TWIN CITIES 128
The COMMODORE 128 Journal

$3.25 US

ISSUE 29

# RAMLINK Preview

## by Doug Cotton, with Loren Lovhaug

For some time now bits and pieces of information about a new RAM based product from Creative Micro Designs have been popping up in various publications and online services. While there has been many delays in getting this product into thê hands of the consumer, RAMLink should be released by the time you read this. TC128 was the first publication to bring news of RAMLink to the general public in an interview with CMD back in issue #26. In issue #27, TC128 was the first publication to give you a sneak preview of CMD's HD hard drives. In our continuing tradition of being the first to bring you information about new products for the 128, we now have an opportunity to get an inside look at RAMLink. Because this is a preview rather than a review, we enlisted the services of Doug Cotton of Creative Micro Designs to help us give you an accurate and insightful look at this new product. Please note that we will actually give you our independent thoughts concerning the quality and importance of RAMlink in a future edition of TC128.

### RAMLink - What is it?

Possibly the most difficult task that CMD will face in selling this new peripheral is explaining to potential customers what RAMLink is and what it is capable of. Our imaginary Commodore Computer Users Dictionary might define RAMLink as follows:

**RAMLink (ramlngk)** *Noun.* A device used to connect read/write memory to a Commodore computer.

Unfortunately, this definition is a bit too literal to describe the full scope of RAMLink's capabilities. To get a real insight into what this product is all about, we'll need to break it down into the several functions which it provides.

### RAMLink Compatibility

RAMLink, via its own internal operating system, provides a usable DOS interface for Commodore 1700, 1764 and 1750 RAM Expansion Units and for Berkeley Softworks' geoRAM. This idea of supplying a constant and usable DOS for other REU's has always (according to CMD) been the focal point of RAMLink. How often have you wished that you could use your Commodore REU with your favorite word processor, spreadsheet or communications program, only to find out that your program won't support an REU? How often have you loaded Commodore's RAMDOS only to find out that when you loaded the program you wished to use it with wiped out RAMDOS in the process?

How many more times did you refrain from even trying, knowing that it probably would not work anyway? It's sad (but true) that when Commodore created their RAM expanders for the 64 and 128, compatibility with present and past software packages did not seem to be much of a consideration.

Apparently, CMD designed RAMLink around this flaw in Commodore's logic. RAMLink provides a DOS interface which performs the functions that Commodore's RAMDOS would normally provide, with one very large difference - RAMLink's DOS (RL DOS) doesn't base its

continued on page 3

# Rumor Opinion & Mayhem

## by Loren Lovhaug

During the past five years that I have been producing Twin Cities 128 and working as an information provider on GEnie and Quantum Link, I have experienced a strange honor. At least two terms I coined have fallen into common usage in the wacky world of Commodore computerdom. Several long time 128 owners as well as couple of computer journalists have confirmed that for better or worse I am responsible for terms such as "technolust" and "flat 128". However with this edition of Rumor/Opinion/Mayhem I will attempt the introduction of a new term to the informal language of Commodorespeak that I can not take full credit for.

Today, as Frank and were working on the daily correspondence, we were discussing the state of C128 product development in 1991 and beyond when it happend. Frank uncovered a new software catagory. So now without further adieu let me add to the likes of shareware, freeware, vaporware, and teaseware, his discovery, "zombieware".

The term zombieware is derived from the term zombie, which defined by several bad horror movies I have seen on late night cable TV, refers to dead folks who just can't get the hang of eternal retirement. These dead folks seem to wander aimlessly from place to place without much of a clue as to what they ought to be doing (kind of like Commodore management).

Zombieware is the software manifestation of "the walking dead". Essentially, it is software which is no longer being sold or supported by its manufacturer but is still "walking", or in other words is still being used and distributed in some way.

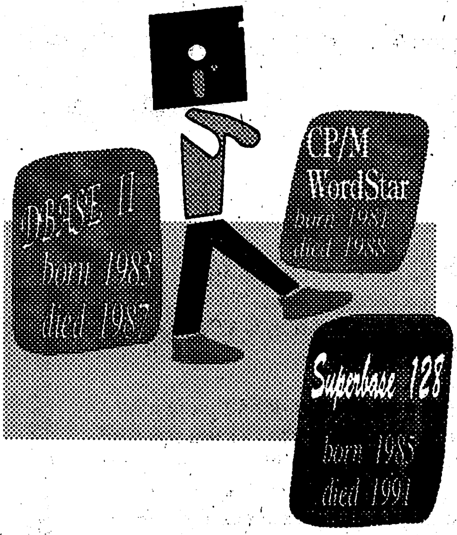
continued on page 2

# Rumor/Opinion/Mayhem continued from page 1

Perhaps the earliest indentifyable forms of Zombieware in the 128 world comes from the CP/M side of the 128. CP/M was on a rapid decline as a mainstream operating environment by the time the 128 (mid 1985) was introduced. Even Digital Research, the creators of CP/M did not seem all that interested in servicing the influx of curious minds that 128 sales brought to the CP/M fold. This meant that 128 owners intrigued by exploring their machine's other triad had a more daunting task than those content with the more familar territory of 64 and 128 modes.

Despite the added challenges involved, however, thousands of 128 owners have found good reasons to explore the CP/M side of the 128. Perhaps paramount amoung these reasons was the extensive library of software that was "sort of" available to run under CP/M. By "sort of" available I am refering to the fact that although several popular applications such as DBASEII and Wordstar were indeed created for CP/M, their creators were not actively marketing them. Due to CP/M's fall from favor in the mainstream computer world several of these companies were concentrating solely on their MSDOS products by the time 128 owners appeared on the scene. The diligent 128 owner, however, could acquire these programs if she/he perservered. And in many cases that perservence was rewarded by popular prices. During 1986 and 1987, a number of liquidation outfits sold off DBASE, Wordstar, and other CP/M packages with complete documentation for a fraction of their original price just a couple of years earlier. For instance, DBASEII was available complete with documentation for $29.95, a slightly better deal than the $495 I had seen it selling for two years earlier. And many folks paid a lot less than that.

The truth is that a vast majority of 128 owners who have copies of DBASEII and other prominent CP/M commericial programs did not buy them at all. They acquired copies of the programs from friends or at user group meetings, even though most user groups have policy against software piracy. And given the fact that even the liquidators have long since exhausted their supplies of most of these programs, this informal distribution network is probably the only method by which many of these programs can be obtained. Of course in the purest interpretation of the copyright laws, these people are guilty of software theft (or piracy). After all I am sure the software



publishers association and the legal staffs at Ashton Tate and MicroPro will be happy to advise you of the fact that these companies have not relinquished their rights to these packages and therefore unauthorized distribution of these products is in fact a crime. However, how likely is a company to spend money protecting a product legally that it is unwilling stock and sell anymore? I know plenty of 128 owners who would have gladly purchased some of their CP/M programs if they could have, since they would dearly love to have the documentation that originally accompanied these packages, but they don't that option.

## Table of Contents

| | | |
|---|---|---|
| RUMOR/OPINION/MAYHEM | 1 | LOVHAUG |
| RAMLINK PREVIEW | 1 | COTTON |
| PRICE & PROGRESS REPORT | 5 | HUDSON |
| THE ANSWER MAN | 7 | VARIOUS |
| ULTIMATE C128 PERIPHERAL | 9 | HUDSON |
| 80 COLUMN SPRITE POINTER | 12 | KANE |
| INTERNAL FUNCTION RAM? | 16 | CURCIO |

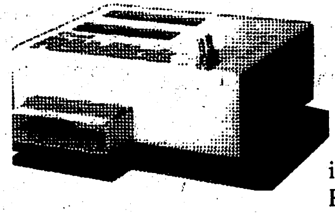
Ironically, the 128 implementation of the CP/M operating system itself is now a member of the walking dead. The final update to CP/M for the 128, dated May 27, 1987, added support for the 1581 disk drive and corrected some minor problems. Originally Commodore had intended this update to be packaged for free with the 1581. However some greedy bean counters and legal eagles at Digital Research decided apply some fine print and grab royalties for distributing an operating system that: 1) they were not supporting and 2) they did not improve (the bug fixes and 1581 support were done in-house at Commodore). Therefore Commodore was forced to charge $20 for the CP/M update. The addition of the $20 fee meant that many 128 owners would forgo the update. Consequently Commodore, which has not cared a whole lot about 128 or CP/M since mid 1987 anyway, stopped actively distributing the upgrade after an initial flurry of orders during the first few months it was available, due to a "lack of customer interest". While it is still claimed by some customer support representatives at Commodore that you can order the CP/M update, several other sources inside West Chester tell me that sending money for a CP/M update is about as likely to get you your update as tossing your checkbook into the ocean. It seems that the only way you can get the CP/M update is to find a friend to make a copy of it for you.

But zombieware in the 128 world is not a concern for the estimated one third of 128 owners who bother to explore CP/M. Much to the contrary, it appears that over the next year many, if not most, popular C64 and C128 titles will "evolve" into the zombieware classification. For instance, there are hundreds of 64 mode games which are no longer available from any legitimate source. And now it appears that many C128 mode productivity titles are going to disappear from distributors shelves.

Of course the de facto bright side of this trend is that the wealth of C128 software will soon be available for little or nothing. However you will have to be "well connected". On the more serious side this also means these packages will not be updated any longer, so they will not improve. New features will not be added, and bugs will not be fixed. And perhaps the worst side-effect of 128 software wandering aimlessly through the night is the fact that many of these packages will do so without documentation. I personally can't imagine using a package like Superbase or Vizastar without documentation. So it is important that you buy these packages now, while they are still available with their complete documentation. Keep your eyes open for liquidation sales, swap meets, and user group meetings as they will likely be your best source for acquiring these packages. In the end, they will be your best defense against the zombies.

# RAMLINK Preview

continued from page 1



operation on a constant need for a page of RAM in the computer. Instead, RAMLink switches in its interface page only when needed, without effecting the contents of that area. RAMLink also doesn't need to resort to using software methods to 'wedge' its way into the operating system of the 64 or 128, but does this by supplying a patched mirror of the proper Kernal ROM. Another benefit of supplying the proper DOS in ROM is that you need never load the DOS - it's always there.

Some other beneficial differences exist between RAMLink's RL DOS and Commodore's RAMDOS. Among these is the ability to us a much wider range of standard Commodore DOS commands. Quite nearly every disk command is emulated by RL DOS, including direct access commands (BLOCK-READ, BLOCK-WRITE, BLOCK-ALLOCATE, BLOCK-FREE, BLOCK-EXECUTE, U1 and U2) and memory commands (M-R, M-W and M-E). Most surprisingly of all, RL DOS can also handle JOB QUEUE instructions and contains DOS buffers, just as it would if it really were a Commodore-type disk drive.

## RAM Expander

While RAMLink can operate with only an REU of the types mentioned previous, it is also capable of operating with internally installed RAM using CMD's optional RAMCard. You can use this optional 'daughter board' to make RAMLink into a stand-alone REU. RAMCard can be purchased with RAMLink (or separately later) in capacities of 1, 2, 3 or 4 Megabytes of RAM. RAMLink may hold only a single RAMCard, but the capacity of that RAMCard may be as much as 16 Megabytes! RAMCard memory is also combinable with any REU memory you may have plugged into RAMLink's external RAM port, but keep in mind that the system can only keep track of a maximum of 16 Megabytes of RAM.

One important aspect of RAMCard is how CMD has provided for the addition of RAM. You may install RAM on RAMCard using industry standard SIMMs (Single Inline Memory Modules). RAMCard contains four SIMM sockets, and each of these may hold one SIMM. You may expand RAMCard using either the common and inexpensive 1 Megabyte SIMMs or the newer (common but still quite costly) 4 Megabyte SIMMs. You may not mix SIMM types, however, so you must use one type or the other. A small trace on the RAMCard circuit board must also be cut to use of the higher capacity (4 Megabyte) SIMMs.

If you buy a partially populated RAMCard, installation of more RAM is very easy. You simply remove four screws which hold RAMLink's cover in place, then remove the cover itself. Place your additional RAM SIMM(s) into the empty socket(s) and then replace the cover and screws. Bear in mind that you must remove all power from RAMLink before doing this, so any contents of RAM will be lost during this process.

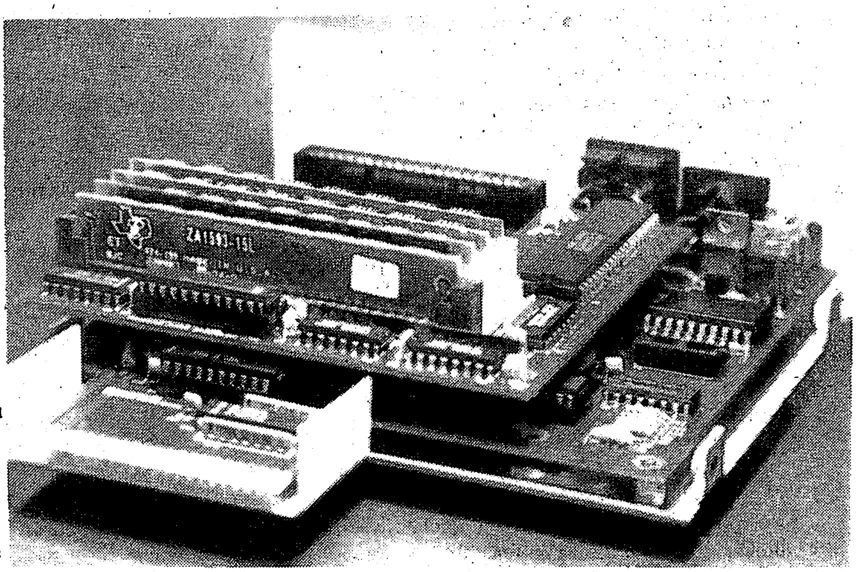
## RAM Data Retention

A separate power supply is supplied with RAMLink which provides power to all the circuitry within RAMLink itself, including to any internal RAM attached via RAMCard or external REU plugged into the RAM Port. By having a separate power supply, RAMLink not only avoids the problems associated with supplying power to an external device with the often under-rated power supplies often included with many Commodore computers, but also provides a means of retaining programs and data stored in RAM when you turn off the computer itself. This can save you the added time needed to reload an REU with your programs and data each time you turn on your computer.

An optional battery is also available to protect against data loss if a power failure should happen. The battery is a sealed lead-acid type, and connects to RAMLink via a cable. Because of the size of the battery itself, it will not fit inside RAMLink. The battery cannot be used to operate RAMLink, and is strictly for retaining the contents of RAM during power failures. Depending on the type and amount of RAM installed, the battery can maintain the data for approximately 20 hours to 2 or 3 days.

## JiffyDOS

RAMLink also contains the computer portion of JiffyDOS for both the Commodore 64 and 128. For those of you who are unfamiliar with JiffyDOS (where have you been?), it provides your machine with the ability to LOAD, SAVE, and do all types of disk access at higher speeds than are possible on a stock system. Attaining the faster disk access speeds with RAMLink requires the purchase of drive ROMs for any disk drive you wish to enhance. JiffyDOS also provides some other goodies, like a built-in DOS wedge for simplifying disk commands, a two drive file copier, screen dump command, output toggle command, text and program file reader, current device toggle and lots more. These added benefits don't need the optional drive ROMs. I could probably go



*RAMLink Exposed: Note the Daughter Card holding the SIMM modules*

continued on page 4

continued from page 3

on and on about the benefits of JiffyDOS, but since that isn't the purpose of this article, I won't. One last comment about the implementation of JiffyDOS in RAMLink is in order, though, and this is that by supplying JiffyDOS in this manner (via RAMLink) CMD has removed the necessity of opening your computer for JiffyDOS installation.

## High Speed Parallel Port

RAMLink is also equipped with a parallel port connector, located on the back panel. This may be used by owners of the CMD HD Series of hard drives and provides the fastest data transfer possible with the HD and a Commodore 64 or 128. Formerly, the HD's speed has been limited by the serial bus, though this restriction is not as severe when using JiffyDOS. Connecting the HD via the parallel bus on RAMLink increases the maximum transfer rate from the current level of around 25K bytes per second (with JiffyDOS) to about 100K bytes per second.

## Cartridge Port Expander

By supplying a PASS-THRU port, RAMLink allows you to use RAM expansion concurrently with many popular utility and I/O cartridges for the Commodore 64 and 128. This helps to alleviate the need for a multi-port expansion device for the cartridge port. CMD has tested the PASS-THRU port with a wide variety of cartridges and all 64 and 128 models and indicates that most cartridges are compatible for use in this port. The exceptions are cartridges which try to permanently replace the computer's Kernal ROM - these will most probably operate in the PASS-THRU port, but will disable RAMLink. Cartridges which have tested successfully with RAMLink include Super Snapshot, Swiftlink-232, Vizastar 64, Vizastar 128, Partner 64 and Partner 128.

## The Controls

Several other hardware features via RAMLink's front (top?) panel controls. The controls include an Enable/Disable toggle switch which can be used to totally disable RAMLink with the exception of the PASS-THRU port. A reset switch which is used to reset both the RAMLink unit itself and your computer is also provided. Because of circuit differences found in the 128D, this reset switch does not have any effect upon the internal 1571 drive.

RAMLink also sports the CMD exclusive SWAP 8 and SWAP 9 switches. These will be familiar to those who have used the CMD HD hard drives, though the RAMLink implementation is slightly different from that found on the HD. The SWAP switches allow RAMLink to exchange device numbers with other serial devices (assigned device numbers 8 or 9) on your system. This makes operating RAMLink with programs which can only recognize these lower device numbers much easier. For example, if you have a 128D system with RAMLink attached as device number 16, pressing the SWAP 8 switch will change RAMLink's device number from 16 to 8. Simultaneously, the internal 1571 drive which is normally assigned device number 8 will become device number 16. While drive swapping happens immediately after pressing the SWAP switch on the CMD HD, with RAMLink the SWAP does not happen until the next serial bus device access occurs. This is because RAMLink does not contain a microprocessor of it's own, and so cannot detect the SWAP switches until it has control of the computer.

One final toggle switch is located on the control panel to select Direct or Normal operation of the RAM port. This allows owners of Commodore 17xx series REU's or Berkeley Softworks' geoRAM to use those devices with software written specifically for them. By placing the Normal/Direct switch in the Direct position, any RAM expander which is plugged into the RAM port will be seen by the computer just as if it were plugged directly into the cartridge port. RAM located in the cartridge is no longer used by RAMLink itself when using this mode, and it's important to note that any data stored there previously by RAMLink will be lost. This adds flexibility to the RAMLink system which can simultaneously provide a high-speed RAM disk if you have RAMCard installed. An REU used with RAMLink in Direct mode will still benefit from RAMLink's data retention capabilities.

## RAMLink's Operating System - RL DOS

While RAMLink is obviously strong and flexible in hardware capabilities, these would be of little value without an equally capable operating system. CMD has shown us that they are very capable of providing complete and compatible operating systems in past products, and RAMLink is no exception to this rule. With RAMLink, CMD has ported their HD Series hard drive DOS to a RAM environment, which makes RAMLink operate very much like a high-speed version of the HD. Full partition support is provided, with emulation partitions for the 1541, 1571, and 1581 disk drives. CMD's own Native partition types are also supported, along with MS-DOS style subdirectories. The only exceptions which were noted were the exclusion of the Burst commands found on the 1571 and 1581 disk drives, and the time and date commands which were standard on the HD because of its built-in real time clock. With these exceptions, RAMLink accepts all standard Commodore and CMD HD DOS and file commands. And with the built-in JiffyDOS DOS wedge, sending commands and manipulating files is easy.

A few features have been added to RL DOS which are noteworthy. Among these are RAMLink's ability to autoboot a file of your choice from any drive on your system whenever the system is turned on or reset. This feature works in both 64 and 128 modes, and in 128 mode will take precedence over a boot block. You may also redefine the JiffyDOS function key assignments and store them permanently in RAMLink. Although the regular versions of JiffyDOS will allow you to redefine the function keys, doing so has always required running a program after turning on your computer.

The DOS emulation in RAMLink is so complete, that it will not only allow you to use direct commands to access particular tracks and sectors of your 'RAM disk', but will also accept and act upon JOB QUEUE commands. RAMLink can also handle all file types, including relative files. No other RAM expansion unit has ever been produced with all of these capabilities. When using RAMLink, you can truly treat it as a disk drive on your system.

# C128 Price And Progress Report

## Goodbye Harry, Hello Jim

This January Commodore announced that its U.S. CEO Harry Copperman was reassigned to the international Commodore holding company Commodore International Ltd. Replacing him as head of US operations is James Dionne, an 11 year veteran and former general manager of Commodore's Canadian sales subsidiary. Mr. Copperman's move was termed a "promotion" in the press release, but some observers believe that it is a cover story for what is essentially a termination of his participation in Commodore operations. At the same time there is evidence that the shuffle is part of a general restructuring of Commodore's North American operations. A layoff of 15% of the employees remaining at West Chester was announced at the same time as the CEO shift, and the latest quarterly report shows that 85% of Commodore International's sales were generated overseas.

As Commodore 128 owners, what does all this mean to you? Not much. With the discontinuance of the 1581 disk drive (see our new Answer Man column in this issue for the details), Commodore has effectively ceased its participation in the C128 market. Unless you still need a RGB monitor or are looking for a 1200 bps modem, Commodore has nothing to sell you for your 128. If Commodore were to leave the North American market entirely, you should barely notice the difference. C128 computer and peripheral choices would remain exactly the same (used equipment and third-party products).

As long-time Commodore watchers, we were always surprised that Commodore refrained from making an organized effort to sell 64 and 128 owners an Amiga or one of its MSDOS line, especially since the non-U.S. arms of Commodore International have had considerable success doing just that. By running the current Power Up promotion, Commodore has at last made such an effort, one we applaud. We can only guess how much more effective this program would have been if it had been combined with a planned and well-organized phase-out of their eight-bit products.

## So Where Can You By That Big, Fast Drive Now?

Maybe you'd rather have a 20 megabyte drive instead of a 1581 anyway. Creative Micro Designs has taken a bit step toward making itself the answer to more users mass storage needs with a price reduction on its HD series hard drives.

| Model# | New Price | Old Price | Reduced |
|---|---|---|---|
| HD-20 | $499.95 | $599.95 | $100.00 |
| HD-40 | $649.95 | $899.95 | $150.00 |
| HD-100 | $999.95 | $1149.95 | $150.00 |
| HD-200 | $1299.95 | $1499.95 | $200.00 |

Shipping and handling charges remain the same for all models. Orders may be placed by calling 1-800-6383-CMD. Information, questions, and support may be handled by calling 1-413-525-0023. Or you could talk with CMD software engineer Doug Cotton directly during the weekly CMD QuantumLink online conference Monday nights 9 pm central time/10 pm eastern.

## Ante Upper

Ante Up Version 2 is now available. Its developer lists improvements such as multi-drive support, 100% memory residence, and three more Geos translations. The upgrade from earlier version is $6.85 postpaid. If you are on Qlink you can purchase this graphics format translation package for a special price of $17 plus an unstated postage charge. Contact address for Ante Up is: Starfollower Productions, 4224 Bakman, N. Hollywood, CA 91602.

## Speaking of QLink

Our corporate arm, Voyager Mindtools, has taken over expanded duties on QuantumLink since the beginning of 1991. We now handle information provider duties throughout the Hardware Support area of the Commodore Information Network. Those of you that have participated in TC128's existing area on this network can now find us in areas supporting other hardware as well. Our new arrangement is going to give us more direct control over C128 file areas, so there should be more to download from TC128 in 1991. Former TC128 editor Loren Lovhaug will be available online every Sunday night 9 pm central time, 10 pm eastern in the Hardware Support area Conference Room during the coming months. To sign up for QLink you can call 1-800-782-2278 or write to them at 8619 Westwood Center Drive, Vienna, VA 22180.

## Who Knows What Evil Lurks in the Heart of East Longmeadow, MA?

Creative Micro Designs has taken over the sales of Dr. Evil Laboratories SID Symphony Stereo Cartridge and the Swiftlink 232 fast serial communications interface. Effective January 1, 1991, CMD began handling all sales and service of new units of both products.

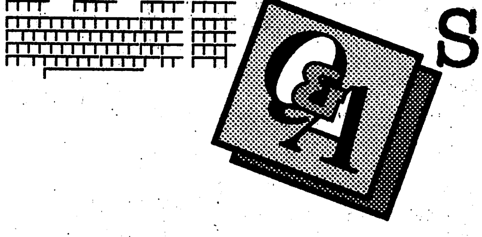
If you're looking for the accompanying software and books that Dr. Evil used to distribute, CMD will be handling them too. CMD has the Enhanced Sidplayer book/disk 3-voice music editing package as well as the powerful 6-voice Stereo Editor and a disk of Sidplayer extensions for MIDI playback.

Any orders received by Dr. Evil Labs which are postmarked after December 31, 1990 will be forwarded to CMD for processing. Customers are urged to use CMD's toll-free order line and credit card capabilities for quick, hassle-free ordering. Their order line is 800-638-3263 (US and Canada). For orders outside the U.S. or for questions call 413-525-0023

Dr. Evil Labs will continue to provide warranty service on any products it has sold, as well as online support on QuantumLink, CompuServe and GEnie. Dr. Evil Labs will also continue to distribute the public domain telecommunications program C-64/128 Kermit v2.2

continued on page 8

# The Answer Man Speaks



Q: I enjoyed your informative article in issue #27 of TC128 in which you wrote about four ways to expand the C128's Random Access Memory (RAM). However lately I have been hearing about other methods for expanding the 128's memory. Would you please tell me about these methods?

A: Well there are at least two new ways to expand the 128's Random Access Memory. Perhaps the one you have been hearing the most about is Creative Micro Designs' RAMLink/RAMDrive products. CMD's products are external RAM expansions which in some respects are simlar to Commodore's RAM Expansion Unit and the Quick Brown Boxes we refered to in issue #27. Like these units their primary use is as a RAMdisk, however CMD has made great strides towards making their RAMDOS extremely compatible with the 1541, 1571, and 1581. This approach makes this products far more likely to function well with existing C128 software. For more on RAMLink see the preview article in this issue of TC128.

For the more adventurous, there is now a scheme for adding additional internal memory to the 128. Unlike the external schemes, this approach actually involves adding additional hardware so that the 128's microprocessor can access this RAM directly, rather than as a RAMdisk. One serious drawback to this approach is that there I do not yet know of any software which is designed to makes use of the extra RAM provided by this scheme. However this could be remedied in the future. Here is a bit more on this technique from its pioneer, Richard Curcio:

Shortly after the publication of Paul Bosacki's C64 expansion project, my speculations on the difficulties involved in doing the same thing for the C128 appeared in the April 1989 issue of Transactor. Most of the suggested ideas in "RAMifications" proved to be un-workable.

I am pleased to announce that the "problem" has been solved, and the C128 can now be provided with the "missing" memory Banks 2 and 3, making it a C256. This doubling of memory would not be worthwhile if it involved an excessive number of integrated circuits, programming "tricks" or "wedges". My solution uses just four ICs (and additional dynamic memory, of course,) and you will not have to remove the existing memory chips. No new I/O locations are created and all memory can be accessed by the existing BASIC and Kernel commands through existing MMU registers.

Complete instructions for carrying out the modifications are available for $25 (US). This price includes two of the required ICs. Or I will modify your machine for $80 plus parts and shipping. See below.

There are a few inadequate schemes circulating to give the 128 Banks 2 and 3, or the 1 Meg "SuperBanking" hinted at in the documentation. The more blatantly BOGUS schemes propose the use of MMU pins that simply DO NOT EXIST! On the whole, the difficulties of any scheme arise from the complexity of what the Memory Management Unit is doing.

Some have suggested a latch to capture bit 7 of the MMU Configuration Register (CFR) at $FF00 and $D500. (Yes, I was one who made such a suggestion). Unfortunately, this is one of those inadequate schemes. Just capturing bit 7 and using it to switch in one of the two pairs of 64K banks would cause Common Memory, essential for normal operation, to become "relative". Part of RAM 0 will be "common" with RAM 1, but not with RAM 2 and 3. A portion of RAM 2 will be common with RAM 3, but not with RAM 0 and 1. There's no simple way around this because the MMU doesn't provide a signal to indicate whether a RAM 0 access is a normal access, or a common access that over-rides the currently selected Bank.

It gets worse. BASIC 7.0 extensively uses the Pre-Configuration Registers (PCR), at $D501-$D504. Store a value in a PCR and it gets transferred to the CFR when ANY value is stored in the corresponding Load Configuration Register (LCR), at $FF01-$FF04. This affects bits 6-7 of $FF00/D500 but WOULD NOT SHOW UP IN AN EXTERNAL LATCH. I didn't even try to figure out that one, and decided that duplicating PCRs, LCRs and Common Memory detection, would take too many chips to just double the amount of memory.

Any additional circuitry would have to "know" ALL the details of what the MMU has been "told" to do and, to an extent, duplicate its actions. Common Memory amount and location, the actions of the Pre- and Load Configuration Registers, VIC and DMA access -- all these would require much more that simply capturing a single bit.

To deal with the complexities, and still keep the number of ICs reasonable, what I have done is installed a second MMU, along with minimal support circuitry, which allows the two MMUs to work together, handling all the messy details mentioned above. The results is a machine that can access the new ROM blocks as if it always had them. The operating system already thinks it has the new banks, and initializes them on start-up. BASIC POKE, PEEK, BLOAD, BSAVE and SYS can all access the new banks. So can the Machine Language Monitor. In machine language, you need only store the proper configuration value in the appropriate MMU register, By changing bits 6 and 7 of the RAM Configuration Register at $D506, the VIC chip can display the new banks. I have tested my modification with Speedscript 128, GEOS, Spectrum, CP/M and other software, and encountered no incompatibilities. Should any arise, a switch can be installed to make the machine appear to be un-modified.

There are two limitation, however. None of the additional memory is available in 64 mode. (For now.) And the MMUs cannot relocate zero page and the stack to the added banks. I believe this last could be done but, frankly, I don't think the added complexity would be with worth the effort.

In addition to the DRAMs and 2nd MMU, the circuitry consists of one readily available integrated circuit a 74LS138 plus two more ICs, designated Z3 and Z4 which are available ONLY THROUGH ME. My instruction package includes Z3 and Z4. They may also be ordered separately for $8 (US) that's for both of them.

The modification is not especially difficult, but it is not something for those with no experience in building or modifying electronic equipment. It involves "piggy-backing" of the MMU and DRAMs and wire-wrapping directly to the pins of upside-down ICs. It is simpler than Paul Bosacki's C64 RAM expansion, but somewhat more complex than Andrew E. Mileski's REU expansion. For those who have doubts about their ability to perform my modification, I will do it for $80 (US) plus parts and shipping BOTH WAYS.

Deduct $15 and return and Z3 and Z4 if you've purchased the instruction and changed your mind about doing it yourself. (Owners of low-profile 128s can save on shipping costs by NOT sending the power supply.) For the US and Canada, include your telephone number. My current LOW estimate on parts cost is $35, THIS IS SUBJECT TO CHANGE. The price of memory fluctuates. (I recommend four 64K x 4 DRAMs 4464, 41464, etc. for both the 128D and flat 128s. The instructions explain how to piggy-back 18-pin 64K X 4 DRAMs to the flat 128's 16-pin 64K X 1s.) Jameco Electronics, which advertises in BYTE, COMPUTER SHOPPER and electronics magazines, used to sell the 8722 MMU for $8.95. They no longer list it in their catalog but they might still have some. They have a $50 minimum order. Grapevine, advertising in RUN, list the 8722 for $12.95. By special arrangement, Commodore Service will sell you one for $37!

You can acquire the needed parts and send them to me with your computer and a check for $80, and I will bill you for the return shipping costs. Or, I can get the parts and include them in my final billing.

To summarize, the materials required are (1) 8722 MMU, (1) 74LS138, ICs Z3 and Z4 ($8.00, included with instructions), (4) 64K-by-4bit dynamic RAMs, (2) 68 Ohm, 1/4W resistors, (1) 2K Ohm, 1/4W resistor, 30-gauge wire-wrap wire, solder, electrical tape, "spaghetti", etc.

After much consideration, I have concluded that a modification to provide the 1Meg SuperBanking hinted at in the 128 Programmer's Reference Guide is impractical for both the low-profile and "D" machines. But 512K -- two SuperBanks of 256K each -- seems feasible. Research continues.

**Q:** I have had my CMD hard drive for a few weeks now, and while I am impressed with its speed and capabilities, I have a problem. Although I have had no trouble using Superbase 128 with 1581 mode partitions, my database is quickly growing to the point where I will exceed the capacity of a 1581 partition. Obviously the solution is to use a native mode partition, which I understand can be formatted up to 16 megabytes in size, however in my tests Superbase will not function properly with my data in a native mode partition: HELP?

**A:** One of the advantages of the CMD HD when using programs such as Superbase, is the HD's ability to store more data than most other devices. To realize the potential of the higher capacity of the CMD HD with Superbase does require 'fooling' the software into thinking that a Native Mode partition is really a disk from one of the devices normally supported by Superbase. Two patch programs are provided on the HD UTILITIES disk to assist users in doing this. The programs 1581SUB and 1541SUB will create Native Mode subdirectories on the HD which will emulate the directory placement of the appropriate type of disk drive.

Deciding which of these two utilities to use is determined by two factors: (1) the version of Superbase which you are using, and (2) how large the Native Mode partition is in which you wish to place your Superbase data. If you are not using Superbase version 3 or higher, use 1541SUB (earlier versions of Superbase do not support the 1581). If you are using Superbase version 3 or higher, and the size of the Native Mode partition you wish to use is 38 traeks (9728 blocks) or less, use 1541SUB. If you are using Superbase version 3 or higher, and the size of the Native Mode partition you wish to use is 40 tracks (10240 blocks) or more, use 1581SUB. To avoid problems, it is best not to use partitions with a total number 36 to 39 tracks (9216 to 9984 blocks). If you aren't certain of the number of blocks assigned to a partition, use HD-TOOLS to view the partitions, or create a new Native Mode partition for this purpose.

***Warning: never use a partition which already contains files or data - all data will be erased within the selected partition.***

The following steps outline how to use the 1541SUB or 1581SUB programs to create a special subdirectory:

1. Load the appropriate patch program (1541SUB or 1581SUB).
2. Run the program.
3. Enter the device number of the HD at the prompt.
4. Enter the partition number for the partition you wish to use.
5. Enter a name for the special subdirectory.

The following steps outline the use of the special subdirectory:

1. Before loading Superbase, make the special partition your current partition using the Change Partition (CP) command.
2. Make the special subdirectory your current directory by using the Change Directory (CD) command.
3. If you are booting Superbase from the HD, use the Change Partition command to make the partition which holds the Superbase program your current partition.
4. Boot Superbase and press CONTROL-Q at the first screen in Superbase to bypass the data disk options (if you use the default start.p).
6. If you booted Superbase from the HD, use MAINTAIN/OTHER to make the special partition your current partition (the CP command).
7. From the main menu in Superbase enter: database"database name",dv

Substitute the name of the database you wish to use or create for "database name", and substitute the device number of the HD for dv. Here is an example: database"business",9

Create the database if you are asked to do so. Then enter: file"filename" Substitute the name of the file you wish to use or create for "filename". You will only need to create the database and file the first time you use the database or file in this area of the HD.

**Q:** Is there any advantages to using 100ns or faster chips when I upgrade my VDC RAM?

**A:** Yes, the retailer you bought the more expensive chips from will be able to take a longer vacation trip. The 128 can't access faster VDC RAM chips at a more rapid rate. Anything faster than 120ns is wasted and 150ns chips are adequate. 150ns chips are used in the standard 64K VDC RAM in the 128D.

**Q:** I've read in overseas's Commodore mags that Commodore is introducing a new game console based on the C64. Is this machine the rumored C65, and is it at least partially C128 compatible...What's the scoop?

**A:** It's called the 64GS, and you're right it's essentially a diskless and keyboardless C64, with a cost-reduced motherboard design. Nintendo has yet to make a substantial dent on many European markets and Commodore Europe hopes to grab a share of the lucrative European game playing market with this low cost product. Though it has little chance of being sold in North America, the C64GS is another product designed by Fred "The Father of the C128" Bowen. By the way, don't confuse this machine with the accelerated/improved Commodore 8 bit machine that has occasionally born the GS name in rumors. This machine, which we still call the C65 ourselves, could bear another name when (or more likely if) it is finally introduced. Like the 64GS, it looks like the C65 will not be a North American product either.

continued on page 8

## Continued From Page 7

**Q:** A Freak electrical storm occurred near my home which fried every chip and board in my system. The C128 was irreparable. I made a necklace from the now useless JiffyDOS chips and gave it to one of my daughters. What do you guys think of that?

**A:** Sounds like a great idea, and one worthy of additional commercial exploitation. A motorcycle enthusiast friend of ours thinks that Harley Davidson makes just about as much money from licensing its name and logo as it does selling motorbikes. Perhaps Commodore should start working on a line of Official 8 Bit Jewelry. Think of it: 6581 SID chip ear rings (buy two, get Stereo SID!), 6512 tie tacks, 1351 Mouse belt buckles (left button unclasps your belt, right button as usual, does nothing).

**Q:** I'm trying to locate a 1581 disk drive. Where can I get one?

**A:** Try want ads, user group swap meets, dealers with a dusty box on a back shelf, and buy/sell areas of online services. Unfortunately we have to report that the 1581 has been officially discontinued by Commodore. The Western Digital controller used in the 1581 is no longer available in mass quantities, therefore the current 1581 design can no longer be mass produced. Of course the 1581 could be redesigned to use a different controller chip, however Commodore doesn't feel that expending the R&D costs associated with this endeavor would be justified.

**Q:** Which address should I send orders for TC128 stuff to, the Box 4625 in St. Paul or the Box 11578, 55411 in Minneapolis? Qlink has a message in the TC128 Hardware section using the St Paul address and that is the address on the cover of TC128 Compendium Book #1, but I see the Minneapolis address used in the magazine and many of your other products.

**A:** Parsec Inc. is the new publisher of Twin Cities 128, therefore you should order TC128 subscriptions and TC128 Compendium Book #1 directly from Parsec. However if you are interested in Voyager Mindtools products such as 500 C128 Questions: Answered, IPaint, or Dialogue 128 you should mail your inquires to Voyager Mindtools Inc., P.O. Box 11578, Minneapolis MN 55411. We no longer maintain the St. Paul box.

## Continued from Page 5

### More Deals

OK, you're a new CEO of a major North American computer corporation. You've got a good deal on a used C128 and have hooked up a CMD hard drive and called QLink to get some more stereo SID music and see what kind of graphics you can translate with your copy of Ante Up Version 2. But something is missing. That's right, you need a modern graphical user interface to put it all together. Something like...GEOS! In fact you'd like lots of GEOS stuff. Well the GEOS people have extended their half-price software offer, so GeoEverything will cost you:

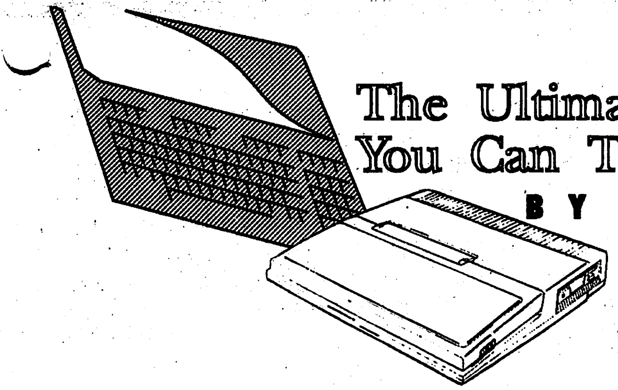
| | Special | Regular |
|---|---|---|
| GEOS Clip Art Disk | $9.95 | $19.95 |
| GeoChart | 14.95 | 29.95 |
| DeskPack Plus | 14.95 | 29.95 |
| FontPack Plus | 14.95 | 29.95 |
| Intnl. FontPack | 14.95 | 29.95 |
| GeoFile | 19.95 | 49.95 |
| GeoProgrammer | 24.95 | 69.95 |
| GeoPublish | 24.95 | 69.95 |
| GeoCalc 128 | 24.95 | 69.95 |

You can send your orders to Berkeley Softworks Fulfillment Center, 5334 Sterling Center Drive, Westlake Village, CA 91361. And Jim, if you are still using GEOS versions that preceded V2.0, you may want to upgrade now, because Berkeley won't be providing any further replacements for V1.2, V1.3, and V1.4 disks that go bad. To upgrade to GEOS 128 V2.0 from an earlier version, send in the top cover of your GEOS manual with a check for a total $19.95, or call the Customer Service Department at (415)644-0926.

### Blast From the Past

One more CEO story. Remember Thomas Rattigan, the man who headed up Commodore during the crucial 1985-1987 period? Mr. Rattigan's departure from Commodore was as rancorous as any up to and including the affair d'Tramiel pere and fils, climaxing with Rattigan being removed bodily from the Commodore headquarters by security guards. The dispute lead to a suit being filed by the ousted CEO against Commodore calling for $9 million in damages. This suit then slowly wended its way through the courts until it was settled this February in New York with a decision favoring Thomas Rattigan. The actual damage award has yet to be announced, but perhaps all this has instantly increased James Dionne's job security, or again maybe this is one of the reasons behind the much more decorous handling of the switch this time.

continued on page 19



# The Ultimate C128 Peripheral You Can Take With You

BY FRANK HUDSON

We have an inside joke here at TC128's world headquarters. Whenever we need to pack up a 128 to take to a show, user group meeting or other site, one of us will pick up the flat 128 keyboard/CPU case, tuck it under a arm while striking a jaunty pose and announce: "Commodore 128 and Student Heading for Class!"

For those of you who can still laugh at a joke after reading a footnote, the explanation can be found in the series of color pictures bound between pages 92 and 93 of the original 128 reference manual. The pose refers to our favorite example from among several knee-slappers included there. Intended to show Commodore's hopes for the kind of uses the C128 would find, half of the systems pictured conspicuously lack disk drives or monitors. The red-shirted student crossing the quad with only a C128 keyboard/CPU under his arm must be trusting that a video monitor will be available at his destination where he will type in his spreadsheet or word processor line-by-line in BASIC 7.0 immediate mode instead of loading it from disk and even this assumes that he has a 128 power brick splitting the seams of his out-of-view left pocket.

Our 128's have many virtues, but portability isn't one of them. What can a 128 owner do when he needs to take computing power along and doesn't want to pack up a desktop-bound system? That's where TC128's look at laptops and luggables for the Commodore 128 owner comes in. One of the 128's strong suits, versatility, means we have a lot of choices to look at, so let's get started.

First off, I'm going to assume that most of you are interested in machines that cost less than $1000. Under our $1000 ceiling, I'm going to divide up things by their level of C-128 compatibility, not cost, since many of the contenders are available chiefly through used and liquidation channels at widely varying prices.

## The Mode Compatibles

There is nothing available that runs the C-128's native mode in a take-out package, but the 128's other two modes are represented in the used market. Both types of machines below are "luggables", suitcase-sized and vacation-packed-weight machines that require an AC power source to run.

## The SX-64

Commodore thought this machine would attract the busy business executive to the C-64 market. It didn't. However it's eventual liquidation at attractive prices meant that this machine is an ever present sight at any Commodore user group meeting and good choice for 128 owners who need to pack up and go. For all intents and purposes it is a C-64 with a built in 1541, and a very small 40 column monitor (external monitors can be used). It shares the 64's relationship with the 128, including a similar DOS and shared disk format. If you have a stash of 64 software that dates back to your pre-128 days, it will run on this machine, and the SX-64 at least gives you a reason to use the top side of those flippy 64/128 Pocket 2 series disks.

**Strengths:** Software and DOS familiarity/Commodore brand-loyalty/color display/fullest disk format compatibility/former CBM head Jack Tramiel picked the SX prefix because he thought it would remind buyers of SeX

**Weaknesses:** Some examples have weak power supplies-this is not the 64 to plug a 1750 into!/small and only 40 column display/weight/AC power only
Price Range: $200-$350

## The CP/M Luggables

The 128's other mode, CP/M, virtually started the transportable computer craze with the release of the original Osborne computer. However the Osborne was soon outgunned by better models in a similar package from Kaypro. Often sold with a considerable amount of bundled CP/M software, these machines were CP/M's last gasp as MS-DOS took over the business market. The Kaypro's especially have a reputation for being quite rugged, but despite this they are rare on the used market, with many machines continuing to do duty as second or third computers with their first families. Like the SX-64, they feature at least one built in drive and a scaled down video monitor. While the CP/M machines are monochrome, their displays are more readable than the SX-64's. 128'ers with 1571's should be able to exchange data on disks fairly easily. Additional formats are supported by programs such as Format Executive and Big Blue Reader. If you use or are comfortable with the CP/M mode of your 128, these machines will be easy for you to use.

**Strengths:** Fair amount of disk compatibility/good choice for C-128 CP/M fanciers/very readable displays

**Weaknesses:** Must learn or understand CP/M to use them/weight/AC power only/eroding software base
Price: $150-$500

## The MS-DOS compatibles

Any owner of 1571 or 1581 disk drive has the ability to read and write to MS-DOS disk formats via Big Blue Reader or Format Executive, making the ubiquitous MS-DOS portable a good choice for 128 owners. However, MS-DOS shares little else with the 128. Software prices can be high, the DOS syntax alien (unless you are a CP/M user), and performance with the older 4.77 MHz models will make you wonder what all the fuss was about the power of the PC back in the early 80's. While many of the latest generation of laptops are above our one grand ceiling, used, liquidation, and low-end models come in under the qualifying figure. There are so many models available that I'll have to batch them together.

## The Discontinued Models:

Too many examples to mention, but any issue of Computer Shopper or a visit to your local electronics liquidation parlor should alert you what is currently being offered. Frequently being sold at less than a

continued on page 10

continued from page 9

quarter of their original price, these machines can feature first-class construction balancing with limited configurations. The cheapest may not have 640K of RAM and therefore will not run current versions of the most popular MS-DOS packages. Other bargains feature a "blazing" 4.77 MHz processor speed and dim hard-to-read LCD screens. Even with these limitations discontinued machines can be still be a good deal if you know what you are getting into, Older software and efficient PD and shareware offerings often run even on the 256K machines, and some even come with a built-in modem. Sizes range from almost modern clamshell laptop designs to "lunchbox" shaped larger battery powered models, all the way up to the suitcase AC powered "luggable" units. Prices for discontinued models are all over the map too. Whatever you do, don't believe anyone who tells you that you can fix some shortcoming in the configuration they're selling with an easy upgrade. Add-ons for laptops are notoriously expensive. Shop carefully, don't expect to be running Windows or Word Perfect 5.1 on it, and you can find some real buys here. Two models we have known and loved despite their warts are the IBM Convertible with a great keyboard, beefy construction, only 256K, and an especially bad LCD screen; and the Zenith 171 lunchbox, with a great backlit display, full complement of RAM, built in 1200 bps modem, worthwhile ROM utilities, small and slow 5.25 inch disk drives, and sluggish screen updates.

As you reach the top of our arbitrary price $1000 price range you can purchase some more broadly capable machines these days. XT class portables with hard drives, a smattering of older and heavier AT class machines, and even the clever lightweight Zenith MiniSport have all been offered in recent ads for less than a grand. The machines with hard drives are an especially good deal if you plan on running modern hard-drive-dependent MSDOS applications.

**Strengths:** Disk format transferable to C-128/reasonable size/wide-based MS-DOS software library

**Weaknesses:** May not be up to some modern MS-DOS software packages (but your wallet may not be either!)/may have hard-to-remedy hardware shortcomings
Prices: $200-$1000

## Budget Priced New MS-DOS laptops

The Toshiba T1000 laptop defined this market for the past few years, and this model remains one of the best deals low-cost portables. Street prices for the T1000 vary from just under $600 to $800 tops, and it is one sweet little machine. Weight is about six pounds, battery life is good (five to six hours), DOS is in ROM, the keyboard is acceptable, and its LCD display is not bad for a non-backlit model. The chief option is a 768K battery-backed REU, the value of which any Quick Brown Box or 1750 user will quickly understand. Surprisingly, there is still little competition for this popular model. Its closest competitors are a pair laptops produced as a joint venture by Tandy and Panasonic. The Tandy 1100FD is priced under $1000 (it comes closer to the T1000's "street" price during one of Radio Shack's frequent sales). By and large, the 1100FD is comparable to the T1000. Differences? There is no RAM disk option for the 1100FD and the Tandy product has their everpresent DeskMate utilities and a newer version of DOS (3.3 vs. 2.2) in ROM. I'd give the Tandy a point for its larger LCD display, but not everyone agrees on that one. Frankly you'd have to like DeskMate in ROM a whole lot to want to pay more for the 1100FD. The Panasonic CB150 is the same machine as the 1100FD, but with two important options unavailable at Radio Shack. The CB150 can be purchased with a backlit screen, and it has a battery-backed RAM disk option like the Toshiba. Both the CB150 and the Tandy 1100FD have double the processor speed of the Toshiba, but this does not translate directly to double the speed in typical user operations. Widely sold at discount houses and through the mail, the Panasonic CB150 and Toshiba T1000 are close enough in capabilities that we can tell you to make your choice between them based on vendor and price. Because none of these machines has a hard drive, they lack the ability to run a fair portion of the most popular MSDOS software. However, those with the RAM disk options can access files and applications with hard drive speed.

**Strengths:** Light weight/disk transferability with the 1581/decent battery life/wide-based MS-DOS software library/buying a new machine means better guarantee and service situation

**Weaknesses:** Non-backlit displays can strain the eyes/will not run software requiring hard drives or more modern processors/MS-DOS operating system not familiar to all 128'ers
Price: $550-$1000

## The Odd-Balls

If you bought a C-128 because of its popularity, you can skip this section. But if you hang onto your 128 because you like it and don't give a clock-cycle about what others think about it, read on. There is a whole gumball machine of weird stuff out there, some of it at very attractive prices. None of these machines come with disk drives - much less a compatible 128 disk format. Some of them are small enough to be crushed under a 6 pound T1000. All of them have enough novelty value to start up a conversation with other portable users you cross paths with. File transfer will have to be via modem or null-modem to and from your 128, but the extra trouble is just part of the experience of hardy individualism.

## The Radio Shack Workslates

The Tandy 102 laptop is the successor to the Model 100 that launched the laptop boom several years ago. The 102 has 32K RAM, a built in 300 baud modem (remember you'll need it to transfer files), limited but functional software in ROM, and a short 40 column by 8 line LCD display. In return for it's limits, it weighs only 3 pounds, is smaller than even the T1000, can function up to 20 hours on the same kind of AA batteries you can grab in any drug store and has the best keyboard of any computer in its size range. Unfortunately, at $600 it's overpriced in the current market. Used or during a Tandy sale it might be available at something like it's true value. The older Model 100's are functionally similar, and also show up used at good prices. The Tandy's are durable machines that can survive being repeatedly dropped.

A less expensive Tandy option for those who want to work with words is the WP-2 Wordprocessor. Not to be confused with those rip-off dedicated word processors being sold in department stores, this unit is about the same shape and weight as the Tandy 102. It has a 80x8 LCD display and like the 102 it runs off AA batteries. The WP-2's ROM software includes a spell checker and thesaurus as well as a limited telecommunications support. There is no built in modem, but a RS232 port is provided. Priced at $350, it's the better Radio Shack deal.

## The Stealth Laptop

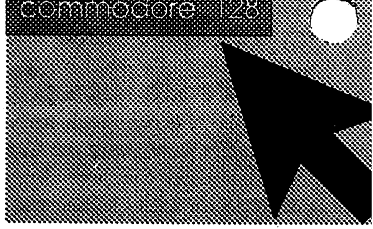
Three pounds too much for you? Nostalgic for your old Timex Sinclair? How about the Cambridge Z88, from the (fertile/febrile) mind of Clive Sinclair? Imagine the already tidy Tandy models above with a few inches and a full pound shaved off them and you've got the Z88: 64K, applications in ROM, 8X80 screen, 20 hours on plain AA's, and file transfer via RS232 idea. True to the Sinclair tradition, the silent rubber keyboard is a weak point. Price is $499.95 and you can contact Sharp's Inc. at Box 326 Mechanicsville, VA 23111 for further information. Tell'em TC128 sent you.

## There's the Computer - Under the Checkbook

Want to spend your $300 bucks on something even smaller? Want to buy a machine from Jack Tramiel's company once again? How about the Atari Portfolio. Sure they talk about this machine being a MSDOS compatible, but that's about as shaky a line as the old C-64 CP/M cartridge promise. So forget that stuff, and dig it for what it is. A one pound, 128K pocket computer. A keyboard right (above/below) the borderline of the smallest possible touch typing surface. Again applications in ROM, but this screen is only 40 columns wide (although 20 lines high). The Portfolio seems to be a litmus test of how much small size means to portable users. It's either the breakthrough to the coat pocket machines of the future or just too tiny to be any use, depending on who you ask. Another catch is that the RS-232 port you'll need to get stuff in and out is an extra cost, hang-it-on-the-outside, option.

**Strengths:** The ultimate in affordable small size and weight/excellent battery life/low prices even on new machines

**Weaknesses:** Software base is very limited compared to other options/no disk transfer possible/cramped screens/limited dealer network (except for the Tandy products)
Price: $300 to $600



# A *Sprite* Pointer for 80 Column Text

by Rick Kane

Almost everyone has experienced the ease of "point and click" selection. Instead of scanning a menu for the option you want, finding out what key (or combination of keys!) to press, hunting for the correct key and pressing, you just move a pointer to the option you want on the screen and select it. While some jobs become tedious if you have to wade through menus without keyboard shortcuts, often the easiest, most intuitive way of getting where you want to go is through the use of Point and Click.

You run into certain difficulties when trying to implement point and click in your own programs for the C=128 . First, while it is very easy to define a VIC sprite for the 40 column screen (which will work with either text OR bitmapped graphics), there is no provision for sprites on the 80 column screen. You could do all your programming in GEOS, but then you are requiring that your end-user 1) own GEOS and 2) be willing to boot it up for what might have been a relatively small utility using the keyboard method. Or you could use BASIC 8, using the Run-Time Library so the user doesn't have to own it. But these are both memory-hungry solutions; you may have wanted to use all that memory for, say, a fast name-and-address database. Bitmapped graphics screens, simple require more memory even for simple things like menus.

Here is a solution to most of these difficulties. "Pointer-80" works on the 80 column text screen much the way a pointer works on the bitmapped screens of GEOS and BASIC 8. This means you can use the standard CBM graphics characters to create boxes, "tools", etc. which can be selected just like on the bitmapped screens. In addition, almost no new commands are needed - this pointer responds to the BASIC 7 MOVSPR command that controls the position of sprites on the 40 column screen. In fact, the 80 column pointer mimics the position of its "sister" sprite on the 40 column screen! What this means is that you don't have to rewrite everything for one screen or the other; the pointer position can be made the same on both screens. Just place your menu selections in the same visual position on both screens, and don't worry whether they are in 40 or 80 column mode. And since it works with sprite registers, it works with the 1351 mouse driver from the Commodore 1351 Mouse Utility Disk, and the "joystick driver" program found in this issue (even both at the same time!).

## How It works

Several things are going on to make all this possible. First, how do you make a pointer that passes non-destructively over text on the 80 column screen? The answer to this lies in the redefinable character sets. The definition of the character under the desired pointer position is copied to that of a non-visible character. Then, the pointer definition is mapped onto that. In this case, first a "mask" definition is used, creating a blank "hole" in our new character. Then the pointer's "positive" image is OR'ed into place. The result is a pointer that is always separated from its surroundings by an OFF outline. This is how GEOS creates a pointer. Without the mask, the pointer can get lost on some graphic or reversed characters; especially if its color is turned off. Try it on some of the reversed spaces in the demo program.

OK, now we have a pointer. What do we do to make it move? Pointer-80 is interrupt-driven. It follows the VIC sprite x and y position registers and calculates the corresponding position on the 80 column screen. (These are "shadows" of the actual VIC registers; the x position is peeked at decimal 4566, with 256 added if bit 0 of location 4582 is set, y position is at decimal 4567.) Then it places the pointer. Of course, it must be sure to erase any previous pointer or things would quickly look pretty lousy. Also, you should note that on the 40 column screen, sprites' coordinates are measured from point 24 (horizontal), 50 (vertical) at the upper left to 343(h),249(v) at the lower right. This is to allow a sprite to move off the visible screen. Pointer-80 preserves this convention. A sprite's on-screen position ranges 320 pixels horizontal, 200 pixels vertical. Since you read the same sprite registers to determine where the 80 column pointer is, remember that its position is HALF what it would be if it were positioned using 80 column coordinates. Also, don't forget the offset (24 or 50) to the upper left corner of the screen.

## Controls

As mentioned, Pointer-80 responds to the MOVSPR command. S your user's guide, or the demo listing, for more information on this command. The following controls are also at work:

| | |
|---|---|
| sys 4864 | sets/releases interrupt sprite link. Pointer will always be ON when this is set. |
| sys 4867 | pointer ON (with interrupt released) |
| sys 4870 | pointer OFF only works with interrupt link released. |
| poke 4873,color | color = 0-15; if color > 127 then color will be taken from the character beneath the pointer. |

## Special Notes

Pointer-80 rounds off the pointer's position to the nearest character. There is nothing smaller that could be selected on a text screen anyway, and it would require more and slower code to "smooth scroll" the pointer (also 4 non-visible characters would need to be redefined instead of just one, to allow the pointer to spill over horizontally and vertically). One note of caution: Pointer-80 doesn't check when you are adding text to the screen, or scrolling; be sure to move the pointer off screen before printing or leftover pointers will remain when the screen scrolls. This is an interesting effect to watch, though, so I'm sure you'll try it!

The demo program simply shows how you can detect which selection has been made, as well as a few interesting MOVSPR demos. Positioning of menu items is critical. When typing it in, be careful to type spaces before and after "Show X-Y", "END", and the numbers 1-5. Also beware of extra carriage returns or cursor-down's.

If you select "Show X-Y", the demo will be in fast mode; which means you won't see the 40 column screen. Select "Show X-Y" again to go to slow mode, then switch your monitor back and forth between the two screens to see the correlation between pointer movement on each.

After typing in the "make.pointer-80" program and saving it to disk, run it with an unimportant disk in the drive. It will make a copy of

After typing in the "make.pointer-80" program and saving it to disk, run it with an unimportant disk in the drive. It will make a copy of "Pointer-80" on the disk. If all went well, copy "Pointer-80", the demo, the 1351 mouse driver ("m1351.128.bin") and the joystick sprite driver from this issue to the same disk.

## MAKE.POINTER-80

```
120 D=PEEK(186):PRINT"READING POINTER-80 DATA ...":RESTORE
125 I=0:DO:READ A$:IF A$="END" THEN EXIT
130 POKE DEC("1300")+I,DEC(A$):I=I+1:LOOP
140 PRINT"DONE!":PRINT"INSERT A DISK INTO DRIVE";D;"AND PRESS RETURN"
150 DO:GETKEYK$:LOOP UNTIL K$=CHR$(13)
160 BSAVE"POINTER-80",U(D),B0,P(DEC("1300"))TOP(DEC("1512")):END
500 DATA 4C,1A,13,4C,6C,14,4C,C7,14,0D,00,60,78,7E,78,4C
510 DATA 06,00,3F,0F,03,00,03,01,F0,F8,08,AD,14,03,C9,54
520 DATA D0,07,AD,15,03,C9,13,F0,19,78,AD,14,03,8D,FC,14
530 DATA A9,54,8D,14,03,AD,15,03,8D,FD,14,A9,13,8D,15,03
540 DATA 28,60,20,C7,14,78,AD,FC,14,8D,14,03,AD,FD,14,8D
550 DATA 15,03,28,60,BD,07,01,C9,C4,90,04,C9,D0,90,23,20
560 DATA A7,14,20,C7,14,20,85,13,C9,50,B0,13,20,9A,13,C9
570 DATA 19,B0,0C,20,A7,13,20,02,14,20,51,14,20,71,14,20
580 DATA B7,14,6C,FC,14,18,AD,E6,11,29,01,F0,01,38,AD,D6
590 DATA 11,6A,4A,38,E9,06,8D,00,15,60,AD,D7,11,38,E9,32
600 DATA 4A,4A,4A,8D,01,15,60,0A,0A,18,6D,01,15,0A,8D,03
610 DATA 15,A9,00,8D,02,15,A2,03,18,0E,03,15,2E,02,15,CA
620 DATA D0,F6,AD,03,15,18,6D,00,15,90,03,EE,02,15,8D,03
630 DATA 15,AD,02,15,A2,12,20,CC,CD,E8,AD,03,15,20,CC,CD
640 DATA 20,D8,CD,8D,04,15,AD,02,15,18,69,08,8D,05,15,A2
650 DATA 12,20,CC,CD,E8,AD,03,15,20,CC,CD,20,D8,CD,8D,06
660 DATA 15,60,A9,00,8D,0F,15,AD,04,15,A0,04,18,0A,2E,0F
670 DATA 15,88,D0,F8,8D,10,15,18,A9,20,6D,0F,15,8D,0F,15
680 DATA AD,06,15,10,09,A9,10,18,6D,0F,15,8D,0F,15,AD,0F
690 DATA 15,A2,12,20,CC,CD,AD,10,15,E8,20,CC,CD,A0,00,20
700 DATA D8,CD,39,12,13,19,0A,13,99,07,15,C8,C0,08,90,EF
710 DATA 60,A9,2E,A2,12,20,CC,CD,E8,A9,00,20,CC,CD,A0,00
720 DATA B9,07,15,20,CA,CD,C8,C0,08,90,F5,60,AD,11,15,D0
730 DATA 35,A2,12,AD,02,15,20,CC,CD,E8,AD,03,15,20,CC,CD
740 DATA A9,E0,20,CA,CD,A9,01,8D,11,15,A2,12,AD,05,15,20
750 DATA CC,CD,E8,AD,03,15,20,CC,CD,AD,09,13,10,05,AD,06
760 DATA 15,29,0F,20,CA,CD,60,A2,12,20,DA,CD,8D,FE,14,E8
770 DATA 20,DA,CD,8D,FF,14,60,A2,12,AD,FE,14,20,CC,CD,E8
780 DATA AD,FF,14,20,CC,CD,60,AD,11,15,F0,2F,A2,12,AD,02
790 DATA 15,20,CC,CD,E8,AD,03,15,20,CC,CD,AD,04,15,20,CA
800 DATA CD,A2,12,AD,05,15,20,CC,CD,E8,AD,03,15,20,CC,CD
810 DATA AD,06,15,20,CA,CD,A9,00,8D,11,15,60,00,00,00,00
820 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
830 DATA 00,00,END
```

### Joystick Driver

If you have a 1351 mouse, you know a little about "drivers". The 1351 Utilities Disk includes a mouse driver which positions a sprite on the 40 column screen as you move your mouse. If you don't have a mouse, you may have dealt with trying to move a sprite quickly from basic. Although BASIC 7 has commands which improve this task greatly over the 64, there is still a need for a transparent "driver" to silently control a sprite. This small program does just that. As with the 1351 mouse driver, it "latches" sprite #0 to the movement of the joystick. It takes the work out of moving a sprite with the joystick - and the sluggishness of doing it in BASIC.

The joystick driver resides at $1900, which doesn't conflict with the 1351 mouse driver. It reads joystick port 2, while the mouse driver generally reads port 1. In fact, if the mouse driver is installed first, the two will coexist! You can move the pointer with the mouse, then pick up the joystick for some critical straight line movement, then go right back to the mouse.

SYS 6400 ACTIVATES THE JOYSTICK SPRITE DRIVER. OF COURSE, YOU NEED TO DEFINE AND ACTIVATE SPRITE # 0, OR USE "POINTER-80", TO SEE ANYTHING!

SYS 6405 DEACTIVATES THE JOYSTICK DRIVER, RESTORING THE VECTORS TO THEIR PREVIOUS SETTINGS. THERE ARE ONLY TWO OTHER IMPORTANT LOCATIONS: DECIMAL 6403 & 6404. THESE HOLD THE X AND Y INCREMENT, WHICH ARE ADDED EACH TIME THE JOYSTICK IS POINTED.

After typing in and saving the "make.joydriver" program, run it using an unimportant disk. It will save a copy of the joystick driver to the disk. You may then copy it to any disk you wish.

## MAKE.JOYDRIVER

```
120 D=PEEK(186):PRINT"READING JOYDRIVER DATA ...":RESTORE
125 I=0:DO:READ A$:IF A$="END" THEN EXIT
130 POKE DEC("1900")+I,DEC(A$):I=I+1:LOOP
140 PRINT"DONE!":PRINT"INSERT A DISK INTO DRIVE";D;"AND PRESS RETURN"
150 DO:GETKEYK$:LOOP UNTIL K$=CHR$(13)
160 BSAVE"JOYDRIVER",U(D),B0,P(DEC("1900"))TO 1xDEC("19B7")):END
500 DATA 4C,1D,19,04,02,AD,15,03,C9,19,D0,10,08,78,AD,B3
510 DATA 19,8D,14,03,AD,B4,19,8D,15,03,28,58,60,AD,15,03
520 DATA C9,19,F0,19,08,78,AD,14,03,8D,B3,19,AD,15,03,8D
530 DATA B4,19,A9,3E,8D,14,03,A9,19,8D,15,03,28,60,D8,AD
540 DATA 7E,11,D0,33,AD,D6,11,8D,B5,19,AD,D7,11,8D,B6,19
550 DATA AD,00,DC,48,29,01,D0,03,20,7A,19,68,48,29,02,D0
560 DATA 03,20,85,19,68,48,29,04,D0,03,20,90,19,68,48,29
570 DATA 08,D0,03,20,9D,19,68,6C,B3,19,AD,D7,11,38,ED,04
580 DATA 19,8D,D7,11,60,AD,D7,11,18,6D,04,19,8D,D7,11,60
590 DATA AD,D6,11,38,ED,03,19,8D,D6,11,90,0E,60,AD,D6,11
600 DATA 18,6D,03,19,8D,D6,11,B0,01,60,AD,E6,11,49,01,8D
610 DATA E6,11,60,00,00,00,00,END
```

## POINTER-80.DEMO

```
5 D=PEEK(186)
10 A$="M1351.128.BIN":OPEN2,D,2,A$+",P,R":CLOSE2:IFDS=0THENBOOT(A$),U(D)
15 A$="JOYDRIVER4.O":OPEN2,D,2,A$+",P,R":CLOSE2:IFDS=0THENBOOT(A$),U(D)
20 A$="TPOINTER3.61.O":OPEN2,D,2,A$+",P,R":CLOSE2:IFDS THEN END
25 BOOT(A$),U(D)
30 TRAP300:DEFFNX(X)=PEEK(4566)+256*(PEEK(4582)AND1):DEFFNY(Y)=PEEK(4567)
32 DIMC(16)
34 DATA  144,5,28,159,156,30,31,158,129,149,150,151,152,153,154,155
36 RESTORE 34:FOR I=1 TO 16:READ CD:C$=C$+CHR$(CD)+"   ":NEXT I
40 GOSUB500:REM DEFINE 40 COL SPRITE
50 SPRITE 1,1,1,0,0,0
60 MOVSPR1,150,150
70 GOSUB1000:REM MENU
100 DO:GOSUB900:LOOPUNTIL(JOY(1)>0)OR(JOY(2)AND128)>0
110 X=FNX(0):Y=FNY(0):IFY>97ANDY<106THENGOSUB600:GOTO100
120 IFY>113ANDY<122THENGOSUB650:GOTO100
130 IFY>129ANDY<138ANDX>43ANDX<84THENGOSUB700:GOTO100
140 IFY>129ANDY<138ANDX>123ANDX<144THEN300
150 IFX>63ANDX<76THENGOSUB400
190 GOTO100
300 GOSUB420:END
399 END
400 J=1+INT((Y-154)/8)
405 I=0:DO:ONJGOSUB450,455,460,465:LOOPUNTIL(JOY(1)>0)OR(JOY(2)AND128)>0
410 MOVSPR1,0#0
420 DO:LOOPUNTILJOY(1)=0:DO:GETK$:LOOPUNTILK$="":RETURN
450 MOVSPR1,180-180*SIN(I/)#15:I=I+1:RETURN
455 MOVSPR1,I# 8:I=I+4:RETURN
460 MOVSPR1,10;I:I=I+4:RETURN
465 MOVSPR1,+160*((I=1)-(I=0)),+0:I=-(I=0):RETURN
500 RESTORE510:A$="":FORI=0TO62:READA:A$=A$+CHR$(A):NEXT:SPRSAVA$,1
505 FORI=0TO16:READA:C(I)=A:NEXT:RETURN
510 DATA 254,0,0,132,0,0,136,0,0
520 DATA 132,0,0,162,0,0,209,0,0,136,128,0,4,128,0,3,0,0
540 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
560 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
580 DATA 0,15,8,7,11,4,2,13,10,12,9,6,1,5,3,14,128:REM POINTER COLORS
600 C=INT((X-84)/12):IFC<0ORC>16THENRETURN
610 POKEDEC("1309"),C(C):RETURN
650 C=INT((X-84)/12):IFC<0ORC>15THENRETURN
660 COLOR6,C+1:RETURN
700 PRINTCHR$(19)CHR$(27)"Q":SH=-(SH=0):IFSHTHENFAST:ELSESLOW
710 RETURN
900 IFSHTHENPRINTCHR$(19)CHR$(27)"Q";FNX(0),FNY(0):RETURN:ELSERETURN
1000 COLOR6,1:COLOR5,6:SCNCLR:PRINT:PRINT:PRINTCHR$(14);TAB(15)"80 COL 'SPRITE'"
1010 PRINT:PRINT:PRINT:PRINT"POINTER COLOR
:";CHR$(18);C$;CHR$(30);CHR$(2);CHR$(146);"OFF";
1015 PRINT TAB(15);C$:COLOR5,6:PRINT"  SCREEN COLOR:";CHR$(18);C$
1020 COLOR5,6:PRINT:PRINT TAB(5);CHR$(18);" SHOW X-Y ";TAB(25)" END "
1025 PRINT:PRINTTAB(5);"MOVEMENT DEMO:":FORI=1TO4:PRINTTAB(10);CHR$(18);
     STR$(I);" ":NEXT
1090 RETURN
```

## TEXT DRIVER Source Code

```
WRITEMEM   = $CDCA
WRITEREG   = $CDCC
READMEM    = $CDD8
READREG    = $CDDA
SPX        = $11D6
SPXHI      = $11E6
SPY        = $11D7
IIRQ       = $0314

           ORG $1300

JINSTALL   JMP INSTALL
JPNTRON    JMP PNTRON
JPNTROFF   JMP PNTROFF

PNTRATTR   HEX 0D                   ;YELLOW
                                    ;POINTER PIXELS
PNTRDEF    HEX 00                   ;........
           HEX 60                   ;.**.....
           HEX 78                   ;.****...
           HEX 7E                   ;.******.
           HEX 78                   ;.****...
           HEX 4C                   ;.*..**..
           HEX 06                   ;.....**.
           HEX 00                   ;........
                                    ;BACKGROUND BITS
MASK       HEX 3F                   ;..******
           HEX 0F                   ;....****
           HEX 03                   ;......**
           HEX 00                   ;........
           HEX 03                   ;......**
           HEX 01                   ;.......*
           HEX F0                   ;****....
           HEX F8                   ;*****...

INSTALL    PHP
           LDA        IIRQ
           CMP        #<POINTER
           BNE        START
           LDA        IIRQ+1
           CMP        #>POINTER
           BEQ        QUIT          ;INSTALLED?
START      SEI
           LDA        IIRQ          ;SAVE IRQ
           STA        OLDIRQ
           LDA        #<POINTER     ;SET TO PTR ROUTINE
           STA        IIRQ
           LDA        IIRQ+1
           STA        OLDIRQ+1
           LDA        #>POINTER
           STA        IIRQ+1
           PLP
           RTS

QUIT       JSR        PNTROFF       ;TURN POINTER OFF
           SEI
           LDA        [illegible]   ;RESTORE PREV
           STA        IIR[illegible] ;IRQ VECTOR
```

```
           LDA        OLDIRQ+1
           STA        IIRQ+1
           PLP
           RTS

POINTER    LDA        $107,X        ;CHK 8563 ACTIVITY
           CMP        #$C4
           BCC        DOPNTR
           CMP        #$D0
           BCC        NORMIRQ       ;IF NOT QUIET

DOPNTR     JSR        GETUPDAT      ;SAVE UPDATE
           JSR        PNTROFF
           JSR        SETX          ;CONVERT X POS
           CMP        #80           ;IS IT ONSCREEN?
           BCS        SKIPOUT       ;NO- EXIT

           JSR        SETY          ;CONVERT Y POS
           CMP        #25
           BCS        SKIPOUT       ;IF OFFSCRN-EXIT

           JSR        GETADDR       ;GET LOC IN 8563
           JSR        GETDEF        ;COMB CHAR&PTR
           JSR        DEFPNTR       ;SEND 8563 MEM
           JSR        PUTPNTR       ;PTR TO SCREEN
SKIPOUT    JSR        REUPDT        ;RESTORE ADDR
NORMIRQ    JMP        (OLDIRQ)      ;REST INTERRUPT

SETX       CLC                      ;CNVRT SPRITE POS
           LDA        SPXHI         ;CHECK HI BIT
           AND        #1
           BEQ        NOHI
           SEC
NOHI       LDA        SPX
           ROR
           LSR                      ;DIVIDE BY 4
           SEC
           SBC        #6            ;ADJ OFFSCR LEFT
           STA        SSPX
           RTS

SETY       LDA        SPY
           SEC
           SBC        #50           ;ADJ OFFSCR TOP
           LSR                      ;DIVIDE BY 8
           LSR
           LSR
           STA        SSPY          ;0 - 24
           RTS

GETADDR    ASL                      ;COMP 8563 ADDR
           ASL                      ;START WITH SSPY
           CLC                      ;FIRST MULT BY 10
           ADC        SSPY
           ASL                      ;ALWAYS <= 240
           STA        CHARPLO
           LDA        #0            ;STORE  BYTES
           STA        CHARPHI

           LDX        #3
MULT2      CLC
```

```
           ASL        CHARPLO       ;MULTIPLY BY 8
           ROL        CHARPHI
           DEX                      ;(*2 *2 *2)
           BNE        MULT2

           LDA        CHARPLO       ;LO BYTE OF Y*80
           CLC
           ADC        SSPX          ;ADD X (0-79)
           BCC        SKIPHI
           INC        CHARPHI
SKIPHI     STA        CHARPLO

           LDA        CHARPHI       ;UPDATE TARGET
           LDX        #18
           JSR        WRITEREG
           INX
           LDA        CHARPLO
           JSR        WRITEREG
           JSR        READMEM       ;SAVE OLD CHAR'S
           STA        OLDCHAR       ;SCREEN CODE

           LDA        CHARPHI       ;GET ATTRIBUT ADDR
           CLC
           ADC        #8            ;(NORMALLY)
           STA        ATTRIBHI
           LDX        #18           ;UPDATE TO ATTRIB
           JSR        WRITEREG
           INX
           LDA        CHARPLO
           JSR        WRITEREG
           JSR        READMEM       ;SAVE ATTRIBUTE
           STA        ATTR
           RTS

GETDEF     LDA        #0            ;DEFINE CHAR+PTR
           STA        CHARDHI       ;FIRST GET
           LDA        OLDCHAR       ;"OLD" CHAR CODE
           LDY        #4
MULTCHAR   CLC                      ;MULTIPLY BY 16
           ASL
           ROL        CHARDHI
           DEY
           BNE        MULTCHAR      ;(*2 *2 *2 *2)
           STA        CHARDLO       ;= OFFSET

           CLC
           LDA        #$20          ;SET HIGH BYTE
           ADC        CHARDHI
           STA        CHARDHI
           LDA        ATTR          ;CHECK ATTRIBUTE
           BPL        STDDEFS       ;STANDARD DEFS
           LDA        #$10          ;SET TO ALT DEFS
           CLC
           ADC        CHARDHI
           STA        CHARDHI

STDDEFS    LDA        CHARDHI       ;SET ACTUAL DEF
           LDX        #18
           JSR        WRITEREG
           LDA        CHARDLO
           INX
```

```
           JSR        WRITEREG
           LDY        #0            ;MERGE CHAR&PTR

GETBYT     JSR        READMEM       ;GET CHAR DEF BYT
           AND        MASK,Y        ;KEEP THESE BITS
           ORA        PNTRDEF,Y     ;OVERLAY PTR BYTE
           STA        CHAR,Y        ;SAVE IT IN RAM 0
           INY
           CPY        #8
           BCC        GETBYT
                      RTS

DEFPNTR    LDA        #$2E          ;SET UPDATE TO PTR
           LDX        #18           ;CHAR'S ADDRESS IN
           JSR        WRITEREG      ;8563 MEMORY
           INX
           LDA        #0
           JSR        WRITEREG

           LDY        #0
DEFLOOP    LDA        CHAR,Y
           JSR        WRITEMEM
           INY
           CPY        #8
           BCC        DEFLOOP
           RTS

PNTRON     LDA        PNTRFLAG
           BNE        OUT           ;IF ON, EXIT
PUTPNTR    LDX        #18           ;PTR ON SCREEN
           LDA        CHARPHI       ;TO CHAR POS
           JSR        WRITEREG
           INX
           LDA        CHARPLO
           JSR        WRITEREG

           LDA        #224          ;PTR CHAR CODE
           JSR        WRITEMEM
           LDA        #1            ;SET FLAG
           STA        PNTRFLAG

PUTATTR    LDX        #18           ;PUT PTR ATTRIB
           LDA        ATTRIBHI      ;SET UPDATE ATTRIB
           JSR        WRITEREG
           INX
           LDA        CHARPLO
           JSR        WRITEREG

           LDA        PNTRATTR      ;COLOR OF POINTER
           BPL        PNTRCOLR

           LDA        ATTR          ;IF HIGH BIT SET
           AND        #15           ;CURRENT COLOR
PNTRCOLR   JSR        WRITEMEM
OUT        RTS

GETUPDAT   LDX        #18           ;SAVE UPDATE POS
           JSR        READREG       ;R18,R19
           STA        UPDATEHI
           INX
           JSR        READREG
```

```
          STA      UPDATELO
          RTS
REUPDT    LDX      #18        ;RESTORE UPDATE
          LDA      UPDATEHI   ;ADDRESS
          JSR      WRITEREG
          INX
          LDA      UPDATELO
          JSR      WRITEREG
          RTS
PNTROFF   LDA      PNTRFLAG   ;IF FLAG=0
          BEQ      OFFDONE    ;PTR ALREADY OFF

          LDX      #18        ;SET UPDATE TO
          LDA      CHARPHI    ;PTR TEXT POSITION
          JSR      WRITEREG
          INX
          LDA      CHARPLO
          JSR      WRITEREG
          LDA      OLDCHAR    ;REPLACE STORED
          JSR      WRITEMEM   ;CHARACTER

          LDX      #18        ;SET UPDATE TO
          LDA      ATTRIBHI   ;POINTER ATTRIBUTE
          JSR      WRITEREG
          INX
          LDA      CHARPLO
          JSR      WRITEREG
          LDA      ATTR       ;REPLACE THE
          JSR      WRITEMEM   ;ATTRIBUTE

          LDA      #0         ;TURN OFF FLAG
          STA      PNTRFLAG
OFFDONE   RTS

OLDIRQ    HEX 0000
UPDATEHI  HEX 00
UPDATELO  HEX 00
SSPX      HEX 00
SSPY      HEX 00
CHARPHI   HEX 00
CHARPLO   HEX 00
OLDCHAR   HEX 00
ATTRIBHI  HEX 00
ATTR      HEX 00
CHAR      HEX 000000          ;TEMP 8 BYTES
          HEX 000000
          HEX 0000
CHARDHI   HEX 00
CHARDLO   HEX 00
PNTRFLAG  HEX 00
```

## JOYDRIVER Souce Code

```
IIRQ       = $314
JOY        = $DC00

ACTIVE     = $117E
VICDATA    = $11D6
XPOS       = VICDATA+0
YPOS       = VICDATA+1
XMSB       = VICDATA+$10

           ORG $1900

JINSTALL   JMP INSTALL

XINCR      HEX 04                ;SETS "SPEED"
YINCR      HEX 02

DEACTIV    LDA      IIRQ+1       ;IS IT ACTIVE?
           CMP      #>JIRQ
           BNE      DDONE        ;IF NOT, DONE
           PHP
           SEI
           LDA      OLDIRQ       ;RESTORE VECTOR
           STA      IIRQ         ;PREVIOUS SETTING
           LDA      OLDIRQ+1
           STA      IIRQ+1
           PLP
           CLI
DDONE      RTS

INSTALL    LDA      IIRQ+1       ;IS IT ACTIVE?
           CMP      #>JIRQ
           BEQ      IDONE        ;IF YES, DONE
           PHP
           SEI
           LDA      IIRQ         ;PRESERVE PREV
           STA      OLDIRQ       ;IRQ VECTOR
           LDA      IIRQ+1
           STA      OLDIRQ+1
           LDA      #<JIRQ       ;INSTALL JOY IRQ
           STA      IIRQ
           LDA      #>JIRQ
           STA      IIRQ+1
           PLP
IDONE      RTS

JIRQ       CLD
           LDA      ACTIVE       ;IF MOVING SPRITE
           BNE      JIRQDONE     ;SKIP

           LDA      XPOS         ;SAVE CURRENT POS
           STA      OXPOS
           LDA      YPOS
           STA      OYPOS
           LDA      JOY
           PHA
           AND      #1           ;CHECK UP
           BNE      CKDOWN
           JSR      UP
CKDOWN     PLA
           PHA
           AND      #2           ;DOWN?
           BNE      CKLEFT
           JSR      DOWN
CKLEFT     PLA
           PHA
           AND      #4           ;LEFT?
           BNE      CKRIGHT
           JSR      LEFT
CKRIGHT    PLA
           PHA
           AND      #8           ;RIGHT?
           BNE      CKDONE
           JSR      RIGHT
CKDONE     PLA
JIRQDONE   JMP      (OLDIRQ)     ;EXIT

UP         LDA      YPOS
           SEC
           SBC      YINCR
           STA      YPOS
           RTS

DOWN       LDA      YPOS
           CLC
           ADC      YINCR
           STA      YPOS
           RTS

LEFT       LDA      XPOS
           SEC
           SBC      XINCR
           STA      XPOS
           BCC      TOGLMSB
           RTS

RIGHT      LDA      XPOS
           CLC
           ADC      XINCR
           STA      XPOS
           BCS      TOGLMSB
           RTS

TOGLMSB    LDA      XMSB         ;TOGGLE X HIGH BIT
           EOR      #1
           STA      XMSB
           RTS

OLDIRQ     HEX 0000
OXPOS      HEX 00
OYPOS      HEX 00
```

# *An Internal Function RAM for the C128*

## *...RAM-that-shouldn't-be-there*

by Richard Curcio

Inside the C128 is an empty 28 pin socket to hold a 32K ROM or EPROM. This "internal function ROM" can contain an "application" program that is auto-starting, taking complete control of the system, a wedge to add features, or a collection of routines called with SYS or JSR. With minimal support circuitry, the socket can be made to hold a 32K static RAM (SRAM,) which can be written to "in-circuit", repeatedly, without the expense of an EPROM burner and eraser. With battery backup, the RAM will retain its contents even when the computer is turned off. One use for a battery-backed RAM is as a simple RAM disk, something not possible with an EPROM.

### Theory

The C128 selects internal function ROM when in the BANK 4-7 configurations. In these standard banks, ROM exists from $8000 to $ffff with a 4K gap at $d000 for I/O. The ROM overlays system RAM 0 for even-numbered banks 4 and 6, and RAM 1 for odd-numbered banks 5 and 7. All ROMs in the C128 are always selected, since their Chip Enable pins /CE are held at ground. When an access of ROM is needed, the PLA generates a logic 0 on the ROM's Output Enable pin, /OE, allowing the ROM to be read. In the case of internal function ROM, the enabling line is labeled "/FROM."

After opening my C128, I placed a logic probe on /OE of the empty socket and ran this line:

```
bank4: fori=0to255: poke32768+I, I: next
```

The logic probe indicated that /OE was indeed pulsing high and low, even though I was trying to write to non-existant ROM. (Sometimes, the computer locked-up. This will be explained shortly.) Examining RAM 0 with the MLM ".m" command, I found that locations $08000-$080ff contained the values $00-$ff. As expected, the memory writes were "bleeding" through to underlying RAM. So far, so good.

Next, I installed a 32K-by-8 bit static RAM in the socket with the pins /CS and /WE (Write Enable,) bent away from the body of the chip so that they weren't inserted in the socket. I then wired /CS to /FROM and /WE to system R/W and ran my one-liner again. Once past the occasional lock-up, I found that locations $48000-$480ff (BANK 4,) did not contain the expected values. The logic probe indicated that /FROM was taking place, but apparently, when the PLA detects a Write, /FROM is suppressed. It was there long enough for the logic probe to detect, but not long enough. (This is one of the pit-falls of attempting this sort of thing with un-sophisticated test equipment.) I had to somehow extend /FROM long enough to keep the SRAM enabled while R/W went low.

### Practice

Figure 1 shows the solution in a mix of mechanical and schematic representation. The 74LS75 is a quad "transparent" latch. When enable E is logic 1, the Q output follows the D input. Inverted data appears at /Q. When E goes low, the data on D at that time is latched at Q. Connecting D to /FROM and E to R/W, /CS will remain low if a write to banks 4-7 occurs. The signal used to latch /FROM is supplied by FR/W, a slightly different Read/Write inside the C128. This makes the SRAM inaccessible to an REU, but the circuit didn't work correctly using R/W.

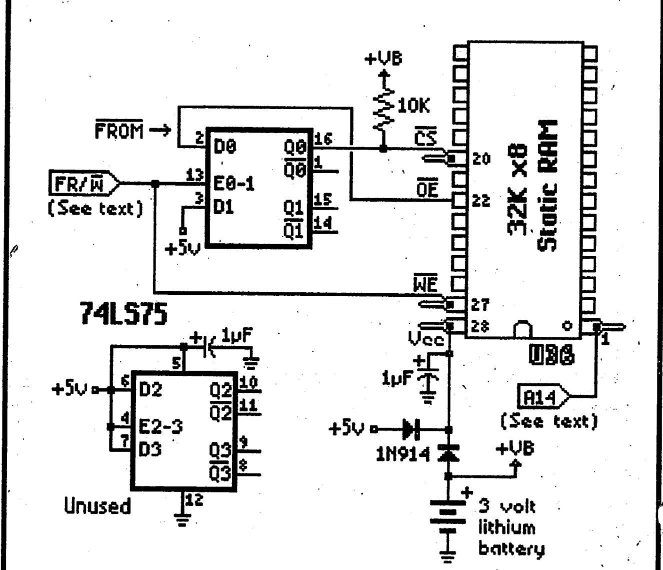


**Figure 1**

**Internal Function RAM**

Capacitors = 12 volts
Resistors = 1/4 Watt
Diodes = 1N914 or 1N4148

R. Curcio '90

Note that each E of the 'LS75 serves two latches. Note also that the 'LS75 is powered by the C128's 5 volt supply, the resistor on /CS is pulled-up by the battery voltage VB, and the SRAM is powered by the computer OR the battery. Notice also that +5 volts and ground for the 'LS75 are not on the usual "corner" pins 16 and 8. An equivalent chip with the more conventional power connections is the 74LS375. Indeed, as shown in figure 2, all the pins of the 'LS375 are arranged more sensibly. The 32K static RAM I used is a Sony CXK58256. Equivalent SRAMs are 62256 and 43256.

Note further that the bypass capacitors on both chips' supply pins should be more substantial than the usual 0.1 microFarads. I used 1 mF tantalum electrolytics.

### Installation

Unless otherwise noted, the following information applies to my computer, a fairly early C128. The only advice I can offer owners of later models, especially 128Ds, is to strongly recommend that you acquire a schematic and/or service manual for your particular 128 if it differs from the information supplied here. The schematic in the back of the C128 Programmer's Reference Guide is helpful, but it has a few errors.

Begin by disconnecting all peripherals and the power supply from your computer. Remove six screws from the bottom of the case, noting the different sizes. Partially lift the top cover and remove the three wire LED power indicator on the left side. (When you reconnect it, the polarity won't matter.) At this point, you might want to discharge any static charges from yourself by touching a cold water or radiator pipe. Shift the top cover to the left and unscrew the large ground strap connecting the keyboard to the main board. Carefully remove the keyboard connector on the right side of the board. Do this by gently rocking it from side to side while maintaining an upward pull. This connector is "keyed" and will go back on only one way. If any of the pins bend, you can gently straighten them. If you break any pins, you will lose use of your keyboard. Remove the screws holding the RF shield in place, again noting any difference in sizes. The metal shield is soldered to the board at one point. This will have to be unsoldered, using a solder vacuum and/or solder braid. There are also small metal tabs holding the top and bottom metal covers together. Gently un-bend these and lift the top cover to expose your computer's innards. The white gooey stuff on the larger chips and the underside of the RF shield is heat-sink compound. Be careful not to get any on clothing, carpets, etc. You don't have to remove the main board from the lower case.

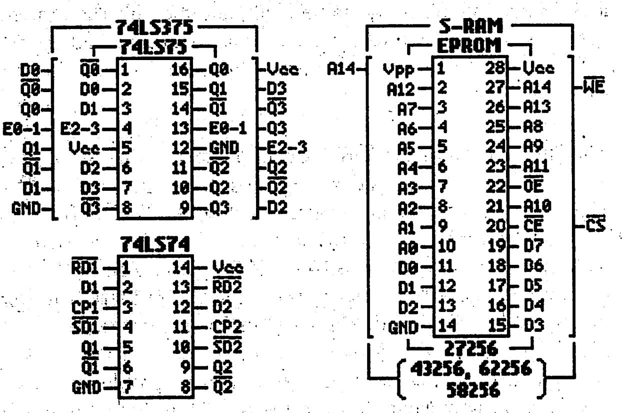


**Figure 2** pin configurations

R.C. '90

With small pliers, bend pins 1, 20, 27 and 28 away from the body of the static RAM you'll be installing. Don't try to make them completely horizontal, just enough to clear the socket. Make sure they don't touch any pins of the adjacent ICs. You can find ground and +5 at pins 1 and 2 respectively of the User Port. The /FROM signal is available at pin 22 of the socket the static RAM is installed in. When soldering to socketed IC pins, solder to the "shoulder" -- the wide part of the pin. Don't use globs of solder. This can cause the pin to become soldered to the socket, making the chip difficult to remove. FR/W is available at pin 1 of U57, a 7407 inside the video "box", or pin 6 of U8, a 74LS08 some distance away. (The P.R.G. schematic erroneously shows this part of U57 as an inverter. It also shows FR/W at pin 9 of the 80 column VDC. My 128 did not jibe with this. An inexpensive multi-meter is invaluable when comparing schematics against the actual equipment.) If you go inside the video box, insulate a portion of the edge with a small piece of electrical tape and position your FR/W wire so it doesn't get crushed when the cover is replaced.

In the 128D FR/W (labeled R/WA in the SAMs schematic,) is at pin 3 of U61, or pin 9 of U57. Because the pin-out of the SRAM is slightly different from that of a ROM or EPROM, pin 1 must be bent out and connected to address bit A14. That signal is available at pin 27 of the socket but, being in such close proximity to the bent pin 27 of the SRAM, I felt that connection hazardous. In 128s that use four 16K ROMs, A14 is not available at the other ROM sockets. I found A14 on my C128 at the feed-through immediately to the right of the "2" of the identifier "R32" (near U32 and U33). On the "D", A14 is available at pin 22 of the 8502 (U6), pin 4 of the Z80 (U10), or pin 2 of the 8721 PLA (U11).

Since the circuit uses so few components, I wired the 'LS75 on a low-profile 16-pin dip socket without a circuit board. When I was satisfied that the thing was working, I enfolded the boardless assembly in a piece of paper, secured with adhesive tape, and let it sit in the empty space near the User Port. There is enough room near the User Port for a very small perforated board. (I have other plans for the larger open area.) Insulate that area of the main board with two or three layers of masking or electrical tape. You can bend the rear lip in that vicinity of the RF shield if it hits anything when replaced. (You can also get creative with tin snips, if necessary.) Use the top of the rectangular video box to judge how high you can go before hitting the shield. Use the top of the square RF Modulator to judge where the upper plastic housing begins. There is also some usable space outside the RF shield, along the left edge of the case. Be sure to insulate your 'LS75 assembly in some manner. Another strategy is lay down some doubled-stick foam tape, stick the 'LS75 to it, on its back, and wire it upside-down.

The battery used is a 3 volt lithium "coin" type, with tabs for vertical p.c. board mounting, Panasonic # BR2330-1VC. Wires were soldered to the tabs and electrical tape wrapped around to insulate it. I let it sit just behind the video box.

All this trickery was needed because of the "tightness" of the 128 housing. "D" owners should have no trouble finding room closer to the empty socket.

Check your wiring and inspect your work for solder "bridges" and other errors. When you're confident, reconnect the keyboard, power supply and monitor. Don't replace the screws just yet. Turn on your monitor first so you won't have to wait for it to warm up. Power up the C128 and you should get the familar start-up screen. If so, you can go on to the next step. If not, power down everything and inspect your work again.

## Testing/Software

Enter lines 100-130 of Program 1. To handle interrupts, on start-up the C128 Kernal copies routines into the last page of RAM 0 and 1. If an IRQ or NMI should occur while the Kernal is switched out, these routines save the current configuration, along with .a, .x, .y and the status register before switching to the BANK 15 "system" configuration. Without these routines, the computer would get lost. The absence of these routines was the cause of the lock-ups while I POKEd BANK 4 in my preliminary experiments. Save this short program and run it. Do not attempt to access your IFR (Internal Function RAM,) from Basic before running these lines or their ML equivalent. These lines turn off interrupts and copy the routines and vectors from the last page of RAM 0 into the same locations in BANK 4. (You can also use the MLM Transfer command; .t 0ff05 0ffff 4ff05. Ignore the "?" that appears upon completion.) Enter the MLM and enter .d 0ff05. You should see SEI, PHA, etc. Cursor up to the .d command and change the address to 4ff05. The disassembly should be identical. If not, your new circuitry still isn't right. If page $ff of Bank 4 matches Bank 0, turn your 128 off, wait a few seconds, then turn it back on. Examine $4ff05 again with the MLM and the IRQ and NMI handlers should still be there, demonstrating that the battery backup is functioning.

*continued on page 18*

continued from page 17

Load the fragment of program 1 that you saved earlier and type in the remaining lines of the program. When run, it pokes a machine language program into the applications area beginning at 4864. This program will allow you to move data to and from your new RAM-that-shouldn't-be-there and is relocateable by changing the variable SA in line 180. Assuming that SA is the Start Address of the routine, it is accessed with:

### bank15: sys sa, host bank, direction,,, host start, host end, ifr start

The host bank is system memory in the standard BANK configurations; 0-3, 14 and 15. Banks 4-13 are not allowed, for reasons that will be explained. This restriction can be bypassed. Direction is zero to move data TO internal function RAM ("save"), and greater than zero to recall data FROM that RAM ("load"). Host start and end are self explanatory, while IFR start must be 32768 or greater. The three commas must be present. Addresses greater than 65279 are not allowed, as that would affect the MMU registers at $ff00-$ff04 and the important routines and vectors in page $ff. As the routine moves data, it checks its pointers and will set the Carry bit if either the source or destination reaches page $ff. Carry is also set if either pointer "rolls over" to zero page -- which theoretically can't happen, but one never knows. Use the RREG function to test the Carry from Basic.

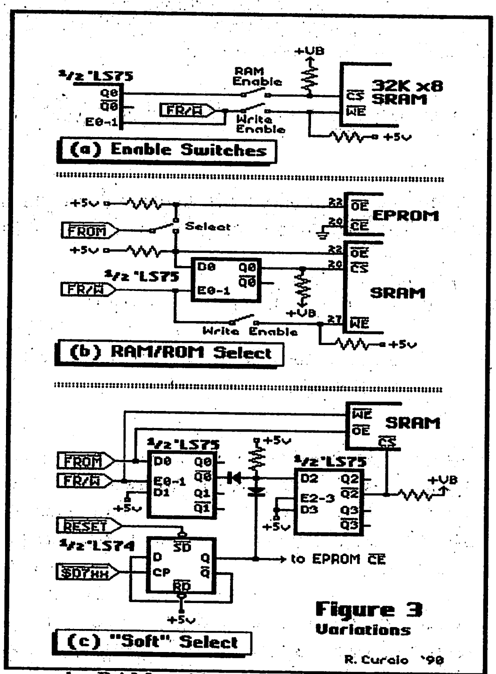


Figure 3
Variations

R. Curcio '90

The routine uses the RAM-resident portions of the Kernal INDSTA and INDFET routines. Calling these routines via the Kernal jump table was far too time consuming, because the bank number is converted to a configuration value for each byte. Instead, the routine performs the bank to configuration conversion only once, and bypasses the jump table.

No tricks are needed to recall data from the IFR, but because of the "bleed through" to underlying system RAM when writing, two loads and stores are performed for each byte moved to the IFR. (So it's just as well that the SRAM cannot be accessed by an REU.) The byte "under" the IFR destination is read and stored on the stack. Then the host source byte is stored in the IFR. Then the byte on the stack is restored to the location under the IFR. In this way, we do not lose the use of the underlying system RAM. (Do any light bulbs go on over the heads of owners of a certain battery-backed RAM cartridge?) This is one reason why BANKs 4-13 are not allowed as "host." If we tried to use the "load" portion of the routine to move data from an IFR location (or an external ROM,) to an IFR location, the byte under the destination would be lost. The "save" operation would work properly, but this is a "dumb" mover, so if the source and destination are in the same bank, and they overlap, the move becomes a fill. If you really need to move data around inside the IFR, and don't care about the under-bytes, call the routine by sys sa+16.

The "to" portion of the IFR Mover contains a rare use of the MMU Pre- and Load Configuration Registers. It assumes that PCRA at $d501 contains its default value, selecting the BANK 0 configuration when any Write is directed to LCRA at $ff01.
You may freely PEEK or BSAVE IFR locations from Basic (assuming page $ff is intact,) except for $d000-$dfff. The standard banks 4-7 include I/O in this range and you can't get at it with the MLM either. The IFR Mover uses a non-standard BANK 4 configuration, and accesses this area with no problems. This is the main reason why I couldn't use the Kernal entry points for INSTA and INDFET. You can also POKE and BLOAD to BANK 4 (excluding d-block!) if you don't care about the under-bytes.

Since the IFR Mover contains no JSRs or JMPs within itself, it can be relocated without adjustments. (This made the routine longer, but faster. Early versions took as long as eight seconds to save 32K!) It doesn't disable IRQs, so it will not disrupt split-screens or other interrupt driven activities. This is another reason why page $ff must remain inviolate.

Once the routine is ensconced in system RAM, it can be used to move itself into internal function RAM. It can't execute from there, (at least, not in its present form,) but, thanks to the battery, it will remain there when you turn your machine off and back on. Being only 198 bytes in length, it won't take very long to move it from the IFR to system RAM using Basic:

bank4: forl=0to197: poke5120+l, peek(34000+l): next

assuming the routine is stored at 34000 in the IFR, it's being moved to 5120, and page $ff of bank 4 contains a copy of that page of RAM 0.

## Variations

In figure 3, some changes to the circuitry are suggested. Figures 3a and 3b are fairly straight forward but 3c may require some experimentation. I stress that these are un-tested ideas. In 3a, two switches and another pullup resistor are added. Opening the switch on /CS disables the RAM, perhaps to prevent an auto-starting program from doing so. (There are ways to do this in software. C128 auto-starting is too involved to go into here.) When open, the switch on /WE provides a "Write Protect", preventing changes to the RAM's contents. This provides ROM emulation and could serve as a system for developing auto-starting ROMs. The switches could be positioned on the left side of the computer case. I can understand why anyone would be reluctant to cut holes in their computer.

What if you already have a ROM or EPROM in that empty socket? Figure 3b shows how you can have your SRAM and EPROM too. A 2-position switch sends the enabling /FROM signal to the EPROM or the 'LS75 latch. The EPROM would be inserted into the socket with its pin 22 bent out. The SRAM would be piggy-backed to it with its pin 22 bent out, in addition to the pins shown in figure 1. The /FROM signal is obtained at pin 33 of the 8721 PLA, U11. When the RF shield is replaced, the heat sink tab should be straightened. Insulate the underside in that area with electrical tape. You still have to find a place for the switch(es).

Figure 3c shows a software RAM/ROM Select. The latched /FROM is inverted at /Q0 and one of the unused 'LS75 latches is used as an inverter by tying E2-3 to +5 volts and using D2 and /Q2 as input and output. The diodes at D2 turn this inverter into a NAND gate. The 'LS74 is an edge-triggered D flip-flop. With its /Q fed around to D, each low-to-high transition on CP toggles the flip-flop from 0 to 1 or from 1 to 0. The CP clock pulse is supplied by /$d7xx, a page of I/O refered to in Commodore literature as "reserved." It is found at pin 12 of I/O decoder U3, a 74LS138. With I/O visible, any access of $d7xx, Read or Write, will toggle the 'LS74. With Set input /SD connected to the C128 RESET line, the flip-flop powers up with Q high, disabling the EPROM and allowing /FROM to select the SRAM. If RESET were instead connected to the Reset input /RD, the opposite condition would be established on power-up. This circuit should just barely fit in the aforementioned User Port vicinity.

Once you start toggling the 'LS74, it's easy to lose track of the state it's in, since it can't be read. One strategy is to select BANK4 and attempt to read a sequence of values known to exist in EPROM. If that sequence is not found, the flip-flop is in the other state. As mentioned earlier, this circuit may require some experimentation, particularly around the 'LS74. No doubt, more elaborate schemes can be devised, perhaps extending selection to more than one SRAM or EPROM. The major challenge would be getting things to fit -- in the 128, that is. The 128D has plenty of room inside.

There is an alternative to piggy-backing available to C128 owners who have upgraded their system ROMs. Besides correcting some firm-ware bugs, that upgrade replaces four 16K ROMs with two 32K ROMs, thus freeing two more 28 pin sockets, U33 and U35. Keep the function ROM in the socket assigned to it, U36, with pin 22 bent out of the socket for figure 3b, or pin 20 for figure 3c. The SRAM can then be inserted into either U33 or U35. If U35 is used, pin 2 must be bent out in addition to the other pins. That socket uses Translated Address bit TA12, which complicates matters! Wire the bent pin 2 of the SRAM (A12,) to pin 2 of U33, U34, or U36, but not U32. That socket also uses TA12.

## Other Possiblities

Alas, this new 32K of RAM isn't accessible in 64 mode. (I do have a plan, but it takes too many chips. And getting it to coexist with external cartridges could be tricky.) But it's possible that on 128 mode start-up, an auto-starting program could check for a certain keypress (possibly CONTROL or ALT,) and if found, copy part of itself to $8000 in RAM 0, complete with the C64 auto-start sequence, GO64, and come up running. For now, additional ideas must come from you.

continued from page 8

## The Dialogue Continues

This June, Voyager Mindtools Inc. took over the exclusive distribution of the powerful menu-driven terminal package, Gary Farmaner's Dialogue 128. Those who have read our praise of this product in past issues can easily understand how proud we are to be associated with Dialogue.

TC128 #27 first reviewed Dialogue 128 V2.0 over a year ago, and called it "a leading contender for the finest piece of code ever written for the Commodore 128". Dialogue V2.2b continues to feature a rare combination of sophistication and ease of use that makes it a great example of C128 software that meets or beats products running on other computers. It compares well feature for feature with other "power user" terminal programs, offering a wide selection of transfer protocols, sophisticated macros, multiple buffer options, a built-in editor, and full support for the 1351 mouse, the SwiftLink RS232 interface and Commodore's RAM expansion units. While Dialogue 128 can call up all its features using a conventional mnemonic key command method, the less expert among us will be able to use all this power by dropping down the menus from the top of the screen with a touch of the HELP key or a click of the mouse. Even the process of setting up the initial configuration of the Dialogue program (choosing your modem type, mono or color screen, and other default settings) is handled by a menu-driven program."

Dialogue 128 lets the Commodore 128 telecommunicator use the modem features, hardware, and online options they may find missing in older products:

o Supports SwiftLink RS232/UART interface for higher effective baud rates on the C128

o Supports Commodore 1650, Commodore 1660, and Commodore 1670 (3 or 4 dip models), Pocket Modem 300/1200, Total Telecommunications 64, MNP 1064, Sharedata, Westridge, Master Modem, Mitey Mo, Hes-II, Volks 6420, Volks 6480, Aprotek 12c, Volksmodem 12, Aprotek 1200c, Aprotek Minimodem, and any Hayes command compatible modem with appropriate RS232 interface including support for those modems with MNP 5 error correction/data compression

o Modern drop down menu and requestor system makes Dialogue easy to learn and use - menu access is via mouse, joystick, trackball, or keyboard

o Alternate quick command interface using ALT, CBM, and CONTROL key combinations

o Installs on any disk drive, utilizes RAMDOS RAM disks, multiple disk drives, and can issue direct disk drive commands from within the program

o Flexible buffer configuration options (8 options for non-REU owners, even more for owners of 1750 or compatible REU's, including a massive 512K buffer!)

continued on page 20

continued from page 19

- Integrated text editor

- Terminal emulations: CBM graphics, VT52 and VT100, or ANSI graphics.

- Batch transfer protocols, including Punter and YModem as well as XModem, XModem 1K, and Compuserve QuickB

- Full-featured BASIC-like macro language that lets you automate as much or as little of your telecommunications as you'd like

- Multiple phone books with each entry separately configurable for default terminal emulation, transfer protocol, password macros, etc.

- Extensive control of auto-redial activity

- 10K scroll-back buffer recalls your most recent online activity

- Chat mode

- Optional 50 line and 40 column modes without switching from the standard C128 80 column display

- Support for burst mode drive activity on 1571 and 1581 disk drives and always uses the C128 2 MHz Fast mode

The new list price for Dialogue 128 V2.2b is $29.95. Dialogue requires a C128 with disk drive and 80 column monitor (color or monochrome) and a compatible modem. It supports (if present) a 1351 or equivalent mouse, standard joystick or trackball; 64K VDC RAM, Commodore RAM expansion units, a printer and the CMD hard drives or RAMLink. A demo disk which includes a working subset of the full Dialogue program is available for $2.50. Upgrades from previous versions are $15 with the return of the original disk. Those with version 2.2 who want to upgrade to V2.2b can either download a freely distributable patch program from various online sources or purchase the Dialogue demo disk which includes the patch. Address all orders to the Voyager Mindtools P.O. Box 11578, Minneapolis MN 55411.

## Zippity-do-dah

The drive to produce a speed up chip for the C128 has produced some returns at last. Parsec Inc. has reached an agreement with Zip Technology to develop an 8 to 10 MHz speedup board for the the 128, provided that certain conditions are met. And Parsec has begun to implement a program to meet those conditions. Pursuant to this aim, "order forms" for the Zip chip are being circulated via online services this June. Essentially these forms are promises to send in a $50 deposit at a later date. In capitol letters the forms remind users "DO NOT SEND ANY MONEY NOW!!!!!"

Parsec promises that this proposed C128 Zip chip will be sold "For $200 or less" and "That it will be software compatible with the majority of software and hardware including CMD products". They add that "Z80 - CP/M mode is not included in the above promise at this time".

The proposed design is based around a daughterbord that includes not only a faster 6502 compatible central processor but circuitry that lets this chip "shift-down" in speed when it needs to talk to the other chips used in the C128. Similar methods have proven workable with Zip Technologies Apple II product (see a review of their products in the June 1991 issue of the Apple II magazine inCider - go ahead, we give you special TC128 dispensation to buy an Apple II mag just this once!). Furthermore, similar ideas are utilized in the architecture of the Apple IIgs and some of the speed up chips for the Amiga. It's an ambitious project, but one that if completed should produce noticeable performance gains.

Those interested in downloading the "order form" for their own use or redistribution to local BBSs, gaining more information on the project, or talking directly with developer John Brown can download the C128 Zip Chip files available on Qlink in the Hardware Support Area of the Commodore Information Network (to get there, just follow the menus) and the GEnie Commodore area (m625 at any prompt), or leave EMail to:

Qlink = Parsec<br>
Genie = C128.JBEE

or write:

Parsec, Inc.
P.O.Box 111
Salem, MA 01970-0111

# CMD • Your Memory Location • CMD

# HD Series™ Hard Drives

## The New Standard in 64/128 Hard Drives

The CMD HD series of hard drives for the Commodore 64 and 128 have been designed to provide the highest level of compatibility and performance. These drives incorporate the latest in SCSI technology with advanced circuit design to bring you the most features at an affordable price. CMD drives allow you to achieve the higher level of productivity you require to get the most from your existing software - today!

- **Compatible with GEOS™, CP/M™, BBS programs and most commercial software.**
- **3 1/2" SCSI technology for quiet reliable operation in a compact case about the size of a 1581.**
- **Built-in Commodore compatible DOS responds to all 1541, 1571 and 1581 commands.**
- **Up to 254 partitions in sizes ranging from 256 blocks to 65,280 blocks each.**
- **Built-in real time clock automatically time and date stamps all files.**
- **Compatible with Amiga, IBM and Macintosh systems for convenient upgrading.**
- **Connects easily to the serial bus, leaving ports open for use with cartridges, REU's and RAMLink.**
- **Supports all serial and JiffyDOS protocols, plus parallel interface for connection to RAMLink.**
- **Easy-to-use utilities allow backing up and copying files to and from 1541, 1571, or 1581 drives.**
- **Supports an unlimited number of true subdirectories which share all blocks within a partition.**
- **Built-in Q-Link software allows easy access to America's most popular 64/128 online service.**
- **External power supply avoids overheating and wear on computer power supply.**
- **Includes standard SCSI port and all utilities needed to expand the system (up to 4 Gigabytes).**

**HD-20 $599.95 • HD-40 $799.95 • HD-100 $1149.95 • HD-200 $1499.95**

### CMD SD Seiries Add-on Drives

**SD-20 $459.95 • SD-40 $559.95 • SD-100 $899.95 • SD-200 $1299.95**

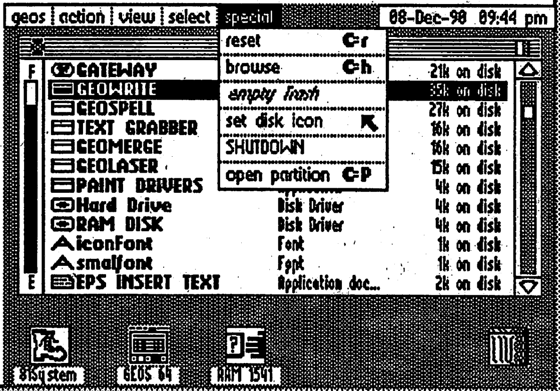
# gateWay™

## A Whole New Way of Working with GEOS

More than a replacement for the deskTop, the gateWay offers a unique, streamlined approach to file management under GEOS. With a resizable Note Pad, proportional slider and fuel gauge, the gateWay makes it easier to get where you're going. In designing the gateWay, we took a good hard look at the deskTop to make sure that none of the functionality would be lost. Then we added feature after feature - capabilities that you, the serious GEOS user, have found lacking in the deskTop. At CMD, there's a lot more going on than powerful new hardware development - we're creating new ways to use it.

- **Full three drive support, including auto-swapping for applications which will not allow use of drive C.**
- **Loadable disk drivers become memory resident and replaces the need for CONFIGURE.**
- **Full CMD device support for RAMLink, RAMDrive and HD hard drives allowing for partition sizes up to 16 Megabytes and Macintosh style folders.**
- **Supports using the CMD HD hard drive via parallel for faster hard drive access.**
- **RAM disk drivers allow use of as much available RAM as possible - no longer limiting you to a 1541 or 1571 sized RAM disk.**
- **Integrated Control Panel replaces the Preferences Manager and allows you to contour the gateWay to your choice of colors, screen pattern, printer driver and input driver.**
- **Task switching via our Switcher allows you to move instantly between two applications.**
- **An all new trash can structure which allows you to retrieve multiple files from the trash can.**
- **A proportional gadget for moving easily through the directory listing on the Note Pad.**
- **Status boxes to indicate files remaining for copy and erase features, plus new file and disk info boxes.**
- **A browse feature which helps in quickly locating files on the currently active device.**

Note: The gateWay requires a Commodore 64 or 128, minimum of one disk drive, a mouse or joystick and GEOS 2.0 for operation. Available first quarter of 1991. Orders are not yet being accepted. Projected retail prices for the gateWay are $29.95 for gateWay 64, $29.95 for gateWay 128 and $39.95 for both versions.



```
GATEWAY                              21k on disk
GEOWRITE                             35k on disk
GEOSPELL                             27k on disk
TEXT GRABBER                         16k on disk
GEOMERGE                             16k on disk
GEOLASER                             15k on disk
PAINT DRIVERS                         4k on disk
Hard Drive        Disk Driver         4k on disk
RAM DISK          Disk Driver         4k on disk
iconFont          Font                1k on disk
smalfont          Font                1k on disk
EPS INSERT TEXT   Application doc...  2k on disk
```

# JiffyDOS™ V 6.0

## The Ultimate Disk Drive Enhancement System

**JiffyDOS 64 or SX-64 - $59.95   JiffyDOS 128 or 128D - $69.96**

Each system includes computer Kernal ROM(s) and one drive ROM.
Please specify computer and drive model numbers and serial numbers when ordering.
**Additional Drive ROMs - $29.95 each**

## OTHER CMD HARDWARE & ACCESSORIES

| | |
|---|---|
| Lt. Konnector - Allows use of Lt. Kernal drive mechanism with CMD HD System | $19.95 |
| Disk Drive Power Supply - Heavy Duty Replacement for 1541-II, 1581, & CMD HD | $49.95 |
| JiffyMON ML Monitor - 64 Mode ML and disk drive Monitor (requires JiffyDOS) | $19.95 |
| Serial Cables- Quality 36" Commodore Serial Cables (6 pin DIN male to male) | $3.95 |

**Shipping (U.S.):** Lt. Konnector $5.00, Power Supply $6.00, JiffyMON $2.00, Serial Cables $2.00

## • SUPER SAVINGS COUPON •

## Any JiffyDOS System ONLY $15.00*

***with the purchase of any HD hard drive**

To receive these low prices (US & Canada), send in this coupon with your order. For ordering, shipping and credit card details see the adjoining box. Offer expires 2/31/91. This coupon may not be used with any other offer.

## Ordering Information and Shipping Charges

| | |
|---|---|
| HD and SD Hard Drives: | Continental US: $25.00 per drive (UPS ground), $35.00 (2nd-Day), $45.00 (Next-Day). Canada: $50.00 (Airmail). COD to U.S. only $4.00 add'l charge. Foreign prices: 20Mb, $699.95, 40Mb, $899.95, 100Mb, $1249.95, 200Mb, $1599.95 Postage: $35.00 |
| JiffyDOS: | Add $5.00 per order (UPS ground), $9.00 (2nd-Day Air), plus $4.50 for APO, FPO, AK, HI, and Canada, or $15.00 for overseas orders. No additional shipping if ordered with any hard drive.COD's to US only - add'l $4.00 |
| Tax: | MA residents add 5% sales tax. |
| Terms: | We accept VISA, MasterCard, Money Orders, C.O.D., and personal checks (allow 3 weeks for personal checks to clear). Credit card orders provide the following: Card holders name, billing address, home/work phone, card number, expiration date and issuing bank. |

**• WE VERIFY ALL CREDIT CARD INFORMATION AND PROSECUTE INDIVIDUALS ATTEMPTING TO PERPETRATE FRAUD •**

Note: Prices and specifications subject to change without notice. GEOS, GEOS 2.0, GEOS deskTop, GEOWRITE, GEOPAINT, GEOS CONFIGURE, and GEORAM are registered trademarks of Berkeley Softworks, Inc, CPM is a trademark of Digital Research, Commodore 64, Commodore 128, Commodore SX-64, Commodore 128D, Commodore 1700, 1764, 1750 and Commodore RAMDOS are registered trademarks of Commodore International, Amiga is a trademark of Commodore-Amiga. Macintosh is a trademark of Apple Computer, IBM is a trademark of International Business Machines.Lt. Kernal is a trademark of Fiscal Incormation, Inc. RAMLink, RAMDrive, CMD HD Series and gateWay are registered trademarks of Creative Micro Designs, Inc.

## CMD / Dr. Evil Labs *SWIFT*Link-232 Serial Cartridge

### A Real RS-232 Serial Port for the Commodore 64 or 128

SWIFTLink-232 gives you a true RS-232 port with the ability to communicate at speeds from 300 to 38,400 bps, plus true 1200 and 2400 bps using Hayes compatible modems. Experience the power of high speed modems and fast data transfers directly to other computers. SWIFTLink plugs easily into your computer's expansion (cartridge port) and provides an industry standard IBM-AT style serial port. As an added bonus, SWIFTLink comes complete with file transfer software (FTP) plus many popular shareware terminal programs such as Desterm 128 and NovaTerm for the C-64. SWIFTLink is compatible with most other hardwre devices designed for the Commodore 64 and 128. Note: SWIFTLink serial Cable (DB-9 to DB-25) is required for use with most modems.

| | | | |
|---|---|---|---|
| SWIFTLink-232 Cartridge | $39.95 | SWIFTLink Cable | $9.95 |

**Shipping:** US: **$5.00** (Cartridge), **$3.00** (Cable), **$6.00** (Both). Canada add **$4.50.**

## CMD / Dr. Evil Labs SID SYMPHONY Stereo Cartridge

### The Realism of Stereo Sound on your Commodore 64 or 128

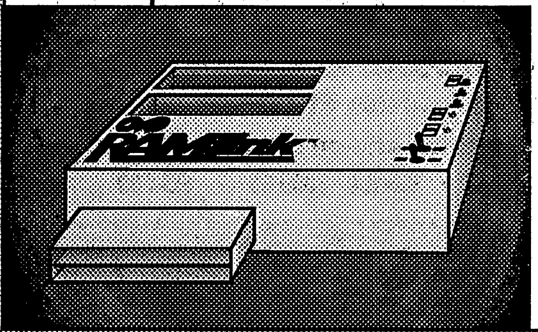
The STEREO SID Cartridge effectively adds a second complete SID chip to your Commodore 64 or 128, giving you a total of 6 separately controllable voices. SID SYMPHONY may be attached to your home stereo or most amplified speaker systems to provide you with whole new dimension in sound. Create your own stereo music using the Robert Stoerrle's Stereo Editor (which we include free with *Compute!'s Enhanced SIDPlayer* book), or just enjoy the many hundreds of available songs by using Mark A. Dickinson's Stero Player software (included free with the SID SYMPHONY Cartridge). If you're interested in creating your own music files, be sure to get a copy of *Compute's Music System for the Commodore 128 and 64: The Enhanced SIDPlayer.*

| | | | |
|---|---|---|---|
| SID SYMPHONY Cartridge | $39.95 | Compute's Music System Book | $22.95 |

**Shipping:** US: **$5.00** (Cartridge), **$3.50** (Book), **$7.00** (Both). Canada add **$4.50.**

# RAMLink™

## Power Backed REU Interface and Expandable RAM Disk



RAMLink is a multi-purpose hardware interface designed to overcome the limitations of existing RAM expansion units (REUs). RAMLink delivers the performance and compatibility lacking in past REUs by incorporating CMD's proprietary RL-DOS and advanced features into a compact unit which allows most software to utilize Commodore 1700 series REUs, GEORAM, PPI's RAMDrive, and CMD's RAMCard as if it were a disk drive. Optional RAMCard installs internally and can be used alone or in combination with the storage of an external REU for a maximum capacity of 16 Megabytes.

- **Supports Commodore 1700, 1764, 1750, Berkeley Softworks GEORAM and PPI's RAMDrive.**
- **Optional RAMCard allows RAMLink to be used as a RAM Disk with or without a separate REU. User Expandable from 1 Mb to 16 Mb using standard SIMMs.**
- **Pass-Thru connector allows use of cartridge port peripherals such as utility cartridges.**
- **Reset, Disable, Direct Access mode and SWAP functions are all standard features.**
- **RAM port provides power back-up to REUs.**
- **Parallel port provides ultra-fast data transfer when connected to CMD HD series hard drives.**
- **Includes separate power supply - optional battery available to protect against power failure.**

| | | | |
|---|---|---|---|
| RAMLink (w/out internal RAM) | **$149.95** | RAMLink Battery back-up unit | **$24.95** |
| RAMLink w/ RAMCard (1 Mb) | ***$259.95** | Parallel Cable for CMD HD | **$14.95** |
| RAMCard w/ 1 Mb (if purchased separately) | ***$149.95** | | |

*Add $90.00 per Mb for additional RAM up to 4Mb. Higher capacities - CALL. Prices and specifications subject to change without notice.

**Shipping:** US: **$10.50** (UPS), **$18.00** (2nd day), COD add **$4.00.** Canada **$20.00.**

## Both RAMLink and RAMDrive Include RL-DOS

- Allows commercial software to access all available RAM as a high-performance RAM Disk.
- gateWay software for GEOS use available at no additional charge.
- RL DOS performs up to 20 times faster than Commodore RAMDOS, up to 400 times faster than a stock 1541 and provides full DOS command compatibility.
- Up to 31 1541/1571/1581 emulation or Native mode partitions, plus true subdirectories.
- Includes built-in JiffyDOS Kernal for high-speed access to JiffyDOS equipped disk drives.
- Software for copying files (FCOPY) and complete disks (MCOPY) included.
- Many other features, including CMD's exclusive device number SWAP functions.

## RAMLink or RAMDrive - Which one is right for you?

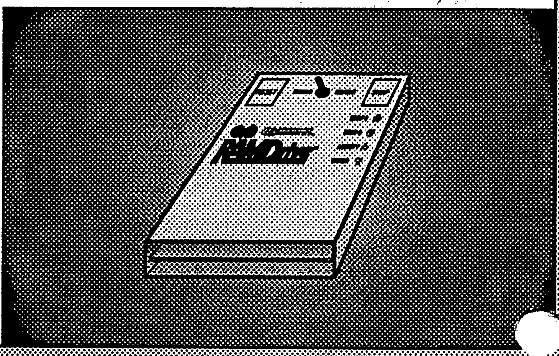
Both RAMLink and RAMDrive give Commodore users powerful features never before available in a RAM expander. RAMDrive is intended mainly for those who do not require the high RAM capacity of RAMLink ( up to 16 Mb ), RAMLink's Pass-Thru port or the parallel port for communicating with the CMD HD Series of hard drives. RAMDrive is an excellent choice for those who desire portability since the internal battery pack can keep the contents of RAMDrive intact for several days. RAMLink is the perfect choice for GEORAM owners who want the ability to use that device with programs other than GEOS. RAMLink has a built-in RAM Port for easy use with a GEORAM or Commodore REU. No matter which CMD RAM device you select, you'll benefit from the fantastic capabilities of a high speed RAM disk equipped with RL-DOS and the outstanding value of CMD support.

## RAMDrive - A High Performance Battery Backed RAM Cartridge

# RAMDrive™

RAMDrive is a fresh new approach to RAM expansion for the Commodore 64 and 128. RAMDrive is a self contained battery backed cartridge with capacities of 512K, 1 Mb and 2 Mb. RAMDrive incorporates RL-DOS, an ultra-fast, easy-to-use operating system similar to the HD-DOS found in CMD HD Series hard drives. This DOS allows various types of partitioning along with compatibility and speed unsurpassed by previous RAM Expansion systems.

- **Compatible with vast amounts of commercial software as an ultra-fast RAM disk.**
- **Power back-up eliminates the loss of files upon powering down the computer while the built-in battery protects data during transportation or power outages.**
- **Available in 512K, 1 Mb and 2 Mb models at incredibly low prices.**
- **DOS includes JiffyDOS Kernal routines for high speed disk access with JiffyDOS equipped disk drives.**
- **Software for copying and GEOS compatibility included.**
- **Reset switch allows computer reset without loosing data in RAMDrive.**
- **Reset, Disable and SWAP functions are all standard features.**



| | | | | | |
|---|---|---|---|---|---|
| **RAMDrive w/512K** | **$149.95** | **RAMDrive w/1 Mb** | **$199.95** | **RAMDrive w/2 Mb** | **$289.95** |

**Shipping:** US: **$7.50** (UPS), **$14.00** (2nd day), COD add **$4.00**, Canada **$15.00**. See Ordering and Shipping information elsewhere in this ad for credit card information. RAMDrive designed and manufactured by Performance Peripherals, Inc. Prices and specifications subject to change without notice.

CMD

## Creative Micro Designs, Inc.

50 Industrial Dr., P.O. Box 646, East Longmeadow, MA 01028
**ORDERS ONLY:** 800-638-3263 **BBS:** 413-525-0148
**Questions/Support:** 413-525-0023 **FAX:** 413-525-0147

CMD

## The Final Horizon
## Commodore 64 & 128
## Newsletter

This monthly newsletter contains articles about the C64 and C128 computers. It features a disk with 64 or 128 public domain programs every other month. You also have the option to receive both disks. To subscribe, fill out or circle the information below and send your remittance to THE FINAL HORIZON. We publish on the fifth of the month.

Annual Membership

| USA | Canada |
|---|---|
| $19.00 per year for either disk | $25.00 per year for either disk |
| $25.00 for both 64 and 128 disk | $33.00 per year for both 64 and 128 disk |

Choose the disk you want C-64 C-128 Both

First name:

Last name:

Address:

City:

State:

Zipcode:

Computer: C-64 C-64c C-128 C-128D

Disk Drives: C-1541 C-1571 C-1581

Ram Expanders: C-1700 C-1764 C-1750

Modems: 300 1200 2400

Monitor type:

Printer:

For a FREE sample issue drop us a note with your address and we will send one.

* PAID FOR ADVERTISEMENT * PAID FOR ADVERTISEMENT * PAID FOR ADVERTISEMENT * PAID FOR ADVERTISEMENT *

## Twin Cities 128 - The Commodore Specific 128 Computer Journal

Twin Cities 128 is written and proudly produced on C128s by C128 owners and users. Formly owned and produced by Loren Lovhaug of VMI it is now owned and produced by Parsec, Inc. Doctors and lawyers specialize in their fields because doing so makes them experts, we specialize in the C128 !

Info magazine once wrote "If you are serious about using or programming your C-128, you need this publication." and in the July 1990 issue of INFO they added Twin Cities 128 "has always been the best source of C-128 information and editor Loren Lovhaug is dedicated to keeping TC128 up and running for a long, long, time."

The editorial staff and owners have changed (though they are still staying on as authors, writers, and reviewers) but Twin Cities 128 is still your best and only place to get those articles and programs that let you utilize your C-128 to it's fullest and realize the maximum return on your computing dollar. Starting in issue number thirty-one Parsec is adding a "C" language, CP/M, Geos, and a question, answer, and comment column and much more. We will also provide type in program listings for these columns and offer a TC128 disk that can be ordered with a subscription or from our catalog.

| | | | |
|---|---|---|---|
| Subscription information: 6 issues a year | $20.00 for the U.S. | $30.00 for Canada | $40.00 Overseas |
| Subscription with a disk: 6 issues a year | $36.50 for the U.S. | $46.50 for Canada | $56.50 Overseas |

Single TC128 disks starting with issue #31 are available from our catalog for $3.00

---

## Twin Cities 128 - The Commodore 128 Specific Computer Journal
## User Group Special

You must have three or more subscriptions to start the special, each additional subsciption can be added at the below rates. Please provide the full name and address of each subscriber. All issues will be delivered by UPS (vs. bulk mail!) to your User Group address. You will recieve the issues well before regular subscribers. A bonus for your club members! If you have a POB please provide an alternate street address or use the Canadian rates. Canada User Groups will be shipped by First Class/Air mail. Overseas will be shipped by US ground.

| | | | |
|---|---|---|---|
| Subscription information: 6 issues a year | $17.00 for the U.S. | $25.00 for Canada | $35.00 Overseas |
| Subscription with a disk: 6 issues a year | $26.00 for the U.S. | $34.00 for Canada | $44.00 Overseas |

---

To add subscriptions to your User Group subscription (either subcriptions with or without the disk) just subtract the last issue received from your last issue number due "a", divide the yearly subscription by six "b", and multiple your User Group issues due "a" times "b". Example:

```
        Subscription issue end      : XX
        minus (-)
        Subscription recieved so far: XX
                                      -----
                    Issues due: (A)         Yearly subscription requested divided by six = (b)
                                                Total due for additional subscription = (b) x (a)
```

---

## Twin Cities 128 Compendium Book #1

Compendium Book #1 is the best of the best from issues #1 - #18 of Twin Cities 128, the C128 specific computer journal.

From hard hitting reviews, tips, tricks, CP/M articles, gossip, features, interviews, opinions, editorials, and how to squeezing the most performance from your C128 VDC 80 column controller chip. Written by C-128 experts and users such as Loren Lovhaug, Avonelle Lovhaug, John Kress!, Bruce Jaeger, Miklos Garamszeghy, Randy Margolis, Jim Butterfield, Frinkle Prindle, and Fred Bowen (Commodore's engineer).

The best of the best and it is all in here for only $16.95 plus S&H !

© Published by Parsec, Inc. P.O.Box 111 Salem,MA 01970-0111 U.S.A. Copyright 1991 ©