

It now appears possible that the techniques of handling liquid helium will eventually make this device practical for vehicle-borne slave computers.

System Compatibility. When computers are integrated into a system, there immediately arises the problem of their compatibility. In some instances, slave functions may be quite loosely coupled to the master control center. Logical devices operating in the submicrosecond range must take into account the fact that transit time of pulses, even at the speed of light, becomes exceedingly significant at these high switching speeds. The netting of several slave computers in the millimicrosecond range with their central master is impossible under present data link standards. Added to this is the Doppler effect incurred when some of the slave functions move about the region at a speed several times sonic velocity. The need for absolute compatibility could rule out completely recent ideas for the use of burst communication techniques between master and slave stations.

Radar Accuracy. Serious limitations might very well be placed on the theoretical accuracy with which both the position of a target and its velocity can be determined simultaneously. It appears to be an extension of Heisenberg's uncertainty principle, which places an upper limit on the accuracy with which both position and velocity may be obtained simultaneously.

Information Capacity. Another area of interest occurs when one considers the gigantic task of performing a running inventory of our aero-space for the exact position of all friendly nonintruding vehicles, yet such a feat may well be necessary. This problem must be con-

sidered along with the problem of equivocation, which subject this paper has touched on most gingerly, as the more quantitative aspects of this subject cannot be discussed in a paper of this sort. Shannon and others have pointed out that as long as the information rate is kept below the capacity of the system, techniques exist whereby nearly errorless information retrieval is theoretically feasible. The price, other than high redundancy, which must be paid will result in intolerable time delays, or enormous increases in information capacity. This, coupled with the gargantuan task of keeping track of all vehicles in the aerospace, appears to be almost beyond the bounds of practical solution.

Equivocation and Redundancy. It appears from here that the most pressing problem facing the air defense system designer, and consequently the designer of air defense computers, is that of processing information in the presence of almost over-riding equivocation. Unfortunately, where redundancy is needed to buy off this equivocation it tends to manifest itself in the costly function of weapon commitment. Here, in the limiting case (that of completely over-riding equivocation), one can envision the illogical solution of committing every available weapon simultaneously. Even solutions approaching this to a small degree can become logistically nonfeasible; thus, everyone in air defense is faced with this most formidable challenge. Perhaps there is another approach to this problem, along the lines of the gaming computer which was mentioned earlier. A system controlled by such a computer might, in the presence of completely over-riding equivocation, be able to draw judgments, through the process of

"statistical inference," from the meager information available. This presupposes, then, that such a computer could retain in storage all conceivable intrusion patterns, together with their appropriate counter-moves, and has the facility to select from meager inputs the most probable of all statistically possible solutions. Here in this ultimate form one departs from the cybernetic character of the problem, only to become involved in a sophisticated game of chess against a player who is a recognized master of that game.

Epilogue

It seems reasonable that a group of engineers with backgrounds in computers would also have a primary interest in the economics involved in the use of computers for air defense. Therefore, the author will conclude on this note, for there is considerable correlation between the professional challenge involved and the fraction of the military budget directed toward computer functions.

It appears at this time that the total investment in ground-based computer functions will not be less than 3.3 billion dollars by the end of the 1960 decade. The investment in vehicular-borne computer functions will start to become an important factor during this period and will undoubtedly exceed 78 million dollars. In addition, leased facilities for the various support functions involved could very well amount to 11 million dollars annually. These figures include only those computer functions which are considered closely associated with the Air Defense Mission. There is good reason for the computer engineer to look upon both the material and professional roles implied herein with unusual seriousness.

Microprogramming

M. V. WILKES

BEFORE going into the subject matter of this paper, the author will report, briefly, the state of the digital computer art in Britain. Just before leaving England for Philadelphia, the author attended an electronic computer exhibition which opened in London. This was the first exhibition of its kind to be held in Europe. It was a genuine computer exhibition, not only an exhibition of computer components, and there were 11 stored-

program digital computers being demonstrated. The exhibition was at Olympia, a large exhibition building in the west of London, where many of the big exhibitions are held. The computers were not occupying the largest hall at Olympia, but, nevertheless, it is a source of pride to those who have been associated with the British computer industry that it should be possible to stage an exhibition on such a scale.

Anyone who promises to give a lecture or an address and is asked for a title in advance is likely to give a vague one, in order that he need not decide what to say until immediately before he is due to speak. The author's title perhaps comes into this category since "microprogramming" is a term which has been used in so many different senses that no one is now quite sure what it means. Generally, however, it is used in some context which has to do with the design of the control circuits of a digital computer,¹⁻⁴ and that is the sense in which it will be used here.

M. V. WILKES is with Cambridge University, Cambridge, England.

Wilkes—Programming

The author feels some responsibility for the term since he was, he believes, the first to use it in this way when he gave a lecture at a conference held at Manchester University in July 1951.^{5,6}

Up to that time it appeared that the sequencing units of digital computers had generally been designed on an *ad hoc* basis. The design engineer would cover a sequence of large sheets of paper with block diagrams of gates and flip-flops until he found an arrangement which produced the waveforms he required, and one that appeared to be reasonably economical. The author's aim was to suggest a mode of design which would not only be more systematic, but would lead to a sequencing unit with the same underlying structure whatever the order code of the machine.

The need for a sequencing unit arises because the basic operations which a digital computer is required to perform, in response to orders written by the programmer, are more complex than the basic operations which take place inside the machine itself. A machine (it is convenient to think in terms of a parallel machine) contains a set of arithmetic registers together with an adder, and also a set of control registers (a register which holds the address of the next order to be executed, index registers, etc.) which also have an adder associated with them. It is the sequencing unit which brings the machine to life by supplying the pulses which cause the suboperations required in the execution of an order to take place. These consist simply of transfers of numbers from one register to another, either direct or via an adder. An addition, for example, requires two such suboperations. The first is the simple transfer of the number in the accumulator to a shifting register and the second is its transfer back to the accumulator via the adder. Similar suboperations take place on the control side of the machine when, for example, the number in an index register is being added to an address.

There is an analogy between the way in which the sequencing unit causes the various suboperations comprising an order to be performed and the way in which the machine as a whole executes the program of orders. This analogy becomes clearer when it is pointed out that some of the suboperations may be conditional; for example, in a multiplication one must add and shift if the multiplier digit under examination is a 1, and shift without adding if it is a 0. It is this analogy which gives rise to the term "microprogramming." The microprogram is the list of suboperations or micro-operations re-

quired to execute all the orders in the order code.

It will be seen that the sequencing unit must contain storage for the microprogram. In a machine which has been designed on an *ad hoc* basis, this storage will be provided by the wiring of the control unit. In the scheme which was suggested by the author in 1951, it was provided in a diode matrix, or rather in two diode matrices. The first matrix determines the transfers which take place during each micro-operation and the second matrix determines which micro-operation in the microprogram is to be executed next. The analogy is thus with a 1+1 address order code such as is often used in a machine provided with a magnetic drum as its main store. If the choice of the next micro-operation can be made conditional, the analogy with ordinary programming is very close; there is, however, no reason why the choice of the transfers which take place during the micro-operation should not also be made conditional.

It should be noted that conditional micro-operations are used not only in arithmetic contexts such as that mentioned previously in connection with a multiplication order, but also for switching from one part of the microprogram to another; for example, in a machine with both fixed and floating point orders, the same basic division sequence may be used for both modes, switching to and from it being arranged by means of conditional micro-operations.

In a machine designed along the principles described, the engineering problems to be tackled in the design of the sequencing unit do not depend on the details of the order code which are determined by the disposition of diodes in the matrices. Two advantages follow from this. One is that the order code need not be decided on until a late stage in the construction of the machine, and will therefore be more up-to-date than it might otherwise be. The other advantage is that people with programming experience can be called in to help with the design. Apart from division of labor, this has the advantage that the resulting order code is more likely to be free from those little twists and exceptions which may seem unimportant to an engineer but which are very irritating to a programmer.

Now in operation in Cambridge is a machine with a sequencing unit designed along these lines, but making use of a matrix of ferrite cores instead of diode matrices.⁷ One matrix only is used and each core corresponds to a micro-operation in the microprogram. One set of

wires determines the order in which the various cores are switched and corresponds to the second diode matrix previously mentioned. The other set of wires control gates throughout the machine and determine what transfers take place during each micro-operation. Conditional action is secured by arranging that at certain intersections there are two cores. At any given time one of the cores is biased off and the matrix behaves as though the other core only were present. Which core is biased off is determined by the value of the binary digit being sensed. Some of the intersections have four cores and provide four-way conditional action.

The control matrix of the machine, which is known as EDSAC 2, contains 1,024 cores of diameter of 8 millimeters, and arranged geometrically at the intersections of a 32×32 grid. The wiring, is such that at some electrical intersections there are single cores, at others two or four cores, and at some no cores at all. By this device it was possible to construct the matrix in two dimensions rather than in three, and to secure uniform loading of the tubes driving it.

The author has written mainly about his own work in this subject and that of his colleagues. He is well aware that others have been thinking along similar lines, and that various machines have been built with sequencing units designed along principles similar to those indicated. A reaction of many engineers when confronted with these ideas is to suggest that a sequencing unit should be composed of a series of delay lines, one for each order in the order code. When a particular order is to be executed, a pulse starts along the delay line corresponding to that order, and branches off to operate gates in the machine at appropriate points on its passage. Sequencing units of this type have been built, the first of which the author became aware being in the Memory Test Computer at the Lincoln Laboratory of the Massachusetts Institute of Technology. In its simplest form this scheme requires that entirely separate sequencing equipment should be provided for each order, even though there may be some overlap of function. Conditionally controlled cross-connections can be provided between the various chains in order to obviate this objection. It may be found, however, that the complexity of the cross-connections is such that it is better to go back to the idea of using a matrix. This is certainly true if magnetic cores are used; many engineers would now agree that magnetic cores pay off as switching elements only if they are used in matrices.

A distinctive approach to the rational design of the sequencing unit of a digital computer is that cultivated on the West Coast. This makes use of Boolean techniques, and instead of the design engineer calling in some one with programming experience to help with the design of the sequencing unit, he calls in a Boolean analyst. The relationship between this approach to the subject and the one described earlier will perhaps become clearer as time goes on.

So far the discussion has been about ways of designing the sequencing unit of an otherwise conventional machine. It is an advantage of these methods that changes may be made in the order code of the machine at a late stage in its design, or even after it has been built. It is not intended, however, that such changes should in general be made after the machine has been accepted for service. Nevertheless, many people have become fascinated by the idea of a machine in which the order code would not be fixed, but could be chosen at will by the programmer. In such a machine the microprogram would not be wired permanently

into a matrix but would be held in an erasable store. The programmer would set up the order code by transferring an appropriate number of words from the main store of the machine to the erasable microprogram store. Each programmer would thus be at liberty to use his own private order code, and indeed could change codes as often as he liked in the course of the same problem.

This idea is certainly a fascinating one, and it is what is meant by many people when they use the term "microprogramming."

However, the view is expressed in the lecture the author gave at Manchester in 1951 that probably there was no real requirement for such a machine, and that it can be said that events have confirmed this view. There has been no lack of engineers who would have been glad to try their hands at designing such a machine, or of programmers who would have liked to have one to play with. As far as the author knows, however, none of them have succeeded in persuading a sponsor to put up the necessary money. Perhaps this is just as well because the many

problems involved in running a computing laboratory are bad enough as it is, without the additional license which would be created by a system of private order codes.

References

1. MIKROPROGRAMM-STEUERWERK, H. Billing, W. Hopmann. *Elektronische Rundschau*, Berlin-Borsigwalde, Germany, vol. 9, 1955, pp. 349-53.
2. A NOTE ON MICROPROGRAMMING, H. T. Glantz. *Journal of the Association for Computing Machinery*, New York, N. Y., vol. 3, Apr. 1956, pp. 78-84.
3. MICRO-PROGRAMMING, R. J. Mercer. *Ibid.*, vol. 4, Apr. 1957, pp. 157-71.
4. LOGICALLY MICRO-PROGRAMMED COMPUTERS, J. V. Blankenbaker. *Transactions, Professional Group on Electronic Computers, Institute of Radio Engineers*, New York, N. Y., vol. EC-7, June 1958, pp. 103-09.
5. THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE, M. V. Wilkes. *Manchester University Computer Inaugural Conference*, Manchester, England, July 1951, pp. 16-18.
6. MICRO-PROGRAMMING AND THE DESIGN OF THE CONTROL CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER, M. V. Wilkes, J. B. Stringer. *Proceedings, Cambridge Philosophical Society*, Cambridge, England, vol. 49, pt. 2, Apr. 1953, pp. 230-38.
7. THE DESIGN OF THE CONTROL UNIT OF AN ELECTRONIC DIGITAL COMPUTER, M. V. Wilkes, W. Renwick, D. J. Wheeler. *Proceedings of the Institution of Electrical Engineers*, London, England, vol. 105, pt. B, Mar. 1958, pp. 121-28.

The Athena Computer, A Reliability Report

L. W. REID G. A. RAYMOND

THIS PRESENTATION provides actual performance data, which makes feasible digital computer applications that require equipment to operate 24 hours a day for a period of 7 days without error or failure, with no preventive maintenance required, and involve real time problems in which a human life is involved.

These data have been derived from the performance figures of Remington Rand Univac's Athena Guidance Computer, Fig. 1. This computer is a large-scale general-purpose, transistorized digital computer. It is part of the Radio Inertial Guidance System for the Air Force ICBM Titan. This guidance system has been developed in co-operation with Bell Telephone Laboratories, Whippany, N.J. Bell Telephone Laboratories has designed the associated radar equipment.

The guidance system consists of a ground based computer, which com-

municates with the missile through the radar. Simply stated, the system continually determines the missile's present position and relates it to a desired position, computes and furnishes to the missile necessary steering orders to perform corrective action so that the assigned destination is reached.

Operational Summary

In the contract issued by the United States Air Force to Remington Rand Univac, the Air Force placed specific requirements on the manufacturer as to the desired level of reliability which was to be attained. This requirement is most easily explained by saying that no more than six out of every 1,000 missile firings shall fail to hit the target due to errors or failures on the part of the computer. This converts to a mean time to failure requirement of 55.4 hours.

All information presented is based on the following definition of a failure: A failure is any computer malfunction which affects guidance. This definition includes any and all computer errors or malfunctions, which occur at any time during the operation of the computer, whether it be in performing actual guidance or in test or maintenance routines.

The sections of the computer, included in the failure reports, are those sections active during guidance or used to confidence check the equipment prior to guidance. The failures are reported against a component population in each computer system of approximately 7,000 transistors, 21,000 diodes, and 24,000 resistors. Equipment concerned with monitoring guidance operation such as magnetic tape, typewriter, and tape punch is excluded.

The information presented is based on operational experience obtained on three Athena Guidance Computers, one of which is installed at the Air Force Missile Test Center, Cape Canaveral, Fla. This system is fully integrated with the Bell Telephone Laboratories' radar system and is operated on sched-

L. W. REID and G. A. RAYMOND are with Remington Rand Univac, St. Paul, Minn.