MILLERS GRAPHICS

THE SMART PROGRAMMER

VOLUME TWO

A lot certainly has happened since the last newsletter. I'm quite sure that TI's announcement took many of us by surprise and that by now you have heard all the rumors. I won't go over all of them but here are a few that are worth mentioning.

TOYS R US and/or GE are apparently trying to obtain the rights to manufacture the TI 99/4A consoles, software and maybe the peripherals.

The International Users Group is trying to obtain the rights to Extended BASIC and the Editor/Assembler command modules.

The 99'er magazine might be adding the IBM, Commodore and Atari computers to its' ges.

A California company by the name of CorComp Inc. is manufacturing a 32K memory expansion card, an RS232 card, a double sided double density Disk Controller card and a new peripheral expansion box. The cards have a 1 year warranty and are FULLY compatible. This same company is working on a replacement for the 99/8 computer with many more enhancements over its' unreleased TI predecessor. From what we have heard so far about CorComp's 99/64, which is what it may be called, it sounds like it is going to be an extremely nice computer for a very reasonable price. We will be sure to publish more about CorComp's products in future issues. Interested dealers may contact CorComp Inc., 23461 Ridge Route Dr., Suite H, Laguna Hills, CA 92653 or phone 714 855-8033.

If you are interested in buying some of these cards contact your local dealer to see when they will have them in stock. The 32K card has a suggested retail price of 119.95, the RS232 is 79.95 and the disk controller ould be around 119.95

A local dealer to us has latched on to one heck of a deal for color monitors. He has a very nice 13 inch color monitor that includes the cable for the TI 99/4A and it is only 224.95 plus shipping. He also has a working light pen for your TI 99/4A that he is inviting inquires on. We just found out about the light pen so we are uncertain about the price, but if it is like the rest of his products I'm sure you'll like the price. If you are interested contact Ron Wells at 5523 San Jose, Montclair, CA 91763 or call after 4:00 PM P.S.T. (714) 983-2878.

Millers Graphics will be publishing a monthly 16 + page newsletter FULL of tips, tricks and many undocumented items of interest about the 99/4A and its' peripherals. We've been working on detailed memory maps of ROM, RAM, VDP RAM and GROM which we will be publishing. We have also looked at the disk controller card and memory dumps of the various command modules, through which we have discovered many interesting items that we will be sharing with you in future issues.

Along with sections on getting the most out of Extended BASIC, there will be columns on TI Forth, the TI PC computer and a question and answer section where we will try to answer all the questions you've had but did not know who to ask.

The items slated for the first issue, which will be mailed at the end of January are; The start of a continuing series on the detailed memory maps and what you can do with the various addresses (PEEKS and LOADS). How to change the screen and/or the text color in the immediate or program entry mode, without memory expansion! How to get into Multicolor and Text modes from Extended BASIC! Answers to some of the questions we have received regarding Extended BASIC programming. An overview of TI Forth and a nice little program that is written in Extended BASIC that lets you draw on the

screen in all the colors, using the joystick will also be in the first issue.

Yes, we are still going to publish the book we promised in the last newsletter. TI's announcement and some new discoveries that we want to add to the book have placed us behind schedule. We will be sending out full color brochures about the book as soon as it is complete and ready to ship.

Please keep sending in the questions they help us with on future topics for the newsletter!

That is enough editorial lets move on to programming and the other fun stuff.

Here is a new address for you to play with >83C2, it is located in the 256 byte area of memory know as Scratch Pad Ram. This address is know as the 'Interrupt Flag' and it is a 1 byte bit mapped address.

If you have memory expansion you can turn the bits on and off by using the following CALL LOAD's. Don't forget about CALL INIT.

Bit	Use
0	On disable all of the following
1	On disable Sprite motion
2	On disable Auto Sound processing
3	On disable the QUIT key (FCTN =)
4-7	not used

>83C2 = decimal 33730
33730-65536 = -31806

CALL LOAD(-31806,128) = bit 0 on
CALL LOAD(-31806,64) = bit 1 on
CALL LOAD(-31806,32) = bit 2 on
CALL LOAD(-31806,16) = bit 3 on
CALL LOAD(-31806,0) = All bits off

CALL LOAD(-31806,48) = bits 2 & 3 on
CALL LOAD(-31806,80) = bits 1 & 3 on
CALL LOAD(-31806,96) = bits 1 & 2 on

If you type in CALL INIT :: CALL LOAD (-31806,16) before you start programming you won't have to worry about FCTN QUIT ever again! If you disable the Auto Sound processing after a CALL SOUND statement the sound will stay on forever but don't execute another CALL SOUND until you turn off bit 2 or you will lock up your computer and you will have to turn it off & on to regain control, this includes the beep and honk sounds the computer generates for inputs and warning sounds.

Listed below is a little program that turns Sprite motion on and off. Even though auto Sprite motion is off all the other sprite related statements still work, such as CALL COINC, CALL LOCATE etc. After you have run the program change the color parameter in the CALL SPRITE to 7 instead of I+2.

```
100 CALL INIT :: CALL SCREEN
(2):: CALL LOAD(-31806,64)::
 CALL MAGNIFY(4):: CALL CHAR
(64,RPT$("F",64))

110 A=10 :: FOR I=1 TO 10 ::
 CALL SPRITE(#I,64,I+2,I*16,
110,0,I*A):: NEXT I

120 RANDOMIZE :: CALL PEEK(-
31880,B):: IF B<25 THEN CALL
 LOAD(-31806,64):: A=-A :: C
ALL COINC(#1,#2,30,C)ELSE 12
0

130 CALL SOUND(-100,660,(NOT
 C)*-30):: FOR I=1 TO 10 STE
P 2 :: CALL MOTION(#I,0,I*A,
#I+1,0,I*-A):: NEXT I :: CAL
L LOAD(-31806,0):: GOTO 120
```

You can also stop sprite motion by loading -31878 with 0 and then start them back up by loading -31878 with the highest numbered sprite that is in motion. Both -31806 and -31878 work but -31806 brings them to a stop a little faster, on the other hand, by using -31878 you can selectively stop the higher numbered sprites and keep the lower numbered sprites in motion.

Since we are going to be doing a lot of talking about PEEK's in future issues of The Smart Programmer here are a few tips on what all those numbers are in memory.

Our computers can access an address as either a byte or a word. A byte is a string of binary digits (bits). In the majority of cases, including the TI 99/4A, a byte is 8 bits long. A 6-bit byte is commonly called a character, not to be confused with the characters that are printed on the screen. A 4-bit byte is usually called a digit or nibble. Our bytes are made up of 8 bits and a word is 2 bytes or 16 bits. A bit can be either on, equal to 1, or off, equal to 0. In a byte, the bits are numbered 0 through 7, with bit 0 being the most significant bit

(msb) or highest value bit and 7 being the least significant bit (lsb) or lowest valued bit.

0	1	2	3	4	5	6	7
128	64	32	16	8	4	2	1

A byte can also be split in half to form a nibble or 4 bits, such as it is for the Hex codes in the character definitions.

00000111 in binary equals 7 in decimal and 07 in hex. 00001110 equals 14 in dec. and 0E in hex. 00011100 equals 28 in dec. and 1C in hex.

When you execute CALL PEEK(xxxx,A,B), both A and B contain one byte in decimal form. A and B will always be a value between 0 and 255 since 255 is the largest value that one 8 bit byte can contain. 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255. To convert this into a word multiply A times 256 and add B to it.

The following program is a number converter. You can input a number in decimal, hexadecimal or binary and the program will return the number in the other two number bases. It will also return the PEEKable decimal address if the number is larger than 32767 as well as displaying the number as a word and split into two bytes. The conversion formulas were written as subroutines so that you could easily use them in any of your other programs.

Here are the inputs, printouts and conversion formulas listed by line number.

130 Decimal input, range checking and subroutine execution.
140 Hexadecimal input, range checking and subroutine execution.
150 Binary input, range checking and subroutine execution.
160 Decimal printout (word and bytes).
170 Hexadecimal and Binary printout.
180 Hex to Decimal conversion.
190 Binary to Decimal conversion.
200 & 210 Decimal to Binary conversion.
220 & 230 Decimal to Hex conversion.

```
100 ON WARNING NEXT :: CALL
CLEAR :: H$="0123456789ABCDE
F" :: PRINT "DEPRESS YOUR AL
PHA LOCK KEY": :"PRESS LETTE
R FOR INPUT BASE": :

110 PRINT : :"D=DEC #   H=HE
X #   B=BIN #": : :: CALL SO
UND(80,660,6)

120 CALL KEY(0,K,S):: IF S<1
 THEN 120 ELSE ON POS("DHB",
CHR$(K),1)+1 GOTO 110,130,14
0,150

130 INPUT "DEC #=":DEC :: IF
 DEC<-32768 OR DEC>65536 THE
N 130 ELSE A,DEC=INT(DEC-655
36*(DEC<0)):: GOSUB 200 :: G
OSUB 220 :: GOTO 160

140 PRINT "HEX #=" :: ACCEPT
 AT(23,7)BEEP SIZE(4)VALIDAT
E(H$):HEX$ :: GOSUB 180 :: G
OSUB 200 :: GOTO 160
```

GOTO page 4

```
150 PRINT "BIN #=" :: ACCEPT
 AT(23,7)BEEP SIZE(16)VALIDA
TE("10"):BIN$ :: GOSUB 190 :
: GOSUB 220 :: GOSUB 210

160 A=INT(DEC/256):: PRINT :
"D=";DEC;TAB(12);A;DEC-A*256
 :: IF DEC>32767 THEN PRINT
"  ";DEC-65536

170 PRINT "H= ";HEX$:"B= ";S
EG$(BIN$,1,8)&" "&SEG$(BIN$,
9,8):: HEX$,BIN$="" :: A,DEC
=0 :: GOTO 110

180 HEX$=SEG$("0000",1,4-LEN
(HEX$))&HEX$ :: FOR I=1 TO 4
:: A,DEC=DEC+(POS(H$,SEG$(H
EX$,I,1),1)-1)*16^(4-I):: NE
XT I :: RETURN

190 FOR I=1 TO LEN(BIN$):: D
EC=DEC-2^(I-1)*(SEG$(BIN$,(L
EN(BIN$)+1-I),1)="1"):: NEXT
 I :: RETURN

200 A=A/2 :: BIN$=STR$(-(A-I
NT(A)<>0))&BIN$ :: A=INT(A):
: IF A THEN 200

210 BIN$=SEG$(RPT$("00",8),1
,16-LEN(BIN$))&BIN$ :: RETUR
N

220 A=DEC+65536*(DEC>32767)

230 HEX$=SEG$(H$,(INT(A/4096
)AND 15)+1,1)&SEG$(H$,(INT(A
/256)AND 15)+1,1)&SEG$(H$,(I
NT(A/16)AND 15)+1,1)&SEG$(H$
,(A AND 15)+1,1):: RETURN
```

MILLERS GRAPHICS
1475 W. Cypress Ave.
San Dimas, CA 91773

BULK RATE
U.S. POSTAGE
PAID
San Dimas, CA 91773
PERMIT NO. 191