A DESCRIPTION OF THE ELECTRONIC COMPUTER AT THE INSTITUTE FOR ADVANCED STUDIES

By

Gerald Estrin
Institute for Advanced Studies, Princeton

In June, 1946 John von Neumann, Herman H. Goldstine, and Arthur W. Burks set forth logical principles on the basis of which they proposed to construct a general purpose computing machine at the Institute for Advanced Study in Princeton, New Jersey.

Something less than six years later the results of labors directed by Julian H. Bigelow bore fruit. Twenty-three hundred vacuum tubes cooperated to solve the machine's first major problem at the beginning of this year.

My allegiance to the Institute machine goes back two years when the memory and arithmetic units were first joined in wedlock. Throughout the six years a group of about five engineers labored to execute the logical design. Some of the engineers who were members of the Princeton group from time to time and many of the design ideas in the Institute machine may now be found at computing centers all over the globe.

Since this is one of the first papers describing the Institute machine, I will take just a moment to mention some of the people for whom I am reporting.

Aside from those mentioned above the present engineering group consists of J. H. Pomerene, E. Frei, H. D. Crane, G. Kent, R. W. Melville, N. Emslie and the reporter.

Some of those who are not now with the Princeton group but whose names may be familiar to you are: R. Slutz, W. Ware, J. Rosenberg, M. Rubinoff, J. Davis, T. Hildebrandt, and R. L. Snyder.

Since the beginning of the year, the Institute machine has been used on several problems with most of the computing time going to the solution of meteorological problems formulated by a group led by J. G. Charney.

The 1946 von Neumann report set the following requirements for a high speed general purpose computer:

1) Produce four basic organs: Memory, Control, Arithmetic, and Input-Output for a 40 binary digit, parallel, general purpose machine.

2) Produce a parallel memory which can store 4000 forty digit words with an access time of less than 50 μ.s. Let this memory store both orders and numbers with the orders expressed by numerical code.

3) Produce a control organ which can differentiate between numbers and orders and which can interpret and command the execution of a vocabulary of 21 orders without human interference. Let each 40 digit order be an order pair, each half containing a 12 digit address and an eight digit instruction.

4) Produce a parallel arithmetic organ with the ability to execute binary addition, subtraction, multiplication, and division. To save memory capacity, use a fixed binary point and place the burden of scaling on the mathematician.

5) Produce an Input-Output organ which can bridge the gap between the slow speed human and the high speed internal memory. The initial suggestions were teletype tape and magnetic wire installations.

The von Neumann report went into greater detail in the discussion of the logical design of the Institute machine but I repeat the above merely as a background from which to view its eventual organization.

In the remainder of my discussion, I shall describe the organization of the Institute machine as it now stands and shall attempt to avoid enmeshing you in the mass of detailed circuitry.

Figures 1 and 2

These illustrations show two views of the Institute machine. The row of circular apertures at the bottom mark the positions of the 40 high speed memory cathode ray tubes. The upper portion contains the parallel accumulator, arithmetic register and memory register on the left and the adder and control circuitry on the right.

Figure 3

Schematic of Basic Interconnections. This slide is a schematic establishing the physical layout of the Institute machine and displaying the interconnections between the Memory, Control and Arithmetic Units.

The large block at the bottom contains the 40 cathode ray tubes (5CP1A), 20 on each side. Each memory tube has an associated three stage amplifier and an eight tube discriminator unit which controls the beam turn-on.

The upper portion contains the arithmetic and control organs. On the left there are three double rank registers each containing two rows of 40 flip-flops and two rows of gate tubes for entry into them. The left most stage contains the most significant digit. This is chosen as the sign digit with the binary point fixed to the right of it.

The register labelled RI is the Accumulator. It is a double rank shifting register which holds the Resident Number and the Final Sum during addition; the Dividend and Partial Remainder during division and the most significant digits of the Partial Products during multiplication.

RII is the Arithmetic Register. It is a double rank shifting register which holds the multiplier and the least significant digits of the partial products during multiplication and the quotient during division.

RIII is a non-shifting Memory Register. In the upper row of 40 flip-flops it receives numbers from the memory and presents them to the adder via the 40 complementing gates. It contains the Incident Number during addition, the Multiplicand during multiplication and the Divisor during division. In the lower row of flip-flops it receives order-pairs from the memory -- each half containing a ten digit address and a ten digit instruction.

On the right the row of Adder chassis contains 40 parallel circuits each receiving three inputs; the resident digit from RI, the incident digit from RIII via the complement gates and a carry digit from the next less significant stage. Since each of these inputs may be a one or a zero, there are eight possible combinations entering each stage of the adder. The adder circuit contains four tubes which propagate a carry to the next stage and reduce the eight possible inputs to four possible voltage levels. The row of chassis above the adder is called the digit resolver and reduces the four voltage levels coming out of the adder to two voltage levels representing a "1" or a "0". This information is then communicated to RI.

The lower row of chassis comprises the control organ containing four major units, the Main Control, the Shift Counter, the Address Counter, and the Memory Control.

1) The Main Control establishes the sequence for observing the two halves of any order pair and bringing up a new order. It decodes the order digits and commands the execution of the resulting processes. It is completely asynchronous. It does not go through the expense of matrix decoding, but rather utilizes unique combinations of two or three order digits to command the desired processes.

2) The Shift Counter counts the number of shifts in the shifting registers and contains recognition circuits which determine when shifting shall be terminated appropriately to the order being done.

3) The Memory Control with its associated Pulse Chain establishes regeneration, reading and writing routines under the command of the Main Control.

4) The Address Counter remembers the locations of the last point regenerated on the face of the cathode ray tube and of the last order-pair executed; advancing these addresses by one each time. It is also capable of receiving from RIII the address associated with any instruction.

Figure 4

Machine Characteristics.

Overall

The primary tube types are the

6J6, 5670, 6AL5, and 5687.

Arithmetic

The addition time of 62 μ.s. is the measured average time for a routine containing a long sequence of summation orders. It therefore includes memory access, a full regeneration and half the time to bring up a new order pair. A less realistic though meaningful figure is the interval from presentation of resident and incident numbers to delivery of the final sum to the accumulator which is of the order of ten μ.s.

The multiplication time is conditional on the number of one's and zero's in the multiplier. The 500 μ.s. time is for a half-half mixture.

The division time of 800 μ.s. is unconditional upon numbers processed.

Design Criteria

Finally a description of overall machine characteristics would be incomplete without a statement of the design criteria which have time and again demonstrated their validity.

The logical circuitry is designed to operate satisfactorily if the resistors do not vary more than $\pm$ 10 % from their original value, each in the worst direction.

The logical circuitry is designed to operate satisfactorily if the cathode emission of the tubes does not drop below 50 % of the manufacturer's rating.

The vacuum tubes are designed to operate at half the manufacturer's plate dissipation rating.

All circuits with exception of the Williams Tube Amplifiers are direct coupled.

Figure 5

Double Rank Shifting Registers (RI and RII)

The double rank shifting register illustrated here is used in both the Accumulator, RI and the Arithmetic Register, RII. The register consists of two rows of 40 toggles each. There are two diagonal gates which can shift to the left and right, respectively, from the temporary or upper rank to the main or lower rank. There are two vertical gates into the upper rank, one of which is used for communication with another part of the machine. A typical shifting operation would start with meaningful information in the lower rank -- clear the upper to "1", gate up zero's, clear the lower to "1", shift down left zero's. The 40 clears and gates are operated in parallel. Note that at no time is the information purely "in transit."

It should be observed here that there is no vertical down gate. Hence when information is communicated to the upper rank via the dashed line gate it can be delivered vertically down only by means of a zig-zag operation -- down left -- up -- down right.

Figure 6

This illustration demonstrates the use of the shaded vertical up gates in RI and RII.

In the Accumulator, RI, the contents of the lower rank are always statically presented as an input to the adder. The adder also receives statically the contents of the upper rank of the Memory Register RIII via the complement gates. The sum is statically resolved by the Digit Resolver and presented as the input to the shaded gate into the upper rank of the Accumulator RI. The information is stored in RI when that gate is enabled.

In the Arithmetic Register, RII, the shaded up gate receives the contents of the upper rank of the memory register.

Figure 7

The Memory Register RIII is also of the double rank type but does not have shifting properties. Information is presented from the memory to gates into both the upper and lower ranks of RIII. The choice of gate is dictated by the control circuits, numbers being gated into the upper rank and orders into the lower rank. Numbers may also enter the upper rank from the lower rank of RII, the Arithmetic Register via the right hand gates. The lower right hand set of gates is used to communicate instruction digits to the main control and address digits to the shift and address counters. This fourth set of gates unlike any of the others described is enabled in groups of ten instead of 40 to separate address digits from instruction digits and the first order-half from the second.

The set of complement gates at the top is capable of taking the information from the upper rank of RIII and either transferring, complementing, or nullifying that information before it is presented to the adder.

Figure 8

Kirchoff Adder. The essence of the Static Kirchoff Adder is illustrated in Figure 8.

A carry arriving from the previous less significant stage establishes the voltage level at the top of the 10K resistor at 200v for a "0" or 150v for a "1". If either the Incident or Resident digit is a "1" 5 m.a. is drawn through the summing resistor to produce an additional 50v drop at the bottom. The carry gate compares the summing point level against a standard 125v and propagates a carry on to the next more significant stage. The four voltage levels derived from the eight possible input combinations appear at the summing point and are then presented to the Digit Resolver. There they are separated and reduced to two voltage levels which are then presented to the gates into the upper rank of the accumulator, RI.

The adder circuit requires four double triodes per stage with a similar number used in the digit resolver. The time for a carry to propagate or decay through 40 stages is from eight to ten μs.

Figure 9

Memory Block Diagram. This slide illustrates the general organization of the Memory.

When the memory is not being called upon by the Main Control to deliver or receive information, it sequentially regenerates the information stored on the 32 x 32 raster of each 5CP1A cathode ray tube. This is accomplished under the dictate of the Memory Control and Pulse Routine Generator. A clock issues a pulse

every 24 μ.s. and initiates a pulse chain created by ten pulser units. These pulses are routed by the Memory Control to turn the beam on at an address specified by the Regenerate Rank of the Address counter; to observe and store in a toggle the amplified output of the CRT screen signal; to turn the beam on to restore a dot; to displace the beam if a dash is to be restored; to turn the beam on to restore the dash; and to clear and gate from one rank of the address counter to another and back. The address is presented to a balanced deflection generator which raises the deflection signal to a 1200 volt level and applies it to the CRT deflection plates.

At one moment during each clock cycle the memory control is receptive to intervention by the main control. The main control may then specify that the memory bring up the next order of a consecutive series into the lower rank of RIII utilizing the address recorded in the order rank of the address counter or it may depart from this series to specify that a number or order be brought out from an address residing in either order half in RIII or it may specify that the number in the accumulator RI be stored in the memory at the address residing in either half of the order in RIII.

For the case where new information is to be stored in the memory special beam-turn-on pulses are utilized whenever a dot is to be replaced by a dash or vice versa.

To establish the beginning and end of any memory process commanded by the main control, two feedback pulses are issued by the Memory Control. It acknowledges receipt of the Main Control intervention at the start and the pulse which finally delivers information into RIII is fed back to establish the completion of the memory cycle.

Figure 10

The address counter summarily described in connection with the previous slide is an adder type counter in which one is added to the previous count and a full carry is allowed before the next count may proceed.

The shift counter illustrated in this slide is a scaling type counter used to count the number of shifts in any arithmetic process. It is a double rank counter in which the lower rank holds a "true" count and the upper rank contains a "false" or non-monotonic function of the "true" count. When the up gates are enabled they deliver the contents of the true rank stage into the false rank.stage. When the down gates are enabled they transfer the inverse of the false rank stage into the true rank stage. The up gates of the n + 1st stage are enabled whenever the lower nth stage contains a "1". The down gates of the n + 1st stage are enabled whenever the lower nth stage contains a "0". The up and down gates of the least significant stage are enabled sequentially by feedback signals from the shifting registers and will change the state of that stage with each shift.

The counter has six stages permitting a count up to 63 but a modified sixth stage only permits recognition of a maximum of 47 shifts. A count of 41 is the maximum required for any of the arithmetic operations.

The count in the true rank may be compared with the address in either order half of the memory register, RIII or with a count specified by the main control. Whenever a coincidence exists between the "1's" in the count requested and the "1's" in the "true" rank of the shift counter, a counter satisfaction signal is issued to the main control.

Figure 11

Timing Chain. The key point in rapid and reliable arithmetic operations is the clear-gate timing chain, an example of the control's asynchronous operation.

A sequence of clear-gate operations is always initiated by the coincidence of the go-ahead state of a toggle in the main control and the fact that the shift counter has not achieved a number to be recognized. The choice of the preclear of the upper rank is previously established and the clear invitation is issued. When the register clear bus drops down in level it also feeds back to change the state of a timing toggle in the chain which in turn invites the following gate request. The up phase clear and gate requests also act as the enabling signals into the first stage of the shift counter. The choice of the left or right diagonal down shift is made before the up gate is completed and the drop in level of the register up gate bus is fed back to request the diagonal preclear invitation. In like manner the drop in level of the diagonal clear bus sets a timing toggle which in turn produces the diagonal gate request. If the count to be recognized was not achieved during the up gate, this cycle is repeated with the choice of up phase clear being made during the down phase. In this manner prototypes of the toggles being cleared and gated into control the timing process itself and permit it to proceed at maximum speed.

Figure 12

Execution of a Summation Order.

(1) The main control observes the order in the lower rank of the memory register, RIII.

(a) An "action" request is issued to the Memory Control and conditions are established whereby the memory will "read" the number at the address in the first ten digits of the order half being observed and will transfer that number into the upper rank of the RIII.

(b) The complement gates are set for either addition or subtraction of magnitude or number.

(c) The condition is established to permit a 12 μ.s. delay after receipt of the incident number by the Memory register.

(d) The clear selection circuitry is instructed to execute a "zig-zag" operation of two shifts in order to bring the sum of the resident (R_1) number and the incident (R^3) number from the adder into the upper rank of RI and then get it vertically down.

(e) The recognition circuitry of the shift counter is instructed to stop the timing chain after the count of "2".

(2) The memory control accepts the request for "action" and issues a "sync" signal to the main control.

(a) The pulse routine then proceeds to demand execution of the process in (1)(a).

(3) The Main Control uses the "sync" signal to:

(a) Clear the shift counter to "0".

(b) Clear the accumulator if so requested by the order being observed.

(c) Prevent any further action requests during the period in which this order is being executed.

(d) Prepare for observation of the second order half.

(4) When the memory control operates the gates into the upper rank of RIII, it thus indicates the completion of the memory cycle.

(5) The main control uses the indication of (4) to issue a "go ahead" signal to the timing chain conditional on the completion of the 12 μ.s. carry delay.

(6) The timing chain demands the execution of the "zig-zag" shifting operation.

(7) After the second "up" gate, the shift counter recognition circuits signal "counter satisfaction" to the main control.

(8) The main control:

(a) withdraws its "go ahead" signal to the timing chain, preventing any further shifts.

(b) terminates observation of the first order-half

(c) requests observation of the second order-half.

Figure 13

Execution of the Multiplication Order. The starting and terminating processes of the multiplication order are the same as those previously described for the summation order.

The multiplier is previously established in the lower rank of the arithmetic shifting register, RII. The multiplicand is delivered into the upper rank of the memory register. The least significant digit in RII decides whether or not the multiplicand is to be added into the partial product in the accumulator. The partial product and the multiplier are then shifted right. By initially establishing the sign of the multiplicand in the accumulator and by propagating that sign digit during each right shift, the cases of either positive or negative multiplicand are executed correctly with the exception of a final correction of 2^{-39} which must be added to the final product if the multiplicand is negative.

If the multiplier is negative, the multiplicand must be subtracted from the final product.

These two corrections are accomplished in 40th and 41st steps during which the accumulator is shifted to the left and right respectively in order to maintain correct positioning of the most significant digits of the final product. During the left shift operation the contents of RII are not shifted. Hence at the end of the 41st step, the final right shift leaves the 39 least significant digits of the final product correctly positioned with respect to the fixed binary point in RII.

If a round off is requested a "1" is added to the 2^{-40} position of the final product. To permit resultant carries, this correction is accomplished during the 39th step of the multiplication.

Thus, if a round off is desired, the main control requests a recognition signal after 38 shifts and sequences the final three steps. If no round off is desired, a recognition signal is requested after 39 shifts and the final two steps are sequenced.

The completion of the 41st step signals the termination of the multiplication order permitting the main control to observe the next order half or to bring up a new order pair.

Figure 14

Division. The division process is somewhat more straightforward than multiplication -- containing only 39 steps with no clean up processes.

The dividend resides in RI and its sign is stored in the main control. The divisor is transferred into RIII and its arrival initiates the arithmetic operation. The quotient digit in each step is inserted

into the least significant stage of the upper rank of RII and then both registers are shifted left.

The decision to carry out a subtraction of the divisor from the partial remainder is based upon the recognition of an overdraft. This is accomplished directly by comparison of a carry arising from the most significant stage (sign digit) of the adder with the sign of the original dividend. The nature of the quotient digit to be inserted is determined directly by a comparison of the above carry and the sign of the divisor.

These rules are summarized as follows:

If Sign of Dividend = Carry, do not add only shift left.

If Sign of Divisor = Carry, Quotient digit = "0".

The round off in division is accomplished by arbitrarily setting the 2^{-39} digit of the quotient to "1".

The total vocabulary of the Institute machine consists of 19 basic orders described below:

(1-8) Bring a number from position x in the memory and add it to the accumulator with the following choices:

(a) "Clear" or "Hold" where "clear" signifies the desire to preclear the accumulator to zero.

(b) "Magnitude" or "Number" where "magnitude" signifies that the absolute magnitude of the quantity at x is to be added in.

(c) "-" or "+".

(9) Multiply: Bring a number from position x in the memory and multiply it by the number residing in the Arithmetic Shifting Register, RII. This order has choices of "Round off" and "clear" (preclear of R_1).

(10) Divide: Divide the number in the Accumulator by a number to be brought up from position x in the memory.

(11) Load RII: Bring a number from position x in the memory and place that number in the Arithmetic Shifting Register, RII.

(12) Left Shift: Shift the numbers in the Accumulator, RI and the Arithmetic Shifting Register, RII $n(\leq 47)$ places to the left. During this process the contents of RI shift end around into the right end of RII. However, only zeros are propagated through right end of RI. The number n is here the address portion of the order.

(13) Right Shift: Shift the numbers in the Accumulator, RI and the Arithmetic Shifting Register, RII $n(\leq 47)$ places to the right. During this process the contents of RI shift end around into the left end of RII. However, only the sign digit of the original quantity in RI is propagated through the left end of RI. The right shift order has the choice of Round Off. If Round Off is chosen, 2^{-39} is added into the quantity in RI at the end of "n" right shifts and a total of n + 1 shifts are executed.

(14) Store: Take the number existing in the Accumulator, RI and store it at position x in the memory.

(15) Parity Unconditional Transfer: Bring a new order pair into the lower rank of the Memory Register, RIII from position x in the memory and proceed to observe that order which occupies the same half of the order pair as the unconditional transfer being executed.

(16) Non-Parity Unconditional Transfer: Bring a new order pair into the lower rank of the Memory Register, RIII from position x in the memory and proceed to observe that order which occupies the half of the order pair opposite to that occupied by the unconditional transfer being executed.

(17-18) Parity and Non-Parity Conditional Transfers: If the number in the accumulator is positive, execute operations described in 15 and 16.

If the number in the accumulator is negative, ignore the request for transfer and proceed to the next order in sequence.

(19) RII => RI: Take the number in the Arithmetic Shifting Register, RII and add it to the quantity in the accumulator, RI. This order includes all of the eight variants described in orders (1-8).

Two partial substitution orders contained in the original 21 were deleted during the construction of the machine.

There has now been added a 20th Input-Output order permitting automatic communication with either a magnetic drum memory of 2000 word capacity or with an IBM Punched Card Reproducer.

Conclusion

The foregoing remarks represent an attempt to give you a picture of the skeleton of the Institute machine, the criteria of its creators, and the basic functions it can now perform.

It is not possible in so short a discussion to describe the work experiences of this machine since its completion. However, I shall attempt to give you a qualitative description of its accomplishments.

The bulk of the machine's useful computation time has been devoted to the solution of a set of partial differential equations representing a simplified model of the atmosphere. In addition problems in number theory, matrix inversion and integral equations have been solved by its circuitry.

Measures are now being taken to define the present level of reliability and to establish procedures which can at the same time raise that level and prevent deterioration over long periods. This program has three major categories:

(a) Decrease Input-Output time since within the period spanned by a computation, any decrease in time reduces the probability of a given noise level affecting the outcome.

(b) Continue development of better inspection techniques such that deterioration of any element may be discovered soon after it occurs. The time for diagnosis of multiple deterioration is extremely disproportionate to the number of elements failing. This process requires a theoretical and experimental evaluation of the weakest links in the machine by means of marginal checking techniques and a hierarchy of test codes.

(c) At the top of the hierarchy of test codes is the running experience of the machine when doing useful computation. No test codes can exactly reproduce all of the combinations of orders and information handling presented by diverse problems over a long period of time. Hence a maximum amount of information must be drawn from the computation experiences -- particularly in the case of malfunctions.

In parallel with these processes must run complete open mindedness of research people in the computer field. None of us has written the last introduction to this subject much less the last chapter.

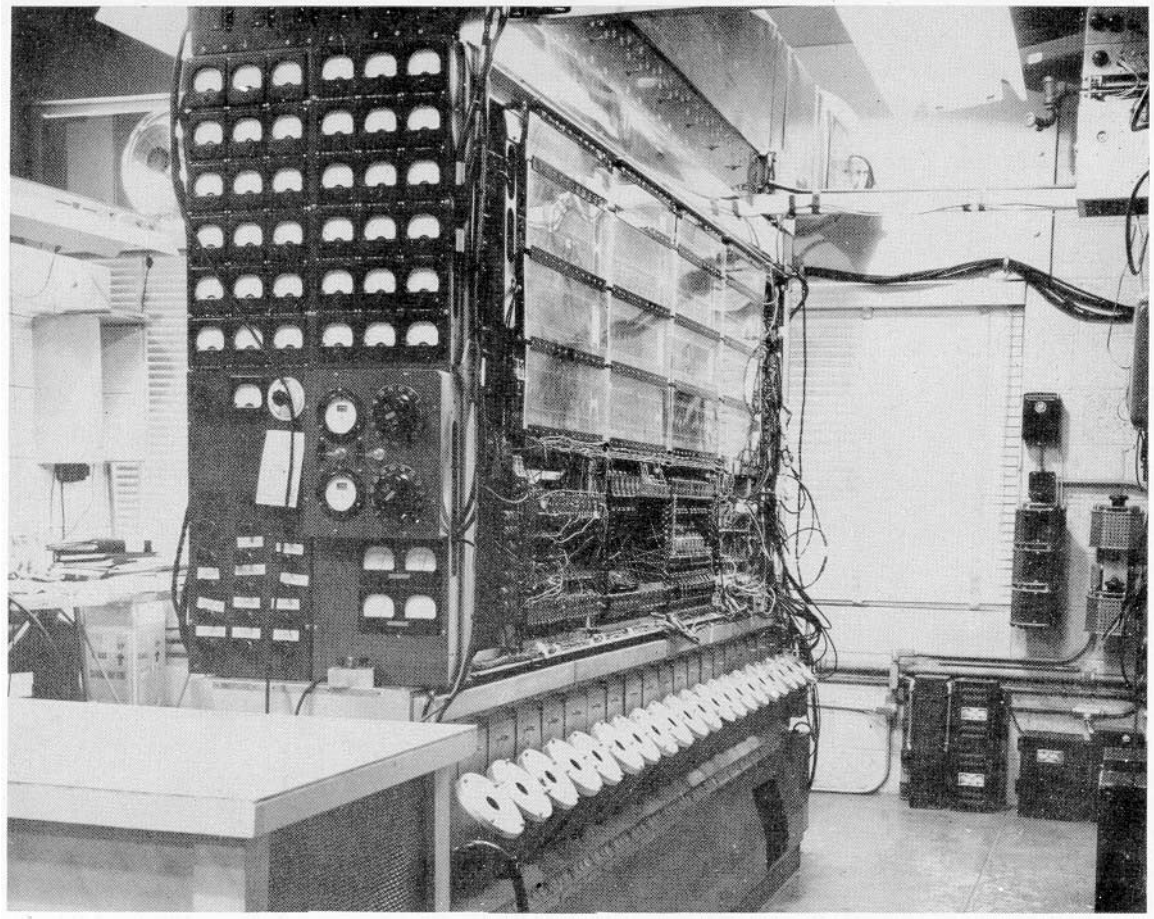


FIG. 1

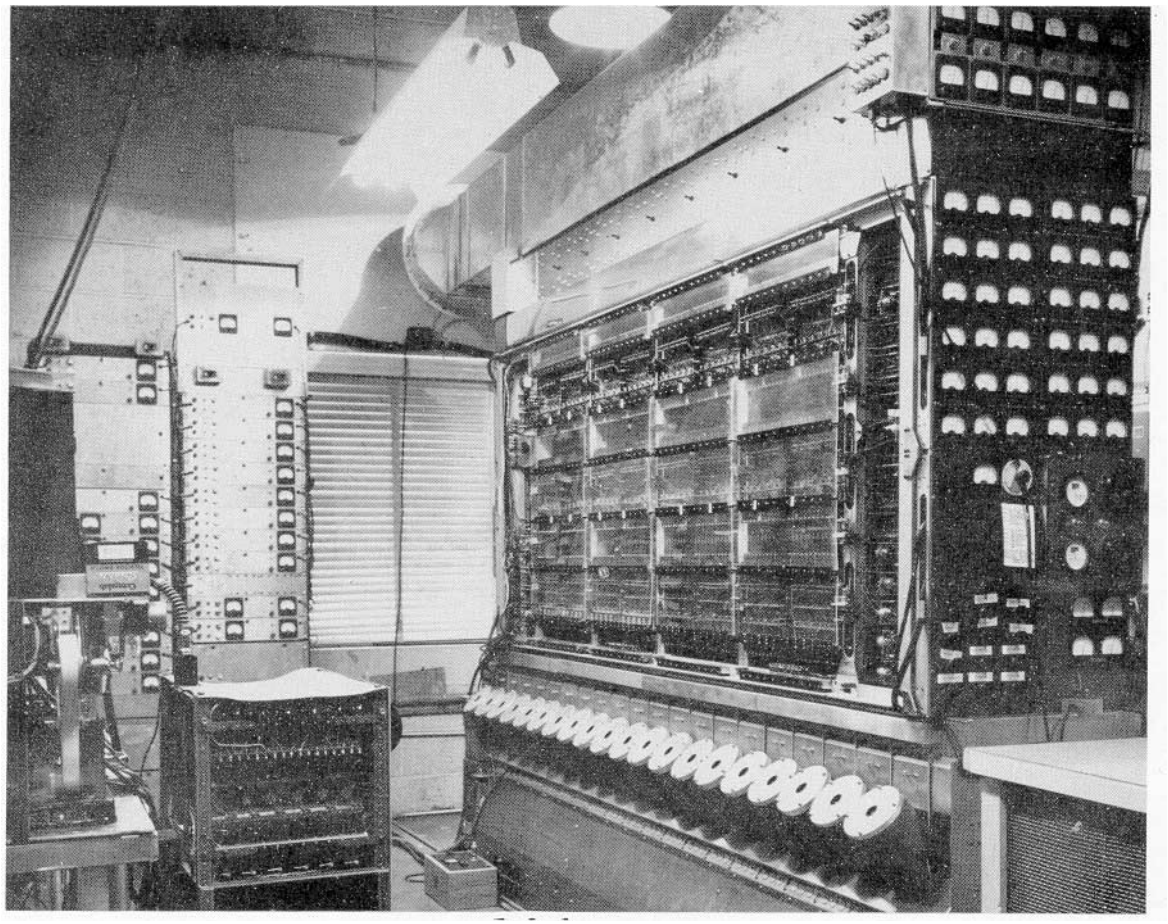


FIG. 2

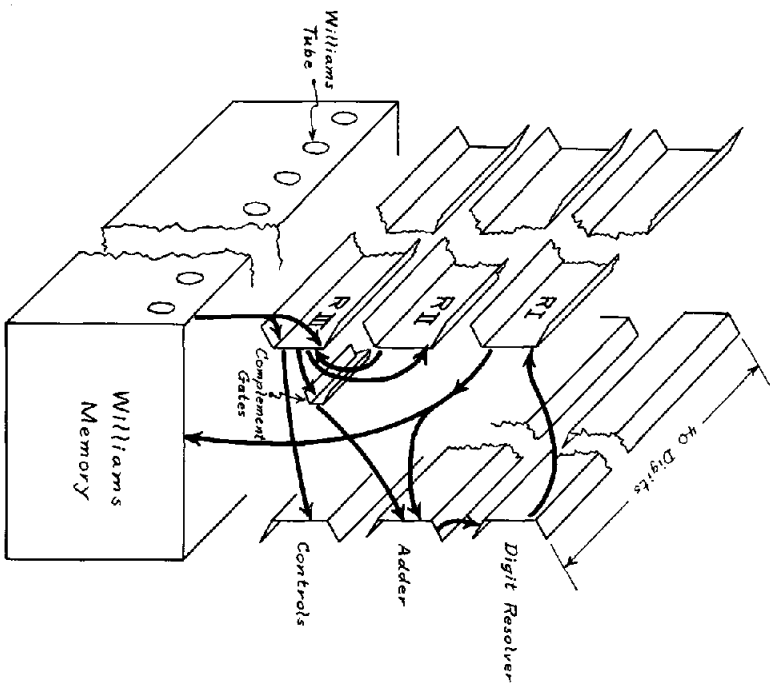


FIG. 3

SCHEMATIC OF BASIC CONNECTIONS BETWEEN WILLIAMS MEMORY, CONTROL, AND ARITHMETIC UNITS

MACHINE CHARACTERISTICS

OVERALL	MEMORY	ARITHMETIC
Size: 2'x 6'x 8'	Williams Type	Add: 62 μs
Parallel	5CPIA	Mult: 500 μs av.
40 Bit Words	1024 Words	Div: 800 μs
Power: 10 kw	12 μs Access	Asynchronous Control
Tubes: 2300	40 Parallel	8 Bit Orders

INPUT – OUTPUT

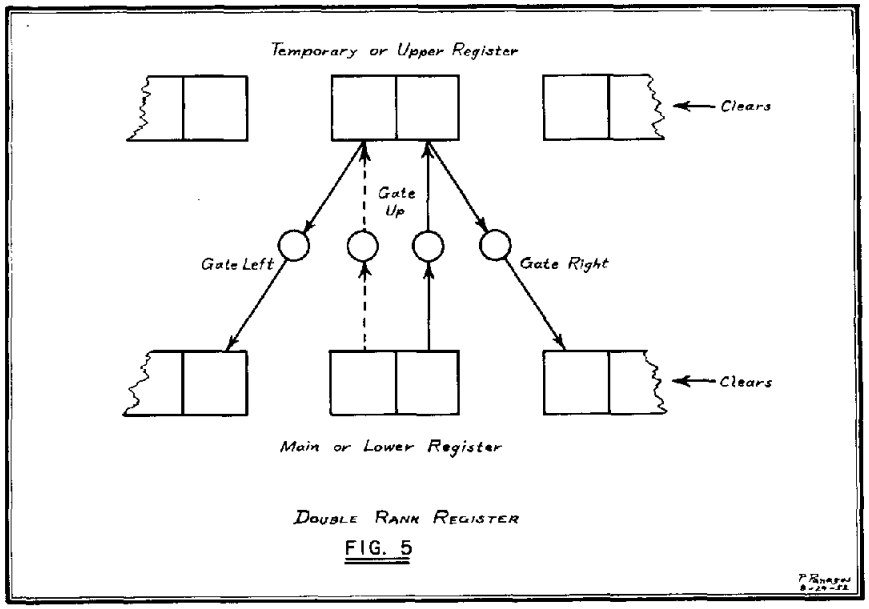
Teletype Tape: 5 holes across, 4 bits used

Full Load: 8 min Full Punch: 16 min

IBM Card: } In process

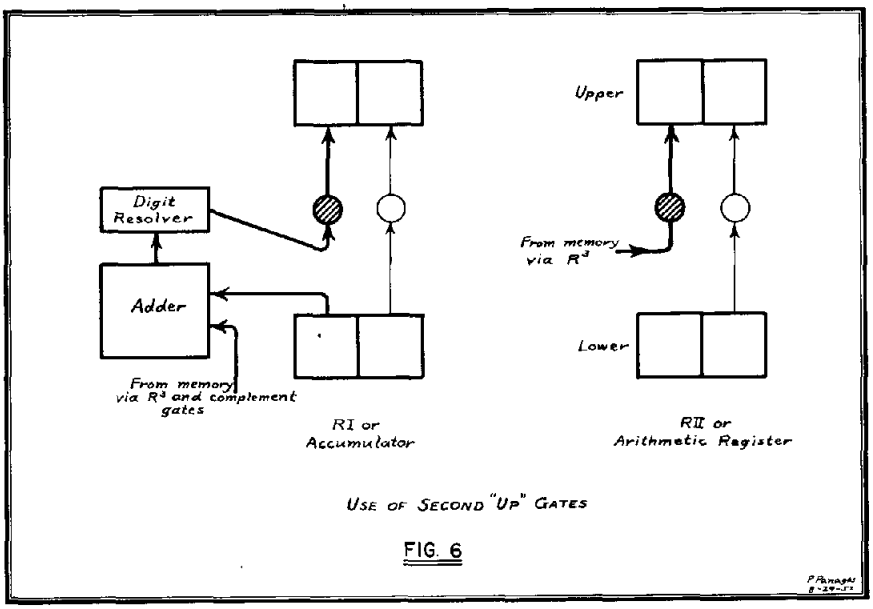
Magnetic Drum: }

FIG. 4



DOUBLE RANK REGISTER

FIG. 5



USE OF SECOND "UP" GATES

FIG. 6

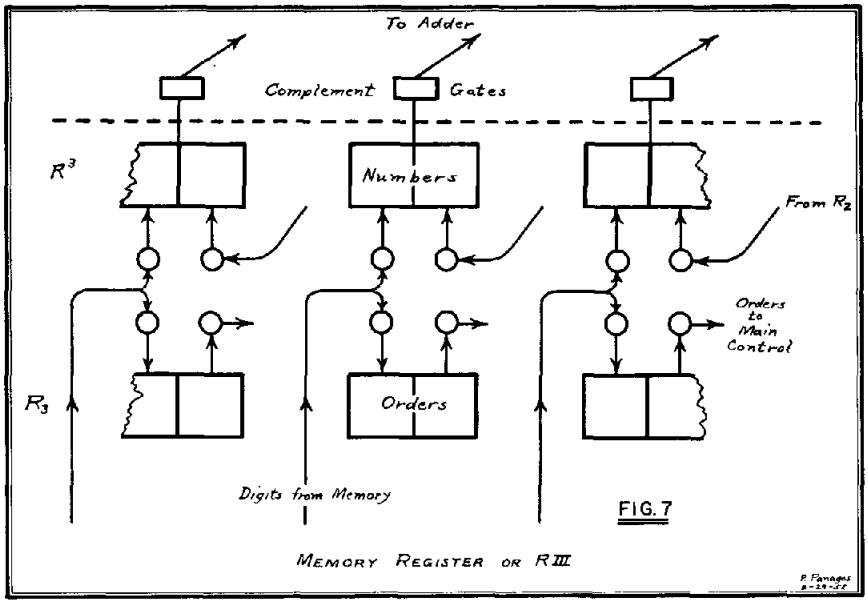
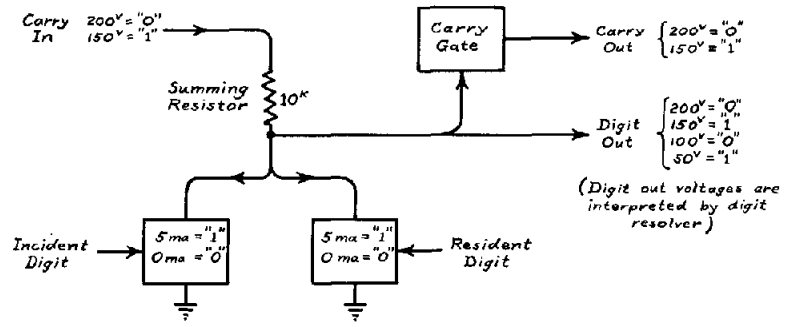


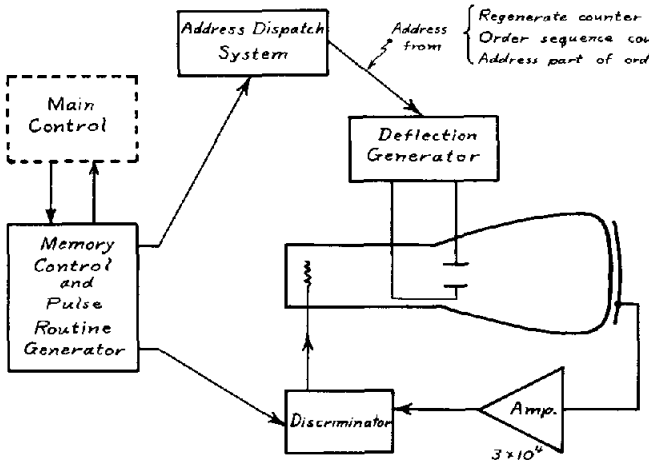
FIG. 7

MEMORY REGISTER OR RIII



STATIC KIRCHHOFF ADDER

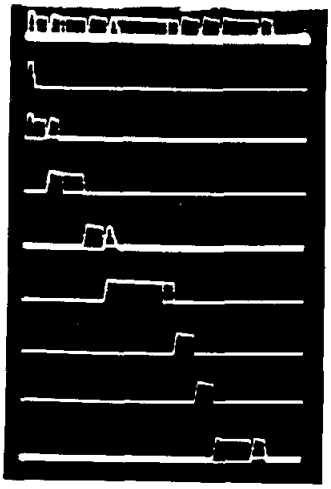
FIG. 8



MEMORY BLOCK DIAGRAM

FIG. 9

WILLIAMS PULSE CHAIN



Total Chain (20 μs)

Clock Pulse

S.D. and Strobe

Normal and Super Dot

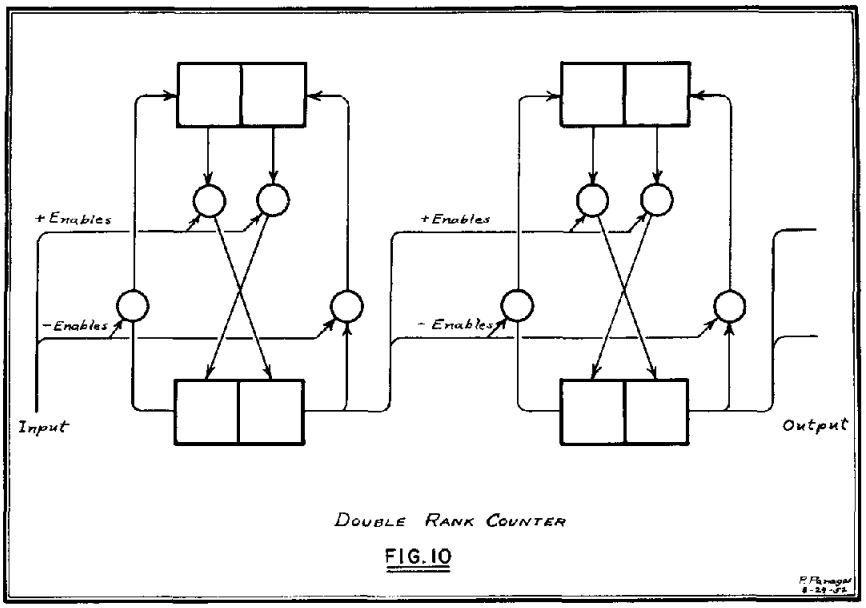
Twitch Settling Delay (with termination pulse)

Normal and Super Dash

Action/Regen. Decision

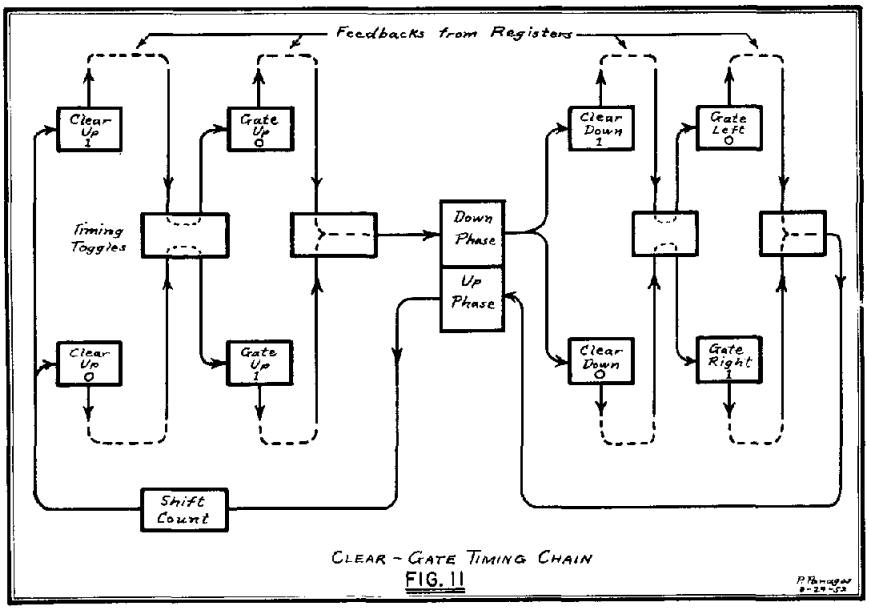
Clear and Gate to get New Address

FIG. 9a



DOUBLE RANK COUNTER

FIG. 10



CLEAR - GATE TIMING CHAIN

FIG. 11

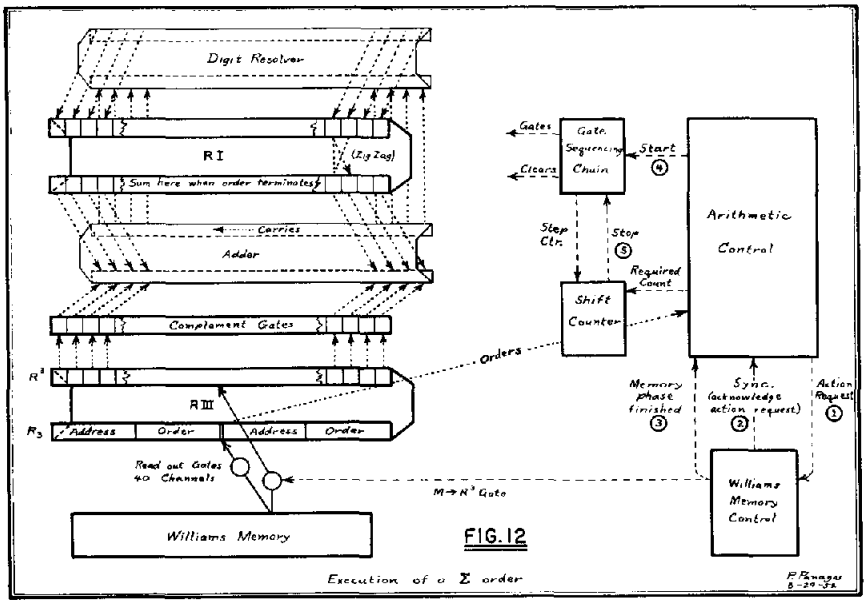


FIG. 12

Execution of a Σ order

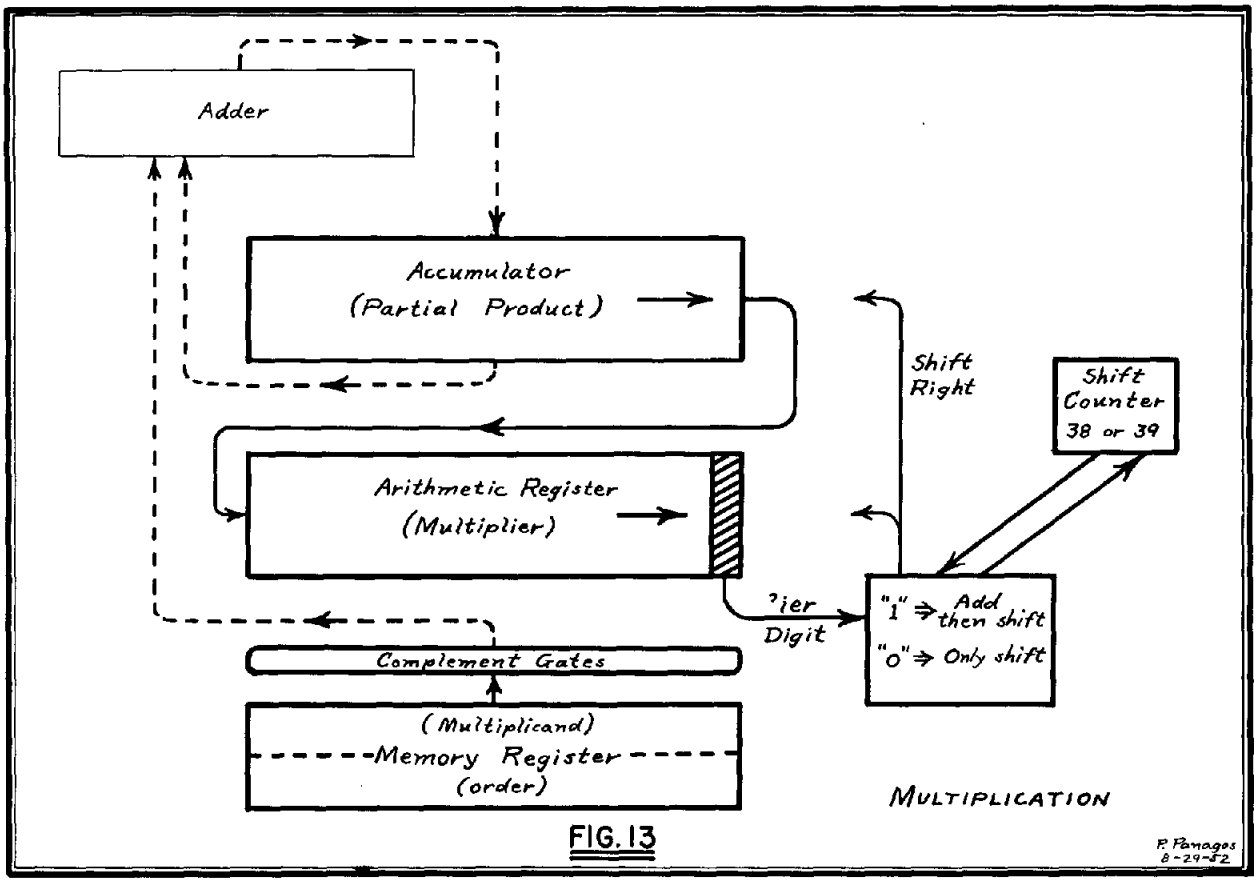

Adder
Accumulator
(Partial Product)
Shift
Right
Shift
Counter
38 or 39
Arithmetic Register
(Multiplier)
ier
Digit
"1" ⇒ Add then shift
"0" ⇒ Only shift
Complement Gates
(Multiplicand)
Memory Register
(order)
MULTIPLICATION
P. Panagos
8-29-52

FIG. 13

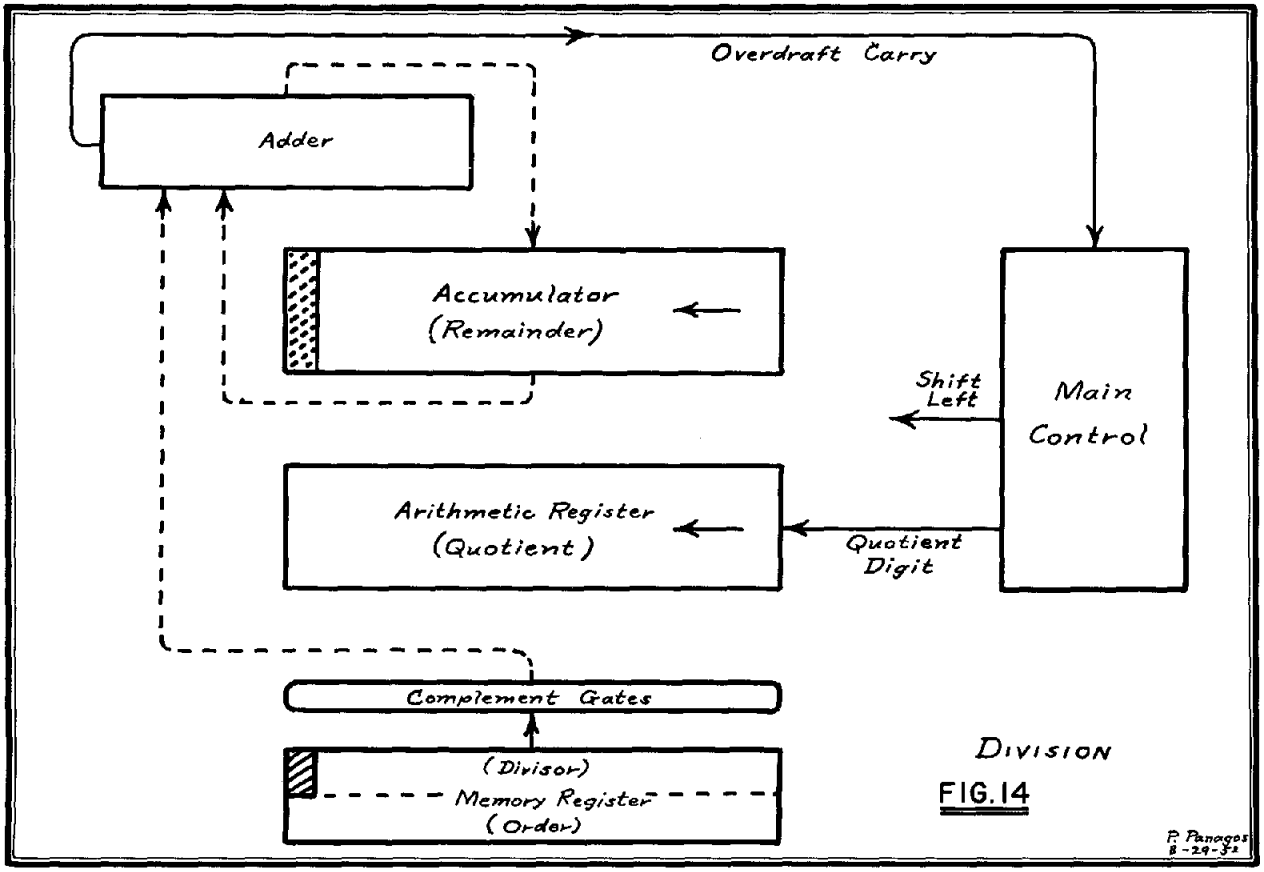

Overdraft Carry
Adder
Accumulator
(Remainder)
Shift
Left
Main
Control
Arithmetic Register
(Quotient)
Quotient
Digit
Complement Gates
(Divisor)
Memory Register
(Order)
DIVISION
P. Panagos
8-29-52

FIG. 14