The MMSFORTH Newsletter

MILLER MICROCOMPUTER SERVICES
61 LAKE SHORE ROAD, NATICK, MASS. 01760
(617) 653-6136

September-October 1980

Vol.1 No.4

GROWING, GROWING, GROAN! (Editorial)

It's history now! August's special Forth issue of BYTE Magazine has launched Forth into the limelight - and MMS into a whirlwind of additional activity. We've been working overtime to keep up with MMSFORTH's increased popularity and are doing just a tad of additional work "on the side" - our continuing development of MMSFORTH, our custom software projects and system selection work for a growing group of clients, et al! If our phone has been overly busy or a particular book arrived late, please be assured that we are adding staff and increasing our stock of items as quickly as we dare, carefully increasing quantity while maintaining the quality of our services. Nearly all orders and projects have remained on schedule despite the increased work load. For those who have felt the pinch, thank you for bearing with us! (Good grief! What will we do when they report on THE DATAHANDLER?!)

Rip-offs hurt you and MMS. MMS because software piracy costs us sales, you because our necessary response is to increase our prices or to shift our attention further towards custom programming instead of inexpensive packaged software. We want to continue providing affordable and excellent products to our now over 1,000 users and more to come. So we ask you to participate actively by complaining publicly about bootleg copies of our software (and that of others!) and reporting violations to the vendor and author. The latest scam is lending libraries, and some unethical user groups also need watching and reporting - hopefully, educating as well. As a legitimate user of MMSFORTH and our other systems, keep our "front screen" (showing the item, copyright notice and your serial number) aboard on all your copies. Consider disguised versions and multiple users of the same serial number to be suspect, and report them to us. We are hearing of the problems, and will be measuring the level of group protectiveness as one important input on future projects and pricing. Please help us all!

-- Dick Miller, Editor 4th Class

FORTH MEETINGS COMING UP!

FORML CONFERENCE (Forth Modification Laboratory):

A three-day ADVANCED technical workshop for the discussion of topics related to FORTH implementation, language and application. No introductory tutorials will be held. Wednesday-Friday, November 26-28, 1980 at the Asilomar Conference Grounds, Pacific Beach, CA, $100. Asilomar is rustic and nice, on the ocean near Monterey, hosted by the Forth Interest Group, provision for families AND there will be a Thanksgiving Dinner, too! MMS will attend, and recommends this event to serious advanced users only. (This meeting should put the final touches on 79-STANDARD FORTH.) Preregistration was requested by Oct. 15th, but ask anyway: FORML Conference, P.O. Box 51351, Palo Alto, CA 94303.

AND FORTH NATIONAL CONVENTION:

Immediately after the FORML get-together and just up the road onto the San Francisco Peninsula, all Forth enthusiasts are invited to attend the second Annual Forth Convention. FIG put on a winning show last year, and this one promises to be even better! MMS will exhibit and give talks and so will many others. We hope to meet many of our West Coast friends, so please drop by and say hello! It's Saturday, November 29th, 1980 at the Villa Hotel, San Mateo, CA - 8:30 a.m. to 4:30 p.m., $4.00 preregistration. 7:30 p.m. dinner with speaker Alan Taylor (The Taylor Report, Computer World), $15.00 preregistration. For action or more information, contact: Forth Interest Group, P.O. Box 1105, San Carlos CA 94070 (FIG's general information contact is Roy Martens, 415/962-8653).

NORTHEAST COMPUTER SHOW:

This is the biggest and best show of the year for New England, and MMS will be exhibiting its wares here at our own end of the country. November 20-23, 1980 (Noon-10 p.m. Thurs.-Sat., 11 a.m.-6 p.m. Sun.) in the Hynes Auditorium, Prudential Center, Boston MA, $5.00 admission. Say hello to MMS at Booth 925 or attend our scheduled talks on Forth!

AT MILLER MICROCOMPUTER SERVICES

MMSFORTH UTILITIES DISKETTE AVAILABLE NOW:

MMS has upgraded its popular Floating Point Math/Z80 Assembler diskette to include XREF and ALL-CAPS. XREF, a powerful new cross-referencing routine by Paul van der Eijk, provides ASCII-ordered listings with block and line numbers, for one word or for all words except those you list in "ignore" blocks. Serious Forth programmers should find it as invaluable as we do here at MMS!

Andy Watson's ALL-CAPS routine converts all occurences of lower-case characters to their upper-case counterparts and rewrites these throughout a range of diskette blocks. This will permit the distribution of lower-case information for the growing number of compatible users, as it offers an easy conversion process to avoid the "garbage" display of lower-case on unmodified TRS-80's.

Our new MMSFORTH UTILITIES DISKETTE is just $39.95 plus $1.00 shipping/handling. For an upgrade, send $25.00 and your original FP/Z80 diskette. (That's the usual $10.00 re-write charge, plus the $10.00 price increase, $1.00 S/H, and a $4.00 deposit in case your diskette must be replaced.)

TAPE STILL V1.8:

As of this writing, the MMSFORTH System Diskette is at Version 1.9 but the cassette version is still 1.8. Many V1.9 changes are inappropriate to the cassette format, and extensive MMS support of the tape system is limited by its lower sales volume (caused in large part by the rapid conversion of tape users to disk users). MMS plans further additions to the tape system and will keep you tape users posted as changes are implemented. (Say, can we sell you a disk drive?)

DATAHANDLER CUSTOM MODIFICATIONS:

MMS is very busy customizing THE DATAHANDLER for a wide variety of clients. So far we have produced dedicated versions to do specific tasks involving inventory, payroll, order entry and analysis, repair logging, and more. All out-perform their BASIC counterparts (often spectacularly so!) and were delivered in less time with less expense. In one of its more versatile roles, MMS has even modified THE DATAHANDLER to replace a fleet of IBM keypunch machines with one-disk Model I's for preprocessed high-volume data entry to a minicomputer system!

Other programmers as well as MMS now provide custom modification of THE DATAHANDLER. Typical prices are $500 for a simple system, including all software and service time, $1,000 for a moderately complex project. That's a good profit for a good programmer; for the client, it's a good price, early delivery, and a lot of user satisfaction as well.

Because many of you are expressing interest in this as a personal or commercial activity, MMS intends to explain key internal words and modification procedures for THE DATAHANDLER in an upcoming issue.

DATAHANDLER UPGRADES:

MMS has discontinued the special reduced price for upgrading THE DATAHANDLER V1.0 to V1.1 . Such upgrades now require a $5.00 documentation price in addition to the prior $10.00 for disk rewrite and $1.00 shipping/handling. As usual, include $4.00 for possible diskette replacement; we return this if we needn't use it. (Translation: send your original MMS-labelled DATAHANDLER diskette and $20.00.)

Some users say we haven't emphasized this major upgrade from V1.0 as much as it deserves. We thought we had, although our most glowing description was trimmed from a full Newsletter #3. We agree with them - upgrading THE DATAHANDLER will prove well worthwhile!

STOP MASHING THOSE DISKETTES!:

MMS asks you to package your diskettes carefully before shipping to us. Tight-wrapped layers of cardboard in envelopes often fail, and "Floppy Armour" doesn't always survive the stamping (and stomping?) process at the Post Office. Follow our recommendation in the MMSFORTH and DATAHANDLER instructions: seal your original MMS-labelled diskette and its diskette envelope in a baggie, then "float" it on tissue in a box. It may be worth $4.00 to you and save us a hassle, to boot!

AND DON'T TAPE YOUR CASSETTES!:

Don't use tape to stick your cassette to a backing, etc., without first enclosing it in a box or baggie. The recording tape itself occasionally gets sticky due to careless packaging. MMS won't put such a cassette into our recorders and you shouldn't, either!

USING REAL-TIME INTERRUPTS IN MMSFORTH:

by MMSFORTH author Tom Dowling

The current version of MMSFORTH for TRS-80 runs with the interrupt system off. The disk routines are not configured to work with the interrupt system on; therefore interrupts must be disabled before any disk operations may take place.

Interrupt processing requires three routines. The first handles the necessary interrupt processing. The second initializes the interrupt link and any required interrupt routine initialization. The third disables interrupt processing. Here is an example for advanced programmers.

WORD	STACK	DEFINITION
CLK	(-> n)	Constant containing address of time buffer (4-byte buffer)
INTERRUPT	(-> n)	Constant containing address of interrupt latch (status)
FDC	(-> n)	Constant containing address of floppy disk interrupt register
INTAD	(-> n)	Constant containing address of interrupt link
CLOCK	(-> n)	Leaves address of clock interrupt processing routine on stack
OCLOCK	(->)	Resets time to zero
CLOCKON	(->)	Turns clock processing on (56ED req'd in some ROMs to set Interrupt Mode 1)
CLOCKOFF	(->)	Turns clock processing and interrupt system off
TIME	(->)	Prints current time in clock buffer

BLOCK : 92

```
( TIMER ROUTINE, TD/MMS, 7/80 )              : TASK ;
HEX       405E CONSTANT CLK          37EC CONSTANT FDC
          4012 CONSTANT INTAD        37E0 CONSTANT INTERRUPT
LABEL CLOCK  HL PUSH  PSW PUSH  INTERRUPT LDA  A L MOV  40 ANI
   #0 IF FDC LDA
      ELSE L A MOV  80 ANI
         #0 IF  CLK LHLD  HL INX  CLK SHLD  H A MOV  L ORA
                 =0 IF  CLK 2 + LHLD  HL INX  CLK 2 + SHLD  THEN
      THEN  THEN  INTERRUPT LDA  PSW POP  HL POP  EI  RET
CODE OCLOCK   0 HL LXI  CLK SHLD  CLK 2 + SHLD  NEXT
CODE CLOCKON  DI  C3 A MVI  INTAD STA CLOCK HL LXI INTAD 1+ SHLD
              56ED ,  EI  NEXT
CODE CLOCKOFF  DI  C9F3 HL LXI  INTAD SHLD  NEXT
DECIMAL
: TIME  CLK 2 + @ CLK @ 40 M/MOD . " ." 25 * <# # # # #> TYPE ;
: TIMER  CLS CLOCKON  BEGIN  8 25 PTC  TIME  0 END ;
```

SHARE YOUR DATABASE WITH OTHERS

Here's a new idea which may catch on among user groups and individual MMSFORTH fans. Users of THE DATAHANDLER are busily compiling data into files and some of these should be of general value to many others. Why not share these as a common data bank? MMS has started the ball rolling by putting one of its own files, ASSEM, aboard on DATAHANDLER V1.1 diskettes. (ASSEM is a very useful comparison table of Assembler instructions in Z-80, Forth Z-80, Forth 8080, plus the appropriate reference page for each instruction in Radio Shack's Editor/Assembler Manual. We created it as an easy way to print the tables in our Z80 Assembler documentation.)

Perhaps you have built a file of all last year's articles in one of the computer magazines, all computer bulletin board systems in the country, all Radio Shack TRS-80 catalog items, etc. Others could use it, and might reciprocate by developing additional entries. Let MMS know what you've got, and we'll start our DATAHANDLER file of YOUR DATAHANDLER files to share with others! For starters, tell us what you've got, how much RAM each file occupies, plus your name, address, phone and MMSFORTH Serial Number. Are you interested?

READ-ONLY PROGRAM FOR DATAHANDLER FILES, AND MMSFORTH GLOSSARY QUICK-REFERENCE FILE

If you own THE DATAHANDLER you have a great tool for creating data files for flexible use. Now there is a new program in Disk MMSFORTH that can be used without THE DATAHANDLER to review files created by it. It's designed for broad read-only distribution of your data files. This READ-ONLY program was written for the MMSFORTH Users Group of Eastern Massachusetts by Jim Gerow after a suggestion by Gene Walenciewicz. That group is using it to provide QGLOSS, a MMSFORTH Quick-Reference Glossary file which can be looked up by functional grouping, by word-name, etc. Owners of THE DATAHANDLER are equipped to modify this file, as well. MMS has helped and thinks well of this user group project. Interested users of MMSFORTH can get a diskette with the READ-ONLY program and the MMSFORTH Quick-Reference Glossary file by sending Jim Gerow (see User Groups listing) $11.00 with your name, address, MMSFORTH Serial Number, and whether you want the lower-case version or the upper-only version of QGLOSS.

MMSFORTH QUICKIES

STACK-DUMP, REVISITED:

In MFNL1:3 we shared a nifty non-destructive stack-dump word, .S . Now redefine it to print with top-of-stack always on the right, more in keeping with Forth convention. (The S0 definition remains as before.)

```
: .S 'S S0 - IF 'S S0 DUP ROT DO 2 - DUP ? 2 +LOOP DROP THEN ;
```

Here's another word which makes .S even handier for learning Forth!

```
: TEST  CR .S "  ->" ', ' EXECUTE .S CR ;
1 2 3 4 (Enter)
TEST SWAP (Enter) ...etc.
```

Remember to put appropriate items on stack first; stacks may be cleared by hitting Break.

COUNT:

Use this beginner-level routine to experiment with DO-LOOPs, +LOOP steps, CONSTANT and VARIABLE, etc.

```
: COUNT  0  CR DO  I .  LOOP  CR ;
```

Now 10 COUNT displays the numbers 0 through 9. Note that ten items were counted but unlike BASIC the final (eleventh) item, 10, caused conclusion of the LOOP operation without being printed.

Let's redefine COUNT as follows.

```
: COUNT   0  CR DO  I .  2 +LOOP  CR ;
```

Calling 10 COUNT now shows 5 numbers, stepping by 2 from 0 though 8. Again, the terminating 10 is not displayed.

Let us remove these two definitions of COUNT by entering FORGET COUNT FORGET COUNT . Then define it again, but first initialize a variable for the step.

```
2 VARIABLE STEP
: COUNT   0  CR DO  I .  STEP @  +LOOP  CR ;
```

(@ means "fetch the contents of this address".) 10 COUNT performs as before, but now we can easily vary the step as follows.

```
5 STEP !    100 COUNT
```

The above operation, of course, stored the 5 as the new contents of STEP .

In Forth, constants can also be redefined. Let's initialize one and see.

```
FORGET STEP
5  CONSTANT  STEP
: COUNT   0   CR DO  I .  STEP +LOOP  CR ;
100 COUNT
2  ' STEP  !
```

(' STEP means "address of STEP".) 10 COUNT

Defined as a constant instead of a variable, the value is easier to call to stack; however, it is less efficient to change it. As a variable, calling STEP got the RAM address; as a constant, it gets the content of that address. So in Forth, the use of constants and variables optimizes differently but the two perform the same tasks. As a rule of thumb, if you will be changing the value a lot more than you are calling it, make it a variable; otherwise a constant (see table below). But first, consider just entering it on stack. If you don't require maximum clarity of Forth source blocks, this is usually most efficient of all!

GET-TOGETHER

Consider sharing your questions and answers with a MMSFORTH User Group, or contact MMS for help in starting one in your metropolitan area. Here is our present list of contacts for local MMSFORTH User Groups:

Morris Herman, 503 Rosario Drive, Santa Barbara CA 93110 (805/964-7144).
Rich Royea, 6456 Lubau, Woodland Hills CA 91367 (213/704-6859).
Ed Laughery, 1222 Jason Drive, Denham Springs LA 70726 (504/665-7537).
Jim Gerow, 1630 Worcester Road, Framingham MA 01701 (617/443-9521 x3562 days, 617/872-1882 eves.).
Kim Watt, Box 1013, Berkeley MI 48072 (313/288-9422).
Bob Zwemer, 6408 South Washington, Lansing MI 48910 (517/393-9287).
Larry Goforth, 10203-J Golden Meadow, Austin TX 78758 (512/836-0981).
Jim Shepard, 16210 Arbor Downs Drive, Dallas TX 75248 (214/661-9702).
Paul van der Eijk, 5480 Wisconsin Avenue #1128, Chevy Chase MD 20015 (301/656-2772).
Rod Proctor, 13520 N.E. 29th Place, Bellevue WA 98005 (206/885-4171 days, 206/883-1923 eves.). Rod also is on THE SOURCE.

NOTE: Program trading is one popular facet of these meetings, but NOT commercial programs and WITHOUT MMSFORTH SYSTEMS aboard! Promote legitimate sharing, discourage pirating, and take care not to jeopardize your own MMSFORTH serial number.

SETTING FORTH (for beginners)

ON INTEGER ARITHMETIC VS. FLOATING-POINT MATH:

In this issue we will explore some double-precision integer arithmetic operations. But first, a few thoughts about integer vs. floating point arithmetic.

Single-precision integer math is sufficient for most purposes and where it fits, it fits smallest, fastest, easiest and without floating-point's inherent round-off error. For most business accounting applications and the like, single-precision's values from 0 to 65535 or from -16767 to 16767 won't hack it, but double-precision will count the pennies across a range of better than +/- $21,000,000.00 . (If your assets exceed this amount, you can afford to hire MMS to add triple-precision routines for your use!) Although that decimal point isn't printed out by the double-precision operation, it is easily added to the output with an appropriate word, as has been done in the Checkbook Balancing program on your MMSFORTH System Diskette or Cassette. Again, double-precision integer math outperforms floating-point in compactness, speed and lack of round-off error.

Unlike most versions of Forth, MMSFORTH does offer a floating-point package as an option. This option leans heavily on your TRS-80's Level II BASIC ROM to save space and speed (it's fast machine code and it's already aboard, so no extra RAM). For numbers which vary over a wide dynamic range, think scaling for advanced users or floating-point for beginners. For more usual applications, MMS strongly recommends you get to meet your double-precision instruction set.

DOUBLE-PRECISION INTEGER MATH IN MMSFORTH
by MMSFORTH Author Tom Dowling

A double-precision number is entered in MMSFORTH by including a decimal point in the numeric input. Thus, 5 places a single single-precision element on the stack, while 5. places a pair of elements on the stack. The double-precision representation has the top-of-stack equal to the least significant 16 bits of the number and the second-on-stack is the most significant 16 bits. Actually, all numbers entered in MMSFORTH are remembered as double-precision quantities. The value of the byte variable #PT indicates if one or two values have been pushed on the stack. If #PT = 0 then one value has been pushed on the stack because no decimal point was included in the input; the variable HI# contains the high order bits of this number and can be used to check for overflow or to create a double-precision entry by HI# @ SWAP . If #PT <> 0 then two values have been pushed on the stack and #PT indicates the position of the decimal point in the input. #PT=1 if the decimal point was the last character input, =2 if next to last, etc. Thus, 5. has #PT=1; 5.7 has #PT=2.

Double-precision variables are defined by putting a initial double-precision value on the stack and using the word DVARIABLE to enter it into the dictionary:
5. DVARIABLE XX

Likewise a double-precision constant is created by putting its value on the stack and using DCONSTANT to enter it into the dictionary:
5. DCONSTANT FIVE

DOUBLE-PRECISION OPERATIONS:

Stack inputs and outputs are shown with top of stack on right.

OPERAND KEYS:	d,d1,...	double prec. #'s
	n,n1,...	single prec. #'S
	ub1,ub2	unsigned byte #'s.
	un1,un2	unsigned single prec. #'s.
	ud1,ud2	unsigned double prec. #'S.
	ut1,ut2	unsigned triple prec. #'S.
	f,f1,...	flag: true=1, false=0.
	addr	address

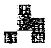
Word	Stack	Action
M+	(d1 n1 -> dsum)	add single to double.
D+	(d1 d2 -> dsum)	add double to double.
M-	(d1 n1 -> ddif)	subtract single from double.
D-	(d1 d2 -> ddif)	subtract (d1-d2).
M*	(n1 n2 -> dprod)	multiply single ans. double.
M/	(d1 n1 -> nquot)	(d1/n1) single result.
M/MOD	(d1 n1 -> nq nr)	(d1/n1) single quot + remainder.
M*/	(d1 n1 n2 -> d2)	(d2=d1*n1/n2)
DU/	(ud ub -> dq nr)	(ud/ub) d quot. s remainder.
DU*	(ud ub -> ud1)	(ud1 = ud*ub).
D*S	(ud ub -> ut)	(ut=ud*ub).
T/S	(ut un -> ud)	(ud=ut/un).
DABS	(d1 -> d2)	d2= abs(d1).
DMINUS	(d1 -> d2)	d2=-d1.
DO=	(d -> f)	f=1 if d=0. else f=0.
DMIN	(d1 d2 -> d3)	d3= minimum of d1 or d2.
D@	(addr -> d1)	fetch double contents.
D!	(d addr ->)	store double quantity.
2DUP	(d -> d d)	duplicate dquantity on stack.
2DROP	(d ->)	throw away double quantity.
2SWAP	(d1 d2 -> d2 d1)	reverse top 2 double items.
2OVER	(d1 d2 -> d1 d2 d1)	make copy of 2nd double on stack.
DCONSTANT x	(d ->)	create double constant x with value d.
DVARIABLE y	(d ->)	create double variable y with initial value d.
D#	(d -> d1)	converts least significant digit of d to ASCII and places on output string. d1=d2/(base).
D#S	(d -> 0.)	converts remaining digits of d to ASCII and places on output string.
D.	(d ->)	print double number.
D.R	(d ub ->)	print d right justified in fld ub.
D#IN	(->)	input double number from keyboard; if number does not have "." convert to double.

FUN & GAMES

THE SECRET VIKING, AND A SEAT-EJECTOR:

A mysterious Viking face has been discovered lurking in Generation #9 of a rather plain pattern in MMSFORTH's Game of Life. Conduct your own investigation to determine whether Norsemen beat Columbus to, umm, would you believe to Fort Worth? Create a symmetrical hour-glass-like pattern by placing the Life dot at 4,4 and drawing a diagonal to 32,32. Then draw left to 15,32, diagonal to Line 4, and back to the start. Press G to go, and keep your eye peeled for the mystery warrior!

In the same run, a pair of gliders (repetitive moving pattern sequences) appears in Generation 31 and moves toward top center until about Generation 60. To observe one of these "seat-catapult" gliders in detail, clear the LIFE screen and draw the pilot and his seat in the lower left:



"G" is the seat-eject button for this remarkable pentomino (5-cell) pattern!

LET THERE BE MUSIC!:

Arnold Schaeffer, the high-school student who wrote BREAKFORTH, now sends MUSIC. Arnold notes, "It is expandable by simply adding a second scale or have another modifier that will vary the wavelengths and frequencies or music can be transposed the same way. I did this program using wave theory and simple physics as I have little musical capability." Follow our earlier Breakforth instructions to listen with a speaker and your cassette recorder. At MMS, we've already added the Rest capability, high-C, a few bars of The Blue Danube Waltz, and a live keyboard. Does anyone out there care to tune up this MUSIC routine further and play Arnold a tune?

BLOCK : 90

```
( MUSIC BY ARNOLD SCHAEFFER/MMS )
0 VARIABLE N/R  ( HEX 0102 = NOTE & 0202 = REST )
CODE MUSIC  DE POP  HL POP  BC PUSH   ( ED57 INSTR = 9 STATE NOP
  HEX     BEGIN  N/R 1+ LDA  FF OUT  H B MOV L C MOV  0060 CALL
  57ED , 57ED ,      N/R LDA  FF OUT  H B MOV L C MOV  0060 CALL
  DECIMAL       DE DCX  D A MOV  E ORA  =0 END  BC POP NEXT

: NOTE <BUILDS , C,
        DOES> DUP 2+ C@ SWAP @ ROT / 258 ( 0102H ) N/R ! MUSIC ;
: R  ( REST )  132 262 ROT / 514 ( 0202H ) N/R ! MUSIC ;

132 262 NOTE C  124 277 NOTE C# 117 294 NOTE D  111 311 NOTE D#
104 330 NOTE E   99 349 NOTE F   93 370 NOTE F#  88 392 NOTE G
 83 415 NOTE G#  78 440 NOTE A   74 446 NOTE A#  70 494 NOTE B
 66 524 NOTE C'

```

BLOCK : 91

```
( MUSIC & KEYBOARD BY MMS )

: DANUBE 4 C  4 C 4 E 4 G  4 G 4 R 4 C'  4 C' 4 R 4 G  4 G 4 R ;

: KEYBOARD   BEGIN 0 4 KEY ACASE ↑1Q2WE4R5T6YU"
   C C# D D# E  F F# G G# A A# B  C'  CASEND
  DUP IF DROP THEN END ;

```

MMSFORTH MODIFICATIONS

GAME OF LIFE (CASSETTE VERSION):

User Alan Wolf of Austin, Texas reports a problem common to all copies to date of the MMSFORTH System Cassette's LIFE/DOODLE demo program. If you attempt to store a pattern to tape, screen prompts guide the operation. Unfortunately, the pattern stored to tape will include these prompts! Jill Miller of MMS provides this mod within Block 52 which corrects the problem by moving the screen data up to PAD before the prompts, then returning it after the tape-writing operation:

```
 9 ELSE 1 THEN END DUP 48 >= OVER 57 <= AND IF DUP ECHO 48 - SWAP
10 10 * + 0 ELSE DROP 1 THEN END ;  : %L %NB RBLK %CC ;  : %S %NB
11   SWAP PAD 1024 MOVE  PAD SWAP WBLK %CC  PAD SCREEN 1024 MOVE ;
```

AVAILABLE RAM:

MMS, like Volkswagen, quietly makes many changes between official revisions. On our latest MMSFORTH V1.9 diskettes, a DIR also reports the remaining free RAM. The end of Block 11 now LISTs like this:

```
12 CR " BREAKFORTH  REAL-TIME GRAPHICS GAME WITH SOUND"
13 CR 12 SPACES " ** "  'S PAD -  <# #S #>  TYPE
14  "  BYTES AVAILABLE **"  16 SPACES
15
```

This clever choice of location (suggested by new MMS programmer John Rible) results in absolutely no additional overhead because in V1.9 the DIR menu screen is not compiled into RAM. Try it, you'll like it!

SHIFTY WORDS:

We've quietly moved two Version 1.9 MMSFORTH System words down to where a FORGET SCR (to drop the EDITOR) won't drop them, too. ALLOT has moved from Block 23 to Block 17, and DIR has moved from 23 to 21. New diskettes and new Glossaries will note this.

More subtly, we've relabeled the old code definition of PCRT to PCRT1 . Then later in Block 19 we've redefined PCRT as

```
: PCRT  CRT PCRT1 ;
```

This guards against surprises when calling PCRT twice without a CRT between, but still allows the use of PCRT1 to address two or more outputs with MMSFORTH's virtual I/O.

Other changes in the very latest copies of V1.9 include reset of number of tracks/disk by the CUSTOMIZE routine, and a moderate neatening of those tight and hard-to-read MMSFORTH source blocks. Upgrade diskettes are available as usual, $10.00 plus $1.00 shipping/handling (foreign orders extra). Except for recent copies, we strongly recommend new MMSFORTH Instructions at the same time, for $8.00 additional.

MISUSE OF VOCABULARIES:

We plan to discuss the topic of vocabularies, the subsets of Forth's dictionary, in a future issue. Meanwhile, here are two warnings from present experience.

The first will require a minor patch for most users of MMSFORTH. If your MMSFORTH System Diskette or Cassette was manufactured prior to October 1980, examine Block 31 Line 15. If it doesn't have the word FORTH at the right, use the editor to add it now! Why? When you load the strings routines which are on Blocks 29-31, the context vocabulary becomes ASSEMBLER and at the end we have neglected to switch it back to FORTH. In most instances this doesn't matter: a DIR or a new colon definition will reset to FORTH as would any subsequent routine which calls FORTH . And while in ASSEMBLER, the Forth vocabulary is still available anyway, as soon as the word is not found in the Assembler vocabulary. The problem occurs only when the word is found in the Assembler vocabulary. For a nasty example of this, consider our Glossary example for $VARIABLE , without FORTH on Block 31 Line 1. 20 $VARIABLE AB works fine - prove it by entering this and then defining $L NAME-AB" AB $! AB $. . Then repeat the same activity with a new variable name, A . It will fail upon print-out. This rather unlikely and undesirable problem occurs because we didn't first return to Forth and a different definition of A happens to occur in the Assembler vocabulary which we searched first. We're sorry, you're warned, now make that change!

The second problem is to be avoided rather than fixed. Trouble can develop quickly when FORGETting a vocabulary such as EDITOR. FORGET must address a word in the CURRENT vocabulary, not the CONTEXT vocabulary. In this case, EDITOR itself is in the FORTH vocabulary so one must be in FORTH in order to say FORGET EDITOR. More on this later, just this "advance warning" for now.

MODIFICATION TO DATAHANDLER MODIFICATION DEPT.:

Yes, I KNOW we said 512 in the last issue, but go back to THE DATAHANDLER source diskette, Block 25 Line 0, and change those three 512's to 256's instead. Also make the same change on Line 8 of Block 13. MMS finds its prior recommendation was overly conservative, as no DATAHANDLER operation at PAD which also used a routine at PAD + 512 could be larger than 255 bytes anyway. With this change, 300 bytes of available RAM will suffice for normal applications rather than the previously recommended 650.

ON REVERSE POLISH NOTATION

'RPN stands for "reverse Polish notation" and the "Polish" refers to Jan Lukasiewicz (1878-1956), a great Polish logician and mathematician. "Polish" is easier to pronounce than "Lukasiewicz" (wu-ka-sha'-vich) and "reverse Polish" is much easier to pronounce than "Zciweisakul".'
-From "Algorithms for RPN Calculators", by John A. Ball, John Wiley & Sons, New York, 1978. MMS recommends this book as a source of many mathematic algorithms, optimized for RPN users.

THE LAST WORD: "What do you mean, I'm forthing at the mouth?"
- rabid Forth enthusiast

MMS
MILLER MICROCOMPUTER SERVICES
61 LAKE SHORE ROAD, NATICK, MASS. 01760
(617) 653-6136