May 5, 1970 J. C. SIMS, JR 3,509,817

LINE PRINTING WITH PROPORTIONAL SPACING AND JUSTIFICATION

Filed Nov. 21, 1968 9 Sheets-Sheet 1

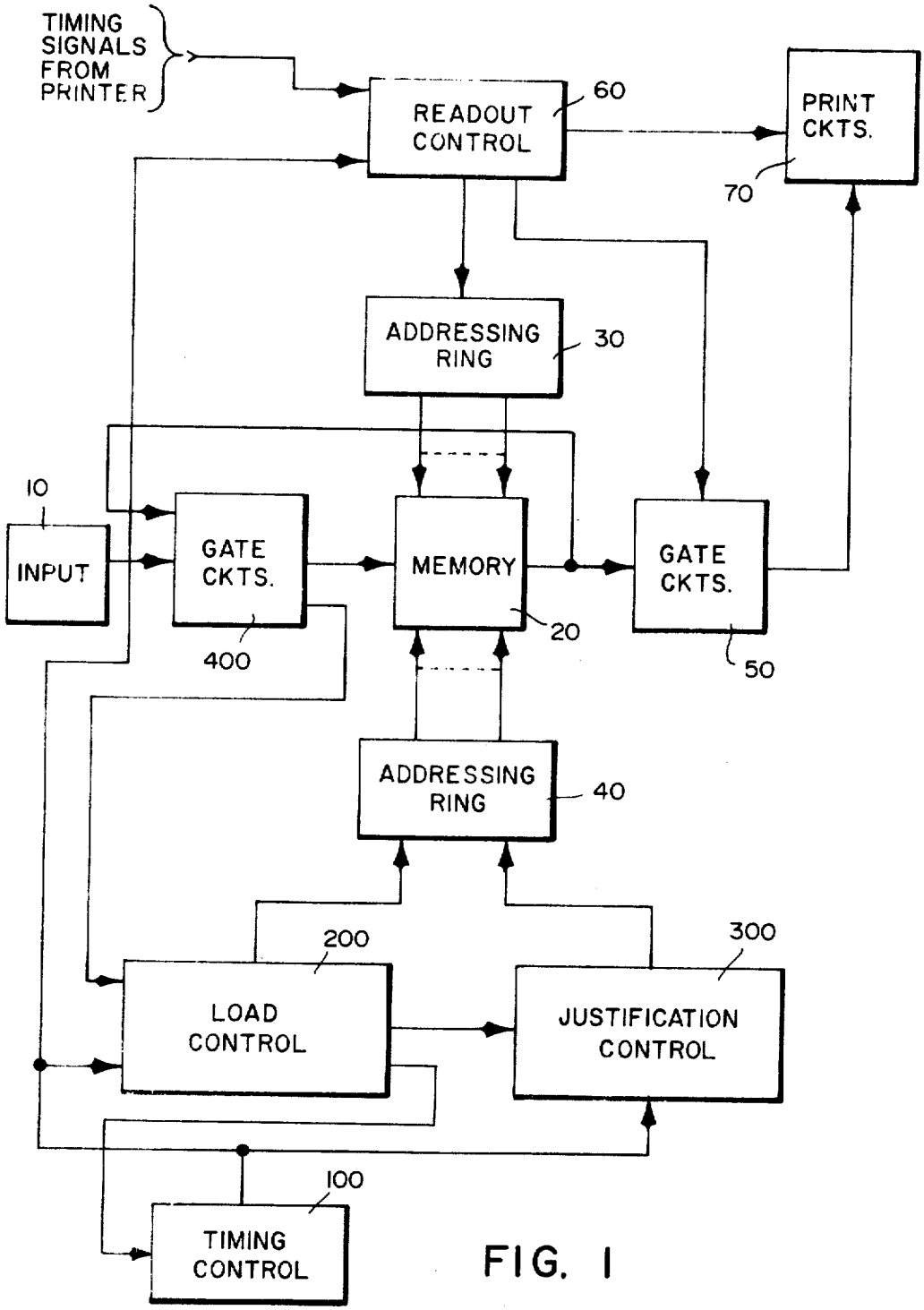


FIG. 1

INVENTOR.
JOHN C. SIMS, Jr.
BY Richard H. Smith
ATTORNEY

LINE PRINTING WITH PROPORTIONAL SPACING AND JUSTIFICATION

Filed Nov. 21, 1968 9 Sheets-Sheet 2

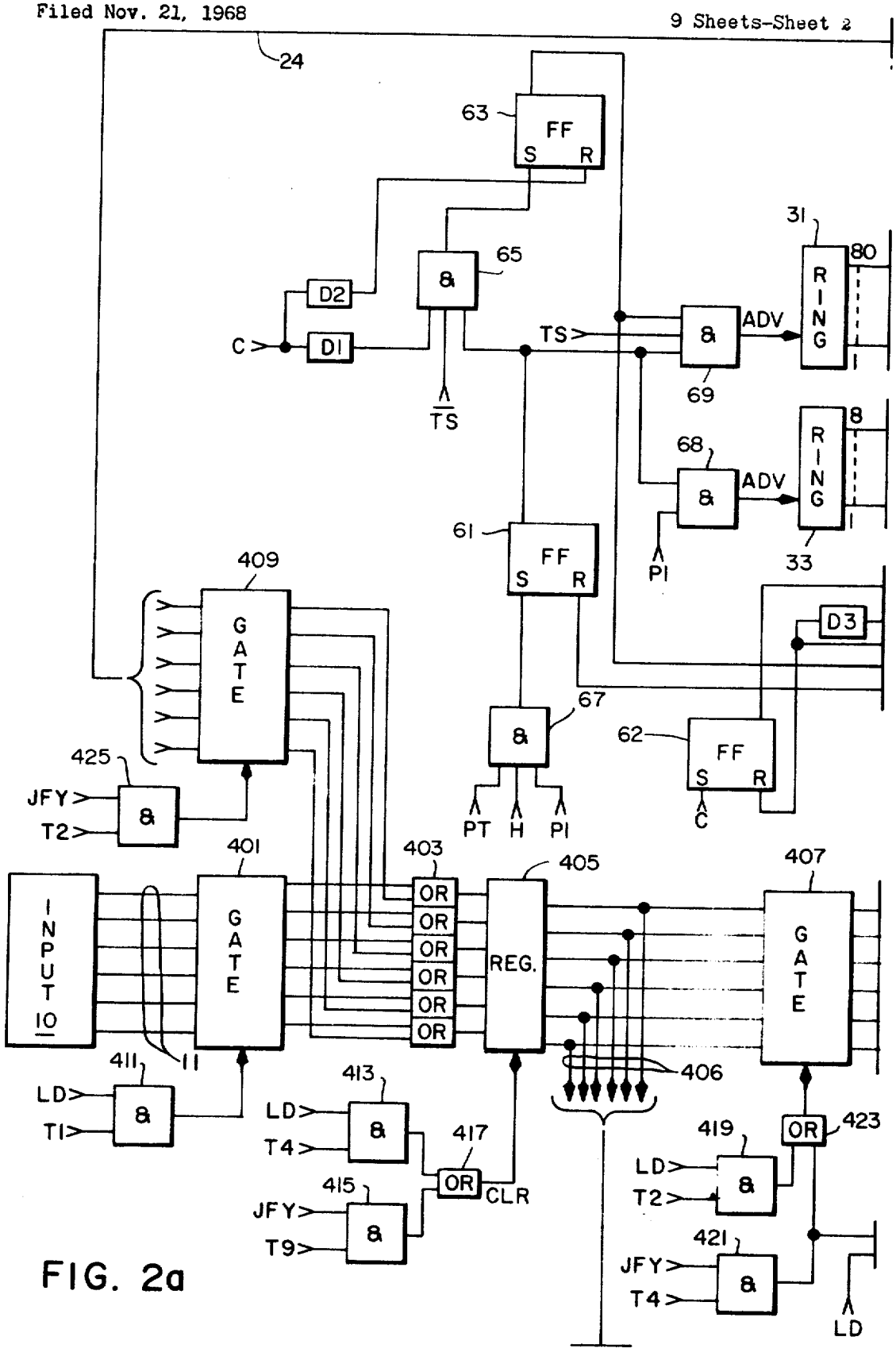


FIG. 2a

LINE PRINTING WITH PROPORTIONAL SPACING AND JUSTIFICATION

Filed Nov. 21, 1968 9 Sheets-Sheet 3

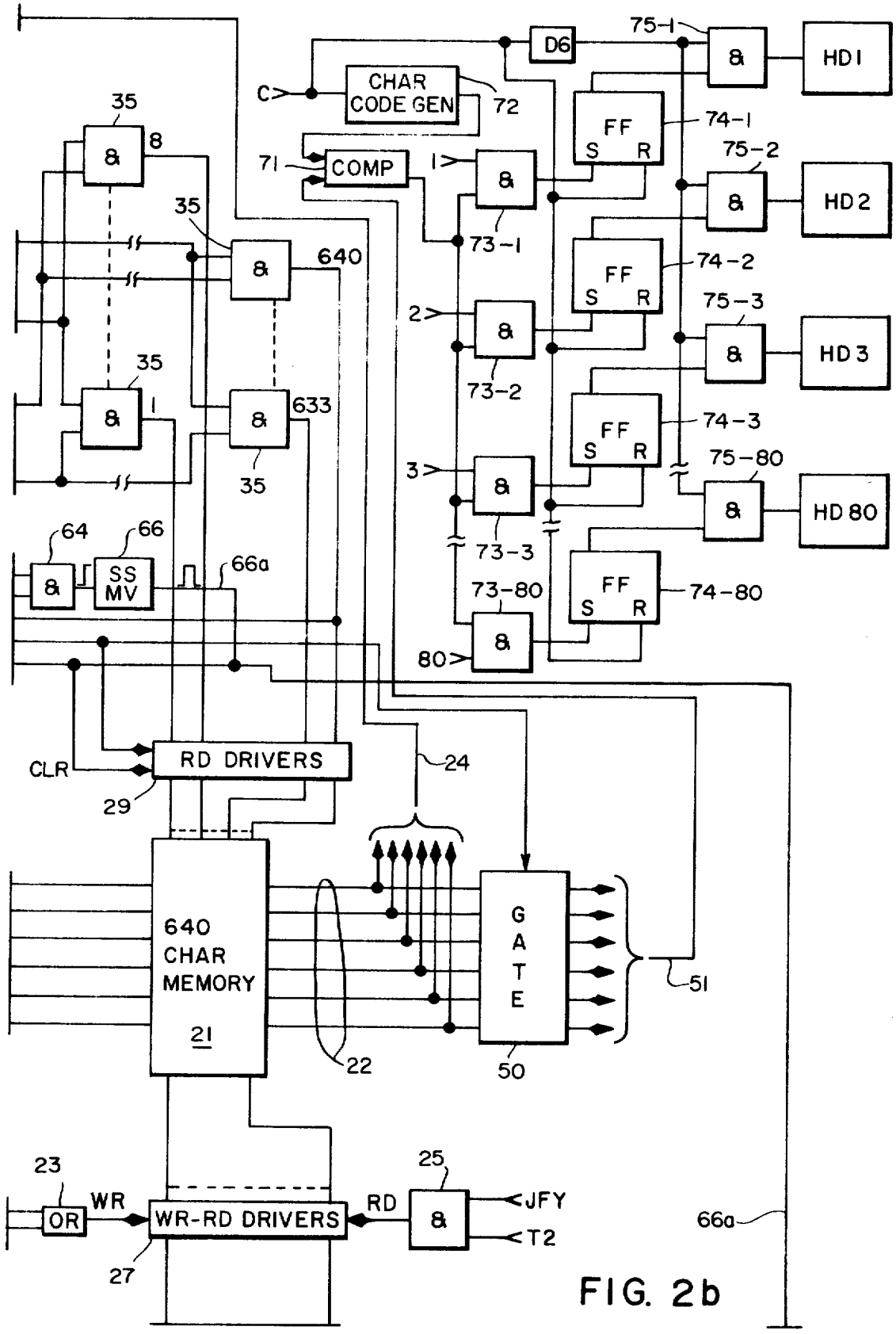


FIG. 2b

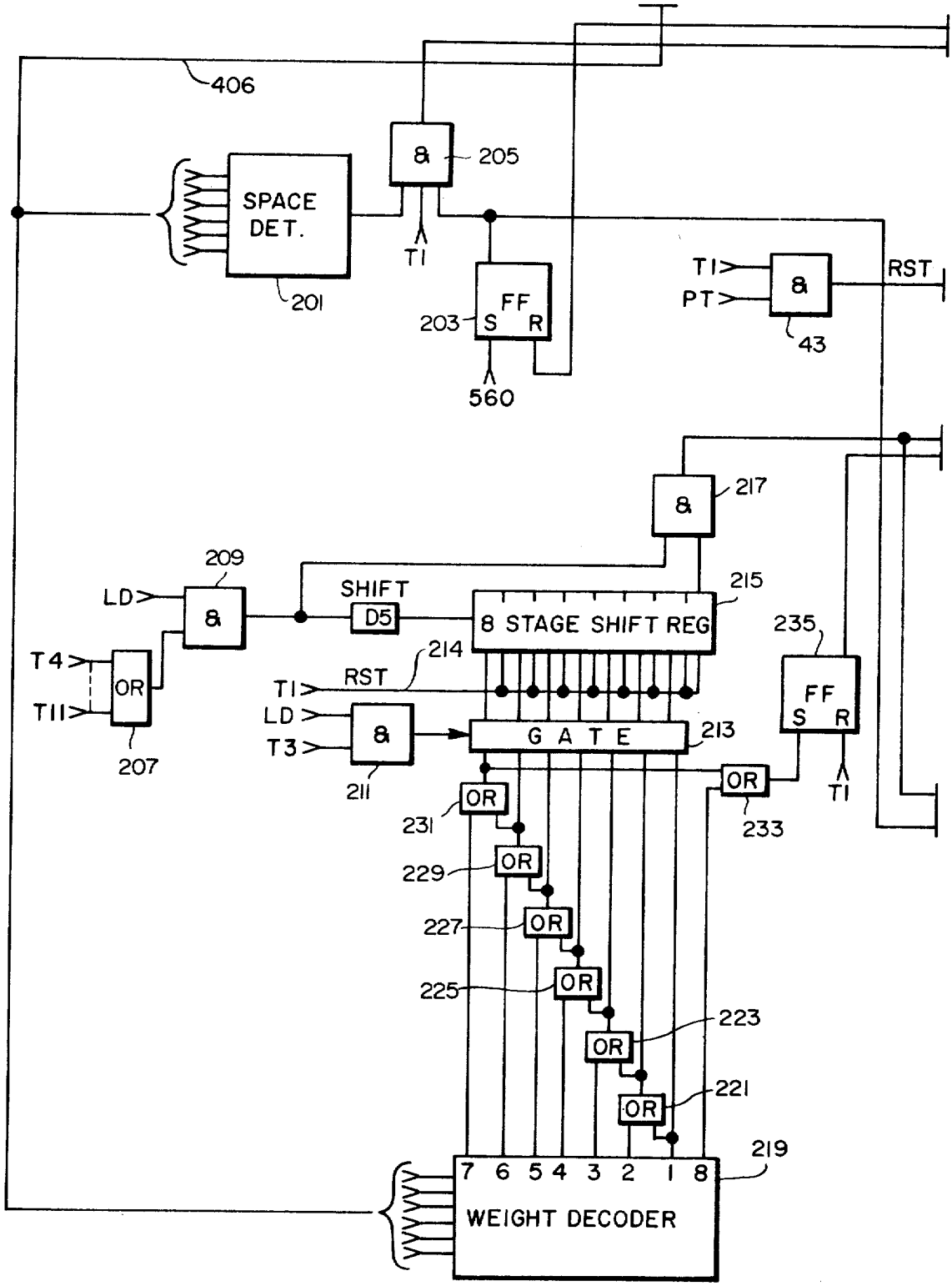

406
&
205
SPACE DET.
201
T1
FF
S R
203
560
T1
PT
&
RST
43
&
217
209
LD
&
SHIFT
D5
8 STAGE SHIFT REG
215
235
T4
T11
OR
207
214
T1
RST
LD
T3
&
211
G A T E
213
FF
S R
T1
OR
233
OR
231
OR
229
OR
227
OR
225
OR
223
OR
221
7 6 5 4 3 2 1 8
219
WEIGHT DECODER

FIG. 2c

LINE PRINTING WITH PROPORTIONAL SPACING AND JUSTIFICATION

Filed Nov. 21, 1968 9 Sheets-Sheet 5

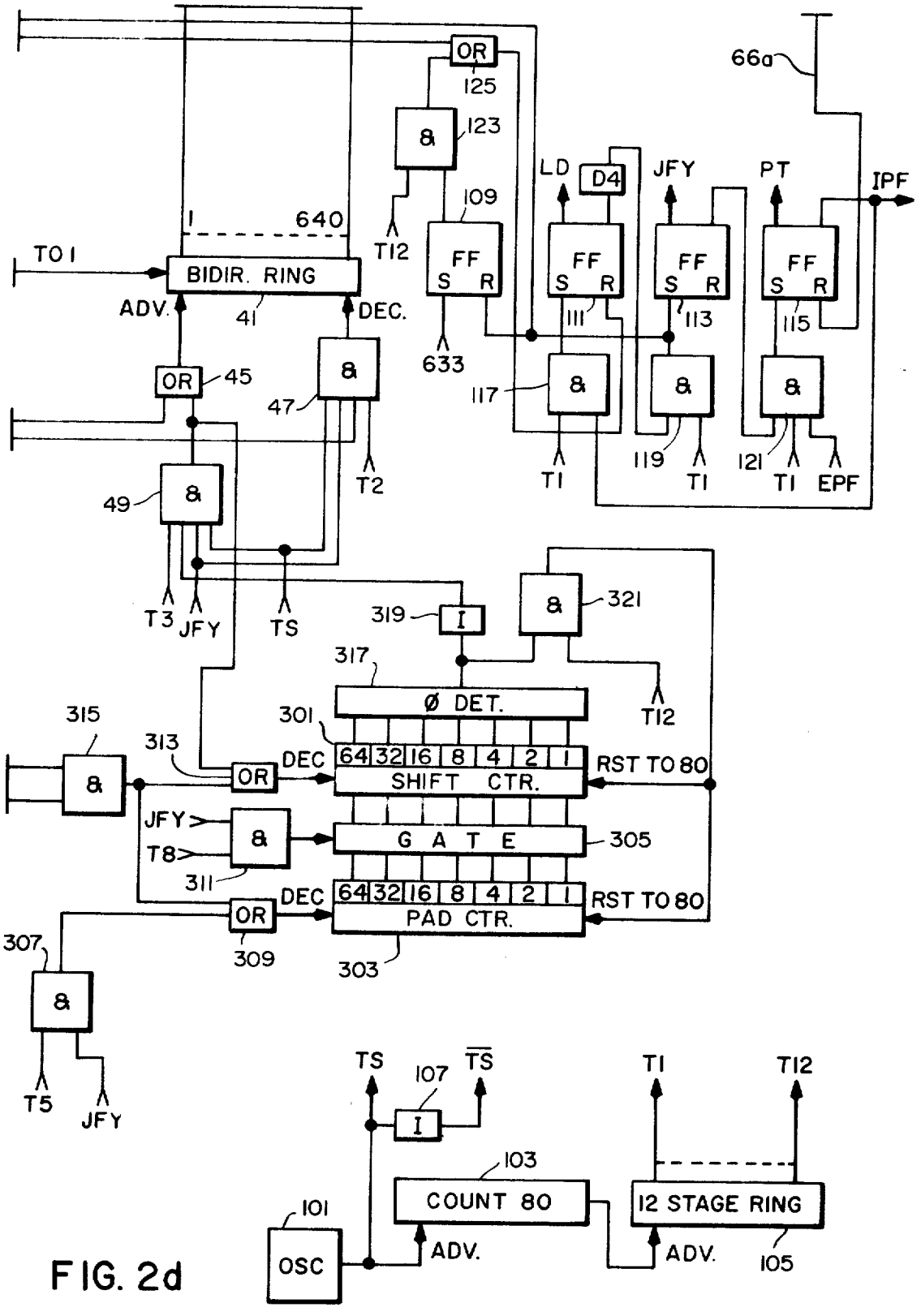


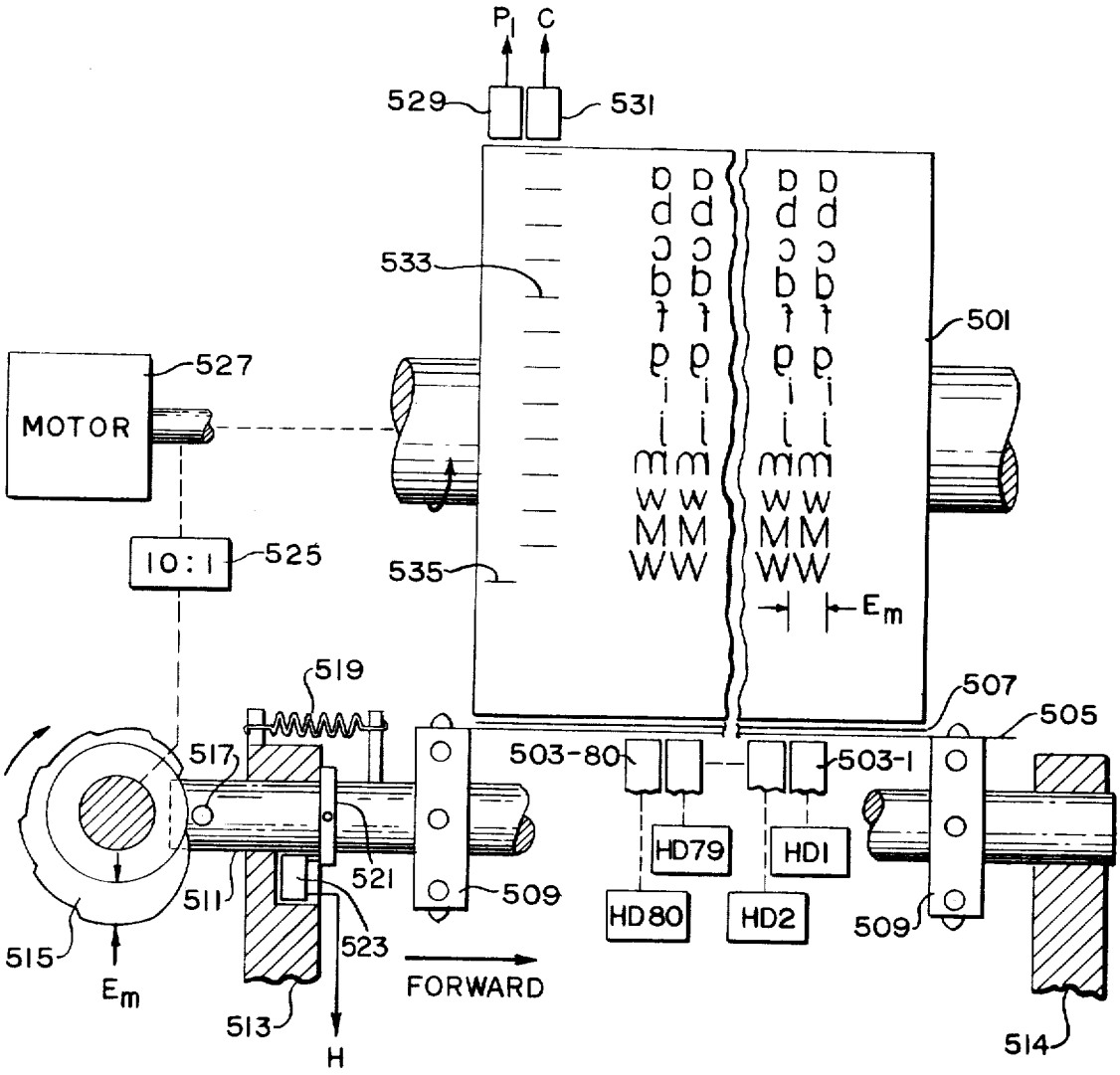
FIG. 2d

May 5, 1970 J. C. SIMS, JR 3,509,817

LINE PRINTING WITH PROPORTIONAL SPACING AND JUSTIFICATION

Filed Nov. 21, 1968 9 Sheets-Sheet 6

FIG. 3



INVENTOR
JOHN C. SIMS, JR.
BY Richard H. Smith
ATTORNEY

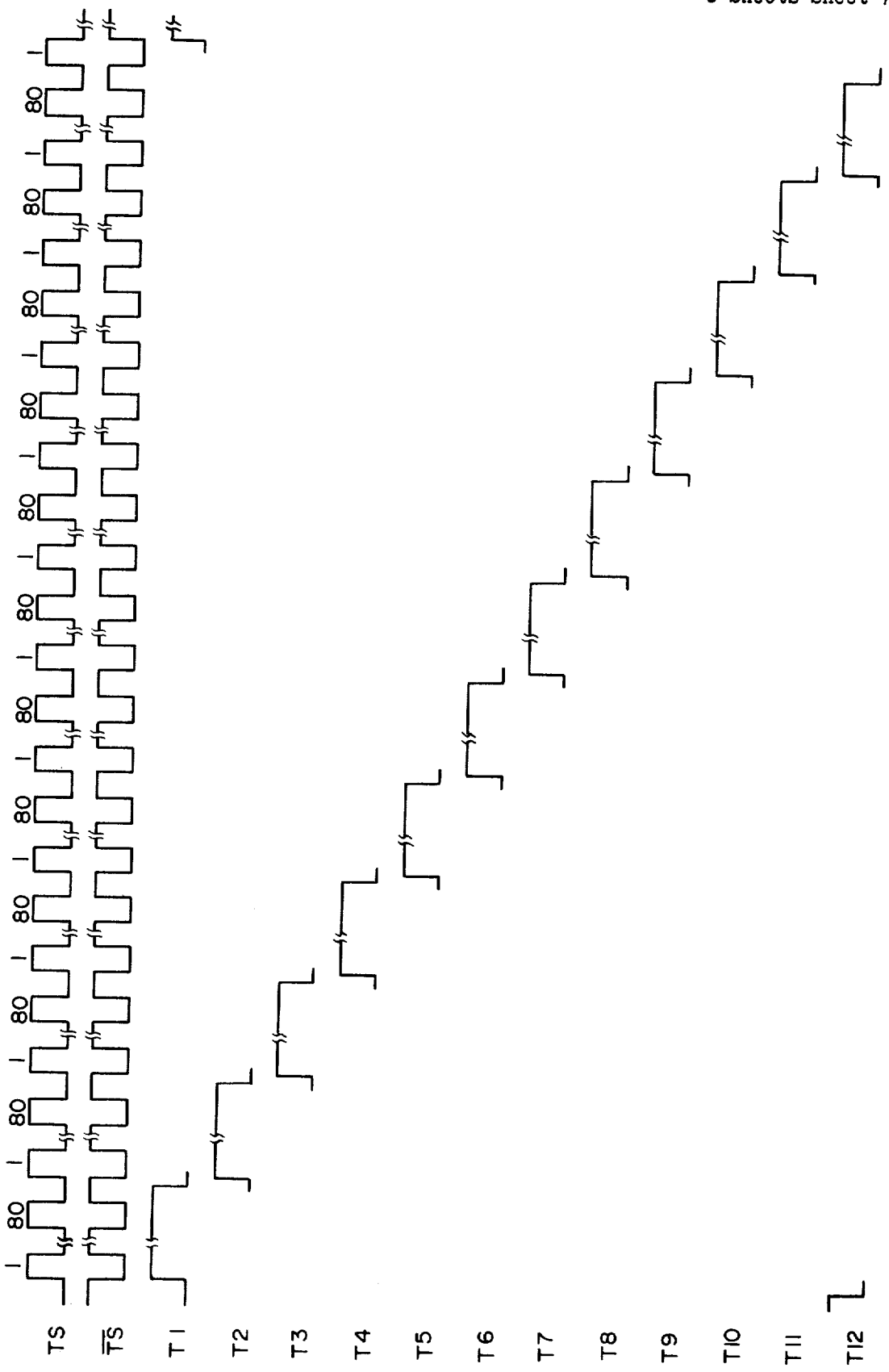


FIG. 4

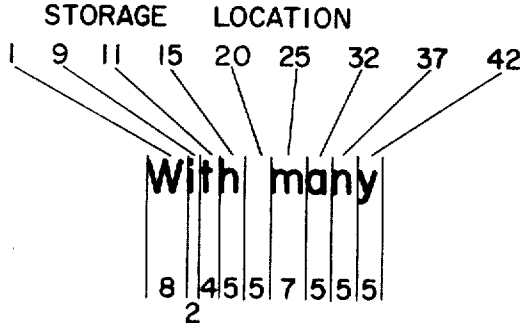


FIG. 5

REV 1	(1)	(9)	17	(25)	33	41	49	57
REV 2	2	10	18	26	34	(42)	50	58
REV 3	3	(11)	19	27	35	43	51	59
REV 4	4	12	20	28	36	44	52	60
REV 5	5	13	21	29	(37)	45	53	61
REV 6	6	14	22	30	38	46	54	62
REV 7	7	(15)	23	31	39	47	55	63
REV 8	8	16	24	(32)	40	48	56	64

Wi m
1 2 4
Wi m y
6
Wit m y
2
NO PRINT
Wit m ny
5
NO PRINT
With m ny
2
With many
4

FIG. 6

INVENTOR
JOHN C. SIMS, JR.
BY Richard H. Smith
ATTORNEY

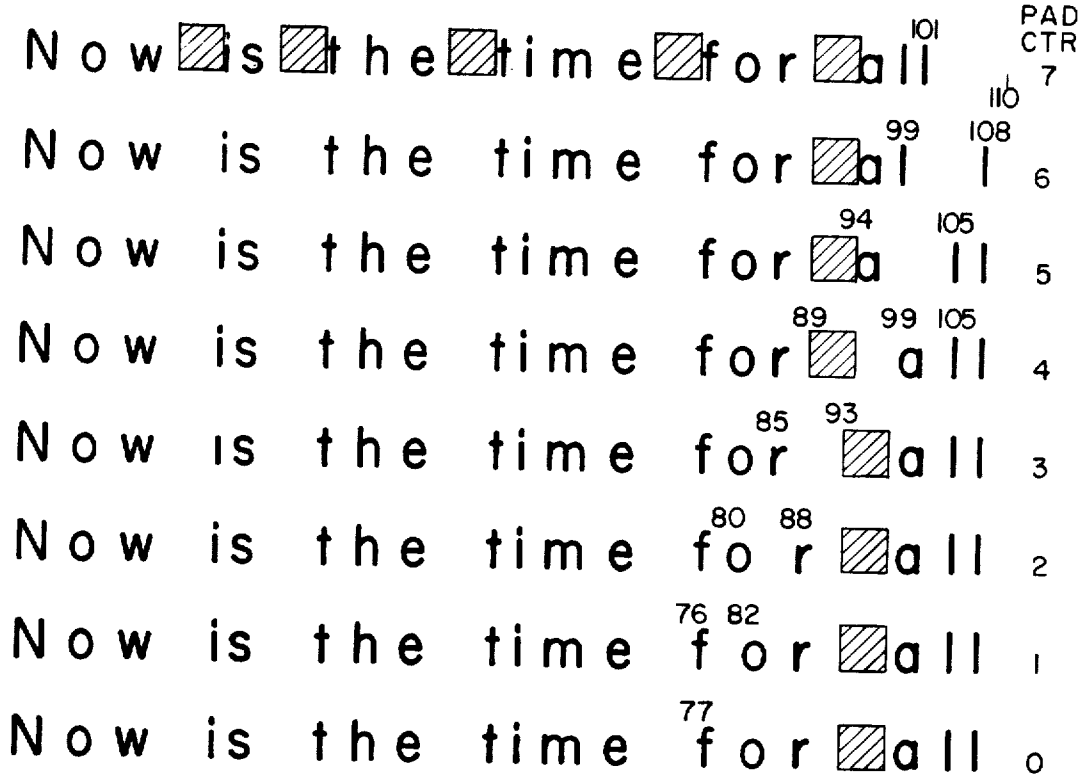


FIG. 7

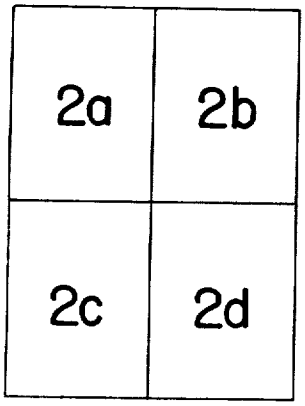


FIG. 8

INVENTOR
JOHN C. SIMS, JR.
BY Richard H. Smith
ATTORNEY

United States Patent Office 3,509,817
Patented May 5, 1970

3,509,817

LINE PRINTING WITH PROPORTIONAL SPACING AND JUSTIFICATION

John C. Sims, Jr., Sudbury, Mass., assignor to Mohawk Data Sciences Corporation, Herkimer, N.Y., a corporation of New York

Filed Nov. 21, 1968, Ser. No. 777,763
Int. Cl. B41j 9/14, 19/58
U.S. Cl. 101—93 8 Claims

ABSTRACT OF THE DISCLOSURE

An on-the-fly line printer prints justified, proportionally spaced lines by stepping the paper laterally during the print cycle. A print line is subdivided into N character position increments and the widest printable character occupies X increments. There are N/X print hammers and the paper is laterally stepped $X-1$ increments during the print cycle. Prior to printing, the characters are stored in a buffer memory having N character locations and each character is assigned to a location dependent on the cumulative widths of the previously loaded characters. Justification is effected by shifting the characters in memory so that the last character in the line is moved to a location corresponding to the right margin and the remaining characters are shifted in behind the last character one at a time, one position increment being added to the inter-character spaces at the time of shifting until the line is filled in. After justification the memory is read out for printing in X basic scan sequences, each sequence covering 1/X of the memory locations in multiples of X. After each scan-print sequence the paper is laterally shifted one increment until the full line is printed after X scan-print sequences.

BACKGROUND OF THE INVENTION

This invention relates to line printing, and more particularly, to a line printer capable of printing justified lines of proportionally spaced characters at relatively high speeds.

Heretofore, production of proportionally spaced, justified print copy has been a relatively low speed, high cost operation. Typewriters and Linotype machines, for example, have been equipped with means to enable the production of justified and proportionally spaced copy but both types of apparatus require a relatively lengthy period for preparation of the print master since both are operator time-limited. Furthermore, the Linotype technique is relatively expensive and is economically feasible only on a high volume production basis.

OBJECTS AND SUMMARY OF THE INVENTION

It is therefore an object of the invention to provide an improved apparatus for producing proportionally spaced and justified print copy at several times the speed possible with a typewriter and at only a slight increase in cost as compared to the typewriter technique.

Another object is to provide an improved on-the-fly line printer capable of printing proportionally spaced and justified print lines.

Still another object is to provide an improved data buffering memory and controls therefor which enable the use of a conventional line printer, with slight mechanical modification, as a means for generating proportionally spaced and justified master sheets which can be economically reproduced on a volume basis by conventional processes.

In accordance with the invention, a print line is subdivided into a plurality of discrete character position increments each equal to the narrowest printable character.

A buffer memory is provided having a number of character storage locations equal to the number of character positions increments in a line. Each character to be printed is received by the buffer memory and stored at a character location corresponding to the position in the line at which it is to be printed, taking into account the widths of the previous characters of the line. Reception of data characters into the buffer is automatically terminated at some point before the memory has been completely filled. When memory loading has been completed, the storage positions of certain characters in the line are shifted to effect justification of the right-hand margin. This is done by systematic shifting of the last characters of the line to the right while slightly increasing the spacing between them until the unfilled portion of the line is completely taken up.

Once the loading and justification cycles are terminated, the memory is systematically read out and the characters are printed by print means including a continuously moving type carrier and a plurality of print hammers. Each hammer is wide enough to print the widest character and the number of hammers represents the maximum number of widest characters that can be printed in a line. Means are provided for shifting the print medium in the direction of the print line relative to the print hammers by an amout equal to the width of the widest printable character. During this shifting movement, the type carrier presents a full type font to each hammer each time the print medium advances one basic character position increment so that after the full shift movement has been completed each character position increment of the line has been presented with all printable characters. Thus, the system is able to print any character at any character position increment in the line. During this process the buffer memory is read out in synchronism with the shifting movement of the print medium so that each character in memory is printed in the line at the character position increment corresponding to its memory storage location.

These and other objects, features and advantages will be made apparent by the following detailed description of a preferred embodiment of the invention, the description being supplemented by drawings as follows:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic diagram showing the overall arrangement of the buffer memory and control circuits of a preferred embodiment of the invention.

FIGS. 2*a*–2*d*, when assembled in the manner indicated in FIG. 8, constitute a detailed schematic diagram of the circuits generally depicted in FIG. 1.

FIG. 3 is a schematic diagram illustrating the basic mechanical components of a preferred embodiment of the invention, including the print drum, print hammers and paper feed mechanism.

FIG. 4 is a timing diagram showing the basic timing signals generated by the timing circuit **100**.

FIG. 5 is a schematic diagram illustrating certain principals of the proportional spacing aspects of the invention.

FIG. 6 is a schematic diagram illustrating the operation of the invention in generating a portion of a proportionally spaced print line.

FIG. 7 is a schematic diagram illustrating the operation of the preferred embodiment during execution of a justification cycle.

FIG. 8 is a schematic diagram depicting the proper orientation of FIGS. 2*a*–2*d*.

GENERAL DESCRIPTION

FIG. 1 shows the general arrangement of the buffer memory and print control circuits in accordance with a preferred embodiment of the invention. Also, in accordance with the preferred embodiment, one complete print

line is subdivided into 640 character position increments and the widest printable character W is eight position increments wide. Therefore, a maximum of eighty such characters are printable in one line and there are thus eighty print hammers in the print mechanism together with a print drum having eighty identical character font columns.

Generation of each line of print requires three basic operational cycles: a load cycle, a justification cycle and a print cycle.

During the load cycle, the characters to be printed are presented to the system serially by an input circuit **10**, as shown in FIG. 1. A buffer memory **20** is provided with 640 character storage locations to permit accumulation of a full line before any printing takes place. Each character is admitted by a gate circuit **400** and is temporarily held in a one-character register therein until it is transferred to the proper location in memory **20**. The first character of the line is automatically stored in character location **1** of memory **20** under the control of an addressing ring **40**. After the first character is stored, a load control circuit **200** causes the addressing ring to advance a number of character locations equal to the width (in character positoin increments) of the first character. When the second character of the line is received, it is entered into the newly accessed storage location of memory **20** and thereafter the load control circuit **200** causes ring **40** to advance a number of storage locations equal to the width of the second character.

This sequence of steps continues until the load cycle is terminated by receipt of a "space" character after the addressing ring **40** has advanced into the "justification zone." The latter defines those character storage locations in memory **20** corresponding to a predetermined number of character position increments at the right-hand end of a line. The number of increments in the justification zone constitutes the maximum number of increments that will have to be "filled up" during the justification cycle. Thus, for appearance sake, it is desirable to have the justification zone as short as possible. However, as the length of the zone is decreased the likelihood that hyphenization will be required at the end of a line is increased. Thus, in order to shorten the justification zone as much as possible while still virtually eliminating any need for hyphenization, the justification zone in the preferred embodiment begins at storage location **560**, thereby resulting in a zone which is eighty character position increments long. Since the average character width may be presumed to be four position increments, this zone provides space for approximately twenty characters whereby chances of a space character falling in the zone become almost 100 percent, virtually guaranteeing that each load cycle will be terminated before the maximum capacity of the buffer memory is reached.

Immediately upon termination of the load cycle, the system switches into the justify cycle. The justify cycle is controlled by a pair of binary counters in the justification control circuit **300**. One counter is a "shift" counter and the other is a "pad" counter. These counters are set to a count indicative of the number of character position increments between the position of the last-loaded character of the line and the right-hand margin. This count is entered into the counters by resetting them to a count of eighty prior to commencement of the load cycle and then decrementing them by one for each step addressing ring **40** is advanced beyond storage location **559** during the load cycle.

The first operation in the justify cycle causes the addressing ring **40** to be decremented to access memory **20** in reverse sequence until a stored character is detected. When a character is detected it is loaded into the one-character register in gate circuits **400** and the operation of the ring **40** is arrested.

Next, ring **40** is advanced in the forward direction toward storage location **640** and each time the ring advances one position the shift counter is decremented by one.

When the counter reaches a count of zero, advancement of the ring is halted and the character in the one-character register is loaded into the memory storage location then accessed by the ring. This storage location corresponds to the right-hand margin of the print line i.e., it represents the extreme right-hand character position increment which that particular character can be permitted to occupy. It will be understood that since characters have different widths the narrower ones will have their left-hand edges closer to the right margin than the wider ones. Since the particular memory storage location that a given character occupies represents the location of its left-hand edge on the print line, justification requires that the last character in the line be stored at location **638** if it is a two-increment character, at position **637** if it is a three-character, etc.

After the last character in the line has been shifted in the memory to a storage location representing the right-hand margin, the count in the pad counter is decremented by one and the shift counter is loaded with the count represented at the pad counter output. If this count happens to be zero the justification cycle terminates. If it is not zero the cycle continues with repetition of the above-described sequence of operations.

During the justification cycle, therefore, the last character in the line is shifted to the last permissible storage location in memory **20** and then the characters preceding it are sequentially moved to the right and stored in locations such that an additional storage location (position increment) is added between each character. This process is continued until the pad counter reaches zero, which is an indication that all of the unused character position increments at the end of the line are taken up.

The print cycle is best understood with joint reference to FIGS. 1 and 3. As schematically indicated in FIG. 3, a continuously rotating drum **501** is arranged to operate in a conventional manner with a group of eighty print hammers **503–1** through **503–80**. The print drum has engraved about its circumference at least one complete font of type characters for each of the print hammers. In FIG. 3 only the first and last two type fonts are shown. Each font is abbreviated for simplification and includes only a sampling of characters to show the range of different character widths. For example, the "i" is the narrowest character and is two position increments wide and the "W" is the widest character and is eight position increments wide. It is noted that the width of the character is measured from a point in alignment with the right edge of the character to a point extending two position increments beyond the left edge of the character. In FIG. 3, the distance *Em* defined by the arrows on the drum surface constitutes eight position increments. It is also to be noted that the right edges of all the characters of one font are in alignment with the right-most portion of the face of the print hammer aligned with that font. Also, the distance from the right edge of one print hammer to the right edge of the adjacent print hammer is equal to the distance *Em*. Approximately one position increment (*Em*/8) is provided between hammers to prevent frictional interaction therebetween. The drum rotates in the direction indicated by the arrow in FIG. 3 and a blank space is provided on the drum surface between the first and the last characters of the font for timing purposes, the reason for which is made apparent subsequently.

Each print hammer is associated with one of eighty driver circuits HD1 through HD80. These circuits are of conventional construction and are part of the overall print circuits **70** (FIG. 1). When a driver circuit is actuated, the associated hammer is propelled against print medium **505** and in turn causes it and an ink-containing ribbon **507** to impact against the surface of the drum. This prints a character on the medium **505** in the usual manner. It is noted that the font characters are engraved on the drum **501** in mirror-image fashion. This, of

course, results in correct character printing on the upper surface of the medium **505**.

Print medium **505** is edge-perforated in the usual manner and is supported in print position by a pair of sprocketed feed tractors **509**, which may be of conventional construction. After each line of printing is complete, means (not shown) operate to index the tractors to feed the print medium **505** out of the plane of the drawing to place the next print line in print position. In accordance with the invention, the paper feed mechanism is also equipped with means for incrementally shifting the print medium in a series of seven steps in the forward (right-ward) direction from a home position. Each step spans a distance equal to *Em*/8).

The feed tractors **509** are mounted on a laterally shiftable shaft **511** which is journaled in machine frame members **513** and **514**. A collar **521**, pinned to shaft **511**, is adapted to butt against the side of frame member **513** to establish shaft **511** in the home position. A tension spring **519** which is connected at one end to the frame and at the other end to shaft **511** biases the shaft in home position. A switch **523** is mounted in a slot in frame **513** and is arranged to detect the presence of collar **521** when the shaft **511** is in home position. A signal H is generated by the switch to denote this condition.

A stepping cam **515** is mounted to rotate in a clockwise direction and cooperates with a follower lug **517** on shaft **511**. A motor **527** is arranged to drive the print drum **501** and the cam **515**. Gear reduction means **525** are provided to reduce the drive imparted to cam **515** by a factor of ten. That is, for each ten revolutions of the motor output shaft and drum **501**, cam **515** makes one revolution.

The cam has eight circular camming segments. The first segment (which is shown aligned with follower **517** in FIG. 3) has the shortest radius and permits shaft **511** to reside in the home position for one-tenth of the cam revolution which represents one revolution of drum **501**. The next circular cam segment shifts shaft **511** to the right one character position increment (*Em*/8) and holds it there for one revolution of the print drum. The next six cam segments each shift the shaft one additional character position increment to the right and each of them also holds the shaft stationary for one revolution of the print drum. As shown in FIG. 3, the radial difference between the highest and lowest cam segments equals *Em* so that the total rightward shaft movement equals *Em*.

After the high dwell of the cam has passed follower **517**, a transitional segment on the cam surface permits spring **519** to return the shaft to home position. This occurs over two tenths of a cam revolution (two print drum revolutions). During this time the print medium is also undergoing a line feed operation. The drive linkage between cam **515** and motor **527** is arranged such that the incremental rightward shifting of shaft **511** always occurs when the blank space on the print drum is opposite the print hammers.

The print drum has inscribed about its periphery a plurality of magnetically detectable marks **533** and **535**. There is one mark **533** for each horizontal row of type characters. A magnetic sensing transducer **531** is positioned to detect the marks **533** as the print drum rotates and produces a pulse C each time a mark is sensed. The single mark **535** is positioned to coact with a transducer **529** so that a pulse P1 is produced for each revolution of the print drum. The transducers **529** and **531** are mounted, as is conventional, for angular adjustment about the center of rotation of the print drum. This, of course, is necessary to permit proper phasing of the timing pulses C and **P1** to compensate for the various mechanical and electrical delays inherent in the printing mechanism and control circuits.

A print cycle is initiated when the justification cycle has terminated and when the electronic controls have established that the shaft **511** has returned to home position and that paper feeding has terminated. The full print cycle consists of eight revolutions of drum **501**. During each drum revolution a complete scan of every eighth character storage location of memory **20** is effected once for each C pulse. During the first drum revolution the eighty character positions **1**, **8**, **15**, **22**, etc., through **633** are scanned. As each storage location is read out, the data stored therein is fed through gate circuits **50** (FIG. 1) to a comparator in the print circuits **70**. The comparator matches each of the eighty characters from memory **20** with a coded representation of the character on the print drum which is then coming into print position. Each time an equal comparison occurs, the output of the comparator is used to condition a hammer driver circuit for later firing. After all eighty comparisons have been completed, a single fire pulse is supplied to all eighty hammer driver circuits to simultaneously fire those which had been conditioned by the comparator.

Upon completion of the last comparison and hammer firing sequence of the first drum revolution, the lateral shift cam **515** shifts the print medium **505** one character position increment to the right (FIG. 3). During the second drum revolution a second eighty character scan and comparison sequence is performed for each C pulse. The second scan sequence causes readout of character storage locations **2**, **9**, **16**, **23**, etc., through **634**. On the third drum revolution, after the paper has again been laterally shifted one character position increment, character storage locations **3**, **10**, **17**, **24**, etc., through **635** are read out for printing.

The above continues until all character storage locations of memory **20** have been read out, compared and printed. At that time the full print line has been completed and the system is ready to print the next line after the print medium has been incremented one line position and has been shifted leftwardly back to home position.

Since the control circuits operate completely independently of the mechanical elements of the print head during the load and justification cycles, the next load cycle can begin the instant the last hammer fire pulse of the print cycle has terminated. Thus, during the two drum revolutions required for paper feeding between print lines, the load and justification cycles can be completely performed so that by the time the paper feed and return-to-home operations are complete another print cycle can begin. This means that the system can print a line of proportionally spaced, justified copy for each 10 revolutions of the print drum. Thus, if the drum rotates at 1800 revolutions per minute, a maximum print rate of 180 lines per minute (3 lines per second) is achieved.

DETAILED DESCRIPTION OF CONTROL CIRCUITS

FIGS. 2*a*–2*d*, when assembled in the manner illustrated in FIG. 8, provide a detailed illustration of the control circuits generally shown in FIG. 1 and described above.

Timing control circuit **100** is shown in FIG. 2*d*. The circuit includes a free-running, fixed frequency oscillator verter **107** provides a train of inverted pulses $\overline{TS}$, for purposes to be described subsequently. A counter **103** advances on each TS pulse and after each group of 80 such pulses the counter supplies an advance pulse to a 12 stage ring circuit **105**. Ring circuit **105** generates at its output the 12 timing signals T1 through T12. Each of the pulses T1 through T12 therefore delineates a time period 80 TS pulses long. Delays built into the ring **105** cause a slight dwell between the time one T pulse goes negative and the next T pulse begins. This is done for reasons which will become apparent. The waveform diagram of FIG. 4 illustrates the relationship between the various timing pulses.

In addition to the circuits just described, the timing control circuit **100** also includes flip-flop circuits **109**, **111**, **113** and **115** which operate to define the three basic operational cycles, load, justify and print. Flip-flop **111** generates at its set output the load control signal LD.

This signal goes positive to initiate a load cycle when flip-flop **111** is set by an output from an AND circuit **117**. AND **117** operates to set the flip-flop on the first T1 pulse occurring after flip-flop **115** has been reset.

Flip-flop **113** generates at its set output a signal JFY which controls the operation of the system in the justify mode. JFY goes positive when flip-flop **113** is set by an output from an AND circuit **119**. AND **119** operates to set flip-flop **113** on the T1 pulse occurring after the resetting of flip-flop **111**.

Flip-flop **115** generates at its set output the print control signal PT which delineates the print cycle. Flip-flop **115** is set by an output from an AND circuit **121** which operates on the first T1 pulse after flip-flop **113** has been reset and after the end-paper-feed signal EPF has been received. When a signal on line **66***a* indicates the completion of the print cycle, flip-flop **115** is reset and generates an initiate-paper-feed signal IPF to cause the print head to index the print medium to the next printline. At the same time IPF is transmitted to the input of AND **117** so that a new load cycle commences on occurrence of the next T1 pulse.

A delay circuit D4 is inserted between the reset output of flip-flop **111** and the input of AND **119** to prevent flip-flop **113** from being set during the same T1 pulse that resets flip-flop **111**. The EPF signal is provided from the print head by conventional means (not shown) which indicate full completion of the print medium indexing operation.

Flip-flop **109** functions to force termination of the load cycle in the unlikely event that the input fails to provide a space character in the justification zone. During any load subcycle when the addressing circuits are driven past character location **632** in the memory, flip-flop **109** sets and at T12 of that subcycle an AND circuit **123** is enabled and transmits a pulse through OR **125** to reset flip-flop **111**. In this way no data character can be lost or only partially printed at the end of a line.

Input circuit **10** (FIG. 2*a*) supplies parallel-by-bit, serial-by-character data signals on six input lines **11**. Each data character includes six binary bits. The repetition rate at which data is presented on input lines **11** must be at least equal to or less than the repetition rate for system print cycles to avoid loss of data.

The gate circuits **400** comprise first and second character gates **401** and **409**, a set of 6 OR circuits **403**, a one-character storage register **405** and an output character gate **407**. OR circuits **403** channel characters from either gate **401** or gate **409** into register **405**. Gate **407** transmits the character stored in register **405** to the buffer memory for storage. An AND circuit **411** opens gate **401** for the duration of each T1 pulse occurring during a load cycle. An AND circuit **425** opens gate **409** for the duration of each T2 pulse occurring during a justify cycle. Register **405** is periodically cleared (reset to the all-zero state) by an output from an OR circuit **417**. OR **417** responds to an output from either an AND circuit **413** or an AND circuit **415** to generate the clear signal. AND **413** is actuated each T4 of a load cycle and AND **415** is actuated each T9 during the justify cycle. An OR circuit **423** operates to open gate **407** under the control of a pair of AND circuits **419** and **421**. AND **419** provides a gating signal each T2 during a load cycle and AND **421** provides the gating signal each T4 during a justify cycle. A set of six lines **406** transmit the character bit data from the output of register **405** to a space detector circuit **201** (FIG. 2*c*) and a weight decoder circuit **219** which are included in the load control circuits **200**.

The load control circuits **200** are shown in FIG. 2*c*. Space detector **201** provides an output signal whenever a space character is stored in register **405**. The output from detector **201** partially conditions an AND circuit **205**. AND **205** is enabled on coincidence of the output from detector **201** with a T1 timing pulse and a set output from a flip-flop **203**. The latter circuit is set in response to a signal from addressing circuit **40** indicating that storage location **560** of memory **20** has been accessed during the load cycle. The output from AND **205** is transmitted to OR **125** (FIG. 2*d*) in the timing control circuits to set flip-flop **111**, terminating a load cycle.

The weight decoder **219** provides an output signal on one of eight output lines, labeled **1** through **8**, in response to each character stored in register **405**. The output from the circuit **219** indicates the width weight of the character. The 1 through 7 outputs from circuit **219** are transmitted through a series of OR circuits **221**, **223**, **225**, **227**, **229** and **231** to the input side of a gate circuit **213**. Due to the action of the OR circuits the number of input lines to gate **213** which are activated is inversely proportional to the weight of the character being decoded. For example, if the 1 output line goes positive signifying a character which is one position increment wide, all seven input lines to gate **213** are activated since the signal on the one output line also feeds through the six OR circuits **221** through **231** to activate the six other inputs to gate **213**. Similarly, if the 2 output lines goes positive, all input lines to gate **213** except the right-hand input line are activated. If the seven output line goes positive, only the left-hand input to gate **213** is activated.

An AND circuit **211** opens gate **213** each **T3** during the load cycle to transmit the outputs from the weight decoder into an **8** stage shift register **215**. At **T1** a reset line **214** was activated to set a one bit into each stage of the shift register and when gate **213** opens, each activated output therefrom enters a zero bit into the corresponding stage of the shift register. Therefore, for a character having a weight of one, a zero bit is set into each of the first seven stages of the shift register and the last (right hand) stage is left at one. Similarly, for a character having a weight of seven, a zero bit is set into the first (left-hand) stage of the shift register and the remaining seven stages are left set to one.

After the shift register has been loaded with the weight count at T3, an AND circuit **209** generates eight shift pulses, one during each of the timing pulses T4 through T11. An OR circuit **207** feeds the timing pulses to AND **209** and the output from the latter circuit conditions an AND circuit **217** and drives the shift register through a delay circuit **D5**. The delay of D5 is equal to substantially the full length of a T pulse so that each of the timing pulses **T4** through **T11** causes the data in register **215** to shift one stage to the right at the end of the time pulse. AND **217** receives as its second input the output from the last stage of the shift register. Therefore, so long as a one bit remains stored in the last stage of the shift register the output of AND **217** goes positive in response to each of the shift pulses. Each output from AND **217** is used to advance the addressing circuit to the next higher storage position in memory **20**. Shift register **215** therefore acts to advance the addressing circuit a number of storage positions equal to the width weight of the character stored in register **405**.

FIG. 2*b* shows the construction of the buffer memory **20**. It includes a 640 character magnetic core memory matrix **21**, a set of write-read drivers **27** and a set of read-only drivers **29**. Each character storage location in matrix **21** includes at least six magnetic cores for storing the six bits of a character. Character locations are accessed one at a time by the drivers **27** and **29** under control of either the addressing circuits **40** or the read-out control circuits **60**, respectively. The write-read driver circuit **27** has 640 output lines, each one of which links one character storage location in matrix **21**. When the drivers are conditioned by an output from an OR circuit **23** for writing data into the memory, the activated driver line provides a select current to the six cores of its associated storage location so that when gate **407** is opened the character presented to the memory matrix is written into the accessed storage location. The write drivers are ac-

tivated by OR **23** during the full load cycle and during each T4 time of the justify cycle.

When the read drivers of circuit **27** are activated by an output from an AND circuit **25** each T2 time of a justify cycle, each energized driver output line supplies a full switching current to the six cores of the associated storage location so that the data stored in the location is destructively read out on the six memory output lines **22**. The character on lines **22** is fed back, via lines **24** to the inputs of gate **409** (FIG. 2*a*).

The read-only driver circuits **29** also have 640 output lines linked one each to the storage positions of matrix **21**. These drivers are conditioned during a print cycle to sequentially read the characters out of memory nondestructively. During this read-out operation gate **50** is opened to pass the characters to the print circuits **70**.

The addressing circuit **40** is shown in FIG. 2*d* and comprises a 640 stage bidirectional ring circuit **41**, three AND circuits **43**, **47** and **49** and an OR circuit **45**. Each pulse applied to the ADV input of ring **41** causes it to advance one position toward 640 and each input pulse applied to the DEC input terminal causes the ring to be decremented one position towards stage one. The 640 output lines from ring **41** are energized one at a time to activate one of the write or read drivers in circuit **27**. AND **43** generates a pulse each T1 time of a print cycle to reset ring **41** to stage one. OR **45** pulses the ADV input of ring **41** in response to pulses from AND **217** whereby ring **41** is caused to advance. OR **45** also receives advancing pulses from AND **49** which supplies one advancing pulse each TS time during T3 of a justify cycle so long as the output of a zero detecting circuit **317** in the justification control circuits **300** is negative (signifying the absence of a zero output from a counter **301**). AND **47** supplies a decrementing pulse to the ring **41** for each TS pulse occurring during T2 of a justify cycle when flip-flop **235** in the load control circuits is in its reset state.

The read-out control circuit **60** is shown in FIG. 2*a* and functions to control the operation of the addressing circuit **30** to cause nondestructive readout of the data in memory matrix **21** during a print cycle. An AND circuit **67** is enabled during the first P1 pulse of a print cycle after the presence of the print medium in home position has been indicated by an H signal. The output from AND **67** sets a flip-flop **61** and the set output therefrom partially conditions AND circuits **65**, **68** and **69**. AND **65** generates an output that sets a flip-flop **63** in response to the first $\overline{TS}$ signal that occurs after the setting of flip-flop **61** and after passage of a delay period D1 following the occurrence of a character pulse C. The set output of flip-flop **63** partially conditions AND **69**, conditions read drivers **29** and opens gate **50**.

AND **69** responds to each TS pulse occurring after the setting of flip-flop **63** to transmit an advance pulse to a ring circuit **31** in the addressing circuits **30**. AND **68** responds at the time of the setting of flip-flop **61** to transmit an advance pulse to a ring **33**. The first P1 signal during the print cycle causes ring **33** to be advanced from its 8 state to its 1 state and each P1 pulse occurring thereafter during the print cycle advances the ring one stage toward the 8 state.

A delay circuit D2 generates a pulse to reset flip-flop **63** at a time following the occurrence of the C pulse. The delay provided by circuit D2 is exactly timed to permit the ring **31** to advance from stage **1** to stage **80**. When the latter stage is reached, the flip-flop **63** is reset and AND **69** is disabled so that the ring remains at stage **80**. The read drivers **29** and gate **50** are also disabled by the resetting of flip-flop **63**.

A flip-flop **62** is provided to terminate the print cycle after the last C pulse following the eighth P1 pulse of the cycle. Each C pulse sets flip-flop **62**. During the eighth print drum revolution of the cycle the addressing circuits **30** provide a pulse to reset flip-flop **62** following each C pulse. A delay circuit D3 feeds this resetting pulse to an AND **64**, delaying it a length of time greater than the time between the resetting pulse and the following C pulse. After the last C pulse flip-flop **62** remains in its reset state so that after the D3 delay period AND **64** pulses a single-shot multivibrator **66** and the output therefrom resets flip-flop **61**, causes drivers **29** to clear memory matrix **21** and resets flip-flop **115** via line **66***a* to terminate the print cycle and initiate paper feeding.

Addressing circuit **30**, shown in FIGS. 2*a* and 2*b*, includes the ring circuits **31** and **33** and further comprises 640 AND circuits **35**. Ring circuit **31** has eighty stages and therefore has eighty output lines, only the first and last of which are shown. Ring **33** has eight stages and eight output lines. The AND circuits **35** are connected in eighty groups of eight circuits each. Only the first and last circuit of the first AND group (labeled **1** and **8**, respectively) and the first and last circuit of the last AND group (labeled **633** and **640**, respectively) are shown. The first stage output line from ring **33** is connected to the first AND's of each of the eighty groups. The second stage output line from ring **33** is connected to the second AND of each group, etc. on up to the eighth output line from ring **33** which is connected to the eighth circuit of each of the eighty groups of AND's **35**.

The second input to each of the AND's **35** is supplied from ring **31**. The stage **1** output line is connected to all eight AND's of the first group. Similarly, the stage **2** output line is connected to all eight of the AND circuits of the second group, etc. on up to the eightieth output line which is connected to all AND circuits in the eightieth group, as shown.

Thus, on the first revolution of the print drum during a print cycle, AND **69** steps ring **31** to sequentially energize AND circuits **35** in the pattern **1**, **9**, **17**, **25**, etc., up to **633**. At the beginning of the second drum revolution AND **68** steps ring **33** to its second output stage so that when AND **69** drives ring **31** through the next eighty step sequence AND circuits **2**, **10**, **18**, **26**, etc., on up to **634** are activated. This continues for eight revolutions of the drum until, at the end of the eighth revolution the output from the 640th AND circuit **35** resets flip-flop **62** and triggers the print cycle termination sequence, explained above.

Therefore, during a print cycle the AND circuits **35** control the read-only drivers **29** to nondestructively read out the data stored in matrix **21** in eight basic scan patterns. Each scan pattern reads out every eighth storage location in the memory. During each revolution of the drum a basic scan pattern is repeated a number of times equal to the number of C pulses generated during the drum revolution.

The print circuits **70** are shown in FIG. 2*b*. Each character read from the memory through gate **50** is fed by lines **51** to the input of a comparator circuit **71**. Also supplied to the input of comparator **71** is the six bit output from a character code generator **72**. The code generator supplies at its output a sequence of character codes defining the order in which character rows on the print drum come up for printing. Each C pulse generated by the print drum steps the code generator to the next character. A set of eighty flip-flops **74–1** through **74–80** are provided to temporarily store the results of each comparison performed during a scan sequence. The C pulse is fed to reset each of these flip-flips just prior to the time that the comparisons take place. As each memory character is presented through gate **50** to the input of the comparator, one of the eighty AND circuits **73–1** through **73–80** is partially conditioned by the activated output line from ring **31**. If the memory character matches the output of code generator **72**, the partially enabled AND circuit **73** is activated to set its corresponding flip-flop **74**. As soon as the eighty character scan sequence is completed, a delay circuit D6, which was activated by the C pulse, generates an output which activates all of the eighty AND circuits **75–1** through **75–80** which are

connected to flip-flops **74** which were set during the compare sequence. The outputs from the activated AND circuits **75** cause the corresponding hammer driver circuits **HD1** through **HD80** to drive their corresponding print hammers. The hammer driver circuits HD1–HD80 correspond to those shown in FIG. 3.

The justification control circuits **300** are shown in detail in FIG. 2*d*. A seven stage binary counter **301**, herein called a shift counter, and a matching seven stage binary counter **303**, herein called a pad counter, operate to provide the basic control of memory **20** during the justify cycle. At the beginning of each load cycle, the count in each of these counters represents a count of eighty. This count was set into the counters at the end of the previous justify cycle by a pulse from an AND circuit **321**.

During the load cycle an AND **315** provides an output pulse each time ring **41** is advanced a storage location past location **559**. Each output from AND **315** is fed by OR circuits **309** and **313** to decrement the pad counter and the shift counter, respectively, one count. As previously described, the count of these counters gives an indication of the distance, in terms of character position increments, between the right hand margin of the line and the last character loaded into the memory.

Therefore, when the justify cycle begins, the count represented at the outputs of the counters **301** and **303** represents the number of unused positions increments in the line which have to be filled in during justification. The output generated by AND **49** of the addressing circuit **40** is also connected to the input of OR **313** so that each time ring **41** is advanced during a justify cycle, counter **301** is decremented toward zero. Whenever the count in counter **301** reaches zero, zero detector **317** provides a positive signal to the input of inverter **319** and partially conditions AND circuit **321**.

An AND circuit **307** operates each **T5** time during a justify cycle to feed a decrementing pulse through OR **309** to counter **303**. At **T8** time of the justify cycle, an AND **311** opens gate circuit **305** to cause the state of counter **301** to be set equal to the state of counter **303**.

OPERATION-LOAD CYCLE

The load cycle comprises a sequence of subcycles wherein one character from input circuit **10** is loaded into the buffer memory each subcycle. Each subcycle is made up of one sequence of the timing pulses **T1** through **T12**. The following table summarizes the operation of the control circuits for one load subcycle:

T1—Strobe gate **401** and load input character into register **405**.
Reset shift register **215**.
Terminate load cycle by resetting flip-flop **111** if output from space detector **201** is positive and flip-flop **203** is set.
T2—Load character from register **405** into memory **21** at location accessed.
T3—Set shift register **215** in accordance with weight of character.
T4—Clear register **405**.
T5
T6
T7—Advance ring **41** a number of character positions equal to character weight.
T8
T9
T10
T11
T12—Terminate load cycle by resetting flip-flop **111** if flip-flop **109** is set.
Repeat **T1**–**T12** if flip-flop **111** is still set.

At **T1** time AND **411** strobes gate **401** and enters an input character into register **405**. Also at this time, reset line **214** goes positive to enter a one bit into all eight stages of shift register **215**. AND **205** is strobed by timing pulse **T1** and if space detector **201** is indicating the presence of a space character in register **205** and flip-flop **203** is set, indicating that ring **41** has advanced to character position **560** or beyond, AND **205** produces the output pulse which feeds through OR **125** to reset flip-flop **111** and thus terminate the load cycle and throw the system into a justify cycle. However, if AND **205** does not generate a pulse the load subcycle continues.

At **T2** time AND **419** opens gate **407** and enters the character from register **405** into the storage location of memory matrix **21** which is then being accessed by ring **41** and the write driver circuits **27**.

At **T3** time AND **211** opens gate **213** and sets a number of zero bits into shift register **215** in accordance with the width weight of the character in register **405**.

At **T4** time AND **413** generates a pulse which is fed through OR **417** to clear register **405**. Also, the first shift pulse is fed to shift register **215** to generate the first pulse to advance ring **41**. Thereafter, during **T5** through **T11**, seven more shift pulses are fed by AND **209** to the shift register to complete the advancement of ring **41** to a new storage location in accordance with the width of the last entered data character.

At **T12** time AND **123** generates an output pulse if flip-flop **109** had been set during the preceding ring advancement operation. The output from AND **123** is fed through OR **125** to reset flip-flop **111** and terminate the load cycle.

When the next **T1** pulse occurs, gate **401** is again strobed and the next input character is loaded into memory and the cycle repeats itself as described above and continues to repeat until an output is generated either by AND **205** or by AND **123** to terminate the load cycle. It is to be noted that during a load cycle the first input character is always stored in memory location **1** in matrix **21** and each succeeding character is stored in a storage location determined by the width of the last-loaded character.

This aspect of the operation of the system is illustrated diagrammatically in FIG. 5. As there shown, the first portion of a print line, beginning with the phrase "With many," is illustrated with the width of each character (in character position increments) indicated below the character and the memory location into which the character is loaded shown above the character. As noted from the diagram, a space character is five position increments wide.

OPERATION-JUSTIFY CYCLE

As in the case of the load cycle, the justify cycle is made up of a series of subcycles each subcycle being defined by one sequence of the timing pulses **T1** through **T12**. As previously described, termination of the load cycle causes flip-flop **111** to reset and the justify cycle commences at the beginning of the ensuing **T1** pulse when AND **119** sets flip-flop **113**. The operations occurring during one justify subcycle are summarized as follows:

T1—Reset flip-flop **235**.
T2—Decrement ring **41** until last-loaded character in memory **21** is detected and loaded into register **405**.
T3—Advance ring **41** and simultaneously decrement counter **301** until counter **301** reaches zero.
T4—Enter the character from register **405** into memory at the accessed location.
T5—Decrement counter **303** by a count of one.
T6—- - -.
T7—- - -.
T8—Set counter **301** with a count equal to the count in counter **303**.
T9—Clear register **405**.
T10—- - -.
T11—- - -.
T12—Reset flip-flop **113** to terminate justify cycle if output of zero detector **317** is positive.
Repeat **T1**–**T12** if flip-flop **113** is still set.

At T1 time or the first justify subcycle flip-flop **235** is reset. This conditions AND **47** to transmit decrementing pulses to ring **41** during the ensuing T2 time period. Each time ring **41** is decremented, it steps the read driver circuits **27** one storage location in the reverse direction and causes the character in that storage location to be destructively read out and fed via lines **24** and gate circuit **409** into the character register **405**.

Since all data characters, including the space character, have at least one "one" bit and since at the end of the previous print cycle the output from single-shot **66** was used to simultaneously actuate all read drivers **29** to clear memory **21** (leaving all character locations in an all-zero condition), weight decoder **219** is used to detect the presence or absence of a data character in register **405** since the decoder does not generate an output in response to all-zero character. Thus, when the first character is encountered in the reverse accessing operation, the output of OR **233** (FIG. 2*c*) goes positive to set flip-flop **235** whereby the enabling input to AND **47** is removed. This arrests the reverse accessing operation at the storage location of the first encountered character and leaves that character stored in register **405**.

At T3 time the TS pulses feed through AND **49** to step ring **41** forward. The output from AND **49** also feeds through OR **313** to decrement the shift counter so that when that counter reaches a count of zero detector **317** produces a positive output signal whereby inverter **319** deconditions AND **49** to arrest the forward accessing operation. At this point ring **41** is accessing the storage location in the memory representing the extreme right hand end of the print line insofar as the character in register **405** is concerned. This is due to the fact that at the beginning of the subcycle the count in counter **301** represented the exact number of position increments from the right-hand edge of the last-loaded character to the right-hand margin of the print line. When ring **41** was stepped in the reverse direction during T2 to locate the last-loaded character, it was driven backwards a number of character locations equal to the width of that character. Thus, when ring **41** was stepped forward during T3 under control of counter **301** it was arrested at a storage location removed from storage location **640** by the exact number of locations required to compensate for the width of the character.

At T4 time AND **421** opens gate **407** and the character in register **405** is entered into memory.

At T5 AND **307** decrements the count in the pad counter by one. At T6 and T7 no operations are performed. At T8 AND **311** opens gate **305** to load the shift counter with a count equal to that appearing at the output of the pad counter. At T9 AND **415** clears register **405** and at T12 AND **321** samples the output from zero detector **317** and if that output is positive the output from AND **321** resets flip-flop **113** to terminate the justify cycle. If the output of the zero detector is not positive the justify cycle commences a new subcycle identical to that just described.

Since the pad counter is decremented by one at T5, the next-to-last character in the print line, when it is shifted under control of the shift counter during the next subcycle, is shifted a number of storage locations to the right which is one less than the number which the last character was shifted. This inserts an extra position increment of space between the two characters. Similarly, if the justify cycle is not terminated at the end of the second subcycle, the third-from-last character in the line is shifted to a new storage location so that an additional increment of space is inserted between it and the next-to-last character. This operation of padding position increments into the line continues, of course, until the pad counter is decremented to zero whereupon the justify cycle is terminated.

FIG. 7 illustrates the operation of the justify cycle in diagrammatic form. For purposes of simplification, FIG. 7 assumes that a full print line constitutes only 110 print position increments rather than 640 as in the case of the system of the preferred embodiment. Also, in FIG. 7 the justification zone begins at position **101** so that at the beginning of the justify cycle the pad counter is preset to a count of 10.

The first line shows the characters of the print line as distributed in the buffer memory at the completion of the load cycle. The last character of the line, "1," is stored in character location **101** meaning that when the load cycle terminated the ring **41** was being held at storage location **103** (since the width of the character "1" is two) and the count in the shift counter and pad counter is seven.

As shown in the second line of FIG. 7, the last character of the line is shifted during the first justify subcycle seven increments to the right from storage location **101** to storage location **108**. The count in the pad counter is reduced by one so that at the beginning of the second justify subcycle (third line) it is set to a count of six. Thus during the second subcycle the next-to-last character of the print line is shifted six positions from location **99** to location **105** and the count in the pad counter is reduced to five.

During the third subcycle the character "a" is moved five positions from location **94** to location **99** and during the fourth subcycle the space character is moved four positions from location **89** to location **93** etc., until during the seventh subcycle (bottom line of FIG. 7) the "f" is moved one position from location **76** to location **77**. The justify cycle then terminates at the end of the seventh subcycle since the pad counter is reduced to zero and flip-flop **113** is reset at T12 of the subcycle.

OPERATION-PRINT CYCLE

The basic timing for the print cycle has to be supplied from the print head mechanism due to the natural cyclic variations inherent in the operation of the mechanism. The only timing pulses from the circuit **100** which are employed during a print cycle are the TS and the $\overline{TS}$ pulses which are used to control the scanning of the memory between each C pulse. For this reason, it is apparent that the repetition rate of TS must be high enough to permit at least 80 TS cycles between each C pulse. The print cycle consists of eight subcycles, one for each revolution of the print drum. After flip-flop **61** has been set to initiate the print cycle each print subcycle proceeds as follows:

P1—Advance ring **33**.

C1—Advance code generator and reset all flip-flops **74**.

$C+D1$—Set flip-flop **63** on $\overline{TS}$ to condition AND **69** and read drivers **29** and to open gate **50** whereby during the next 80 TS pulses ring **31** reads out the 80 storage locations determined by ring **33** and flip-flops **74** are set each time an equal comparison is detected by comparator **71**.

$C+D2$—Reset flip-flop **63** to arrest ring **31** at position **80**.

$C+D6$—Actuate AND circuits **75** to energize hammer driver circuits.

Repeat C through $C+D6$ for each remaining C pulse during the drum revolution.

During the eighth print subcycle the 640th AND circuit **35** is actuated at the end of each scan cycle of ring **31**. At the end of the last scan cycle, after the last C pulse has occurred, the output from single-shot **66** on line **66***a* terminates the print cycle by resetting the flip-flop **115** and the print control flip-flop **61**. The pulse on line **66***a* further operates to clear memory **21**.

FIG. 6 diagrammatically illustrates the operation of one print cycle wherein the first six print hammers are shown printing the phrase "With many." The characters of the phrase are stored in the memory at the storage locations indicated in FIG. 5. During the first print subcycle and the first revolution of the print drum the rings **31** and **33** operate to nondestructively read the data out of the storage locations **1, 9, 17, 25, 33, 41, 49** and **57**. As indicated by the circles in FIG. 6 the data characters "W," "i" and "m" appear in the three storage locations **1, 9,** and **25**. Thus, during the first revolution of the print

drum hammers 1, 2, and 4 are activated by the associated hammer driver circuits to print the characters "W," "j" and "m." Thereafter, the stepping cam 515 (FIG. 3) operates to shift the print medium one position increment to the left so that the characters assume a position relative to the print hammers as depicted in line 2 of FIG. 6.

During the second print subcycle character locations 2, 10, 18, 27, 34, 42, 50 and 58 are read out for comparison and as noted by the circle in FIG. 6 the "y," which is stored in location 42, is the only data character encountered. Hammer number 6 is actuated during the second subcycle to print the "y."

The print medium is stepped one more position to the left and in the third cycle hammer 2 fires to print the "t." In the fourth subcycle no printing occurs since no data characters are encountered in the memory scan. In the fifth subcycle the "n" is printed by hammer 5 and in the seventh and eighth subcycles hammers 2 and 4 are fired to print the "h" and "a," respectively. After the eighth subcycle the full line has been printed and the print cycle is terminated.

It will be appreciated that various changes in the form and details of the above-described preferred embodiment may be effected by persons of ordinary skill without departing from the true spirit and scope of the invention.

What is claimed is:

1. In a device for printing characters in marginally justified lines having a predetermined number of character position increments, the combination comprising:
- a character memory having a number of character storage locations equal to said predetermined number of character position increments;
- means for loading into said memory characters representing a line to be printed, each said character being stored in the storage location corresponding to the position increment at which it is to be printed;
- counter means for indicating the number of storage locations between the location in which the last-loaded character is stored and the last location in said memory;
- shifting means operable after storage of said last-loaded character to shift said characters toward said last location by a number of locations equal to the count in said counter means, said characters being shifted in an order reverse to the order in which they were loaded; and
- control means for altering the count in said counter means by a predetermined amount after each said character is shifted.

2. The apparatus set forth in claim 1 wherein said control means comprises:
- means for decrementing the count in said counter means by one after each said character is shifted.

3. The apparatus set forth in claim 2, further comprising:
- means for terminating the operation of said shifting means when the count in said counting means is reduced to zero.

4. The apparatus set forth in claim 1 wherein said loading means comprises:
- means for accessing said storage locations for sequential entry of characters therein;
- means for setting said counter means to a predetermined count prior to character loading; and
- means for altering the count in said counter means by a predetermined amount each time one of a predetermined plurality of storage locations is accessed.

5. The apparatus set forth in claim 4 wherein said predetermined plurality of storage locations comprises the storage locations corresponding to a row of adjacent character position increments extending leftwardly from the right-hand end of the print line.

6. In a device for printing a line of proportionally spaced characters on a print medium, said line having a predetermined number of character position increments, the combination comprising:
- a memory having a character storage location for each character position increment in said line;
- means for sequentially accessing said storage locations for entry of characters therein;
- means for sequentially presenting the characters of a line for storage in said memory;
- decoding means for generating a signal output representative of the width of each character presented; and
- means responsive to said decoding means after storage of each presented character for advancing said accessing means a number of storage locations proportional to the width of the last-stored character.

7. The apparatus set forth in claim 6, further comprising:
- means for setting said accessing means to access the first storage location in said memory prior to presentation of the first character of said line.

8. In a device for printing a line of proportionally spaced characters on a print medium, the combination comprising:
- a character memory;
- loading means for storing the characters representing a print line in said memory at character locations determined by the width of said characters, said loading means comprising: first means for sequentially accessing the storage locations of said memory beginning with a first location thereof, means for presenting the first character of said line to said memory when said first location is accessed whereby said first character is stored in said first location, decoding means for generating a signal output representative of the width of each character presented to said memory, and means responsive to said decoding means after storage of each presented character for advancing said accessing means a number of storage locations proportional to the width of the last stored character;
- second means for sequentially accessing the character locations of said memory;
- means for printing each stored character as it is accessed by said second accessing means;
- means for shifting said print medium relative to said print means in the direction of said print line whereby successive locations along said line are brought into registration with said print means; and
- means for operating said second accessing means in timed relation with said shifting means whereby said characters are printed at locations along said line dependent upon their respective storage locations in said memory.

References Cited

UNITED STATES PATENTS

2,806,575	9/1957	Higonnet et al.	197—84
3,174,427	3/1965	Taylor	197—1 X
3,354,816	11/1967	Giannuzzi	101—93
3,404,766	10/1968	Castle et al.	197—19 X
3,420,164	1/1969	Lee	101—93

WILLIAM B. PENN, Primary Examiner

U.S. Cl. X.R.

197—84, 133; 276—28; 340—172.5