

Rochester Institute of Technology

RIT Digital Institutional Repository

Theses

2-1-1984

Digital Font Design: Alice

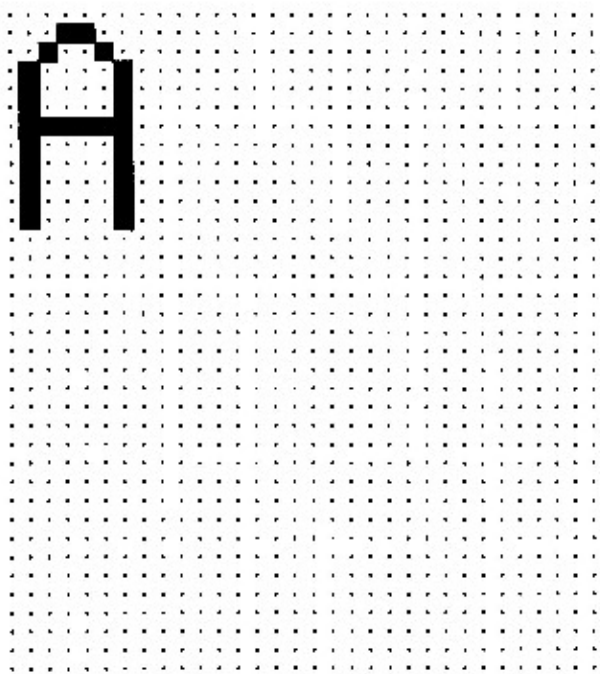
Lynne Garell

Follow this and additional works at: <https://repository.rit.edu/theses>

Recommended Citation

Garell, Lynne, "Digital Font Design: Alice" (1984). Thesis. Rochester Institute of Technology. Accessed from

This Thesis is brought to you for free and open access by the RIT Libraries. For more information, please contact repository@rit.edu.



Rochester Institute of Technology

**A thesis submitted to the faculty
of the College of Fine and Applied Arts
in candidacy for the degree of
MASTER OF FINE ARTS**

Digital Font Design: Alice

**Lynne Garell
February, 1984**

Thesis Committee

James Ver Hague, Chief Advisor
Heinz Klinkon
Mihai Nadin

Approvals

Chief Advisor: James C. Ver Hague

Date: 2/21/84

Associate Advisor: Heinz Kinkon

Date: 2/26/84

Associate Advisor: Mihai Nadin

Date: February, 20, 1984

Graduate Academic Council Representative:

Date: 3/3/84

Dean, College of Fine and Applied Arts:

Date: 3/10/84

I, Lynne Garell, prefer to be contacted each time a request for production is made. I can be reached at the following address:

Lynne Garell
785 Rhode Island Drive
Reno, Nevada 89503

2-27-84

Acknowledgements

**This book is dedicated to my parents.
Thanks, Mom and Dad, for 25 years
of your love, support and encouragement.**

**I would also like to thank Mark Brucks
and Chris Brown of the University of
Rochester for their valuable help and input.**

**And finally, thanks to my committee
members: Jim, Heinz and Dr. Nadin.**

Table of Contents

1. Introduction	1
2. Background	4
3. The beginning	7
4. The development of Alice	9
5. Conclusion	19
Notes	23
Bibliography	24

2 that are used to display text on the screen. Most of these fonts are 5 X 7 dot matrix fonts. Few engineers are trained to design or understand letterforms, with the result that they copy or base their designs on existing types.

Type designers, on the other hand, seem to be mainly concerned with digital typesetting equipment. The typefaces are stored digitally, rather than on film or in the form of cast metal. This means that the finetuning of the typeface takes place at the computer terminal, but the main concern here is with the output, the hard copy.

Some type designers, such as Charles Bigelow and Donald Knuth with Metafont and C. H. Cox with his skeletal approach to font design, have concerned themselves with the design of fonts for digital display. However,

most of this work has been at the experimental level rather than the applied level, and the result is that we still must read poorly designed characters when we use a computer.

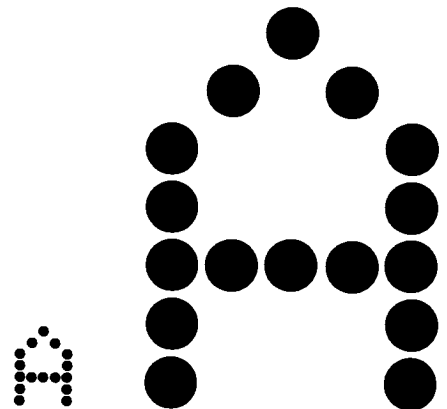
One of the major problems I see is that people want to digitize existing typefaces for use on a cathode ray tube. Types such as Helvetica and Times Roman, designed in 1957 and 1932 respectively, are very popular CRT fonts. These types were designed for the print medium, for ink on paper, and as such they are not meant to be displayed as dot patterns on a computer screen.

It is time for designers to learn computer technology and apply this knowledge to font designs that are produced solely for use in digital display.

Screen resolutions are

constantly becoming greater and greater. The Apple II, a low resolution screen, is not capable of displaying a finely detailed font at text size. The computer I used, the Xerox Alto, is a medium resolution display. A font designed on the Alto would be too detailed for display on a low resolution screen such as the Apple II, and not detailed enough for display on the high resolution screens that are becoming more prevalent. A resolution of 1024 X 1024 or 2000 X 2000 could handle characters with more complex features such as serifs, curves or a variety of angles for diagonal lines. Therefore, a font that is designed at a particular resolution is applicable only to other systems which have similar resolutions.

A "meta-font" is a schematic description of how to draw a family of fonts, not simply the drawings themselves. Such descriptions give more or less precise rules about how to produce drawings of letters, and the rules will ideally be expressed in terms of variable parameters so that a single description will actually specify many different drawings.¹



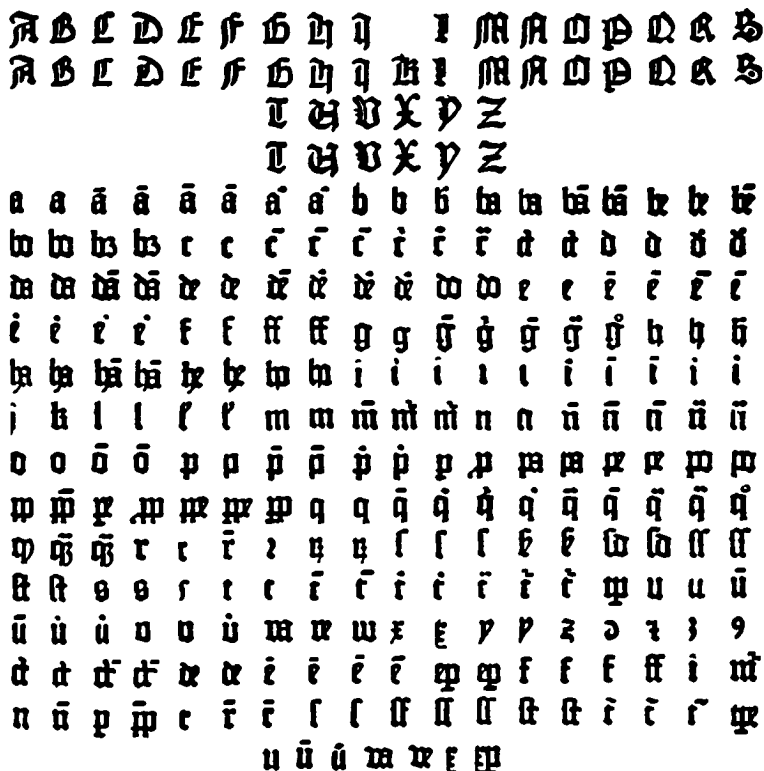
A 5 X 7 dot matrix character.

In about 1453 Johann Gutenberg perfected movable type, which enabled a printer to accurately produce large quantities of type. The result was that printers tried to mass-produce in a short time what scribes had previously spent many tedious hours creating. D. B. Updike writes:

There appears to have been no thought in the minds of early printers other than to reproduce manuscripts quickly and inexpensively. . . . Intent upon imitating manuscripts, they felt obliged to reproduce the kind of letters that a reader had been accustomed to in volumes written by hand. . . . In other words, to the first type-cutters printing was merely an evolution, and did not appear a new invention in the sense that it obliged them to decide what forms of letter were best adapted to the new medium they had to employ.²

Over five centuries later we are experiencing the same reluctance to accept a new technology for what it is, and to base our designs on the capabilities of that technology.

Type designers, engineers and computer scientists must recognize that computers are, in fact, a new medium, and therefore require "new forms of letter". Wim Crouwel says, "The memory of a computer is an assembly of cells, charged positively or negatively. This assembly of cells . . . could be a new starting point for the development of new characters."³ My thesis is an attempt to understand this technology and to use that understanding to design a font that fits the medium of the CRT. What follows is a description of how I designed my font, which is called Alice.



Gutenberg's type, which had 288 different characters.

We will doubtless stop the reproduction of Bodoni and Garamond on the supersonic machines. It is an error!⁴

Writing by hand is fortunately a vanishing skill. In the future it will serve only for making rapid abstract notes, which will be of no value except to the writer, and undecipherable except by him. For true communicative purposes its role is finished. The letter-type for our time will, therefore, certainly not be based on the written or drawn examples of the past.⁵

1. m i

2. 

1. A lower case "m" and "i" as typed on a manual typewriter. Notice the narrow spaces between the stems of the "m" and the extra-long serifs of the "i"
2. Diagonal characters often cause problems in letter-spacing. Notice the space between the "r" and the "v".

2. Background

As I began to work on Alice, there were several preliminary questions to be answered: Would my font be purely technical and highly functional or would it be purely experimental and not necessarily highly functional? What are the advantages and disadvantages of proportional spacing versus monospacing? Do I want a single alphabet, or one with upper and lower case characters? What are the advantages and disadvantages of serif versus sans serif? Will the design process take place at the terminal or at the drawing board?

The first question is the most important because it determines the guidelines I follow in answering the remaining questions. I decided to approach my font from a purely technical point of view, with my goal to be a highly legible and functional font. I had in mind the user who would be typing and reading text matter on the screen; therefore, legibility and ease of reading were very important facets of my design.

Letterspacing is another important problem. Most existing digital fonts are monospaced, for the same reason that early typewriters were monospaced: it's easier to implement a monospaced alphabet. However, these fonts are also awkward and often difficult to read. Putting a lower case "m" in the same set width as a lower case "i" is awkward and it's an eye-catcher. The optimum font is one which reads smoothly without catching the reader's eye. Proportional spaced alphabets are more difficult to implement, but the legibility and eye flow are much improved. Because the computer I used has the ability to do proportional spacing, I decided to design Alice to be proportionally spaced.

The question of a single alphabet is another one of functionality. A single alphabet is one in which there are not two separate designs for the upper case letter and the lower case letter. Instead, one design suffices in both cases. This means that the design for a particular character could be a traditional lower case form or it could be a traditional upper case form, or it could be a totally new form. Some of the single alphabets have also been phonetic alphabets. Many designers who have been concerned with letterforms and legibility have designed single alphabets in an attempt to minimize the number of characters in our alphabet. However, we are taught to read a dual alphabet; we are used to seeing different designs for a capital "A" and a lower case "a". Because of this, a single alphabet would be extremely difficult to implement, and it could be a very slow process. Although the idea is intriguing, I chose to design Alice as a dual alphabet to maintain legibility.

1.

abcdefghijklmnopqrstuvwxyz
a d d

1. Herbert Bayer's universal alphabet of 1926.
2. Single alphabet of 1927 by Kurt Schwitters.
3. Anthony Rozak's phonetic alphabet of 1971.

2.

MÜSİK IM LEBEN DER VÖLKER AM 2. JULI
20 UHR DARJEGERT IM OPERNHAUS
WARSAU BERÜHMTER DARJEGERT WERKE
POLNISCHER MEISTER PREISE 1.5 Mk.

3.

ᄠ ᄠᄡᄢᄣᄤᄥᄦ ᄦ ᄠᄡ ᄠᄢᄣᄤ ᄠᄡᄢᄣᄤ
ᄠᄡ ᄦ ᄠᄢᄣᄤ ᄠᄢᄣᄤ ᄠᄡᄢᄣᄤ
ᄠᄡ ᄠᄢᄣᄤ ᄠᄢᄣᄤ ᄠᄢᄣᄤᄥᄦ

The main purpose of letters is the practical one of making thoughts visible. Ruskin says that all letters are frightful things and to be endured only on occasion, that is to say, in places where the sense of the inscription is of more importance than external ornament. This is a sweeping statement, from which we need not suffer unduly; yet it is doubtful whether there is art in individual letters. Letters in combination may be satisfying

The main purpose of letters is the practical one of making thoughts visible. Ruskin says that all letters are frightful things and to be endured only on occasion, that is to say, in places where the sense of the inscription is of more importance than external ornament. This is a sweeping statement, from which we need not suffer unduly; yet it is doubtful whether there is art in individual letters. Letters in combination may be satisfying and in

Text set in 9/10 Baskerville
(top) and 9/10 Avant Garde
(bottom).

The debate between serif and sans serif fonts is an ongoing one. Many people think that for reading printed text matter the best type is a basic serif face such as Baskerville or Times Roman. It is thought that the serifs help in distinguishing one letter from the next, thereby causing less eye strain and a greater ease of reading. However, I'm not sure that this is true in the case of a CRT. A serif face on a low resolution screen is often less legible than a sans serif because the serifs are more noticeable; they take up a greater proportion of the character. It is very difficult to accurately reproduce finely detailed serifs on a low or medium resolution CRT. Therefore, the serifs are often large and clumsy. I decided to approach my font as a sans serif, with the possibility of using occasional serifs to distinguish certain characters.

When I began to work on Alice my idea was to do all the designing at the terminal, and none on paper. My thought was that typefaces for the print medium are designed with pencil on paper, then transferred to metal or film or bit map. Because I felt so strongly about designing a font solely for display on the CRT, I thought that the design process should take place at the terminal.

3. The beginning

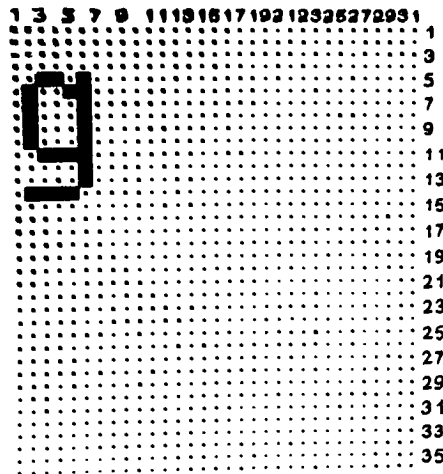
The computer I worked on is the Xerox Alto, with a screen resolution of 606 X 808, and containing 489,648 points. It is a vertical-format 875-line raster-scanned TV monitor with a refresh rate of 60 fields per second from a bit map in the main memory.

The software I used is Flash and Fontedit, written by Clint Parker of the University of Rochester. The font editor has a display that consists of the bit map representation (32 pixels by 36 pixels), the menu and the full font displayed at actual size. Selections were made by operating a mouse. This allowed me to either "ink" or "erase" a pixel; in other words, this is how I controlled whether each pixel

was turned on or off. I also used the mouse to move the entire character either vertically or horizontally within the bit map; I used it to store a completed character, to generate a character onto the bit map representation, and to exit from the font editor.

The font editor appears on the screen as black characters on a white background, but when I viewed the font in text form, I had the option of either white on black or black on white. I think that white on black is easier on the eyes, so from the beginning I designed my font as white characters on a black field. The final design for Alice is also intended to be read as white characters on a black field. When the screen display is reversed to black on white,

FONT EDIT 1.0 -- C. W. Parker

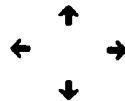


• MOUSE:
 INK --> Right Button
 ERASE --> Middle Button
 PUT --> Right Button
 GET --> Middle Button

TARGETS

WIDTH 7
 CLEAR
 HEIGHT 16
 VERTICAL OFFSET
 EXIT (no WRITE-OUT)
 EXIT (WRITE-OUT)

CHARACTER MOVE



! " # \$ % & ' () * + , - . /
 0 1 2 3 4 5 6 7 8 9 : ; < = > ?
 A B C D E F G H I J K L M N O
 P Q R S T U V W X Y Z [\] ^ _
 a b c d e f g h i j k l m n o
 p q r s t u v w x y z { | } ~ •

Digital type design raises a unique problem of combining computer technology with the type designer's art and artistry. The existing designs have been prepared by mathematicians and engineers aided by a small amount of consultation with a type designer. Eventually, to achieve high-quality fonts, type designers will have to assume this job and will have to learn enough computer technology to express their designs in this new digital language. Merely copying existing fonts has severe limitations.⁶

8 the characters appear thinner and the screen flicker is more apparent.

When I began working on the Alto I spent several days experimenting with the font editor. I used Helvetica to experiment with because it was the simplest available font.

Eventually I was ready to begin the task at hand. I realized that I probably wouldn't get good results by starting with a blank screen, because I had never before attempted a font design. I decided to select a typeface and digitize it, then use that font as a point of departure for Alice. The typeface I selected is Univers 55, designed in 1957 by Adrian Frutiger.

Univers is a well-designed

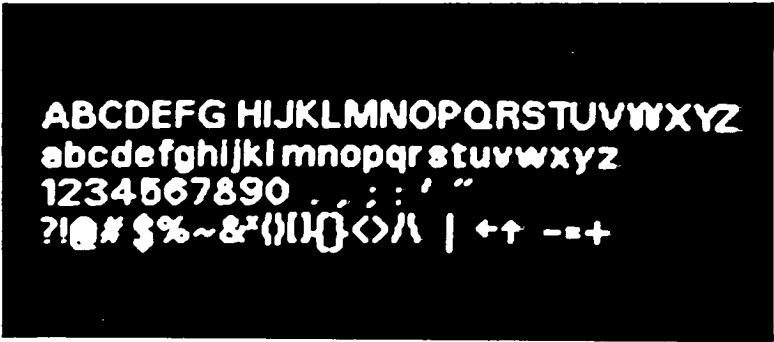
family of 21 sans serif faces that range from condensed to expanded and from light to bold. It is an elegant and simple type, and I felt that it would be a good starting point for my font.

The unretouched Univers font has a character height of 12 pixels. On the bit map representation, the baseline is line nine; the x-height is line four; the cap height is line two and the descender line is line eleven. The average width of a lower case letter is seven pixels, and the average width of an upper case letter is eight pixels.

This unretouched version of Univers 55 is extremely crude. When I read text matter on the screen I realized that one of the most important characteristics of my font would have to be smoothness. Earlier fonts, as well as my version of Univers,

tend to have rough curves and jagged diagonals which are even more obvious when juxtaposed with straight vertical or horizontal lines.

I worked toward an overall smoothness by looking at the characters and trying to analyze places where I could insert vertical or horizontal lines without disturbing the regularity of the letter. Therefore, I wanted as regular a character as possible, because irregularity or unusualness would disrupt eye flow.



ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890 . , : ; ' "
?!@#\$%^&*(){}|<>~`| +- = +

Univers 55, unretouched, as it appeared on the Alto screen.

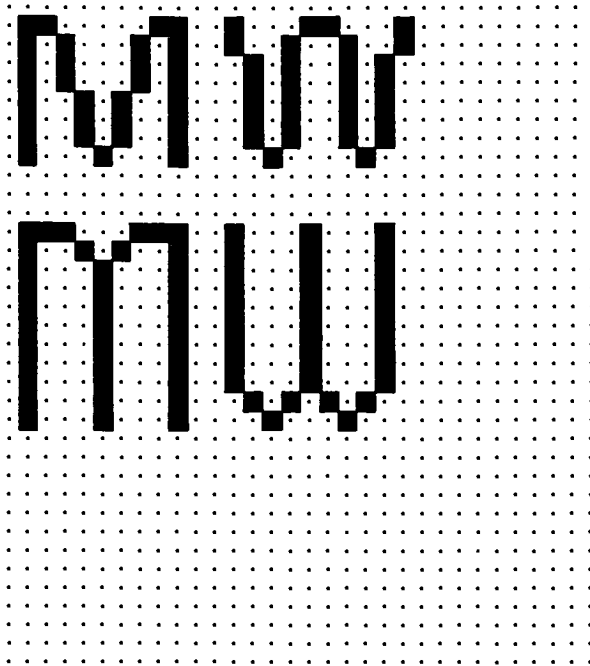
4. The development of Alice

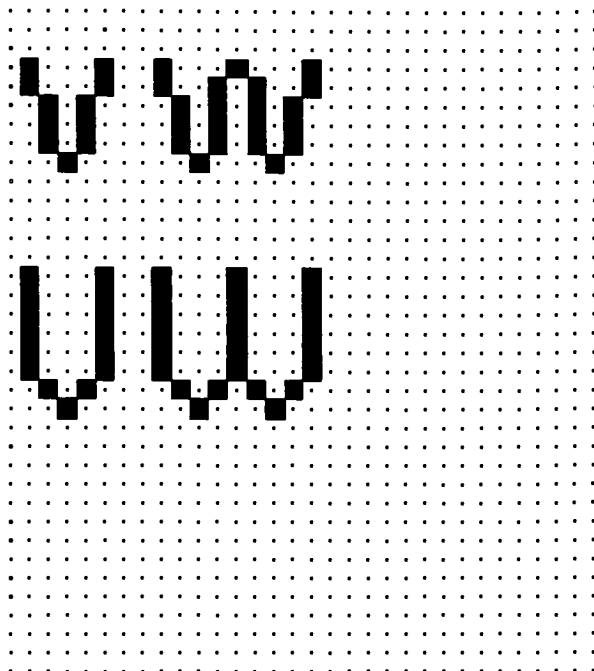
With my goal of a highly legible font, I had to aim for a simple, open design. One way to open up the design was to separate strokes by at least two pixels. Two parallel strokes that are separated by only a single pixel tend to blur together. This happens for two reasons: One, the scanning beam must turn on the first pixel, then turn off, then immediately turn back on. Because of the close proximity of the pixels, the beam probably isn't able to completely turn off before it has to turn back on. This means that what should be a black background could appear grey.

The second cause of blurring is the spread of the phosphor. Although each pixel is technically either turned on or off, there is some spread beyond the exact boundaries of a given pixel, and this small amount of spread can cause some blurring.

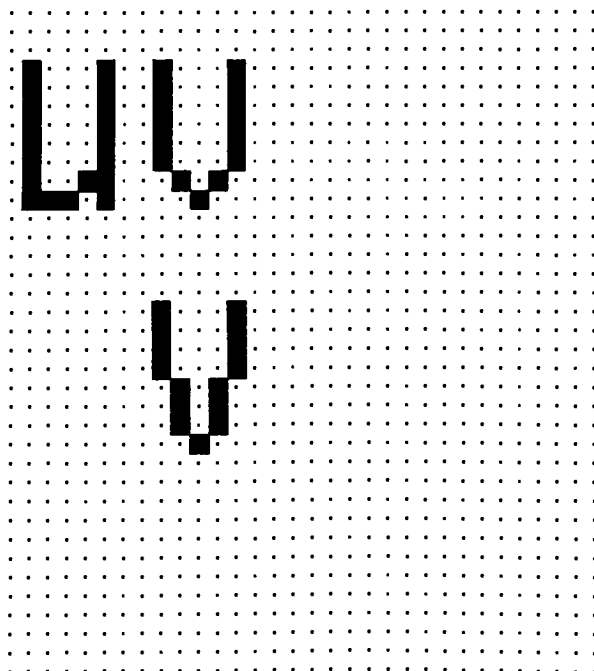
Another way that I simplified the font was to minimize the number of diagonal lines and curves. I replaced these with vertical and horizontal lines whenever it was possible without destroying the identity of each character.

One of the first areas where I applied these two principles was in the upper case "M" and "W". First, I altered each character so that instead of having four strokes, it has three. This allowed me to open up the area between the strokes, as well as to design the strokes as verticals rather than as diagonals.

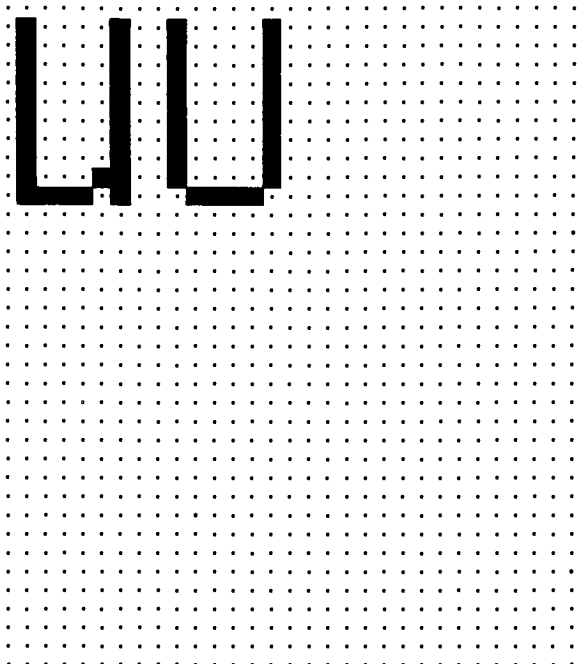




I also altered the lower case “v” and “w” so that the stems are vertical rather than diagonal. The vertices of the characters are formed by short 45° angles that end with a single pixel.



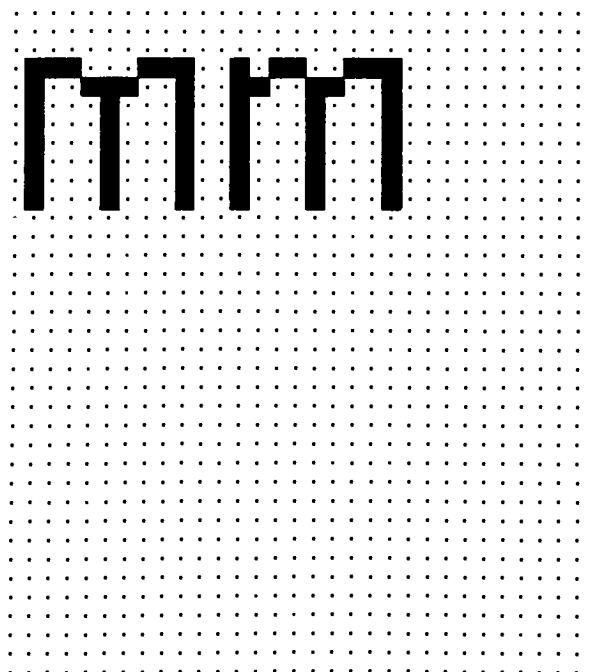
My designs for the lower case “u” and “v” are easily differentiated when placed side by side, but the “v” could have been mistaken for a “u” when seen separately. One suggestion was that the outside edge of the “v” appeared to have the sharp vertex of a “v”, but that the inside edge blurred so that it appeared rounded like a “u”. The bottom design for the “v” is one that I tried in an attempt to further differentiate the two characters. However, this “v” is very awkward; the lines are rough and the character appears blurred near the vertex. I ended up using the original design because I don’t think that there is a big problem in differentiating between the “u” and the “v”.

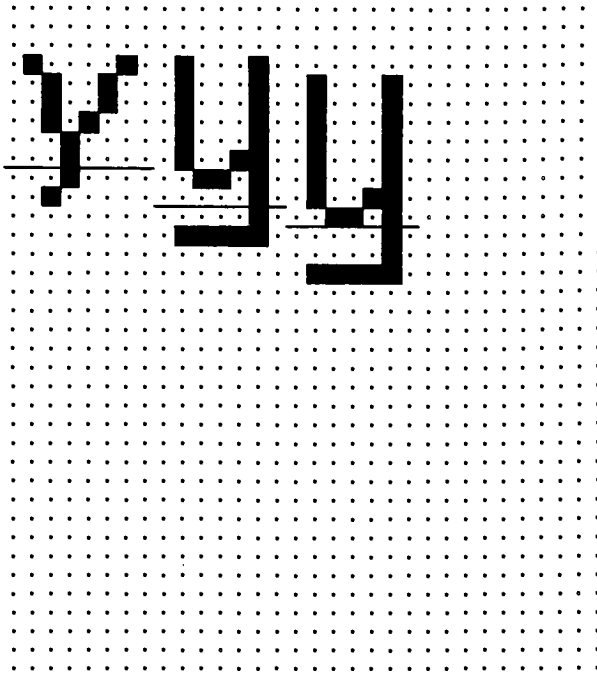


I designed two versions of the upper case "U". The design on the left was an attempt to unify the upper and lower case designs; the capital letter is a larger version of the lower case letter. This design appeared awkward and didn't fit in with the remainder of the font, so I designed the version on the right. It is smoother and not as awkward, and this is the final design of the character.

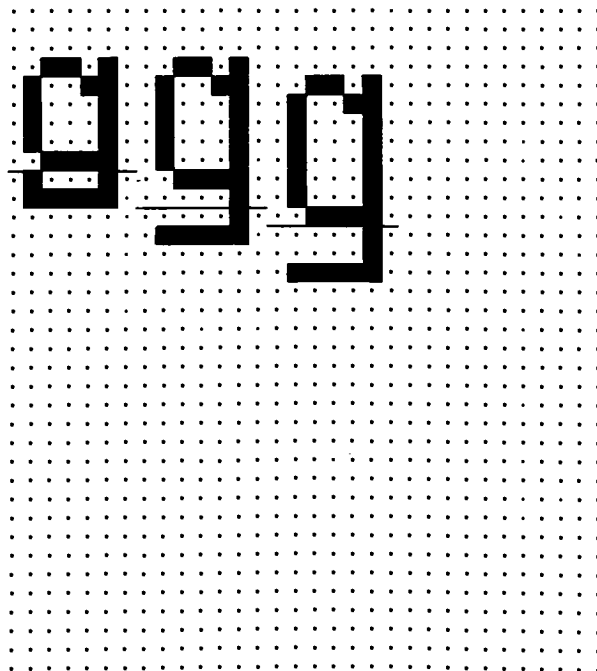


I designed two versions of the lower case "m". The design on the left is the first design. I altered it to form the design on the right, which is more consistent with other lower case characters.



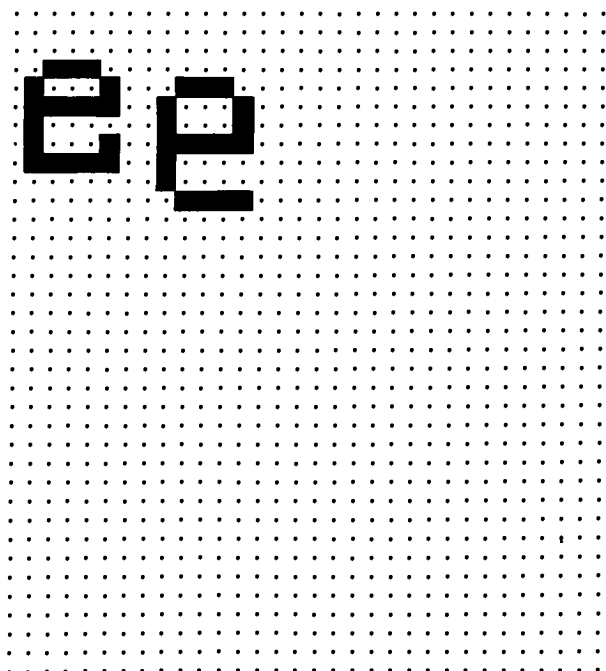
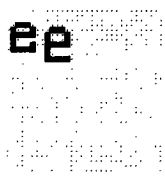


The Univers lower case “y” is extremely rough and jagged. Instead of a letter made of two diagonal lines, I altered the “y” so that it is made almost entirely of smooth horizontal or vertical lines. I also put two pixels of space between the bottom of the bowl and the tail. This was to open up the character so that there wouldn’t be blurring. Initially, I designed the character so that the bowl rested one pixel above the baseline. This was to allow enough space between the bowl and the tail, while maintaining a short overall character height. This created a slightly uneven baseline, and I eventually increased the character height so that the bowls could rest on the baseline.

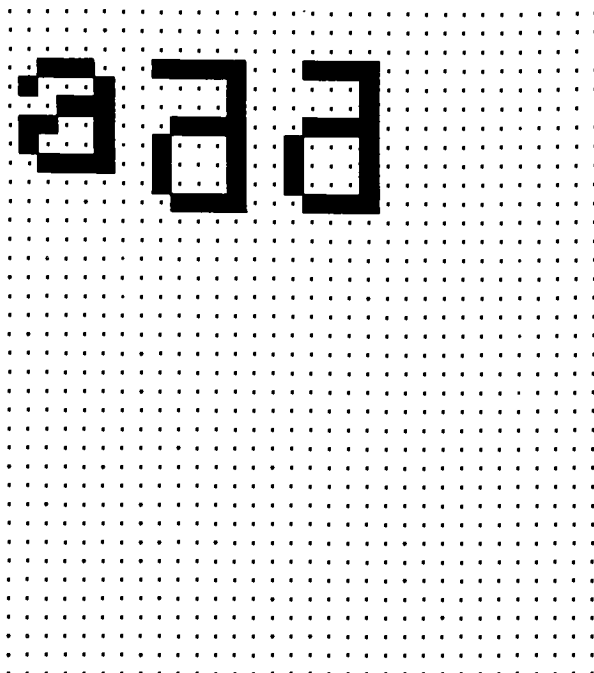


I designed the lower case “g” to follow the design of the “y”. The bowls connect the same way, and the bowl is two pixels above the tail. Again, the bowl of the center design rests one pixel above the baseline, but this was eventually corrected.

Another of the ways in which I opened up the characters of Alice was to expand the x-height. An expanded x-height lends itself to the design of characters with strong verticals rather than diagonals and round curves. It also allows for more vertical space within the character; this is particularly relevant with characters such as the lower case “a” and “e”. The bowls of these two letters often appear crowded together and blurred; in Alice, they appear clean and open.

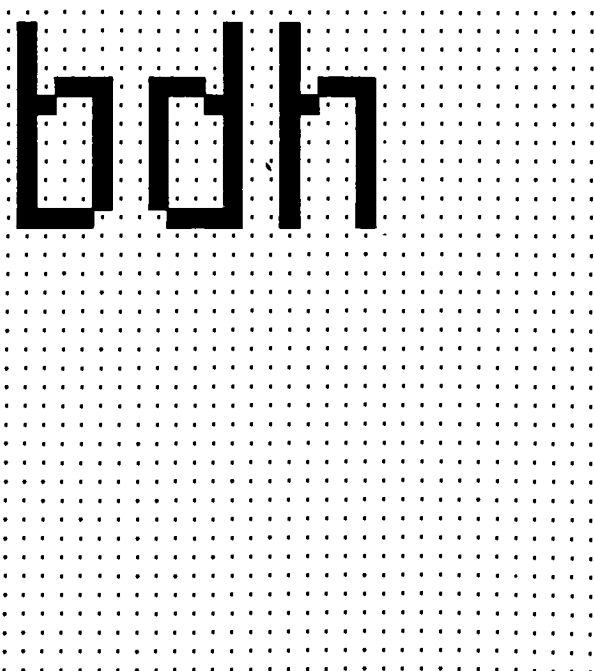


The most commonly noted problem of digital video graphics is that of “jaggies”. A sloped line cannot be represented smoothly by discrete points, and appears rather as an annoying “stair step” pattern. The insertion of intermediate points, halfway between the line and background colors, along the stair steps, smooths the line edges; the visual effect is a continuous, solid line.⁷



bdh

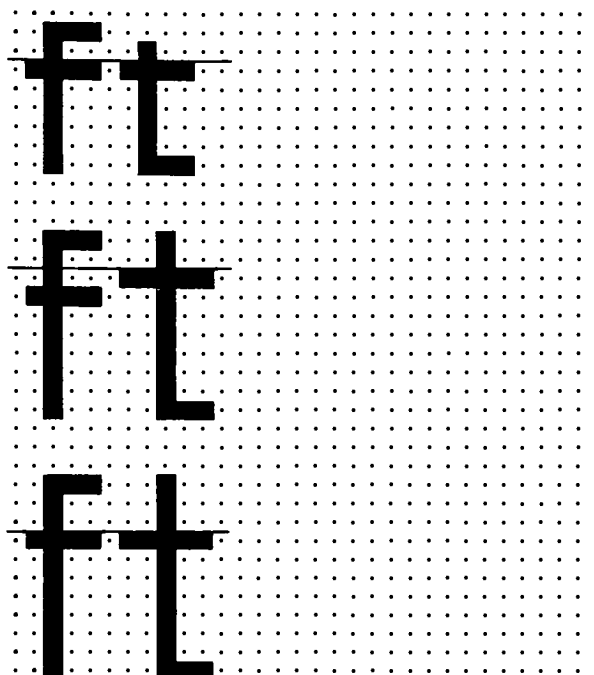
aaa



One of the problems with designing a font for a resolution as low as this is that each pixel makes up such a large proportion of the character. If a pixel is turned off, the character isn't quite right, but if the same pixel is turned on, the character also isn't quite right. This problem was obvious when I worked on the lower case "a". The shoulder curved downward in the original letter, but I converted it to a straight horizontal line. When the shoulder extended the full width of the character, the letter appeared to tip to the left. When I erased the first pixel, the character appeared to tip to the right. Either design was a compromise, but I chose the center design. Similar characters, such as the "e", "g" and "y", have tails that extend the full width of the character, so this design for the "a" was consistent with the remainder of the font.

This illustration shows letters with similar characteristics.

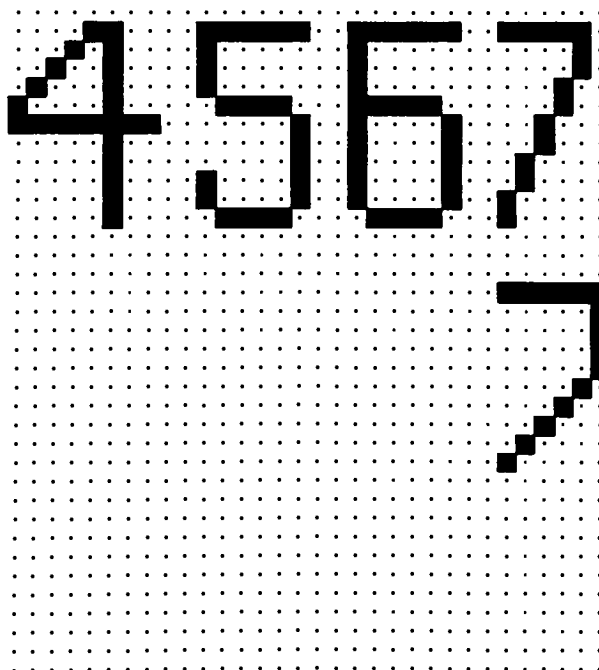
My initial approach to the design of the lower case “f” was similar to that of the lower case “g” and “y”. I wanted to allow two pixels of space between the shoulder and the crossbar, but the character height was such that the only way to do this was to lower the crossbar so that it was one pixel below the x-height. This was another compromise in design that was particularly obvious in “ft” combinations. The crossbar of the “t” remained at x-height, which made an awkward “ft” combination. When I expanded the overall character height, I could put the crossbar of the “f” at x-height and still have two pixels between the shoulder and the crossbar.

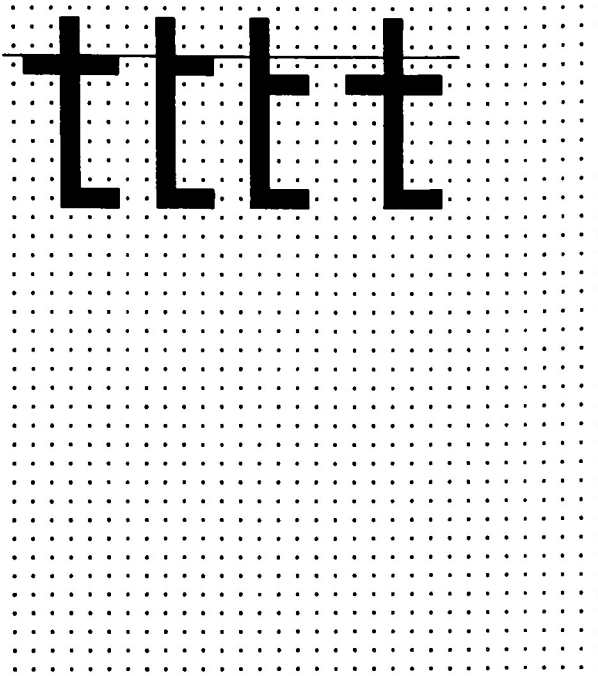


ft
ft
ft

4567
7

The numeral seven is the only character that has a diagonal line that is not a 45° angle. The lower design is one I did in an attempt to design the character with a 45° angle.





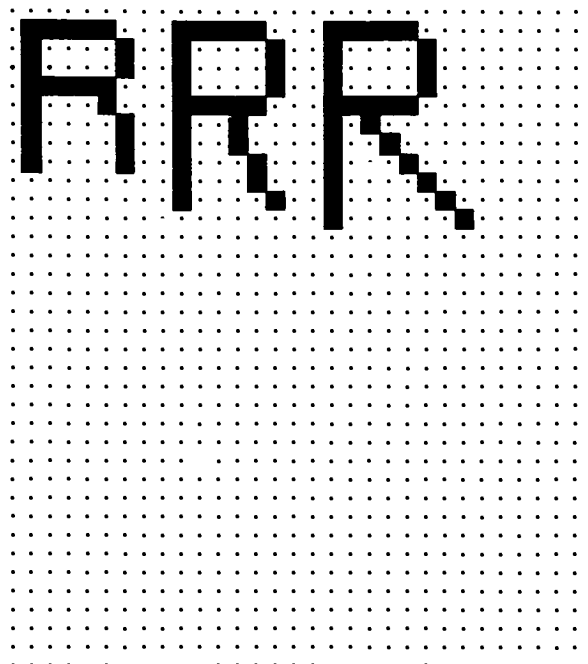
I tried a variety of designs for the lower case “t” in an attempt to resolve the problem with the “f”. First I cut off the left side of the crossbar of the “t”, then I lowered the crossbar by one pixel. This cut-off crossbar worked when the “t” was preceded by other characters, but when the “t” began a word, it wasn’t as quickly recognized as a “t”. The strong vertical line created a channel that the eye wasn’t used to seeing.

My next solution was to leave the crossbar one pixel below the x-height, but to return to the traditional design of a crossbar. This solution, when paired with the lowered crossbar of the “f”, created a very wavy line of text. The only acceptable solution was to expand the character height.

tttt

today

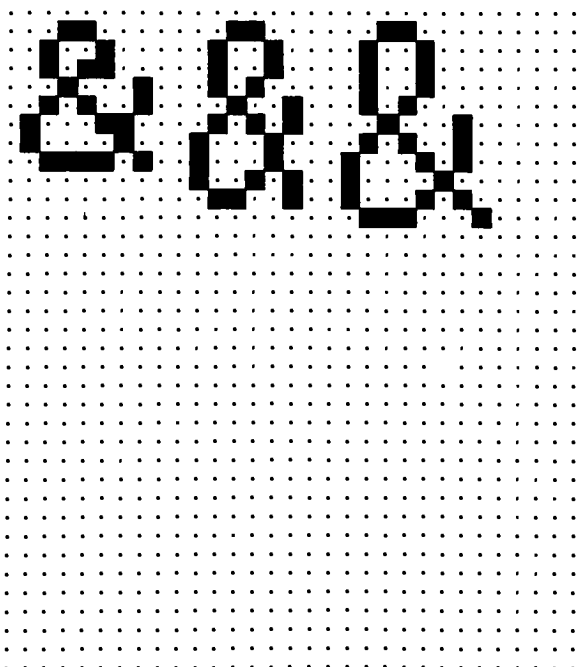
night

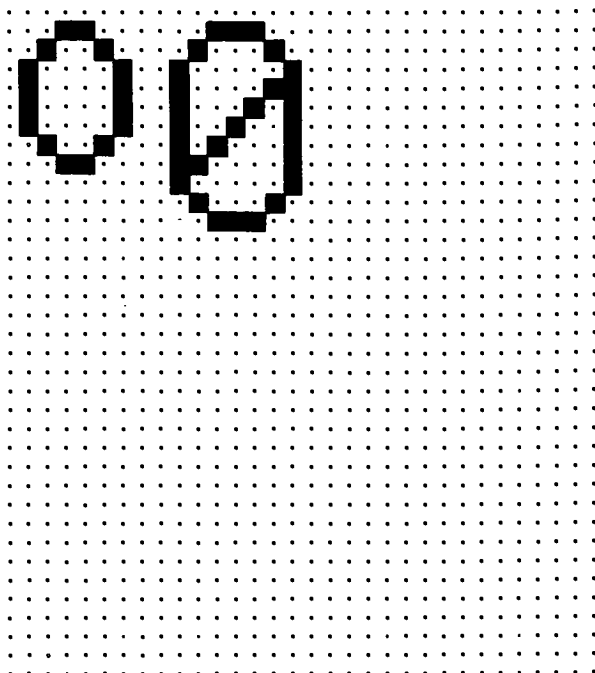


Whenever possible, I tried to use a 45° angle for diagonal lines. I began this with the upper case “R”, which had a rough diagonal line in early designs. After that, I implemented the 45° angle in the ampersand, the slashes and the parentheses.

RRR

&&&

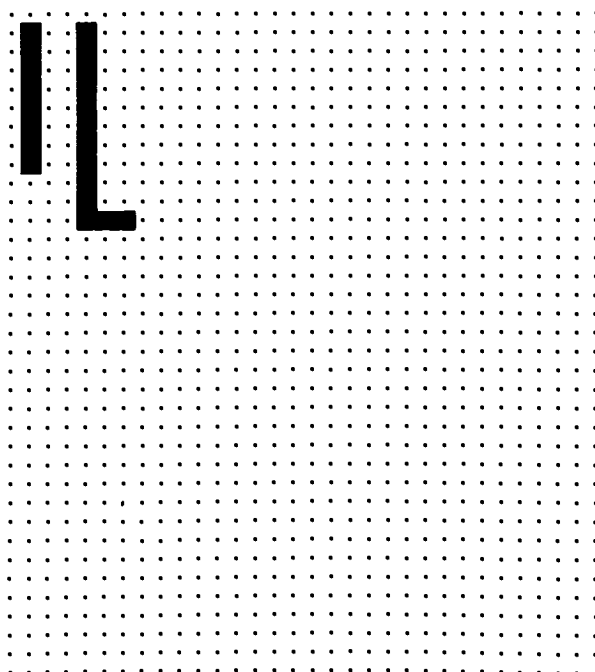




There is often confusion between the numeral 0 (zero) and the letter O (oh). One simple solution to this problem is the diagonal line through the zero. This is the final design of my numeral zero, compared with the original Univers zero.

IL

00



Another area of confusion is between the upper case "I" and the lower case "l". This problem is quickly illustrated in this text, which is set in Helvetica. I solved this problem by adding a foot to the lower case "l"; the foot is of the same design as the foot of the lower case "t".

5. Conclusion

The development of Alice, as with any font, was a tedious process of major and minor changes and adjustments. But it was also a process of experimentation. I had never before designed a font, so this was an attempt to combine what I knew of traditional type designing with what I could discover about the capabilities of a cathode ray tube.

I believe the result is a clean, smooth and elegant font. It is easy to read on the screen. There are few jaggies or rough curves. The characters have an overall consistency and evenness of color. And finally, Alice was designed specifically for digital display.

A grey scale font, while providing smoother curves and diagonals, requires a much larger amount of memory than a two-bit font like Alice. Grey scaling is also an attempt to put a print-oriented typeface where it doesn't belong — the computer screen.

Helvetica, which is available on the Alto, is a much rounder font than Alice, as well as having a rougher overall appearance. It is also wider. The average number of characters per line for Alice is 85, or 3570 characters per screen. Helvetica allows an average of 78 characters per line, or 3276 characters per screen.

I tried putting Alice onto an Apple II. At the same number of pixels per character, the font was much larger on the Apple than on the Alto. At that size the font was more applicable as a display font than as a text font. The letterspacing was also completely off. In most cases, there are two pixels between any two letters in Alice. On the Apple, one pixel would have been sufficient.

Ultimately, Alice should have a contrasting partner font, such as a bold version, an italic or a larger size. Given more time, I would have designed a bold version of the font, as I think that is the most elegant solution to the problem of a contrasting font design.

One of the other problems I had with Alice was in letterspacing. Ultimately, each font that is implemented on a computer should have a letterspacing table as part of the software or hardware. This table would allow for the variety of character combinations, and give each possible combination its own letterspacing parameter. I would like to have been able to do a table like this for Alice, but on the Alto that would have required a programming knowledge that I don't possess. However, the problem of letterspacing has in part been solved with the use of vertical stems rather than diagonal stems. This enabled me to bring characters closer together and avoid the large inter-character spaces that often occur with diagonal stems (for example, the lower case "v").

The following page shows Alice in use on the Alto as a text font, and at the bottom of the page is the full font.

Locate:

Change:

Command:

window: 0 mode: INSERT input: googs2 output: googs2

'Twas brillig, and the slithy toves
Did gyre and gimble in the wabe;
All mimsy were the borogoves,
And the mome raths outgrabe.

"Beware the Jabberwock, my son!
The jaws that bite, the claws that catch!
Beware the Jubjub bird, and shun
The frumious Bandersnatch!"

He took his vorpal sword in hand:
Long time the manxome foe he sought-
So rested he by the Tumtum tree,
And stood awhile in thought.

And as in uffish thought he stood,
The Jabberwock, with eyes of flame,
Came whiffling through the tulgey wood,
And burbled as it came!

One, two! One, two! And through and through
The vorpal blade went snicker-snack!
He left it dead, and with its head
He went galumphing back.

"And hast thou slain the Jabberwock?
Come to my arms, my beamish boy!
O frabjous day! Callooh! Callay!"
He chortled in his joy.

'Twas brillig, and the slithy toves
Did gyre and gimble in the wabe;
All mimsy were the borogoves,
And the mome raths outgrabe.

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz 1234567890
!@#\$%^&*()[]+=-|< > . , ; ' "

1. Donald Knuth, "The Concept of a Meta-Font," **Visible Language** 16 (Winter 1982): 4.
2. Daniel Berkeley Updike, **Printing Types: Their History, Forms and Use**, 2 volumes (Cambridge: Harvard University Press, 1922), 1: 5-6.
3. Wim Crouwel, "Type Design for the Computer Age," **The Journal of Typographic Research** 4 (Winter 1970): 54.
4. Wim Crouwel, "Type Design for the Computer Age," **The Journal of Typographic Research** 4 (Winter 1970): 54.
5. Wim Crouwel, "Type Design for the Computer Age," **The Journal of Typographic Research** 4 (Winter 1970): 53.
6. M. V. Mathews, Carol Lochbaum and Judith Moss, "Three Fonts of Computer-Drawn Letters," **The Journal of Typographic Research** 1 (October 1967): 352-353.
7. Christopher Schmandt, "Soft Typography," **Proceedings of the IFIP Congress 1980**: 1029.

Bibliography

- Bass, Rudi. "The Development of Vidifont." **Visible Language** 5 (Winter 1971): 33-48.
- Bigelow, Charles. "Aesthetics vs. Technology: Does Digital Typesetting Mean Degraded Type Design?" **The Seybold Report** 10-11 (24 August 1981, 8 February 1982, 22 February 1982): 1-16, 1-23, 10-19.
- Bigelow, Charles and Day, Donald. "Digital Typography." **Scientific American** 249 (August 1983): 106-119.
- Boyd, C. E.; Demke, K. R.; and Mumola, J. L. "Font Design Minimizing Flickering in Interlaced Raster Scan Display Using Single Dot Characters." **IBM Technical Disclosure Bulletin** 21 (February 1979): 3496-97.
- Cox, C. H.; Coueignoux, P.; Blessner, B.; and Eden, M. "Skeletons: A Link Between Theoretical and Physical Letter Descriptions." **Pattern Recognition** 15 (1982): 11-22.
- Crouwel, Wim. "Type Design for the Computer Age." **The Journal of Typographic Research** 4 (Winter 1970): 51-59.
- Crow, Franklin. "The Aliasing Problem in Computer-Generated Shaded Images." **Communications of the ACM** 20 (November 1977): 799-805.
- Demke, K. R.; Mumola, J. L.; and Schwartz, A. A. "Dot Matrix Display Font of Small Size But Increased Readability." **IBM Technical Disclosure Bulletin** 21 (April 1979): 4343-46.
- Duncan, W. S. "Proportional Spacing Character Generator CRT Display." **IBM Technical Disclosure Bulletin** 23 (January 1981): 3791-92.
- Edwards, J. C. and Paterson, R. L. "Limited Element Fonts." **IBM Technical Disclosure Bulletin** 25 (June 1982): 381-385.
- Frutiger, Adrian. "OCR-B: A Standardized Character for Optical Recognition." **The Journal of Typographic Research** 1 (April 1967): 137-146.
- Gould, John. "Visual Factors in the Design of Computer-Controlled CRT Displays." **Human Factors** 10 (1968): 359-376.
- Hofstadter, Douglas. "Metafont, Metamathematics, and Metaphysics." **Visible Language** 16 (Autumn 1982): 309-338.
- Knuth, Donald. "The Concept of a Meta-Font." **Visible Language** 16 (Winter 1982): 3-27.
- Kolers, Paul. "Clues to a Letter's Recognition: Implications for the Design of Characters." **The Journal of Typographic Research** 3 (April 1969): 145-168.
- Maddox, Michael. "Two-Dimensional Spatial Frequency Content and Confusions Among Dot Matrix Characters." **SPIE** 199 (1979): 109-116.

Maddox, Michael. "Font Comparisons for 5 X 7 Dot Matrix Characters." **Human Factors** 19 (1977): 89-93.

Mathews, M. V.; Lochbaum, Carol; and Moss, Judith. "Three Fonts of Computer-Drawn Letters." **The Journal of Typographic Research** 1 (October 1967): 345-356.

Mergler, H. W. and Vargo, P. M. "One Approach to Computer Assisted Letter Design." **The Journal of Typographic Research** 2 (October 1968): 299-322.

Negroponte, Nicholas. "Soft Fonts." **SID 80 Digest**: 184-185.

Ruder, Emil. **Typography**. (New York: Hastings House, 1982).

Schmandt, Christopher. "Soft Typography." **Information Processing 80: Proceedings of the IFIP Congress 1980**: 1027-1031.

Shurtleff, Donald. "Relative Legibility of Leroy and Lincoln/Mitre Fonts on Television." **The Journal of Typographic Research** 3 (January 1969): 79-90.

Snyder, Harry. "The Sensitivity of Response Measures of Alphanumeric Legibility to Variations in Dot Matrix Display Parameters." **Human Factors** 21 (1979): 457-471.

Snyder, Harry and Maddox, Michael. "On the Image Quality of Dot-Matrix Displays." **Proceedings of the SID 21** (1980): 3-7.

Snyder, Harry and Maddox, Michael. "Legibility Optimization of Computer-Generated Dot Matrix Alphanumerics." **Proceedings of the International Conference on Cybernetics and Society, Part Two** (November 1978): 965-968.

Spencer, Herbert. **The Visible Word**. (New York: Hastings House, 1969).

Spencer, Herbert. **Pioneers of Modern Typography**. (London: Lund Humphries Publishers Limited, 1969).

Tannas, Lawrence. "Fail-Safe Matrix-Fonts." **SID 77 Digest**: 54-55.

Updike, Daniel Berkeley. **Printing Types: Their History, Forms and Use**, 2 volumes. (Cambridge: Harvard University Press, 1922).

Warnock, John. "The Display of Characters Using Gray Level Sample Arrays." **Siggraph 1980 Conference Proceedings** 14: 302-307.

Winkler, Robert and Konz, Stephan. "Readability of Electronic Displays." **Proceedings of the SID 21** (1980): 309-313.

Zapf, Hermann. "The Changes in Letterforms Due to Technical Developments." **The Journal of Typographic Research** 2 (October 1968): 351-368.

Zimmermann, Roland and Fleischhaker, Heinrich. "Balancing Type Widths and the Different Effects of Metal, Photo, CRT and Laser Setting Systems." **Typografische Monatsblätter** 5 (1982): 2-18.

