First Draft of a Report on the EDVAC[*]

By John von Neumann

Von Neumann's computing responsibility on the Los Alamos Manhattan Project led him to Bell Laboratories and Harvard University in search of additional computing power, but neither facility could meet his needs. In 1944 he heard by accident of the construction of computing instruments at the University of Pennsylvania's Moore School of Electrical Engineering. On his first visit to the Moore School a few weeks later, he found that a programmable electronic calculator (ENIAC) was completely designed but not yet completed and that work on a stored-program computer (EDVAC) was just beginning. He arranged for the Los Alamos project to be the first major user of the ENIAC and served for the next year and a half as a consultant to the EDVAC project.

In 1945 *von Neumann wrote First Draft of a Report on the EDVAC. Using logical symbolism from the research of Warren McCulloch and Walter Pitts on neural networks in the human brain and such biological vocabulary as "organ" and "neuron," von Neumann was able to describe the stored-program computer in a way that illuminated the logical functions of the necessary constituent parts without having to consider the particular engineering components that might be employed in any given computer. His report also pointed to the analogy between the stored-program computer and human brain. The first widely circulated document on the stored-program computer, it served as a primer for subsequent stored-program computer design based on the serial delay line memory.*

Credit for the intellectual content of this document has been disputed for forty years. The Draft Report, as its name indicated, was intended as a preliminary document. However, Herman Goldstine, the Army's project administrator and a close associate of von Neumann, distributed it throughout the United States and Britain without von Neumann's knowledge. Because von Neumann's name appeared alone on the report, he received all the credit for the stored-program concept. Although there is no question that he was the sole author, several members of the Moore &hool staff claim that the ideas described in the report were group results and should not be credited to von Neumann alone. J. Presper Eckert and John W Mauchly, the senior members of the Moore School computer team, have contended further that the stored-program concept was fundamentally theirs. The Moore School staff had begun to consider the prospects of storing instructions in the machine at least eight months before von Neumann's involvement with the project, and the ideas that went into the report were discussed in great detail by Eckert, Mauchly, Arthur Burks, Goldstine, and von Neumann before the writing of the report. However, it is also clear the von Neumann was an important catalyst and major contributor to these discussions and that the description of these ideas in a theoretical form, stripped of all engineering details, was his contribution alone. Much of the controversy that ensued unquestionably stems from the differing perspectives of practicing scientists and engineers and from the values they attach to theoretical constructs, technical feasibility, engineering design, and construction.

The Draft Report was the source of an additional friction. Eckert and Mauchly were interested in the commercial prospects of the computer and expected to obtain the patent rights to both the ENIAC and EDVAC. They were concerned that the Draft Report would jeopardize these rights, for it disclosed the technology before they had the opportunity to flle patent. As they had feared, a federal court ruled many years later that the distribution of the Draft Report in 1945 rendered it a public disclosure, thereby invalidating their patent claim flied in 1947.

Excerpted with permission from Papers of John von Neumann on Computing and Computer Theory, by William Aspray, Charles Babbage Institute Reprint Series, MlT Press, 1987.

[*] Contract No. W-670-ORD-4962. Between the United States Army Ordinance Department and the University of Pennsylvania. Moore School of Electrical Engineering, University of Pennsylvania, June 30, 1945

1.0 DEFINITIONS

1.1. The considerations which follow deal with the structure of a *very high speed automatic digital computing system,* and in particular with its *logical control.* Before going into specific details, some general explanatory remarks regarding these concepts may be appropriate.

1.2. An *automatic computing system* is a (usually highly composite) device, which can carry out instructions to perform calculations of a considerable order of complexity-e.g. to solve a non-linear partial differential equation in 2 or 3 independent variables numerically.

The instructions which govem this operation must be given to the device in absolutely exhaustive detail. They include all numerical information which is required to solve the problem under consideration: Initial and boundary values of the dependent variables, values of fixed parameters (constants), tables of fixed functions which occur in the statement of the problem. These instructions must be given in some form which the device can sense: Punched into a system of punchcards or on teletype tape, magnetically impressed on steel tape or wire, photographically impressed on motion picture film, wired into one or more, fixed or exchangeable plugboards - this list being by no means necessarily complete. All these procedures require the use of some code, to express the logical and the algebraical definition of the problem under consideration, as well as the necessary numerical material (cf. above).

Once these instructions are given to the device, it must be able to carry them out completely and without any need for further intelligent human intervention. At the end of the required operations the device must record the results in one of the forms referred to above. The results are numerical data; they are a specified part of the numerical material produced by the device in the process of carrying out the instructions referred to above.

1.3. It is worth noting, however, that the device will in general produce essentially more numerical material (in order to reach the results) than the (final) results mentioned. Thus only a fraction of its numerical output will have to be recorded as indicated in 1.2, the remainder will only circulate in the interior of the device, and never be recorded for human sensing. This point will receive closer consideration subsequently, in particular in

1.4. The remarks of 1.2 on the desired automatic functioning of the device must, of course, assume that it functions faultlessly. Malfunctioning of any device has, however, always a finite probability- and for a complicated device and a long sequence of operations it may not be possible to keep this probability negligible. Any error may vitiate the entire output of the device. For the recognition and correction of such malfunctions intelligent human intervention will in general be necessary.

However, it may be possible to avoid even these phenomena to some extent. The device may recognize the most frequent malfunctions automatically, indicate their presence and location by externally visible signs, and then stop. Under certain conditions it might even carry out the necessary correction automatically and continue. (Cf.)

2.0 MAIN SUBDIVISION OF THE SYSTEM

2.1. In analyzing the functioning of the contemplated device, certain classificatory distinctions suggest themselves immediately.

2.2. First: Since the device is primarily a computor, it will have to perform the elementary operations of arithmetics most frequently. There are addition subtraction, multiplication and division: +, □, □, ÷. It is therefore reasonable that it should contain specialized organs for just these operations.

It must be observed, however, that while this principle as such is probably sound, the specific way in which it is realized requires close scrutiny. Even the above list of operations +, □, □, ÷, is not beyond doubt. It may be extended to include such operations as $(\ )^{1/2}$, $(\ )^{1/3}$, sgn, 1 1, also $\log_{10}$, $\log_2$, ln, sin and their inverses, etc. One might also consider restricting it, e.g. omitting ÷ and even □. One might also consider more elastic arrangements. For some operations radically different procedures are conceivable, e.g. using successive approximation methods or function tables. These matters will be gone into in. At any rate a central arithmetical part of the device will probably have to exist, and this constitutes the first specific part: CA.

2.3. Second: The logical control of the device, that is the proper sequencing of its operations can be most efficiently carried out by a central control organ. If the device is to be *elastic,* that is as nearly as possible *all purpose,* then a distinction must be made between the specific instructions given for and defining a particular problem, and the general control organs which see to it that these instructions - no matter what they are - are carried out. The former must be stored in some way - in existing devices this is. done as indicated in 1.2 - the latter are represented by definite operating parts of the device. By the *central control* we mean this later function only, and the organs which perform it form *the second specific part: CC.*

2.4. Third: Any device which is to carry out long and complicated sequences of operations (specifically of calculations) must have a considerable memory . At least the four following phases of its operation require a memory:

(a) Even in the process of carrying out a multiplication or a division, a series of intermediate (partial) results must be remembered. This applies to a lesser extent even to additions and subtractions (when a carry digit may have to be carried over several positions), and to a greater extent to $(x)^{1/2}$, $(x)^{1/3}$, if these operations are wanted. (Cf.).

(b) The instructions which govern a complicated problem may constitute a considerable material, particularly so, if the code is circumstantial (which it is in most arrangements). This material must be remembered.

(c) In many problems specific functions play an essential role. They are usually given in form of a table. Indeed in some cases this is the way in which they are given by experience (e.g. the equation of state of a substance in many hydrodynamical problems), in other cases they may be given by analytical expressions, but it may nevertheless be simpler and quicker to obtain their values from a fixed tabulation, than to compute them anew (on the basis of the analytical definition) whenever as value is required. It is usually convenient to have tables of a moderate number of entries only (100-200) and to use interpolation. Linear and even quadratic interpolation will not be sufficient in most cases, so it is best to count on a standard of cubic or bi-quadratic (or even higher order) interpolation, cf.

Some of the functions mentioned in the course of 2.2 may be handled in this way: $\lg_{10}$, $\lg_2$, ln, sin, and their inverses, possibly also $(\ )^{1/2}$, $(\ )^{1/3}$. Even the reciprocal might be treated in this manner, thereby reducing ÷ to □.

(d) For partial differential equations the initial conditions and the boundary conditions may constitute an extensive numerical material, which must be remembered throughout a given problem.

(e) For partial differential equations of the hyperbolic or parabolic type, integrated along a variable t, the (intermediate) results belonging to the cycle t must be remembered for the calculation of the cycle t + dt. This material is much of the type (d), except that it is not put into the device by human operators, but produced (and probably subsequently again removed and replaced by the corresponding data for t + dt) by the device itself, in the course of its automatic operation.

(f) For total differential equations (d), (e) apply too, but they require smaller memory capacities. Further memory requirements of the type (d) are required in problems which depend on given constants, fixed parameters, etc.

(g) Problems which are solved by successive approximations (e.g. partial differential equations of the elliptic type, treated by relaxation methods) require a memory of the type (e): The (intermediate) results of each approximation must be remembered, while those of the next one are being computed.

(h) Sorting problems and certain statistical experiments (for which a very high speed device offers an interesting opportunity) require a memory for the material which is being treated.

2.5. To sum up the third remark: The device requires a considerable memory. While it appeared, that various parts of this memory have to perform functions which differ somewhat in their nature and considerably in their purpose, it is nevertheless tempting to treat the entire memory as one organ, and to have its parts even as interchangeable as possible for the various functions enumerated above. This point will be considered in detail cf.

At any rate the total *memory* constitutes *the third specific part of the device: M.*

2.6. The three specific parts CA, CC (together C) and M correspond to the *associative* neurons in the human nervous system. It remains to discuss the equivalents of the *sensory* or *afferent* and the *motor* or *efferent* neurons. These are the *input* and the *output* organs of the device, and we shall now consider them briefly.

In other words: All transfers of numerical (or other) information between the parts C and M of the device must be effected by the mechanisms contained in these parts. There remains, however, the necessity of getting the original definitory information from outside into the device, and also of getting the final information, the results, from the device into the outside.

By the outside we mean media of the type described in 1.2: Here information can be produced more or less directly by human action (typing, punching, photographing light impulses produced by keys of the same type, magnetizing metal tape or wire in some analogous manner, etc.), it can be statically stored, and finally sensed more or less directly by human organs.

The device must be endowed with the ability to maintain the input and output (sensory and motor) contact with some specific medium of this type (cf. 1.2): That medium will be called the *outside recording medium of the device: R*. Now we have: 2.7. Fourth: The device must have organs to transfer (numerical or other) information from R into its specific parts C and M. These organs form its input, the fourth specific part: *I*. It will be seen, that it is best to make all transfers from R (by I) into M, and never directly into C (cf.).

2.8. Fifth: The device must have organs to transfer (presumably only numerical information) from its specific parts C and M into R. These organs form its *output,* the *fifth specific part: O*. It will be seen that it is again best to make all transfers from M (by O) into R, and never directly from C (cf.).

2.9. The output information, which goes into R, represents, of course, the final results of the operation of the device on the problem under consideration. These must be distinguished from the intermediate results, discussed e.g. in 2.4, (e)-(g), which remain inside M. At this point an important question arises: (Quite apart from its attribute of more or less direct accessibility to human action and perception R has also the properties of a memory. Indeed, it is the natural medium for long time storage of all the information obtained by the automatic device on various problems. Why is it then necessary to provide for another type of memory within the device M?) Could not all, or at least some functions of M - preferably those which involve great bulks of information - be taken over by R?

Inspection of the typical functions of M, as enumerated in 2.4, (a)-(h), shows this: It would be convenient to shift (a) (the short-duration memory required while an arithmetical operation is being carried out) outside the device, i.e. from M into R. (Actually (a) will be inside the device but in CA rather than in M. Cf. the end of 12.2) All existing devices, even the existing desk computing machines, use the equivalent of M at this point. However (b) (logical instructions) might be sensed from outside, i.e. by I from R, and the same goes for (c) (function tables) and (e), (g) (intermediate results). The latter may be conveyed by O to R when the device produces them, and sensed by I from R when it needs them. The same is true to some extent of (d) (initial conditions and parameters) and possibly even of (f) (intermediate results from a total differential equation). As to (h) (sorting and statistics), the situation is somewhat ambiguous: In many cases the possibility of using M accelerates matters decisively, but suitable blending of the use of M with a longer range use of R may be feasible without serious loss of speed and increase the amount of material that can be handled considerably.

Indeed, all existing (fully or partially automatic) computing devices use R - as a stack of punchcards or a length of teletype tape - for all these purposes (excepting (a), as pointed out above). Nevertheless it will appear that a really high speed device would be very limited in its usefulness, unless it can rely on M, rather than on R, for all the purposes enumerated in 2.4, (a)-(h), with certain limitations in the case of (e), (g), (h), (cf.).

3.0 PROCEDURE OF DISCUSSION

3.1. The classification of 2.0 being completed, it is now possible to take up the five specific parts into which the device was seen to be subdivided, and to discuss them one by one. Such a discussion must bring out the features required for each one of these parts in itself, as well as in their relations to each other. It must also determine the specific procedures to be used in dealing with numbers from the point of view of the device, in carrying out arithmetical operations, and providing for the general logical control. All questions of timing and of speed, and of the relative importance of various factors, must be settled within the framework of these considerations.

3.2. The ideal procedure would be, to take up the five specific parts in some definite order, to treat each one of them exhaustively, and go on to the next one only after the predecessor is completely disposed of. However, this seems hardly feasible. The desired features of the various parts, and the decisions based on them, emerge only after a somewhat zigzagging discussion. It is therefore necessary to take up one part first, pass after an incomplete discussion to a second part, return after an equally incomplete discussion of the latter with the combined results to the first part, extend the discussion of the first part without yet concluding it, then possibly go on to a third part, etc. Furthermore, these discussions of specific parts will be mixed with discussions of general principles, of arithmetical procedures, of the elements to be used, etc.

In the course of such a discussion the desired features and the arrangements which seem best suited to secure them will crystallize gradually until the device and its control assume a fairly definite shape. As emphasized before, this applies to the physical device as well as to the arithmetical and logical arrangements which govern its functioning.

3.3. In the course of this discussion the viewpoints of 1.4, concerned with the detection, location, and under certain conditions even correction, of malfunctions must also receive some consideration. That is, attention must be given to facilities for *checking* errors. We will not be able to do anything like full justice to this important subject, but we will try to consider it at least cursorily whenever this seems essential (cf.).

4.0 ELEMENTS, SYNCHRONISM NEURON ANALOGY

4.1. We begin the discussion with some general remarks:

Every digital computing device contains certain relay like *elements,* with discrete equilibria. Such an element has two or more distinct states in which it can exist indefinitely. These may be perfect equilibria, in each of which the element will remain without any outside support, while appropriate outside stimuli will transfer it from one equilibrium into another. Or, alternatively, there may be two states, one of which is an equilibrium which exists when there is no outside support, while the other depends for its existence upon the presence of an outside stimulus. The relay action manifests itself in the emission of stimuli by the element whenever it has itself received a stimulus of the type indicated above. The emitted stimuli must be of the same kind as the received one, that is, they must be able to stimulate other elements. There must, however, be no energy relation between the received and the emitted stimuli, that is, an element which has received one stimulus, must be able to emit several of the same intensity. In other words: Being a relay the element must receive its energy supply from another source than the incoming stimulus

In existing digital computing devices various mechanical or electrical devices have been used as elements: Wheels, which can be locked into anyone of ten (or more) significant positions, and which on moving from one position to another transmit electrical pulses that may cause other similar wheels to move; single or combined telegraph relays, actuated by an electromagnet and opening or closing electric circuits; combinations of these two elements; - and finally there exists the plausible and tempting possibility of using vacuum tubes, the grid acting as a valve for the cathode-plate circuit. In the last mentioned case the grid may also be replaced by deflecting organs, i.e. the vacuum tube by a cathode ray tube - but it is likely that for some time to come the greater availability and various electrical advantages of the vacuum tubes proper will keep the first procedure in the foreground.

Any such device may time itself autonomously, by the successive reaction times of its elements. In this case all stimuli must ultimately originate in the input. Alternatively, they may have their timing impressed by a fixed clock, which provides certain stimuli that are necessary for its functioning at definite periodically recurrent moments. This clock may be a rotating axis in a mechanical or a mixed, mechanico-electrical device; and it may be an electrical oscillator (possibly crystal controlled) in a purely electrical device. If reliance is to be placed on synchronisms of several distinct sequences of operations performed simultaneously by the device, the clock impressed timing is obviously preferable. We will use the term *element* in the above defined technical sense, and call the device *synchronous* or *asynchronous,* according to whether its timing is impressed by a clock or autonomous, as described above.

4.2. It is worth mentioning, that the neurons of the higher animals are definitely elements in the above sense. They have all-or-none character, that is two states: Quiescent and excited. They fulfill the requirements of 4.1 with an interesting variant: An excited neuron emits the standard stimulus along many lines (axons). Such a line can, however, be connected in two different ways to the next neuron: First: In an *excitatory synapsis,* so that the stimulus causes the excitation of that neuron. Second: In an *inhibitory synapsis,* so that the stimulus absolutely prevents the excitation of that neuron by any stimulus on any other (excitatory) synapsis. The neuron also has a definite reaction time, between the reception of a stimulus and the emission of the stimuli caused by it, the *synaptic delay.*

Following W. PITTS and W. S. MACCULLOCH ("A logical calculus of the ideas immanent in nervous activity", Bull. Math. Biophysics. Vol. 5 (1943), pp. 115-133) we ignore the more complicated aspects of neuron functioning: Thresholds, temporal summation, relative inhibition, changes of the threshold by after effects of stimulation beyond the synaptic delay, etc. It is, however, convenient to consider occasionally neurons with fixed thresholds 2 and 3, that is neurons which can be excited only by (simultaneous) stimuli on 2 or 3 excitatory synapses (and none on an inhibitory synapsis). Cf.

It is easily seen, that these simplified neuron functions can be imitated by telegraph relays or by vacuum tubes. Although the nervous system is presumably asynchronous (for the synaptic delays), precise synaptic delays can be obtained by using synchronous setups. Cf.

4.3. It is clear, that a very high speed computing device should ideally have vacuum tube elements. Vacuum tube aggregates like counters and scalers have been used and found reliable at reaction times (synaptic delays) as short as a microsecond (= 10^{-6} seconds), this is a performance which no other device can approximate. Indeed: Purely mechanical devices may be entirely disregarded and practical telegraph relay reaction times are of the order of 10 milliseconds (= 10^{-2} seconds) or more. It is interesting to note that the synaptic time of a human neuron is of the order of a millisecond (=10^{-3} seconds).

In the considerations which follow we will assume accordingly, that the device has vacuum tubes as elements. We will also try to make all estimates of numbers of tubes involved, timing, etc. on the basis, that the types of tubes used are the conventional and commercially available ones. That is, that no tubes of unusual complexity or with fundamentally new functions are to be used. The possibilities for the use of new types of tubes will actually become clearer and more definite after a thorough analysis with the conventional types (or some equivalent elements, cf.) has been carried out.

Finally it will appear that a synchronous device has considerable advantages (cf.).

5.0 PRINCIPLES GOVERNING THE ARITHMETICAL OPERATIONS

5.1. Let us now consider certain functions of the first specific part: the central arithmetical part CA.

The element in the sense of 4.3, the vacuum tube used as a current valve or *gate* is an all-or-none device, or at least it approximates one: According to whether the grid bias is above or below cut-off, it will pass current or not. It is true that it needs definite potentials on all its electrodes in order to maintain either state, but there are combinations of vacuum tubes which have perfect equilibria: Several states in each of which the combination can exist indefinitely, without any outside support, while appropriate outside stimuli (electric pulses) will transfer it from one equilibrium into another. These are the so called *trigger circuits,* the basic one having two equilibria and containing two triodes or one pentode. The trigger circuits with more than two equilibria are disproportionately more involved.

Thus, whether the tubes are used as gates or as triggers, the all-or-none, two equilibrium arrangements are the simplest ones. Since these tube arrangements are to handle numbers by means of their digits, it is natural to use a system of arithmetic in which the digits are also two valued. This suggests the use of the binary system.

The analogs of human neurons, discussed in 4.2-4.3 are equally all-or-none elements. It will appear that they are quite useful for all preliminary , orienting considerations on vacuum tube systems (cf.). It is therefore satisfactory that here too, the natural arithmetical system to handle is the binary one.

5.2. A consistent use of the binary system is also likely to simplify the operations of multiplication and division considerably. Specifically it does away with the decimal multiplication table, or with the alternative double procedure of building up the multiples by each multiplier or quotient digit by additions first, and then combining these (according to positional value) by a second sequence of additions or subtractions. In other words: Binary arithmetics has a simpler and more one-piece logical structure than any other, particularly than the decimal one.

It must be remembered, of course, that the numerical material which is directly in human use, is likely to have to be expressed in the decimal system. Hence, the notations used in R should be decimal. But it is nevertheless preferable to use strictly binary procedures in CA, and also in whatever numerical material may enter into the central control CC. Hence M should store binary material only.

This necessitates incorporating decimal-binary and binary-decimal conversion facilities into I and O. Since these conversions require a good deal of arithmetical manipulating, it is most economical to use CA, and hence for coordinating purposes also CC, in conjunction with I and O. The use of CA implies, however, that all arithmetics used in both conversions must be strictly binary. For details cf.

5.3. At this point there arises another question of principle.

In all existing devices where the element is not a vacuum tube the reaction time of the element is sufficiently long to make a certain telescoping of the steps involved in addition, subtraction, and still more in multiplication and division, desirable. To take a specific case consider binary multiplication. A reasonable precision for many differential equation problems is given by carrying 8 significant decimal digits, that is by keeping the relative rounding-off errors below 10^{-8}. This corresponds to 2^{-27} in the binary system that is to carrying 27 significant binary digits. Hence a multiplication consists of pairing each one of 27 multiplicand digits with each one of 27 multiplier digits, and forming product digits 0 and 1 accordingly, and then positioning and combining them. These are essentially $27^2 = 729$ steps, and the operations of collecting and combining may about double their number. So 1000-1500 steps are essentially right.

It is natural to observe that in the decimal system a considerably smaller number of steps obtains: $8^2 = 64$ steps, but possibly doubled, that is about 100 steps. However, this low number is purchased at the price of using a multiplication table or otherwise increasing or complicating the equipment. At this price the procedure can be shortened by more direct binary artifices, too, which will be considered presently. For this reason it seems not necessary to discuss the decimal procedure separately.

5.4. As pointed out before, 1000-1500 successive steps per multiplication would make any non vacuum tube device inacceptably slow. All such devices, excepting some of the latest special relays, have reaction times of more than 10 milliseconds, and these newest relays (which may have reaction times down to 5 milliseconds) have not been in use very long. This would give an extreme minimum of 10-15 seconds per (8 decimal digit) multiplication, whereas this time is 10 seconds for fast modem desk computing machines, and 6 seconds for the standard I.B.M multipliers. (For the significance of these durations, as well as of those of possible vacuum tube devices, when applied to typical problems, cf.).

The logical procedure to avoid these long durations, consists of telescoping operations, that is of carrying out simultaneously as many as possible. The complexities of carrying prevent even such simple operations as addition or subtraction to be carried out at once. In division the calculation of a digit cannot even begin unless all digits to its left are already known. Nevertheless considerable simultaneisations are possible: In addition or subtraction all pairs of corresponding digits can be combined at once, all first carry digits can be applied together in the next step, etc. In multiplication all the partial products of the form (multiplicand) □ (multiplier digit) can be formed and positioned simultaneously - in the binary system such a partial product is zero or the multiplicand, hence this is only a matter of positioning. In both addition and multiplication the above mentioned accelerated forms of addition and subtraction can be used. Also, in multiplication the partial products can be summed up quickly by adding the first pair of pair sums together simultaneously with the second one, the third one, etc.; and so on until all terms are collected. (Since 27 □ 2^5, this allows to collect 27 partial sums - assuming a 27 binary digit multiplier - in 5 addition times. This scheme is due to H. Aiken.)

Such accelerating, telescoping procedures are being used in all existing devices. (The use of the decimal system, with or without further telescoping artifices is also of this type, as pointed out at the end of 5.3. It is actually somewhat less efficient than purely diadic procedures. The arguments of 5.1 - 5.2 speak against considering it here.) However, they save time only at exactly the rate at which they multiply the necessary equipment, that is the number of elements in the device: Clearly if a duration is halved by systematically carrying out two additions at once, double adding equipment will be required (even assuming that it can be used without disproportionate control facilities and fully efficiently), etc.

This way of gaining time by increasing equipment is fully justified in non vacuum tube element devices, where gaining time is of the essence, and extensive engineering experience is available regarding the handling of involved devices containing many elements. A really all-purpose automatic digital computing system constructed along these lines must, according to all available experience, contain over 10,000 elements.

5.5. For a vacuum tube element device on the other hand, it would seem that the opposite procedure holds more promise.

As pointed out in 4.3, the reaction time of a not too complicated vacuum tube device can be made as short as one microsecond. Now at this rate even the unmanipulated duration of the multiplication, obtained in 5.3 is unacceptable: 1000-1500 reaction times amount to 1-1.5 milliseconds, and this is so much faster than any conceivable non vacuum tube device, that it actually produces a serious problem of keeping the device balanced, that is to keep the necessarily human supervision beyond its input and output ends in step with its operations. (For details of this cf.).

Regarding other arithmetical operations this can be said: Addition and subtraction are clearly much faster than multiplication. On a basis of 27 binary digits (cf. 5.3), and taking carrying into consideration, each should take at most twice 27 steps, that is about 30-50 steps or reaction times. This amounts to .03-.05 milliseconds. Division takes, in this scheme where shortcuts and telescoping have not been attempted in multiplying and the binary system is being used, about the same number of steps as multiplication (cf.). Square rooting is usually and in this scheme too, not essentially longer than dividing.

5.6. Accelerating these arithmetical operations does therefore not seem necessary - at least not until we have become thoroughly and practically familiar with the use of very high speed devices of this kind, and also properly understood and started to exploit the entirely new possibilities for numerical treatment of complicated problems which they open up. Furthermore it seems questionable whether the method of acceleration by telescoping processes at the price of multiplying the number of elements required would in this situation achieve its purpose at all: The more complicated the vacuum tube equipment - that is, the greater the number of elements required - the wider the tolerances must be. Consequently any increase in this direction will also necessitate working with longer reaction times than the above mentioned one of one microsecond. The precise quantitative effects of this factor are hard to estimate in a general way-but they are certainly much more important for vacuum tube elements than for mechanical or for telegraph relay ones.

Thus it seems worth while to consider the following viewpoint: The device should be as simple as possible, that is, contain as few elements as possible. This can be achieved by never performing two operations simultaneously, if this would cause a significant increase in the number of elements required. The result will be that the device will work more reliably and the vacuum tubes can be driven to shorter reaction times than otherwise.

5.7. The point to which the application of this principle can be profitably pushed will, of course, depend on the actual physical characteristics of the available vacuum tube elements. It may be, that the optimum is not at a 100% application of this principle and that some compromise will be found to be optimal. However, this will always depend on the momentary state of the vacuum tube technique, clearly the faster the tubes are which will function reliably in this situation, the stronger the case is for uncompromising application of this principle. It would seem that already with the present technical possibilities the optimum is rather nearly at this uncompromising solution.

It is also worth emphasizing that up to now all thinking about high speed digital computing devices has tended in the opposite direction: Towards acceleration by telescoping processes at the price of multiplying the number of elements required. It would therefore seem to be more instructive to try to think out as completely as possible the opposite viewpoint: That one of absolutely refraining from the procedure mentioned above, that is of carrying out consistently the principle formulated in 5.6. We will therefore proceed in this direction.

Reprinted from "Great Papers in Computer Science", Phillip Laplante (ed), Burlington County College, IEEE Press, New York.