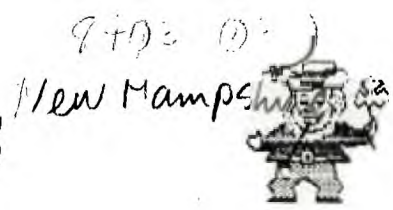




NEW HAMPSHIRE 99'ers



MARCH 1987
NEWSLETTER VOL 5 NO. 3
POB 5991 MANCHESTER, NH 03108

>OLD

The last meeting saw me expertly disassemble the major components of our trusty TI. The major components were explained, as well as how to replace some of them (keyboard, power supply, GROM extender). Although we didn't do any mod's, the RESET and LOAD switch options were addressed. Fortunately, we started the raffle while I fumbled trying to get the system back together.

Helene donated a copy of SAM'S COMPUTER FACTS for the TI which Glen Hammond won during the raffle. The system was eventually re-assembled and actually worked the first time it was powered up!

As I have been mentioning for the last few months, I am not running again for President. I have enjoyed serving the club, and will continue to be an active member (including demo's). Is the President's job that taxing? Not really. I suppose my major task is to organize and run the monthly meetings. I do answer correspondence sent to the club, which is more of a priveledge than a responsibility (since I get to see everything first!). I also have been supplying the color monitor each meeting, as well as toting a variety of manuals, hardware, disks, etc. Let it be known that I can't thank Roger Roy enough for bringing his system to the meetings. Unfortunately, Roger has taken his system to Maine (his sister I believe) and we will be reduced to using the club system.

Which brings me to another point! When are we going to get the club system upgraded to two 1/2 heights, DS/DD? Also, when are we going to get a color monitor for the club? The latter item is a pet project for obvious reasons - I will not continue to haul my monitor to meetings each month! I propose that we all search for the cheapest color monitor available and purchase it while we have the resources. The extra drive and capacity upgrade can wait a few months.

>NEW

Don't forget the fair coming up on April 4th. It is to be held in Waltham, though I am not sure of the location. I will upload more data to the Progressive Connection (644-3507) when I get it. You may call me after April 1st for directions (if you don't have a modem to call the Progressive Connection).

For those of you south of Manchester, the Progressive Connection is about to install a new 434-**** number. This will relace the 434-6225 number which was disconnected a few months ago. Fritz Muller, owner operator of the Progressive Connection, apologizes for the whole situation and assured me that members' expiration dates would be adjusted to reflect your availability. Simply CHAT with him or drop some email. After hard disk wipe outs, program back up failures, and losing his southern New Hampshire line, Fritz is still optimistic and has come out of the corner fighting. Let him know that there are TI enthusiasts looking for a 'home'. I should mention that he is seriously interested in purchasing a complete TI system in the near future, just to upload, download, and check our programs for us! You can't beat that for service!

I will make every attempt to demo the telecommunication abilites of the TI at the next meeting. This will require two TI's (which I have) a null modem cable (which I have) and a set of BBS software and assorted telecommunication programs (I have no BBS and only FASTTERM). Those of you with /4TALK or MASSTERM or whatever, please feel free to bring them along.

Many people not at the January meeting have asked about the MYARC computer. "What do you think...?" or "I was going to get a clone, should I wait and get a MYARC...?" I would rather address those questions (and others) in a separate article.

This month I have included a brief example on C. I would like to do something like this each month, if there is interest. I will warn you right now that I am not an expert in anything (tho' I don't think I'm too bad at Assembly or the BASIC's). I would also welcome articles from any member.

IF YOU WOULD LIKE TO SUBMIT AN ARTICLE:

Please submit the article on a disk. I will gladly return the disk with the article intact, as well as some programs from my collection of over 200 disks (my choice - no copyrighted material, unless my own). If you absolutely can't send a disk, please use the following formatter commands:

```
.LM B;RM 62;FI;AD
```

You must also use compressed type. On the TI Impact printer (and most other printers that I know of) you can get that from the editor by pressing CTRL U then SHIFT O then CTRL U again. Thanks; and see you in April at the fair and the meeting!

BYE

ASGARD'S TOTALLY
UNBELIEVABLE OFFER

Richard J. Bailey
68A Church Street
Gonic, N.H. 03867
NH99ER USER GROUP

While poking around on Comuserve one evening I ran across an offer from Asgard Software for RLE pictures. All I had to do was send 20 initialized disks and \$10.00 to Asgard. I decided to also order GRAPHX Companion III for \$9.50 at the same time and sent my order off on November 9th, 1986.

By the middle of January, I had received nothing from Asgard so on January 19th, 1987, I sent a letter to Asgard asking if there was a problem. Now it's the middle of February and I have heard nothing from the people at Asgard.

So I can't suggest you not order from Asgard because that's a decision you have to make for yourself but if you want to throw 20 disks away, my address is at the start of this article and I consider myself more needy than Asgard. I will also send you a nice reply which is a lot more than you can expect from Asgard.

Even if I eventually get my disks back (which isn't likely) the service I've got from Asgard is unforgivable. Fortunately for you I can't say exactly what I think of Asgard in this article but if you have a good imagination and a rich vocabulary, I believe you'll get the idea.

OH SAY CAN YOU 'C'?

Curtis Alan Provance
New Hampshire 99er's User Group

Tho' the title would have been more appropriate this July, the topic is hot NOW. C is quickly becoming the language of choice of many programmers for a variety of reasons. C is a high level language with similarities to PASCAL, and therefore is a structured language. Programs written in C are generally quite readable and easy to understand. C also contains commands and functions normally found only in Assembly. It therefore allows many operations not available in PASCAL or other high level languages. Finally, although there is no standard for C, the language is very portable. I have seen numerous programs in C written for the IBM (or compatibles) which require no modification for compilation on a TI.

NOW! What is 'compilation?' C is a compiled language, which means you write a program in C and save the program as 'source code.' You then run a compiler which translates the source code into machine instructions and saves them as 'object code.' This code must then be 'linked' to external routines which you may have used in your program, or the address pointers set, or whatever. It may sound like a lot of work, but it really isn't as complicated as I make it seem. The best part is that after the object code has been loaded and linked, it runs faster than any BASIC program could possibly run. I'm not talking twice as fast, I'm talking tens, or hundreds times faster, depending on what you're doing. There is one other really nice aspect to C:

There is an excellent C compiler for the TI.

Clint Pulley has done a fine job of writing 'c99' which is a collection of files used to write C programs on the TI. This disk is in the club library and copies are available. As with all 'FAIRWARE' disks, the minimal copying charge is not a substitute for paying the author for his or her labor.

We will be sporting a C column in the newsletter shortly. Programs appearing in this newsletter will be developed, compiled, etc. using the c99 disk mentioned above. Anyway, back to C -

As I said, C is a compiled language, which allows your program to be executed very quickly. Unfortunately, errors in machine language programs are harder to find than in BASIC. If your C program screws up, you will not receive a nice message such as 'STRING - NUMBER MISMATCH IN 200.' Instead, your program may go wacky, or you may not notice any error at all - you'll just get the wrong results. For this reason, you should take care to write programs anticipating mistakes. As I mentioned in last month's BASIC column, assume the user knows nothing. Expect strings instead of numbers, division by '0', or whatever.

What does C look like? Let's take a small program

in BASIC and C and compare them. The following program determines the greatest common divisor of two positive integers, based on an algorithm developed by Euclid around 300 B.C. I made every attempt to use exactly the same logic in both programs (otherwise, it wouldn't be a comparison). Bear in mind that this may not be the best way to write each program; I am simply providing them for comparison. I will comment each program in the code itself.

BASIC VERSION

```
100 REM CALCULATE GREATEST
COMMON DIVISOR OF TWO
POSITIVE INTEGERS
110 INPUT "INPUT HIGHER #, L
OWER #:";HIGHER,LOWER
120 REM MAKE SURE NUMBERS
ARE POSITIVE INTEGERS
130 HIGHER=INT(ABS(HIGHER))
140 LOWER=INT(ABS(LOWER))
150 PRINT "GREATEST COMMON
DIVISOR OF";HIGHER;"AND";LOW
ER
160 REM IF EITHER NUMBER IS
ZERO, QUIT ROUTINE
170 IF HIGHER=LOWER THEN 180
ELSE 290
180 REM MAKE SURE 'HIGHER'
GETS THE HIGHER NUMBER
190 IF HIGHER>LOWER THEN 240
200 REMAINDER=HIGHER
210 HIGHER=LOWER
220 LOWER=REMAINDER
230 REM GET THE REMAINDER
OF DIVIDING HIGHER BY
LOWER. IF NO REMAINDER,
THEN LOWER WAS GREATEST
COMMON DIVISOR.
240 REMAINDER=HIGHER-LOWER*I
NT(HIGHER/LOWER)
250 HIGHER=LOWER
260 LOWER=REMAINDER
270 IF REMAINDER THEN 240
280 PRINT "IS";HIGHER
290 END
```

C VERSION

```
/* CALCULATE GREATEST COMMON DIVISOR
OF TWO POSITIVE INTEGERS */

main ()
(
    int higher, lower, remainder;

    printf ("INPUT HIGHER #, LOWER #:\n");
```

```

scanf ("%d,%d", &higher, &lower);

/* MAKE SURE NUMBERS ARE POSITIVE INTEGERS */
if ( higher < 0 )
    higher *= -1;
if ( lower < 0 )
    lower *= -1;

printf ("\nGREATEST COMMON DIVISOR\n");
printf ("OF %d\nAND %d\n", higher, lower);

/* IF EITHER NUMBER IS ZERO, QUIT ROUTINE */
if ( higher * lower )
{
    /* MAKE SURE 'HIGHER' GETS THE HIGHER NUMBER */
    if ( higher < lower )
    {
        remainder = higher;
        higher = lower;
        lower = remainder;
    }

    /* GET THE REMAINDER OF DIVIDING HIGHER
    BY LOWER. IF NO REMAINDER THEN LOWER WAS
    GREATEST COMMON DIVISOR. */
    do
    {
        remainder = higher % lower;
        higher = lower;
        lower = remainder;
    }
    while ( remainder != 0 );
    printf ("IS %d\n", higher );
}
}

```

Several comments are in order before you pass judgement: 1) I am not a professional C programmer (or BASIC programmer for that matter); 2) as of the printing of this article, I have not yet compiled this source code; 3) there are a few differences between the two programs which I must address.

Note that the C program is (for the most part) in lower case. C makes a distinction between cases; i.e. a variable named 'higher' and a different variable named 'Higher' (or 'hIgher', or....) could all exist in the same program. Don't do it, tho'; it makes reading the program harder.

You may also have noted the liberal use of spaces and lines. This is not only perfectly acceptable in C, but also encouraged. Spacing makes the program easier to read, as well as setting off loops, subprograms, etc.

The 'printf' function allows many more formatting commands than in BASIC. For example, \n means advance to the next line (like : only it can be in the middle of a printf statement); %d means decimal integer. There are format commands for floating point, strings, accuracy (decimal places), leading zero's, etc.

The 'int' declaration in the sixth line means that the three variables following it are declared to be integers. This has several advantages over floating point (BASIC) numbers: 1) integers take up two bytes, floating point takes up eight; 2) integer arithmetic is **much faster** than floating point; 3) displaying integers requires less formatting. Integers do have the problem of being inaccurate, i.e. if you divide an integer 10 by an integer 3, you get an integer 3; the remainder of 1 is discarded. Sometimes this is OK and exactly what you want. There is also a function called 'MOD' which returns the remainder of integer division. This function is represented by the percent sign '%' (don't confuse it with the formatting symbols which are % followed by other characters) and is used in our program above.

Other interesting C commands include the 'assignment' operators. If you want to multiply a variable by a number, then store the result in the variable, use the '*' operator. This can be seen in the makeshift 'absolute' function. If higher is less than 0, multiply it by -1 to make it greater than zero: higher *= -1. There are comparable operators for +, -, /, MOD, OR, AND, XOR, SHIFT RIGHT, and SHIFT LEFT (the word commands represented by special symbols).

As you may have guessed, there is no ABS command. That doesn't mean we can't write one, though! One of the nice features about C is that you may write a function and call it as if it were built in. This is similar to BASIC's DEF, only much more powerful. You may pass multiple parameters, and have the function definition consist of multiple lines of code.

The indentation helps to track program flow. For example, on the 12th line, there is an 'if' statement which says (in English) 'If higher*lower doesn't equal 0, then both must be non-zero. Go ahead and execute the program contained in the brackets.' If the product had been zero, all the lines in the brackets would be skipped. This being the case, if either number is zero, you may drop straight down from the open bracket '{' to the close bracket '}'. This marks the end of the 'if' statement and is the point where you pick up the rest of the program.

There are many more things we need to talk about before we hope to become fluent in C, but they must be addressed over volumes of newsletters. I would like to point out one more item: the 'do - while' loop. C provides not only a 'for - next - step' loop (of sorts) but also the 'do - while' loop and a 'while' loop. There are also provisions for 'switch' which is similar to ON..GOTO only much more powerful and expressive.

If you are interested in obtaining a good starting book on C, I recommend PROGRAMMING IN C by Stephen G. Kochan. Other book titles will follow.

THE POWER OF THE TI
Curtis Alan Provance
New Hampshire 99er's Users Group

If I had a nickle for every time someone said "Why don't you get a **real** computer..." I'd have enough money to buy a MYARC 9640! Actually, I have gone through various stages of responses ranging from "Oh yeah! Well @&Z\$! you!" to laughing in their face.

Let's be realistic, though. The TI has had a lot of bad PR (public relations) since it first came out years ago. Why is that? First, look at the crap that TI passed off as software. I have written BASIC programs that were more powerful than some of the modules TI sold for \$30 and up. How about peripherals? You want a PEB for \$300? It doesn't do anything except let you add other \$300 peripherals. I'll take a dozen, please.

Those were my "Oh yeah!..." days, when I felt that I had been taken. Black Friday (when TI pulled out) was like a knife in the back. However, now that I can look back on my first year (and laugh!) I can honestly say that TI dropping us was as if our chains of bondage had been broken! Let's look at what's available....

Memory: this is frequently where the 'power' of a system shows up. You can't read an ad without getting the RAM spec's, what comes with it and what it is capable of doing. I say this is the 'power' of the system because anything that is done must be done by a program. The more memory available, the bigger (i.e. powerful) the programs may be. The stripped down TI only came with 8K of ROM, 18K to 6ROM, 16K of RAM (some used by the screen and BASIC interpreter) and a 256 byte scratch pad (working) CPU RAM. A cartridge adds as much as 42K of ROM - RAM - 6ROM. This was great in the early 80's, but peanuts now.

Along came peripherals. Most peripherals require some type of machine code interface (disk controllers, RS232's, etc.). These peripherals typically have at least 4K of ROM (RS232) and in some instances (CORCOMP and MYARC disk controllers) have more than one bank of 8K. Now we have a big pile of peanuts.

But wait! Look! Up in the peripheral expansion box! It's a green light! It's a red light! It's **RAMDISK!**

Whether Foundation, CORCOMP, MYARC, New Horizon, Mechatronic, or whatever, RAMDISKS have revolutionized my TI life. Each with merits of its own, these cards (more appropriately called RAMCARDS since they do more than emulate disks) can put the TI back into the lead concerning powerful home computers.

You think I've lost my mind? Consider this: a modified New Horizon RAMCARD (my term) contains 256K of RAM. Since this card may occupy any CRU address, you can fill any empty slot in your PEB with one of these beauties. I have four slots open in my PEB at home - that's 1 megabyte of CPU memory available to me **battery backed!** These cards can be used as extremely fast RAM disks or can contain CALL routines for BASIC or Extended BASIC (with appropriate linking routine)

programs. I also have a 512K MYARC RAMDISK. The machine code provided with the card makes using it as a disk or print spooler extremely easy - but I could use it to store routines if I wanted to! CORCOMP also has a similar card and has developed a word processor - formatter - spell checker which will (if the description is true) blow the overlays off an IBM or clone.

Have you heard of a 'super cartridge?' This module doesn't add a lot of memory to your system, but it is usually battery backed and is like having a portable chip set. Plug it in, and your program is available (machine code, not BASIC or Extended BASIC).

I have only mentioned the memory aspect of the TI, without going into comparing the colors, speech, sound registers, etc. of other home computers.

If I may tell a short story I believe I can get my point across: the company I work at has 38 VAX 11/7XX's, IBM mainframes, hundreds of micro's, etc. I have access to most of these systems through a Local Area Network. After trying MASS11 (word processor) on the VAX, I broke down and brought in a TI setup. Three office buddies (including my boss) who own IBM's (or clones) started the ribbing. I plugged in my Qubert cartridge and showed them the graphics - they stopped laughing. I plugged in the Terminal Emulator and let the TI talk to them - they listened. I played 'Axel F' with three sound registers and a noise register (IBM can only make one noise at a time). One of them wanted me to see 'CENTIPEDE' as written for the IBM. It was pitiful. I walked them over to my cube and demo'd the real thing for them. They haven't asked me to look at any more of their programs.

Sure TI has far better games, but what about Lotus? Flight Simulator? dBASEIII? You've got me there. I admit there aren't programs of that magnitude available - yet! These RAM cards I mentioned are still in the infant stage. Give programmers some time (and incentive) to develop programs that use TI's new found power.

The TI is capable of programs such as these, but we must want them! Millers Graphics has developed a fantastic emulation program called Explorer. A finer program hasn't been written for the TI (personal opinion). Yet, within a few months (weeks?) of its release, there was a pirated version available. If TI owners supported both commercial developers and fairware authors, instead of pirates, more excellent programmers would have stayed with us. Unfortunately, judging by the speed in which pirated programs make it around the country, I'd say as a large group of computer enthusiasts, we have a bad reputation.

Moral of the story: for the price of a RAMCARD (couple of C notes) you could have it all. You must also 1) not support pirates (which costs you nothing) and 2) support commercial developers and fairware authors (which typically costs \$10 to \$20 for a good program). If you can't do either of these, go buy a clone (\$600 to \$2000) and start shelling out hundreds of dollars for "the neat stuff."

PATCHES
Curtis Alan Provance

EDITOR LOCKUP
Curtis Alan Provance
New Hampshire 99er's Users Group

If you are unhappy with the way a BASIC or Extended BASIC program works, you can usually fix it. If, however, your problem is with a machine code program, chances of fixing it aren't very good. There are some simple changes that may be made, and we will discuss one or two each month.

First, you need a disk editing (or patching) program. Funnelweb comes with a simple disk patch program. There is also Advanced Diagnostics, John Birdwell's excellent fairware disk editor, DISK+AID, and others. The only requirement is that you be able to read sectors, modify them and write them back to the disk.

This month's patch concerns the formatter used with TI-WRITER (it also applies to the formatter used in Funnelweb, BA-WRITER, or whatever). You may change the formatter to recognize different bold and underline codes.

Why would you want to do this? I don't know about anyone else, but I use @ and & quite a lot. I can't get by without @ in my Assembly source code, and & is very common in BASIC string commands. Anyway, I looked at my keyboard and decided to use ! for @ and _ for &. You can change bold and underline to anything you want. Here's how:

Using a disk editor, search through the first formatter file (using ASCII, not hex) until you find the two format characters @&. There is only one place where these appear together. Now type over them with the two characters you want for bold and underline. Save this changed sector back to disk and viola! New formatter codes.

As with any disk modification, I recommend you do this on a copy first, test it to ensure it does what you want, then use it as your master once you're satisfied.

Another patch: this one slows down the auto repeat speed in DM 1000. Search through the first file (in hex this time) for 00A0. The word before should be 0380 and the word after is FF00. 00A0 is the delay; the larger this is, the longer the delay between auto repeat. I changed mine to 0200 and it is a little too slow. Experiment with several values until you find one that suits you.

Ever since I began using the TI-WRITER editor, I have experienced lockups with the REPLACE STRING subroutine. In particular, lockups seem to occur only when I do a replace string involving space characters. Last night it finally dawned on me! It wasn't a lockup afterall! Let me give you an example:

If I want an entire line to be bold or underline, I use the bold character before each word. I do this so that the line will wrap around naturally, fill, and adjust. If I had used the bold character in front of the line, then the required space between each word, the fill, adjust, etc. wouldn't work properly.

Anyway, I don't type dozens of @'s, I simply type in the line with normal spaces, then use REPLACE STRING. For example, I could put the cursor at the beginning of this sentence, go to REPLACE STRING, and type / / @ / for my choice. Here is where the lockup can manifest itself. As long as you answer the prompt (All, Yes, No, Stop) with 'Y' for each character, everything is OK. If you answer the prompt with 'All', you might as well go get a snack - it could be five or ten minutes. This is what's happening:

When you press 'A' (for All) the editor starts at the cursor location and replaces each space with a space and bold character. If you are still in word wrap, this may push some characters (even if they are only spaces) to the next line. After the original line, the editor will look at the next line - which may be only spaces! As each of these spaces is converted, space characters will be pushed to the next line - creating another line to be checked ... and so on ... After five or ten minutes, your text buffer will be full of ' @ @ @ @ @'. You will end up with the message to save or purge (because the buffer is full). Don't bother to do either. Press E for edit, then return to the last line of real text using FCTN 6. Delete from the first line of ' @ @ @ @ @'s to the end of the file - FCTN 9, D (enter), (first line # of @ @ @ @ @'s) E (for end) and enter.

WARNING: be careful when using REPLACE STRING with space characters!

EQUIPMENT FOR SALE!!!!

CONTACT OWNERS FOR DETAILS

PAUL STELLO RRA BOX 148 WEST BRATTLEBORO, VT 05301 (802) 257-1758

EDWARD R. AHO PO BOX 145 - TEMPLE ROAD NEW IPSWICH, NH 03071 (603) 878-3098

RALPH KING RFD6 - COTTON HILL ROAD LACONIA, NH 03246 (603) 528-3188

RICHARD V. MORIN 32 CROWN STREET NASHUA, NH 03060 (603) 883-8756

IBM CLONE HELENE LABONVILLE 121 CAMELOT DRIVE RFD5 BEDFORD NH 03102

(603) 472-3369

HEADS IT'S MYARC, TAILS IT'S TURBO

Curtis Alan Provance
New Hampshire 99er's Users Group

There are numerous options available for TI owners these days. I would like to focus on two which have received much visibility of late. The first is the MYARC 9640 computer which is truly an upgrade to the TI system. The second is the TRITON (read that Millers Graphics) TURBO XT which is not. If you think I am prejudice toward the MYARC, you're wrong! I simply am a stickler for clarity in communication. I would like to discuss both systems briefly. As with the majority of this newsletter, the opinions expressed here are solely my own and not those of the club. I also don't claim to have any better information than anyone else. All I know is that a lot of people ask me questions at meetings, so somebody thinks I know what I'm talking about!

First the MYARC:

This is the GENEVE you have heard about for over a year. Yes, it has had production problems and has been 'available' for so long that I almost believe the expression 'vaporware' was coined based on this machine alone. The 9640 (as GENEVE has come to be called) is truly an upgrade. What I mean is that your programs (most) will run on this machine. It will also run your cartridges, after you have saved the cartridge contents to disk with software provided by MYARC. There are some programs which in their original form would not work on the 9640. These include FASTTERM and FUNNELWEB. These don't work because they don't always use the console's built in key scanning routine. For example, in FASTTERM there is a three key sequence to send a file (FCTN SHIFT X). In FUNNELWEB, you page through the disk catalog with CTRL and SHIFT (nothing else). Anyway, FASTTERM has been rewritten to work with the 9640; I'm sure that FUNNELWEB could be patched, too.

The 9640 is also an upgrade in that it doesn't require a massive system expenditure. You still use your PEB, drives, controller, RS232 (and of course PIO). Contrary to what I've read in other newsletters, these cards do not have to be MYARC cards. This info is from the people who are developing the 9640 - I trust they are telling the truth.

The 9640 is also an upgrade in that it provides fantastic speed and graphics. Don't worry, your TI games won't go into warp drive, because the interrupts which control sprite motion are still at 60 hertz. The memory is great, with half a megabyte on the board and the capability of a total of over 1 megabyte of CPU. PLUS! The video can drive an analog RGB to produce graphics that rival Amiga's or those of the Atari ST.

Are there dark clouds inside this silver lining? Yes and no. The operating system is a DOS look alike. DOS has become synonymous with the IBM. DOS is the collection of machine code which allows users to load

and run programs as well as other computer peripheral interactions. The TI 'DOS' is mostly contained in the console and is limited (as you know). Anyway, the 9640's DOS is not MS-DOS. If you don't know MS-DOS, this isn't a problem. If you do, there may be some slight variations between the two to keep straight.

Will the 9640 run software written for the IBM? Yes and no. Many fairware programs written for the IBM are stored in executable format, i.e. the machine language of the INTEL 8088 microprocessor (or upgrade). However, the source code for many of these programs (which is almost always available from the author) is in C, or PASCAL, or even IBM BASIC (which the 9640 claims it can run directly). The source code can be recompiled to work with the 9640's TMS9995 microprocessor.

What about the TURBO XT? This is not an upgrade. This is an IBM clone which uses the TI as a keyboard. You still have to supply your own monitor, though. You can't run your TI programs through the TURBO. You can run your TI while the TURBO is running, though. A junction box which attaches to the side of the TI allows you to switch between your old TI setup (if you had one) and the TURBO. Is the TURBO IBM compatible? The makers claim it is. Is it a good deal? NO! The TURBO costs around \$500 and comes with a single DS/DD drive, 256K RAM (upgradable to 640 for a cost) and a color graphics card. With the exception of the color graphics card, this is as 'bare-bones' a system as you can get. Please note that to run many of the popular IBM software requires more memory than the TURBO comes with. Any issue of Computer Shopper is full of clone deals which have this beat hands down (with a much nicer keyboards, besides).

What am I going to do? Someday I will buy an IBM clone. Please note that I said 'buy' not 'switch to'. Since I don't have any money anyway, I will wait for the release of the 9640 and examine a production line version before I cast judgement on it. In the meantime, I will continue to develop my TI either through home brew modifications or some of the excellent fairware available to us. Please read the accompanying article about the power of the TI found elsewhere in this issue.

e
e THE NEW ENGLAND TI99 COMPUTER FAIRE e
e

APRIL 4, 1987 - 10am to 6pm
WALTHAM HIGH SCHOOL, WALTHAM, MASS.
From Rte.128/95 just N of Mass. Pike, take Exit 27A
East on Totten Pond Road Turn L at end, follow 99 signs
to WHS.
SEE NEW MYARC, TRITON COMPUTERS

CONTROL CODES FOR THE MORE POPULAR PRINTERS

	GEMINI 10X	SG-10 STAR MODE	SG-10 IBM MODE	TI IMPACT	MX-80	FX-80	STAR LU1210	STAR NP-10	CNIRNICS GLP	PANASONIC 1091
ITALICS	27 52	27 52	27 73 1		27 52	27 52		27 52		27 52
ELITE	27 66 2	27 66 2	27 77			27 77		27 77		27 77
CONDENSED	27 15	27 15	27 15	27 15	27 15	27 15	27 15	27 15	27 15	27 15
PICA	27 66 1	27 66 1	27 80				27 80	27 80		27 80
ENLARGED	27 87 1	27 87 1	27 87 1	27 14	27 87 1	27 87 1	27 87 1	27 87 1	27 87 1	27 87 1
SUPERSCRIPIT	27 83 0	27 83 0	27 83 0		27 83 0	27 83 0	27 83 0	27 83 0	27 83 0	27 83 0
SUBSCRIPT	27 83 1	27 83 1	27 83 1		27 83 1	27 83 1	27 83 1	27 83 1	27 83 1	27 83 1
NLQ		27 66 4	27 52			27 120 1	27 52	27 120 1	27 120 1	27 110
EMPHASIZED	27 69	27 69	27 69	27 69	27 69	27 69	27 69	27 69	27 69	27 69
UNDERLINE	27 45 1	27 45 1	27 45 1			27 45 1	27 45 1	27 45 1	27 45 1	27 45 1
DOUBLE STRIKE	27 71	27 71	27 71	27 71	27 71	27 71	27 71	27 71	27 71	27 71
SLASHED ZERO		27 92 1	27 92 1					27 126 1		
1/8 LINE SPACE	27 48	27 48	27 48	27 48	27 48	27 48	27 48	27 48	27 48	27 48
1/6 LINE SPACE	27 50	27 50		27 50	27 50	27 50	27 50	27 50		27 50
7/72 LINE SPACE	27 49	27 49	27 49	27 49	27 49	27 49	27 49	27 49	27 49	27 49
n/72 LINE SPACE	27 65 n	27 65 n	27 65 n 27 50	27 65 n	27 65 n	27 65 n	27 65 n	27 65 n	27 65 n 27 50	27 65 n
n/144 LINE SPACE	27 51 n	27 51 n								
n/216 LINE SPACE			27 51 n		27 51 n	27 51 n	27 51 n	27 51 n	27 51 n	27 51 n
TOP MARGIN	27 82 n	27 82 n	27 114 n					27 114 n		
BOTTOM MARGIN	27 78 n	27 78 n	27 78 n	27 78 n		27 78 n	27 78 n	27 78 n		
LEFT MARGIN	27 77 n	27 77 n	27 108n			27 108 n		27 108 n		
RIGHT MARGIN	27 81 n	27 81 n	27 81 n			27 81 n		27 81 n		
COLUMN WIDTH				27 81 n	27 81 n					
PAGE LENGTH	27 67 n	27 67 n	27 67 n	27 67 n	27 67 n	27 67 n	27 67 n	27 67 n	27 67 n	27 67 n
PAGE LENGTH IN.	27 67 0 n	27 67 0 n	27 67 0 n		27 67 0 n	27 67 0 n	27 67 0 n	27 67 0 n	27 67 0 n	27 67 0 n
PAPER OUT OFF	27 56	27 56	27 56	27 56	27 56	27 56	27 56	27 56	27 56	27 56
PROPORTIONAL		27 112 1	27 112 1			27 112 1		27 112 1		27 111
RESET PRINTER	27 64	27 64	27 64		27 64	27 64	27 64	27 64		27 64



NEW FROM MONTY SCHMIDT: **GPL LINKER V1.1 Run Time Version**

GPL Linker is an ingenious program that places the power of Graphics Language Programming (GPL) at your command. No extra hardware is required beyond standard 32k and disk system. In short, Linker creates runnable program files from compressed (or uncompressed) GPL Assembler object files. You can then run these programs with "Option 5 Run Program Files" of the Editor Assembler Module.

Up to 24k GPL programs can be developed and run on standard 32k systems. included in the run time version are two demonstration programs and "CONVERT," a public domain conversion program that converts MS BASIC statements to TI BASIC statements. Price: \$21.00 CDN funds \$15.00 US funds.

now \$49.95
w/Linker \$59.95
plus Intern \$69.95
add \$3 shipping

GPL price Reduction

ENHANCED GPL Assembler V2.1 NOW with high memory loader package

UNLOCK ALL THE SECRETS! New GPL Assembler Version 2.1 available exclusively through RYTE Data.

This program provides the power to write, edit and assemble true GPL programs for the TI 99/4A. Create code that accesses console operating system routines directly. Develop programs that use the GPL Interpreter and all the features of the TI 99/4A.

This package includes the GPL Assembler disk, printed documentation, GPL tips and hints, update support service and commented GROM/ROM listings (with the book "INTERN"). An example for a command module type GPL program is included with source, object and list files on disk.

Requires: 32k memory, disk drive(s), TI Editor Assembler package. Printer/RS-232 recommended.

R/D COMPUTING Technical Newsletter

with Bill Gronos on assembly!

We have a vision. Our vision is one of continued TI 99/4A support. We're dedicated to the power of the machine. From the novice to the experienced computer user; for management, home, education, entertainment or advanced applications our publication "R/D COMPUTING" is for you. TI never revealed all the important inner workings of the 99/4A. We bring you this vital information every month.

A major feature of R/D COMPUTING is the regular "upgrade projects." These electronic construction projects are designed to give the 99/4A owners more features and improvements. For example, it is possible to increase the speed of your computer with a very simple part and switch. Each month we present new circuits, diagrams and projects for your computer.

From the moment your new subscription arrives at your home, you will have access to critical technical information that makes your computer more valuable, powerful and versatile.

We believe that the TI 99/4A deserves new products, innovative hardware, software, information and a dedicated technical publication. This is what makes a computer "viable" in the fast paced microcomputer industry. Now that the 99/4A has been "opened up," all the secret information is available. You can have all these benefits and more each month. **SUBSCRIBE NOW!**

\$14/year - back issues 3-15 available

THANK YOU! Our business has grown 300% this year. To show our appreciation we are giving away hundreds of dollars in TI products to 99/4A owners or users groups. To enter drawing, (no purchase necessary) send your name and address on a postcard to RYTE Data. For subscribers to R/D Computing we are giving away XBII plus, 32k memories, GPL Assembler package, etc. Enter your subscription today!

Prices listed in U.S. funds.

New catalogue available.



Designed for the CorComp Clock Peripheral—Triple Tech Card or Stand-alone models. This utility package provides more functions for use in your Extended Basic programs. Direct access to the clock ROM at assembly speed gives you these features: three independent timers to set and read; alarm function; two interrupt routines to display time and date on screen with CTRL T—continuously or on your

command; all time and date displays are in 12 or 24 hour format using TEXT. This program also allows the week, date and time to be set independently rather than all together.

Program disk is not copy protected to allow you full use in your Extended Basic programs. Package includes disk and instructions. Only \$17.95 plus \$2

XBII plus

As reviewed in Micropendium October 1985. This command module gives you all the features of Extended Basic PLUS 40 new commands.

Totally compatible with TI's XB, this enhanced version gives your programs more power to access your 99/4A. Commands such as MLOAD, MSAVE, VPEEK, VPOKE, GPEEK are superior to most other Basic environments. Various demo programs and new applications using high resolution graphics make this module a "must" for Extended Basic users. Comes complete with a 95 page manual. Requires console and 32k. \$75.00 (US) plus \$2 shipping.

BASIC V1.1 Compiler

New Basic Compiler that is finally easy to use! Supports virtually all Basic and Extended Basic commands in

existing programs. Simply load and compile programs from a menu driven directory on your screen. No extensive re-writing, variable declarations or conversions are required. Compiler produces code-list in one pass containing all variable addresses and jump list. Package includes Extended Basic Loader, Floating Point Loader, Integer Loader, Disk Menu program and DSR program for the Compiler support. This Compiler cannot unravel DEF statements and stops on the END statement—no SUB's allowed. TRACE, BREAK, ON ERROR, CALL LOAD and CALL LINK may produce execution errors. Requires 32k, disk. Price: \$20.00 plus \$2 shipping (US funds).

Ryte Data (705) 457-2774

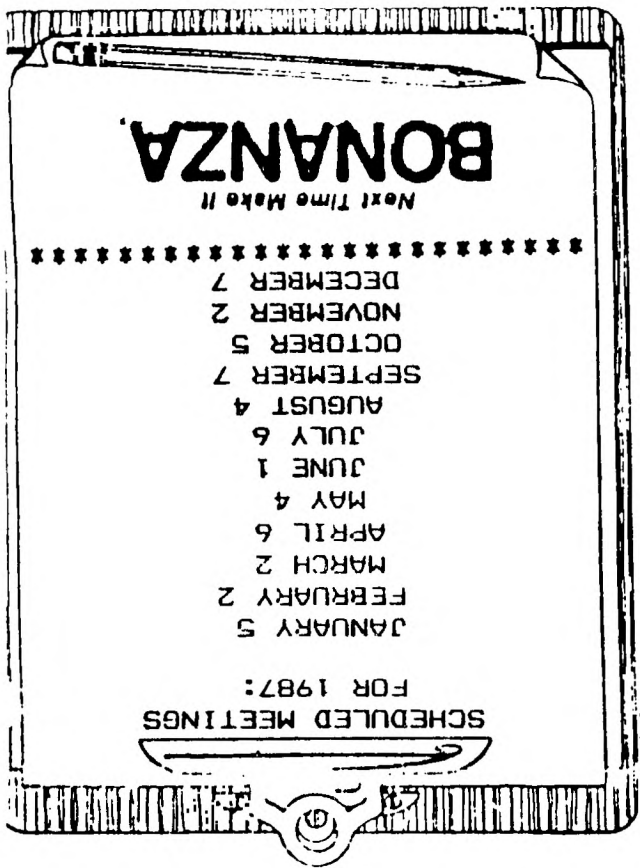


MILLENNIUM COMPUTERS
210 MOUNTAIN STREET,
HALIBURTON, ONTARIO K0M 1S0
TELEX 06-986766 TOR. ATTN: RYTE DA

BONANZA
Next Time Make It

DECEMBER 7
NOVEMBER 2
OCTOBER 5
SEPTEMBER 7
AUGUST 4
JULY 6
JUNE 1
MAY 4
APRIL 6
MARCH 2
FEBRUARY 2
JANUARY 5

SCHEDULED MEETINGS
FOR 1987:



456 Beech St. Manchester, NH
OBIS
Co. (603) 668-4245

Your Computer Forms & Supplies Headquarters
ALL SIZES AVAILABLE — MOST IN STOCK

2500	BX	21.60
1000	BX	11.30
250	BX	3.25

20* BLANK EASI-PERF
9 1/2 x 11

—Free Local Delivery—
COMPUTER PAPER



NEW HAMPSHIRE 99'ERS USER GROUP, INC.

PO BOX 5991
MANCHESTER, NH 03108-5991



EDMONTON USER'S GROUP 01/C
PO BOX 11983
EDMONTON, ALBERTA
T5J-3L1

**PLEASE SUPPORT BONANZA
- THEY SUPPORT US!**