United States Patent

Dillon et al.

[15] **3,652,838**

[45] **Mar. 28, 1972**

[54] DEMAND AUTOMATED COMPUTER

[72] Inventors: **Vearl Joseph Dillon**, Chicago; **Walter M. Threewitt**, Carpentersville, both of Ill.

[73] Assignee: **Edac Company**, Chicago, Ill.

[22] Filed: **Mar. 2, 1970**

[21] Appl. No.: **15,348**

[52] **U.S. Cl.** **235/151.31**, 235/151.21

[51] **Int. Cl.** **G06f 15/56**

[58] **Field of Search** 235/151.31, 151.21, 183; 307/31, 34, 35

[56] **References Cited**

UNITED STATES PATENTS

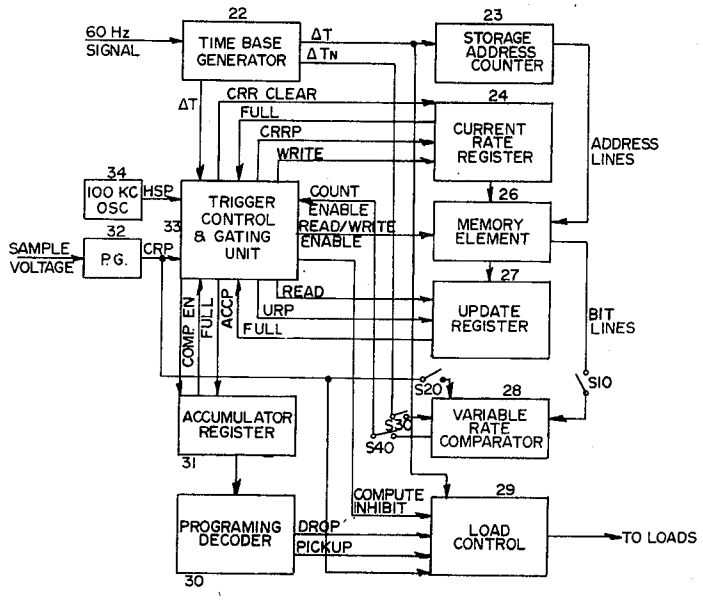
3,539,785	11/1970	Baker	235/183 X
3,387,121	6/1968	Maczuzak et al.	235/151.21
3,522,421	8/1970	Miller	235/151.21
3,296,452	1/1967	Williams	235/151.21 X
3,517,168	6/1970	Gates et al.	235/151.21 X
3,517,310	6/1970	Gates et al.	235/151.21 X

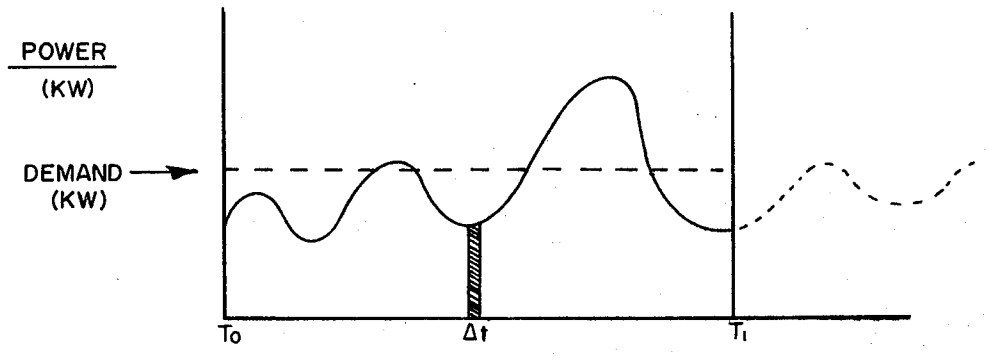
Primary Examiner—Malcolm A. Morrison
Assistant Examiner—Edward J. Wise
Attorney—Hill, Sherman, Meroni, Gross & Simpson

[57] **ABSTRACT**

A computer and controls system which monitors the application of power or other quantities subject to demand in which increased costs results due to periodic periods of high demand and controls loads on the system so that the cost is minimized while maintaining the desired service. This invention also allows continuous demand monitoring on industrial processes which are necessary for a specific application.

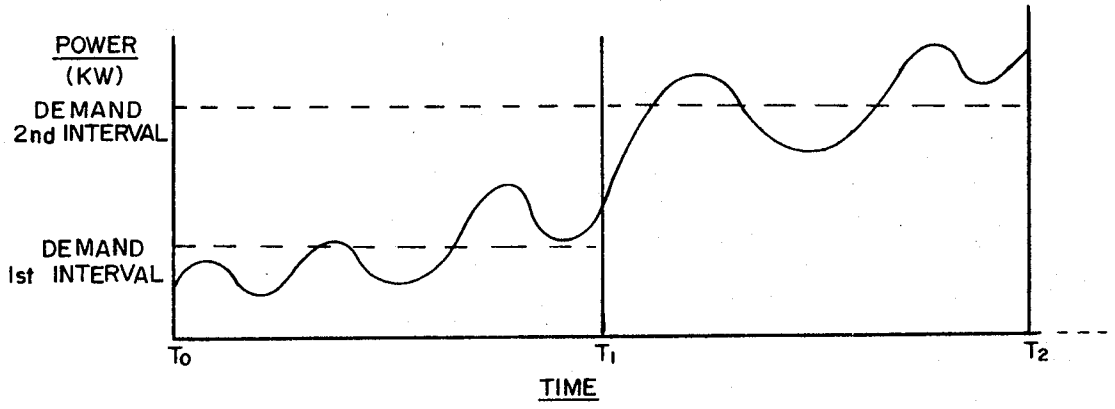
3 Claims, 31 Drawing Figures





$$\text{DEMAND} = \frac{\int_{T_o}^{T_1} P(t)\,dt}{T_1 - T_o}$$

Fig. 1



$T_1 - T_o$ = 1st DEMAND INTERVAL
$T_2 - T_1$ = 2nd DEMAND INTERVAL
$T_n - T_{(n-1)}$ = nth DEMAND INTERVAL

Fig. 2

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

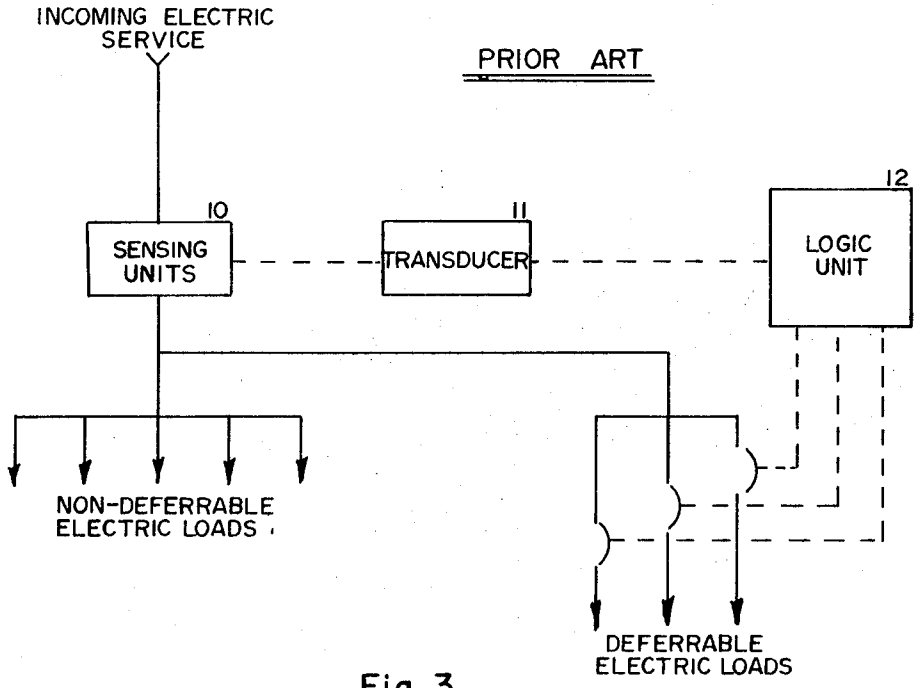


Fig. 3

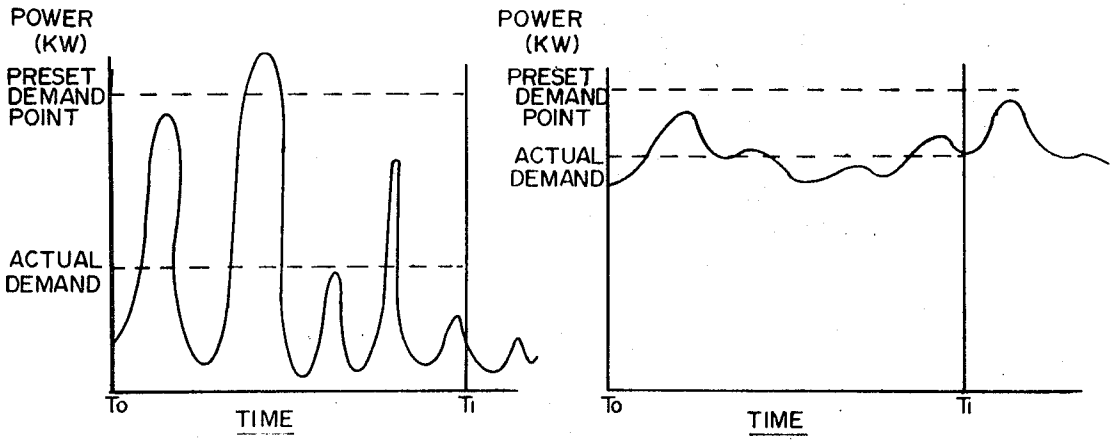


Fig. 4a

Fig. 4b

$T_i - T_o$ = DEMAND TIME INTERVAL

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

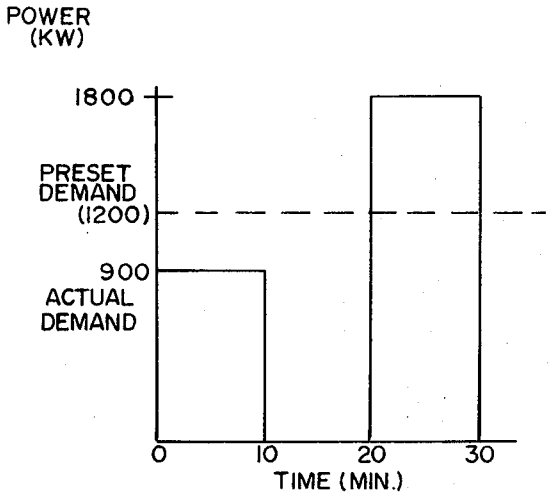


Fig. 5a

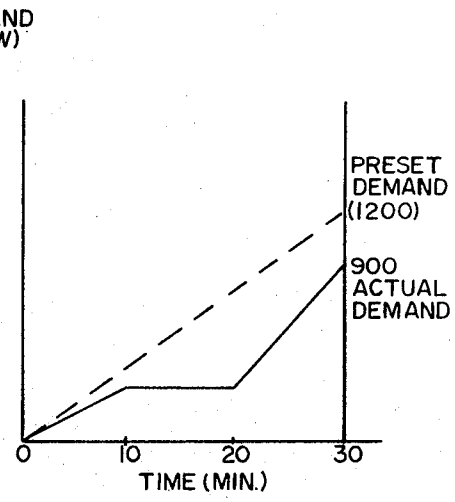


Fig. 5b

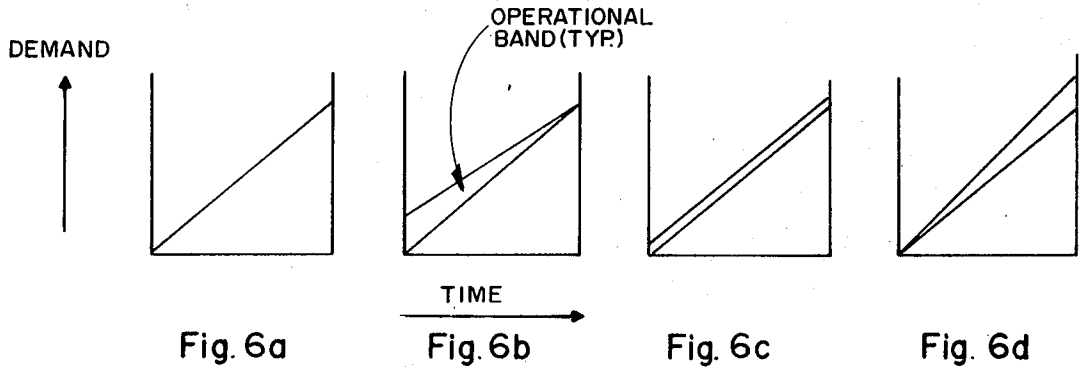


Fig. 6a Fig. 6b Fig. 6c Fig. 6d

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

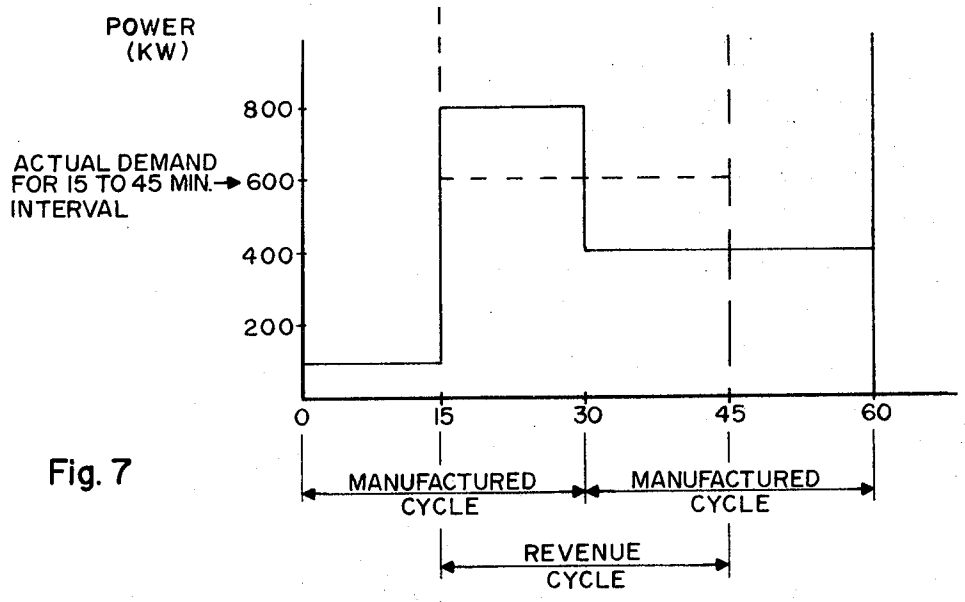


Fig. 7

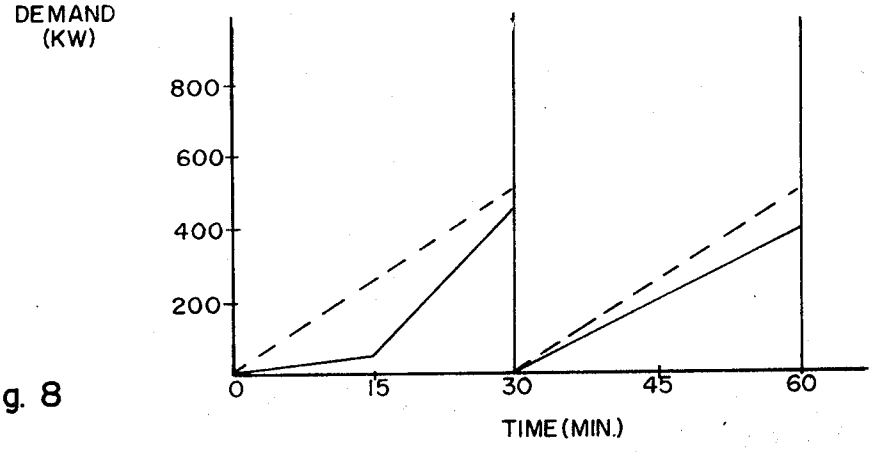
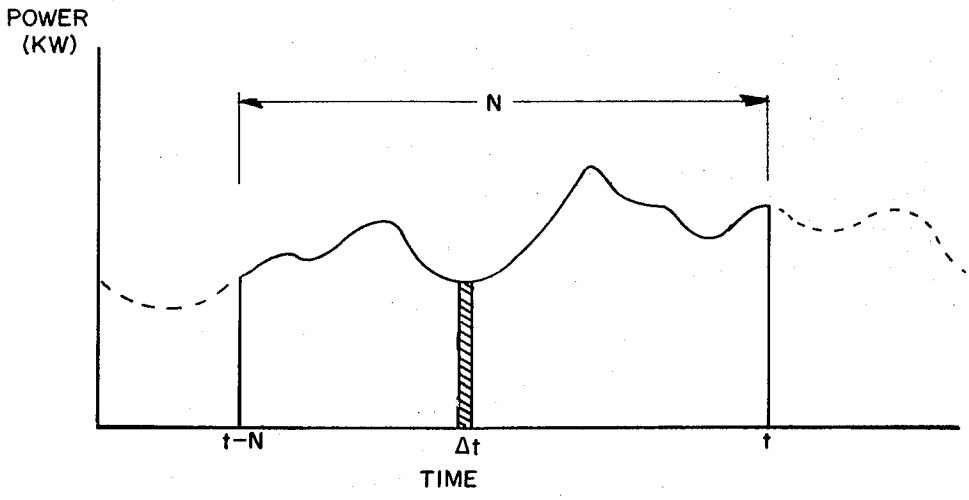


Fig. 8

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Dill, Sherman, Meroni, Gross & Simpson ATTORNEYS



$$\text{DEMAND}_{\text{(MEASURED CONTINUOSLY)}} = \frac{1}{N}\int_{t-N}^{t} P(t)\,dt$$

N = FIXED DEMAND TIME INTERVAL.
t = PRESENT TIME WHEN DEMAND IS COMPARED TO PRESET DEMAND.
t-N = BEGINNING OF FIXED TIME INTERVAL.
P(t) = QUANTITY BEING MEASURED AS A FUNCTION OF TIME.

Fig. 9

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt
BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

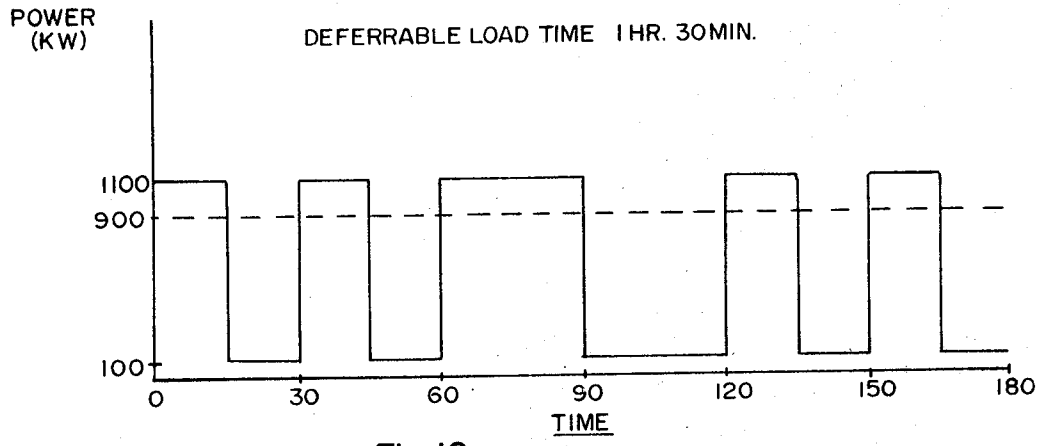


Fig. 10a

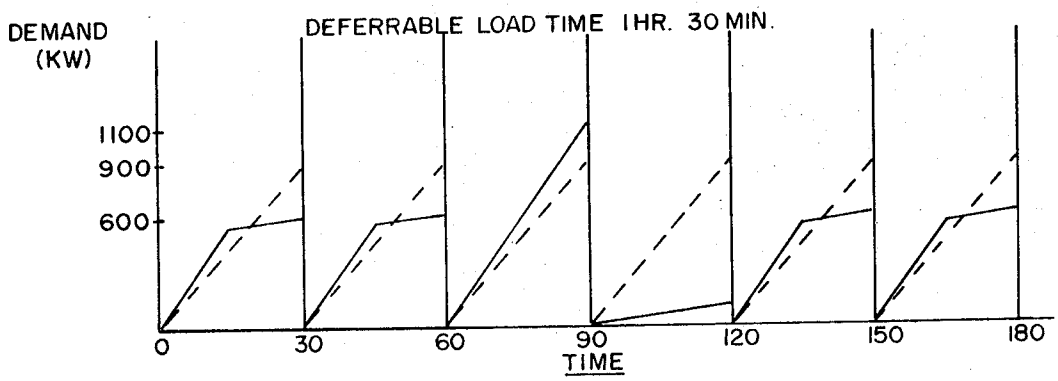


Fig. 10b

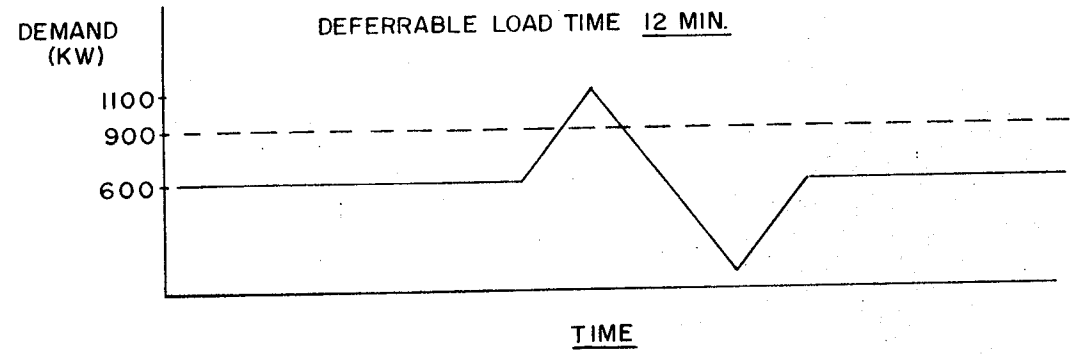


Fig. 10c

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

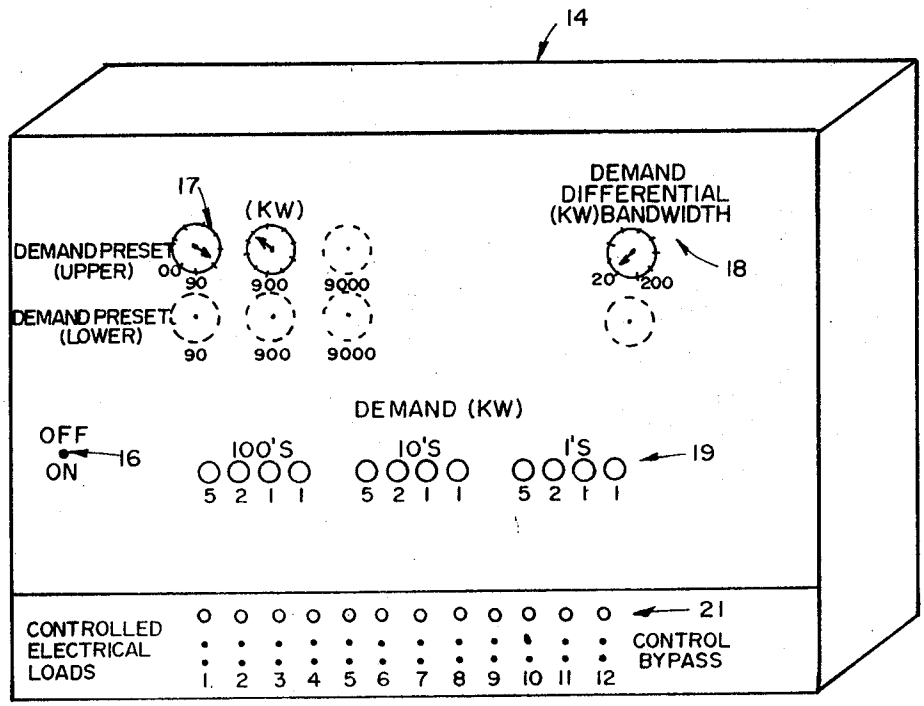


Fig. 11

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

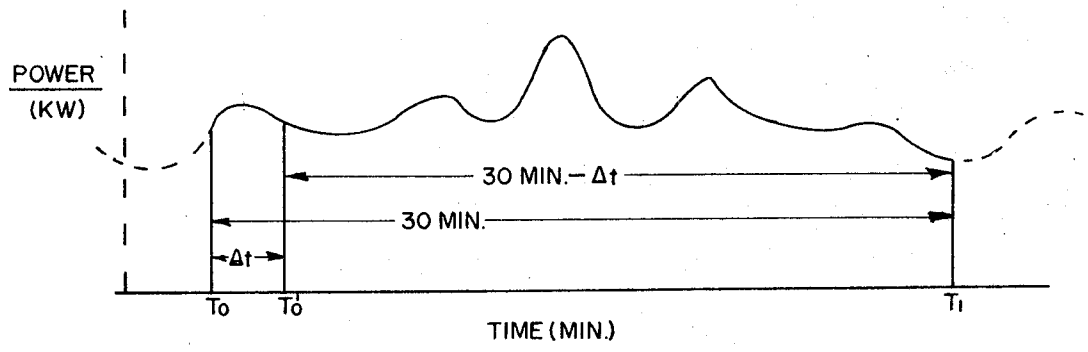


Fig. 12

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

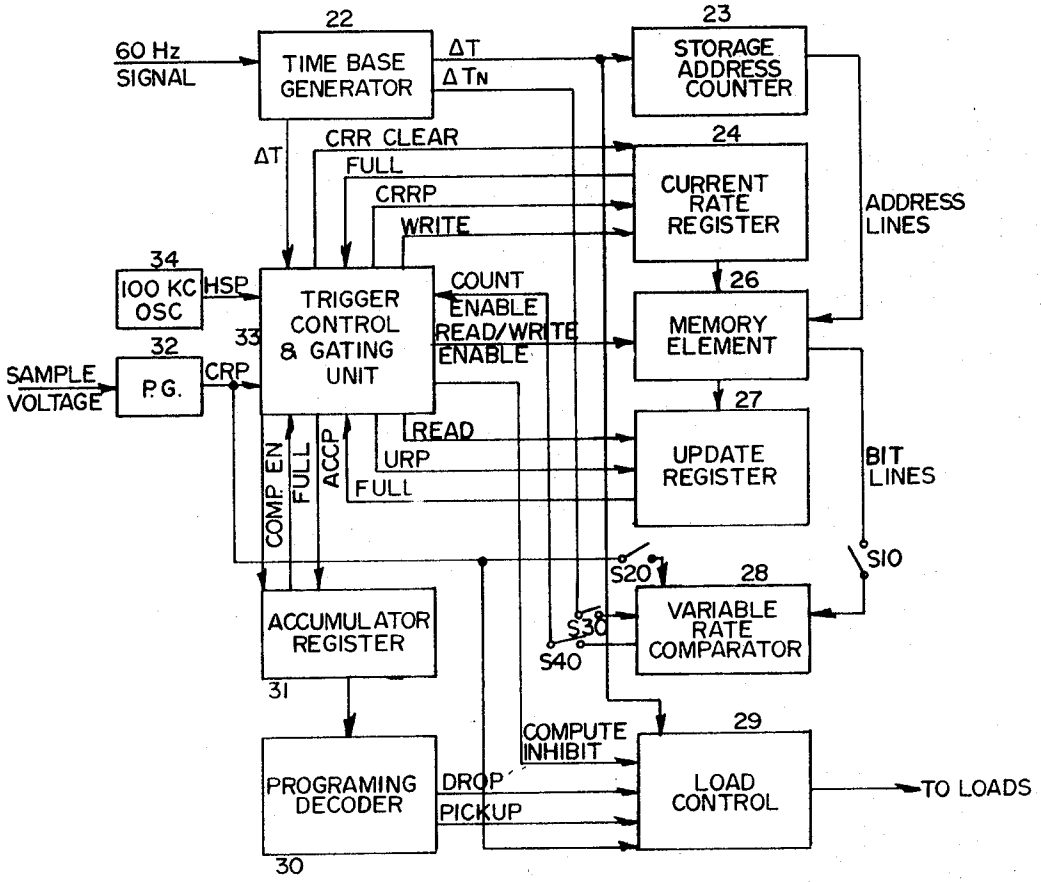


Fig. 13

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

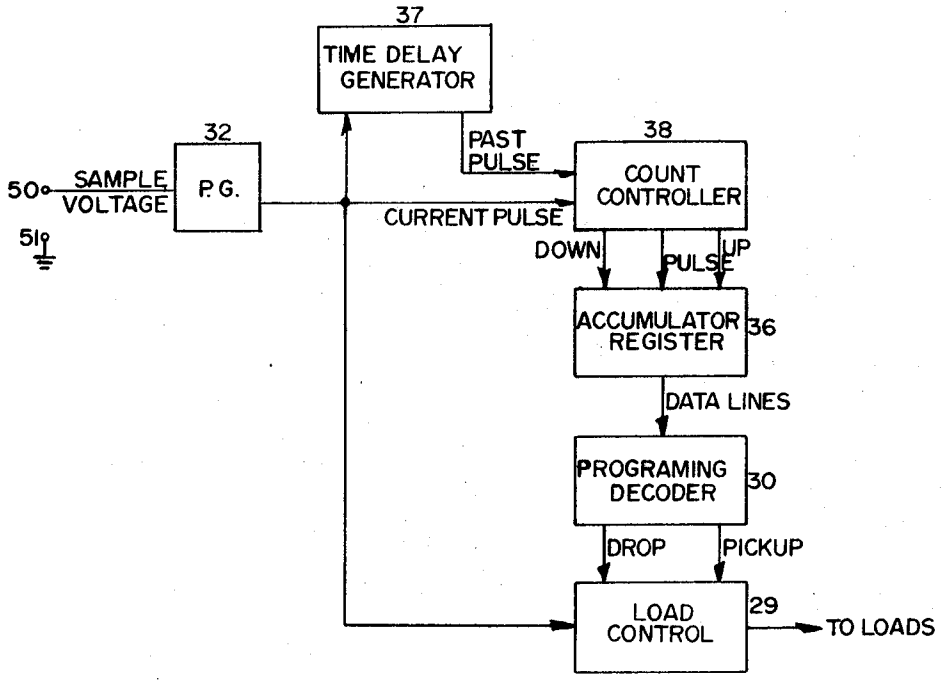


Fig. 14

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt
BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

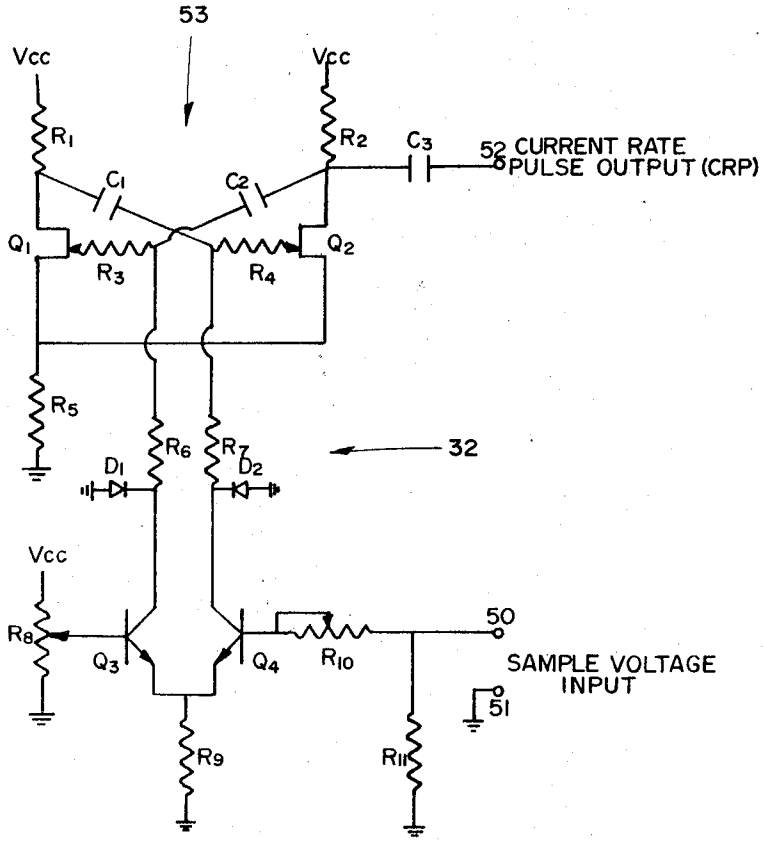


Fig. 15

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

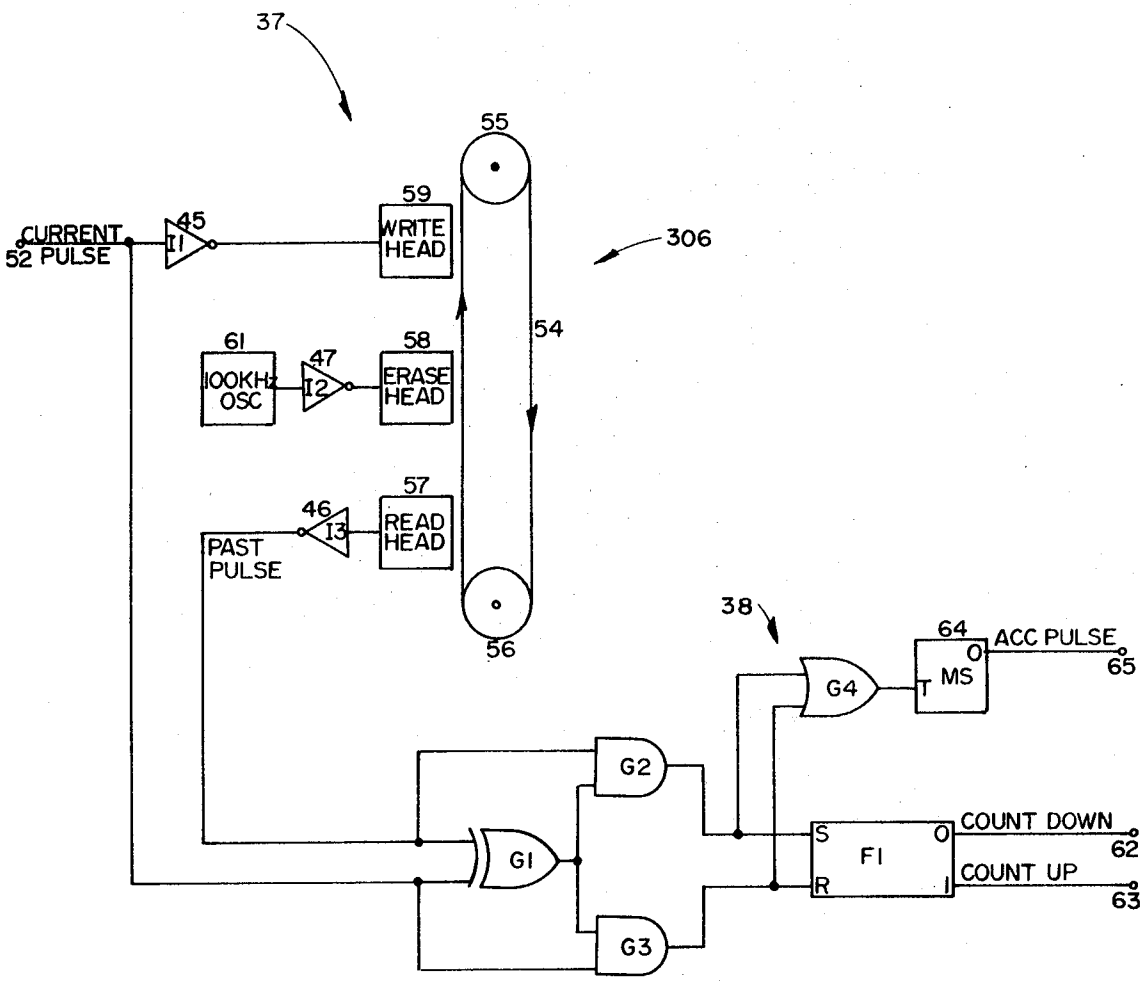


Fig. 16

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

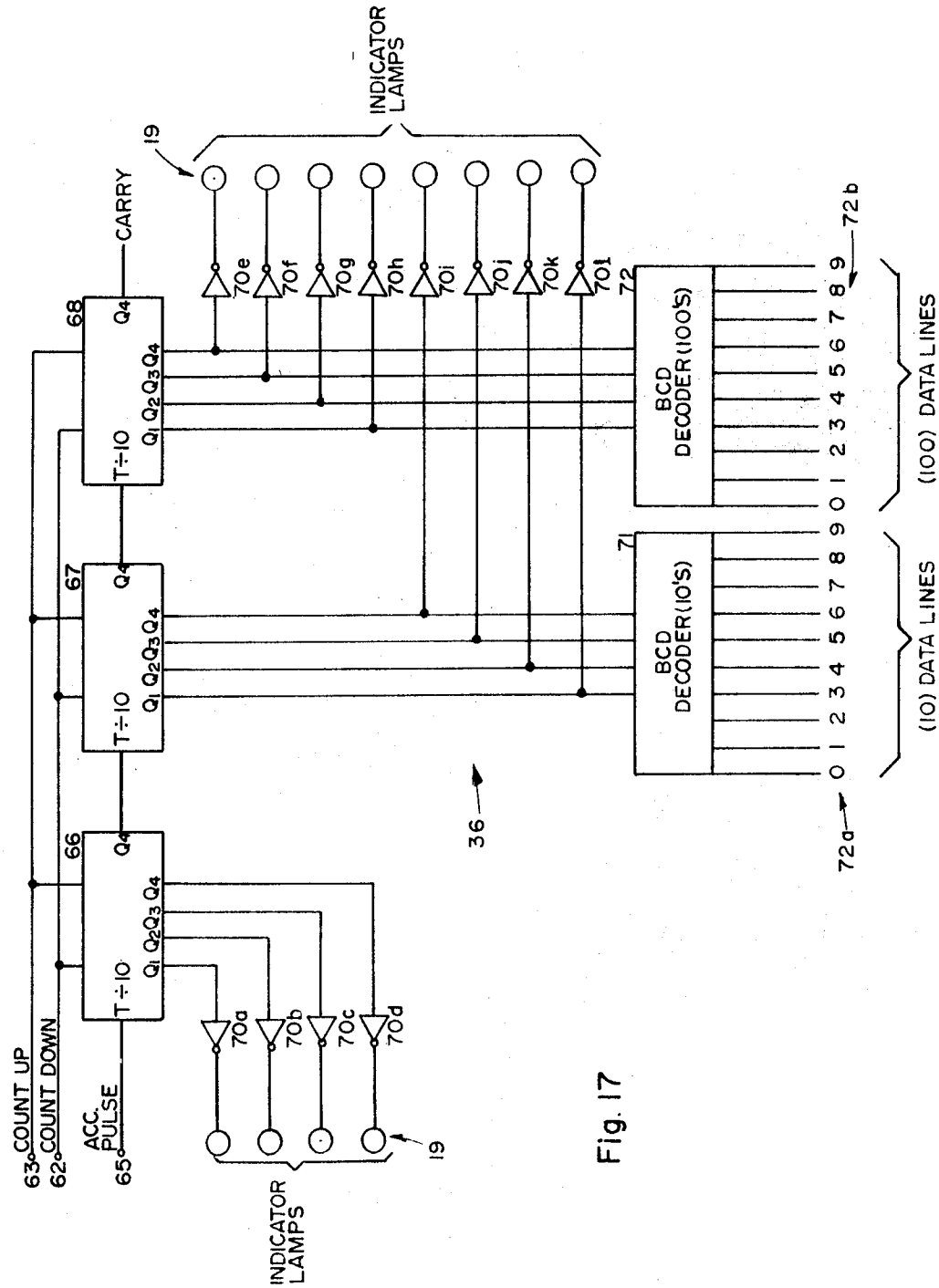


Fig. 17

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

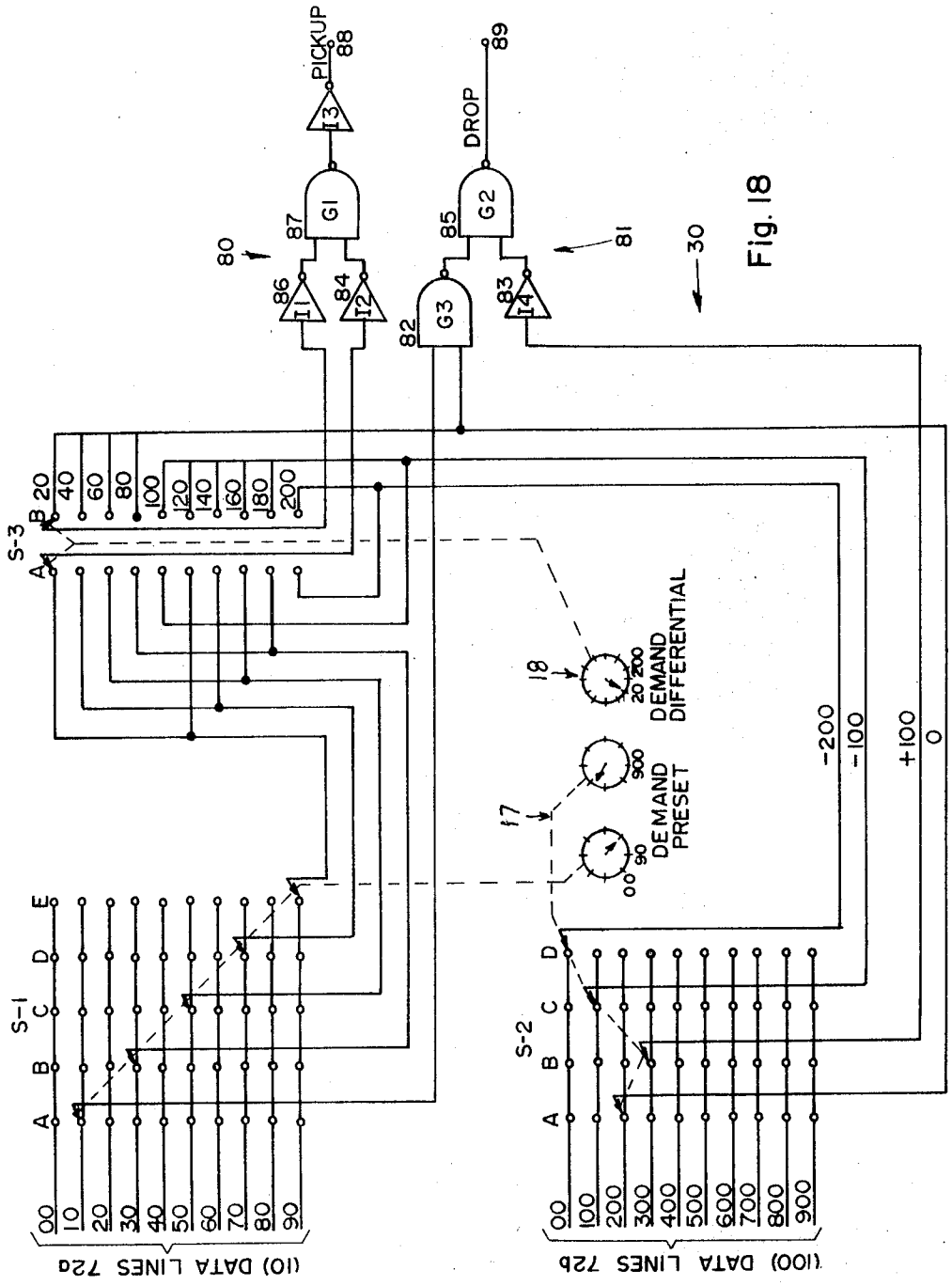


Fig. 18

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

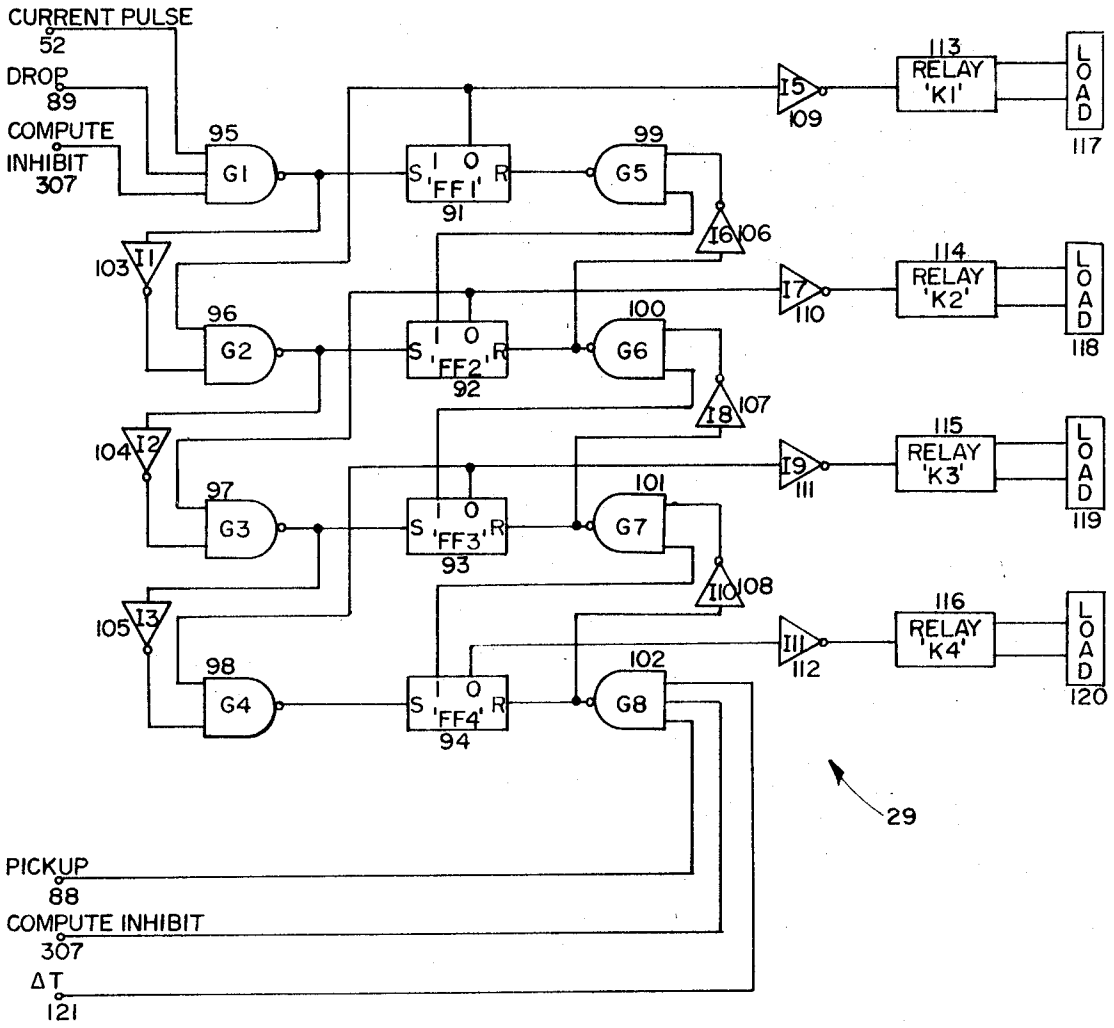


Fig. 19

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Kills, Sherman, Meroni, Gross & Simpson ATTORNEYS

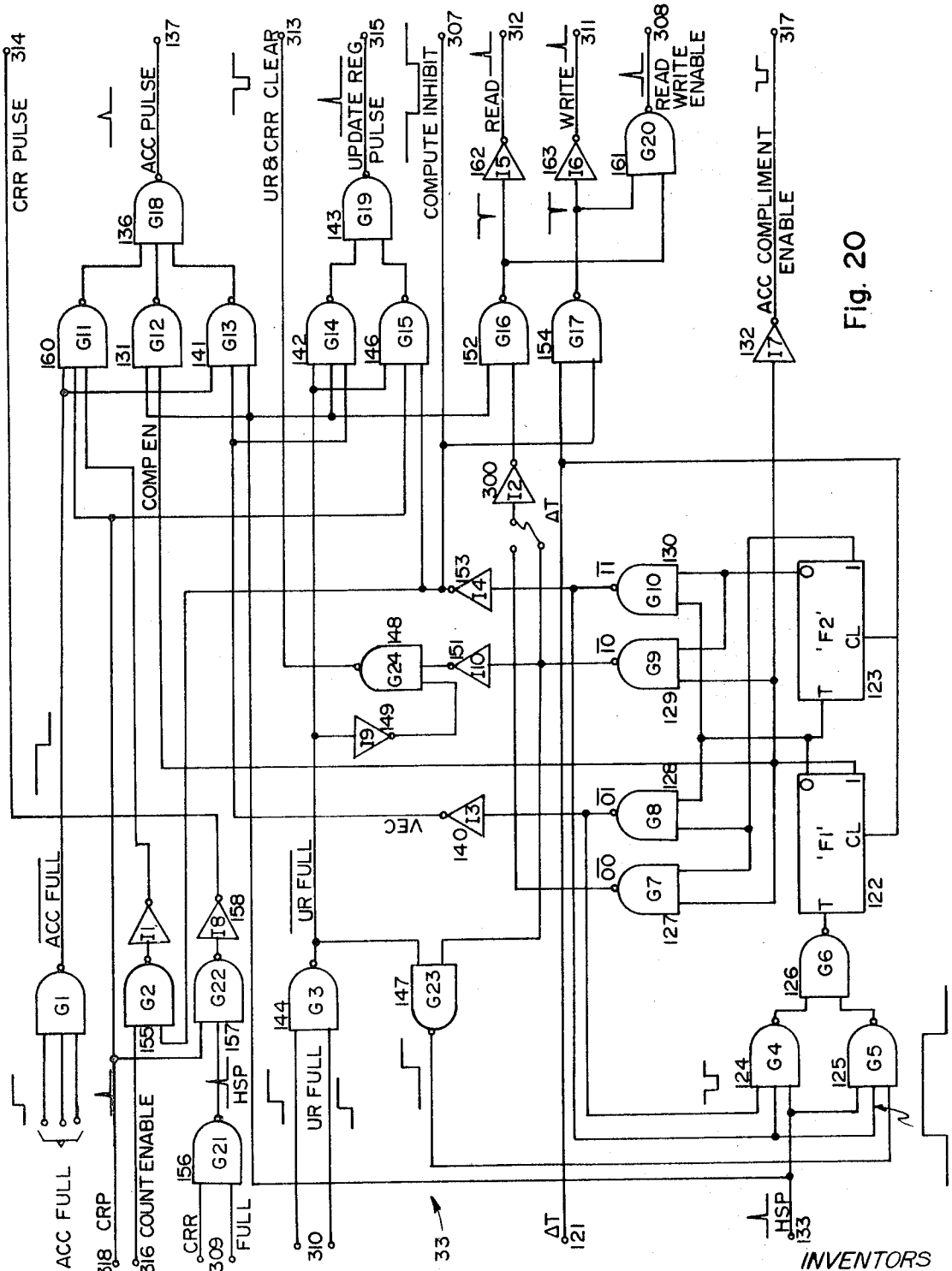


Fig. 20

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

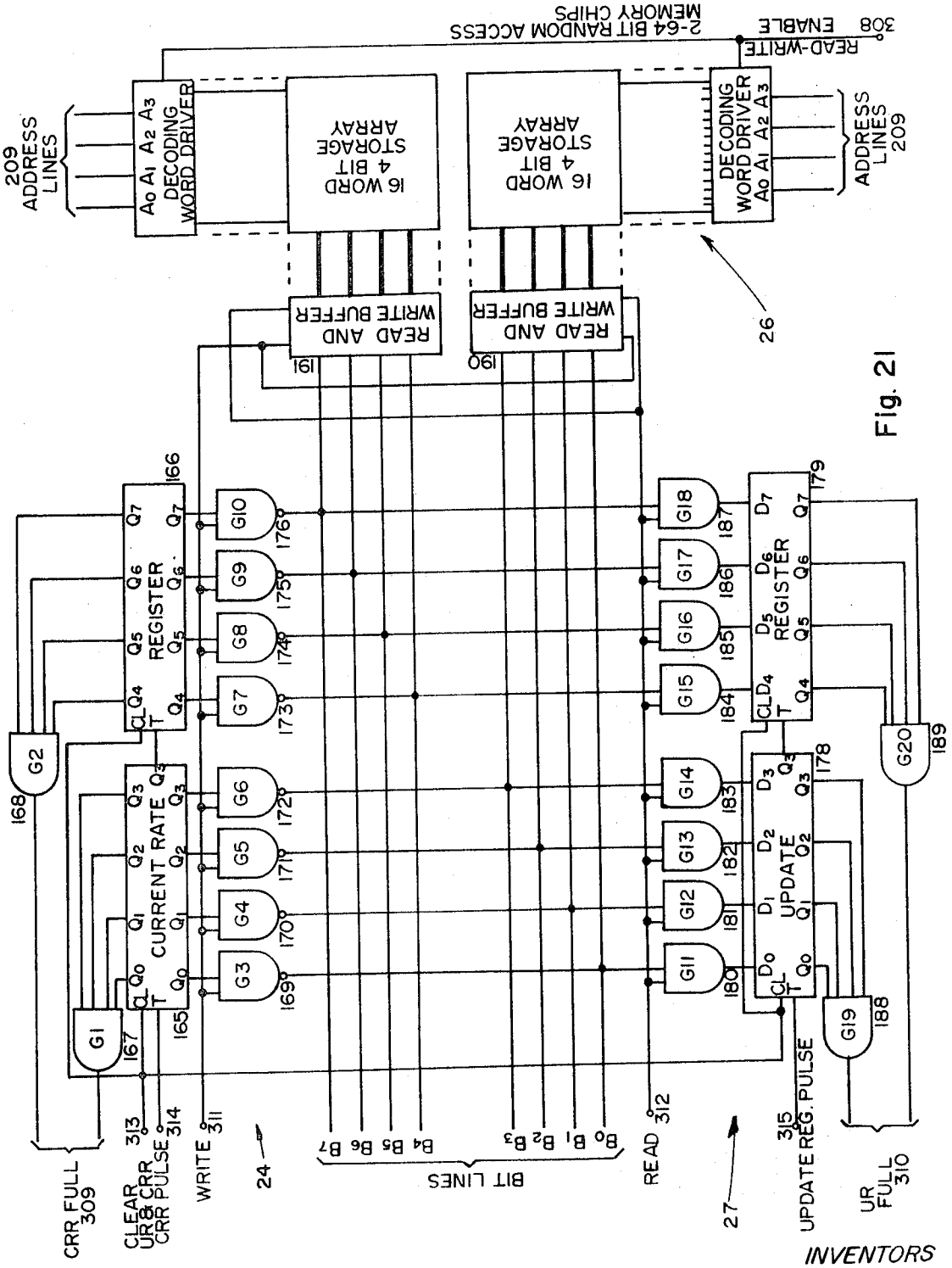


Fig. 21

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt
BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

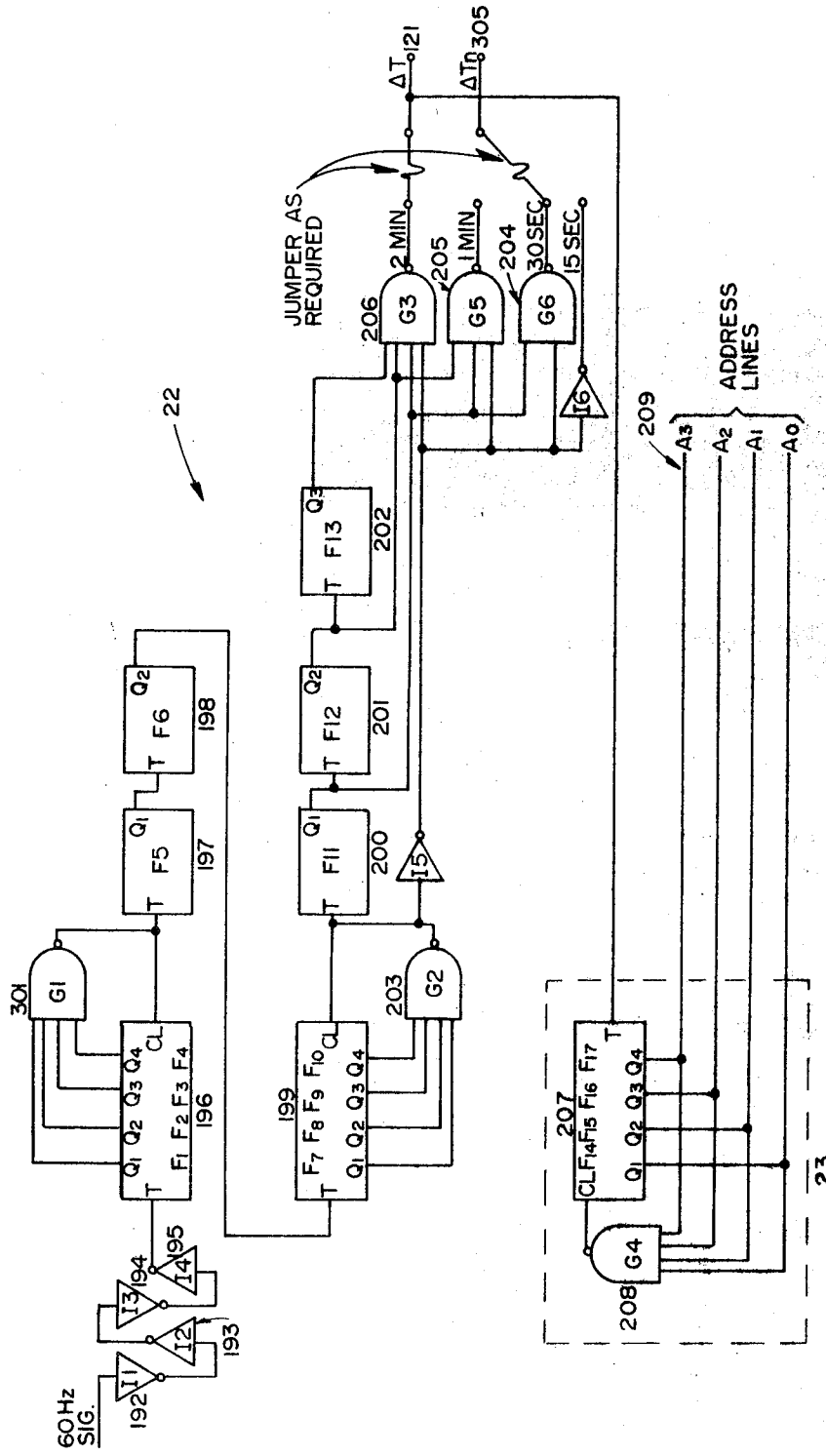


Fig. 22

INVENTORS
Pearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

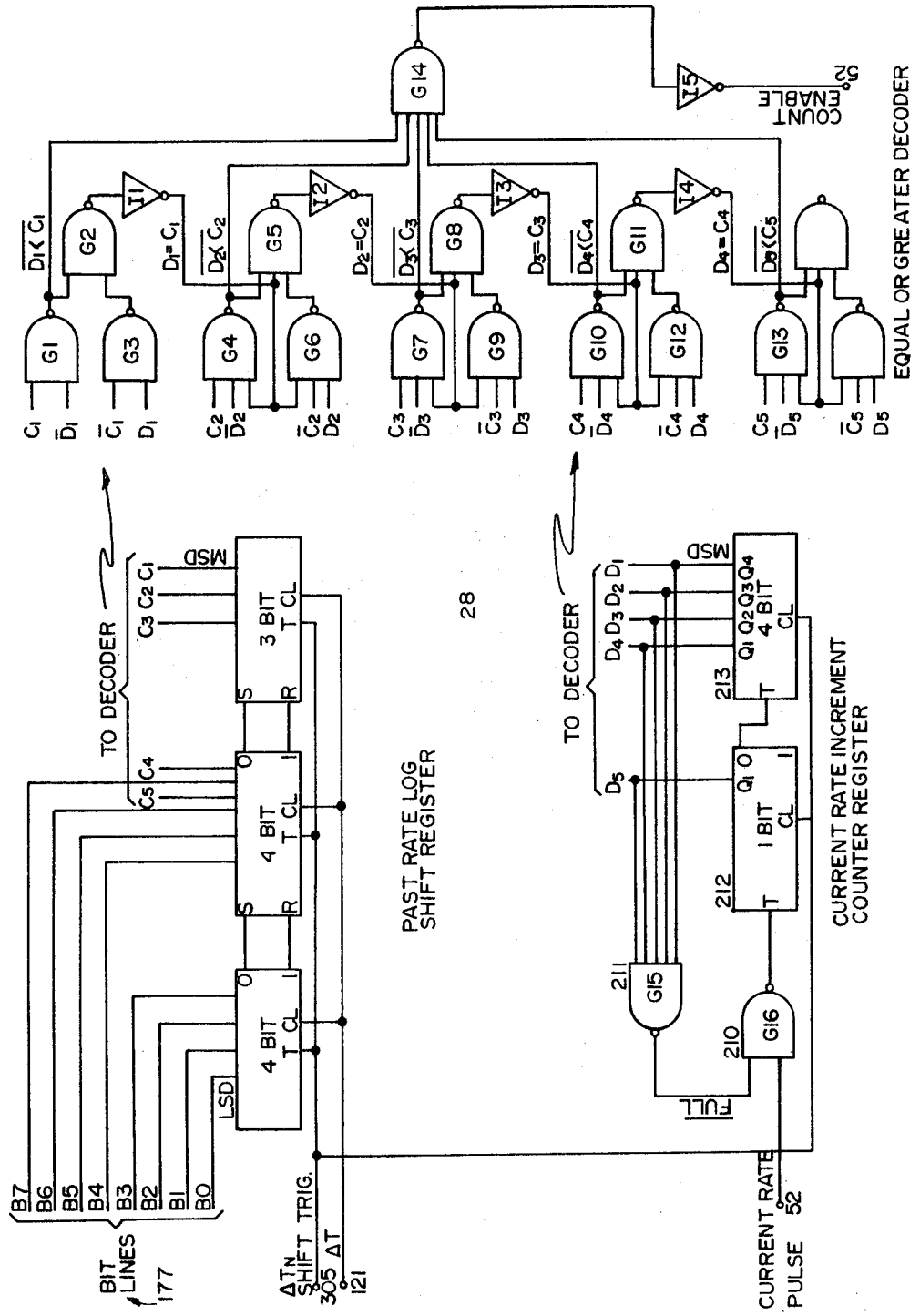


Fig. 23

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt
BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

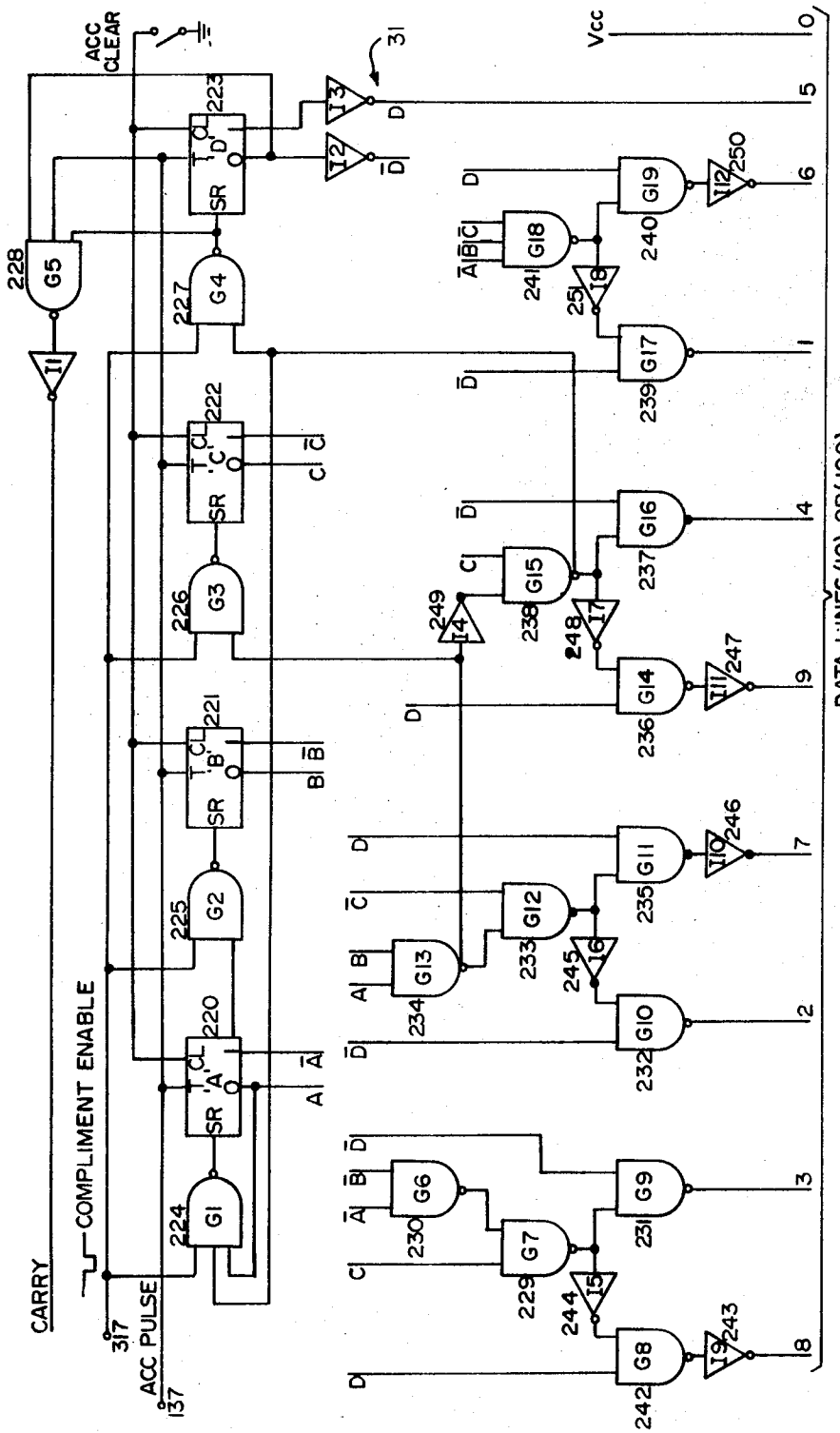


Fig. 24

INVENTORS
Vearl Joseph Dillon
Walter M. Threewitt

BY Hill, Sherman, Meroni, Gross & Simpson ATTORNEYS

DEMAND AUTOMATED COMPUTER

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates in general to the measurement and regulation of demand parameters for industrial applications, utility generation and distribution, and demand revenue economizing.

2. Description of the Prior Art

Most gas and electric utility companies establish commercial rate schedules for measurement of power consumption and peak demands, and due to irregular rate of power consumption by many customers, the demand peak is usually very high, which produces a high cost from the utility. It has previously been recognized that if the irregularities of power consumption could somehow be reduced without reducing the total power consumption, then peak demand would be reduced without adversely affecting the customer's facility operation.

Attempts have been made to regulate deferable loads for reduction of peak demand in recent years, which basically consist of an input sensing unit, a transducer conversion unit, and a logic unit. The sensing unit has generally comprised common manufactured items such as current and potential transformers, and the conversion unit has usually been a common watt transducer which supplies a millivolt output directly proportional to the kilowatts consumed. The logic unit has received the total consumer power intelligence and has attempted to regulate the deferable electrical loads in order to prevent the total consumed power from reaching a preselected demand point. One of the types of logic units presently available simply monitors any short interval peak kilovolt ampere above a pre-selected setting to regulate deferable loads. This unit has two gross deficiencies; (1) the KVA monitoring is not a measurement of power consumed, and (2) monitoring short interval peaks of a few seconds will regulate deferred loads unnecessarily. This unnecessary regulation, in addition to possibly being a nuisance and harmful to electrical loads, will necessitate increasing the demand set point, thus resulting in lower efficiency and perhaps no control at all when irregular load patterns are present. The preset demand point must be set very high in order to ensure that deferred loads are not operated too frequently. Also, during changing load patterns, an existing high preset point affords absolutely no control.

Another type of logic unit measures power consumed and integrates intervals of power sequentially. This type of unit provides a somewhat better percent of efficiency, since part of the unused power of time cycle is available until reset time and the start of a new cycle. The deficiency of this type of unit is that it must be synchronized to the power utility meter for the same sequential power intervals for total efficiency, and many power utilities are reluctant to allow synchronization since customer equipment would be connected to the revenue-measuring equipment. Therefore, without synchronization high blocks of power occurring within the revenue demand cycle may not be monitored, since they could fall in between two manufactured metering cycles.

SUMMARY OF THE INVENTION

The present invention resulted from a consideration of the inherent deficiencies of the two types of logic units mentioned above and it was recognized that a new device was needed which could monitor any revenue demand cycle, while not being physically connected or synchronized to the demand metering equipment. Thus, any irregular or smooth load pattern, or a combination of both, would not defeat control of a preset demand and would avoid all nuisance or harmful, unnecessary regulation. Such a system results in maximum efficiency. The present invention comprises an electrical demand automated control computer which maintains a continuous memory and integrates the previous amount of power consumed in a demand time interval divided by that time interval

at each instant in time. In other words, it calculates demand at each instant in time. Now, whenever a revenue demand cycle occurs, the present invention records that particular demand and compares it to the preset demand point for possible regulation of deferable loads. Thus, the invention monitors a continuous average of power being consumed since the beginning of the interval which is updated each instant minus an amount of power consumed in the instant when the demand interval begins. Thus, a very efficient device results in that no wave form of demand in any shape, such as irregular, erratic, spiked or smooth, will pass detection of the present invention.

Other objects, features and advantages of the invention will be readily apparent from the following description of certain preferred embodiments thereof taken in conjunction with the accompanying drawings, although variations and modifications may be effected without departing from the spirit and scope of the novel concepts of the disclosure, and in which:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a plot of power versus time;

FIG. **2** is a plot of power versus time;

FIG. **3** is a block view of a system incorporating a prior art device;

FIG. **4***a* is a plot of power versus time;

FIG. **4***b* is a plot of power versus time;

FIG. **5***a* illustrates power versus time;

FIG. **5***b* illustrates demand versus time;

FIGS. **6***a*–**6***d* illustrate variations in preset demand limits and operational limits of demand versus time;

FIG. **7** is an illustration of another power versus time;

FIG. **8** is another plot of demand versus time;

FIG. **9** is a plot of power versus time;

FIG. **10***a* is a plot of power versus time;

FIG. **10***b* is a plot of demand versus time;

FIG. **10***c* is a plot of demand versus time;

FIG. **11** is a perspective view of a power computer according to this invention;

FIG. **12** is a plot of power versus time;

FIG. **13** is a block diagram of the invention;

FIG. **14** is a block diagram of a modification of the invention;

FIG. **15** illustrates the pulse generator illustrated in FIGS. **13** and **14**;

FIG. **16** illustrates the time delay generator and count controller of FIG. **14**;

FIG. **17** discloses the accumulator register of FIG. **14**;

FIG. **18** discloses the programming decoder of FIGS. **13** and **14**;

FIG. **19** illustrates the load controller of FIGS. **13** and **14**;

FIG. **20** illustrates the trigger control and gating unit of FIG. **13**;

FIG. **21** discloses the current rate register, the update register and the storage element of FIG. **13**;

FIG. **22** illustrates the time base generator and the storage address counter of FIG. **13**;

FIG. **23** discloses the variable rate comparator and the past rate log register of FIG. **13**; and

FIG. **24** illustrates the accumulator register and decoder of FIG. **13**.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

In order to illustrate and explain peak demand and the electrical demand automated control computer of this invention the electrical power consumption technology will be presented although similar applications in gas consumption and industrial processes, for example, exist.

Electrical demand is measured by a demand meter and is expressed in real power (watts) averaged over a specified interval of time as shown in FIG. **1**. The most common demand intervals used in commercial metering are 15, 30 and 60 minutes. The "peak" demand is the maximum recorded demand averaged in the 15, 30 or 60 minute interval which has

occurred during a given period of time, usually the billing period. Each demand interval is recorded sequentially as illustrated in FIG. 2.

In order to reduce "peak" demand, certain electrical loads must be regulated, that is, they must be turned off (deferred) as the total facility electrical load is approaching peak usage. After the critical peak time period, these deferred loads must be reinstated systematically. For example, certain air conditioning loads, refrigeration loads, electric water heating loads, snow melting systems, auxiliary air compressor units, and electric furnace salvage operations are examples of deferable loads. These are electrical loads which can be disconnected for short periods of time without adverse effects upon the system as a whole.

Attempts have been made to regulate deferable loads for reduction of peak demand in recent years. As shown in FIG. 3, manufacturers have provided systems which basically consist of an input sensing unit **10**, conversion unit **11** (transducer), and logic unit **12**. The sensing unit **10** is usually comprised of common manufactured items such as current and potential transformers. The conversion unit **11** is usually a common watt transducer which supplies a millivolt output in direct proportion to kilowatts consumed. The logic unit **12** receives the total consumed power intelligence and attempts to regulate deferable electrical loads in order to prevent the total consumed power from reaching a pre-selected demand point.

Certain types of logic units have previously been available. One type logic unit simply monitors any short interval peak KVA above a pre-selected setting to regulate deferable loads. Such units have two deficiencies. First, KVA monitoring is *not* a measurement of power consumed, and second, monitoring short interval peaks of a few seconds will regulate deferred loads unnecessarily. Such unnecessary regulation, in addition to possibly being a nuisance and harmful to electrical loads, will necessitate increasing the demand set point thereby resulting in lower efficiency and perhaps no control at all when irregular load patterns are present. The preset demand point must be set very high as in FIG. **4***a* in order to insure that deferred loads are not operated too frequently. However, if the load pattern changes as in FIG. **4***b*, the existing high preset point affords absolutely no control.

Another type logic unit measures power consumed and integrates intervals of power sequentially. This type of unit provides a somewhat better percent of efficiency than the first type since part of the unused power of a time cycle is available until reset time and the start of a new cycle. FIGS. **5***a* and **5***b* illustrate the system. FIG. **5***a* represents power versus time and FIG. **5***b* represents demand versus time for the same load. As shown, the power has exceeded 1,200 KW for the last 10 minutes (FIG. **5***a*) of the demand time interval, but has not exceeded the accumulative power of 1,200 KW for 30 minutes as indicated by the preset demand limit (dotted line) in FIG. **5***a*. Variations of the preset demand limits and operational limit bands are shown in FIGS. **6***a* through **6***d*.

This type unit *must* be synchronized to the power utility meter for the same sequential power intervals for total efficiency. Many power utilities may be reluctant to permit synchronization since customer equipment would be introduced in the revenue measuring equipment. Therefore, without synchronization, high blocks of power occurring within the revenue demand cycle may not be monitored since they could fall in between two manufactured metering cycles as shown in FIG. **7**. If a preselected demand limit of 500 KW is chosen, the preset demand limit graph will be as shown in FIG. **8**. In both demand intervals, the preset demand of 500 KW is not exceeded and therefore the logic unit will not regulate any of the deferable loads. However, a demand of 600 KW was attained in the interval between 15 minutes and 45 minutes, a possible time interval as measured by the revenue metering equipment. Therefore, 100 KW of demand has slipped by and the efficiency has been considerably reduced.

The deficiencies of the two types of logic units available led to the present invention which monitors any revenue demand cycle while not being physically connected or synchronized to the demand metering equipment. Therefore, any irregular or smooth load pattern or a combination of both patterns will not defeat the control of a preset demand and will avoid all nuisance or harmful unnecessary regulation. This gives *maximum efficiency*. The invention comprises an electrical demand automated control computer which maintains a continuous memory and integrates the previous amount of power consumed in a demand time interval divided by that time interval *at each instant in time*. In other words, it calculates the demand at each instant in time. Now, when every revenue demand cycle occurs, the computer has recorded that particular demand and compared it to the preset demand point for possible regulation of deferable loads.

The graph of FIG. **9** shows a continuous average of power being consumed since the beginning of the interval which is updated each instant minus an amount of power consumed in the instant when the demand interval began. The computers of this invention monitor this average (demand) continuously by constantly measuring, storing, and calculating the definite integral of a quantity as a function of time from the beginning of a fixed time interval to the present time divided by the same fixed time interval as shown in FIG. **9**. This results in a totally efficient device in that *no* waveform of demand in any shape (irregular, erratic, spiked, or smooth) will pass detection from the computer of this invention.

The efficiency can be easily shown by taking an erratic waveform in a sample time period of three hours as shown in FIGS. **10***a*–**10***c* and comparing present devices with the computer of the invention. FIG. **10***a* represents a power waveform with a peak load control unit preset for 900 KW. To insure that 900 KW of demand is not exceeded, the peak load control must defer electrical loads for 1 hour and 30 minutes in the sample time period. FIG. **10***b* represents demand as measured by sequential integral monitoring units with a preset demand for 900 KW also. Again, to insure that 900 KW of demand is not exceeded the sequential integral monitoring unit must defer electrical loads for 1 hour and 30 minutes in the sample time period. Now, FIG. **10***c* represents the actual demand of the sample waveform at each instant in time which is measured by the computers of the invention. To insure that 900 KW of demand is not exceeded, the computer of the invention defers electrical loads for only 12 minutes in the total sample time period. This example shows without doubt that when demand actually becomes larger than the preset demand in any short or long period of time, the computer of the invention has it recorded 100 percent of the time and does not indicate any false demand message in any other period of time as the other devices repeatedly indicate.

The computer of the invention is simple and electronic solid state components are used although the same logic operation could be performed by using fluidic components.

As shown in FIG. **11**, the exterior controls on the case **14** are minimal. An on-off switch **16** is provided. Demand preset and demand differential switches **17** and **18** are provided. The demand preset control consists of decade position switches **17** for the deferable load drop out point. Additional decade switches can be added easily depending upon change in the application. The demand differential control is also a decade switch **18** which controls the deferable load pickup point from the established demand preset point. This switch can easily be changed for higher or lower differential settings depending on the application. In addition to these controls, indicating lights **19** are provided to indicate demand at every instant. A bank of deferable load by-pass switches **21** are provided. The by-pass switches **21** allow individual loads to be removed from control for testing or maintenance purposes.

Three separate embodiments of the computer of the invention have been constructed and are designated as Models 1, 2 and 3 herein. These models accomplish the same result (continuous memory and monitoring of demand), however, slight variations in the memory circuit have been designed to provide versatility and economic options depending on application.

The Model 1 computer monitors a very slightly changing time interval as shown in FIG. 12. As T_o, T_o' and T_1 advance thru time, the power intelligence between T_o and T_o' is removed from memory Δt minutes successively. The Model 2 computer also monitors a very slightly changing time interval as shown in FIG. 12. However, as T_o, T_o' and T_1, advance through time, the power intelligence between T_0 and T_0' is synthesized during the Δt successive periods before removal.

The Model 3 computer monitors the demand time interval as shown in FIG. 9. The memory function of this computer is provided by a tape or wire recording system, or suitable mechanical time delay line.

Much of the prior discussion has been directed toward the new theory of demand sensing. However, load programming is also essential to the present invention. Four types of load programming systems are available for the computers of the invention. These four programs are not the only available ones but are presented here for possible requirements in various applications. The four programs are identified as follows:

1. Sequential priority load
2. The indefinite deferable load
3. Equal cycling priority load
4. Control limited deferable load

1. The sequential priority loads are devices which have a distinct precedence which will allow them to be assigned in increasing priority status. Example: hot water heating takes a lower priority than air conditioning or heating.

2. The indefinite deferable loads are devices that do not have a mandatory requirement for operation in a specific time period. Example: salvage operations which use electric furnaces to melt and prepare metal for re-use, snow-melting systems, etc. It differs from the lowest priority of sequential programming in that it is regulated at a second demand set point.

3. The equal cycling priority loads are devices which can be deferred as a composite frequently but individually cannot be deferred repeatedly, but require regulation from unit to unit (cycling). Example: multiple air conditioning units, multiple heating units, etc.

4. The control limited loads are loads that can be deferred but have control limits on the regulation. Examples of such loads are: air compressors, sump pumps, heating ovens, etc. The control limits can be time, pressure, volume, flow or any measurable limit. These devices must operate at a fixed set control point. Therefore, their operation as a deferable load depends on fixing a point of regulation above the mandatory regulation point which will cause the unit to cycle sooner as long as demand is below the preset point, but defer it if demand is above the preset point. But no matter what the demand, the regulation will be overridden at the load cover limit point.

The controllers of this invention are designed for monitoring the integral of some quantity continuously in any fixed or variable time interval and will provide control after comparison of this integral to a preset integral. The application of electrical demand control is used for illustration purposes only. The gas consumption technology on demand usage is similar except that the demand interval is usually larger, typically a 24 hour period. Therefore, the logic unit of the invention would remain the same except for time interval, preset demand, and demand differential adjustments. The sensing units and transducers would be units available from manufacturers in gas flow sensing technology.

Application for controllers of the invention can be found in any industrial process, laboratories, portable testing, and revenue economizing whenever continuous integral monitoring of some parameter is required. Measurements of temperature, pressure, volume, flow, density, etc., can be measured by sensors with applicable transducers for proportional millivolt input to the logic units of the invention.

The computers of this invention provide true "demand" monitoring and control for gas and electric utility revenue economizing. Furthermore, this continuous up-to-date average of any represented quantity of measure in a preselected time interval has many industrial process control applications to which the invention may be applied.

The continuous integral concept is achieved by the computers of the invention by using three similar but distinct methods. These three separate methods are embodied in each of the three computers which provide a wide latitude in matching the degree of control to the requirements of the application, as well as optimizing the cost versus benefits considerations.

Although each of the three models of computers provide a distinct variation in achieving the end result of a definite integral moving through time, many of the units are used in each system. Therefore, a general discussion of each system will be presented separately using a system block diagram to show the variations in system makeup as well as the interrelationship of each unit within the system. Each individual unit will then be treated separately to define its theory of operation.

The Model 1 computer achieves a continuous integral moving through time by acquiring data as a contiguous group of time defined increments registered as received in two units, one called the current rate register **24** and the other called the accumulator register **31** (see FIG. **13**). At the end of each time defined increment, the data accumulated in the current rate register is transferred to storage and the increment from the passing period is removed from storage and subtracted from the accumulator. This is a repetitive process producing a definite integral accurate to the value of one increment continuously registered in the accumulator. This method of computation is defined by the system block diagram of FIG. **13** and is designated the incremented integral.

The heart of the incremented integral system is the trigger control and gating unit **33** which monitors the activity of the system, receiving or directing commands to almost all of the other units in the system.

The current rate pulses are generated by the pulse generator **32** (PG) which vary in frequency as a function of the input sample voltage. The high speed trigger is generated by an astable multivibrator **34** operating at a frequency of 100 KHz. These two inputs to the trigger control and gating (TCG) unit **33** are directed by the TCG unit to the update register (UR) **27** and the accumulator register (ACC) **31** during alternate compute and update cycles.

Two general time cycles are generated by the TCG unit **33**, the longest of the two being the compute cycle during which time the current rate pulses (CRP) are directed to the current rate register **24** and accumulator register **31** to be accumulated. The second cycle is the update cycle, which is a very small increment of time by comparison to the compute cycle, during which time the high speed trigger (HSP) is gated to the update register **27** and the accumulator register **31** to subtract the past increment from the accumulator register **31** an amount equal to that contained in the update register **27**.

A read and a write command is also generated by the unit **33** to control the flow of data into and out of storage in the memory element **26** during the update cycle.

A compute inhibit command prevents the load control unit (LCU) **29** from making any decisions during an update cycle while the data contained in the accumulator register **31** is being updated.

The storage address counter (SAC) **23** maintains the proper sequential store and fetch addressing for reading and writing data into and out of storage.

The time base generator **22** maintains the proper time relationships for the various commands and cycles of the computer.

The programming decoder **30** continuously monitors the contents of the accumulator register **31** and generates a drop or pickup command which is sent to the load control unit **29**, which provides the logic functions and commands to decide what loads will be dropped or picked up and in what order this will occur.

The Model 2 computer is the same as the Model 1 computer except in the method of transition from one period to the next.

A synthesized flow is generated by the addition of a variable rate comparator 28. Switches S_{10}, S_{20}, S_{30} and S_{40} allow this unit to be connected for Model 2 operation. This flowing transition is accomplished by retaining the passing increment in the variable rate comparator 28 where it is modified by time to produce a synthesized past rate which is compared to the present rate occurring in the current rate register 24. As long as this comparison shows the present rate to be less than the synthesized past rate, through the incremental time difference from one period to the next, the data stored in the accumulator 31 remains static, neither acquiring or disposing of data until the end of the increment at which time an update cycle subtracts the past increment minus the current count from the accumulator 31 and begins a new comparison. If, however, the present rate begins to exceed the synthesized past rate the accumulator 31 will at once begin to accumulate at the present rate, doing so until the end of the increment when an update occurs. This system is defined by the system block diagram of FIG. 13 with switches S_1–S_4 closed and is designated the synthesized flow integral.

The Model 3 computer achieves a continuously flowing definite integral. It receives and records each bit of data in its specific real time position maintaining the actual time displacement of each bit throughout a monitoring period. The retention of data in this form produces a definite integral in a high degree of accuracy with a minimum manipulation of the accumulated data. This method is achieved by use of a time delay generator capable of generating long periods of delay at minimal cost. This system is illustrated in FIG. 14 and is designated the continuous integral.

FIG. 14 is a block diagram of the continuous integral system, which will be used as the reference in this explanation.

The sample voltage received by the pulse generator 32 in continuously variable analog form is converted by the pulse generator to pulses varying in frequency as the input voltage varies. These pulses are routed in two directions, one direct to the count controller 38 which generates an up count signal to the accumulator register 36 while producing a small time delay before counting the current pulse.

The second direction is to the time delay generator 37 which delays the pulse for the time period being monitored, i.e., 15 minutes, 30 minutes, 1 hour, etc. The output of the time delay generator 37, designated past pulse, generates a down count signal in the count controller 38 which is sent to the accumulator register 36. The past pulse receives a small delay before being sent to the accumulator register 36 to be counted down.

The decoded output from the accumulator register 36 is continuously monitored by the programming decoder to generate drop or pick up commands to the load controller 29 as directed by the programming of the decade switches 17 and 18 in the programming decoder 30.

The load controller 29 provides the sequencing logic for dropping or picking up loads as the demand integral and the programmed decoding dictate.

FIGS. 15–19 disclose in detail the Model 3 computer according to this invention (illustrated in block form in FIG. 14). For example, the voltage being sampled is applied between terminals 50 and 51 of the pulse generator 32 which is shown in detail in FIG. 15. The pulse generator 32 comprises an astable multivibrator 53 having two field effect transistors Q1 and Q2 connected in a standard cross-coupling network. The drain of Q1 is capacitively coupled to the gate of transistor Q2 and the drain of Q2 is coupled to the gate of Q1. Both gates are returned to ground through two separate constant current sources comprising the transistors Q3 and Q4. Transistor Q3 is forward-biased through the positive voltage divider variable resistor R8. This circuit provides an adjustable time base for calibrating the pulse generator to the appropriate CT ratios which are used to supply the sample voltage. The transistor Q4 is biased by the sample voltage at terminal 50 which is applied to its base through resistor R10. The sample voltage varies the charging current available to the transistor Q1 cross-coupling capacitor and thereby alters the timing of the astable multivibrator as a function of the output of the watts transducer. The resistor R9 provides temperature compensation for the pulse generator.

The output of the pulse generator is applied to terminal 52 which is connected respectively to the time delay generator 37, the count controller 38 and the load controller 29 as shown in FIG. 14.

As shown in FIG. 16 the time delay generator 37 consists of a tape or wire recorder 306 comprising a tape or wire 54 which passes over the reels 55 and 56 and which has read head 57, erase head 58 and write head 59. The input current pulse from pulse generator 32 is applied through the buffer amplifier 45 to the write head 59. After a time delay determined by the length of the tape 54 and the speed of the recorder, the current pulse is read out of the recorder by the read head 57 and sent to the count controller 38 through an amplifier 46. This pulse is identified as the past pulse signal. A 100 KHz oscillator 61 supplies an output through amplifier 47 to erase head 58 so as to continuously erase the pulse information on the tape before it passes the write head 59.

The count controller 38 receives the past pulse and the current pulse which are fed to an exclusive OR gate G1. The past pulse is also applied to an AND gate G2. The current pulse is applied to AND gate G3. The AND gates G2 and G3 receive the output of the exclusive OR gate G1. A flip-flop circuit F1 receives the output of the AND gates G2 and G3 and provides a countdown output to terminal 62 and a countup output to terminal 63. An OR gate G4 receives the output of AND gates G2 and G3 and supplies an output to a monostable multivibrator 64. An accumulator pulse is supplied to terminal 65 at the output of the monostable 64.

If either a past pulse or a present pulse arrive at the exclusive OR gate G1 an enable trigger is generated at gates G2 and G3. If, however, both pulses arrive simultaneously no enable is generated and the two pulses cancel each other without any action occurring at the accumulator register. The outputs of G2 and G3 go to the flip-flop F1 and to an OR gate G4 which triggers the monostable 64. If a past pulse arrives from G2, the flip-flop will set, which provides a countdown command to the accumulator 36 and at the same time triggers the monostable multivibrator 64 to delay the past pulse from going to the accumulator until the count command has been generated.

If a current pulse arrives from gate G3 the reverse process occurs, resetting the flip-flop F1 and generating a countup command while delaying the pulse via the monostable multivibrator 64.

FIG. 17 illustrates the accumulator register 36 and input terminals 62, 63 and 65 correspond to these terminals in FIG. 16. The accumulator register comprises three divide by 10 updown counters 66, 67 and 68, respectively, each of which receive the up and down count commands from terminals 62 and 63. Counter 66 receives the pulse from terminal 65 and supplies an output to counter 67 which in turn supplies an output to the counter 68.

The indicator lamps 19 illustrated in FIG. 11 are respectively connected to the counters 66, 67 and 68 through the indicator driver amplifiers 70*a* through 70*L*. The count contained in the accumulator registers is continuously decoded in the BCD decoders 71 and 72 and sent to the programming decoder 30 through the data lines 72*a* and 72*b*. The indicator drivers 70*a* through 70*l* provide the amplification and buffering between the accumulator and the indicator lamps on the control panel of the computer illustrated in FIG. 11. The count contained in the accumulator represents the accumulated integral of the time period in use adding and subtracting on a continuous basis as commanded by the past and current pulse.

FIG. 18 illustrates the programming decoder 30 which receives the data lines 72*a* and 72*b* from the accumulator register 36. These are decoded equal to or greater than signals from the accumulator registers TNO most significant digit BYTE counters. The inputs from the most significant digit are termed 000 through 900 and labeled 72*b* and the inputs from

the next most significant digit are labeled 00 through 90 and indicated by 72*a*.

A switch S1 comprises a five section 10 position switch (five pole 10 position). The input to Section A (SIA) is routed to the drop decoder to provide a 10 KW per step change in the drop decoder programming. The other four sections of switch S1, B, C, D and E are each offset by 20 KW from each other and the signal sent to the drop decoder. These provide negative offset differences from 20 to 80 which are routed to switch S3, Section A.

Switch S2 is a four-section 10-position switch (four pole 10 position) and the input to Section A (S2A) is routed to the drop decoder to provide 100 KW per step changes in the drop decoder programming. Section B is offset from A by plus 100 KW and is routed to the drop decoder to provide an override of the 10 KW steps in the decoder programming whenever the programmed 100 KW step is exceeded by the next higher amount.

Sections C and D of switch S2 provide a negative offset in 100 KW steps below the drop decoder programming and are routed to switch S3, Sections B and A.

Switch S3 is the pickup differential programmer with outputs going to the pickup decoder **80**. Switch S3 is a two-section 10-position switch (two pole 10 position). Each of the 10 positions of the switch S3 provide a 20 KW step change from −20 to −200 for programming the pickup decoder.

The operation of switches S1 and S2 programs the drop decoder in 10 KW steps from 10 to 990. When the accumulator count registers any value equal to or greater than this programmed count, the drop decoder **81** detects and generates a drop command which is applied to the load control unit **29**. The pickup decoder **80** detects whenever the accumulator registers below the drop point by the amount programmed by switch S3 and generates a pickup command for the load control unit **29**.

With the programming decoder set at the values shown in FIG. **18**, the peak limit at which a drop command would be generated is determined by the A section of switch S1 and the A section of switch S2 which in this case is 200 and 10 or a setting for 210 kilowatts limit. If the contents of the accumulator registered 200 or over the A contact output, shown as 0 for reference to the program setting would be positive and enable one input to the gate **82**. If the accumulator count in the tens digit was 10 or over, the other input to gate **82** would be positive and the output of gate **85** would be positive which is the condition desired when a drop signal is active. If the accumulator count should for some reason exceed 300 the plus line would be positive and override the gate **82** input to gate **85** through amplifier **83**. This prevents a dropout of the gate **82** enable as the tens signal passes through a lower than programmed value although the high order digit is above its setting.

With the programmed decoder set at the value shown in FIG. **18**, the pickup command would be generated for any value less than 190 in the accumulator register. The switch S3 A section contact picks up the signal designated −20 which comes from the E contact of switch S1. This is the **90** line output from the accumulator and will be low for any value less than 90 in the tens digits. Inverted by inverter **84** it will enable one-half of gate **87**. The S3 B section contact picks up the **0** line from Section A of switch S2 which will be low for any value less than 200. This is inverted by inverter **86** to complete the enable of gate **87** and generate a positive pickup gate. By changing the setting of switch S3 any offset value lower than the program limit can be established in 20 KW steps from 20 to 200.

FIG. **19** illustrates the load controller **29** which is comprised of four flip-flops **91**, **92**, **93** and **94**, eight NAND gates **95** through **102**, 11 inverters **103–112** and four load control relays **113–116**. Gate **95** receives three signals: the compute inhibit which is in the enable state except during a compute update cycle; a drop command which is active only when the programmed limit is exceeded; and, the current pulse from the pulse generator terminal **52**. If all three inputs are active, flip-flop **91** will set generating an enable gate for the next higher priority gate **96** and an activate signal to relay **113** whose contacts will open to disconnect the load **117** connected to them. If the flip-flop **91** is already set the drop command is gated to flip-flop **92** via gate **96** and so on as each higher priority is activated.

Gate **102** receives three signals, a pickup command from terminal **88**, compute inhibit and ΔT triggers. The first ΔT trigger following a pickup enable will reset flip-flop **94** which will enable the next lower priority gate **101** and activate relay **116** to pickup load **120**. The pickup action operates in reverse order of picking up each relay until the pickup command ceases.

Thus, it is seen that the Model 3 computer illustrated in FIGS. **14–19** is a continuous integral system which provides sequencing logic for dropping or picking up loads as the demand integral and the program decoding require.

The Models 1 and 2 controllers illustrated in FIG. **13** will now be described.

The Model 1 system is described in FIGS. **13, 15, 18, 19, 20, 21, 22** and **24**. The Model 2 system is described with these figures plus the addition of FIG. **23**.

The trigger control and gating unit **33** illustrated in FIG. **20** provides a timing logic and gating function which controls the update and compute cycles, trigger switching and read-write timing of the memory storage. A four state controller called the cycle generator provides a sequence and timing control of the update cycle. Combinations of logic gates and inverters develop the control commands.

The start of an update cycle begins with a ΔT trigger at terminal **121** which clears the four state cycle generator comprised of the flip-flops **122** and **123** and the gates **124–130** which set the flip-flops **122** and **123** to the 00 state.

In the 00 state, gates G**130** and **128** provide positive gates to enable gate **124**. The output of flip-flop **122** at the No. 1 output, complement enable, will gate **131** and inverted by inverter **132** the accumulator complement enable.

With gates **124**, **131** and the accumulator complement enabled, the next high speed pulse at terminal **133** from oscillator **34** will pass through gates **131** and **136** to complement the contents of the accumulator register and through gate **124** and **126** to toggle the cycle generator to the next state, 01.

In the 01 state, gate **128** will disable gate **124**, inverted by inverter **140** it will enable gates **141** and **142**. The high speed pulse at terminal **133** will pass through gates **141** and **136** to down count the accumulator and through gate **142** and **143** to down count the update register. All down counting is actually accomplished by up counting the binary complement of the data to be counted down.

This period of the update cycle will continue until the update register is full indicating the down count is complete. The UR full command enables gate **144**, the output of gate **144**, UR full disables gates **142** and **146**, enables gate **125** via gate **147** and enables gate **148** through inverter **149**. The next high speed pulse will now pass through gates **125** and **126** to toggle the cycle generator to the 10 state.

In the 10 state, gate **129** maintains the enable gate **125** through gate **147**, enables gate **152** through inverter **300** and gate **148** through the inverter **151**. The flip-flop **122**, one output, again enables the gate **131** and the accumulator for a complement function, as soon as gate **129** enables gate **148** the update register and the current rate register are cleared by the output of gate **148**. This drops the UR full command and disables gate **148**.

With gate **125** and gate **152** still enabled the next high speed pulse will toggle the cycle generator to the 11 state, generate a read pulse, which takes the next passing increment out of storage putting it into the update register and the past rate log register of the variable rate comparator.

In the 11 state, gate **130** disables gates **124** and **125** inverted by inverter **152** it enables gates **154** and **155** and puts the compute inhibit line in the enable state. The cycle generator will

remain in this state as the compute cycle until the next ΔT trigger.

The current rate register pulse is controlled by gates **156** and **157** and inverter **158**. As long as the current rate register is not full, gate **157** will pass the current rate pulses through inverter **158** to the current rate register. The CRR pulse is therefore always sent to the current rate register **24** no matter what cycle the computer is in. This is of no conflicting consequence since the rate of the high speed pulse processes the update function in so short a time compared to the slow rate of the current rate register pulses.

The accumulator pulse output from gate **136** is controlled by three gates **160**, **131** and **136**. Gate **160** will pass the current rate pulses if the accumulator register is not full and the cycle controller is in the compute state 11 and the count enable from the variable rate comparator is positive. Gate **131** as previously described, passes the complement toggle trigger during the update cycle. Gate **141** passes the high speed pulses to the accumulator during the update cycle as long as the ACC is not full and the cycle controller is in the 01 state.

The update register pulse output at gate **143** is controlled by the two gates **142** and **146**. As long as the UR is not full or the count enabler is not positive during the compute cycle, gate **146** will pass the CRP pulses. Gate **142** is active when the update register is not full and the cycle generator is in the 01 state. At this time it passes the high speed pulses to the update register.

Gates **152**, **154** and **161** and inverters **162** and **163** perform the read-write decoding. At the end of the compute cycle while the end of the compute inhibit is positive the ΔT pulse is passed by gates **154** and inverter **163** as the write command and gate **154**, **161** as the read-write enable command. After ΔT when the cycle controller is in the 10 state gate **152** will pass one high speed pulse through inverter **162** as the read command and gate **161** as the read-write enable.

The compute inhibit output is positive throughout the compute cycle. The cycle generator is locked in the 11 state. The output from gate **130** is inverted by inverter **153**. During the update cycle the compute inhibit becomes negative.

The update register and the current rate register are cleared simultaneously at the end of the update cycle. When the full command from the update register (UR FULL) is received indicating that the update register has a full count in it the output of gate **144** goes to 0. This provides an enable to gate **125** through **147** and allows the high speed pulse to clock the cycle generator to the 10 state. The 10 state decoded by gate **129** maintains the gate **125** enabled through the other input to gate **147**. It is also inverted to enable gate **148** by inverter **151**. The second enable to gate **148** is the inverted UR FULL. As soon as the cycle generator is clocked to the 10 state by the first high speed pulse after the UR FULL command gate **148** decodes a clear pulse to the UR and CRR. When the UR clears the UR FULL command drops and the UR and CRR clear is terminated. This action takes place at such a speed the UR is cleared before the next high speed pulse which will clock the cycle generator to the 11 state, and therefore the **10** line can also enable the read command through gate **152**.

FIG. **21** illustrates the current rate register **24**, the update register **27**, and the memory element or storage **26**. The current rate register is composed of two binary **16** counters **165** and **166** in series, and 10 gates identified by numerals **167–176**. The current rate register pulses from the trigger control and gating unit **33** are counted up through the compute cycle and cleared for a new count during every update cycle. Gates **167** and **168** decode a full state to inhibit CRR pulses in the event the counter reaches the full state before the end of the compute cycle. This prevents an overfill which would pass through zero.

The output of the CRR is gated to the bit lines **177** by gates **169–176** on the write command from the trigger control and gating unit **33** at the end of each compute cycle. The output of the CRR is sent on the bit lines to the storage element **26** in complement form to be stored at the location decoded by the decoding word drivers in the storage element.

The update register **27** is comprised of two binary **16** counters **178** and **179** in series and 10 gates **180–189**. The UR receives a preset count from the bit lines on the read command at the end of the update cycle.

The update register pulse (URP) received from the TCG33 is counted up from the preset count that was removed from storage via the bit lines. The count up occurs until the UR is full at which time a UR FULL command is sent to the TCG33.

The storage element **26** contains 16 eight-bit storage addresses of which only 15 are used. Data to and from the storage element **26** is transferred via the bit lines through the read and write buffers on the read or write command. The address is sequentially changed each ΔT trigger. Moving through the 15 addresses in a continuously repetitive sequence, these addresses are decoded by the decoding word drivers from the address lines. The read and write buffers are identified by numerals **190** and **191**, respectively.

The storage element **26** has two 64-bit random access chips which contain the address decoding, storage array, and read/write buffers.

FIG. **22** illustrates the time base generator and storage address counter. The time base generator receives a 60 hertz sample signal from the power supply which is fed to the inverters **192** through **195** to form a squaring circuit by over-amplifying and clipping the AC signal. Flip-flops **196** and gate **301** form a count to 15 counter. Flip-flops **197** and **198** count by 4. The combined count of flip-flops **196–198** is 60 and the output of **198** is 1 second. Flip-flops **199** to **201** form a second count to 60 counter providing three time outputs, 15 seconds apart decoded by gate **203**, 30 seconds by gate **204** and 1 minute by gate **205**. The force to clear action of gate **203** allows the four flip-flops **199** to remain in the decoded **15** state for only a very short time thus providing a trigger every 15 seconds which is then gated by gate **206** as ΔT and **204** as ΔT_n. The strapping terminals for ΔT and ΔT_n are provided for flexibility in increment timing as may be required by the application.

The storage address counter **23** is comprised of four flip-flops **207** and gate **208**. These are configured to count to 15 and clear, which provides a new output on the address lines **209** repeating itself every 15 ΔT's.

FIG. **23** illustrates the variable rate comparator **28** which is used only in the Model 2 configuration. The variable rate comparator **28** receives two types of data; the full value of the passing increment in the past rate log register and the current value of the present increment as it occurs in the current rate increment register **24**.

The data in the past rate log register is shifted each ΔT_n trigger while the current rate register is reset to acquire a new count each ΔT_n at the new current rate. This process adds greater and greater values to the past increment data and less and less weight to the present increment data as time progresses through the present increment.

The equal or greater decoder compares the five highest order bits of the past rate log register to the five bit contents of the current rate register. If the value of the CRI equals or exceeds the value of the PRL at any time, a count enable will be generated and sent to the trigger control and gating unit **33** to allow the accumulator register **31** to count up.

The current rate increment counter register comprises the gates **210** and **211** and registers **212** and **213** respectively. The past rate log shift register is a shift register composed of 11 flip-flops. The first eight flip-flops are preset from the data on the bit lines at the end of the update cycle having already been cleared by the ΔT trigger at the beginning of the update cycle. The data read into the past rate log register is shifted to the right each time a ΔT_n trigger occurs. There are three ΔT_n triggers which occur at ¼, ½ and ¾ of the ΔT time value.

Each shift doubles the amount of the input data that is to be compared to the current rate. This value is therefore the least at the start of a compute cycle and grows larger as it approaches the end of the cycle.

FIG. **24** illustrates the accumulator register **31** which comprises a binary coded decimal up-down counter which uses a

weighted 5211 BCD code and synchronous triggering of the flip-flops. Each digit of the accumulator contains four flip-flops A, B, C, and D, designated by numerals 220–223 and five gates designated by numerals 224–228. Two inputs from the trigger control and gating unit 33 go to the least significant digit, the ACCP and the complement enable, the higher order digits receive the complement enable and the carry output from the next lower order digit.

The accumulate decoder comprises gates 229–242 and inverters 243–251 and decodes the contents of the accumulator register as equal to or greater than signals to be sent to the programming decoder. The nine outputs of the accumulator decoder are returned to the trigger control and gating unit 33 as the accumulator full command (ACC FULL).

In a typical configuration, three units such as illustrated in FIG. 24 would be utilized to make up a three digit accumulator with the decode outputs of the least significant digit unused except for nine and the carry output of the most significant digit unused.

The basic difference between the Model 1 and Model 2 is provided by the addition or deletion of the variable rate comparator (VRC) and the associated lines to the VRC. Model 2 contains the VRC. Model 1 deletes it.

The difference in action by the use or non-use of the VRC is essentially how the update function will subtract the past increment from the count contained in the accumulator register. Without the VRC, the past increment is counted down in one lump sum at the start of the present increment, by placing the past increment in inverted true binary form in the update register, and counting down the count in the ACC while counting down the count in the update register until it generates a full command to the TCG to stop the update process. This removal of the total past increment at the start of a new increment requires the load controller to be programmed for pickup using the ΔT trigger rather than the current rate pulse so that a full increment will pass before a load will be restored and then only one load per ΔT is allowed to be restored.

When the VRC is used, the past increment is placed in the update register in inverted true binary form and also sent to the VRC. As the current rate proceeds through the current increment, the update register is counted down at the current rate and no pulses are sent to the ACC. This is equivalent to subtracting the present count from the past count at the present rate. However, if the VRC detects a surge it will generate a count command to the TCG which stops the down count in the update register and activates an up count in the ACC. The remainder contained in the UR will then be subtracted from the ACC at the end of the present increment in the same manner as previously described.

When the VRC is used it is not necessary to wait a full increment to restore loads, therefore the load controller is programmed for load restoring with the current rate trigger rather than the ΔT trigger. This provides a restoration time of much shorter duration.

It is seen that this invention comprises a new and novel computer of various modifications in which substantial savings of power rates may be obtained by preventing excessive power charges due to the continuous monitoring by the computer.

Although the invention has been described with respect to preferred embodiments it is not to be so limited as changes and modifications may be made which would be within the full intended scope as defined by the appended claims.

We claim:

1. A demand computer for continuously accumulating a quantity over a fixed time interval and said interval being divided into a fixed number of lesser time intervals designated delta T increments and said fixed time interval having starting and ending delta T increments which are continuously changing as the fixed time interval moves through time, comprising:

a pulse generator means receiving a variable voltage indicative of the quantity and converting it into pulses whose frequency is indicative of the rate of usage of said quantity;

a time base generator means for converting 60 cycle timing signals into short term timing intervals designated delta T increments;

means for counting said delta T increments to define a demand period as a fixed number of delta T increments receiving an input from said time base generator;

an accumulator means for storing the integral of said quantity which can be continuously altered by removing pulses representative of the rate of usage of said quantity during an update timing period or acquire pulses representative of the rate of usage of said quantity during an alternate compute timing period, said accumulator having contents at any moment in time equal to the sum of all pulses generated by the pulse generator over the demand period from the present delta T increment through the past fixed number of delta T increments defining the demand period;

a current rate register means for storing the number of pulses developed by the pulse generator in one delta T increment and capable of holding the maximum expected number of pulses until a new delta T increment begins;

a memory element composed of solid state memory devices divided into discrete storage addresses, each address of which is capable of storing the maximum value that can be acquired by the current rate register;

means for entering into the enabled address of the memory element on a write command the data stored in the current rate register and means for removing from the memory element on a read command the data stored in said memory element from the address which is enabled and means for storing said data removed from said memory element in an update register;

a storage address counter means for counting the delta T increments such that a full count is equal to the fixed number of delta T increments in a demand period and means for decoding the count in the storage address counter for enabling one discrete address in the memory element for each count value in the storage address counter;

an update register means for receiving data stored in the memory element equal to the amount acquired in one delta T increment and having been stored in the memory element for the length of one demand period minus one delta T increment and having means to decode a full state and generate a control gate to control pulses to the accumulator register;

a trigger control and gating means receiving the output of said pulse generator and supplying an input to said accumulator, said current rate register and said update register, and including means for generating read and write commands to control said memory element and further including means for generating timing cycles for compute and update functions as required to maintain a continuous integral in said accumulator register;

a programming decoder means monitoring the contents of said accumulator and initiating a control command when said contents exceed a preset limit and also containing means for presetting an offset value lower than the preset limit which will initiate a pickup control command when the contents of said accumulator drop below the preset limit minus the offset value; and

a load controller means connected to said programming decoder for receiving pickup and drop commands from said programming decoder and sequentially transferring said commands into open or closed relay contacts to control peripheral equipment.

2. A demand computer according to claim 1 wherein said accumulator means includes decoding means for continuously decoding the binary contents of said accumulator to provide a decimal output.

3. A demand computer according to claim 1 further including a variable rate comparator receiving inputs from said pulse generator, said memory element and said time base generator,

and supplying an output to said trigger control and gating means.

* * * * *