EDMONTON
99'er
COMPUTER
USERS'
SOCIETY

The Edmonton 99'er Computer Users' Society
p.o. box 11983
Edmonton, Alberta
CANADA T5J 3L1

99'er ON LINE... is the news letter of the Edmonton 99'er Computer User's Society published ten times a year. Unless otherwise stated, all articles may be republished in other news letters provided that source and author are identified. We will credit authors quoted in 99'er ON LINE.

CORRESPONDENCE: Newsletter editor: Bob Pass, 59 Labelle Cr, St. Albert, Alberta, Canada T8N-2G6. (403)458-7658. All other correspondence should be sent to the address at left.

OFFICERS: president Tom Hall, vice-pres Ken Godbeer, treasurer Jim Mulligan, secretary Roxanne Appelt

DISCLAIMER: Information published in this newsletter is created by and for amateurs; therefore, we cannot guarantee the accuracy or use of presented information.

REGULAR MEETINGS... of the Edmonton 99'er Computer User's Society are held on the second Tuesday of each month in room 849 of the General Services building of the University of Alberta from 7:00 till 10:00 PM and are open to all members in good standing. Non-members may attend their first meeting free of charge.

ADVERTIZING... Commercial space is available in this news letter at the following rates: full page $20.00, half page $15.00, 1/4 page $10.00. Discuss your needs with Jim Mulligan at 467-6021, at the next meeting, or send "photo ready" copy to the P/O Box above. Members may advertise their personal computer related items for free but are asked to limit their ads to about 50 words. Mail your ads to the editor's address or hand it to him at the general meeting; newsletter deadline 3'rd Monday of the month.

MEMBERSHIP FEES: Family 12 months, $20.00, 6 months, $15.00. Students 12 months, $15.00, 6 months, $10.00. New member initiation, $20.00.

DECEMBER MEETING

Minutes of November executive meeting were adopted as read.

CLUBLINE/99 -- The december issue had not arrived as of the meeting date so there was no distribution made. If there are more people interested in participating in this offer, please see Bob Pass or Jim Mulligan.

Tom Hall gave a good demonstration of a new operating system for the ramdisk. This system presents you with a menu of programs to run at a key stroke and adds some more "CALL" functions to the system. It is freeware and should be available in our library soon.

Tom completed his presentation on TI-BBS and gave a complete demo of the Horizon Ramdisk. The latter was in conjunction with a video tape that was shown featuring John Clulow speaking about the development of the ramdisk at the Ottawa TI Fair last spring.

Attendance at the meeting was 31 people, a fairly good showing over the last two meetings. Undoubtedly some were enticed by the home baking supplied by Lois Meunier and Roxanne Appelt and also by the donuts supplied by Ken Godbeer. This helped add a festive spirit to our Christmas season meeting. Thank you ladies and Ken.

Tom mentioned that Miller Graphics will be discontinuing production of the GramKracker in the early new year. If you want one, you are advised to phone Craig Miller right away as future production is in doubt. Something to spend Aunt Harriet's Christmas cheque on!

NEXT MEETING

The next meeting will be Tuesday, January 13'th at 7:15 PM. Same place as usual; General Services Building, U of A campus in room 849.

Featured this month will be the next section of the video tape of the Ottawa TI fair held last spring. Sorry, I cannot recall what the subject matter is on the next section.

Due to some crossed signals, the door prizes promised for the December meeting did not materialize. They will be there for the January meeting. If you have any suitable items that could be used for a door prize, bring it along. We will be glad to give it away for you!

Planned for the January meeting is a tutorial on MULTIPLAN by Bob Chapman. If you own that package and do nothing with it, circle that January meeting date and make that the day open up a new world in computing.

NEWSLETTER DISKETTE AWARDS

This month's issue is comprized almost entirely from files downloaded from our bulletin board, TI-BBS which is operated by Tom Hall. Therefore, this month's newsletter disk award goes to Tom. But I would also like to thank those people who took the time to leave their messages on the board. In doing so, they have helped spread the latest news in the TI world and helped keep our machines from becomming true orphans.

If you have an item that you would specifically like to be published here and you cannot get access to our newsletter disk system, simply leave a message or file on the board to my attention and I will see that you too get an award disk once your item is published. Thanks to all of you for supporting this newsletter.

Your Fearless Leader Returns!

Some of you have probably been wondering what happened to that guy you elected President way back in June. Well, he's been pretty busy, and unfortunately not all of the busy-ness has been with the TI!!! However, as we launch into 1987, perhaps we should take a look at a few things and try to get some perspective on where the TI community is headed in the new year.

Perhaps the most significant news is the advent of the long-awaited 9940 Computer from Myarc -- or "Geneve", as it's affectionately called in the community. Since making it's official premiere in Chicago in November, the computer seems to finally be in production, and we may see the first one in Edmonton very early in the new year! The retail price of the new computer is said to be around $450 U.S., and includes a card for the expansion box and a special IBM-type keyboard. The keyboard actually contains 640K of RAM, and most of the power of the new system is said to reside there. Eighty-column versions of TI WRITER and MULTIPLAN are said to be included. Sorry, but I haven't heard anything about a monitor; guess that's an extra.

The most critical need of our group continues to be information. Your loyal prez and sysop has been rapidly shoveling himself into the poorhouse with his recently-acquired subscription to COMPUSERVE, but there have been some fantastic dividends in terms of information, sources of yet more information, and some exciting new software, which has all but revolutionized our bbs. Barry Traver's ARCHIVER program (permanently residing on the club bbs) now ensures that we get complete packages of software, since the program allows you to "pack" any number of files into one file, with one directory listing, and then later "unpack" that file, and get back all the files that originally went into the "packed" file, in their original condition!

Indirectly, this change in the bbs is an encouragement for 300 baud users to upgrade to 1200 baud, since it's now quite common to see a file larger than 100 sectors on the board. Ask our secretary's husband how long it takes to upload 273 sectors at 300 baud! Fortunately, upgrading to 1200 baud is no longer quite as painful as it once was, since there are now several Hayes-compatible or near-Hayes-compatible 1200 baud modems on the market, at less than $300 CDN.

Other hardware innovations of 1986 were the Gram Kracker, from MG (formerly called Millers Graphics), and the Horizon RAM disk from Horizon Computers of Toledo, Ohio, both of which have greatly improved the operation of our bulletin board. With the advent of the Geneve, the future is indeed looking bright for the TI, as there is now the very real possibility of IBM compatibility for our tenacious "orphan". Whatever you do, KEEP YOUR TI! You'll be glad you did!

The year does end, however, on a slightly ominous note, as there are rumblings from the U.S. that the Federal Communications Commission is considering legislation which would effectively eliminate *ALL* free local telephone access to information databases such as Compuserve, The Source, GEnie, and PC Pursuit. This legislation, if passed, could set a dangerous precedent and threaten the future of low-cost modeming in Canada as well. Anyone interested in more information on this topic should contact me either through the bbs or at the next meeting.

In closing, I'd like to once again remind all of you that there are considerable untapped resources in our club; we have innovators both in software and hardware. If hardware is your passion, catch Yves Chevalier at a meeting and ask him about his home-made modem! Or ask Michal Jaegermann about FORTH or the "C" language. Don't be afraid to ask questions! That's how we learn things!

On behalf of the club executive I'd like to take this opportunity to wish you all a happy and prosperous new year --- and KEEP ON TI-ing!!!!

SELECTED ITEMS FROM TI-BBS

by: Bob Pass

The following items were gleaned from our bulletin board. As they are of general interest, I thought that most of you would enjoy them and perhaps encourage you to spend more time on the board.

From: DON BILYEA
Chess challenge. To anyone willing to accept the challenge of a chess game I issue the first move of P-K4. Please respond through E-mail. I have more than one set prepared so more than one of you can reply. Hoping for a response soon. Thank you, Don.

The challenge was issued in August but but go ahead and contact Don by leaving him a message on the board. Nothing like a good game of chess to while away wintery evenings!

From: Wolly Barabash

20% SPEED UP FOR TI. (THIS IS FROM RYTE DATA NEWSLETTER OF MAY/JUNE.)

Desolder one lead of the TI 12 MHz crystal. Solder a 6" wire in its place. Solder the other end of this wire to the middle contact of a single pole double throw switch. Solder a new wire to the desoldered end of the 12 Mhz crystal and to one of the other switch contacts. Solder one lead of a 14.318 MHz crystal to the 12 Mhz lead that is still wired to the TI motherboard, solder the other end of the 14.318 MHz crystal to the last bare lead of the switch. You should mount this outside onto the console. There the switch can be toggled back and forth between the two speed, in mid program yet so you can see the difference. I haven't done this myself yet, but if help is required, leave me a message and I will call back to this board in two weeks. See you later, Wolly

From: James McBride
TMS9900/Z80 ASSEMBLER. I woulD like to obtain a program that will assemble Z80 code on the TI and produce an Intel hex object code file. Please leave a message for me on this board if you have any information on such a program.

From: Tony Van Muyden
FORTH MEETINGS. For those of you interested in FORTH there is a way to meet with other enthusiasts. Meeting place ROOM A134 at NAIT on the following dates: Nov 22, Dec 20, Jan 17, Feb 14, Mar 14 Apr 11, May 09. Mark these dates on your calender in pencil as they are to be confirmed on the next meeting. Time of meeting will be from 14:00 till 16:00. We also need help from the members in the organisation. If you are a FIG member bring your membership card so we can add you to the data-base. Everybody is welcome and see you there.

From: Danny Ochitwa
TI SYSTEM FOR SALE. TI Console, Peripheral Expansion box including: TI Disk Controller, TI RS232 and Full-height SS/SD Drive, Stand alone 32K Memory Expansion, Navarone Widget, Extended BASIC Module, Disk Manager 2 Module, Editor/Assembler Module (with 8K RAM added), Black&White TV, Miscellaneous Game Modules. Asking $500 or best offer. Call Danny Ochitwa 482-1702 after 6pm

From: Rob Regular
FOR SALE. I have for sale 1 TI-99/4A home computer, expansion box, disk drive, joysticks, assembly lan., extended basic, munch man, Zork III (disk), memory expansion. Good condition. For more info, phone Rob 467-7994 4-11 pm.

From: Keith Nickerson
WANTED. A friend of mine is looking for a stand alone parallel printer interface for his TI. Anyone know of one for sale please leave me message here.

From: Keith Nickerson
FOR SALE. I have for sale a Signalman 300 Baud modem for the TI. New cost was $150, sell for $75 or best offer. Works great! Leave message if interested.

From: Earl Toman
GRAPHX. I have recently been musing about the possibility of buying a Graphx package ie)Printshoppe package as per

Texcomp. Have any of you readers used this program? If so, drop me a line with your comments as to whether it is worth the $50.00 investment.

Also I would like to get a different printer to use the above package. I have considered Panasonic or Roland. Would these printers work with Graphx?

From: Jim Beck
ANYBODY want to sell me a used extended basic? I need a spare.

From: JIM MULLIGAN
AGRICULTURE DATABASE. I'm looking for a good agriculture database, primarily for livestock records. If anyone has one for sale or knows of a good one (freeware is also acceptable), please leave a message on this board.

From: KIM SPENCER
PRINTER FOR SALE. A friend has for sale an Extel B208L printer. It prints at 300 baud and is capable of printing all control characters. It is equipped with a keyboard so it can be used as a terminal. He wants $200.00 obo. Call him at 723 3555 after 6 PM; ask for Allan or leave a message for me (Kim Spencer) on this board.

From: Garry Salisbury. GENIEVE is now available from Myarc. It is $409 U.S. My sources have talked to one owner he seemed satisfied. More to come as it arrives.

The Companion word processor would not work with the Myarc disk controller. There is a fix for it from a fellow in Vancouver. I apologize but I can't remember his name. The problem is on the "B" file. Use the search function of DISK+AID disk fixer to find the following string in "B" (You should save "B" to a blank, newly initialized disk for the easiest searc): hex C016 2B06 0215 FA02. I think it's the last sector on the file. Change 2B06 to 1000. Now it will work.

FOR SALE. I have an extra 12 voice music synthesizer for sale. Includes documentation, disks and speaker wire to hook up to your amplifyer. Plays stereo or quad. Uses Forth to program music; ideal for someone with a knowledge of music. Write to GARRY SALISBURY BOX 1971 R.R.1 CLEARWATER, B.C. VOE 1NO or call 604-674-2326 after 6 pm.

PEEKS AND POKES

This file was downloaded from The TI Forum section of Compuserve. It is the most extensive list of peeks and pokes I've seen for the TI.

24K OF DATA STORAGE
If you need to work with quite a bit of data or would like to change programs, but save the data after you press CALL QUIT then you can set up the 24K of High-Memory in the PEB as a single data file called "EXPMEM2". You open this file just as you would a disk file with one exception - you must precede the OPEN statement with a CALL LOAD to the location -24574 as follows:

For INT/VAR files - 24
For DIS/VAR files - 16
For INT/FIX files - 8
For DIS/FIX files - 0

For example: if you want to open up the Expansion Memory for Display, Variable 80 files this is what you'd do:

```
100 CALL INIT
110 CALL LOAD(-24574,16)
120 OPEN #1:"EXPMEM2",RELATIVE,
    UPDATE,DISPLAY,VARIABLE 80
```

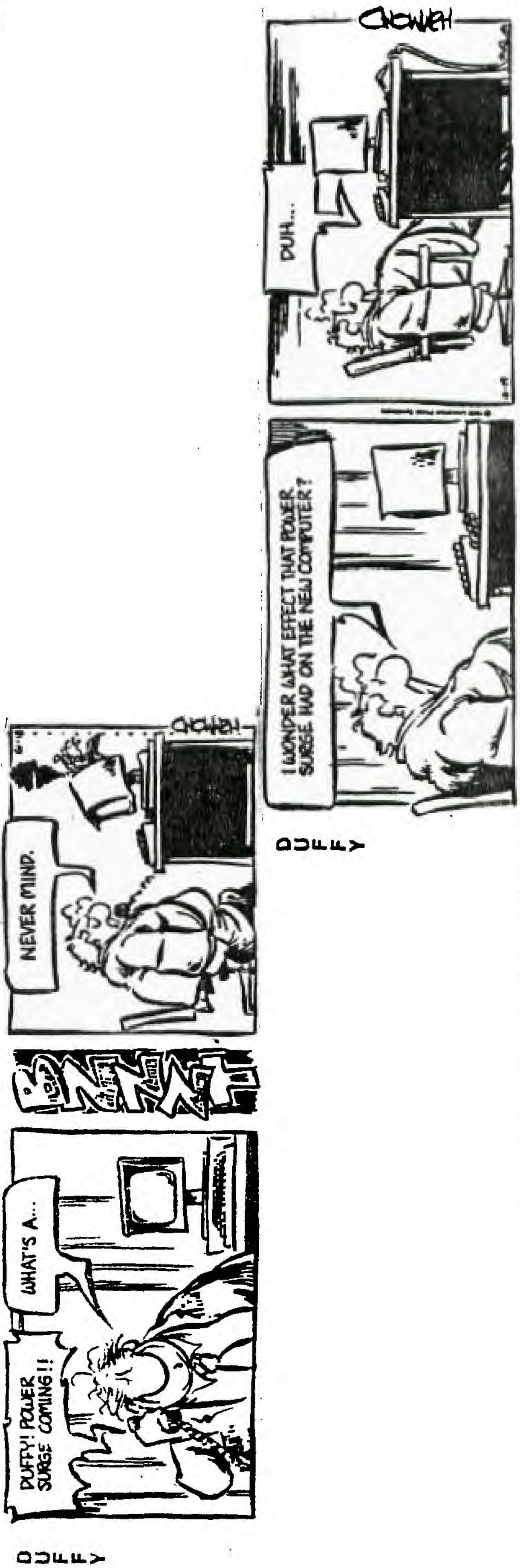
Then continue on as you normally would. If you want to store both data and assembly language routines at the same time do this:

```
100 CALL INIT
110 CALL LOAD(-24574,-16)
120 OPEN #1:"EXPMEM2"
130 CALL LOAD ("DSK1.ASSM1")
140 CALL LOAD ("DSK2.ASSM2")
150 CALL LINK ("START")
160 REM REST OF PROGRAM
```

In the above example the 24 K of high memory was saved for use as a DATA file (DIS/VAR 80 format) then the assembly routines were loaded. The computer will look for the best place to put the routines and will adjust the pointer accordingly. After the routines are loaded, a LINK statement starts the first routine and off we go. If that's not enough for you, you can also use the MINI-MEMORY for 4K more of storage of assembly routines! Now that's 16K of program space, 12K of assembly routine space!

THESE ARE ALL OF THE PEEKS & POKES THAT I HAVE COME ACROSS FOR USE WITH X-BASIC AND 32K MEMORY EXPANSION (BE SURE TO DO A "CALL INIT"). THE P & Q VARIABLES ARE USED FOR "PEEK" - THE NUMBERS ARE FOR "POKE" OR "LOAD". IF YOU KNOW OF ANY OTHERS PLEASE LET ME KNOW AND I WILL ADD THEM IN.

ADDRESS	VALUE(S)	MEANING IN EXTENDED BASIC
		CALL VERSION(X) IF X=100 100= NEWEST VERSION OF X/B CART
8192	P	USE (PEEK,P) IF P<> 70 OR <>121 THEN DO A CALL INIT
8194		FIRST FREE ADDRESS IN LOW MEMORY
8196		LAST FREE ADDRESS IN LOW MEMORY
-28672	P	P=0 SPEECH NOT ATTACHED P=96 OR P=255 SPEECH IS ATTACHED
-31572	0 TO 255	VARY KEYBOARD RESPONSE
-31740	P , Q	PUT IN DIFFERENT TO CHANGE BEEPS,WARNINGS, ETC
-31744	0 TO 15	CONTINUATION OF LAST SOUND (0=LOUD AND 15=SOFT)
-31745	0	PRODUCES A FROZEN SCREEN. THEN BLANKS. RESTORE WITH FTCN(-)
-31748	0 TO 255	CHANGE THE CURSOR FLASHING AND RESPONSE TONE RATES
-31788	160	BLANK OUT THE SCREEN (MUST PUSH A KEY TO ACTIVATE)
	192	NO AUTOMATIC SPRITE MOTION OR SOUND
	224	NORMAL OPERATION
	225	MAGNIFIED SPRITES
	226	DOUBLE SIZE SPRITES
	227	MAGNIFIED & DOUBLE SIZED SPRITES
	232	MULTICOLOR MODE (48 BY 64 SQUARES)
-31794	P	TIMER FOR CALL SOUND (COUNTS FROM 255 TO 0)
-31804	X , Y	RETURN TO THE TITLE SCREEN (USE "PEEK (2,X,Y)")
	P	CHANGE THE CURSOR FLASH RATE (0 TO 255)
-31806	0	NORMAL OPERATION
	16	DISABLE QUIT KEY (FCTN =)
	32	DISABLE SOUND (USE NEG DUR FOR CONTINOUS SOUND)
	48	DISABLE SOUND & QUIT KEY
	64	DISABLE AUTO SPRITE MOTION
	80	DISABLE SPRITES & QUIT KEY
	96	DISABLE SPRITES AND SOUND
	128	DISABLE ALL THREE
-31808	P , Q	DOUBLE RANDOM NUMBERS (0 TO 255) NEED "RANDOMIZE"
-31860	4	GO FROM EX-BASIC TO CONSOLE BASIC (NEED "NEW")
	8	AUTO RUN OF DSK1.LOAD
-31866	P , Q	END OF CPU PROGRAM ADDRESS (P6+Q)
-31868	0	NO "RUN" OR "LIST" AFTER "BREAK" IS USED
	0 , 0	TURNS OFF THE 32K MEMORY EXPANSION
	255 , 231	TURNS ON THE 32K MEMORY EXPANSION
-31873	3 TO 30	SCREEN COLUMN TO START AT WITH A "PRINT"
-31877	P	P&32 = SPRITE COINCIDENCE P&64 = 5 SPRITES ON A LINE
-31878	P	HIGHEST NUMBER SPRITE IN MOTION (0 STOPS ALL)
-31879	P	TIMER FOR VDP INTERRUPTS EVERY 1/60 OF A SEC (0 TOP 255)
-31880	P	RANDOM NUMBER (0 TO 99) NEED "RANDOMIZE"
-31884	0 TO 5	CHANGE KEYBOARD MODE (LIKE "CALL KEY(K,...)")
-31888	63 , 255	DISABLE ALL DISK DRIVES (USE "NEW" TO FREE MEMORY)
	55 , 215	ENABLE ALL DISK DRIVES (USE "NEW" TO FREE DRIVES)
-31931	0	UNPROTECT X-B PROTECTION
	2	SET "ON WARNING NEXT" COMMAND
	4	SET "ON WARNING STOP" COMMAND
	14	SET "UNTRACE" COMMAND
	15	SET "UNTRACE" COMMAND & "NUM" COMMAND
	16	SET "TRACE" COMMAND
	64	SET "ON BREAK NEXT" COMMAND
	128	PROTECT X/B PROGRAM
-31952	P	PEEK P=55 THEN 32K EXPANSION MEMORY IS OFF <>55 MEANS ON
-31962	32	RETURN TO THE TITLE SCREEN
	255	RESTART X/B W/DSK1.LOAD
-31974	P , Q	END OF VDP STACK ADDRESS (P6+Q)
-32112	8	SEARCHES DISK FOR ?
-32114	2	RANDOM GARBAGE
	13	SCREEN GOES WILD
	119	PRODUCE LINES
-32116	2	RANDOM CHARACTERS ON SCREEN
	4	GO FROM X/BASIC TO BASIC
-32187	0	UNPROTECT XB PROGRAM
	2	SET "ON WARNING NEXT" COMMAND
	4	SET "ON WARNING STOP" COMMAND
	9	SET 0 LINE NUMBER


CHOWTEH
DUH...
I WONDER WHAT EFFECT THAT POWER SURGE HAD ON THE NEW COMPUTER?
NEVER MIND.
DUFFY
DUFFY! POWER SURGE COMING!!
WHAT'S A...
DUFFY

	14	SET "UNTRACE" COMMAND
	15	SET "UNTRACE" COMMAND & "NUM" COMMAND
	16	SET "TRACE" COMMAND
	64	SET "ON BREAK NEXT" COMMAND
	128	PROTECT XB PROGRAM
-32188	1	CHANGE COLOR AND RECEIVE SYNTAX ERROR
	127	CHANGE COLOR AND RECEIVE BREAKPOINT
-32572	128	DISABLES KEYBOARD
-32630	128	RESET TO TITLE SCREEN
-32699	0	UNPROTECT XB PROGRAM
	2	SET "ON WARNING NEXT" COMMAND
	4	SET "ON WARNING STOP" COMMAND
	14	SET "UNTRACE" COMMAND
	15	SET "UNTRACE" & "NUM" COMMAND
	16	SET "TRACE' COMMAND
	64	SET "ON BREAK NEXT"
	128	PROTECT XB PROGRAM
-32700	0	CLEARS CREEN FOR AN INSTANT
-32729	0	RUN "DSK1.LOAD"
-32730	32	RESET TO TITLE SCREEN
-32961	51	RESET TO TITLE SCREEN
	149	SETS "ON BREAK GOTO" LOCKS SYSTEM

THE FOLLOWING LOADS REQUIRE E/A OR MM

ADDRESS	VALUE(S)	MEANING
784	P	USE POKEV(784,P) (WHERE P IS 16 TO 31) CHANGES BACKGROUND COLOR OF CURSOR
-24574	8	I THINK THIS ALLOWS THE MINI-MEM TO USE THE 24K FOR STORAGE
-30945	0	WHITE EDGES
-32272	0	WILL PUT YOU IN TEXT MODE
-32766	0	BIT MAP MODE
-32768	0	GRAPHICS (NORMAL MODE)
-32280	0	MULTI-COLOR MODE
-32352	107	WILL BLANK THE SCREEN, ANY KEY PRESS WILL RESTORE

* PASCAL LOADS

14586 , 0 , 0 THIS ALLOWS YOU TO DO A "RUN-TIME WARM START" FROM PASCAL TO BASIC.

OH REALLY!

The following gems were written by: Sean Philip Engelson, Carnegie-Mellon University, Computer Science Department.

REAL PROGRAMMERS DON'T WRITE SPECS

Real Programmers don't write specs -- users should consider themselves lucky to get any programs at all and take what they get.

Real Programmers don't comment their code. If it was hard to write, it should be hard to understand.

Real Programmers don't write application programs; they program right down on the bare metal. Application programming is for feebs who can't do systems programming.

Real Programmers don't eat quiche. In fact, real programmers don't know how to SPELL quiche. They eat Twinkies, and Szechwan food.

Real Programmers don't write in COBOL. COBOL is for wimpy applications programmers.

Real Programmers' programs never work right the first time. But if you throw them on the machine they can be patched into working in "only a few" 30-hour debugging sessions.

Real Programmers don't write in FORTRAN. FORTRAN is for pipe stress freaks and crystallography weenies.

Real Programmers never work 9 to 5. If any real programmers are around at 9 AM, it's because they were up all night.

Real Programmers don't write in BASIC. Actually, no programmers write in BASIC, after the age of 12.

Real Programmers don't write in PL/I. PL/I is for programmers who can't decide whether to write in COBOL or FORTRAN.

Real Programmers don't play tennis, or any other sport that requires you to change clothes. Mountain climbing is OK, and real programmers wear their climbing boots to work in case a mountain should suddenly spring up in the middle of the machine room.

Real Programmers don't document. Documentation is for simps who can't read the listings or the object deck.

Real Programmers don't write in PASCAL, or BLISS, or ADA, or any of those pinko computer science languages. Strong typing is for people with weak memories.

Real programmers never make up schedules. Only planners make up schedules. Only managers read them.

Real programmers never deliver programs on schedule. Either the program is "done" in two days or it is never finished. In any case, it is never delivered when it was scheduled.

Real programmers never eat at restaurants. If the vending machine sells it they eat it. If it doesn't, they don't. Recently real programmers discovered that popcorn was being sold in vending machines. Common coders discovered that it could be popped in the microwave oven in the vending-machine room but real programmers use the heat escaping from the top of the CPU.

Real programmers never deliver programs on Wednesdays. Real programmers never deliver programs on the first day of any month.

Real programmers know that good human factors design requires only the application of common sense. Besides, no one cares about users. The program's written for aesthetic beauty.

Real programmers know every nuance of every instruction and they use them all in every program.

Real programmers do not clear registers twice before using them. In fact, if you annoy a real programmer, he/she won't clear the registers at all. And that goes for your memory too!

Real programmers do not wonder where the bits went following a shift operation. They do not care.

Real programmers are not in it for the money. Most of them are secret millionaires.

REAL COMPUTER SCIENTISTS DON'T WRITE CODE

Real computer scientists don't write code. They occasionally tinker with 'programming systems', but those are so high level that they hardly count (and rarely count accurately; precision is for applications.)

Real computer scientists don't comment their code. The identifiers are so long they can't afford the disk space.

Real computer scientists don't write the user interfaces, they merely argue over what they should look like.

Real computer scientists don't eat quiche. They shun Schezuan food since the hackers discovered it. Many real computer scientists consider eating an implementation detail. (Others break down and eat with the hackers, but only if they can have ice cream for desert.)

If it doesn't have a programming environment complete with interactive debugger, structure editor and extensive cross module type checking, real computer scientists won't be seen tinkering with it. They may have to use it to balance their checkbooks, as their own systems can't.

Real computer scientists don't program in assembler. They don't write in anything less portable than a number two pencil.

Real computer scientists don't debug programs. they dynamically modify them. This is safer, since no one has invented a way to do anything dynamic to FORTRAN, COBOL or BASIC.

Real computer scientists like C's structured constructs, but they are suspicious of it because its compiled. (Only Batch freaks and efficiency weirdos bother with compilers, they're soooo un-dynamic.)

Real computer scientists play go. They have nothing against the concept of mountain climbing, but the actual climbing is an implementation detail best left to programmers.

Real computer scientists admire ADA for its overwhelming aesthetic value, but they find it difficult to actually program in, as it is much too large to implement. Most Computer scientists don't notice this because they are still arguing over what else to add to ADA.

Real computer scientists work from 5 pm to 9 am because that's the only time they can get the 8 megabytes of main memory they need to edit specs. (Real work starts around 2 am when enough MIPS are free for their dynamic systems.) Real computer scientists find it hard to share 3081s when they are doing 'REAL' work.

Real computer scientists only write specs for languages that might run on future hardware. Nobody trusts them to write specs for anything homo sapiens will ever be able to fit on a single planet.

Real computer scientists like planning their own environments to use bit mapped graphics. Bit mapped graphics is great because no one can afford it, so their systems can be experimental.

Real computer scientists regret the existence of PL/I, PASCAL and LISP. ADA is getting there, but it is still allows people to make mistakes.

Real computer scientists love the concept of users. Users are always real impressed by the stuff computer scientists are talking about; it sure sounds better than the stuff they are being forced to use now.

Real computer scientists despise the idea of actual hardware. Hardware has limitations, software doesn't. It's a real shame that Turing machines are so poor at I/O.

Real computer scientists love conventions. No one is expected to lug a 3081 attached to a bit map screen to a convention, so no one will ever know how slow their systems run.

REAL SOFTWARE ENGINEERS DON'T READ DUMPS

Real software engineers don't read dumps. They never generate them, and on the rare occasions that they come across them, they are vaguely amused.

Real software engineers don't comment their code. The identifiers are so mnemonic they don't have to.

Real software engineers don't write applications programs, they implement algorithms. If someone has an application that the algorithm might help with, that's nice. Don't ask them to write the user interface, though.

Real software engineers eat quiche.

If it doesn't have recursive function calls, real software engineers don't program in it. Real software engineers don't program in assembler. They become queasy at the very thought.

Real software engineers don't debug programs, they verify correctness. This process doesn't necessarily involve executing anything on a computer, except perhaps a Correctness Verification Aid package.

Real software engineers like C's structured constructs, but they are suspicious of it because they have heard that it lets you get "close to the machine."

Real software engineers play tennis. In general, they don't like any sport that involves getting hot and sweaty and gross when out of range of a shower. (Thus mountain climbing is Right Out.) They will occasionally wear their tennis togs to work, but only on very sunny days.

Real software engineers admire PASCAL for its discipline and Spartan purity, but they find it difficult to actually program in. They don't tell this to their friends, because they are afraid it means that they are somehow Unworthy.

Real software engineers work from 9 to 5, because that is the way the job is described in the formal spec. Work woulfl like usg anundocumented externa l procedure.

Real software engineers write in languages that have not actually been implemented for any machine, and for which only the formal spec (in BNF) is available. This keeps them from having to take any machine dependencies into account. Machine dependencies make real software engineers very uneasy.

Real software engineers don't write in ADA, because the standards bodies have not quite decided on a formal spec yet.

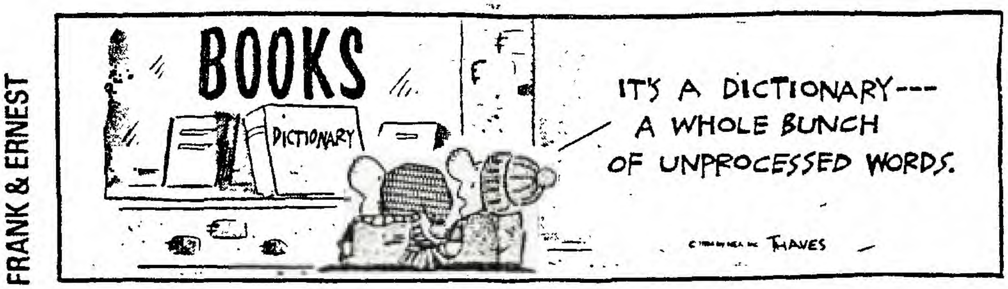
Real software engineers like writing their own compilers, preferably in PROLOG (they also like writing them in unimplemented languages, but it turns out to be difficult to actually RUN these).

Real software engineers regret the existence of COBOL, FORTRAN and BASIC. PL/1 is getting there, but it is not nearly disciplined enough; far too much built in function.

Real software engineers aren't too happy about the existence of users, either. Users always seem to have the wrong idea about what the implementation and verification of algorithms is all about.

Real software engineers don't like the idea of some inexplicable and greasy hardware several aisles away that may stop working at any moment. They have a great distrust of hardware people, and wish that systems could be virtual at ALL levels. They would like personal computers (you know no one's going to trip over something and kill your DFA in mid-transit), except that they need 8 megabytes to run their Correctness Verification Aid packages.

Real software engineers think better while playing WFF 'N' PROOF.



The Night Before Christmas, 2001

By: David A. Tarr

Twas the night before Christmas, and all through my home
Not a creature was stirring, not even my clone.

The test tubes were hung by the burner with care,
In hopes that Saint Nicholas soon would be there.

The androids were nestled all snug in their beds,
While visions of mc**2 danced in their heads.

My wife in her jumpsuit, and I in my vest,
Had just settled down to some drug-induced rest.

When, out by the labs, there arose such a clatter,
My bed woke me up to see what was the matter.

Away to the window I hastened my mass,
Tore open the blast shields, and threw up the glass.

The refraction of moonlight through smog-ridden air
Gave a luster of midday to everything there.

When what to my bionic eyes should appear
But a mass-driven sleigh with some strange landing gear,

With a quick little pilot, a company man,
Who did what was asked and followed the plan.

More rapid than phantoms his coursers they came.
He impulsed his crew, then called them by name.

"Now, Redox! Now, Hewlett! Now, Quasar and Photon!
"On, Laser! On, Xerox! On, Pulsar and Proton!

"To the top of the dome, by the air-intake vent.
"Now dash away quickly before our fuel's spent."

So, up to the air vent his coursers they flew,
With a craft full of toys and Saint Nicholas, too.

And then, in a flash, on the dome I did hear
The scratching and scraping of stout landing gear.

I steadied my blaster, my chest to the ground,
And then, through the air vent, he came with a bound.

He was dressed in a three-piece he'd rented near here.
(Why purchase an outfit you wear once a year?)

A life-support system he wore on his back,
While the toys for the 'droids he took out of his pack.

A bottle of synthroid he held in his hand
(He was quite overweight from a poor thyroid gland).

He brought out the toys that department stores sell;
The elves at the Pole could not make them as well.

He checked with the base ship, while doing his work,
And filled all the test tubes, then turned with a jerk.

His anti-grav belt was secure, I suppose,
And, pressing the keys, up the air vent he rose.

He sprang to his craft, to the crew gave a shout;
The ship heaved a shudder, then blasted them out.

But I heard him exclaim, as he flew out of sight,
"Merry Christmas to all, and to all a good flight."

FLOPPY DRIVES

Problem:
Some floppy disk drives will write on floppy diskettes with black write protect tabs.

Solution:
Use only silver write protect tabs. Some floppy drives depend on reflected light to determine if a write protect tab is present.

"But you distinctly said to dump the files."

Compact disc could store encyclopedia

Computers

By Ron Chalmers

The compact disc (CD), which already is better than the long-playing record for storing music, could add massive memory to microcomputers.

It will come soon, but only after somebody finds a friendly use.

The compact disc, also known as the optical disc, is "written" or "recorded" upon with a laser beam that burns tiny pits into the subsurface under a transparent covering that protects it from physical abuse. The pits — or absence of pits — comprise digital messages that can be "read" or "played" by another laser beam.

CDs are increasingly popular for music because they store information digitally, with high accuracy, and don't deteriorate. Neither stylus nor dirty fingers ever touch the subsurface.

Now, computer engineers have found that a single CD stores 540 million bytes of information, as much as 1,500 floppy discs. The CD is cheap ($10 to $20) and small (4.72 inches in diameter).

But here's the rub: while a CD *reader* costs under $300, a CD *writer* costs tens of thousands of dollars.

And the CD, once written upon, can't be erased or edited.

In other words, it only provides Read-Only Memory (ROM). But it does give lots of it.

The obvious application is storing major reference works: the entire Encyclopedia Britannica would occupy less than a single CD.

A micro with a stack of CDs could save shelf space at a library or law office.

But most users don't have many reference works, so where's the big payoff?

One exciting prospect is that the CD could change the way we learn.

This argument is made by Dr. Gabriel Ofiesh, Emeritus Professor of Educational Technology at Howard University, Washington, D.C.

In a recent article he says that a great store of information on CD could be much more than just an on-screen encyclopedia with an electronic page-turner *if the memory is linked to fast-switching hypertext software.*

This combination would escape the linear style that a book imposes on your parallel-thinking mind.

You seldom think in a straight line. Even when you do, your straight line is different from my straight line. Yet books are designed for thinking along a given straight line. That's fine for a novel, where you want to follow the author, but not for non-fiction where you want to follow your curiosity.

With CD storage and fast-switching hypertext software, Ofiesh suggests, "one could hop from article to article, following a given idea."

You wouldn't even need to know what articles were stored on the disc; you'd just specify a keyword and the software would search out relevant articles. You'd choose one, read as far as you wished, then take any tangent you wanted simply by specifying a new keyword.

As usual with computer hardware, CD ROM's possibilities exceed the ability of existing software. Borland Turbo Lightning — an online dictionary with other online reference works promised — probably is the closest of any micro-based software to realizing such potential.

ON THE FASTRACK

MOM, I KNOW THAT WITH YOUR [illegible] YOU DON'T ALWAYS [illegible] TIME TO CLEAN HOUSE!
...BUT IT DOESN'T TAKE THAT LONG! THE MAIN THING IS TO JUST GET ORGANIZED!
B. Holbrook
PILE D
PILE B
PILE A
PILE C
1/2

ZIGGY

© 1986 Universal Press Syndicate
BALANCE YOUR CHECKBOOK $1.00
7-24
Tom Wilson