XEROX PALO ALTO RESEARCH CENTER
Computer Science Laboratory

January 13, 1976

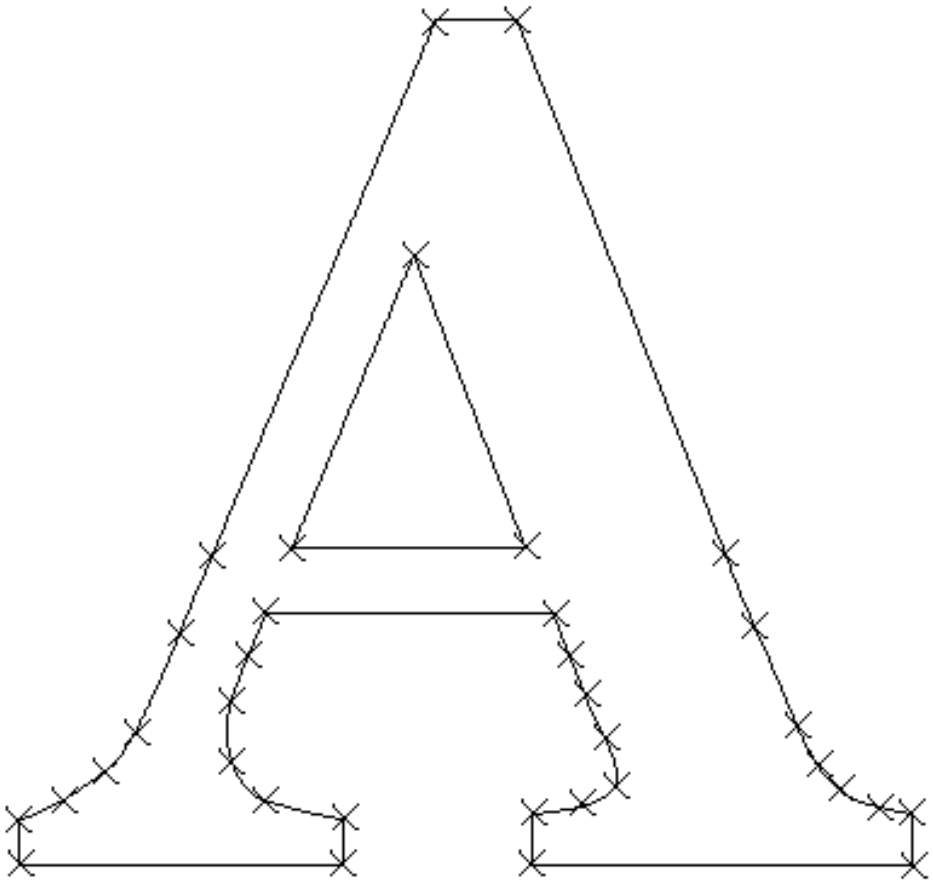
To: CSL/SSL

From: Patrick Baudelaire

Subject: FRED

1. Introduction

FRED is an interactive editor of curves, intended to be used mainly for creating fonts. FRED is used to define outlines of characters.



FRED manipulates spline curves, which are piecewise parametric cubic functions fitting a set of points called knots (shown as "x" above). Spline curves are created and modified by simple operations on these knots.

Usually, when creating font outlines, the curves should correspond to the shape of a character, perhaps designed by a graphic artist. To help define such outlines, FRED will display a "background" image to use as a reference when editing the curves.

2. Summary of commands

The Alto screen is divided into three areas: a display area for drawing spline curves, a menu area and a message area. User input comes mainly from the mouse, when the cursor is in the display and the menu areas. The result of an interaction usually shows as a new curve in the display area.

FRED displays a menu of commands which are invoked by pointing at them with the cursor and pressing any mouse switch. In response to certain of these commands, another menu of subcommands may in turn appear. Subcommands are invoked in the same fashion. FRED commands are described in the following sections of this document:

section 3: basic operations

3.1: m a ke

3.3: r e p l a ce

3.5: n e xt

section 4: transformations

4.1: m o ve

4.2: c o py

4.3: d r ag

4.4: r e p e at

section 5: other operations

5.1: w i pe

5.2: u n do

5.3: b r e ak

5.4: j o in

section 6: refresh

6.1: r e f r e sh

6.2: s h i ft

6.3: n ew b a c k g r o u nd

section 7: r e ad, w r i te, p l ot

section 8: f o nt

In addition, the main menu offers two simple commands:

k n o t s: spline curves are drawn with or without their knots explicitly represented, depending on the context. This command is used for displaying all the knots on all the curves (they are drawn as "x" shaped symbols).

q u i t: for returning to the Alto operating system. This command expects confirmation with a key stroke (Y or return).

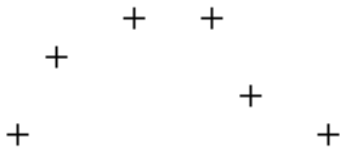
Certain commands use keyboard interaction. When inputing a text string (such as filename) or a number, terminate with return or escape; edit with backspace which deletes the last character, and delete for starting over. Entering only return usually aborts the command. Entering only escape may either abort or imply some default value.

3. Basic operations

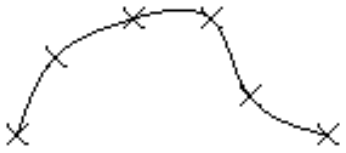
Spline curves can be created with the command m a k e. They can be deleted and modified (by deleting knots, moving knots or adding new knots) with the command r e p l a c e. The operation r e p l a c e applies to a section of a curve, that is to say an ordered set of contiguous knots of the curve. Since the commands m a k e and r e p l a c e are the two most frequently used, they do not appear on the menu but are invoked by pressing switch 3 of the mouse.

3.1 M a k e:

1



2



This is the operation for creating a new curve. First press switch 3. The editor goes into knot input mode (see below): a new menu appears and a small symbol "+" is now attached to the cursor. Now define the knots of a new spline curve. When all the knots of the spline have been defined, terminate knot input mode. The new spline is displayed with its knots turned on. A maximum of 40 new knots can be accepted at one time. However this restriction does not limit the number of knots for a curve since new knots can be added with a r e p l a c e operation.

3.2 Knot Input Mode

Knots are input in the display area by pressing switch 1 or 2 of the mouse. A symbol "+" is displayed at that location, and the number and coordinates of the point are shown in the message area.

If switch 1 is used, a knot is placed at the exact location pointed at by the cursor.

Alternatively, if switch 2 is used, a knot is input only if the cursor is in the vicinity of either a knot on a curve or a previously input knot (i.e a symbol "+"). The new knot will fall exactly at the location of this adjacent knot. The message "overlap" will confirm the input.

Switch 3 is used to terminate knot input, execute the operation and return to the main menu.

In addition, the following actions are available from the knot input mode menu:

e r a se: erase the last knot input;

a b o rt: abort knot input; do not make a spline;

x & y: input a knot by its coordinates.

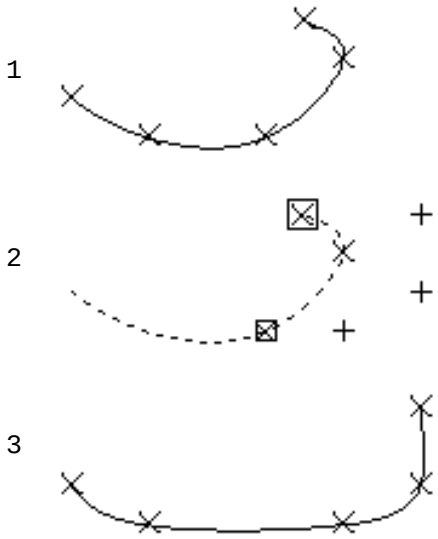
Keys delete and backspace have the same action as the command e r a se.

The menu area also contains an 11 x 11 grid, with a black square in its center which is used for moving the last knot input. When the cursor is placed in the grid and a switch depressed, the last knot will be moved by an amount equal to the distance between the black square in the center of the grid and the square pointed at by the cursor, multiplied by the "resolution" of the grid which depends on the switch used:

switch 1: 1 grid unit equals 1 screen units;
switch 2: 1 grid unit equals 10 screen units;
switch 3: 1 grid unit equals 100 screen units.

For instance, if one points at the square immediately to the right of the black square using switch 2, the last input knot will be moved by ten screen units; if one points at the top left square of the grid using switch 1, the last input knot will be moved up and left diagonaly by five screen units in each direction.

3.3 R e p l a c e:



This operation replaces a curve section by a set of new knots. First specify a curve section (see below). Then press switch 3. The editor goes into knot input mode (already described in section 3.2). Now input new knots. When the set of new knots has been defined, the modified spline is displayed with its knots turned on. The set of new knots may be empty (in this case, the curve section is deleted).

3.4 Specifying a curve section

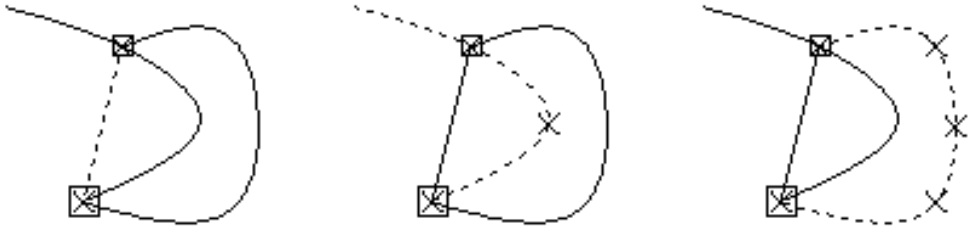
A curve section is an ordered set of contiguous knots of a curve. It is defined by its end knots. Switch 1 and switch 2 are used to specify a section. As seen above switch 3 is used for invoking the commands m a ke and r e p l a ce: if a curve section is currently selected the operation r e p l a ce is invoked; otherwise the operation m a ke is invoked. An unwanted selected section may be suppressed with either delete or backspace.

The first knot of the section is specified by pointing at it with the cursor and pressing switch 1 of the mouse. It is displayed with a small square surrounding it. The last knot of the section is specified similarly with switch 2, and is displayed with a slightly larger square surrounding it. The first and last knot will coincide, when either switch 1 or switch 2 is used, in the following two cases: no section was previously selected, or the previously selected section was on a different curve from the one just pointed at.

The entire curve containing the selected section is drawn as a dotted line, with only the knots of the section turned on. The end knots of the section are surrounded by a square. In addition to the visual cues, a message is displayed indicating the spline number and the knot numbers of the selected section; that information may be helpful in some ambiguous cases.

3.5 N e x t:

There may be ambiguity about which curve is selected by the specified section when two or more curves share end knots of the section, or when one of the end knots is a multiple knot of a single curve. The command n e x t may then be used to cycle through the possible choices. In most cases, the visual cues (dotted curve and visible knots) should be sufficient to indicate which is the current choice. The following figures illustrate typical examples of the use of n e x t.

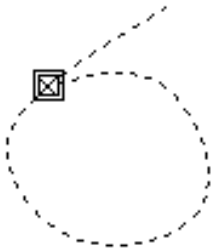


1 2 3

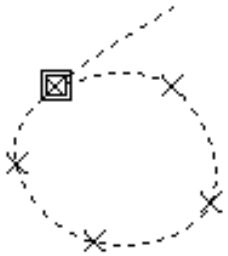
Three curves having two common knots; the possible sections which may be selected by pointing at these common knots are: 1) the leftmost spline, which is a line segment since it has only two knots; 2) three knots from the four-knot spline in the middle; 3) the whole five-knot spline on the right.

1 2

A closed curve; the possible selected sections are: 1) knot 1 through 7 (i.e. the whole curve); 2) knot 1 through 2.



1



2

A closed curve; the possible selected sections are: 1) knot 2 or knot 7; 2) knot 2 through 7, or knot 7 through 2.

The sense of the selected section of the curve (observable by the relative size of the square symbols defining the beginning and end of the section) is important: the designated knots are replaced in that order. There can be ambiguity only when the section contains exactly one knot. Then the order in which the new knots are inserted into the curve is the internal order of the knots of the curve. This order may be found by observing the direction in which the curve is drawn or deleted. Alternatively, the problem can be circumvented by always replacing at least two knots.

3.6 Summary of mouse switche use:

Top level:

switch 1	curve section (first knot)
switch 2	curve section (last knot)
switch 3	m a ke or r e p l a ce

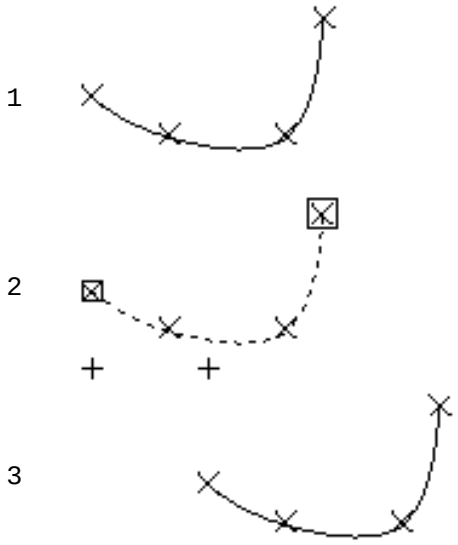
Knot input level:

switch 1	knot input
switch 2	knot input (overlap)
switch 3	execute

4. Transformations

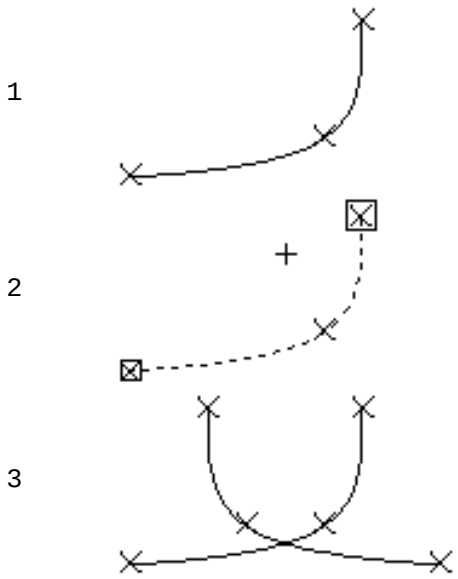
Splines curves may also be modified with several transformation operations: m o ve, c o py and d r ag. These operations all apply to a section of a curve.

4.1 M o v e:



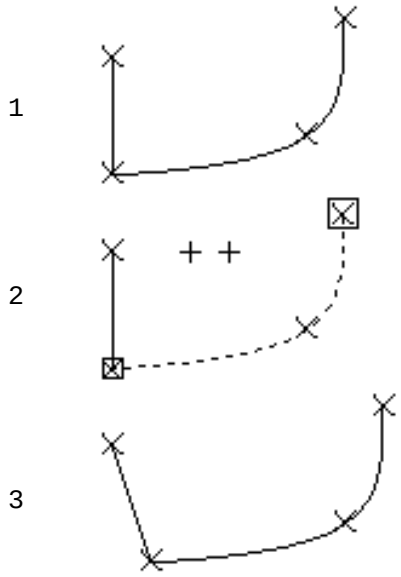
This command does one of three geometrical transformations on a curve section: a translation, a vertical symmetry or a horizontal symmetry. First specify a curve section (see above: 3.4). Then point at one of the three options of the command m o ve: t r a n s l a t e, h o r i z o n t al s y m m e t r y, v e r t i c a l s y m m e t r y. Then the editor goes into a mode identical to knot input mode (see above: 3.2). However only one or two points are specified. They define the geometrical parameters of the transformation. For a t r a n s l a t e, define the origin point and the destination point (this is illustrated on the left). For a h o r i z o n t al s y m m e t r y, define one point on the horizontal axis of symmetry; For a v e r t i c a l s y m m e t r y, define one point on the vertical axis of symmetry (this is illustrated below, in the context of a copy command).

4.2 C o p y:



This command makes a transformed copy of a curve section. It is otherwise the same as the m o ve command. The illustration on the left demonstrates v e r t i c al s y m m e t ry.

4.3 D r a g:



This is a version of the command m o ve (translate) in which all the curves sharing the knots of the translated curve section are modified accordingly. Knots common to several curves, such as end knots of connected curves, may thus be translated in one single operation.

4.4 R e p e a t:

This command will repeat the most recently applied transformation(m o ve, c o py or d r ag) to the current selection with the same parameter (i.e. same translation vector or same symetry center).

4.5 Simple combinations:

Deleting a knot, a curve or a portion of a curve is easily done by executing a r e p l a ce and then a do it without supplying a set of new knots.

Moving a single knot can be done in two ways: r e p l a ce or m o ve.

Inserting N new knots between two consecutive knots k_1 and k_2 is done with a r e p l a ce selecting k_1 and k_2 respectively as the end knots of a section; then input N+2 points such that point 1 coincides with k_1 (using switch 2), points 2 to N+1 are the N new knots, point N+2 coincides with k_2 (using switch 2).

Appending N new knots at either end of a curve is done in a similar way: select the end knot as a single knot section, and r e p l a ce by N+1 new knots. However, be aware of the ambiguity associated with single knot sections (3.5).

5. Other operations on spline curves

5.1 W i p e:

This operation deletes all displayed curves. Beware: no confirmation is expected. An accidental w i p e may be recovered from with the u n do command (5.2). A w i p e is actually equivalent to a succession of single curve deletions. Therefore it will take an equal number of successive u n do operations to recreate all the deleted curves.

5.2 U n d o:

Spline curves are created, go through a history of modifications, and may eventually be deleted. The u n do feature is provided for recovering from destructive events in the history of curves, that is modifications and deletions. It applies to the operations r e p l a c e, v and w i p e. It does not apply to other types of operations (i.e. m a ke, c o py, r e a d and j o in), since they are easily invertible.

All deleted curves and all modified curves are chronologically "remembered," up to some finite variable depth. The most recently deleted or modified curve is recreated when the command u n do is invoked. If that curve had originally been modified (through a r e p l a c e or m o ve) the curve that was substituted for it disappears permanently. The depth of "memory" is variable, because it is a function of the internal storage available to the spline editor. The "memory" will be expunged of its oldest items according to these requirements. It is believed that if FRED is not used extravagantly, the depth of "memory" is about a dozen items. Immediately after a w i p e, all deleted curves should be recoverable.

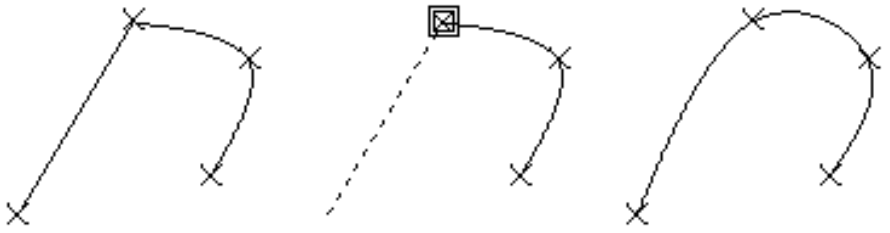
5.3 B r e a k:

This operation is used to break one single curve into two connected curves. First select the knot where the "breaking" is to happen, and then execute this command.

1 2 3

5.4 J o i n:

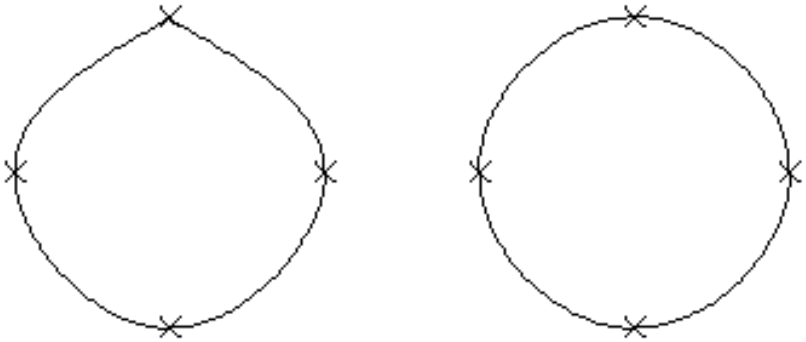
This is the inverse of the b r e a k operation. First select the common end knot of two connected curves, and then execute the command. The two connected curves are joined into one single smooth curve. The command is not executed if there is ambiguity, namely if there are more than two curves with the same end knot.



1 2 3

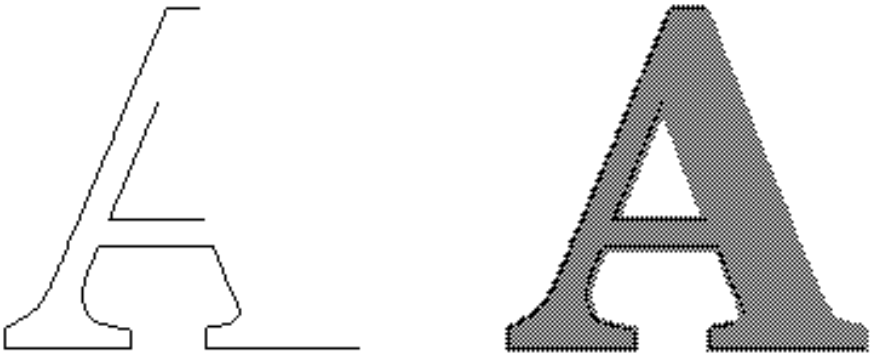
5.5 Cyclic curves:

The j o i n operation may also be applied to a closed curve. This will produce a cyclic curve with a smooth junction. A cyclic curve does not have any end points. It may be broken at any of its knots.



closed cyclic

6. Refresh



The display area may be viewed as a background overlaid with a transparency on which curves are drawn. The background picture is a "one bit per point bitmap" where dark areas are represented as gray halftone.

6.1 R e f r e s h:

Because of the particular way in which curves are drawn and deleted, the display area may get dirtied in regions where curves cross or overlap each other, and where knots coincide. Therefore a command is provided for refreshing the display area. This is a reasonably fast operation which regenerates the background and produces a clean display of spline curves without knots. The current selected section, if any, disappears. The r e f r command comes in two flavors: with a clear background or with the current background.

6.2 S h i f t:

This is a r e f r command combined with a translation of all the curves. The translation is specified as for a m o v e command: source point and destination point. The background, if displayed, is not translated.

6.3 N e w b a c k g r o u n d:

In order to obtain a new background, a character dot matrix may be read from a file in CU format. This character matrix will be expanded so as to fill a maximum area of the display, and the character will be displayed in gray halftone. The expansion factor is the same for all the characters in the same CU file, as it is determined by the constant height of the matrix and the width of the widest character. The interaction scenario is as follows: type the name of the CU file which will cause the file to be scanned for its content (be patient); alternatively, if the same CU file is used as before, only type escape, since the file does not need to be scanned again; then the list of the characters the CU file contains is displayed; now type the desired character (or type escape followed by the octal code).

7. File input/ouput and plotting

7.1 R e ad and W r i t e:

Two commands permit reading and writing the displayed splines, without concern for whether these splines form a well-defined character outline. Arbitrary sets of splines may thus be stored and retrieved. This is the same file format as used by the illustrator program DRAW[1] (the recommended file name extension is DRAW). When reading pictures generated by DRAW, text and curve brushes are ignored.

7.2 P l o t:

Plotting of the picture is done using the PRESS file format. The command p l ot outputs the picture as a bitmap in PRESS format.

The file may be printed on EARS through MAXC; for this you may use the command file PRINT.CM which FRED generates. The file may also be used by programs accepting PRESS files: for instance, MARKUP[2] may be used for inserting the bitmap picture into a document.

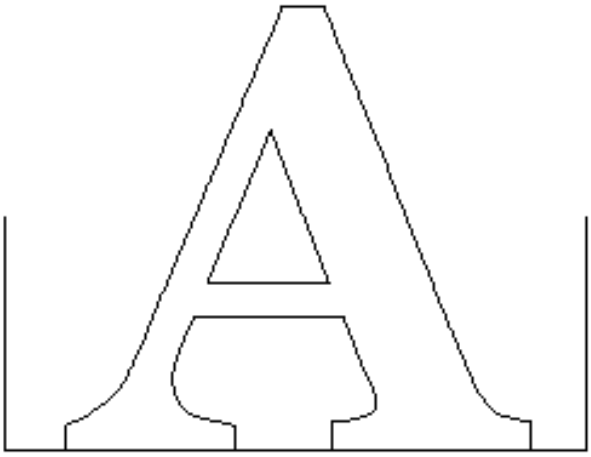
---

[1] Documentation on <GR-DOCS>DRAW.EARS

[2] Documentation on <ALTODOCS>MARKUP.EARS

8. Making a font

The main intended use of FRED is for making fonts, or more precisely creating spline outlines of characters.[1] Spline characters are generated using the curve editing features of FRED (described in section 3 and 4). Additional commands are provided for storing in a file and retrieving from a file such a character description, as well as for specifying the additional information necessary for fully defining the font. These commands are available from a submenu which scrolls in when the command font is invoked.



Section 8.1 first describes the various elements composing a spline font description. Then section 8.2 explains the various commands and methods for creating and modifying these components. Section 8.3 presents the file input/output commands.

Generation of the appropriate fonts for various devices, using spline fonts, is done with the program PREPRESS.[2]

8.1 Description of a spline font:

A special LISP-compatible text format is used for spline fonts (given in appendix, for the very curious). The recommended extension for such a file is SF. A spline font description contains the following components for each character:

a) character outline: it is composed of a number of closed curves made of a number of end-to-end connected splines.

---

[1] R.F. Sproull, "Fonts project", September 9, 1974.

[2] Documentation on <GR-DOCS>PREPRESS.BRAVO

b) base line and width: or more precisely, the position relative to the outline of the character of the horizontal base line, the left side of the character slug and the right side of the character slug.

c) fiducials: the spline outlines generated by FRED are intended to be used by the program PREPRESS which "scan-converts" the character, i.e. generates the actual dot matrix used on a printing or displaying device. The actual resolution of the dot matrix will be a function of the resolution of the device (for instance 500 lines/inch) and the desired point size of the displayed or printed character (say 12 points).[1] In order to guarantee that the scan-converting process will produce an appropriately scaled dot matrix font from a given spline font, there must be some means to relate the particular coordinate system used for the spline outline to the size of the final dot matrix. For that purpose, each character definition contains a set of two numbers called fiducials. These two numbers are respectively equal to the height and width in the coordinate system of the spline outline of a square whose side is equal to the point size of the character. These numbers are used to determine the scale factor to apply both vertically and horizontally to the spline coordinates for producing a dot matrix for a particular point size.

d) character identification:

family: e.g. Baskerville;

character: e.g. "A", or octal ASCII code 101;

face (or style), which has three components: bold or medium or light, regular or italic, condensed or regular or expanded (defaulted to medium, regular, regular).

e) bookkeeping information: version number, creation date, and name of file used for background.

8.2 How to create a spline font:

FRED can define all the components of a spline font with a number of special purpose commands.

a) character outline: in general practice, this outline is generated by creating and editing splines to follow the contours of a halftone character displayed as a background (section 5.2). There are two typical cases. The background character could be obtained from an existing font (in dot matrix format) for a device such as Alto, VTS or SLOT, which one wants to convert to the more general spline font format. Alternatively (and the most likely in the future), one could create an original font in spline outline format. For this purpose one would first create a digitized picture

---

[1] The point is a unit of type measurement equal to 1/72 inch (vive le systeme metrique...).

of the type font to use as the background. In either case, CU file format is the standard, since it is the format used by the video font digitizing system. The recommended resolution for digitized type font pictures is 256 by 256; this creates rather large files but provides a background with minimally jagged contours which are easier to fit with spline curves.

b) base line and width: current base line and width may be modified or redefined in only two ways: with the command b a s e & w i d t h, by reading a character definition from a spline font file. The command b a s e & w i d t h actually activates a special mode for defining an arbitrary rectangle in the display area (which is also used for defining fiducials). A submenu scrolls in, with the following commands:

l e f t a n d r i g h t: when that mode is activated, switch 1 is used for defining the left side of the rectangle, switch 2 the right side. Switch 3 is unused.

t o p a n d b o t t o m: when that mode is activated, switch 1 is used for defining the top side of the rectangle, switch 2 the bottom side. Switch 3 is unused.

m o v e: use any switch to reposition the bottom left corner of the rectangle, its dimension unchanged.

h e i g h t w i d t h: input at the keyboard the desired dimensions of the rectangle (in screen units), the bottom left corner remaining fixed.

ok: terminate, i.e. return to f o n t command.

When the command b a s e & w i d t h enters the rectangle defining mode, a rectangle is displayed corresponding to the current values of base line and width. You may then modify base line and width by redefining the bottom, left and right side of this rectangle, moving the rectangle around (which affects only the base line) or eventually typing in the value of the width.

As an additional option, width (but not base line) may be automatically obtained from the CU font character currently used as a background. This is useful when converting an already existing font. The option comes in the form of a question when entering the command b a s e & w i d t h.

c) fiducials: current fiducials may be modified or redefined in only two ways: with the command f i d u c i a l s, by reading a character definition from a spline font file. The command f i d u c i a l s activates the same mode as the command w i d t h (for defining an arbitrary rectangle in the display area). It is described above (8.2 c).

When the command f i d u c i a l s enters the rectangle defining mode, a rectangle is displayed corresponding to the current values of the fiducials. However only the dimensions of this rectangle (or square) are important. Its position on the screen are irrelevant. You may then modify the values of the fiducials by redefining the top, bottom, left and right side of this rectangle, or eventually by typing in the values.

As an additional option, fiducials may be automatically computed from the CU font character currently used as a background. This is useful when converting an already existing font to spline format. The option comes in the form of a question when entering the command f i d u c y. You must prepare for that option when reading a new CU file: answer yes to the question "Do you want FIDUCIALS automatically computed?"; then enter the point size of the font to be converted, and the resolution of the printing device (500 lines/inch for EARS fonts).

However, when creating an original font, the recommended practice is to digitize a picture of the font type also containing some marks or graduation indicative of the point size of the font. These marks will appear on the screen as part as the background, and the fiducials will be defined by pointing at them.

d) character identification and bookkeeping information are defined or modified through a command labelled m i s c e l l which provides some self-explanatory keyboard interaction.

8.3 Reading and writing spline font files

One SF file may be opened at a time, for reading, writing or both. Opening a file, or creating a new file, is done with the command g e t. The filename must have extention SF. Getting a font file (say FOO.SF) may take some time if it contains many characters, as it implies scanning the file and duplicating it under the name FOO.XF. Beware that SF files grow fast: for efficiency, it is recommended not to store much more than a dozen characters into one single SF file. When quiting or when getting another SF file, the previously opened SF file is closed. Confirmation is expected before closing the file. Confirming with a V (for verify) allows selective deletion of unwanted characters from the file being closed. After file FOO.SF has been closed, FOO.XF will be a copy of the initial file FOO.SF.

Do not exit from FRED by any means other than q u i t. There are ways to recover from the effect of a crash or other similar disruption, but they require expertise.

Characters may be randomly read from, or written on the currently opened SF file. Specify a character by typing either a single key, escape followed by octal code, or return to abort. Overwriting a previously stored character requires confirmation. The r e a d c h a r a command displays a character directory of the opened file.

The command d e f i n e differs from w r i t e c h a r a in that it automatically goes through the commands b a s e w i d t h, i d u and m i s c e l l before proceeding to write out the font definition. When writing out the font outline, all splines not forming a closed curve will be ignored. This means that auxilliary curves created as templates or used as constructive elements, that is to say not actually part of a character outline, do not have to be deleted at the time of writing.

9. Keyboard commands

Command input may be done on the keyboard (as well as from the menu) for most operations at the top level. This allows faster interaction for the experienced user.

The key corresponding to a command is simply the first letter of that command: e.g. key command U is equivalent to menu command u n do. There are only a few exceptions:

- r e p e at is escape;

- keys M and C are used to set the meaning of keys T, V and H to be either a m o ve or a c o py operation (t r a n s l a t i o n, t i c al s y m e t ry, h o r i z o n t al s y m e t ry);

- mainly for safety reasons, w i pe is done with <control>W;

- background and refresh operations also use control keys:

  <control>B r e f r e sh with background

  <control>C r e f r e sh with clear background

  <control>N n ew b a c k g r o u nd

- in addition, delete and backspace are used to suppress an unwanted selection.

10. Getting started

Obtain the file <GRAPHICS>FRED.DM and LOAD it. It contains the following files:

- the program files: FRED, FREDOV1.BB to FREDOV5.BB;

- the menu picture files: MENU1.FRED to MENU4.FRED;

- a utility program SFMUNCH for processing spline font files (described in Appendix A).

Acknowlegments

This document greatlybenefitedfrom help and suggestionsby BillBowman and Bob Sproull.

A p p e n d i x A

SFMUNCH

This is a utility program for processing spline fonts: concatenation of SF files, setting fiducials, and character transformations (shearing-for italics-, condensing and expanding). The syntax of the command is as follows:

SFMUNCH <output SF file> <operations> <list of input SF files>

The available operations are:

i/I	incline characters by the specified slope percentage i;
e/E	expand characters by the specified percentage e;
c/C	condense characters by the specified percentage c;
xf/X	set x fiducials to the given value xf;
yf/Y	set y fiducials to the given value yf.

If no operation is specified, a simple concatenation of the SF files is done. Transformation specifications may be mixed with the list of input files. They take effect only for the input files following them.

In addition, when /V is used, confirmation is expected before processing and writing out each character.

Examples:

```
SFMUNCH METEOR.SF METEOR*.SF
```

concatenates all METEOR characters into a single file;

```
SFMUNCH/V METEOR.SF METEOR*.SF
```

or

```
SFMUNCH METEOR.SF/V METEOR*.SF
```

selectively concatenates METEOR characters into a single file;

```
SFMUNCH METEORI.SF 10/I METEOR.SF
```

generates a font file of pseudo-italics (10 per cent incline);

```
SFMUNCH NUMSYM.SF SYMBOLS.SF 15/E NUMERALS.SF
```

generates a font file of symbols and expanded numerals.

A p p e n d i x B

Font file format

The following description uses the notation:
<...> is a list,
{...} is a string,
[...] is a number.

A spline font file has the form:
<character description> ... <character description> STOP

where <character description> is either of the form:
((FAMILY {family name})
(CHARACTER [code])
(FACE { B | M | R } { R | I } { C | R | E })
(WIDTH [width in x] [width in y])
(FIDUCIAL [dimension in x] [dimension in y])
(VERSION [number] {date})
(MADE-FROM {file name}
[x character origin] [y character origin]
[x fiducial origin] [y fiducial origin])
(SPLINES <closed curve> ... <closed curve>))

or of the form:
((FAMILY {family name})
(CHARACTER [code])
(USE {family name} [code]
{ B | M | R } { R | I } { C | R | E }))

where <closed-curve> is:
(<spline> ... <spline>)

where <spline> is:
([n] <knot list> <weight list> <derivative list> {solution method})

where [n] is the number of knots,
and <knot list> is:
(([X_1] [Y_1]) ([X_2] [Y_2]) ... ([X_n] [Y_n]))
and <weight list> is:
([W_1] [W_2] ... [W_n])
and <derivative list> is:
(([X_1'] [Y_1'] [$X_1"$] [$Y_1"$] [$X_1"'$] [$Y_1"'$]) ...
... ([X_{n-1}'] [Y_{n-1}'] [$X_{n-1}"$] [$Y_{n-1}"$] [$X_{n-1}"'$] [$Y_{n-1}"'$]))
and {solution method} is:
{ NATURAL | CYCLIC | PSEUDO-CYCLIC }

Comments of the form:
(COMMENT {any string})
may be inserted in a <character description>.

FACE information stands for:
BOLD | MEDIUM | LIGHT
REGULAR | ITALIC
CONDENSED | REGULAR | EXPANDED