United States Patent [19]

Mathews

[11] **3,712,443**

[45] **Jan. 23, 1973**

[54] APPARATUS AND METHOD FOR SPACING OR KERNING TYPESET CHARACTERS

[75] Inventor: **Max Vernon Mathews**, New Providence, N.J.

[73] Assignee: **Bell Telephone Laboratories Incorporated**, Murray Hill, Berkeley Heights, N.J.

[22] Filed: **Aug. 19, 1970**

[21] Appl. No.: **65,056**

[52] **U.S. Cl.** .....**197/84 A**, 197/19, 95/4.5 R, 340/324 A

[51] **Int. Cl.** **B41j 19/58**

[58] **Field of Search** .....197/19, 20, 84, 84 A, 84 B; 199/18; 95/4.5; 340/324, 324 A, 325

[56] **References Cited**

UNITED STATES PATENTS

3,267,454	8/1966	Schaaf	95/4.5 R X
2,672,794	3/1954	Freund	95/4.5 R
2,890,634	6/1959	Rondthaler et al.	95/4.5 R
3,165,045	1/1965	Troll	340/324 A UX
3,188,929	6/1965	Higonnet et al.	95/4.5 R
3,311,908	3/1967	Halsted et al.	340/324 A
3,422,419	1/1969	Mathews et al.	340/324 A
3,471,848	10/1969	Manber	340/324 A
3,513,968	5/1970	Hanson	197/84 R X

FOREIGN PATENTS OR APPLICATIONS

665,889	1/1952	Great Britain	95/4.5 R

OTHER PUBLICATIONS

IBM Technical Disclosure Bulletin, Selective Adjacent Character Space Control, J. O. Schaefer, Vol. 11, No. 11, April 1969, pp. 1480, 1481

Primary Examiner—Ernest T. Wright, Jr.
Attorney—R. J. Guenther and William L. Keefauver

[57] **ABSTRACT**

Method and apparatus for producing improved print quality of alphanumeric character combinations. Each character is preassigned closest spacing points at the top, middle, and bottom. Successive characters are positioned so that one left-most point of a new character coincides with one right-most point of the previous character, thereby insuring close intercharacter spacing. Distance measurements from the character body to each spacing point are retained in memory. The left measurements at top, middle and bottom of a new character to be positioned in the print line are subtracted from the right measurements of the top, middle and bottom, respectively, of the previous character in the print line using subtractor circuitry. Maximum value circuits are used to select the largest of the differences thus formed. The print line is displayed on an oscilloscope. The invention may also be practiced using a general purpose digital computer suitably programmed.

4 Claims, 8 Drawing Figures

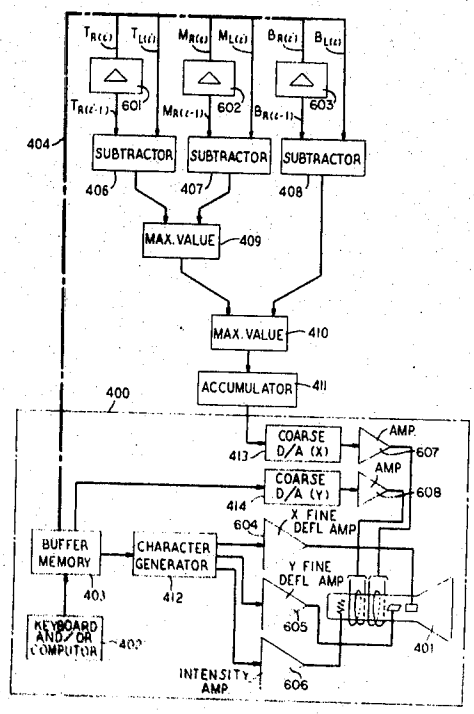


FIG. 1A PRIOR ART

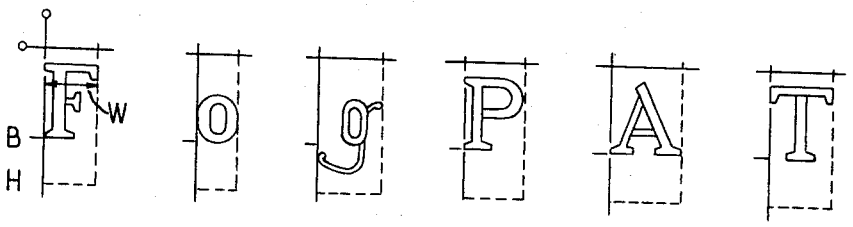


FIG. 1B PRIOR ART

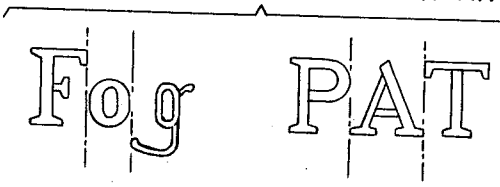


FIG. 2

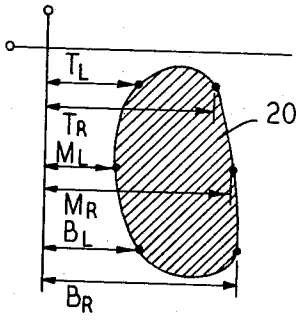


FIG. 3A

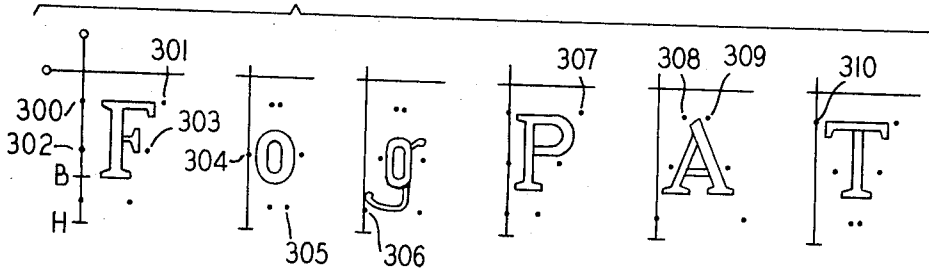
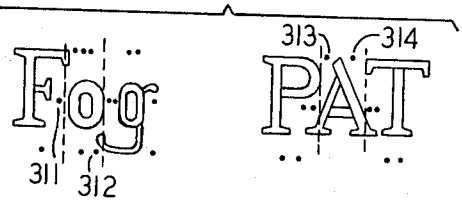
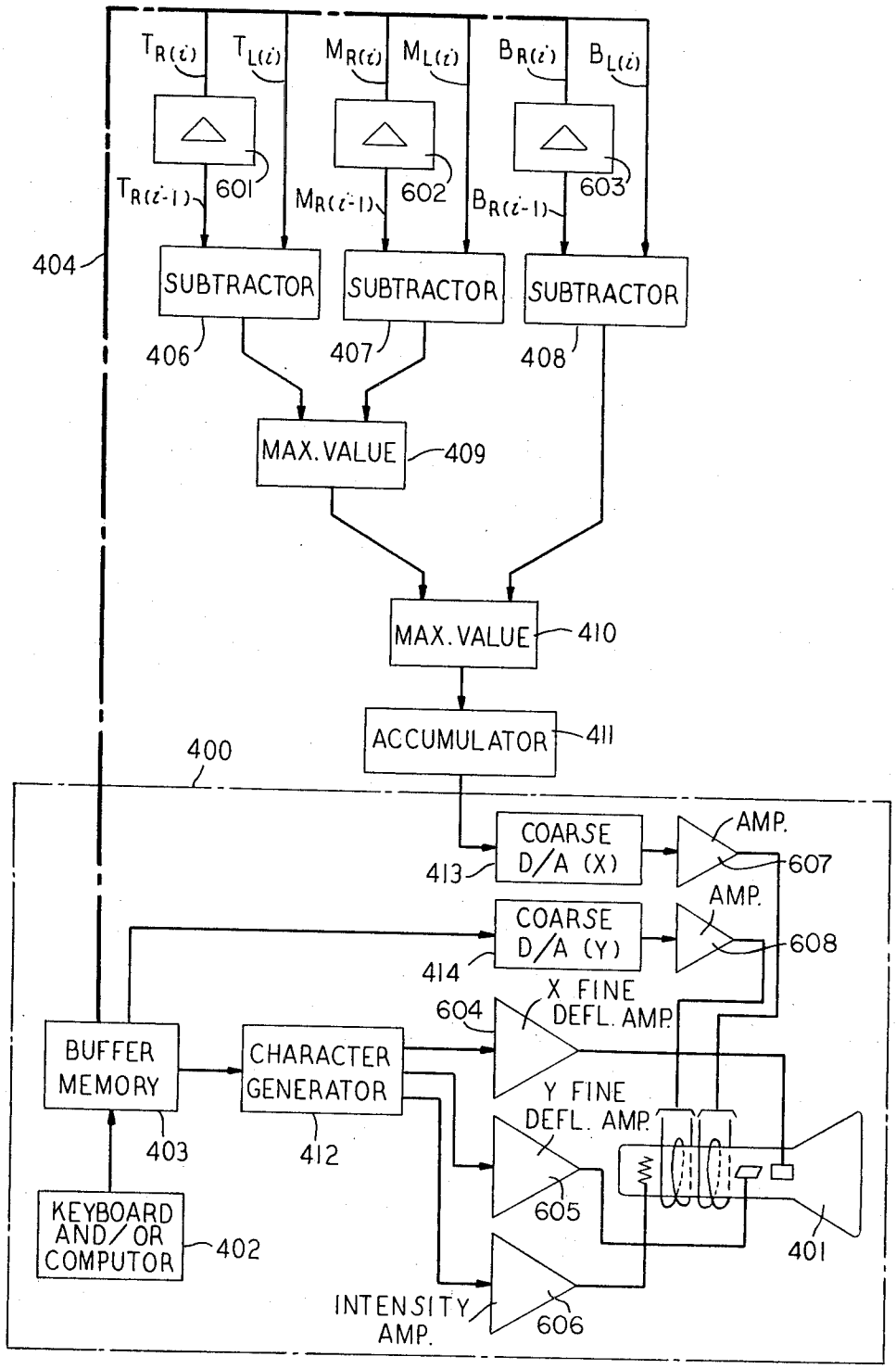


FIG. 3B



INVENTOR
M. V. MATHEWS
BY R. O. Nissen
ATTORNEY

FIG. 4


$T_{R(i)}$
$T_{L(i)}$
$M_{R(i)}$
$M_{L(i)}$
$B_{R(i)}$
$B_{L(i)}$
601
602
603
$T_{R(i-1)}$
$M_{R(i-1)}$
$B_{R(i-1)}$
404
SUBTRACTOR
SUBTRACTOR
SUBTRACTOR
406
407
408
MAX. VALUE
409
MAX. VALUE
410
ACCUMULATOR
411
400
COARSE D/A (X)
413
AMP.
607
COARSE D/A (Y)
414
AMP.
608
X FINE DEFL. AMP.
604
Y FINE DEFL. AMP.
605
BUFFER MEMORY
403
CHARACTER GENERATOR
412
KEYBOARD AND/OR COMPUTOR
402
INTENSITY AMP.
606
401

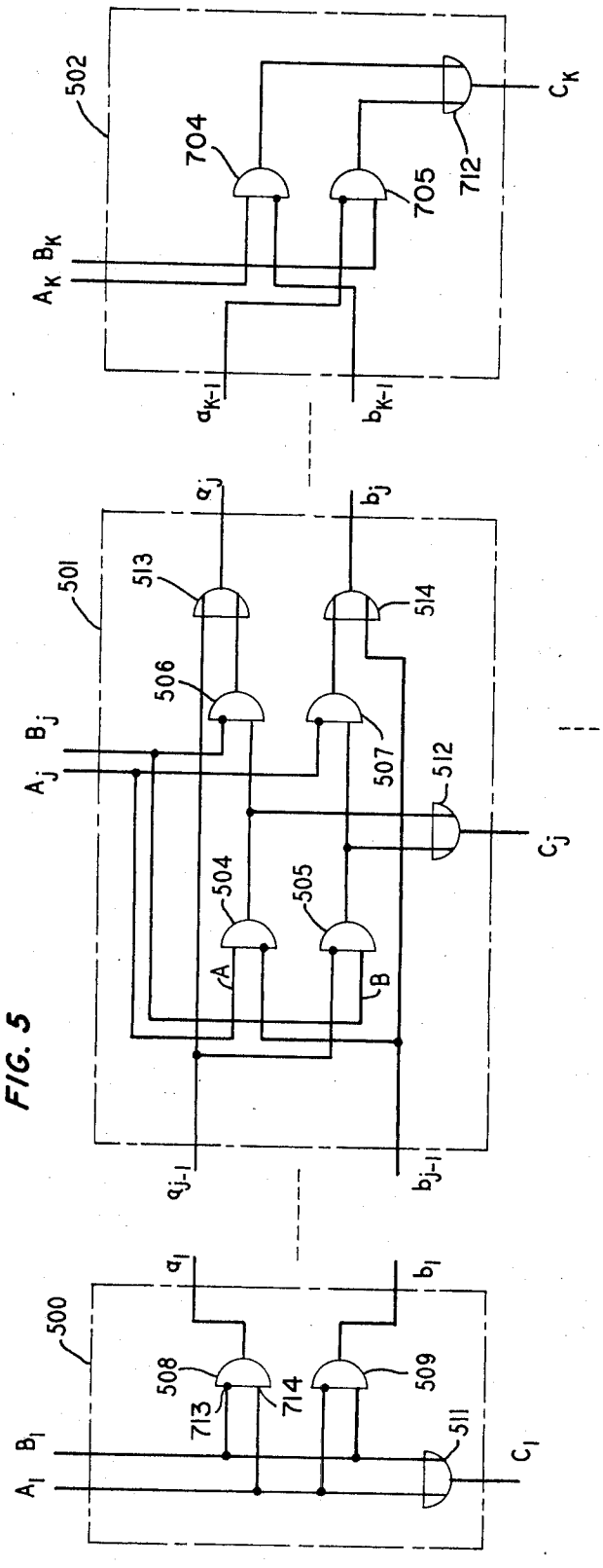


FIG. 5

```
100  W1 = TR(I-1) - TL(I)
101  W2 = MR(I-1) - ML(I)
102  W3 = BR(I-1) - BL(I)
103  W = W1
104  IF(W2.GT.W) W = W2
105  IF(W3.GT.W) W = W3
106  K = K+W
107  CALL PLACE (K,I)
```

FIG. 6

APPARATUS AND METHOD FOR SPACING OR KERNING TYPESET CHARACTERS

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to typesetting and particularly to typesetting methods which use automatic means for character spacing.

2. Description of the Prior Art

The history of printing has been characterized by a quest for mechanized methods for producing high quality print. Letters individually drawn by hand may be freely placed and styled for maximum readability and quality. Drawn letters may include portions which overhang or even intersect their neighbors when the artist feels it adds to the quality of the script. In prior art mechanical printing, the freedom of the manuscript artist is lost. Each type face is cast on a rectangular metal matrix of a specified line width, and all portions of the body of the letter appear within the boundaries of the type matrix.

Since the methods of this invention have broad application to prior art systems, the term "typesetting" is used herein and in the appended claims in the broad sense. It covers the operation of any character compositor including the placement of characters by typewriter, computer-driven composition machines, certain line printers, and any other system where characters are placed sequentially in a print line whether for display (for example screen viewing), or production (for example typewritten copy), or for reproduction (for example printing plates).

The typewriter is properly included as a typesetting device under the above broad description of typesetting. It also qualifies under the more general narrow view of typesetting, i.e., only reproduction, since typewritten copy is frequently directly photographed to produce offset printing plates. Generally, typewritten print is of lesser quality than type matrix print.

A normal typewriter—that is, a non-proportional spacing typewriter—produces characters which occupy a uniform, unvarying line width. Thus the narrowest letter "i" is given the same space to occupy, the same line width, as the widest letter "M".

Proportional spacing typewriters improve this situation by providing variable line widths for the letters. The letter "i" might occupy three units of space, while "M" might occupy six units. Such print is still of lesser quality than type matrix print, since the line width of metal matrix characters can be more closely controlled to give fine gradations of spacing.

Limitations of print quality imposed by the typewriter and type matrix printing are of mechanical origin. The same limitations do not exist, for example, in computer controlled typesetting systems wherein characters are drawn, or painted, or projected on a film or cathode ray oscilloscope face by an electron gun or light beam. However, prior art systems do not take full advantage of their freedom of such limitations.

An exemplary system is described in U. S. Pat. No. 3,471,848 issued Oct. 7, 1969, to S. Manber wherein each character is formed on the face of a cathode ray oscilloscope by a large number of vertical scans. In the Manber system, character widths are divided into small increments allowing relatively precise widths to be assigned to each character. None the less, the line width of a given character in a given font is fixed. The space alotted to a character is as definite as though the type were cast of metal. Print quality would be improved if means were provided for allowing adjacent character identity to effect spacing. This would lead to the overhang of adjacent type bodies and reduced intercharacter spacing for many letter pairs.

SUMMARY OF THE INVENTION

The present invention includes apparatus and methods for spacing typeset characters within words for printing, projecting, or other presentation in order to reduce the space between characters. Each character is assigned, not as a single spacing measurement, as in prior art systems, but a multiplicity. Each time the character appears an appropriate width measurement is calculated for the current instance of the character, thus allowing words to be formed with reduced intercharacter space. In this system, it should be clear it is not the body of the character which is variable but the spacing between the body of the character and its neighbor on either side.

Because this invention entails spacing measurements taken at diverse vertical levels, some type fonts will produce character combinations with overhanging portions of adjacent type bodies. This result should not be confused with fonts which are normally overhanging, e.g., italics. Such normally overhanging type styles may also benefit from the closer spacing provided for by this invention.

Without importing limitations to the invention herein fully described and claimed, assume, for example only, the arrangement wherein six spacing measurements are provided for each character—two at the top, middle, and bottom of the character body, to the left and right sides. Assume further that characters are generated from left to right. As each new character is appended to the right of the character previously placed in the word being generated, the three right spacing measurements of the previous character are mathematically combined with the three left spacing measurements of the new character. From these calculations, a width measurement is selected which displaces the new character from the previous character. Certain character pairs thereby appear with reduced intercharacter spacing, thus producing a print line with improved appearance and print quality. Other characteristics and advantages of the invention will become evident from the following description of the drawing.

BRIEF DESCRIPTION OF THE DRAWING

FIGS. 1A and 1B represent prior art character spacing;

FIG. 2 is one character spacing measurement embodiment conforming to the invention;

FIGS. 3A and 3B represent character spacing according to the present invention;

FIG. 4 is a circuit diagram of a system for performing the inventive method;

FIG. 5 is a detailed diagram of a maximum value circuit used in the system of FIG. 4;

FIG. 6 is a segment of a computer program suitable for performing the invention.

DETAILED DESCRIPTION

FIGS. 1A and 1B show an example of character spacing performed by prior art typesetters whether characters are produced by metal type matrix, proportional spacing typewriter, photocomposer or computer driven display. The common characteristic is that each letter occupies a fixed width on the print line.

FIG. 1A illustrates a particular system in which individual letter characters are stored in the memory of a computer system. A pair of coordinate axes is provided as a reference frame for each character. Each character is described in the computer memory in terms of the lines or structures or patches of area which comprise the body of the type face. See, for example, U. S. Pat. No. 3,422,419 "Generation of Graphic Arts IMages" issued to M. V. Mathews and H. S. McDonald, assigned to the present assignee. The line B forms the base on which the letters sit. The line H represents the limit of the lowest descender in a given type font which extends below the base line. Width W represents the line width of the character and may be different for each character in each font.

Letters are concatenated in order to form words as shown in FIG. 1B. For metal matrix type, the slugs of metal are simply placed side by side. In a proportional spacing typewriter, the carriage or printing mechanism is spaced over by an amount W after printing the character. In prior art computer typesetting systems, the same effect is achieved by adding the quantity W to the stored value of the previous character displacement in the print line. This produces the displacement of the new coordinate axis to form the next letter.

FIG. 2 demonstrates how in one embodiment of the invention a multiplicity of spacing measurements are assigned to a single character. The cross-hatched area **20** is a general representation of the outlines of the body of a character of type. Three pairs of measurements are shown in the Figure, though some type fonts may be adequately described with fewer than three pairs and others may be improved by the use of more than three pairs.

Each pair, for example, T_L and T_R, represents spacing measurements of the character taken at the same vertical level on the left and right of the character body—i.e., Top Left (T_L) and Top Right (T_R). This assures that the T_R measurement of a character already placed on the print line will appear at the same vertical level as the T_L measurement of the new character to be placed next in the print line. The same is true for measurements M_L and M_R (Middle), and $B_{L \text{ and } BR}$ (Bottom).

FIGS. 3A and 3B illustrate the use of multiple spacing measurements to improve the quality of print. FIG. 3A shows six measurements applied to a number of letters. The measurements T_L, T_R, and the rest, are shown in FIG. 3A as six points associated with each of the illustrated characters. The location of point **300**, for example, illustrated by a dot on the Figure, represents the measurement T_L for the letter "F". Since dot **300** happens to appear on the coordinate axis, in this example, $T_L = O$. The location of point **301** represents the measurement T_R. The location of points **302** and **303**, respectively, represents the measurement M_L and M_R. B_L and B_R are also represented. It will be understood that the dots on the drawing are physically representated by the stored values of T_L, T_R, etc. These dots are shown to make visible the concatenation points where characters are joined into words; otherwise, they have no physical significance.

The six letters of FIG. 3A are formed into words in FIG. 3B. By utilizing the multiplicity of spacing measurements, the letters in FIG. 3B have been spaced closer together than those in FIG. 1B. The letter "F" and the letter "o" have been displaced by an amount which corresponds to the coincidence of point **303** and **304** in FIG. 3A into the single point **311** in FIG. 3B. Of the three right-hand points of the letter "F" and the three left-hand points of the letter "o", only the middle points coincide. Similarly, the letters "o" and "g" are joined at point **312** in FIG. 3B representing points **305** and **306** in FIG. 3A. Points **307** and **308** are joined in point **313**. Points **309** and **310** are joined in point **314**.

The invention will be more clearly understood from the following illustrative remarks. Consider that a first character, for example "P" as shown in FIG. 3A, has been placed in the print line. The position of this first character is specified by the location of its origin, that is, the horizontal displacement of its origin from the left margin. When a second character is to be placed in the print line, for example "A" as shown in FIG. 3A, its displacement must be greater than that of the first character in order that the second character will appear to its right.

If the value of T_R for the first character is added to the value of the displacement of the first character (that is, if T_R is taken as the width of the first character), the origin for the second character will fall on top of the T_R concatenation point of the first character, for example, point **307** in FIG. 3A. Now, if T_L for the second character is subtracted from this sum, the origin of the second character is moved to the left by an amount which causes the T_L concatenation point of the second character, point **308** in FIG. 3A, to fall on top of the T_R concatenation point for the first character. This appears as point **313** in FIG. 3B.

Addition of the value of T_R of the first character and subtraction of the value of T_L of the second character is equivalent to using the difference $T_{R1} - T_{L2}$ (where subscripts **1** and **2** indicate the first and second characters respectively) as the effective width of the first character. This effective width is added to the displacement of the first character to obtain the new displacement of the second character.

Although the T_{R1} and T_{L2} measurements were used illustratively in the above discussion, it is clear that the same procedure could be followed for M_{R1} and M_{L2} which would cause the coincidence of the middle concatenation points, and also for the B_{R1} and B_{L2} which would cause the coincidence of the bottom concatenation points. It should be clear that as the effective width of the first character increases, the second character moves farther to its right. Therefore, by calculating all three width values and then choosing the greatest of the three as the effective width of the first character, the second character is displaced to the right of the first by the greatest amount of the three possibilities while still retaining the coincidence of a pair of concatenation points. The result achieved is the same as though the second character were moved closer and closer to the

first character until two concatenation points first coincide.

It should be again emphasized that the dots have no physical significance outside of their representation of the spacing measurements which have been shown diagramatically in FIG. 2. The physical displacement of the characters in the print line is determined by mathematical manipulation of these spacing measurements according to the following algorithm.

Three tentative width measurements are produced combining the right-hand spacing measurement at each vertical level of a character already positioned in the print line with the appropriate left-hand spacing measurement of the new character to be placed. Obviously, similar results may be obtained when the characters are appended from right to left. Let the index value i identify the new character to be placed on the print line. The index value $i-1$ then indicates the previous character—that is, the one already positioned in the print line. The three tentative width values W_1, W_2, and W_3 are then formed by the expressions:

$$W_1 = T_{R(i-1)} - T_{L(i)}$$
$$W_2 = M_{R(i-1)} - M_{L(i)} \quad (1)$$
$$W_3 = B_{R(i-1)} - B_{L(i)}$$

The actual displacement of the origin of the new character is then given by the maximum of these three tentative width values. That is, the displacement W is given the expression:

$$W = \text{Max}\,(W_1, W_2, W_3). \quad (2)$$

In this system, a multiplicity of spacing measurements are preassigned to each of the characters. From these spacing measurements, a multiplicity of tentative width measurements are derived. From these tentative width measurements a suitable width measurement is chosen for displacing the new character from the previous one. In prior art systems, the characters are each preassigned a single measurement which is both the width and spacing measurement.

FIG. 4 is an embodiment of one hardware implementation of the above-described algorithm. System **400** is a display system and is found in the prior art. See, for example, U.S. Pat. No. 3,311,908 by C. P. Halsted et al., issued Mar. 28, 1967. In system **400**, cathode ray oscilloscope **401** is used to display sequential characters in the print line. The electron beam in oscilloscope **401** is coarsely positioned by digital-to-analog converters **413** and **414** and amplifiers **607** and **608** which produce the horizontal and vertical deflection signals, respectively, for positioning the origin of each character to be generated. Characters are imputed by unit **402** which may simply be a keyboard or, preferably, a digital computer which contains a character string in storage. As a digital computer, unit **402** may have any of the usual input devices such as a character keyboard, a magnetic tape unit, or a telecommunications channel for supplying the character string.

Characters are conveyed from unit **402** to the buffer memory **403** in which is stored information needed to generate the selected character and also spacing measurements needed to implement the instant invention. The identity of the character to be displayed is conveyed to character generator **412** which controls the details of shaping the selected character. Character generator **412** exerts fine control of the electron beam necessary to form the selected character on the face of oscilloscope **401** with the aid of X deflection amplifier **604**, Y deflection amplifier **605** and intensity amplifier **606**.

For this embodiment, again assume three pairs of width measurements are employed and that the characters are positioned from left to right. This spacing information is conveyed through cable **404** to delay units **601**, **602** and **603** and subtractor circuits **406**, **407**, and **408**. The three right-hand spacing measurements T_R, M_R, and B_R of the requested character are conveyed through separate delay units **601**, **602** and **603** to the minuend inputs of subtractor circuits **406**, **407**, and **408**, respectively. The left-hand width measurements T_L, M_L, and B_L are conveyed directly to the subtrahend inputs of subtractors **406**, **407**, and **408**, respectively.

In the embodiment of FIG. 4, it is assumed that characters are sequentially positioned at a fixed rate of one character-time per character, Delay units **601**, **602**, and **603** cause a single character-time delay to be applied to the signals conveying the right-hand measurements from buffer memory **403** to subtractors **406**, **407**, and **408**. As a result, the outputs of delay units **601**, **602**, and **603** represent the three right-hand measurements of the character previously placed in the print line, one character-time before the positioning of the present new character.

In the embodiment of FIG. 4 delay units **601**, **602**, and **603** may be any suitable devices for providing delay to electrical signals such as electro-acoustic or resistor-capacitor delay lines. In other embodiments it may prove advantageous to position characters asynchronously and not at a fixed rate. In this case delay units **601**, **602**, and **603** may be replaced by other suitable storage elements for registering and retaining the measurement of the character previously placed in the print line.

Subtractors **406**, **407**, and **408** form the three tentative width measurements W_1, W_2, W_3 in accordance with Equations 1. Maximum value circuits **409** and **410**, shown in detail in FIG. 5, combine to select the greatest of these three tentative width values, according to Equation 2, which is subsequently conveyed to accumulator **411**. Maximum value circuit **409** selects the larger of the two tentative width measurements appearing on the outputs of subtractors **406** and **407**. Maximum value circuit **410** selects the larger of the quantities produced by maximum value circuit **409** and subtractor **408**. Accumulator **411** sums the selected width measurement with the displacement of the previous character to form and store the displacement of the requested character.

The embodiment of FIG. 4 is implemented with spacing measurements expressed as binary numbers appearing on a number of parallel leads. Subtractor circuits and accumulator circuits suitable for this implementation are well known in the art. For details see any of a number of standard texts on logic circuit design, for example, Chapter 9 of *Pulse, Digital and Switching Waveforms by* Millman and Taub (McGraw-Hill, 1965).

FIG. 5 shows an embodiment of a maximum value circuit suitable for use in circuits **409** and **410** in the

embodiment of FIG. 4. Two multi-bit binary numbers A and B comprised of parallel bits A_1 through A_k and parallel bits B_1 through B_k, respectively, where k represents the number of bits in A or B, are logically compared on a parallel bit-wise basis to produce an output number C comprised of bits C_1 through C_k, which is equal to A or B, whichever is numerically the larger of the two. High order bits A_1 and B_1 are compared in cell **500**; low order bits A_k and B_k are compared in cell **502**. A cell of the type shown at **501** compares each of the $(k-2)$ intervening bits of A and B, represented by bits A_j and B_j, where $O<j<k-2$.

The larger binary number A or B is identified as being the number which is equal to one in the highest order bit positioned in which the two numbers differ. That is, either A or B will be larger, depending on which number has the highest order bit equal to one while the corresponding bit of the other number is equal to zero. If A and B do not differ in any bit position, then A and B are equal and $C=A=B$.

The communicating signals a_1 through a_{k-1}, and b_1 through b_{k-1} indicate to adjacent cells whether A or B has been determined to be the larger number. At any given point in the circuit shown in FIG. **5**, $a=1$ and $b=0$ indicates that A has been determined to be the larger. Consequently, all subsequent cells will produce the bits of A as their output. When $a=0$ and $b=1$, B has been determined to be the larger, and the output of all successive cells will be the bits of B. When $a=b=0$, neither number has yet been identified as the larger, and each subsequent cell is free to make its own comparison of the bits of A and B. In this way, the highest order cell to find the bits of A and B to be different will determine the output of all of the succeeding, i.e., lower order, cells.

The operation of this circuit of FIG. **5** is as follows. High order cell **500** provides OR gate **511** to logically combine the two input bits A_1 and B_1 and form the output bit C_1. Thus, if either or both of the input bits is a logical one, the output bit will also be a logical one, thereby functionally producing the numerical maximum value of the numerical value represented by the two input bits. AND gate **508** includes a normal input terminal **714** and an inhibiting input terminal **713**, denoted in FIG. **5** by the circular dot associated with input terminal **713**. The construction of such an AND gate is well known (see Section 9–6 of the Millman and Taub reference especially FIG. **9–18** of page 325). The following truth table relates the logical inputs and outputs of AND gate **508**:

Input Terminal 714	Inhibiting input Terminal 713	And Gate 508 Output
1	1	0
1	0	1
0	1	0
0	0	0

The application of a logical one to inhibiting input terminal **713** may be said to "inhibit" AND gate **508** since the output of AND gate **508** will be a logical zero regardless of the logical value at normal input terminal **714**. Similarly, the application of a logical zero to normal input terminal **714** may be said to inhibit AND gate **508**. The application of a logical zero to inhibiting input terminal **713** may be said to "enable" AND gate **508** since the output of AND gate **508** will take on the same logical value of normal input terminal **714**. When AND gate **508** is thus enabled the logical signal on the normal input terminal **714** may then be said to be "gated through" AND gate **508**. AND gates **504**, **505**, **506**, **507**, **509**, **704** and **705** each have one normal input terminal and one inhibiting input terminal denoted in each instance by a circular dot in the drawing. Each of these AND gates will be understood to operate in a manner analogous to that described with reference to AND gate **508**. AND gate **508** will produce $a_1=1$ when $A_1=1$ and $B_1=0$. Similarly, AND gate **509** will produce $b_1=1$ when $A_1=0$ and $B_1=1$. When $A_1=]B_1$, $a_1=b_1=0$. Thus, the correct communication is provided to the next cell.

Low order cell **502** accepts information bits a_{k-1} and b_{k-1} from the next higher adjacent cell and input bits A_k and B_k. When previous cells have determined that the number A is the larger, AND gate **705** will be inhibited by the one appearing on lead a_{k-1}, AND gate **704** will be enabled by zero appearing on b_{k-1}. Thus the bit A_k will be gated through AND gate **704** and OR gate **712** to appear as the output bit C_k. Similarly, bit B_k will be gated through AND gate **705** and or gate **712** as output bit C_k when communication from the previous cell makes lead $A_{k-1}=0$ and $b_{k-1}=1$. When no determination has been made by prior cells, and both communicating leads are equal to zero, both AND gates **704** and **705** are enabled, thereby gating bits A_k and B_k through to OR gate **712**. Output C_k thus takes on a logical value equal to the numerical maximum of the two input bits A_k and B_k by the action of OR gate **712** in the same fashion as OR gate **511**, described above.

Generic cell **501** processes communication signals entering on inputs a_{j-1} and b_{j-1} the same way as above described for cell **502**. AND gates **504**, **505**, and OR gate **512** are assembled in the same fashion and perform the same functions as described for AND gates **704** and **705** and OR gate **712**, and produce an output on lead C_j. What remains is a description for cell **501** of the manner in which OR gates **513** and **514** produce communication signals to the next successive cell over leads a_j and b_j.

If $a_{j-1}=1$ and $b_{j-1}=0$, it means that binary number A has previously been determined to be the larger; this information must be passed to the next cell. The one on lead a_{j-1} is conveyed through OR gate **513** directly onto lead a_j. The values of A_j and B_j can thus have no effect on the value of a_j when $a_{j-1}=1$. The one on lead a_{j-1} also inhibits AND gate **505**, thereby inhibiting AND gate **507**. The values of A_j and B_j can thus have no effect on the value of b_j when $a_{j-1}=1$. Since lead $b_{j-1}=0$, OR gate **514** produces a zero on lead b_j, all inputs to OR gate **514** being logically zero. Therefore, the combination entering on the left side of cell **501** is preserved on the right side regardless of the values of A_j and B_j.

Analogous activity takes place when it is determined in the previous cell that B is the larger of the two numbers, thereby producing a zero on output lead a_j an a one on output lead b_j.

When there has been no prior determination of which number has the larger magnitude, i.e., $a_{j-1}=b_{j-1}=0$, cell **501** may be called upon to perform this function. If A_j and B_j bits are both equal to one, AND gates

506 and 507 are both inhibited, insuring that a_j and b_j are zero. If A_j and B_j are both zero, AND gates 504 and 505 are inhibited, thereby inhibiting AND gates 506 and 507 and assuring an output of $a_j = 0$ and $b_j = 0$. If $A_j = 1$ and $B_j = 0$, AND gate 504 is enabled, thereby enabling AND gate 506 and providing a one output through OR gate 513 on lead a_j. AND gates 505 and 507 are inhibited, thereby assuring $b_j = 0$. Analogous signals can be traced in the case of i $A_j = 0$ and $B_j = 1$.

The circuit of FIG. 5 is constructed of OR gates and of AND gates with inhibiting inputs. Such elements are well known to those versed in the logic design art. Circuit details of these elements will also be found in Chapter 9 of the above-referenced text by Millman and Taub.

FIG. 6 represents an additional embodiment of the typesetting algorithm described above. FIG. 6 shows a segment of a computer program suitable for compiling and execution on a general purpose digital computer which provides the control signals for the typesetting or display mechanism.

The computer program statements shown in FIG. 6 are written in FORTRAN IV. For this program, information relating to each text character is arranged in a number of one dimensional arrays of data. The array TR(I) is a one dimensional array storing a number of values T_R as defined for FIG. 2, where I is an index which points to the characters in the string of text which is to be displayed. Similar one dimensional arrays store the values of the variables T_L, M_R, and the rest. The variables W1, W2 and W3 represent storage locations which contain the results of the tentative width calculations as described in Equations 1 above.

Statements 100, 101, and 102 direct the computer to retrieve from memory the three left-hand spacing measurements for the new character indicated by index I, subtract them from the corresponding right-hand spacing measurements for the previous character in the print line, and store the difference as tentative result measurements in the variable locations W1, W2, and W3.

Statements 103 through 105 provide the maximum value function described above in Equation 2. The variable W is a storage location where the maximum of the three tentative width values will be stored after the execution of these three statements. In statement 103 storage location W is initialized with the value of the first tentative width measurement W1. Statement 104 is a conditional substitution statement. If the quantity stored in location W2 is greater than the quantity stored in location W, the quantity stored at location W is replaced with the quantity stored at location W2. Statement 105 performs similarly. If W3 is greater than W, W is replaced by W3.

Execution of these three statements will result in the conditional assignment of W1 as the maximum of the three tentative widths. If W2 is greater than W1, W2 will be conditionally assigned as the maximum value. W3 will then be compared with the results and assigned as the maximum value if it is larger than the other two. Having reached statement 106, the maximum of the three tentative widths is stored in location W. Statement 106 now adds the width value to the storage location K wherein the displacement on the print line of the new character is stored. Statement 107 is a subroutine call to a segment of program by the name of PLACE which places the character pointed to by the current value of the index I on the display surface with a horizontal displacement equal to the value stored in location K. The values of K and I are arguments to the subroutine PLACE.

Further instructions in the program, not shown, select sequential characters from a string of text, increment index pointer I, and define in detail machine steps of the character generation subroutine PLACE. Also not shown in FIG. 6 are various control statements and additional FORTRAN IV statements which are necessary to compile and execute a working program. Such details are well known to those versed in the programming art. See, for example, *Basic FORTRAN IV for the GE-600 Series Computers*, by Rudolph G. Schenbeck (Addison-Wesley, 1968).

What is claimed:

1. The improved machine method of typesetting characters comprising the steps of:
   - generating for each character a multiplicity of spacing measurements directed to the right and to the left of each character;
   - subtracting the right directed measurements associated with a first character from the left directed measurements associated with a second character;
   - selecting a maximum of the resulting differences; and
   - spacing said second character from said first character according to said selected maximum.

2. The method of spacing alphanumeric characters along a linear display comprising the steps of
   - establishing, for each character, optimum spacing values for a plurality of registered positions vertically displaced from one another on each said character;
   - comparing said spacing values at all registered positions for pairs of said characters to be displayed juxtaposed; and
   - displaying said pairs with a spacing corresponding to the optimum compared spacing values.

3. Apparatus for spacing a new character over from a previous character in a character compositor comprising
   - storage means containing data describing each character and data comprising a multiplicity of spacing measurements for each character;
   - means for subtracting the multiplicity of spacing measurements of the previous character from the multiplicity of spacing measurements of the new character;
   - means for selecting the maximum of the differences thus formed; and
   - means for positioning the new character at a position wherein the new character is spaced over from the previous character by said maximum difference.

4. In a character compositor, an improved means for spacing characters comprising:
   - means for storing a multiplicity of spacing measurements for each character, some of said measurements directed to the right of each character, some of said measurements directed to the left of each character;
   - means for subtracting the right directed measurements of a first character from the left directed

measurements of a second character thereby producing a multiplicity of tentative width measurements;

means for selecting the maximum of said width measurements; and

means for displaying said second character at a position spaced from said first character by said selected measurement.

* * * * *