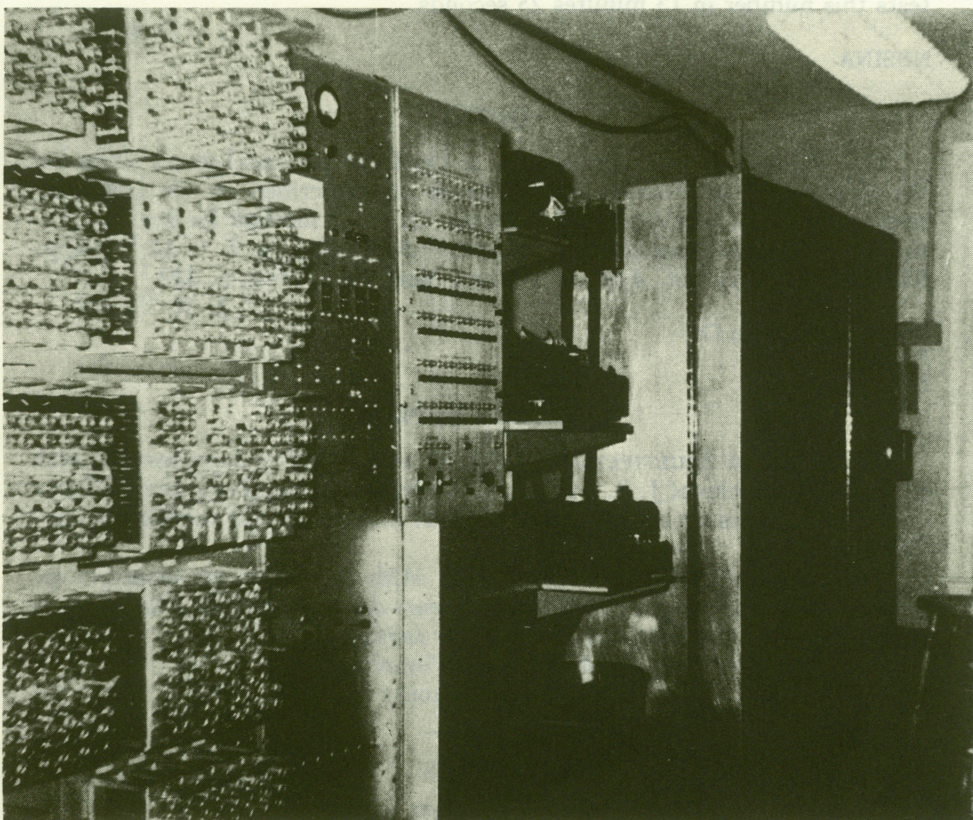


prepared to act as a clearing house, supplying information about the most urgent tasks that call for key-punching.

G. BRANCH
E. G. YOWEN

NSINA

138. A New Message Form.—The program described in Note 131 (c) has produced the 15th Message Form: 200-1 on June 25. The SWAC has the number of 15 minutes.



THE ONR RELAY COMPUTER

AN XIII (1965)

The first page bears the following annotation written in pencil: "Revised by Ste Anne 71, the second time 131 by M. Oyon. I have no information about Oyon (he may have been an engineer in the Land Registry Service)."

R. LÉVARD

Tulle (Corrèze) France

The Office of Naval Research Relay Computer

1. Introduction. This paper describes a general purpose relay computer. This digital machine was installed in Staughton Hall on the campus of The George Washington University on May 9, 1951. At that time it was renamed the "ONR Relay Computer" and was made available by the Office of Naval Research to the Logistics Research Project for operation and maintenance under an ONR-GWU contract. The machine is a slow-speed, single-address computer with magnetic drum storage of 4094 numbers or instructions, each consisting of 24 binary digits including sign (equivalent to about seven decimal digits). It is capable of performing 42 arithmetic, logical, transfer, output and stop instructions through use of 734 mechanical relays and 655 electronic tubes. Input to the computer is on mechanically sensed, seven-level teletype paper tape, while output may be on electric typewriter, or on punched tape. A view of the computer is shown in the frontispiece.

2. Storage. The storage element is a cylinder or drum 12 inches in diameter, $8\frac{1}{2}$ inches long. The cylindrical surface of this drum is coated with a ferromagnetic material, small areas of which may be magnetized in either of two directions to represent the binary digits 0 or 1. The drum rotates at 440 revolutions per minute, and the magnetization configurations can be played back from or recorded (stored) on it 24 at a time by a group of 24 magnetic heads placed .002 inches from the surface. Two additional playback heads are required for timing. All electronic tubes in the computer are used in connection with the storage unit; they accomplish amplification, shaping and timing of signals in the memory unit. These groups of 24 binary digits represent both numbers and coded instructions to the computer, either of which may be stored on any part of the drum. Successive memory locations are normally loaded from punched paper tape sensed by a standard mechanical tape reader, but any can be loaded singly by manual controls. Each group of 24 binary digits has an address of 12 binary digits which refers to and specifies its particular location on the drum. There are 4094 such addresses, from octal 0002 through 7777. Given any address, the operator can inspect by manual control the contents of the address in a display of neon lights.

3. Arithmetic Unit. All arithmetic is performed by mechanical relays within the computer. A relay may be either open (nonconducting) or closed (conducting), representing the binary digits 0 and 1, respectively. The computer contains three basic arithmetic registers which are groups of relays: the Accumulator, the Q register and the X register. The Accumulator, A, is capable of binary representation of a number of 48 digits. It is the principal arithmetic register, the place where the actual arithmetic is performed, and possesses subtracting and shifting properties. The 24-binary-digit Q register holds the multiplier during multiplication and the quotient after division. The 24-binary-digit X register serves as a temporary storage for both numbers and commands as they come from the memory. In it are

held instructions during decoding and data while being operated upon. For example, the X register contains the multiplicand during multiplication.

The binary number system is used throughout. On the control panel lights and buttons are grouped in triplets for easier octal representation. All numbers are considered as integers by the computer, with the left-most digit indicating the sign (0 designates positive; 1, negative). A negative number is represented by the "one's" complement of the corresponding positive number. The Accumulator is basically a subtractor, so additions are actually performed by complementing one of the numbers to be added and subtracting it with end-around borrow from the other number. Such subtractions are performed in parallel; i.e., all 24 digits are operated on at once. Multiplication is a series of subtractions and shift operations.

Conceptually, the arithmetic unit is much like a desk calculator where the Accumulator corresponds to the product or upper dials, with twice as many digits as the keyboard; the Q register corresponds to the lower or multiplier dials, equal in length to the keyboard; and the X register is similar to the keyboard itself. But the Relay Computer's "keyboard" X can effectively be filled from any of the 4094 addresses under control of instructions on the drum. Basically the arithmetic unit is like an automatically sequenced desk calculator with 4094 keyboards.

4. Instructions. Each instruction occupies one address. Such an instruction always contains an operation code for the left-hand six binary digits. These two octal digits designate which of the 42 instructions the computer is to perform. The right-hand four octal digits usually indicate the address from which data are to be taken for use with the instruction being performed. Some instructions do not need an associated address; they use the allotted digits for other purposes. In the shift instruction, for example, these digits indicate the number of digit-positions to be shifted. In any case the third and fourth octal digits from the left are not interpreted by the computer in the execution of an instruction.

While the computer is in operation, the arithmetic section operates asynchronously with the memory section, each waiting in turn for the other to perform its function. In normal operation, when the address of the first instruction has been sent to the memory unit (to be held in the Drum Address Register, DAR), further action is suspended until that address has passed under the playback heads and its contents have been transferred to the X register. While the 6-binary-digit operation code is being decoded, the 12-binary-digit address, if contained in the instruction, is transferred back to the Drum Address Register. No arithmetic operation can be performed until the specified address has been located on the drum and its contents have been transferred to the X register. Actually an address is found electronically by circumferentially counting timing channel pulses and matching the count with the contents of the DAR. While the instruction is being executed, action in the memory unit is suspended. Then the address from which the next order is to be taken is transferred to the DAR. In this way the computer normally proceeds in linear sequence from one instruction to the next. The Instruction Address Register, IAR, holds the address of the next instruction in linear sequence to be sent to the DAR before the execution of each instruction. In normal sequence "one" is automatically added to the contents of the IAR as each instruction is carried out.

Four "jump" instructions provide important exceptions to the linear sequence described. Each one of these can specify the address from which the computer is to take its next instruction under a certain condition. That is, the four right-hand octal digits of these instructions themselves are transferred to the DAR under certain conditions; this transfer interrupts the normal transfer of the contents of the IAR to the DAR. These four octal digits are also transferred to the IAR, so that operation again proceeds in a linear sequence starting with the instruction at this new address. This ability to alter the linear sequence of instructions, together with the ability to alter instructions by performing simple arithmetic with them, is the basis for the real flexibility of this general purpose computer.

Following is a brief description of the instructions:

1) Fourteen insert instructions: These cause the transfer of 24 binary digits replacing the previous contents of a receiving location. Included are transfers between the drum, Accumulator and Q register. Besides these normal transfers, provision is made for the transfer of absolute values, complements and specified digits, and for the storage of the right-hand 15 digits (address part) from A.

2) Eleven arithmetic operations: They include special additions and subtractions to facilitate double precision (48-binary-digit) arithmetic and manipulations with absolute values, and summation of the contents of successive storage locations. In division the non-negative remainder is retained in A. Both clear multiply and accumulate multiply produce 48-binary-digit products in the Accumulator.

3) Four jump instructions: One is unconditional. Two others depend on the sign of A or Q (control is transferred if the register contains a negative number). The fourth is a zero test for the Accumulator.

4) Two shifting operations: These instructions provide for the circular left shifting of the contents of A or Q.

5) Three stops: optional, intermediate (which depend on pre-set switches), and final.

6) Three output instructions: One causes the typewriter to print a single character or perform a single function (carriage return, space, etc.). Another causes the typewriter to perform as above and also causes the punching of the corresponding teletype code on tape. The third causes the contents of successive storage locations to be punched in a form suitable for input or later printing.

7) Five logical operations: These include two digit-by-digit multiplication instructions, noncarry addition, clear half of A and a special "transfer and add one" instruction.

Division and multiplication (depending on the number of one's in the multiplier) require approximately $2\frac{1}{2}$ seconds. All other operations which refer to storage can be performed at the rate of 220 per minute, i.e., one every two drum revolutions or .272 seconds per single address operation. Those which do not require reference to the memory (shifts, jumps, transfers between A and Q) are executed with twice the speed. An exception to these rules is the zero-jump test of the Accumulator. This instruction, though reference to the memory is not necessary, requires the time of 2

drum revolutions since "one" is automatically subtracted from the contents of A after the zero test.

5. Input, Output, and Operation. Normal perforated input tapes contain seven channels across the tape, six of which are used for binary information. Four lines, therefore, contain 24 binary digits of information—a complete number or instruction. Each line can be considered to have one octal digit on each side of the tape feed hole. The seventh channel normally contains a punch in every fourth line of the tape to signal the end of a 24-digit group. For normal input the operator inserts the tape in the reader and indicates the address into which the first group is to be stored by pressing digit keys of the Drum Address Register. When reading has started, successive groups on tape are automatically stored at successive addresses as long as holes continue to appear in the seventh channel in every fourth line. This normal tape input takes about $\frac{2}{3}$ second for each 24 binary digits, i.e., 90 addresses are loaded per minute. Another type of drum loading from tape is possible. In this case each line (6 binary digits) of tape is loaded into the least significant six places of consecutive storage locations. This type of drum loading proceeds at 220 storage locations per minute.

Under manual control, not only can the operator inspect in lights the contents of any desired memory location, but he can cause insertion of 24 binary digits there. The contents of A, Q and X are always displayed in lights even during computation. Under manual control of the operator, as well as when program-controlled, the machine is capable of automatically punching the contents of successive storage locations onto tape in a form directly suitable for input when the first address is specified by the operator. Output of this kind is at the rate of 110 addresses per minute. Also, when operator- or program-controlled, the machine can sum the contents of consecutive storage locations. This operation, used for checking the storage of a complete program or a set of data whose sum is known, proceeds at the rate of 440 addresses per minute, leaving the cumulative sum in the Accumulator.

Actually the operator, when performing the manipulations discussed above, and the computer itself, while under control of the program, make use of three additional 24-binary-digit registers. The Storage Insertion Register contains a number to be stored in the memory unit; the Storage Output Register holds the number when it comes from the drum; and the Storage Blocking Register holds binary "1's" in digit positions where storage of a number is to be blocked. All three may be loaded manually by pushing buttons for binary 1's.

After both data and instructions are in the memory, the calculation may be started (there is no program instruction by which the computer can call for additional data to be stored on the drum during computation). When starting computation, the operator by means of a switch may cause the machine to perform a single instruction or to execute instructions successively from one to the next in normal operation. The computer stops automatically and indicates the trouble if the operation code it receives is a nonpermissible one, if it reaches a meaningless print code, or if in division the quotient is too large for the Q register.

6. Maintenance and Use. Ideally, after a 15 to 30 minute warm-up

period and a successful run of the test program each morning, the computer would be available for operation during the remainder of the day, since no routine maintenance periods are scheduled. The warm-up period may be prolonged or the operation interrupted for needed nonscheduled maintenance. Since no marginal checking features, computational checks or indications as to the faulty section are provided internally, much time is sometimes consumed in finding and remedying trouble. Ordinarily programs are written to include computational checks or are written so that a complete duplication of each computation section is performed before proceeding to the next.

The Logistics Research Project usually operates the computer $8\frac{1}{2}$ hours a day during a five day week. For the first eleven months after delivery in May '51, the computer was maintained by several of The George Washington University undergraduate student engineers on duty a total of about 70 hours per week. This indicates that computer time was devoted to education of new maintenance men. Since then, one full-time engineer and one half-time undergraduate student have been maintaining the computer. During the first year of operation at the University, about \$400 was spent for replacement parts and approximately \$100 for parts for computer modification. This includes the cost of replacement of 21 electronic tubes and 50 relays. The maintenance men have undertaken projects other than routine maintenance. As an example, a test rack has been assembled which allows checking of any of the computer's 48 removable electronic chassis apart from the computer. During this first year of operation, approximately 42.5% of the time during which the machine was on was spent in maintenance, original checkout and education of maintenance men, and 2% for improvement or modification of the computer itself. Thus 55.5% of the time was available to the operator-programmers for the year. This last figure includes time spent in input, output, program checking and actual computation (which sometimes turns into time spent pinning down the cause of a machine error). For the first six months of operation the operator-programmer time was 43.4% of the total time; this increased to 68.3% for the other six months of the year. The computer, then, is purposely a research and educational tool in maintenance as well as operation.

Several problems from organizations outside the Logistics Research Project have been allotted machine time. These problems have been discussed in advance to determine eligibility for use of the machine with the Logistics Branch of the Office of Naval Research, Department of the Navy, or with the staff of the Logistics Research Project.¹ Tape preparation, duplication and print-out accessories are available to those whose problems have been accepted.

Following is a selected list of programs which were run during the first year, together with pertinent comments:

- 1) The Project used the machine to compute quarterly manpower requirements implied by a proposed four-year shipbuilding schedule. Given the building dates and total manpower requirement of a ship, a fourth-degree polynomial was evaluated to distribute the man-hour requirements into yearly quarters. These requirements were summed for the fleet.
- 2) A program was prepared for the iterative solution of games by

Brown's method—each player having 5 strategies. The method provides convergence to the value of the game within .01 after about 8 hours of computing (1000 iterations) for the games computed.

3) Solution of n simultaneous linear equations by Crout's method was performed for $n = 5$, and a general program was prepared for $n \leq 10$.

4) The Project used the computer to assist in obtaining a research solution to a fleet logistics planning problem.

5) Currently, a calculation is being performed which obtains information pertaining to feasible schedules for tanker delivery of fuel. Later this information will be used to obtain an optimal schedule with the hope of application to monthly delivery of petroleum products by the Military Sea Transport Service.

J. JAY WOLF

The George Washington University
Washington, D. C.

This work was sponsored by the Office of Naval Research.

¹ Publications available from the Logistics Research Project are: JOS. B. KRUSKAL, JR., *A Programmer's Description of ABEL, A Magnetic Drum Relay Computer, Orders for the ONR Relay Computer*, a detailed description of the instructions, and *ONR Relay Computer Operators Manual*.

On the Rounding Off of Difference Tables for Linear Interpolation

In order to simplify linear interpolation many tables contain the differences $\Delta(x) = f(x + d) - f(x)$ between two consecutive values of the function. Since the values of $f(x)$ given in the tables are usually rounded off the question arises whether the value of $\Delta(x)$ given in the table must be the difference between the rounded off values $\overline{f(x)}$ of $f(x)$ and $\overline{f(x + d)}$ of $f(x + d)$ or $\bar{\Delta}$, the result of the rounding off applied to $\Delta(x)$. It appears on the first view plausible that we obtain better results in the second case since we use here more information about $f(x)$. However, the detailed analysis shows that this is not so. If the values $f(x)$ are given with n decimals so that the rounding off errors do not exceed $h = \frac{1}{2} 10^{-n}$, the part of the interpolation error due to the rounding off of $f(x)$ and $f(x + d)$ does not exceed h , if the difference used is $\overline{f(x + d)} - \overline{f(x)}$, while if we use $\bar{\Delta}$, this error can come arbitrarily near to $2h$.

Since this situation is not apparently realized by all computers of tables, I should like to develop an observation on this subject which was published elsewhere.¹

We give first an example.

In computing the decimal logarithm $\log 9684.8$ we start from the values $\log 9684 = 3.986\ 054\ 78$; $\log 9685 = 3.986\ 099\ 63$ and from the rounded off values $\log 9684 = 3.986\ 05$; $\log 9685 = 3.986\ 10$ with an error $< h = \frac{1}{2} 10^{-5}$.

We have

$$\Delta = 4.485 \cdot 10^{-5}, \quad \bar{\Delta} = 4 \cdot 10^{-5}.$$

We obtain then by the "complete" interpolation

$$\log 9684.8 = 3.986\ 054\ 78 + 0.8 \cdot 4.485 \cdot 10^{-5} = 3.986\ 090\ 66,$$