tion in one instruction. A 64-bit instruction would be wasteful for the highly stylized floating-point and indexing operations with their fixed data formats.

The concept of varying the instruction length to suit the operation was explored. Vocabularies of half-length, quarter-length, and even eighth-length instructions were tried. While short instructions save space, it was found that the saving can be quickly offset by the extra bits needed to tell the computer how to interpret each format. The greatest economy of memory space and memory references was gained in a mixture of half-length and full-length instructions.

A floating-point "add" illustrates the half-length format. Only 18 bits of address are needed because a floating-point number occupies a full 64-bit word.

It has been found already that the flexibility and power of the new vocabulary will materially shorten the length of programs. In programming comparisons with the IBM *704* or *709*, reductions in program length up to a factor of 3 have been observed.

The Simplicity of Complexity

One may ask whether a more complex instruction set does not lead to more difficult programming. One answer is that programming can be simplified by adding instructions to complete a set (branch on plus, as well as branch on minus) and arranging them systematically. Another answer can be obtained by looking at the other extreme.

Van der Poel has shown[1] that the simplest instruction set theoretically consists of just one instruction. The instruction contains no operation code, only an address. Every instruction causes a combination of "subtract" and "store" to be executed; the difference replaces the contents of both the accumulator and the specified memory address. All other computing operations, including conditional branching, can be built up from this one instruction which is a very easy one to learn. But the programs needed to simulate no more than the elementary instruction set of early computers would be enormous. It is quite a task just to estimate the size of the program for a real job. It seems safe to say that the storage required would be gigantic, and a desk calculator would probably be faster.

A complex, but appropriate, language will, in fact, simplify the programmer's task as the problems to be done become more complex.

Relationship to Automatic Programming Languages

The advent of more powerful computers designed to tackle larger problems is thus accompanied by more elaborate and versatile instruction sets. Programs to do the same job require considerably fewer instructions and fewer references to memory. Or, to look at it another way, sequencing of simpler instructions stored in a relatively slow memory is replaced by internal sequencing with high-speed control circuits. This is a form of microprogramming using the fastest available memory, one made of transistor flip-flops.

Such an instruction set is still a long way from the "superlanguages" being developed under the heading of automatic programming. These languages are intended to simplify the task of the problem coder, not to raise the performance of the machine. The instruction set is an intermediate level between the programmer's language and the language of the elementary control steps inside the machine.

A 2-step process of translation is thus required. One is the programmed assembly of machine instructions from the statements in the superlanguage. The other is the internal translation of instructions to control sequences. The 2-step process is a matter of necessity at this stage of development to keep the complexity of the computer within bounds. It has the advantage that each language can be developed independently of the other to be most effective for its purpose.

At the level of the user, there may be a need to develop specialized languages which facilitate programming of different jobs with varying emphasis on arithmetic, logical operations, data manipulation, and input-putput control. At the machine level, where all these jobs come together, the need is clearly for a versatile and relatively unspecialized language. Perhaps the greatest demand on versatility is made by the process of translating from an automatic programming language to machine language. The performance of a computer on translating its own programs is a significant measure of how effective a tool the instruction set really is.

Reference

1. The Essential Types of Operations in an Automatic Computer, **W. L. Van der Poel.** *Nachrichtentechnische Fachberichte*, Fredr. Vieweg & Sohn, Braunschweig, Germany, vol. 4, 1956, p. 144. (Proceedings of a conference on Electronic Digital Computing and Information Processing held at Darmstadt, Germany, Oct. 1955.)

System Design of the Gamma 60

PHILLIPPE DREYFUSS
NONMEMBER AIEE

THE BULL Gamma *60* is a large-scale electronic data-processing system designed for general business use, as well as for engineering calculations.

The Gamma *60* incorporates many new design concepts both in the structure of its information handling devices, and in its novel control unit organization. It uses a resistor diode logic.

The Gamma *60* opens a new area in data processing whereby different functions, or even distinct independent problems can be solved simultaneously on a single machine without any previous planning or programming of the possible occurrence of these parallel operations.

The Gamma *60*, like other data-processing machines, handles different sets of information, and the difference depends upon:

1. The nature of the information; qualitative data, quantative data, or instructions.
2. The origin or destination of the information; input devices, internal processing and storage units, and output devices.

It is well known that no single code is optimally suited to the requirements of these different classes of information. These requirements are: information density, minimum storage requirements, coding efficiency, ease and speed of transcribing input and output codes to and from internal machine codes, high transfer flow, parallel versus serial, number of parallel channels, programming ease and economical storage allocation, and fixed versus variable "word lengths."

In order to remain as close as possible to an ideal code which would optimize

Phillippe Dreyfuss is with the Bull Company, Paris, France.

these often conflicting requirements, different codes are used in the Gamma *60*, depending upon the nature and location of information throughout the system.

Internal data code is used: Quantitative (numerical) data are coded in a 4-bit decimal code; qualitative (alpha-numerical) data are coded in a 6-bit alphanumerical code. The internal instruction code means that the instructions are coded in straight binary code.

As to the internal information length, the information quantum is called a "catena," and it is composed of 24 bits representing either 6 decimal digits, or 4 alphanumerical characters. This quantum must contain a multiple of 4 and 6 bits to represent a whole number of decimal or alphanumeric characters. Twenty-four bits was found to be a good compromise between the minimum 12 bits, which would lead to a too-low transfer flow from a parallel readout core memory, and 36 bits or more, which was judged as too large an information quantum. The catena is to be considered as the equivalent of a character in variable word length machines, but it cannot be called so, as it may contain several characters. It is transferred in series to and from the main memory.

Not wanting to call a "quantum" a word, or a set of characters a letter, (a word is a word, and a quantum is something else), a new word was made, and it was called a "catena." It is an English word and exists in Webster's although it does not in French. Webster's definition of the word catena is, "a connected series;" therefore, a 24-bit information item. The word catena will be used hereafter.

The internal code, therefore, has been defined. Now what are the external data codes? These depend primarily upon the information handling device involved. The Gamma *60* is designed to handle information relevant to any binary coded structure. Thus an 80-column punched card is considered as a 960-bit information item; 12 rows multiplied by 80 columns equals 960 possible punches; is stored as an exact image in 960 magnetic cores of the main memory with 2 card columns occupying one catena.

If necessary for further processing, this card code will be translated into internal code in a later operation. Some of the usual codes are: punched card codes, 80 or 90 columns; International Business Machines Corporation, Bull, or Remington line printer codes; punched paper tape codes, 5 and 8 channels; magnetic tape codes, etc.

A novel approach in logical structure has been developed for the Gamma *60*. The machine can roughly be divided into a main unit, including the magnetic core memory, and a super control unit, composed of a program distributor and a data distributor. There are an unlimited number of elements, each of which constitutes a complete data-handling device with its input-output storage and control units. These elements belong to one of these categories:

1. The arithmetic unit.
2. The logical unit.
3. The comparison unit.
4. The translating unit.
5. The magnetic-drum auxiliary storage unit.
6. The card reader input unit.
7. Tape reader input unit.
8. Card punch output unit.
9. Tape punch output unit.
10. Line printer output unit, etc.

The fundamental common characteristic of these elements is that each one is rectly connected to the main memory by two (some by only one) transfer buses, a distributing bus from the main memory to the elements, and a collecting bus from the elements to the main memory. No direct transfers between elements are possible. However, each element contains its own program and control unit, and can operate completely independently of any other element in operation, once it has been provided with an instruction.

The only common devices shared by all the elements are the main memory and transfer buses. The program distributor handles the super control problem of providing each element automatically and in due time with instructions, and of setting and removing all the necessary interlocks to avoid program interference. The data distributor handles the priority problem of time-slot allocation (one slot equals one catena) to demanding elements on the transfer buses.

Thus the Gamma *60* allows a completely integrated on-line operation resulting from the possibility of running several independent problems at the same moment and on a single line. To illustrate this feature the following problems could be performed simultaneously:

1. A general data-processing problem such as a payroll.
2. A scientific computation such as a matrix inversion or linear programming.
3. One, or several magnetic tapes to printer conversions.
4. One, or several punched cards to magnetic conversions.
5. One, or several punched tapes to magnetic tape conversions.

This would, of course, require an adequate number of elements, magnetic tape units, printers, card and tape readers, and enough magnetic core storage and drum storage to accommodate all of these programs and their corresponding working storage. But, this condition being fulfilled, nevertheless, the different programs would have been planned, written and loaded in the machine independently, and in complete ignorance of one another. As a result of on-line operation a large economy of hardware is achieved, as well as a higher degree of automation; all of the phases of a problem can remain under the control of a single stored program.

The dominant feature of this organization is, however, the over-all saving of time. In effect, the time loss resulting from solving several problems simultaneously will only exceptionally exceed a few per cent for each problem of the time that would be required to solve that problem alone.

As for operations generally handled off-line, such as card-to-tape or tape-to-printer conversions, they will always keep pace with the maximum speed of the slow mechanical device, card reader, or printer.

Everthing is quite conventional. There is no high speed; there is nothing extraordinary in technology. The question was raised before by someone as to the possibilities of a symbolic code. The printer allows one to write a symbolic formula code such as: $y = \Sigma X_i(IJK)\text{-}P(1/2)$.

Now study the main unit and the elements. The following are characteristics of the main memory which is a saturated magnetic core memory. It has a capacity of 32,768 catenae representing 786,432 bits, 196,608 decimal digits or 131,072 alphabetic characters. The access time is 11 microseconds for 1 catena. Every catena is addressable.

The arithmetic unit performs the 4 arithmetic and the comparison operations both on program and on floating decimal-point numbers. The numbers are represented by two catena; the algebraic sign 1 bit, 10 coded decimal digits 40 bits; the decimal exponent (range 0 to 79) 7 bits. The arithmetic unit instruction allows 1 to 3 address operations, that is, 2 to 4 catenae.

A 3-address floating decimal-point addition takes 150 microseconds; a 1-address floating decimal-point addition takes 88 microseconds. The arithmetic unit also comprises four particular index registers for automatic address modifications.

The logical unit performs logical and

straight binary operations and some specialized operations, such as program loop counts, transfers in and out of subroutine, etc. It contains two registers.

The alphanumerical comparison unit performs the comparison of two items stored in the main memory. The length of the items in catenae is determined by the operation code. It can take any value from 1 to 16,384 catenae.

The comparison unit is used to translate codes and edit input and output information. None of the input or output devices are provided with any electromechanical means of editing or selection, and no plugboard or relay networks. All these functions are performed by the translating unit for which special instructions are provided. The line-printer paper skip is itself programmed. The speed and flexibility thus achieved surpasses greatly those obtained with electromechanical means and further increases the integrated on-line character of the Gamma *60*.

The comparison unit instruction allows a 2-address variable-length operation. The same unit handles transfers from one location of the main memory to another. The same instructions, two address and length, are used.

The magnetic drum units are auxiliary storage units exchanging information with the main memory. Their capacity is 32,768 catenae each; the average access time is 11 milliseconds. A magnetic drum unit instruction is of the same 2-address type and length as the comparison unit instruction. The first address is that of the magnetic drum, the second is that of the main memory; the operation code determines the direction of the transfer and length. A single operation can control the transfer of a full magnetic drum unit (32,768 catenae) to the main memory, or vice versa.

The magnetic tape units are auxiliary storage units exchanging information with the main memory. They use one-half inch Mylar tape comprising 8-information-carrying channels. Pulse packing density is 200 bits per inch; tape speed is 50 inches per second leading to a frequency of 10 kc. This frequency corresponds to a rate of 20,000 decimal digits or 13,333 alphabetic characters per second. The tapes may be read in forward and backward directions. A fast rewind is provided.

The card reader units will read 80 column cards at a rate of 300 cards per minute. Three reading stations allow a complete check. The first reading is compared with the second reading; if these differ, the third reading may either agree with one of the first two, or give a third different value. In that case, the card will be directed to a special card stacker.

Punched paper tape may be directly fed into the Gamma *60* through the tape reader units at a rate of 200 characters per second. Any type of tape with from five to eight channels and code may be used.

The card punch units are in fact card reader punch units. They read and punch 80-column cards at a rate of 300 cards per minute. Two card hoppers are provided; one for reading cards already punched, which can then be read by two reading stations before reaching the punch mechanism; the other for eventually feeding blank cards directly into the punch mechanism. Two stackers are provided.

The line printer units are based on the flywheel principle. They will print 300 lines a minute each. The line holds 120 characters; there are 60 characters for each position. The paper skip is controlled by the program. A skip instruction specifies the number of lines to be skipped or a skip to the next form. Printer checking uses the echo principle.

The number of elements that can be connected to the main unit is unlimited. To illustrate the possibilities of simultaneity, the following figures show the number of elements of the different categories that could operate simultaneously if they were to saturate the rate of flow of the transfer buses and main memory:

Comparison unit: 1
Logical unit: 1
Arithmetic units: 2 to 3 (depending upon the operation)
Translating units: 3
Magnetic drum units: 8
Magnetic tape units: 28
Line printer units: 312
Card reader punch units: 472

These figures are the justification of the control organization previously described. It shows the momentous gain that can be achieved in time and hardware with the adoption of on-line data processing in a system which allows simultaneous operations.

It is the author's firm belief that this will be one of the major trends in system development of the data processing machines of the 60's.

This is all that can be told about the Gamma *60* at this time. There is not enough time to go into more detail, but the author wanted to show that it is possible to run a computer data-processing machine with parallel operations, and that it is possible to have simultaneous functions, going along at the same time without any interconnection between the programs of these functions. Of course, it is a great time saver because the over-all loss in time of running, say a 5-tape-to-printer conversion, and running a tape sort at the same time, and running, say, invenory control, or any of ten programs—is not more than 5 %.

It is a question of putting in more units. Of course there is one limitation, the 2.4 megabits of the code storage. One cannot go further than that, and that is the limitation of the amount of simultaneous programming that can be handled by this conception.

Discussion

Chairman Carr: The first question is for Mr. Shaw from Quentin Correll, IBM: "What has the command structure you described been used for to date? What future uses do you see for it?

J. C. Shaw: There are other questions here that are along a very similar vein, so let us get them all in at the same time and I will answer them together.

Chairman Carr: The next question for Mr. Shaw is from L. D. Yarbrough, North American Aviation: "I understand that an information-processing language has been coded for Johnniac. How about other existing computers, (*704, 1103,* etc.). Will you publish details of this project?"

It seems as if everyone has asked the same question. The next question for Mr. Shaw is from Mrs. Joanna Wood Schot, David Taylor Model Basin: "For what machine is the program you have discussed designed? Or is it designed for an arbitrary machine? What types of data have already been successfully processed using this routine, or is the whole thing indeed hypothetical in spite of your claim of its existence?"

J. C. Shaw: Three of these questions are along the idea: What has the command structure been used for to date? What are the future uses?

We are in the field of simulating human problem-solving behavior. To put it bluntly, we would like to create a machine that thinks.

Last year at Western Joint Computer Conference we presented two papers on the logic theory machine. At the 1955 Western Joint Computer Conference, a paper on the chess machine was presented.

The logic theory machine has done some interesting things. One which I might mention here is that it has discovered a new proof to a theorem in Chapter II of *Principia Mathematica.*

We are only now achieving the chess machine. However, I am looking forward to a paper in this conference by Dr. Bernstein and others having to do with the chess machine that they have coded at IBM. There has been a series of these information processing languages, so that the one I described is really IPL VI. The prior ones, which are essentially the same, were realized on the Johnniac, the Princeton-type computer built by the RAND Corporation.

The people working along the same lines at the Carnegie Institute of Technology have put an information processing language on the *650,* and also on the *704.* So, in the pure interpretive form, IPL's essentially the same as the one I described exist and are being used.

Chairman Carr: The first question for Mr. Dreyfuss is from myself: "Wasn't this differential address device discovered or invented in this country first by Samuel Alexander and others at NBS and the University of Michigan?"

Phillippe Dreyfuss: We were badly informed, and we had to rediscover it.

Chairman Carr: There is a question from the floor: "Are you able to use the instruction location counter as an index register?"

Phillippe Dreyfuss: Yes, any instruction, any current address counter can be used as an index register; but any internal memory location in fact can be used as an index register by substituting the address automatic feature, the automatic counter of the address. This is made necessary by the fact of having simultaneous operations. Let us take one like a drum transfer load, a current address of the drum telling it where the first address is, the first catena to be transferred. This transfer is going to take place only once in every eight cycles; it is not continuous, and it must be stored somewhere. The point which we have reached in the transfer is to know that the next transfer arrives eight cycles later, and it will have to translate it next. You have to have the feature of an automatic counter anyhow. We added another constant or even a variable.

Chairman Carr: I would like to express what I think a lot of people are thinking, and that is that the Bull Company is to be congratulated for very imaginative design. I would also like to ask you how far along is this design?

Phillippe Dreyfuss: Thank you. Well, the machine is being developed, and the first delivery is scheduled for delivery in one year. We have seven systems on order.

There is one other inevitable question that you all have in mind, so before anybody puts it to me, I will give it to you right away. The cost of the system is hard to determine, as it depends on how much and how many units you put in. Let us say that the most basic system would sell for about a million dollars.

Chairman Carr: A question from the floor: "How many people made up the group who designed the machine?"

Phillippe Dreyfuss: Four or five people.

Chairman Carr: Another question from the floor: "When you have several problems going on, how do you know where to start the new ones, how do you know that you have enough core memory?"

Phillippe Dreyfuss: You have to have some kind of core planning. It can be done by the supervisor, or by the machine itself. At some time the machine always counts how much core storage is available. It knows the loaded programs. You generally know how many core storages you are going to use, so the machine can do its own accounting. If you are trying to put in a problem that does not fit, it would just write out, "It does not fit."

As to how you are going to start a problem, there are just the sequences as I have mentioned. There is the cut operation, the sort of branch cut operation; every time we start a new sequence it comes from a branch or cut. A very special cut operation is put in on the console which is the start operation. The difference between this start operation and that of other machines is that you can restart the machine when it is once started by setting a new first address. If you have a new first address that you want, you press the button, and you start a new program. The other one may stop; in fact, I did not have a chance to tell you about all of the cuts. There are eight types of cuts. Also, I did not tell you about internal checking, but the machine has internal checking.

Chairman Carr: Again from the floor: "Do you anticipate trouble-shooting in the equipment as well as trouble-shooting programs?"

Phillippe Dreyfuss: That is why I started to tell you about the checking. Whenever an error occurs, if our checking is correct, this error should be automatically detected. What happens when it is detected? It puts one of the bits in a qualitative catena and it stops the sequence on hand, and it sends us to a diagnostic program. The first thing we have is 24 portions of elements to find out what happened. Two things may have happened: Either there is a possible corrective step taken by the machine itself; or it will stop the reading of the tape through the diagnostic program.

Of course the arithmetic program will just try to find out if it is a random error or not, and repeat the operation, if the operand is still available.

There is another type of cut that does exist. When a unit breaks down, the sequence in question has stopped, in fact just as if you had programmed a branch operation branching over to a diagnostic program. This diagnostic program cannot use the unit that was broken down, and, most important, it should not stop by a cut operation which would liberate the unit taken over by another program which is waiting for it. In this case, unfortunately, I did not have time to explain how the waiting line is established. It is automatically established there and there may be any number up to a hundred programs waiting for a certain program to become available. A chain-like system inside it allows it to repick the program as it arrives, whenever the unit becomes available.

Chairman Carr: From the floor: "Is this a relative diagnostic program that you are talking about?"

Phillippe Dreyfuss: It is a program; you can have as many as you can program.

Chairman Carr: From the floor: "Speaking of programming, have you developed a programming system for this machine, and can you give a brief discussion about it?"

Phillippe Dreyfuss: I wrote out one formula for you, and we have written a compiler which is in the mathematical type. We have more symbols to use on the printer, and a few modifications on the program. This program is now being coded, it is not coded as yet.

Chairman Carr: From the floor: "What are you coding it in, that is, what language?"

Phillippe Dreyfuss: In the machine we have used a 4-level code and it is a pure binary code. The first program interpretive routine that we have is what is called a second-level code, an alphanumeric symbolic code; one to one for each catena. This code that I have here uses a one-to-four-letter code to specify the type of catena. Then the addresses are in decimals. Of course the relative addresses are just alphanumerical codes.

A Direct Read-Out Bistable Circuit and Some Applications of It

IR. H. RODRIQUES DE MIRANDA
NONMEMBER AIEE

I. RUDICH
NONMEMBER AIEE

THE *6977* is a high-vacuum subminiature triode with a fluorescent anode primarily designed for display in transistor computer circuitry. It consists of a single-strand direct-heated filament having long life properties and requires 1 volt and 30 milliamperes (ma) a-c or d-c. Around this single-strand cathode is a cylindrical control grid. The anode is a gridlike structure coated with a *P-15* phosphor optimized for maximum brightness. The light output of the tube is visible from the side as an area approximately the diameter of the tube with a length of 1/2 inch. At zero bias, the plate current is approximately 0.6 ma and the tube can be cut off with from 2 to 4 volts, depending upon whether the filament voltage is a-c or d-c.

This tube, being a vacuum triode, can without objections be used as an active element with the advantage of combining two functions in this one device. As this tube will be mainly used for indicating the position of a bistable circuit, it is worth while considering this tube as an active part of this circuit. Fig. 1(A) and (1)B shows the average characteristics of the *6977* and its construction.

The use of complementary transistors in bistable circuits is well known. This principle of direct coupling of a p-n-p and n-p-n transistor providing a regenerative loop can be applied as well to the combination p-n-p vacuum tube.

A vacuum triode can very well be compared with a n-p-n transistor. However, there are some important differences to consider:

1. Operation can be in the negative grid voltage region.
2. When overdriving the tube, the anode-cathode voltage remains considerable.
3. The grid-input impedance stays very high in the operating region.

While points 1 and 2 are only facts which are not particularly advantageous or disadvantageous, the high-input impedance of the tube is definitely an appreciable feature. This gives the possibility of triggering the circuit with very little power.

The introduction of the *6977* only as an indicator tube for transistor flip-flops already means a big saving in components. It has made the amplifier stage between the indicator and the bistable circuit obsolete. The circuit to be described here is a further attempt in this direction.

In Fig. 2 there is a positive feedback from anode via the transistor to the grid of the tube.

"Off" Condition

Assume that for $V_g < -4$ volts, the anode current is zero. This means that there is no base current and no collector current (assuming the transistor can stand the high collector voltage). Nevertheless, because of the small currents in the loop $R_1R_2R_3$, the grid voltage stays < -4 volts. This situation therefore is stable. It is referred to as the off condition.

Transient I

Now consider that in some way, i.e., by externally applying a positive going pulse to the grid, the tube starts to conduct. This means that there is base current, and consequently, collector current in the transistor. The potential of A grows in the positive direction and because of the voltage division $R_1 - R_2$ point B also rises. This tends to increase the anode current more. This is (or can be) an unstable situation. (For the stability criteria see circuit analysis.)

"On" Condition

The current increases until the voltage drop over R_3 approaches 50 volts. The collector-emitter voltage becomes nearly zero and the transistor ceases to be an active element. This again is a stable situation. R_1 and R_2 must be chosen so that the level of B is, in this condition, about zero volt. R_3 can be chosen so that the transistor just reaches saturation. The analysis will go over these considerations in more detail.

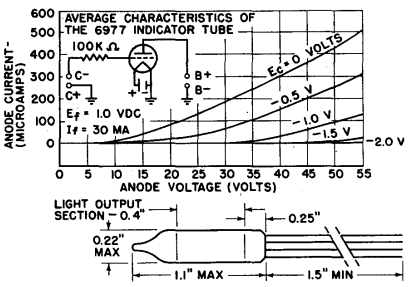

Fig. 1(A). Construction of the 6977

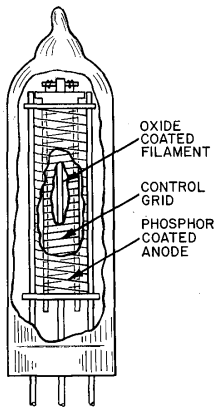

Fig. 1 (B). The 6977 is a high-vacuum subminiature indicator triode with a type P15 phosphor-coated anode

Transient II

Now, if by some means the anode current decreases somewhat, e.g., because a negative going pulse is applied to the grid, the base and therefore the collector current decrease. This makes point B negative and therefore decreases the anode current even more. This situation can be made unstable. Finally, the circuit terminates in the off-condition again.

Most fast-switching transistors cannot withstand such a high collector-emitter voltage as 50 volts. As for proper operation of the indicator tube, this supply voltage is needed and, as this voltage will come across the transistor in the off condition, a possible change of the circuit must be considered in order to protect the transistor. One of the possible solutions to this problem is given in Fig. 3(A).

In this circuit, as well as in Fig. 1, assume that R_1 and $R_2 >> R_3$ and R_4.

By inserting R_4 across the transistor, the collector-emitter voltage in the off condition becomes

$$\frac{R_4}{R_3 + R_4} \times V_{\text{battery}}$$

For instance, if $R_3 = R_4$, the maximum collector voltage will become 25 volts.

In the on condition, R_4 has no influence upon the operation because the transistor then represents a very small impedance. In this kind of circuit, the transistor acts merely as a switch which is

IR. H. RODRIQUES DE MIRANDA is with Philips Research Laboratories, Eindhoven, Holland.

I. RUDICH is with Amperex Electronic Corporation, Hicksville, N. Y.

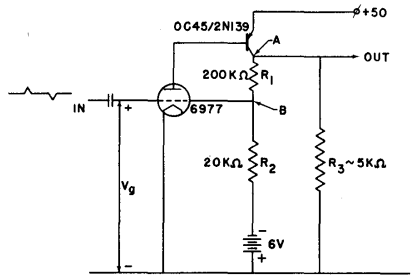

Fig. 2. Description of basic circuit

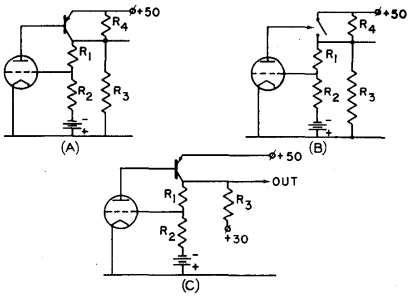

Fig. 3. Protection against high collector voltage

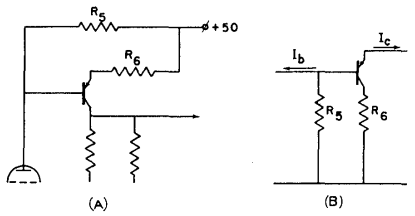

Fig. 4. Stabilization against spread and variation of β

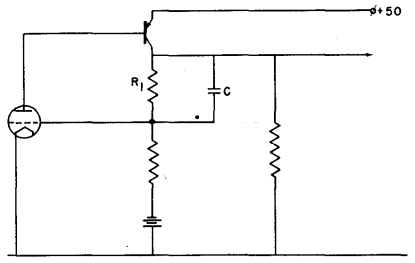

Fig. 5. Improving speed and sensitivity

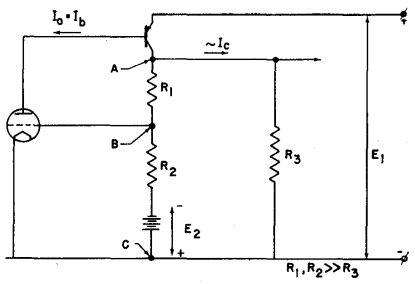

Fig. 6. Analysis of basic circuit

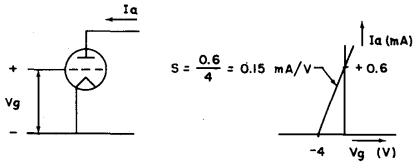

Fig. 7. Linear approximations

controlled by the base current (anode current of *6977*); see Fig. 3(B). This protection must be compensated for with a smaller output voltage and a higher battery power consumption.

In the circuit of Fig. 3(C), which is another possible solution, the power consumption is not increased but decreased. In this circuit, however, an extra battery-voltage level is needed. (This protection may not be necessary with some of the newer diffused-base transistors because of their higher breakdown voltages.)

Of course, the circuit could be designed so that even for the lowest expected value of β (base collector-current multiplication) the transistor saturates ($V_{ec} \approx OV$) in the on condition. This means, however, that for higher values of β, the transistor is overdriven very badly, which gives higher dissipation and, what is even more important, lower switching speed.

There is, however, a very simple method of stabilizing β at the cost of some loop amplification; see Figs. 4(A) and (B).

If R_6 is chosen so that it is several times the emitter resistance and R_5 is several times smaller than β times R_6 (which is the approximate base input impedance), the effective β of the circuit $\bar{\beta} = I_c/I_b$ will approximate R_5/R_6.

Reasonable values for R_5 and R_6 are 600 and 40 ohms respectively. This stabilization is set the expense of a somewhat reduced power gain.

In general

$$\frac{1}{\bar{\beta}} = \frac{1}{\beta} + \frac{R_6 + r_e}{R_5}$$

in which

$\bar{\beta}$ = effective current gain of the circuit of Fig. 4(B)
β = current gain of transistor
r_e = internal emitter resistance

The stabilization of the current gain is defined as

$$S_\beta = \frac{d\beta}{d\bar{\beta}}$$

$$\bar{\beta}^{-1} = \beta^{-1} + \frac{R_b + r_e}{R_5}$$

Differentiated to $\bar{\beta}$ gives

$$-\bar{\beta}^{-2} = -\beta^{-2}\frac{d\beta}{d\bar{\beta}}$$

or

$$\frac{d\beta}{d\bar{\beta}} = \left(\frac{\beta}{\bar{\beta}}\right)^2$$

If, for instance, the effective β is lowered by a factor 4, a stabilization of 16 results.

$$S_\beta = \frac{d\beta}{d\bar{\beta}} = \left(\frac{\beta}{\bar{\beta}}\right)^2 = \left(1 + \beta\frac{R_6 + r_e}{R_5}\right)^2$$

For example, if

$\beta = 60$
$R_6 = 40\Omega$
$r_e = 10\Omega$
$R_5 = 600$

then

$$S_\beta = \left(1 + 60\,\frac{50}{600}\right)^2 = 36$$

and

$\beta/\bar{\beta} = 6$ or $\bar{\beta} = 10$

As another example, if

$\beta = 60$
$R_6 = 40\Omega$
$r_e = 10\Omega$
$R_5 = 3{,}000\Omega$

then

$$S_\beta = \left(1 + 60\,\frac{50}{3{,}000}\right)^2 = 4$$

and

$\beta/\bar{\beta} = 2$ or $\bar{\beta} = 30$

In Fig. 5 a method is given to improve the speed and sensitivity. By shunting the resistance R_1 with a capacitor, a much greater positive feedback is obtained at the beginning of the transient, which will make the switchover much faster.

The R_1C time should be much smaller than the output pulse time.

The analysis of the circuit in Fig. 6 is given in the following.

Assumptions

The approximate electrical characteristics for the indicator tube *6977* are as follows:

heater voltage = $V_f = 1$ volt
heater current = $I_f = 30$ ma
anode voltage = $V_a = 50$ volts
maximum light output at $V_g = 0$ volt ($I_a = 0.6$ ma)
zero light output at $V_g = -4$ volts (actually 3 volts or less) (one side of filament tied to ground)

For the analysis, the following assumptions are made:

1. In this first-order approximation neglect

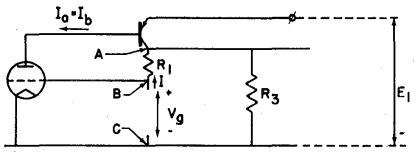

Fig. 8. Circuit analysis

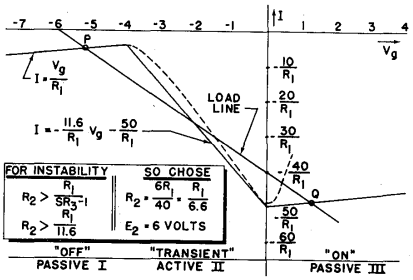

Fig. 9. Input characteristic showing negative resistance

the influence of voltage applied to the filament.

2. Assume a linear relation between grid voltage and anode current. A constant slope S is used throughout the working region of the tube; see Fig. 7.

3. Neglect the grid current in first approximation.

4. Assume β of the transistor is constant.

5. Assume the collector breakdown voltage is higher than E_1.

Analysis

One must now determine two battery voltages, E_1 and E_2, and values of three resistances, R_1, R_2, and R_3, to obtain bistable action of the circuit. (See Fig. 8.)

The value of E_1 is given by the tube requirement $E_1 = 50$ volts. The value of R_3 is determined by the fact that in the on condition the transistor should be near saturation. So, $I_c R_3 \cong E_1$, but $I_c = \beta I_a$,

thus $R_3 \cong \dfrac{E_1}{\beta I_a}$

With

$\bar{\beta} = 15$
$E_1 = 50$ volts
$I_a = 0.6$ ma
$R_3 = 5.6$ kilohms

Now there is the somewhat more complicated matter of choosing a right combination of R_1, R_2, and E_2. This choice determines the stability and triggering sensitivity. That the voltage V_{CB} should be 4 volts for off and zero volt for the on condition is only one of the requirements. Of course, R_1 and R_2 are chosen as high as possible because this will result in a high-input impedance level which makes the triggering power low.

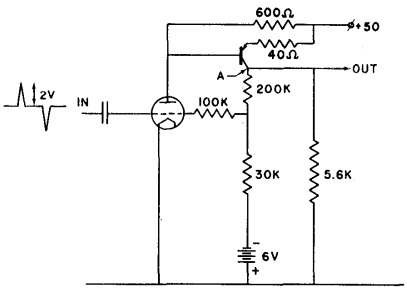

Fig. 10. Calculated basic circuit

To determine R_1, R_2, and E_2 take E_2 and R_2 out of the circuit. Looking at the terminals $B-C$ (Fig. 8) one sees negative resistance region limited on both sides by passive, and therefore, positive resistance regions.

In Fig. 7 the positive directions of V_g and I were chosen so that if an increase of I results from an increase of V_g, a passive element exist. If, however, I decreases if V decreases, an active or negative resistance element exists.

Consider how I and V_g are related.

The anode current is

$$I_a = S(V_g + V_0)$$

where $V_0 = 4$ volts.

The collector current is

$$I_c = \beta I a = \beta S(V_g + V_0)$$

The voltage across R_3 is

$$V_{AC} = R_3 1_c = \beta S R_3 (V_g + V_0) \quad (\text{assuming } R_1 >> R_3)$$

$$I = \frac{V_g - V_{AC}}{R_1}$$

$$I = \frac{-(\beta S R_3 - 1)}{R_1} V_g - \frac{\beta S R_3 V_0}{R_1} \qquad (1)$$

This equation indicates that a negative resistance can be obtained if and only if

$$\beta S R_3 > 1 \qquad (2)$$

where

$\beta = 15$
$S = 0.15$ ma/volt
$R_3 = 5.6$ kilohms $\beta S R_3 = 12.6 > 1$

If the known values are substituted in equation 1 the following results

$$I = \frac{-11.6}{R_1} V_g - \frac{50}{R_1} \qquad (3)$$

In Fig. 9 this relation of V_g and I plotted. The passive regions on both sides limiting the active or transient domain are interesting.

Region I: Off, Passive

If the grid voltage becomes more negative than $-V_0$ (-4 volts), anode and collector current stops. The differential resistance of $B-C$ now becomes R_1 (substitute in equation 1 $S=0$).

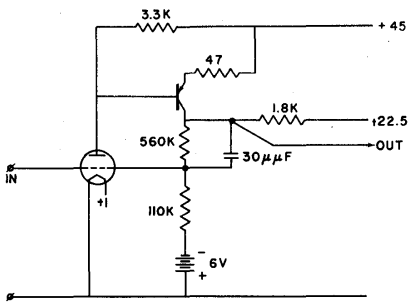

Fig. 11. Practical design

Region II: Transient, Active

This is our calculated negative resistance. The dotted line gives the better approximation.

Region III: On, Passive

If the voltage becomes greater than 0 volt, the transistor saturates (this also can occur by limiting the grid voltage swing because of the grid current or because of external diode action) and consequently stops being an active element. Now again there is a positive differential resistance of the value R_1.

If a load line is drawn, one sees that there are two stable situations (P=off and Q=on).

It was not easy to foresee that a stable on situation would result, which implies a positive grid voltage of 1.5 volts. The best solution to this dilemma is by inserting a series grid resistance of 100 kilohms as is in fact ordered by the tube manufacturer. It is not important that point B rises to a positive voltage, the grid stays very nearly at zero volt because of the grid current which starts to flow. Another way would be to choose a slightly higher value for R_3 in order to saturate the transistor at $V_g \approx -1$ volt.

Now back to the determination of R_1, R_2, and E_2. From the load line, Fig. 9, it is learned that:

1. Bistable action can only be achieved if the load line intersects the S characteristics at three points. This means that the instability criterion is

$$R_2 > \frac{R_1}{SR_3 - 1} \qquad (4)$$

or in the case

$$R_2 > \frac{R_1}{11.6} \qquad (5)$$

2. Looking at the graph it seems to be a reasonable compromise between stability and triggering sensitivity if the following is chosen

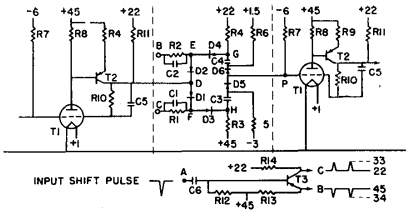

Fig. 12. Application of the circuit in a shift register

$$R_2=\frac{6R_1}{40}=\frac{R_1}{6.6}$$

and

$E_2=6$ volts

3. Now there is still some freedom in choosing the absolute value of R_2, and consequently R_1. It is chosen as high as possible, limited by the grid-cathode capacitance and switching speed. A value of 30 kilohms seems reasonable.

$$R_2=30 \text{ kilohms}$$
$$R_1=6.6\times R_2 \rightarrow R_1=200 \text{ kilohms}$$
$$E_2=6 \text{ volts}$$

4. The triggering sensitivity now is 2 volts toward both sides. Finally the basic circuit now looks like Fig. 10. Though the circuit of Fig. 10 operated satisfactorily, the actual circuit used was a little different, see Fig. 11.

As mentioned before, a collector voltage of 50 volts is above the ratings of the *OC45* which was used for its speed. As described previously (Fig. 4) R_3 now is connected to a positive voltage. Further it is seen that the feedback resistance of 560 kilohms is bypassed by a 30-micro-microfarad ($\mu\mu$f) capacitor to improve the speed. A rise time of 0.26 microsecond (μsec) and a fall time of 0.86 μsec were found.

The impedance level of the feedback voltage divider is chosen somewhat higher in the final circuit. By doing this one could avoid the 100-kilohm grid series resistance of Fig. 10 and still have a grid current limiting of the on position.

The purpose of the coupling circuit between the dashed lines is to provide point P with a positive pulse at the time a shift pulse appears at A, if the previous stage was on, and a negative pulse if the previous stage was off.

$r_1,r_2,r_3,r_4=56$ kilohms
$r_5,r_6=220$ kilohms
$r_7=100$ kilohms
$r_8=3.3$ kilohms
$r_9=47$ kilohms
$r_{10}=560$ kilohms
$r_{11}=1.8$ kilohm
$r_{12}=15$ kilohms
$r_{13},r_{14}=1$ kilohm
$c_1,c_2=47$ $\mu\mu$f
$c_3,c_4=100$ $\mu\mu$f
$c_5=30$ $\mu\mu$f
$c_6=0.001$ μf
$T_1=6977$
$T_2=OC\ 45$
$T_3=OC\ 45$
$d_{123456}=OA\ 86/1N480$

As is shown later the capacitors c_1 and c_2 act as a temporary memory during the shift pulse. During this pulse the previous stage has to be decoupled and the polarity of the output pulse at P should only be determined by the state of the previous stage before shifting.

The way the pulses take in this coupling circuit is now followed. There are three inputs B, C, and D. Points B and C receive positive and negative going pulses respectively from an external supply which serves the whole n-stage register. Point D is connected with the output of the previous stage. Therefore, point D can have two voltage levels, 45 volts (on condition) and 22 volts (off condition).

Assume that $V_D=45$ volts. The following conditions are considered:

1. Condition between two shift pulses.
2. Condition during shift pulse.

With reference to condition 1, the following applies:

$V_D=45$
$V_B=22$

d_2 is reversed-biased; c_1 is not charged.

$V_C=22$

d_1 is forward-biased and c_2 charges up to $\sim 45-22=23$ volts. d_4 is reversed-biased. d_3 is not carrying appreciable current because $V_F \sim V_H \sim 45$.

$V_H=45$
$V_G=22$

During shift pulse, the following applies:

$V_B=34$

Because of the voltage inertion of c_1, V_E lowers to 34. d_2 and d_4 are still nonconducting.

$V_C=33$

Because of the voltage inertion of c_2, V_F rises ~ 9 volts.

$V_F \approx 45+9=54$

d_3 is, therefore, during the shift, forward-biased and d_1 is reversed-biased.

$V_H \approx 54$

Therefore the positive going pulse is transferred through the lower branch and passes via c_4 and d_5 to the input of the second bistable circuit.

This circuit receives a positive pulse which switches it on if it was off and leaves it on if it was already on.

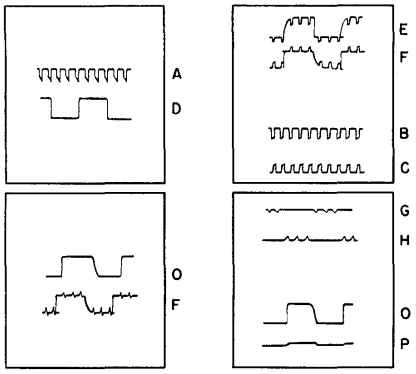

Fig. 13. Operation of shift register as shown by waveshapes

Note that a change of voltage on D, because of a possible change of state of the previous stage, does not influence the circuit during the shift pulse. Diodes d_1 and d_2 are both reversed-biased.

Assume that $V_D=22$ volts. A similar reasoning as in the 45-volt case can be given for the case in which the previous stage is off, and consequently the potential of D is 22 volts.

In this case a negative going pulse on P is found, which takes care that after the shift pulse the second stage is in the off condition.

To summarize, the voltage levels at the different points are given in Table I, which follows.

Table I. Voltage Levels

	Between Shift Pulse S		During Shift Pulse	
Points	$V_D=45$	$V_D=22$	$V_D=45$	$V_D=22$
D	45	22	45	22
B	45	45	34	34
C	22	22	33	33
E	45	22	34	11
F	45	22	56	33
G	22	22	22	11
H	45	45	56	45
P	0/−2	0/−2	>0	<−2

The purpose of c_3 and c_4 obviously is to separate the d-c level of points G and H from the pulse signal.

The diodes d_5 and d_6 in combination with r_5 and r_6 and the battery voltages −3 and +1.5 take care that the grid voltage of the *6977* can change between −3 and +1.5, without being loaded with the preceding circuit.

Testing the Circuit

For the test the input voltage at D was simulated with a square-wave generator with a voltage swing of 22 to 45. This generator was synchronized to 1/6 of the

shift-pulse frequency applied to A. The maximum shift-pulse frequency was 0.3 megacycles per second.

In the following figures the pulse patterns on the indicated points of the circuit of Fig. 12 are seen. The shift frequency was 100 kilocycles per second and the block frequency at point D was 16.7 kilocycles per second.

Conclusions

A direct read-out bistable circuit using one transistor and one high-vacuum indicator triode is described. The advantages of this circuit are:

1. Very low triggering power (~30 microwatts) because of high-input impedance (~100 kilohms).
2. High-output power available (~200 megawatts) on low-impedance level (~2 kilohms).
3. Simple circuit with fewer components than the conventional circuit.

Further, this paper contains a description of a coupling circuit which, together with a bistable circuit, provides one stage of a shift register.

Flow Gating

W. J. POPPELBAUM
NONMEMBER AIEE

Synopsis: The standard arrangement used to transfer information from one flip-flop to another requires severing elements in the connection between them. The number of diodes and triodes in these "gates" is often comparable to that in the flip-flop. This paper describes a scheme in which the transfer is effected by a simple diode going from the sending to the receiving flip-flop and by changing the supply voltage of one of the two circuits. The method appears to be attractive for small memories composed of flip-flops. It is also shown that all inputs and outputs can be tied together and that information can then be exchanged between two partially selected units.

A SIMPLE problem occurring in a computer is to transfer a zero or a one from one place to another in a selective fashion. The problem may be presented by discussing the transmission of information from one Eccles-Jordan flip-flop to another. It will be assumed that nonoverlapping voltage bands represent the zero and one signals, the bands being caused by parameter drift. An Eccles-Jordan flip-flop then has the following abstract properties (see Fig. 1): There are two low impedance outputs and two high-impedance trigger points. Points with the same number are in phase; points with a different number are out of phase. (In general, especially in so-called last-moving-point flip-flops, there may be a time lag between a trigger point and the corresponding "in phase output.") "Out 0" in the one state corresponds to the zero state of the flip-flop. "Out 1" in the one state corresponds to the one state of the flip-flop. Transmission of information between two flip-flops can be accomplished by connecting the outputs of the first to the inputs of the second by means of switches. The direction of flow of information will be determined by the asymmetry of the impedances; the lower impedance drives the higher impedance. This does not mean that the voltages in the connecting wires lie necessarily in the zero or one bands while the flip-flops are tied together. It does mean, however, that once the switches are opened, flip-flop no. 1 has copied the state of flip-flop no. 2 independently of the order in which the switches were opened.

The important point to note is that a gate in the form of two switches severs the connections which transmit information, thus allowing the triggered circuit to seek its levels inside the permitted bands. This severing action can be achieved by adding two diodes in front of trigger point 0 and trigger point 1. In Fig. 2 the situation is indicated for the case of positive logic (one signal voltages greater than zero signal voltages). If the points marked "in" are kept at the zero level, the trigger points are effectively disconnected, except perhaps for some small currents due to the difference in the applied zero voltage and the natural zero voltage of a trigger point. In order to allow the trigger points to seek their own level, the two inputs can be held at the most negative zero voltage. This will be termed a floating output of the diodes.

Note that one can also pull the trigger points down by using diodes in the opposite direction. Then, the floating output would be caused by the most positive one voltage. The discussion will, however, be limited to the first case.

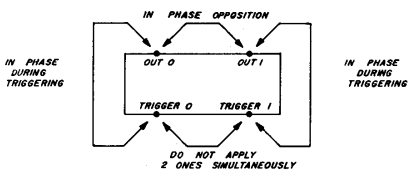

Fig. 1. Abstract representation of an Eccles-Jordan flip-flop

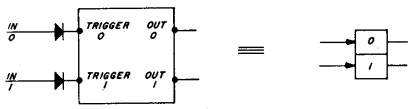

Fig. 2. Make-up of an Eccles-Jordan flip-flop

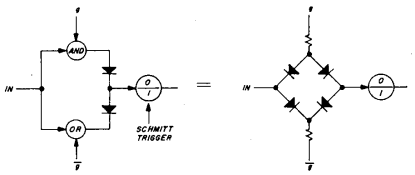

Fig. 3. Double gating a Schmitt trigger-type flip-flop

Transmitting information between two Eccles-Jordan flip-flops with input diodes now only necessitates the use of two "and" circuits with a sufficiently low output impedance. To inhibit the flow of information, a zero signal is injected into the second inputs of the "and's". This double-gating system can be simplified by setting the first flip-flop to the standard zero state by an initial clearing signal which is turned off before a single "and" (connected between the one sides) receives the gating one-signal which causes the conditional transfer. This clearing and gating is naturally slower than double gating since it essentially involves two distinct operations. It should be noted that in both these gating systems the trigger point is either left as it is (floating or zero input) or pushed up.

In flip-flops of the nonsymmetric variety, like a Schmitt trigger (operationally equivalent to an Eccles-Jordan with only two in-phase points accessi-

---

W. J. POPPELBAUM is with the University of Illinois, Urbana, Ill.

The author wishes to thank Mr. Neil Wiseman and Mr. Sylvian Ray of the Digital Computer Laboratory for their help with calculations and experiments. This work was supported in part by the Office of Naval Research and the Atomic Energy Commission under AEC Contract AT(11-1)-415.