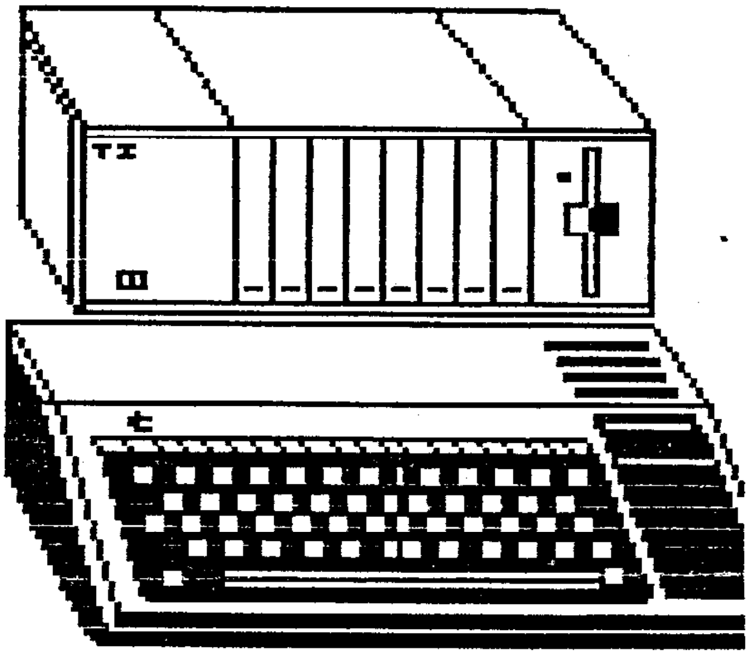
WICHITA AREA 99ERS



NOVEMBER 1985

STAN
I am working on a
catalog of program & Have
in Hand when I Finish IT
I'll send you a copy. Do
you Have any new programs
not listed on your last catalog?
I Have some pretty neet STuFF
From East coast. will Have
to get together and
do some more
Trading

Thanks, Guy Hulsey

The next meeting of the WICHITA AREA 99er's USERS GROUP will be on FRIDAY the 15 th. of NOVEMBER 1985. The meeting will be held at the TRIANGLE BARBER SHOP at CENTERAL AND EDGEMORE. The meeting will start at 7:PM.

SUBJECTS TO BE COVERED THIS MONTH

EXTENDED BASIC--MULTI COLUMN PRINTING AND VARIABLE LENGTH LISTING

ASSEMBLY LANGUAGE--DISK ENITIALIZER

FORTH--A REVIEW

TI-WRITER--USING TI-WRITER CHAPTER 1 GENERAL

NEW PRODUCTS--SUPER WIDGIT

Last month I we held a software swap at my home. Only two people came down but we still had a good time. Several people said they would like to have come down but they had to work on saturday. I am going to hold another software swap meeting, but this time it will be on SUNDAY THE 17th. Of NOVEMBER. I have recived 10 more disks full of programs from the EAST cost, there are some very good programs in this lot.

Every one is invited to come down, bring your blank disk and any programs that you would like to trade.

To find my home go south from Winfield on Highway #77 cross the Walnut river go on South aprox. 1/4 mile. Turn West cross rail road tracks continue west to the second road that goes south (Medowlark Lane) turn South on this road, My house is the first house on the East side of the road. If you can't find it call me at 221-7140 and I'll come lead you in.

MULTI COLUMN PRINTING and VARIABLE LENGTH LISTING by George F. Steffen

For the past several months I have used two programs to list programs in our newsletter 28 characters wide as they appear on the screen and three columns wide so they do not waste space. I received a request for the method and, at the same time, I saw a program to list programs on a wide printer. So i adapted my programs to be more versatile instead of single purpose.

VARYLIST will take a program listing and convert it to whatever line length you desire. There is one bug: if the listed line is an exact multiple of 80 characters in length, the next line will be appended to it. I can think of no simple solution to this and it is an infrequent occurence, so it remains in the program. This program works on a program LISTed to disk. If your desired length is 80 or less, the disk file will be opened as VARIABLE 80 so that it may be edited with TI Writer. If you wish to list to a wide printer, the file will be opened with the correct length.

MULTIPRINT will take a text file and output it to the printer in multiple columns so that it may be read in normal newspaper fashion, one column after another. You determine the number of columns, but you must inform the program of the length of the input text and the length of the output device. This program has no provisions to enable the output text to be edited. Editing must be done before using it.

Before using MULTIPRINT you should prepare your text file. You should first use VARYLIST or the Formatter of TI Writer to create a text file of the desired width. Then examine the file and delete any unneeded blank lines. Make sure that the number of lines is an exact multiple of the number of columns you will be using. Insert blank lines to reach this number. You may put these blank lines any place in the text, but they should be placed so as to form pleasing column breaks. If you have used the text formatter to print the file, you should use the Replace String command to change all Line Feeds (Control U, Shift J, Control U), Carriage Returns (Control U, Shift M, Control U) and New Page (Control U, Shift L, Control U) to spaces. Because the text is reformatted after these changes, be sure you are not in Word Wrap Mode when you do this. If you make the first line of your text longer than the line length you plan to tell the printer, it will print across the page as on this article. In this case, you must be sure that the first two lines of succeeding columns are blank. Then save the text file or print it to disk and run MULTIPRINT. The program is designed to accept 300 lines of text, enough for five columns of 60 lines each. If the number is increased too much, the computer will run out of memory.

The programs are listed herewith, each giving an example of itself. Both programs are available in the club library.

```
100 REM VARYLIST
110 REM THIS PROGRAM WILL CONVERT ANY PROGRAM LISTED TO DISK INTO A LISTING OF ANY WIDTH YOU DESIRE
120 REM IT MAY BE A 28 COLUMN LISTING SIMULATING A SCREEN LIST
130 REM IF LISTED TO DISK AND OUTPUT WIDTH IS 80 OR LESS, OUTPUT MAY BE EDITED WITH TI WRITER
140 REM IF A NUMBERED LINE IS EXACTLY 80, 160, OR 240 BYTES WHEN LISTED, THIS PROGRAM WILL COMBINE IT WIT
H THE FOLLOWING LINE
150 DATA 3,DSK,WDS,RD
160 CALL CLEAR :: PRINT TAB(11);"VARYLIST"
170 PRINT :: LINPUT "NAME OF INPUT PROGRAM LIST? ":IP$
180 PRINT :: LINPUT "NAME OF OUTPUT FILE          ":OF$ :: IF OF$=IP$ THEN PRINT :"INPUT AND OUTPUT NAMES
MUST BE DIFFERENT!" :: GOTO 170
190 PRINT :: INPUT "WIDTH OF OUTPUT FILE?  ":OW :: ODW=OW :: IF OW>79 THEN 220
200 READ N :: FOR I=1 TO N :: READ DN$ :: IF SEC$(OF$,1,LEN(DN$))=DN$ THEN ODW,I=80
210 NEXT I
220 OPEN #1:IP$,DISPLAY ,VARIABLE 80,INPUT :: OPEN #2:OF$,DISPLAY ,VARIABLE ODW,OUTPUT
230 FOR I=0 TO 9999 :: L1$=""
240 IF EOF(1)THEN I=I+10000 :: GOTO 250 ELSE LINPUT #1:L2$ :: IF LEN(L2$)=0 THEN GOTO 240 ELSE L1$=L1$&L2
$ :: IF LEN(L2$)=80 THEN GOTO 240
250 FOR O=1 TO LEN(L1$)STEP OW :: PRINT #2:SEG$(L1$,O,OW):: J=J+1 :: NEXT O :: NEXT I
260 CLOSE #1 :: CLOSE #2 :: PRINT :I-10000;"NUMBERED LINES":J;"OUTPUT LINES" :: END
```

(cont. from page 12)

```
100 REM MULTIPRINT
110 REM TI EXTENDED BASIC AN
D MEMORY EXPANSION
120 REM WILL PRINT MULTIPLE
COLUMNS OF ANY TEXT FILE
130 DIM L$(300):: CALL CLEAR
 :: PRINT TAB(10);"MULTIPRIN
T"
140 PRINT :: LINPUT "NAME OF
 INPUT FILE?          ":IF$ :
: INPUT "LENGTH OF INPUT LIN
ES?      ":LL
150 PRINT :: LINPUT "NAME OF
 PRINTER?             ":P$ ::
 INPUT "PRINTER LINE LENGTH?
  ":PL
160 PRINT :"COLUMN SEPARATIO
NS WILL BE  CALCULATED." ::
INPUT "LEFT AND RIGHT MARGIN
 SIZE? ":M :: INPUT "NUMBER
OF COLUMNS?  ":C
170 IF (2*(M+C-1)+C*LL)>PL T
HEN PRINT "WILL NOT FIT" ::
GOTO 160
180 OPEN #1:IF$,INPUT ,DISPL
AY ,VARIABLE :: FOR I=1 TO 3
00 :: IF EOF(1)THEN 210
190 LINPUT #1:L$(I):: IF ASC
(L$(I))>127 THEN L$(I)="" ::
 GOTO 210 ! DISREGARD TAB SE
TTINGS
200 NEXT I
210 CLOSE #1 :: S=INT((PL-(C
*LL+2*M))/(C-1))+LL :: M=M+1
 :: OPEN #2:P$,DISPLAY ,VARI
ABLE PL+1,OUTPUT
220 N=INT((I-1)/C):: FOR I=1
 TO N :: FOR J=0 TO C-1 :: P
RINT #2:TAB(J*S+M);L$(I+J*N)
;:: IF I=1 AND LEN(L$(1))>LL
 THEN J=C
230 NEXT J :: NEXT I :: CLOS
E #2 :: END
240 STOP
```

<*><*><*><*><*><*><*><*><*><*><*><*><*>

Keyboard Conversion

courtesy TOM FREEMAN

49	50	51	52	52	54	55	56	57	48	61	SHIFT UP
33	64	35	36	37	94	38	42	40	41	43	SHIFT DOWN
1	2	3	4	5	6	7	8	9	0	=	
3	4	7	2	14	12	1	6	15	188	5	FTCN
177	178	179	180	181	182	183	158	159	176	157	CTRL

113	119	101	114	116	121	117	105	111	112	47
81	87	69	82	84	89	85	73	79	80	45
Q	W	E	R	T	Y	U	I	O	P	/
197	126	11	91	93	198	95	63	39	34	186
145	151	133	146	148	153	149	137	143	144	197

97	115	100	102	103	104	106	107	108	59	13
65	83	68	70	71	72	74	75	76	58	13
A	S	D	F	G	H	J	K	L	;	ENTER
124	8	9	123	125	191	192	193	194	199	13
129	147	132	134	135	136	138	139	140	156	13

SHIFT	122	120	99	118	98	110	109	44	46	SHIFT
	90	88	67	86	66	78	77	60	62	
	Z	X	C	V	B	N	M	,	.	
	92	10	96	127	190	196	195	184	185	
	154	152	131	150	130	142	141	128	155	

CTRL	32 32 SPACE 32 32	FCTN

DISK INITIALIZER AND GRAPHPAPER MAKER

by Tim Jobe

Dne of the quickest ways to learn a programming language is to experiment with someone elses programs and see what changes will do. I ran across this super program in the Chicago TImes latest newsletter and couldn't resist keying it in. It s an assembly language disk initializer. The neat thing about this program is the documentation. Most ssembly programs don't have enough o realy help you, but this one loes. It was written by Don Cook of lamilton, Ontario and modified by ohn Moeller of Woodridge, Il.

he version you see listed also ontains modifications of my own. y appologies to Mr. Cook and Mr. Moeller if the changes are too adical, but I couldn't make the ersion in the TImes work. The eferences to screen locations did ot work properly. I changed the creen color to white and added some ext on th screen to help the user. also detected an error in the uplicate byte in VDP ram routine nat caused the routine to only rite zeros to the ram instead of e character desired. I had to nsert a SWPB R1 to get the character (32) into the MSB because the OVB instruction acts only on the SB. Other than that, the program s a dream to work with, if you just ddle with assembly programs like I o.

e program is run using option 3 of A and will auto-start. You first lect the mode (SS/SD, DS/SD, SS/DD DD/DD) with the space bar. Each ess will change the selection to a fferent format. When the desired rmat is shown, press ENTER. You en enter the disk drive number (1 3), then hit any key except QUIT start the process. During the ecking process, the sector numbers e displayed like Disk Manager 2. exit, just press FCTN= (QUIT).

Also included here is another neat program found in the same issue of the Chicago TImes. It will print a sheet of graph paper for you.

Tim

```
100 REM +---------------+
110 REM +GRAPHSHEET MAKER+
120 REM + BY JOHN BEHNKE +
130 REM +               +
140 REM +EPSON OR GEMINI +
150 REM +PRINTER REQUIRED+
160 REM +BASIC OR X-BASIC+
170 REM +---------------+
180 CALL CLEAR
190 INPUT "NUMBER OF SHEETS?
 ":A
200 CALL SCREEN(2)
210 @$=CHR$(27)
220 FOR I=1 TO 228
230 A$=A$&CHR$(128)
240 NEXT I
250 B$=SEG$(A$,1,7)
260 C$=CHR$(255)&SEG$(A$,1,6
)
270 FOR I=1 TO 4
280 FOR J=1 TO 8
290 E$=E$&C$
300 NEXT J
310 E$=E$&CHR$(255)
320 NEXT I
330 F$=@$&"K"&CHR$(484)&CHR$
(0)&E$
340 G$=@$&"K"&CHR$(228)&CHR$
(0)&A$
350 OPEN #1:"PIO.CR"
360 FOR B=1 TO A
370 FOR C=1 TO 11
380 PRINT #1:@$&CHR$(64)&@$&
"3"&CHR$(16)
390 FOR D=1 TO 8
400 PRINT #1:F$;F$;CHR$(10)
410 NEXT D
420 PRINT #1:G$;G$;@$&"3"&CH
R$(2)
430 NEXT C
440 PRINT #1:@$&"3"&CHR$(17)
450 FOR I=1 TO 9
460 PRINT #1:CHR$(13)&CHR$(1
0)
470 NEXT I
480 NEXT B
490 CLOSE #1
500 END
```

```
*****************************************************
* DISK INITIALIZER by Don Cook, Hamilton, Ontario *
* MODIFICATIONS BY JOHN MOELLER, WOODRIDGE,IL     *
* AND TIM JOBE, CORPUS CHRISTI, TX                *
*****************************************************
        DEF  SLOAD
        REF  VSBW,VMBW,DSRLNK,KSCAN,GPLLNK,XMLLNK,VWTR
ENTER   BYTE 13
SPACE   BYTE 32
ONE     BYTE 49
THREE   BYTE 51
ZERO    BYTE 0
D3      BYTE 3
MODTBL  TEXT 'SS/SD '
        BYTE >01,>01
        TEXT 'DS/SD '
        BYTE >01,>02
        TEXT 'SS/DD '
DDSS    BYTE >02,>01
        TEXT 'DS/DD '
        BYTE >02,>02
INIT    DATA >0111
*INITIALIZE DSR SUBROUTINE
RDWRT   DATA >0110
*READ/WRITE DSR SUBROUTINE
SECTO   TEXT 'BLANK        DSK (  '
USE     BSS  200
SAVB    BSS  8
THOU    DATA 1000
HUN     DATA 100
TEN     DATA 10
CHARS   TEXT '0123456789'
TITLE   TEXT 'PRESS ANY KEY TO INITIALIZE    '
        TEXT '      OR (FCTN =) TO QUIT '
TEXT1   TEXT 'DISK'
TEXT2   TEXT 'INITIALIZER'
TEXT3   TEXT 'MODE:SS/SD'
TEXT4   TEXT 'DRIVE: '
TEXT5   TEXT 'SECTOR:    '
TEXT6   TEXT '* WRITING *'
TEXT7   TEXT '  SPACE BAR TO CHANGE MODE     '
        TEXT '        <ENTER> TO ACCEPT  '
TEXT8   TEXT ' ENTER DRIVE NUMBER (1-3)      '
        TEXT '                          '
ERROR   TEXT 'DISK ERROR'
QUIT    BYTE 5
*FCTN = KEY VALUE
        EVEN
MYWS    BSS  32
*CLEAR SCREEN
SLOAD   LWPI MYWS
DTITLE  BL   @DUPBYT
        DATA 768,32,0
        LI   R0,>070F          MAKE BORDER COLOR WHITE
        BLWP @VWTR
        LI   R0,>0384          SET THE COLORS OF THE
        LI   R1,>4F00             CHARACTERS BLUE ON
COLOR   BLWP @VSBW                WHITE
        INC  R0
        CI   R0,>0391
        JLE  COLOR
        LI   R0,45             VDP ADDR
        LI   R2,4                 LENGTH
        LI   R1,TEXT1             TEXT ADDR
        BLWP @VMBW
        LI   R0,106            VDP ADDR
        LI   R2,11
        LI   R1,TEXT2             TEXT ADDR
        BLWP @VMBW
        LI   R0,514            VDP ADDR
        LI   R2,56                LENGTH
        LI   R1,TEXT7             TEXT ADDR
        BLWP @VMBW
        LI   R9,>FFFF             STALL FOR A WHILE
```

```
LOOP1   DEC  R9
        JNE  LOOP1
        LI   R0,202            SET UP TO SELECT MODE
        LI   R1,TEXT3             MODE MSG ADDR
        LI   R2,10                LENGTH
        BLWP @VMBW
        AI   R0,5                 SAVE ADDR OF MODE FIELD
        MOV  R0,R7                IN R7 FOR LATER USE
RSET1   LI   R5,MODTBL            TABLE ADDR
        LI   R6,4                 # OF TABEL ENTRIES
        JMP  WRTMOD               GO WRITE IT
CHKEY1  BLWP @KSCAN               WAIT FOR A KEY PRESS
        MOVB @>837C,R0
        JEQ  CHKEY1
        CB   @>8375,@ENTER        IF ENTER WAS HIT
        JEQ  SETDRV               USE CURRENT MODE
        CB   @>8375,@QUIT         IF QUIT
        JEQ  BYE                  GO BYE BYE
        CB   @>8375,@SPACE        IF SPACE WAS HIT
        JNE  CHKEY1
        DEC  R6
        JEQ  RSET1                END OF TABLE START OVER
        AI   R5,8                 SELECT NEXT TABLE ENTRY
WRTMOD  MOV  R5,R1                WRITE NEW MODE
        LI   R2,5                 TO SAVED ADDR
        MOV  R7,R0                TO SEE IF ITS OK
        BLWP @VMBW
        JMP  CHKEY1               WAIT FOR NEXT KEY PRESS
SETDRV  MOV  @6(R5),R3            MODE IS OK, GO SEE WHAT #
        LI   R0,514            VDP ADDR
        LI   R1,TEXT8             TEXT ADDR
        LI   R2,56                LENGTH
        BLWP @VMBW
        LI   R0,266            VDP BUFFER ADDR
        LI   R1,TEXT4             TEXT ADDR
        LI   R2,7                 LENGTH
        BLWP @VMBW
        AI   R0,6
        MOV  R0,R7                SAVE ADDR TO WRITE DRIVE #
LOOP2   BLWP @KSCAN               SEE WHAT KEY THEY HIT
        MOVB @>837C,R0
        JEQ  LOOP2
        CB   @>8375,@ONE          IF ITS BETWEEN 1 AND 3, PUT
        JLT  LOOP2                INTO THE DRIVE FIELD AND GO
        CB   @>8375,@THREE        IF ITS NOT, THROW IT AWAY
        JH   LOOP2                AND WAIT FOR THEM TO HIT
        MOVB @>8375,R1            THE PROPER KEY
        MOV  R7,R0                OK, WRITE DRIVE TO SCREEN
        BLWP @VSBW
        SRL  R1,8                 TURN ASCII DRIVE #
        LI   R6,48                TO ITS HEX EQUIVALENT
        S    R6,R1                IN THE MSB
        SWPB R1                   FOR THE DSR ROUTINE
        AI   R1,40                # OF TRACKS TO FORMAT
        MOV  R1,R4
        MOV  R3,@>8350
        LI   R0,514            BEFORE WE FORMAT THE
        LI   R1,TITLE             DISK, MAKE THEM HIT A KEY
        LI   R2,56                SO WE DON'T FORMAT THE
        BLWP @VMBW                WRONG DISK
LOOP3   BLWP @KSCAN
        MOVB @>837C,R0            GO ON ANYTHING BUT QUIT
        JEQ  LOOP3
        CB   @>8375,@QUIT
        JNE  DOIT
BYE     BLWP @0                   ITS "QUIT", SCRAM!
DOIT    LI   R0,362            VDP ADDR
        LI   R1,TEXT6             TEXT ADDR
        LI   R2,11                LENGTH
        BLWP @VMBW
*: INFORMATION FOR DSR ROUTINE
        CB   R3,@DDSS
*: >834C HIGH NYBLE 0=SSSD,1=DDSD ETC.
        JNE  TESTDS
```

(CONTINUED ON P. [illegible])

(CONTINUED FROM P.5

```
*:
*LOW NYBLE  1=DRIVE #1 ETC.
       ORI  R4,>2000
*: >834D NUMBER OF TRACKS ON DISK
TESTDG SWPB R3
*: >834E BUFFER LOCATION
       MOV  R3,@SECTO+18 : >834F (MINIMUM 3.2K)
       CB   R3,@DDSS
*: >8350 1 FOR SINGLE DENSITY
       JNE  DTYPE
*:
*2 FOR DOUBLE DENSITY
       ORI  R4,>1000
*: >8351 1 FOR SINGLE SIDED
DTYPE  MOV  R4,@>834C
*:
*2 FOR DOUBLE SIDED
       CLR  R10
*DO ERROR CHECK
       LI   R1,INIT
*INITIALIZE SOUBROUTINE DSR VALUE
       BL   @DSRSUB
       MOV  @>834A,@SECTO+10    TOTAL NUMBER OF SECTORS
       MOVB @>834D,@SECTO+12    SECTORS/TRACK
       BL   @DUPBYT
*PUT ZEROS IN PABBUF FOR SECTOR 1
       DATA 256,0,>1002
       CLR  R4
*WRITE TO DISK
       LI   R3,1
*SECTOR 1
       BL   @DRDWRT
*WRITE ZEROS ON SECTOR 1
       MOVB @D3,@USE
*SHOW SECTORS 0 AND 2 AS USED
       LI   R2,199
*:
       BL   @MEMDUP
*: SHOW REMAINING SECTORS AS USED
       DATA USE+1,>FFFF  :
       MOV  @SECTO+10,R2 TOTAL NUMBER OF SECTORS
       SRL  R2,3
*DIVIDE BY 8
       DEC  R2
       BL   @MEMDUP
*SHOW FREE SECTORS
       DATA USE+1,0
       SETO R10
*REMOVE ERROR DETECTION
       SETO R4
*READ FROM DISK
       LI   R3,2
*START AT SECTOR 2
CHKSEC BL   @DRDWRT
*READ DISK SECTOR
       JEQ  NSECT
*SECTOR SCREWED UP?
       MOV  R3,R1
       MOV  R3,R0
       SRL  R1,3
*DIVIDE BY 8
       LI   R2,>0100
       ANDI R0,7
*CALCULATE SECTOR BIT LOCATION
       JEQ  SUSE
       SLA  R2,0
*SHIFT BIT TO ALIGN WITH SECTOR NUMBER
SUSE   SOCB R2,@USE(R1)  SHOW SECTOR AS USED
NSECT  INC  R3
*NEXT SECTOR
       C    R3,@SECTO+10  LAST STRING?
       JNE  CHKSEC
       LI   R0,>103A
```

```
*BUFFER LOCATION
       LI   R1,USE
*MEMORY LOCATION FOR IN USE INFORMATION
       LI   R2,200
       BLWP @VMBW
       LI   R0,>1002
*:
       LI   R1,SECTO
*: TRANSFER SECTOR 0 DATA
       LI   R2,20
*: TO PABBUF
       BLWP @VMBW
*:
       BL   @DUPBYT
       DATA 36,0,>1016
       CLR  R10
*SET FOR ERROR DETECTION
       CLR  R4
*SET TO WRITE
       CLR  R3
*SECTOR 0
WRITEO BL   @DRDWRT
*WRITE SECTOR 0 ON DISK
       B    @DTITLE
*NEXT DISK
****************************************
* DUPLICATE BYTE IN VDP RAM SUBROUTINE *
****************************************
DUPBYT MOV  *R11+,R2
*NUMBER OF TIMES TO REPEAT BYTE VALUE
DUPBY2 MOV  *R11+,R1
*VALUE TO BE SENT TO VDP RAM IN MSB
       MOV  *R11+,R0
*VDP RAM STARTING LOCATION
       ORI  R0,>4000
*SET TO WRITE TO VDP RAM
       SWPB R1
*GET CHARACTER TO WRITE INTO MSB
       SWPB R0
*LSB FIRST
       MOVB R0,@>8C02
*LSB TO VDP
       SWPB R0
*MSB
       MOVB R0,@>8C02
*MSB TO VDP
SHWBYT MOVB R1,@>8C00
*SEND BYTE TO VDP RAM
       DEC  R2
*NEXT VDP LOCATION
       JNE  SHWBYT
*LAST ONE?
GBACK  RT
********************************
* WRITE TO SECTOR SUBROUTINE *
********************************
DRDWRT MOV  R3,R6      SECTOR #
       CLR  R5
       DIV  @THOU,R5     # OF THOUSANDS
       MOVB @CHARS(R5),@TEXT5+7
       CLR  R5
       DIV  @HUN,R5      # OF HUNDREDS
       MOVB @CHARS(R5),@TEXT5+8
       CLR  R5
       DIV  @TEN,R5
       MOVB @CHARS(R5),@TEXT5+9
       MOVB @CHARS(R6),@TEXT5+10
```

(CONTINUED ON P.7)

(Continued From P.6)

```
       LI   R0,362
       LI   R1,TEXT5
       LI   R2,11
       BLWP @VMBW     WRITE OUT SECTOR #
       MOVB R4,@>834D
*>00 FOR WRITE (>FF FOR READ)
       MOV  R3,@>8350
*SECTOR NUMBER
       LI   R1,RDWRT
*SECTOR READ/WRITE SUBROUTINE VALUE
*****************************
* DSR SUBROUTINE EXECUTION *
*****************************
DSRSUB LI   R0,>1000
*:
       LI   R2,2
*: CREATE PAB IN VDP RAM
       BLWP @VMBW
*:
       MOV  R0,@>8356
*POINTER TO THE PAB
       INCT R0
       MOV  R0,@>834E
*PAB BUFFER LOCATION
       BLWP @DSRLNK
       DATA 10
       MOVB @>8350,@>8350   DISK ERROR?
       JEQ  GBACK
       NOV  R10,R10
       JNE  GBACK
*IGNORE ERROR?
       LI   R0,587
*:
       LI   R1,ERROR
*: SHOW "DISK ERROR"
       LI   R2,10
*:
       BLWP @VMBW
*:
       B    @DTITLE
****************************************
* DUPLICATE BYTE IN MEMORY RAM ROUTINE*
****************************************
MEMDUP MOV  *R11+,R0
*GET MEMORY LOCATION
       MOV  *R11+,R1
*GET BYTE TO DUPLICATE
SETUSE MOVB R1,*R0+
*PUT BYTE AT MEMORY LOCATION
       DEC  R2
*NEXT LOCATION
       JNE  SETUSE
*LAST LOCATION?
       RT
       END  SLOAD
```

NEW PRODUCTS--SUPER WIDGIT

The Osram Industries of Victoria B.C. is making plans to manufacture and market a inexpensive "Super Widgit". Initial plans are for a monster box holding up to 16 cartridges, with each cartridge available from the "Review Module Library option on the title screen. No more troublesome switches. The software looks for only those modules with GROM or a combination of ROM and GROM. Third party modules will not be acessed. The GPL (Graphics programing language) system is designed so that with this hardware, the built in software will allow one cartridge to acess the devices and call in another module. This allows , for example, console Basic to access all of the plugged in modules call routines and devices names at one time. Osran Industries can be accessed through the Victoria 99'er Users Group, 1369 Finlayson ST., Victoria, British Columbia, v8t-2v5, Canada. NO price has been set as yet.

EDITORS NOTE

We now exchange newsletters with 46 other users groups accross the US and CANADA. This brings in alot of very usefull information on the TI computer.

Anyone that would like to look the newsletter colection over or would like copies, let me know at the users group meeting.

If anyone has something they would like to have included in the newsletter you can give it to me at the users group meeting or mail it to me at "Guy Hulsey rr#5 Box 13 Winfield,Ks. 67156.

I could use some usergroup member input

<><><><><><><><><><><><><><><><><><><><>

BY
LEE LAMAR, P.E.

Being an engineer, I am more at home with figures than words, especially when organizing words to transmit thoughts such as required for newsletter articles I do, however, feel obligated to provide something in return for all the help which I have received from the numerous members of the UG who have helped me i various computer matters. Several have indicated interest in using TI-WRITER and others have indicated that I have apparently used TI-WRITER more than most of the membership. This seems, therefore, to be the area where I can help the most. I will plan to write one chapter per month covering my experience with TI-WRITER if time permits.

TI-WRITER has been used for almost a year in my consulting engineering busines: for various purposes---correspondence, specifications for construction projects, preparation of forms for various use in the office, etc. Since TI-WRITER is the only word processor which I have used, I cannot compare it to others. I have, however, found it to be completely satisfactory for my needs. It has been relatively easy to learn. The manual provided is well written.

Using TI-WRITER requires a printer, of course, as well as a disk drive, memory expansion and an RS-232 card. My business requires a letter quality printer especially for construction specifications. Only a daisy wheel printer seemed suitable for this purpose. I chose a Radio Shack DWP 410 printer, with which am well pleased. This is a parallel printer using the Centronics system. It has one disadvantage in that the line feed cannot be turned off manually. When printing through the Text Formatter, of TI-WRITER, it is necessary to electronically turn off the line feed. I will explain how later. Otherwise the TI-99/4a, RS 410 printer and TI-WRITER have worked together quite well.

The most valuable use for me is no doubt the ability to type constuction specifications and keep them stored on diskettes. I can bring these documents up on the screen when needed for minor changes here and there and have the printer assemble and number the pages consecutively. Often these documents such as a grading specification are 25 to 30 pages long. After starting the printer I often go to my desk for other work. When I look over to the computer typing out these specifications and realize that I am not paying it even the minimum wage, I realize its value. I have no doubt that the computer and printer has paid for itself in time and money saved during the past year.

TI-WRITER has two main menu choices, TEXT EDITOR and TEXT FORMATTER. All original typing is done in TEXT EDITOR. Material can be sent to the printer from TEXT EDITOR if desired. I generally do this only for one page letters, forms etc. Printing from the TEXT EDITOR is exactly as shown on the screen. Therefore margins must be set, page breaks made properly before starting printing. I always type longer documents in TEXT EDITOR including insertion of the format commands, save to a diskette, enter TEXT FORMATTER and print from that point. There are a number of reasons for this which I will explain later. This Chapter 1 has been printed directly from TEXT EDITOR.

With TI-WRITER the full width of the page is not shown on the screen at one time. Slightly less than half the page width is shown at any one time, first the left side, then the middle part and last the right side. This can be somewhat confusing to a typist who is used to seeing the full width of their work at one time. It takes a little getting used to. Keyboard functions allow you to quickly and easily move from window to window. There is very little problem when typing narrative, but there is some lost motion when editing material already entered. The ability to have the letters on the screen large enough for easy reading, however, more than makes up for any problem with the three windows.