The Origins, Uses, and Fate of the EDVAC

MICHAEL R. WILLIAMS

The EDVAC computer was the first modern, electronic stored-program computer to be designed. *It was, however, never produced to the original plan. When eventually redesigned and constructed, it was unreliable and heavily modified. This article sets out the basic facts about the machine, the uses to which it was put, the software used in an attempt to ensure its reliable operation, and its eventual fate.*

It is well known that the EDVAC was the first general-purpose electronic digital stored-program computer to be *designed.* This fact is clearly repeated in almost every elementary textbook and, occasionally, even a one-line description of the machine can be found. What is not so easy to find, however, is any indication that the EDVAC was nowhere near the first computer to be operational, was not actually constructed according to the initial design, was not reliable when constructed, and was eventually so heavily modified that it would have been almost unrecognizable to the original design team. This article will set forth the basic facts about EDVAC in the hope that it will serve as a reference point for future investigation into the influence of this machine.

EDVAC origins

In 1944, at the Moore School of Electrical Engineering at the University of Pennsylvania, the world's first *large-scale* electronic calculating machine was under construction. The ENIAC was a leap forward in calculating technology because its basic building block, the vacuum tube, was thousands of times faster than the electromechanical devices that had been used in earlier machines. It is certainly the case that several other projects (Atanasoff in America and Zuse/Schreyer in Germany, for example) had experimented with the use of vacuum tubes for constructing arithmetic units, but the ENIAC was the first machine to incorporate these high-speed devices into fully functional control and memory elements as well as the arithmetic facilities.

Any of the earlier machines, such as the Zuse, Harvard, or Bell Laboratories devices, that were capable of automatically executing a series of instructions, did so by reading one instruction at a time from long loops of paper tape, executing it, then reading the next instruction. This technique was well known to the major ENIAC designers, J. Presper Eckert and John Mauchly, but had to be replaced by a different control mechanism for the ENIAC. It was a waste of resources to use a mechanism that was only capable of reading, at most, a few instructions per second from a paper tape to control an arithmetic machine capable of executing 5,000 operations per second.

The ENIAC, of course, took its "instructions" in the form of electrical activation signals that were routed from unit to unit by an extensive network of plug-board panels and bus wires that ran around the lower exterior portion of the machine (see Figure 1). The ENIAC progress report of December 31, 1943, parts of which are reprinted in *From ENIAC to UNIVAC*,[1] makes it quite clear that no attempt had been made to "make provision for setting up a program automatically" because this would have added extra complications to the design. Some concept of what was implied by "setting up a program automatically" can be obtained from reading a report, written in January 1944 (reprinted, in part, in *From ENIAC to UNIVAC*,[1] p. 28), on the design of a magnetic calculating machine that was to have its instructions recorded either magnetically on a special alloy disk or etched permanently on the disk when the program was to be kept for repeated use. Thus, the concept of the stored-program computer, while not completely developed, appears to have been in the initial stages of formation some time in late 1943 or early 1944.

By early 1944 the pressure of design and early supervision work on the ENIAC project had eased off to the point where Eckert and Mauchly could safely leave most of the construction details to other technical staff. They were able to spend some time considering just how they would have gone about the ENIAC design, if they were ever to attempt such a project again, without the urgency imposed by military wartime conditions, something that they had been hoping to do for at least the previous six or eight months.[2] In fact, the group had asked their military backers, through their army liaison officer Herman Goldstine, for some extra money to finance just such a reevaluation project in August 1943, but it was not granted at that time.

Although written two years later, some indication of the progress made on the concept of a new machine can be gleaned from the first major report written on the project.

Short preliminary versions of this article have been presented at the New Advances in Computer Science conference held in Graz, Austria, 1991, and at the History of Mathematics Conference held in Baotao, Inner Mongolia, 1991.

1058-6180/93/0100-0022$03.00 © 1993 IEEE

Figure 1. A close-up view of the "programming" system for the ENIAC. (Photograph courtesy of the US Army Ballistic Research Laboratories.)

During the construction of the ENIAC, Eckert and Mauchly reported,

> It became apparent that serial operation was in general advantageous and that when serial methods were used whenever possible the equipment was used most efficiently. Hence, in January, 1944, a "magnetic calculating machine" was disclosed, wherein the successive digits of a number were transmitted in timed sequence from magnetic storage or memory devices through electronic switches to a central electronic computing circuit and similarly returned to magnetic storage. An important feature of this device was the operating instructions and function tables would be stored in exactly the same memory device as that used for numbers... Therefore in July, 1944, it was agreed that when work on the ENIAC permitted, the development and construction of such a machine should be undertaken.[2]

When John von Neumann learned of the ENIAC project in the summer of 1944, he became a regular visitor to the Moore School and, although too late to participate in the design considerations of the ENIAC, he eagerly joined in the discussions concerning a new machine, eventually to be called EDVAC (Electronic Discrete VAriable Computer). This machine was to be capable of storing its instruction "tape" internally in a memory system and issuing instructions, one at a time, at electronic speeds comparable with those available in the rest of the machine. In October 1944 the Army Ordnance Department granted a second Moore School request for additional funds to explore this new concept. According to Stern[1] (p. 59), this $105,600 addition to the ENIAC budget may well have been given because of the influence and prestige that John von Neumann now added to the project.

From September 1944 on, von Neumann took an active part in the EDVAC design discussions, even writing letters to the Moore School group when his other duties required him to be absent. It is undoubtedly the case that all members of the group made contributions to the eventual design of EDVAC, but it was von Neumann's genius for organizing material and his penchant for producing written reports that eventually led him to write down the results of these design meetings in a document which he called "First Draft of a Report on the EDVAC" in June 1945. It was this, copies of which were circulated to a wider audience than von Neumann had intended, which first described, in any reasonable detail, the concept of the stored-program digital computer.

As the name implies, this document was intended as a first draft of a report on the investigations taking place under the Army contract. The fact that von Neumann is listed as

the only author has led to his name becoming closely associated with the concept of the modern stored-program computer. Needless to say, several other members of the Moore School staff were annoyed to find little or no mention of their own contributions and this, combined with later patent right disputes, led to several confrontations. These altercations resulted in the leading members of the EDVAC design team, together with several of the best technical personnel, leaving the Moore School and joining other academic institutions, founding their own electronic or computer-related firms, or simply moving on to other projects. This, in turn, caused an almost complete halt to any further design or development work on the EDVAC.

EDVAC of the Moore School lectures

The next major event in the EDVAC story concerns the activities at the Moore School that took place immediately after the end of the Second World War. The concept of using the electronic vacuum tube as a basis upon which to construct a calculating machine became widely known through various newspaper stories, technical articles, and cinema newsreel features about the ENIAC. This led to the Moore School receiving a number of inquiries from industrial firms, academic institutes, and government agencies concerning the technical details of the ENIAC construction. To accommodate these numerous requests for information, the Moore School decided to present a course during the summer of 1946. Although a number of the senior members of the ENIAC and EDVAC teams had already left, people like Eckert, Mauchly, Goldstine, von Neumann, and others returned to give the majority of lectures in the course.

This two-month course, "Theory and Techniques for the Design of Electronic Digital Computers,"[3] was the turning point in the spread of information about the electronic digital computer. It officially attracted 28 people from both sides of the Atlantic, but there were many others who were known to have attended one or more of the lectures. Most of the "students" were under the impression that the main subject would be the ENIAC developments and were surprised when the hardware lecturers spent their time discussing the new design for EDVAC. About two thirds of the way through the course, one of the students, Sam Alexander (who was later a computer pioneer in his own right), obtained the support of a number of his fellow attendees and demanded that the course lecturers stop all this talk about a hypothetical EDVAC machine and get back to describing the construction and operation of ENIAC.[3] Despite this insistence on ignoring the stored-program concept, enough information was disseminated about EDVAC that this design became the basis for several machines, the most famous being the Cambridge University EDSAC, which were to be constructed immediately after the course was complete.

It is worth examining the EDVAC concept as it existed in the summer of 1946 because it was, through its presentation at the Moore School lectures and technical reports, highly influential in the design of many of the early British and American machines.

The EDVAC was the major subject in a number of different lectures during the course. On July 15, for example, Eckert and Brad Sheppard gave two talks that looked first at the ENIAC and its control system, then at how a similar machine control could be implemented by storing information in a high-speed memory on an EDVAC type of machine. This was likely the first time that most of the attendees had ever heard of the concept of a stored-program computer. As the course progressed, the different parts of an EDVAC-like machine were discussed in detail. Often alternative implementations for each component were presented and even radically different designs, such as the trade-offs to be expected between binary- and decimal-oriented machines, were discussed. Finally, on August 28 and 29 (1946), Kite Sharpless presented the current Moore School plans for the complete EDVAC in a section of the course entitled "Description of Serial Acoustic Binary EDVAC."[3]

Figure 2 is a simplified version of the original blueprint shown during the lectures as the most up-to-date version of the machine. The large open square labeled "computer" was actually to contain the arithmetic unit, while the blank "reader and recorder" section was for the input and output equipment. There were plans for each of these sections; in fact, Sharpless went on to discuss a possible design for "the serial binary computer" later in the session, but they were only preliminary sketches and still subject to extensive revision.

In Figure 2, the boxes at the top represent the memory, which was intended to be constructed from mercury delay lines. The horizontal lines (S1, S2, and S3) under the memory units are switches through which the various computer components are connected to the control (CPU) mechanism shown in some detail at the lower left. The memory consisted of two distinct types: The ones at the upper left represent long delay tanks, capable of holding 1,024 binary bits with 1-μs spacing (equivalent to 32 words of 32 bits each), while those at the right are shorter versions of the same system but are each limited to holding one word of 32 bits. The dotted lines between the memory components in Figure 2 represent a bus transmitting words between the long and short tanks — this was to be used to break up the 32 words stored in a single long tank and distribute them into the 32 individual short tank units.

The operation of the machine in executing a single instruction would proceed as follows:

1. An instruction would be selected from the memory units, via a mechanism not shown here, and proceed down the dotted bus, through the gates shown immediately below the memory units, and into the distribution network to store individual bits in flip-flops labeled KF1-KF36. (The exact distribution can be found in the Charles Babbage Institute reprint of the lecture notes[3] on pp. 553-554.)
2. The 5-bit operation code, now residing in K24-K28, would control the action of the large matrix switch immediately below these flip-flops, and this would open the appropriate switches to allow the operands to flow into the "computer" and also activate the *s*, *a*, and *m* lines (subtract, add, or multiply lines shown entering the left-hand side of the computer box) controlling the action of the "computer."

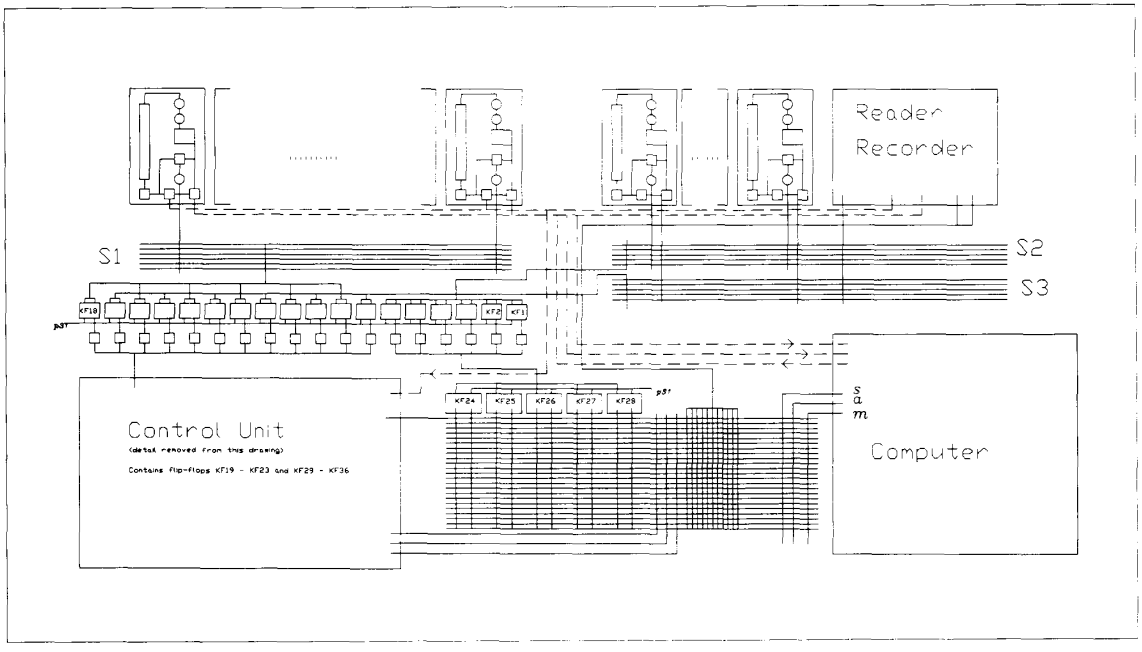


Figure 2. The August 1946 design for a "Serial Acoustic Binary EDVAC," as it was shown to students during the Moore School lectures (redrawn from the original with some detail removed to avoid clutter).

3. The bits in KF1-KF6 would control the switch S2, which allows one-word operands from the short tanks and the reader/recorder unit to flow to the "computer" via the dotted buses.
4. The bits in KF7-KF18 control switches S1 and S3 to allow data movement involving the long tanks.

This scheme would allow an instruction to consist of a 5-bit operation code, three 6-bit addresses, and other control bits. The proposed instructions are listed below; *a*, *b*, *g* are memory addresses, while a, b, g are actual numbers, that is, immediate operands:

- add *a b g*: Add the contents of *a* to the contents of *b* and put the result in *g*.
- sub *a b g*: Subtract the contents of *a* from the contents of *b* and put the difference in *g*.
- mul *a b g*: Multiply the contents of *a* by the contents of *b* and store the product in *g*.
- neg *a b g*: Multiply the contents of *a* by the contents of *b* and store the negative of the product in *g*.
- c *a b g*: If the number stored in *a* is greater than the number stored in *b*, then take the next instruction from memory location *g*.
- x a b *g*: If a is greater than b, then take the next instruction from memory location *g*.
- t - - *g*: Transfer control to the instruction in memory location *g*.
- p *a b* g: Shift the number stored in *a* by g places to the left and store the result in location *b*.
- q *a b* g: Shift the number stored in *a* by g places to the right and store the result in location *b*.
- i a b g: Increase *a*, *b*, and *g* in the memory location following this one by the values of a, b, and g from this instruction.
- e *a b* g: Extract some of the bits in the contents of *a* and use these to replace bits in the same position of the word *b*. The bits in question are determined by the contents of g.
- fn - *b* g: Read g words of information into memory location *b* moving tape n in the forward direction.
- bn - *b* g: Read g words of information into memory location *b* moving tape n in the backward direction.
- fn *a* - g: Write g words of information from memory location *a* moving tape n in the forward direction.
- bn *a* - g: Write g words of information from memory location *a* moving tape n in the backward direction.

This machine is clearly a workable modern stored-program computer capable of performing any of the usual types of jobs that could be done on a machine with its memory capacity. It was, however, a pipe dream. It was never con-

structed in this form and, in fact, no further development work was done on this design after Eckert and Mauchly left the Moore School in the spring of 1946. Its major influence was that it was the first stored-program electronic digital computer to be described at this level of detail and, as such, set the paradigm for many of the first-generation machines. The design, and in particular the concept of using mercury delay lines for the memory, influenced several of the early machines, the Cambridge EDSAC (for which Maurice Wilkes deliberately chose a similar name to show the connection) and the SEAC being the most famous.

Revised EDVAC design

When Eckert, Mauchly, and others left the Moore School early in 1946, the job of heading up the EDVAC project fell to T.K. (Kite) Sharpless who, after graduating with an MS in electrical engineering from the Moore School in 1943, stayed on to become a teacher and member of the ENIAC and EDVAC projects. Sharpless himself left in 1947 to become a founding partner in the firm of Technitrol, Inc., which specialized in the production of electronic components for the developing computer industry (among other items, it produced the memory components for EDVAC and sold a duplicate unit for use in the SEAC). The next manager to be appointed was Louis Tabor, whose tenure lasted for only a few months. Finally, the task of project manager and chief engineer was given to Richard L. Snyder, who saw the project through to the point where a machine was actually shipped from the Moore School to the Army's Ballistic Research Laboratories at the Aberdeen Proving Ground in Maryland. Snyder left the Moore School at this time and followed the machine to the Ballistic Research Laboratories. The final report on the EDVAC was actually written, for the Moore School, by S.E. Gluck and W.H. Boghosian.[4]

Although the situation, particularly the staffing, at the Moore School had changed considerably from what it had been in the late stages of the war, it was still the case that the staff had a contract with the Army Ballistic Research Laboratories (BRL) for research on the EDVAC. It was, however, obvious that some changes in the project were required before any real progress could be expected. As a consequence, Harold Pender (dean of the Moore School) and Irven Travis (director of research for the Moore School) met with Colonel Paul Gillon of BRL and John von Neumann (now back with the Institute for Advanced Study) to discuss the next phase of the EDVAC project. During this meeting, von Neumann (who, of course, had plans to attempt the construction of a machine himself) pointed out that no machine of this type had ever been constructed but several groups had preliminary plans to do so, and thus information about coding problems and the operating characteristics of an EDVAC-like machine was urgently needed. It was therefore decided

> that the Moore School should immediately proceed with the design and construction of a small preliminary model of EDVAC for the B.R.L., while the Institute for Advanced Study should undertake a study program leading to the establishment of a large scale comprehensive computer.[5]

To agree on what constituted "a small preliminary model of EDVAC" a meeting was held at the Ballistic Research Laboratories in Aberdeen, Md., on October 9, 1946. The attendees were Dean Harold Pender and Dr. Irven Travis representing the Moore School, Col. G.F. Powel and Mr. S. Feltman from Army Ordnance, Col. L.E. Simon from the Ballistic Research Laboratories, Dr. J. von Neumann from the Institute for Advanced Study, and Mr. H. Diamond from the National Bureau of Standards. The Moore School suggested three alternatives:[5]

- *EDVAC 1.* A simple serial binary computer with the ability to add, subtract, and multiply (but division had to be programmed), with no internal checking of operations and a memory capacity of 1,000 words.
- *EDVAC 2.* A simple binary coded decimal computer that could perform fixed decimal point arithmetic with all four standard arithmetic operations, complete internal checking of results of each operation, and a memory capacity of about 1,000 words.
- *EDVAC 3.* A machine similar to EDVAC 2 but augmented by a floating-point arithmetic unit and a total memory capacity of about 4,000 words.

Von Neumann suggested that the EDVAC should be a binary machine[6] and thus the meeting decided on EDVAC 1, but with the addition of facilities for a hardware division instruction and the complete checking of all arithmetic operations. This machine was known as the EDVAC 1.5. Another decision made at this meeting, and one that was later to be regretted, was that the machine was to rely on a magnetic wire input/output system with stand-alone equipment to transcribe information to and from the wire. The National Bureau of Standards undertook to develop the magnetic wire reading and recording equipment, which was then to be actually constructed by the Reeves Instrument Corporation.

Some seven months later (May 27, 1947) another meeting was held because it had become apparent that the machine could be produced in two slightly different versions: EDVAC 1.5A or EDVAC 1.5B. The B version would have a slightly extended instruction set that, besides the elementary arithmetic and control instructions, would contain more sophisticated facilities such as floating-point operations and the "extract" (see below) instruction that had been described during the Moore School lectures. Dr. Richard Clippinger of BRL was asked to study the situation. He quickly decided that it should be EDVAC 1.5B that was constructed, but the floating-point facilities were later abandoned because of the extra complexity they introduced into the hardware design. A software floating-point system was eventually produced to overcome the lack of appropriate hardware.

It was decided that the machine should take advantage of the possibility of optimum coding. The mercury delay line memory, while fast, is not completely random access — any

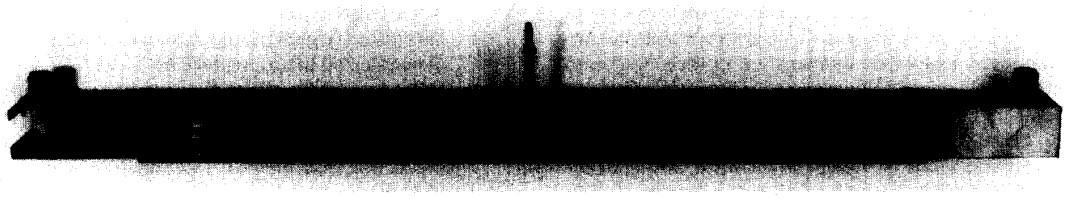


Figure 3. A photograph of a single mercury delay line, in its metal housing. The photograph is actually of the SEAC memory, which was identical to that in the EDVAC. (Photo courtesy of the National Museum of American History, Smithsonian Institution, Washington, D.C.)

particular memory item is only available as it emerges from the delay line. Thus the machine might have to wait up to a millisecond for the next instruction to become available. Rather than have control normally go to the next instruction in sequence in the memory, an additional address field was to be included in each instruction, which gave the location of the next instruction to be executed. This allowed the programmer, if the exact timing of each instruction was known, to code a problem so that as one instruction was finishing its execution the next instruction to be obeyed would be just about to exit from the delay line memory. This meant that each instruction had to be composed of an operation code and four addresses. It was usually denoted by

op a b g d

where *op* was the operation code, *a* the address of the first operand, *b* the address of the second operand, *g* the address of the destination of the result, and *d* the address of the next instruction.

It was decided that the memory should have a capacity of 1,024 words (which automatically meant that *a*, *b*, *g*, and *d* each had to be 10 bits long) and that no more than 16 instructions would be allowed (thus *op* had to be 4 bits long) which, as they wanted one instruction per word of memory, established the word size at 44 bits. The delay lines actually stored 48 bits for each word, the extra four bits being "constant characters" or "non-pulses," which simply provided a time interval between successive words so that the electronic circuits could switch and stabilize properly. A preliminary analysis[5] showed that a 1,024-word memory would be enough to do useful problems, while reducing it to 512 words would only save about 25 percent of the cost and limit the usefulness of the resulting machine. On the other hand, increasing the memory to 2,048 words would have increased the cost of the machine by about 50 percent.

To ensure that the delay in waiting for a particular word to emerge from a memory line was kept as short as possible, the memory was divided into 128 individual delay tanks, each holding eight words. The 10-bit address of each word in memory was to be constructed from a 7-bit integer specifying the tank and a 3-bit integer (often called the minor cycle number) indicating which of the eight words was to be used. An additional consideration in the construction of each of these eight-word tanks was that, because they were quite short, it was possible to allow the temperature to vary by plus or minus 2.5 degrees C. On the other hand, a 16-word delay line required an operating temperature within 1.25 degrees C of optimal conditions to keep the timing within allowable limits.

An engineer on the project, Herman Lukoff, had earlier been given the job of designing a memory temperature compensator that would change the spacing between the acoustic pulses to offset the effects of temperature variations. Although he had some success with this design, even demonstrating it at an IRE meeting in New York in March 1947,[7] the circuit was very sensitive to outside electrical noise. When Lukoff left the Moore School in September of that year to join the fledgling Eckert-Mauchly organization, the decision was made to contract the memory requirements out to the firm of Technitrol, which had recently been founded by Kite Sharpless.

Sharpless decided to change the design slightly to produce a mercury memory system that relied mainly on a thermostatically controlled oven to house the delay lines rather than to completely trust Lukoff's questionable temperature-compensation circuits. The basic reason for this decision was that the original Lukoff memory would have required 128 different temperature control circuits, while Sharpless' design required only two, one for each bank of memory. Each of the two independent memory banks held 512 words in 64 mercury delay lines.

Mercury undergoes chemical reactions when in contact with most metals, which result in contamination of the mercury and the precipitation of a powdery deposit on all surfaces. In the case of the walls of the tube, this powder could be tolerated, but it interfered with the acoustic transmissions when deposited on the surfaces of the quartz crystal transducers. The contamination problem was overcome by using 3/8-inch diameter glass (rather than metal) tubes to contain the mercury and unreactive tungsten electrodes on the quartz crystals.

Each glass tube was 58 cm long from one quartz crystal to the other. This gave a total delay time of 384 microseconds: 8 words of 44 bits + 8 interword 4-bit gaps = 384 bits, each of which was represented by a pulse 0.3 μs long with a gap of 1 μs between pulses. The individual tubes were encased in a heavy U-shaped magnesium alloy with a plate over the top of the U (Figure 3). Thirty-two of these metal enclosed units were bolted to one side of a large plate made from the same magnesium alloy. Two of these large plates were mounted back to back with thermostatically controlled heating elements between them. The entire assembly of 64

delay tanks was mounted inside a heavily insulated cabinet, which was kept at 50 degrees C.[8] The two major memory cabinets were completely independent in their operation, EDVAC being specifically designed to operate using only one unit if the other had to be taken out of service for maintenance.

In addition to the 64 "long" (eight-word, 384-μs delay) mercury lines, each memory cabinet also contained three "short" (single-word, 45-μs) nonaddressable delay lines capable of acting as internal registers for the machine. Both the long and the short tank assemblies were self-contained removable units consisting of the mercury delay line, magnesium mountings, electrical connections, and shielding.[4]

The main decision for the control system concerned the speed at which the basic clock would operate (modified slightly by the speed requirements of the delay line memory system). A final decision to have the basic pulse rate of 1 megacycle per second was made, based primarily on the characteristics of the commercially available vacuum tubes to be used in the machine. It was possible to obtain tubes that would operate at significantly higher pulse rates, but the designers did not want to have to use any specially designed tubes. They also wished to keep the vacuum tubes operating at less than their nominal ratings, a design technique that had proved its worth in the earlier ENIAC project.[5]

The instruction set initially consisted of 12 elementary instructions:[9]

- A *a b g d*: Add the contents of location *a* to the contents of location *b*, put the result in location *g*, and proceed to the next instruction in location *d*.
- S *a b g d*: Subtract the contents of location *a* from the contents of location *b*, put the result in location *g*, and proceed to the next instruction in location *d*.
- M *a b g d*: Multiply (rounded) the contents of *a* by the contents of *b*, giving an 86-bit product which was then rounded down to 44 bits and stored in location *g*, the next instruction being taken from location *d*.
- m *a b g d*: Multiply (exact) the contents of *a* by the contents of *b*, store the 86-bit product in locations *g* and *g* + 1, the next instruction being taken from memory location *d*.
- D *a b g d*: Divide (rounded) — similar to instruction "M" above.
- d *a b g d*: Divide (exact) — similar to instruction "m" above.
- C *a b g d*: Compare the contents of *a* with the contents of *b*; take the next instruction from either location *g* or location *d*, depending on the results of the comparison.
- W *a b g d*: Wire — read or write information to or from the magnetic wire recorder. Later changed to "Write" for punching paper tape (this was also known as the "T" or "transfer" instruction).
- R *a b g d*: Read — not in the original set of instructions, but added when the I/O system was changed to use paper tape.
- E *a b g d*: Extract — shift a word left or right and replace some of the bits with the bits from another word. This was used to manipulate the addresses in an instruction (eight different standard extracts existed, depending on the contents of *b*).
- MR *a b g d*: Manual read — take the bit pattern set up on the front panel input switches and store this in memory locations *a*, *b*, *g*, and then take the next instruction from memory location *d*.
- H: Halt.

It was initially also intended to have a "Visual" instruction that would emit an *x* and *y* coordinate to the CRT on the front control panel to permit the graphical display of results. This seems to have been abandoned during the final construction process. As will be described later, this set of instructions was modified several times during the EDVAC's operational existence, when extra equipment and facilities were added to the original machine. Timing and internal representation of these instructions can be determined from Table 1. As a very rough first approximation, the speed of the EDVAC was about 150 times slower than that of an original IBM PC.

The input/output system was to consist solely of the magnetic wire recorders. All input to the machine was to be recorded on magnetic wires by off-line equipment and read into the machine via the "inscriber" readers. Output from the machine was to be recorded on the wire by the "outscriber" units and then printed via off-line equipment. These were devices that would convert standard punched paper tape to and from the magnetic wire form. The nickel-coated bronze wire would then be mounted on any of three on-line reader/recorders. It was assumed that each reel of magnetic wire would provide a data storage capacity of about 50,000 words, giving a total on-line auxiliary magnetic wire memory capacity of 150,000 words or about 6,600,000 decimal digits.[5] The reels of magnetic wire were specially formatted by having a 0.6-inch "marker" pulse recorded at the end of each word of data.

The process of creating data on a magnetic wire was to have had a number of safeguards to ensure that the transcription was done correctly. First, the information was to have been punched into special "chadless" paper tape. The feature that the hole was not completely removed from the paper tape stock was provided to allow the information to be printed on the paper tape at the same time as it was punched. This would allow an operator to easily read the punched tape. The chadless tape was to have been mounted in a verifier and the data rekeyed: If the second typing agreed with the first, a second standard-format punched paper tape would be produced. This second tape was to be the one used to transfer data onto the magnetic wire.

There was a concern that this machine was to be so fast that, if an error did occur, it might never be noticed. This was partially resolved by the expedient of designing EDVAC to have two identical arithmetic units. All arithmetical instructions would be performed in synchrony by these two units, and the results cross-checked at five different points within the arithmetic circuitry.[5] This is essentially the same technique that Eckert and Mauchly were to use when designing the BINAC.

Table 1. Timing and internal representation of instructions.

Instruction	Internal op code representation* (octal digit)	Timing in microseconds (dependent on operand access) Minimum	Average	Maximum	Average number performed per second
d	–5	2,256	2,930	3,600	341
m	–4	2,256	2,930	3,600	341
E	–3	192	700	1,200	1,436
W**	–2	0.25 words/second for paper tape			
MR	–1	450 μs/position			
R	0	50 words/second for paper tape			
C	1	192	700	1,200	1,436
A	2	192	860	1,536	1,157
S	3	192	860	1,536	1,157
M	4	2,208	2,880	3,552	347
D	5	2,256	2,930	3,600	341
H	6				

* The importance of this octal digit representation of the operation code can best be explained by the following quotation: "The sign of the order type is important because in modifying an order by means of an arithmetic operation, the order is considered as a signed number by the computer. This means that if it is desired to increase an address in an order which has a negative order-type by some positive number, this positive number must be subtracted from the order word."[8]

** The original W (also known as "T" or "transfer") instruction for writing information on the magnetic wire system would have taken about 35 ms to read or write a word.

Once all these concerns had been resolved, the actual detailed design was finalized in May 1947. Needless to say, some detailed design work was done during the process of finishing the conceptual design, and this introduced further delays in the project when it had to be redone to accommodate some of the later decisions. Total time taken on the design effort was about three years and, as more knowledge and design experience was gained during that time, some parts of the machine were more "primitive" than others.[5]

EDVAC construction

Once the final design decisions had been made, the task of actually constructing the EDVAC could begin. The Moore School was capable of constructing sophisticated electronic projects, but their ability to manage the fabrication of delicate mechanical components was not up to the same level. For this and similar practical reasons, the National Bureau of Standards agreed to assume responsibility for the design and construction of the magnetic wire input and output system. NBS, in turn, subcontracted it out to the Reeves Instrument Corporation.*

The Moore School's final report to the Army Ordnance Department[4] lists 46 people as having contributed to the design, construction, and testing of the EDVAC. The basic construction was performed at the Moore School and, in late 1949, it was moved to its permanent quarters at the Army Ordnance Department, Aberdeen Proving Ground, for final assembly and testing. At the time of the move it was complete except for the input/output units. Each section had been constructed and undergone initial testing, but the sections had not been combined and tested as a single integrated computer.[8]

The computer was constructed in a series of standardized cabinets 30 inches wide by 87 inches tall, the depth varying depending on the contents. A total of 12 basic units were built,[4] nine of which are shown schematically in Figure 4.

Figure 5 shows construction technicians at the Moore School standing in front of the units. At its final site at the Ballistic Research Laboratories, it sat on a cork foundation in a 46 × 20-foot room — of which the EDVAC proper occupied a space 30.5 × 14 feet. The rest of the space was eventually taken up by various bits of input/output equipment. (The cramped quarters help to explain the small size and low quality of the photographic record of EDVAC. Figure 6 is an exception.) Because, as will be detailed below, the EDVAC underwent considerable modification during its lifetime, it is not possible to say exactly how many components it contained. However, in 1949 (essentially as it was delivered to the Aberdeen Proving Ground), its circuitry was based on some 3,000 vacuum tubes — about half again

* Their experience with EDVAC was such that the Reeves Instrument Corporation, which up to that time was only involved in analog computing devices, decided to manufacture a reengineered version of the EDVAC which was to be called the REEVAC. As early as October 1, 1947, the company was contacting prospective clients and indicating that they had started work on five copies (three for the military, one for the Moore School, and one for themselves). They expected these machines to be finished by May 1948 and were willing to consider selling their own copy to the Rand Corporation for under $100,000.[10] By April 8, 1948, when a Rand representative went to investigate further, he noted, "The situation at Reeves is far different than I had expected. When I last visited them in September, they led me to expect Edvacs rolling off the production line at this time. However, considerable development work has been done in the interim, changing the previous design in several respects, and production will not start until June or July."[10]

It was only a few months later that Reeves announced it was going out of the computer business, never having even come close to having an EDVAC-like machine on the production line.[11]

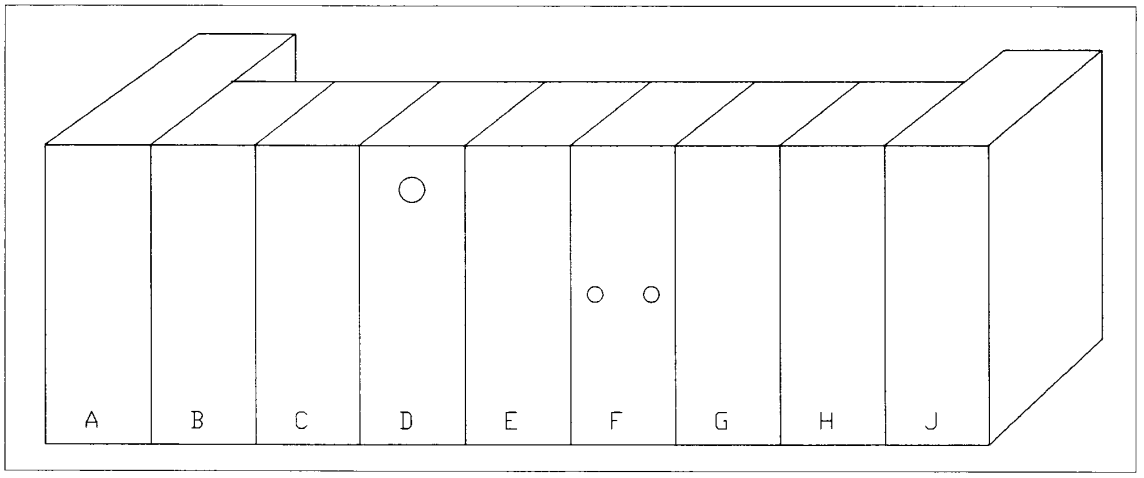


Figure 4. A drawing of the layout of EDVAC: (A) Left memory unit (2,700 lbs.), (B) dispatcher (CPU) (850 lbs.), (C) timer (clock) (850 lbs.), (D) control unit (850 lbs.), (E) dispatcher (850 lbs.), (F) magnetic wire reader/recorder (850 lbs.), (G) computer (arithmetic unit) (850 lbs.), (H) computer (arithmetic unit) (850 lbs), (J) right memory unit (2,700 lbs.), (K, L, M, not shown) power supply (4,000 lbs.).

as many as was initially envisioned by Eckert and Mauchly in their 1945 report. Table 2 gives the full inventory of components. Construction costs were said to have been $467,000.[12]

EDVAC at Aberdeen

I happened to overhear a chance remark that Mario Juncosa made at a 1988 meeting on the History of Scientific and Numeric Computation in Princeton, N.J.[13] When discussing the contributions of the Ballistic Research Laboratories, he talked of the ORDVAC computer and casually said, "Of course, the EDVAC was always threatening to work." During my investigation of the history and performance of EDVAC, I found that Juncosa's remark was quite accurate: The delivery and setup of the EDVAC was a process fraught with difficulty. As many of the early computer construction people were to discover, it was one thing to get a machine partly working on the construction floor and another entirely to get it fully functional at its final site.

Table 2. Components included in the EDVAC circuitry. Blank entries indicate no information is available. The source of the figures is noted beside the year.

	1945[2]	1949[4]	1951[9]	1957[12]	1961[12]
Vacuum tubes	1,925	3,000	3,500	4,000	5,937
Transistors	0	0	0	0	328
Relays		180			
Potentiometers		100			
Resistors		12,000		26,000	
Capacitors		5,500		6,000	
Crystal diodes		8,000		10,000	12,000
Neon bulbs		320		500	
Chokes and coils		1,100			
Transformers		485			
Wire (feet)		50,000			

Although the EDVAC was reported as being basically complete in April 1949 and was shipped from the Moore School to the Aberdeen Proving Ground during a period that began in September 1949,[14] it did not run its first application program until two years later on October 28, 1951: The program was to diagonalize a symmetrical matrix by doing 500 rotations. (This information comes from a private communication with G.W. Reitwiesner in 1988.) Even then it took a further three months before it was considered reliable enough to run a large calculation — to find the eigenvalues of a 12×12 matrix[14] in January 1952.

Even if the EDVAC was capable of performing some computational tasks in early 1952, it was still to be a long time before its use could be considered routine. Almost two years later, when a report was issued describing some proposed additions to the peripheral equipment, it began with G.W. Reitwiesner (then in charge of EDVAC operations) apologizing for the informal look of the report and saying:

> There is a certain ironic justice in the form of the present paper — reminiscent of the many occasions [before the statement (on February 2, 1953) of the director of the Ballistic Research Laboratories that the Computing Laboratory was to be congratulated upon the "successful development" of the machine] upon which the author found it necessary to accept the results furnished by the machine only with the reservation that that which was printed was not that which was intended for printing by the machine.[15]

This lengthy delay (during which time several other computers had been designed, constructed, and put in regular use in both Britain and America) in producing an opera-

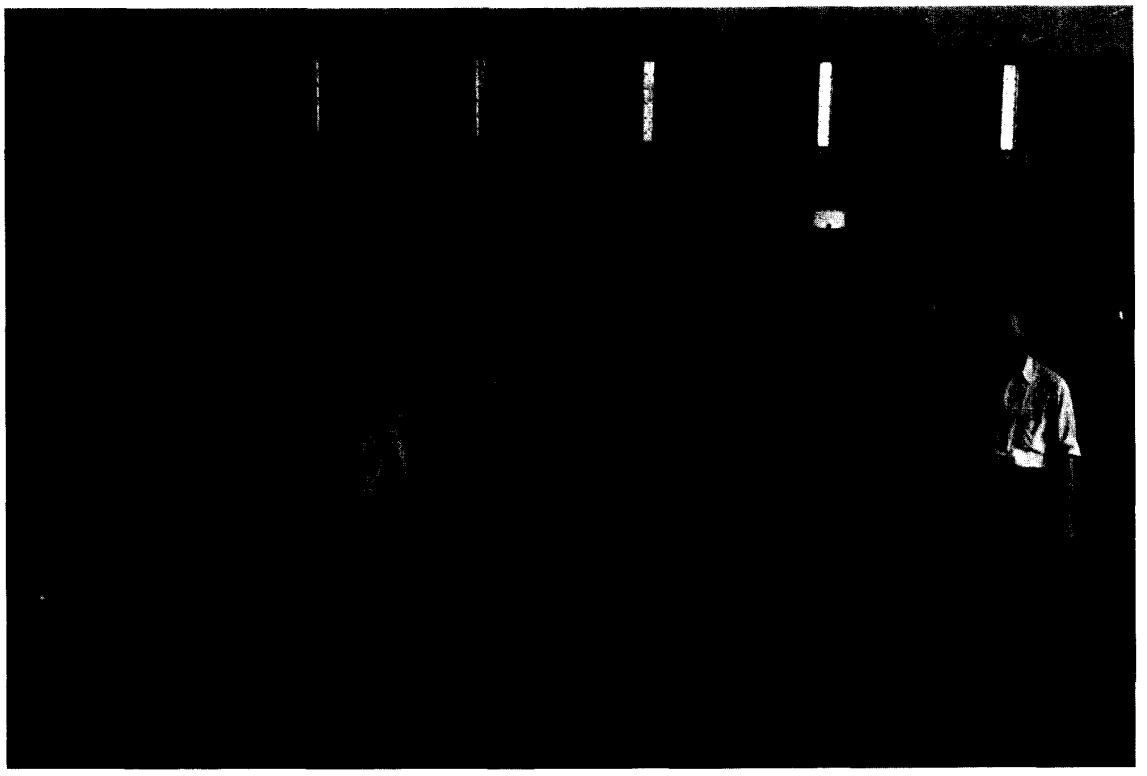


Figure 5. The EDVAC and some of the construction technicians at the Moore School. Names of the technicians are not known with any certainty. (Photograph courtesy of H.J. Gray, Moore School.)

tional device was caused by several different factors, most of which can be summed up by the fact that the EDVAC design and construction team had been breaking new ground and, as a consequence, made a lot of mistakes. The fact that the errors appeared to be more frequent and of larger magnitude than those of many of the other early computer construction teams can possibly be attributed to the lack of strong leadership that resulted from the breakup of the Moore School group at the end of the war.

Another major contributing factor was that by the time the EDVAC had started to show signs of becoming a reliable machine, the Ballistic Research Laboratories had taken delivery of another machine, the ORDVAC, patterned after von Neumann's machine at the Institute for Advanced Study. The ORDVAC was very much more reliable and faster, and could take advantage of some of the programming experience that had been gathered at other places with similar machines based on the IAS design. It was also generally easier to maintain.[16] The Ballistic Research Laboratories appear to have given EDVAC a secondary role in its computational program, behind the major machines of ENIAC and ORDVAC, and this benign neglect was a major factor in not putting out the utmost efforts to ensure a reliable machine. It was only after ENIAC died (during a severe electrical storm on the night of October 2, 1955) that a major program of enhancements to EDVAC was undertaken for that machine to assume the extra computational load.

One of the first major difficulties with EDVAC was the provision of reliable input/output equipment. As mentioned earlier, the plan had been to use three magnetic recorders for all program and data I/O. When discussing the design of the magnetic wire I/O system, Patterson et al. noted

> This decision was unwise. The decision to use wire was because it would be ready soonest — that was an error. The magnetic clutches in the wire servos were very troublesome.[5]

In fact, the wire servos were so troublesome that it quickly became obvious that the EDVAC had to have a different form of I/O system if it were ever to operate successfully. As a consequence, the wire recorders were scrapped and a paper-tape system was quickly thrown together as a substitute.

By May 1950 it was reported that the design of the paper-tape I/O system was complete and that construction had begun.[14] It was, however, a rather primitive system, even by the standards of the day. For example, the photoelectric paper-tape reader was controlled by the operator

Figure 6. The EDVAC at BRL. William Monzel is at the operator's panel, while Richard J. Bianco is attending to the paper-tape reader. (Photograph courtesy of the Ballistic Research Laboratories.)

physically pulling the tape past the read head — there was no drive motor. Enough space characters (gaps of 8 to 12 inches) had to be left between blocks of information to ensure safe stopping and starting of the tape. Each time the machine attempted to execute a "T" (transfer or read) instruction, the operator had to pull one block of information past the read head. Some concept of the slow speed of this system can be gained by noting remarks made in BRL documents such as, "Normally the entire memory can be filled in a matter of a *few minutes* employing the current photoelectric tape reader"[15] (emphasis added).

The limitations were obvious. This reader was eventually replaced by an automatic high-speed paper-tape reader in a further redesign of the I/O system (which allowed the addition of punched-card equipment) in the spring of 1954. Major problems were found to exist in the circuits used in the memory system and the power supply. In operating a recirculating mercury delay line, the pulses exiting from the mercury tube are amplified, reshaped, and then reinjected into the mercury delay system. The amplification circuitry did not perform in a satisfactory way and had to be redesigned.[14] After new memory amplifiers had been constructed in the late spring of 1951, the power supply began to have major reliability problems. Part of the difficulty was the fact that EDVAC required 15 different voltages ranging from −175 to +400 volts with currents from 5 to 22 amps.[6] Although it was too late to change the complexity of the power system, it was possible to create a more robust unit. This forced yet more redesign and reconstruction, lasting until the fall of 1952.[14] These setbacks caused a major delay of any form of regular use or testing of the EDVAC. However, by January 1953, the EDVAC was seemingly in partially productive use (despite G.W. Reitwiesner's remark about the machine's accuracy, quoted above). It was reported that

> [The EDVAC]...averaged only 37 hours per week of productive time in the last 6 months...partly due to the fact that extensive engineering changes were being done. These changes should ultimately result in increased operating efficiency of the machine.[14]

Because of all these changes to EDVAC, it is not possible to give a specific date as to when it was actually operational. It certainly ran with the old memory amplifiers, the old power supplies, and other examples of the original Moore School equipment, but became progressively more reliable

Table 3. Hours per week that the EDVAC was available for productive operation. Figures are the author's estimates based on several different sources of information.[12,14] Blank entries indicate no information is available.

	1952	1953	1954	1955	1956	1957-1960
Low	0	0	0			
High	56	159				
Average	42*	49**	72		132	145

* Average is only for the last few months of the year. EDVAC was unavailable for the majority of the time.

** This does not include a period of six weeks when the EDVAC was completely unavailable due to a major refurbishing.

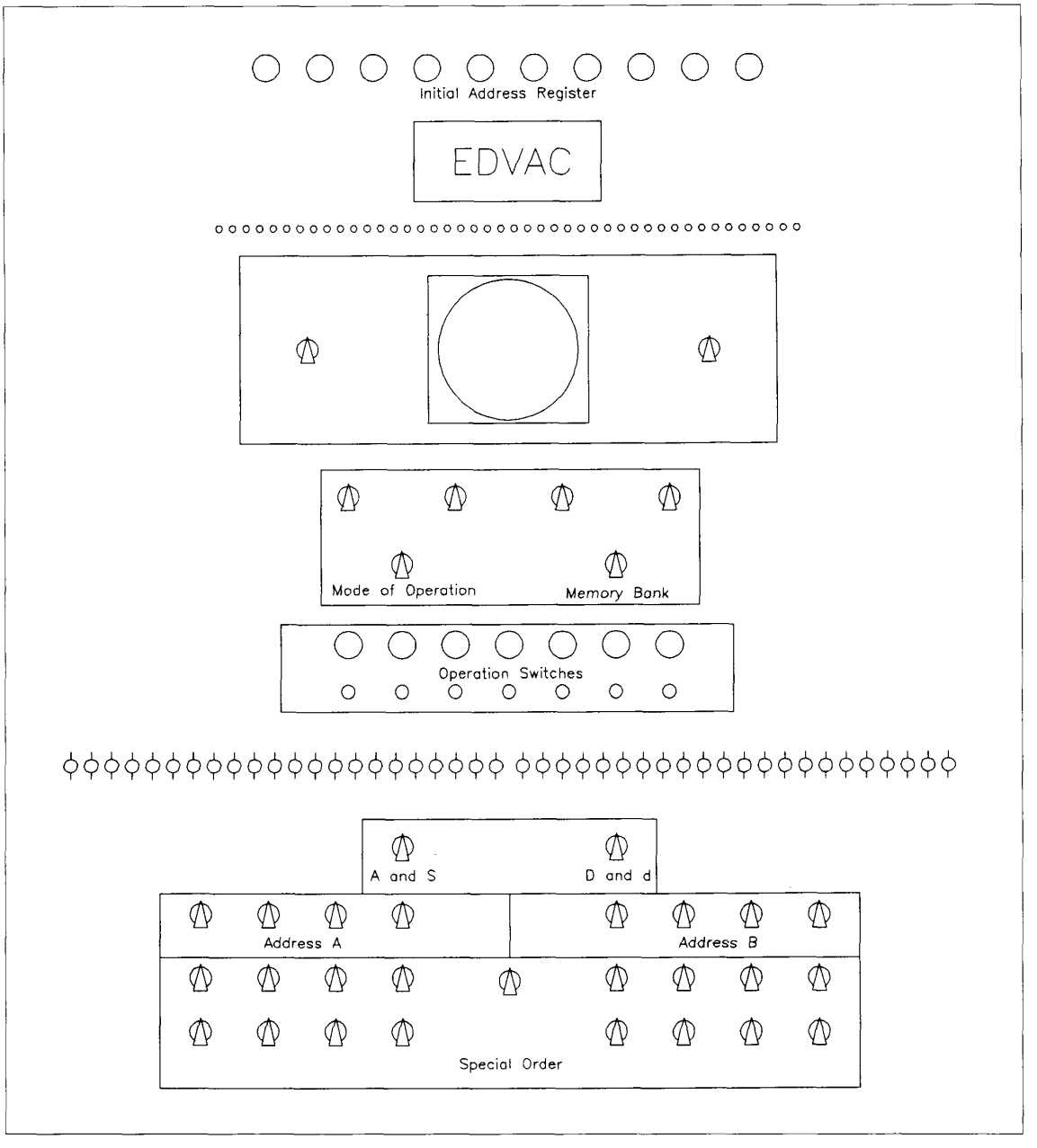


Figure 7. A diagrammatic representation of the EDVAC operator's console.

as these were gradually replaced. Performance figures are difficult to acquire, and those that are available show a wide diversity in value. However, an estimate of the number of hours per week that the machine was actually available is shown in Table 3. In 1957 it was reported that the average error-free runtime was approximately eight hours.[12]

EDVAC control console

The EDVAC was operated from the main control console — panel D in Figure 4. This consisted of an oscilloscope surrounded by a multitude of neon lights and switches, as illustrated in Figure 7.

The 10 neon lamps at the top of the panel were known as the *initial address register*. As the name implies, they indicated the contents of what would now be known as the program counter. Below this display were 44 neon lamps that could display the contents of any selected word from the memory. Below that was the oscilloscope and its associated switches, which allowed various readings from the system to be displayed for the operator or engineer, the display being controlled by the selector switches. Also within the selector switch group were two controls of importance.

The *mode of operation* switch could be set to cause the machine to operate in a number of different ways (only four of which were used during normal operation):

1. Normal high-speed operation.
2. Run until attempting to execute an instruction located at the address set on the "address A" switches, then halt (i.e., a breakpoint mode).
3. Execute a single instruction.
4. Execute the instruction set on the control panel "special order" switches (the bootstrap loader mode).

The *memory bank* switch controlled which of the two memory banks was active. The settings allowed were L0, LR, and R1. Normal operation was to use the LR position, which allowed for 1,024 words of memory (addresses 0000-0777 in the left unit and 1000-1777 in the right), and the two arithmetic units were set to check the other's operation. If either the L0 or R1 setting was used, it disabled either the left or the right memory unit (addresses were then considered as being in the range 0000-0777, whichever unit was in use), and it also disabled the dual arithmetic unit checking because each ALU required some of the short registers from an adjacent memory bank for the storage of intermediate results.

The group labeled "operation switches" consisted of a "clear" button (left) and "initiate operation" button (center) — about to be pressed by the operator in Figure 6 — and a "halt" button (right).

The 44 toggle ("knife") switches that came next on the panel were used to set up a full-word bit pattern that could then be accessed via an execution of the MR (manual read) instruction. This allowed an operator to interact with a running program by providing a facility by which single integers, or occasionally a bit pattern representing an instruction, could be accessed by the machine. This facility, in combination with the "special order" switches, was the way the machine was loaded with a simple bootstrap loader.

The machine could generate two different types of arithmetic overflow. One overflow would result from inappropriate values in the addition (A) or subtraction (S) instructions, while the other was generated by the divide (D, and d) instructions. The action taken by the machine was dependent on the settings of two (one for A and S, the other for D and d) switches known as the "excess capacity option switches" and the series of "address A," "address B," and "special order" switches in the lower section of the console. The excess capacity option switches could be set to four different positions to cause the machine to take different actions upon an overflow being detected:

1. Halt.
2. Ignore the overflow.
3. Perform the instruction set on the special order switches.
4. Jump to the instruction at the address set on the address B switches.

Thus the operating console, while not as sophisticated as that found in later equipment, was rather more convenient to use than many of the contemporary machines.

Software

Initially, of course, the EDVAC was simply a piece of hardware with no provision for software of any kind. The users were well aware of the necessity for software development. However, the constant redesign, particularly the operation of the new peripherals, which required the transfer instruction (T, –2) to be entirely redesigned,* led G.W. Reitwiesner to remark, "Ah, the task of planning programming for the EDVAC is sometimes outright maddening."[15]

One of the first software projects was to construct an adequate series of test routines. An early test routine, the EDVAC "Leap Frog Test,"[18] was based on a similar test scheme developed by David Wheeler (from Cambridge University), the concept having come to the Ballistic Research Laboratories when they took delivery of the ORDVAC.** The Leap Frog Test was a program that contained all the executable instructions on the machine — after executing once it would move

* The "T" instruction was modified so that the *d* address (the one normally used to indicate the next instruction in sequence) was changed to use bits 5, 6, and 7 to encode a 3-bit integer to control the direction of data transfer and the peripheral involved as follows: 0, transfer out to the drum; 1, transfer in from the drum; 2, transfer out to the IBM card punch; 3, transfer in from the IBM card reader; 4, transfer out to the paper-tape punch; 5, transfer in from the paper-tape reader; 6, transfer out to the high-speed printer; 7, unused.

The block of data to be transferred was specified by having the starting memory location in the *a* address and the ending in the *c* address of the instruction. This resulted in a very awkward instruction in that the *b* address area was unused, while the *d* address, which usually specified the next instruction to be executed, was used for other purposes. Thus, unlike any other instruction on the machine, the new "T" required that the next instruction to be obeyed must be the one in the next physical location in memory.

This seemingly absurd departure from the norm was, like many aspects of the EDVAC, created because of a change of direction partway through the design process. At one time, someone in BRL had proposed to install a million-word external memory.[17] This would have required the address specification in some form of I/O instruction to be 20 binary bits, and it was proposed that addresses *a* and *b* comprise one of these 20-bit addresses and *c* and *d* the other. The use of *a* and *c* as the start and end of the memory transfer block was a holdover from this scheme.

** The Ballistic Research Laboratories had the ORDVAC computer operational in 1952.[19] The ORDVAC design was based on the von Neumann IAS machine and was constructed by the University of Illinois, essentially as a copy of their ILLIAC computer. David Wheeler had spent some time in Illinois, where he helped in the development of the ILLIAC software and, presumably, introduced the concept of the Leap Frog Test while there.

itself one position further along in memory and begin execution again. After 1,024 such "leaps," every instruction had been executed from each memory location at least once. It took about 20 minutes to complete a full Leap Frog.

By 1954 a more sophisticated version, called the "EDVAC No Scotch Test," was devised.[20] This routine was essentially the same type of test as the Leap Frog, with the exception that at each step it also filled memory with a bit pattern taken from the knife switches on the EDVAC control panel. It was noted that

> While the EDVAC is passing the Leap Frog Test as part of its regular daily acceptance, the No Scotch Test will usually fail to complete 1024 leaps when put on the machine at the time of this writing (January 1, 1954).[20]

While the passing of the No Scotch Test was usually considered as a reliable indication that the EDVAC was performing properly, it was not very useful as a diagnostic tool. When the test failed, it usually left no indication of exactly what had caused the problem. This was rectified by creating a series of "Tadpole Tests," each of which contained only a few instructions.[21] If the No Scotch Test failed, the Tadpole Tests would be run, one after the other, until the problem instruction was isolated.

There was some forethought in the design of the hardware that helped with debugging the software. Most of the early computer people, particularly those using "optimal coding machines," suffered from the fact that a wild jump instruction (or incorrect *d* address) would leave them with no clue as to how the execution of a program arrived at its present location. The EDVAC contained a register that kept the last instruction executed, together with the location from which it came, and this was a great asset in the debugging process.[12] The operator's console was also handy in that it provided the facilities for a trace routine.[22] After loading the trace routine into memory, the panel switches could be set to indicate the location of the first instruction in a program, the location of the first instruction to be checked, and the address of the last instruction to be checked. The trace routine would then run the program and print out the results of each instruction as it was executed. The concept of this trace routine, like that of the Leap Frog Test, was credited to David Wheeler.[15]

The Cambridge EDSAC group had shown just how important it was to provide a reliable set of subroutines to be used for programming by publishing their famous "WWG" (Wilkes, Wheeler, and Gill) book[23] in 1951. It is ironic that the Cambridge machine, whose concept was based on the EDVAC described during the Moore School lectures, should be so far in advance of the EDVAC that its designers were already developing the science of software while the EDVAC was still not operational. The EDVAC group was, by this time, well aware of the need for subroutine software, but the design of the instruction set had been fixed before the experience of others could be taken into account. As a consequence, the EDVAC subroutines were anything but convenient to use. The passing of a return address was a difficult problem — the calling program had to put the return address in a known memory location and then the subroutine had to use the extract (E) instruction to incorporate this into the *d* address portion of the last instruction in the subroutine.

Table 4. Optimal programming addresses.

	Operand			
Instruction	***a***	***b***	***c***	***d***
A, S	x	$x+1$	$x+2$	$x+3$
E	x		$x+2$	$x+3$
M	x	$x+1$	$x+4$	$x+5$
D	x	$x+1$	$x+5$	$x+6$
d, m	x	$x+1$	$x+4$	$x+6$
C	x	$x+1$	$x+3$	$x+3$

The machine language subroutine system eventually adopted divided each routine into three sections, two of which could contain constants and variables but not instructions, while the third could contain instructions and variables but not constants — each subroutine was headed by a code word which gave the upper and lower addresses of each section. This scheme allowed some elementary "relocation" of the subroutine within the memory. Despite the best efforts of the early software designers, the limited memory size and the very elementary instruction set led to the advice that

> In the development of subroutines, coding tricks frequently are extremely advantageous...and therefore the design of subroutines should be accomplished by...programmers who have the ingenuity to find the most economical organization...to accomplish the action required of the routine and to develop methods (which may be completely unorthodox from the standpoint of conventional coding procedures) for the most economical performance of the required machine activity.[15]

Although the machine was designed with the concept of optimum coding in mind, it was little used in practice because of the extreme difficulty of making sure you had all the parameters correct. Not only did the programmer have to scatter the program's instructions around the memory space (to ensure that once an instruction was finishing the next instruction needed was just about to emerge from the delay line memory), but the operands of each instruction had to also be positioned correctly to ensure maximum speed from the machine. Each memory delay line could hold eight words and, if an operation was being started on minor cycle x (as word x was emerging from the line), Table 4 shows the addresses that had to be used to be sure of optimum speed.[8] A few moments thought will convince any programmer that attempting to satisfy all these demands at one time is extremely difficult in any except trivial programs.

By June 1953 EDVAC's programmers were making limited use of a software floating-point monitor, which was a

start in producing a workable system.[24] The system was a hybrid of an interpreter and a monitor: If an instruction would cause no problem it was executed. If it was such as to potentially take control away from the monitor routines, it was either changed or the action simulated by the monitor and then the next instruction would be examined. There was also a rather complex "break code" that would allow the execution of many EDVAC instructions in the machine's native mode. Then, when the break was encountered, the interpreter would regain control of the machine to execute any sections containing the interpreted floating-point instructions. This scheme appears to have been created in an attempt to speed up the very slow execution speed of a program when run entirely under control of the monitor. Reitwiesner does point out that

> Primary concentration in the design of these routines has been on the minimization of the storage requirement, time considerations being regarded as secondary to the desirability of the accommodation of the largest (in number of words) possible program in the memory of the machine.[24]

The system relied on the fact that EDVAC did not use all the 16 possible operation codes. The four unused opcodes became the software-interpreted instructions for the four floating arithmetic operations. Comparison between two floating-point numbers was done via the hardware compare command, as the floating-point number representation was compatible with the native numerical representation. In 1954 the floating-point monitor system was augmented by the addition of facilities for manipulating complex numbers and, to a limited extent, matrix and vector operations.[25,26]

By 1961 it was reported[12] that the EDVAC was regularly used in three major application areas: exterior ballistics (solar and lunar trajectories, guidance control data for free flight and guided missiles), interior ballistics (computations involving rocket propellants), and satellite calculations (spin calculations, tracking data, and computing orbital elements). While this statement cannot be considered definitive proof that the machine was in regular useful operation, it does indicate that it was, at last, functioning for some productive calculations and that the software base had grown to the point where it could support a range of numerical applications.

Later modifications

The very small memory of the EDVAC was one of the major problems encountered in attempting to perform any realistically large computations. This deficit was recognized as early as 1951 when the BRL had asked the Brush Development Company of Cleveland, Ohio, to design a 10,000-word drum memory. In late 1952 the drum was undergoing tests.[14] This drum did not prove satisfactory,* so BRL decided to attempt the design of its own, smaller, drum system. In the first half of 1953 the machine was shut down for extensive modifications to the basic machine,** part of which involved providing the control circuits for the future use of the drum and standard IBM punched-card I/O equipment. The results of this major refurbishing can be noted by the substantial increase that resulted in productive use of the machine, shown in Table 3, between 1953 and 1954.

The drum itself proved as difficult to bring into operation as the basic machine had been. It was two years later, in March 1955, that a synchronous magnetic drum actually became a working reality. The drum had a total capacity of 4,608 words, an average access time of 15 ms, and a transfer rate of 20,000 words per second.[14] The 1953 modifications to the "T" instruction were such as to cause the *a* and *c* addresses to specify the beginning and end of the block of memory (between 1 and 384 words in length) to be transferred, and the bit pattern in the *d* address indicated the direction of data transfer and the equipment to be used (as outlined in the first footnote on page 34). Even though this drum had been several years in the design and construction, the reliability was, at first, problematical. A few months after it had been installed, it was reported that, to improve its performance, the entire drum should be cleared before starting a program.[29]

Late in 1960 the external storage capabilities of EDVAC were again upgraded by the installation of magnetic-tape units. A second drum (16,128 words) using high-speed transistorized track-switching circuits was also planned at this time but, although some of the installation was accomplished, it never became fully operational.

Although the machine had software capable of dealing with floating-point numbers, the speed of this system was

* The Brush drum was a very complex piece of equipment. It was 18 inches long and 12 inches in diameter, and rotated at 3,254 rpm. It was held in synchronization with the EDVAC by a servo mechanism which matched the pulses generated by the mercury delay line memory with those generated by the photocell scanning a synchronizing wheel attached to the drum. This allowed the transfer of information between the memory and the drum without any intermediate buffer registers. Each word was broken into six segments, each of which was recorded on a different drum track — of course, they had to be reassembled when reading the information back into the machine. There was a total of 25 of these six-track bands on the drum, requiring 150 individual read/record heads. The various heads were switched by a bank of six-channel relays which were, in turn, themselves set by pulses from banks of vacuum tubes.[27]

** These modifications, hinted at earlier in the article, involved a complete redesign and construction of a large number of circuits. The major problems were connected with the fact that the electronics were underdesigned. There were not many logical errors but the design had incorporated a large number of crystal diode gating circuits in an effort to keep down the number of vacuum tubes required. At that time there was very little experience in the design of this type of circuit, and a large number were either marginal in operation or quickly became that way due to the aging of the components. Some indication of the state of this branch of engineering can be found in an internal Moore School document[28] that lists very simple items, such as AND and OR circuits, together with hints such as, "Don't use this one." Other critical problems were that the pulse amplifiers associated with the recirculating delay line memory system were simply not providing the required gain and had to be replaced, all the circuits in the machine had to be checked for reliability and replaced if necessary, and the input-output system had to be redesigned, adding new buffer registers for the paper-tape system and modifying the control to accommodate teleprinters, punched-card equipment, and the drum.

such as to discourage its use. Finally, in early 1958[14] BRL installed hardware floating-point operations. This increased the speed of this aspect of the machine by a factor of 12.[30] The reports of the speed vary, but it would appear that the floating-point additive operations took about 1 ms, floating multiply about 25 percent longer, and a floating divide just over 2.3 ms.

EDVAC's fate

By about 1960, when scientific machines such as the IBM 7090 were readily available, it was becoming increasingly obvious that the EDVAC was entirely outclassed and should be retired. As with a lot of machines, the decision to scrap EDVAC was not a high-priority item; besides it was actually producing some useful work.

In late 1962 the decision was finally made to scrap EDVAC at the end of January 1963. The machine was shut off for the 1962 Christmas holiday period and, when it was turned on again in January 1963, it was not possible to get it working.[31] (Here I also draw on private communications from Lloyd W. Campbell and Michael J. Romanelli.) Rather than waste a lot of time attempting to get it operational for a few extra days, the demise was simply brought forward and EDVAC ended its days, much as it had begun them, as an unreliable relic of the early days of the computer age.

Today nothing remains of the EDVAC except a few small plug-in circuit elements stored carefully away in the Smithsonian's National Museum of American History and a larger section of electronics in the basement of the Moore School of Electrical Engineering, said to have been part of the original construction effort.

The EDVAC — the machine that launched us into the computer age — is no more, and many who worked on her would say, "Just as well." ■

Acknowledgment

Support for this work was provided by the Social Sciences and Humanities Research Council of Canada.

References

Math. Coll., Smithsonian Inst. after a reference indicates that the document is available in the Mathematics Collection of the National Museum of American History, Smithsonian Institution, Washington, D.C. *Oral History Archive, Smithsonian Inst.* indicates that it is available in the Oral History Project Archive at the National Museum of American History, Smithsonian Institution.

1. N. Stern, *From ENIAC to UNIVAC, An Appraisal of the Eckert-Mauchly Computers*, Digital Press, Bedford, Mass., 1981.

2. J.P. Eckert and J. Mauchly, "Automatic High Speed Computing: A Progress Report on the EDVAC," Moore School of Electrical Engineering, Univ. of Pennsylvania, Philadelphia, Sept. 30, 1945. (50 numbered copies of this report, each page of which was stamped "CONFIDENTIAL," were produced. Copy no. 43 is available in Math. Coll., Smithsonian Inst.)

3. M. Campbell-Kelly and M.R. Williams, *The Moore School Lectures: Theory and Techniques for Design of Electronic Digital Computers*, Charles Babbage Inst. Reprint Series for the History of Computing, Vol. 9, MIT Press, Cambridge, Mass., and Tomash Publishers, Los Angeles, 1985.

4. S.E. Gluck and W.H. Boghosian, *A Functional Description of the EDVAC*, Vols. 1-2, Moore School of Electrical Engineering, Univ. of Pennsylvania, Philadelphia, Nov. 1, 1949. (Subtitled "A final report on the work performed for the Ordnance Department, Department of the Army under contract W36-034-ORD-7593.")

5. G.W. Patterson, R.L. Snyder, L.P. Tabor, and I. Travis, "The EDVAC, A Preliminary Report on Logic and Design," Report No. 48-2, Moore School of Electrical Engineering, Univ. of Pennsylvania, Philadelphia, Feb. 16, 1948.

6. G.W. Patterson and R.L. Snyder, "Recent Developments at the Moore School," *Symposia on Modern Calculating Machines and Numerical Analysis*, Univ. of California, Los Angeles, July 29-31, 1948. (Also published as Report No. 49-5, Moore School of Electrical Engineering, Univ. of Pennsylvania, Philadelphia.)

7. H. Lukoff, *From Dits to Bits... A Personal History of the Electronic Computer*, Robotics Press, Portland, Ore., 1979.

8. H.B. Hilton, "Problem Programming for the EDVAC," Memo Report No. 771, Ballistic Research Laboratories, Aberdeen, Md., Mar. 1954.

9. M. Lotkin, "Coding of Problems on the EDVAC" (duplicated lecture notes from the "Ballistic Institute" series of BRL lectures), Ballistic Research Institute, Aberdeen, Md., March-June 1951. (Math. Coll., Smithsonian Inst.)

10. A.S. Mengel, letter to J.R. Goldstine of the Rand Corp., Oct. 1, 1947. (Copies in Rand Corp. files and Oral History Archive, Smithsonian Inst.)

11. S. Snyder, transcript of a 1970 oral interview, pp. 25-28. (Oral History Archive, Smithsonian Inst.)

12. M.H. Weik, "A Second Survey of Domestic Electronic Digital Computing Systems," Ballistic Research Laboratories Report No. 1010, US Dept. of Commerce, Washington, D.C., 1957, also "A Third Survey," Report No. 1115, 1961.

13. S.G. Nash, *A History of Scientific Computation*, Addison-Wesley, Reading, Mass., 1989.

14. *Digital Computer Newsletter*, Vols. 1-14, Office of Naval Research, Washington, D.C., 1949-1962.

15. G.W. Reitwiesner, "The EDVAC from a Programming Point of View" (alcohol duplicated document), Branch Note No. 65, Analysis and Computational Branch of the Computing Laboratory, Aberdeen Proving Ground, Md., 1953. (Math. Coll., Smithsonian Inst.)

16. R.F. Clippinger, transcript of a 1970 oral interview. (Oral History Archive, Smithsonian Inst.)

17. "Programming of Input-Output Orders" (alcohol duplicated sheets), Ballistic Research Laboratories, Aberdeen, Md., no date. (Math. Coll., Smithsonian Inst.)

18. W.W. Leutert, "The EDVAC Leap Frog Test" (alcohol duplicated document), Jan. 13, 1952. (Math. Coll., Smithsonian Inst.)

19. C.R. Williams, "A Review of ORDVAC Operating Experience," Memo Report No. 756, Ballistic Research Laboratories, Aberdeen, Md., Jan. 1954.

20. W.W. Leutert, "The EDVAC No Scotch Test" (mimeographed document), Jan. 1, 1954. (Math. Coll., Smithsonian Inst.)

21. W.W. Leutert, "The EDVAC Tadpole Test" (mimeograph duplicated document), no date. (Math. Coll., Smithsonian Inst.)

22. "Edvac Code Checker" (alcohol duplicated sheet), Ballistic Research Laboratories, Aberdeen, Md., no date. (Math. Coll., Smithsonian Inst.)

23. M.V. Wilkes, D. Wheeler, and S. Gill, *The Preparation of Programs for an Electronic Digital Computer*, Addison-Wesley Press, Cambridge, Mass., 1951; Charles Babbage Inst. Reprint Series for the History of Computing, Vol. 1, MIT Press, Cambridge, Mass., and Tomash Publishers, Los Angeles, 1982.

24. G.W. Reitwiesner, "EDVAC Floating Point Monitoring Routines" (alcohol duplicated document), Feb. 1953. (Math. Coll., Smithsonian Inst.)

25. G.W. Reitwiesner, "An EDVAC Complex Number Monitoring Routine" (alcohol duplicated document), Mar. 1954. (Math. Coll., Smithsonian Inst.)

26. G.W. Reitwiesner, "An EDVAC Matrix and Vector Monitoring Routine" (alcohol duplicated document), Mar. 1954. (Math. Coll., Smithsonian Inst.)

27. D. Eadie, "EDVAC Drum Memory Phase System of Magnetic Recording," *Electrical Eng.*, Vol. 72, No. 7, July 1953, pp. 590-595.

28. T.C. Chen, "Standard Crystal-Diode Gating Circuits for EDVAC," Report No. 49-3, Moore School of Electrical Engineering, Univ. of Pennsylvania, Philadelphia, July 12, 1948.

29. T. McGann, "EDVAC Drum Clear Card" (alcohol duplicated single sheet), Ballistic Research Laboratories, Aberdeen, Md., July 18, 1955. (Math. Coll., Smithsonian Inst.)

30. K. Kempf, "Electronic Computers within the Ordnance Corps," Ballistic Research Laboratories, Aberdeen, Md., 1961.

31. H.D. Huskey, "EDVAC," in *Encyclopedia of Computer Science and Engineering*, A. Ralston, ed., Van Nostrand, New York, 1983, p. 599.

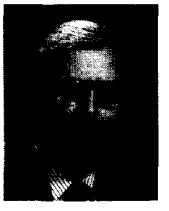
Related reading

J.P. Eckert, "Disclosure of Magnetic Calculating Machine," 1944, excerpts reprinted in N. Stern, *From ENIAC to UNIVAC*, Digital Press, Bedford, Mass., 1981, pp. 28-75, and reproduced in full in H. Lukoff, *From Dits to Bits*, Robotics Press, Portland, Ore., 1979, pp. 207-209.

H.D. Huskey, "Electronic Digital Computing in the United States," *Report on a Conference on High Speed Automatic Calculating Machines*, Cambridge, UK, June 22-25, 1949, pp. 109-111. Reprinted in *The Early British Conferences*, Charles Babbage Inst. Reprint Series for the History of Computing, Vol. 14, MIT Press, Cambridge, Mass., and Tomash Publishers, Los Angeles, 1989.

A.S. Mengel, letter to John Williams of the Rand Corp., April 8, 1948. (Copies in Rand Corp. files and Oral History Archive, Smithsonian Inst.)

J. von Neumann, "First Draft of a Report on the EDVAC," Moore School of Electrical Engineering Report, Univ. of Pennsylvania, Philadelphia, 1945, reprinted in N. Stern, *From ENIAC to UNIVAC*, Digital Press, Bedford, Mass., 1981, pp. 177-246. (The best reproduction of this report is available from Michael Godfrey. See page 12 in this issue for information on how to obtain it.)



Michael R. Williams is a professor of computer science at the University of Calgary. Educated in Canada and Scotland, he joined the University of Calgary in 1968, where he has held various teaching and administrative positions. He has also worked and lectured at various institutions in North America, Europe, and China. His main teaching and research interests are in the history of computation, with an emphasis on the people and the technical aspects of the subject.

Williams can be reached at the Department of Computer Science, University of Calgary, Calgary, Alberta T2N 1N4, Canada. e-mail: williams@cpsc.ucalgary.ca.