

JPDRAWING by Jean-Pierre Morin (Version 3.0)

INTRODUCTION

JPDRAWING program is designed to draw color graphics in the TI-FORTH environment without having prior knowledge of TI-FORTH on the part of the user. Therefore, it is accessible and easy to assimilate. A few minutes per command should amply suffice to become familiar with the single-key and LOGO commands.

JPDRAWING requires the following minimal TI-99/4A system configuration:

- TI-99/4A console with monitor or TV set (preferably color)
- One disk drive storage system
- 32K expansion memory system
- Editor/Assembler module

Optionally, a printer will provide black & white screen dumps in two different formats.

FEATURES

- Full bit map or split screen (SPLIT2) screen formats
- Line, circle, rectangle, arc and many more drawing commands
- 7 different paint brushes
- Full drawing LOGO facilities
- Paint command
- Screen save and load to/from disk
- 2 print screen commands
- Some forth utilities like SAVE, ST-STTS, ASC
- 2 easy ways of adding new drawing commands
- Big letter typing
- Command saving and re-execution. (User-defined commands.)

LOADING THE PROGRAM

Using the Editor/Assembler's Option 3, type "DSK1.FORTH". The program will load automatically and execute a small demo upon your request. When the demo is through, you are free to take a crack at it! This program capabilities are best learned through experimentation; don't be afraid to try your fancy. Naturally, you will have put a write-protect tab on your program diskette first!

SCREEN DESCRIPTION

The screen consists of a column of 6 two-digit numbers at the top right corner; these 6 numbers are referred to as P1 to P6 (Parameter 1 to parameter 6). Immediately below them is the 15-color tablet and at the bottom right corner is the angle value used by LOGO commands (0=right, 90=down, 180=left, 270=up). A sprite (the cursor) defined with one of eight different shapes is used to show the type of paint brush in effect. This sprite is called the cursor in the following text. You can use the joystick #1 or arrow keys to move the cursor around. A second sprite, called the beginning sprite, can be anchored anywhere on the screen to draw lines, circles and others between it and the cursor.

FULL SCREEN BIT-MAP DRAWING COMMANDS

For each of the commands below (words in parentheses), a single key is assigned (far left) which takes effect immediately upon being pressed. The words themselves are used in COMMAND LISTS or in DESSIN2 mode to achieve the same results.

- 0 (KEY0 or PU): No trailing line appears on the screen when the cursor is moved.
- 1 (KEY1 or PD): The cursor is assigned a 1-pixel paint brush; when moved, a thin line is trailing.
- 2 (KEY2): The cursor is assigned an 8-pixel horizontal paint brush; when moved, a "ribbon" effect can be achieved.
- 3 (KEY3): The cursor is assigned a solid square of 3 by 3 pixels; when moved, it leaves a thick line behind.
- 4 (KEY4): The cursor is assigned a 3-pixel paint brush with this shape "/".
- 5 (KEY5): The cursor is assigned a 3-pixel paint brush with this shape "\".
- 6 (KEY6): The cursor is assigned an 8-pixel vertical paint brush.
- 7 (KEY7): The cursor is assigned a 3-pixel vertical paint brush with two pixels in between.
- = (+BCKGRND): Advances the color tablet pointer to the next color in effect. See also the G command.
- b (>BEG): Anchors the beginning sprite to be used by the a, c, d, g, l, P, R, or s commands.

- l (S-LINE): Draws a smart line between the beginning sprite and the cursor position (See L (MLINE) command). The l command is slower than the L command, but works better in multicolor drawings. (See F (FST-MD) and S (SLW-MD) commands).
- c (CIRCLE): Draws a circle between the beginning sprite and the cursor.
- s (SQUARE): Draws a rectangle between the beginning sprite and the cursor.
- h (HOME): Places the cursor in the middle of the screen.
- / (NXT-NMBR): Advances the parameter pointer and sets it in input mode (See i (INPT-NMBR) command).
- !, @, #, \$, % and ^ (SHIFT 1, 2, 3, 4, 5 and 6): Places the parameter pointer at P1 to P6 respectively, and sets it in input mode (See i (INPT-NMBR) command).
- i (INPT-NMBR): Allows the input of a value beside the parameter pointer (P1 to P6). These parameter numbers are used as input by many functions (FXX), mainly LOGO functions. A key other than 0 to 9 will terminate the input; use space bar to exit and activate the use of the joystick.
- d (ARD): Draws a right arc between the beginning sprite and the cursor:
 - P1 is the radius of the circle
 - P2 is the number of degrees of curvature of the arc.
 - (EX.: 6 will give you a 60 degrees arc, 18 will give you a 180 degrees arc (half a circle)).
- e (EXCT): Executes a COMMAND LIST at block P1 line P2. This COMMAND LIST can be entered via the editor or automatically while you draw, after having pressed the O command, and terminated with the E command. See the COMMAND LIST FORMAT section.
- g (ARG): Draws a left arc. (See d (ARD) command).
- n (ST-ANGLE): Allows input of the drawing angle. A key other than 0 to 9 will terminate the input.
- o (ST-CLR): Sets old color. The selected color will appear just below the white color. Used by CONTROL C and P commands.
- p (PNT): Paint command. First set color using = command, put cursor inside what you want to paint and press p.
- r (READ): Puts the current cursor column position in pixels into P1 and P2; if column=123, P1=1 and P2=23, and row position in pixels into P3 and P4.
- q (ADJST): Adjusts cursor at the beginning of an 8-pixel structure, to fix more than 2 adjacent colors.

- t (CLR-TBL): Restores the color tablet to its original setting. Very useful after a CONTROL C or P command.
- a (ATTCH): Attaches the beginning and the cursor sprites together. Use the same command to de-attach.
- w (WRITE): Puts cursor at position defined by P1 and P2 for column and P3 and P4 for row. If P3=01, P4=12 then row=112. Opposite of the r command.
- x, y, and z: Saves the cursor position into internal variables (3 available). Commands X, Y and Z put the cursor back to these positions. Useful to "reference" a cursor location on the screen.
- A (APPND-SV): Similar to O command, saves commands in COMMAND LIST at the end of the existing COMMAND LIST.
- B (BCKGRND): Changes color of the top and bottom of the screen. First select the color using = command.
- C (CHGN-SPRT-CLR): Changes sprites color (Sprites are the beginning sprite, the cursor, the parameter values, pointer arrows, and the angle). First select color with = command.
- D (DBC-ON-OFF): Toggles debouncing ON and OFF. With debouncing ON, a key pressed down will not automatically repeat, allowing greater precision drawings.
- E (END-SV): Closes (ends saving) the previously opened COMMAND LIST. See O and A commands.
- F (FST-MD): Restores fast drawing mode for LOGO commands.
- G (GT-CLR): Sets the color tablet pointer to P1.
- H (F04): Draws a house of height P1 pixels.
- L (MLINE): Draws a line between the beginning sprite and the cursor (faster than l, but not as smart).
- O (OPN-SV): Initiates the save mode for a COMMAND LIST. "EX" is automatically written at block P1, line P2. Each and every command following the O command will be saved until an E command. The saved commands can be executed with the e command.
- P (PRTL-MD-CLR): Modifies color on a portion of a screen delimited by the beginning and cursor sprites. First select color to be changed with the o command, then select the new color with = command (See CONTROL C command).
- R (RPLCT): Copies part of the screen delimited by the beginning sprite and the cursor to column (P1,P2) and row (P3,P4). The r command can be used to setup P1, P2, P3 and P4.
- S (SLW-MD): Selects slow drawing mode for LOGO commands.

T: Selects typing mode. All capital letters will be drawn with the specified angle. Use] to exit typing mode.

> Similar to T command. Advances an extra P1 pixels and rotates (P2, P3) between each letter. If P2=03 and P3=30, the rotation angle is 330 degrees to the right (equivalent to 30 to the left).

X, Y, and Z: Restores the cursor position. See x, y and z commands.

"(" and ")": All commands between (and) are repeated P5 times. P5 must be setup first. This command only works in save mode (command O or A) or when executing a COMMAND LIST (e command).

FUNCTION CLEAR (CONTROL B): Exit. Use DESSIN or JS in DESSIN2 mode to go back in drawing mode. Use MON to exit to the master console title screen.

CONTROL C (MD-CLR): Modifies color. First select the color you want to change using the = and o commands, then select the new color and press CONTROL C. To change a portion of the screen use the P command.

CONTROL E (CLS): Clears the screen.

CONTROL L (LD-SCRN): Loads screen P1, where P1 is an arbitrary screen number between 01 and 07. Be sure to have the appropriate diskette before executing this command. The screen must have been previously saved with the CONTROL S command.

CONTROL S (SV-SCRN): Saves screen P1 where P1 must be an arbitrary screen number between 01 and 07. Be sure to have the appropriate diskette before executing this command.

CONTROL T (CLT): Clears the first 4 lines in DESSIN2 mode.

CONTROL P (PRNT-SCRN): Prints screen in condensed format with a 90-degree rotation to the right. (. command is nicer but slower). Press Q for a while to cancel printing.

. (PRNT-SCRN2): Prints screen. Press Q for a while to cancel printing.

FUNCTION BEGIN or CONTROL N (EXCT-FXX): Executes function XX (See FXX list below). You can create your own FXX functions; this command will execute them. You must first specify the function number XX in P6.

CONTROL R (RPT-FXX): Executes FXX P5 times rotating 360/P5 degrees each time.

FUNCTION REDO or CONTROL R (PTR-FXX): Similar to the CONTROL R command, but P4 pixels from the center for each rotation.

CONTROL W (RPTWR-FXX): Executes function FXX P4 times,
advancing P5 pixels each time.

_ : Rotates left P4 degrees.

? : Rotates right P4 degrees.

& : Saves angle value in an internal variable.

" : Restores angle value from the internal variable.

{ : Advances P3 pixels in the direction of angle.

} : Advances P3 pixels in the opposite direction from angle.

FXX_FUNCTIONS

- F01 Draws a star. P1=Hexagone side size in pixels.
- F02 Advances P3 pixels, rotates right $360 * P1/P2$ P2 times.
- F03 Advances 0, 2, ..., 198 pixels rotating $(P1 * 100 + P2)$ each time 100 times.
- F04 Draws a house. P1=Height of the house in pixels.
- F05 Draws a polygone (PG).
P1=Side length in pixels
P2=Number of sides
- F06 Draws P1 polygones with a rotation of $360/P1$ (RPG)
P1=Number of rotations
P2=Side length in pixels
P3=Number of sides
- F07 Draws P1 polygones whit a rotation of $360/P1$ P4 pixels away from the cursor.
P1=Number of rotations
P2=Side length in pixels
P3=Number of sides
P4=Radius of rotation in pixels
- F08 Draws a flower.
P1=Radius length in pixels
P2=Number of degrees for arc (60 is a good choice)
P3=Number of petals
- F12 Similar to F08, puts a circle in the middle.
- F09 Draws and rotates until you type "/".
P1=Side length in pixels
P2 and P3=Degrees for rotation. (EX.: 144 -> P2=1, P3=44).
- F10 Repeats 720 times [advance P1, turn right P2, increment P2 by P3].
- F11 Draws a tree.
P1=Longest branch in pixels
P2=Smallest branch in pixels
P3=Number of divisions for each branch
P4=Number of degrees between two branches.
- F13 Similar to F11, using random numbers.

LOGO COMMANDS AND DESSIN2 MODE COMMANDS

You can create your own drawing functions using LOGO type commands. The best way to learn these commands is to go in DESSIN2 mode and try them. You can also use the full screen commands by using their names in full.

xx FD: Advances the cursor xx pixels in the direction of angle.

xx BK: Advances the cursor xx pixels in the opposite direction from angle.

xx RT: Right turn of xx degrees.

xx LT: Left turn of xx degrees.

ANGLE@: Returns the angle value.

xx ANGLE!: Sets the angle value.

PD: Puts cursor in drawing mode.

PU: Puts sursor in non-drawing mode.

XX [...]: Repeats instructions between [] XX times. Can only be used in definition mode.

xx COLPOS !: Sets the column position to xx.

xx ROWPOS !: Sets the row position to xx.

xx COLPOS+!: Adds xx to COLPOS.

xx ROWPOS+!: Adds xx to ROWPOS.

xx GET: Puts on the stack the first xx parameters.

xx TAB-NMBR @: Puts on the stack Pxx+1 (For internal offset).

JS or JOYSTICK: Goes in joystick drawing mode.

ASC x PKEY: Will execute the x command, where x is any one of single-key commands.

EXAMPLE

You want to create a rectangle function where P1=Height and P2=Length (both in pixels). Exit drawing mode and type:

```
: RECTANGLE PD 2 GET 2DUP 4 [ FD 90 LT ] ;
```

You can assign RECTANGLE to the R key so that every time the R key is pressed RECTANGLE is executed, using the ASC and DFN commands, as follows:

```
ASC R DFN RECTANGLE
```

Next type DESSIN, setup P1 and P2, press R and a rectangle will be drawn. Try changing the angle value with the n command. Press R again. You got it! You can also assign a function number to the RECTANGLE function as follows:

```
: F33 RECTANGLE ; (33 is an example, suit yourself!)
```

To execute, setup P1 and P2, set P6 to 33 (or what have you) and press FUNCTION BEGIN. You can also use FUNCTION REDO, CONTROL R or CONTROL W to draw multiple rectangles.

COMMAND LIST FORMAT

Using the O command, you can save all the commands up to the E command (END-SV). You can re-execute this COMMAND LIST with the e command. In both cases, you must first setup the TI-FORTH block number in P1 (any of the 90 blocks (0 to 89) available on a blank initialized diskette), and the line number within that block where your function starts in P2. You can use the editor to create or modify a COMMAND LIST stored on a block, or input a block in "real-time", as you draw, with the O command. The first 2 characters of a COMMAND LIST must be "EX", otherwise the e command will not execute it. The last character must be ";".

A few changes were made to the TI-FORTH editor:

- Control characters are displayed in reverse video.
- You can insert a control character by typing CONTROL I followed by the control character.
- CONTROL INSERT and CONTROL DELETE scroll the whole screen instead of only the line containing the cursor.

This TI-FORTH application program was designed and written by Jean-Pierre Morin. Have a good time !

JP GRAPHIC V3.1 ADDENDUM

V3.1 adds music to V3.0. To make room in memory for the music routines, the following functions were removed:

- F04 (HOUSE), F08, F12 (FLOWER), F09, F11, F13 (TREE).

These functions are available in source code and can be loaded into memory if desired. To load them, press "FCTN CLEAR" to exit drawing mode and type:

- 37 LOAD to load F08, F12 (FLOWER) AND F09
- 38 LOAD to load F04 (HOUSE)
- 39 LOAD to load F11, F13 (TREE)

Type DESSIN or DESSIN2 to go back to drawing mode.

Here is the DISK ORGANIZATION by BLOCK:

0 RESERVED FOR FILENAMES AND BIT MAP
1 FREE BLOCKS
2 DEMO PROGRAM
3 BOOTING SCREEN
4-5 ERROR MESSAGES
6-7 26 CAPITAL LETTERS ROUTINES
8-19 FORTH SYSTEM
20-36 JP GRAPHIC BINARY IMAGE PROGRAM
37 F08, F12 (FLOWER), F09
38 F04 (HOUSE)
39 F11, F13 (TREE)
40-48 FREE BLOCKS
49-60 FOREST SAVED IMAGE (P1=05)
61-72 HOUSE SAVED IMAGE (P1=06)
73-79 FREE BLOCKS
80-81 MUSIC SOUND LIST
82-89 FREE BLOCKS

To stop the music, exit drawing mode with "FNCT CLEAR" and type:

0 >83C4 ! 7 EMIT

*** JPDRAWING VERSION 3.0 ***