

VAST



NEWS

VOLUME 12 NUMBER 7

Established 1984

JULY 1996

The Newsletter of the VALley of the Sun TIGG/ers

REVIEWED

REVIEWS

ISSUE

INSIDE INFORMATION

ITEM	PAGE	ITEM	PAGE	ITEM	PAGE
GRANKRACKER.....	2	GK UTILITY1.....	2	GRANKARTE.....	4
TRIPLE TECH.....	5	PDM MODULE.....	5		

DISCLAIMER

the VAST COMPUTER USERS GROUP can assume NO responsibility for the accuracy of the information in this newsletter or for the programs or for construction projects tried by its members or others. YOU try them at your own risk!

Material contained herein may be republished as long as credit is given to the original source.

GRAM KRACKER - SO WHAT!!

from
RAY'S RACK 248

Some of you may be wondering what all the fuss is about over GRAM KRACKER. Of course you can save your modules to disk, but that's only the tip of the iceberg.

I suppose some technical talk is in order for many of you to fully appreciate what we have here.

The TI99/4A has a 16-bit address buss. If you convert 16 binary bits to the equivalent decimal number you will discover that the CPU (Central Processor Unit) which is the brain of the computer can access 65536 memory locations.

Much of this memory does not exist in the TI console but instead in the P-Box (32K) and the GRAM port (8K). In fact, there is only 256 bytes of CPU addressable RAM in the console. The 16K that comes with the console is not directly accessible by the CPU, but is a memory mapped device. Most computers make use of mapping to expand the memory capabilities beyond the capability of CPU address space.

The mapped devices generally communicate with the CPU by having several common addresses. One (or several) addresses tell the mapped device what address in it's space you want to communicate with. Other common addresses are used to transport data back and forth. If a device is RAM, it can be both written to and read from. If a device is ROM it can only be read from.

The TI is sort of like an octopus with arms extended to devices in all directions. In particular is the concept of GRAM which is one of the mapped devices in our machine that has had so much to do with the 99's history and uniqueness in the computer world.

GRAM is sequentially accessed memory which has some unique characteristics relative to other areas of the machine. First, the GRAM chip itself is proprietary and was basically TI's method of keeping control of the software. The use of GRAM in modules was intended to stop piracy and to control software for the machine.

The use of GRAM in the console was a memory space saver, and enabled coding a large part of the operating system and console BASIC in GPL. Graphic Programming Language (GPL) is an assembly type language. It's characteristics are somewhat different from 9900 assembly in that it's a byte oriented language (8 bits = 1 byte) as opposed to 9900 code which has both byte and word (two byte) instructions. GPL seems to make widespread use of macros (canned routines which can be CALLED in any program) and supposedly is easier to program in than 9900 code.

TI never came out with a GRAM version of 9900. You see, 32RAM can be reprogrammed just like 32R, and 32R cannot.

What all this means is that the 99/4A has been described as having closed architecture. Architecture refers to the way the electronic building blocks (CPU, RAM, ROM) are interconnected to form a computer. A machine with open architecture with the same addressing capability as ours would have 64K of RAM that could be addressed by the CPU. Any ROM (such as the operating system code which is the wake-up routines in the machine) would be dumped to RAM on power-up. It would be simple to change this code. It would also be simple to download any module, as TI gave us a closed machine.

The GRAM KRACKER has opened it back up!!! It is now possible to change the operating system which resides in GRAM 0 because 6K has a GRAM 0. As more programmers become familiar with GPL code, you'll see more changes becoming available to the TI community.

The code is already there to bypass the color bar screen, auto-start a menu option, slash the zeroes in both console and Extended Basic. Similarly, GRAM 2 of 6K can be used to modify console Basic. The remaining 56K in GRAM KRACKER simulate the GRAM and ROM that the modules use.

The point is that not only can the modules be copied, but they can also be altered. Already a number of changes have been offered (most by Millers Graphics), but look for a major overhaul of Extended Basic. We already have such new commands as CALL CLOCK and CALL CRT, as well as the ability to change the default screen/text colors. The number of possibilities is almost endless. You can now customize the modules to YOUR liking and to adapt to specialized situations and applications.

Hopefully you can see that the GRAM KRACKER represents a major step in the evolution of our dear system.

OK UTILITY DISK 1

from
RAY'S RACK 248

Millers Graphics has finally released the first utility disk for GRAM KRACKER. Since the disk is numbered, I assume we may expect more in the future.

The utilities were done by Danny Richard (HEATLIST and SCREEN DUMP). There are 22 pages of documentation on the disk and a short program is included to print them out in an orderly fashion.

First, you have to save your GROM 0, Extended Basic, and E/A cartridges. Then E/A is loaded in and moved to GROM 7 with the E/A-MOVER program that came with the GK. Next, XB is loaded and then you use the Load Module function of GK to load the XBEAPATCH program.

Follow the prompts on the screen and the whole process takes just a couple of minutes at the most.

What you will end up with is a combined version of XB and EA which will load into the GROM KRACKER in one big chunk. You will now have seven disk files, but that includes not only XB and EA, but also the EA Editor and Assembler.

You should also save the contents of GROMs 1 & 2 for reasons I'll explain later.

Now here's the advantage of all of this...lots of enhancements to XB and EA. A number of new calls and commands have been added to XB.

CALL INIT - this command loaded more data into expansion RAM than necessary and this has been corrected.

LIST - the LIST command can now be output to a device using a specified line length of 1-255.

CALL LOAD - if you are poking data into a memory location, CALL INIT does not have to precede the command, but is still required for loading object code.

RESEQUENCE - has been abbreviated to RES. You can now RES just a part of a program using any increment you want.

TRACE - can now be output to a device. You have to open a file in a program line as OPEN #123:"DEVICE" and then you can toggle the TRACE function on and off within the program as you desire. I imagine most people would output to a printer, but it will work with a disk file.

COPY - lets you copy a line or block of program lines to any other location in a program. The original line or lines remain intact. This routine does not change any references to the copied lines, it is an exact copy of the source lines with new line numbers.

DEL - will delete a line or group of lines from a program. Sure beats zapping them one at a time.

MOVE - moves a line or block of lines to another location in the program. When the block is moved, it will be renumbered and all references to those lines will be changed to reflect the new line numbers.

CALL PEEK6/POKE6 - allow you to read/write data to/from GROM/GROM at a specified address.

CALL PEEKU/POKEU - lets you read/write data to/from UEP RAM.

CALL QUITON/QUITOFF - lets you disable/enable the FCTN - Keystroke.

CALL EA - moves you directly to the EA menu, assuming you originally set up the GK to hold both EA and XB.

You can now hold down any key to abort the DISK LOAD when you start up XB. There is a new character set and the colors for XB command mode are now white on blue with the old colors remaining in RUN mode. Error messages are now in upper/lower case.

CURSOR MOVEMENT - Ever try to edit one of those lines where the programmer packed everything he could into one line....but you needed to change something at the end or the middle of it? It's a real hassle to hold down FCTN D and wait for the cursor to move across each line until it hit the place where you wanted it. Now, SHIFT FCTN D will place you at the end of the program line and SHIFT FCTN S puts you at the first (the program line, not the screen line). Also, SHIFT FCTN E and X move the cursor up and down screen lines just like you do when using a text editor. It isn't full-screen editing, but it's got to be the next best thing.

Of course, all of the CALLs from the utility disk that came with GK are still valid, such as CALL CLEAR, CALL CAT, etc.

Editor/Assembler has not been left out of this fine enhancement package. Most of the EA changes are cosmetic, but very useful.

Repeating keys have been added, along with FCTN 3 (CHANGED). The SHIFT FCTN S and D work similarly in the EA menus.

You now have filename recall at any point that requires a filename, device name, or option entry. The last thing input at any of these places will stay in memory until you use it next. That is the reason for saving GROM 1 & 2 because this information is stored there along with the Assembler. Since the Assembler is stored in GROM, loading is virtually instantaneous.

Option #6 on the EA menu lets you jump immediately to XB. Option #7 lets you format a Myarc RANdisk, and Option #8 lets you catalog any device that conforms to standard catalog routines.

Also presented in this enhancement package is an explanation of what data is in each of the 7 GROMs so you can change colors, the format defaults for the Myarc RANdisk, and other technical goodies.

GRAM-KARTE
from
RAMM's RUMOR MAG

DSK1.MBASIC1
DSK1.MBASIC2
DSK1.MBASIC3
DSK1.MBASIC4
DSK1.MBROM1
DSK1.MBROM2

I've had lots of people calling and writing me asking about the Mechatronic hardware from Germany. Since I am not affluent, I can't afford to buy every piece of hardware and software that comes out.

I'd like to extend many thanks to Franz Wagenbach of T.A.P.E. for loaning me a card to try out. It is through the kind efforts of people like this that I can check out all the new things coming to the marketplace. THANKS Franz!!!!

The GRAM-KARTE comes in a 128K and 512K version. What they do is remove five chips, a DIP switch, and the PROM and piggyback a board in their place. I tested the 128K unit and I think the 512K version would be far more useful. What the GRAM-KARTE does is allow you to store your modules on disk and load them into the card when you want to use them. Since I have had a GRAM-KRACKER for some time, I wanted to see how this device differs and which one is the best (if either).

Just put the card in whatever slot you desire in your P-Box. When you turn on the computer (with no module in place) you will see **GRAMCARD**, **19800**, and **REVIEW MODULE LIBRARY** on your menu screen. The **REVIEW MODULE LIBRARY** is there in case you have multiple GRAM-KARTEs in your system. According to the docs, you can have your P-Box filled with them (just 2 512K units per box).

Selecting **GRAMCARD** from the menu brings you to a master menu with eight files listed (another eight can be toggled in by pressing FCTN 7). If you have saved some modules already, you can load them with one keystroke.

OK, so you want to save a module... just insert it as you normally would and enter Basic. Type in **CALL GRAM** and you will see the loader menu for the card. Before saving any modules, you will need to know what kind of memory the cartridge has, such as **GRAM/ROM**. Each bank of **GRAM/ROM** must be saved as a separate file. The documentation has a listing of Ti modules that tells what banks are in what places. It would be much simpler if there were a way to determine what banks are in a cartridge so a non-technical person could have an easier time of it.

You also need to know the addresses of each bank, but this information is given in the documentation.

Once your **GRAM/ROM** banks are saved, you need to create a **D/V 80** file that tells the loader which files to load. For example, Extended Basic has 4 **GRAM** banks and 2 **ROM** banks, so your file would look like:

Loading a module is as simple as entering the filename of the **D/V 80** file. If you want to erase whatever module is in the card, just hit **FCTN 4** upon reset (such as after entering **BVE**).

There are several **DIP** switches on the card for setting the **CRU** base and **GRAM** address.

Comparisons to **GRAM KRACKER** are inevitable, so here are a few of my observations.

First, you cannot have a **GRAM KRACKER** and a **GRAM-KARTE** in your system at the same time. When you go to load files, part goes to each device and things get real screwed up.

The card does not have any type of battery or AC back-up and I think this would be an enormous improvement. Also, you will find that 128K isn't much to work with. The docs say you can store 2 cartridges in the 128K model, but I could not figure out how to do it. Loading files with the card is a lot faster than with **GRAM KRACKER**, especially when using a **RANDISK**.

The documentation needs to be written for less experienced users. I had a real tough time figuring out what was going on. There needs to be some way to know what **GRAM/ROM** banks are in a cartridge. Although the docs list many cartridges, there are a lot not on the list and I have no way of knowing if there are 2 **GRAM** banks and one **ROM** or whatever. How do you know?

The necessity of writing a **D/V 80** file to tell the loader which files to load is a bit "Nicky Name". **GRAM KRACKER** just increments the files with numbers.

Between the two similar types of products, **GRAM KRACKER** is definitely the easier of the two to operate. The memory editor for the card does not have the search and copy features of the **EK**.

In all, I would probably like the card if it were easier to use with better written documentation.

The **GRAM-KARTE** does have the advantage of not needing anything plugged into the console to operate, however.

TRIPLE TECH
from
ROBERT'S RASCAL RAG

As you may be aware, CorComp has released a new piece of hardware for the TI called the Triple Tech card. I would like to devote a fair chunk of space this month to this incredible piece of hardware.

The Triple Tech card is actually three devices in one. There is a clock/calendar, 64K print spooler, and the ability to put your speech box inside the P-Box.

First remove the circuit board from your speech synthesizer and plug it onto the Triple Tech board. Triple Tech doesn't use a metal case like TI's cards. Now, when you plug Triple Tech into your P-Box, you can access the speech capabilities of the TI just as when it was hanging on the side of the console.

My #1 console has had trouble with the speech box, so I have left it off and gone without speech for some time. Now, I have my speech back and it's well out of the way.

The second feature of the card is the clock/calendar. The clock is addressed very simply as:

OPEN #1=CLOCK

Then, you use an INPUT statement to set the clock and either a PRINT or DISPLAY AT statement to put the information on the screen. Even a non-programming stooge like me can figure that out.

The clock is powered by a 3 volt lithium cell which is very accessible and will last for 6 months (according to CorComp).

It should be very easy to incorporate the clock feature into your existing Basic and Extended Basic programs. The clock keeps time in military (24 hr) format.

Finally, there is the most useful feature, the 64K print spooler. When use with TI-Writer's Formatter, or LISTing a program or whatever, the spooler will load up with whatever you're printing and return control of the computer to you.

I was able to print a document from TI-Writer at the same time as I was running a game in Extended Basic (that also used speech)!! This spooler is a dream to use. Simply specify PIO for your printer name as always and that's all there is to it. Operation of the spooler is totally transparent to the user.

The cable from your printer attaches to the Triple Tech card and the card has a cable which hooks onto the parallel port of your RS232.

The card has a built-in diagnostic which will check the print spooler.

There are two buttons on the card labeled COPY and CLEAR. CLEAR obviously clears the buffer and COPY lets you make copies of what is in the buffer.

So, let's say that you have started to print a document and then decided to play Pacman. But after the thing has printed, you remember that you need another copy.....just hit the copy button and that's all there is to it.

The card seems to have provision for parallel printers only, so I don't know what you people with a serial printer will do. I wish there were an easier way to get to the COPY and CLEAR buttons instead of having to reach behind your computer, but I don't know how else they could have done it.

The documentation tells everything you need to know in language that anyone can understand (I did).

PERIPHERAL DIAGNOSTIC MODULE

from

ROBERT'S RASCAL RAG

This module will test both RS232 ports for baud rate and I/O. A special plug will be included that will allow you to check both ports without having a modem in the line or a printer connected.

The cartridge will also check the 32K card to make sure everything is in order.

Finally, the PDM will do a destructive disk test, display head step times, and initialize disks in all of the four common formats (SSSD, DSSD, SSSD, and DSSD). The program will also perform a motor speed test similar to Craig Miller's Diagnostic.

CorComp tells me that the module will be compatible with the Myarc RS232 card with the exception of the loopback tests, and is compatible with both the TI and CorComp disk controllers. They told me that every effort had been made to make it compatible with the Myarc controller, but they were unable to do so.

You may think that many of these features are redundant when compared to Craig's fine software, but remember you need a working 32K and disk system to use his. Since the PDM is cartridge based, a working disk system and 32K are not required.

VAST USERS GROUP INFORMATION

The **VAST COMPUTER USERS GROUP** is a support group for the Texas Instruments TI-99/4A Home Computer and Geneve.

The **VAST COMPUTER USERS GROUP** operates a BBS 24 hours a day, 7 days a week. the phone number is:

CURRENT OFFICERS

(602) 267-1419

PRESIDENT

Ralph Rees.....582-0800

VICE PRESIDENT

Jack Workman.....437-3187

SECRETARY/TREASURER

Hazel Knight.....938-5446

LIBRARIAN

Wallace Knight....938-5446

NEWSLETTER EDITOR

Ralph Rees.....582-0800

BBS STAFF

Jim Ely.....UNLISTED

ADVERTISING: There isn't any charge to paid members for **PERSONAL** advertising. Non-members will be charged at a cost of \$1.50 per ad.


NEWSLETTER EXCHANGE: We exchange club newsletters with many TI Users Groups. Contact Our secretary.

Opinions expressed herein are those of the writer and not necessarily those of the **VAST COMPUTER USERS GROUP**.

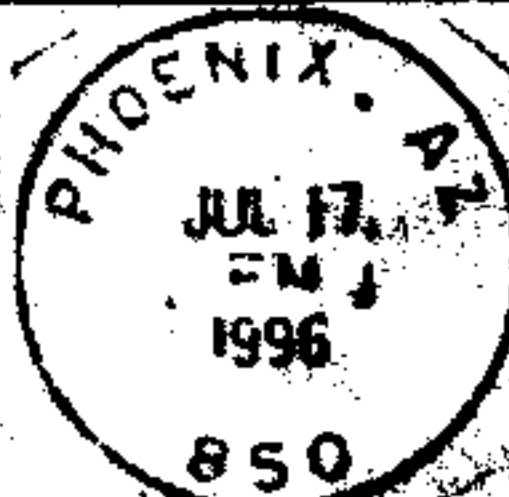
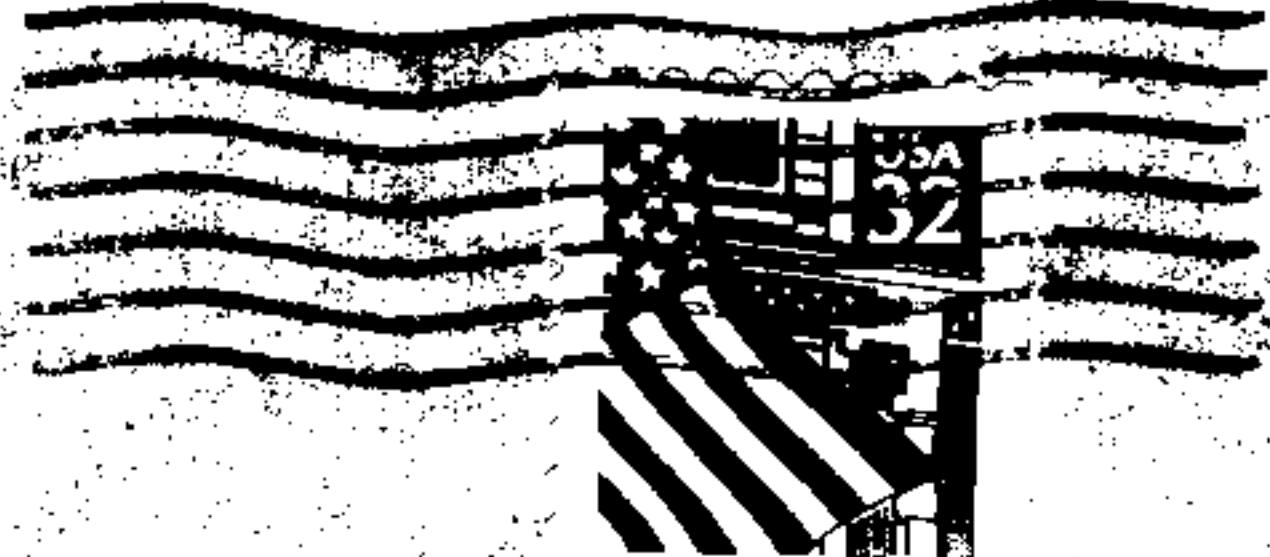
Send any correspondence to the address below.

NO JUL. AUG. OR SEP. MEETING

THIS NEWSLETTER IS COMPOSED IN ITS ENTIRETY USING A **TEXAS INSTRUMENTS TI-99/4A** COMPUTER.



VAST NEWS
c/o Ralph E. Rees
P.O. Box 37725
Phoenix, AZ. 85069

FIRST CLASS MAIL

MR. BILL GASKILL
2310 CYPRESS COURT
GRAND JUNCTION, CO 81506

FIRST CLASS MAIL

