


West Jax 99er News



SEPTEMBER 1988

The WEST JAX 99'ERS is a non-profit computer users group for the TI-99/4A Home Computer. NOT affiliated in any way with Texas Instruments. The club's mailing address is PO BOX 176 Orange Park Florida 32067.

MEETINGS are held on the Second and Fourth Tuesday of each Month in the auditorium of the Webb Library. It is located two lights west of Blanding Boulevard on 103rd Street. The first meeting of the month is the Business meeting with workshop time after adjournment. The second meeting is strictly workshop time.

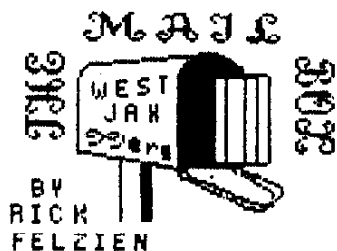
OFFICERS

President...Rick Felzien.....(904) 772-9162
Treasurer...Thomas LeMay.....(904) 282-5220
Secretary...Ralph Glattli.....(904) 751-1308
Librarian....Zach Ziegler.....(904) 389-2194

For newsletter suggestions and submissions, contact Rick Felzien.

This month we have the regular mail box column and another Basic Assembler installment plus a nice article on the Font Writer II publishing program written by Richard Kotrba.

For those Exchange Newsletter Editors who may think tat they missed a couple of issues of our Newsletter, we did not have a January or an April issue. We are a small group, about 12 paid members and it is hard at times to obtain material to make a publication feasable.



Tidewater 99er's Jun 88

1. Info on Weathermation
2. Utility useful with save utility

Dallas Interface Jul 88

1. An article on Compute mag.(rag?)
2. An interesting tinygram

Southern Nevada 99er's

1. Well written review of Lima faire
2. Hint for Star printer owners
3. Getting the most from cassettes

Charlotte Newsletter

1. Subroutine Extractor program
2. newsletter index

Victoria 99er's Mar/Apr 88

1. P-box dual drive mod.

Victoria 99er's May/June 88

1. F'web flowchart
2. list of available GROM's

Wordplay, Portland Or Jul 88

1. Easy multicolumn
2. Review of latest word processor
3. Read and Write, a nice program
4. Program to merge DU/80 files
5. Concept of arrays

SFU 99er times Jul 88

1. Remind Me vs Memo It prog. review
2. Impact 99

LITI 99er's Jul 88

1. What is a nibble anyway?
2. Putting it all together #3
3. ExBasic tutorial #3

Kansas City 99er's Jul 88

1. Are you grounded?
2. Programs that write prog. #5
3. PR-Base bug report
4. Putting it all together #1

ROM newsletter Jun 88

1. And so forth
2. E/A accept at routine
3. PRBaseology

HUG newsletter Jun 88

1. TI-Writer tips #2
2. Back to basics

Central Pa 99'ers Jul 88

1. Review of TI-Base
2. A poor man's loader
3. TI-Writer #7

Wordplay Aug 88

1. Super maze program
2. Partial files in TI-writer
3. High Res graphics #5
4. Relational expressions
5. Mystery program

N.O.V.A. Jul 88

1. Customizes forms in Graphx
2. Review of Plus!

West Penn 99'ers Jul 88

1. Let's talk Ramdisks #7
2. Pascal article
3. The UCR connection

Cleveland 99'ers Jul/Aug 88

1. Review of TI-Base
2. Plus! article
3. Impact 99
4. Clock/Calendar project
5. Review of Picture It

TICO topics Jul 88

1. Bypassing 16 bit bus wait state

ROM newsletter Jul 88

1. Assembly Windows and Invers Video
2. And so forth

Ozark 99'ers Aug/Sep88

1. TI-Writer tips
2. DIY surge protector

Boston Computer Society Jul 88

1. Review of Myarc ramdisk
2. E/A music program
3. Intro to Pascal
4. List of books for the 99/4A

The Computer bridge Jul 88

Bluegrass 99er Sep 88

1. Article, prog, on G-Graphic
2. Print Stylist sets up printer

Southwest 99er Sep 88

1. Trigonometry-computer applications
2. Children's corner

West Penn 99er Aug 88

1. Pascal/P-code article
2. 2-way communications in EXBasic
3. TI Exploits Pt.1
4. TI-Articst to TI-Writer, prog.
5. Review of TI-Base

Philadelphia area 99er Jun/Jul 88

1. Review of G-Graphic
2. What's next for the TI community

Bluegrass 99'ers Jul 88

1. BigBucks program set

LITI newsletter Aug 88

1. EXBasic tutorial from F'web Farm

Erie 99er Aug 88

1. Elements of Basic Pt. 21
2. Name that phone (tinygram)
3. TI-Writer tip
4. A unique Dec.to binary prog.
5. Getting most from cassette sys.

Pittsburg 99er Aug 88

1. Starting a new FORTH series
2. Tips for beginners
3. Getting most from cassette sys

North Jersey 99'ers Jul 88

1. Lightning protection
2. customizing F'web

Delarare Valley 99'ers Jun 88

1. Interesting interfacing articles
2. About screen dumps

FONT WRITER II

PART 2

FONT WRITER II IS WRITTEN IN EXTENDED BASIC AND IT CONSISTS OF 9 MAIN SECTIONS. HERE IS A BRIEF DESCRIPTION OF EACH SECTION.

- 1.EDITOR - THIS PART ALLOWS THE CREATION AND MODIFICATION OF FONTS,IMAGES AND EVEN SPRITES.
- 2.FORMATTER - THIS IS A SPECIAL FORMATTER TO FORMAT BOTH TEXT AND GRAPHIC FILES.
- 3.MANAGER - THIS IS A DISK MANAGER SYSTEM TO CATALOG,COPY AND DELETE FILES. IT ALSO HAS THE ABILITY TO CONVERT BETWEEN CSGD AND TI-ARTIST FONTS AND PICTURES.
- 4.DISK DUMP - THIS IS A VERY HANDY UTILITY THAT WILL PRODUCE A PRINT OUT OF EVERY CHARACTER IN ALL FONTS ON A DISK. THIS CREATES HANDY REFERENCE SHEETS.
- 5.BANNER - CREATES BANNERS IN VARIOUS SIZES FROM ANY CSGD FONT.
- 6.TI-WRITER EDITOR -- WORKS LIKE THE REGULAR TI-WRITER EDITOR WITHOUT THE SHOW DIRECTORY FUNCTION.
- 7.TI-WRITER FORMATTER -WORKS THE SAME AS THE REGULAR TI-WRITER FORMATTER.
- 8.UTILITY LOADER --ALLOWS THE LOADING OF ANY EDITOR/ASSEMBLER 5 (E/A 5) PROGRAM SUCH AS DM1000.
- 9.DEFAULTS - WRITES THE DEFAULTS OF YOUR PARTICULAR SYSTEM TO DISK SO THAT THEY DON'T HAVE TO BE TYPED IN EACH TIME. THESE INCLUDE PRINTER ACCESS CODES AND DISK DRIVE NUMBERS.

1.EDITOR IS THE MOST COMPLEX SECTION AND THE MAJORITY OF THE DOCUMENTATION THAT COMES WITH THE PROGRAM DEALS WITH THIS SECTION. I FOUND,HOWEVER,THAT THE 2.FORMATTER SECTION IS THE BEST PLACE TO START. THE FORMATTER USES TI-WRITER FILES OR IT CAN BE USED IN THE IMMEDIATE MODE. TO GET TO THE IMMEDIATE MODE, CHOOSE 2.FORMATTER FROM THE MAIN MENU. WHEN "FONT FILE" APPEARS,PRESS ENTER. WHEN "INPUT FILE" APPEARS,PRESS ENTER AGAIN. YOU WILL NOW BE IN THE IMMEDIATE MODE.
USE THE DOT COMMANDS WHICH ARE EXPLAINED NEXT.

FONT WRITER II USES DOT COMMANDS (.) TO GIVE INSTRUCTIONS TO THE PROGRAM. THE DOT IS FOLLOWED BY 2 LETTERS. IN SOME CASES THERE ARE OPTIONS AND THE FORMAT IS DOT,FOLLOWED BY 2 LETTERS,FOLLOWED BY A SPACE,FOLLOWED BY THE OPTION. THE DOT COMMANDS CAN BE GROUPED INTO FOUR CATAGORTES.

- 1.TI-WRITER COMMANDS USED BY FONT WRITER II
- 2.GRAPHIC AND GRAPHIC/FONT COMMANDS
- 3.EXTERNAL FILE COMMANDS
- 4.OTHER COMMANDS

HERE ARE THE COMMANDS THAT ARE IDENTICAL TO TI-WRITER COMMANDS

- 1 .FI (FILL -EXPANDS THE SPACING BETWEEN WORDS SO THAT THE LINE FILLS THE SPACES BETWEEN MARGINS.)
- 2 .NF (NO FILL -WORDS WILL PRINT AS TYPED, MARGIN SETTINGS ARE IGNORED:)
- 3 .AD (ADJUST -ADJUSTS LINES OF GRAPHICS SO THAT RIGHT MARGIN IS EVEN. REQUIRES THAT .FI COMMAND IS USED FIRST.)
- 4 .NA (TURNS OFF ADJUST COMMAND.)
- 5 .BP (STOPS THE PRINTING ON ONE PAGE. ANY OTHER TEXT OR GRAPHICS WILL PRINT ON THE NEXT PAGE.)
- 6 .CE n (CENTERS PRINTER TEXT, GRAPHICS TEXT AND GRAPHICS ON A LINE. THE NUMBER OF LINES CENTERED IS DETERMINED BY THE NUMBER(n).
- 7 .IN n (INDENT THE FIRST LINE OF EACH PARAGRAPH BY THE NUMBER(n). IF (n) IS NEGATIVE THE FIRST LINES WILL BE OUTDENTED IF POSSIBLE.)
- 8 .LM n (SETS THE LEFT MARGIN (n) SPACES. FIRST SPACE IS NUMBERED 0, NOT 1.)
- 9 .RM n (SETS THE RIGHT MARGIN.)
- 10 .LS (LINE SPACING. THIS IS THE NUMBER OF SPACES BETWEEN EACH LINE.)
- 11 .SP n (NUMBER OF BLANK LINES. WORKS ONLY ONCE EACH TIME IT IS USED.)
- 12 .PL n (PAGE LENGTH -SETS NUMBER OF LINES THAT WILL BE PRINTED BEFORE THE PAPER IS ADVANCED TO THE NEXT PAGE.)
- 13 .CO (COMMENT -USED TO PLACE COMMENTS IN FILE. USED LIKE A REM STATEMENT.)
- 14 .IF (INCLUDE FILE -USED TO ADD OTHER FILES TO BE MERGED IN. CORRECT FORMAT IS .IF DSKn.FILENAME)
- 15 .BL n (BLANK LINES -SETS THE NUMBER OF BLANK LINES THAT WILL BE PRINTED BETWEEN PAGES.)
- 16 .GR (PLACES THE PROGRAM IN GRAPHICS MODE. THIS IS THE MODE THE PROGRAM IS IN WHEN IT IS STARTED. THE .GR COMMAND WOULD BE USED TO RETURN FROM TEXT(.TX) MODE.)
- 17 .II (INCLUDE IMAGE -PLACES AN IMAGE AT CURRENT LEFT MARGIN. RESPONDS TO THE CENTER COMMAND (.CE). PROPER FORMAT IS .II DSKn.FILENAME)
- 18 .IL n (IMAGE LEFT -SETS THE LEFT MARGIN OF AN IMAGE USED WITH TEXT. USED WITH THE .IM COMMAND.)
- 19 .IM (IMAGE MERGE -USED WHEN AN IMAGE IS PLACED ON THE SAME LINE AS GRAPHIC/TEXT. CAN BE USED WITH .IL n COMMAND. CORRECT FORMAT IS .IM DSKn.FILENAME. FILENAME MUST BE AN IMAGE FILE AND END WITH _I.)
- 20 .IC (IMAGE COMPLETE -INSURES THAT THE IMAGE IS COMPLETED WHEN USING THE IMAGE MERGE COMMAND.)

- 21 .IR (IMAGE ROW -PRINTS A ROW OF A SINGLE IMAGE ACROSS THE PAGE.)
- 22 .IP (INCLUDE PICTURE -PRINTS A TI-ARTIST OR GRAPHX PICTURE. FORMAT IS .IP m,DSKn.FILENAME_P WHERE m=THE LEFT MARGIN OR .IP DSKn.FILENAME_P)
- 23 .GI ON/OFF (GRAPHICS INVERT -.GI ON REVERSES IMAGE COLORS. .GI OFF RESETS COLORS.)
- 24 .FR ON/OFF (FRAMES GRAPHICS - USED WITH .II, .IR OR .IP COMMANDS. PUTS A FRAME AROUND THE IMAGE OR PICTURE. USED WITH .FSn COMMAND AND AFFECTED BY THE .BI COMMAND.)
- 25 .BB (BEGIN BOX -STARTS A BOX ON THE PAGE. UP TO 8 BOXES CAN BE OPEN AT ANY ONE TIME. PROPER FORMAT IS .BB n,s,lm,rm WHERE n=BOX NUMBER(1 TO 8),s=BOX STYLE (1-6),lm=LEFT MARGIN,rm=RIGHT MARGIN. NOTE-NO TWO BOXES CAN BEGIN OR END ON THE SAME LINE.)
- 26 .BE n (BOX END -DRAWS THE BOTTOM OF THE BOX. n=BOX NUMBER USED IN .BB NOTE THE SIDES OF A BOX WILL CONTINUE TO BE DRAWN UNTIL .BE n IS USED.)
- 27 .BI ON/OFF (PRINTS ALL BOXES AND FRAMES IN INVERSE WHEN .BI ON IS USED.)
- 28 .LF (LOAD FONT -LOADS THE GRAPHIC FONT INTO MEMORY. WILL LOAD BOTH TI-ARTIST AND CSGD FONTS. PROPER FORMAT IS .LF DSKn.FILENAME. FOR TI-ARTIST FONTS THE FILENAME ENDS IN _F, FOR CSGD FONTS THE FILENAME ENDS WITH /CH.)
- 29 .CS n (CHARACTER SPACING -PLACES EXTRA SPACES BETWEEN FONT CHARACTERS. AMOUNT OF SPACING DEPENDS ON NUMBER GIVEN TO n.)
- 30 .DS ON/OFF (DENSITY ON/OFF -USED FOR SINGLE AND DOUBLE DENSITY PRINTING OF FONTS AND GRAPHICS. THIS COMMAND WILL CONDENSE PRINT IMAGES AND FONTS. USE WITH THE FOLLOWING COMMAND TO RETURN TO NORMAL WIDTH.)
- 31 .DB ON/OFF (DOUBLE WIDTH ON/OFF -USED TO PRINT A FONT OR IMAGE IN DOUBLE WIDTH.)
- 32 .TI ON/OFF (USED TO INVERT THE TEXT COLORS.)

THESE ARE THE EXTERNAL FILE COMMANDS.

- 33 .OX DSKn.FILENAME_X (THIS COMMAND DIRECTS THE GRAPHICS AND GRAPHIC/FONTS TO AN EXTERNAL FILE RATHER THEN TO THE PRINTER.)
- 34 .CX (CLOSES THE EXTERNAL FILE OPENED WITH THE .OX COMMAND.)
- 35 .IX DSKn.FILENAME_X (INCLUDES AN EXTERNAL FILE THAT WAS CREATED WITH THE .OX COMMAND INTO THE FILE BEING CREATED.)

OTHER COMMANDS ARE:

- 36 .TX (TEXT-ALLOWS THE PRINTERS FONTS TO BE USED.)
- 37 .DF m,DSKn.FILENAME (DUMP FILE-THIS PRINTS THE FILE EXACTLY AS WRITTEN. THE m DETERMINES THE LEFT MARGIN.)

38 .KB n (KEYBOARD ENTRY-ALLOWS THE INSERTION OF INSTRUCTIONS FROM THE
KEYBOARD. THE n INDICATES HOW MANY LINES WILL BE ENTERED FROM THE KEYBOARD.)

39 .ME (PRINTS ANY MESSAGE TYPED AFTER THIS COMMAND ON THE SCREEN. THIS COMMAND
DOES NOT DIRECT THE MESSAGE TO THE PRINTER. IT IS USED AS A PROMPT FOR KEYBOARD
ENTRIES.)

40 .QU (QUIT-USED TO EXIT THE IMMEDIATE MODE.)

AS YOU CAN SEE, THIS IS A VERY FLEXIBLE PROGRAMMING ENVIRONMENT. NEXT TIME
WE'LL COMBINE SOME OF THE COMMANDS TO PRODUCE MIXTURES OF GRAPHICS AND TEXT.

RICHARD KOTRBA...WEST JAX 99'ERS USER GROUP.

T H E B A S I C A S S E M B L E R #4 By Steve Peacock

DEMONSTRATION OF HOW TO USE THE JOYSTICK

This month I am presenting a program that will demonstrate how to use the joysticks. Two new codes are introduced, they are KSCAN and MOV8. KSCAN is the routine that reads the joysticks. It will also read the keyboard. MOV8 stands for MOVE Byte. This command moves the left (most significant) byte of a word. For example the command -MOV8 @>8374,R1- will move the left (most significant) byte that is in the address >8374 into the left (most significant) byte of R1. The value in >8374 is not changed. This is used when you want to compare two values, as you can see in this months program.

If you have an assembly program written for the MINI-MEMORY and want to type it in using the EDITOR/ASSEMBLER, you will have to convert some numbers to mnemonics. Below is a table that lists the changes.

MINI-MEMORY	EDITOR/ASSEMBLER
>6018	GFLLNK Link to GROM Routine
>601C	XMLLNK Link to ROM Routine
>6020	KSCAN Keyboard Scan
>6024	VSBW VDP Single Byte Write
>6028	VM8W VDP Multiple Byte Write
>602C	V8BR VDP Single Byte Read
>6030	V8BR VDP Multiple Byte Read
>6034	VWTR VDP Write to Register
>6038	DSRLNK Link to Device Service Routine
>603C	LOAD5R Tagged Object Loader
>6040	NUMAS8 Numeric Assignment Routine
>6044	NUMREF Get Numeric Parameter
>6048	STRAS8 String Assignment Routine
>604C	STRREF Get String Parameter
>6050	ERR Error REporting Routine

For example if you had a line that looked like this -BLWP @>6024-, it would be changed to -BLWP @VSBW-, for the EDITOR/ASSEMBLER.


```

*****
*
*PROGRAM BA4A==>Basic Assembler #4 Assembly Version
*DEMONSTRATION OF HOW TO USE THE JOYSTICK
*(C)1985 S. PEACOCK
*
*****
REF  KSCAN,VSBW  *KEYBOARD SCAN/WRITE SINGLE BYTE TO SCREEN
DEF  START      *START OF PROGRAM
START LI  R0,300  *START POSITION TO PRINT 'X'
      LI  R1,>5800 *LOAD HEX CODE FOR 'X' INTO REG. 1
      BLWP @VSBW  *PRINT 'X'
      LI  R1,>0100 *>0100 FOR JOYSTICK NUMBER 1 (>0200 FOR JOYSTICK #2)
      MOVB R1,@>8374 *MOVE LEFT BYTE (>01) INTO THE ADDRESS >8374 (KSCAN)
LF    BLWP @KSCAN *BRANCH TO KEYSKAN ROUTINE
      CLR  R1      *CLEARS REG. 1 WHERE THE VALUE >8376
*****WILL BE PLACED. RIGHT BYTE ZERO WILL BE ZERO FOR
*****COMPARISION.
      MOVB @>8376,R1 *CHECK 'Y' RETURN. WHEN THE JOYSTICK IS USED THE 'Y'
*****RETURN WILL BE PUT IN THE ADDRESS >8376. THIS VALUE
*****IS MOVED INTO REG. 1. THE 'Y' RETURN CAN BE
*****>FC, >00, >04 (-4, 0, 4)
      CI  R1,>0400 *IF 'Y' RETURN >04 THEN JOYSTICK PUSHED UP
      JNE T1      *JUMP TO NEXT COMPARISION IF 'Y' RETURN NOT
*****EQUAL TO >04
      CI  R0,32   *TO SEE IF TOP BORDER IS HIT. THE LARGEST POSITION
*****POSSIBLE IS 32 (TOP RIGHT)
      JLT  LF    *IF TOP BORDER HIT GO BACK TO MAIN LOOP AND READ KSCAN
      LI  R5,>2000 *IF NOT HIT, LOAD 'SPACE'
      BLWP @VSBW *PRINT 'SPACE'
      AI  R0,-32  *DECREASE PRINT POSITION ONE ROW
      JMP  PG    *JUMP TO PRINT
T1    CI  R1,>FC00 *IF 'Y' RETURN >FC00 THEN JOYSTICK PUSHED DOWN
      JNE T2      *JUMP TO NEXT COMPARISION IF 'Y' RETURN NOT
*****EQUAL TO >FC00
      CI  R0,735 *TO SEE IF BOTTOM BORDER IS HIT. THE SMALLEST POSITION
*****POSSIBLE IS 735 (BOTTOM LEFT)
      JGT  LF    *IF HIT GO BACK TO MAIN LOOP AND READ KSCAN
      LI  R5,>2000 *IF NOT HIT LOAD 'SPACE'
      BLWP @VSBW *PRINT 'SPACE'
      AI  R0,32  *INCREASE PRINT POSITION ONE ROW
      JMP  PG    *JUMP TO PRINT
T2    MOVB @>8377,R1 *CHECK 'X' RETURN. WHEN THE JOYSTICK IS USED THE 'X'
*****RETURN WILL BE PUT IN THE ADDRESS >8377. THIS VALUE
*****IS MOVED INTO REG. 1. THE 'X' RETURN CAN BE
*****>FC, >00, >04 (-4, 0, 4)
      CI  R1,>0400 *IF 'X' RETURN >0400 THEN JOYSTICK PUSHED RIGHT
      JNE T3      *JUMP TO NEXT COMPARISION IF 'X' RETURN NOT
*****EQUAL TO >0400
      LI  R8,31   *SEE !START AT TOP RIGHT POSITION COMPARE THIS VALUE
RL    C  R0,R8    *IF !WITH THE VALUE OF REG 0 (THE MAIN PRINT
      JEQ  LF    *RIGHT !POSITION) IF EQUAL JUMP OUT OF THE LOOP. IF
      AI  R8,32  *WALL !NOT ADD 32 (1 ROW) AND COMPARE. DO THIS
      CI  R8,768 *IS !UNTILL THE BOTTOM RIGHT CORNER IS REACHED.
      JLT  RL    *HIT !IF NO HIT, LOAD A 'SPACE' AND PRINT
      LI  R5,>2000 *LOAD 'SPACE'
      BLWP @VSBW *PRINT 'SPACE'
      INC  R0    *INCREASE PRINT POSITION ONE COLUMN

```

```

      JMP PG          *JUMP TO PRINT
T3   CI R1,>FC00     *IF 'X' RETURN >FC00 THEN JOYSTICK PUSHED LEFT
      JNE LF        *JUMP TO TO MAIN LOOP. NO OTHER COMPARISON TO MAKE.
      LI R0,0       *SEE I START AT TOP LEFT POSITION COMPARE THIS VALUE
LL   C R0,R8        *IF I WITH THE VALUE OF REG 0 (THE MAIN PRINT
      JEQ LF        *LEFT I POSITION) IF EQUAL JUMP OUT OF THE LOOP. IF
      AI R8,32      *WALL I (NOT ADD 32 (1 ROW) AND COMPARE. DO THIS
      CI R8,737     *IS I UNTILL THE BOTTOM RIGHT CORNER IS REACHED.
      JLT LL        *HIT I IF NO HIT LOAD A 'SPACE' AND PRINT.
      LI R5,>2000   *LOAD 'SPACE'
      BLWP @VSBW    *PRINT 'SPACE'
      DEC R0        *DECREASE PRINT POSITION ONE COLUMN
PG   LI R1,>5800    *ASCII CODE FOR 'X'
      BLWP @VSBW    *PRINT 'X'
*    LI R4,4000    *A SHORT I TO SLOW DOWN THE PROGRAM
*    DEC R4        *DELAY I (REMOVE THE THREE ASTERICKS.
*    JNE $-?      *I NOP I
      JMP LF        *BACK TO READ JOYSTICK AGAIN
      END

```

```

100 REM PROGRAM BA4B==>Basic Assembler #4 Basic Version
110 REM DEMONSTRATION OF HOW TO USE THE JOYSTICK
120 REM (C)1985 S. PEACOCK
130 REM YOU MAY WANT TO PUT IN A 'CALL CLEAR' HERE
140 R=10
150 C=13
160 CALL HCHAR(R,C,88)
170 CALL JOYST(1,X,Y)
180 IF Y<>4 THEN 230
190 IF R=1 THEN 170
200 CALL HCHAR(R,C,32)
210 R=R-1
220 GOTO 370
230 IF Y<>-4 THEN 280
240 IF R=24 THEN 170
250 CALL HCHAR(R,C,32)
260 R=R+1
270 GOTO 370
280 IF X<>4 THEN 330
290 IF C=32 THEN 170
300 CALL HCHAR(R,C,32)
310 C=C+1
320 GOTO 370
330 IF X<>-4 THEN 170
340 IF C=1 THEN 170
350 CALL HCHAR(R,C,32)
360 C=C-1
370 CALL HCHAR(R,C,88)
380 GOTO 170
390 END

```