United States Patent [19]

Blejwas, Jr. et al.

[11] **3,930,250**

[45] **Dec. 30, 1975**

[54] **SYNCHRONIZING SYSTEM FOR REFRESH MEMORY**

[75] Inventors: **Walter T. Blejwas, Jr.**, Somerville, N.J.; **Peter D. Dickinson**, Monte Sereno, Calif.

[73] Assignee: **Vydec, Inc.**, Whippany, N.J.

[22] Filed: **May 6, 1974**

[21] Appl. No.: **467,270**

[52] **U.S. Cl.** ..... **340/324 A**; 328/72

[51] **Int. Cl.²** **G06F 3/14**

[58] **Field of Search** ..... 340/324 A; 328/72, 229

[56] **References Cited**

UNITED STATES PATENTS

3,548,402	12/1970	Schumacher	340/324 A
3,598,911	8/1971	Helbig	340/324 AD
3,742,484	6/1973	Rosenthal	340/324 A
3,786,482	1/1974	Puckett et al.	340/324 A

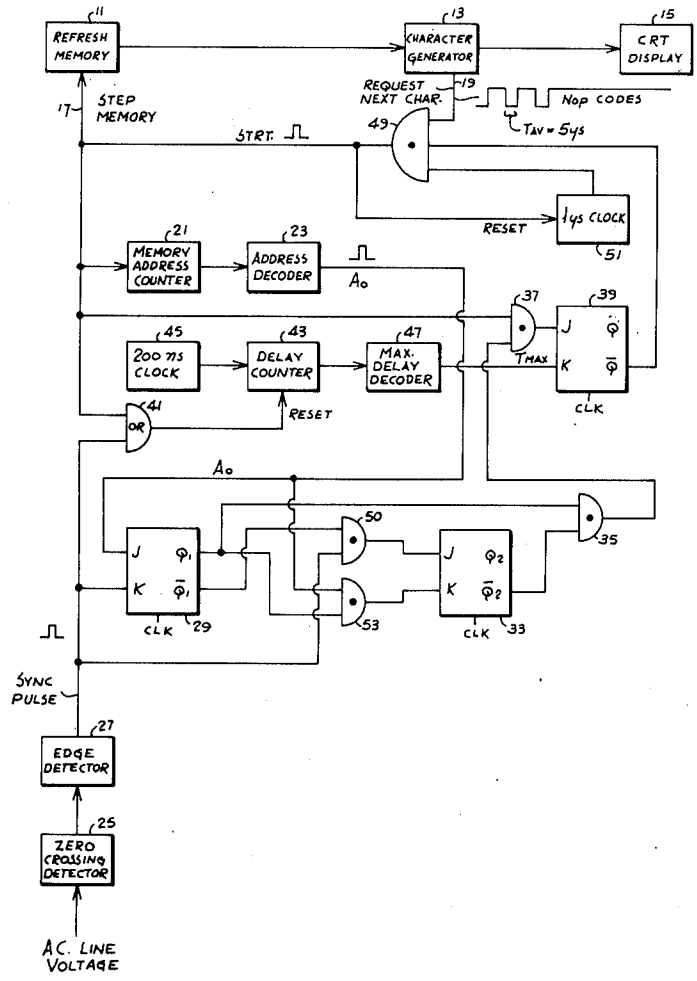
Primary Examiner—David L. Trafton

Attorney, Agent, or Firm—Stephen P. Fox

[57] **ABSTRACT**

A recirculating refresh memory comprising dynamic shift registers provides character codes in sequence to a character generator, which in turn draws the characters on a cathode ray tube display screen. The memory output is stepped from one character to the next at either fixed or variable time intervals, depending on the position of the memory in its refresh cycle and the occurrence of sync pulses from the AC line. The sync pulses may be referenced to any point in the memory and are allowed to move relative to text stored in memory.

10 Claims, 6 Drawing Figures



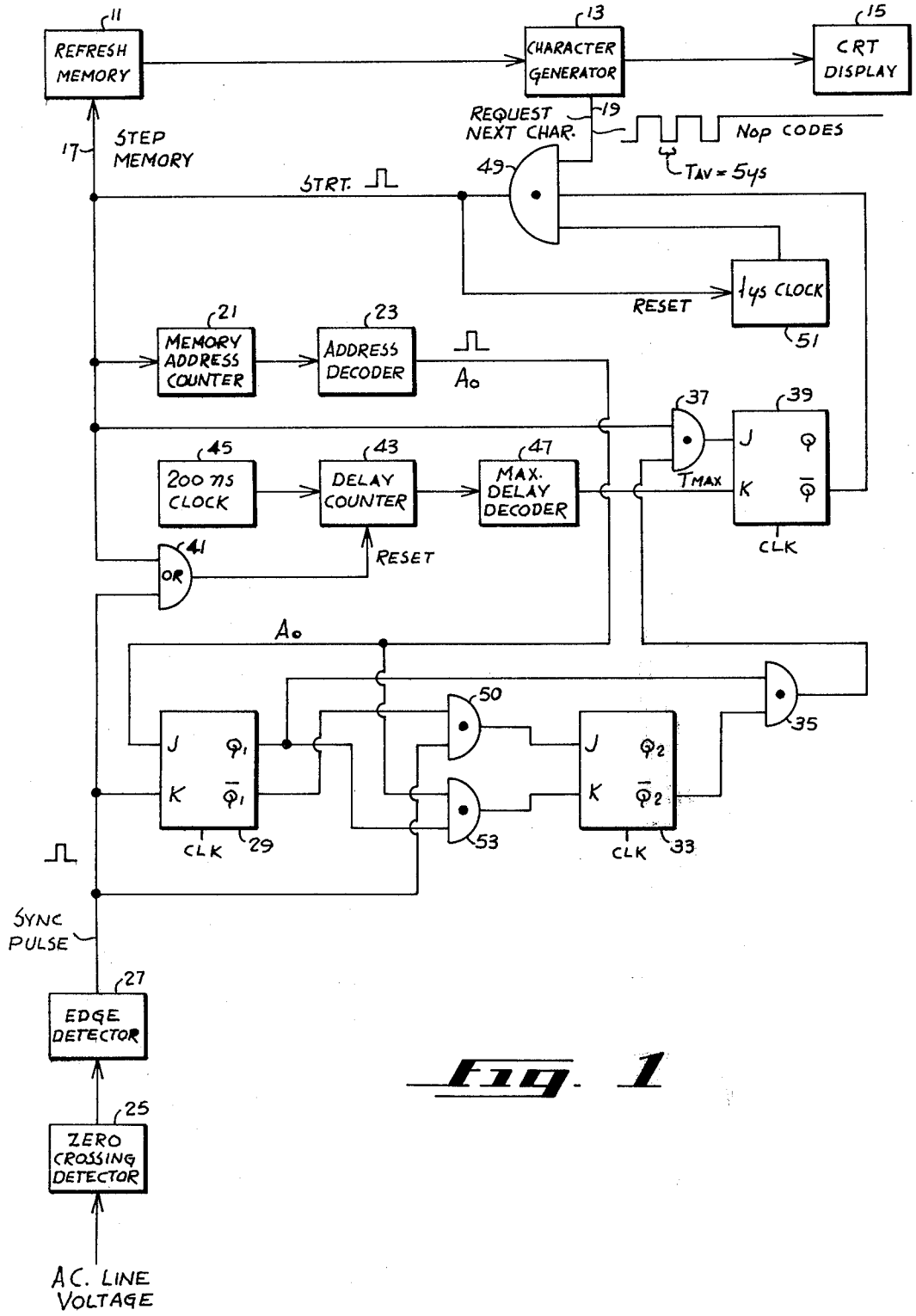

11
REFRESH MEMORY
13
CHARACTER GENERATOR
15
CRT DISPLAY
REQUEST NEXT CHAR.
19
Nop CODES
TAV = 5μs
17
STEP MEMORY
49
STRT.
RESET
1μs CLOCK
51
21
MEMORY ADDRESS COUNTER
23
ADDRESS DECODER
A0
37
39
J
Q
K
Q̄
TMAX
CLK
45
200 ns CLOCK
43
DELAY COUNTER
47
MAX. DELAY DECODER
RESET
41
OR
A0
50
35
J
Q1
K
Q̄1
CLK
29
53
J
Q2
K
Q̄2
CLK
33
SYNC PULSE
27
EDGE DETECTOR
25
ZERO CROSSING DETECTOR
AC. LINE VOLTAGE

fig. 1

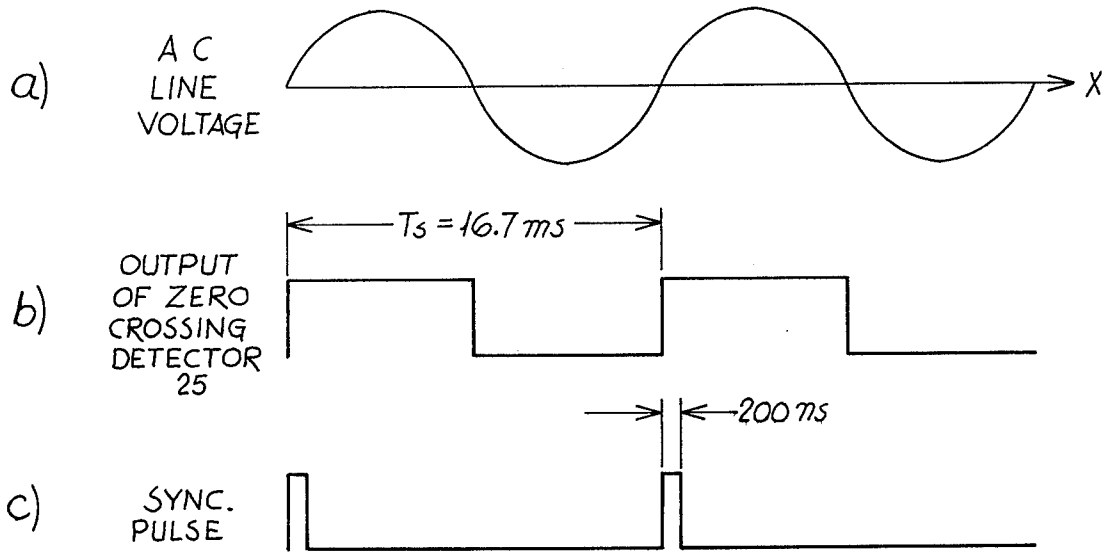

a)
A C LINE VOLTAGE
X
b)
OUTPUT OF ZERO CROSSING DETECTOR 25
Ts = 16.7 ms
200 ms
c)
SYNC. PULSE

Fig. 2

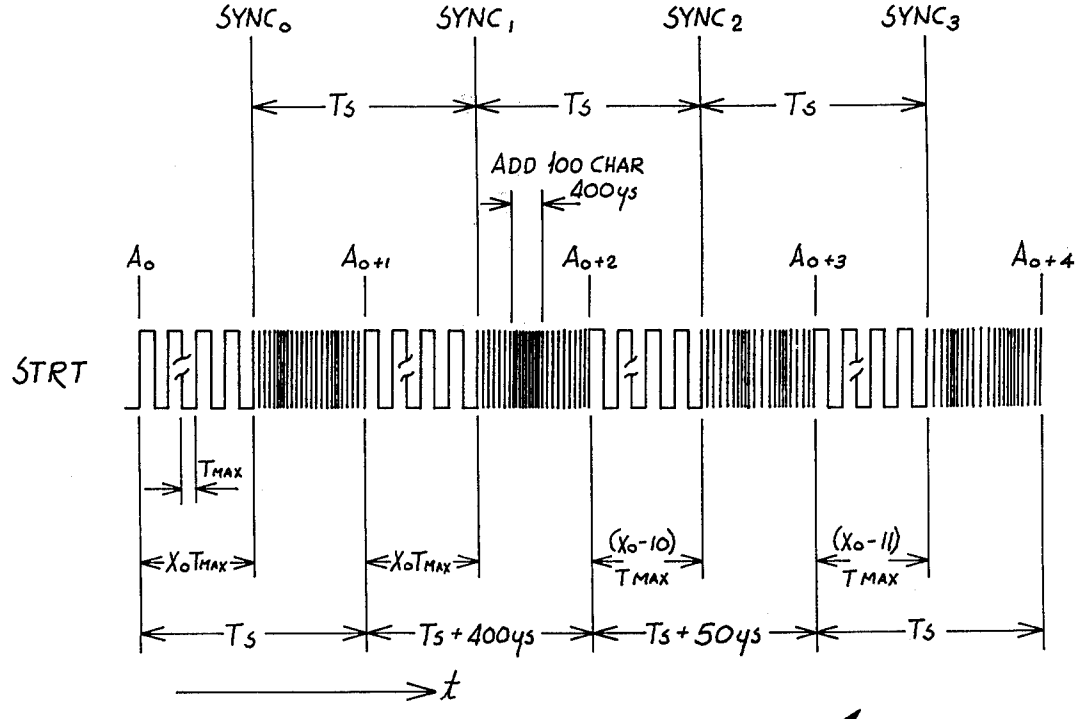
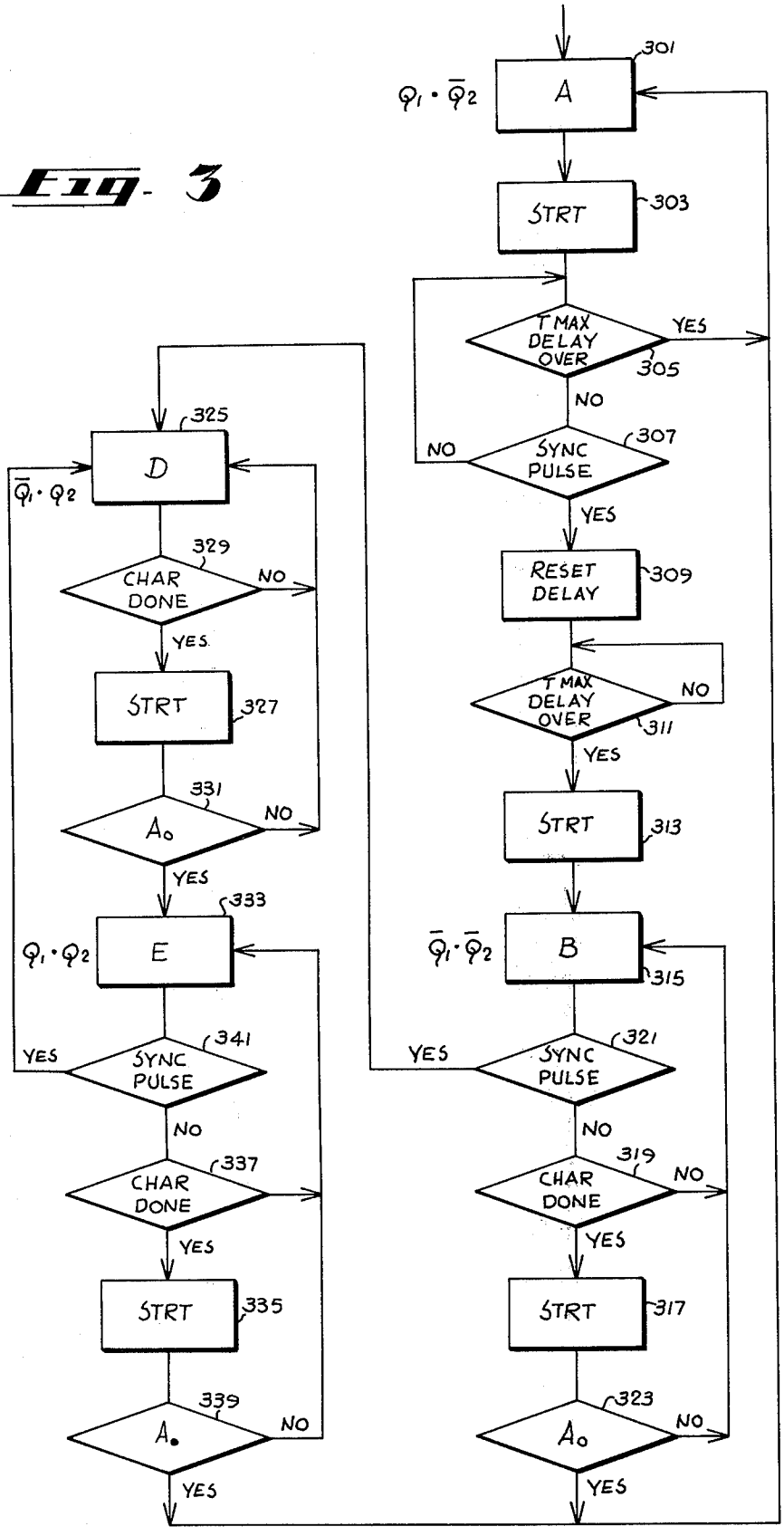

SYNC0
SYNC1
SYNC2
SYNC3
Ts
Ts
Ts
ADD 100 CHAR
400μs
A0
A0+1
A0+2
A0+3
A0+4
STRT
TMAX
X0TMAX
X0TMAX
(X0-10)
TMAX
(X0-11)
TMAX
Ts
Ts + 400μs
Ts + 50μs
Ts
t

Fig. 4

Fig. 3


301
$Q_1 \cdot \bar{Q}_2$
A
STRT
303
T MAX DELAY OVER
YES
305
NO
NO
SYNC PULSE
307
YES
RESET DELAY
309
T MAX DELAY OVER
NO
311
YES
STRT
313
$\bar{Q}_1 \cdot \bar{Q}_2$
B
315
YES
SYNC PULSE
321
NO
CHAR DONE
319
NO
YES
STRT
317
A_0
323
NO
YES
325
$\bar{Q}_1 \cdot Q_2$
D
CHAR DONE
329
NO
YES
STRT
327
A_0
331
NO
YES
$Q_1 \cdot Q_2$
E
333
YES
SYNC PULSE
341
NO
CHAR DONE
337
YES
STRT
335
A_0
339
NO
YES

SYNCHRONIZING SYSTEM FOR REFRESH MEMORY

CROSS-REFERENCES TO RELATED PATENTS AND APPLICATIONS

The present application relates to U.S. Pat. No. 3,660,833, and copending U.S. Pat. application Ser. No. 464,990, filed Apr. 29, 1974 in the names of Patrick P. de Cavaignac and Peter D. Dickinson.

BACKGROUND OF THE INVENTION

The present invention relates generally to a system in which information contained in a recirculating refresh memory is repeatedly applied to a display device such as a cathode ray tube. More particularly, the invention relates to a method and apparatus for synchronizing the refresh memory with periodically recurring synchronization signals, derived from an AC power line for example.

The display of a page of alphanumeric characters on a CRT screen is accomplished by moving an electron beam across the phosphor surface inside the tube. The beam is magnetically or electrostatically directed to each character position on the screen and then controlled to produce the desired character shapes. The beam energizes the phosphor which in turn emits light, but the light diminishes rapidly and the characters fade in a fraction of a second. To sustain the image, the characters on the screen must be drawn repetitively, thereby to "refresh" the display. In order to prevent the display from flickering to the eye of an observer, the refresh should occur at least 40 times per second.

Since the electron beam is directed by magnetic or electrostatic fields, it can be misdirected if influenced by unwanted interference fields. Such interfering fields are most commonly caused by the alternating line current which supplies the electrical power to the equipment. The disturbances are alternating ones. They change cyclically at the constant line frequency of 60 cycles per second. Thus character jitter will occur at the rate of 60 times per second.

If a page of characters on the CRT screen is refreshed at a rate different than the AC line frequency, say 40 times per second, the character signals and the disturbance signals will shift with respect to each other with each refresh. The effect is that the display will appear to wobble at a rate equal to the difference between the line frequency and the refresh frequency, namely 20 cycles per second. A wobbling display soon becomes unpleasant and annoying to an observer. Elimination of most of the wobble can be achieved by making the refresh rate the same as the line frequency, i.e., 60 cycles per second. In this case, the display refresh is synchronized to the power line frequency. Typically, each refresh cycle is initiated by a trigger pulse derived from the power line.

Most schemes for drawing characters on a CRT screen have the common characteristic that each character is allotted the same drawing time as all others, regardless of character complexity. The average drawing time per character is largely a function of the most difficult and time-consuming characters to draw. Thus, a complex number 8 will be given the same drawing time as a period (.). The refresh memory is stepped from character to character at a constant speed. This type of system limits the total number of characters that can be drawn during each refresh cycle, which is

1/60 of a second long. One way to increase the number of characters drawn is to lengthen the time of each refresh cycle to 1/40 of a second for example, but the consequence is a wobbly display, as described above.

Another way to increase the number of characters displayed is to allot different drawing times to the characters depending on their complexity, as described in U.S. Pat. No. 3,660,833, issued May 2, 1972 and assigned to the same assignee as the present invention. With this technique, the time given to draw an 8 is much greater than that for a period. The refresh memory is stepped at a slower speed for complex characters than it is for simple characters. The problem with this technique is that memory stepping is not constant. The refresh memory does not operate at a constant 60 cycle per second refresh rate, and synchronism with the power line may be lost and the display may wobble.

One way to preserve synchronism when memory clocking is not constant is to structure the refresh memory with a sequential array of static shift registers. These devices may be clocked at any rate from zero to their design maximum. Thus, when the drawing of a page of text on the CRT screen has been completed, the refresh memory can remain static until the next power line synchronizing pulse is received. The disadvantage of static shift registers is that they are more expensive and require more complex circuitry than dynamic shift registers. For these reasons, it is desirable to structure the refresh memory with dynamic shift registers; however, such registers must be continually clocked at some specified minimum rate to prevent them from losing information. Thus, it is not possible to delay clocking of a dynamic shift register memory until a synchronization pulse arrives.

SUMMARY OF THE INVENTION

The present invention provides a synchronizing system which permits characters to be drawn at different rates on a CRT screen using non-constant clocking of a refresh memory which comprises dynamic shift registers. The characters displayed on the screen are substantially flicker-free and do not wobble. The number of characters per page that can be displayed is substantially larger than with other systems employing constant character drawing times, yet the circuitry required is relatively inexpensive and not complex. The quality of the characters displayed is as good or better than other systems.

According to the illustrated embodiment of the invention, there is provided "floating reference" synchronizaion of the refresh memory and the power line frequency. Synchronizing pulses derived from the power line may be referenced to any point in the memory rather than only to a fixed point. The beginning of each refresh cycle is varied relative to the beginning of each cycle of the AC power line. The variations in the sync pulse reference point depend on the time needed to draw all of the characters on the display screen in each refresh pass.

During a refresh cycle, the memory may be stepped from character to character at three different clock rates: (1) a fixed slow rate which takes longer than the drawing time for the most complex character to be displayed; (2) a varying rate which depends on the time required for the CRT beam to draw each character; and (3) a fixed fast rate which steps the memory quickly through unused memory locations and which is generally too short in time to permit drawing of charac-

ters. The character-by-character drawing of each page displayed is started by stepping the refresh memory at the slow clock rate. This continues until a power line synchronizing pulse is received. Thereafter, the memory is stepped at a variable clock rate as determined by the complexity of the characters themselves until all the characters on the page are drawn by the CRT beam. From then on, the memory is stepped at the fast clock rate until it reaches the beginning character of the page. The three clock rates are used in a manner such that the total clocking time spent per page of characters displayed on the CRT screen will equal the refresh period of 1/60 second. When all the characters in memory cannot be drawn in this time, the slow clock rate is not used, and instead the memory is stepped as fast as possible after each character is drawn.

In effect, the slow clock rate is used to fill in unused refresh time, thereby to provide the timing adjustment for variations in text. As characters are added in the refresh memory, more time is spent in drawing them and the number of slow memory clock pulses produced during each refresh cycle will automatically be reduced. Conversely, when characters are removed from the memory, more fill-in time is required and thus more slow clock pulses are used.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of the preferred embodiment of the system incorporating the present invention.

FIGS. 2 (a)-(c) are waveform diagrams illustrating the generation of synchronization pulses.

FIG. 3 is a flow diagram illustrating the operation of the system of FIG. 1.

FIG. 4 is a timing diagram illustrating one mode of operation of the system of FIG. 1.

DESCRIPTION OF THE PREFERRED EMBODIMENT

Referring now to FIG. 1, a refresh memory 11 provides characters to a character generator 13, which in turn controls a cathode ray tube (CRT) display 15 to draw alphanumeric characters on the CRT screen. Refresh memory 11 comprises a plurality of dynamic shift registers in a recirculating loop. Characters are stored in coded form and applied one-by-one to character generator 13. Memory 11 has a character capacity large enough to produce a page of text on the CRT screen. With each circulation of the memory, the page of text is refreshed on the CRT screen. Memory 11 is stepped from one character location to the next by a STRT signal at its control input 17. After a character at the memory output is drawn on the CRT screen by character generator 13, it produces a "Request Next Character" pulse on output line 19. This pulse occurs at the end of each character and remains high until the memory is stepped to the next character. Memory organization is such that any unused memory locations at the end of the memory following a page of text contain special NOP codes. These codes are recognized by character generator 13 to maintain the signal at output line 19 high, as shown in FIG. 1.

A memory address counter 21 is incremented one count by the STRT pulse each time memory 11 is stepped to its next character location. An address decoder 23 produces an output signal A_o when the count in counter 21 indicates that the first character location in the circulating memory 11 has been reached. In operation, the A_o pulse occurs at the beginning of each refresh cycle and at that time counter 21 is reset by means not shown to begin the count for the next refresh cycle.

The configuration and operation of the character generator and CRT display system is described in more detail in U.S. Pat. No. 3,660,833 issued to Walter T. Blejwas et al on May 2, 1972 and in copending U.S. Pat. application Ser. No. 464,990, entitled "Cursor Find System For the Display of a Word Processing System," filed in the names of Patrick P. de Cavaignac and Peter D. Dickinson. Said patent and application are assigned to the same assignee as the present invention.

The system of FIG. 1 also includes a first detector 25 which receives AC line voltage and detects the zero crossover point thereof. The AC line voltage is shown in FIG. 2(*a*). The output of detector 25 is a square wave signal, as shown in FIG. 2(*b*). Each edge of this signal represents a cross-over point and the positive going edges are detected by a second detector 27 which produces a synchronization pulse in response thereto, as shown in FIG. 2(*c*). The AC line operates at 60 Hertz so the time between synchronization pulses is 16.7 milliseconds (ms). Each sync pulse is 200 nanoseconds (ns) long.

Refresh memory 11 is not stepped from character to character at a constant clock rate throughout each refresh cycle. Instead, memory clocking varies and depends on the text displayed and the occurrence of a sync pulse generated from the AC line voltage. As will be described in detail hereinafter, the sync pulse may be referenced to any point in the memory and allowed to move as a function of the text displayed. Hence, this system is referred to as "floating reference synchronization."

The system operates to step refresh memory 11 in three different ways: (1) at a predetermined slow rate wherein the period between clocking (STRT) pulses is T_{max}; (2) at a variable rate dependent on the time required for the CRT beam to draw each particular character; and (3) at a predetermined fast rate which serves to move the memory quickly through unused memory locations that contain no text. In general, the system will remain sychronized to the AC line if the following boundary equations are satisfied:

EQUATION 1

$$N\,T_{max} > T_s$$

EQUATION 2

$$N\,T_c < T_s$$

Where N is the number of characters that may be stored in memory 11, T_{max} is the maximum memory clock period used; T_s is the time between sync pulses; and T_c is the time spent at each character location on the CRT screen. For example, with a 60 Hertz line frequency and a memory with capacity N = 2048 characters, the time T_{max} is 40 microseconds, T_s is 16.7 milliseconds, and T_c varies from 1 microsecond for unused (NOP) character locations to 40 microseconds (T_{max}) for complex characters, with an average of 5 microseconds.

The circuitry that maintains the non-constant memory stepping synchronized with the AC line is shown in FIG. 1 and the operation thereof is illustrated by the flow diagram of FIG. 3. Assume that memory 11 is beginning a refresh cycle and that the initial address

pulse A_o has just been produced. The A_o pulse is applied to the J input of a J-K flip-flop **29**, thereby to set this flip-flop. Assume also that a flip-flop **23** is initially in its reset state so its Q_2 output is high. The Q_1 and Q_2 signals enable an AND gate **35**, which in turn conditions an AND gate **37** to be enabled by the memory clocking signal STRT. At this time, the system is in state A, indicated by block **301** in FIG. **3**. The Q_1Q_2 designation adjacent block **301** illustrates the condition of flip-flops **29, 33**, respectively, in FIG. **1**.

Initially, memory **11** is stepped at fixed slow speed at intervals T_{max} (40 microseconds, for example). This is achieved as follows: the STRT pulse at the beginning of the refresh cycle enables AND gate **37**, the output of which is applied to the J input of a J-K flip-flop **39**, thereby to set this flip-flop. As a result, the Q output thereof goes low. The STRT signal is also applied through an OR gate **41** to the reset input of a counter **43**. Thereafter, counter **43** counts pulses (at 200 nanosecond intervals, for example) from a clock **45**. A decoder **47** monitors the output of counter **43** and produces an output pulse when the contents of counter **43** reaches a count indicating that the time interval T_{max} has elapsed. The output pulse from decoder **47** is applied to the K input of flip-flop **39**, thereby to reset it and cause the Q output thereof to go high. This Q output is applied to an AND gate **49** along with the output **19** of character generator **13** that is held high after a character is completed. Thus, after the delay T_{max} occurs, the next pulse from a clock **51** enables AND gate **49** to produce STRT which steps the memory to the next character location. Clock **51** produces pulses at 1 microsecond intervals, for example, and is reset by each STRT pulse to keep it synchronized with the rest of the system. The STRT pulse also advances counter **21**, resets delay counter **43** and sets flip-flop **39**, thereby to start the entire T_{max} delay cycle over again.

The above-described operation is shown in FIG. **3**, beginning at state A (block **301**). Thereafter, a STRT signal is produced (block **303**) and the T_{max} delay is monitored (by flip-flop **39**) as indicated by decision block **305**. During the T_{max} delay interval, the system also monitors for the occurrence of a sync pulse, as indicated by decision block **307** and as described later. As long as no sync pulse occurs, the system waits for completion of delay T_{max}, after which operation returns to state A. Thus, STRT pulses are produced at intervals T_{max} to step the memory. In this mode of operation, memory stepping is constant at T_{max} intervals even though each character is drawn on the CRT screen in substantially less time than T_{max}.

Memory stepping progresses at T_{max} intervals until a sync pulse occurs at the output of detecor **27**. In response to the first sync pulse, the following occurs: (**1**) flip-flop **29** is reset, so the Q_1 output goes low, thereby to disable AND gate **35** which in turn prevents STRT pulses from setting flip-flop **39**; (**2**) the Q_1 output of flip-flop **29** goes high to condition an AND gate **50** to be enabled and set flip-flop **33** in response to the next sync pulse; and (**3**) delay counter **43** is reset. The resetting of counter **43** operates to extend the last memory stepping interval so it ends exactly T_{max} microseconds after the arrival of the sync pulse, thereby to maintain precise synchronization and prevent jitter of the display. When the last T_{max} pulse does occur, it resets flip-flop **39**, which thereafter is held in its reset mode because AND gate **37** is disabled as described above and the J input cannot receive STRT pulses. In FIG. **3**, the resetting of delay counter **43** is indicated by block **309** and monitoring for the completion of a full T_{max} delay after the sync pulse is indicated by decision block **311**.

When flip-flop **39** is reset by the last T_{max} pulse, the Q output thereof goes high and the next STRT pulse is produced (block **313** in FIG. **3**). Thereafter, the Q output stays high and AND gate **49** produces a STRT pulse in response to a clock pulse from clock **51** as soon as character generator **13** completes drawing one character and provides a high level signal on line **19** to request the next character from memory **11**. In this mode of operation, a STRT pulse is produced as soon as a character is drawn and memory **11** is stepped from character to character at a varying rate, depending on how long it takes to draw each character. In FIG. **3**, operation cycles through state B (block **315**), producing a STRT pulse to step the memory (block **317**) after each character is drawn, as indicated by decision block **319**. This continues as long as a sync pulse does not occur (decision block **321**) and as long as an A_o pulse is not produced to indicate that memory **11** has started the next refresh cycle (decision block **323**). With each new refresh cycle, operation returns to state A (block **301**) and the memory is again stepped at T_{max} intervals, as described above.

If memory **11** is not filled to capacity, the unused memory locations at the end of the memory contain special NOP codes which cause character generator output **19** to remain high. STRT pulses are then produced by clock **51** at a fixed fast rate, e.g., at one microsecond intervals. Thus, memory **11** is stepped quickly through unused memory locations to the beginning of the next refresh cycle.

Occasionally, when the memory used has a large capacity, there are so many characters in memory that the time required to draw all of them is longer than the time between two sync pulses, e.g., longer than 16.7 milliseconds for a 60 Hertz AC line. In this case, a second sync pulse occurs in the same refresh cycle. Equation 2 above is violated and unresolvable hunting of the synchronization circuit may occur. This problem is prevented by allowing the memory stepping to run free for all memory locations until there is a complete refresh cycle without the occurrence of a sync pulse. In operation, the second sync pulse enables AND gate **50** to set flip-flop **33**. As a result, the Q_2 output thereof goes low and prevents AND gate **35** from being enabled the next time Q_1 goes high in response to an A_o pulse at the beginning of a new memory refresh cycle. As shown in FIG. **3**, the second sync pulse causes operation to branch from decision block **321** to state D (block **325**). Thereafter, the memory continues to be stepped in response to STRT pulses (block **327**) after each character is drawn (decision block **329**) as long as a new refresh cycle is not indicated by an A_o pulse (decision block **331**).

When A_o occurs and the next refresh cycle begins, flip-flop **29** (FIG. **1**) is set but operation does not return to state A (block **301**) because STRT pulses are still prevented from setting flip-flop **39**. Thus, at the beginning of the refresh cycle, the memory is not stepped at the longer time intervals T_{max}. Rather, memory **11** continues to be stepped as soon as each character is drawn. This reduces the total refresh cycle time and in effect permits the memory cycle to catch up with the the sync pulses. In FIG. **3**, this operation at the begin-

ning of a refresh cycle starts at state E (block **333**). The memory is stepped by STRT pulses (block **335**) after each character is completed (decision block **337**) until a new refresh cycle begins (decision block **339**), unless in the meantime a sync pulse occurs (decision block **341**), in which case operation branches back to state D (block **325**). Operation through states D and E insure that the memory is stepped as fast as possible until one complete refresh cycle takes place without the occurrence of a sync pulse. With reference to FIG. **1**, the first A_o pulse sets flip-flop **29**, which in turn conditions an AND gate **53** to be enabled by the second A_o pulse, providing no intermediate sync pulse occurs to reset flip-flop **29**. When this condition is met, AND gate **53** resets flip-flop **33**, so its Q_2 goes high, operation branches from decision block **339** back to state A (FIG. **3**) and the above-described memory stepping at T_{max} intervals resumes.

Operation of the syncing system in the steady state is defined by the following equation:

EQUATION 3

$$X\ T_{max} + (N\text{-}X)T_{av} = T_s$$

Where X is the number of memory locations, starting at A_o to use T_{max} clocking time, T_{max} is the maximum memory clock period used; N is the number of characters that may be stored in memory; T_{av} is the average time per character required to step through all characters and unused memory locations (NOP's) on a page of text; and T_s is the time between sync pulses. As the quantity of text increases, the number of NOP's decreases and T_{av} becomes larger. This forces the first term in Equation 3 to reduce in magnitude, and X is therefore reduced (T_{max} is a fixed value). Conversely, when text is removed, T_{av} reduces and causes a corresponding increase in the value of X.

FIG. **4** illustrates the STRT pulses which step memory **11** as a function of time. The mode of operation shown is that in which only one sync pulse occurs in each refresh cycle. The beginnings of successive refresh cycles are designated A_o, A_{o+1}, A_{o+2}, etc. The sync pulses that occur during the refresh cycles are designated $Sync_0$, $Sync_1$, $Sync_2$, etc. As shown, memory stepping takes place at fixed intervals T_{max} until a sync pulse occurs. Thereafter, the memory is stepped as soon as each character is drawn on the display until a new refresh cycle begins.

FIG. **4** also illustrates how the synchronizing system adapts in response to the sudden insertion of additional characters into memory **11**, as may occur during a text editing operation for example. Assume that a block of **100** characters is suddenly added to memory during the second refresh cycle, i.e., between A_{o+1} and A_{o+2}. If the average per-character drawing time T_{av} is 5 microsecond (μs), the minimum memory clocking interval is 1 microsecond (μs), the total increase in character drawing time is $100 \times (5\ \mu s - 1\ \mu s) = 400\ \mu s$. As a result, the starting point of the next refresh cycle is delayed by 400 μs. If T_{max} is 40 μs, there will be $400/40 = 10$ fewer T_{max} intervals produced than the initial number X_o by the time the next sync pulse ($Sync_2$) arrives. In FIG. **4**, this reduction is indicated as $(X_o - 10)\ T_{max}$. The 10 characters originally drawn at T_{max} intervals are now drawn at the average intervals of $T_{av} = 5\ \mu s$ each. Hence, the disturbance has been reduced to 50 μs, the time required to draw the 10 characters ($10 \times 5\ \mu s = 50\ \mu s$). The following equation governs the response of the system to a disturbance caused by the insertion of additional characters in memory:

EQUATION 4

$$D_n = (D_{n-1})\ (T_{av}/T_{max})$$

Where D_n is the disturbance time n refresh cycles after the initial disturbance when $n = o$. In the example given above, the disturbance after the first refresh cycle is $D_1 = (400\mu s)\ (5\mu s/40\mu s) = 50\ \mu s$. After the second refresh cycle, the disturbance $D_2 = (50\mu s)\ (5\mu s/40\mu s) = 6.25\ \mu s$. The system stabilizes as soon as the disturbance time becomes less than the interval T_{max}. Hence, in this case, the system will stabilize after the second refresh cycle.

We claim:

1. In a system for producing characters of a page of text on a display from a dynamic recirculating refresh memory, a method for synchronizing the system to the power line frequency to prevent movement of the display, the method comprising:

stepping said refresh memory from character to character at a predetermined slow clock rate until a power line synchronizing pulse occurs, wherein the time interval T_{max} between each clock pulse is longer than the time required to produce the most complex character to be displayed; and

stepping said refresh memory at variable clock rates after said synchronizing pulse occurs until all characters of a page are produced on the display, wherein a clock pulse occurs in response to the completion of each character displayed.

2. The method of claim **1**, further including:

stepping said refresh memory at a predetermined fast clock rate after the last character of a page is produced and until the beginning of the next refresh cycle of the memory.

3. The method of claim **2**, wherein the time interval between each clock pulse of the fast clock rate is at least as short as the time required to produce the simplest character to be displayed.

4. The method of claim **1**, wherein the stepping of said refresh memory at a predetermined slow clock rate includes:

extending the memory stepping time interval in which said synchronizing pulse occurs so that said interval ends at a time T_{max} after said synchronizing pulse, thereby to prevent jitter of the display.

5. A synchronizing system comprising:

means for producing periodically recurring sync pulses;

a dynamic recirculating refresh memory for storing a plurality of characters in coded form, said memory having a control input for stepping said memory from one memory location to the next to output the characters in sequence;

a character generator coupled to the output of said refresh memory for producing characters in a manner for display, said character generator having an output indicating when the generation of each character is complete;

means for sensing the beginning of each recirculation of said refresh memory;

first means coupled to the control input of said memory for stepping said memory at predetermined long time intervals T_{max} which are longer than the time required to generate the most complex character to be displayed;

second means coupled to the control input of said memory and to said character generator output for stepping said memory at varying time intervals in response to the generation of each character; and

control means responsive to said sync pulse producing means and said sensing means for enabling said first stepping means at the beginning of a memory refresh cycle and for thereafter disabling said first stepping means and enabling said second stepping means when a sync pulse occurs.

6. The system of claim **5,** wherein said character generator output also indicates unused memory locations and further including third means coupled to the control input of said memory and responsive to said character generator output for stepping said memory at predetermined short time intervals when a memory location contains no character information.

7. The system of claim **6,** wherein said control means also enables said third memory stepping means when a sync pulse occurs.

8. The system of claim **5,** wherein said first stepping means has a control input coupled to said sync pulse producing means and is responsive to a sync pulse to extend the next succeeding memory stepping interval to step said memory at time T_{max} after the occurrence of said sync pulse.

9. The system of claim **5,** wherein said sync pulse producing means includes:

means responsive to a source of AC line voltage for detecting the zero crossover points of said line voltage; and

means responsive to said zero crossover point detector for producing a sync pulse during each cycle of the AC line frequency.

10. The system of claim **5,** wherein said control means includes:

means responsive to the occurrence of a second sync pulse in the same memory refresh cycle for holding said first stepping means disabled and said second stepping means enabled at the beginning of the next memory refresh cycle; and

means for disabling said holding means in response to the absence of a sync pulse during a complete refresh cycle.

* * * * *