PORT FE

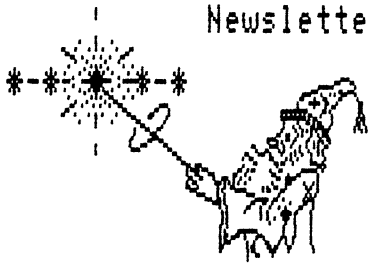
SORCERERS USERS' GROUP

(Toronto)

P.O. Box 1173 Sta. 'B'
Downsview, Ontario,
Canada. M3H 5V6

SORCERER

Newsletter



The Toronto Sorcerer Users' Group was founded in the Spring of 1979, a handful of willing and eager to learn members.

This newsletter shall at all times keep in mind the goal at its conception. To spread the seeds of knowledge.

Articles printed in this newsletter shall be free for all Sorcerer Users' groups to reprint or comment on as they see fit.

Articles submitted for this newsletter must be in no later than the beginning of the 1st of every month.

September 1982 ISSUE

TABLE OF CONTENTS

GENERAL INTEREST

CP/M RELATED

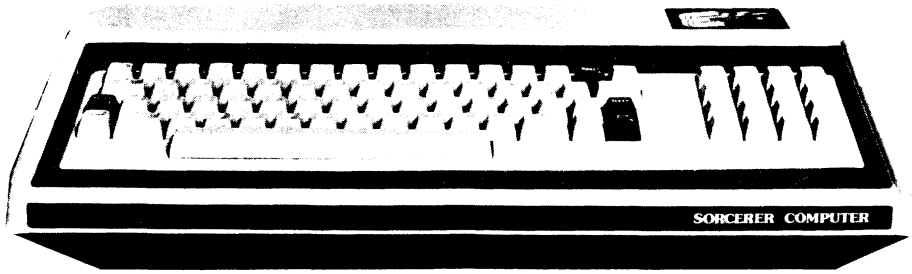
Technical Tips

MEETING PLACE

Location : Bathurst Heights Library - 7:00 PM 3170 Bathurst St.

Thur. Oct. 14
Thur. Nov. 18 Wed. Dec. 15

One block north of Lawrence on the west side of Bathurst.



CADAS changes for compatibility with EXMON2
by: Heinz Benedikt

The following address changes must be made with either DDT or SID for the CP/M version of CADAS.

NOTE: The listing below shows the information below the brackets to be as actual addresses. When changing the CADAS program remember that you must reverse the high order and low order bytes.

ex: to change address 02E6 which contains '3AE1'
and we want it to contain.....'5FE1'
using DDT *S2E6
02E6 3A 5F
02E7 E1 . (CR) don't change already is E1
*

CHANGE FROM OLD MONITOR	CHANGE TO EXMON2	ADDRESSES	
(information)		actual addresses	
E13A	E15F	02E6,03E2	LINEIN: (routine)
E1BA	E1CF	02DB,033F,04EB,07F0,0D25,0F8E,10DF,125B	
			MSGOUT: (routine)
E205	E1F8	032F,0332,035B,04B5,07F3,0945,0A93,0B05,0B15 0B1B,0B33,0CB4,0CED,0D0F,0D75,0F7F,0FA8,0FE7 10CA,1170,1197,11A8,125E,132D	
			CRLF: (routine)
E23D	E29B	03E8	CONV: (routine)
E28A	E2E7	04E1	MCTRON: (routine)
E2D2	E333	045B,04AD,112E	SPACES: (routine)
E6DE	E77A	04EE	HEDPRT: (routine)
E679	E73A	0A5D	SAVE1: (routine)
E71B	E7B7	04E4	GETHED: (routine)
E734	E7D0	0506	SKIPFL: (routine)
E7FE	E88F	0580	Point to loading message
E993	E030	03C7	(or use E033 printer call c/w LF)

After all the changes have been made exit DDT or SID with G0 or ^C

A>SAVE 27 CADM2.COM

You now have a copy of CADAS that will be compatible with EXMON2.

As you can see from the listing above that all of the changes involve jumps that are made directly into the monitor. This program I would consider one of the more involved programs that required a 'great deal of changes'. These changes took all of about one hour. Between SID (or any Z80 disassembler) and the SR (Search) from to xxxx function in EXMON2 it makes it almost child's play to change any old program over to be able to make it run under EXMON2.

EXMON2 saved a great deal of time in weeding out all those calls that used some of the monitor routines.

==========

EXMON2 information

Some people have exercised a lot of caution before getting involved with EXMON2, and I don't blame them for being hesitant at all. Some of the queries that have surfaced are mainly connected with Sorcerer II owners. They too would like to be able to switch back and forth, between the new and old monitor. Well stay tuned for in the next issue of PORT FE we shall give you a circuit board layout that will let you have a possible (FIVE) different monitors even in the Sorcerer II.

CP/M SECTION xxxx

SPELLBINDER / WORDPROCESSOR PAC .....&....EXMON2

We have found the necessary addresses in the WORD PROCESSOR PAC that need to be changed. For some of you that are running the PAC either relocated for disk operation or the actual PAC itself. In the PAC the last PROM/EPROM is the only one affected. The following are the addresses that have to be changed:

	add.		was/is		change to
WORD PROCESSOR PAC	DF21 -	LD HL,	EC6E	------> GRATEL:	ED4C
==========	DF3F -	LD HL,	ECBE	------> CONTBL.	ED9C
	DF46 -	LD HL,	ED0E	------> SHITBL.	EDEC
CHANGES	DF51 -	LD HL,	ED5E	------> SLOTBL.	EE3C
=======	DF56 -	LD HL,	EDAE	------> UNSTBL:	EE8C

As for the **SPELLBINDER** file, there may be several versions in use, however many versions there are, will not affect 'what you must change' only the addresses where you find them may differ from version to version. After having changed the file it works just the same with EXMON2 as it did with the old monitor. (still memory bound !!! unfortunately).

	add.		was/is		change to
SPELLBINDER Vrs xx	45EC -	LD HL,	EC6E	------> GRATBL.	ED4C
==========	460A -	LD HL,	ECBE	------> CONTBL:	ED9C
	4613 -	LD HL,	ED0E	------> SHITBL.	EDEC
CHANGES	461C -	LD HL,	ED5E	------> SLOTBL:	EE3C
=======	4621 -	LD HL,	EDAE	------> UNSTBL:	EE8C

FOR SPELLBINDER - SAVE 89 SPELLM2.COM or SAVE 78 SPELLM2.COM
(Installed version)

Those are the only things that need to be changed to make it compatible with EXMON2.

==========

EXBASIC & EXMON2

The following changes make EXBASIC compatible with EXMON2.

The first part contains the changes to standard EXBASIC. The second part are the additional code changes for EXBASVE - (EXBASIC installed with the full Screen Editor).

******

part one.

address:	old code:	new code:
3934H	CALL E1A2H	CALL E045H
39A0H	CALL E1A2H	CALL E045H
39A9H	CALL E9E8H	CALL E04EH
39E8H	CALL E9C0H	CALL E04BH

part two. (EXBASIC installed with Full Screen Editor)

5917H	CALL E018H	(SAME)
591BH	CALL E9D6H	CALL E051H
597DH	LD HL,E018H	(SAME)
59ACH	JP E9F0H	JP E01BH
5A07H	CALL E9B1H	CALL E046H
5A6CH	CALL E045H	CALL E00CH
5A73H	CALL E205H	CALL E1F8H
5D11H	LD DE,E01EH	(SAME)

As you may gather from the information regarding changes above, most programs require very little change. EXMON2 is proving to be quite the handy tool.

We shall try to keep you updated on all programs and their changes as they become available. If you have anything to add to the expanding list of modified programs, please write in and let us know what they are so that others may benefit.

yours : The chief cook and bottle washer.

SUPERCALC on the SORCERER !!!!!!

YES YES YES

---- Now were getting somewhere. For those of you not familiar with this " piece the resistance ", I will attempt to enlighten the avid person who has not had the oportunity to view or use such a masterpiece of programming forethought. I could go on and on for quite some time on this subject, but I won't bore you with my enthusiasm.

EXMON2 made it possible again. What else can I say.

To see this program up and running on the Sorcerer has made my day. This is a full fledged spread sheet calculator with full scientific math functions built right in. Yes it even can do the mundane tasks like addition etc.. It even lets you do split screen work, hidden entries (for your eyes only).

Cursor positioning has a built in memory that orients the direction of movement, for cell entry, automatically in whatever direction that was last used.

It even has a screen dump routine so that you can dump whatever strikes your fancy. One of the OUT-PUT functions allows you to dump the cell formulas and not the contents, great for tracking down formula errors. For the brief time that I've been exposed to the program, I really like what I've seen.

Functions -- functions -- many of the functions and the formats in which you can mix text and even protect areas from being disturbed are terrific. The built in help menu's are through out the command levels, at whatever point you're at, all you have to do is hit the '?' and up comes some help. Don't let me lead you into thinking that you don't need a manual, because you do. But for the most part it really is a pleasure to work with.

This is ideally suited for the engineering field where data variables entered, have a dramatic effect on either costing or performance of various pieces of equipment. The following will give you some idea as to its potential

Information returned and values contained in Cells

Symbol	Value	Units		CELL #	formula used
Tr	-50F		<---	D5	= -50
Te	50F		<---	D6	= 50
dT	100F			D7	F= D6-D5
Hd	10ft		<---	D8	= 10
Wd	10ft		<---	D9	= 10
Pr	0968lb/ft**3			D10	F= -.0002*D5+.0868
Pe	.0768lb/ft**3			D11	F= -.0002*D6+.0868
Vt	554.689102lft/min			D12	F= 4000*SQRT(.1923*(D10-D11)*(D8/2))
Vd	3175.855813ft/min			D13	F= (3.435+.32721*(D8-3))*D12
Va	1641.619038ft/min			D14	F= (2.625+.04779*(D8-3))*D12
Ws	3.25inches			D15	F= LOOKUP(D9,H5:H23)
As	2.708333333ft**2			D16	F= D9*(D15/12)
Q	8601.276161cfm			D17	F= D13*D16
Theta	19.74846418degrees			D18	F= ASIN(D12/D14)*(180/PI)
D	12.57336669inches			D19	F= 10*TAN(D18)
B	4.423963134inches			D20	F= D8/((D10/D11)+1.0)
Vp	100ft/min		<---	D21	= 100
	Blower Pressure			D23	F= 4*(D14*(D21/4000)^2)
SpH	4.104047596In. W.G.			D24	F= 3.3*(D14*(D21/4000)^2)

As you can see, the math involved will handle the most demanding type of calculations. For normal use, spread sheets, costing, etc... it works very well

SUPERCALC really works very well with the Sorcerer. The on screen formating and display of information is in no way hindered by only displaying 64 characters. Direct cursor addressing is really not used with this program, but some of the other features such as reverse video, clear to end of line, clear to end of screen are used.

On the last three pages is an actual installation procedure that one goes thru for installing SUPERCALC on the Sorcerer. For those of you that have it, or for those of you who would like to get it, I can only say ' I think it's just great ' Good luck and enjoy.

by: H.A. Lautenbach

The PL/I-80 System.

PL/I-80 is the ANSI standard subset G of the powerful PL/1 programming language. PL/1 has existed on large mainframe computers since the late sixties but has only been available on microcomputers for the last couple of years. It is compiled like 'C' and FORTRAN as opposed to an interpeted language like basic or forth. The compiler produces true 8080 machine language as opposed to P-CODE which some pascal compilers produce. This fact, along with a very efficient code generator make this one of the fastest languages on the market today. PL/I-80 was written by DIGITAL RESEARCH.

Besides being a fast language, PL/I has several other features which make it a very attractive language. Firstly it is a block structured language, that is to say that one statement may be replaced by a group of statements by delimiting the group with DO and END statements. Blocks of code may also be delimited with BEGIN and END and then temporary variables may be defined which only live as long as that block is active.

Another advantatge is the clean interface between a PL/I program and its subroutines. This is accomplished through formal parameter passing. Parameter passing in PL/I is done by a method known as call-by-reference, that is, when a parameter is passed to a sub-routine a pointer to the variable is passed, not the value of the variable itself. To the programmer, though, it looks like the value of the variable is passed because the compiler handles all the juggling of pointers.

PL/I also has a very impressive file handling functions. There are two kinds of files in PL/I, sequential, and random access. Sequential files may either be record oriented or just a stream of ascii characters, a serial file may be opened as either an input file or an output file. Random access files are, by their nature, record oriented. They are keyed on a spesific field of a record. A random access file may be opened as input, output, or update. If the keys of a file are not known, a file may be read through sequentially and the key field of each record stored into an array.

As well as disk files, PL/I may open a file to any CP/M device.

PL/I also has very flexible output. The PUT LIST and PUT EDIT statements provide output flexibility equalled by none. There are facilities for formatting output which make report generating a snap.

PL/I supports a rich set of data types. Data may be declared as ;

FIXED BINARY, to a precision of 15 bits.
FIXED DECIMAL, bcd with 15 digit precision.
FLOAT BINARY, floating point 24 bit precision
CHARACTER, any ascii characters, the maximum string length is 255 characters.
POINTER, an 8 or 16 bit pointer value.
BIT, a bit string up to 16 bits long.

Variables may be any one of the above data types and have any of the following attributes;

AUTOMATIC, lives within a block or procedure and dies when the program exits the block or procedure
STATIC, lives within a block or procedure but retains its value after the program leaves its block or procedure.
EXTERNAL, The variable is global to all procedures.

In addition to the above attributes a variable may also be,

BASED on a certain memory location and accessed using pointers. These variables are allocated storage dynamically at runtime by the functions ALLOCATE and FREE.
ENTRY, a variable which gets its value from a separately compiled and linked, or an assembly language subroutine.

PL/I also supports a powerful feature called STRUCTURES. A structure is a data record containing several fields which may be accessed as a unit. When two separate structures, or two structures in an array of structures, are to be equated, you need not equate the individual fields one at a time, all you have to do is say STRUCTA=STRUCTB.

There are examples of most of these features in the example programs after this article. I will be printing a continuing column on PL/I here, space permitting, and will be presenting in much greater detail the true power of PL/I. Any questions on the subject of PL/I may be addressed to PORT-PE or myself. I would be happy to be of assistance

by Tom Gottweis ,4662 Kingston Rd. Unit 115
West Hill, Ontario. M1E 4Y7

IDS460 CP/M PRINTER SET CALL ROUTINE

by: Tom Gottweis

This is a PL/I listing for doing printer setup direct from a menu driven CP/M file. It allows the printer to be set up first, before sending text. This feature takes advantage of the programable features, like setting form size, character size, auto justify, proportional spacing etc.... comes in very handy when you just don't want to reselect the default switches just for a specific printout. This program (CP/M

```
460SET:
        PROC OPTIONS(MAIN);
%include 'DIOMOD.DCL';
%REPLACE NUMFUNC by 27;
DCL
(c,i,j) fixed binary,
1 d static,
 2 maxsize bit(8) init('2'b4),
 2 inchars char(3) var,
prfn(1:30) STATIC CHAR(64) VAR INITIAL(''),
tabstr char(32) var,
formln(1:30) static char(15) var initial('');
prfn(1)='1)Carriage return.^M^J';
prfn(2)='2)Line feed.^M^J';
prfn(3)='3)Form feed.^M^J';
prfn(4)='4)Enter enhanced mode.^M^J';
prfn(5)='5)Exit enhanced mode.^M^J';
prfn(6)='6)Text justify mode on.^M^J',
prfn(7)='7)Text justify mode off.^M^J';
prfn(8)='8) Spare not used .^M^J';
prfn(9)='9) Spare not used .^M^J';
prfn(10)='10)Set graphics mode on.^M^J',
prfn(11)='11)Set 6 lines per inch.^M^J';
prfn(12)='12)Set 8 lines per inch.^M^J';
prfn(13)='13)Set form length.^M^J';
prfn(14)='14)Set 10 cpi.^M^J',
prfn(15)='15)Set 12 cpi.^M^J',
prfn(16)='16)Set 16.8 ^M^J',
prfn(17)='17)Set vertical advance 1.^M^J';
prfn(18)='18)Set vertical advance 2.^M^J',
prfn(19)='19)Set vertical advance 3.^M^J',
prfn(20)='20)Set left/right margins.^M^J',
prfn(21)='21)Set vertical tabs.^M^J',
prfn(22)='22)Set proportional on.^M^J',
prfn(23)='23)Set proportional off.^M^J';
prfn(24)='24)Set horizontal tabs.^M^J';
prfn(25)='25)Select printer.^M^J';
prfn(26)='26)deselect printer.^M^J',
prfn(27)='27)RETURN TO CP/M.^M^J',
      formln(1)='0"^M^J',
      formln(2)='3.0"^M^J',
      formln(3)='3.5"^M^J',
      formln(4)='4.0"^M^J',
      formln(5)='4.5"^M^J',
      formln(6)='5.0"^M^J',
      formln(7)='5.5"^M^J',
      formln(8)='6.0"^M^J',
      formln(9)='7.0"^M^J',
      formln(10)='8.0"^M^J',
      formln(11)='8.5"^M^J',
      formln(12)='9.5"^M^J',
      formln(13)='10"^M^J',
      formln(14)='11"^M^J',
      formln(15)='12"^M^J',
      formln(16)='14"^J^M';
do while ('1'b),
   call conout('^L'),
     do i=1 to numfunc,
      do j=1 to length(prfn(i)),
      call conout(substr(prfn(i),j,1));
   end,
end,
      call rdbuf(addr(d));
      c=binary(inchars),
      if c=1 then call setlpt(13,00),
      if c=2 then call setlpt(10,00),
      if c=3 then call setlpt(12,00),
      if c=4 then call setlpt(00,01);
      if c=5 then call setlpt(00,02);
      if c=6 then call setlpt(00,04);
      if c=7 then call setlpt(00,05),
      if c=8 then call setlpt(00,00);
      if c=9 then call setlpt(00,00);
      if c=10 then call setlpt(00,03);
      if c=11 then do;
         call setlpt(27,44),
         call setlpt(66,54);
       end;
         if c=12 then do,
           call setlpt(27,44),
           call setlpt(66,56);
         end;
         if c=13 then do;
           call setlpt(27,44),
           call setlpt(76,44);
           call setlpt(coninp(),coninp());
           call setlpt(coninp(),44);
           call setlpt(coninp(),coninp()),
           call setlpt(coninp(),13),
         end;
         if c=14 then call setlpt(00,29),
         if c=15 then call setlpt(00,30),
         if c=16 then call setlpt(00,31),
         if c=17 then do;
           call setlpt(27,44);
           call setlpt(66,44);
           call setlpt(coninp(),coninp()),
           call setlpt(coninp(),13);
          end;
          if c=18 then  do;
           call setlpt(27,44);
           call setlpt(67,44);
           call setlpt(coninp(),coninp());
           call setlpt(coninp(),13);
          end,
          if c=19 then  do,
           call setlpt(27,44),
           call setlpt(68,44),
           call setlpt(coninp(),coninp()),
           call setlpt(coninp(),13);
          end,
          if c=20 then  do;
           call setlpt(27,44);
           call setlpt(74,44);
           call setlpt(coninp(),coninp());
           call setlpt(coninp(),44),
           call setlpt(coninp(),coninp());
           call setlpt(coninp(),13);
          end,
          if c=21 then  do;
           call setlpt(27,44),
           call setlpt(69,44);
           call rdbuf(addr(tabstr));
            do j=1 to length(tabstr);
             call wrlst(substr(tabstr,j,1)),
            end,
           end,
          if c=22 then call setlpt(00,18),
          if c=23 then call setlpt(00,06),
          if c=24 then  do;
           call setlpt(27,44),
           call setlpt(70,44);
           call rdbuf(addr(tabstr)),
            do j=1 to length(tabstr),
            call wrlst(substr(tabstr,j,1));
           end;
          end,
         if c=25 then call setlpt(00,17),
         if c=26 then call setlpt(00,19);
         if c=27 then call reboot(),
        end,
    setlpt:
      proc(contval,setval),
      dcl (contval,setval) fixed binary;
      call wrlst(ascii(contval)),
      call wrlst(ascii(setval)),
    return;
    end;
    end 460SET,
```

FROM: Thomas Hill
200 Oklahoma
Anchorage, Ak. 99504

SUBJECT: Undocumented CP/M BDOS Features August 12,1982

Just a short note to aquaint you with an "undocumented feature" I have encountered in the CP/M 2.2 BDOS. While developing an assembly program which read and wrote disk files, an early version did not open the output file before writing to it. Oddly enough, the BDOS accepted the write and did not return an error condition. Being a curious soul (and cautious), I sidetracked to investigate this effect. A call to Digital Research resulted in a letter informing me that they knew of the effect and told me it was an "undocumented feature" of CP/M. They also told me that it was the programmer's responsibility to open and close his files properly, to which I heartily agree.

However. I wrote some test programs to determine WHERE on the disk the information was going, and WHAT happened to the valid data on the disk. Writing to an unopened file apparently writes information beginning at Group 0, sector 1 and continues in a sequential manner thru the allocation map. (I lost three directories that way). No change is made in the allocation map, however, and the only change in the File Control Block is the Current Record and Next Record fields are incremented. NO CHANGE occurs in the FCB allocation map.

While it is, of course, the programmer's responsibility to control the file accesses, and proper opening and closing is mandatory, in some cases (particularly during program development), proper file access may not take place. If this occurs, a possible loss of data may result.There may be a BDOS patch which will clear this up, or someone out there may already have one. If anyone knows more about this, I would appreciate it if you would drop me a line at the above address.

Thanks,
Thomas Hill

==========

Sb: CP/M 3.0 31-Aug-82 23:50:15 Fm: Digital Research

Well, I haven't the foggiest what the enduser upgrade will be. Or if there will be one. Requires a lot of mods to a 2.2 BIOS to take advantage of some of the performance enhancements.

CP/M3 is capable of doing disk I/O 10-20 times faster than optimized 2.2 systems, especially on hard disk systems. It can run with the BDOS mostly banked, and use alternate banks as L.R.U. directory caching and deblocking buffers. The deblocking is done by the BDOS, not the BIOS. There is a whole new set of utilities, it will come with RMAC, LINK, LIB, a MP/M-like SET and SHOW instead of STAT, rename with wildcards, console I/O redirection to/from disk, lots of other goodies like that. Most users should wait for their system suppliers to offer preconfigured systems, unless they have implemented their own 2.2 BIOS's. We are finding it increasingly hard to give tech-support to individuals trying to write BIOS's.

I guess I could go on for a while, but I mainly know about the internals of the BDOS-BIOS interface, and not much about the enduser interface level of the system. Ask specific questions, I'll try and answer them.

-jrp

==========

Re: Shugart disk drive lubrication by: Charlie Strom

I recently received some valuable info re lubrication of Shugart 8" drives, prompted by intermittant squeaking from a drive! Carefully, spray a small amount of aerosol silicone lubricant on the lead screw (without getting it all over the place!) I was amazed at the noise reduction I saw immediately. According to my source, Shugart has done this themselves in spite of no mention in service manuals. It certainly hasn't hurt my drive operation. Don't use a lubricant with any petroleum base... pure silicone (food grade) only! The brand I have is called Sprayway, from Sprayway, Inc., Addison, Ill 60101.

==========

FEEDBACK RE: QUESTIONNAIRE REQUESTED IN THE AUGUST ISSUE OF PORT FE
General concensus so far has been as follows:

1) Yes 79% ,No 20%,Don't care 1%
2) Yes 24% ,No 76%,Don't care 0%
3) Yes 79% ,No 20%,Don't care 1%
4) Yes 80% ,No 16%,Don't care 4%
5) Yes 50% ,No 5% ,Don't care 45%
6) Yes 50%, No 50%, Don't care 0%
7) Yes 35%, No 20%, Don't care 45%
8) Yes 62%, No 20%, Don't care 18%
9) Keep it as is 82%,Change it - 2%
Don't care - 13%

Thank you to those that have sent in your returns so far. We shall be publishing some/all of the comments that you submit,(pending room in PORT FE).

Some of you have not sent your returns to the club address,(ahem !!! saving postage eh!!). Please direct these thru proper channels please.

Please Send in YOUR OPINION to: C/O The PREZ ,PORT FE, P.O. Box 1173,Stn.'B', Downsview,Ont.,Canada, M3H 5V6

FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS FDS

Exidy Bios Modification.

Have those of you with EXIDY FDS systems ever had the flaky disk blues, you know, when you know that a sector is questionable but you think you can still get it off disk. And you have to WAIT through hours of RETRYS before you finally get it off. Well there is fortunatly a cure. There is a modification which can be made to the bios which will disable the step in/out and the deselect/reselect phases of the bios retry cycle. This in effect cuts out the track seek time, which is by far the most time consuming part of disk operations. What the bios will then do, is to re-read the track 5 times as is doccumented in the CBIOS section of the CP/M manual, and repeat the series of re-reads 6 times. I have found that on most questionable sectors I have been able to read the sector with only 2 or 3 sets of re-reads. Be forewarned though, after the sixth set of retrys if the sector still has not been read, the system will crash with a very heavy thud. You will then not be able to do anything but cold boot. I think that after 5 sets of 5 retrys you can be pretty sure that the sector really has been stepped on. You can at any time during the retrys hit ^C just like in the regular version. Anyway, on to the mod.

First make sure that you have a backup copy of MOVCPM.COM .
Then get into SID by typing

```
SID MOVCPM.COM
```

Sid will then respond with

```
SID VERS. 1.4
SYMBOLS
NEXT   PC   END
2D00  0100 (dependant on your memsize)
```

You now type

```
S1F02(cr)
```

Sid will respond with

```
1F02 01 FF(cr)        You now type
1F03 .(cr)            You now type
#^C (cr)              You type (to return back to CP/M
When you get back into CP/M type
SAVE 44 MOVCPM.COM
```

and you have a patched version of CP/M which doesn't take hours to tell you a sector is bad.
To get a version of this CP/M onto disk just follow the normal MOVCPM and SYSGEN procedures as outlined in the CBIOS manual.
NOTE WELL. THIS MODIFICATION WILL ONLY WORK WITH CP/M VERSION 1.4. If you are using CP/M 2.2 then I can't guarantee this modification won't do bizzare things.

By Tom Gottweis.

==========

RE: Hayes Smartmodem Speedup modifications.
Messages received from various sorces BBS's etc...

According to the supplied literature of the D.C.Hayes MicroModem-100, it cannot be operated above 300 baud The limiting factor is not the Uart or it's clock frequency but the Microcoupler itself (according to its designer). Not accepting that their design was such that its upper limit was no higher than 300 I decided to experiment. If you replace the diodes that determine the 12-bit divider for the clock frequency to a decimal 17 - 00010001 - (I used a 16-pin socket and a removeable header), you can squeeze out 450 Baud. I couldn't get 600 baud, but the upper limit probably varies from unit to unit.
If you wish you had a PMMI, give this a try. Anything is better than 300.
Bob Loesch Bel Air, Md. (301) 879-7135

Msg 11796 02/18/82 from GENE PLANTZ (Messages from CBBS/Chicago)
to DAV HOLLE re: SMARTMODEM AT 600

Dav.. thanks for the info on the baud rates for the Smartmodem. I tried it at work at 600 baud and I only got about 50% intelligible data. Can't get the system at work to do 450 because the baud rate generator won't support it. Am anxious to try it tonite on my IBMPC. What system are you using, and what values are you using to get your UART to run at 450? Will let you know how it goes with the PC. Thanks Gene Plantz

Msg 11800 02/18/82 from GENE PLANTZ
to DAVID DUNKLEMAN re: RUNNING AT 600 ON SMARTMODEM

David. I am connected and entering this message at 600 baud. I have only seen 2 characters garbled so far Am very surprised. The IBMPC baud rate generator will not accept 450 as a valid rate. So it is either 300 or 600. The communications pgm I am using is written in BASIC and is called CMX (modified some by me). Apparently, with the buffering that BASIC is doing, it works. Try it with whatever TP program you have. Gene Plantz

Msg 11828 is 05 line(s) on 02/20/82 from GEORGE BRICKNER
to ALL re SMARTMODEM AT 600 BAUD

I am entering this message at 600 baud using a Hayes Smartmodem and a TRS-80 Model III and Omniterm. The TRS-80 RS-232 interface does not support 450 baud either. George Brickner

Msg 11829 is 06 line(s) on 02/20/82 from DALE SMITH
to ALL WHO'VE TRIED re: SMARTMODEM @)300 BAUD

We have been totally unsuccessful online @) 300 baud. Modem has no problem local at any baud. Our baud rate generator is a 555 with a 25K trimmer pot on pins 6,7,8; frequency counter reads 7.2 KHz +/- 200 Hz (least significant digit usually flutters) for 450 Baud but CBBS doesn't recognize my C/R to sign on. Any suggestions anyone? Tnx

==========

MBASIC

THE CALL STATEMENT

There have been a number of requests for clarification of the CALL statement in the various Microsoft languages. Hopefully, the following explanations will clear up several misunderstandings.

First, let's look at the "Microsoft CALL standard". When any Microsoft program issues a CALL statement, such as "CALL MYROUT", control is passed to the address specified by MYROUT. In all cases except the Basic Interpreter, MYROUT must be the name of a SUBROUTINE program, or a GLOBAL/PUBLIC label within a Macro-80 routine. The calling and the called program must be linked together by L80 into a single file.

In the Basic Interpreter (MBASIC), the variable MYROUT must be set to the beginning address of the subroutine. If we complicate things by adding a few paramaters to the CALL, such as, "CALL MYROUT(A,B)", the parameters are passed as follows. The *ADDRESS* of the first parameter is passed in registers HL, the second in DE, and the third in BC. If there are more than three parameters, BC points to a block of memory containing the *ADDRESS* of parameters three through N. Note that the address and *NOT* the parameter itself is passed.

The arguments themselves correspond to the standard Microsoft variable format. (2 bytes for an integer, 4 bytes for single precision floating point, and 8 bytes for a double precision floating point number) with two exceptions.

COBOL passes variables as they would appear to another COBOL program (DISPLAY, COMP, or COMP-3).

Strings are also handled a bit differently. The address pointed to by the register contains a three byte "string descriptor". This string descriptor contains the length of the string in byte one, and the address of the string in bytes two and three. When passing strings, take care not to modify this string descriptor, or unpredictable results will occur.

In all cases, it is the user's responsibility to ensure that the arguments correspond *EXACTLY*, in both type and number. Also, be sure to preserve *ALL* of the registers and use your own local stack when you call Macro routines.

With the preliminaries out of the way, let's look at which languages can call which other languages. In the following table, "B" represents the Basic Interpreter (MBASIC), "BC" the Basic Compiler (BASCOM), "F" the Fortran Compiler (F80), "C" the Cobol Compiler, and "M" the Macro Assembler (M80). A "Y" in the appropriate entry means that a CALL is possible.

CALLing Program	CALLed Program: B	BC	F	C	M
B					Y
BC			1		Y
F			Y		Y
C			2	Y	Y
M			3		Y

Notes: 1 When calling a FORTRAN routine from the Basic Compiler, only one of the two languages may be used to perform I/O. When the programs are linked, link the Basic program first, then search BASLIB, then load the Fortran program, then search FORLIB. The multiply defined global message may be ignored.

2 When calling FORTRAN from COBOL, remember that the variable types are different. Only COMP data items will be passed in such a way that FORTRAN can deal with them without an ENCODE statement.

3 While Macro-80 may not directly CALL Fortran subroutines, you may make use of the routines in the FORTRAN Library. For more information, see the Fortran Manual. Of course, from within M80, you may initiate execution of any other .COM file by reading the file and then jumping to the appropriate address.

```
A0)
A0>INSTALL

          SUPERCALC Install Program
              Version : 1.21

This program will Install your SuperCalc file
for the terminal that you wish to use.

Do you wish to proceed (y/n) ? Y

----------------------------------------


Enter name of SuperCalc(tm) file as .
"d:filename" where "d" is the drive.
Enter "?" for more Help.
Enter name : A:SC

Now reading data from A:SC.com...


----------------------------------------

These are the terminals supported for use
     with SUPERCALC :
 A. ADDS                    H. Televideo
 B. Apple II                I. Visual
 C. Beehive                 J. Xerox
 D. DEC                     K. Zenith
 E. Hazeltine               L. Infoton (GTC)
 F. Lear Siegler            M. ANSI Standard
 G. Soroc

 Y. Write SuperCalc to disk
    (with any changes made so far.)
 Z. Edit printer and terminal data

Enter A-M, Y, Z or ? . X



----------------------------------------
These are the items you may now edit .

 A.  Edit screen controls.

 B.  Edit attribute data.

 C.  Edit input keys.

 D.  Edit GotoXY, printer init. string.

 E.  Edit miscellaneous data.

 F.  Edit terminal name.

X.  Finished editing data.
Enter A-F or X or ? : A


This is the current data in Hex.

 1. Clear Screen : Unconfigured.
 2. Clr. to EOL  : Unconfigured.
 3. Home Cursor  : Unconfigured.
 4. Clear to EOS : Unconfigured.
X. Exit to Menu.
Enter 1-4 or X or ? . 1


The current value for Home and Clear Screen
 is . Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) . 1

Enter data byte in Hex (or ?). 0C


This is the current data in Hex.

 1. Clear Screen :  1, 0C
 2. Clr. to EOL  : Unconfigured.
 3. Home Cursor  : Unconfigured.
 4. Clear to EOS : Unconfigured.
X. Exit to Menu.
Enter 1-4 or X or ? : 2


The current value for Clear to End of Line
 is : Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) : 2
Enter data byte in Hex (or ?): 1B
Enter data byte in Hex (or ?): 31


This is the current data in Hex.

 1. Clear Screen .  1, 0C
 2. Clr. to EOL  .  2, 1B, 31
 3. Home Cursor  . Unconfigured.
 4. Clear to EOS : Unconfigured.
X. Exit to Menu.
Enter 1-4 or X or ? : 3


The current value for Home Cursor
 is : Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) : 1
Enter data byte in Hex (or ?): 11


This is the current data in Hex.

 1. Clear Screen .  1, 0C
 2. Clr. to EOL  :  2, 1B, 31
 3. Home Cursor  .  1, 11
 4. Clear to EOS . Unconfigured.
X. Exit to Menu.
Enter 1-4 or X or ? : 4


The current value for Clear to End of Screen
 is : Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) : 2
Enter data byte in Hex (or ?): 1B
Enter data byte in Hex (or ?): 32


This is the current data in Hex.

 1. Clear Screen :  1, 0C
 2. Clr. to EOL  :  2, 1B, 31
 3. Home Cursor  :  1, 11
 4. Clear to EOS :  2, 1B, 32
X. Exit to Menu.
Enter 1-4 or X or ? : X
```

```
---------------------------------------
These are the items you may now edit .

 A.  Edit screen controls.

 B.  Edit attribute data.

 C.  Edit input keys.

 D.  Edit GotoXY, printer init. string.

 E.  Edit miscellaneous data.

 F.  Edit terminal name.

X.  Finished editing data.
Enter A-F or X or ? : B


This is the current data in Hex.

 1. Set Inv. Video    . Unconfigured.
 2. Clear Inv. Video : Unconfigured.
 3. Set Underscore   : Unconfigured.
 4. Clear Underscore : Unconfigured.
 5. Set Highlight    : Unconfigured.
 6. Clear Highlight  : Unconfigured.
X. Exit to Menu.
Enter 1-6 or X or ? : 1


The current value for Set Inv. Video
 is : Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) : 2
Enter data byte in Hex (or ?): 1B
Enter data byte in Hex (or ?): 36


This is the current data in Hex.

 1. Set Inv. Video    .  2, 1B, 36
 2. Clear Inv. Video : Unconfigured.
 3. Set Underscore   : Unconfigured.
 4. Clear Underscore : Unconfigured.
 5. Set Highlight    . Unconfigured.
 6. Clear Highlight  : Unconfigured.
X. Exit to Menu.
Enter 1-6 or X or ? : 5


The current value for Set Highlight
 is : Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) : 2
Enter data byte in Hex (or ?): 1B
Enter data byte in Hex (or ?): 34


This is the current data in Hex.

 1. Set Inv. Video    .  2, 1B, 36
 2. Clear Inv. Video : Unconfigured.
 3. Set Underscore   : Unconfigured.
 4. Clear Underscore : Unconfigured.
 5. Set Highlight    :  2, 1B, 34
 6. Clear Highlight  : Unconfigured.
X. Exit to Menu.
Enter 1-6 or X or ? : 6


The current value for Clear Highlight
 is : Unconfigured.
Do you wish to change this (y/n) ? Y

Enter 0 to unconfigure function or
Enter the no. of bytes to follow (in Hex)
(max 9 bytes) : 2
Enter data byte in Hex (or ?): 1B
Enter data byte in Hex (or ?): 35
```



```
This is the current data in Hex.

 1. Set Inv. Video    :  2, 1B, 36
 2. Clear Inv. Video : Unconfigured.
 3. Set Underscore   : Unconfigured.
 4. Clear Underscore : Unconfigured.
 5. Set Highlight    :  2, 1B, 34
 6. Clear Highlight  :  2, 1B, 35
X. Exit to Menu.
Enter 1-6 or X or ? : X


---------------------------------------
These are the items you may now edit .

 A.  Edit screen controls.

 B.  Edit attribute data.

 C.  Edit input keys.

 D.  Edit GotoXY, printer init. string.

 E.  Edit miscellaneous data.

 F.  Edit terminal name.

 X.  Finished editing data.
Enter A-F or X or ? : C


This is the data to edit.
1. Keyboard lead-in character  : No lead-in
2. Key to input for - up       : ^K
3. Key to input for - down     : ^J
4. Key to input for - left     : ^H
5. Key to input for - right    . ^L
6. Key to input for Help : }
X. Exit to menu.
Enter 1-6 or X . 2


The current cursor Up key is : ^K
Do you wish to change this character (Y/N) ? Y

Enter character to use or
 enter space for no change .

This is the data to edit.
1. Keyboard lead-in character  : No lead-in char
2. Key to input for - up       : ^E
3. Key to input for - down     : ^J
4. Key to input for - left     : ^H
5. Key to input for - right    . ^L
6. Key to input for Help : }
X. Exit to menu.
Enter 1-6 or X : 3


The current cursor Down key is : ^J
Do you wish to change this character (Y/N) ? Y

Enter character to use or
 enter space for no change :

This is the data to edit.
1. Keyboard lead-in character  : No lead-in char
2. Key to input for - up       : ^E
3. Key to input for - down     : ^X
4. Key to input for - left     : ^H
5. Key to input for - right    : ^L
6. Key to input for Help : }
X. Exit to menu.
Enter 1-6 or X : 4
```

```
The current cursor Left key is . ^H
Do you wish to change this character (Y/N) ? Y

Enter character to use or
 enter space for no change .

This is the data to edit.
1. Keyboard lead-in character   : No lead-in char
2. Key to input for - up        : ^E
3. Key to input for - down      . ^X
4. Key to input for - left      : ^S
5. Key to input for - right     . ^L
6. Key to input for Help : )
X. Exit to menu.
Enter 1-6 or X . 5


The current cursor Right key is : ^L
Do you wish to change this character (Y/N) ? Y

Enter character to use or
 enter space for no change .

This is the data to edit.
1. Keyboard lead-in character   . No lead-in char
2. Key to input for - up        : ^E
3. Key to input for - down      . ^X
4. Key to input for - left      : ^S
5. Key to input for - right     . ^D
6. Key to input for Help : )


X. Exit to menu.
Enter 1-6 or X . X

-----------------------------------------
These are the items you may now edit .

 A.  Edit screen controls.

 B.  Edit attribute data.

 C.  Edit input keys.

 D.  Edit GotoXY, printer init. string.

 E.  Edit miscellaneous data.

 F.  Edit terminal name.

X.  Finished editing data.
Enter A-F or X or ? . E


This is the data to edit
1  No. of video lines (down)      - 50
2  No. of video columns (across)  - 64
3  No. of printer lines           - 66
4  No. of printer columns         - 132
5  Dumb terminal cursor brackets.
   Left = >     Right = <
6  CPU Frecuency (in 1000s Hz.)   - 2000
7  Baud rate used (for delays)    - 96
8  No. of CRT attributes          - 0
9  Use printer status (in BIOS)
   (1 = yes, 0 =  no)             - 1
X. Exit to menu.
Enter 1-9 or X . X
```

```
-----------------------------------------
These are the items you may now edit .

 A.  Edit screen controls.

 B.  Edit attribute data.

 C.  Edit input keys.

 D.  Edit GotoXY, printer init. string.

 E.  Edit miscellaneous data.

 F.  Edit terminal name.

X.  Finished editing data.
Enter A-F or X or ? . F


The current terminal name is
To change, enter new name or press return to
leave name as is.
The maximum number of characters is 14.

Enter name : SORCERER


-----------------------------------------
These are the items you may now edit .

 A.  Edit screen controls.

 B.  Edit attribute data.

 C.  Edit input keys.

 D.  Edit GotoXY, printer init. string.

 E.  Edit miscellaneous data.

 F.  Edit terminal name.

X.  Finished editing data.
Enter A-F or X or ? . X


-----------------------------------------
This is what you may now do:

 A.  Save SuperCalc on disk.

 B.  Return to first menu.

X.  Quit Install Program
```

Membership Application Form Covering Jan. to Dec.1983

Membership to the group is not restricted to the TORONTO area. All persons willing to participate are invited to join.

As a member of the Sorcerer Users' Group (Toronto), I enclose the annual membership fee and agree to the following Terms.

1. That I will not, without the authorization of the board of directors, represent myself or take any action as agent, or representative or become spokesperson of the group.

2. That I will not use any software obtained from the SUGT library for any commercial purpose or financial gain. The library shall be available to me should I wish to obtain programs donated by other members. These programs shall not be distributed without the owners consent and/or the consent of the board of executive officers.

3. That I have the right to vote for the officers and directors of the organization at the annual general meeting.

4. That any breach of the above conditions and any other restrictions that the Officers of the Club may invoke in the future on my part may result in suspension or termination of my membership without refund.

Annual Membership Rates: (Jan - Dec)

Canadian - $21.00 Cdn
U.S.& Foreign $25.00 (U.S Funds)

Payable to - SORCERER USERS' GROUP (TORONTO) - by Cheque or Money Orders.

The SUGT program library is available to all members in the following manner.

You may send $6.00 + $1.50 postage for each volume as they become available and we shall supply the cassette/s. Program cassettes shall be sent via Air Mail.
All issues of PORT PE shall be mailed first class, in the case of non local issues, they are mailed via Air Mail. Past issues of PORT PE are only available for the current calendar year. Contact the editor, he will advise the amount of payment for previous issues.

NAME(print):.....

ADDRESS:.....

CITY:.....

POSTAL CODE:.....

TELEPHONE. Res..... Bus.....

Payments enclosed (membership):..... Library tape/s.....Vol 1 or 2

Signature:.....

Please list the type of equipment you are using etc...

Sorcerer size: 8... 16... 32... 48... other..... S100... Graph board.....

Disk system - Micropolis..... Discus.... Exidy.... other... Size.....

Other Equipment .....
.....
.....
.....
.....

If you belong to any other Sorcerer Users' Group please list it below.
.....