DM 9.17 · Euro 4.69

GO64?!

HARDWARE

TIPPS ZU SCHWACHEN C128D NETZTEILEN

SZENE

COMMODORE SCAPE - RETRO-SPEKTIVE ZUR COMMODORE EXPO IN DEN USA

GEOS

MIT GEOS ONLINE: GEOTERM, GEOFAX UND THE WAVE FÜR WHEELS - DER NEUE WEBBROWSER!

STARTERS

DAS MYSTERIUM DER FLOPPY-KOMMANDOS GELÜFTET

ONLINE

DFÜ MIT DEM COMMODORE 64

**AUSFÜHRLICHE GRUNDLAGEN
CHATS, FOREN, WEB
E-MAIL ZUGANG ÜBER BOARDS**

**GRAFIKKURS - DIE PERSPEK-TIVE MACHT ES MÖGLICH
GODOT: KONVERTIE-RUNG NACH FLI**

CONTENTS

- December 1999 -

7

Online

Online with your C64? Of course! Read our report featuring new projects and established software solutions for your beloved Commodore and learn about the use of boards and some of the possibilities to enter the Internet.

20

Graphic-Tutorial

Three dimensional pictures on the C64; It's just a question of the proper use of colors and vanishing points. Learn more about it in part three of our tutorial, starting at page 20!
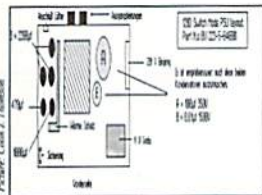
24

Commodore Expo in the USA

The scene is alive! Not only in Europe, but also in the US, many highlights are being introduced to all who wish to attend. Read our retrospective about two expos and their highlights on page 24.

26

PSU problems?

Have you seen mysterious character effects in C64- and C128-mode of your C128D when equipped with a SuperCPU? Well, either you are watching a demo, or your power supply unit is getting weaker and weaker. We will show you how to tune and tweak your PSU like Tim Taylor would. The grunting starts at page 26!

A

Graphics Tutorial - Files
How to GoDot - Files
Easter Sunday (USR)
DesTerm

B

Novaterm 9.5
The Wave

Into retirement with the C64

Greetings,

Hard times have come for the gamers among us: New releases have become rare. Although most games show a high standard of quality, they're not sufficient joystick fodder for a real player - there are simply too few.

You just too easily forget all the games getting dusty in your own diskbox. Many of you have collected lots and lots of games through the years, but never really spent any significant time with them. But, is there anything more enjoyable than sitting down on a cosy winter evening to have a closer look at all those great pieces of software you never had time for? There are thousands of games for the C64, with many a pearl among them. Games with timeless concepts and innovative designs you hardly find on today's PC or even the Playstation. The only innovation today is in graphics, to sell off the newest Command&Conquer clone, and raise the producers' profite margin.

One important thing to note is that people on the C64 were mostly enthusiasts. People like Manfred Trenz were writing games that are still without competition on the PC today. Even today, there are still people with visions, realizing their concepts on the machine that reached the masses, making computers commonly presentable.

In the future, along with the new releases, we will continue to present you aged games that didn't meet due attention at the time of their release - often without reason, as we find. Some classics will be presented as well, especially ones that are nearly forgotten, but still worth a play.

With this in mind, it surely won't be as hard to wait for new games. After all, a good thing needs its time, and some new hit games are in the making already - just have a look at the news column.

Recently at a meeting, a first hour's enthusiast who's not active anymore today, told me, "I'll never give my C64 away. Right now, I'm already looking forward to trying out all those games when I'm retired."

Why wait until then? Get to your C64 today and see what you missed!

Helmut Bieg and Volker Rust

News Flash

News from D2-SMS-Service

The BTX offer from Matting will go into effect on 12/31/99, and so the SMS offer from Mannesmann is over. Likewise, FUN will end its BTX-EMail-Gateway: Email from Telekom is not available with Mega-Online or the BTX-Module. Wolfgang Grimm is not planning any future development of his BTX-Decoder "Mega-Online". *(st)*

Year 2000 Patches

Werner Weicht is planning an adaptation of GeosLQ and TextPrint for MegaPatch 3. Additional information will come shortly. *(st)*

RE-System for the REU

Johannes Schultze-Oechtering has released some new software for his RE-System for the REU. The software is a fast access system for programs which reside in the REU. After starting the server program at the press of a key, all programs are ready to be executed. Up until now it was easy to forget the hotkey list of the 32 programs which could be stored; the serverscreen solves this problem nicely with an onscreen list. Work is being done to create a version of the RE-System that will work with the CP/M operating system. *(st)*

Disk Contents For GO64! Disks

Peter Karlsson, from Sweden, has drawn up a list of the contents of the GO64! coverdisks, complete with a table of contents and illustrated cover. They are in postscript format and are available for free download from:

http://www.softwolves.pp.se/cbm/c64-sleeves/
(st)

New Literature Concerning Computer Hackers

Due to the success of the book "Hackerland - Das Logbuch der Szene" (Hackerland - The Logbook of the Scene) Denis Moschitto and Evrim Sen are in the process of writing the sequel, "Hackertales", which will contain stories from the lives of computer hackers. In contrast to the purely factual "Hackerland", the sequel presents the information in a both a literary and sensational way. In this book, the authors present tales of their own life experiences as well as what they have achieved legitimately. *(st)*

Correction

The given telephon number in issue 11/1999 of Oliver Biasin (Commodore Connection Line) is wrong. The correct address must be:

Oliver Biasin
Schwalbenberg 15
91220 Schnaittach
Germany
Tel/FAX +49 9153 - 1831
Connection-Line@gmx.de
http://home.t-online.de/home/mysoft/ccl.htm
(gb)

HELLO WORLD,

just what is really the difference between a C64 and a modern computer system? At first sight, I admit that this may seem an odd question to ask. After all, everyone knows that even today's tiny PDAs (also called palmtops) pack more computing power than our breadbox. However, let's not be fooled by bare MHz numbers, for here, too, the saying 'not all that glitters is gold' holds true. In spite of an identical clock rate, AMD's K6 processors will always be slower under certain circumstances than Pentium IIs at the same clock rate. The secret behind this is the mathematical co-processor, also known as the FPU (Floating Point Unit) - which is exactly what I would like to discuss in this issue's World Watch.

FPU - What's that for?

Let's have a look at one of today's typical computer games. With all that technical brilliance, a game is still nothing but mathematical battles being waged - vector geometrics, ray tracing, light diffraction, and many others. These effects require a vast number of multiplications and divisions every second. Things really get complicated when objects are to be rotated, since the trigonometrical functions (sine, cosine, tangent) then also join the party. The fun really starts when you also have a high resolution. In this case, you have to be really precise; otherwise truncation errors will inevitably catch the viewer's attention.

But it doesn't always have to be one of the current high-tech games. What is the first thing a demo program writes into the memory? Correct - a sine table. What really eats computing time when doing vector geometry? The multiplications. On the other hand, an 8x8 bit multiplication table would be too big (64KB) to be kept in the RAM. That should be reason enough to have a look at where and how a mathematical co-

processor can be attached to the C64.

Major Problems

The heart of the C64 is the 6510, a descendant of the 6502 processor family. Commodore really loved these processors, and this was because they had, together with MOS Technologies, the possibility to produce these processors in ever new variations (6502, 6504, 6510, 7501, 8501). What made this processor really appealing in those days was the fact that creating a multi-processor system with it was very easy. This is because the 6502 is a quite frugal processor. It accesses the bus only during half a cycle. Commodore made use of this in their IEEE disk drives, in which, roughly put, one processor handles hardware control while the other does the communications. They also took advantage of it in their computers, so that, as is commonly known, the VIC-II is active during the half of the cycle the 6502 doesn't use.

That's quite good. However, the 6502 lacks an interface for connecting a co-processor, like the one the 68000 (Amiga) has. Those who believe that the 68000 is that much older are mistaken; actually, both are but a few years apart. Though they are almost the same age, a 68000 has the opcodes to include co-processors while a 6502 doesn't have such opcodes. Consequently, a co-processor can't be accessed using reserved opcodes. But is this true?

Sick Minds also Brood...

...and sometimes even get results. Let's imagine there is some circuitry (pre-processor) sitting between CPU and memory and taking in the CPU's reading orders to the memory. If the memory returns a valid 6502 opcode (before everyone starts shouting for LAX - globally accepted illegal opcodes are also valid, of course), this will be sent to the CPU. If that is not the case, a FPU of whatever kind will be given that command while the CPU receives NOPs instead, during the FPU command's execution.

Certainly a good idea, but there are dozens of problems connected to it. The pre-processor has to be able to distinguish between an opcode fetch and a data byte. It also needs to be faster than the CPU, because in that period of time in which the CPU simply wants to read the opcode, the pre-processor not only has to recognize this fact, but it also has to read the opcode and interpret it. The respective reply to the CPU has to be delivered just in time within first half of the cycle.

The advantage of this scheme is certainly the co-processor's transparent integration, that is, it is not taking up any memory and can be accessed directly by its own, discrete commands. I don't have to point out the catch, do I?

There has to be a Simpler Way!

And there is. What are the essential contents of such a co-processor? Tables, tables, and some more tables. Memory has become cheap these days; a single EPROM can contain up to 1 MB. So why should we keep being stingy about it? Considering this, we could have a little hardware module offer the following functions, for example:

- sine, cosine, and tangent, each normed to a certain range, as well as a domain of 0-255 instead of 0-1 or 0-359 degrees
- multiplication table 8x8 bit
- division table 8x8 bit
- addition and subtraction tables, which would allow, for instance, for an addition/subtraction like A=A+/-X via LDA mytab,X.

The only catch is that we have to access the EPROM data somehow. For simplicity's sake, we make use of the I/O area. Some logics are needed around it, but the means by which 64KB can be accessed using few registers has already been exemplified by the C128's VDC. We can use segment registers (has anybody said INTEL?). By means of a segment register, we pick the desired table, which is mapped into the memory one page at a time. Then the desired data is accessed using simple indexed addressing. In order to

access several 64KB address areas (the multiplication table needs 64KB all by itself), we need a second segment register. Too complicated? Let's have another look at a graphic which describes the tables' memory integration into the C64 memory map:

And Who Cares?

That is the basic question. Is something like this really of interest for the C64 user? Of course, such a module would be useful only if the software exploits it. This is why I would like to ask here if there is interest in such a module. This is just a conceptual sketch at the moment, but it certainly has potential for development... if there is interest.

Therefore, call! Send me your opinion via e-mail (**rainer@buchty.net**)

Adress	Register	Function
$de00	segment register 1	selection of functions collection
$de01	segment register 2	selection of function
$df00-$dfff	data area f(x)	

Or as code:

```
CLR $de00    ; select SIN/COS/TAN
CLR $de01    ; select SIN
LDX #$40
LDA $df00,X  ; 'compute' the sine of 90 degrees
```

Alternatively for multiplication:

```
LDA #$01     ; select multiplication
STA $de00

LDA #factor1 ; do multiplication
LDX #factor2
STA $de01
LDA $df00,X
```

We want our email!

Put radically, this could be the motto of our campaign. We the staff of the GO64!/Commodore World Magazine are planning to improve communication between all C64 fans, as nowadays very few still have a next-door neighbor who has got a C64, too. "Swapping" C64 software by way of diskette letters has lost much of its attraction due to a lack of new publications. The electronic letters that were written with Notemaker and added to the software on the discs are a complete waste of postage if you don't have any software to send them along with.

One way of getting into contact again is to communicate via e-mail, but now the last possibility of sending e-mails with BTX, which could be used by C64, has expired in Germany.

The reason is that #Fun*, the only BTX provider allowing this option has announced its retreat from business. In the US, Genie recently closed their e-mail accounts. PPP dialers and TCP-IP stacks for C64s are not available yet, so it is impossible to log on at many providers. The provider would need to offer one of the few UNIX shell accounts that can hardly be found anymore.

But there is in fact a way out of this dilemma: Some mailbox providers (BBS, 'Bulletin Board System') are able to offer e-mail accounts at their mailbox. This also becomes increasingly interesting for German users now that the fees for long-distance calls have reached a quite acceptable level. Moreover, many BBSystems offer a wide range of software, which can be

downloaded for little or no money. We contacted some sysops (System Operators, i.e. mailbox providers) and tried to find out whether they have e-mail accounts. Until now, two answers have reached us.

We are aware of the fact that this a bit meager. Therefore we would like to ask all sysops offering e-mail accounts to contact us and support our campaign.

Here are the addresses of René Schwarte's and John Guillory's BBSs, followed by the necessary technical information. *(wg)*

GERMANY			
Telephone Number*	+49-(0)341-2613589	Comments	FIDO 2:249/1210 - Online 24 hrs except 1:30 am - 4:00 am MEZ.
Location	Leipzig, Germany		Email access at SachsenGEOS-BBS costs 3,- DM per month.
BBS Name	SachsenGEOS		
BBS Software	Remote Access 2.2+		
BBS Platform/Processor	OS/2 Warp3 / AMD K6 2/ 333	*In Germany, dial 0-341-2613589. When you are abroad leave out the initial 'O'. '49' is the country area code you have to dial from abroad in order to be able to call Germany. '+' signifies the special combination of numbers you have to dial to make an international call from your country. In many countries this combination is different, in Germany e.g. it would be '00'.	
Graphics Support	ANSI, ASCII		
Commodore Support	Messages: Fidonet, GEOSNet, InterNet, CBMNet, VirNet Files: GEOS Files, Emulator Files, CBM-Files, VirFiles		
System Operator Name	René Schwarte (Schwarte@SachsenGEOS.FIDO.DE)		
System Operator Handle	Rene Schwarte		
Modem Speed/Settings	300-33.6K, V90, K56Flex		
WWW Presence	None		
Open Status Verified	Nikolaus Metz		

UNITED STATES			
Telephone Number	318-431-7189	BBS Platform/Processor	PC/486 DX4/100
Location	Westlake, LA	Graphics Support	Unknown
BBS Name	Mainline Consulting	Commodore Support	Files: Commodore 64, Commodore 128, CP/M Echo-Mail: CBM, CBM128, CBMGEOS
BBS Software/Networks	Wildcat 4.0/Fidonet: 1:19/11.0	System Operator Name	John H. Guillory (john.guillory@mainline.nu)
		System Operator Handle	John H. Guillory
		Maximum Modem Speed	Unknown
		WWW Presence	Unknown
		Open Status Verified	21 May 1999
		Comments	E-Mail available to anyone that logs on, their e-mail address is their login-name @mainline.nu Some usenet newsgroups are available.

Coverstory

Picture: Christoph Bach

Is BTX going to suffer the same fate as Qlink has?

There are a lot of rumors, as of late, that those using their c64 to go online via BTX, will very soon find their BTX cartridges and the respective software completely useless. It's about time we take a closer look at this issue, because the similarities to the online service Quantum Link (later known as AOL) are distressing.

by Wanja Gayk

Quantum Link and AOL

When a firm by the name "Playnet" released a c64 online-game system, it became the prime choice for Steve Case, who was then programming a mailbox network for c64 computers. In the end, he bought the firm, merging both parts into the online service Quantum Link ("QLink"). He gave away the access software for free, which led to a great number of users having a lot of fun for $10 US a month via QLink. On the QLink system you would find online games, information about Commodore computers, news, contact with other users (including live chat), courses in the so-called "Learning Center" and products in "The Mall".

With the decline of the Commodore market, access software was programmed for PC,
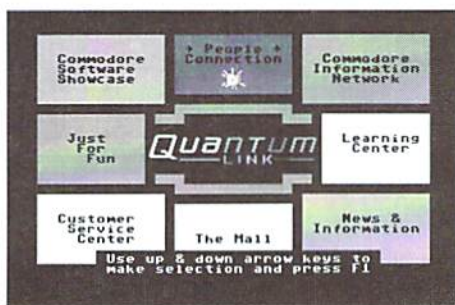
Apple II, and Macintosh, and the service was renamed AOL, or America Online. The c64 fans were left behind with access only to the old QLink system, because newer AOL software had not been programmed for the old breadbox. Later on, as AOL's online services were connected to the Internet, QLink service was discontinued without any warning, leaving the

c64 users - the very people that have allowed for Steve Case's success - stranded. Case had deserted them (s. also the box "last words").

Interestingly enough, some remnants of the QLink system, the limitation of user names to 10 characters for instance, were removed from the AOL software just a few years ago. This limitation had been introduced so that four names could be printed next to each other on the c64's 40-character lines. All that remains of those old QLink days now is the addresss of AOL's headquarters, and the fact that the access software is still given away for free.

BTX and T- Online

In 1979, a British Telekom engineer from research and design presented the ViewData system - he transferred text pages via telephone lines onto his screen. This was a technology which obviously inspired a certain Eric Danke, currently manager at T-Online, and making this one of the dates considered to be the beginning of BTX. There are however some who claim that 1978 ha d been the real birth of BTX, when a council of experts from all over Europe determined the CEPT standard, which is the foundation of the BTX system.

Whatever the case, under Eric Danke's supervision, a system was developed in the research labs of the post, which was introduced in Germany under the name "Bildschirmtext" (roughly translated, "screen text"), or BTX for short. You could access the BTX system via a settop box by the name "Multilink" and a tv set. For data transfer, Multilink initially used an integrated 1200 bps modem. The post then installed public Multilink BTX terminals in its branches. Soon thereafter, BTX cartridges, which allowed access to the BTX net via the telephone lines, became available for the c64 also.

The reason for BTX's success was in fact not only the technical team of the postal services,

QLink Shots by Jim Parkes (http://www2.airtel/home/jparkes)
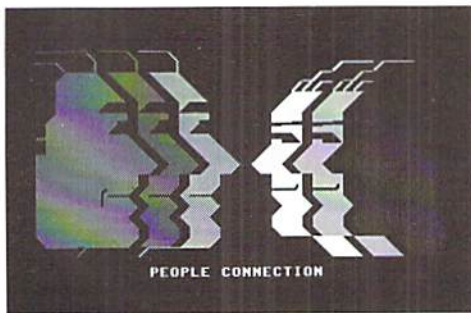
but even moreso mail-order stores like Quelle and Otto, who had been investigating the possibilities of electronic commerce since the beginning. Banks also showed great interest, and supported the BTX concept, having realized the advantages of home banking. Today this is commonly known everywhere by the nice name "Online Banking", and "E-Commerce", which is buying via the computer. Today, both are praised as revolutionary advancements.

Frightening Parallelisms

In 1995, BTX was finally linked to the Internet. The telecommuncations department of the former Deutsche Post AG had by then been partially privatized and was called "Telekom". It rechristened the emerging combination of Internet and BTX due to marketing reasons, so that BTX turned into "T-Online". The access computer was configured so that it could tell CEPT, SLIP and PPP access apart from each other automatically. Former BTX clients still had access to the good old CEPT standard, while T-Online customers had direct access to both the Internet and BTX via (C)SLIP at first, and then PPP. The old BTX address format has never been changed, so that in addition to the common internet URLs (http://... and so on), you can still find BTX addresses like #matting*. It's astonishing how strong the symbiosis between CEPT and PPP has become. For instance, bank services are still handled by T-Online using the CEPT standard, effectively becoming a part of the "old" BTX world.

Is BTX going to die, just as Quantum Link has?

Starting in January, BTX telebilling will be discontinued. Until now, telebilling has provided the ability to charge users the normal BTX service fees by simply adding to the users' Telekom telephone bill. From January

on, it will the BTX provider's problem to finance their pages by other means, for instance by advertisment. Because the fees of the Telekom are not going to change, some services are likely to be finished, considering that the C64 service "Brotkasten Corner" (#matting*, which was the last bastion of C64 services in BTX), Email services handled by "#fun*", and the SMS service of Mannesmann-D2, which handled sending text messages to cellular phones, have already announced their withdrawal. As far as Westdeutsche Fernsehen's "#wdr*" service is concerned, there hasn't been a decision yet. However, I'm quite certain a decision will be made by the time this issue of GO64! goes to print.

You shouldn't forget though, that even without c64 forums, BTX still offers a lot of information you can access. There is still online banking, although more and more banks are changing to the more secure HBCI Internet banking system, and telephone numbers for access via PPP (0191011) and CEPT (01910) have been separated from each other. Does this already signal the end? Representatives of Telekom have told us that there will be access to BTX as long as the old access software is employed, although they are planning to reduce the access capacities. This would seem to indicate that there is no end for German C64 users in sight, at least not yet.

One must consider though, whether the obviously reduced BTX service is really worth the price, especially since the announced price reductions of T-Online are only for users of the new software whilst the pure BTX customers still have to pay the old, higher prices. Wolfgang Grimm (Megacom) for one seems too pessimistic, judging from information in the Brotkasten-Corner stating that he is not going to continue his T-Online software any longer.

Dear Members,

As you know, QLink was originally launched in November, 1985. In the years that followed you, as our loyal members, have helped us build a unique online community for Commodore computer users. I want to thank each of you for your contribution, your support and your feedback over the years.

The computing industry has changed dramatically since those first days of online communications. Commodore ceased to produce Commodore brand computers in 1993. Sadly, the company has recently closed its doors entirely. The Commodore computer, once a leader in the industry, has been replaced by faster, more powerful systems. Many software vendors no longer support the Commodore operating system.

Now we find, with great regret, that we simply can no longer support the QLink service. It has become impossible for us to maintain the product up to a standard of quality that we can be proud of. Many of you I'm sure have noticed a diminished level of product quality in the last few months due to these technical limitations. Without technical support from the industry, we are not able to add new services, fix existing problems, or prevent new ones. Therefore we have made the sad decision to discontinue QLink as of November 1, 1994.

We would like to thank each of you for your long and continued support and, if at all possible, keep you as part of our online community.

If you now have the ability to use America Online (PC-DOS, Windows or Macintosh), we invite you to convert your membership to one of these other systems. For details on what these versions have to offer and the system requirements needed to run them, see the document in this area entitled "Converting to America Online."

For details on the last month of service for QLink, important dates and billing information, see the document in this area entitled "Your Final Bill."

We have enjoyed serving you. Thanks again.

Sincerely,
Steve Case
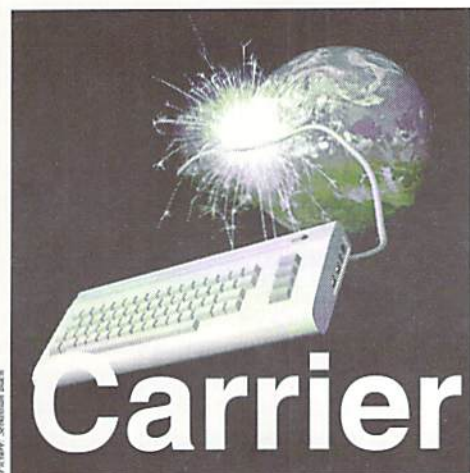
CarrierDetect

by Gaelyne R. Gasson

I'm Online. Now What?

One of the questions I'm often asked is "What can I do on the Internet?". This issue, we'll touch on a few online activities and we may even uncover some surprises along the way. Email addresses or links to web sites with more information are listed in the sidebar for your reference.

Email

Once you're online, you'll have an Email address that people can use to contact you, and you'll be able to send messages to others as well. Email is one of the easiest (and least expensive) ways to communicate with people - either nearby or on the other side of the planet.

Aside from sending messages to friends and family, Email can also be used for other things. For instance, on a daily basis, I receive news headlines, currency rates, an Email TV guide for the nights entertainment and "A Word A Day" that teaches me about new and different words and how to use them. It includes a separate "Quote of the Day" that's also usually thought provoking. The above are examples of "one-way" type mail lists, as they only send information to people. I also belong to several mail lists that could be considered as "virtual/ Email" communities. These allow me to converse with others interested in the same things. There are quite a few Commodore related mail lists, such as COMMODOR, Tifcu and some that are just for specific programs or hardware.

If you've never used Email before, I recommend using a program online named "pine". It uses menus that are easy to follow, and its features can be discovered a little at a time as you become more accustomed to it. One of the best tips I can pass along is when first trying out a new Email feature (or Email in general), you can send Email to yoursel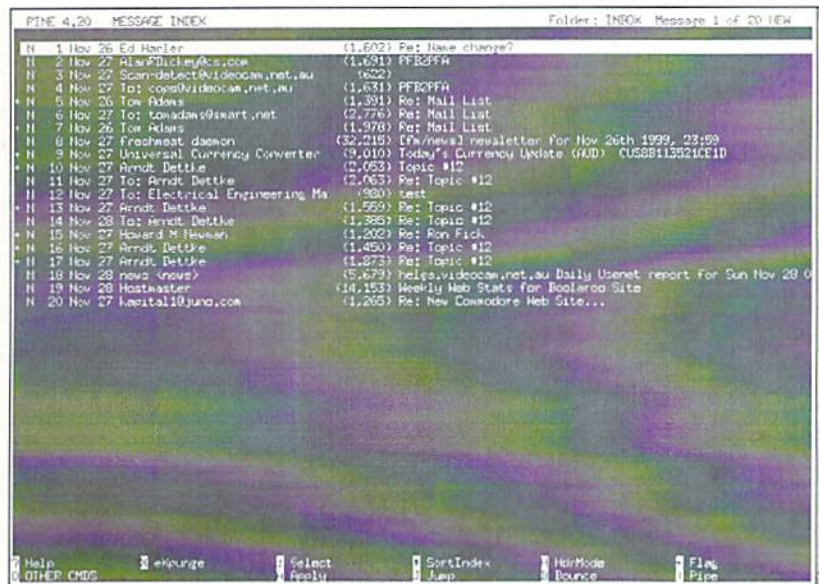f. This lets you see what your messages will look like to others, and if you've made any errors, only you will know about them.

Newsgroups

Similar in some ways to Email (especially mail lists), Usenet news, or "newsgroups" are message areas that cover a specific topic. There are thousands of newsgroup topics to suit everyone's interest. One of the common misconceptions I've run into is that newsgroups are either all about either computers or sex. Well, there's that, but there are also a great many newsgroups covering things like cooking, needle craft, gardening, self help as well as other (general or specific) interest topics.

Some newsgroups are moderated, but most allow anyone to post messages in them. One might think that an argument in a newsgroup can be a bad thing, but it can also be beneficial as it can give a better idea of the people involved in a newsgroup. For instance, if there's someone who's always argumentative or insulting, you get to know that this is part of this person's personality, and if the group were moderated, you may never see this aspect.

Usually to read news you would use a newsreader type program such as "tin" or "nn". However, some versions of Pine (the Email program mentioned above) can also be used for reading newsgroups, and if your Internet Provider has Lynx online for web browsing (discussed below), you may also be able to use it to access newsgroups. If your provider

EMail with Pine

Newsgroups with Lynx

A chat session with IRC, Talk and ICQ

doesn't offer news, you can even use the Web and visit either Deja News or Supernews to read newsgroups.

When new to following a newsgroup, it's best to start out by only reading the posts. This gives you a feel for what the group is like. Later, once you know more about the topics discussed you can post messages with confidence. Although not everyone would agree, I feel it's best to use your best behaviour (and grammar skills) when you post newsgroup articles. Some places, such as Deja News, archive newsgroup messages so what you type in a newsgroup today could be around in archives for years to come.

Chat

Sometimes instead of writing long messages, you might like to "talk" with others online 'in real time'. While I often share Email with my children (who live in America), they much prefer when we are online at the same time and can chat with each other - responding to comments immediately rather than hours later. Sometimes it's more fun too. Other times you might like to get help right away with a problem you're having online, and using chat programs such as IRC (Internet Relay Chat), talk or Icq (as discussed in Issue #X), can get you to a solution much faster than waiting for an answer in Email.

IRC (Internet Relay Chat) is like an Internet version of CB radio. There are a number of different servers that act as relayers, and there are hundreds (thousands?) of different channels of varying topics. Each server has different channels. IRC is a very large topic and the best source of information on how to use it can be found in the "IRC Primer" (check the sidebar).

Talk is a Unix utility that lets you "talk" to another user who's online at the same time, provided they are using a Unix shell account.

You can use it by typing "talk" followed by the Email address of the person you'd like to chat with. For example: talk mailaddress@isp.com. The talk program will send a message that will appear on your friend's screen telling them you'd like to talk with them, and to respond, type "talk youraddress@another.isp.com". Once the two of you are using the talk program, your screen will be split and you will see your friend's typing in one section of the screen, and what you type will be displayed in the bottom section.

ICQ allows you to send short notes to other people that also use ICQ. If the other person is online and is using their ICQ program, they'll receive the note immediately. Otherwise, they'll receive the note the next time they use ICQ. Many people use ICQ, as it's a nice cross between online chatting in real time and sending Email messages. One text version of ICQ that was discussed in Issue #X is Micq. Others include CenterICQ and Zicq. These programs are probably not already installed on your Internet Provider's system, but you can ask for one to be installed for you.

Information and Fun (World Wide Web)

This afternoon, I searched the White Pages phone book for a friends phone number, looked at numerous pumpkin pie recipes, and then as the recipe gave oven temperatures in fahrenheit and our oven uses centigrade, I found and used a simple conversion utility online to find out what temperature to set the oven. This isn't a typical day (I don't bake very often), but I knew the quickest and easiest way to get the information I needed would be on the World Wide Web.

The Internet and the World Wide Web are not one and the same. You might have had this impression due to the news and other media

showing only web sites when they mention anything relating to the Internet, but the Web is only a part of the whole. There are thousands of individual web pages about the Commodore computer, and hundreds of different web sites that cater specifically to Commodore related topics. As with other areas of the Internet, there's at least one web page for any subject you can think of, and probably several thousand for subjects you haven't yet imagined. The web can be used for fun or for serious uses such as research or education. Something to keep in mind when doing research is to check a number of sources to make sure the information you've found is correct. There are a number of reasons information can be incorrect - a couple examples are personal bias, and inaccurate research.

When we use our Commodore computers online, we use a text-based web browser such as Lynx or W3M to view web pages. Usually, your Internet Provider will have Lynx available, either through a menu system or you can simply type the word "lynx" on the command line to start it. Another (highly recommended) web browser is W3M, as it will display web pages formatted somewhat better than Lynx. W3M was covered in depth in Issue #X.

Your Own Web Pages

Most Internet Providers allow their users to have their own personal web site online, and the great thing is, you can create your web pages offline using any program that will let you save your text in ASCII format, such as GEOS WrongIsWrite, The Write Stuff, Zed, etc. Your web page can be about anything you like - yourself, your family, your favourite computer, or your hobbies or even your bookmarks of favourite web sites.

Files

Some people enjoy acquiring new programs and files to use on their computer, whether it be games, utilities, graphics to view, or files to read. There are many different ways to get files online. One of the key methods is to use a program named FTP (File Transfer Protocol) or NCFPT and visit "FTP sites" that have files available to the public. When you access an FTP site, normally you don't download files directly to your computer. The FTP program transfers the file to your home directory on your Internet Provider and you then download it from there.

You can also visit an FTP site using a web browser such as Lynx and download directly to your disk drive if Lynx is configured to let you do this. One thing to keep in mind when retrieving files using Lynx is to press the "d" key when your cursor is highlighting a link. This tells Lynx to download the program

Web with Lynx

without making any changes to it.

Another way to receive files is by having a friend send them to you in Email. Normally, you would save the received file to your home directory from the Email program, and then when you exit the Email program, you download the file to your own computer.

Doing Stuff Online you can't do Offline

If you have a UNIX shell account, you can use many of the utilities that are online for your own purposes. For instance, there are two very powerful text editing utilities (Emacs and Vi) that are on most Unix systems that can be used to edit really large text files, and if you want to split text files into smaller sized files you can with a program named "split". Graphics can be converted from one format to another (e.g. Jpeg to GIF format), and there's a number of PostScript printing utilities you can take advantage of too. Usually there are two ways of finding out how to use a utility online. One way is to type "man name" ('name' would be the name of the utility you want to know about), or you can type either: "name -h" or "name --h". Sometimes if you just type the name by itself, it will tell you how to use it, but often times this will run the utility, and this may not be desirable.

A good way to find out about what utilities are at your disposal is by typing "apropos subject", so if you wanted to know about utilities that manipulate GIF files, you could type "apropos gif". A list of utilities will be displayed, with a description of what each does. If the list is long, it will fill the screen and you can press the spacebar to see the next page. Once you know the name of the utility that you want to use, you can then use the man command described above to find out more about it.

Telnet

What if you don't especially like your Internet Provider but they're the only one in your area that offer Shell accounts? If your provider has telnet available, you can use it to access another Internet Provider. You log into the other system

with a user name and password, and it's almost exactly as if you'd used your modem to dial directly, but instead, you've connected from your usual provider to another one across the Internet. There's not much to tell about telnet as once you've used it to connect to another system, you can do all the same things online as I've mentioned above, such as use Email, read newsgroups, chat online, etc.

Another article in this issue discusses Bulletin Board Systems (BBS's for short). One of the ways you can access a BBS when you have an Internet account is by way of using telnet. There are a couple really nice BBS systems that offer Commodore Fidonet echos (message bases, similar but in some ways nicer than newsgroups). I've listed two such BBS's and how to access them using telnet in the sidebar. VideoCam Services also offers telnet memberships to Commodore users.

Inter-Relationships

With many of the abilities mentioned above, you can use more than one Internet utility to access another one. For instance, there are places on the web such as FriendlyEmail where you can read and send Email online using the web instead of an Email program. Or, you can read newsgroups in a few different formats using a web browser or Email program such as Pine.

Some Surprises .....

Online Romances

I met my husband online in the Fidonet CBM echos and helped him develop his offline mail reader for the C128, later we used Email to send each other daily messages. Regardless of what the press would have you believe, online romances are not something new or only belonging to the realm of the Internet. I have heard of a few Commodore couples that met on Qlink too. It was never our intention to "meet someone special" online, and I have a feeling this is the way it should be. If you go online expecting to find Mr. or Ms. Right, it'll likely never happen. If you find someone you share common interests with, take things slowly.

World Perspectives

After you've become accustomed to using the Internet for awhile, your perspective on some things such as distances and the world as a whole may change. Last week when a construction worker cut through one of the major phone cables in our neighbourhood, about 200 customers lost use of their phones for most of a day. The experience caused me to think about what it is that I appreciate about having access to the rest of the world through our telephone lines. Without the use of the Internet, suddenly family in the USA seemed

very far away from us in Australia, and oddly enough, even the people who live in the next neighbourhood may as well be in another country.

Wrap Up

Back when I first got online, my interests were learning more about my computer and meeting new friends that also enjoyed using Commodore computers. Today, I live in another country, and together with my husband, we run a small Internet service. I'm still always looking for information on getting the most from my C128, and supporting other Commodore users with files and information. I've made many friends that also share my interests, and have even mourned the loss of Commodore friends that though I've never met them in person, their support and friendship have left lasting impressions.

This article is dedicated to the memory of Ron Fick, a CNet BBS operator and member of many user groups. With his assistance many user groups were able to have their own Bulletin Board Systems at little or no cost. His dedication and support will long be remembered. *(ad)*

Gaelyne Gasson is the author of "The Internet for Commodore C64/128 Users" and can be contacted at: gaelyne@videocam.net.au
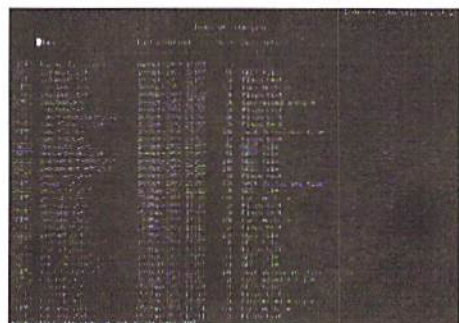
FTP Session with Lynx

IRC for real time chatting. I keep channel #tifcu going whenever I'm at the keyboard so I'm available for help if needed.

Djpeg converting JPEG graphics to GIFs
Cjpeg creating JPEG files from GIFs (C= GEOS users can use this with their postscript level printers)
Ghostscript A swag of graphics conversion utilities

We also have a pretty decently stocked C= FTP site. Lynx and Tin have been updated so they can be used in full colour, and Lynx is updated to allow access to secure web sites for ordering items or banking and such. Other shell utilities and programs are online, but I've listed the most used ones above.

In the newsgroups, we also have the Commodore related Fidonet echos: CBM, CBM-128, CBM-GEOS and CLASSIC_COMPUTER (all these are Fidonet but are in the newsgroup area with a "fido." prefix).

Members get their own home directory, Email address and Web URL (http://videocam.net.au/~yourname/), and a total of 1mb disk space to work in (more is available if needed).

We asking $30 Australian ($19.95 US) a year for membership, which basically helps to cover our expenses.

If you're interested, and like filling out forms, you can sign up from the web at: http://videocam.net.au/telnet.php3 or for secure ordering, it's https://secure.vcsweb.com/videocam/telnet.php3

Otherwise, you can send us a check and a letter - be sure to include what you'd like your login name to be and a password to use (up to 8 characters, preferably mixed with numbers for better security).

Our address is:

VideoCam Services
90 Hillier Rd
Reynella SA 5161
Australia

Phone: +61 8 8322-2716
Fax: +61 8 8387-5810
Fax: (707) 897-0676 (USA)

Email

Pine http://www.washington.edu/pine/index.html
Friendly Email http://friendlyemail.com

List of Commodore Related Mailing Lists:
ftp:/videocam.net.au/cbm/info-txt/cbm-mailing-lists-list.txt

Mailinglisten
NewsScan Daily
Send a blank message to NewsScan@NewsScan.com with 'subscribe' in the subject line.

AWAD (A Word A Day)
http://www.wordsmith.org/awad/subscribe.html
Send a blank message to wsmith@wordsmith.org with 'subscribe' in the subject line.

Tourbus (a twice a week Email on varying Internet topics)
http://www.TOURBUS.com
Send a message to LISTSERV@LISTSERV.AOL.COM and inthe body of the message type:
SUBSCRIBE TOURBUS Your Name

TIFCU (The Internet for Commodore Users) 1 or 2 times a week digest of Commodore - Internet related messages..
http://cbm.videocam.net.au/tifcu/mail_list/
Send a message to tifcu-request@videocam.net.au and in the subject AND body of the message type the word subscribe.

Newsgroups

What is Usenet? http://www.ualberta.ca/HELP/news/usenet.html
Supernews http://www.supernews.com
Deja News http://www.deja.com
tin http://gpu.srv.ualberta.ca/lynx/lynx/HELP/news/Tin_help.html
nn http://www.ii.com/internet/faqs/launchers/usenet/software/nn/getting-started/

Some Interesting Newsgroups
comp.internet.net-happenings
comp.sys.cbm
comp.binaries.cbm
alt.c64

Chat

IRC Primer http://www.irchelp.org
IRC How To http://cbm.videocam.net.au/irc-howto.html

IRC channels:
#c-64 IRCnet 24hrs a dayh
#c-64 Efnet 24hrs a day
(most active during N.American

afternoons)
#c64ntsc IRCnet Thursday Evenings 6pm PST
#c-net Dalnet Sundays starting at 3pm EST

Talk http://videocam.net.au/cgi-bin/man-cgi?talk
Micq http://phantom.iquest.net/micq/
CenterICQ http://konst.org.ua/software.centericq.html
Zicq http://www.asenteck.com/~zicq/

World Wide Web

lynx http://www.hicom.net/~oedipus/help.html
w3m http://ei5nazha.yz.yamagata-u.ac.jp/~aito/w3m/eng/

Your Own Web Pages
Anchors Away http://cbm.videocam.net.au/anchors.html
HTML Primer http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html

FTP

FTP Site-List http://www.wipd.com/snakeman/ftpsites.html
FTP How To http://cbm.videocam.net.au/ftp-howto.html

UNIX-Utilities
Referenz http://www.geek-girl.com/unix.html
Emacs http://www.geek-girl.com/emacs/emacs.html
Vi http://www.geek-girl.com/Unixhelp/vi/index.html
split http://videocam.net.au/cgi-bin/man-cgi?split
man http://videocam.net.au/cgi-bin/man-cgi?man
apropos http://videocam.net.au/cgi-bin/man-cgi?apropos

Telnet

Neverending BBS telnet bbs.neverending.com
Cereal Port BBS telnet 199.125.78.133
VideoCam Services Sign up at http://cbm.videocam.net.au/telnet.html

Other Web Sites of Interest
CBM Fidonet Echos http://cbm.videocam.net.au/cbmfido.html
White Pages http://www.whitepages.com/cgi/intl_sites.cgi
Recipes http://www.ichef.com
Temperature Conversion http://www.qsl.net/wb8tif/temp.htm

Sites with lists of C= Web Sites:
Cocos http://www.aniston-park.de/cocos/
Clinks http://www.cmdweb.com/cbm/clinks/

Against Oblivion - Save the Docs!

The market for new software for the Commodore 64, and other eight-bit computers, isn't what it used to be. Sure, programs are still developed, by devoted companies like CMD and others, but the majority of the programs that are available are quite old and aren't sold any longer. Fortunately, there are many places on the Internet where people collect old software for our veteran computers, like / Arnold/ for C64 games, but what about the documentation?

by Peter Karlsson

Preserving copies of the software is quite simple, you just need to copy the disk (or tape or cartridge, or whatever kind of medium it was originally stored on), but preserving documentation requires a bit more effort. The manuals are most often on paper, and paper isn't something that you easily can shuffle onto a hard disk. Also, paper tends to wear out over time, to completely disappear in the end, and it is not as simple as to copy it over to a fresh disk when the old one is near the end of its life-time.

So, what can be done about it? We could transfer the manuals to an electronic form, something that you can save on your hard-disk. This digital copy can then be downloaded and perhaps printed by those that have found the old software.

The problem is to reach out to others in a good, coordinated fashion once you have transfered the documents to an electronic format. Fortunately for us, such efforts exist, thanks to the hard work of several volunteering Internet users.

The first project of this kind that I heard of when it comes to the Commodore 64, and also the most well-known, is / Project 64/, founded in 1995 by Cris Berneburg in the USA. He started to collect English language Commodore 64 related documentation ranging from books to game manuals. Project 64 is currently being integrated into Dean Thompson's /Project Commodore/, expanding the project to carry documentation for all of Commodore's computers, not just the Commodore 64.

Project 64 and Project Commodore only handle documentations in English, but there are many other languages in which software for the Commodore computers where available. After giving it some consideration, I was convinced by Cris Berneburg to start up /International Project 64/ in March of 1997, covering the non-English part. In March of 1999, IP64 was reformed into /iDOC=/, International Documentation Project for Commodore Eight-Bit Computers (I intentionally chose a very long name so that people would stick to the abbreviation), broadening its scope to other Commodore eight-bitters, and to non text-only documentations as well. To date, iDOC= carries 57 documents in text-only form, in five different languages. Seven documents are available in non text-only form (mostly HTML with illustrations), one of which is not included in the above count.

Other computers are not left out in the cold, either. There is a sister project of Project 64 and

iDOC= called /Project 81/, which carries Sinclair ZX81 (Timex-1000) documentation. The page hasn't been updated since 1997, and there seems not to be a great interest in that project, however. For the Sinclair Spectrum (Timex-2000), there is /Spectrum Game Instructions/, a part of /Planet Sinclair/. For the Apple II there are the /Apple II archives/, documenting a lot of games, among other things, and a friend of mine has scanned the documentation he got with his Apple at / AppleII-iskt/.

If you have any old documentation that is not yet preserved in an electronic format, don't hesitate to scan it in and send it to a suitable project. Especially interesting is the non-English part of the world, since there are quite a number of hardware and software products produced for the Commodore machines outside of the English speaking territories. There are a number of such products that have never reached these areas. (ak)

Arnold: **http://arnold.c64.org/**
Project 64: **http://project64.c64.org/**
Project Commodore: **http:// members.xoom.com/advc/pcbm/**
iDOC: **http://www.softwolves.pp.se/idoc/**
Project 81: **http://www.hh.schule.de/hhs/ mjaap/project.htm**
Spectrum Game Instructions: **http:// www.nvg.ntnu.no/sinclair/instructions/**
Planet Sinclair: **http://www.nvg.ntnu.no/ sinclair/**
Apple II archives: **http:// www.apple.asimov.net/**
AppleII-iskt: **http://www.mds.mdh.se/ ~frv95pen/apple2/** (überwiegend schwedisch)

(ak)
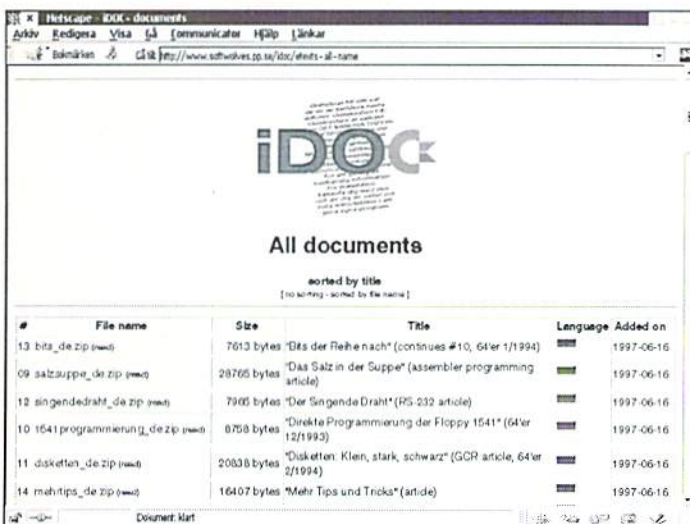
Remote Data Transmission under GEOS

Graphical Transfer

Normally, you'd think that remote data transmission and a user interface that slows a system exclude each other. Yet, there will be some choices for Wheels and MegaPatch users in the future.

Wheels, because they seem to interfere heavily with the o.s.. MPatch3 allows you to start them but preserves the right to crash at any time (no kidding!)

This program has more adjustments to offer than the 1.0 version, but it's still limited to 2400 baud on the 128; also I don't have a suitable interface to try a dial-up. With a SwiftLink and Turbo232, it didn't work, and

by Nikolaus Metz

Once again, GEOS users are startled from their peaceful sleep by a scorcher nobody expected until now - more on this later. First, I'd like to present some programs from the past which I found in my vast archives.

GeoTerm 64/128

The software known as GeoTerm isn't suitable for today's everyday use, and it's not likely it's ever been. For instance, take the shareware GeoTerm V1.0 by William C. Coleman from 1988, which is a program for the 40-column mode supporting 300 and 1200 baud as well as half and full duplex. With today's common transfer rates of 14400 baud or 56800 bps (bits per second), it can't be recommended. Besides that, it doesn't run correctly under Wheels and neither under MPatch3. It also seems to require an older interface on the user port which I don't own.

It's similar to the 'professional' version GeoTerm 2.1 made by Carsten Clasohm in 1991, which exists in 3 versions (all in German language) - one for the 64 and two for the 128 (one for each of the 40 and 80-column screens). None of them runs under

an old 'Drews' BTX-cartridge for the user port failed as well. The earlier versions 1.1 and 1.2 published by the 64'er magazine couldn't be tested due to bad disks. All in all, it seems there's no convenient, up-to-date GEOS terminal software that suits present requirements.

geoFAX

Maurice Randall's 'geoFAX' stands alone in the field of fax software. It's a highly recommended program for GEOS users who can't afford or don't want to pay for a fax machine. Any geoPaint file can be converted into a fax document and sent with speeds up to 14400 baud (interface speed - 38400 bps). To send geoWrite text, you first have to convert it into a bitmap image using a special printer driver. Although this might seem a bit complicated, it's exactly what modern PC software does, except that the user doesn't notice it.
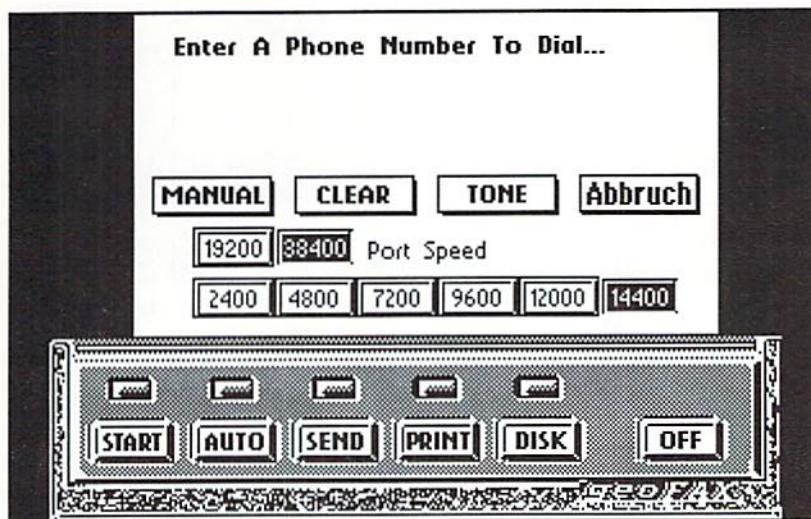
geoFAX runs under GEOS, Wheels, and MPatch3 on the 64 as well as on the 128 in 40-column mode. It requires a Turbo232 interface (V2.0 and up) or a SwiftLink (V1.7 and up) as well as a class 2 fax modem which should have an internal 16K memory (easy to check with the command 'at+fbuf'). In the current version 2.1, the program offers you a large number of options to customize the software to your own needs and also some utilities that allow you to copy, delete, and rename files without leaving geoFAX, for example.

Unfortunately, fax polling - a service which is increasingly offered by companies and publishing houses - is not directly available with geoFAX. Maurice suggested dialing the number on the phone and then switching to the modem, but this idea failed on the German phone system; it's said to work in the US. Maybe this option will be featured in a future update; it's almost the only thing this program lacks for perfection. For convenience sake, I imagine a function to convert a geoWrite document from inside geoFAX; however, the question arises how large the program (217 blocks = 55 K) may

become if there's to be enough space for documents left on the disk. Ah well, you can't have it all...

The Wave

It's been four years now since the first demo of this program came out, and I was quite astonished when I saw it for the first time. But before continuing it, Maurice Randall turned to his 'Project G' (namely Wheels). The Wave was originally intended as an all-purpose program, at least that's the impression you get from the opening screen. From here without actually leaving the program, it should be possible to switch between the important components, which are the Desk Accessories, the Internet/Web browser (The Wild Surf), an ANSI terminal to dial into BBS's (The Sightseer), and an ASCII terminal (The Beachcomber).

The Beachcomber is the only thing that can be started from here and is more or less functional. If you'd like to verify this by yourself, you should have a C128 along with a SwiftLink cartridge and use Wheels to run the Wave Demo from 1996; MPatch3 causes trouble. I can confirm this terminal program
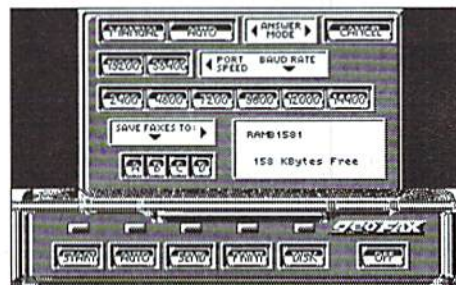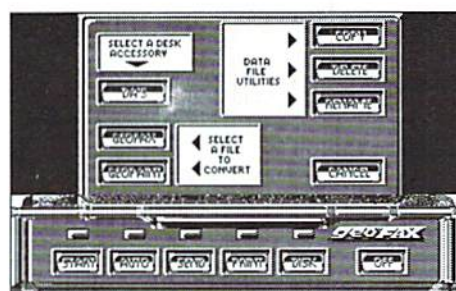
works, because I dialed into the Sachsen GEOS-BBS with it. The interface speed is adjustable in the range between 300 and 38400 bps, limited by the speed of the SwiftLink. Unfortunately, this is the only working function; neither file transfer (in both directions) nor the directory nor the planned online help are available. But still, it's a proof of what you can get out of an aged computer with a graphical interface.

The new Wave

For the time being, work on The Wave continues at full speed. According to an official announcement made by Maurice Randall in the Internet newsgroup 'comp.sys.cbm', a web browser is going to be written which will rank up there with those used on PCs. He's still looking for programmers who are willing to contribute a part of their work to help complete the project more quickly. And it seems he has already gotten some help; the progress reports give reason to hope we'll still see a more or less working version in 1999. The finished program will be non-commercial, so every owner of a C64 or 128 can use it for free. Also, the source code will be freely available to enable further development and ports of the program (e.g., to MegaPatch3). Even an automatic update function is planned, since there might be updates every week - simply go to Maurice Randall's homepage, click on the update link, and the computer does the rest, just as PCs do.
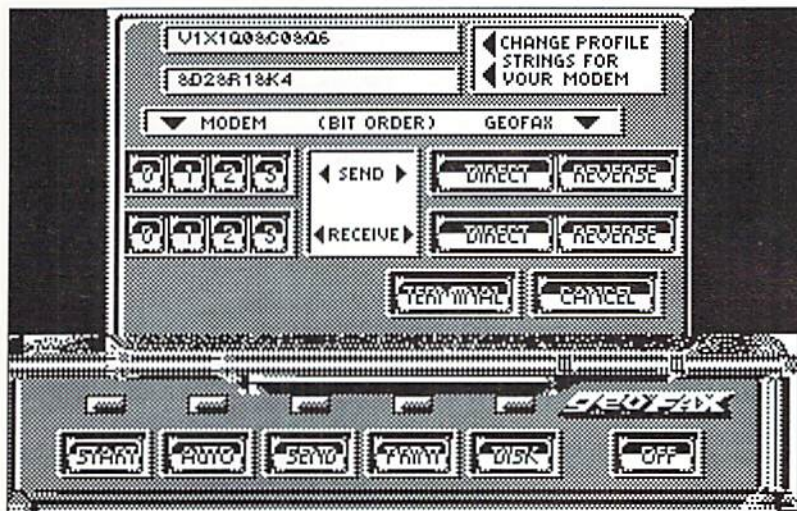
The HTML interpreter is almost finished and processes 37 of the HTML commands, like font size, forced line breaks, enumerated lists, citations with link to the source in the Internet, various text marking functions, as well as columns. The interpreter knows 40 of
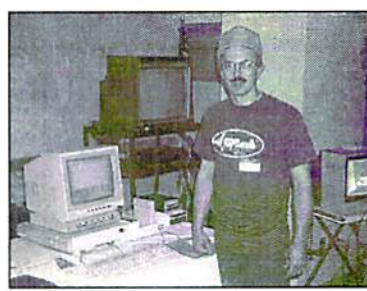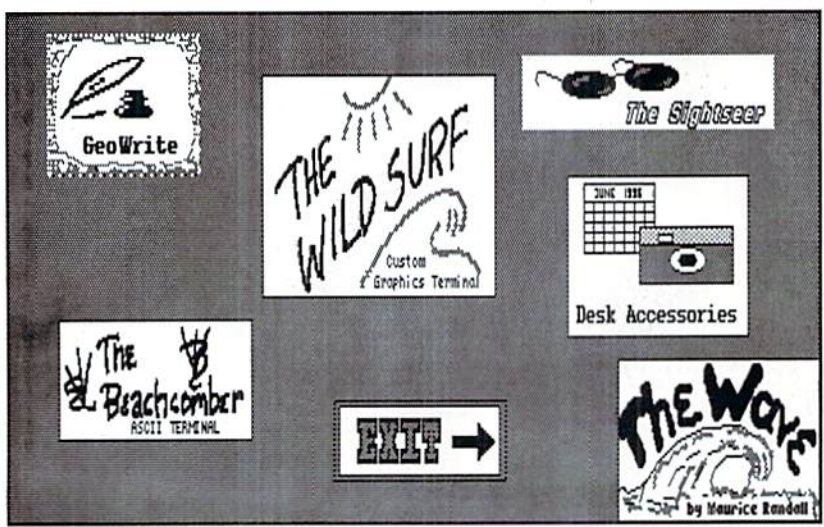
Maurice 'Wheels' Randall
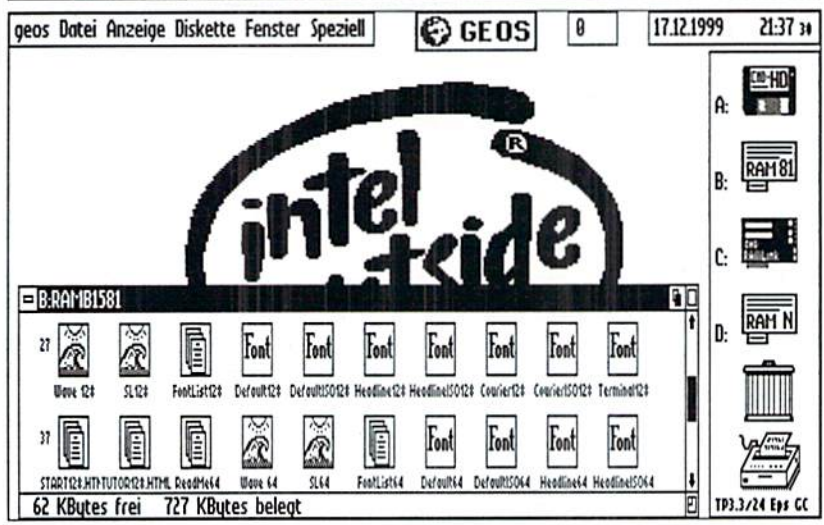
the attributes, of which the six most important ones are realized, e.g., the font size.

Also in the near future, we can expect that rendering images in the GIF or JPEG formats commonly used on the Internet won't be a problem anymore; it's the display capabilities of the computer that are limited - after all, our computer doesn't have 16 million colors - but what we don't have, we don't have. For those who want to offer Internet pages for Commodore users, it's planned to allow the inclusion of GEOS photo scraps, which are relatively easy to display on the C64/128.

For me personally, the most important point thing is how you get 'into the Net'. Since there are a lot of different ISPs, it must be possible for the user to choose freely; there's more than just T-Online, AOL, and Compuserve. The TCP/IP stack and the PPP/SLIP dialer are currently being implemented. Whether it's going to be possible to connect via any provider depends on the complexity of the protocols and on more help from other experienced programmers. Connecting via AOL will not be possible, because they're using a protocol of their own, and the programmers would need the source code to

this protocol. But in my opinion, The Wave would work with T-Online (PPP).

In order to use the browser, besides your C64/128 you'll need a good monitor - the 128 version runs on the 80 column screen only - and Wheels, at least for the moment. Additionally, a SCPU equipped with a RamCard with 4 megs or more of memory is recommended for web page display, 1 meg mandatory. A fast storage device with lots of

space is recommended for temporary cache; a RamLink or a hard drive would be ideal. Of course, you can use 'standard' disk drives, but you won't save much to them. Not to forget, you'll need a modem with an interface (SwiftLink/Turbo232).

Such hardware requirements are justified, since 'The Wave' is designed as a graphical browser, and you wouldn't want to wait eternally until a web page appears on the screen. Even if you can't afford such hardware or don't want to or simply lack some components like an SCPU or the RamCard, there's still hope you'll be surfing the Internet from your C64 in the near future. As I read recently, there are several other people working on projects to write web browsers that work on the minimum configuration of a C64 (e.g. LUnix NG). With these, however, you'll have to go without the graphics and the convenience, which require lots of memory and processing time.

All in all, I look forward to what the year 2000 will bring after a really turbulent 1999. Anyway, I will keep an eye on these projects. Of course, the first working version of 'The Wave' (tested by myself) will be put onto our coverdisk. Right now, I can only guess the possibilities from what I've tested so far. Scrolling through a HTML page without delays definitely makes me hungry for more.
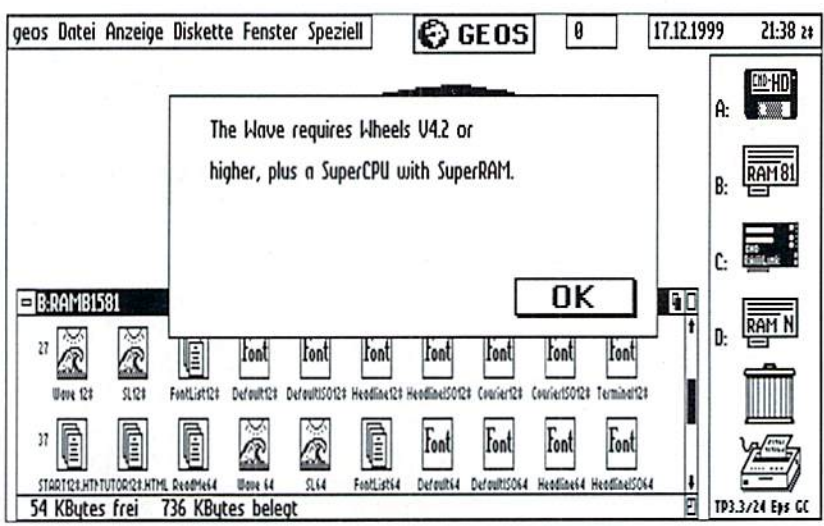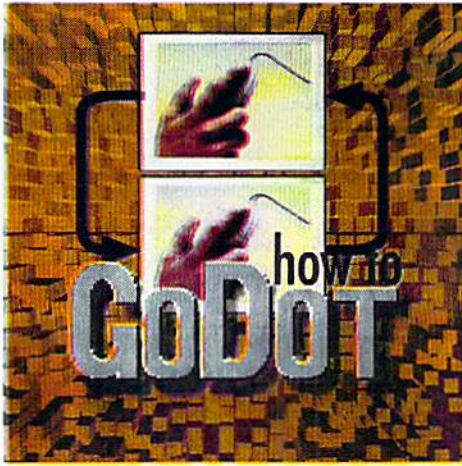
by Arndt Dettke

Converting to FLI, MCI and IFLI

There are three display modes, namely FLI, MCI and IFLI, that make the C64 (well, actually the VIC-II) do things on the screen its designers wouldn't have dreamed of. FLI (Flexible Line Interpretation) breaks the barrier of up to four colors per card (a card is an 8x8 pixel field): all of the 16 colors can be used, four different ones per scanline, per card. MCI (MultiColor Interlaced) breaks the 16 color limit itself, although this mode underlies the normal limitations of the multicolor mode. By means of a physiological trick, the C64 displays up to 136 different colors in MCI mode, but with a very low frame rate. This means the screen may show extreme flickering. Finally, IFLI (Interlaced FLI) combines the advantages of the two other modes: it shows up to 136 colors on the screen, which may be used more flexibly than with MCI. Unfortunately, it has the same low frame rate as MCI. American C64's have an advantage here, as these machines have a slightly higher frame rate than European machines.

All these modes are supported by GoDot, and converting into them is really simple: just load an image and save it again in the desired format, that's all. (Savers for MCI are currently being written.)

But, not so fast. There are some details that we have to take into account. Which loaders and savers do you have to use for the aforementioned display modes? What peculiarities do they have? For the first question, turn to the extra box, where you'll find a listing of all loaders and savers that involve FLI etc. About the second question, I'd like to discuss that a bit more intensively.

FLI's features

Generally, displaying FLI images causes a problem with the image information at the left border. This mode is such a strain to the 64 that it loses the color information for the first 24 pixels of each scan line, and shows some ugly gray specks instead. How can we avoid these?

Help comes in the form of GoDot's modules **mod.Scroll**, **mod.PixelEdit** and **mod.ClipWorks**. If the C64 can't display colors in this area, we remove everything that has a color, leaving only black, which is commonly used as the background color by most programs that support FLI.
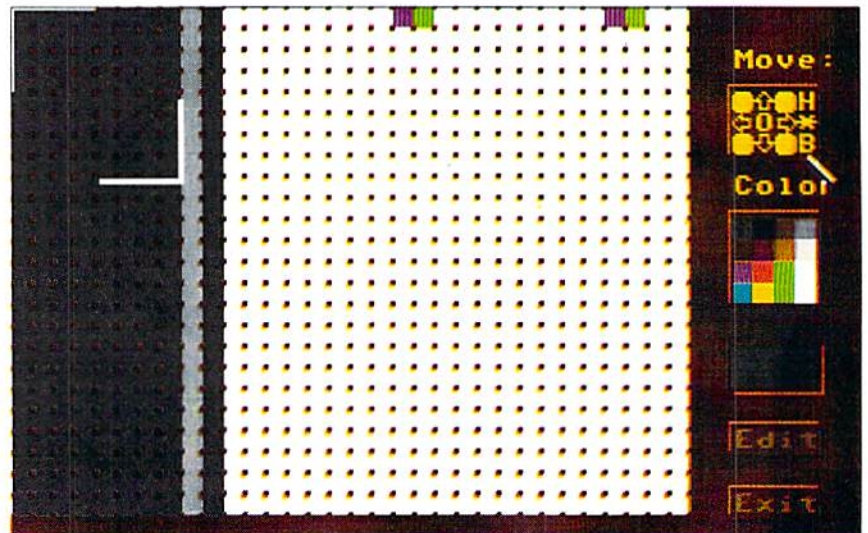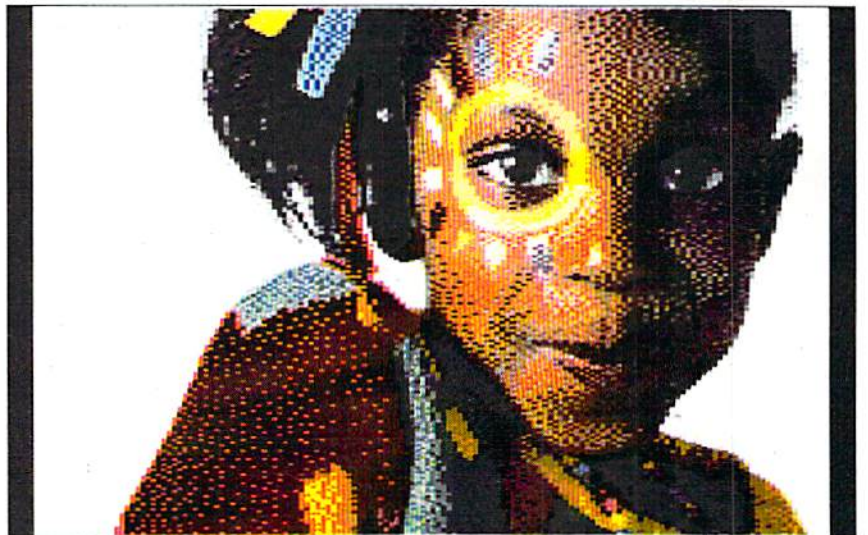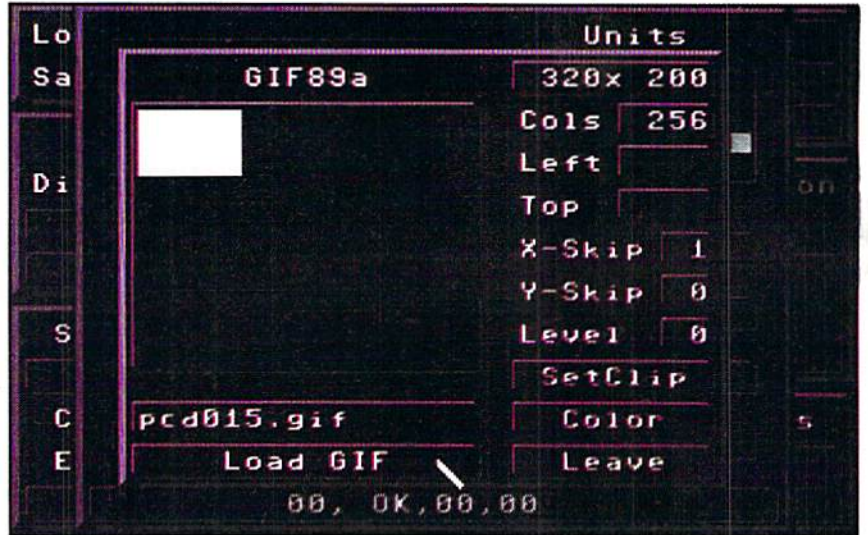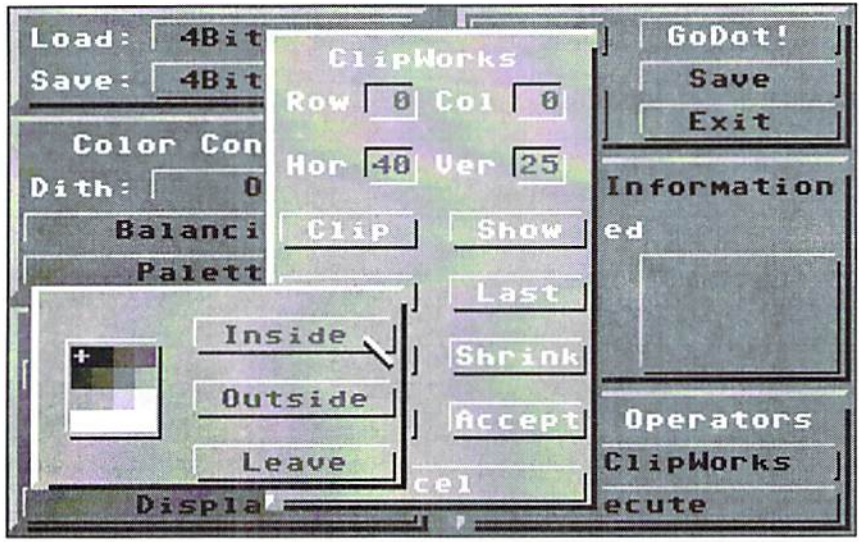
```
Load: GIF
Inst: DecodeGIF
Load "pcd015.gif"
Load GIF
Display
```

A brown girl's face on a white background appears, with a narrow black border to the left and right, the width of which we have to know in order to position the image such that it is centered to a width of 296 pixels (320 minus 24). For this, we use **mod.PixelEdit**:

```
Inst: PixelEdit
Execute
```

In the magnified image we can easily see that the left border is 12 pixels wide. Aha. And to the right? Clicking on the letter "B" (which stands for "Bottom", under the "Move" section) gets us to the lower right corner, and the border width is 12 pixels, too. Fine! 12 plus 12 makes 24, which is exactly the number of pixels we want to eliminate at the left border! Ok, so let's eliminate:

```
Exit
Inst: Scroll
Execute
Set Amount: 12
Direction: -(to the right)
Execute
Leave
Display
```

Unless we're processing a hires image, the value entered under Set Amount in mod.Scroll has to be even, otherwise GoDot would separate the bit pairs that make up the multicolor pixels! We've now moved the image one and a half cards to the right. Next, we use **mod.ClipWorks** to erase at the left:

```
Inst: ClipWorks
Hor: 3 (click on
the number 40 and
enter 3
instead)
ClrClp
(Set:) black
Inside
Leave
Accept
Exec Area: Full
Display
```

After this, the first 3 cards at the left of the image are free of any pixels irritating the FLI mode, and yet the image looks balanced because it's centered. In order to save an image in the MCS format, you have to use a special module instead of the ClrClp function in ClipWorks, since MCS may use a background color different from black (MCS automatically finds the most suitable color). After centering the image, simply execute **mod.MCSprep** and the left border will have the color MCS chose for the background (red in this case).

Next time we'll put our attention to the (unpleasant) properties of the interlace mode, and how we can handle them with GoDot.

Loader	Saver	Description
ldr.AFLI,	svr.AFLI	A hires FLI format, allows two colors per scan line
ldr.DrazPaint	-	An interlaced multi-color format limited to 16 colors, allows you to create multi-colored "hires" images
ldr.FLI-Designer	-	A non-compressing multicolor FLI format, examples can be found on FUNET
ldr.FlipRaw	svr.FlipRaw	A multicolor FLI format with an excellent editor
ldr.GIF	-	A loader that prepares standard GIF images for the IFLI mode, generates up to 55 colors
ldr.HiManRaw	svr.HiManRaw	A hires FLI format similar to AFLI
ldr.MagicDiskEMC	-	A multicolor FLI format from Magic Disk magazine
ldr.M.C.S., ldr.MCS-dat,	svr.MCSfli	Multicolor FLI format used by the program Mandelbrot Construction Set (from 64'er Magazine)
ldr.PCXprp4IFLI, ldr.IFLIprep4PCX,	svr.FunPaintII, svr.PCX-VGA	A loader for PCX which prepares images for the IFLI mode, a loader and a saver for the IFLI editors Fun Painter II (and Gunpaint), and a saver for IFLI images to be displayed on a PC.
ldr.Truepaint	-	An MCI format with up to 136 colors.

GO64! Bazar Form

Please insert the following advertisement in the next issue of GO64!

My advertisement is

☐ private/non-commercial (four (4) lines, 40 characters wide.)
US Dollar 3 (Euro 2.50) are ☐ included in cash ☐ included as cheque ☐ chargeable from my credit card (VISA, AMEX, Master Card)

☐ commercial (any reasonably sized ad, US Dollar 7 (Euro 6) plus taxes

The item(s) I am advertising about are related to:

☐ C64   ☐ C128   ☐ Software   ☐ Accesories   ☐ Misc

Address
CSW Verlag
Goethestraße 22
D-71364 Winnenden
Germany

I assure that I own all rights concerning my offers.

From:

Last Name/First Name _____

Street _____

ZIP/Town/State _____

Credit Card no./expire date/holder _____

Date/Signature _____

GO64! Bazar

Where can one buy or sell second-hand hardware, accessories, literature and software? Is there a convenient solution? Yes, there is! Check out our Classified Advertisements section, the GO64! Bazar! Above, you'll find a small registration form. Just clip it out, fill in the blanks, and send it to the publisher. For every advertisement you submit. For personal/private (non-commercial) advertisements, you may have up to four lines of text, each up to 40 characters in length. Please see the coupon for pricing and other details. We will make every effort to publish your ad in within the next issue of GO64!. If your ad doesn't make it by the next issue, it will appear in the issue following that.

For Free - Contact Corner

We're going to offer a free service to people who are looking for other C64 enthusiasts. Just send a contact advertisement in the form of a letter, FAX, or email to our editorial staff, care of: Contact Corner. The editorial staff reserves the right to shorten parts of letters which are not immediately important, in order to save space.

Commodore 64

RAMLink 4MB RTC, with PSU, Battery, owner's guide, diskette, DM 350,-, Floppy 1581 with PSU, cord, diskette, DM 150,-, Floppy 1571, with parallel cord, power cable, DM 130,-, Floppy 1541 I, with parallel cord, power cable, DM 50,-, Floppy 1541 II, with PSU, in original package (new!), DM 80,-, RAM-Expansion, Commodore REU, 1MB DM 150,-, RAM-Expansion, Commodore REU, 2MB DM 230,-; Battery-control unit for RAMLink (from GO64! Nr. 1/99) DM 50,-; Real Time Clock for cassette port (for GEOS) with drivers disk DM 40,-; keyboard C128DB, new, DM 90,-; keyboard SX64, new, DM 60,-; GEOS-Games, Vol. 1-4, new, in original package, all 4 editions (published by Markt & Technik), DM 30,- Tel.: +49 2053-7765, from 7.00 p.m. central european time

64'er S-CPU DM 285,-, RAMLink 16MB-RAM+battery DM 400,-, FD2000 DM 285,-, 2x1581, 1x1571, 2x1541II, REU512Kb List with SASE, prices after negotiating. W. Rielicke +49 30/7039886 (Fax) Bürgermeisterstr. 55a, 12103 Berlin, Germany

I am searching for blueprints or detailed instructions or complete assemble kit of a radio controlled clock DCF-C64/128 of the company ELV from Leer. Uwe-Klaus Schmidt, Hannover Tel. +49 (0) 511/3523910

Alles eine Frage des Blickpunkts

It's the point of view that matters

Part 3

Have you recovered from the beer can revelry last issue? This time we'll explain how to create a 3-D, spatial impression in your images, since it is the successful use of perspective foreshortening that makes a vivid image.

by Alexander "Mr. Quark" Teufel and Volker Rust

Generally, objects in the foreground (the ones that are nearer to the viewer) appear larger than others. However, while one person may intuitively know how to draw perspectives, another may lack such knowledge altogether. For those who have difficulties in this respect but don't want to go without spatial images, the easiest solution is to use a vanishing point.

Vanishing point perspective

Known from school art classes, this technique is based on the following principle - you mark a point in the image as the vanishing point. In the image, it's the meeting point of the four, long lines.

In order to draw objects according to the vanishing point, you simply draw lines from the lower left and lower right corner of the field of view, ending in the vanishing point. All objects on the plane formed by these lines have to be oriented to the vanishing point. Two more lines going from the upper ends of the field of view to the vanishing point allow you to fix the height of the objects. All objects in our example are oriented according to this

rule. This technique is particularly helpful and frequently used for images where geometric forms play an important role, e.g., when drawing the outlines of towns, buildings, or interiors.

Mixing colors

Flat areas and curvatures can be made to look more spatial by coloring them with several tones of the same primary color. However, we need some more creativity to obtain this effect, because Commodore only gave us 16 colors as compared to machines with millions of colors available. On the other hand, this might just be part of the charm in painting with the 64. Still, the graphician's job doesn't become easier with lots of colors.

Now we want to give you an example of how you can create a spatial impression by mixing colors. By only cleverly choosing the colors, we'll turn a red circle into a red ball.

Let's imagine a billiard ball (if you like, place one in front of you as a model). It has a glossy, reflective surface that reflects the light if the ball is lit, resulting in a highlight, say, in the upper left area. If you look at this highlight more closely, you can see it consists of a small, white circle surrounded by an

COLORTABLE

The demon above the town is painted in black, dark gray, blue, light blue, cyan, and white

oval-shaped color gradation getting darker as it extends to the outside.

You can portray this gradation in a similar way. Start with white as the drawing color (or a light grey if white seems too bright) and extend outwards using the nearest darker color, which is light red. If you want to go further outwards and need another color, you'll have to improvise, because there are no more than two different red tones available; orange might be a possible choice. The lower part of the ball lies in shadows and can be painted with darker colors like mid and dark gray, or dark brown. The shadow's usually
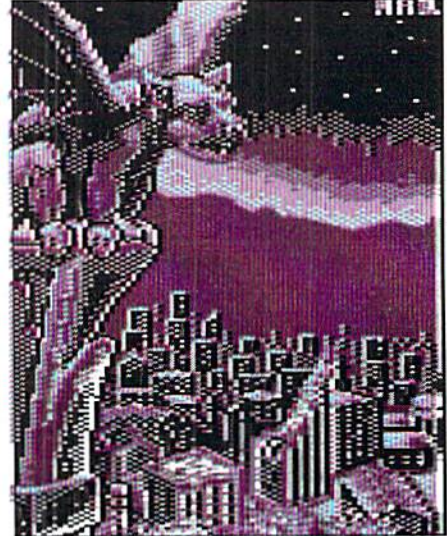
An eerie atmosphere using only six (!) colors

shaped like a half moon, which fits the 3-D shape of the ball.

At this stage, the color edges still look somewhat stepped. Therefore, we'll checker the edges where any two colors meet, coloring every second pixel on the outer side with the color on the inner side, which results in a pattern that looks like a smooth transition.

Color transitions

With rainbow colors, patterns, or simple gradations from bright to dark, color transitions are commonly used to fill areas that would look boring otherwise. It's important not to align all the transitions to the
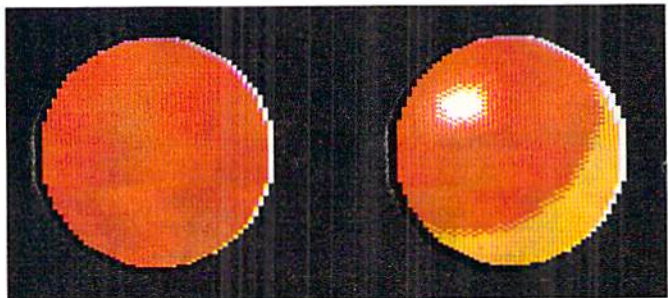
From circle to sphere, the colors do it all
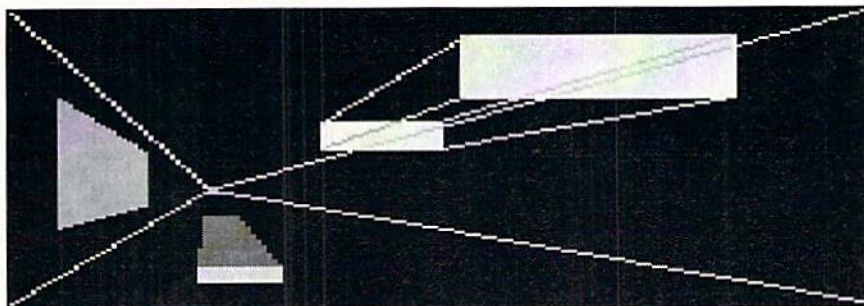
Facilitates foreshortening - the vanishing point

same straight line or to some other fixed point. Variety is the magic word here. It's better to place your pixels a bit more randomly than with a regular, dull pattern that's visible at the first glance. Of course, the smooth color transitions are obtained with the same color mixing technique as described above. In extensively detailed images, they're mostly used for skies, water surfaces, or light reflections. Unfortunately, this is one of the occasions where you have to pay special attention to the limitations of the multicolor mode that prevents you from using too many colors in a 4x8 pixel field (see part 1 of the course in issue 10/99).

You might find it interesting to use the interlaced FLI mode (IFLI, e.g. with the program Fun Painter II), because the increased resolution and variety of colors let you create much more beautiful, smoother transitions and more detailed effects.

> For technical reasons, the colors in the screenshots may change in the magazine printing process, spoiling the graphical tricks. To help you see them accurately, the pictures from the course can also be found on the coverdisk in Amica-Paint format, so you can reconstruct our tips in detail.

An example for the use of color mixing and transitions

Let's put it into practice. Showing a town at night and a demon statue in the foreground, our sample picture uses no more than six colors, but the skillful use of color mixing makes it look much more complex. The color spectrum is mostly located toward blue, which is absolutely sufficient for this motif. The sky in the background is an example of a simple but very nice-looking color transition.

Also, foreshortening was used to get an interesting perspective. The statue in the front gives the viewer an impression of looking down at the sleeping town from above (like from a steeple). Down below, there are buildings at a smaller scale and mountains at the horizon. This picture was inspired by a CD cover; good models can be a valuable help.

If you dare, you can now try projects smaller than a full screen which don't make excessive use of colors.

Tutorial

Assembly Language

by Wanja Gayk

Part 9

Today I would like to start off from where we stopped last time, namely with graphics. After having spent two parts dealing with charset graphics, let's get to bitmap graphics.

The Bitmap

A bitmap is somehow like a gigantic charset with an independent char definition for every block of 8x8 dots (also called cursor block or card). Sounds complicated? Don't worry, it isn't. It's even easier than a charset. A bitmap always has a size of 8000 bytes ($1F40), because every cursor block consists of 8 bytes (just as with a charset). The screen contains 1000 cursor blocks (40x25), and every cursor block encompasses 8x8 dots. This means that we have a maximum resolution of 320x200 dots.

The dots' organization is quite simple: The first eight bytes of a bitmap form the dot pattern for the upper left cursor block, the next eight bytes describe the one to its right, and so on. The bitmap's last eight bytes contain the dots for the cursor block in the screen's lower right corner. The VIC's designers have simply rounded that value of $1F40 bytes up to $2000 bytes, and have divided the memory visible to the VIC into segments of just that size. Therefore, if the VIC bank ranging from $0000 to $3FFF is active, you could have a bitmap either at $0000-$1FFF or at $2000-$3FFF.

Hold it! If you paid attention during the earlier parts, you will know that the VIC can't really use the area from $1000 to $1FFF anyhow: there are ROM images mapped into the upper part of this space, and the range from $0000 to $073F contains the zeropage, the stack, and usually the video/text-display RAM. So let's just forget about $0000- $1FFF. So, in bank 3 (the lowest VIC bank), we only have room from $2000 to $3FFF. But in bank 2 ($4000- $7FFF) and bank 0 ($C000-$FFFF), we have space for two

bitmaps each. Bank 1 ($8000-$BFFF) suffers from some of the same problems as bank 3. Starting at $9000, there is a copy of the ROM charset, so that only the upper half of a bitmap at $8000- $9FFF would show correctly, while you would see the ROM charset in the lower half.

Seeing Colors - without the need for drugs

When Ada Byron was still preparing punched cards for Charles Babbage's "Difference Engine", becoming effectively the first programmer in world history, she was also taking drugs in addition to her alcohol addiction. Maybe that gave her an incredible impression of colors, all without color monitor and video chip. There are nasty rumors that all subsequent programmers have been following her tradition, but the c64 programmers today have a much easier stance, which, before anything else, is advantageous for their health. This is because we have the video RAM. You may now protest because the screen codes reside there, but they are meaningless in bitmap mode, because we have one single gigantic pattern - the bitmap - instead of the charset graphics with their screencodes and their respective small patterns.

Its organization resembles that of the color RAM. The first memory location represents the colors of the card in the upper left corner, and the last memory location for the one in the lower right corner. Every byte in the video RAM contains two colors: The one for the "on" dots, and the background color . The background color resides in the low nibble, while the dot color uses the high nibble. Let's imagine that the bitmap mode was activated, with a bitmap at $2000 and the video RAM, as usual, at $0400. If $0400 contained the value $A9, the cursor block in the upper left corner would have a foreground color of light red ($A), and brown ($9) would be its background color. Background color!?

Yes, you got that right: the background color

register ($D021) can be thrown out in this mode, it will only be of importance again in the multicolor mode, which will be introduced in a later part of this course. The limitations, as far as $1000- $1FFF and $9000- $9FFF are concerned, hold true for the video RAM as well. You simply can't place it there.

"Scotty, Beam me up to the Bitmap Mode!"

No, we are not going to violate the Prime Directive, influencing foreign cultures indifferently, we are simply going to use register $D011, by activating its bit 5. From that moment on, our c64 doesn't display any charsets, but rather bitmaps - resistance is futile. The only things missing now are the positions for bitmap and video RAM in the memory. These are controlled, as we already know from our dealings with charset graphics, by registers $D018 and $DD00. For this purpose, have a look at the following table:

Bitmap/ Video RAM / $DD00 /$D018

Bitmap	Video RAM	$DD00	$D018
$2000-$3F3F	$0400	%xxxxxx11	$18
$2000-$3F3F	$0800	%xxxxxx11	$28
$2000-$3F3F	$0C00	%xxxxxx11	$38

There can't be any video RAM residing at $0000-$03FF or $1000-$1FFF

Bitmap	Video RAM	$DD00	$D018
$4000-$5F3F	$6000	%xxxxxx10	$80
$4000-$5F3F	$6400	%xxxxxx10	$90
$4000-$5F3F	$6800	%xxxxxx10	$A0
$4000-$5F3F	$6C00	%xxxxxx10	$B0
$4000-$5F3F	$7000	%xxxxxx10	$C0
$4000-$5F3F	$7400	%xxxxxx10	$D0
$4000-$5F3F	$7800	%xxxxxx10	$E0
$4000-$5F3F	$7C00	%xxxxxx10	$F0
$6000-$7F3F	$4000	%xxxxxx10	$08
$6000-$7F3F	$4400	%xxxxxx10	$18
$6000-$7F3F	$4800	%xxxxxx10	$28
$6000-$7F3F	$4C00	%xxxxxx10	$38
$6000-$7F3F	$5000	%xxxxxx10	$48
$6000-$7F3F	$5400	%xxxxxx10	$58
$6000-$7F3F	$5800	%xxxxxx10	$68
$6000-$7F3F	$5C00	%xxxxxx10	$78

```
$8000-$9F3F / $A000 / %xxxxxx01 / $80
$8000-$9F3F / $A400 / %xxxxxx01 / $90
$8000-$9F3F / $A800 / %xxxxxx01 / $A0
$8000-$9F3F / $AC00 / %xxxxxx01 / $B0
```
*Up at $8000, only the upper half can be
displayed correctly!*

```
$A000-$BF3F / $8000 / %xxxxxx01 / $08
$A000-$BF3F / $8400 / %xxxxxx01 / $18
$A000-$BF3F / $8800 / %xxxxxx01 / $28
$A000-$BF3F / $8C00 / %xxxxxx01 / $38
```
*No video RAM may reside in $9000-
$9FFF!*

```
$C000-$DF3F / $E000 / %xxxxxx00 / $80
$C000-$DF3F / $E400 / %xxxxxx00 / $90
$C000-$DF3F / $E800 / %xxxxxx00 / $A0
$C000-$DF3F / $EC00 / %xxxxxx00 / $B0
$C000-$DF3F / $F000 / %xxxxxx00 / $C0
$C000-$DF3F / $F400 / %xxxxxx00 / $D0
$C000-$DF3F / $F800 / %xxxxxx00 / $E0
$C000-$DF3F / $FC00 / %xxxxxx00 / $F0
```

```
$E000-$FF3F / $C000 / %xxxxxx00 / $08
$E000-$FF3F / $C400 / %xxxxxx00 / $18
$E000-$FF3F / $C800 / %xxxxxx00 / $28
$E000-$FF3F / $CC00 / %xxxxxx00 / $38
$E000-$FF3F / $D000 / %xxxxxx00 / $48
$E000-$FF3F / $D400 / %xxxxxx00 / $58
$E000-$FF3F / $D800 / %xxxxxx00 / $68
$E000-$FF3F / $DC00 / %xxxxxx00 / $78
```

The table contains all usable combinations and
makes the registers' cooperation quite clear.
The lower two bits in register $DD00 define
the bank, in which the bitmap resides.
$D018's high nibble positions the video
RAM. Register $D018's low nibble holds the
bitmap's position within the bank, which is
calulated just as with charsets: (bitmap start -
bank start)/$0400. Let's take
a bitmap at $6000 as an example: ($6000-
$4000)/$0400=$08. Because there are
effectively but two positions for a bitmap in a
bank (either in the lower or the upper half),
values other than 8 and 0 make no sense for
the low nibble. The values 0-7 would place the
bitmap in the bank's first half, while the values
8-F would place it in the second half. The high
nibble's value is
calculated in the same manner and describes
the video RAM's position.

How Do I Draw Bitmap Graphics?

The bitmaps I have discussed thus far have
been two colored (in effect), with a resolution
of 320x200 dots. They are also known as hires
bitmaps, hires being the abbreviation for "high
resolution". In order to draw hires bitmaps,
there are drawing programs like Art Studio,
Doodle, Hi-Pic-Creator/Oxyron, or the very
simple Hi-Eddi. The GEOPaint format (also
hires), is rather inadequate for the common

programmer, because it has a packed and quite
complicated file format.

Can Bitmaps only Be Used for Pictures?

The answer is quite simply "No". Bitmap
graphics in hires mode have the excellent, but
not yet mentioned characteristic, of every card
having its individual background and
foreground color. This is something the
charset mode does not offer. You can make use
of this characteristic for some effects, as the
following programm shall show. Basically,
this effect is quite simple: You
simply fill the bitmap's uppermost cursor line
(40 cards) with one bit pattern (in this case
%11110000, $F0). Then you change the pixel
and background color for every card, by
writing values from a table into the high and
low nibbles of the video RAM's uppermost
line. When you start rotating that table, you
get a nice little effect. Try changing its looks
by using the value $AA (%10101010) instead
of $F0 as the bitmap's pattern.

One more word on setting the VIC bank
($DD00): Some programmers may be
wondering why I start by clearing the two
lowest bits (AND #$FC, binary %11111100)
and then again setting them (ORA #$03,
binary %00000011). This is only meant to
demonstrate how you should set these values.
If you wanted to use bank 2, the AND
command would be required. For VIC bank 3,
which requires
setting the two lowest bits, simply using ORA
#$03 would suffice. Those starting this
programm from BASIC (after a reset, for
instance) could even skip setting $DD00
completely, because the video RAM resides at
$0400-$073F in the c64's basic mode, which
means that the VIC is already using bank 3
($0000-$3FFF).

The following source code is written in the
ACME cross assembler's syntax (**http://
home.pages.de/~mac_bacon/acme**), and can
be transferred quite easily to other systems.
Those of you using a machine language
monitor have to replace the labels (clear,
pattern, sync, wait, video, cycle, and colors)
by the respective addresses.

```
*=$0810 ;programm start at $0810 (SYS
2064)
!To "Effekt_0810" ;define the created
code's filename

JSR $E544 ;Clearscreen (ROM routine)
LDA #$3B
STA $D011 ;bitmap mode 'on'
LDA #$18
STA $D018 ;video RAM = $0400, bitmap =
$2000
LDA $DD00 ;fetch value from $DD00
AND #$FC ;clear the two lowest bits
```

```
ORA #$03 ;VIC bank number 3 (set bits 0
and 1)
STA $DD00 ;write value back to $DD00

LDA #$20 ;clear bitmap, start address
STA $FB ;$2000 as high/low byte
LDY #$00 ;at zeropage address $FA/$FB
(vector)
STY $FA
TYA ;low byte into accumulator (replaces
LDA #$00)
clear STA ($FA),Y ;copy accumulator to
address in vector ($FA/$FB) plus Y
INY ;increase Y
BNE clear ;branch as long as not 0
INC $FB ;otherwise increase high byte of
address
LDX $FB ;load high byte into X
CPX #$40 ;and compare to $40 (checking
for end address $4000)
BNE clear ;branch as long as not $4000

LDX #$00
LDA #$F0 ;<- bit pattern for the effect
(%11110000)
pattern STA $2000,X ;fill bitmap's first
cursor line ($0140 bytes)
STA $2040,X ;with the bit pattern
INX
BNE pattern

SEI ;IRQ 'off'
sync LDA #$FA ;wait for
wait CMP $D012 ;rasterline $FA
BNE wait ;for even timing

LDX #$28 ;$28 runs (dec. 40)
LDY #$00 ;color table's lower end
video LDA colors-1,X ;value from color
table-1+X
ASL ;shift it
ASL ;into
ASL ;the
ASL ;high nibble
ORA colors,Y ;logical "or" with value
from color
table+Y
STA $03FF,X ;to video RAM minus 1 plus X
INY ;increase Y
DEX ;decrease X
BNE video ;branch as long as X not zero

LDY colors+$27 ;memorize last color
value
LDX #$27 ;$27 runs (dec. 39)
cycle LDA colors-1,X ;move all color
values
STA colors,X ;backwards by one place
DEX ;decrease X
BNE cycle ;branch as long as X not 0
STY colors ;write memorized value to
color table's first position
JMP sync ;jump back to the timing
routine

colors !Byte
$00,$00,$00,$00,$06,$06,$04,$04,$0e,$0e
;color table
consisting
!Byte
$03,$03,$0d,$0d,$01,$01,$01,$01,$01,$01
;of $28
values (dec. 40)
                                    !Byte
$01,$01,$01,$01,$07,$07,$0f,$0f,$0a,$0a
                                    !Byte
$08,$08,$02,$02,$09,$09,$00,$00,$00,$00
```

Commodore Scape

Expo retrospective

By the time this is read, two yearly events will have happened. The Chicago Commodore Expo in Lansing, Illinois, and the Vintage Computer Festival 3.0 in Santa Clara, California. The Chicago Expo is a Commodore-only celebration at the Day's Inn (formerly, Holiday Inn) on September 25. The Vintage Computer Festival is a celebration of all vintage computers, held at the Santa Clara Convention Center on October 2-3.

by Robert Bernardo

These two autumn shows, along with the Lansing Area Commodore Club Expo (Okemos, Michigan) in May, comprise the best-known events in which we Commodorians can attend here in the states.

All three shows have a different flavor, a different style, and each complements the other. The LACC Expo seems very local, like a big, club activity. The Chicago Expo gathers more people from afar, due to its more central location. VCF is very California, Silicon Valley-based, and caters to archivists, historians, and collectors interested in the computer industry. There is something in each which would please a Commodore fan.

Need to feel a closeness and a personal touch in all the C= activities going on at a show? The LACC Expo I attended last year had about 50 people show up. Though the number was small in comparison to other computer shows (or large in comparison to other C= meetings), the attendees could really focus on the demonstrations and luminaries who appeared, such as Dale Sidebottom (president of Louisville Users of Commodore, Kentucky club) and Maurice Randall, Geos programmer extrordinaire. Dale showed what could be done with a Commodore and a Postscript printer, and Maurice officially introduced Wheels 64, the replacement/updated operating system for Geos 64. At this year's LACC Expo, Maurice introduced the beta version of GeoShell 3.0. Size doesn't matter; the LACC Expo makes up for it by having big C=
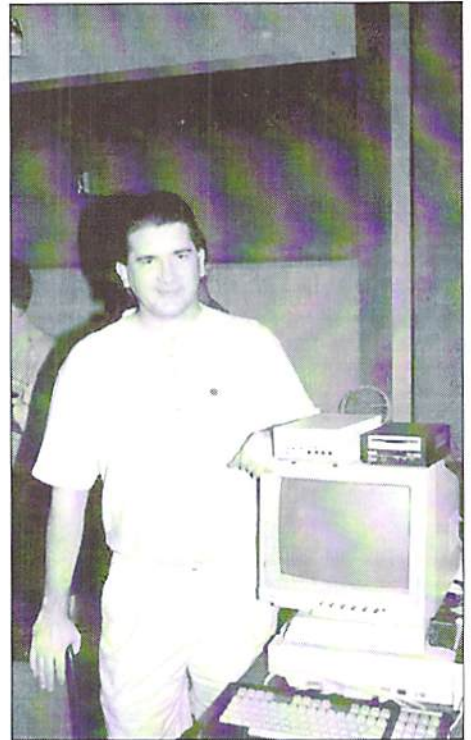
Todd Elliott and his power system: C128D (CR), SuperCPU 128, RAMLink, CMD HD and FD 4000

releases.

Need to a bigger C= Expo? The Chicago Expo I attended last year had about double the number of attendees with even more C= stars, Jason Compton (writer for CU Amiga magazine and Commodore World), Steve Judd, Mark Seelye, Nate Dannenberg, Todd Elliott, and Robin Harbron (programmers), Fender Tucker (Loadstar disk magazine and newsletter), Jim Butterfield (programmer/ writer), Roger Lawhorn (Geos programmer), Scott Parker (Centsible Software) and Dale Sidebottom and Maurice Randall again. At that Chicago show, Maurice introduced Wheels 128, the replacement/updated operating system for Geos 128. Nate Dannenberg showed off his C128 Tower and Quickcam interface/program. Roger Lawhorn and Dale Sidebottom promoted their Geos applications. Jim Butterfield reminisced about Commodore programming in the 1980's. Three Commodore C65's were shown; less than 200 of these prototypes were ever made by CBM. Like the LACC Expo, these are the two shows to go to if you want your fill of Commodore.

Need a show that's more general in nature? The Vintage Computer Festival may be the ticket. At last year's VCF 2.0, many computer platforms from the 1980's and earlier were represented there, anything from

Larry Anderson, sysop of the C= bulletin boards, Silicon Realms and avid C= collector, shows off his award-winning PET computer.
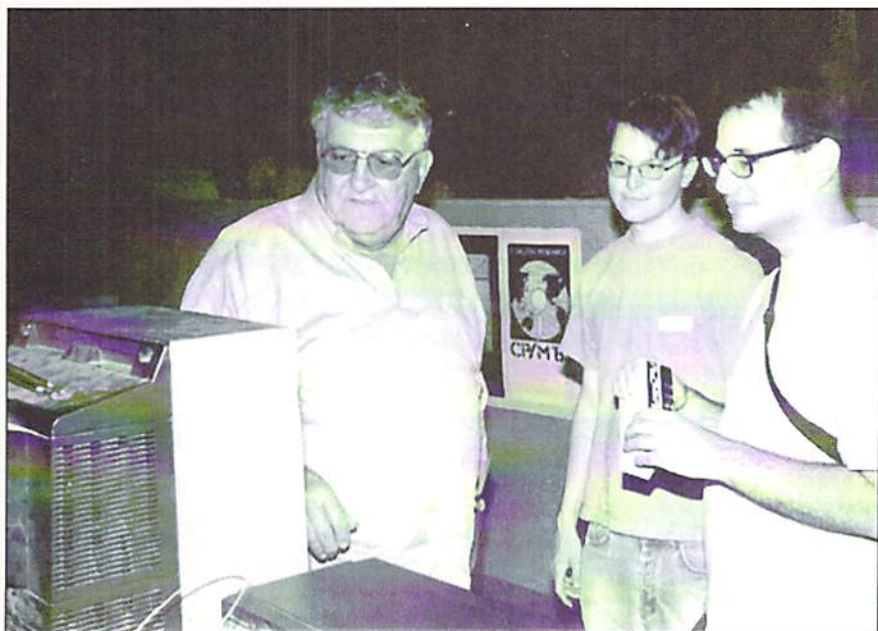
Clark Murphy, vice-president of FUNHUG, (far left) explains the wonders of Wheels to interested on-lookers.
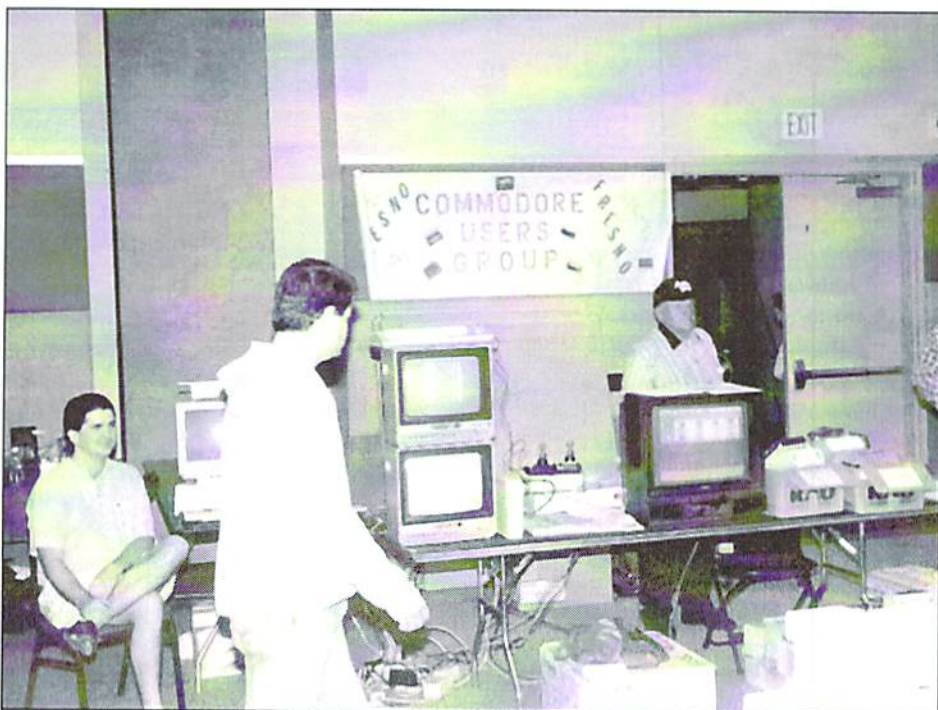
parts for 1950's behemoths to IMSAI 8080's to Commodore PET's to Timex/Sinclair's to Apple II's to Tandy Color Computers to obscure, unknown computers. The Computer Museum History Center had a large exhibit of historical computers, including many different Commodores. These were usually hands-on displays; you actually got to work the computers.

Larry Anderson, one of the exhibit docents and also sysop to the Silicon Realms BBS (209 754-1363), was showing his collection of PET computers. Larry's 15 minutes of fame came when Desmond Crisis, t.v. reporter for the Sci-Fi cable show, CNet Central, interviewed him in front of the cameras. In fact, Desmond's entire report was on how people still used their vintage computers and attended a show such as VCF. (Unfortunately, I did not ever videotape the report from the program. Desmond can be contacted at: CNet, 150 Chestnut St., San Francisco, CA 94111. Phone: 415 395-7805 ext. 1554. FAX: 415 395-9205. E-mail: desmondc@cnet.com)

The VCF flea market had a wide variety of items for a wide variety of computers. Commodore items were there for sale; from various vendors I was able to buy some Warpspeed and FastLoad cartridges, a Commodore Magic Voice cartridge; the programs, World Cup Soccer, Wizard of Oz, Alice in Wonderland, and Totl.Infomaster 3.6; an Animation Station drawing pad, a carrying bag for SX-64 cables (the bag in mint condition!), and a few 1541 and 1571

disk drives (given for free, though I did make a donation).

The featured speakers gave their talks in conference rooms, separate from large, flea market room. At VCF 2.0 a lot of their talk went over my head as they spoke about the computing industry and programming in the 1960's or the 1970's. Their insiders' jokes drew appreciative laughter from the programmers in the crowd, but because it

was before my time in computers, I could only wonder.

As you can see from the above, there is a type of show for most any Commodore fan. To find out information about next year's shows, contact the following people:

LACC Expo - Tim Lewis, president of LACC (lewist@arq.net), LACC, P.O. Box 1065, East Lansing, MI 48826, U.S.A.

Chicago Commodore Expo - Randy Harris, president of the SouthWest Regional Association of Programmers (rgharris@aol.com), SWRAP User Group, Inc., P.O. Box 528232, Chicago, IL 60652-8232, U.S.A.. Phone: 773 375-9017

Vintage Computer Festival - Sam "Sellam" Ismail, organizer of VCF (siconic@jasmine.psyber.com), Vintage Computer Festival, 4275 Rosewood Dr. #29-161, Pleasanton, CA 94588, U.S.A. Phone: 888 SICONIC ext. 823.

Without these people and their partners, we wouldn't have these fine shows in which we could celebrate Commodore. *(ad)*

An overall view of the FCUG table. Todd Elliott, GEOS programmer, is seated to the left. Ed Hart is standing behind the black, 20-inch, RGB/composite monitor

Power Supply tips for the European 128D + SCPU/RamLink/REU Combo

In the pages of "Commodore Scene" and various discussion groups on the Internet, I have read about many people having problems with their SCPUs. Well, a few months back I did have problems with screen artifacts (odd characters) appearing in the 64 mode and also very unreliable operation in all modes as well.

by Colin J. Thomson

I also reccomend changing these two Capacitors.
A = 100µf 350v
B= 0.01µf 1500v

128D Switch Mode PSU layout.
Part N.o BU 223-5-04008

It became more apparent after adding a 2 meg REU to the system, which does put even more load on my 128D's power supply! After no help at all from CMD, I thought I better take a look.

I know there are solutions for the C64's and C128's which have external PSU's (In an early GO64! there is a excellent article about using a PC PSU), and also changing the power switch can make all the difference. But what about us 128D users with internal PSU's? Well, read on...

After some investigating, I found out that my 128D (plastic case version) was low on volts and not by a lot! The initial reading was around 4.93v. Ok, so it was near the 5v CMD recommended, but it wasn't near enough.

A hardware specialist from Germany recommends that it should be between 4.95v to 5.00v (the closer to 5v the better) measured at the user port, pin 2 (+5v) and pin 1 (ground), with everything connected! Measuring it this way, the PSU is under load and gives a more realistic reading. Measuring the PSU with no load, for example disconnected from the computer, is no good at all and may even give a reading above 5v.

PLEASE NOTE
Remember that all your measurements must be made with a digital test meter for accuracy with everything connected - SCPU/RL/REU, etc.. I suggest that you should measure the volts before the modification and after modification so you can compare your results.

After receiving some e-mail's, we came up with a solution to bring up the PSU to the magic 5v. The components to change are some of the electrolytic capacitors around the 5v output side of the PSU. There are five in all, three 2200uf 25v, one 1000uf 16v, and one 470uf 25v (uf = microfarad). I replaced the old capacitors with high temperature radial types. Our model of 128D (European, plastic case) has a "switch mode" power supply, and they

lose volts over the years as do all PSU's.

It's very easy to remove the PSU in the 128D. When you remove the lid of the 128D, it is the metal box in the bottom left corner. Remove the front panel which is fixed by 3 small screws and pull off the drive lever. Then you can remove the PSU which is held in by 4 small screws and unplug the connectors. Remember to take note of the connections when you unplug the connectors, although you can't really go wrong.

After removing the PSU and taking off the lid, you will see on the 5v output side of the power supply and the 5 large, electrolytic capacitors. These tend to become "leaky" over the years and can drag down the voltage slightly.

To replace these capacitors, you will have to remove the PCB (printed circuit board) from the box, but again it's very easy. It's only attached in the case with some screws and is easily removed. Or you could take the PSU to a repair shop and ask them to change the offending parts. The parts are very cheap and are easily available from any electronics shop, but make sure you buy good quality capacitors.

PLEASE NOTE
These capacitors are polarized, i.e., they have a positive and a negative terminal. So before you take out the old ones, make a note of which way they go and mark the PCB accordingly. The best way is to replace them one at a time. I would also recommend that you check for "dry or dull joints" (cold solder joints) and resolder them where necessary while you have the PCB out of its case.

Double-check your work and then reassemble your 128D and check all is well. Now you can connect up everything else, RL/SCPU/REU, etc., and check the volts. On my system the

volts went up from around 4.93v to 4.98v. It does not sound a lot, but it has made all the difference.

I cannot guarantee this modification will work for you, but it has cured all my RL/SCPU/REU problems I have had over the last few months. So why not give it a go?

Note for 128DCR users
The same can be done to the US version (metal case) 128DCR, but because I have not seen one, I can't help. I believe it does not use a "switch mode" PSU, just the normal type PSU, but it's still worth finding the large electrolytic capacitors that "smooth" the 5v line and replacing them. Even standard PSU's lose volts with age! Remember we are talking small amounts.

Thanks must go to Raymond Carlsen for his advice/tips and to Werner Weicht for passing the information to me for the best place to measure the 5 volts.

Additional NOTES...

I would also recommend changing the large electrolytic capacitor marked "A" and the smaller one marked "B" shown in the diagram. The value for "A" = 100uf @ 350v and "B" = 0.01 @ 1.5Kv (1500 VOLTS) !! That's why there is a high voltage warning!!!

There seems to be three types of "switch mode" PSUs for our 128D's, but if you follow the simple rules and replace ALL the large value electrolytic capacitors around the 5v section, you should see a good increase in volts. *(ma)*

Just for Starters
Disk Doctor
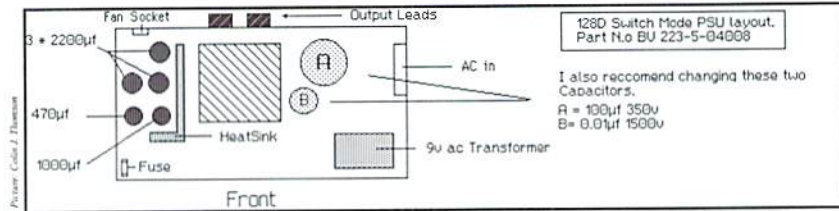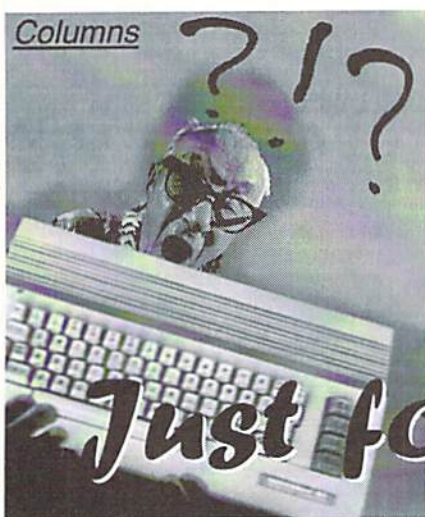
by Jason Compton

This month, we'll take a look at some common issues and questions you might have with standard floppy drives, and touch on some of the less common forms of disk storage.

Even if you're perfectly satisfied with your trusty old 1541, there are still a few things you might want to keep in mind for the months and years ahead.

The Incredible Initialize

If you've ever used a DOS wedge, or read the manual for your floppy drive, or done much of anything on your own with disk and file management, you know that there are a number of commands you can send to the disk drive. If you don't have a DOS wedge, typically you send commands to the drive like this:

```
OPEN 15,8,15,"<command>"
CLOSE 15
```

where 8 is replaced by a higher number if you're using more than one drive on your system. The OPEN command opens up a communication channel to the disk drive's internal brain, sends the command you issue in quotes, then closes that communications channel (a very important step - it lets the disk drive get back to normal operation.) If you have a DOS wedge, you get to skip all that typing and simply type:

```
@<command>
```

which is far more convenient. If you're not using a DOS wedge in BASIC already, they can be found on almost any major utility cartridge, as well as on the "Test/Demo disks" that come with all floppy drives.

Most people usually use DOS commands to do things like rename or delete files, or to format disks. But the most innocuous command is actually one of the most useful: the Initialize command, "I".

The 1541 manual just says that the "Initialize" command "returns the disk drive to the same state as when powered up." That doesn't tell you anything useful. What you should know is that it helps the disk drive adjust to new disks that have been inserted into the drive.

In a perfect world, any Commodore disk put in any Commodore drive should work just fine. But that's not the case. Drives can drift out of alignment, which also causes the data they write to skew on the disk. And sometimes, reading from disks that have been written in differently aligned drives can "confuse" the drive into looking at new disks the wrong way. When this happens, you usually get a "Drive not ready" error, or sometimes you'll get various "Read Errors." Sometimes you can fix this problem by switching the drive on and off, but this causes unnecessary wear.

When you know for sure that a disk is actually good, but doesn't seem to work in your drive especially after you've successfully read different disks there, try a quick Initialize (either with OPEN 15,8,15,"I":CLOSE 15 if you don't have a wedge, or @I if you do.) It doesn't fix all problems, but it can save you a lot of headaches on quirky disks. And remember that many application programs have DOS command channels, so if you're in your favorite program and it's having trouble loading and saving to your project disks, try to send an Initialize command. A quick flash of the drive access light later, and you just might be ready to go!

Disk Drive Compatibility

Have you ever seen "Commodore-compatible" floppy drives advertised and wondered exactly what they meant by "compatible"?

Commodore floppy drives actually have computers inside them: they have their own CPU, memory, and mini operating system. That means they can be programmed - programmed to increase disk access speed, programmed to access disks in an unconventional manner (such as for copy protection purposes), or even to take some of the workload off of the main computer. This makes them versatile and flexible, and it also makes them difficult to replicate perfectly.

Most "Commodore-compatible" floppy drives are not completely compatible, because they are unable to recreate each and every quirk and nuance of the original 1541 drive. Even Commodore's late-model 1541-II disk drive has a few compatibility problems. The most successful compatible drives are those that in essence stole Commodore's ROM code (the operating system of the disk drive) and used it in their own drives. Oceanic was infamous for this.

The less compatible drives work fine in BASIC and with programs that use standard disk access routines, but often choke on more elaborate copy protection or disk turbo schemes. Some, like the Oceanic FSD-2 (or "Excelerator Plus") work with virtually all custom disk access methods.

There are good reasons to think about adding a compatible drive to your system. They're usually smaller and generate less heat than standard 1541s when they have external power supplies. Many have DIP switches for setting the drive number (Commodore didn't start doing that until the 1571 and 1541-II.) You may also be able to find one for cheaper than a Commodore drive because they were cheaper to begin with, or because they are an unusual brand. But remember that for compatibility, the classic 1541 is still the best.

Next month, we'll take a look at what you need to know about other forms of disk storage (including 3.5 inch floppy drives, hard drives, even RAM-based file storage) and more important issues. If you have any questions for Just for Starters, please e-mail

jcompton@xnet.com

(ak)

CMD SuperCPU CORNER

Basically, you can think of things like the SCPU is working at 20 MHz while the computer connected to it sticks to 1 MHz. If both machines are to communicate with each other, the SCPU has no choice but to slow down to the lower speed. In most cases this means waiting for the C64. It would be very wasteful to squander the turbo cartridge's additional power with being idle, which is why CMD has provided aid ment to avoid exactly this. However, the most efficient thing to do would be to have a look at the facts yourself so that your programs can comply with the SCPU as best as possible. The first question that arises then would be when do we have to communicate with the computer at all?

The Helpful Spirits

In the computer there are a lot of specific ICs, those that really make a C64 a C64. The VIC provides the graphics, the SID takes care of the sound, and the CIAs handle input and output. If you want to access one of these, a physical exchange of information to and from the SCPU is required. The actual C64 CPU is not important here, because it is replaced by the SCPU and therefore deactivated. The 65816 is the one and only boss in town then.

All that's left then is the memory, which is kind of a problem. RAM and ROM are by far too slow for 20 MHz, which is why the SCPU has its own RAM and ROM complying with these high standards. However, the original RAM is not totally unused, as all of you who have dealt with the optimization modes know. In the C64 it is not only the processor that accesses the memory but also the video chip which is getting its graphics data from there. Unfortunately, it is impossible for the VIC to access the SCPU's internal RAM, and so the latter has to care for mirroring the data to parts of the C64's memory. Don't forget that the VIC has its own color RAM. Accessing this also forces an exchange. Thus, there are more than enough reasons for the SCPU to exchange information with the computer.

...is quite a waste of the SuperCPU. In spite of this, many programs are running without utilizing the SuperCPU's full speed, because they are not complying with the SCPU's timing. But what does this mean in detail? Information about this and the SCPU in general is offered here!

by Wolfram Sang

Fortunately, there is a way to keep this transfer from delaying things.

The Cache

"Cache" is the name for buffer memory. If the SCPU needs to send a byte to the C64, it writes it to the cache first. The advantage of this is that the SCPU's processor continues to work, because it has gotten rid of that byte. Instead of the processor, the cache now waits for the C64 to get ready, and only then it sends the byte. Hence, there is no delay. The problem is that this cache is but one byte in size! This means if the cache is already full when the processor wants to send another byte to the C64, it has no choice but to wait for the cache to be emptied before it is able to proceed.

This leads to my advice for programmers not to write continuously into mirrored memory addresses, because the SCPU would have to wait for the cache to be emptied all the time. Instead, something useful only dealing with the SCPU's fast RAM should be performed in between repeated accesses. How many cycles have to be bridged doiog this? CMD says you should have write access to mirrored memory every 19th cycle (see note 1). This is really something you need to get used to. For instance, the two programs in our text box would need the same time to be completed in spite of actually having different running times in each loop.

You must also be careful when using the 16-bit registers. If you write a 16-bit value, the processor splits it into two 8-bit (read 1 byte) sized packages which are sent away one after the other. But if the value is to go into mirrored memory, the things discussed above will happen. At the time the second byte

ought to be placed in the cache, the first one isn't gone yet. The unpleasant result is... "wait". The same goes for the assembly language commands MVN and MVP. They allow for moving memory blocks very fast, needing but seven cycles for each byte. In such cases, it could prove to be more efficient to write your own loops instead of using one short opcode and to additionally execute some different code while writing to the mirrored memory that is being dealt with.

I/O Area

All things said so far only apply to the mirrored memory addresses. Within the I/O area - the place where the C64's specific chips have their registers - everything changes.

The cache is not touched here. The processor instantly waits for the next 1 MHz cycle, which leads to a one access every 28 cycles. By the way, this does not only apply to writing but also to reading! This is because, unlike the mirrored RAM, these registers are not copied into the SCPU, and the data must be fetched from the C64 itself. Unfortunately, this also means that Read-Modify-Write Opcodes - commands like INC, ASL, and TRB which read a memory address, modify the value, and immediately write it back - slow down the SCPU twice during their execution. So to speed things up, it may very well be better to separate these commands into simple actions.

```
INC $D020
```

would then, for instance, become

```
LDX $D020
INX
```

```
... (22 cycles for doing different
things)
STX $D020
```

That may seem quite strange, but it's a fact nonetheless. By the way, this behavior in timing does not only occur with the I/O area but also when accessing $01, and if EPROMs are switched on, then in the range from $8000 to $BFFF.

Exceptions

There are two more types of timing. The first one only applies to writing to $DF01, $DF21, and $FF00. These registers are mainly used by REUs, which require some special timing behavior due to their DMA usage. While reading behaves just as within the I/O area, writing to these registers is slightly slower.

The second exception is writing to the CIA registers. After waiting there, the SCPU is halted for about 40 additional cycles. This is meant to keep it from reading from the CIAs within the next two 1 MHz cycles. If it did, erroneous values would be returned, because the CIAs' reaction times are slowed down. This is due to the input and output lines of the CIAs being protected by resistances. By the way, during these occasions, the SCPU is really halted, and you can't run any code during this period!

You should also keep in mind the special status of the color RAM. Usually, you would expect it to comply to the I/O area timing, because it is a part of that area. However, this only applies to the SCPU V1. With the newer V2, the color RAM is handled like mirrored memory. On the one hand, this is quite useful, because things are sped up, but on the

---

Two loops, one speed?

The prerequisite for this comparison is that the range from $0000 to $3FFF is defined as mirrored memory.

```
    LDX #$00                  LDX #$00
L1:                       L1:
    STZ $3F00,X   ; 5          STZ $3F00,X    ; 5
    INX           ; 2          TXA            ; 2
    BNE L1        ; 3          STA $010000,X  ; 5
                               STA $010100,X  ; 5
                               INX            ; 2
                               BNE L1         ; 3
```

The sample routine to the left only needs 10 cycles per loop for execution, while the one to the right needs 22. Nevertheless, both routines require the same amount of time when running on a SCPU, because both are accessing mirrored memory which depends on the 1 MHz clock rate. The only difference is that while the routine to the left spends most of the time waiting, the routine to the right is going to perform something different yet. Making skillful use of this is the real art of efficiently programming the SCPU!

---

other hand, it is quite annoying since it may cause incompatibility.

There are even more differences between V1 and V2. The V2 device can work at its full 20 MHz when dealing with hardware registers (SD07x, $D0Bx, and $01) without having to wait for the next 1 MHz cycle. Finally, it can also access the extra RAM ($D2000-$D3FF) without any delays, while its predecessor always needed one additional cycle to do so.

To be continued...

This should be enough for a first entry into the wonderful world of SCPU timing. C64 programmers should get into this first, and it

really pays off. Those who make the SCPU wait a lot may perhaps only get an increase of speed by less than half of what is possible. However, those who spend a little bit of effort can really achieve maximum speed. The files aren't closed on this topic yet, because the SuperRAM card's behavior has to be discussed as well. Some new aspects will also be discovered there. In the meantime, you can start getting used to the concept of half cycles. Explanation soon to come here, of course!

[1] www.cmdweb.com/cbm/supercpu/suprtime.pdf

The USR-Function what is it good for?

One of the most over-looked functions in the Commodore BASIC repertoire is USR(), the user-defined function. Perhaps it's because by default it doesn't do anything, or perhaps it's because most people have never heard of it.

by Peter Karlsson

It is, however, very useful for adding functions to the BASIC, without having to modify the whole of the BASIC.

An Example

In his 2nd Basic Project, starting in the 8/1999 issue of GO64!, Arndt Dettke mentioned the lack of a modulo operator, one that calculates the remainder of an integer division. In his example, he managed to do without it, but what if it were really necessary? Sure, it could be implemented in BASIC by using a DEF FN user-defined function, or directly in-line using INT() and some other interesting commands, but that is slow.

To alleviate the speed problem, we could always try to write the routine in assembly code, and call it with SYS. The problem is how to pass along the arguments, and how to get the result back from the routine. This is where USR() comes in; it implicitly takes one argument and makes it available to your assembly routine, and it also allows you to return a result back to BASIC.

Argument Passing

Now, there is just one problem, the modulo operator takes two input values, the numerator and the denominator. Well, I said that it implicitly takes one argument, the one within the parentheses, and while this is true, nothing stops us from taking our own arguments outside of the parentheses. We just have to know our way around the BASIC ROM to call the correct routines.

The implicit argument is passed to the USR() routine in FAC1, the first of BASICs three floating point accumulators, located at addresses 97-102 (hex $61-$66), which will also be used to hold any results generated by your code.

The vector for the USR() function is located at addresses 785-786 (hex $311-$312) in the standard low/high format. Upon startup, the

contents of this vector is initialized to 45640 (hex $B248), which will just print the "ILLEGAL QUANTITY" error message on the screen. To test it out, we can point it to an RTS instruction, which would just return the argument passed to it. Try this:

  POKE 49152,96
put a RTS instruction at 49152 (hex $C000)
  POKE 785,0:POKE 786,192
set the USR() vector to $C000
  PRINT USR(42)
should print 42 back to the screen

A very nice property about the USR() function is that it's always available for redefinition, even if other BASIC extensions are available. Thus, using USR() when all you need is one function ensures maximum compatibility.

Easter Sunday

The date of the (Christian) Easter Sunday is determined by the time when the full moon appears in early spring. Thanks to the nice regularity of the moon's orbit around the earth, this date can be calculated by a mathematical formula (and as a nice coincidence, it heavily relies on the modulo operator):

$$A = YEAR\ MOD\ 19$$
$$B = YEAR\ MOD\ 4$$
$$C = YEAR\ MOD\ 7$$
$$D = (19*A + M)\ MOD\ 30$$
$$E = (2*B + 4*C + 6*D + N)\ MOD\ 7$$

The sum of D and E gives the date, counting zero as 22nd March, with two exceptions: If you get the result 26th April, or if you get 25th April with D=28, E=6 and A greater than ten, you should subtract one week. The numbers for M and N depend on the century in question, according to the table below:

Century	M	N
1583-1599	22	2
1600-1699	22	2
1700-1799	23	3
1800-1899	23	4
1900-1999	24	5
2000-2099	24	5
2100-2199	24	6
2200-2299	25	0
2300-2399	26	1
2400-2499	25	1
2500-2599	26	2

Implementation

The modulo operator MOD(X,Y) can be implemented by calculating X-INT(X/Y)*Y, which is exactly the assembly code we are implementing. We are using a number of the BASIC ROM floating point routines to implement this, since we both receive the arguments and have to give the return values in floating point.

To get the second argument (the denominator) we first call the MOV2F routine to move the input argument out to memory, in the BASIC temporary storage area, addresses 87-91 (hex 57-5B, FAC3). We then call the FRMNUM routine, which loads the next numeric argument in the BASIC source code and puts it in FAC1. If the argument is omitted, it will generate a SYNTAX ERROR, and if it is not a number, an expression or a numeric variable, it will generate a TYPE MISMATCH. This suits our purposes well.

This tweaking of the input arguments will result in the syntax USR(X)Y for the MOD(X,Y) function. A problem with this implementation is the operator precedence, to write MOD(X,Y)*2 you need to use an extra set of parentheses: (USR(X)Y)*2, because USR(X)Y*2 will be interpreted as MOD(X,Y*2). While this does not look as good as a "real" implementation of a MOD function would do, you cannot really add a MOD function with a correct syntax in only 33 bytes, so we have to live with it.

The assembly code is available on this month's cover-disk, as well as an implementation of the Easter Sunday algorithm. Please note that you need to RUN the program once, or issue the command POKE 43,1:POKE 44,16 to see the BASIC listing, as it loads a custom character set at the normal start of BASIC (address 2048, hex $0800). The USR() routine is activated by SYSing the start address, in this case, SYS 49152.

Recommended Reading

"Mapping the Commodore 64" from Compute! Publications is a good source of information about what ROM calls are available, as is Marko Mäkelä's Commodore 64 ROM disassembly. Both are available for download from the Project 64 website

http://project64.c64.org *(ak)*

The 1581 ROM-Listing

There are commented ROM listings of the operating system of the 1541 disk drive in various books. For the 1581 disk drive, such a thing has never been published. Since a book about a C64 disk drive would probably only be sold in small numbers today, we bring you exclusively a fully-commented ROM Listing of the 1581 disk drive in GO64!. Because it is very large, we are spreading it over several issues with one page per issue -- this listing certainly not being interesting for everyone. To help collect these pages, we have specifically provided a margin for hole-punching on the left side.

```
Commented 1581 ROM Listing
by Peter Steiner, Heusenstamm


important memory addresses

--------------------------------
Zero page:
--------------------------------
$00- $01 0- 1 Free (probably meant for the insertion of a 6510)
$02- $0a 2- 10 Job memory for buffers 0 - 8
$0b- $1c 11- 28 T&S for every job
$1d- $1e 29- 30 ID
$1f- $24 31- 36 Current block header
$1f 31 Track number
$20 32 Side number (0: Front/ 1: Back)
$21 33 Sector number
$22 34 Sector size
$23- $24 35- 36 CRC checksum
$25 37 Flag for disk changing (also set if disk error occurred)
$26 38 Drive mode 0: drive not active
 b4=1: deactivate engine (still running after last usage)
 b5=1: drive engine is activated
 b6=1: step engine is activated
 b7=1: drive is starting up
$27 39 Current position of head
$28 40 Temporary memory for job code
$29 41 Length of command line
$2a 42 Start up counter for drive engine
$2a- $2b 42- 43 Slow down counter for drive engine
$2c 44 Stack pointer at control loop entry
$2d 45 0: initialize automatically
 1: only initialize by hand
$2e 46 Sector displacement (1)
$2f 47 Free blocks on current track on BAM check
$30 48 b0- 5: number of tries at reading
 b6: no function (at 1541: adjust head)
 b7: 0: at reading error go to track 0
$31- $32 49- 50 Pointer into BAM
$33- $34 51- 52 Pointer to user jump table
$35 53 Flags for BAM status
 $00: BAM has not been changed
 $01: BAM has been changed and needs to be rewritten
$36 54 Temporary memory
 (turn LED on/off)
 (Choose side)
$37 55 Mode of job $9e: BUFMOV_DV (program error)
 data will only be copied if b5=1!!!!
$38 56 Temporary memory for burst load/ 'u9' (warm start)
$39 57 Temporary memory:
 Mode of job $9e: BUFMOV_DV
 1st logic sector on current disk side (JTRANS_TS)
 current disk side
$3a 58 Temporary memory
 for reading/writing/comparing track
$3b 59 Position of '*' in command line
$3c 60 Flag for Burst Load: file has but one block
$3d 61 Counter for found sectors for QUERY DISK FORMAT
$3e 62 1st sector number found for QUERY DISK FORMAT
o3f 63 Temporary buffer: current buffer (for working on BAM)
$40- $41 64- 65 Pointer for various purposes (for instance jmp ($0040),
 ...)
$40- $42 64- 66 Memory for ASCII number for translation to 2 byte
$40- $44 64- 68 Temporary memory for BAM creation
$42- $43 66- 67 Pointer for various purposes (for instance, copying
 buffer contents)
$41- $42 65- 67 Length of a REL file (without super side sectors)
$43 67 Number of data blocks in last side sector * 2
$44 68 Temporary Memory for side sector numbers
 number of side sectors
$45 69 (Free)
$46- $47 70- 71 Jump vector for user commands
$48- $49 72- 73 Buffer address for jobs
$4a- $4b 74- 75 Pointer into cache buffer
 Buffer pointer for jobs: SP_READ/SP_WRITE
$4c 76 Track number of last file (LOAD***)
$4d 77 Current track number
$4e 78 Current sector number
$4f 79 Memory for upper stack border
$50 80 Current channel number
$51 81 0:EOI from bus/ EOF
$52 82 Current secondary address
$53 83 Secondary address from IEC bus
$54 84 Temporary memory for current data byte
$55 85 Temporary memory for side sector number
 number of 'files scratched'
 flag of utility Loader: 'start address found'
$56 86 Temporary memory for:
 track number of current side sector
 position of desired T&S in side sector
 pointer to last used record byte
 checksum for autoboot & utility loader
$55- $56 85- 86 Pointer for creation of error report
$57 87 Sector number of current side sector
```

```
 used record length
 flag:'create but 1 new data block'
$57- $58 87- 88 Pointer for autoboot & utility loader
$58 88 Pointer on T&S of data block in side sector
 number of existing groups
$59 89 Pointer into SS table of current SS
 block size for autoboot & utility loader
$5a 90 When enlarging a REL file: currently worked-on group
$5b- $5d 91- 93 Calculating register 1 (for instance division)
$5e- $5f 94- 95 (Free)
$60- $62 96- 98 Calculating register 2
$63 99 (Free)
$64- $65 100- 101 Temporary buffer pointer (may be set by any routine for
 own purposes)
$66 102 Bit counter for serial bus
$67 103 Counter (for instance for drive/channel search)
$68 104 Position in record
$69 105 Side sector number of record
$6a 106 Position of T&S searched in side sector
$6b 107 Position of record in data block
$6c 108 Temporary memory for job number
 current buffer
$6d 109 Buffer occupation (b0-6: buffer 0-6), 1=occupied)
$6e 110 Drive status
 $00: OK
 $ff: error occurred/foreign format
$6f 111 Format marker of disk
$70 112 Channel occupation table
 b0-6: channel 0- 6 (1= channel free)
 b5-6: reserved for DOS (always 0)
 b7 : unused (always 0)
$71 113 Dir block file was found in
$72 114 1st dir block with free entry
$73 115 Position of free entry in dir block
$74 116 Pointer to current buffer number for channel
 (0-6: 1st buffer; 7-13: 2nd buffer)
$75 117 Number of sectors per track
$76 118 Bus mode
 b0: 1: ATN occurred
 b1: 1: ATN mode; 0: data mode
 b2: last burst clock state
 b3: enabling/ disabling of FSM bus
 1: FSM enabled, 0: always serial slow
 b4: 1: VIC20 mode, 0: C64 mode
 b5: mode request from computer
 1: FSM bus, 0: serial slow
 b6: 1: work on LISTEN
 b7: 1: work on TALK
$77 119 LISTEN drive number (address+ $20)
$78 120 TALK drive number (address+ $40)
$79 121 LED flag
 b5: LED blinking
 b6: drive LED on
$7a 122 Temporary memory for secondary address
$7b 123 Command mode flag; >0 means that a command has been
 sent to the drive via the bus, and that it needs to
 be analyzed.
$7c 124 Burst command byte
$7d 125 Temporary memory for job status (only set, not read)
$7e- $7f 126- 127 Cache buffer pointer for burst commands
$80 128 Burst status
 b0- 3: error code
 b4- 5: sector size
 b6: drive number
 b7: 1: foreign format
$81 129 DOS error suppression (burst access)
 $80: suppress error message
 $00: send error message
$80 130 Temporary memory in controller program during 'real'
 loading/saving (not merely cache access)
$83 131 Number of active job
 $80: no job requested/active (is used as counter and
 index during job handling)
$84 132 Number of active job routine (0-12)
$85- $86 133- 134 Control bits for job execution (see $c163)
$87 135 b7: 1: Cache buffer changed
$88 136 Target track (for head movement)
$89 137 AND mask for burst status
$8a 138 OR mask for burst status
$8b- $8c 139- 140 Pointer to cache buffer
$8d 141 I/O byte
 b5: 1: suppress super- side- sectors
 b6: 0: no creation of checksum
 b7: 0: turn off automatic verify
$8e 142 Number of first physical track
$8f 143 Number of last physical track
$90 144 Starting track of current partition
$91 145 Sector size
 $00: 128 bytes
 $01: 256 bytes
 $02: 512 bytes
 $03: 1024 bytes
$92 146 Number of sectors on track
$93 147 Last sector on track
```

$94 148 1st sector number on track
$95 149 Current track (in cache buffer)
 $80: cache is empty
$96 150 Disk side of current job
$97 151 Current side (in cache buffer)
$98 152 Head settling time after movement
$99 153 Index for 2 byte job memory (e. g. T&S)
$9a 154 Size of hole between two sectors
$9b 155 Filling byte for data block
 $00: during normal formatting ('n: xx,y')
 $e5: during normal burst formatting
 $f5: instead of filling byte, the cache's contents will
 be used. (If there's no formatting so that the verify
 will work; see
$c4e5, $ca73).
$9c 156 Counter for cache delay
$9d 157 Cache delay
$9e 158 Blinking counter for LED
$9f- $a7 159- 167 b0 -6: Position of sector in cache buffer (high byte)
 b7: T&S have already been translated to physical
 format. Usually the execution of a job requires several
 passes by the controller program, but T&S need to be
 translated only during the first pass.
$a8- $ba 168- 186 Secondary address table
 b0- 3: channel number for secondary address
 b76: 00: reading
 01: reading/writing
 10: writing
$bb- $cc 187- 204 Buffer pointer 0- 8 ($bb- $bc: buffer 0)
$cd- $ce 205- 206 Pointer into INPUT buffer
$cf- $d0 207- 208 Pointer into ERROR buffer
$d1- $d7 209- 215 1st buffer for channel
$d8- $de 216- 222 2nd buffer for channel
 b0- 3: buffer number
 b6: 1: flag 'buffer changed'
 b7: 1: buffer not active
$df- $e4 223- 228 Table: channels inactive for the longest time
$e5- $e9 229- 233 Directory block containing file entry
$ea- $ee 234- 238 Pointer into directory block containing entry
$ef- $f3 239- 243 Parameters' drive numbers
$f4- $f8 244- 248 File type and flags for joker
 b0- 2: file type
 $00: DEL file
 $01: SEQ file
 $02: PRG file
 $03: USR file
 $04: REL file
 $05: CBM range
 $07: direct access
 b5: 1: open file
 b6: 1: scratch protection
 b7: 1: joker in file name
$f9- $ff 249- 254 File type table
 b0: drive number
 b1- 3: file type (see $f4- $f8)
 b4: ???
 b5: 1: current record is full
 b6: 1: REL file has been changed (update directory)
 b7: 1: desired record does not exist/end of file reached

Area of hardware stacks

$0100 256 Group number of current record
 $ff: no group chosen
$0101 257 Mode of REL file
 b4: 1: load second part of record
 b5: 0: use super- side- sector (see I/O byte b5)
$0102- $0108 258- 263 Current group number
 $fe: super- side- sector loaded
 $ff: no (super-) side- sector loaded
$0109- $010f 265- 271 Sector of super- side- sector
$0110- $0117 272- 277 Track of super- side- sector
$0117- $018f 264- 399 Stack area

$0190- $01bb 400- 443 Jumping table of the DOS
$0190 400 w b0f0 main empty loop
$0192 402 w dafd interrupt routine
$0194 404 w afca warm start vector
$0196 406 w b262 VALIDATE
$0198 408 w 8ec5 INITIALIZE
$019a 410 w b781 partition
$019c 412 w 892f memory read/write
$019e 414 w 8a5d block commands
$01a0 416 w 898f USER
$01a2 418 w a1a1 POSITION (RECORD)
$01a4 420 w a956 utility loader (&)
$01a6 422 w 876e COPY
$01a8 424 w 88c5 RENAME
$01aa 426 w 8688 SCRATCH
$01ac 428 w b148 NEW
$01ae 430 w a7ae controller's error routine
$01b0 432 w abcf handle ATN
$01b2 434 w ad5c data to serial bus
$01b4 436 w aeb8 data from serial bus
$01b6 438 w cfbe controller routine
$01b8 440 w cedc logical to physical format
$01ba 442 w a7f1 error routine of the DOS

$01bc- $01cd 444- 461 Job memory HDRS2
 physical track/sector for every job
$01ce- $01d6 462- 470 Job memory SIDS
 physical side for every job
$01d7- $01d8 471- 472 Timer B value in high/low format
$01d9 473 Starting up time of engine
$01da 474 Controller command: 'Restore'
$01db 475 Controller command: 'Seek'
$01dc 476 Controller command: 'Step'
$01dd 477 Controller command: 'Step- in'
$01de 478 Controller command: 'Step- out'

$01df 479 Controller command: 'Read sector'
$01e0 480 Controller command: 'Write sector'
$01e1 481 Controller command: 'Read address'
$01e2 482 Controller command: 'Read track'
$01e3 483 Controller command: 'Write track'
$01e4 484 Controller command: 'Force interrupt'
$01e5 485 1st sector of directory
$01e6- $01e7 487- 487 Temporary memory for jumping vector of controller
 error routine
$01e8- $01e9 288- 489 Temporary memory for jumping vector of DOS error
 routine
$01ea 490 Temporary memory: (burst) status/error code
$01eb 491 1581 format signature 'D'
$01ec 492 DOS version number '3'
$01ed- $01ee 493- 494 Length of partition
$01ee 494 Temporary memory for file type
$01ef 495 Smallest sector number found (burst)
$01f0 496 Largest sector number found (burst)
$01f1- $01f9 497- 505 High bytes of buffer addresses
$01fa 506 Write protection status
 $00: writing allowed
 $08: disk write-protected
 other values result in faulty error codes
$01fb 507 Autoboot flag
 b6: 1: autoboot on if INITIALIZE (disk info)
 b7: 1: autoboot on if warm start (user option)
$01fc- $01fe 508- 510 JMP $ba40: send byte to FSM bus
$01ff 511 (free)

Page 2

$0200- $0229 512- 553 Command buffer
$022a 554 Current command number ($80- $89: block commands)
$022b 555 Track of directory
$022c 556 Last track in partition +1
$022d 557 Current file type
$022e 558 Number of parameters (in front of '=') (temporary memory)
$022f 559 Number of parameters in command line
 number of file names in command line
$0230 560 Number of parameters in front of '*'
 counter for file search
$0231 561 Counter for search in directory
$0232 562 Pointer at directory entry found
$0233 563 File type from command line
$0234- $023a 564- 570 Channel status table
 b0: 1: get data from computer (write)
 b3: 0: EOI (End- Or- Identify)
 b7: 1: send data to computer (read)
$023a 570 Channel status for error channel
$023b- $0241 571- 577 Output bytes from channels to serial bus
$0241 577 Output byte from error channel
$0242- $0248 578- 584 Number of valid bytes in data block
 0: not last data block yet
$0249- $024f 585- 591 Current record number low
 current file length low
$0250- $0256 592- 598 Current record number high
 current file length high
$0257- $025d 599- 605 Position of next record in block
$025e- $0264 606- 612 Current record length
$0265- $026b 613- 618 Side- sector buffer for channel
$026c 620 Current parameter length (see $8526)
$026d 621 Number of file name found
$026e 622 >0: Formatted directory will be shown on SA 0
$026f 623 Current record length
$0270 624 Track of current super- side- sector
$0271 625 Sector of current super- side- sector
$0272- $027a 626- 634 Temporary memory for job codes
$027b- $027c 635- 636 (free)
$027d- $0283 637- 643 Directory block with channel's file entry
$0284- $028a 644- 650 Position of file entry in directory block
$028b 651 1st sector number of last file ('*')
$028c 652 Temporary memory (channel number)
$028d 653 Directory: BASIC line number low
 (drive number, file length, 'blocks free')
$028e 654 Directory: BASIC line number high
$028f 655 Temporary memory for char search
$0290 656 Pointer behind current parameter (see $8526)
$0291 657 Position of drive number in command line
 position of 1st file name
$0291- $0296 657- 662 Position of parameters in command line
$0297- $029b 663- 667 Values from command line high bytes
 track number(s) of file(s)
$029c- $02a0 668- 672 Values from command line low bytes
 sector number(s) of file(s)
$02a1 673 Joker flag: current file name contains '*' or '?'
$02a2 674 Bit mask for command line check
$02a3 675 Number of drives
$02a4 676 Kind of a drive number (during loading directory)
$02a5 677 $00: There are file names not found yet
 $ff: All given file names have been found
$02a6 678 Flag for end of directory (0: end)
$02a7 679 Access type from command line (R, W, A, M)
$02a8 680 Deactivate error checking
 b6: 1: don't check for valid T&S
 b7: 1: turn off job error handling
$02a9- $02aa 681- 682 Number of blocks free
$02ab 683 Error number
$02ac- $02c7 684- 711 Directory line
$02c8- $02cf 712- 719 (free)
$02d0- $02ff 720- 767 Error buffer for messages in normal text

----
DOS buffer

$0300- $03ff 768- 1023 Buffer 0
$0400- $04ff 1024- 1279 Buffer 1
$0500- $05ff 1280- 1535 Buffer 2 (user buffer)
$0500 1280 <- U3
$0503 1283 <- U4
$0506 1286 <- U5

Readers' Corner

E-mailing with the C64

Here's a letter concerning Niko Malecki's answer in the 7/99 GO64! issue to Martin Poleska's question about e-mailing with the C64:

There's another possibility to get/send e-mails with C64s and C128s, although it is less widely known. You can work with a

mailbox. There are very few mailboxes, and there remains only one that I know of as far as C64s are concerned; it's my own (I don't want to hurt any sysops' pride, so tell me if you know any other mailboxes that offer this service!!!) It enables you to write e-mails on C64s and C128s using average off-line readers. The only special feature you have to pay attention to is the way you have to address your e-mails.

You can easily read/write e-mails off-line. It is also possible to download FTP and HTML files. Everything is sent to a pseudo address that gets things on the Net; only e-mails go directly on the Internet. I have to admit that the mail delivery times are not all too fast, because I have to make a phone call every time to send them on the Internet. Normally these calls are between 6 pm to 3am CET, which leaves the possibility to answer the messages the same day you get them. This is the only service you have to pay for on my mailbox (there are also FIDO-, GEOS- and CBMNet), because I have to pay for the phone connection. Complete access costs 3 DM per month which is still cheaper by far than a connection via t-online (I'm not a shark; I haven't been making any profits on the mailbox for five years now!). You can define the part of the e-mail address left of the @ in any way that you want. Only the domain to the right of the @ is fixed: SachsenGEOS.FIDO.DE. Anyone interested can contact me; the mailbox number is +49 (0) 341-2613589 (open 24 hrs.). The telephone number is +49 (0) 341-2612789.

René Schwarte

Seikosha SL 80-VC

Not too long ago, I bought a used Seikosha SL80-VC pin printer which has two DIP switches. Can anyone send me the correct position of the switches? Who can help me? I would be grateful for any letters, or I could call you at night from 8-12 pm.

Wilfried Sopart
Baerstr. 8
D-37290 Meissner Abterade
Germany
Tel +49 5657 378

Who can help?

A floppy that starts to work by itself and other problems

Here's an update to Hansjörg Meduna's

problem in the 4/99 GO64! issue. He had a floppy drive that started working without any given reason.

I was really satisfied to find my question and my suggestion in the 4/99 issue. In the meantime, I discovered the reason why the floppy 1541 drive started working without any reason and only functioned properly again after turning it off (and losing the program). The fault was in the mains power supply. The operational voltage of 6 volts jumped to 8.3 volts all of a sudden. The voltage control IC had to be out-of-order. The required voltage of 6 volts was there again after restarting the floppy drive. I had to buy a new mains power supply.

I've got two more questions. The writing on some of the Commodore 64's keys has begun to disappear. I tried to restore the signs on the keys with a special pen for writing on plastic (Lumocolor 318 waterproof). But the ink doesn't stay on for long. Who could tell me a better way of restoring the writing?

Are there any further GEOS graphics (clip arts) apart from the Mega Pack 1 & 2 and Tommy's Superscanner?

Hansjörg Meduna

Concerning the GEOS graphics, there are tons of other graphics collections, most of them available everywhere as part of public domain collections in all kinds of GEOS libraries. We tested two of the available graphics collections in the GO64! 4/98 issue - the Mystery Art Collection (available from Günther Rauscher, Grieswang 3, A-3262 Wang, Austria for 20 marks) and the Big Clip Art Collection (39.95 marks from the GEOS Users Club, Moerser Str. 11, D-46286 Dorsten, Germany).

GO64! editors

Turbocard

Not long ago somebody gave me a small C64 system (with a Turbocard by Roßmöller). There weren't any manuals. The system works fine; only the program execution has a severe fault that's definitely getting on my nerves. There is a continuous flickering of the screen, as though the synchronization was out-of-time. I've already changed the mains power switch, but the fault has remained. Some programs that include still images do not show the flickering. Of course, it is hard to diagnose the difficulty from afar, but can anybody give some helpful advice to solve the problem? Anyone who wants to contact me personally, please ask the editorial staff for my address.

G. Keuenhof

Who can help here?

The CMD SuperCPU - it's got a will of its own!

Let's think back to Herbert Bock's problems with the power supply in the 7/99 issue and to the article about power supplies in the 10/98 issue.

1. The power supply jack you were looking for is still available, i.e. at Kessler Elektronik in Meinerzhagen or at Reichelt Elektronik in Sande. It is the usual seven-pin DIN jack and costs about 1 Mark. A built-in jack really looks better than a cord that's hanging from the case, doesn't it?

2. Here's a general trick for building your own power supplies. Normally, they tend to be much better than the originals because of their higher amperage, so you should not try to imitate the kind of economy class wiring with only four wires that the originals use. Nearly all C64s (except for the really antique ones where one of the 5 volt wires is not connected) have a power supply jack with three ground wires and two +5V wires. To be on the safe side, you should use the five wires. The full wiring looks like this:

```
9V   *   *   9V

GND  *   *   GND

+5V  *   *   +5V
          *
GND
```

Nicolas Welte

A Virus in GEOS?

I've been using GEOS for about a year. Recently, I opened the GO64! cover disk from issue 7/99 from the system disk and printed the order form and the questionary. From that day, an error keeps on occuring when I work with GEOS: load error on system disk (system error near $bf41). It happens regularly; by now, I can't even boot the system disk. I can only boot using the backup system disk, but even with it, the above error near $bf41 appears from time to time.

Something else has happened: My trusty MPS1230 printer starts to malfunction more and more frequently: it starts printing with a paragraph sign in the first line, and after that, it prints double-shifted. Sometimes it prints control characters in the middle of a line, just to go on printing as usual some time later. Could this be a virus?

Harry Ehrenhaus

To the first topic: I instantly noted that you're booting from the original boot disk. Even though this is not actually wrong, it's still not recommendable. It also seems to me that you used to write onto this disk, which you should avoid doing by all means. Writing to the GEOS system disks is delicate because they have a copy protection applied to them, which can be damaged by writing accesses. If this happens, the system thinks at boot time that it's an (illegal) copy, and refuses to start. It's a well-known problem in the GEOS scene, and I had to make the same experience. The best you can do is to make backup copies of your original disks that are still working. There is software for this purpose, CMD's 'GeoMakeBoot' for example, available from software retailers at a price around $12. After making the backup copies the original disks should be stored safely in a cupboard (or a safe?)

I'm also familiar with the second problem, however, it's not likely it has something to do with the printer; it rather seems to me that the geoWrite file involved is corrupted: it contains characters that aren't displayable on the screen but appear on the printout because the printer doesn't know what to do with them. You can resolve the problem by deleting and retyping the affected parts of the text. Sometimes, the problem with the paragraph sign occurs when printing a geoWrite document that contains a corrupted image.

Nikolaus Metz, GO64!

CMD WAREHOUSE BARGAINS
1-800-638-3263

CMD's Warehouse Bargains include many hard-to-find used and (where noted) new items. In most cases we have only one or two of each item in stock, so be sure to place your order before your selection has been taken! A minimum order of $20.00 applies to all Warehouse Sale orders. Shipping charges are not included in the prices shown. (See our main advertisement elsewhere in this issue for shipping information).

TELEPHONE ORDERS ONLY ON THESE ITEMS

Mailed in orders for non-reserved out-of-stock items will result in a merchandise credit. No cash refunds. All items listed here are sold "as-is", and all sales are final. Prices and specifications subject to change without notice. Not responsible for typographical errors.

BOOKS

32 BASIC Programs for the C64	$8.00
Anatomy Of The 1541	$8.00
Basic Computer Games	$6.00
C128 Computer Aided Design (new)	$8.00
C64 Programmer's Reference Guide	$25.00
Commodore 64 Favorite Programs Explained	$10.00
Commodore 64 For Kids from 8 to 80	$8.00
Commodore 64 Games Book	$5.00
Commodore 64 Programs for the Home	$8.00
Compute's First Book of C=64 Sound &Graphics	$10.00
Computes Second Book of Commodore 64	$10.00
Computes Second Book of Machine Language	$10.00
"Guide to Care, Feeding and Training of C-64"	$6.00
How To Use The Commodore 64 Computer	$10.00
Inside Commodore DOS	$10.00
Osborne 1 User's Reference Guide	$15.00
Superbase The Book	$10.00
The Elementary Commodore 64	$10.00
The Official Book Of King's Quest-Daventry	$10.00
Using & Programming the C64	$8.00
Using Newsroom	$10.00
Your Commodore 64-A Guide to the C 64	$10.00

BUSINESS

64 Powerpack-NEW	$10.00
Better Working File & Report (new)	$15.00
Business System- Sales Analysis Management	$10.00
Datamanager 128	$15.00
Datamanager 128	$20.00
Datamanager 2 (new)	$15.00
Easy Finance 1	$5.00
EPYX Microsoft Multiplan Spreadsheet	$10.00
Financial Cookbook-NEW	$8.00
General Ledger for the C=64-NEW	$15.00
Paperback Filer 64	$10.00
Paperback Planner 64	$10.00
Payables for the C=64-NEW	$15.00
Planner's Choice -Personal Planning System	$8.00
Power Assembler-(new)	$10.00
Practicalc 64	$5.00
Practicalc-(new)	$10.00
Practifile 64-(new)	$10.00
Receivables for the C=64-NEW	$15.00
RUN Productivity Pak II	$8.00
Silent Butler (Check Writing system)	$15.00
Superbase Starter-NEW	$10.00
Swiftcalc 128	$20.00
Swiftcalc 64	$10.00
The Consultant (Database Management)	$15.00
The Home Accountant	$15.00
The Home Banker	$10.00
The Home Manager	$10.00
The Home Manager	$10.00
The Word Machine/Name Machine (new)	$5.00
Valuecalc	$8.00

CARTRIDGES

Alf in The Color Caves	$10.00
Alpha Build	$10.00
Big Bird's Special Delivery	$10.00
Bubble Burst	$10.00
Dragonsden	$8.00
Facemaker-(new)	$10.00
Heskit	$10.00
Kids on Keys-(new)	$10.00
Kindercomp	$10.00
Lemans	$8.00
Linking Logic	$10.00
Mach 5 Cartridge	$15.00
Mastertype (new)	$10.00
MS Pac-Man	$8.00
Pinball Spectacular	$8.00
Radar Rat Race	$8.00
Sea Speller	$10.00

Speed Bingo/Math	$10.00
The Final Cartridge	$15.00
Up & Add 'em	$10.00

CARTRIDGES (No Docs)

Blueprint	$8.00
Clowns	$8.00
Ducks Ahoy	$8.00
Kindercomp	$8.00
Lemans	$8.00
Linking Logic	$8.00
Magic Desk I	$8.00
Maze Master	$8.00
Memory Manor	$8.00
Popeye	$8.00
Speed/Bingo Math	$8.00
Up & Add 'em	$8.00
Vic 20 -Hesware-Gridrunner	$5.00
Vic 20- Avenger	$5.00
Vic 20-Jupiter Lander	$5.00
Vic 20-Omega Race	$5.00

DESIGN

Awardware	$8.00
Delta Drawing-(new)	$6.00
Jingle Disk-NEW	$6.00
Printer's Devil for Print Shop	$8.00
The Christmas Kit-NEW	$10.00
The Newsroom	$10.00
The Newsroom (new)	$15.00
The Print Shop	$15.00
The Print Shop Companion	$10.00

DISK UTILITY

Commodore 64 Software Bonus Pak	$5.00
Disk Utilities	$5.00

EDUCATIONAL/CHILDREN

Computer Science	$5.00
Early Learning Friends	$10.00
Easy Lesson and Easy Quiz	$5.00
English I	$5.00
ENGLISH II	$5.00
First Men on the Moon -Math-(new)	$6.00
Grade A Math	$5.00
In Search of the Most Amazing Thing-(new)	$6.00
Jungle Book Reading-(new)	$6.00
Koalagrams Spelling 1-use with Koala Pad (new)	$6.00
Logic Builders	$8.00
Math Busters	$8.00
MECC Expeditions	$8.00
MECC Odell Lake	$8.00
Paint-A-Rhyme -for use with Koala Pad-(new)	$6.00
Peter Rabbit Reading-(new)	$6.00
Snooper Troops Case #1 (new)	$6.00
Snooper Troops Case #2	$5.00
Snooper Troops Case #2 (new)	$6.00
Snoopy's Skywriter Scramble	$10.00
Story Machine	$8.00
SUNBURST-Memroy Building Blocks	$10.00
Tales of Mystery	$8.00
The Argos Expedition	$8.00
Up for Grabs-(new)	$6.00
Weather Tamers	$6.00
Where in Europe is Carmen San Diego (new)	$20.00
Where in the USA is Carmen Sandiego	$20.00
Where in the World is Carmen San Diego (new)	$25.00
Wizard of Words	$10.00

GAMES

20 Solid Gold C64/128 Entertainment Hits	$10.00
20 Solid Gold Hits -5 Disks	$10.00
Aliens	$4.00
Amnesia	$6.00
Archon	$10.00

Ardok-The Barbarian	$6.00
Batman	$10.00
Batman-The Caped Crusader	$10.00
Beach-Head	$6.00
Below The Root (NEW)	$10.00
Beyond Zork-NEW	$6.00
Blood Money-New	$6.00
Blue Angels-Formation Flight Simulation	$10.00
Bop N Wrestle-NEW	$6.00
Bureaucracy	$10.00
CavemanUgh-lympics	$4.00
Championship Baseball	$6.00
Championship Golf (new)	$6.00
Championship Wrestling	$6.00
Chomp (new)	$4.00
Classic Concentration	$6.00
Commodore 64C Test Pilot-NEW	$6.00
Conflict in Vietnam-NEW	$6.00
Countdown to Shutdown	$4.00
Crossfire	$4.00
Cyrus Chess	$6.00
Deadline	$4.00
Demon's Winter	$6.00
Dig Dug	$4.00
F15 Strike Eagle	$10.00
Feud	$6.00
Fist - The Legend Continues	$6.00
Flight Simulator II	$15.00
GBA Championship Basketball-2 on 2-NEW	$6.00
GFL Championship Football	$6.00
Gold Record Race-(new)	$6.00
Grand Prix Circuit-(new)	$10.00
Guerilla (new)	$6.00
Hacker	$4.00
Hardball (new)	$6.00
Heartland (new)	$6.00
High Roller-NEW	$6.00
Howard the Duck-Adventure on Volcano Island	$4.00
Invaders of the Lost Tomb-(new)	$6.00
Jack Nicklaus Greatest 18 Holes of Golf-NEW	$6.00
Jaws	$6.00
Jeopardy-Junior Edition	$6.00
John Elway's Quarterback	$6.00
Jordan vs Bird	$10.00
Karate Chop (new)	$6.00
Kings of the Beach (new)	$6.00
Knight ORC	$6.00
Kung Fu II-Sticks of Death (new)	$6.00
Last Ninja 2-NEW	$6.00
Legionnaire	$5.00
Lords of Conquest-(new)	$6.00
Lords of Conquest-NEW	$6.00
Mean Streets	$10.00
Megaplay Volume 1	$4.00
Minit Man	$6.00
Murder by the Dozen	$8.00
Navy Seal	$6.00
Navy Seal (new)	$8.00
Operation Whirlwind	$6.00
Pathwords	$6.00
Power Drift	$4.00
Predator	$6.00
Presidential Campaign	$6.00
Project Space Station	$4.00
Q-Bopper	$6.00
Qix (new)	$6.00
Questron II	$4.00
Raid Over Moscow	$4.00
Raid Over Moscow	$6.00
Rambo III	$4.00
Rick Dangerous	$4.00
Robbers of the Lost Tomb	$4.00
Samxion	$6.00
Sargon II-(new)	$6.00
Sargon III-(new)	$8.00
Serve and Volley (new)	$6.00
Shinobi	$6.00
Silent Service	$4.00
Sky Fox	$6.00
Snow strike-NEW	$6.00
Spiderman	$4.00
Sporting News Baseball	$6.00
Star Trooper-(new)	$6.00
Stationfall	$6.00
Summer GamesII	$4.00
Superman	$4.00
Superstar Ice Hockey	$6.00
Suspended -A Cryogenic Nightmare	$5.00
Sword of Fargoal	$4.00
Tag Team Wrestling	$4.00
TETRIS	$6.00
The Last Ninja	$4.00

The Lords of Midnight	$6.00
The President is Missing (new)	$5.00
The World's Greatest Baseball Game	$4.00
Thud Ridge-NEW	$6.00
Times of Lore	$4.00
Titan-NEW	$6.00
Total Eclipse (new)	$5.00
Ultimate Wizard	$6.00
War in Middle Earth-NEW	$6.00
Wheel of Fortune	$8.00
Wild Streets-NEW	$6.00
Word Flyer	$6.00
Z-Pilot	$4.00
Zenji	$6.00
Zork I	$6.00
Zork II	$10.00
Zork III-NEW	$8.00

GEOS

Deskpack 1-NEW	$10.00
Fontpack 1-NEW	$10.00
GEOBASIC (new)	$10.00

HARDWARE

1351 Mouse w/Manual-no disk	$20.00
1702 Monitor -Color 40 Column-refurbished	$99.00
1802 Monitor-Color Monitor 40 Column-refurb	$139.00
Aprotek 1200 Baud Mini-Modem (new)	$30.00
Assorted Joysticks	$5.00
C128D Keyboards (NEW)	$69.95
C128D Keyboards-REFURBISHED	$59.95
C64 Keyboard (NEW)	$49.95
C64 Keyboard (refurbished)	$39.95
CPU64 Version 1 Base Model	$139.00
Datasette	$5.00
EPYX Fast Load Cartridge w/Manual	$10.00
Homewriter 10 Printer w/manual	$45.00
Koala Pad Only no docs	$20.00
KXP 1080I Printer	$75.00
Laser 190E Printer	$30.00
Magic Mouse	$10.00
Magic Voice Speech Module	$50.00
Multiplexer	$50.00
Okimate 10 Color Printer w/manual,disk,ribbons	$45.00
STAR Gemini 10X w/Manual	$25.00
Super Expander 64	$8.00
The Voice Messenger -Speech 64	$20.00
Used SlimLine cases for Commodore 64	$10.00
VICMODEM for VIC20	$5.00

MISCELLANEOUS

10 Assorted Commodore Magazines 1985-86	$10.00
64 Doctor	$5.00
Home Video Producer	$10.00
Instant Music-NEW	$6.00
LOADSTAR Disks#46,65,66,67,68	$10.00
Synthesound 64	$5.00
Vic 20 Cassette-Intro to Basic Part 2	$10.00

MISCELLANEOUS SOFTWARE

Aerobics	$6.00
Bobsterm Pro (new)	$10.00
Commodore	$5.00
Commodore Technology	$5.00
Music Construction Kit	$5.00
SAT Score Improvement System	$5.00
Writer/File Pak 1541	$8.00

PROGRAMMING

Better Working Power Assembler	$10.00
GEOBASIC (new)	$10.00
Hesware - Graphics Basic	$10.00
Introduction to BASIC/Part I	$8.00
Oxford Basic	$10.00
Oxford Pascal	$10.00
Simon's Basic	$10.00
Start Programming w/Gortek and the Microchips	$5.00

WORD PROCESSING

Easy Script	$5.00
Mastertypes Writer	$8.00
Mirage Concepts-Word Processor -Professional	$10.00
Outrageous Pages-NEW	$20.00
Super-Text Professional	$5.00
Supertext Word Processor	$10.00
The Printed Word	$8.00
The Write Stuff 128-NEW	$20.00
Word Writer 128	$20.00
Word Writer 6-NEW	$15.00
Wordpro 3 Plus/64-NEW	$15.00
Wordpro w/turbo Load & Save-NEW	$15.00
Writers Choice	$8.00