ABSTRACT: Each character to be displayed by a cathode-ray tube is assigned unique control and mask words. A portion of the assigned mask word designates one of a plurality of predetermined masks best suited for generation of a character; each mask defining a unique plurality of character stroke endpoints. The control word determines which of the various character strokes defined by a mask are to be executed by the CRT beam, while the remaining portion of the mask word determines which of the executed character strokes are to be visible.
Fig. 2.

SERVICE CONTROL:
CR moves CPR to margin and down one line
H tab moves CPR to tab stop
V tab erases screen and sets CPR to top left margin

INITIALIZE:
1. Using ASCII code, retrieve mask and control WDS.
2. Enter control WD in REG A
3. Enter mask WD in REG B
4. Using mask WD, enter address -1 of 1st mask endpoint into marker
5. Rotate REG B to insert \( \theta \) in 1st bit for blanking char, advance
6. Index CPR to next char position

TEST FOR EXECUTION:
1. Increment marker
2. Load control WD in accumulator, rotate 1st bit into link and \( \theta \) into last bit

LOAD AND TEST FOR BLANKING:
1. Load mask endpoint coordinates in accumulator using marker
2. Add endpoint coordinates to CPR, to form stroke coordinates for display
3. Load mask WD in accumulator and rotate 1st bit into link

EXECUTE VISIBLE STROKE
EXECUTE INVISIBLE STROKE

INVENTOR.
Murray H. Ruben

BY
Blair, Cesari & Stonge
ATTORNEYS
**Fig. 3**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
<th>M</th>
<th>N</th>
<th>O</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>R</td>
<td>S</td>
<td>T</td>
<td>U</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

7890+-! ? : ; = [ ] <> $ 
←↑→'. #, % & @ * / \ > < " " |

**SPACE**

**Fig. 4a**

| 9 | 7 | 10 |
| 8 | 6 |
| 1,11 | 32 | 4 |
| 2,12 | 3 | 5 |

**MASK φ**

**Fig. 4b**

| 1,3 | 7 | 6,12 |
| 4 |
| 5,8 |

**Fig. 4c**

| 3,7 | 12 | 4,8 |
| 6 |
| 5 |

**Fig. 4d.**

| 7 | 6 |
| 1,12 | 2,11 | 3 | 4,10 |
| 8 | 5,9 |

**MASK 3**

**Fig. 4e.**

| 1,9 | 11 | 2,10 |
| 2,11 | 9 | 10 |

**MASK 1**

**Fig. 4f.**

| 10 | 9 | 7 |
| 6 |
| 2 |
| 4,5,6 |

**MASK 5**

**Fig. 4g.**

| 1,2 | 12 |
| 3,4 | 5 |
| 6 |
| 7,8 |
| 9,10,11 |

**MASK 6**

**INVENTOR.**

Murray H. Ruben

**BY**

Blair. Cesari & St. Onge

**ATTORNEYS**
1

VARIABLE STROKE CHARACTER GENERATOR

BACKGROUND OF THE INVENTION

It is becoming commonplace to employ cathode-ray tube displays as an input-output device for an electronic computer. In many situations, such displays may take the place of printers if a permanent record of the computer output is not desired.

Numerous techniques have been employed for generating alphanumeric characters for display on the screen of a cathode-ray tube. Probably the most commonly used technique of character generation employs a fixed, universal stroke pattern from which all of the alphanumeric characters to be displayed are generated. In operation, the beam of the cathode-ray tube is controlled so as to execute each stroke of the pattern in sequence. The beam is blanked except during the times it executes those strokes making up the particular character to be displayed. In order to provide sufficient character legibility, the stroke pattern must contain a large number of different strokes, some patterns including as many as 20 strokes. It will be appreciated that the greater the number of different strokes in the pattern, the more legible the characters can be made.

An analogous technique employs a dot matrix. The beam of the cathode-ray tube is positioned to each of the dots in the matrix in sequence and is blanked to display particular ones of the dots necessary to form a particular character. Again, the greater the number of dots in the matrix, the more legible each of the characters can be made.

In each of these two techniques, it is seen that there is a high degree of redundancy. This is particularly so for those characters having simple configurations. That is, some characters require only several visible strokes or dots. Yet, all of the other strokes of the stroke pattern (or dots of the dot matrix) are in effect, generated or executed. It is therefore apparent that there is considerable time waste involved, even when displaying complex characters.

A further disadvantage of the above-noted character generation techniques involves the fact that it is extremely difficult to devise a universal stroke pattern from which all of the characters can be generated and displayed in legible fashion. Typically, a compromise is struck between legibility and system complexity. Also at stake in this compromise is character execution time. If the number of strokes in the stroke pattern is increased, legibility, character execution time, and system complexity all increase. Typically, this compromise has been struck in favor of simplicity and decreased character execution time, and as a consequence, the resulting character fonts leave much to be desired from the standpoint of legibility and esthetics.

SUMMARY OF THE INVENTION

By virtue of the instant invention, the drawbacks of known alphanumeric generation techniques are substantially overcome. Legibility of the generated characters is materially enhanced without adding proportionately to system complexity. Moreover, character execution time, rather than being increased, is drastically reduced.

Rather than employing a single universal stroke pattern, I employ a plurality of different stroke patterns, hereinafter referred to as "masks." In the disclosed embodiment of my invention, seven masks are used as basis for generating at least 64 different alphanumeric characters. It will be appreciated that a greater or lesser number of masks may be used in the practice of my invention.

Each mask establishes a unique series of stroke endpoints so formulated as to handle a plurality, but not all of the alphanumeric characters to be generated. Each mask is formulated to handle a different group of characters having similar configurations, i.e., requiring some mutually common strokes, although this is not always the case. Although there is some compromise of legibility in making a particular mask compatible to a plurality of characters, legibility is significantly enhanced as compared to those techniques using a single universal stroke pattern for all of the characters to be displayed.

As a further significant feature of my invention, stroke redundancy in composing the characters to be displayed is reduced to an absolute minimum. Only those strokes which are necessary in generating a particular character are actually executed. That is, those strokes of a particular mask which form no part in the generation of a particular character are not executed. Thus, in most instances all executed strokes are actually displayed on the cathode-ray tube screen. For certain characters, certain strokes are not intensified, as when they would retrace a stroke already displayed pursuant to positioning the beam for execution of the next visible stroke.

As a consequence, character execution time is also substantially decreased and is, in fact, held to an absolute minimum.

In accordance with my invention, I provide a dispatch table containing a mask word and a control word associated with each of the alphanumeric characters to be generated. This dispatch table is held in storage. Also held in storage is mask data associated with each of the plural masks provided by my invention. This mask data is in the form of X- and Y-coordinate data of the various stroke endpoints established by each mask.

Upon receipt of a character, typically coded in ASCII format, the mask word and the control word for the character are retrieved from storage and held in a pair of registers. The mask word includes two parts. One part designates which of the plural masks is to be used as basis for generating the particular character. The control word designates which of the strokes of the designated mask are to be executed, while the remaining part of the mask word designates which of the executed strokes are to be visible and which are to be invisible.

On the basis of the stroke data of the designated mask, the beam of the cathode-ray tube is stroked from its position at the endpoint of the last executed stroke of the previously displayed character to the starting point of the first stroke to be executed pursuant to generating the next character. The mask and control words are interpreted such that this first stroke, which is not a character stroke by merely an indexing stroke, is blanked. Under control of the mask data, the beam is then addressed to the endpoint of the next stroke designated by the mask. The control word is then processed to determine whether this next stroke is to be executed. If it is to be executed, the mask word is processed to determine whether this stroke is to be a visible or invisible. If the stroke is not to be executed, the mask data is in effect indexed to address the beam to the next in the sequence of endpoints established by the mask. Again the control word and the mask word are interpreted in determining if the beam is to be stroked to this endpoint and, if so, whether or not the beam is to be intensified.

Each time a stroke is executed, the control word is interpreted to determine whether any additional strokes are to be generated. In other words, the control word is repeatedly monitored to determine when the entire character has been generated and displayed. As soon as this has been accomplished, the next character to be displayed is accepted.

It is to be noted that the mask data is effectively indexed to establish successive stroke endpoints in ordered sequence, not strokes. Consequently, the number of stroke possibilities established by a particular mask are considerable.

The invention accordingly comprises the several steps and the relation of one or more of such steps with respect to each of the others and the apparatus embodying features of construction, combination of elements and arrangement of parts which are adapted to effect such steps all as exemplified in the following detailed disclosure, and the scope of the invention will be indicated in the claims.

For a fuller understanding of the nature and objects of the invention, reference should be had to the following detailed description taken in connection with the accompanying drawings in which:
FIG. 1 is a generalized block diagram of data processing apparatus suitable for use in practicing my invention; FIG. 2 is a flow diagram illustrating the manner of operation of the apparatus of FIG. 1 pursuant to the practice of my invention; FIG. 3 is a pictorial representation of a portion of an alphanumeric character font generated by the disclosed embodiment of the invention; FIGS. 4a through 4g are graphical representations of the various masks employed in the disclosed embodiment of my invention.

DETAILED DESCRIPTION OF THE INVENTION

The essential means for carrying out my invention are conveniently found in typical small-scale, general purpose digital computers, such as the Programmed Data Processor-8 (PDP-8), manufactured by Digital Equipment Corporation of Maynard, Mass. A generalized block diagram of this computer is shown in FIG. 1. It will be appreciated however that by virtue of the teachings of the instant invention, a special purpose computer or data-processing system may be provided to perform the various operations, to be described, pursuant to practicing my invention. However, the block diagram of a general purpose computer is disclosed herein since the essential operating elements and controlling logic for carrying out the invention are found in any computer of this type.

Referring now to FIG. 1, the accumulator 10 is a 12-bit register which operates in conjunction with the control logic circuit 12 in performing various data-processing operations pursuant to the invention. The accumulator 10 also serves as an output register, transferring data to cathode-ray tube (CRT) interface logic circuit 14. The interface logic circuit 14 controls a cathode-ray tube 16 pursuant to displaying characters on the screen thereof. This cathode-ray tube 16 may be a Tektronix Type 611 Storage Tube Monitor, or its equivalent.

The link 18 in FIG. 1 is a one-bit register operating in conjunction with the accumulator 10. Under program control by the computer control logic 12, the link 18 is cleared and rotated as part of the accumulator 10. The core memory 20, in the case of the PDP-8, is a random access magnetic core memory. As will be seen, the core memory 20 holds in storage all of the data necessary for the performance of the present invention.

The memory buffer register 22 handles data transfers between the core memory 20 and the other registers shown in FIG. 1. The core memory 20 is selectively addressed for reading and writing by the memory address register 24. Addresses are entered into the memory address register 24 either from the memory buffer register 22 or the control logic circuit 12.

Pursuant to the invention, I provide a plurality of masks, which in the illustrated embodiment of the invention are seven in number as shown in FIGS. 4a through 4g. These masks define a plurality of stroke endpoints in a character position as indicated by the variously numbered dots in each of FIGS. 4a through 4g. Each mask can have a maximum of 12 stroke endpoints in the illustrated embodiment, however, a greater or lesser maximum number may be adopted without departing from the invention. These stroke endpoints are numbered from 1 through 12. However certain stroke endpoints are given several numbers. In the case of mask 0 (FIG. 4e) the stroke endpoint designated by the reference numeral 30 is assigned two numbers, 1 and 11. Stroke endpoint 32 has two number designations, 2 and 12. The remaining stroke endpoints in mask 0 have only one number designation. As will be seen, the purpose in giving certain stroke endpoints plural number designations is to handle the situation where the same endpoint is used twice in generating certain characters, as when a subsequent stroke ends at the starting point of a previous stroke or retraces a previous stroke.

Each mask 0 through 6 (FIGS. 4a through 4g) may be considered as being 6 units high and 4 units wide. The maximum of 12 stroke endpoints established by each mask are so positioned to accommodate all of those characters to which a particular mask is assigned. The endpoints of each mask are resolved into X- and Y-coordinates which are digitally encoded and stored in the memory 20 (FIG. 1). This X- and Y-coordinate data of each mask is preferably referenced to the lower left-hand corner of the character space in which a character is to be displayed on the CRT screen. The X- and Y-coordinate data of each mask is stored in consecutive address locations in the memory according to the numbering sequence given the mask endpoints. Thus, the memory 20 can be sequenced through consecutive addresses to access the stroke endpoint coordinate data in the same sequence in which they are numbered in the mask. Since the X- and Y-coordinate data of the stroke endpoints is accessed in sequence, the system cannot return to a particular endpoint after it has been once accessed unless its coordinate data is repeated in a subsequently accessed memory address location. This is precisely what is done in the case of stroke endpoints 30 and 32 in FIG. 4a. Thus, the first and 11th memory address locations of mask 0, designating the single endpoint 30, have the same content. This is also true of the second and 12th memory address locations of mask 0, stroke endpoint 32. This greatly reduces the logic circuitry required in accessing the memory.

In addition to mask coordinate data, the core memory 20 also stores a dispatch table format which is a control word and a mask word for each character to be generated. Each control word contains as many bits as there are stroke endpoints in a particular mask. Thus in the illustrated embodiment of the invention, the maximum number of stroke endpoints in any mask is 12, and thus the control word contains 12 bits. It will be appreciated that this limitation is dictated by the word length capabilities of the apparatus of FIG. 1 and is not an inherent limitation of the invention.

The mask word on the other hand has two parts. Three bits of each mask word are used to designate the appropriate mask for generating the character. Since seven masks are used in the illustrated embodiment, three bits are necessary to designate which is to be used in any particular instance. The number of remaining bits of each mask word corresponds to the maximum number of strokes to be executed in generating the most complex character in the font. For the particular font shown in FIG. 3, it turns out that the maximum number of strokes required is nine, despite the fact that each mask may have the capacity to define 12 separate character strokes. Consequently, the mask word contains 12 bits, three for designating the appropriate mask and nine to handle a maximum of nine character strokes.

In accordance with the invention, the bits of the control word determine if a character stroke terminating at the various mask endpoints is to be executed pursuant to generating a particular character. Thus, for example, the first bit of the control word is a binary one if a stroke terminating at the first mask endpoint is to be generated. If not, the first bit is a binary zero. Similarly, the second bit of the control word is either a binary one or binary zero depending upon whether a stroke terminating at the second mask endpoint is to be executed, etc.

The first nine bits of the mask word, on the other hand, indicate which of those executed strokes, as designated by the control word, are to be visibly displayed on the face of the cathode-ray tube 16 (FIG. 1). If the first executed character stroke is to be visible, the first bit of the mask word is a binary one, for example. If the first executed stroke is blanked, the first bit of the mask word is a binary zero. The rest of the first nine bits of each mask word are similarly coded in sequence to designate which of the remaining strokes executed in corresponding sequence are to be visible and which are to be blanked.

As will be seen, by virtue of logical manipulation of the mask word, the first stroke to be executed pursuant to the generation of each character is always an invisible stroke. This
is due to the fact that, in accordance with the invention, the first executed stroke is actually an indexing stroke, moving the beam from the endpoint of the last stroke executed in generating the last character to the starting point of the first visible stroke of the next character to be generated.

As an example of the manner in which control of the generation of a character is handled by the control and mask words and the masks of the invention, consider the letter Y seen in the character font of FIG. 3. The mask word is coded as follows:

\[ 110100000000 \]

The last three bits of the assigned mask word, decoded as a decimal one designates mask 1 (FIG. 4b) for use in generating the character Y. The control word assigned to the character Y is coded as follows:

\[ 001011011000 \]

As seen, the first binary one in this control word is contained in the third bit position. This designates that endpoint 3 of mask 1 (FIG. 4b) is the starting point of the first visible stroke to be executed in generating the character Y. Actually, since endpoint 3 is also endpoint 1, the first bit of the above control word could just as well be a binary one and the third bit a binary zero.

As will be seen, endpoint 3 (or endpoint 1) is actually the endpoint of an indexing stroke bringing the CRT beam from the endpoint of the last stroke executed for the previously displayed character to endpoint 3, mask 1, preparatory to generating the character Y.

It is seen that the fourth bit of the control word is a binary zero and thus a stroke is not executed from endpoint 3 to endpoint 4 of mask 1. The fifth bit of the control word is a binary one, and thus a stroke is executed from endpoint 3 to endpoint 5 of mask 1. Since the first bit of the mask word is a binary one, this executed stroke is made visible to form one leg of the character Y.

The sixth bit in the control word is also a binary one and a stroke is executed from endpoint 5 to endpoint 6. The second bit of the mask word is also a binary one, consequently this stroke is also visible to form the second leg of the character Y.

As seen from the control word, the next stroke to be executed is from endpoint 6 to endpoint 8 (also endpoint 5) of mask 1. However, the third bit of the mask word is a binary zero and consequently this third stroke is blanked. As seen in FIG. 4b, the beam, in moving from endpoint 6 to endpoint 8, retracts the previously executed stroke from endpoint 5 to endpoint 6. This retrace is blanked as so as to not to overintensify the second leg of the character Y. It is thus assumed that the two upper legs of the character Y have been generated and the beam has returned to the junction thereof pursuant to generating the third leg downward to endpoint 9, which is visibly executed by virtue of the fact that the ninth bit of the control word is a binary one and the fourth bit of the mask word is a binary one.

The flow diagram of FIG. 2 shows how the apparatus of FIG. 1 is operated to generate alphanumeric characters for display on the screen of the cathode-ray tube 16. The characters to be generated are inserted in the accumulator 10, either from the core memory 20 by way of the memory buffer register 22 and connections 21, 23 or from an outside source (not shown). Each character is coded in any suitable format, such as the ASCII binary format. With the character to be generated held in the accumulator 10, the flow diagram of FIG. 2 is entered at entry point 1. The content of the accumulator is applied to the control logic circuitry 12 over connection 11 where the character is decoded to determine whether it is an alphanumeric character to be generated or a control character. If the character is recognized as a control character, the system of FIG. 1 goes into a Service Control subroutine, as indicated in FIG. 2. The only control characters recognized for purposes of the present invention, are carriage return CR, horizontal tab H-Tab, and vertical tab V-Tab.

Included in predetermined address locations in the core memory 20 are a pair of registers termed character position registers CPR. One of the character position registers CPR contains X-coordinate data used to horizontally position the beam of the cathode-ray tube 16 (FIG. 1). The other character position register CPR contains Y-coordinate data for vertically positioning the beam of the cathode-ray tube.

If the control character is carriage return CR, the contents of the character position registers CPR are read from the memory 20 into the accumulator 10 by way of the memory buffer register 22. The addresses of the character position registers CPR are generated by the control logic circuitry 12 and fed to the memory address register 24 over connection 13. The memory address register, in turn, address the memory 20 over connection 25. The X- and Y-coordinate data of the beam position in the character position registers CPR are augmented by the logic circuitry 12 to, in effect, address the beam of the cathode-ray tube 16 to the left-hand margin of the screen and down one line. The new X- and Y-coordinate data in the character position registers CPR is returned to their assigned locations in the core memory 20 but the beam of the cathode-ray tube 16 is not then stroked to the position specified by the contents of the character position registers CPR. In practice the character position registers are handled separately and in succession. That is, for example, the X-coordinate data in one character position register is accessed, augmented and returned to memory, and then the Y-coordinate data in the other character position register is similarly processed.

For the control character H-Tab, the contents of the character position registers CPR are manipulated in the same fashion to address the beam to a predetermined tab stop to the right of the left-hand margin of the cathode-ray tube screen. As is apparent, only the X-coordinate data is augmented in this case.

In the case of the V-Tab control character, the X- and Y-coordinate data of the character position registers CPR held in the memory 20 is altered so as to address the beam of the cathode-ray tube 16 to the upper left-hand margin of the screen. In addition, the logic circuitry 12 in response to V-Tab control character, signals the cathode-ray tube interface logic 14 over connection 15 to produce erasure of the screen, thereby removing all previously generated and displayed characters.

After the Service Control subroutine has been carried out, the system is conditioned to accept the next character in accumulator 10, corresponding to entry of the flow diagram of FIG. 2 at entry point I. Alternatively, the system receives a stop signal, in which case shutdown is effected.

An alphanumeric character to be generated is decoded by the logic circuitry 12 to derive the addresses for appropriate entry into the dispatch table stored in the memory 20 (Step 1). The control word and the mask word assigned to the character to be generated are transferred in succession from their permanent storage locations in the memory 20 to temporary storage locations, referred to as Register A and Register B. These transfers are carried out by way of the memory buffer register 22, while the logic circuitry derives the addresses for Register A and B. The control and mask words are also reentered in their permanent memory locations to preserve the integrity of the dispatch table.

The mask word is then retrieved from Register B in the memory 20 for entry into the accumulator 10. The last three bits of the mask word, designating one of the seven masks in the illustrated example to be used in generating the character, are decoded in the logic circuitry 12. The address immediately preceding the address containing the first endpoint data of the designated mask is generated by the logic circuitry 12 and entered into the memory buffer register 22 over connection 19. The control logic circuitry then generates the address of a location in memory termed "Marker" which is fed to the memory address register 24. The contents of the memory buffer register 22, address -1 of the first mask endpoint, is
then entered into the Marker location in memory 20 (Step 3). The reason that the address immediately preceding the address of the first endpoint data is entered into the Marker location in the first instance is that, as will be seen, the first executed stroke is an indexing stroke moving the beam from the endpoint of the last stroke of the previously displayed character to the starting point of the first character stroke of the next character to be displayed. Thus, to access the first endpoint data, as well as subsequent endpoint data, the Marker is merely incremented.

Next, in step 4 of the Initialize subroutine, the mask word stored in accumulator 10, is rotated under the control of the logic circuitry 12 to insert binary zero in the first bit position. In effect, the bits of the mask word are shifted one position to the right and a binary zero is inserted in the first bit position. This destroys the mask-designating significance of the last three bits, but this is of no matter since the Marker has already been set. As will be seen, this is done to blank the indexing stroke.

The mask word, as rotated, is returned to the Register B location in memory. The contents of the X- and Y-Character Position Registers CPR are next successively retrieved from the memory 20 and placed in the accumulators 7 and 8. This data is transferred to the logic circuitry 12 to, in effect, add the beam of the cathode-ray tube 16 to the bottom left-hand corner of the next character space on the screen where the next character is to be displayed. This X- and Y-coordinate data, as incremented, is then returned to the assigned Character Position Register CPR locations in the memory (Step 5). Having completed the Initialize subroutine, the Test For Execution subroutine is entered.

The Marker is retrieved from memory, entered into the memory buffer register, and incremented to thus establish the address in the memory 20 of the X- and Y-coordinate data for the first mask endpoint. The Marker, as incremented, is returned to its assigned location in memory (Step 1). The control word in Register A is retrieved and entered into the accumulator. The control word is rotated under the control of the logic circuitry 12 such that the first bit is shifted into the link 18 and a zero is inserted in the last bit position. Thus, the individual bits of the control word are shifted to the left one place, with the leftmost bit being shifted into the link 18. The 12th bit position of the control word is always made a binary zero each time this shift is performed (Step 2). The logic circuitry 12 then looks at the content of link 18 manifested over connection 17. If the link is set, i.e., a binary one, the subroutine Load Stroke And Test For Blanking is entered. If the link is not set, the logic circuitry 12 tests the Register A content in the accumulator to determine if any of the bits are binary ones. It will be seen that if the control word (Register A) does not contain any binary ones, all of the character strokes to be executed have been executed, and the character has been fully generated and displayed. The system is then conditioned to accept the next character or stop.

In the example of character Y considered above, it is seen that at this point in the flow diagram, the link would not be set but Register A still contains binary ones. Consequently, the system exits at point 2 which is common to entry point 2 in advance of the Test For Execution subroutine. The Test For Execution subroutine is thus performed again. With Register A still in the accumulator, the Marker is retrieved, incremented in the memory buffer register 24 and returned to memory. Register A is rotated again, shifting a binary zero into the last bit position and the second bit of the control word into the link. For the character Y, the second bit of the control word is again a binary zero. The link is not set, and Register A still contains binary ones. Consequently, the Test For Execution subroutine is repeated.

Register A has a binary one in its first bit position, and upon rotation, this binary one is shifted into the link. The Marker, having been again incremented, contains the address of the X- and Y-coordinate data of endpoint 3 of mask 1 (FIG. 4b). Since the link is now set, the Load And Test For Blanking subroutine is now performed. The thrice-shifted control word (Register A) is returned to its assigned location in the memory 20. The binary one in the link 18 is not returned to memory and is erased. The Marker, addressing the storage location of the X- and Y-coordinate data for endpoint 3 of mask 1 is retrieved from the memory and loaded into the memory address register 24 by way of the memory buffer register and connection 27. The Marker is also rewritten in its assigned memory location for later use. The thus-accessed X- and Y-coordinate data for endpoint 3 is added into the accumulator (Step 1). The X-coordinate data in one Character Position Register CPR is retrieved from memory and under control of the logic circuitry 12, added to the X-coordinate data of mask endpoint 3 held in the accumulator (Step 2). The augmented X-coordinate data is then transferred over connection 29 to the CRT interface logic circuitry 14. The same operation is performed with the Y-coordinate data of the other Character Position Register and the mask endpoint 3. Since the Character Position Registers address the beam to the lower left corner of the character space, addition thereto of the data for mask endpoint 3 (FIG. 4b) addresses the retrieve of both the Character Position Register data and the mask endpoint data, a rewrite function is performed to immediately restore this data in unaltered form in their assigned memory locations in addition to supplying it to the accumulator.

Register B, containing the mask word, is retrieved from the memory 20 and entered into the accumulator 10. The accumulator content is shifted to the right one-bit position, entering the first bit into the link 18 (Step 3). The logic circuitry 12 then tests to determine if the link is set. If the link contains a binary zero (not set), the logic circuitry signals the cathode-ray tube interface logic circuitry 14 over connection 15 such that the cathode-ray tube beam executes an invisible stroke. If the link is set, the cathode-ray tube interface logic circuitry 14 is signaled to cause the cathode-ray beam to execute a visible stroke.

The cathode-ray tube interface logic circuitry 14 is of conventional design, containing a digital-to-analog converter for converting to corresponding analog voltages the summed X- and Y-coordinate data of the Character Position Registers CPR and the appropriate mask endpoint as received from the accumulator. These voltages are ultimately integrated to produce suitable voltage ramp functions which are concomitantly supplied to the X- and Y-deflection plates of the cathode-ray tube 16. The beam of the cathode-ray tube is thus deflected to the position address by the combined Character Position Register and mask endpoint data.

It will be recalled that according to Initialize step 4 the mask word (Register B) was rotated to insert a binary zero into its first bit position. As a consequence, for the first stroke executed in generating a character the link 18 always contains a binary zero. Consequently, this first stroke is blanked, inasmuch as it is an indexing stroke moving the beam from the endpoint of the last character stroke in the previously displayed character to the starting point for the first visible stroke of the next character to be generated. In the specific example of the character Y, this indexing stroke brings the beam of cathode-ray tube 16 to mask endpoint 3 as seen in FIG. 4b. Whether the executed stroke is visible or invisible, the flow diagram is reentered at entry point 2 in advance of the Test For Execution subroutine. Register B is returned to its assigned memory location.

Repeating this subroutine, the Marker is retrieved from memory and incremented while in the memory buffer register to develop the memory address for mask endpoint 4. Register A is retrieved and entered into the accumulator. The fourth bit of the control word, a binary zero is shifted into the link, and a binary zero is inserted in the last bit position. Since the link is not set, the Test For Execution Subroutine is repeated.

This time, a binary one is shifted into the link and mask endpoint 5 is accessed. The link is set and the Load Stroke And
Test For Blanking subroutine is initiated. The coordinate data for mask endpoint 5 is added to the X- and Y-coordinate data of the Character Position Registers and fed to the cathode-ray interface logic 14. Register B is then entered into the accumulator and rotated to shift the first bit of the mask word, a binary one, into the link. As a consequence, an intensified stroke is executed from mask endpoint 3 to mask endpoint position 5 (FIG. 4b).

Reentering the flow diagram at entry point 2, the next stroke to be executed is from mask endpoint 5 to mask endpoint 6 since the sixth bit of the control word is a binary one. This stroke is intensified since the second bit of the mask word is a binary one.

It will be noted that the next stroke to be executed under the control of the control word is from mask endpoint 6 to mask endpoint 8. However, this stroke retraces the previous stroke from endpoint 5 to endpoint 6. Consequently, this retraced stroke must be blanked and is so blanked by virtue of the binary zero in the third bit position of the mask word for the character Y.

Finally, the visible stroke from mask endpoint 8 to mask endpoint 9 is executed to complete the generation and display of the character Y. The system nevertheless reenters the flow diagram at entry point 2 and proceeds through the Test For Execution subroutine once again. Now Register A contains no more binary ones. Consequently, the link is not set. When Register A is tested for binary ones by the logic circuitry 12 and found to have none, the apparatus is automatically conditioned to accept the next character or stop.

It is thus seen that the Test For Execution subroutine is not needlessly performed for the remaining binary zeros of the control word after the last binary one has been processed. Considerable time is thus saved. Moreover, the Load Stroke And Test For Blank subroutine is only performed when a stroke is to be executed, thus saving even more time.

The following is a chart for the characters of the font of FIG. 3, indicating for each mask, the sequence of stroke endpoints (those in parentheses being endpoints of blanked strokes), the number of strokes, and the mask and control words (in octal code); all pursuant to the invention:

<table>
<thead>
<tr>
<th>Character</th>
<th>Mask No.</th>
<th>Sequence</th>
<th>No. of Control Strokes</th>
<th>Word</th>
<th>Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>126.7.10.11.12</td>
<td>8</td>
<td>7745</td>
<td>6347</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>1.4.5.6.7.8.9</td>
<td>6</td>
<td>6001</td>
<td>1376</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
<td>6.7.8.9.10.12</td>
<td>7</td>
<td>7703</td>
<td>2175</td>
</tr>
<tr>
<td>D</td>
<td>9</td>
<td>2.3.4.5.6.7.8.9</td>
<td>5</td>
<td>5706</td>
<td>3553</td>
</tr>
<tr>
<td>E</td>
<td>12</td>
<td>1.3.6.7.8.9</td>
<td>4</td>
<td>4803</td>
<td>5170</td>
</tr>
<tr>
<td>F</td>
<td>15</td>
<td>1.3.6.7.8.9</td>
<td>4</td>
<td>4803</td>
<td>5160</td>
</tr>
<tr>
<td>G</td>
<td>18</td>
<td>6.7.8.9.10.11</td>
<td>3</td>
<td>6703</td>
<td>4176</td>
</tr>
<tr>
<td>H</td>
<td>21</td>
<td>1.4.5.6.7.8.9</td>
<td>3</td>
<td>6703</td>
<td>4760</td>
</tr>
<tr>
<td>I</td>
<td>24</td>
<td>1.3.6.7.8.9</td>
<td>3</td>
<td>6703</td>
<td>4156</td>
</tr>
<tr>
<td>J</td>
<td>27</td>
<td>1.3.6.7.8.9</td>
<td>3</td>
<td>6703</td>
<td>4544</td>
</tr>
<tr>
<td>K</td>
<td>30</td>
<td>1.2.6.7.8.9.10.11</td>
<td>2</td>
<td>6593</td>
<td>6172</td>
</tr>
<tr>
<td>L</td>
<td>33</td>
<td>1.3.6.7.8.9</td>
<td>3</td>
<td>6703</td>
<td>4070</td>
</tr>
<tr>
<td>M</td>
<td>36</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3065</td>
</tr>
<tr>
<td>N</td>
<td>39</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3670</td>
</tr>
<tr>
<td>O</td>
<td>42</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3670</td>
</tr>
<tr>
<td>P</td>
<td>45</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>Q</td>
<td>48</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>R</td>
<td>51</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>S</td>
<td>54</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>T</td>
<td>57</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>U</td>
<td>60</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>V</td>
<td>63</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>W</td>
<td>66</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>X</td>
<td>69</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>Y</td>
<td>72</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
<tr>
<td>Z</td>
<td>75</td>
<td>1.2.3.6.7.8.9</td>
<td>2</td>
<td>7401</td>
<td>3560</td>
</tr>
</tbody>
</table>

It will thus be seen that the objects of the invention, made apparent from the preceding description, are efficiently attained, and since certain changes may be made in carrying out the above method and in the constructions set forth without departing from the scope of the invention, it is intended that all matter contained in the above description or shown in the accompanying drawings shall be interpreted as illustrative and not in a limiting sense.

Having described my invention, what I claim as new and desire to secure by Letters Patent is:

1. The method of generating a font of characters for display on the screen of a cathode-ray tube, said method comprising the steps of:
   a. Storing:
      1. A plurality of masks, each mask consisting of coordinate address data for positioning the cathode-ray tube beam to a unique plurality of stroke endpoints, b. The endpoints of each mask being allotted a number sequence.
   2. A control word assigned to each character, each control word consisting of a plurality of bits with each bit associated with a different endpoint of a particular mask, and
   3. A mask word assigned to each character, each mask word having:
      a. A first part designating a particular one of said masks, and
      b. A second part consisting of a plurality of bits, each associated with a different one of those bits of the commonly assigned control word having a first logical significance;
   B. Accepting a character to be generated;
   C. Locating the storage location of the control and mask words assigned to a character to be generated;
   D. Locating on the basis of the first part of the assigned mask word the storage location of the first endpoint coordinate address data of the designated mask;
   E. Receiving the assigned control word from storage;
   F. Interpreting a first bit of the assigned control word to determine its logical significance;
   G. When said first bit is determined to be of a first logical significance, retrieving the coordinate address data of the associated mask endpoint and supplying it to cathode-ray tube control circuits,
   H. Retrieving the assigned mask word; and
   I. Interpreting the bit of the assigned mask word associated with said first bit of the assigned control word to determine the intensity of the beam as it generates a stroke of...
the endpoint whose coordinate address data was supplied to the cathode-ray tube control circuitry. 2. The method defined in claim 1, and
A. determining when said first bit of the assigned control word is of a second logical significance, and then
B. interpreting the next bit in sequence of the assigned control word without retrieving the endpoint coordinate address data associated with said first control word bit. 3. The method defined in claim 2, and
A. sensing the remaining bits in sequence from said first bit of the assigned control word to determine the logical significance thereof, and
B. when said remaining bits are all of a second logical significance, immediately accept a next character to be generated.
4. The method defined in claim 3 wherein the coordinate address data of consecutive mask endpoints of each said mask is stored in consecutive storage locations, said method further including,
A. establishing a marker register containing a storage address for accessing endpoint coordinate address data; and
B. incrementing said marker register pursuant to interpreting each bit of the assigned control word.
5. The method defined in claim 1; and
A. executing, under the control of the mask and control words assigned to the characters to be generated as the initial executed stroke of each character to be generated, an indexing stroke moving the beam from the endpoint of the last-executed stroke of the previously generated character to the endpoint corresponding to the starting point for the first stroke of the next character to be generated and displayed.
6. A method of generating a font of characters for display on the screen of a cathode-ray tube, said method comprising the steps of:
A. providing a plurality of masks, each said mask consisting of coordinate address data for positioning the beam of the cathode-ray tube to a unique plurality of stroke endpoints,
B. assigning coded mask identification data to each character of the font,
1. said mask identification data for each character designating a particular one of said masks to be used in generating the associated character;
C. assigning coded stroke control data to each character of the font,
1. said stroke control data for each character designating those endpoints of a particular one of said masks to which the beam is to be moved pursuant to executing the strokes necessary to generate the associated character;
D. assigning coded display control data to each character of the font,
1. said display control data for each character designating which of those executed beam strokes are to be displayed on the screen;
E. selecting from said plurality of masks the one identified by the mask identification data assigned to a particular character to be generated;
F. selecting from the pluralities of stroke control and display control data the control data assigned to the particular character to be generated;
G. testing said selected stroke control data and said selected display control data to determine those stroke endpoints of said selected one mask to which a beam stroke is to be executed and which of said executed beam strokes are to be displayed; and
H. controlling the cathode-ray tube beam in accordance with the determinations obtained from said testing step to display the particular character.
7. The method defined in claim 6, wherein said testing step is performed as a succession of test cycles, one test cycle being normally devoted to each stroke endpoint of said selected one mask in predetermined sequence.
8. The method defined in claim 7, wherein said testing step includes the steps of:
1. interpreting said selected stroke control data at the beginning of each said test cycle to determine if a beam stroke is to be executed to a particular endpoint of said selected one mask,
2. interrupting said test cycle when it is determined that a beam stroke is not to be executed, and
3. immediately initiating a new test cycle.
9. The method defined in claim 8, wherein said testing step further includes the step of:
interpreting said selected display control data during the concluding portion of each uninterrupted test cycle to determine if the beam stroke to be executed to said particular endpoint is to be displayed.
10. The method of claim 9, wherein said testing step further includes the steps of:
1. evaluating said selected stroke control data upon each said test cycle interruption to determine if a beam stroke was executed at the conclusion of the last uninterrupted test cycle completed the display of said particular character, and
2. if the character display is completed, immediately proceeding to the generation of the next character to be displayed.
11. The method defined in claim 6, wherein said stroke control data is in the form of a unique digitally encoded control word, and each said mask identification data and each said display control data are combined as first and second parts, respectively, of a unique digitally encoded mask word, one of said control words and one of said mask words being assigned to each character of the font, and wherein the method further includes the step of:
A. storing all of said mask and control words and all of said coordinate address data of each said mask in a memory;
B. said testing step includes retrieving from the memory the selected mask and control words and the coordinate address data of those endpoints of said selected one mask to which a beam stroke is to be executed.
12. The method defined in claim 11, wherein said testing step is performed as a succession of test cycles, one test cycle being normally devoted to each stroke endpoint of said selected one mask in predetermined sequence, and wherein said selected control word is retrieved from the memory at the beginning of each test cycle, and said mask word and the coordinate address data of a particular endpoint are retrieved from the memory during the concluding portion of a test cycle only if a beam stroke is to be executed thereto.
13. The method defined in claim 11, wherein the stroke endpoints of each said mask are numbered in sequence and the coordinate address data of consecutively numbered stroke endpoints are stored in consecutive address locations in the memory, and wherein said testing step is performed as a succession of test cycles, one test cycle being normally devoted to each stroke endpoint of said selected one mask in numerical sequence, said method further includes the steps of:
A. establishing a marker on the basis of said selected one mask,
1. said marker addressing the location in the memory of one stroke endpoint;
B. incrementing said marker at the beginning of each said test cycle to access the memory for the coordinate address data of the next stroke endpoint in sequence.
14. The method defined in claim 11 which further includes the step of:
A. providing a character position register in the memory containing coordinate address data establishing a reference position for the beam in each character space on the screen where a character is to be displayed;
B. said testing step includes adding the mask endpoint coordinate address data to said reference position coordinate address data.
15. The method defined in claim 6, and
A. executing, pursuant to the generation of each character and under the control of said selected one mask and said selected stroke and display control data, an invisible first beam stroke moving the beam from the endpoint of the last executed stroke for the previously displayed character to the mask endpoint corresponding to the starting point for the first visible beam stroke of the next character to be generated and displayed.

16. The method defined in claim 7, wherein said selected stroke control data includes a plurality of bits, each bit associated with a different endpoint of said selected mask, said bits being coded a first logical significance when a stroke is to be executed to their associated endpoints, said testing step including

1. interpreting the bits of said selected stroke control data in sequence, one bit being interpreted during each of a succession of test cycles, and

2. feeding the coordinate address data of the mask endpoint associated with the interpreted stroke control data bit to the cathode-ray tube pursuant to executing a beam stroke thereto if said interpreted bit is of said first logical significance.

17. The method defined in claim 16, wherein said selected display control data includes a plurality of bits, each said display control data bit associated with a different bit of said selected stroke control data coded in said first logical significance, said testing step further including

1. interpreting said bits of said selected display control data in sequence, and

2. initiating the display of an executed beam stroke when an interpreted bit of said selected display control data is of a first logical significance.