm very Thankful this time to the author Brian Lloyd for his companion book, as listed in program, for both the above explanations. They are the best I can find with the limited information available to me today. I think you'll agree they are well written -if you have the book ...On the other hand if you just have my hash up of it to go from, just do the best you can. And Good Luck -you'll need it!

**DEF FN**

The function command takes a variable and does something with it before giving you the answer. The statement allows the programmer to make their own function/s if those they want aren't already available. Like "cos", "sin" etc. Which are yet to be covered properly. The term is an abbreviation of "DEFine FuNction" and is used in numeric calculations.

The included program -which will go on and on until you hold shift and press 'runs/stop' keys, or type in an added exit to it. (like the Trap command).- And gives this as the repetitive calculation:-

$$((ZZ/5)*32)^2$$

Where ^ means raised to power of, an arrow up in program.

110 defines the calculation and gives it the name A1 for future reference.
ZZ is the variable that will make each repetition of the sum

bring out a different answer.
  - divides ZZ by 5. multiply the result by 32then square the result. -So he says! Thank fully he gave us the rest f the listed program to show various outcomes. Now all you need to do is type it in and RUN it. Only line 50 may need explaining. It says to the computer use N. as part of this calculation where you saw ZZ before. This is because any number (or variable letter/s that represent a number) can be used in place of ZZ.

Like the weird monkey says "Simples!"
  ..."pff!"

**(EDITOR I think is a meerkat)**



The more I look at The Commodore16/plus4 Companion, A beginners guide, Brian Lloyd, 1984, Sunshine books (an imprint of Scot Press Ltd.), London.  0-946408-64-5

The more I like it. It's explanations are clear throughout. For Instance, I'm now able to better understand the correct form with RDOT, RGR, RCLR, RLUM etc. (pg. 92) is : Variable=RGR(x) where x in above case is a dummy number that the computer ignores. From the variable it is determined what the value of said code is. Though why any programmer would need such things when they programmed it in in the first place still eludes me. I suppose "You never know when something will come in handy!"
?"(xx!xx?)" - ??? "Is that what that bloke TedMon said in his Proverbs Granny?! -???
  "Yes dear, ..., ... "

**Machine Code**
From Chapter 11 of The c16/+4 of The Companion book,  B.

Lloyd.  p.107 - 121

After a brief but crystal clear reminder that RAM is lost when the machine is reset, whist ROM is kept.  ROM is permanent, and there a many types, for instance the CPU, the brains of the unit. This however understands only its own language.  So it needs another type of ROM, The Interpreter to understand the BASIC commands given to it.

RAM is volatile.  You can't use all of RAM for a program, because everything on screen is stored in part of it.  Hence the "Commodore Basic x bytes free" etc. on the startup screen when you turn your computer on.  This is the workspace the user is given to play with.

I also found interesting the analogy that ROM is like a glass jar with a lid you can't open.  You can see what is there without changing it.  While RAM is like a glass jar with no lid so you can change as well as PEEK at the number that it holds.

Try,
POKE 3072,1
You get an 'A' in the top left of the screen.  1 is the ASCII code for A.
 If later you wish to look at this code for this part of the screen.  You would type,
PRINT PEEK (3072) And of course 1 is the outcome.

Next to it at 3073 is the result of 32 because this is the code for a blank space.

Now try,
POKE2048,128
Amazingly -to me- The 'A' that was put in by the above POKE now flashes.

2048 is the colour memory which starts at the top left of the screen from 2048.  To put colour in a letter at say 3116, which it is assumed a character up to the maximum 255 has been chosen.  So, deduct 1024 from this.  We should now have POKE 2092,

 First you must decide what colour you want.  From 1 to 16 as with the command of the same name.  Then multiply the Lumi-

nance level (0 to 7) by 16 and add this to the number for the colour.  If you want the character to flash then add 128 to the result.

Now RUN or type in the program listed as "p.109".  A full description of this is given in The Companion book.  It puts pretty randomly coloured dots onto the screen.  When you've had enough press runs/stop key.

Mute point:  I can't help wondering what the same program would look like with CHAR in place of POKE.  The main difference is while POKE has 1 number for its co-ordinates CHAR requires two -as with CHARts!-  Following on from this, how complicated would the same program be with PRINT statements?  It would be interesting to see how the colours fit in.  Have a go if you wish but this is moving away from true assembly.
Try,

POKE3072,81
To understand line 60 a bit better.
It's all pretty much in the above explanation, or what has gone before anyway, so I won't repeat myself here.

### An Introduction to TedMon
(pg.110 and beyond.)

ML (Machine Language) is what the CPU inside your computer understands.  As the CPU does not therefore need an interpreter with this programs are much quicker than in BASIC.  (To be honest, though I am enjoying learning about our Ted.  I'd sooner have BASIC with a Compiler, at least for the things I've been doing thus far.)
 To learn Assembler etc. properly -if you haven't already- it would be better to seek out books dedicated to the subject.  This is a discovery rather than a lesson.  If you like it, you can
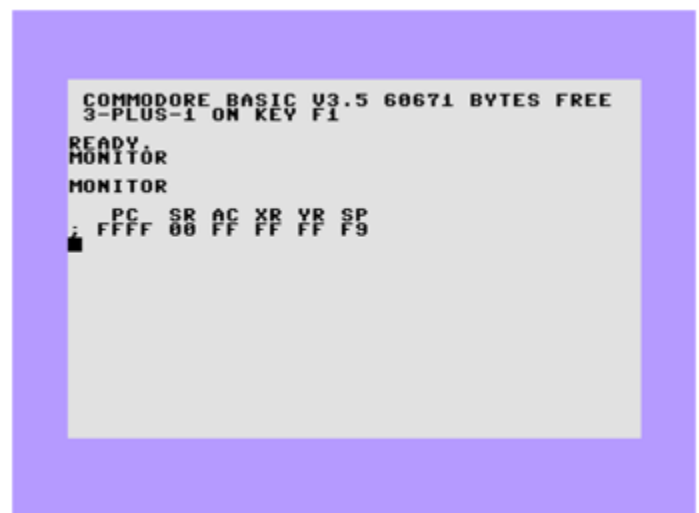
look into it further elsewhere.

The Commodore 16 and The plus 4 came with an Assembler built in.  Called TedMon.  The way to access this from BASIC is to type

MONITOR

(another command from The c16 manual now covered!!!)

Though the numbers may vary, you get something like:-

MONITOR
  PC   SR  AC XR YR SP
;FFFF 00  FF  FF  FF  F9

we will be told later what this all means.  But we are now in TedMon -so 'listen carefully my son, these words will give you wisdom...' or a bit to start us off at least.

Try typing
M 8188
(include spacing or results may be different.)

The four digit number immediately after the greater than sign is a memory address in Hex.  After that  the eight 2 digit numbers -I'm told these can be changed...
In Hex A=10, ... F is 15, also called base16 numbers because of the 0 being also or pseudo 1.  i.e. considered the first number in the sequence.  Therefore the list does not go 8188, 8189, then (8190) -this must be 818A, 819B and so on.  Thus the 8 two digit numbers contain what is in memory. 8188 holds A2, 8189 holds A9 and so on until we do actually get hexadecimal to 8190 which is the next line and contains C4.  And so on again.

8188 is 33160 in decimal.  In a prior issue we've looked at conversions.  The area of ROM that we have looked at is the reserved word table of the interpreter.  A . is given in place of any unprintable character.

Press M if you wish to continue examining ROM.  Alternatively try,
M 8188 8382
Pressing The Commodore key will slow the speed of the scrolling.  Ctrl and S -if memory serves- to pause screen.
A sort of M =
List Memory (line) 8188 - 8382,  but without the extras.

If the above wasn't ROM, you could change the listing.  For a bit of fun with RAM try,
M 3000
Where ROM changes itself back, RAM can be altered.  But beware of possibly strange occurrences when you blindly play around without knowing what things do.

To leave TedMon press,
X

and you're back in BASIC.  What's more, if you didn't delete them, you still have the dots from the prior program in memory!  Type,
LIST

Again for TedMon type
MONITOR
here's some more tips.
F 2400 2A00 A3

Will fill all that area of memory locations with the value A3.

 H 7000 9000 C0
Will search through all the listed memory locations Hunting for C0.  And should display a list of the memory locations that hold this value.
H 8000 9000 'COMMODORE
will return
80CF
Now type
M 80CF
and you will find it is written.  Be sure to use the 'apostrophe to search for strings.

T 0C00 0FFF 0BD8
Transfers everything up one line.
We may work out how to move this down again later.  Have a go. 08D8 is 40 memory locations before 0C00 which is the start of screen memory.

**Writing Machine Code programs**
A is for Assemble.

A 2000 LDA # $01
A 2002 STA $0C00
A 2005 LDA # $80
A 2007 STA $0800
A 200A BRK

As you type the computer adds things, working out the next memory location when you press return, like AUTO in Basic.  And it will Assemble the line as you go into its own machine code from the Assembly Language you type in.  LDA has the code A9 for example.  You will see how things change depending on their use.

A at the start of each line stands for Assemble (pg. 114), having stated the start the computer works out the next line (a sort of auto statement!  Wait till you see 'Teddies' alternative for RUNning the Assembled program.)

Every Machine code command has a code number which the computer adds in the line before these.  The command LDA (Load Accumulator) has the code number A9.  So A901 being the equivalent of  LDA #$01.  (surely it would be easier just to put the former! -ah, well -'tis not mine to question why' as the saying goes 'it is but mine to -' get utterly flumaxed!!!

 (p115) The computer likes to have the low byte part of the number before the high byte.  So, in the second line; when a memory address is used  STA $0C00 becomes 8D 00 0C.

I'm still unclear as to the terms "Low Byte", and "High Byte" and what the differences are etc.  Also, B. Lloyd says he 'doesn't intend to cover machine code programming in depth'.  So, much is lost on me here.  Nor does my Computer Dictionary seem to want to tell me the difference though the term "bytes" are in.  And we know already these are linked to bits but I'm still a bit fed up that I can't find the explanation; and can only guess that the high byte may be something to do with or same as the first number in a POKE statement in BASIC.

Whether or not this is true, to start the program type,
G 2000

G is short for GO, but if you type in GO; You'd get a question mark and nothing much else would happen. You will see that this is the ML. equivalent of the single line POKE command (In the segment titled "machine code" above, or Pg. 108 of the companion book)

## DisAssembly
OK!!!
The teachers at school liked attention quite a lot, I remember being dragged in as a class and having to sit still for ages on end and listen to a waffling old - ...err, wait a minute... oh! (**#!#**!! -AKA 'Ooops!') err, ... "ahem!" .."Ah, yes!"

Pg. 115-116 of Companion book.

Assembling is to converting a series of machine code commands into a series of numbers. TedMON can also ~Disassemble a machine code program, that is convert all the numbers to the commands which they represent.

Try,
D 9000
just for an alternative that we cannot change as this is a part of memory known as ROM described earlier. For RAM and our little Proggie; type
D 2000 200A
the space making the form to numbers readable.

The dissasembly is in the format which the computer converts your ml program to when it assembles them. To alter a line of our program. change the first line to,
.2000 A9 01 LDA #02

When you press return A9 01 is now A9 02.
The D command takes the same format as the memory dump.
D 4000 4020 will Disassemble the contents of these hexadecimal memory locations.

(Pg.117) As it comes next, though the author considers this less useful but to type,
C 1000 2000 4000
C stands for compare, and would in the above case compare memory locations from 1000 to 2000 with those from 4000 to 5000, the last number (5000) isn't needed as it occurs by default with the total amount to be compared -(Clever calculators, these things!)
   The computer displays a list of memory locations which differ.

## Saving & Loading ML. Progs.
To Save our above machine code program. We need to give it a name, then tell the computer tape or disk, I chose disk given that is all I have -or need for that matter- and finally the memory locations from and to that indicate where our program is.
In other words type,
S"pg114to117",8,2000,200A

To Load this back later. All we need is the name of the pro-

gram, and whether it's on tape or disk. So, if you put the prog on tape. Having turned the computer off then coming back to it after awhile to continue. You'd type.
L"pg114to117",1
   where "...",1 is ,8 if on disk.

We can even Verify programs in TedMON with
V"pg114to117",8

All the usual messages, screen going blank to load/save etc. occur as usual. If Verify OK. the flashing cursor is returned, otherwise VERIFY ERROR is displayed.

## The Registers (Pg. 118)
Not a lesson to be late for -as this refers to the numbers when typing
MONITOR
or holding the runs/stop key and pressing reset. And on so doing opening TedMON.

PC is Program Counter (I'm so pleased; had this been for that moister which is political correctness -AKA sensible Incorrectness then I would've ended the session here!!!) The Program Counter always points to the part of the machine code program that the computer is carrying out. If monitor was the first thing you typed when turning the machine on 0000 will be displayed under PC. Now, it occurs that loading in the Assembly program will alter these numbers. However, using The usually Excellent Yape Emulator for the c16/+4. The L"pg114to117",8 Having changed the filename to PG, through the menu of Yape. I then opened it with the menu. It will go as normal. And now it has it anyway, it will Load no problem -though you don't need to with the emulator if you've gone through the file.

I was right. PC has changed. But so have two other Register Counters. I pressed "R" for Register to find out. Note: If you go too far down the page so that the computer has to scroll lines upon return. You won't see the change as the top line will be wiped off by the scrolling.

The next Register we come across is SR, The Status Register contains information about the operations which have just been carried out.

Next The Accumulator. AC can be thought of as similar to a variable, although it is used in a slightly different way from variables. Next XR and YRegister are similar to the AC but each can do things the others cannot have.

The SP. Last on the list is the Stack Pointer. The stack is an area of memory where numbers are stored or STACKed! Like a pile of books on a shelf. Only with the specific rule that -due to health and safety, back issues?- You can only move one book at a time. And this must be from the top of the pile on a first in last out basis. Stack can hold 256 bytes. The number points to the first free memory location in the stack.

Now type,
R
change XR from BF to 2A. This can be done, thanks to ";" the semicolon at the start of the line.

Nothing really happens as we haven't engaged the X Register -yet.

**The SYS Command and USR Function**
 pg.119-121

There are two ways to execute a machine code program from BASIC. ( err, The title is a clue!)

 First -for both- Dissasemble the ML.
 2. Change 200a from BRK to    RTS
- Short for ReTurn Subroutine.
 3. type,
 S"PG121",8,2000 200A
- hence saving the new layout as a new file.
 X
- And on typing that you will see "READY" and we're back in BA-SIC.

You should have the "pg109" program in basic memory. If not load it -through BASIC. And LIST the program.
Alter from line 70 as follows.

 70 k=k+1
 80 ifk<25 then 20
 90 k=0
 100 sys8192
 110 forj=1to300: nextj
 120 goto 20

Save it on disk as,
SAVE"PG120",8
or
DSAVE"PG120"

As there is no switch over with the letter. We might try to expand the ML. My next guess was that there isn't anything important after our first RTS at 200A. So, with a bit of trial and error I Disassembled 2000 to 2012,
D 2000 2012 And from the next line after 200A, I wrote essentially the same program. Only at 200B -the counter-part of 2000 "A9 03" has A9 09 for its trouble. RTS is at 2015, the ??? at 2016 I changed to a BReaK command for good measure.

Now, I am to change the basic prog. So,
S"Pg121b",8,2000 2016
Saving...
X
LIST

 change line 90 to
 90 K=0:  KK=1-KK
 100 if k=0 then sys8192 :else sys dec("200B")
   Note,  for quicker code. i.e. else sys dec...
     try, making the calculation before the code, in a PRINT statement.
 ? dec("200B")
-in basic of course...
 I get 8203, so you may wish to remove the DEC... after SYS and type the actual number -this is 8 commands on.

Now type,
SAVE"PG120B",8

You can give each program a better more memorable name if you wish. The USR function seems essentially very similar -at this point to the SYS command. But as a function is more complicated. I have not used it in place of the SYS command for the simple reason that SYS works, having been there first:. And I had to experiment a lot to make loading the assembled sub-routine from the basic program work. But so as to not hold any-one back in our process of learning. The USeR function is given in the companion book *Pg.120) as follows:-

The USR function is an alternative way to execute a machine language program from BASIC. This function can be used to pass a number or string of letters to a machine code program, as well as to execute the ml.
 However, you must store the start address of the machine code program in memory locations 1281 and 1282. For instance, to execute our machine code program using the USR command we must carry out this procedure.

PRINT DEC ("2000")
The c16/+4 returns the number 8192. This shows in DECimal where in memory dear old TedMON holds his Assembly!

Now I don't know how this is the high byte calculation. I only know the book I'm working from says it is. So,

PRINT8192/256
32

READY.
POKE 1281,0 : POKE 1282,32

READY
X=USR(0)

Because it is a function it must be used with a variable, in the above format. The value in brackets is the one passed to the machine code program. Because we don't need to pass a value to our particular program, it does not matter what value is used.     The variable (x)  which is used will have a value stored in it  when the machine code routine ends. -So, do not use a variable which is already in use in the program elsewhere.

The companion has reached the closing paragraph for the chapter. And I'm feeling I have gained some understanding that I didn't have before of ml. And especially how to use TedMON. If you can get hold of the book it is well worth reading. There are more books of course but this is the best start I found for people with no knowledge to get started. Where I felt let down is the apparent reluctance of this and other books (including those specifically written to teach ML!) to explain clearly and with examples how to use a saved machine code sub-routine in a basic program, bearing in mind that both the main basic program as well as the machine code has to be loaded in again after the machine is switched on. Learning how to chain was something of a leap of faith!

So... Good luck with that...

("hee-he-hee!")

I'm not that cruel -often! And as I know there are people also struggling on this. I am of course including it, though tempted to charge! Here it is in these pages. After all, what use is either basic or ml. without the ability to interact!

OK. A bit Scooby-doo, but my first clue came from that useful of all sources. The c16 user manual. (Pg. 104 to be exact). This explains the numbers after the LOAD, and confirms that you can indeed use this statement within a program. A process known as Chaining. I knew that the first number being either 1 for tape, or 8 for disk, but the second number had been a mystery. Just there! Ever since the supplement from the original c16 explained that the only way to load the four games that came as part of the introductory pack (namely, in my case; Chess, Rolf Harris Picture Builder, Xzap, and Punchy) had to be loaded thus,
LOAD"",1,1
I remember my dad calling the " ", tick tick. As usual, I didn't have a clue what he was on about at first. But it seems easier than saying "two quotation marks". -very posh!

The second number is labelled as U for Unit, and known as the relocate flag. it determines where in memory a program is loaded. 0 is for loading at the start of BASIC. That is $1001. The part of memory where basic naturally starts. 1 tells it to load into the part of memory to where it was Saved, $2000 in the case of our machine code routine. Default is 0. And usually only used for ML prog's.

Things moved on from pages 120 and 121b. programs. Though 121b is still the ml code called from basic. The basic program had to be altered, modified and generally experimented on. So, for safeties sake 120b became asm along the way. There were other files including an experimental 'loader' program which tried to load both the basic and ml. With various results from freezing at the load command, (despite the ml. having been loaded), basic code that changed from what I'd typed in. To one coloured dot in a random place on screen before the program stopped having not loaded the ml. So, when I say it took a lot, it did!

The ASM file shows some of the previously tried ideas, as well as the way that worked. It seems you can't load in the subroutine first -at least not at this point of knowing. And the screen will come up with a round of dots then clear the screen again before putting in the letters and setting off with the dots again. While a file called ASM1 tried to call the ml. first, at an rts -having altered the ml, and start area- loading in the ml at the start. But this again just froze the program. I don't know how ASM2 works. That is to say the in-program load requires the sys command somewhere near it. Here, I purposely turned it off so that it could not call the sub-prog in ml. -at first. This is most like how it was when I had to load the two programs separately. ("aah!" how long ago that now feels!)

I'm chuffed because I always wanted to have background music in projects. It just wasn't possible before! I think though as

much ground has been covered, we'll end there for now.

So, on with the listings.

### The amended machine language code.  (I C.)

```
READY.
LIST


READY.
MONITOR

MONITOR

  PC  SR AC XR YR SP
; 0000 00 FF 00 00 F9
D 2000 2016
. 2000  A9 03    LDA #$03
. 2002  8D 00 0C STA $0C00
. 2005  A9 80    LDA #$80
. 2007  8D 00 08 STA $0800
. 200A  60       RTS
. 200B  A9 09    LDA #$09
. 200D  8D 00 0C STA $0C00
. 2010  A9 80    LDA #$80
. 2012  8D 00 08 STA $0800
. 2015  60       RTS
. 2016  00       BRK
```

---------------------------------------------------------------------------

### The amended BASIC program code.

```
ASM

10 REM  POKE1281,0 :POKE1281,32
20 A$="PG121B"
30 B$="ASM"
40 REM  LOAD (A$),8,1, 2000
50 REM  LOAD"ASM",8,0
100 REM POKE1281,28 : POKE1282,153
1000 REM *ASM*
1010 REM *PG.109* *COMPANION*
1020 COLOR4,2,7:COLOR0,2,7:SCNCLR
1030 X=INT(RND(0)*999)+1
1040 C=INT(RND(0)*15)+1
1050 I=INT(RND(0)*6)
1060 C=C+I*16
1070 POKEX+2047,C:POKEX+3071,81
1080 K=K+1
1090 IFK<25 THEN1030
1100 KK=1-KK :K=0
1105 IF LL=0 THEN LL=1:LOAD"PG121B",8,1
1110 IFKK=0 THEN SYS8192 :ELSE SYS8203
1120 FORJ=1 TO 300 :NEXTJ
1130 GOTO 1030

READY.
```
---------------------------------------------------------------------------

**Final Breakthrough 7 March 2012**
```
10 REM  POKE1281,0 :POKE1281,32
20 A$="PG121B"
30 B$="ASM"
40 IF LL=0 THEN LL=1:LOAD"PG121B",8,1
50 IFKK=4 THEN SYS8192 :ELSE SYS8203
100 REM POKE1281,28 : POKE1282,153
1000 REM *ASM2*
1010 REM *PG.109* *COMPANION*
1020 COLOR4,2,7:COLOR0,2,7:SCNCLR
1030 X=INT(RND(0)*999)+1
1040 C=INT(RND(0)*15)+1
1050 I=INT(RND(0)*6)
1060 C=C+I*16
1070 POKEX+2047,C:POKEX+3071,81
1080 K=K+1
1090 IFK<25 THEN1030
1100 KK=1-KK :K=0
1105 IF LL=0 THEN LL=1:LOAD"PG121B",8,1
1110 IFKK=0 THEN SYS8192 :ELSE SYS8203
1120 FORJ=1 TO 300 :NEXTJ
1130 GOTO 1030
```

READY.

Along the way I renumbered the basic code from pg. 109 of the companion book. This was done as I blindly searched for a way to load in the programs from a program. It seems basic loads out the old program, but if there are numbers within numbers in one program it may trip the main program up. But as this was not so much about learning to chain programs, we can leave such for another day.

**Lines**
10 - 30, (even to 100) are remnants from previous experiments on loading the ml. sub-routine in. Ideas when looking through books and having a go from vague comments that I'm sure the authors understand. It turns out you can use LOAD with a variable, so long as that variable is assigned a name$.

40-50 Early attempts to load in the ml. on start up failed, so as you'll see from the original ASM listing these lines are modified here; the idea from the plus 4 reference guide. Another vague offering! So,

40 loads in our precious ml. It will never do it again as the variable upon which it is allowed to has been changed. (Note: If the computer encounters a LOAD statement within a program, it will not only attempt to load it, but then RUN it automatically as well. *But as BASIC comes first therefore that is being run again.

50 Now to allow the computer 'closure', it seems we must offer it a SYS command or it will refuse to move. In this case we give it a conditional offering. If the named variable KK = 4 then it can run the ml. However, it doesn't as we don't want it yet. So it will skip it. (Very logical of the makers of basic to make sure there is this though!)

100 you will see is REMarked, and is to do with USR which I haven't tried yet. I was seeing if it would unfreeze the comp. before I 'found' the conditional operand. (ie. figured it might work this way!)

1000 name over original.
1010 named prog.

1020 - 1070 is about the same as pg. 109 in the companion. Thus needs no further explanation as they put the random dots on the screen.

1080-1090 create a count so that the repeat is interrupted. Currently set to 25 but change this to whatever you wish, if you wish.

1100 When the limit is reached the program continues, instead of jumping back. And we find that the count we've used is reset. And we again use a switch for the following lines.

1105-1110 The latter having been modified earlier to show the machine code being called. Is therefore already explained. Instead of one SYStem call there are two depending on the letter that is currently flashing at the top left corner of the screen.

   1105 on the other hand was inserted as that leap of faith I mentioned earlier, when I was on the point of woefully telling the editor I'd bitten off more than I could chew.

  In this program it is ignored due to lines 40-50. But was essential without them. It still only worked to load in ml. once because once is all that is required. After the "program name",8 denotes that I have used disk. Change this to 1 if you are using tapes. The second number,1 tells the computer to load in the program at the memory where it was saved. By default this is 0 for the start of basic. The Unit is called a relocate flag. See the respective manuals for more details.

1120 A pause on return to basic. Probably unnecessary except for the purpose of basking in glory!!!
1130 In case I have to explain this line. It keeps the program going. But if I did, you either haven't been reading from Iss. 28 (on and off), or I've wasted my time!

**Having done all that.**
There's still more for the sake of thoroughness. If you type, NEW You only erase basic. The assembled memory remains intact. To prove it having typed NEW.
type, SYS 8192
C flashes in the top left, control is returned to basic again. As there is nothing else down to do, we get the prompt,
READY.

Incidentally, we could've called the part of ml. that prints the 'I'. had we wished. Now to clear this.
  A reset is probably safest, though any program needing that space would probably just write over it. Alternatively you can make it safe.

type the above, NEW
after pressing return go to TedMON. ... ie. type,
MONITOR
D 2000 2016
The list will appear.
Option one. Go over each STA, each LDA, RTS and so on in the list. type, ???
and space out any letters after. And go down the list this way.
Or, we can use option two, and get it all done with one command.
F 2000 2016 02
but! Remember to put the odd BRK in here and there or the machine will crash or reset. Still better to reset anyway.

Happy Prog'ing
John Fielden

# Jack Tramiel, The Founder Of Commodore International, Dies At the age of 83

## My personal memories of Commodore BY COMMODORE FREE

Sometimes a date sticks in your mind; and when someone says to you "oh were you on X date" you immediately remember where you were and are transported back in time as if the event had just happened. The news about the death of Jack is one of those dates. When I heard the Death of Jack Tramiel and was so saddened by the news, and indeed it took some time for the information to sink in, It didn't occur to me that Jack wouldn't live forever. I went up to my "Man cave" after work (computer room) and looked at all the Commodore equipment I had and said "Thanks Jack for all the memories" it took some time for the sad feelings to subside, and I was quiet for some days, rolling over the memories and sadness.

Sadly I never personally met Jack so can't comment on him and his personality, All I know is what Jack left me, and the many memories I have of Commodore and my Commodore equipment. So in honor of Jack and his family here is my story about how Commodore touched me and how a man with a vision changed my life. I am sure as a reader you have your own stories and they are possible similar to mine.

### School

I never liked school, and it was made worse for me by the fact that I wasn't allowed to use the schools computers. I think at the time these were some form of Commodore Pets, but as I never saw them and I can't really be 100% sure. You needed to be in the top math classes to use them due to the complexity of the machines (or so school thought). I had friends who used them and brought home homework from computer lesions. I did this homework for them in exchange for them helping me with English and maths.

### VIC

I used to buy Computer magazines from our local newsagent; and had read about the Commodore Vic20, I can't remember the year but the machine must have just been launched in the U.K. as I do remember the magazine had a full review. The Vic was keenly priced and of course a number one item on my Christmas list. My Dad had a mutual friend of the family who had just bought a Vic20 and said he could show it to us and explain how it all worked. So we boarded the car and shot down to his house to see this wondrous machine. Not only did the machine look beautiful it smelt new and felt like touching gold. To be brutally honest it seemed more like I was demoing the machine to him; as I loaded some of the games from the supplied tape on the datasette and then loaded Blitz bomber, and we all played for what seemed like 1 hour, until someone actually landed the plain and a little ma came out waving. My father's friend seemed baffled at how to actually load things, but with the research I had done I managed to step up to the mark and show off my skills. We all played on the machine for a while longer and thought was the most amazing invention ever.

After some begging and saying I would wanted nothing else, I somehow convincing my parents that missing out on a computer would hurt my school work, especially English and math's, they agreed to the purchase of a Vic20 with datasette for my Christmas present.

## Christmas

Christmas day couldn't come fast enough for me, I knew exactly what to expect and although I did receive other presents, I don't know what they were as the Vic took over my whole Christmas period. I remember unboxing it and setting it all up on a portable TV and a card table with green top.

I managed to do some rudimentary programming and I wrote a program for the machine to display my name, and then accept a key input and print out the key pressed on screen, I even managed to get the machine to play out the tune "we wish you a merry Christmas" although I think some artistic licensing had to be used for some of the notes and rests. I also realized at that time I would never make a programmer, but I did have some sort of aptitude at diagnosing problems. So maybe something other than programming would exist for me.

## 1541

I found Tape loading slow, especially while I worked on things like databases. So I decided to save up for a disk drive, my savings were also helped with birthday money and doing odd jobs for family members etc, basically anything I could do to earn money for a disk drive I did, and soon I was the owner of a Commodore 1541 disk drive. The VIC machine was a wonder to behold color and sound like nothing else around at the time. I know some friends looked in envy at the machine.



## Commodore 64

The Vic soon gave way to the Commodore 64 and although I couldn't part with the Vic (yes I still have it today) the new Commodore machine took over my life and it seemed my whole being, everything I did was revolving around the Commodore 64, I would save for games read reviews of games play games and use various utilities, I even managed to haunt a few bulletin board services were I would chat and download the odd SID file. A few fiends would meet at a local computer meeting to worship the gods that the club called "hackers" these guys could take a copy protected disk and by the end of

the meeting they had beat the protection and they would have a version they could pass around to all the members of the club.

Now I come to think about the process I thinks this was Not exactly the right thing to do, and yes I know it's not 100% legal either. However If I played a game and liked it I would buy the disk just to physically own the thing, so for me it worked as an advantage, any games I disliked the disks would be re-used so I suppose from me no one actually lost any money from the process. I did however still enjoy loading from Tapes, as this was the popular format in the U.K. with the introduction of the fast loading  for the Commodore 64 I didn't have to wait too long and the music and loading screens made up for the speed, I also still feel that even now the anticipation of a tape loading is well worth it.



## SID

The thing about the c64 over the vic20 was the sounds, the Sid chip in the c64 was something special and when programmed to an inch of its life by people like Rob Hubbard, well it was an instrument in its own right and still is. I used to tape game music onto audio cassette so I could listen to it on my cassette player, something I still do now! only they are on Mp3 and flac files and I usually listen to them in the car. The tunes were so catchy and the sounds so cool, I would even buy games just because of the music. I still remember loading "Monty on the run" and listening to the music for ages before even playing the game, classic stuff!
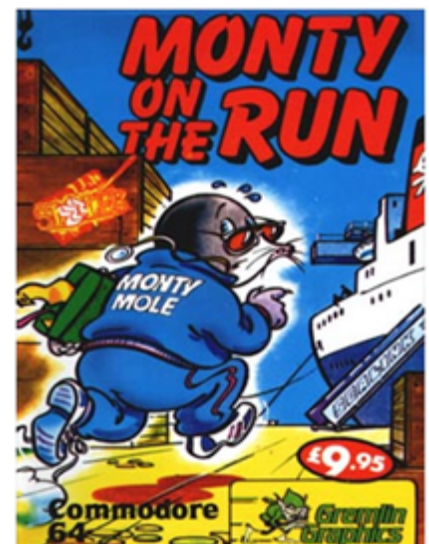
I dabbled with programming as I said, but realized I was no good! and so had to find another method to pay bills and forge out a life. I left school and entered a day job I went to college and qualified as a carpenter/joiner, I worked for a local company before being made redundant and moved to another company working as a joiner, but in the back of my mind I wanted to do something with computers, I didn't know what, but just wanted to do something. I seemed good at helping others and explaining about machine terminology, so was there a job in this, My father purchased an IBM compatible machine but couldn't really work it, I studied the machine and soon found out how to trouble shoot problems and what the Config and Autoexec files were for. I also scored as a Drum programmer in a local band, mainly because I had an add-on for the commodore 64 that replayed sampled sounds. One of the sound kits supplied sounded like "neworder" blue Mondays drum kit so I got the job! The band didn't last very long and it was so frustrating listening to drum sections over and over and then re-creating them, but at least when playing live it was just a case of pressing a button and then relaxing and looking cool! I was soon called on to help others setting up computers at home, tuning in TVs loading files general helping out really, this started to increase and I thought, maybe I do have something here.

At least this spawned my interest in more musical add-ons for the C64 like the Fm digital sound expander, midi cartridges and the purchase of a real midi keyboard for my commodore 64 to trigger the sounds, very cool at the time. My programming experience increased and I could write some simple applications, my favorite was a wall paper estimator I still use this today when decorating.

**AMIGA**

I moved to an Amiga 500 for a 21$^{st}$ birthday present and again was totally stunned by things like tracker music programs, heck you could sample bits and pieces and put together your own remix just on 1 machine. I had always been musically inclined having been taught the piano from an early age and soon the Amiga had an add-ons like me C64 in the way of a Midi interface and a copy of music X, My equipment grew also with various keyboards and sound modules being added and then I purchased Drt`s KCs software and again did nothing much but compose music.

Some friends pestered so much I actually made a few demos but rely the music was for my own enjoyment. As I was made redundant again I thought I would try working as a shop assistant or a PC technician as I had built some IBM clones for friends and diagnosed various problems. I went for an interview with Silica shops, after a successful interview I worked as the Amiga expert. I was promoted to assistant manager then for a short time as manager, it was around this time Commodore filed for bankruptcy and I realized that Silica wouldn't be able to function without the Commodore range of machines, so I left and worked as a manager in a local tool hire shop, then I moved to my current position as a System support engineer looking after computer/printer/network and server problems. If it wasn't for Commodore some things would have never happened. Using the Commodore range of machines meant I was 1 step ahead when it came to using machines and finding problems.
Here are some of the things Commodore gave me

1 Fun
2 my career progression
3 lots of friends, because I loved computers and so did they
4 music
5 fascination about how machines work

So it's a Bye Jack
and thanks for all of the above, you will be very sadly missed! Most of All thanks for bringing Computers to the masses and not just the classes and of course thanks for Commodore and your visions of what could be.

**Nigel**
**Commodore Free (Editor)**

**On the Way to Fun:
An Emotion-Based Approach to
Successful Game Design**

**Roberto Dillon (Author)**

**Product details
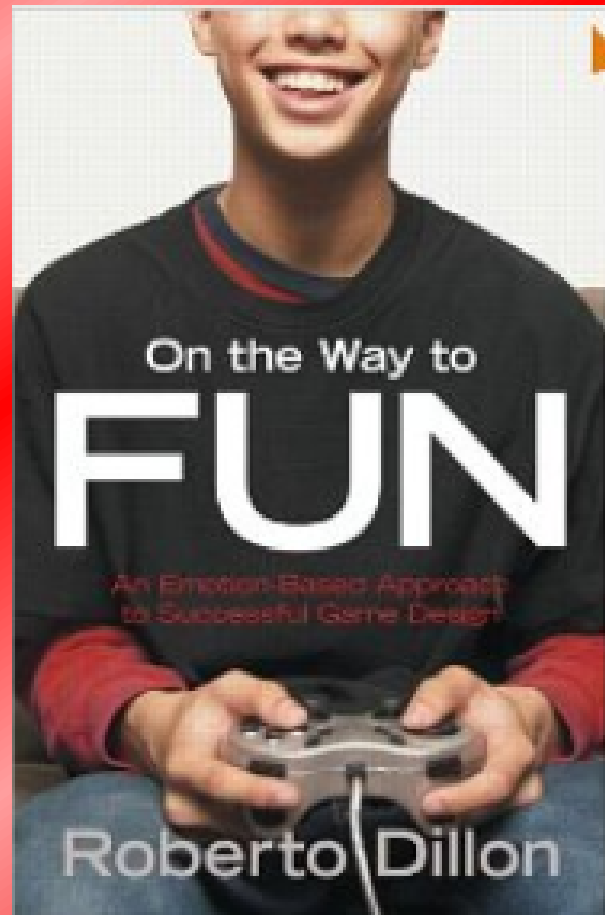Paperback: 200 pages
Publisher: A K Peters/CRC Press
(8 Mar 2010)
Language English
ISBN-10: 1568815824
ISBN-13: 978-1568815824**

**Golden Age of Video Games:
The Birth of a Multi billion
Dollar Industry [Paperback]**

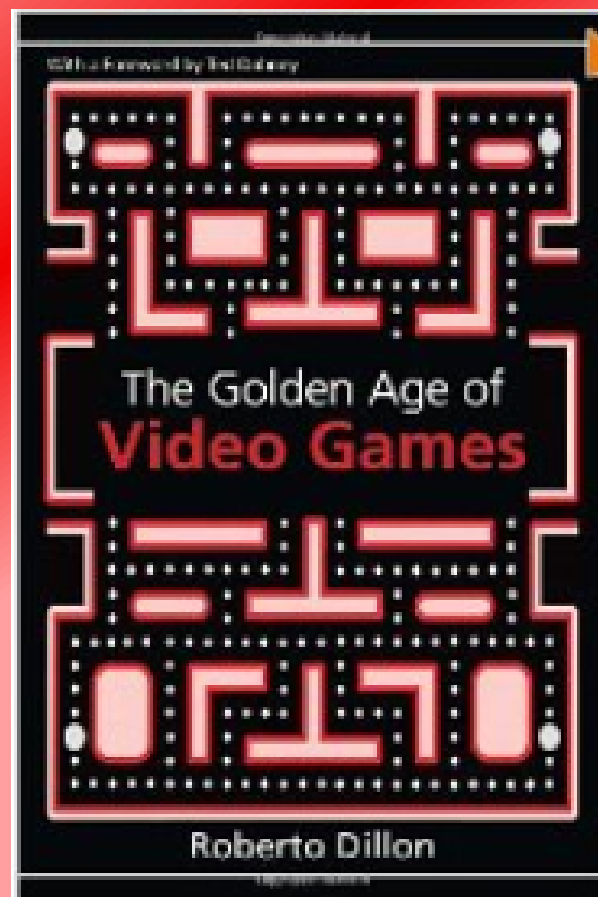**Roberto Dillon (Author)**

**Product details
Paperback: 209 pages
Publisher: A K Peters/CRC
Press; 1 edition (28 April 2011)
Language English
ISBN-10: 1439873232
ISBN-13: 978-1439873236**

# You Don't Know Jack!
# Interview with Jack Tramiel
## Reprinted with permission from  www.commodore.ca

**Presenter:** I don't know if you like computers or if you use computers but computers are so much a part of life that it's hard to resist them I think and Commodore 64 is still the best selling home computer in Australia, in fact the whole world, and my next guest is the man who created the Commodore 64,

**Jack Tramiel**. He's now heading another computer company which is called Atari, he actually bought up the company in 1984 and we'll learn about that in a moment, but when he took over Atari the company was said to be losing an enormous amount of money every day, but within only a couple of years it was out of the red, into the black, and Atari is churning out more than 70-million dollars worth of profit these days, which is pretty healthy indeed.
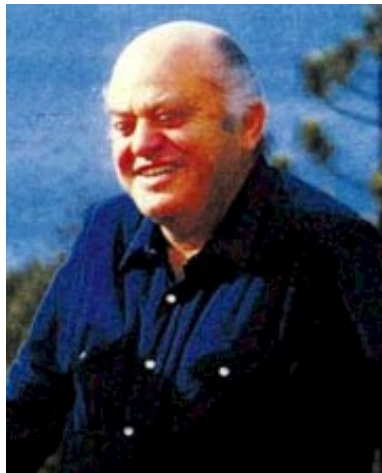
**Jack Tramiel** is here in Australia to open the PC '89 exhibition at Darling Habour which starts today, and he's given us a little bit of time in the studio this morning. Good morning.

**Jack Tramiel:** Good morning.

**Presenter:** We work our guests hard when they come to Australia and you're finding that out, aren't you?

**Jack Tramiel**: I do yes.

**Presenter:** You've had a very busy time.

**Jack Tramiel:** But I'm enjoying it.

**Presenter:** Are you?

**Jack Tramiel:** Yes.

**Presenter:** Well I'm glad to hear that. In fact we've got a line of Americans in this hour of the program which is interesting because it's a lot of people from your part of the world making their way to our shores. Commodore 64 is big here, it's big just about everywhere isn't it?

**Jack Tramiel:** Yes it is.

**Presenter**: Did you really invent it?

**Jack Tramiel**: No I'm not the engineer. I'm a businessman, but I do know what the public wants and I know how to bring technology and people together. And by living in Silicon Valley where the technology was born I know it's available, I know how to bring like I said people together and sell volume to bring the price down that the average person can afford to buy. In this we've made the 64 so successful.

**Presenter**: Commodore started out as a fairly small company I think didn't it? What did they start out as?

**Jack Tramiel**: Well I founded Commodore and that was in 1955 in Toronto Canada.

**Presenter**: Where did you get the name from?

**Jack Tramiel**: From the back of a Opal Commodore of a car.

**Presenter:** It was as ordinary as that was it?

**Jack Tramiel**: I was sitting in a taxi cab and I was trying to get the name for the company which I was building, and I was really looking to make it call it General, I'd just come out of the army and I was in the army for three years and seven months, so I was looking for something strong, so I was looking for a name like General which I couldn't get because it was taken. Then I was looking for a name like Admiral, and that was taken, and as I was talking to a friend of mine in the cab right in front of me this car with the name Commodore. I said well let's try this one.

**Presenter**: And that was it.

**Jack Tramiel**: And that was it.

**Presenter:** What a wonderful story, I love it, I love it.  We're back in 1955, let's go a little bit further back, life started for you in Poland, is that right?

**Jack Tramiel:** Yes I was born in Poland, I was born in 1928, in 1939 the war started and that's the time when I to a certain extent left Poland. Auschwitz was still in Poland but it was not Poland for me.

**Presenter:** When you were in Auschwitz.

**Jack Tramiel:** Yes.

**Presenter**: For how long?

www.commodorefree.com

**Jack Tramiel:** I was in the camps altogether for five years and a few months.

**Presenter**: And then America?

**Jack Tramiel:** Then I spent two years in Germany from April 10th 1945 till November 19th 1947. In November 1947 I left Germany and went to the United States.

**Presenter:** Did you have a lot of money?

**Jack Tramiel:** I had absolutely no money at all when I arrived in the States, I'm Jewish, that's the reason I was in camp, and a Jewish organisation paid for my ticket and they also gave me 10 dollars spending money. And when I arrived I was in a hostel like which was done by the Jewish Immigration Association and for three weeks I had to find my own way and I started to work for whatever job I could find. But when I did arrive in New York I did not believe that I'm in the United States.

**Presenter:** Why?

**Jack Tramiel:** Because it was just like being back in Poland, same language, the area I was in it was lots of immigrants and it had the same smell of pickles and of herring and all that which was very nice but this is not what I came for. And I made a decision that the United States was extremely good to me, I was liberated by the Americans and I felt I wanted to learn more about America so I joined the army.

**Presenter**: Where did you fight, or did you not fight? Did you go to Korea?

**Jack Tramiel**: No I didn't go anywhere.

**Presenter**: You just joined the army.

**Jack Tramiel:** I joined the army and it did me a lot of good to learn all about America because it was a peopledom. Washington State, from California, from Texas and from New York and black and green and white - all kinds of different people, and I found that America is not New York City only, there is much more to it.

Then I left the army and after two years and the Korean War started and I was recalled, but I was still lucky that I was not shipped to the front and there the second time around I made a decision that I'd better learn a trade, and the army gave that opportunity to start repairing office equipment like typewriters and adding machines etc. And before I left I was in charge of the First Army Office Equipment Repair Department which we had something like 25-thousand pieces of equipment in there for repairs. And when I left the army I actually continued working in the same field.

**Presenter**: And that was the genesis of your interest in computers.

**Jack Tramiel:** Correct. And so during the day I was working in an office equipment repair shop, at night I was driving a cab to be able to feed my family and after a while I decided I'd better use my allowance which I received from the United States Army I was entitled to borrow 25-thousand dollars from the bank with a government guarantee. And I actually took that money and I started my own business, my own little shop. And after I have done that I found that New York City in which I was in the Bronx, it was just a little too big, people were too smart and 25-thousand dollars which is not enough. And my wife had lots of family, she's also a survivor, also from Poland, and she had a lot of family in Toronto and we used to go there every once in a while, so we decided to move to Toronto. And there I repaired again the typewriters and adding machines in a company by the name of Sears Roebuck liked my services and they asked me if I'd possibly could find a way how I could assembly a typewriter for them. Being young enough and foolish enough I figure it's an easy task, as long as you have money you can do almost anything. Well I get 176-thousand dollar loan from Sears and I started to try to find a license to build typewriters. Well no American or West European country or company wanted to give me a license, so I wound up getting a license from Czechoslovakia.

And I actually brought 50 technicians over the counter, we started building typewriters. And we built so many that we could not sell them all in Canada and I had to start exporting them back to the States and that's the way Commodore started.

**Presenter**: That's a fantastic story and it tells me because you haven't said this, but it tells me that you're a man of great determination, that you don't think that any obstacle can get in your way. Do you feel that?

**Jack Tramiel**: I don't feel that. When I look back you know there must be something you know. In the camp that I told you I was in there was 10-thousand people in 1944, and we moved from Auschwitz to Hanover and when the war came to an end of these 10-thousand there was only 60 left.

**Presenter**: 60 individuals?

**Jack Tramiel:** 60 individuals from 10-thousand people. I was one of those 60. So from there on nothing was difficult to me.

**Presenter**: No, having survived that.

**Jack Tramiel**: Right. So I believe when a person has a goal, when a person is willing to work hard, the person does not want to become rich the same day but he looks at it in the long term, he can make it. The key is to give first and receive after. We live in a society today that's just the opposite.

**Presenter**: Everybody wants to take, take, take don't they.

**Jack Tramiel**: Right now.

**Presenter**: Instant gratification.

Page 42

**Jack Tramiel:** But if you are willing to invest your time, your effort and you're willing to serve society, society will reward you in time.
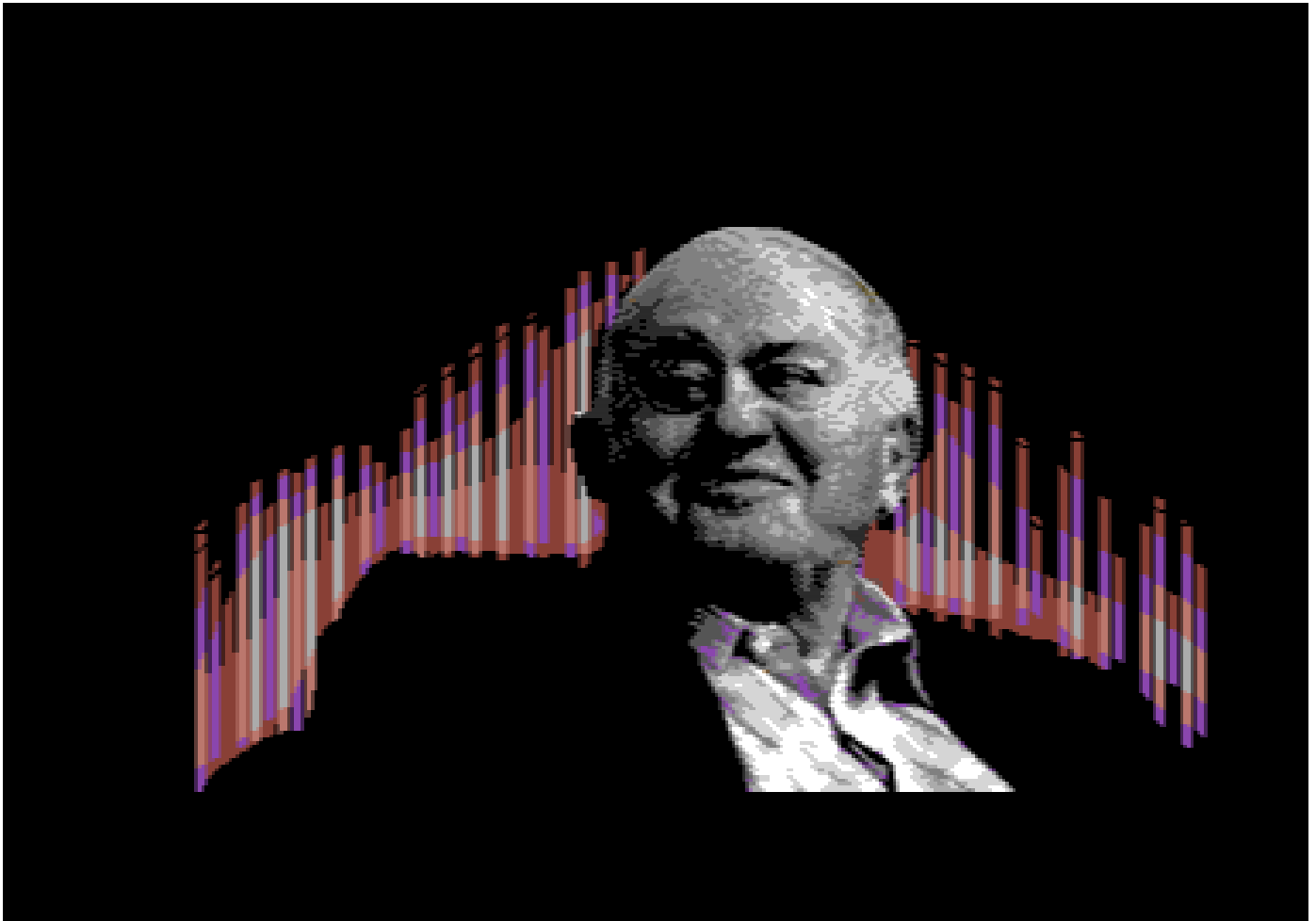
**Presenter**: Extraordinary, an amazing story. And you've got a very happy face. Are you a happy person do you think?

**Jack Tramiel**: Well a very happy person because I, I'm just looking at that in 1945 I was reborn, I don't look back, I do remember but I don't have any hate in me. I have built a company, I have built a family, I have three sons and four grandchildren and they all know about my background and about success and they're all working together with me, my three sons are part of my company, and we are very happy with what we've accomplished.

**Presenter**: Well welcome to our country, I hope you have a happy stay, and I hope they don't work you too hard, not in this humidity anyway.

**Jack Tramiel**: Thank you

# http://www.commodore.ca/



# Business Is a War [2012]
http://noname.c64.org/csdb/release/?id=108403&rss

# Jack Tramiel, a man who changed the world.
## By Robert Bernardo

The evening was over, and Jack was leaving, but as he left, several people were asking him for his autograph. Heck! I wasn't expecting that. I scrambled to the other side of the Computer History Museum, to the table where I had set up a Commodore 64 display in honor of the 25th Anniversary of the Commodore 64. I disconnected the brown C64 off that set-up and rushed back to where Jack was. He was walking out, surrounded by friends and fans.

I rushed up to him, "Mr. Tramiel, could you sign my C64?" He graciously smiled at me, took my pen, and autographed the top of the Commodore. "Thank you, Mr. Tramiel." He and his retinue then continued walking out the door with me watching as they left.

Later, CBM engineer Bil Herd, who also had attended the event, sent me an e-mail remarking, "It was a special night… very rare…". History had been made. Jack Tramiel, the leader of Commodore Business Machines, had made a rare appearance and had spoken at length about Commodore. The auditorium had been packed with his family members, former employees, industry leaders, news people, and fans like me, all there to honor Jack Tramiel and Commodore. There was much love and respect shown to the man… standing ovations… plenty of laughter and smiles… everybody on the edge of their seats to hear his words.

And I thought that there would be another such celebration for Jack and Commodore. After that December night in 2007, I regularly checked the Computer History Museum's website to see if they would have another Commodore night, another appearance of Jack. Nothing. Every year in the planning of the Commodore Vegas Expo, I would create the guest list, and our treasurer, Dick Estel, one time flew the idea about inviting Jack. How could I, a measly person, do such a thing, invite such a legendary character? Dick said it wouldn't hurt to try. I never tried.

With the release of Brian Bagnall's expanded edition, "Commodore: A Company on the Edge", in 2011, I was anticipating the 30th anniversary of the C64 in 2012. I was hoping that Jack would make an appearance again and speak about Commodore, and we would honor his accomplishments. We would laugh and smile and listen to his words again.

On Monday evening, April 9, I was on Commodore business; I was at Duncan MacDougall's house in Santa Clara in order to pick up a repaired Amiga 3000 computer. I waited in my car awhile, and finally he showed up. He opened up his garage workshop, and as he dug out the A3000, he said, "You know about Jack Tramiel dying?" Jack had died the day before.

I was shocked. I was dumbfounded. Forget about any repaired machine. Duncan said, "I thought he'd always be around." And I agreed… I thought that Jack would always be around, in the background, watching over his Commodore creations and his fans.

Duncan went on to talk about the repairs he had made to the A3000 and about other hardware matters, but Jack's death weighed heavily on me. I couldn't get over it. I left Santa Clara for the long drive back to the valley, and I was thinking about Jack. For the next 5 days, I thought about Jack, and I read all the news and talk about his passing. Whereas others were very vocal, I didn't say much on-line. I suppose it was my own form of mourning his death.

Whether as a hobby or in my work or in my social life, Commodore had been a significant part of my life for nearly 30 years. Hence, Jack had been a part of my life for nearly 30 years.

At the end of mourning comes acceptance. I've decided that Jack should be honored. The Commodore Vegas Expo 2012 will be devoted to his memory. My video of him at the 25th Anniversary of the Commodore 64 will be shown, Larry Anderson is planning a Jack Tramiel-labeled wine bottle, and photos of Jack will be distributed to each attendee. The 30th anniversary of the C64 will be a celebration of Jack, a man who developed computers that affected my life, that affected the lives of millions of others, that affected the development of all future computers. Jack Tramiel, a man who changed the world.



Jack Tramiel, founder of Commodore International and former CEO of Atari International, died on Sunday 8th April at the age of 83

**Computers "for the masses, not the classes"**

# Issue 61
# 2012

**Editor**
**Nigel Parker**

**Spell Checking**
**Peter Badrick**

**Text & HTML Conversion**
**Paul Davis**

**D64 Disk Image**
**Al Jackson**

**PDF Design**
**Nigel Parker**

**Website**
**www.commodorefree.com**
**Email Address**
**commodorefree@commodorefree.com**

**Submissions**
Articles are always wanted for the magazine. Contact
us for details .We can't pay you for your efforts but you are safe in the knowledge that
you have passed on details that will interest other Commodore enthusiasts.

**Notices**