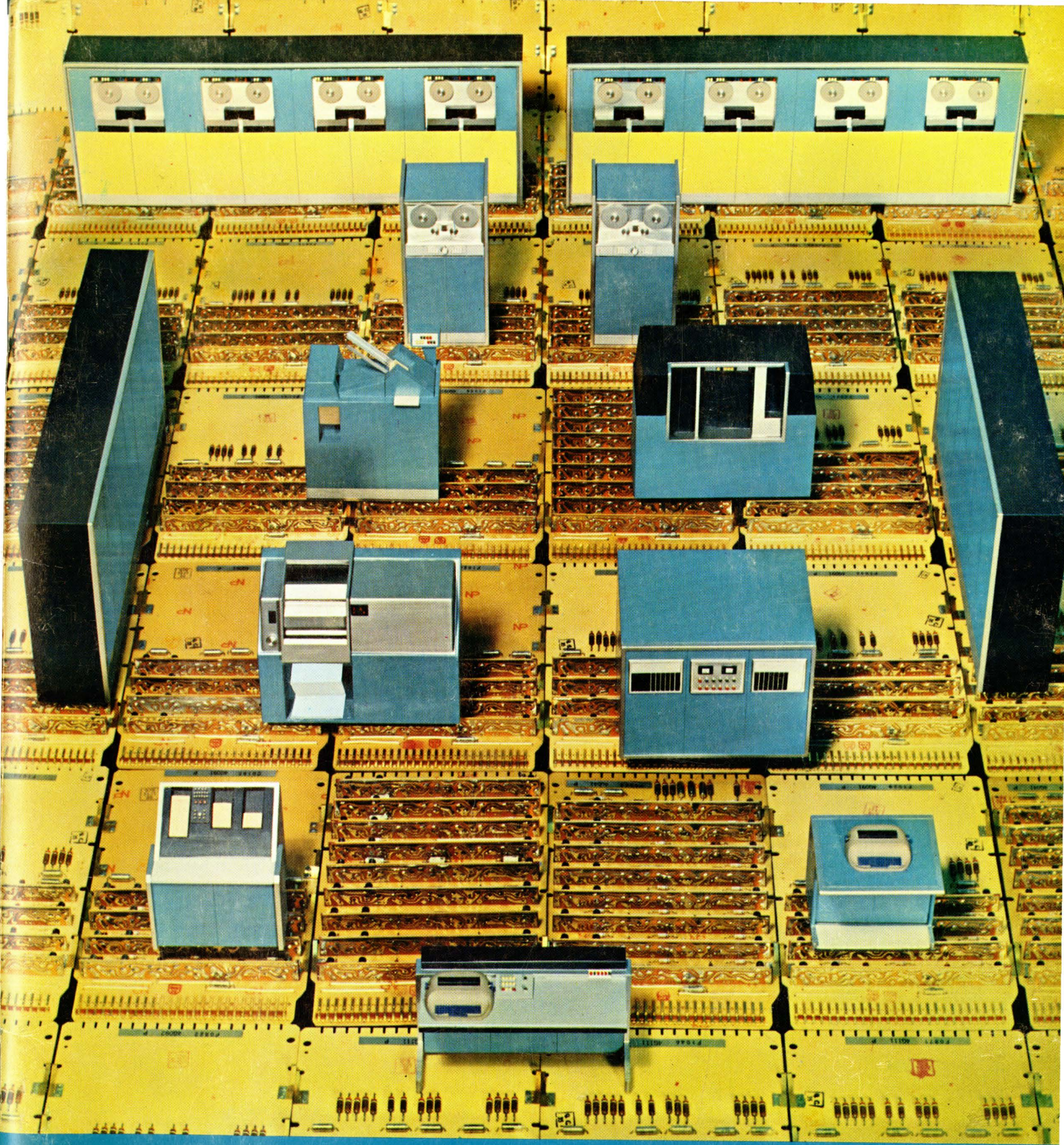


RCA ACHIEVEMENTS IN COMPUTER SYSTEMS AND COMPONENTS

The Tidal Wave of Electronic Data Processing

Electronic data processing—from a small ripple in the late 1940's—has surged into a powerful force for business, industry, and the military, and its momentum is increasing at a rapid rate. Technological and application developments have reinforced each other, providing additional impetus and directions for the field to propagate.

ENIAC, the first electronic computer, was delivered to the Aberdeen Proving Grounds in 1947. Mathematicians used its high-speed computation ability, provided by vacuum tubes, to solve artillery ballistic problems. In the early 1950's, magnetic memories gave the computer increased power and flexibility. In the same era, magnetic tape stations and drums began providing large and rapid bulk files for data and instruction storage. Augmented by high-speed input equipment and printers, the computer—which had been the almost-exclusive tool of the mathematician—became part of a data-processing system that could serve the business world in areas such as stock control, payroll application, and insurance billing.

The RCA 501 heralded the advent of the transistorized computers, which increased reliability, performance, and compactness and opened up new doors in the fields of military and industrial control.

Today, the practical applications of data-processing systems are being expanded by the development of micromagnetic and cryogenic memories, random-access files, optical character reading, data input and collection devices, fast and flexible display systems, and powerful means for integrating communication facilities with computers. The host of technological advances not only are opening up new functions but also are providing increased system performance at lower cost, bringing many new applications into economic practicality.

But in the final analysis, technological advances are only generating better and better tools. Far more people are required *to effectively utilize* these tools than to develop them. The challenges of successful application are in many ways more difficult. Historically, it has been difficult and slow to overcome the inertia and smugness of present practice. The rewards are great, however, and one only has to view the many application advances to "get the picture."

Engineering is presently utilizing computer systems for mathematical reductions, statistical analysis, circuit design, generation of wire lists, drawing control, schedule and cost control, and simulation studies, to name a few. *Increased productivity* is a primary challenge to all of us, and it is the goal that we *must* achieve to stay competitive—whether from an individual, or company, or national point of view. Electronic data processing provides a tremendous opportunity to achieve this goal. It will require that the individual become knowledgeable and conversant with this tool and that he apply it imaginatively to his area of responsibility.

The wave is here. *Are you prepared to swim?*

Arthur D. Beard

A. D. Beard
Chief Engineer
Electronic Data Processing
Radio Corporation of America

PE-152

This publication is a collection of technical papers by RCA scientists and engineers describing a series of Corporate achievements in the evolution of the electronic computer. Originating entirely within RCA, these advances depict the many facets of its in-house capabilities from computer components to complete systems.

COVER PHOTO—Models of the new RCA 3301 REALCOM atop an array of plug-in boards dramatically visualize this advanced new system.

FIVE YEARS OF PROGRESS ... A Review of RCA Electronic Data Processing Product Engineering

The last five years have been very significant technically in the engineering of RCA product-line computer systems. This paper reviews those technical landmarks and provides an over-all view of the present product-line state of the art in RCA Electronic Data Processing Engineering.

J. N. MARSHALL, Mgr.

Computer Advanced Development
Technical Programs—EDP & DEP
Cherry Hill, N.J.

J. N. MARSHALL graduated from Cambridge University, U.K., with an Honors Degree in Mathematics and Physics in 1939. He then worked on rapid-scanning radar and identification at the R.A.F. Research Establishment. In 1943, he was assigned to the U.S. Naval Research Laboratory as project engineer on RF test equipment and check-out systems for a radar identification system. With RCA since 1946, he has been project engineer on part of the Teleran Air Navigation System and a Target Designation System for the Navy. As Mgr., Advanced Development, for Commercial Electronic Products, he supervised development of a computer, a complex telemetering system, and industrial instrumentation of various types. Since joining EDP four years ago, Mr. Marshall has headed up development engineering groups on industrial computer applications and ultra-high-speed computers. He is now Mgr. of Computer Advanced Development, reporting to C. A. Gunther, Vice President of Technical Programs for EDP and DEP. He is an IEEE Member and has several U.S. Patents.

TWO MILESTONES were passed by RCA Electronic Data Processing late in 1962: *the first* RCA 601 computer was delivered to New Jersey Bell Telephone Company; and *the first* Air Force DATA-COM (Phase I, COMLOGNET) installation containing two Communication Data Processors (CDP's) was accepted by the Air Force.

The RCA 601 and the CDP are large computers with 5×10^5 to 1.5×10^6 bits of memory capacity operating at a 1.5-μsec cycle, a long word length of 56 bits, and an unusual flexibility of instruction handling, both word and character. They are the most powerful machines designed for data processing and are exceeded in size and capability only by the very largest scientific machines, such as STRETCH and LARC. These achievements make RCA a *full-line supplier* of EDP equipment.

EARLY COMMERCIAL COMPUTERS

The delivery of the first RCA 501 system in 1958 was the start of RCA's large-scale entry into the data processing field. Prior to that, the delivery of five BIZMAC systems represented an enormous engineering achievement, and provided an education in marketing, planning, programming, and fabrication which made the RCA 501 possible. But it is these last five years that have seen EDP mature. A wide variety of products were introduced between the RCA 501 and 601, of which the RCA 301 was unquestionably the most important. Approximately 300 RCA computing systems are now in the field and 100 systems are on order; in addition to RCA's computers going to external customer applications, the various divisions of RCA are making good use of EDP's commercial computer designs (see Table I).

This five-year period can be considered as a first full cycle for RCA, and makes it suitable at this time to review what has happened technically during this period.

ON-LINE OPERATION

One of the features of the RCA 501 was its capability to operate printers *on line* without use of tape as an intermediary. A simultaneous mode was provided in which an input or output instruction could be held and operated as required at the relatively slow rate of the peripheral devices; at the same time, computing instructions were executed in the normal mode, and only interrupted when the memory was actually needed for transfers to and from peripheral equipment. Tape read or write could be performed simultaneously with computation, or the computer could be assigned to operate the printer essentially whole-time with a limited amount of editing. This welcomed innovation permitted an improvement in sorting and merging functions on tape and avoided the high cost of the electronics in existing off-line printer equipment.

More sophisticated solutions for operating multiple peripheral equipment have since been developed. The improvements in techniques to maximize the utilization of peripheral equipment, and to work on-line to multiple communication channels or data sensors, are important areas of present development.

The RCA 301 control units permit (with some restrictions) the use of simultaneous modes by different peripheral equipment alternately, provided they do not use too large a percentage of the memory cycles all together. In this way, more than one simultaneous function is possible. Recently, a control unit for multiple communication channels has been added which can receive or transmit on 80 telegraph channels. The *Communications Mode Control* (CMC-80) samples single-channel buffers on each line and loads or empties them to the memory as required. The memory addresses are decided by the count in the sampling mechanism so that each transfer automatically occurs to (or from) the memory without need for any routine to find the appropriate address. Block transfers are then arranged between the general work areas of the memory and this staging area which directly transfers data to the CMC-80. A system using the CMC-80 and RCA 301 provides immediate connection to *Electronic Data Gathering Equipment* (EDGE) units installed at a Lockheed plant in California.

The RCA 110 control computer includes another feature in that any input device such as a pressure or temperature sensor could automatically interrupt the computer and go to a specialized subroutine, without any input-output instruction having appeared internally. These interrupts have five levels of priority so that, for example, alarm warnings can take top priority. Those interrupts set flip-flops which are sensed automatically by the computer between instructions.

The RCA 601 has a sophisticated scheme of this kind which permits running multiple input-output functions up to a level set by the individual speed-weight ratios of the functions. Memory interlocks are provided so that input-output routines, or simultaneous computing routines cannot have access to the wrong areas and damage information belonging to other programs.

The RCA 4101 military computer also has a priority-interrupt scheme and multiple simultaneous operations.

The problem is to minimize the hardware required for buffers and storing of alternate addresses, instructions, etc.,

Table I—RCA 301, 501, and 601 Computers at RCA Locations

RCA LABORATORIES, PRINCETON, N.J.
RCA 601; RCA 301
Scientific Calculations
Computer Research

RCA VICTOR HOME INSTRUMENTS DIV., INDPLS., IND.
RCA 501; RCA 301
Warehousing Shipping
Analysis of Finished Goods
General Accounting

NATIONAL BROADCASTING CO., NEW YORK CITY, N.Y.
RCA 301
Payroll
General Accounting
Station Time Billing
Station Switching
Cost Accounting

RCA SERVICE CO., CHICAGO, ILL.
RCA 301; RCA 501
Demand Deposit
Bank MICR Check Sorting
Payrolls
Subscription Fulfillment
Reader Inquiry Service
Mailing List Maintenance
Broker Accounting
Bookkeeping Service
Test Scoring and Analysis
Records and Statistics

RCA SERVICE CO. (FINANCIAL) CHERRY HILL, N.J.
Two RCA 501's, RCA 301
General Accounting
Gross Margin Reporting
Demand Service
Contract Fulfillment
Payroll
Physical Inventory
Demand Deposit
Reader Inquiry Service
Broker Accounting
Government Statistics
Subscription Fulfillment
Mailing List Maintenance
Bookkeeping Service
Stock Transfers

RCA SERVICE CO., WASHINGTON, D.C.
RCA 501
Demand Deposit
Payrolls
Subscription Fulfillment
Reader Inquiry Service
Mailing List Maintenance
Broker Accounting
Bookkeeping Service
Government Statistics
Service Billing
Federal Credit Union Accounting
International Trade Statistics
Test Scoring Analysis
Records Statistics

ELECTRONIC COMPONENTS & DEVICES, HARRISON, N.J.
RCA 301; RCA 501
Sales Reporting
Profitability of Items
Gross Margin Records
Financial Planning
Factory Loading
Quality Control
Statistical Reporting
Financial Reporting
Inventory Control
Central Billing

RCA COMMUNICATIONS, INC., NEW YORK CITY, N.Y.
RCA 501
Billing
Communications Statistics
General Accounting
Payroll
Traffic Bureau Statistics
Accounts Receivable
FCC Statistics
Foreign Message Recording

COMMUNICATIONS SYSTEMS DIVISION, DEP, Cambridge, Ohio
RCA 301
Payroll
Manufacturing Control
General Accounting

ELECTRONIC COMPONENTS & DEVICES, LANCASTER, PA.
RCA 301
Payroll
Marketing Analysis
Factory Loading
Material Control
Cost Standard
Life Test

DEFENSE ELECTRONICS PRODUCTS, CAMDEN & MOORESTOWN
RCA 501 in Moorestown;
Two RCA 501's and
RCA 301 in Camden
Payroll
Material Operations
MINUTEMAN Project
Quality Control

AEROSPACE SYSTEMS DIV. VAN NUYS, CALIF.
RCA 301
Engineering Simulation
General Accounting
Payroll
Purchasing

DATA SYSTEMS CENTER, BETHESDA, MARYLAND
RCA 301
Information Retrieval
Equipment Design
Research New Techniques
Data Reduction
Engineering Evaluations

ASTRO ELECTRONICS DIV., DEP, HIGHTSTOWN, N.J.
RCA 501
Scientific Simulation
Information Retrieval
BMEWS Point-to-Point Contact
Scientific Space Calculations
Data Reduction

RECORD DIV., INDIANAPOLIS, IND.
RCA 301
Distribution Warehousing
Production Control
Material Control
General Accounting

EDP, NEW YORK SERVICE CENTER, N.Y.C., N.Y.
Two RCA 501's; RCA 301
Brokerage Service
Savings Accounting
Subscription Fulfillment
Mailing List Maintenance
Property Title Records
Demand Deposit
Payrolls
Reader Inquiry Service
Billing and Accounts

EDP, CHERRY HILL DATA CENTER, CHERRY HILL, N.J.
RCA 501; Two RCA 301's
Customer Support
Payroll
Programming Support
Sales Demonstrations

Note: RCA 301, 501, and 601 computer installations at RCA plants in West Palm Beach, Camden, Pennsauken, and Van Nuys are also primarily used for engineering design and development techniques, rather than the commercial-type computer uses shown above.

and still provide a break-in on interrupt, to indicate completion of a read-in cycle and provide lock-outs for memory. At the same time, maximum performance is required in terms of using a minimum number of memory cycles for servicing input or output. For a data-processing machine, elegant solutions in this area are more important than arithmetic capability.

PROGRAMMABILITY

The RCA 501 was the first machine to be designed after a real programming capability had been developed in RCA. Its specification was determined by a team of programmers and logic designers working together to explore trade-offs between hardware costs and program convenience, and to design a machine that fits neatly together.

The RCA 501, unlike the BIZMAC machines, had no drum. The drum had been used to provide program storage at reduced cost, and instructions were surged into the core memory in groups of 64 in order to be executed. Part of the reason was the reduced cost of core memories, but more important was the view that the automatic programming techniques then envisaged would be severely complicated by two levels of storage.

Automatic programming—the use of the computer to convert a program written in simplified form to detailed machine language—was already well developed for scientific applications. Assembly routines using simplified instruction formats (corresponding to one or two machine instructions) and assigning memory locations for data were in general use. Algebraic compilers, such as FORTRAN, took convenient mathematical statements (not simply related to machine instruction) and constructed programs to execute them. However, there were no sophisticated business compilers such as the *Common Business Oriented Language* (COBOL) which was first used by RCA and Univac in 1960.

An alternate form of automatic programming, first thoroughly developed at Bell Laboratories, is the *interpreter*. This type of program is initially developed in a convenient language (such as algebraic forms in scientific work) and then interpreted by routines stored in the computer's memory. This is inefficient in terms of time to do the job on a computer and wastes memory space. However, as it involves no previous compiling time, it is economical for single-shot jobs and is widely used for engineering purposes.

An ideal computer would interpret from instructions in natural language, with an efficiency in performance equal to compiled programs. Attempts are now being made in this direction. An example is the English Electric KDF9 which uses push-down memories to provide powerful arithmetic functions fitting well with an algebraic notation called *Reverse Polish* in which a sequence of data symbols and operators describe the arithmetic functions to be performed.

In business data processing, it has so far appeared that a good data processor is automatically good at compiling and convenient to compile for. There has not been a major effort to design machines performing more of the automatic programming in the hardware. This relation will, however, become more important in the future and machines using instructions in natural language directly are a possibility.

CHARACTERS AND WORDS

Important to RCA was the introduction of variable word length in BIZMAC, where data was handled character-by-character and symbols were used to designate item and message separation. This organization of data had proved

convenient for business users. It was important in conserving tape, since data was packed tightly without spaces or zeros to fill out standard word lengths. It also saved programming effort in that the symbols could automatically complete instructions and complicated multiple precision arrangements were avoided since there was no fixed word to overflow. It was adopted by other manufacturers (the best confirmation) and included in the RCA 501.

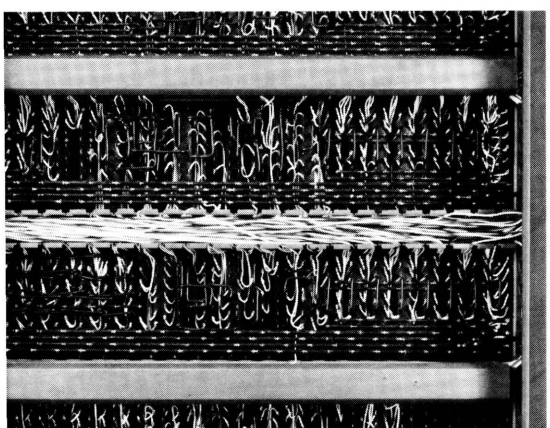
However, character-by-character operation is slow and in one place in particular, the reading of instructions from memory, was too great a restriction. The RCA 501, therefore, reads four characters (28 bits) at a time from a 28-bit-deep memory and staticizes instructions (8 characters or 56 bits) in two reads. This feature was also adopted for data transfers, which take place in *tetrads*, (four characters at a time). The RCA 501 therefore has fixed-word features—so does the RCA 301, but being a small machine, it staticizes and transfers two characters at a time.

Though the earliest electronic computers were serial decimal, machines which were designed for high-speed arithmetic functions have been parallel-binary fixed-word-length ever since the early 1950's. At the end of the 1950's, machines of this type appeared with the ability to address parts of words by additional bits on the address, and thus had some character-by-character capability and adaptability to variable word length. Some machines were also built using parallel-decimal operation, such as the *Livermore Atomic Reactor Computer* (LARC) and the IBM 7074. The 7074 has capability for handling 10 decimal digits in parallel, as well as character-by-character operations.

The RCA 601 is a sophisticated composite machine having both fixed-word and variable-word character-by-character operations. It can address either words or any fraction of a word, such as a character, and can operate character-by-character with either symbol control to indicate the end of a variable-length word, or a fixed number of characters not necessarily forming a number of complete words. It also has variable instruction length. It has built-in arithmetic capability which treats the 56-bit word as 8 decimal digits operated on in parallel; it can also operate character serially. The RCA 604 high-speed arithmetic unit, an attachment for the processor in the RCA 601 system, provides parallel binary and decimal floating-point operations.

What was a direct choice between fixed- and variable-word machines in the BIZMAC days has become a complex issue of internal organization, and a three-way problem between programming convenience, complexity and speed. The RCA 601 to a considerable extent achieves the best of both worlds, but it has not been determined how to accomplish this in smaller machines.

Fig. 1—This photo of the RCA 501 back-panel wiring is an example of neat, clean, business-like wiring by use of panduits.



COMPUTER HARDWARE

One of the problems in the early days of the RCA 501 development was to decide between transistors for logic circuits and a variety of pulse and AC-driven magnetic circuits. The appeal of the magnetic circuits was reliability. It took a full-scale study of a magnetic computer on a government contract to show that magnetic circuits driven by a 100-kc sine wave permitted the transit of only one logic stage in each half cycle (5 μsec); in contrast, transistor circuits using a 200-kc clock could transit through five to ten logic layers in the 5 μsec of a single time-slot, and were therefore five to ten times faster. Today, with stage or pair delay a standard parameter, this would be more quickly recognized. It also turned out that magnetic circuits needed about ten diodes per gate to operate at this speed—this undermined the reliability argument. At intervals during the RCA 501 development, rumors were received that a competitor had fast, simple, cheap and reliable magnetic circuits, and the issue came up for discussion. One of the arguments for transistors was their broad application in non-computer applications to ensure low device cost and good reliability as a result of the high volume and the resultant funds for development and design.

The RCA 501 was the first large transistorized computer to be delivered to a customer. We have no cause to regret the decision to transistorize. The reliability has indeed turned out to be excellent even though we can now design circuits with much better margins than were then possible. Within a year of the first delivery, it was apparent that transistor failure would not be an important source of system downtime; in fact, the lifetime maintenance cost of a transistor circuit was not more than 10% of its original cost.

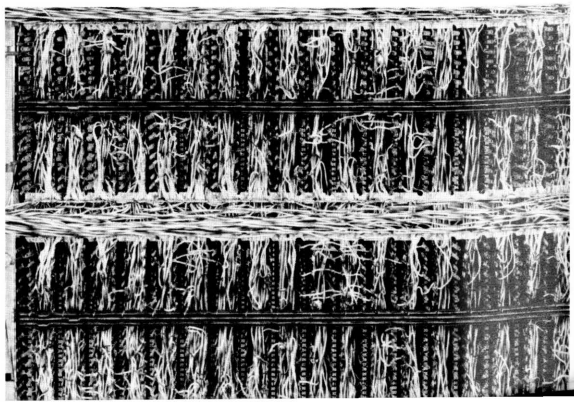
When transistors were first used, there was some interest in point-contact units because of their higher speed; there was no real belief that junction transistors would ever be as fast as tubes. The junction units were preferred because of reliability. Within a few years, the RCA 601 circuits appeared with 50-nsec stage delays instead of the 500 nsec of the RCA 501 circuits; these were as fast as any tube circuits ever built. As far as the author knows, this was the carry propagation circuit in the IBM 704 with the identical 50-nsec stage delay. Now, we can see another improvement factor of ten with transistors. These improvements will be a major factor in increasing computer performance. The cost of transistor circuits is now 25 to 30% that of the tube circuits of 1957.

The RCA 501 memory problem was to transistorize all associated circuits. The magnetic core stack is only slightly changed from the BIZMAC stack and operates at essentially the same speed. The combination of high current, speed, and back voltage required in memory drivers was a real problem for the devices. The prototype RCA 501 went into test with a tube-driven memory built as an insurance program, just because of these difficulties. Yet, in a few short years, the RCA 601 memory appeared at ten times the speed (1.5- instead of 15-μsec cycle time). Even in the RCA 301, where cost was the prime factor, the memory operates at 7 μsec.

INTERCONNECTIONS

One area which has rapidly grown in importance since the RCA 501 design, is interconnections. In BIZMAC days, signal wires were on stringers and were separated by about an inch to reduce capacity coupling which was a problem in the high impedance tube circuits. Transistor circuits are lower impedance and tests indicated that signal wires could be bundled together without undesirable levels of cross-talk. This turned out to be justified in all cases except the

Fig. 2—Higher speeds and advanced programming needs have created increased wiring densities as shown in this photo of the RCA 601 back-panel wiring.



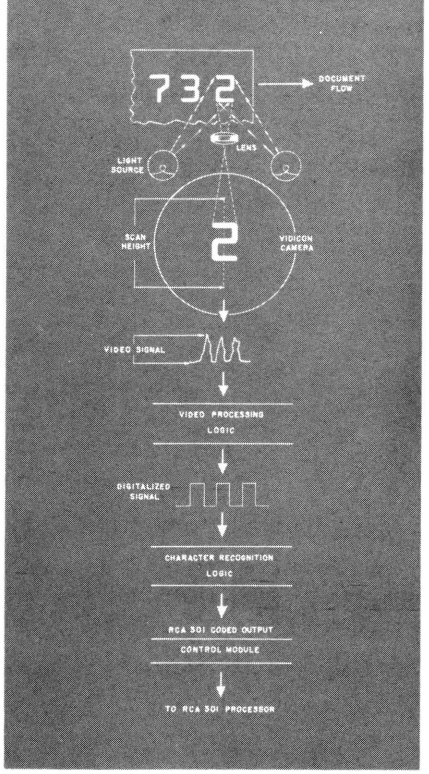


Fig. 3—Principle of the RCA VIDEOSCAN optical document reader.

Fig. 4—A prototype VIDEOSCAN document reader which, utilizing the TV vidicon tube, allows high-speed reading of input documents—for example 90,000 utility bills, premium notices, and similar turnaround documents—directly into the RCA 301 computer, here shown in the background.

long cables to the memory where coax had to be used. The placement of wires in *panduit* or covered slots (instead of using harnesses) made the signal wiring of the RCA 501 (Fig. 1) much simpler than BIZMAC. Unfortunately, increasing speed and decreasing size of modules are again making this problem increasingly difficult. The RCA 601 and future faster computers must use some kind of transmission line, twisted pair or coax, for all but the shortest connections (Fig. 2).

The first BIZMAC installation used 30,000 tubes, and since each tube had an eight-pin socket, and was also on a plug-in board, there were about 350,000 spring contacts. This received careful attention but it was never a major reliability problem. The lower voltages and higher currents of transistors made this a more difficult problem even though transistors were soldered into circuits. A much greater effort went into the plug and socket; but by careful design and the use of gold plating, successful results were achieved.

For new machines, the interconnections and fabrication scheme are a major problem. The increasing number of circuits on a board has led to serious problems in controlling the number of different types of boards. The logic designer is deeply involved in questions of board types, wiring rules and physical layout. A whole new area of expert knowledge has developed about the problems of wire, contacts and circuit assembly. It involves the physics, metallurgy, and chemistry of the devices themselves, the radiation and propagation properties of assemblies, and immensely complicated organizational problems in machine layout. It is an area where computer assistance to design can be of great importance. Under the title of *packaging*, which is not a suitably romantic title, a whole new skill is growing up.

The problem of interconnections has also become important in memories where the ringing of a stack structure following application of a drive pulse is a major source of noise, reducing operating margins and limiting speed of operation. It is controlled by the physical arrangement of wires and ground planes.

PERIPHERAL EQUIPMENT

The problems of peripheral equipment (reliability of tape recording, tape and head wear, card jams, print quality, and printer reliability) are old, and antedate the RCA 501. However, we are presently solving these problems at a different performance level. Tape-recording speeds went to 33 kc with the RCA 501, (BIZMAC II was 16.7 kc) and are now going to 120 kc, though the mechanisms have an obvious family resemblance. An enormous amount of work on recording techniques, tape guiding, error control, tape quality, head design, and many other areas has made possible this increase in performance.

Recording techniques have changed substantially. The RCA 501 tape station used pulse recording and the key problems of the day were eliminating crosstalk in heads and obtaining tape with few dropouts. The crosstalk problem is now relatively well controlled. Present stations use non-return-to-zero recording at much higher densities. The key head problem is *gap-scatter*—the variation in the position of effective magnetic gap. Tape quality and dropouts are still very much of a problem, and tape testing and qualifying is an important field.

The RCA 501 printer produced 600 lines/min; now 1000 lines/min is standard. The quality is much improved and the alignment and uniformity of charactors of the 1958 printer would not be acceptable today. We also read and punch cards at 600/min and 200/min, instead of 400/min and 150/min.

THE NEXT FIVE YEARS

With all these changes in performance, the nature of the problems has changed to a lesser degree. The revolution in peripheral equipment is just coming after a phase of consolidation and squeezing the most out of the available techniques. Optical character recognition, mass memories with the capacity of tape and access times below a second, broader use of on-line communications to remote points, and multiple inputs using EDGE and similar equipment are becoming major factors. The next five years are likely to see a radical new and advanced group of peripheral equipments, just as the RCA 501, 301, and 601 were radically different and better than the BIZMAC generation which preceded them. Meanwhile, a major forward step has been taken by the RCA 3301 REALCOM system (see p. 6).

THE NEW RCA 3301
. . . An Advanced, User-Oriented Family of Medium-Priced Computer Systems

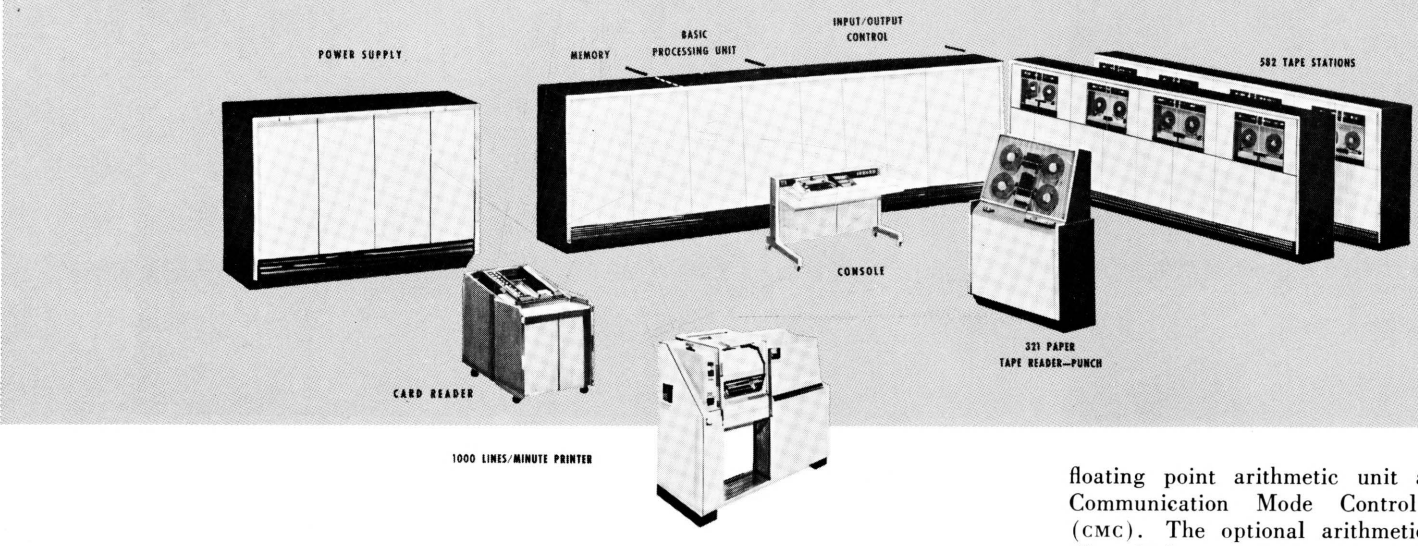


Fig. 1—The prototype complement of the RCA 3301 computer system.

Introduction of the new RCA 3301 Data Processing System (Fig. 1) culminates an intensive engineering effort to produce a distinctly advanced, user-oriented system. The RCA 3301 represents a logical extension of the RCA 501 and RCA 301 with particular emphasis on enhanced performance, greater reliability, and adaptability to both real and non-real-time environments. The RCA 3301 is a two-address, bit-parallel, character serial processor. Through the judicious design of both the command structure and the optional features, the RCA 3301 is effective and efficient in an especially wide range of applications—a goal for a medium-priced system against which all major engineering decisions were measured.

T. A. FRANKS, Ldr.
Design and Development Engineering
Electronic Data Processing, Camden, N.J.

THE "RCA 3301 System" actually refers to an extensive family of systems, because of flexibility incorporated into the basic processing unit. The many possible system configurations allow 1) efficient tailoring of the initial equipment complement to suit existing requirements, and 2) convenient capability to expand and/or reorient an installed RCA 3301 system to meet changing needs.

BASIC SYSTEM AND OPTIONAL FEATURES

Fig. 2 illustrates both the basic RCA 3301 elements and its optional features. These basic elements are:

1) *micromagnetic memory*—replaces most of the hardware registers normally required in the basic processor and operates at an independent read or write cycle time of 250 nsec;
2) *program control and central bus*—contains the main communication artery and the addressing control logic;
3) *logical processing unit*—includes the instruction execution controls and the logical, arithmetic, and comparison processing functions;
4) *main memory*—provision for storing 40,000 seven-bit characters with a complete read-write cycle time of 1.75 μsec;
5) *console and operator's typewriter*—for all operator-system communications; and
6) *input-output control unit*—includes interfaces for connecting various peripheral-device control modules capable of being operated in either of two independent simultaneous modes.

Enhancements to the processing functions of the RCA 3301 include a fixed-floating point arithmetic unit and a Communication Mode Control unit (CMC). The optional arithmetic unit allows higher-speed execution of wired-in *add*, *subtract*, *multiply*, and *divide* instructions with either 8-character fixed-point operands or 10-character floating-point operands. The CMC unit provides the connecting link between the basic processor and low-speed real-time data links. Up to 160 such data links can be handled in an independent "CMC mode" so that normal data processing is unaffected.

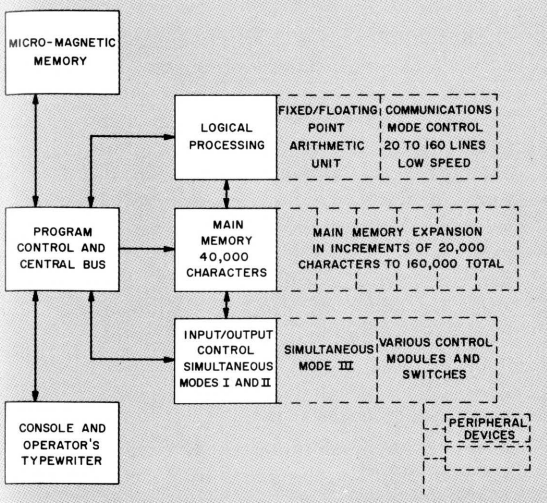
As another option, additional modules of main memory may be added in increments of 20,000 characters up to a total maximum of 160,000 characters.

The remaining optional elements consist of various control modules to link peripheral devices and the processor's input-output control. Another mode of simultaneity is available for on-line operation of up to three functions concurrently—in addition to the normal processing and CMC functions. Typical performances characteristics of these devices are included in Fig. 2.

The characteristics of the RCA 3301 System are compatible with the existing RCA 301 System;[1,2] however, the RCA 3301 provides a faster and larger main memory, higher-speed circuitry, enhanced input-output capabilities including program interrupt, and a more powerful command structure. The internal representation of data is identical with the RCA 301, and the RCA 301 instruction repertoire is a subset of the RCA 3301 instruction repertoire.

In the RCA 3301, the major hardware features include:

1) a compact 200-character micro-



PERIPHERAL DEVICES INCLUDE:

1. Magnetic Tape Stations: alphanumeric character rate options: 33, 66, and 120 kc
2. Random Access Disk Files: total capacity: 528 x 10^6 alphanumeric characters min-max access: 70-120 msec transfer rate: 32 x 10^3 character/sec
3. EAM Cards: reading: 1,500 cards/min punching: 300 cards/min
4. Paper Tape: reading: 1,000 characters/sec punching: 100 characters/sec
5. Line Printers: asynchronous mode (64 printable symbols): 800 lines/min synchronous mode (47 printable symbols): 1,000 lines/min
6. Computer to Computer Exchange Channels
7. High Speed Communications Channels
8. Interrogating Typewriters
9. Custom Real Time Interfaces
10. Real Time Clock
11. Communication Mode Control: capacity: 20, 40, 60, 80, 120, 140, 160 lines data rate: 10,000 characters/sec maximum total

Fig. 2—RCA 3301 system functions and peripheral devices.

T. A. **FRANKS** graduated from Rensselear Polytechnic Institute with a BEE in 1956 and received the MS from Stanford University on a fellowship program in 1957. Since joining RCA in 1956, he has taken part in the development of solid state memory circuitry; was a project engineer on several computer programs; worked on logic design of checkout equipment, and system organization of advanced computers; and was a project engineer on a BMEWS computer installation. Since joining EDP in 1962 he has been the Project Leader coordinating engineering of the RCA 3301 System. Mr. Franks is a registered Professional Engineer-in-Training in New York State, and a member of the IEEE, Tau Beta Pi, Sigma Xi and Eta Kappa Nu.

magnetic memory, used in lieu of flip-flop registers;

2) a program-interrupt feature, permitting efficient use of input-output and error-recovery capabilities;
3) a high-speed main memory, organized in 70-bit words, and available in a basic size of 40,000 characters with optional modules to raise total capacity to 160,000 characters;
4) a flexible input-output bus interface for adding various combinations of control modules; and
5) an advanced circuit and packaging design concept that balances operating speeds and design margins against ease of manufacture.

Basic Processor and Micromagnetic Memory

Fig. 3 shows the basic processing unit and illustrates how the micromagnetic memory has replaced many of the hardware registers normally required in a processor. The extremely fast cycle time is effectively employed for the same tasks normally assigned to flip-flop registers. This memory is significantly less expensive and more compact than the total number of registers it replaces. Thus, it is one of the *major state-of-the art advances* incorporated in the RCA 3301.

The micromagnetic memory consists of 200 seven-bit characters (six bits of information and parity) arranged in 50 locations of four characters each. To achieve the extreme performance required, this memory system used the concept of the RCA microferrite array.[3,4] It is a word-oriented, two-core-per-bit, linear-select configuration with three wires threaded through each core. Two are used for the driving system (one for read, one for write) and one printed wire is used for the digit-sense circuits. The independent read or write cycle time is 250 nsec. This permits reading one micromagnetic memory location into the micromagnetic memory register, or writing from the micromagnetic memory register to one micromagnetic memory location within one time-pulse period. The micromagnetic memory is functionally used for address registers, control registers, various indicators, and as temporary storage during instruction execution and progam-interrupt sequences. All of the contents are addressable by specific instructions.

The equality detection between the micromagnetic register and the main-memory address register is required to determine the termination of address-controlled instructions and input-output service sequences. Fig. 4 illustrates the use (and timing) of the micromagnetic memory during a typical instruction-fetch cycle.

Multilevel Program Interrupt

Another unique feature of the RCA 3301 is the multilevel program interrupt system. This facilitates real-time programming, servicing of multiple input-output devices, error-recovery procedures, program debugging techniques, and compatibility routines for execution of other RCA computer programs.

The three levels of program priority, in order of their priority, are: 1) *real-time interrupt*, 2) *general interrupt*, and 3) *normal processing*. Thus (1) cannot be interrupted, (2) can be only by (1), etc. There are 18 conditions which will cause an interrupt process to take place, of which 5 are designated as real-time and 13 as general interrupt conditions. The mechanization of the interrupt process involves:

1) interrupt sequence (all hardware);
2) interrupt routine (software using hardware indicators and instructions);
3) return after interrupt (an instruction); and
4) program control of interrupt (an instruction);

Step 1, the *interrupt sequence*, is initiated when a bit of the interrupt register is set, the appropriate inhibit-interrupt interlock indicator is clear, and the execution of the current normal processing instruction is completed. The interrupt sequence automatically stores the appropriate registers in standard micromagnetic-memory locations, one set for a general interrupt, another set for a real-time interrupt. The associated inhibit-interrupt indicator is set, i.e. general or real-time. The automatic interrupt sequence then obtains the *jump* address from another standard location in the micromagnetic memory and transfers control to the interrupt routine.

In Step 2, the *interrupt routine* determines which condition(s) of the 18 caused interrupt. As a result of a programmed scanning operation, this routine branches to the appropriate program. Further instructions test the various status conditions for proper recovery. After the interrupt condition has been dealt with, the software interrupt routine scans the interrupt register

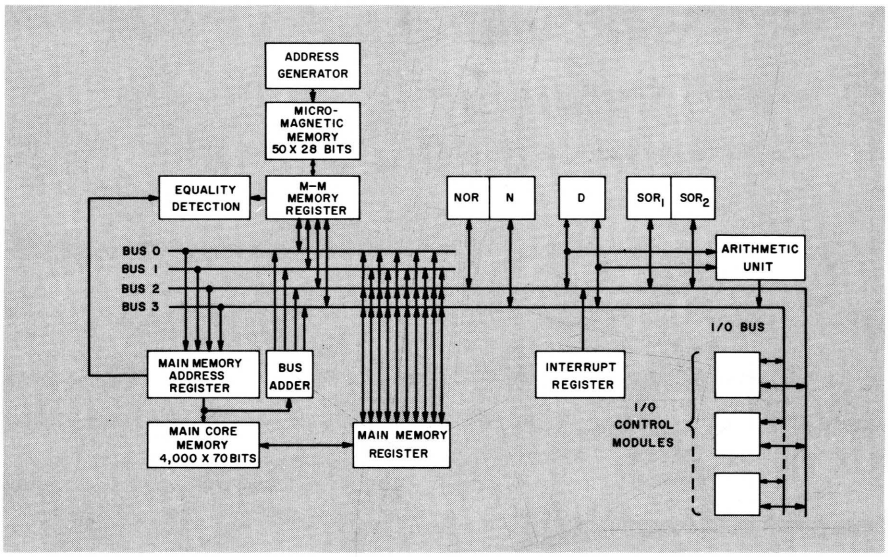


Fig. 3--Basic processing unit.

again to determine if another bit is set. If so, that condition is dealt with. This cycle is repeated until all conditions have been accommodated. Then the interrupt routine exits by means of a *return-after-interrupt* instruction.

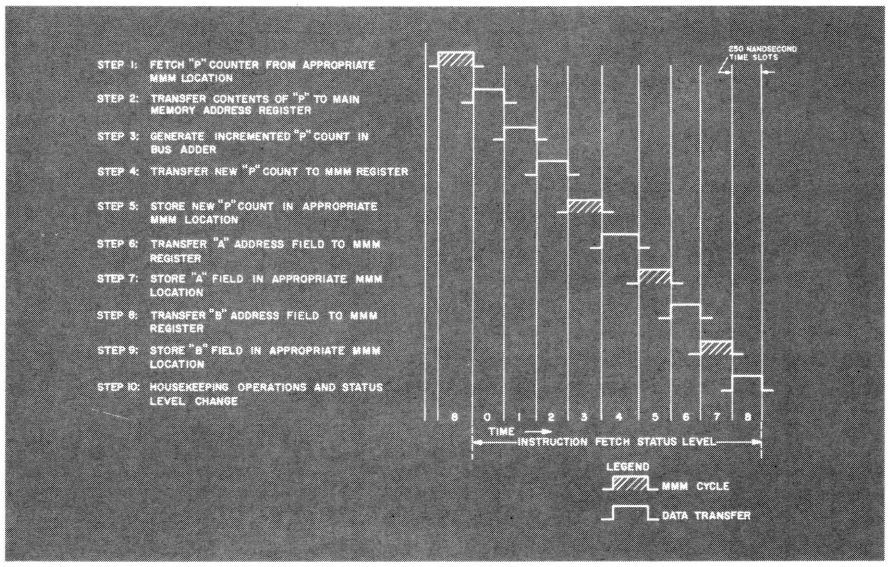
Step 3, *return-after-interrupt* instruction, automatically restores all the register and flip-flop settings that were stored in the micromagnetic memory by the most recent interrupt sequence and clears the appropriate inhibit-interrupt indicators. Then the instruction addressed by the instruction-counter register is fetched and normal processing continues.

Step 4, *program control of interrupt* by the programmer is affected by another instruction which allows the selective setting or clearing of the two inhibit-interrupt indicators.

Main Memory

The RCA 3301 random-access main memory is a magnetic-core design supplied in a basic size of 40,000 alphanumeric (7-bit) characters, with optional 20,000-character modules available for increased capacity (up to a total of 160,000 characters). Each location is individually addressable and can store one character. The main memory cycle is designed for 1.75 μsec and is subdivided by the basic processing unit into seven 250- nsec slots. The machine cycle, however, can be either 1.75 or 2.25 μsec, depending on the amount of control and data manipulation required during a given cycle. The 2.25-μsec cycle consists of nine 250-nsec time pulses, and is required for operations such as instruction fetch or input-output service (Fig. 4).

Fig. 4—Simplified micromagnetic memory example.



The multiplicity of connections shown between the main memory register and the central bus in Fig. 3 are a logic network. This allows the ten character positions of the memory to be interchanged with the four character positions of the bus, thus, providing the selective character-oriented operation of this system. Address incrementing (or decrementing) is performed by the bus adder in order to properly sequence instructions and operands located in the main memory.

Input-Output Devices

A broad range of input-output devices are available for any given RCA 3301. The basic processing unit incorporates the essential control logic necessary for generalized operation of peripheral devices and, by means of an extension to the central bus (Fig. 3), a standardized interface for connection of control modules. Thus, enhancements can be conveniently added to an operating system by simple field modifications. The schematic configuration of the control-module connections are shown in Fig. 5. The system design provides for accommodating up to six input-output control-module racks.

Circuitry and Packaging

The circuit and packaging concepts were selected to allow maximum performance with ease of manufacture and maintenance. The basic high-speed logic circuit is a multi-input diode gate followed by a transistor inverter. A zener diode provides proper voltage offset and charge storage characteristics between the input-diode gate and transistor base (Fig. 6). This circuit accounts for 94% of all the active elements in the basic processing unit. The other 6% are special circuits required for line drivers, line receivers, delay lines, oscillators, etc.

To minimize signal-wire lengths, a compact method of packaging circuits on plug-in module assemblies was chosen. Twenty-four basic plug-in types similar to that shown in Fig. 7 were selected for use in packaging the processor logic. These plug-ins employ vertical submodules to provide additional room for components. Appropriate test points are included. Additional ground contacts are used on the top and bottom edges of the plug-in to minimize ground impedances and increase circuit noise-immunity margins. All external signal wiring is wire-wrapped to further ensure reliability of connections. Fig. 8 shows a typical configuration of plug-ins mounted in the basic-processing and main-memory racks.

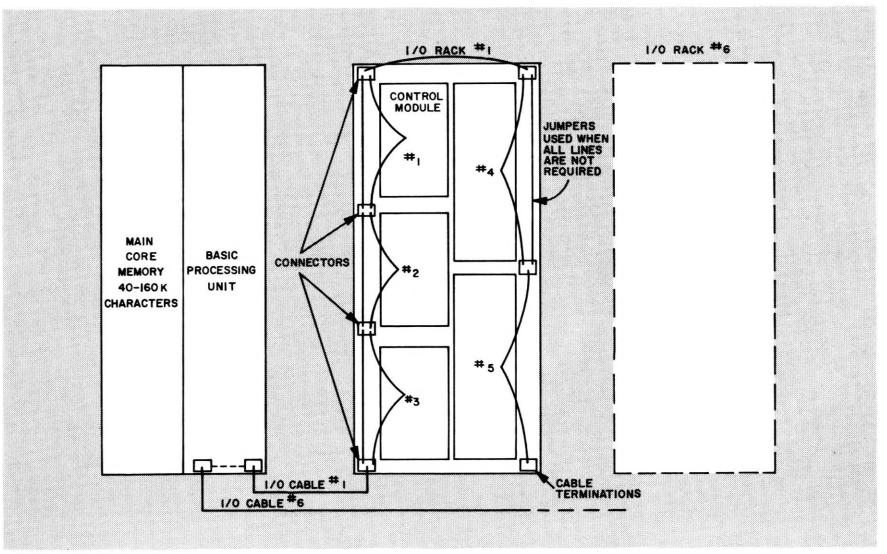


Fig. 5—Control module connections.

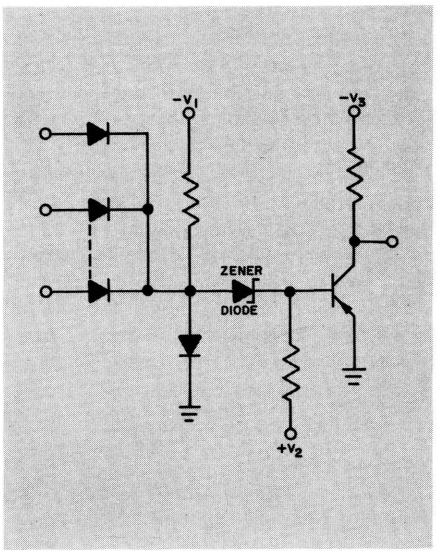


Fig. 6—RCA 3301 basic high-speed logic gate.

PROGRAMMING FEATURES

The RCA 3301 operates by character-oriented, two-address instructions. The instruction format consists of ten characters interpreted as follows:

OP	*N*	*A*	*B*
X	X	XXXX	XXXX

The first character, *OP*, specifies the basic operation to be performed. The second, *N*, indicates a count, a specific symbol, or a device identification number depending on the operation character. The next four specify the first address (*A* address field); the remaining four specify the second address (*B* address).

Indirect addressing is indicated by a bit in the least-significant character of the address. After instruction fetch, if this bit is present in either (or both) the *A* or *B* address field, the processor will automatically replace the contents of the *A* or *B* locations in the micromagnetic memory with the contents of the main-memory location addressed by the previous value of the *A* or *B* field. This process will repeat for as many levels of indirect addressing as necessary until this bit is zero.

Similarly, indexing of either (or both) the *A* or *B* address fields is indicated by two bits of the second least significant address character. There are three address fields, each with an associated increment field and all are located in the micromagnetic memory. Indexing always precedes indirect addressing. Identification of indirect addressing and indexing are indicated by the original instruction, but the effect of indirect addressing occurs on the address formed after indexing.

For descriptive purposes, the instruction repertoire may be classified into four general categories: 1) input-output, 2) data handling, 3) arithmetic, and 4) decision and control. (Typical execution times are shown in Table I.)

Input-output instructions link the processor with the peripheral devices (through the control modules) to position and/or search tapes and disk files, bring data from an input medium into the processor, or send data from the processor to an output medium. Five basic functions are provided that can be executed in any one of the two (or optionally three) simultaneous modes (plus one special instruction for specifying operation of the communications mode control). These functional varieties are: 1) input-output control, 2) read, 3) read reverse (magnetic tape only). 4) write, 5) erase (magnetic

Fig. 7—Typical plug-in unit.

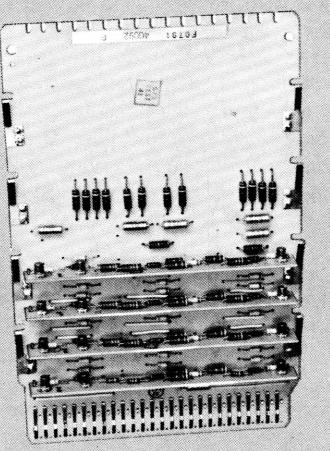


Fig. 8—Basic processing and main-memory racks.

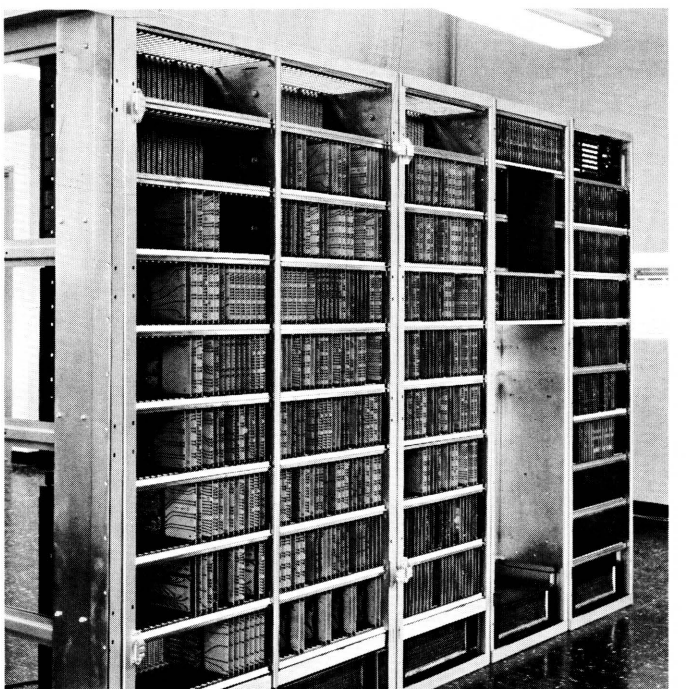


TABLE I—Typical Instruction Execution Times

Instruction	*Time, μsec*
DATA HANDLING:	
translate by table (6-character field)	33.75
transfer data left (6-character field)	23.25
ARITHMETIC:	
data add (6-character operands)	
no end-around carry condition	37.25
with end-round-carry condition	61.75
logical or (6-character operands)	33.75
DECISION AND CONTROL:	
Unconditioned Transfer of Control	2.25
Repeat	5.75
Set register	5.75

Notes: Indirect addressing adds 3.5 μsec for each level of modification. Indexing adds 2.25 μsec for each address modified.

tape only), and 6) communication mode control (one instruction only). Operations such as tape rewind, track select (disk file) and paper advance are initiated by the processor (via an input-output control instruction), but once underway, are completely independent in execution.

The data-handling instructions are nonarithmetic operations for manipulation of data stored in the main memory. The instructions included in this group permit control of data fields by symbol, address, or count. Instructions for edited manipulation of varied fields are also included in this group.

The arithmetic instructions include: 1) four decimal operations, *add*, *subtract*, *multiply*, or *divide*; 2) three operations used to alter the bit configuration of an operand through the use of logical commands; and 3) three operations for arithmetically manipulating four character fields in accordance with the rules of addressing. The decimal instructions operate in accordance with standard arithmetic rules and are designed to handle operands of mutually equal lengths. Three instructions, *logical or*, *logical and*, and *exclusive or* constitute what may be considered as a separate arithmetic category. They can alter the bit configuration of an operand by the employment of a second operand to "mask out," or insert 1 bits. The three address-oriented instructions allow operations of *address add*, *address subtract*, and *address compare* on four-character operands consistent with the progression rules of memory addresses.

The decision and control instructions influence the sequence of operation. Four instructions enable the programmer access to registers of machine indicators directly and one instruction provides conditional control; that is, it chooses a path according to selected conditions. Another instruction either halts or causes a program interrupt in the processor's operation. A *repeat* command enables the execution of loops a designated number of times. The *compare* instruction enables the programmer to determine the relative magnitude of two operands of equal length. The last two instructions in this group enable program control and restoration of machine conditions after interrupt sequences, as described in a previous section.

In addition to this command structure, the overall system efficiency is further improved by:

1) built-in and programmed accuracy controls;
2) automatic storage of the contents of various working program-control locations in the micromagnetic memory;
3) character addressability providing completely variable data organization; and
4) machine code covering the full range of numerics, alphabetics, and special symbols.

The accuracy-control philosophy of the RCA 3301 System includes not only error detection, but also error recovery. When an error occurs and is detected by wired-in parity and invalid-code checking circuits, program control is transferred (by the interrupt feature) to an executive error-recovery routine. Appropriate actions can be taken at this time. Transient processor malfunctions, as well as input-output-equipment errors, can be handled by these techniques.

Facilities for automatically storing various program-control address fields are included in the basic processing unit. These are called STA, STP, and STPr.

STA automatically occurs at the conclusion of selected instructions. In STA, the final contents of the *A* field located in micro-magnetic memory are automatically stored in standard main memory locations. This permits the subsequent use of the final *A* field contents and is a convenient programming technique to eliminate memory searching time.

STP occurs whenever program control is to be transferred out of immediate sequence; STP automatically stores the contents of the instruction-counter field in another set of standard main-memory locations at the conclusion of those instructions that would cause transfers of control. The stored address is the address of the instruction that would have been executed if the transfer of control had not taken place.

STPr similarly automatically stores in standard main-memory locations the instruction address immediately following the repeat instruction, and is used for looping control reference during a repeat sequence.

CONCLUSION

This paper has described the most significant engineering features incorporated in the RCA 3301 Data Processing System. Obviously, many design details and novel techniques have been either presented very generally or not at all due to space limitations. Readers interested in further information should consult the EDP Marketing Department, Cherry Hill, New Jersey.

ACKNOWLEDGEMENTS

The author acknowledges the intensive efforts expended by the entire RCA 3301 engineering design team on this project. These efforts extended from establishing the initial system concepts, during 1962, to final manufacturing release during 1963. As of the time of publication of this paper, the prototype system will have been in hot tests for two months, and complete operational capability is anticipated during October 1963. The successful conclusion of this project within the original performance, schedule, and cost goals represents a significant achievement for the design team. In particular, the author wishes to acknowledge the efforts of: G. R. Gaschnig, W. F. Glass, B. I. Kessler, J. E. Linnell, J. J. O'Donnell, P. H. Reynolds, G. D. Smoliar, G. J. Waas, and R. H. Yen.

BIBLIOGRAPHY

1. H. Kleinberg, "The RCA 301: Flexibility at Low Cost—An Engineering Challenge"
2. A. Turecki, "The RCA 301 Scientific Computer"
3. Dr. H. Lemaire, "New Techniques for Ferrite Nanosecond Memories"
4. D. F. Joseph, "The NS1—A New 400-nsec Microferrite Memory System"

Fig. 1a — RCA 601 control console.

THE RCA 601 COMPUTER SYSTEM

Design of the RCA 601, largest of RCA's product-line computers, involved solution of major problems in circuit packaging and wiring techniques occasioned by the complexity and speed of the machine. The first RCA 601 System was delivered to the New Jersey Bell Telephone Company and became operational in December 1962. The success of solutions to the design challenges is marked by the fact that machine "up time" has considerably surpassed original expectations. For another user, a second system was recently delivered, and three more are scheduled for 1963. The RCA 601 features many powerful programming and operating features, and has a basic high-speed memory capacity of 8,192 words of 56 bits each which can be expanded by addition of up to twelve memory stacks of 2,048 words each. Data readout from the memory is in less than 0.9 μsec, and the complete memory cycle is 1.5 μsec.

J. J. WORTHINGTON, Mgr.*

RCA 601 Project, Electronic Data Processing

Camden, N. J.

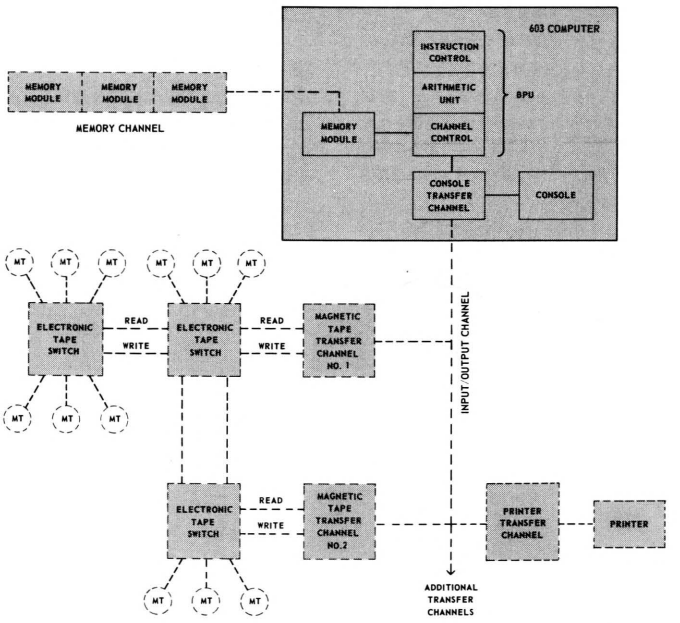


Fig. 1b — RCA 601 System, illustrating expandability.

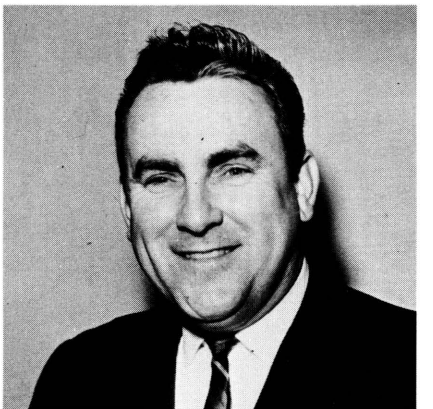
THE RCA 601 is a variable-address computer with an instruction complement that can handle both word-oriented and variable-length-data problems. The computer incorporates an extremely powerful address-modification scheme which allows indirect addressing, many levels of modification, and the ability to increment the modifiers. Another important feature of the RCA 601 is its ability to have automatic interrupts for servicing of input-output equipment or console demand functions. Errors also cause interrupts, rather than machine stoppage, giving a programmer the ability to utilize programming rollback features or diagnostic routines.

BASIC SYSTEM

Fig. 1 illustrates system expandability. The basic computer system consists of a console, a basic processing unit, a high-speed memory (8,192 words of 56 bits each), a power supply, and a console

* In association with J. J. O'Donnel, E. Gloates, B. Freedman, and W. Gibbeson of Electronic Data Processing.

JOHN J. WORTHINGTON studied Electrical Engineering at Oklahoma A & M College and Rittenhouse College (Philadelphia) prior to joining RCA Service Co. in 1947. He was recalled to active duty with the U.S. Navy in 1950. After leaving the U.S. Navy he was RCA aviation equipment engineering's representative at Eglin AFB during the operational evaluation testing of the ARC-21 high-frequency Airborne Communications equipment; and in 1953 was transferred to Engineering Products Division to supervise production testing of the ARC-21. In 1955 he was named superintendent BIZMAC Test and Inspection; and in 1958 he was put in charge of all Data Processing Test and Inspection. In 1959, he transferred to RCA International and was one of five Americans who started RCA Sweden, AB, in Stockholm. He was Manager, Engineering Services, and his responsibilities included sales, installation planning and maintenance of EDP equipment. While there, he sold the second RCA 301 System, installed an RCA 501 Data Center, installed an RCA 501 System in a Swedish bank, and formed an engineering group to handle installation, 60-cycle to 50-cycle specifications, and equipment maintenance. In 1961, he returned to Camden and was assigned as Project Manager, RCA 601 Project; this was expanded to include Engineering responsibility as well as Project responsibility in June, 1962. As Engineering Manager, he supervised the shipping and installation of the first RCA 601 System to New Jersey Bell Telephone Co., October 31, 1962, and of the second RCA 601 System to N. J. Public Service Electric and Gas Co. on January 31, 1963. He is responsible for system performance at these two sites as well as for testing, shipping and installing the remaining systems.



transfer channel. The basic processor contains hardware necessary to mechanize instructions, an arithmetic unit capable of doing word additions, and controls necessary to link the associated units. The arithmetic unit also provides a data handling capability (shifting, masking, merging) of all the various character lengths. Up to 12 additional stacks (2048 words each) of high-speed memory modules can be added to the basic computer.

The logic within the computer is asynchronous. Therefore, the subclocks of the machine are self-timed, and the worst-case delays need not be considered for all paths. During a data transmission, for example, the contents of the register receiving data are compared with those of the transmitting register to determine when the transmission is complete. This comparison is also used to provide a high degree of accuracy control. Each subclock—for example data transfer, high-speed memory, or arithmetic unit, controls a portion of the computer's logic. Each is started individually and operates independently, except for necessary interlocks at points of timing intersection. Because of this type of logical construction, overlapped operation is possible—individual subclocks can operate concurrently, even though they may be under the control of different steps in the execution of an instruction. Since there is no need to wait for each step to conclude before starting the next, and because each step is self timed, considerable computer time is saved.

The console (Fig. 1a) is divided into two major areas:

1) The *operator area* which contains such features as numeric displays of registers or memory contents, a digital keyboard for entering information into the system, a paper tape reader and a flexowriter which may be used for input as well as output.
2) The *supplementary indicator area* which houses the indicators necessary to display many registers and flip-flops in order to facilitate trouble shooting and thereby minimize downtime.

Another major feature of the system is its input-output handling capability. As Fig. 1 shows, there is an expandable input-output channel. Transfer channels are attached via this link to the computer, so that additional input-output devices may communicate with the high-speed memory. Each transfer channel not only adds the ability to interchange information with more peripheral devices, but also adds degrees of simultaneity so that up to 16 devices may be operating concurrently with normal processing. Therefore, it is possible to have eight *read's*, eight *write's* and the computer operating together. The system can handle up to 48 tape stations at speeds of 33-, 66- and 120-kc through electronic switches which allow more than one transfer channel to communicate with each station. The console also communicates with the basic processor through the console transfer channel and the input-output channel. Many functions of the console can therefore be handled while the program is operating. For example, it is possible to read out of or write to the memory from the console while the machine is running a program.

HIGH-SPEED MEMORY

The high-speed memory is an asynchronous unit. When running at maximum speed, the time for a complete cycle (read out the information and write it back into the memory) is 1.5 μsec. Data is read out onto the bus in less than 0.9 μsec. Because it is asynchronous, the memory can perform part of a cycle, stop while the processor makes changes, then continue on with the next part of the cycle instead of having to perform a complete cycle for each part.

Close cooperation between RCA Laboratories, Princeton, the SC & M Needham Materials Operation and the RCA 601 project personnel, produced the fastest large-memory system built by RCA.

The stack is designed so that only two wires, at right angles to each other, pass through the core. One wire is an address-drive line which carries both read and write pulses. The other wire is a digit-sense line which carries the digit pulse during the write part of the cycle and the core output during the read part of the cycle. Coincidence of the write-current and digit-current pulses causes the core to switch to a *1* state. The core will return to a *0* state if the digit write pulse is absent when the write pulse is present.

Address (Drive-Line) System

Fig. 2 illustrates the method of selection of a particular drive line, and how only one wire is used for both read and write pulses.

The emitters of the current switches are connected to the output of a common driver which is used as a constant-current source of the read and write pulses. One write and one read current switch which have their collectors tied together through diodes to a current bus using 32 lines, can be selected. At the bottom (grounded) end of the drive line, one write and one read voltage switch can be selected. Their collectors are also tied together through diodes; however, there is a read and a write diode associated with each drive line to prevent sneak current paths (e.g., path *A, B, C, D, E*). The voltage switch boards, with all their diodes, plug directly into the stack.

Digit-Sensing System

A simplified diagram of the digit-sense system is shown in Fig. 3. It illustrates how only one wire is used to carry both the digit write pulse during writing and to sense the core output during read. The design of the stack is such that the digit-sense winding crosses the address (drive) lines at right angles.

PROGRAMMING

Inasmuch as the overall RCA 601 design philosophy called for a tape-to-tape system (i.e., all peripheral operations are performed via an RCA 301 computer) the RCA 601 software system was designed around a *magnetic-tape library* concept. Because of the complexity of the RCA 601 system, the programming package provided by RCA consists of four primary areas: the *executive system*, the *file-control processor*, the *sort-merge*, and the *RCA 601 assembly system*.

Fig. 2 — High-speed memory drive system.

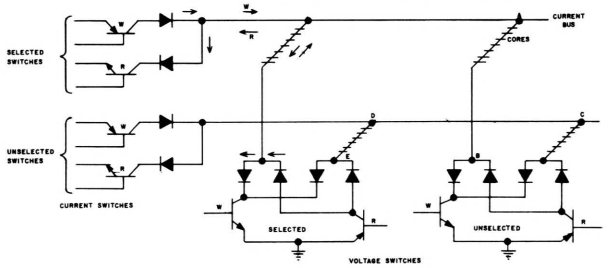
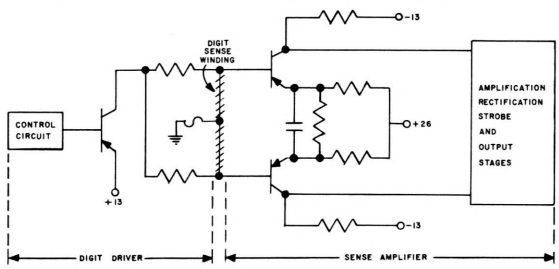


Fig. 3 — High-speed memory digit-sense system.



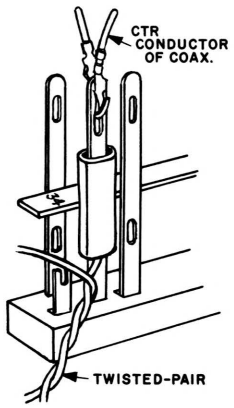


Fig. 4 — RCA 601 metallic clip connector.

In order to properly service the program library tape, the executive system was developed to control program loading and operation. One of the outstanding features of this system is the console demand facility which allows the computer operator to automatically interrupt the program in operation to: take memory or tape prints, discontinue the current program, continue with the next program in a prestored list, reassign input-output devices, and perform other functions common to the operation of a large scale computer system.

The file-control processor automatically directs the simultaneous input and output operations of the RCA 601. With it, the user can easily make use of the flexibility of the RCA 601 in the area of tape processing. The file control processor insures that all available input areas are filled and that data is made available to the user when required by the program. It also performs appropriate tape-label checks and controls input-output error recovery as well as many other functions required for the efficient utilization of the RCA 601 as a data-processing system.

The RCA 601 sort-merge package consists of two powerful generalized sort and merge routines which automatically tailor themselves to best fit the desired application as expressed through a series of parameters entered via the console paper tape reader or prestored on program library tape.

The RCA 601 assembly system links the entire package. In addition to providing the normal features of a good assembly system, such as symbolic representations for all machine instructions and features, subroutines and symbolic memory allocation, the RCA 601 assembly system provides a series of macroinstructions that automatically supply program linkage with the executive system and the file-control processor. Through these instructions, the program may instruct the executive system and file-control processor to perform such functions as calling in additional segments of the program, typing a message on the monitor printer, reading or writing magnetic tape, or even calling in another program to replace itself.

RACK DESIGN

Mechanical design challenges of the RCA 601 racks were created to a large degree by the need to minimize interconnections between racks—and reduce effects of noise and crosstalk. The use of somewhat larger racks and a high-density packaging of the electronic circuits resulted from these new standards of compactness. The 44-inch racks were designed to accommodate 650 plug-in circuits, and the 22-inch racks to house 325 plug-ins.

The basic processor, power supply, and high-speed memory are housed in the 44-inch racks and the remaining units (the transfer channels) occupy racks 22 inches wide. Cooling air is circulated in each rack by centrifugal fans capable of supplying air at 350 cfm against a back pressure of 0.5 inches of water. If the air flow is impaired, an alarm is activated by a protective device consisting of a set of air vanes and switches.

Cabling and Rack Wiring

Peripheral device and power cables enter the rack from the bottom section of the blower assembly. The signal cables (coax wired) connect between racks through side connectors. Power cables are of a new design which provides maximum decoupling. This cable is made up of layers of thin flat conductors employing the necessary circular-mil areas to carry the required current. The new power-cable design has very low inductance due to a favorable ratio of surface area to cross-sectional area.

Cables are sandwiched together—a voltage conductor, a ground conductor, a voltage conductor, etc. Conductors are insulated from each other by using a high-K material. This combination of a small-inductance cable and a very large distributed capacitance is ideally suited for supplying step loads, by assuring a minimum voltage drop and maximum HF filtering.

A number of special techniques were developed to permit the innerconnection of large numbers of logical elements, prevent excessive build-up of wire in certain areas, and reduce the noise due to common impedance. These techniques included a specially developed coax of very small diameter, solder clips which relieved the strain on the center conductor of this coax, multi-level terminals on the connector (Fig. 4), and a special grounding system isolated from the frame (Fig. 5).

The RCA 601 wiring schedule was prepared by an RCA 501 computer utilizing 1) plug-in location and pin number of the source signal from the logic diagram, 2) plug-in location and pin number of the destination from the logic diagram, and 3) the wiring parameters. In order to minimize crosstalk, the following wiring parameters were used: 1) trees less than 6 inches use single wire, 2) trees between 6 inches and 4 feet require twisted-pair wire, and 3) all trees over 4 feet require coaxial cable.

Module Boards

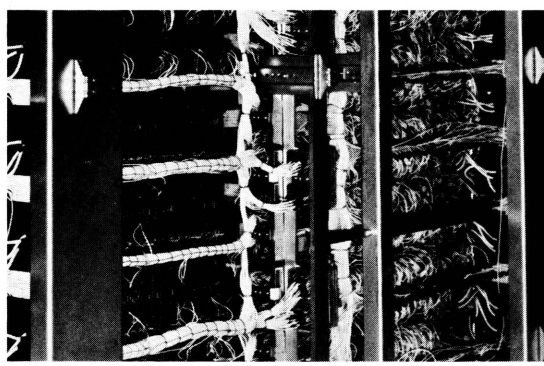
To package the large number of transistors in an RCA 601 rack, it was found that the packing density of the RCA 301 module board was insufficient. For example, the RCA 601 required an average of 14 to 16 (with a maximum of 24) transistors per module board while the RCA 301 averaged only 8.8 transistors per module board (with a maximum of 18). The number of pins best suited to these logic requirements was determined to be 35.

The final pin-connector boards retain the basic features of the RCA 301 and RCA 501 designs, but with additional pins. Fig. 4 illustrates a portion of the connector and the metallic clip presently used on the plug-in to provide suitable board connections.

PLUG-IN-CIRCUIT PACKAGING

The RCA 601 system required a special plug-in-circuit design. The goal was to provide a special-purpose board with greater packaging density. For example, approximately 9500 transistors on 650 plug-ins, were to be packaged in a single 44-inch rack. The final RCA 601 design consisted of plug-in units made up of mother boards and submodules. The submodules serve a dual purpose: 1) to free the main board area for signal and interconnections and 2) to provide maximum exposure of the board-mounted

Fig. 5 — RCA 601 backwiring.



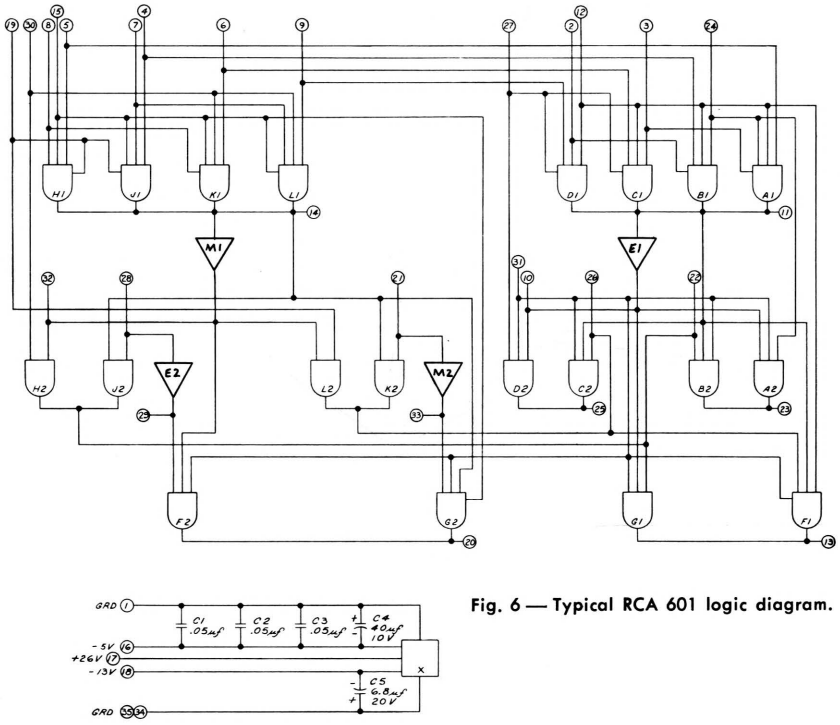


Fig. 6 — Typical RCA 601 logic diagram.

components to the cooling air. Each submodule in the logic and control sections contains two circuits and is easily accessible for repairs and rework. Whenever possible, specialized logic functions are combined on one board. Fig. 6 illustrates the large amount of logic circuitry mounted on the sample plug-in of Fig. 7.

In previous module packaging design, the handling of the input diodes and the mounting of the resistors and capacitors caused problems. Thus, for the RCA 601, the design and development of a new diode package and of an integrated-assembly *printed electronic component* were completed. The printed component contains the required resistor and capacitor elements. The diode packages contain either two or three RCA diodes in a TO-5 transistor case; this highly reliable unit was developed by the Semiconductor and Materials Division to meet RCA 601 specifications.

The final printed-electronic-component package design was to contain the four resistors and a capacitor required for the logic and power gates. Performance specifications called for elements meeting or exceeding the characteristics of standard components. Cost was limited to approximately that of assembling standard components.

The complete RCA 601 plug-in board consists of a filter, a termination board, submodule elements, and the etched circuit pattern. The termination board supplies a resistive termination for any signal pin on the mother board (Fig. 7). The end load on a tree is terminated and discharged so that the line voltage falls as the source decreases. On the average, eight trees terminate on every plug-in; terminating resistors have clips so that connections may be made from any signal pin to the —5-volt return. The ohmic value of these resistors is determined by knowledge of the tree source (high-speed power, high-speed transistor) and the number of loads being driven.

Because of the special-purpose configuration, there are many types of boards in the RCA 601 system; however, some degree of standardization was possible in the submodule area. One basic configuration was used for approximately 80 percent of the submodules in the system. By varying the components, different layouts were accomplished.

Submodules are subjected to strict tests. Each basic submodule must satisfy a detailed performance specification—signal levels, circuit delays, rise times, fall times, etc. After assembly of the submodules to the plug-in boards, a logic test is performed. (No actual performance measurements are made at the plug-in level.) The plug-in and submodule test procedures are incorporated into the drawing structure for future use.

SPARES

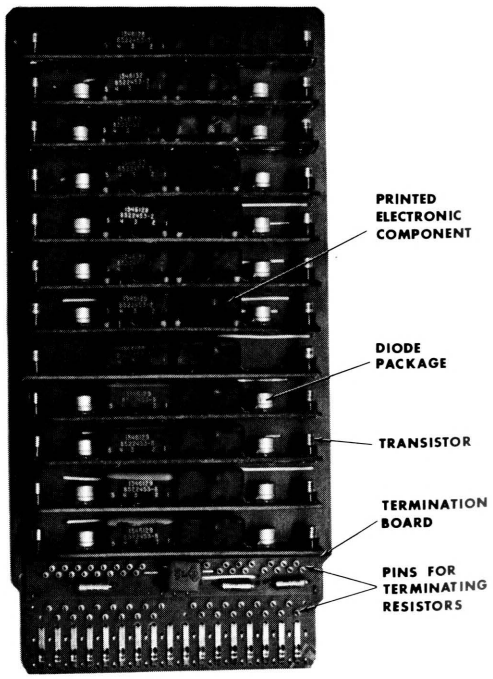
A new approach was developed for the RCA 601 to reduce the spares needed to maintain the system. Studies promised substantial savings if an effective method of plugging in the submodules could be developed. To accomplish this and at the same time stay within stringent height requirements was the main objective (0.8-inch spacing of plug-in units in the rack). Fig. 8 illustrates the pin-and-socket approach which finally proved satisfactory. Cost estimates indicate that this method and a proper sampling plan have reduced the cost of spare plug-ins by more than 50 percent.

CONCLUSION

Experience provided by the RCA 601 program indicates that the limits of using present packaging and wiring techniques on large-scale computers is rapidly being approached. As circuit speeds increase and computers become more complex, wire lengths and termination become more critical. Noise pick-up through capacitive and inductive coupling, and common impedance become problems of increasing concern.

These design problems, however, were successfully solved in the RCA 601. The first system (see front-cover picture) was delivered to the New Jersey Bell Telephone Company and became operational on Dec. 1, 1962. Since then, use of the equipment has rapidly grown from eight to over fourteen hours per day. The machine "up-time" has considerably surpassed original expectations. A second system was recently turned over to Public Service Electric and Gas Company of New Jersey. Completion of three more RCA 601 systems is scheduled for 1963.

Fig. 7—Typical RCA 601 plug-in circuit module.



THE RCA 301 SCIENTIFIC COMPUTER

A. TURECKI
Palm Beach Operations
Electronic Data Processing
West Palm Beach, Florida

A new high-speed arithmetic unit is now available for the RCA 301 Computer to add scientific capability to the basic RCA 301 System. With a new total of 58 instructions, a user can apply the RCA 301 efficiently to both technical and business problems. The new arithmetic unit can be added to any existing RCA 301 system, and a variety of subroutines and programs are made available by RCA for scientific applications.

THE *high-speed arithmetic unit*, latest enhancement of the RCA 301 System, adds scientific capability to the low-cost, high-capacity RCA 301 Computer. With this capability, a customer can acquire greater efficiency for his rental dollars and schedule time for both technical and business assignments.

With the addition of the high-speed arithmetic unit, the present RCA 301 instruction repertoire is increased from 48 to 58 instructions, and the execution of arithmetic instructions is accomplished at least 100 times faster. The 10 new instructions are: *Fixed and floating add, subtract, multiply* and *divide, shift register*, and *store register*. The data format of a fixed-point operand consists of eight digits, and that of a floating-point operand consists of ten digits. The least-significant two digits are the exponent, and the remaining eight digits are the fraction of the operand. The sign and the overflow are indicated by the standard RCA 301 zone bits.

The instruction format is consistent with the present RCA 301 format, the only variation being the interpretation of the *N* digit. The two most significant bits of the *N* digit indicate that the operands *a* and *b* are taken either from the *A* and *B* address locations or from the accumulator. The next bit controls whether the result is stored in the accumulator only or is also transferred to the *A* address location. The last three bits of the *N* digit control the selection of the three index fields for the *A* and *B* address modification which is carried out during staticizing. Also, the modified tally instruction permits the incrementing or decrementing of the index fields on the consecutive runs through a program loop. This adds a very powerful tool to the RCA 301 Scientific Computer.

To be consistent with the original RCA 301 modular philosophy, the arithmetic unit can be added to *any* RCA 301 System in the field. The estimated time of the field installation, which includes some necessary modifications to the 301 basic processor unit, is 40 hours.

RCA makes available a variety of scientific subroutines for matrix operations, linear programming, statistical analysis, differential equations, and curve-fitting, plus scientific and Bell Interpreter Systems, UMAC (an algebraic compiler employing FORTRAN mathematical statements), and RCA 301 FORTRAN. Also available are some special industrial programs, such as lens design, bus scheduling, electrical load distribution, power requirements analysis, and others.

Current marketing reports indicate that the RCA 301 Scientific Computer *has no serious competitors in its price range.*

FUNCTIONAL SPECIFICATIONS

After the modifications required for the addition of the high-speed arithmetic unit, the basic processor unit models 304 and 305 become Models 354 and 355, respectively. The arithmetic unit provides the hardware for high-speed fixed- and floating-point arithmetic and associated address modifications. The unit occupies 12 rows (a half rack) of the control module rack.

The arithmetic unit consists of an addressable upper accumulator and lower accumulator, two nonaddressable arithmetic registers, and a high-speed parallel adder. Standard core memory locations are assigned for address modification. They are the three Index fields used during staticizing and the three increment fields used by the modified tally instruction.

Any quantity falling within the range of 10^{-99} to 10^{+99} can be represented in the floating-point format to the eight-digit precision. All the floating-point results are automatically normalized and rounded. To make double-precision programs possible, the results of fixed point arithmetic are not rounded.

Since the operands of both fixed and floating arithmetic are word-oriented, they must always begin at even locations in the core memory. The floating-point zero is represented by the zero fraction with the exponent of −99.

All the arithmetic-unit errors stopping the computer are indicated by the ARIE alarm on the processor console. There are six different conditions that will ac-

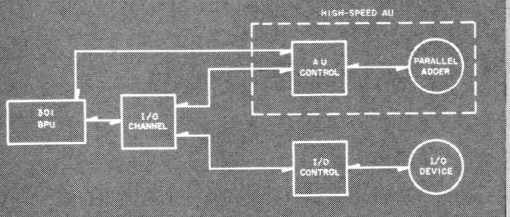


Fig. 1—High-speed AU interface.

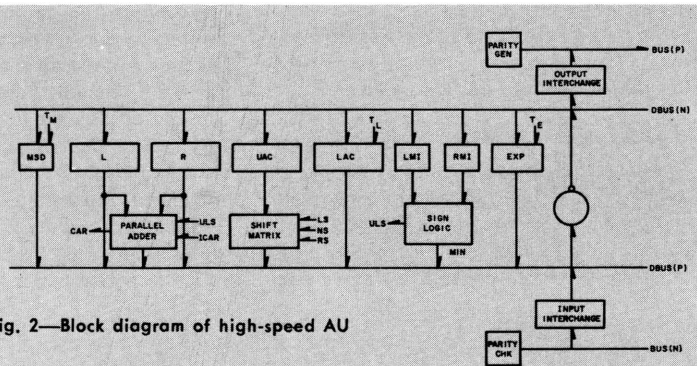


Fig. 2—Block diagram of high-speed AU

tuate the ARIE alarm: 1) *data parity error*, 2) *non-normalized operand*, 3) *dividend greater than divisor*, 4) *zero divisor*, 5) *zero operand with the exponent not equal to −99*, and 6) *modified address exceeding the core-memory capacity*.

The data characters in the arithmetic unit are numeric (straight binary-coded decimal) and therefore the 2^4 and 2^5 zone bits are ignored with the exception of the permissible 2^4 bit in the most-significant digit and the 2^5 bit in the least-significant digit of a quantity, signifying an overflow and negative sign, respectively. Both zone bits can be transferred from the arithmetic unit to the core memory, but only the zone bit signifying the negative sign can be transferred from the memory to the Arithmetic Unit.

Depending on the sign of the result, a corresponding PRI *(previous result indicator)* is set at the completion of any arithmetic instruction. SCAR, signifying an overflow, may be set only in *fixed add/subtract* and in any floating instruction indicating that the resultant exponent exceeds the range of ±99.

FUNCTIONAL DESCRIPTION

Intercommunication with 301 Basic Processor

The similarity between the high-speed arithmetic unit and any input-output control module (besides their common modular structure) can better be seen if the parallel adder is considered to be the arithmetic unit's peripheral device (Fig. 1). Some of the signals, not available from the input-output channel, are obtained through additional lines. These are codes of the new arithmetic unit instructions, new status levels, and special controls.

While the average input-output device speeds are far below the present processor speeds, a good balance was achieved between the 301 basic processor and the parallel adder in the high-speed arithmetic unit. Any further speeding up of the adder would not be justified by the performance-versus-cost relationship.

Although the high-speed arithmetic unit has its own eight-clock-pulse generator, it runs synchronously with the basic processor, BPU. The purpose of the arithmetic-unit clock is to subdivide status levels into more timing slots that are available from the 301 basic processor. This increases the number of performable functions per status level.

The High-Speed Arithmetic Unit

There are eight data buses (DBUS0 to DBUS7) connected, via read-in and read-out gates, to the corresponding digits of all the arithmetic-unit registers. The data from the core memory is transferred one dyad at a time via BUS2 and BUS3 of the input-output channel and the input interchange to the consecutive DBUS's starting from the least significant pair. The input interchange is switched by means of control signals IN1 to IN4 which are generated by a counter running during the loading of operands to the arithmetic unit. Similarly, the output interchange controlled by the signals OUT1 to OUT4 switches the consecutive pairs of DBUS's to the core memory via BUS2 and BUS3. There is a provision made to transfer all the four digits of *A* or *B* address at one time. Moreover, if the address has any zone bits in its most significant digit (signifying either 20,000 or 40,000 character memory) the input interchange generates an equivalent fifth digit. After performing an address modification, the resultant fifth digit is combined with the address most significant digit in the output interchange.

The parity of the incoming data is checked at the input interchange. All the DBUS's consist of four lines with the exception of DBUS0 and DBUS6 which also have 2^4 lines and DBUS7 which has a 2^5 line. After the execution of an arithmetic instruction, the four-bit numeric code is transferred back to the seven-bit code at the output interchange.

There are eight registers in the arithmetic unit (Fig. 2). Starting from the left, the registers are:

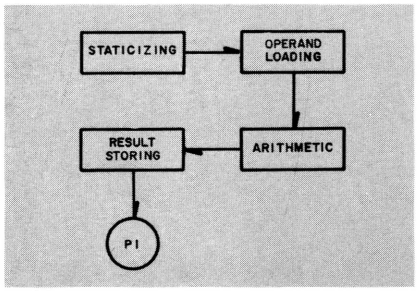


Fig. 3—General fixed point arithmetic flow chart.

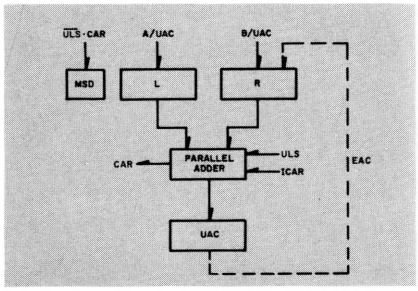


Fig. 4—*Add/sub* data flow chart.

Fig. 3—Fixed-point arithmetic instructions, general functional flow chart. The RCA 301 Computer, being a two-address system, requires *Ai* for the transfer of the result to the memory. This is accomplished by loading the *B* operand into the arithmetic unit and transferring *Ai* to the *B* register. This register is then used in the final step of the arithmetic instructions, which is loading of the result into memory.

Fig. 4—Add-subtract, fixed point. After loading the L and R registers, a minimum of 3 μsec is allowed for the adder to propagate carry. The output of the adder is then transferred to UAC. The result has an overflow if ULS·CAR. This condition is used for triggering up MSD. If MSD = 0, the 2^4 bit is set in the most significant digit of UAC. The condition ULS·CAR initiates the EAC procedure. In this case, L and R registers are reset, UAC→R, and $0 + \bar{R} \rightarrow$ UAC, thus correcting the result in the manner explained above.

Fig. 5—Multiply, fixed point. The loading of operands proceeds in a similar manner as in *fixed add-subtract*, but the destination of operands is different. The multiplicand *a* is transferred into R and the multiplier *b* into LAC. The multiplication algorithm consists of two cycles; a consecutive addition (generating partial product) and a right coupled shift. The consecutive addition proceeds as follows:

L + R is transferred to UAC.
If there is a carry, MSD is triggered up.
LAC7 (l.s.d. of LAC) is triggered down.
UAC is transferred to L and the cycle is repeated until LAC7 = 0.
When LAC7 = 0, the shift cycle performs a coupled right shift of MSD, L, and LAC, and triggers the digit counter down.
If the digit counter XC ≠ 8 and LAC7 ≠ 0, the first cycle is selected again.

After eight shifts, the multiplication is completed and the final product, consisting of 15 or 16 digits, is stored in UAC and LAC. For single precision, only the part in UAC is used. For double precision, the parts in UAC and LAC are used.

Fig. 6—Divide, fixed-point. The loading of operands is the same as in *fixed add-subtract*. The divident *a* is transferred into L and the divisor *b* into R. The division algorithm consists of two cycles; a consecutive subtraction (generating partial remainder) and a left coupled shift. The procedure is as follows:

A left shift of L is executed, transferring the m.s.d. of L into MSD. This step enables multiple precision.
The subtraction cycle is then started with L—R being transferred to UAC.
If there is no carry, MSD is triggered down.
If MSD ≠ 15, LAC7 is triggered up and UAC is transferred to L.
The cycle is then repeated.
When MSD = 15 (which means the difference L—R is negative), the left coupled shift of MSD, L, and LAC is performed, the digit counter is trig-

MSD—*most significant digit*. This is a single-digit register triggerable up or down. As its name implies, it is the most significant digit of sums, partial products, and partial remainders.

L and R—*left operand* and *right operand*. Both registers are eight-digit registers and their contents provide the input to the parallel adder.

UAC—*upper accumulator*. This register stores results and is associated with the shift matrix.

LAC—*lower accumulator*. This register, with its least-significant digit LAC7 (triggerable up or down), is used for generating the quotient in *divide* and exhausting the multiplier in *multiply* instructions. It is also used to store the second part of the product in *fixed multiply* and remainder in *fixed divide*.

LMI and RMI—These are single-bit registers used for storing the signs of operands. In association with the sign logic, they generate the ULS *(unlike signs)* signal and MIN (the sign of the results).

EXP—*Exponent Register*. This register is a two-digit algebraic up or down counter.

Parallel Adder and Sign Logic

The eight-digit parallel adder operates on straight binary-coded decimal digits.

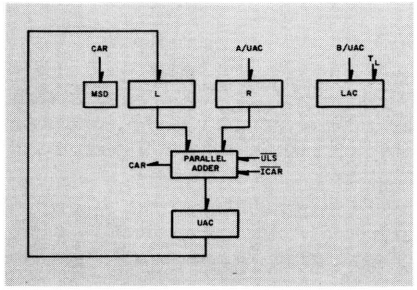


Fig. 5—*Multiply* data flow chart.

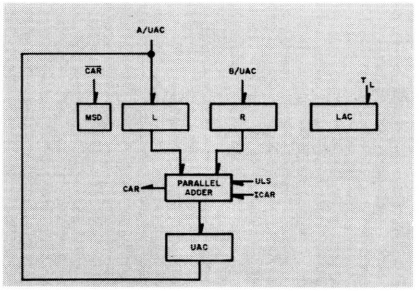


Fig. 6—*Divide* data flow chart.

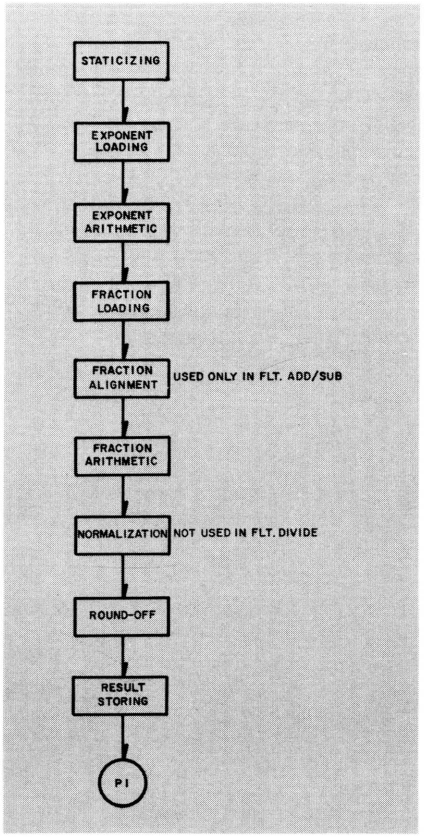


Fig. 7—General floating point arithmetic flow chart.

gered up, and the subtraction cycle is selected unconditionally.

The execution is terminated in the subtraction cycle when MSD = 15 and XC = 7.

Fig. 7—Floating-point arithmetic instructions. Floating-point arithmetic uses only the algorithms previously described and is similar to fixed-point arithmetic with some additional operations. The additional operations are exponent arithmetic, alignment of fractions (used in *floating add-subtract*, only), normalization of results, and round-off. The exponent arithmetic is a straight addition or subtraction depending on the instruction. The normalization is a left or right shift of the result with the corresponding correction of the exponent. The latter is done by triggering the exponent register algebraically up or down. The round-off is carried out on the basis of the least significant digit outside the eight digit precision. The general flow charts contained herein do not indicate any special paths. The charts are of a general nature because, depending on the instruction options and the result itself, some of the functions may be redundant. These procedures are bypassed, whenever applicable and justified by the involved cost, by means of special case status level selections.

Add or Subtract, floating point. The exponent arithmetic carried out three independent functions, and they are as follows: 1) it stores the algebraically larger exponent; 2) computes the exponent difference; and 3) determines which fraction is to be shifted to the right for the alignment purposes. During the alignment, the shifted-off digits from L and R are stored in LAC in a straight or complemented form, depending on the ULS signal state. This is done in order to provide the information for rounding-off. The fraction arithmetic is, in principle, identical to the *fixed add-subtract* arithmetic. During the normalization, the result is either shifted right (if MSD $\neq$ 0) or shifted left (if there are any zeros in the most significant portion of the result and MSD = 0). The rounding-off is performed by adding 1 to the result if LAC0$\geq$5. The solution of the case of an overflow produced during the round-off is left for the readers.

Multiply, floating point. The exponent arithmetic generates an algebraic sum of the operand exponents. The fraction arithmetic is identical to the one of the fixed multiply. As the final product may consist of 15 or 16 digits, the normalization, if any, produces one coupled left shift of UAC and LAC. In the rounding-off, as before, the result is incremented by 1 if LAC0$\geq$5.

Divide, floating point. The exponent arithmetic produces an algebraic difference of the operand exponents. In the fraction arithmetic, if $a<b$, the dividend is shifted one place to the right, thus cancelling the need for the normalization of the result. The following consecutive subtractions and shifts are the same as in the *fixed divide*. For the rounding-off purposes, the ninth cycle of the consecutive subtractions is carried out, but the ninth quotient digit is not allowed to exceed five. If the digit is five, the rounding-off takes place, otherwise, the execution of the division is terminated.

It can be shown that the excess-three code has hardly any advantage over the straight binary code in decimal adders. The excess-three self-complementing code requires some decoding at the adder output and, as a result, is more expensive than the complementing of the straight code. The complementing circuit is associated with the R input only.

The most fundamental operation in the arithmetic unit is addition. The adder performs subtractions using the following algorithm.

Case 1: $a>b$

$$a-b \equiv a+\tilde{b} =$$

$$a+(10^n-b) = 10^n+(a+b)$$

Instead of subtracting b, add its ten's-complement $\tilde{b}$. The presence of 10^n, being a carry of n-digit precision, indicates that the result is correct. In this case, the sign of the result is taken to be the same as one of the operand a.

Case 2: $a<b$

$$a-b \equiv a+\tilde{b} = d$$

As there is no carry, the following procedure, known as EAC *(end around carry)* is initated:

$$0+\tilde{d} = 0+[10^n-(a+\tilde{b})] =$$

$$0+10^n-[a+(10^n-b)] = -a+b$$

In this case, the sign of the result is taken as the opposite of the operand a.

In the adder, the complementing of b operand is initiated by the ULS signal. It is carried out by the complementing of each digit to 9 and then with the generated ICAR *(initial carry)* the result is effectively increased by 1:

$$\tilde{b} = 99999999 - b_6b_7b_2b_3b_4b_5b_6b_7 +$$

$$1 = 10^8 - b$$

The sign logic is mechanized to follow the fundamental rules of the arithmetic.

The parallel adder is also used for the arithmetic on exponents in the floating-point instruction. It was planned originally to provide a special two-digit adder for the exponent arithmetic, but the additional cost would not justify a negligible saving of time.

INSTRUCTIONS

Figs. 3 through 7 and their accompanying captions and text describe both the fixed-point and floating-point arithmetic instructions of the RCA 301 Scientific Computer.

ANATOLE TURECKI received his BSc degree in Engineering from the University of London in 1949. From 1949 to 1956 he was employed by Harris Lebus, Ltd. (London) as a Research Engineer; was employed by Tasma Pty. (Sydney) as a Development Engineer; and was a Lecturer in the Radio School of the Royal Melbourne Technical College. He joined RCA in 1956 in the Broadcast Engineering Division where he designed and tested TV transmitting antennas. He transferred to RCA Electronic Data Processing in 1960 where he assisted in design and testing of the RCA 301 System. He completed all course requirements for his MSEE at the University of Pennsylvania in 1960, and his thesis, "Ternary Numerical System in Digital Computers", is presently being submitted. He is a Member of IEEE and holds a U. S. Patent.

SUMMARY

This article has given a general picture of the high-speed arithmetic unit to readers who are familiar with the RCA 301 System.[1] Some of the minor details, often far from being easy problems for an elegant solution, were omitted. Readers interested in more-complete details of the high-speed arithmetic unit are welcome to request, through the normal channels, the functional specifications for Processor Models 354 and 355, status level flow charts, and logic drawings.

BIBLIOGRAPHY

1. H. Kleinberg, "The RCA 301: Flexibility at Low Cost—An Engineering Challenge"

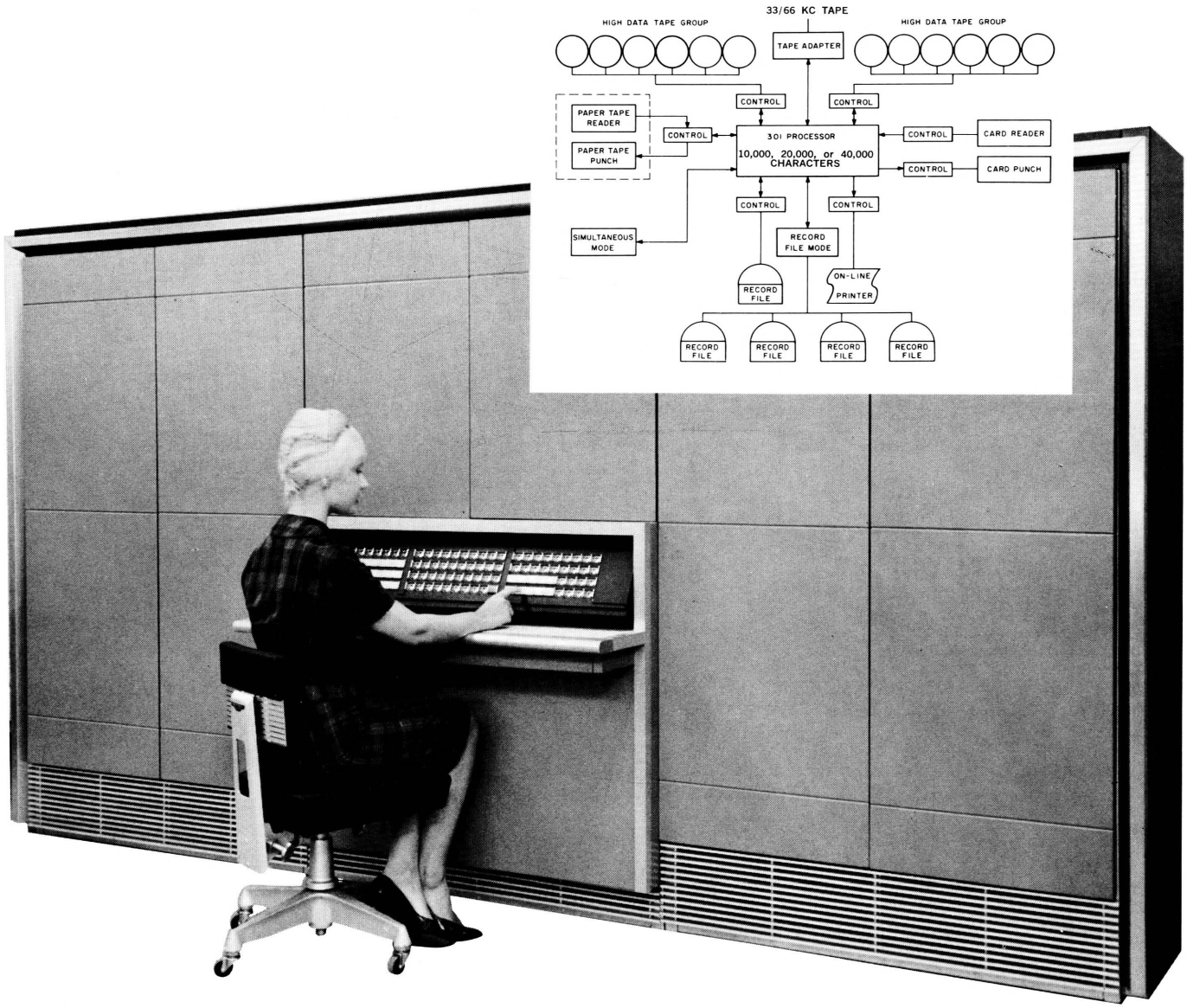


Fig. 1—The RCA 301 basic computer (center cabinet) with modular power-supply and control cabinets at left and right. Inset: The RCA 301 data-processing system.

THE RCA 301 . . . Flexibility at Low Cost – An Engineering Challenge

by **H. KLEINBERG, Mgr.**
Product Line Engineering,
Electronic Data Processing
Camden, New Jersey

MANY PARALLELS CAN be drawn between two apparently very different industries—automobiles and computers—over the course of the past few years. In both of them, a trend toward bigger, more powerful, more expensive machines was the rule, until very recently a reverse toward compact models has set in. While the analogy must not be overworked, one other point of similarity can be noted: Just as a large and a small car perform basically the same function with different load capacities, so the large and small business computers are faced with problems of equal complexity, but with different data volumes.

SMALL-COMPUTER DESIGN PROBLEMS

In the case of computers, small machines with the required capability have not been practical in the past, largely because of the lack of depth in supporting technology. Core memories, for example, presented a fixed cost to the designer of the small computer that invariably led him to the use of drum memories, with a consequent loss of speed by a factor of 1,000. Magnetic-tape units, highly reliable diodes, transistors, and almost every other system component presented a similar problem. It was only when these costs were lowered, through the volume production of large and medium systems, that the

successful design of powerful, low-cost computing systems became possible.

Another problem encountered in the successful design of a small computer lies in the production phase. In order to have a system that markets at a reasonable price, volume production of the same units is necessary. Customizing and tailor-making of systems cannot be economically supported. On the other hand, each customer for a data-processing system has a situation that differs in certain aspects from every other user of the same system. Some users rely on paper tape for their input, others rely on punched cards; some users have a large amount of internal computation, others have a large volume of output printing. Thus, in order to reach a wide market, the system must be flexible. This, of course, works in direct contradiction to the production requirements of a large volume of identical units. It is in this field that the RCA 301 system offers a most novel solution. Modularity, enabling customizing of systems to the specified need of the user, has been made compatible with the volume production required. The RCA 301 system is illustrated in the diagram of Fig. 1.

MODULARITY

This modularity has been achieved through the concept of separate control panels for each piece of input-output equipment. From Fig. 2, it can be seen that the basic computer includes the memory and control hardware necessary for the execution of instructions, but that it includes no input-output equipment. It is, in effect, a central brain with no means of communication with the outside world. This communication is provided by means of peripheral equipment (printer, card punch, etc.) through a set of control panels and associated cables. Each of these panels ties to the central processor and is specifically designed to work with a peripheral unit. Thus, a control panel is associated with the printer, a different control panel with the card reader, a third control panel with the paper-tape reader, and so on. These panels can be attached to the basic computer by means of plug connectors. They can be produced in any volume required and stocked separately. When a customer is to receive shipment of an RCA 301 system, he receives a basic computer, peripheral equipment as required, and a control panel for each of the pieces of input-output equipment that he receives.

INPUT-OUTPUT EQUIPMENT

The choice of input-output equipment for a small system presents a whole set of problems by itself. Again, a balance between cost and performance, between customer requirements and production volume must be found. The past emphasis on speed can no longer be the major determinant in specifying equipment. Two new input-output units have been specifically designed for the RCA 301 system—the *record file* and the *hi-data tape group* (see diagram, Fig. 1).

The record file is a storage device holding information on a set of magnetic recording disks. Each disk as used in the system can store 36,000 characters of information made up of 9,000 characters on each of two bands on both sides of the disk. Access time to information in the record file averages 4¼ seconds. This device is not truly a random-access unit as normally defined. It may not be applicable to a larger, much more powerful system such as the RCA 601 or the RCA 501, but it has specific uses and capabilities within the framework of the RCA 301 system such as program storage and file storage under certain conditions.

The hi-data tape group is a set of six magnetic-tape transports, each of which can handle information at the rate of 10,000 characters per second. A set of common electronics is housed with the six tape drives, permitting access to any one of them at a time. Here again, economy has been the byword. Tape start time, including switching of the desired tape drive, is 30 milliseconds—again, a figure that would hardly be impressive within the framework of a large-scale system, but providing a unit which has a place and a function in the RCA 301 system.

Other input-output equipment is, of course, available. Card reading and card punching are provided, with the reading speed of 600 cards/minute and the punching speed of 100 cards/minute. Paper tape may be utilized, with reading and punching speeds of 100 characters/second. A line printer, providing hard copy at 1,000 lines/minute, 120 characters/line may also be included. The RCA 301 may work with other tape stations of the RCA product line at the rates of 33,333 or 66,667 characters/second. Special input-output equipment is available as required, and here again, the merits of a control-panel system show themselves. In one of the early systems the customer (in this case a large bank) required the ability to handle through the RCA 301 a magnetic-ink check-reading and -sorting device. There was no requirement to modify or redesign the basic computer for this function, since all tie-in to this system is handled by a specially-designed control panel.

BASIC COMPUTER

The basic computer (Fig. 2) is arbitrarily divided into the *high-speed memory* and *program control* units, although both are housed in a single rack (Fig. 3).

The memory comes in 10,000, 20,000 or 40,000 characters. This choice is made at the time of purchase, and corresponds roughly to selecting a 4-passenger or 6-passenger car. The memory is organized so that two characters (14 bits) may be read in or out in a 7-microsecond cycle, which includes the regeneration time.

The program control unit consists of the necessary register for addressing the memory, holding information to be read out of or stored in the memory, and the registers necessary for execution of the instructions. The instructions are two-address, made up of ten RCA 301 characters. The *A* and *B* addresses normally specify the memory locations that are to be used in the instructions; they require four characters each. One character, the *N* character, contains information that usually pertains to the number of characters that are to be handled in this instruction. This is one of the ways in which

H. KLEINBERG received the Bachelor of Applied Science, Engineering Physics, in 1951 from the University of Toronto, whereupon he joined the Ferranti Electrical Ltd. in Toronto as a research engineer in the digital data-handling equipment field. Mr. Kleinberg joined RCA in 1953, where he worked on electronic controls for input-output tape, and film units and circuit standards. He spent three years on logic design, manufacture and test, of the first RCA computer system. He has since been active in the design and development of RCA's transistorized data-processing systems, such as the RCA 501 and RCA 301. In 1959, he was promoted to Mgr., Computer Devices Engineering, and in 1960 was named Mgr., Product Line Engineering. Mr. Kleinberg is a Member of the IRE.

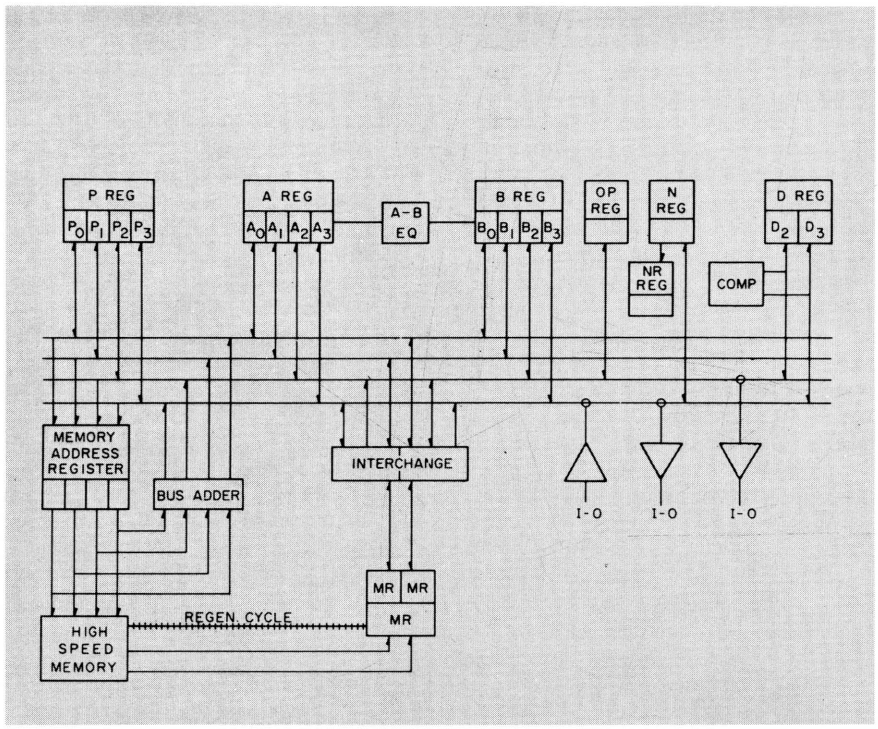
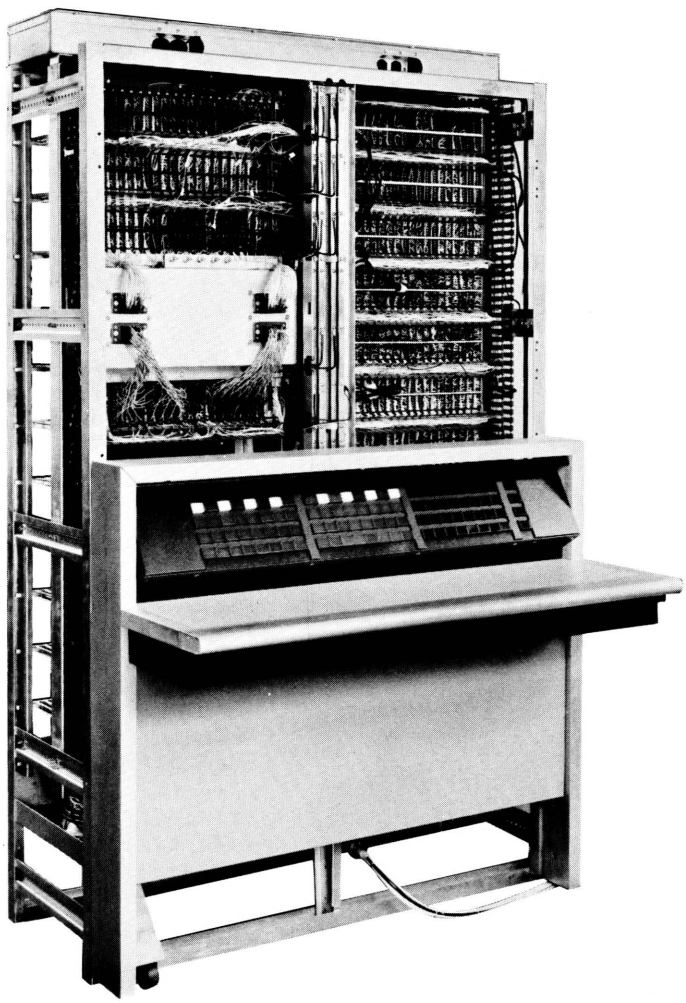


Fig. 2—RCA 301 basic computer.

Fig. 3—RCA 301 basic computer rack and console.



variable word length is handled within the computer. The tenth character of the instruction specifies the operation to be executed by the computer. Instructions are stored sequentially in the high-speed memory, each requiring ten consecutive memory locations.

Much of the packaging for the computer has been carried over from hardware used in the RCA 501 System. To achieve substantial cost reduction, a separate console was eliminated and the control console is now attached to the central computer rack (Figs. 1 and 3). This console swings aside for maintenance and testing of the computer rack itself.

CIRCUITRY

The choice of circuitry for a low-cost system such as the RCA 301 is one of the most basic and difficult portions of the design. Reliability is, of course, the prime consideration. To this has been added increased speed (compare 7-microsecond memory cycle with a 15-microsecond memory cycle in the RCA 501), the requirement for low cost, and the requirement for compactness of the total system. It may be seen that the design problem facing the circuit engineers was indeed formidable.

It should be pointed out that minimum system cost does not always follow from minimum cost of components used in the circuitry. Considerable negotiation and discussion between logic designers and circuit designers is necessary to optimize the final choice of circuits. For example, it may be found that a circuit costing 25 percent more will provide 75 percent more ability to drive loads in the form of other logic circuits. Application of such a new circuit would be worthwhile if enough "chains" of logic are used. Here, too, the problem of volume production versus custom design reappears. The logic designer would like many special-purpose logic plug-ins, while the circuit designer wants a small number of general types that may be produced in volume. A balance must be found that optimizes the over-all system cost.

SUMMARY

As RCA's entry into the low-cost computer field, the RCA 301 System presented a difficult challenge to the design engineer. At every step, the divergent needs of production, user, logic designer, circuit designer, and systems engineer had to be weighed, and the proper compromises made. The result is a powerful, competitively priced system that has received wide acceptance by customers.

ENSURING SUCCESSFUL COMPUTER INSTALLATIONS

. . . the training of Computer Service Representatives

The rapid growth of RCA in the electronic-data-processing field has necessitated, on the part of the RCA Service Company, the development of a new breed — the RCA "computer service representative" (CSR). To install, maintain, and trouble-shoot complex electronic and electromechanical equipment, the CSR must be familiar with all related areas, such as computer programming and operation. Hundreds of such trained technical personnel are needed to install and maintain an ever-increasing number of RCA 301, 501, and 601 data-processing systems — which has led to the comprehensive program of personnel selection and training described herein.

M. H. MACDOUGALL, Mgr.
EDPS Maintenance Training
RCA Service Company
Cherry Hill, N.J.

MOST OF THE training of the RCA computer service representatives (CSR's) takes place at the Electronic Data Processing Services (EDPS) Training Center of the RCA Service Co., located at Cherry Hill, New Jersey. The modern training facilities at the Center include classrooms and laboratories capable of handling an enrollment of several-hundred CSR trainees. The Center has its own RCA 301 and RCA 501 Systems. The equipment complement for each system is carefully selected to be representative of a typical customer's installation. Oscilloscopes and other test equipment, gauges and jigs, hand tools, and operating supplies such as magnetic tape and EAM cards are all duplicates of the equipment and supplies to be used by the CSR when assigned to a customer's installation.

WHO IS TRAINED

There are many classes held simultaneously at the Training Center on a variety of RCA data-processing equipments. Most of the technical personnel attending these classes are new employees who, after completing their training, will be assigned to a forthcoming computer installation. Also, there are a number of experienced field personnel who have returned to the Training Center to receive additional training, either in the form of advanced courses on certain pieces of equipment, or on an entirely new system. The majority of RCA's trained CSR's are capable of maintaining several types of RCA data-processing systems.

HOW TRAINING PROGRAMS ARE PLANNED AND REVIEWED

The development of a maintenance training program for a data-processing system begins many months *before* the delivery of the first system. As soon as preliminary engineering drawings are available, a team of instructors begins working with the design engineering group. By combining the knowledge obtained in working alongside the design group with their experience in developing data-processing training programs, these instructors prepare a preliminary specification for a maintenance training program. These specifications, together with the sales forecast for the system and anticipated delivery dates, provide the basis for estimating manpower requirements and hiring dates.

The training team continues working with the design engineers to develop the preliminary course outlines, lesson plans, and training notes. As the design work proceeds, and engineering drawings become final, the team completes the initial preparation of various sections of the program. These sections then enter the review phase.

Copies of the documentation for these sections such as the preliminary training manuals and notes, outlines, timing charts, and signal source lists, are submitted to the appropriate design activity for technical review. Concurrently, decisions are made to determine the mode of presentation, the types and numbers of training aids required, the method of reproduction and quantities of training manuals required, and the number of lecture and laboratory training hours required for effective presentation.

Close cooperation must be maintained with the engineering activity of RCA Electronic Data Processing throughout the design, development, and prototype testing of a new design. The training team benefits from the experience gained from engineering and manufacturing personnel engaged in the testing of prototype. By examining engineering data compiled during testing, noting areas requiring more precise and more frequent maintenance than others, the training team can modify previously prepared course material by giving proper emphasis to sensitive maintenance areas.

The maintenance training and engineering support capability must be achieved by the EDPS activity prior to the delivery of the first system to a customer (peripheral equipment released at later dates may require a different approach). Therefore, at periodic intervals during the design and development of the computer system, the initial delivery schedule, is reviewed, and estimates of manning requirements and hiring dates prepared. These estimates are reviewed

Fig. 1—A classroom training session on the RCA 301 computer; the Instructor is Don Schulman.

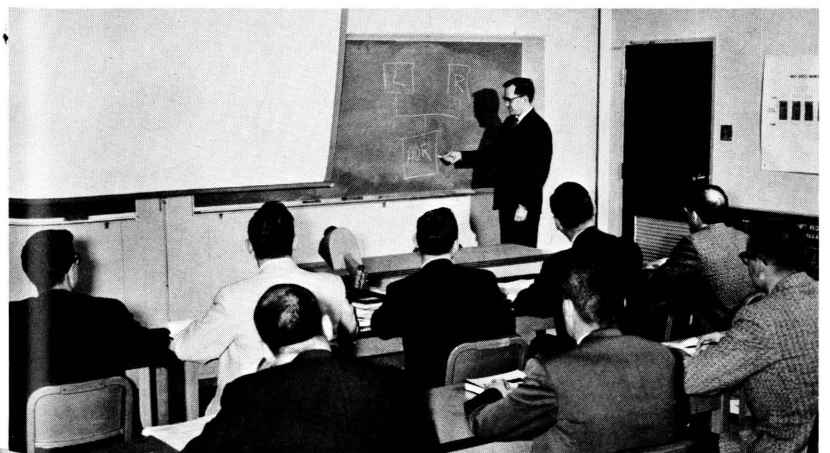
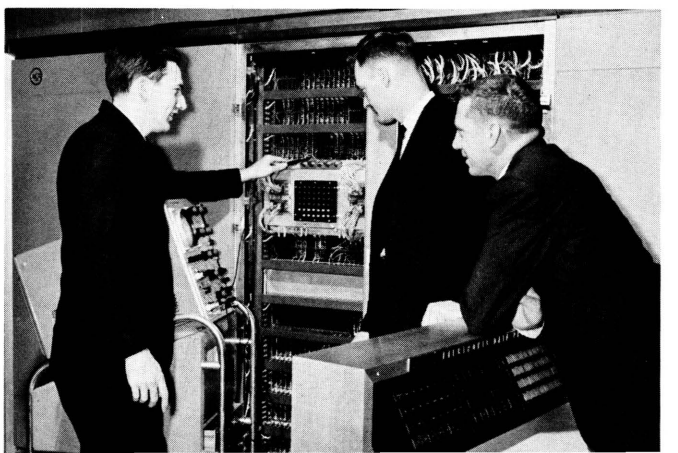


Fig. 2—Trainees shown working on high-speed memory alignment of the RCA 301 during a Training Laboratory session: l. to r. are Instructor, W. Harrison, Kurt Koster and Jim O'Brien.



by the EDPS Personnel activity. An appropriate lead time of one to three months for recruiting is allowed.

SELECTION OF TRAINEES

The selection of CSR candidates is a painstaking process. For every class of ten to fifteen trainees hired, hundreds of applicants may be interviewed and tested. Most candidates are selected from the top graduates of a technical school, such as the RCA Institute. A battery of selection tests is administered to preselected applicants. Tests include aptitude, general intelligence, abstract reasoning, logical reasoning, and mechanical comprehension, as well as technical knowledge. As a result of the close screening in the interviewing and selection process, the number of later dropouts is minimized. Although the present selection of techniques are comparatively successful, the EDPS Personnel and Training activities are continuing in an effort to further define the aptitudes and traits of successful students.

A TYPICAL TRAINING CYCLE

Most of the CSR trainees are presently being trained in the maintenance of the RCA 301 system. This typical training program is 17 weeks in length, and is divided between lecture classes providing a detailed knowledge of system functioning, trouble-shooting, and laboratory training on the equipment at the Training Center. Here, the trainee's skill of utilizing his knowledge to isolate and repair equipment malfunctions is developed.

Courses Studied

The training program begins with a basic course to provide background for the detailed study of the logical design of the system. Some of the topics included in this course are *numbering systems, binary arithmetic operation, transistor and diode switching circuits, logical elements,* and *magnetic core memories.*

The next course covers the *design philosophy, programming, operation,* and *maintenance* of the computer. Other courses cover the various *input-output buffers* and the *peripheral input-output devices* used in the 301 system.

Included in the weeks the trainee spends working with the equipment at the Training Center are courses on the *installation* of a 301 system, *preventive maintenance procedures,* the *assembly and disassembly* of the electromechanical peripheral equipment. Also covered are *test and maintenance programs* in performing preventive maintenance and isolating equipment malfunctions. Equipment malfunctions are induced by inserting defective components into the system. The types of failures simulated are based on a study of equipment failures in the field. However, primary emphasis is not made on specific problems and their solutions, but on the development of the trainee's approach to the solution of the problem.

Upon completion of the training at the Center, most CSR trainees undergo a period of on-the-job training. During this period, they work closely with experienced personnel in maintaining and trouble-shooting data processing equipment. As their experience increases, the degree of assistance required diminishes. This final phase concludes training on the computer. Then, the CSR is considered ready for an assignment.

OTHER TRAINING AIDS

The lecture and laboratory courses which compose the 301 training program are supplemented by a variety of training aids and manuals. Training manuals, encompassing thousands of pages, are utilized not only as texts in the various courses, but also as reference manuals by field personnel. Such manuals are developed by instructors for use in their training courses and are sufficiently comprehensive for experienced field personnel to use in "on-site" refresher courses. Engineering, programming, and methods personnel also find the manuals to be of considerable value. Contents of manuals are translated into French and Japanese.

OTHER TRAINING-CENTER COURSES

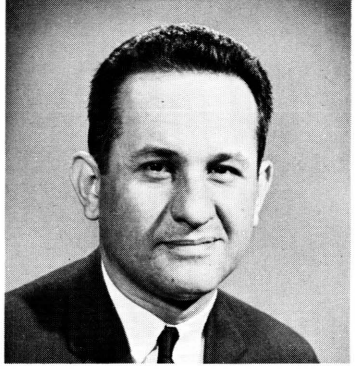
The courses used in the RCA 301 training program are only a few of those conducted at the Training Center. At the present time, courses are conducted on more than fifty individual items of data-processing equipment, ranging in length from a few hours study for a specific course on paper-tape handling equipment to several months on the more complex equipment. Specific courses on data-processing equipment, classes in operating and programming, and introductory seminars in data processing and computer logic are conducted for other activities within RCA. Courses of this type are tailored to fit the needs of the personnel enrolled. As a result of the sale of RCA data processing systems to other countries, the Training Center has conducted training programs for engineers and technicians from Sweden, England, France, Japan, South Africa, and Australia. Upon completion of their training, these people have returned to their native countries to establish their own maintenance, training, and engineering activities.

CONCLUSION

As the types and quantities of RCA data processing equipment continue to multiply, the increasingly diversified training requirements will necessitate further development and utilization of new and more-effective training techniques. The feasibility of using programmed learning techniques in various segments of the training program is being studied. In teaching computer operation, an experimental program using the computer itself as a teaching machine has been developed. Research is also underway in the use of simulators and simulation techniques which would minimize the training time required on certain data-processing equipment.

Fig. 3—Some of the RCA 301 computer system training manuals developed by the Instructors. Manuals are used for classroom instruction and for future reference in the field.

MYRON H. MAC DOUGALL graduated from the RCA Institutes Advanced Technology course in 1959, joining RCA as a Computing Equipment Engineer. In 1960, he was promoted to Manager, Laboratory Facilities, EDPS Training, and in 1961 was promoted to his present position of Manager, EDPS Training. He is a member of the IEEE and the ACM.



RALPH E. MONTIJO, JR. received his BSEE from the University of Arizona in 1952. In 1952, he joined RCA's Specialized Training Program. He was then assigned to the Computing Systems Engineering Section where he participated in the development and design of the BIZMAC. In 1956, he was promoted to engineering management; and in 1957 directed the West Coast Engineering Laboratory of RCA's Electronic Data Processing Division. In 1957-61, he directed systems engineering and development effort on: data systems for airborne vehicle flight testing, random access memories, magnetic tape stations, operations research, and input-output studies. He also made major contributions to the USAF's DATACOM network. In 1961, he was promoted to his present position of Manager, Systems Engineering for Electronic Data Processing. Mr. Montijo has authored several technical papers, and is a member of the IEEE, the ACM, and the AMA.

EDGE . . .
Electronic Data Gathering Equipment

EDGE provides a low-cost, reliable method for remote input of data to computers that is competitive with manual input. The high-volume information supplied directly to the computer enables business decisions on more timely information. This extension of the input-output capability of computers foretells a day when systems like EDGE, coupled with volatile displays (not unlike present military control centers), may be a major tool for decision making.

R. E. MONTIJO, Jr. Mgr.
Systems Engineering
Electronic Data Processing, Camden, N.J.

ALONG WITH the rapid assumption by the computer of its rightful place in our operating systems has come the recognition of its limitations. And, its primary limitation in the area of input-output has been magnified as the internal computing speed and magnetic-tape data rate have risen. Inherent in this problem is the physical separation between the locations where data is generated and subsequently used, and the location of the processing element.

The EDGE (Electronic Data Gathering Equipment) system was developed to implement the tasks of collecting, code converting, and reducing the disparity between present and desired input-output data rates for typical computers.

Simplicity, flexibility, and reliability are fundamental to meet stringent cost-performance requirements for a system such as EDGE. The functions which EDGE automates are far less sophisticated than those performed by the computer. In the kind of locations where EDGE might be installed (for example, a factory work area), EDGE must compete with proven manual methods that are well accepted by local personnel. And because it is intended that EDGE be utilized by such local personnel (not computer personnel), installation and operation must be simple and inexpensive; performance must be consistent and reliable; and the integrity of data handled by the system must be above suspicion.

A two-year program of study, analysis, evaluation, and optimization of customer requirements was performed upon several hundred combinations of available techniques. As a result, EDGE provides the means for reading input documents, accepting data directly from humans, converting the data to a machine-usable code, transmitting it to a distant central point, and recording it for subsequent processing.

When EDGE is combined with other elements of RCA data-communications and data-processing equipment, the entire data gathering, processing, and output cycle may be performed *on-line*; the term *on-line* is used here to describe the processes of both data gathering and computation. (Source-data recording by EDGE is always performed "on-line" in the sense that EDGE input devices are directly linked to the EDGE central recording equipment.) The term *off-line* EDGE (as used here) refers to the setup wherein the EDGE central recording equipment produces a punched paper tape for subsequent use as a computer input. The term *on-line* EDGE refers to a direct link between EDGE and the computer, without intermediate punched paper tape recording. In the EDGE system itself, intermediate storage devices are eliminated, since data is transmitted from its input point(s) through a passive network of telephone lines and line switching devices to the EDGE central recorder. Storage elements are thus only the input media and the ouput store.

The balance of capability between the system components was chosen to provide a simple telephone-subscriber type of interface between remote input devices and the central equipment that was amenable to commonly used telephone-system signalling and installation practices. This interface also enhanced the desired ability to operate over long lines with least future modification.

ELEMENTS OF THE SYSTEM

The EDGE System (Fig. 1) is comprised of three basic types of equipment: remote input devices, passive line or circuit switching devices, and the central recording equipment with its time generation function. Input stations are used to enter data from remote points over two-wire telephone in-plant lines, which are switched by line concentrators into a smaller number of two-wire trunks that are terminated by the central recording equipment.

Input Station

The input station (Fig. 3) is used in great quantity in a given EDGE system application. Physically, it includes two enclosures: a reader unit, and a subset. There are two basic types of reader units used: The general-purpose input station uses a data-input reader (Fig. 2). The alphanumeric input station (Fig. 4) is a more specialized unit and utilizes an alphanumerical reader unit. Both of these devices are designed to sit on a countertop or wall-mounted shelf.

In addition to data read from input documents and manually set data levers, the input station adds certain fixed data and control symbols to each transaction. The subset accepts data from either type of reader unit, translates transaction message characters to the EDGE transmission code, and transmits these characters in a character-serial, bit-serial stream over standard telephone lines using a 2,000-cps phase-modulated tone.

Line Concentrator

The Line Concentrator acts as a switching intermediary to distribute the intermittent demand for service from a large number of input stations to a smaller number of two-wire trunks connected to the central recorders. The line concentrator is mechanized with standard telephone-type, combinational relay logic. Its circuits and components are similar to that used in private-automatic-telephone-exchange (PABX) units. The line concentrator (Fig. 5) accepts up to 25 input lines, which can be switched to four trunks.

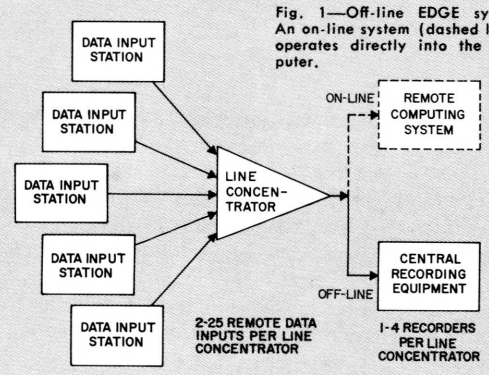


Fig. 1—Off-line EDGE system. An on-line system (dashed lines) operates directly into the computer.

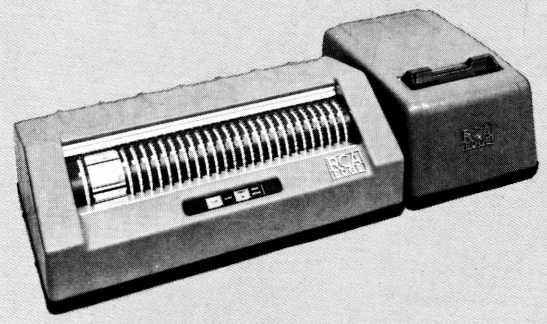


Fig. 4 — The alphanumeric reader unit is used to originate transactions requiring manually settable fields of alphanumeric data. Up to 25 digits of alphanumeric data may be entered into a transaction by this device. Data is entered by operation of its 25 wheel-type switches. This unit is used primarily in the origination of remote inquiry transactions, and to create records with a long alphanumeric data field for which no prepunched cards are available at a location.

Fig. 2—The data reader unit accepts data from an EDGE token that contains up to 12 alphanumeric characters of punched data, an 80-column Hollerith-coded punched card, and up to 11 manually-set numeric characters from 10 multiposition sliding switches. All manually-selected characters are visually displayed. A single manually-set transaction-selector sliding switch is provided for identification of 11 transaction types. Transactions which are comprised of Token Data only; Token Data and Variable data; or Token data, Variable Data, and Hollerith card data can be originated. A key-operated supervisory switch restricts the use of certain pre-designated transactions.

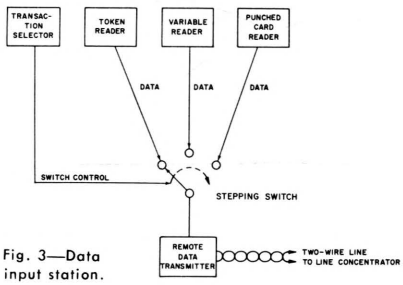


Fig. 3—Data input station.

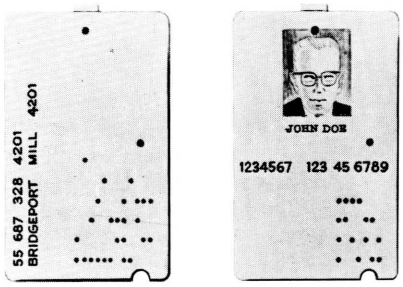


Fig. 7—The EDGE token provides space for punching up to twelve alphanumeric, seven-bit characters in binary-coded-decimal format. The seventh bit allows parity checking of each character read from the token. The EDGE token is credit-card sized (2⅛" or 3⅜"), and is compatible with standard automatic credit-card embossing and punching equipment.

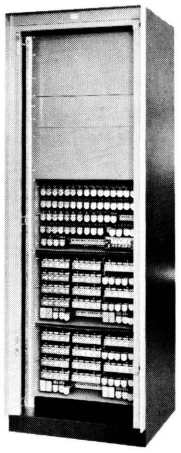


Fig. 5—A Model 6244 line concentrator with the door and cover panels removed. Model 6242 provides two output-trunk connections to two receivers, while Model 6244 provides four output-trunk connections to four receivers.

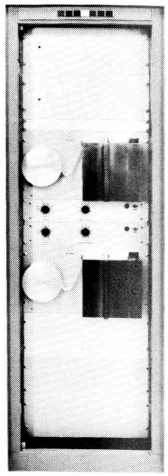


Fig. 6—A two-receiver-punch central recorder assembly. The receivers contain parity checking circuitry and transmission sequence control logic to insure proper reception. Two receivers and recorders with a single power supply are contained in each central recorder rack.

Table I—The EDGE Transmission Code.

CHARACTER MEANING	PUNCHED CARD CODE EQUIVALENT	EDGE CODE
The code is arranged with the least significant bit designated 2^0 and the most significant data bit designated 2^5. Parity in the EDGE code is defined as even.		
Space		000000
1	1	000001
2	2	000010
3	3	000011
4	4	000100
5	5	000101
6	6	000110
7	7	000111
8	8	001000
9	9	001001
&	12	010000
A	12-1	010001
B	12-2	010010
C	12-3	010011
D	12-4	010100
E	12-5	010101
F	12-6	010110
G	12-7	010111
H	12-8	011000
I	12-9	011001
—	11	100000
J	11-1	100001
K	11-2	100010
L	11-3	100011
M	11-4	100100
N	11-5	100101
O	11-6	100110
P	11-7	100111
Q	11-8	101000
R	11-9	101001
O (Zero)	0	110000
	0-1	110001
S	0-2	110010
T	0-3	110011
U	0-4	110100
V	0-5	110101
W	0-6	110110
X	0-7	110111
Y	0-8	111001
Z	0-9	111001
Start Message (SM)	not used	111110
End Message (EM)	not used	111010
Item Separator (ISS)	not used	111101
Delete Message (DM)	not used	101110

Table II—Bit Sequence for One Frame.

Bit Period	1	2	3	4	5	6	7	8	9
Bit Name	Start	2^0	2^1	2^2	2^3	2^4	2^5	2^6	Stop
Bit	1	*	*	*	*	*	*	*	9

* Character bits, 1 or 0

Table III—EDGE Format Symbols.

<	Start message symbol (SM)
•	Item separator symbol (ISS)
>	End message symbol (EM)
■	Delete message symbol (DM)
T_C	Transaction code
I_D	Unit identifier character
T	Token reader character
V	Variable character
C	Chard character
H_{10}	Ten hour character
H_1	Hour character
$H_{.1}$	Tenth hour character
$H_{.01}$	Hundredth hour character
W	Day of Week Character

Table IV—The Transmitted Message Formats.

These formats apply to both an off-line and an on-line system. In an on-line system, the transmitted format is stored in memory where it is combined with the time and day code field output of the time scanner, on-line by computer program. The off-line receiver-punch adds the time and day code field at the central recorder equipment.

TITLE	FORMAT
T-6	$\lceil T \mid \mid \mid \mid C\ D_1\ D_2\ D_3 \bullet T_1—T_6 \rceil$
T-12	$\lceil T \mid \mid \mid \mid C\ D_1\ D_2\ D_3 \bullet T_1—T_{12} \rceil$
V	$\lceil T \mid \mid \mid \mid C\ D_1\ D_2\ D_3 \bullet T_1—T_{12} \bullet V_1—V_{10} \rceil$
C-X	$\lceil T \mid \mid \mid \mid C\ D_1\ D_2\ D_3 \bullet T_1—T_{12} \bullet V_1—V_{10} \bullet C_1—C_X \rceil$
C-Y	$\lceil T \mid \mid \mid \mid C\ D_1\ D_2\ D_3 \bullet T_1—T_{12} \bullet V_1—V_{10} \bullet C_1—C_Y \rceil$

Central Recorder

The modular central recorder used with off-line EDGE consists of receivers and punched-paper-tape recorders (Fig. 6). A time equipment unit is associated with the central recorder to automatically supply time-of-day and day-code data to each receiver-punch. Dual time emitter units provide extra reliability and unambiguous read-outs during time changes without the need for a blanking period.

Central Receiver Terminal

When an EDGE system is used on-line (i.e., linked directly with a computer), the central recorder is replaced by a central receiver terminal designed to operate with an RCA 301 system that is equipped with a communications-mode-control (CMC) unit. The central receiver terminals replace the receiver-punch units in the on-line configuration.

In the on-line configuration, the time equipment unit is replaced by the time scanner on-line dual (TSOL) channels, which are sampled by the CMC in the same manner as it scans all 80 receiver trunks.

In all other respects, the equipment and operation of the basic EDGE is the same for on-line as for off-line operation.

THE TRANSMISSION SYSTEM

The code used in transmission is called the EDGE *transmission code* (Table I) a 7-bit, Hollerith-derived, alphanumeric code. Data read from the punched card and manual levers are code-converted prior to transmission. EDGE tokens are punched in EDGE transmission code and, therefore, bypass the code-conversion process. Although EDGE transmission code is always utilized between the subset and receiver, the output of the receiver may be converted to variations of the basic code.

The transmission of data in the bit-serial form requires parallel-to-serial conversion at the subset and serial-to-parallel conversion at the receiver using shift registers. Serial transmission requires bit synchronization and character framing to properly extract discrete characters from the bit stream.

A frame contains 9 bits. The first bit of the frame is designated the *start* bit and is always a *1*. The succeeding seven bits are six *data* bits and one *parity* bit. The ninth bit is designated the *stop* bit and is always a *0*. The presence of a *1* in the *start* bit position of every frame provides the means by which the serial-to-parallel shift register in the receiver determines the beginning of each frame. Loss of this bit will cause loss of synchronization, failure to satisfy the parity check, and will generally be detected within three frames (or characters).

Table II illustrates the bit sequence for a frame, the relationship between the *data*, *parity*, *start*, and *stop* bits.

Transmission of data from an input station utilizes the 2-kc carrier frequency-generated within the subset from the 4-kc timing source. The 2-kc carrier exists in two forms: a *1* phase, and a *0* phase, which are 180° out of phase. When a marking element is present at the subset output, the *1* phase carrier is transmitted. When a spacing element is present, the *0* phase carrier is transmitted. At the central receiver, a phase detector monitors the 2-kc input signal and generates a pulse for each change of phase of the carrier. This pulse triggers a binary circuit, the outputs of which are read into a shift register each bit time, thus recontracting the transmitted character in binary form.

ORGANIZATION OF DATA

EDGE uses a semivariable data-field and message-length concept to reduce the number of unwanted space characters that would otherwise be required. This feature, together with the concept of masked setup, clearing and system switching time enables EDGE to achieve an effective transmission efficiency equal to or greater than might be achieved with a fixed-message-length format operating at double the EDGE transmission rate of 27.7 characters/sec.

The scanning order for the reader-unit components and their corresponding data fields is fixed in this sequence at the time of factory assembly: *station identification*, *token*, *variable* and *card*. The *token-only* transaction for attendance recording creates the shortest transaction message used and accounts for the greatest number of messages per unit time during clock-in and clock-out periods. Hence, with the token data field occurring first in the scanning sequence, the stepper may be caused to short-scan by homing immediately after the last character of the employee number within the token field is read. Two typical tokens are shown in Fig. 7.

While the volume of *token-variable* transactions is signficantly less than *token-only* transactions, homing of the stepper immediately after scanning of the last character of the variable data field also eliminates unwanted space characters.

A *token-variable card* format repeats the field length requirements of the token-variable format and adds the *card* field. The card field may contain a maximum of 80 characters. Movement of a single taper tab in the reader unit stepper determines the number of columns which are read from a card.

Several different symbols are used in describing the message formats which can be used when transmitting data in the EDGE system (Table III). The format of the transmitted message depends upon the transaction that is selected. Each position of the transaction selector switch is wired to effect reading and transmission of the desired format. Whether or not data must be entered into the card reader, token reader, and/or variable data reader is a function of format and interlock selection.

Data fields within a message may or may not contain data. When data is not required from a particular field for a particular transaction, all character positions of the corresponding data field will contain space characters if the field is scanned. The formats are listed in Table IV.

In an off-line EDGE system, all data are punched on paper tape in the order received. Upon arrival of the *end message* symbol, the receiver completes its various checks on the message. If the message checks satisfactorily, the *end message* symbol is replaced by an *item separator*, wherein the time and day-code field of five data characters is scanned from the time equipment unit and appended to the portion of the message received. An end message symbol is then inserted.

Should the receiver detect a parity or sequence error in an incoming message, the punching of all subsequent characters is inhibited and a *delete message* symbol is punched on tape in place of the character found to be in error. The output from the paper tape recorder is a 7-bit code punched in 1-inch paper tape.

The EDGE token (Fig. 7) is a rugged, multipurpose, machine-readable input medium for use with the input station. This token has many applications, including employee identification, tool crib check-out, standard operation data, and machine utilization data.

ACCURACY AND CHECKING FEATURES

EDGE provides four basic types of accuracy checking: 1) input-station interlocks to assure that messages are properly set-up before transmission; 2) parity check of token characters read from token, and parity-bit generation for variable and card field characters; 3) parity and symbol sequence checks of the transmission to protect against transmission errors; and 4) recording checks to assure accurate recording on paper tape.

Input station checks include:

1) A *punched card insertion check*

which rejects the card if it is inserted incorrectly.

2) A *token insertion check* which rejects token if inserted badly.
3) A *card-attendance exclusive interlock* which prevents transmission of an attendance message when a card is inserted in the card reader and token-insertion transmission-initiation is used.
4) A *ready interlock* which prevents transmission of any selected transaction message until all required inputs are properly satisfied.
5) A *supervisory key-switch* which provides for the restriction of certain transactions.
6) An *automatic reset* which provides automatic return of all input documents and reset of all manually-set levers after a message transmission has been received and verified.

Transmission checks include:

1) Automatic retransmission of messages for transmission errors.
2) Automatic shutdown and alarm of faulty input stations and/or receivers when error incidence is excessive. When an input station fails to transmit a message correctly after 15 to 35 seconds of automatic retransmission, it alarms and retains all documents and settings until manually and deliberately cleared by the operator.

Recording checks include:

1) *low paper indicator* illuminates an alarm at the central recorder.
2) *end of paper alarm* disables the receiver and causes message to be diverted to other receivers.

The coordination method used between the input station and the receiver is closed loop. No transmissions are initiated without issuance of a *request to transmit level* from the input station, and subsequent receipt of *permission to transmit* from the receiver. No message transmission is assumed to be correct without receipt of *verification* by the input station. If *verification* is not received, automatic retransmission is repeated until a 15-to-35-second hold-down time delay expires at the input station. A similar arrangement enables the receiver hold-down to expire if satisfactory transmission is not achieved in 35 seconds. In both cases, hold-down expiration results in an *out-of-service* alarm.

FUNCTIONAL REDUNDANCY

In a typical EDGE installation, all system components contribute to the overall capability of the system. The elements of the system are designed and interconnected in a manner to prevent the failure of any component or major system element from causing a catastrophic loss of all system capacity. Automatic alternate path routing within the network results in partial degradation rather than total failure of the system when a major malfunction occurs. This desirable characteristic has been implemented in the following manner:

Input Station: Because of the multiplicity of stations, loss of a station results only in loss of the time required for an employee to walk to the next nearest input station. Use of separate two-wire communication lines between each input station and the line concentrator minimizes the danger of catastrophic loss of a subsystem because of sympathetic faults that are characteristic of commonly used way-station type operation or wire failures.

Line Concentrator: Alternate path capability through two of four trunks is provided by the internal relay logic of this unit. In addition, telephone switching circuitry of this nature has been perfected to a very high degree of reliability by many years of field operation.

Central Recorder: The multiplicity of receiver-punch combinations and the automatic re-routing by trunk that is characteristic of line concentrator prevents total failure of the central equipment. Thus, failure of any receiver, punch, or power supply results in only partial degradation of system capacity.

Time data for the central recorder is provided from dual time emitter units. Hence, loss of one of the time units results only in small loss in system handling performance.

APPLICATION

In designing an EDGE System for a particular customer application, several factors must be considered in optimizing the equipment complement. These factors include:

1) Volume of messages by type per unit time.
2) Geographic distribution of remote units to reduce long walking distances for operators.
3) Need for two-way transmission.
4) Types of transactions to be originated and set-up times.
5) Ratio of input stations to line concentrator to receivers necessary to satisfy traffic requirements.
6) Allowable attendance recording queues.
7) Physical separation between units in observance of maximum loop resistance.

An EDGE System in combination with communications control computers and other system elements forms a completely integrated data-communication and processing system.

While the detailed procedure for optimizing an EDGE System complement is beyond the scope of this paper, there are several considerations and performance characteristics of general interest:

Traffic Analysis: Each customer's operation must be analyzed to determine the number of transaction types per unit time which the system must handle. Traffic loadings must be determined in terms of messages and characters per unit time for the areas served.

Geographic Layout: Layout and location of the EDGE devices within a plant is generally performed on the basis of employee population density and the average walking distance for employees to and from an input station in their area. One input station is generally assigned to employees located within a radius of 75 feet. Within this rule, the minimum number of input stations for an area is limited by the input station set-up, clearing, and transmission time. A data input station can clock up to 30 employees per minute. A receiver, however, handles 85 employee attendance transactions per second. Therefore, the maximum number of input stations which are connected to a line concentrator is either limited by the receiver message rate and the number of trunks on the concentrator or by the 25 input lines to a concentrator. For low-volume applications, two or more line concentrators may be connected in tandem, thereby increasing the ratio of input stations per receiver. Input-station-to-line-concentrator telephone-line lengths are limited to a total loop resistance of 900 ohms. This resistance represents a distance of 2 to 6 miles depending on wire gauge. The line-concentrator-to-receiver telephone lines may have a loop resistance of 2,400 ohms.

Installation: A major advantage of the EDGE system is its use of standard two-wire telephone lines between all units. Such lines may be installed by private contractors, plant maintenance personnel, or by the local telephone company. Telephone company lines are usually installed at a cost of approximately $5 per drop and rented for $1 to $2 per quarter mile per month. This arrangement minimizes capital investment. Once the telephone lines are installed, removal and relocation of a unit is accomplished with the same ease as moving the telephone instrument.

ACKNOWLEDGMENT

The many contributions to the planning, development, and design of the EDGE System by Messrs. C. J. Pritchard, J. P. Reid, R. R. Helus, W. Saeger, R. P. Graeber and B. P. Silverman are gratefully acknowledged.

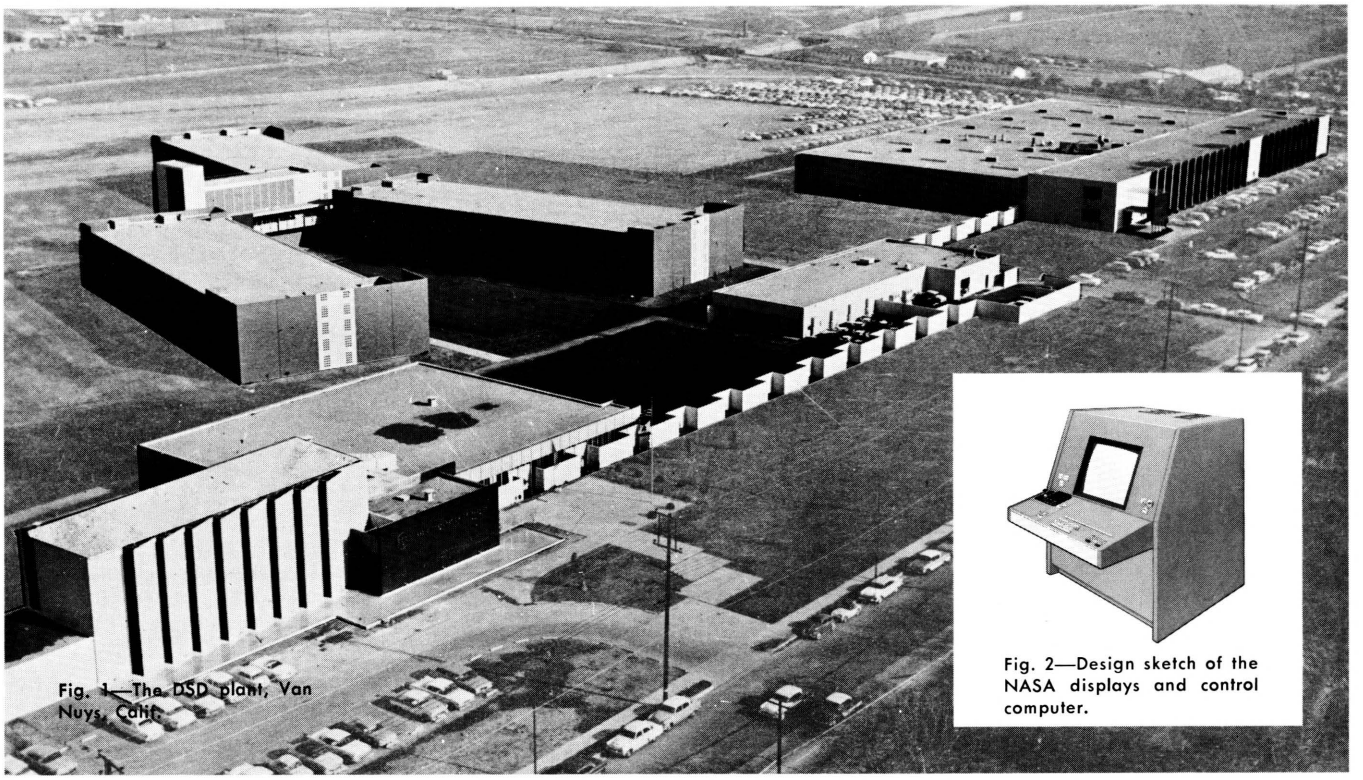


Fig. 1—The DSD plant, Van Nuys, Calif.

Fig. 2—Design sketch of the NASA displays and control computer.

PRODUCT TRENDS IN THE DEP DATA SYSTEMS DIVISION*

G. F. BREITWIESER, (former) Chief Engineer
and
D. J. PIZZICARA, Staff Engineer
Data Systems Division
DEP, Van Nuys, California

THE DATA SYSTEMS Division of Defense Electronic Products is an integrated organization of 2000 people devoted to the data processing and display business. Its basic mission is to pioneer advances in the data arts and carry them to the product-line stage—both individual equipments and total systems.

Some idea of the economic challenge of data processing may be gauged from the national rate of expenditure. Total government and business expenditures for 1962 are in excess of a $2 billion rate, nearly 1/200 of the gross national product. Projections indicate an increase to a rate between $4 billion and $7 billion by 1970. This would place data processing in a position of prominence currently characterized by the apparel or paper industries.

An informative picture of data processing activities in the Data Systems Division may be gained by considering the following questions: *What are our products?; What are our prospects?;* and *What are our personnel skills?*

MARKET AND PRODUCT TRENDS

Near-future projections show that the following product areas will make up the base of data-processing business in the next two to three years:

1) *Digital data processors* — computers and associated devices.
2) *Visual data processors* — displays, film processors, electroluminescent panels, etc.
3) *Informative storage and retrieval processors* — video file, hybrid combinations of Electrofax, RACE (Random-Access Card Equipment), and digital and visual processors.

Market studies of the military, NASA and other areas show an overwhelming trend toward wider application of data-processing technologies. Applications such as weapon-control centers, tracking, data instrumentation, data reduction, check-out, launch control, command and management control, intelligence processing, countermeasures, weather prediction, patent and document processing, and air-traffic control, to name a few, are all dependent upon this advancing art for implementation. In DEP-DSD, numerous products and systems concerned with data processing are being developed and delivered to the military.

For example, the following products were delivered to defense customers during 1962:

Mobile Automatic Programmed Checkout Equipment—MAPCHE

MOBILE APCHE[1], or MAPCHE, is the prime checkout equipment employed to test and validate most subsystems of the various ATLAS missiles and associated air-ground equipment. It performs discrete response tests, analog response tests, timing tests, and servo-response tests. To evaluate a system under test, a deck of punched cards, programmed for an automatically sequenced test, is inserted in a card reader. A program control interprets this data and generates the proper stimuli and response.

RCA-4100 Computer

The RCA 4100 is a digital computer for military use of the internally stored program type.[5] It features 16 priority-interrupt levels, a magnetic core memory of 4096, 8192, or 16,384 words of 30 bits, and adaptability to van, shipboard, and fixed installation. The priority-interrupt feature conveys to the RCA-4100 a powerful real-time capability both from the standpoint of multichannel input-output and multiprogram control.

The RCA 4100 uses solid-state components and satisfies a combined military specification based upon MIL-E-16400C, MIL-E-4158B, and XEL303 requirements. A computer using identical

*Recently renamed Aerospace Systems Division

modules has experienced only 12 minutes downtime in 24-hour-per-day operation over a nine-month period.

High-Speed Arithmetic Unit for RCA 604

The RCA 604 High-Speed Arithmetic Unit and the RCA 603 Processor combine to provide a complex having advanced scientific computing and data processing capabilities. The RCA 604 multiplies, divides, adds, subtracts, and multiply accumulates in either fixed, floating-point, or psuedo-floating-point modes. The machine has a 36-bit mantissa and a 9-bit characteristic. It performs arithmetic in either single or double precision. Computation time for floating-point multiplication can be accomplished in about 6 μsec, exclusive of memory accesses. Exclusive of power supplies, the RCA 604 occupies two racks and employs about 15,000 transistors.

RADCON Digital Computer

The RADCON digital computer is especially adapted to military tactical utilization. It is a bus-organized, parallel, synchronous, single-address machine with a fixed program stored in a special wired-core memory. Loss of program instructions during a tactical situation is virtually impossible, since instructions are represented by wires threaded through pulse transformers contained in plug-in program trays. Programs are altered by changing program trays when rapid changes are desired. The RADCON has the following features: 1) a random-access, coincident-current core memory for data, 2) three index registers, 3) an interpreter for address modification and selection, 4) a sophisticated arithmetic unit, 5) error-detecting facilities, 6) error registers to hold errors for later analysis and action, and 7) real-time input-output facilities. Interface equipments include optical encoder strobing circuitry, digital-to-analog and analog-to-digital converters, a frequency-to-digital converter for encoding doppler, and a 54-kc digital data link required by the dispersion of the RADCON subsystem.

RCA 110 Computer System

The RCA 110 Computer System[6] is furnished to NASA for use as checkout control in the SATURN Program. It is comprised of eight cabinets (three main frame, five peripheral) and ancillary equipment consisting of an electric typewriter, a paper-tape reader and a paper-tape punch. The entire complex meets RFI military requirements. Peripheral equipment consists of analog and discrete switching units, power supplies, and tape stations. The RCA 110 has a serial arithmetic unit with parallel access to core storage. Programming flexibility is enhanced by the large storage capacity provided by both the high-speed working storage (4096 words) and the drum bulk storage (32,768 words).

As a central control unit, the RCA 110 communicates with other equipment through peripheral input-output units. Each class of signals (command, response, analog, discrete) is handled through its own channel under computer control and enters or leaves the computer through the appropriate input-output buffer.

RANGER Video Processor

This program, a subcontract from the DEP Astro-Electronics Division, Princeton, is for the Jet Propulsion Laboratory lunar program, RANGER. The first system has already been delivered for recording spacecraft video signals of the lunar surfaces. Recordings are made simultaneously on magnetic tape and film. The resolution and geometric accuracy achieved in this equipment far exceeds that of previous video recorders. In addition, the video processor checks out spacecraft cameras and associated ground support equipment prior to launch from Cape Canaveral.

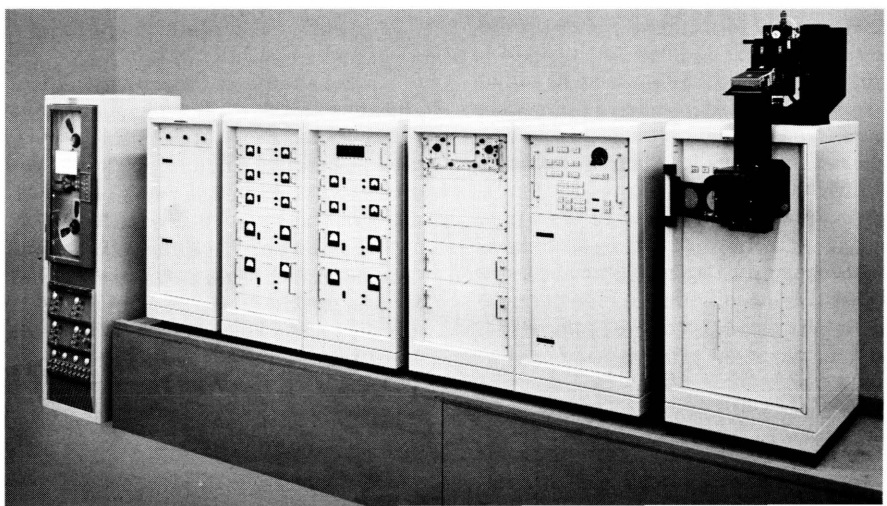
NASA Display and Control Console

The Display and Control Console consists of a desk-type console with a 21-inch display storage tube. It displays information in the form of charts, graphs, or printed pages under operator control. Printed pages can be presented at a rate of 8,000 alpha numeric charac-

D. J. PIZZICARA, Staff Engineer DSD, rceived the BSEE in Electronic Engineering from City College of New York and MSEE from Harvard University. He has completed advanced courses at Brooklyn Polytechnic Institute. His 15 years professional experience includes communication equipment, electronic computer development, and weapons systems direction equipment as well as engineering administration of several major computer systems. Additionally he served with the Air Force, instructing enlisted and officer personnel in electronics, physics, and mathematics. Mr. Pizzicara has been associated with the following engineering projects: the MKIII Computer for TALOS, weapon direction equipment for TALOS, TERRIER and TARTAR; and the data processing central for the MARS. Mr. Pizzicara is a member of Tau Beta Pi and Eta Kappa Nu societies. He received the regents scholarship and faculty award in mathematics. He has U.S. patents pending in the application of aided track techniques to scan conversion system, and has published several papers.

G. F. BREITWIESER Chief Engineer and Projects Manager of DSD when this paper was written, received the BS in Marine Engineering from the United States Coast Guard Academy and BSEE and MSEE from the Massachusetts Institute of Technology. His 18 years' experience include radar, sonar, and weapons control engineering as well as engineering administration of several major weapons systems. Serving with the Coast Guard, he gained experience as a radar, sonar, and/or gunnery officer, and later as officer-in-charge of engineering at the Electronic Engineering Test Station. Among the engineering projects Mr. Breitwieser has been associated with are: design and development of the AN/FPS-10D Height Finder Radar; project engineer on the Army Fire Control system for TERRIER; and manager of electronic development on such systems as the T44 Tank Fire Control system, the AN/SPS-12 Search Radar system, the AN/UPS-1 Assault Search Radar, the AN/FPS-16 Instrumentation and Radar system, the TALOS IM-70 Weapon system, and the MK 8 Shipboard Fire Direction system for TALOS. More recently, he has been concerned with the WS-107A-1 Atlas Launch Control and Checkout system, and the RCA portions of the Ballistic Missile Early Warning system (BMEWS). He is a member of IEEE and was awarded the coveted RCA Victor "Award of Merit."

Fig. 3—RANGER video processor.



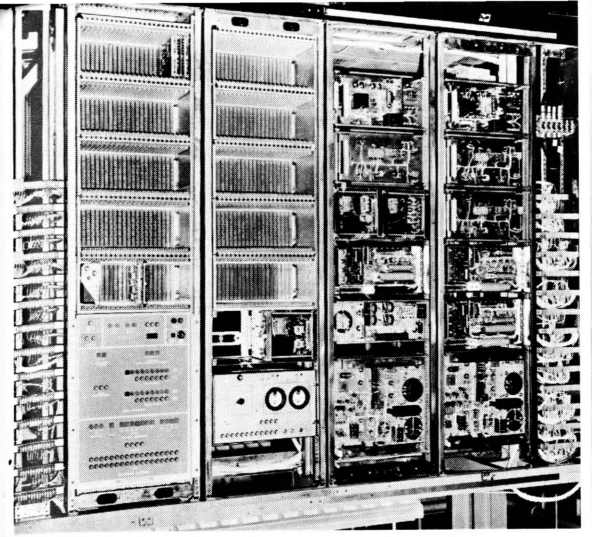


Fig. 4—RADCON computer.

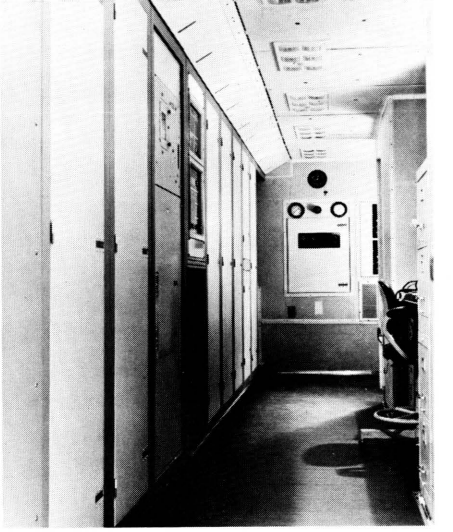


Fig. 5—Mobile APCHE.

ters per second, permitting a full page of 1,600 characters to be presented in 0.2 of a second. Data can be retained on the tube face for up to two minutes and erased under operator control.

A typewriter type keyboard containing 43 alphanumeric characters and symbols controls the display console. In addition, a smaller keyboard containing 18 special symbols is used with the RCA 110 Computer.

FUTURE PROSPECTS

DEP-DSD is currently participating in the following missile or space programs: ATLAS, SATURN C, NOVA, BMEWS, and MINUTEMAN. Adidtionally, proposals are pending on such programs as: LUNAR EXCURSION MODULE, SATURN C5, NOVA, SATURN OPERATIONAL FLIGHT CONTROL SCHEME, APOLLO—GOSS, and ADVENT.

The RCA 110 and RCA 4102 computers launched in 1962 are being offered as on-line, real-time computers. Both machines promise a two- or three-year useful life in the market place and should yield a significant return on investment.

Another step forward is the *video file* concept for electronic image-processing or document storage and retrieval. The video-file development promises to be a cornerstone of DEP-DSD's entry into this important market. Information storage-and-retrieval systems', by 1967, may well represent a volume of business exceeding that of computer data-processing systems.

RACE (Random Access Card Equipment) is a new bulk store which will make large scale information storage and retrieval systems a practical reality. RACE has potential to store a billion bits (digital), or in excess of a million video images or film clips with average random access of 300 milliseconds.

The ready availability of such techniques as ELSI[2] (Electroluminescent Storage Indicator), high-speed jump scan, and digital-generated video will increase RCA's penetration in the command-control and program-management display fields. Typical of future applications are the Navy Strike Command, a flag officer strategic and tactical system; and the Management Program Command System to be located at the Directorate in Washington, D.C. A further development, now in its infancy, is reflective display. This has application to large-screen displays, providing the capability of high information rates. Reflective displays have the advantage that room illumination contributes to the brightness of the observed display, facilitating normal operator functions of writing or reading messages. These developments plus higher circuit speeds (clock rates up to 100 Mc) and memory-cycle speeds (0.2 microseconds or better) are today's technological building blocks.

ENGINEERING

DEP-DSD has grown from the 1959 force of approximately 500 (200 engineers) to 2000 in 1962, including an Engineering Department of approximately 1,100 at Van Nuys and 100 at the Data Systems Center at Bethesda, Maryland. This rapid growth was concurrent with the execution of several military programs grossing $200 million.

To carry out these programs, DSD requires the following broad mixture of skills:

System Application Analyst
Programmers
Operations Analyst
Machine Organization Specialists
Logic Designers
Product Planners
Numerical Analyst
Intelligence Specialists
Library Research Specialists
Language Analysts

The organized efforts of these specialists will conceive tomorrow's data-processing products. Along with advanced skills, the need arises for new concepts in product design and manufacturing techniques. Present-day components, operating well into the VHF region, have pushed conventional wiring and packaging techniques close to the limit of their capabilities. The data-processing art is facing a new challenge to package, interconnect, and manufacture micromodular and microelectronic circuitry for operation in VHF.

An idea of the impending radical changes imposed by molecular electronics can be gained from the fact that thousands of thin-film transistors can be deposited on one square inch of substrate. Clearly, the capabilities of manual dexterity are exceeded—and automation is demanded. Several research and development programs are being pursued to gain insight and solutions to these problems.

Successful prosecution of these endeavors is essential to expanding the base of data processing applications. A few untapped markets are as follows: libraries, the medical profession, and the legal profession. All require the services of the data processing field to contain and harness the aptly termed information explosion.[7]

CONCLUSION

RCA has entrusted DEP-DSD with considerable funding for proprietary data-processing equipment development. These developments will become our base of business operations for the next two or three years. In the data-processing market, an *off-the-shelf* product is mandatory to compete successfully with the rest of the industry. Consequently, we must continue to enter into new and challenging product areas involving advanced concepts.

BIBLIOGRAPHY

1. "Automatic Programmed Checkout and Launch Control for the ATLAS
2. L. E. Haining, "New Solid-State Alphanumeric Display"
3. H. T. Gnuse, "Ferroelectric Decoding and Addressing for "Solid-State Displays"
4. C. P. Hadley, "Electroluminescent-Photoconductive Micromodular Logic Elements"
5. J. A. Cornell, "The RCA 4100 Series Digital Computers"
6. W. M. McCord, "High-Speed, Computer-Controlled Checkout System for the SATURN Program"
7. B. H. Sams, "Programming Information Systems"

Fig. 1—RCA Electronic Data Processing headquarters and engineering in Cherry Hill, N.J.

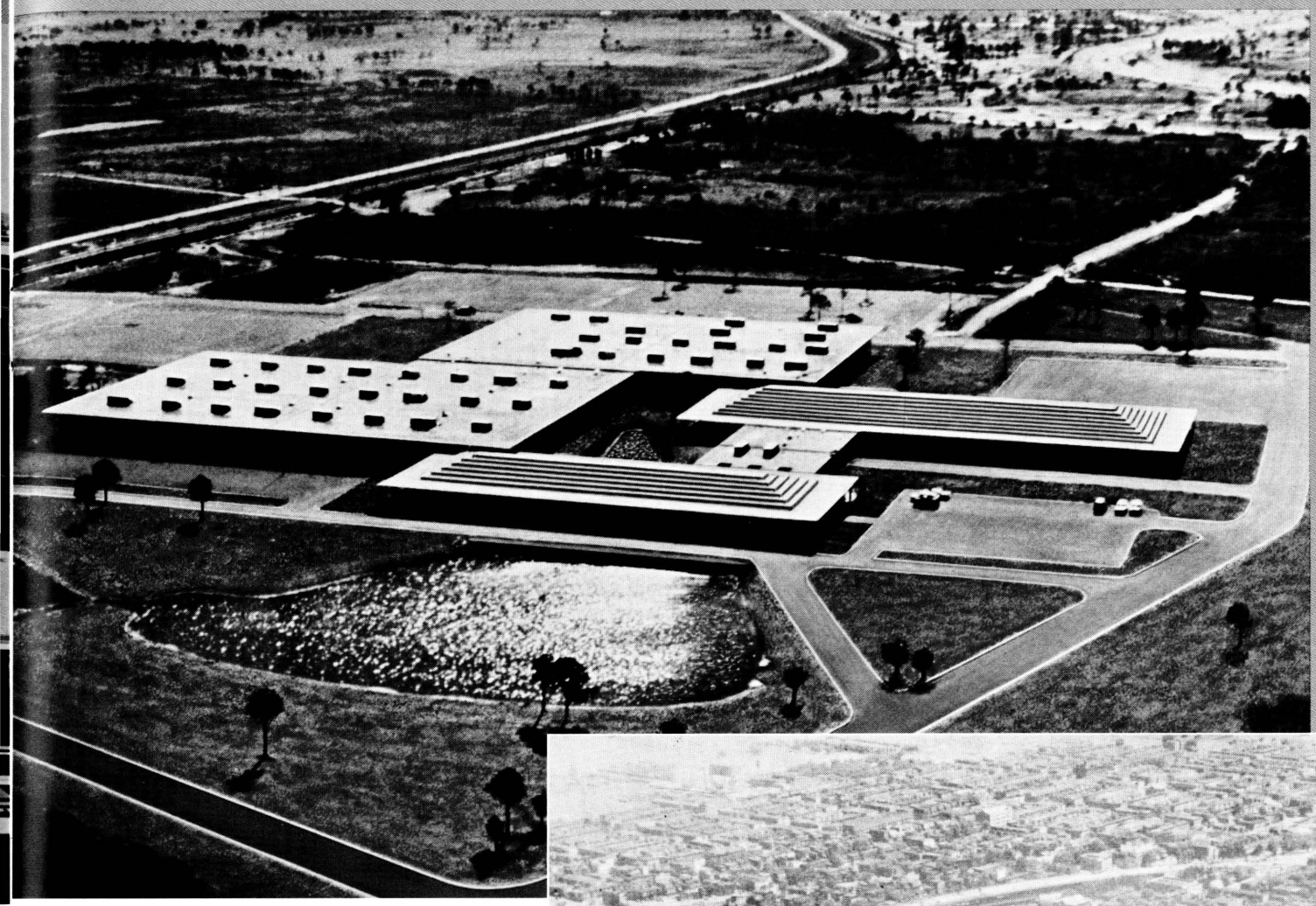
A LOOK AT RCA ELECTRONIC DATA PROCESSING ACTIVITIES AND FACILITIES

T. I. BRADSHAW

Electronic Data Processing, Cherry Hill, N.J.

RCA Electronic Data Processing activities have grown into a multi-plant operation involving thousands of scientists, engineers, and production personnel. From its original Camden location, the EDP organization has spread to its current headquarters in Cherry Hill, N.J., with other major plants in Camden, N.J., and Palm Beach Gardens, Florida. In addition to these and regional offices nationwide, five RCA Data Processing Centers are located in Washington, D.C., Chicago, New York, San Francisco, and at RCA Cherry Hill, N.J. An impressive scope of computer research is underway at RCA Laboratories, Princeton, N.J. Other RCA Divisions are involved in computer installation and maintenance; in engineering and production of computer circuit devices, memories, and special-purpose military computers; and in sophisticated systems analysis and programming research.

Fig. 2—RCA Electronic Data Processing engineering and production, Palm Beach Gardens, Florida.



As far back as 1935, RCA scientists began a study of electronic computing devices. In 1947, at the request of the Navy Department, RCA produced its first computer, an analog device known as Typhoon, which was used to evaluate ship, airplane and submarine performances. Thousands of simulated test runs of proposed guided missiles were made on Typhoon at tremendous time and material savings to the government.

In 1956, RCA produced the Bizmac system for the U.S. Army Ordnance Tank-Automotive Command at Detroit. The computer was designed to control the inventory and supply of parts for military bases all over the world.

RCA's full scale entry into the data-processing market place came with the RCA 501 Computer, announced in 1958 as the industry's first completely transistorized system. In the brief span of two years, the company developed a full line of data-processing equipment to handle the paperwork loads of business and government. To the computers were

Fig. 3—RCA Electronic Data Processing engineering and production occupy major portions of this RCA complex in Camden, N.J.

Fig. 4—RCA Laboratories, Princeton, N.J., where major computer research projects are being pursued.

Fig. 5—Computer memories and materials are engineered in this Needham, Mass. plant of RCA Electronic Components and Devices.

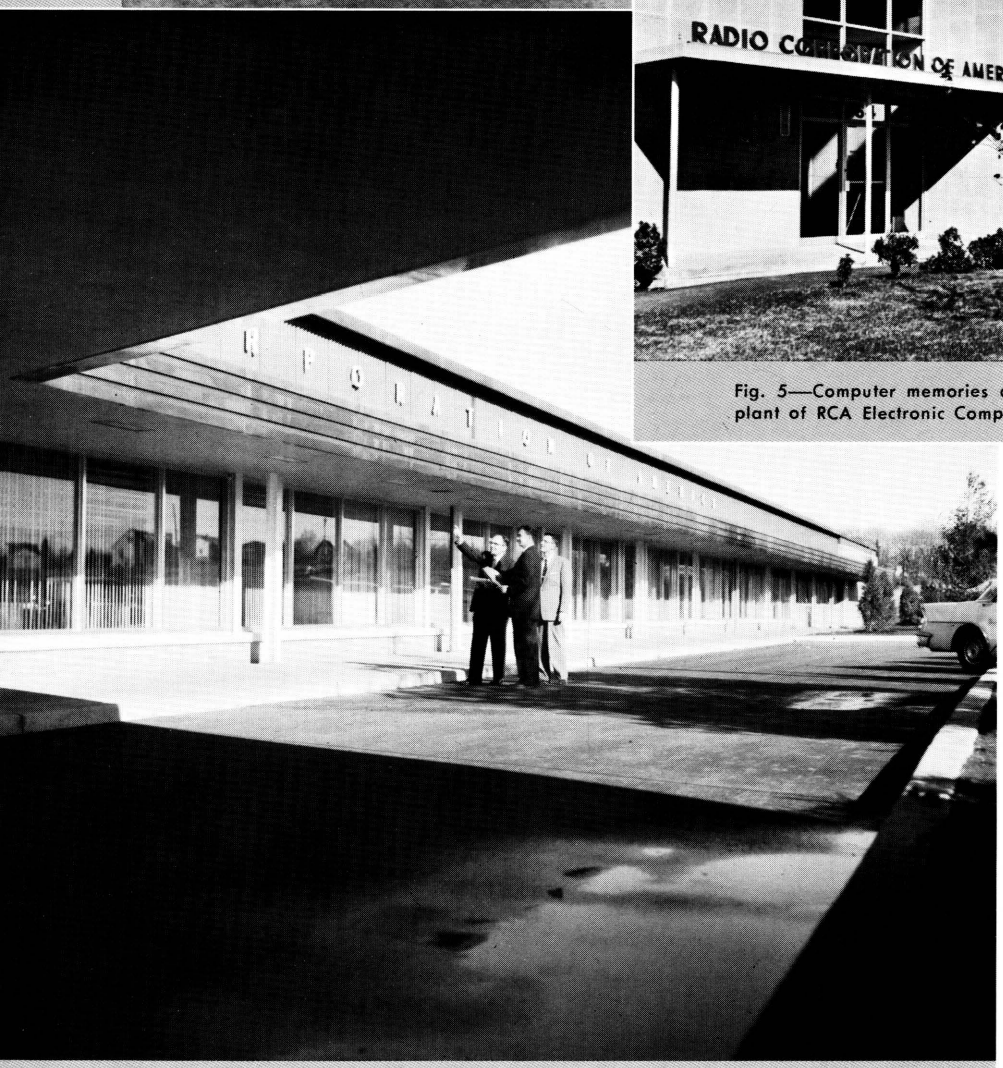


Fig. 6—Engineering and production of computer circuit devices and microcircuits is centered at this Somerville, N.J. plant of RCA Electronic Components and Devices.

added electronic communication devices that relay business data over vast distances.

The Tank-Automotive Command later activated a team of RCA transistorized computer systems which General Thorlin said would do the work of the older machine at two-thirds the cost.

THE RCA 301 AND 601

In 1960, two more transistorized computer systems were added to the growing RCA family, the compact RCA 301 and the bigger, more powerful RCA 601. The RCA 301 is designed to provide full-scale data processing for firms of moderate size. The RCA 301 can serve as a complete data processing system or as an auxiliary to the larger RCA 501 or 601.

The RCA 601, capable of making 700,000 decisions a second, is geared to provide an enormous amount of work

Fig. 7—At this Van Nuys, California plant of the RCA Defense Electronic Products organization, special-purpose military computers and displays are engineered and produced.

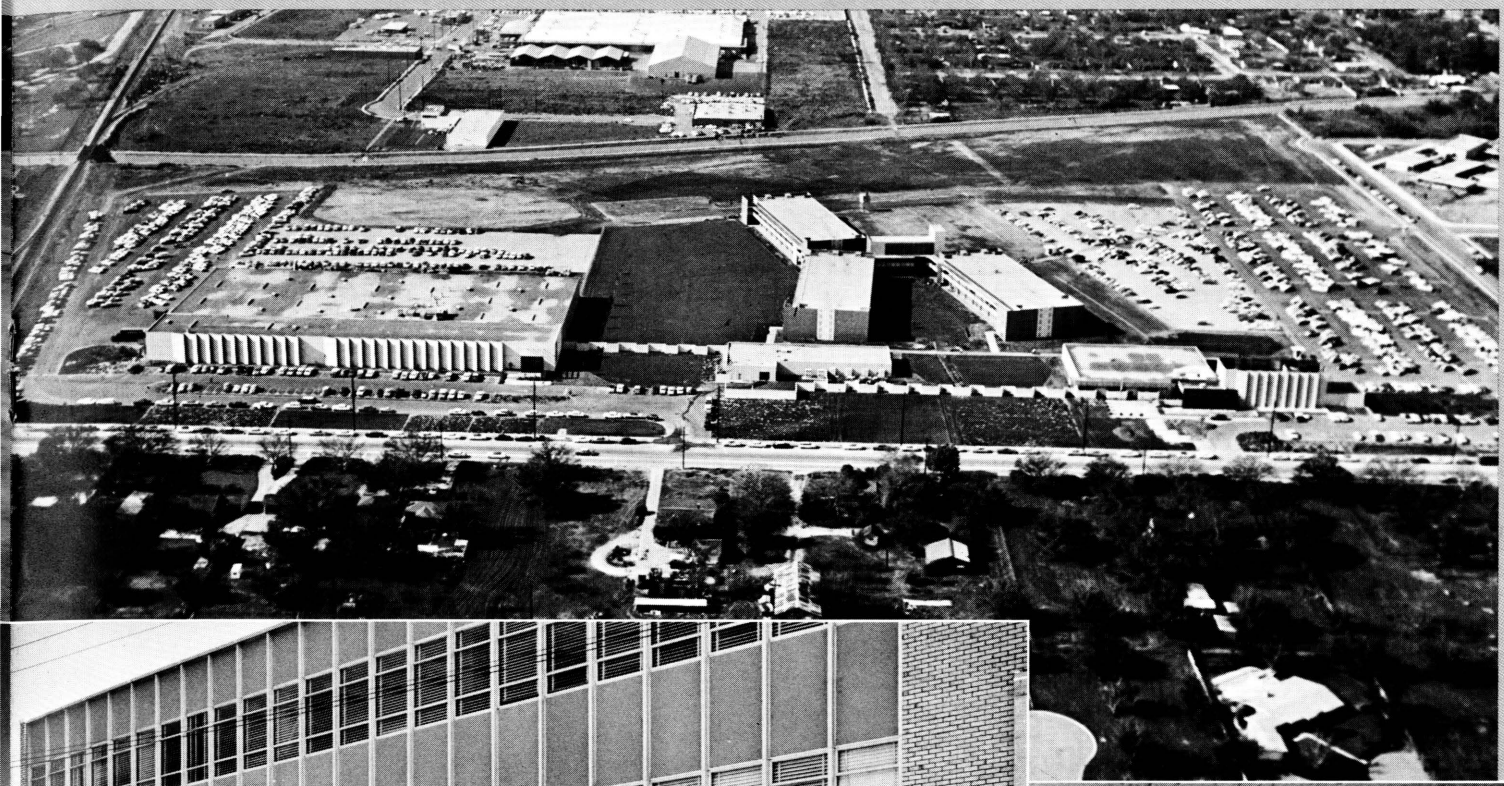


Fig. 8—The RCA Data Systems Center, in Bethesda, Maryland, is engaged in work on advanced information-processing systems and programming research.

power. The system is capable of recalling a fact or figure from its memory equipment in only 1.5 μsec.

Just three years after the 501 unveiling, RCA offered a complete range of transistorized computer services with both basic and peripheral equipment for all types of business, large and small.

FIVE COMPUTER CENTERS FOR BUSINESS

To make complete data-processing services available to the firm not ready for its own system, RCA launched an electronic systems center program. These centers are, in effect, "job-lot" clearing houses for paperwork as well as demonstration points for RCA equipment and training sites for customer personnel. Centers were established in Washington, D.C.; Chicago; the Wall Street district of New York; San Francisco, and at RCA Cherry Hill, New Jersey.

Fig. 9—One of the five RCA Data Processing Centers, this one in Cherry Hill, N.J., that are available to serve business and industry.

COMPUTER RESEARCH, ENGINEERING, AND PRODUCTION FACILITIES

With the extension of its product line and growth in its work force, RCA Electronic Data Processing carried out an accompanying expansion of its physical plant.

In August 1960, a band of bulldozers attacked the scrub pine, palmetto and underbrush on a 115-acre plot near Palm Beach Gardens, Fla., clearing ground for a $4 million plant to turn out the RCA 301 computers (Fig. 2). Six months after the initial groundbreaking work began on the first RCA 301 at Palm Beach Gardens. By late 1962, computer systems were rolling off the production floor at the rate of one a day.

The RCA 501 and RCA 601 computer production was centered at Camden, N.J. (Fig. 3). Administrative and home office personnel of RCA Electronic Data Processing were consolidated at the company's Cherry Hill facility (Fig. 1), situated on 65 rolling acres in New Jersey's Delaware Township.

One of RCA's greatest assets is its mother lode of research and development talent. While it touches all operating divisions of the corporation, it is concentrated in large measure at the RCA Laboratories, David Sarnoff Research Center, Princeton, N.J. (Fig. 4). The RCA Electronic Data Processing organization draws immediate and long range benefits from the research achievements realized there. Research at the Laboratories has many times advanced the state of the art in computer circuit devices and computer memories—and continues to do so.

Another especially important RCA asset for computer products is the engineering and manufacturing of semiconductor circuit devices and ferrite memories (Figs. 5, 6). Within the RCA Electronic Components and Devices organization, the Commercial Receiving Tube and Semiconductor Division is an industry leader in the transistors, diodes, and microcircuits indispensable to modern computers. Work on devices is concentrated at its Somerville, N.J. plant. Similarly, that Division operates, at Needham, Mass., a plant and technical staff that concentrates on computer memories and memory materials.

Further evidence of the breadth of RCA's computer activities is found in the Corporation's Defense Electronic Products organization plant and staff at Van Nuys, California, (Fig. 7) which is heavily engaged in computer systems for the military. Special-purpose machines are also engineered in other RCA Defense Divisions for military field use, for command and control systems, for communications systems, etc.

In the area of advanced systems applications, RCA's Federal Government Systems Support staff activity operates the RCA Data Systems Center in Bethesda, Maryland (Fig. 8) where theoretical and applied work is under way on advanced concepts of information processing and sophisticated programming systems.

Finally, there is the world-wide organization of the RCA Service Company who are available to install and maintain RCA data processing systems.

FOREIGN SALES

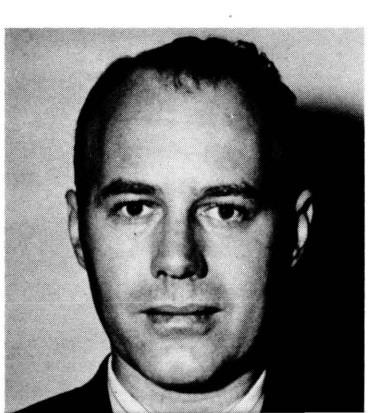
Agreements were reached with three major foreign electronic firms under which RCA Electronic Data Processing would sell these companies RCA 301 systems for re-sale in virtually every part of the world. By mid-1963, a total of 210 RCA 301's had been shipped or were on order to Compagnie des Machines Bull of France, International Computers and Tabulators, Limited, of England, and Hitachi, Limited of Japan.

"Software" developments are giving impetus to increased use of data processing equipment. A *CO*mmon *B*usiness *O*riented *L*anguage, or COBOL, permits the use of simple English words to instruct a computer, rather than the complicated numerical code understood only by computer specialists. RCA was the first to implement the COBOL program.

TWO SPECIAL APPLICATIONS—FORECASTING AND AUTOMATIC TYPESETTING

An RCA sales forecasting technique for business projections was demonstrated dramatically during the National Broadcasting Co. coverage of the last two Presidential elections, pinpointing the eventual winners at an early hour in the vote tally.

THOMAS I. BRADSHAW attended the Hoosac School, Hoosick, N. Y. and Franklin and Marshall Academy, Lancaster, Pa., receiving the B.A. at Franklin and Marshall College in 1937. Excluding three and a half years in the U. S. Navy during World War II, his newspaper career consisted of ten years with the **Philadelphia Bulletin** and eleven years with the **Associated Press.** In 1950 and 1951, he served as an **AP** war correspondent in Korea. He came to RCA in March, 1958, as staff public relations representative for IEP. He now is assigned in a public relations capacity to the RCA Electronic Data Processing.



The RCA 301 was handed a wide variety of assignments. One of the more unusual applications is the electronic compensation or justification of raw news copy to conform with newspaper column width for automatic typesetting. The range of this new tool for the publishing industry was vastly extended by the development of the RCA Transmission and Information Exchange System (TIES), a group of devices which enable two or more RCA 301's to "talk" memory-to-memory or transmit data over great distances.

This latter ability was demonstrated dramatically at an American Newspaper Publishers Association conference in Chicago. News stories were transmitted from Chicago to an RCA 301 at Camden, N.J., then on to linecasting machines in the composing rooms of Scottish and English newspapers via trans-Atlantic cable. Other stories were sent via the RCA-developed RELAY communications satellite from Rio de Janeiro to Chicago through the Camden computer, and from Camden to Great Britain over the same outer space route.

The job-tested RCA 501 was enhanced to increase its speed and productivity up to 33 percent. The four-way "package," which can be applied to RCA 501's on site, enables the computer system to turn out as much as 30 hours of data processing in a 22-hour period.

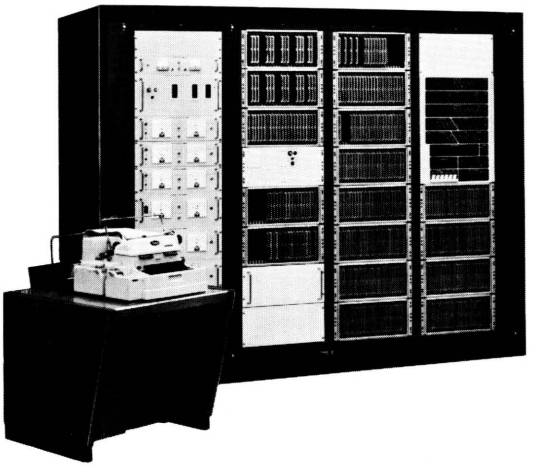
OPTICAL READER—THE VIDEOSCAN

Again taking advantage of its own experience and "tool kit," RCA developed an optical character reader that for the first time combined the scanning ability of the television Vidicon tube with the data-handling capacity of the RCA 301 to process, hourly, up to 90,000 printed documents. The RCA VIDEOSCAN examines the fast-moving documents and feeds the imprinted information directly into the computer.

THE NEW RCA 3301

In August 1963, came announcement of the RCA 3301—fourth in the growing list of RCA transistorized electronic data processing systems and the most unique of all. This computer was developed as a powerful, multipurpose "real-time" machine able to perform each of a variety of functions well. Operating in the nanosecond range, the RCA 3301 combined for the first time functional modularity, serial data processing, a "scratch pad" memory, arithmetic circuitry for scientific work and the ability to communicate with other computers or input-output devices close at hand or far away.

The flood of RCA-EDP activity and achievement indicates that business, science, and government haven't seen anything yet.



THE RCA 4100 SERIES MILITARY DIGITAL COMPUTERS

At top left, the RCA 4101 data processor, military designation CP-685/GPQ, is designed for real-time radar data-processing and control applications. It is constructed on swinging doors intended for van-wall or shipboard-bulkhead mounting and is designed and built to a combined military specification based on MIL-E-16400C, MIL-E-4158B, and XEL303. Several RCA 4101 Digital Computers are in service with RCA MIPIR radars (Missile Precision Instrumentation Radar) in van and shipboard installations on the Atlantic Missile Range and another is scheduled for service on the Pacific Missile Range.

At bottom left, the RCA 4102 data processor is designed for real-time data-processing and control service and general-purpose scientific applications. It is constructed on standard DEP racks and, like the RCA 4101, is designed and built to a combined military specification. The operator's desk, control console, and flexowriter are standard parts of the basic system, and buffer system, and buffer control units for conventional peripheral devices are available. The control panel provides the means for operator control of the central processing unit. It contains sets of indicators for displaying the CPU registers and a set of manual input switches for inserting information into the CPU. Controls are also provided to facilitate manual program tracing and computer maintenance.

The RCA 4100 Series Digital Computers comprise a family of systems designed for real-time control and scientific-data-processing applications requiring simultaneous execution of a number of different programs. The members of the RCA 4100 family, including the 4101, 4102, 4103, and 4150, are solid-state throughout, militarized, and similar in their mechanical, electrical, arithmetic, and programming characteristics.

J. A. CORNELL, Mgr.
Digital Development & Design Engineering
Data Systems Division *
DEP, Van Nuys, California

ALL OF THE RCA 4100 Series Computers feature multiple-function time-shared registers and use of the computer random-access core memory to store 96 *index registers* (the RCA 4150 has 112) and 16 *instruction-location counters* in addition to data and instructions. These features permit a significant reduction in hardware, resulting in machines having the computational power of large-scale data processors but the moderate cost and hardware complement normally associated with medium-scale data processors. The computers also feature a powerful instruction complement enabling programmers to execute simultaneous arithmetic, bookkeeping, test, and branching functions within single instructions. Another feature, common to all members of the family, is a 16-level priority-interrupt system which permits interruption of programs by higher priority programs initiated by external demands occurring at random intervals.

The computers automatically remember the status of any interrupted program and preserve its integrity; the computers return to the original programs even though "interruptions of interruptions" are iterated to a high degree of complexity. The priority-interrupt feature simplifies the scheduling of peripheral devices and permits acquisition of external data from randomly-active channels with short holding times. This is done without periodic scanning and status testing of input channels and the attendant reduction of computing time.

GENERAL DESCRIPTION

All members of the family are parallel, bus-organized machines, and all but the RCA 4150 utilize a binary fractional data word of 30-bit length. They incorporate random-access core memories ranging in size from 4,096 to 32,768 words. The memory read-write cycle time for a typical machine, the RCA 4102, is 5.5 μsec. Most short operations in the RCA 4102, such as addition, are performed in 16.5 μsec, which includes instruction and operand fetching. Indexing adds 5.5 μsec to this time. Multiplication is performed in 75.5 μsec. The RCA 4150 has all of the features of the others, but in addition, has a 36-bit word length, built-in floating-point instructions, and a more powerful instruction complement. Since it is quite different from the other members of the family in some respects, and has not yet reached the hardware development stage, it will not be discussed further in this article.

The machines are designed to handle conventional peripheral equipment, including a flexowriter, paper-tape readers and punches, card readers and punches, magnetic tape stations, and high-speed line printers. Typically, eight magnetic tape stations, four card readers, a card punch, and four line-printers can be handled by a Central Processing Unit (CPU) equipped with four control units. Moreover, different combinations of peripheral equipments or additional equipments may be obtained by adding control units. Peripheral devices such as flexowriters, which transfer data a word or less at a time instead of blocks of words do not require control units.

Standard software packages have been developed and are available for the RCA 4100 Series Computers. These packages currently include two symbolic language assemblers, a library of scientific com-

*Recently renamed Aerospace Systems Division

putation subroutines, input-output subroutines, floating-point subroutines, conversion subroutines, and debugging and maintenance programs. The following descriptions apply to the RCA 4102 Computer; variations relating to the RCA 4101 and the RCA 4103 are described as they are encountered.

MACHINE ORGANIZATION

The simplicity of the machine organization can be seen by reference to Fig. 1. The entire CPU centers around the 30-bit internal data-transfer bus; all register-to-register transfers within the machine are made via this bus. Most of the registers shown in Fig. 1 perform at least two functions; this partly accounts for the small amount of hardware required (less than 3,000 transistors for a 4,096-word machine, including core memory).

The *instruction register* staticizes the operation fields of the current instruction. The content of this register is decoded to implement the micro-operations necessary to perform the instruction. For input-output operations, part of the instruction-register content addresses peripheral equipment and controls sub-operations in the peripheral equipment.

The *flag register* stores a bit for each priority level which must be serviced. Individual priority bits in this register are set by external devices demanding attention, or by internal programs signalling a desired transfer to another program. The most-significant set bit in this register determines the currently-active program.

The *demand register*, connected to bits 23 through 26 of the bus, contains the binary code for the priority currently active, and supplies part of the address of the instruction-location counter for the currently-active priority. This register is also used to address the external peripheral equipment channel of the corresponding priority number.

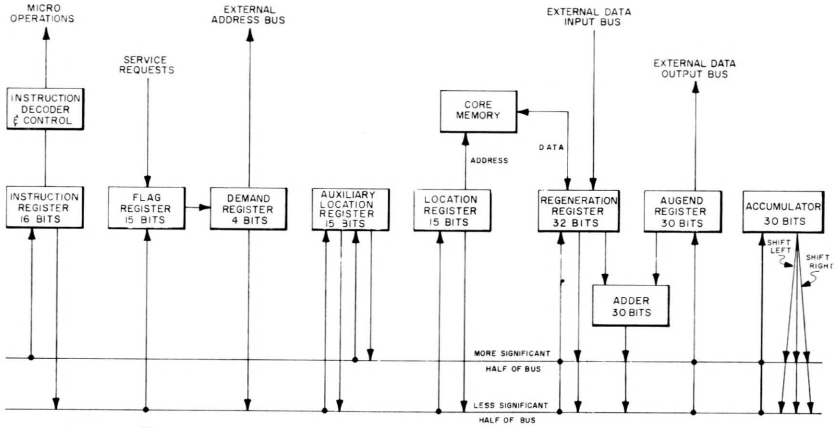


Fig. 1—RCA 4102 block diagram.

The memory used in the RCA 4100 Series Computers is a coincident-current magnetic-core memory with a 5.5-μsec read-write cycle. Standard memory sizes are 4,096 words expandable to 8,192, and 8,192 words expandable to 16,384. The RCA 4103 Computer, which is a special modification of the RCA 4102 to suit a particular application, has a 32,768-word memory. The additional addressing capability is obtained by modification of the operation field of the instruction word. The word length for the RCA 4100 Series Computers is 30 bits; however, two additional parity-check bits are included in the memory system to provide an odd parity check on each half-word for all core-storage transfers.

TABLE I—RCA 4102 Operations

Mnemonic	Instruction Name
ADD	*Add*
ADM	*Add Magnitude*
ADR	*Add and Replace*
ALA	*Shift AC Left, Algebraic*
ALC	*Shift AC Left, Circular*
ALL	*Shift AC Left, Logical*
ALX	*Add L to Index*
AMR	*Add Magnitude and Replace*
ANA	*AND to AC*
ANR	*AND and Replace*
ARA	*Shift AC Right, Algebraic*
ARC	*Shift AC Right, Circular*
ARL	*Shift AC Right, Logical*
DIV	*Divide*
DVH	*Divide Half-Word*
ERA	*Exclusive φR to AC*
ERR	*Exclusive φR and Replace*
FLS	*Flag Set*
HLT	*Halt*
IφC	*Input Output Control*
JAC	*Jump on AC*
JIX	*Jump on Index*
JSX	*Jump and Set Index*
LDA	*Load AC*
LDφ	*Load Output Word*
LDX	*Load Index*
LDZ	*Load Zero*
MPH	*Multiply Half-Word*
MPY	*Multiply*
NφP	*No Operation*
φRA	*φR to AC*
φRR	*φR and Replace*
φRS	*φR to Storage*
PAX	*Place AC in Index*
PXA	*Place Index in AC*
SBM	*Subtract Magnitude*
SBR	*Subtract and Replace*
SKP	*Skip*
SMR	*Subtract Magnitude and Replace*
STA	*Store AC*
STI	*Store Input Word*
STX	*Store Index*
STZ	*Store Zero*
SUB	*Subtract*
TST	*Test Storage*
XAS	*Exchange AC and Storage*

The primary function of the *location register* is to hold the word address for all core-storage transfers.

The auxiliary location registers serves as an auxiliary address register during address modification; this occurs for indexing and branching operations.

The auxiliary location and location registers, together, perform as a multiplier-quotient register during *multiply* and *divide* operations. The auxiliary location register stores the more-significant half of the multiplier during multiply operations, and the location register stores the less-significant half. The auxiliary location register also stores the more-significant half of the quotient during divide operations, and the location register stores the less-significant half.

The *regeneration register* staticizes the core-storage word addressed by the location register. All transfers to or from the core storage take place through the regeneration register. The regeneration register also provides one input to the adder during logical and arithmetic operations, and acts as a slave to the accumulator during shift operations. Additionally, the regeneration register serves as an input register and all computer inputs enter this register.

The accumulator stores one of the initial operands and the final result of most arithmetic, logic, and shifting instructions. This register also provides complementing and shifting capabilities.

The *augend register* provides one of the inputs to the adder and also holds the multiplicand and the divisor during multiply and divide operations, respec-

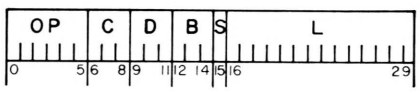


Fig. 2—RCA 4102 instruction word.

JOHN A. CORNELL received his education at the Universities of Wisconsin and Pennsylvania and has 18 years of engineering experience, including 12 with RCA. He has designed laboratory and commercial test equipment, medium-power radio transmitters, radar equipment, radio and television broadcast-station equipment, and analog and digital computers. At present, he is Manager of the Digital Development and Design Engineering Group of the Data Systems Division in Van Nuys, California. This group, responsible for research, development, and design projects in the field of military data processing, has designed several large-scale digital computers for BMEWS, DAMP, TRADEX, and other similar systems. Mr. Cornell is a Senior Member of the IEEE.

tively. The augend register is connected to the computer output bus, and all output words leave the computer through this register.

The adder is not a register but a gating network which continually generates either the algebraic or the logical (modulo two) sum of the contents of the augend and the regeneration registers.

Not shown in Fig. 1 are the index registers and the instruction-location counters, since these registers are located in core storage. Each of the 16 priority levels has its own set of six index registers, an instruction-location counter, and a work space. Use of core-storage locations rather than the more conventional flip-flops as registers results in a significant reduction of the amount of hardware required. Moreover, priority interrupts are simplified because most information in use by an active program (all but the accumulator content) is already in core storage at the time of the interrupt; thus, complicated executive routines are not required to interrupt, hold, and re-initiate programs.

WORD FORMATS

All information in the RCA 4100 Series Computers is handled in parallel, binary words of 30 bits each. The data word is interpreted as a binary fraction using two's complement notation. The range of numbers that can be represented by the data word extends from -1 to $(1-2^{-29})$ (octal representation from *4000000000* to *3777777777*, respectively). The octal representation of the number 0 is always *0000000000* and never has a negative sign. The numbers outside this range are handled by appropriate scaling. The sign bit in a positive number is always *0*, and the sign bit in a negative number is always *1*.

Words used as instructions are interpreted according to the format shown in Fig. 2. The *OP*-code field, the *B* field, and the *L* field comprise a conventional instruction format found in single-address machines with indexing capability. The *OP*-code field (located in bits 0 through 5) controls the execution of 46 different primary instructions. Nine of these instructions are word transmission instructions, seven are logic instructions, six are shift instructions, twelve are arithmetic instructions, nine are control instructions, and three are input-output instructions. Table I shows the set of operations available with the RCA 4102. The *L* field, comprised of bits 16 through 29, represents the core-storage address of the operand to be fetched and operated upon. The *B* field specifies one of six index registers or the instruction-location counter. The sum of the number in the *L* field and the number contained in either the specified index register or the instruction-location counter forms the effective address of the operand. The effective address is the address of the word actually used in the execution of the instruction. If the *B* field specifies the instruction-location counter, as in relative addressing, the sum of the instruction-location counter content and the content of the *L* field becomes the effective address. Most RCA 4100 Series Computer instructions are indexable.

TABLE II—Normal C-Field Interpretation

C Value	Branch Condition	Subsidiary Function
0 (*D* even)	*No jump*	None
0 (*D* odd)	*No jump*	*Do not set Overflow Indicator*
1	*Initial Index Value = 0*	*Decrement Index*
2	*No jump*	*Decrement Index*
3	Final $C(AC) \geq 0$	None
4	Final $C(AC) > 0$	None
5	Final $C(AC) \neq 0$	None
6	Final $C(AC) = 0$	None
7	Final $C(AC) < 0$	None

A secondary instruction, specified by the *C*, *D*, and *S* fields, is included with all primary instructions. The *C* field, in most instructions, specifies conditions for branching and/or subsidiary functions that may be performed concurrently with the operation specified by the *OP*-code field. The normal branching conditions and subsidiary functions applying for most instructions are shown in Table II. The *C* field in input-output operations specifies the location of a peripheral device rather than a branch condition.

The *D* field determines the destination address if the branch condition specified by the *C* field is met. When this occurs, the content of the *D* field is added to the content of the instruction-location counter to yield a destination address. The value of *D* may range from zero through seven. When *D* is *0*, the current instruction is repeated if the condition specified by *C* is satisfied. Other *D* values, with a satisfied normal *C* condition, cause the computer to jump back *D* locations in memory to obtain the next instruction word. When used in conjunction with certain operation codes, the secondary instruction fields operate in other than the normal manner.

The *S* field is called the *suicide* bit. When *S* is *1*, the active priority flag is reset, thus ending the active program and starting the program with the next highest priority.

PRIORITY-INTERRUPT SYSTEM

The priority-interrupt feature of the RCA 4100 Series Computers serves a multiplicity of programs on a priority basis. The primary purpose of the priority interrupt system is to eliminate the need for program-controlled scheduling and status testing of input-output transfers.

The complete set of indicator flags is examined, simultaneously with the execution of each instruction, to determine which demanding program has the highest priority (Fig. 3). When the currently-active program has the highest priority of those demanding attention, the program proceeds without interruption. Otherwise, the system automatically transfers, at the completion of the current instruction, to the highest demanding priority.

During the priority-interrupt cycle, the machine logic ensures that the address of the next instruction of the interrupted priority is saved in its instruction-location counter. The only programming constraint is that each program affecting the accumulator content is responsible for restoring the original value. When all higher-priority demands are satisfied, control returns to the interrupted program, and the latter proceeds oblivious of the interruption. In this way, the priority-interrupt feature enables the computer to interleave a number of different programs.

Priority Assignment

Input-output routines for the RCA 4102 are generally short, and frequently consist of single instructions which do not affect accumulator content. After an input routine transfers data to memory, suitable programs are called on to process the new information. The various input-output and data-handling routines are assigned different priority levels. The priority-interrupt system automatically activates the proper program to process the new data transferred to the memory by other programs; therefore, the programmer need not perform periodic tests for the arrival of data.

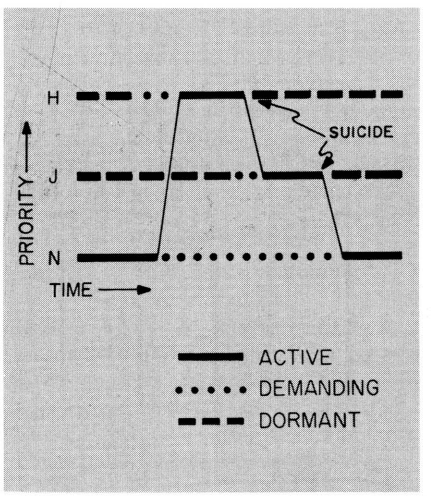


Fig. 3—Program interrupt example.

Because of these considerations, assignment of relative priority among the various input-output programs and devices is principally based on the allowable waiting time between data transfers. Computing programs are assigned lower priorities than those of input-output programs and computing priorities may be assigned arbitrarily, within this constraint, by the user. The lowest priority program, automatically executed when the flag register is cleared, is usually a self-check routine, which automatically exercises and tests the computer during idle time.

TIMING

Instructions in the RCA 4100 Series Computers are executed by sequences of micro-operations capable of clearing registers or transferring data to or from the bus, etc. The execution time of each instruction is divided into a number of status cycles. Each status cycle is subdivided into four, six, or eight ½-μsec time slots. A typical memory cycle (Fig. 4) consists of six time slots. During the first time slot, the address is placed in the location register and a read command is issued. Data are transferred from core storage to the regeneration register before the start of the third time slot. When the transfer is complete, a read echo signal is delivered by the memory to the computer control and timing circuits. During the time between the second time slot and the read echo signal from the memory, the computer clock hesitates, if necessary, to maintain synchronism. The read echo signal starts the third time slot. In the fourth time slot, a word to be placed into core storage is placed in the regeneration register (it could be the word already there as a result of the read operation) and a write command is issued to the memory.

When writing is complete, a write echo signal is delivered by the memory to the computer control and timing circuits. The computer clock again hesitates after the sixth slot, and is started by receipt of the write echo signal. All of the time slots shown in Fig. 4 are used in the execution of various micro-operations in addition to those which control memory access. A register can be set, cleared, or have its content transferred to another register in one time slot. Processing of a word read from memory to the regeneration register takes place during the third, fourth, fifth, and sixth time slots for all short instructions. Subsequent status cycles are required for processing of the word for long operations such as shifting, multiplying, and dividing.

As shown in Fig. 5, all RCA 4102 instructions start with a memory cycle to fetch the instruction. Prior to the start of the memory cycle, the address of the instruction is in the auxiliary location register. During the first time slot of this cycle, the address of the instruction is placed in the location register and a memory read cycle is started. This transfers the instruction from core storage to the regeneration register. Then the address bits of the instruction are transferred to the auxiliary location register and the operation bits are transferred to the instruction register to be decoded.

If indexing is required, as determined by the content of the *B* field in the instruction, another memory cycle occurs. During this memory cycle, the called-for index register is delivered to the regeneration register and the base address which is in the auxiliary location register is transferred to the augend register. The content of the index register is added to the base address in the adder and the sum is delivered to the location register to form the effective address. If indexing is not required, this cycle does not occur.

A third cycle (or second, if indexing is not required) takes place whenever an operation is to be performed on an operand located at the effective address. During this cycle, the operand is transferred from core storage to the regeneration register and the operation called for is completed (for short operations). For MPY, MPH, DIV, DVH, and XAS operations, this cycle is divided into eight time slots.

For long operations, such as multiplication, division, or shifts, additional status cycles are required. These status cycles do not require memory interrogation and are subdivided into four or eight time slots, depending upon the operation required. From zero to 16 of these additional status cycles may be required depending upon the instruction.

The execution of the operation called for by the operation field of the instruction is followed by a memory cycle in which the instruction-location counter is updated. This cycle is common to all instructions. The address of the memory location which contains the instruction-location counter is formed in the location register and a memory read cycle is started, delivering the instruction-location counter content to the regeneration register. Simultaneously, the branch control is set if the conditions specified by the *C* field and the result of the operation are met.

The augend register is set with the proper decrement as determined by the *C* and *D* fields of the instruction. The modified instruction location formed by the adder is transferred to the auxiliary location register, then returned via the regeneration register to core storage. Toward the end of this status cycle, the priority-interrupt flags are examined to determine the highest set priority. If the highest set priority is equal to the content of the demand register, then the program proceeds uninterrupted and the next status cycle, which is the first cycle

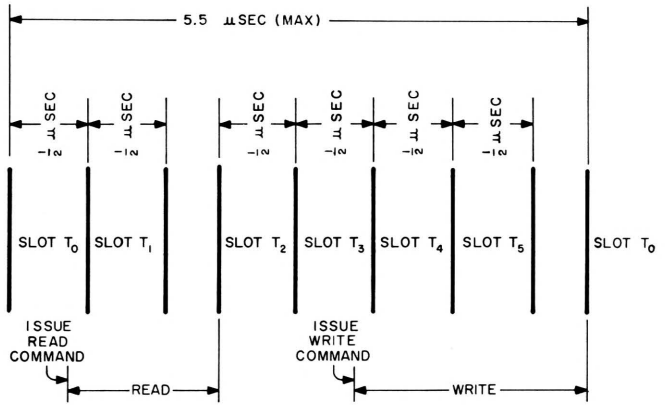


Fig. 4—Basic memory-access status cycle.

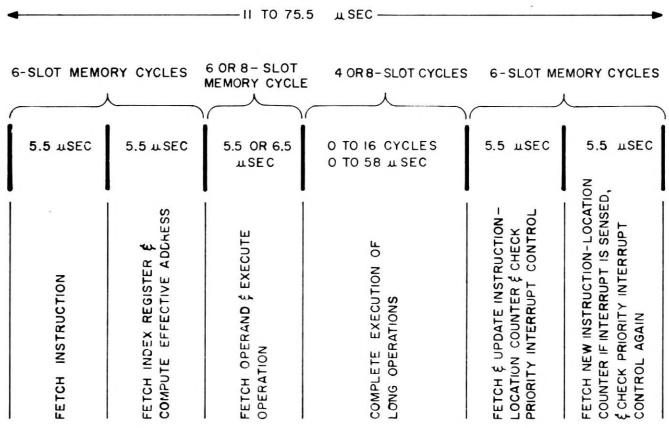


Fig. 5—Basic instruction status-cycle sequence.

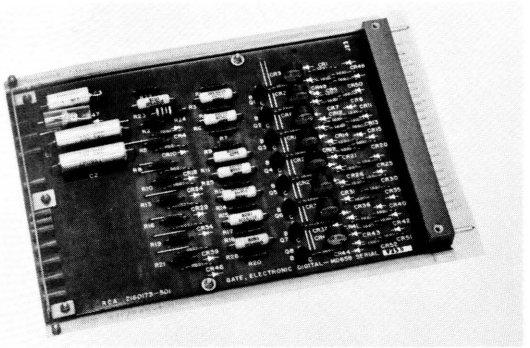


Fig. 6—RCA 4100 series logic module.

of the next instruction, proceeds as already described. Otherwise, another status cycle takes place which transfers to the auxiliary location register the content of the instruction-location counter associated with the interrupting program. Then the machine proceeds with the next instruction.

INPUT-OUTPUT

External devices, such as data links, radars, analog-digital converters, telemetry channels, and teletypes, requiring attention at random and unpredictable intervals, are connected to the RCA 4100 series CPU's by means of the regeneration register and the augend register with little or no intervening buffering or control logic. Each channel is assigned its own priority and the computer will handle up to 14 such channels. More than one device (up to eight) may be connected to a single priority level, provided only one at a time requires servicing. Each device generates a service-request signal whenever it is ready for the transfer of a word to or from the CPU. This service request sets a preassigned priority flag in the computer. The routine executed in this priority contains a single-instruction loop to perform the required data transfer. The single instruction (STI or LDϕ) is executed once in response to each service request, transferring one word to or from memory, and returning control, by means of the *S* field, to the computing routines. Using an index register to tally the service requests, the program can automatically notify another routine whenever a predetermined number of words has been transferred. This automatic notification relieves the program of any need for schedule control or periodic status-testing in the servicing of peripheral devices.

When several devices or channels are connected to the computer, each responds only to STI and/or LDϕ instructions executed in the priority level assigned to the device. The active-priority code, held in the demand register, thus serves as a peripheral channel address for input-output communication.

Other devices, such as flexowriters and perforated-tape equipments, communicate single computer words or less in response to each programmed control command. Communication between the RCA 4102 CPU and such a device is similar to that described above. The distinction lies in the need for a command from the CPU to control the operation of the device. This command is encoded into the *C* field of the STI or LDϕ instruction which performs the data transfer. For example, an LDϕ instruction, which orders the transfer of data to the flexowriter, simultaneously directs whether the data should be punched or typed, or both. After each service request has been processed, and when the device is ready for the next data transfer, the device sets its priority flag, automatically notifying the CPU. Several similar or dissimilar devices can be time-shared on a common priority level, provided no ambiguity or conflict arises in the interpretation of the *C* fields of the input-output instructions. As many as eight input or eight output devices can thus be serviced by each priority level. In this case, the *C* fields of the input-output instructions become a subsidiary peripheral address, supplementing the active-priority code for selecting and controlling peripheral devices.

Input-output devices which communicate blocks of data to and from the CPU, such as magnetic tape stations, line printers, and punched card equipments, use control units between the devices and the CPU for efficient communication. A number of similar block-transfer devices may communicate with the CPU through a single control unit. Each such control unit is connected to a specific CPU priority level, in which all data transfers (STI and LDϕ) for this control unit are executed.

Control commands, for directing the action of these block-transfer control units and their associated devices, are issued from the CPU by means of an IϕC instruction. The IϕC instruction, which may be executed in any priority level, transmits to the peripheral equipment a 15-bit control command which identifies:

1) Priority level of the control unit addressed
2) Specific device to be selected (if necessary)
3) Action to be performed
4) Type of data block

Once the action has been initiated, and for the duration of the specified data block, any required data transfers are performed as already described for word-transfer devices.

At present, control units have been designed and are available for the following: the RCA 381 high-data tape group, an IBM compatible tape station, a high-speed line printer, a card punch, and a card reader. These control units are module nests or groups of nests housed in one, two, or three-rack complexes called input-output equipment control cabinets. The cabinets provide a standard operating environment including common power supplies, blowers, cable assemblies, and common interface logic for several control units. A different control unit is required for each type of block-transfer device; however, one control unit will handle several devices. For instance, only one control unit is required for all six tape stations in the RCA 381 high-data tape group; the line-printer control unit will control up to four line printers; and the card-reader control unit will control up to four card readers.

LOGIC IMPLEMENTATION

The logic building block of the RCA 4100 Series Computers is a *nor* gate—a diode-coupled, diode-biased, grounded-emitter amplifier, combining the advantages of high fan-in, high fan-out, and restandardization of signal levels. Use of this type of gate results in noise-insensitive operation and simplicity of maintenance. High fan-out is achieved, with moderate power drain, through the use of silicon diodes as a nonlinear base resistor. The high storage of these "stabistor" diodes and the polarity of the logic diodes provide compensation for transistor charge storage effects. The gate output is clamped to speed output signal-level transitions and to provide a uniform signal level throughout the system. The worst-case signal delay through an *and/or* pair of these *nor* gates is less than 50 nsec. This standard gate is the only logic element used throughout the entire CPU. The gates are packaged eight to a module board and the entire CPU uses only 12 different types of module boards, including all of the memory boards. Over 70 percent of the module boards used in the CPU are of the type which holds the standard logic elements. A photograph of this board is shown in Fig. 6.

ACKNOWLEDGEMENT

The RCA 4100 Series Computers were the original work of a team of engineers headed by W. E. Woods, a Staff Engineering Scientist of the Data Systems Division, Van Nuys, California. While a great many engineers participated in the successful execution of this program, the primary contributions were made by H. Zieper, W. Helbig, and C. S. Warren.

RCA COMPUTER RESEARCH

... Some History, and a Review of Current Work

Dr. J. A. RAJCHMAN, Director
Computer Research Laboratory
RCA Laboratories
Princeton, N. J.

DR. JAN A. RAJCHMAN, Director, of the Computer Research Laboratory, RCA Laboratories, Princeton, N. J. developed the magnetic core memory system that is now the standard information-storage device in modern computers, and is responsible for pioneering contributions to the development of the electron multiplier tube. Dr. Rajchman received the Diploma E.E. (equivalent to an MSEE) in 1934 and a Doctor of Science degree in 1938 from the Swiss Federal Institute of Technology. His entire professional career has been with RCA. A student engineer at Camden during the summer of 1935, Dr. Rajchman a research engineer in 1936. In 1942, he was transferred to the RCA Laboratories in Princeton. He became Associate Director, Systems Research Laboratory, in 1959. He assumed his present position in September 1961. His first field of work was electron optics. During World War II, he was among the first to apply electronics to computers. Later, he worked on the betatron. After the war he resumed work on computers and developed the selectron, and shortly thereafter, the magnetic core memory. He contributed many magnetic switching circuits, the transfluxor, the magnetic plate memory, and magnetically controlled electro-luminescent display panels. Dr. Rajchman is co-recipient of the 1947 Levy Medal of the Franklin Institute for his work on the betatron. He received the Liebmann Memorial Prize for the year 1960 for his contributions to the development of magnetic devices for information processing. Dr. Rajchman is a holder of more than 90 U.S. Patents and the author of many technical papers. He is a Fellow of the IEEE.

RCA computer research effort is centered in the Computer Research Laboratory of the RCA Laboratories, Princeton, N. J. This paper first traces the beginnings and progress of that research as far back as 1939. Perhaps surprising will be the significance of RCA contributions to early computer development. While RCA was not the first to commercially exploit the computer, it was actually a major pioneer in computer research. The present work of the Computer Research Laboratory is then reviewed with attention to memory speeds and storage capacity, superconductive memories, content-addressable and read-only memories, logic switching circuits, input-output devices, and computer theory.

THE COMPUTER RESEARCH LABORATORY of RCA Laboratories, Princeton, is concerned with conception and realization of computer systems, subsystems, and components, as well as with computer applications. It comprises more than fifty scientists and engineers in a field of growing importance to RCA. Established as a separate entity in September 1961, it is now one of the six main laboratories of the David Sarnoff Research Center.[1]

SOME HISTORY

Computer research started at RCA as far back as 1939. The impetus was the dire need to replace the slow and inaccurate "directors" then used for controlling anti-aircraft guns by rapid and accurate controls made necessary by the fast planes of the day. There was real vision on the part of the military to perceive then that such complex calculations could be achieved by electronics. After all, in those days electronic pulse counters used in cosmic-ray research were the closest thing to electronic computers. In the early war years, a small group at RCA in Camden took this extremely interesting challenge and worked out many of the fundamental principles of electronic computers still in use today. The two main approaches, analog and digital, were considered. (At that time these were called *continuous* and *discrete.*) The analog approach yielded the first practical solution to the fire-control problem, and an actual computer was used in combat.

In the digital approach, of direct interest here, great progress was made in the concepts and prototype circuits, but the technology of those days did not permit the practical realization of such computers for field use. It was realized then that computations in the binary code are the simplest (although manipulation of decimal numbers by properly coded *on-off* signals was also considered). All the elements necessary for a complete binary computer were conceived and incorporated in experimental prototypes. These included registers of flip-flops, shift registers, counters, adders and subtracters, multipliers and dividers, and generalized code converters. Because the vacuum tube was then the only device that provided gain and a switching threshold, a great deal of mixing of signals by resistive networks was used for logic switching. Today, this would be called *threshold logic* or *majority logic.*

For instance, one example of such networks was in connection with carries occurring in binary arithmetic when the sum of the digits exceeds one —the equivalent of exceeding nine in the decimal system. The chain of carries that can run the full length of the number to be added was obtained by resistively coupled circuits. Multiplication was produced by successive additions made in a single adder, alternated with register shifts, as is generally done today. But interestingly enough, an "integrated" multiplier was also made in which all additions were made simultaneously in a single DC-coupled circuit including a number of adders. Fig. 1 shows such a multiplier for two 6-digit numbers (up to 64 by 64).

Another device based on resistive or other high impedance coupling was a code converter that we then called an *arbitrary function generator* and would in some instances today call a *read-only fixed memory*. The device consisted of numerous resistances located at selected locations of a regular array that were connected by columns and rows. When the column wires were energized according to the input code, a certain unique row was selected. The selected row turned on a tube whose output excited a corresponding row wire in another array in which the pattern of the locations of the resistances determined the desired fixed code conversion. The potentials on the column of this second array were indicative of the output. Fig. 2, shows part of a 1942 experimental array which comprised more than 10,000 resistances, and which converted a 7-bit code into a 13-bit code and operated in about 20 μsec.

We soon realized that complete computers could be built for solving intricate equations at high speed, including those of the original fire-control problem, by using a sufficient number of regular radio tubes. However, the tremendous number, running into many thousands, seemed to make the approach impractical. We then developed a special vacuum tube, the *Computron*, in which the many logic functions necessary in an adder and multiplier were obtained by direct connections between electrodes within the tube itself.

About that time, the need became acute for much more rapid computing of ballistic tables for guns because computations by existing methods lagged the constant improvements in weapons. In fact, the need was so pressing that the government decided to undertake the development of an electronic computer despite the requirement for an estimated 20,000 to 30,000 tubes. The project was considered by RCA, but actually was undertaken by the University of Pennsylvania and resulted in one of the first electronic computers, the ENIAC. However, since the work that had been done at RCA was by far the most advanced at the time, we were asked to consult on the project. Many of our ideas were actually the basis of the circuits used, particularly the resistive function generator.

Early Memories

The next chapter in RCA's computer research dates back to the period immediately following the end of the war. At that time, John Von Neumann of the Institute of Advanced Study in Princeton put forward a far-reaching proposal for building a universal machine which would be based on the use of a stored program. The idea was to have a central memory in which numbers could be stored and from which they could be extracted, and a means to solve the problem through a programmed series of coded instructions which themselves could be stored in the memory. A single arithmetic processor and means for input and output completed the computer. By writing various programs, the same machine could be made to solve a great variety of problems; in fact, any problem whose solution could be obtained by a logical sequence of computations. The idea of the stored program was born gradually while the ENIAC and some of its successors were being designed. Von Neumann crystallized all these ideas into an extremely elegant proposal that can well be considered as being the foundation of the present art.

What was needed above all, of course, was an electronic memory and this is what RCA Laboratories undertook to provide. The year was 1947. The memory had to have a so-called *random access*— that is, be able to accept information identified by a coded address and deliver it on the basis of that address. This allowed the information to be stored and retrieved in any order without any sequential scanning of the entire memory as was necessary in the case of some proposals at the time. We conceived a purely "integrated" storage tube which we called the *Selectron* (Fig. 3). The tube had two orthogonal sets of parallel wires by which a flow of electrons could be controlled so as to pass through any desired selected window formed by the wires. The electron current at that window bombarded a metallic element which stored information in terms of one of two stable potentials. *This tube was the first truly digital random-access memory.* It had a storage of 256 bits and an access time of about 15 μsec. At the time, this tube was the ultimate in intricate vacuum-tube technology, and although it had nearly ideal performance, it was relatively expensive. RCA made a few hundred Selectrons for a machine made at the Rand Corporation in California. This machine was patterned after the one at the Institute of Advanced Study at Princeton University. The tubes operated satisfactorily for about five years until the machine was made obsolete by commercial models with magnetic memories.

Magnetic Memories

In the late forties the memory still presented a problem. A variety of beam-deflected cathode-ray electrostatic storage tubes were developed, the simplest in structure and the most widely used being the so-called Williams tube. However, we were looking for a more reliable and less-delicate memory system.

Fig. 1—Digital multiplier (1940 vintage) for numbers up to 64 by 64 = 4096.

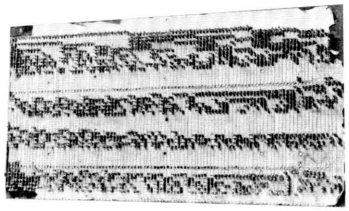


Fig. 2—Resistive matrix for code conversion, or early "read-only" memory (1942).

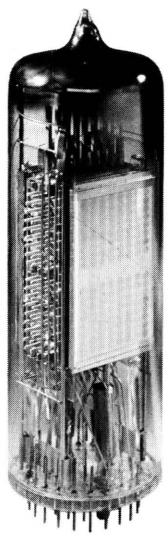


Fig. 3—Selectron memory tube, 256 bits.

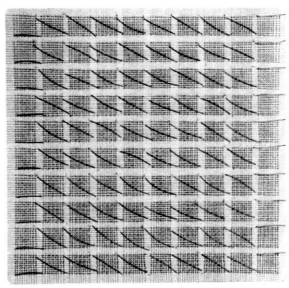


Fig. 4—Myriabit core memory 10,000 cores 0.054 OD, 0.034 ID.

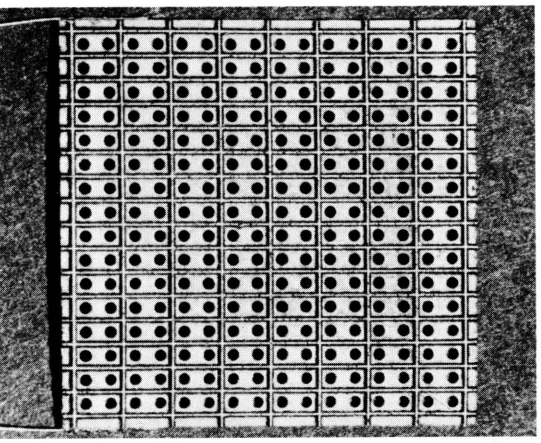


Fig. 5—Ferrite aperture memory plate; 256 bits, ¾" square.

This epoch marks a most important step in computer research at RCA Laboratories: the conception of the magnetic memory. It proved to be far more reliable than the storage tubes, generally extendable to larger capacities, capable of working faster, and also less expensive. At approximately the same period, a group at MIT also worked on magnetic memories.

The principle of magnetic memory operation depends on storing each bit in terms of the direction of remanent magnetization in magnetic cores, which may be shaped as rings. In a sense the core "remembers" in which direction it was excited last. The read-out is obtained by attempting to change the state of the core by sending a current through it in a given direction. If the state changes, the induced voltage due to the change of flux provides the read-out signal and the knowledge permits restoration of the core to its original state. This procedure is the so-called *destructive read-out* mode. The selection of a core among many in an array is obtained with a relative economy of driving circuits by making it necessary that two or more currents energize the core in coincidence for it to switch. This requirement presupposes that the cores have a sharp and well defined threshold of switching—that is, that they are made of a material with a substantially rectangular hysteresis loop. Special alloys of metals having such loops had been developed during the war for magnetic amplifiers and permitted us to make the first magnetic-memory prototypes. But these were delicate, expensive, and relatively slow in switching.

Fortunately, ferrites had been developed at RCA Laboratories for use in transformers and cathode-ray-tube deflection yokes, and proved to be fast switching, rugged, and inexpensive. The problem was to obtain ferrites with rectangular loops. These were obtained from the Materials Section of RCA Laboratories in a relatively short time through the synthesis of ferrites with the proper compositions—with magnesium and manganese activators. Methods were then developed to automatically mold and test small cores. The smaller the core, the less driving current it required and the faster it switched—but, of course, the more difficult it was to handle and wire. The compromise size, an outside diameter of 0.054 inch, that was chosen for the large experimental prototype turned out to be the reduced size reached in production only three years ago. The prototype, dubbed the *Myriabit* (Fig. 4) had 10,000 cores, required several racks of vacuum tubes, and operated in a cycle time of about 25 μsec. (A description of the Myriabit was published in 1953.) Since then, the ferrite-core memory has been highly perfected. The advent of the transistor made it possible to greatly reduce the size and power of the associated circuits. Memories with hundreds of thousands to millions of bits operating in a few microseconds are now standard products.

The fact that in the memory the core is used as switching element, in addition to its main storing function, induced us to consider it as a switch for pure logic functions. Array and combinatorial switches were developed, some of which are still used for addressing memories. The magnetic shift register, developed elsewhere in the mid-forties, was greatly improved. The general possibility of an all-magnetic computer with only a few tube drives was fully demonstrated, but the spectacular advent of transistors provided a much better technique for digital switching. Incidentally, this is an example of the abundance of technical means that are at our disposal today. If it were not for the transistor, all present computers would probably be magnetic. As it happens, there is a revival of interest in magnetic switching for space use where immunity to radiation damage and great reliability are of paramount importance.

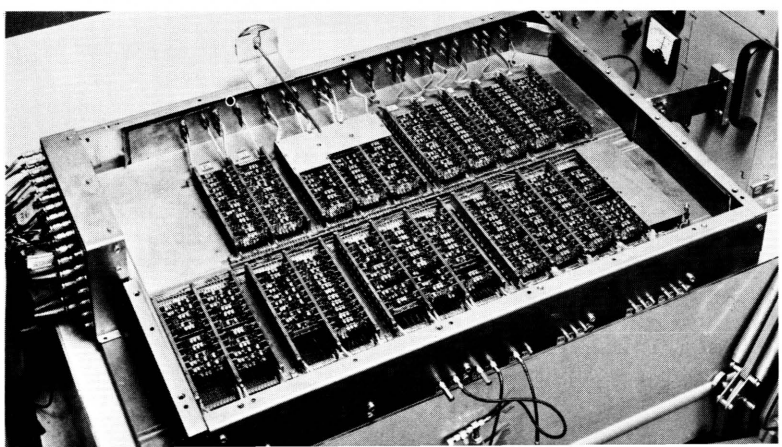
The next step in magnetic memories was toward integrated fabrication in an effort to avoid the necessity of making and wiring each core individually. A plate of ferrite was molded with an array of holes and covered with a metallic coating so patterned as to provide a winding linking all holes. The plate had 256 holes in an area ¾ inch x ¾ inch (Fig. 5). The apertured plate was a significant advance and was the precursor of recent successes in micromagnetics, and at the time, of the transfluxor.

The transfluxor[2] includes a square loop core with two or more apertures. It can store digital and analog information and can be read off nondestructively. The device has found a great variety of specialized applications. It has also been used in all-magnetic logic consisting of circuits with transfluxors and wire only. Memories with nondestructive readout as well as content-addressable memories were made of transfluxors. The transfluxor was the basis of the first electroluminescent display array, a description of which was published in 1957.

Ultra Fast Computers

To complete the historical background leading to our present position, mention must be made of a relatively large Navy-sponsored research program which RCA undertook in mid-1957. This program was aimed at increasing the speed of computers from 1 Mc to 1,000 Mc. The Laboratories played the major role in the project in its early stage, while RCA Electronic Data Processing (EDP) & Materials Division (SC&M) did most of it during the latter part of the program. Many phenomena were canvassed as possible candidates for extremely fast devices, including such diverse effects as vacuum electronics, ionization in semiconductors at very low temperatures, superconductivity, magnetic switching and effects in thin films. However, substantial effort was devoted to only two approaches: The first depended on the use of a variable-capacitance diode in phase-locked oscillator circuits operated at microwave frequencies. The second and more successful approach was that of the tunnel diode. Shortly after the publication by Esaki of the tunneling

Fig. 6—Logic subsystem of tunnel diodes with stage delay of 1 nsec.



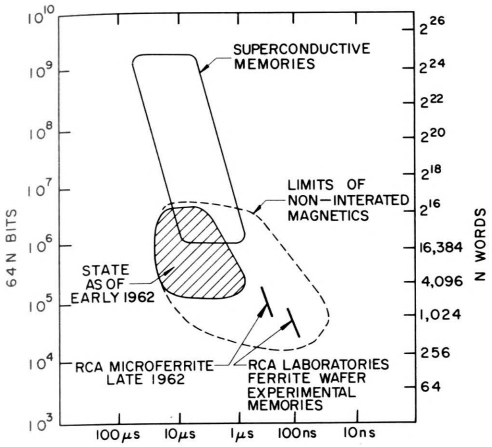


Fig. 7—Storage capacity and cycle time of various memories.

diode effect, work was undertaken at the Laboratories on tunnel diodes. Tunnel diodes operating at the speed of interest were developed in a very short time. Since the tunnel diode is a two-terminal device with negative resistance, entirely new circuit concepts were required to capitalize on the gain inherent in the negative resistance and to provide directionality to the information flow. An exhaustive study of possible circuits was made. A certain type of DC-coupled circuit was finally selected and with it EDP has succeeded in building a prototype subsystem demonstrating the technical feasibility of both tunnel diode logic and memory. The logic stage delay is about one nanosecond, and the essential logic elements can be operated at 300 megapulses/sec, (Fig. 6). This is essentially within the original goal. The project gave birth to the modern high-speed digital circuitry with times expressed in nanoseconds (10^{-9} second) rather than microseconds. It was also largely responsible for such innovations in instrumentation as the fractional-nanosecond sampling oscilloscope. The project was of great help to SC&M in developing commercial tunnel diodes.

This rather lengthy historical introduction is given here because many of the early efforts of RCA in computers are not generally known throughout RCA. While RCA was not the first in the commercial exploitation of electronic computers, it was actually a pioneer in the early research work.

THE LABORATORY TODAY

Today the romantic period in which the challenge was to discover "how to do it" is past and so are the last decade's miseries of computers failing every few minutes. Instead, there is a solid $3 billion industry in which RCA has a strong belief and heavy commitment. The competition is severe. Under these circumstances, *what should be the research program of the Computer Laboratory?* The following is a brief account of the goals chosen and of the progress toward their realization.

Memories

The amount and the accessibility of stored information in computers plays the same crucial role as does the signal-to-noise ratio in communications. The characteristic of the memory of a computer determines the capability of the whole system. This is not surprising, since in the universal stored-program computer of today, all information as to *what to do, how to do it,* and *what data to do it on,* is stored in the memory. Every step involves access to the memory. Obviously, large storage capacities and fast access are essential. The trend of the last few years only accentuates the importance of the memory. Invariably it turns out to be easier to simulate almost any logical task by programming within the memory, rather than building special hardware for the purpose. For his patterns the computer architect has no choice but to prefer the stored states of well-ordered cells of the memory rather than some chaotic wiring of physical parts.

As already mentioned, the magnetic core array, resulting from our research of about a decade ago, has provided computers with a really workable memory that is now the mainstay of the art. We believe that equally significant steps in the art are possible by further innovations in memory, which still represents the most important and fertile ground for research. For that reason, a substantial part of our effort is in that field and is likely to continue for a number of years. We are now engaged in work in magnetic, superconductive, and other types of memories.

The storage capacity and the time of access are the main characteristics of memory. These two parameters are plotted in Fig. 7, where the state of commercial ferrite core memories at the beginning of 1962 is illustrated by the crosshatched area. Maximum capacities are a few million bits and minimum access is about ¾ of a microsecond. We have been working to improve upon these limits, as follows:

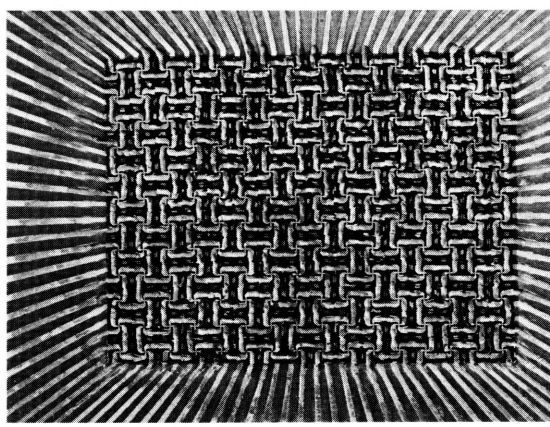
Speed

The speed of a core memory depends strongly on the size of its elements. The smaller the core diameter, the greater is the magnetomotive force due to the maximum drive current possible from a given transistor—and therefore the faster is the switching of the core. Also, the small bit spacings produce corresponding short delays along windings. While a number of other factors must be considered to obtain high speed, miniaturization of the elements is an absolute prerequisite. For that reason we have concentrated our research of the last few years on technologies for making memories with elements drastically smaller than is possible with the conventional molded ring-shaped cores strung on wires.

We have recently worked out a technique for making holes in ferrite by means of electron-beam drilling.[3] Very intense beams of high-energy electrons can produce in microseconds enough heat to sublimate ferrite, even though this ceramic material is very refractory. For example, 1-mil holes can be made quite easily in wafers 10 mils thick. To obtain windings, these small holes are plugged with a conducting paste, the wafer is coated with copper, and a winding is microphotoetched on the two faces of the wafer. A cluster of four holes, each with a single winding, is used and has the effect of four windings through a single hole. The wafers are assembled in a mosaic and the wafer-to-wafer connections for the entire array are completed in a single soldering operation (Fig. 8). Arrays of 16 words of 12 bits each have been operated with a cycle time of 100 nsec. Larger capacity memories would be slower; for example, one with 1024 words would have a cycle time of 150 nsec. This is illustrated on Fig. 7 for the case of a typical word of 64 bits, i.e., a capacity of 65,536 bits. Incidentally, the word length has practically no influence on access time in high-speed word-organized memories.

The Semiconductor and Materials Division, RCA Electronic Data Processing, and the Electron Tube Division worked with the RCA Laboratories on microferrite memories, and have given indispensable assistance in the above project. Concurrently with it, and as a result of it, the SC&M Memory Operation at Needham developed and in mid-

Fig. 8—Mosaic of microferrite wafers; 16 x 12.



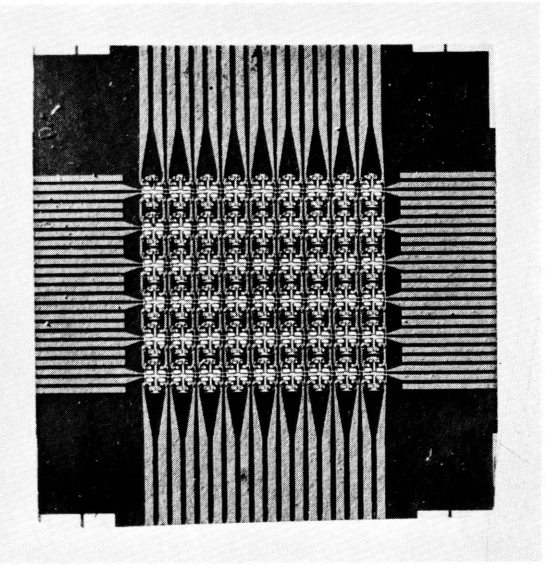


Fig. 9—FLEA memory, about ¾" square.

1962 announced a commercial microferrite memory. This memory has a cycle time of 375 nsec and is the fastest commercially available. A typical memory module may have 256 words of 36 bits each, as illustrated on Fig. 7. The memory is made of cores with 50-mil outside diameter and 10-mil inside diameter, which are metallized on their faces and on the wall of the hole. The cores are mounted on strips and the metallizations are connected so as to produce one winding. Straight wires strung through many parallel strips complete the assembly. This memory represents an important step toward integrated fabrication as only a single hand-threading operation remains.[4,5]

We are presently attempting to develop a microtechnology that provides elements even smaller than those practical with the electron-beam drilling and subsequent microphotoengraved windings, and which at the same time is completely integrated in that a single operation provides elements, windings, and interconnections. We hope thus to both improve performance and lower cost. Promising initial success has been attained.

Storage Capacity

In considering the largest storage capacity attainable in magnetic memories, one could think of the question simply in economic terms. For example, one could obtain a billion-bit memory by buying one thousand conventional memories of one million bits each. Such a memory would cost a good fraction of a billion dollars, but worse yet, the delivery time probably would be very long. This can be appreciated by considering that automatic machines making and wiring elements at the rate of one per second would require more than 30 years to complete a billion-bit memory. It is very unlikely that higher fabrication speeds with obvious parallelings could speed up the present bit-by-bit construction technique by the orders of magnitude required to make it practical. Clearly, batch fabrication or integrated techniques are the real hope.

This hope underlay our work on the apertured ferrite plate mentioned earlier. However, it turned out that the step toward integration, with 256 bits batch fabricated and requiring some hand wiring, has not yet had a great economic impact. More recently, we attempted to take a much larger step by using thin sheets of permalloy. An array of holes is photoengraved in the sheet by a technique similar to that used for the shadow mask of the tri-color cathode ray tube. The sheet is then coated with an insulating layer and a layer of copper. The copper is photoetched to produce winding and connections. The lack of a sufficiently rectangular hysteresis loop and uniformity on the sheet was overcome by the use of miniature transfluxors for each bit, instead of single cores. Sheets with 16 x 8 transfluxors were developed and a stack of 20 was built into the prototype memory (Fig. 9). This prototype, tested by the Surface Communications Division is of interest to the Signal Corps because it can operate in a wide temperature range, typically −70° C to 150° C, owing to the high Curie temperature of the permalloy. Larger sheets can be made, although this would be much easier if materials with better properties were available. At the moment, the ferrite integrated technology alluded to above seems to be more promising. In any case, a vigorous development along either of these two or some other approach is likely to provide the necessary integrated magnetic technology for the practical making of hundreds of millions of elements.

Unfortunately, integration of magnetics alone is not sufficient—the associated electronic circuitry must be integrated as well. It can be estimated that for large capacities, one semiconductor element is required for every 200 magnetic storage elements. This is a conservative estimate; many commercial systems have a greater percentage of circuitry. The several million semiconductor devices required for a billion-bit memory are two to three orders of magnitude more than the number of logic elements in today's largest computers. Thus, it is obvious that the real promise for really large magnetic memories lies in the combined integration of magnetic and semiconductor technologies. It turns out that this is also the best approach to high speed. For these reasons we have started intensive research in this area, and are considering the use of diodes as well as transistors. Some progress has been already achieved. However, it appears that superconductivity may offer an easier road to large capacity memories.

Superconductive Memories

Superconductive phenomena are essentially ideal for computer applications. Persistent supercurrents are a natural form of storage, and sharply defined thresholds between superconductive and normal states permit switching. Moreover, the technology of thin superconductive films offers the unique possibility of simultaneous miniature batch fabrication of storage elements, addressing switches, and all connections. For these reasons, about five years ago we decided to investigate this relatively esoteric phenomenon. We were soon convinced that the cost of cooling to liquid-helium temperatures makes superconductive computer systems of economic interest only if such cost can be spread over a very large number of elements—upwards of millions. On the other hand, as explained above, our magnetic-memory work showed that this was about the limit of nonintegrated magnetic-semiconductor techniques. The conclusions were obvious: work on very-large-capacity superconductive memories, and for computer logic circuits, consider only that work which is necessary in conjunc-

Fig. 10—White room and facilities for superconductive thin film fabrication.

tion with the memory. After a number of relatively complicated arrangements, a solution was found that appears to be the simplest possible: it consists of making only the minimal straight connections of the memory's schematic, and obtaining the necessary storing and switching devices incidentally, as it were, along the connections. The principle is briefly as follows.

Two perpendicular sets of parallel, suitably insulated lead strips are evaporated on top of a continuous film of tin. When an x and y strip carry a current I, the magnetic field pattern at their intersection is at 45° with respect to the strips. The intensity of the field is maximum at the intersection and diminishes gradually with distance. Consequently, there is a sharply defined region within which the field in the tin sheet exceeds a critical value and renders the sheet normal, and outside of which it does not. Within that region, it is possible to induce persistent supercurrents and change the polarity of previously induced persistent supercurrents. The final polarity obtained depends on the polarity of the primary driving currents and determines whether a one or a zero is stored. An element is switched only if the primary excitation is opposite to that which previously established its state, and only if it exceeds a certain definite threshold. The situation is thus quite analogous to that of a hysteretic magnetic element with a perfect square loop. A voltage is induced in a sense winding on the opposite side of the memory plane. No magnetic fields leak through the superconductive memory plane, other than at the selected location, because it is a perfect magnetic shield; thus, there are no disturb signals caused by half excitations.

To address the memory, the drive currents x and y are steered to the selected lines by a network of cryotrons. A *cryotron* is a superconductive gate, described by Dudley Buck in 1956, which depends on the fact that a normal resistive path can be established in a superconductive strip if a sufficiently high current is passed on a superimposed strip. In the selecting networks, the cryotrons are arranged in trees, and at every bifurcation one of the cryotrons is resistive and the other superconductive according to the value of the corresponding binary address bit. The currents are steered through the only completely superconductive path to the desired memory line.

We plan to use the superconductive memory in a semiconductor computer. The drivers for the cryotrons, the sense circuits, and the write and rewrite circuits are of the conventional transistor type. Here, however, in contrast to magnetic memories, the number of circuits and components increase only very moderately with capacity.

Thus far, the progress has consisted mostly of the necessary groundwork: theoretical understanding of all effects and mastery of the thin-film technology. Most samples were made on early experimental vacuum evaporating systems, but systems with a greater degree of control and automation were built in cooperation with SC&M. Also, a special "white room" was built (Fig. 10). The proper functioning of the continuous-sheet memory itself was proven in 10 x 10 and 4 x 4 arrays and signals with 30:1 discrimination were obtained. The cryotrons were tested. A plane with 128 x 128 = 16,384 bits and 508 cryotrons was also built (Fig. 11). The tester for this plane uses electronic circuitry built by EDP. It is hoped that the cycle time will be only one or two microseconds.

Content-Addressable and Read-Only Memories

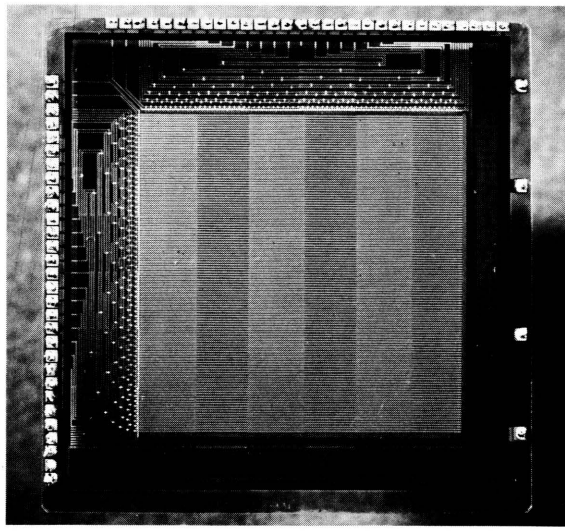
In addition to our efforts to increase the speed and the storage capacity of random access memories, we are also investigating memories with more generalized means of access. When a computer is put to the task of ordering, merging, sorting, and collating information, it does so today by serially examining every word stored in its memory. This is a slow process even with very fast memories. If it were possible to address the memory through the content of part of the information itself and retrieve the remaining part associated with it, these tasks and many others could be done much faster. Such memories are called *content addressable*, *associative*, or *search* memories. The essential of content addressability is to compare simultaneously a given word with all words of the memory and to detect perfect match wherever it occurs. The matching signal provides the location and identification of the associated information. This requires the mixing of logic with storage functions in the storing array itself. Because a content-addressable memory is likely to be the most useful when it has a large storage capacity, its realization presents technological problems akin to those of a large random-access memory, only somewhat more difficult. Therefore, we consider it to be a natural future development following the work toward large capacity *per se*. Nevertheless, we have some research in the field to establish the special logic of switching and methods for using such memories. For example, nondestructive read-out of the storing element is essential, as all bits of the memory must be sensed simultaneously and could not be all rewritten by any sensible amount of electronics. We have found ways to use arrays of transfluxors which are, of course, nondestructive read-out devices. Also, we have conceived very simple modifications of the continuous-sheet superconductive memory to obtain nondestructive and content-addressable memories.

To complete the account of our work on memories, mention must be made of fixed, or *read-only*, memories, of which the resistive matrix of the war years was a very early precursor. Today, we are considering two types: One consists of condensers arrayed in patterns and driven by tunnel diode circuitry. This can be considered as an integral part of the logic of a very fast computer. The other is an attempt to make highly modularized fixed memories which could be used in indexing for information retrieval and, in general, in dealing with tabular data.

Logic Switching Circuits

Increased speed in computers has yielded great returns throughout the short history of the art, as the effective computation speed went up by a factor much greater than the cost, and the computer could attack ever more ambitious tasks. The five-year Navy-sponsored project mentioned earlier has pushed speed almost to its goal of 1000 Mc, momentarily beyond economic profitability, but ultimately only a step toward the profitability of even higher speeds. These may be attained by optical techniques opened by such new developments as the electronically controlled gallium arsenide lasers, fiber optics, and new concepts for optical computers. We have started a modest exploration toward this far-reaching possibility. We are also studying limits to speed imposed by fundamental physical constraints through the analysis of the energy of

Fig. 11—Cryoelectric memory planes with 128 by 128 = 16,384 bits, and 508 cryotrons. Class Plate is 2" x 2".



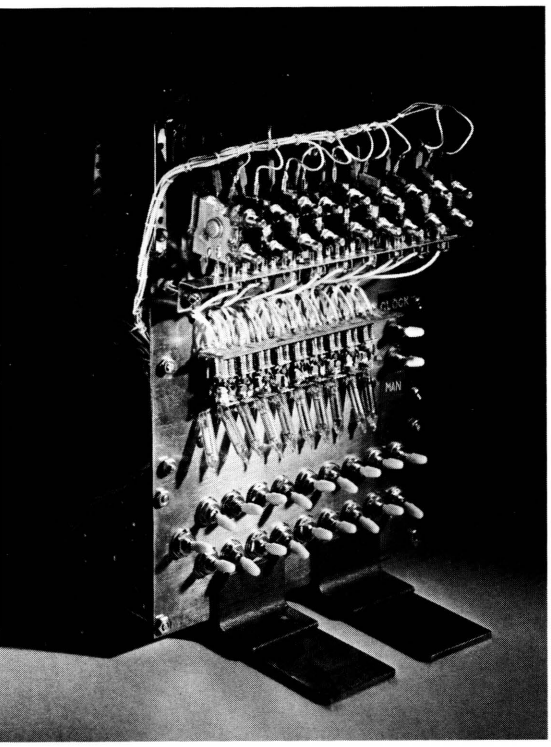


Fig. 12—Nine-stage adder carry circuits; tunnel diodes and transistors.

information-carrying energy in computer networks.

Of more immediate importance is our work with transistor and tunnel-diode circuits which, though somewhat slower, at about 200 Mc, have much wider margins of operation than the pure tunnel-diode circuits of the project just mentioned. Some of the principles of these circuits have been adopted for VANGUARD by the Advanced Development Laboratory of EDP and DEP Applied Research. Among them is a particularly simple and effective parallel adder, in which the carries are propagated from the least to the most significant position by a string of tunnel-diode gates. The propagation depends only on the properties of the tunnel diodes and is, therefore, very fast. Input and output to the tunnel-diode carry chain is by means of transistors which provide the necessary gain and whose somewhat slower response does not appreciably affect the speed of the adder, since it is not cumulative along the string. The nine-stage carry chain shown in Fig. 12 has operated with an average delay of only 0.3 nsec per stage. A 50-bit word length would entail a delay time of only 15 nsec.

The recent advances of integrated semiconductor technology may be a boon to still higher speeds. More likely, it may pave the way to the use of lavish numbers of somewhat slower components and avert the long trend toward the ever-more-efficient use of a few ultra-fast ones. The Computer Laboratory is considering some of the many problems posed by this intriguing possibility. It is not presently involved with the integration technology itself, for which it depends on the Electronic Research Laboratory and SC&M. Integrated technology is best at providing dense arrays of nearly identical elements such as thin-film or metallic-oxide unipolar transistors. To make specific logic networks out of such arrays, we are investigating the use of interspersed photoelements so that any desired microconnection pattern can be obtained optically. Also, we are considering means to tolerate defective elements in large arrays, a condition essential for obtaining reasonable production yields. This is achieved through the redundant use of several elements to do the switching possible with only one.

While we are deferring a substantial effort in superconductive logic elements in favor of large-capacity memories according to a research strategy mentioned before, we have a continuing fundamental effort relating to cryoelectric devices. In addition to computer logic circuits, we are considering the problem of amplifying very small signals. We are exploring various approaches to the problem including superconductive tunneling and magnetic control. The latter method has yielded very promising results and may provide low-noise amplification at microwave frequencies.

Input-Output Devices

The advent of the electronic computer was greatly facilitated by previous developments in typing, recording, and printing of digital data. This inheritance includes punched cards, punched tapes, the magnetic tape, electric typewriters, electric printers, films, and even electronic photography. All of these are electromechanical and originally did not attract as much research interest as the electronic computer itself, the *sine qua non* of the whole system. But today, the emphasis is changing *because the central computer is no longer the bottleneck*, and the input and output devices, which are the interface between man-and-machine, play a dominant role in the extent and convenience of usage.

At the RCA Laboratories, there are a number of projects in the area. In the Acoustic and Electromechanical Laboratory there is work with magnetic recording, high-speed printing, Electrofax, and the phonetic typewriter. In the Computer Research Laboratory there are three projects; large and fixed memories, character recognition, and displays.

The work in large and fixed memories described previously is cited here because these devices are intended to make unnecessary today's use of tapes for auxiliary memory. The tapes are excellent input-output devices and excellent records for dead storage. But their use as auxiliary memories, with fast stop-go mechanisms, is a costly artificiality and a source of much grief in computer maintenance.

The objective of our work in character recognition is to develop the logical principles of a reading machine which can tolerate changes in style — i.e. changes in the font (style or size of typography) and imperfections in printing. Such logic was developed by experimental simulating procedures on a general-purpose computer. Recognition is accomplished by detecting sequentially a set of geometric character features which tend to be independent of type style and printing imperfections. This logic was of help in the design of optical character-recognition machines being worked on by EDP. It is hoped that a relatively simple machine, consisting mostly of a fixed memory containing all the required recognition logic, can be built to recognize characters at rates of 100 characters per second or greater. The design of such a machine was simulated on a large general-purpose computer.

Electronic displays of numerals, letters, and graphs, is an important form of computer output. Furthermore, the problem of a flat display such as mural television is akin to that of a computer memory. There must be rapid access to many elements and each element must have storing properties. The difference is, of course, that the elements must emit or control light in the display but only provide an electric signal to a common circuit in the memory. Computer memory techniques were applied to this problem and published in 1957, as was mentioned, and resulted in a transfluxor controlled electroluminescent display. A similar application was started recently. It is based on a ferroelectric equivalent of the transfluxor, the *transcharger*, which may overcome the complexity in construction and excessive power requirements of the otherwise promising earlier approach. Some success has been obtained with various transchargers made of newly improved ferroelectric materials.

Computer Theory

The physical makeup of computers can still be improved enormously. The majority of our activities are aimed at order of magnitude increases in storage capacity and computer speed, drastic cost reductions, and better inputs and outputs. However, we realized several years ago that great strides are needed in design efficiency, organization and application of computers, and that in the not-too-distant future the really important

progress may be possible only in this field. We have started a theoretical group which has grown gradually and now has an important research program. The program includes studies in *switching theory*, in *system analysis*, and in *artificial intelligence*.

Switching theory is aimed at providing a systematic basis for efficient computer logic and circuit design. Two areas have been studied: threshold logic and reliability of networks. A theory was developed which deals with the threshold gate in much the same fashion that conventional switching theory deals with *and*, *or* or *nor* gates. General theorems were established and applied to specific problems. For example, taking the simplest significant three-input threshold gate (the majority gate), ways to realize all three-input switching functions have been systematized. This theory has also provided insight into a class of proposed adaptive machines based on methods of variable-parameter threshold devices. With respect to the reliability of switching networks, our approach has been the introduction of redundancy at the element level. By using three elements instead of one, and then triplets of triplets, etc., we showed that such a recursive network is in fact efficient in redundancy, although it requires many elements. It is the only way we know to build reliable equipment from really poor components. Other schemes of redundancy are being investigated.

In system analysis, we are concentrating on the problem of developing mathematical criteria for adaptive systems. In systems having a finite number of possible discrete states, the feedback arrangements customary in simple analog systems are not possible. The main problem consists of finding the decision criteria making it possible to jump from any given state of the system to another so that the system progresses toward some desired goal. The mathematical theory of Markov chains, which are ordered series of states each being derivable from its predecessor, is the basis of our study. A number of mathematical inventions have been made beyond the known classical theory, and several concepts have been developed which make it convenient to have an insight on this difficult problem.

Artificial intelligence (a connotation now gaining acceptance) describes a growing activity aimed at mechanizing certain intellectual processes commonly used by people in devising solutions to problems. Such processes involve heuristic methods where guesses, hunches, trial and error, etc. are used. Work in this area is expected to broaden the use of computers to problems for which either a systematic solution procedure does not exist, or if it does exist, is too cumbersome to be of any real value. We are studying the question of automatic problem-solving by attempting to develop procedures that construct simple proofs to theorems in the propositional calculus. This problem, with its simple known formalism, enabled us to work out a general framework for problem-solving procedures. We are also working on the possibility of automatically forming a theory on the basis of examples. Here, we are searching for the proper general program for a computer from which specific programs could be constructed so that an unknown problem would be solved on the basis of a number of given similar solved problems. In general, we found that the subjects of automatic theory formation and problem solving are closely related. Considerable progress in both was recently realized.

In computer theory, we are working in fields which may be considered to be at the opposite ends of a spectrum: on one hand, the theory of building the elementary parts of today's machines, and on the other the possibility of extensions to endeavors which require the conception of new computation schemes and also of new relevant mathematics. In the middle of this spectrum is the large field of the practitioner of computer design and utilization centered on the problems of machine organization and programing. Developments in machine organization and in programing languages and their processors are closely related. They are extremely important and constitute a large activity throughout the country. Machine designers and users that are close to the logic and economics of specific applications are best qualified to establish the requirements and to propose solutions in these areas. We have directed our research in theory of computers to general design principles and to new schemes for extended future applications, rather than the working out of specific machine or software designs. This is consistent with our work on the physical aspects of computers, where we tend to look at new principles rather than build improvements of well-known solutions. For these reasons, we have limited our efforts in machine architecture and software to cooperation with EDP. We plan to enlarge our activity in this field.

CONCLUSIONS

The traditional task of electronics is confined to the faithful transmission of information in space and time. The advent of computers changed this radically. The job is now to combine and cause the interaction of signals so as to create new signals which explicitly represent their useful content. Viewed in this sense, the art of electronics has acquired a new breadth and started on a path even more inspiring than that entered upon by its early pioneers.

Electronic computers have already made a tremendous impact wherever lengthy routine tasks were sufficiently systematized to be ready for rapid automation. These are the well known business applications to general accounting, to insurance, to transportation reservations, to payroll, to inventory control, etc. Of similar importance were, of course, scientific applications which gave birth to the art. Space travel is inconceivable without computers, and so are numerous military systems. The list of new fields could be continued to cover weather predictions, medical electronics, traffic control, management control, automation and many others. These applications are well known, and have already had a great impact on our lives.

Less known perhaps, but of even greater importance, is a revolution in our thinking. No longer is it necessary to confine ourselves to simple thought-patterns striving for economy of manipulative steps. We can now contemplate arbitrarily large numbers of manipulations and thereby handle usefully almost any problem we desire. There hardly seems any intellectual endeavor which is not being profoundly re-examined because of this possibility. Universities and even high schools are bringing computers into their regular curricula, and applied mathematics is permeating all science.

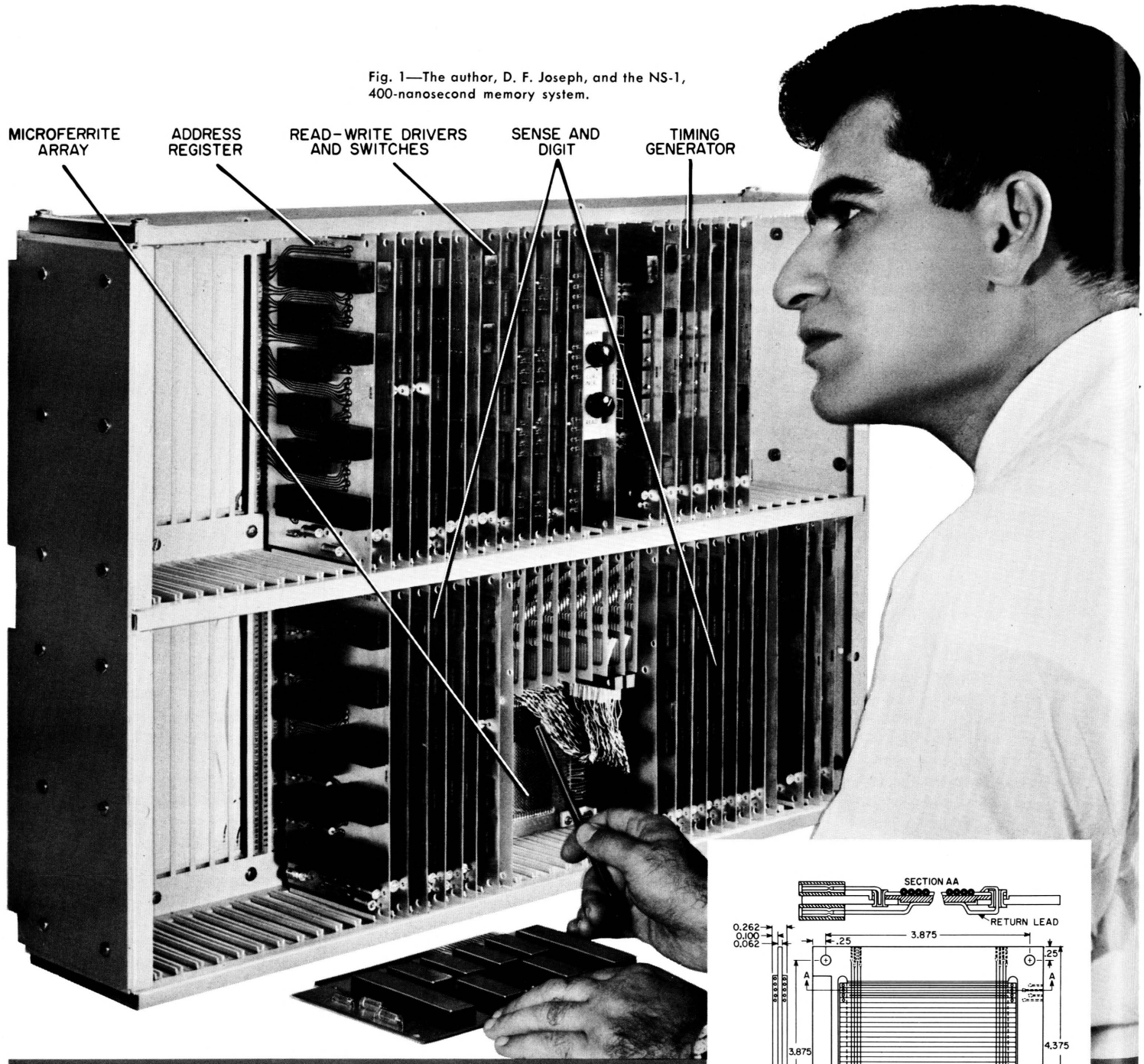
Speaking for myself, I have to admit, along with most early workers in the field, that our wildest dreams of the formative years did not envisage the great developments which have already occurred, and even less the promise of the foreseeable future.

The impact of computers can be expressed more pragmatically in terms of dollars. The electronic data processing industry in 1962 was a business of $2.8 billion, and it is believed that it will reach $3.5 billion in 1963. We believe that RCA must and will play a leading role in this business. The Computer Research Laboratory is striving to help RCA in this leadership.

BIBLIOGRAPHY

1. Dr. James Hillier, "The Engineer and the Corporation: RCA Labs"
2. Dr. Jan A. Rajchman. "The Transfluxor"
3. A. B. El-Karek, "An Electron Beam Machine," *RCA Review*, March 1963.
4. Dr. H. Lemaire, "New Techniques for Ferrite Nanosecond Memories"
5. Dr. Joseph, "A 400-Nanosecond Ferrite Memory"

Fig. 1—The author, D. F. Joseph, and the NS-1, 400-nanosecond memory system.



THE NS1—A NEW 400-NANOSECOND MICROFERRITE MEMORY SYSTEM

This sub-microsecond microferrite memory has a 400-nsec read-write time and is a random-access, word-organized system of 128 thirty-bit words. Since there are no programming restrictions on the memory, random access may be selected repetitively at the full clock rate of 2.5 Mc. Future basic memory module sizes of 500 to 1,000 sixty-bit words are feasible.

D. F. JOSEPH, Mgr.
Memory Systems Product Development
Memory Products Department
Semiconductor and Materials Division
Needham Heights, Mass.

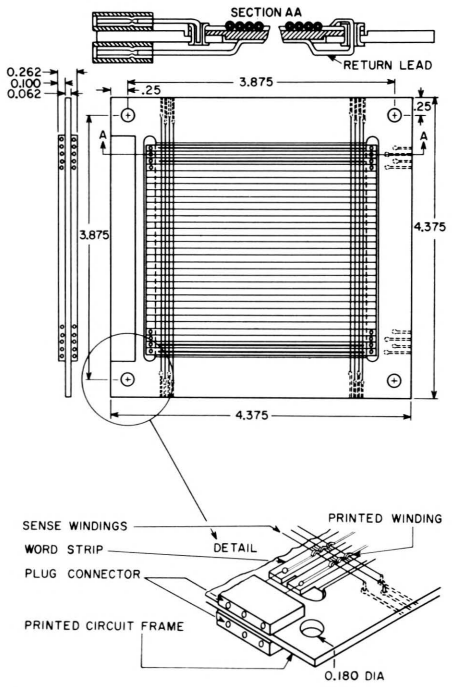


Fig. 2—N7100 microferrite array.

DAVID JOSEPH received his BSEE from the University of London in 1952 and his MSEE from the University of Pennsylvania in 1958. From 1952 to 1953 he was a Graduate Apprentice at the Edison Swan Electric Company in England, associated with the development of vacuum tubes. From 1953 to 1956 he was an engineer at the Bell Telephone Company of Canada, working on transmission problems, and from 1956 to 1958 he was an instructor at the University of Pennsylvania, working full time on a research project pertaining to high-frequency equivalent circuits and measurements of transistors. From 1958 to 1960 he was a Member of the Technical Staff at Bell Telephone Laboratories, working on the development of new magnetic devices and circuitry associated with memory systems. In January 1960, Mr. Joseph joined the RCA Memory Products Operation as Senior Member, Technical Staff in the Memory Systems Engineering Group. He has since been engaged in the design and development of memory systems as an engineer, then as group leader, and now as manager.

IN PRESENT-DAY advanced computer technology, the requirements of memory systems have been directed toward reading and writing in the sub-microsecond region. Of all the techniques proposed for sub-microsecond memories, he word-organized system using partially switched ferrite cores[1] has emerged as the most practical and successful. The ferrite device exists today as an economical production item[2], working models in the laboratory have been reported[3,4] and large sub-microsecond memories have been successfully built and tested.[5]

The Semiconductor and Materials Division's first sub-microsecond memory system, NS1, shown in Fig. 1, operates with a read-write cycle time of 400 nsec. It utilizes microferrite cores as memory elements and is constructed as a word-organized, two-core-per-bit memory system of 128 words of 30 bits each, addressable by random-access means. The memory system is completely solid-state. It uses RCA epitaxial mesa transistors in the logic and sensing circuits, and planar epitaxial transistors in the driving circuits. There are no programming restrictions imposed on the memory; hence, any random address may be selected repetitively at the full clock rate of 2.5 Mc.

Fig. 4—Core states, write *1*, bipolar sensing.

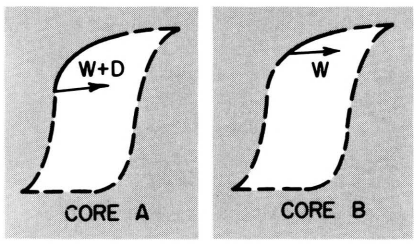
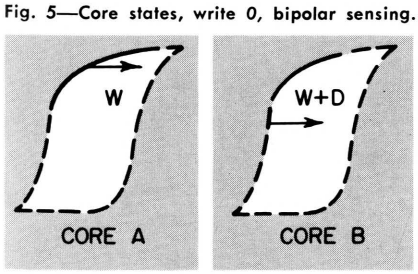


Fig. 5—Core states, write *0*, bipolar sensing.



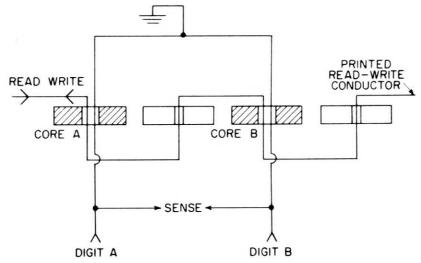


Fig. 3—Bit wiring, two cores per bit.

MICROFERRITE STACK DESIGN

The stack for the NS1 high-speed memory system is constructed of four RCA microferrite N7100 arrays modified for stacking. Each N7100 array consists of 32 word strips (Fig. 2); each word strip is 30 bits (60 cores) long. Microferrite cores have an outside diameter of 50 mils and an inside diameter of 10 mils. The cores are set end to end in each word strip and are connected by electroplated conductors. The word strips are set on 0.1-inch centers. The sense-digit line is strung through the cores in a conventional manner and is tied to terminals at the sides of the frame.

In this application, the printed conductor is utilized as the read-write drive line, and the sense-digit lines are interconnected in series through all four planes.

PRINCIPLES OF OPERATION

The microferrite memory uses two familiar principles of operation, partial switching and two-core-per-bit operation. Partial switching contributes greatly to the speed of operation; the two-core-per-bit arrangement simplifies the driving and sensing circuits. In two-core-per-bit operation, the drive-line load is constant, reversible flux voltages induced by fast-rising pulses are cancelled, and *0* and *1* output signals are relatively easy to discriminate because they are of opposite polarity. Bipolar sensing is used for further improvement of the signal-to-noise ratio.

The above statements are best illustrated by a description of single-bit operation. As shown in Fig. 3, the common read-write conductor is the printed winding along the word length, and the common sense-digit winding is a wire passing through the cores perpendicular to the word direction. For the writing of a *1*, a pulse of proper polarity and amplitude is applied to the read-write winding, while at the same time a pulse of proper amplitude and the same polarity is applied to digit winding *A* of Fig. 3. The resulting core states of the bit are shown in Fig. 4.

For reading, a pulse of proper amplitude but of polarity opposite to that of the write pulse is applied to the read-write winding. The voltages induced by cores *A* and *B* in the sense winding are of the same polarity, but the voltage from core *A* is larger than that from core *B*. The sense amplifier establishes the difference between the two voltages, and the net resulting voltage is read out as a *1*. For the writing of a *0*, the same pulses are applied as in the writing of a *1*, except that the digit pulse is applied to digit line *B* rather than to digit line *A*. The resulting core states of the bits are shown in Fig. 5. When the read pulse is applied to the read-write winding, the

Fig. 6—Digit, read, and write pulses. Horizontal sweep, 100 nsec/cm; vertical sensitivity, 200 ma/cm.

Pulse	Current (I), ma	Rise Time (T_r), nsec	Width (T) (between 50% points), nsec
Read	350	30	100
Write	220	24	48
Digit	70	20	80

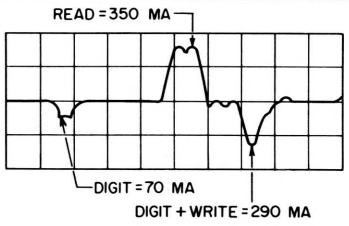


Fig. 7—Bit output from bipolar sensing. Horizontal sweep, 10 nsec/cm; vertical, 50 mv/cm; T_a, 70 nsec.

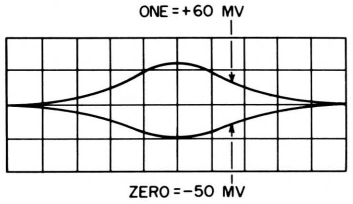
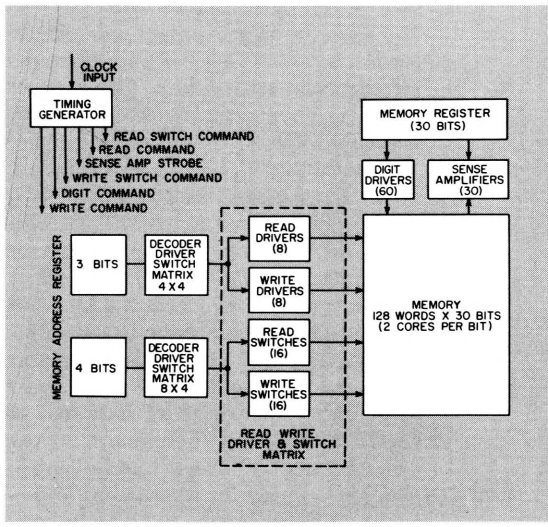


Fig. 8—NS-1 memory system.



voltage induced in the sense winding from core *B* is then larger than that from core *A*. The net resulting voltage in the sense amplifier is then similar to a *1*, but of opposite polarity, and is read out as a *0*.

The read-write and digit pulse characteristics are shown in Fig. 6. Fig. 7 shows typical *1* and *0* outputs of the bit for the pulse characteristics of Fig. 6.

MICROFERRITE ARRAY READ-WRITE DRIVE SYSTEM

Decoding from a seven-bit address register to the 128 word lines is accomplished in two levels by cascaded matrices. These matrices are called the *read-write driver* and *switch matrix*, and the *decoder driver* and *switch matrix*. Fig. 8 is a block diagram of the system.

The *read-write driver matrix* is an 8 by 16 bipolar matrix which drives the word lines through steering diodes. As shown in Fig. 9, there are 8 transformer-coupled read-write driver pairs and 16 transformer-coupled read-write switch pairs. Each word line has 4 matrix diodes located on both the driver and switch sides of the line. Driver current (read and write) is obtained from two current sources, as indicated in Fig. 9.

There are two decoder driver-switch matrices: one 4 by 4 matrix associated with the 8 read-write driver pairs, and one 8 by 4 matrix associated with the 16 read-write switch pairs. The decoder driver-switch is a unipolar-diode selection matrix which drives pulse-transformer primaries (of the read-write drivers and switches) as the matrix element. Fig. 10 shows a typical arrangement.

At the inputs of the decoder driver and switch, *and* gates are required to gate the appropriate read and write timing pulses together with the address levels. Pulse inputs are required only at the decoder driver inputs of the matrix. The decoder switch inputs use address levels to open or close the path to ground for the command pulses. The decoder driver is an emitter follower, and the decoder switch is a power gate.

The driving circuit operates as follows: The memory cycle is initiated by a clock pulse which is fed into a timing generator. (The operation of the timing generator is not discussed in this paper.) Simultaneously, the correct address is selected for the word to be interrogated, and the stabilized address levels appear at the *and*-gate inputs of the two decoder driver-switch matrices, as indicated in Fig. 10. In each of these matrices, one decoder switch is selected and a path is closed to ground for the current to flow in the primaries of the pulse transformers. The read-switch command is then applied, and the correctly addressed decoder driver in the 8 by 4 matrix is selected. A current pulse flows through the primary winding of the selected pulse transformer. The transformer functions as a level shifter to drive the read switch, and sufficient current is produced in the secondary winding to saturate the read-switch transistor.

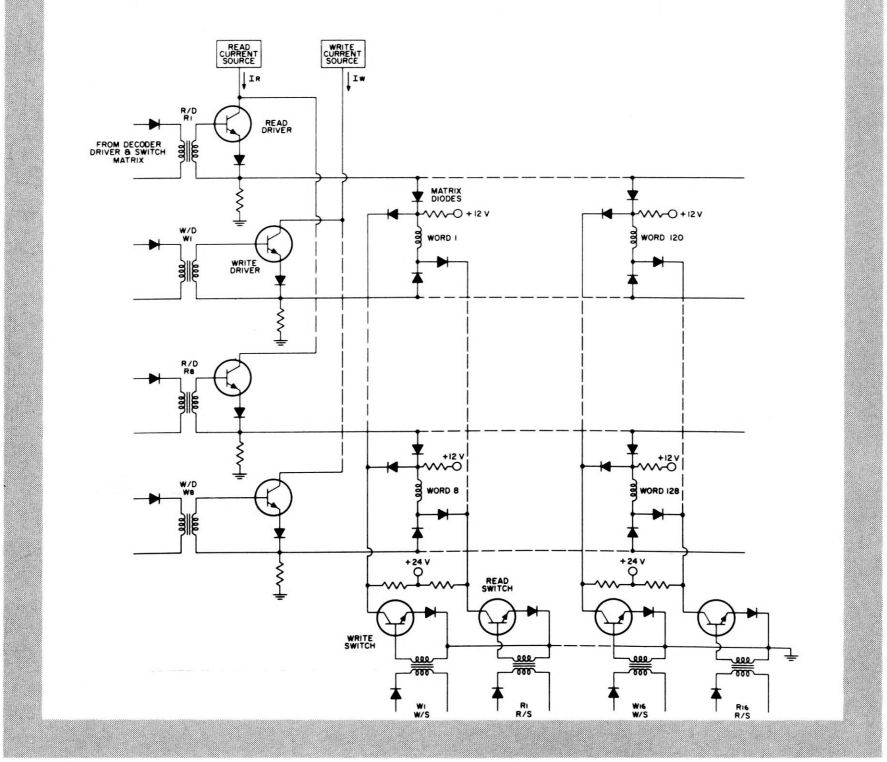
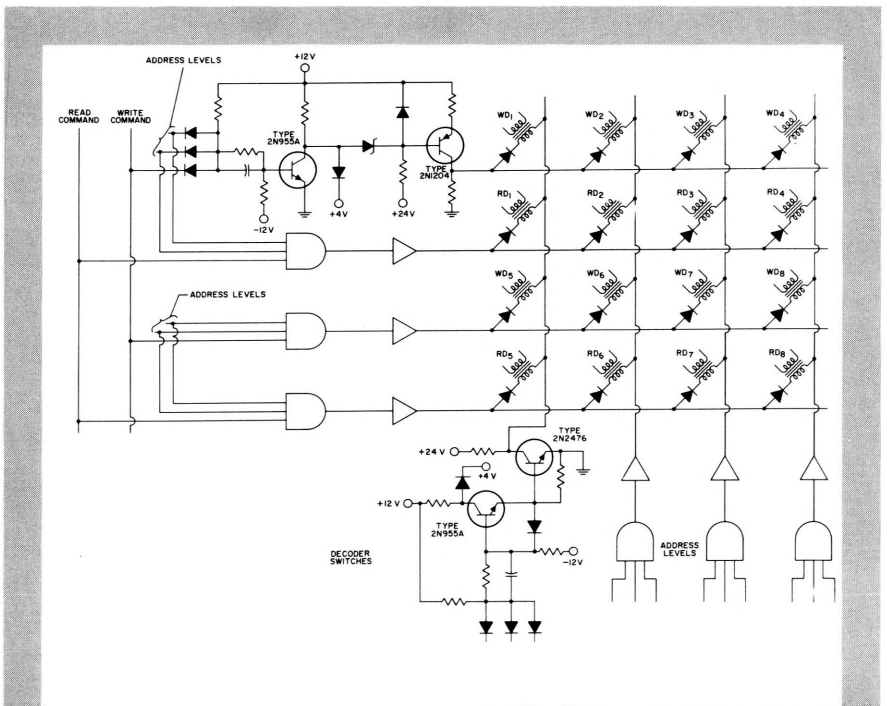


Fig. 9—Read-write driver and switch matrix.

Normally, the word lines are biased to a potential of 12 volts when no drivers or switches are selected, the outputs of the read drivers are at ground, and the outputs of the switches are at a potential of 24 volts, as shown in Fig. 9. When the appropriate read switch is selected, as discussed above, a ground path is established for drive current to flow. The read-drive command is then applied to the 4-by-4-decoder drive-switch matrix, and a read driver is selected by a similar process. The read-driver transistor is driven into saturation, and the current source is gated through. This current tends to flow to all word lines associated with the read driver; however, the full current flows only through the selected word line and through the selected switch to ground. The matrix diodes associated with the word lines block the flow of current into all unselected word lines. Read current flows for a specified period and is terminated when the read command is disabled and the read-switch command is disabled.

Write current, through the selected word line, is generated in a similar man-

Fig. 10—Decoder driver and switch matrix.



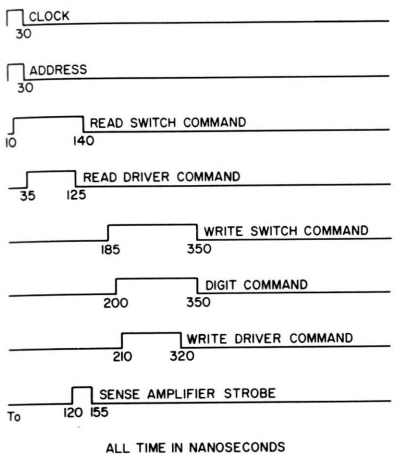


Fig. 11—Timing-generator command pulses.

ner by application of the write-switch command and write command to the decoder driver matrices. Fig. 11 shown the pulse widths and time sequences of these various commands.

SENSE-AMPLIFIER AND DIGIT-DRIVER OPERATION

The sense-amplifier and digit-driver interconnections to a sense-digit line are shown in Fig. 12. (Because the digit driver is only a voltage switch, its operation is simple and is not discussed here.) The sense amplifier is designed to detect usable signals of 30 millivolts and to recover from digit transients (which occur during the turn-on and turn-off of the digit driver) of several volts in about 30 nanoseconds. (This sense amplifier was developed by EDP Advanced Development, Pennsauken, N.J.)

The sense amplifier shown in Fig. 12 is of a differential type for common-mode rejection of noise in a two-core-per-bit sense output. The sense-winding inputs are connected so that when a *1* is read out of the memory, terminal 2 becomes more positive than terminal 1. Transistors Q_1 and Q_2 are used in the differential preamplifier circuit, Q_4 in a strobe gate, and Q_5 and D_1 (a tunnel diode) is the output "comparator" or pulse stretching circuit.

When a *1* is read out, Q_1 tends to conduct more and Q_2 to conduct less, (Q_1 and Q_2 are normally not saturated). More base drive is then supplied to Q_3. The collector current of Q_3 (also not saturated) flows through Q_4. Because diode D_1 is back-biased, there is no output. When the strobe command is applied, however, Q_4 is turned off and the current from Q_3 flows into the low impedance of forward-biased D_1 and switches it. The output transistor is thus driven to saturation for the duration of the strobe command.

When a *0* is read out, Q_2 tends to conduct more and Q_1 conducts less and the base drive of Q_3 is reduced. Because the collector current of Q_3 remains below the peak current of tunnel diode D_1, no output is possible.

The resistor-capacitor-diode network in the emitter circuit of Q_3 is designed to prevent false triggering of the output as a result of the reception of large digit noise.

SYSTEM TEST WAVEFORMS

The waveforms shown in Fig. 13 to 15 were obtained with a read-write cycle time of 400 nsec. The raw sense output of Fig. 13 shows a composite of *1*'s and *0*'s on a typical sense line; the digit and write transient recovery is accomplished within 350 nsec. Fig. 14 shows the strobed output of the sense amplifier for a pattern of all *1*'s and *0*'s. The read-write and digit currents are shown in Fig. 15.

CONCLUSIONS

The fabrication and operation of this ultra-high-speed memory system have been extremely successful. The system has been operated without errors at a cycle time of 375 nsec with read-drive currents of approximately 300 ma.

Investigations have shown that future basic memory module sizes of 500 to 1,000 words with 60 bits per word are feasible. The higher ratio of the sense-digit line delay to the cycle time for these larger modules will require a more complex timing program, and the sense-digit line terminations will require careful consideration.

As mentioned previously, the ferrite devices offer several advantages over other competing materials and devices. An important consideration in this program has been the core and bit uniformity. The ferrite cores that have been developed have shown excellent uniformity characteristics on two-core-per-bit operation. The high outputs (50 mv) and low current levels required for systems operation are another important factor.

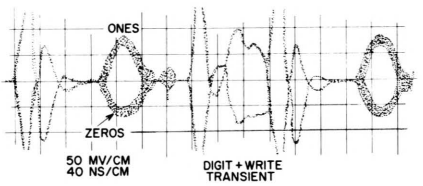


Fig. 13—Differential raw sense output of a bit. Composite 1 and 0 of 128 words are shown.

ACKNOWLEDGMENT

The author acknowledges the assistance in this project of J. Fung, R. Hogan, B. LeBlanc, J. Rodriguez, and U. Strasilla.

The helpful suggestions of Dr. H. P. Lemaire, P. D. Lawrence, and B. Frackiewicz in the applications of the microferrite device; and the help of E. Small for the mechanical packaging.

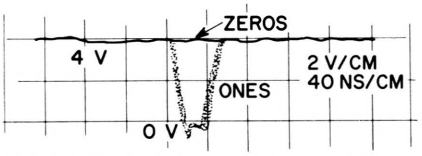


Fig. 14—Strobed output of sense amplifier.

BIBLIOGRAPHY

1. J. A. Rajchman, "Computer Memories: A Survey of the State of the Art," *Proc. IRE*, Vol. 49, No. 1, January 1961, pp 104-127.
2. H. P. Lemaire, "New Techniques for Ferrite Nanosecond Memories"
3. H. Amemiya, H. P. Lemaire, R. L. Pryor, T. R. Mayhew, "High-Speed Ferrite Memories," *Proceedings—F.J.C.C. 1962*, pp 184-195.
4. R. Shabender, T. Nelson, R. Lochinger and J. Walentine, "Micro-aperture High-Speed Ferrite Memory," *Proceedings—F.J.C.C. 1962*, pp 197-212.
5. W. H. Rhodes, L. A. Russell, F. E. Sakaly, R. M. Whalen, "A 0.7—Microsecond Ferrite Core Memory," *IBM Journal of Research and Development*, Vol. 5, No. 3, July 1961.

Fig. 12—Sense-amplifier and digit-driver layout.

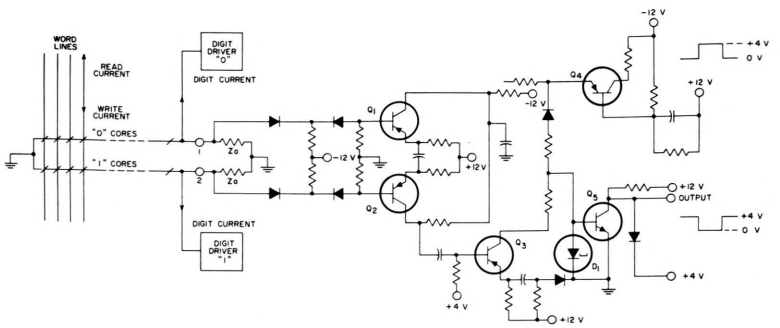
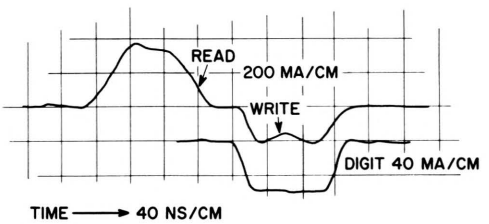


Fig. 15—Read, write, and digit currents.



300-Mc TUNNEL-DIODE LOGIC CIRCUITS

A complete set of tunnel-diode logic circuits has been developed. Average delay per logic level is 0.5 nsec. Average DC power dissipation per gate is 100 mw. To achieve this performance, new techniques were employed: 1) nonlinear biasing using the tunnel resistor, a new tunneling device; 2) trimming, a technique of current bias adjustment to offset initial tolerances; 3) transmission-line terminating without sacrificing signal amplitude. Using these circuits, a 40-gate model was constructed that can shift and count at 300 Mc, and that has operated for 200 hours with very good reliability. The short delay per logic level, high repetition rate, and low average power dissipation make these logic gates a powerful set of building blocks for high-speed digital computers. (The 40-gate model is shown on the front cover.)

M. COOPERMAN
Advanced Development Engineering
Electronic Data Processing
Camden, N. J.

MICHAEL COOPERMAN received the BEE degree from the City College of New York in June 1955. From 1955 to January 1961, he was with the Advanced Development Section of the RCA Home Instruments Division, Cherry Hill, N. J. His work has centered around the application of ferromagnetic and ferroelectric devices to television circuits and, more recently, the development of tunnel-diode circuitry and parametric amplifiers for operation at gigacycle frequencies. He is presently with EDP, Pennsauken, N. J., where he is engaged in development of tunnel-diode logic circuits for digital computers. Mr. Cooperman is a member of the IEEE, Tau Beta Pi, and Eta Kappa Nu.

FOR SEVERAL YEARS, RCA has been developing tunnel-diode logic circuits for high-speed digital computers. A set of circuits was developed by the middle of 1961 which was then used to build a feasibility model of approximately 300 gates. The reliable operation of that model proved the feasibility and practicality of using tunnel-diode logic circuits in high-speed digital computers.[1,2]

The experience obtained with that model suggested numerous ways in which those logic circuits could be improved to provide better performance and reliability. A program was undertaken which resulted in the new and improved set of logic circuits described herein. These new circuits, as compared to those in the 300-gate model, have reduced delays, reduced power dissipation, and increased reliability, repetition rates, fan-in, and fan-out. To demonstrate the improved circuits, a 40-gate model was constructed that can shift and count at 300 Mc.

This paper describes the operation and performance of these circuits, intended for the reader interested in a general understanding of their operation and capabilities. (Additional design considerations, circuit diagrams, layouts, and waveforms, are available in the literature.[5])

Tunnel-diode logic circuits can be designed to operate in either the monostable or bistable mode.[3] Stages operating monostably are used in all of the three gates described in this paper. These are the *or* gate, *and* gate, and bistable circuits. The bistable circuit is the only one using a stage operating in the bistable mode.

BASIC MONOSTABLE STAGE

The basic monostable stage, a tunnel diode in series with an inductance and a tunnel resistor, is shown in Fig. 1. The tunnel resistor is a new device formed by plating a resistive path across a tunnel-diode junction. The two are then mounted in one glass package to form one device. Plating and mounting in one package is used to keep the stray inductance between the two elements to a minimum. Otherwise, the tunnel diode may oscillate, thus preventing the two elements from acting as one device. The characteristics of a tunnel resistor and the elements from which it is formed are shown in Fig. 2. The characteristic of a tunnel rectifier is shown in Fig. 3.

To facilitate describing the operation of this circuit, the tunnel-resistor biasing characteristic is superimposed on the characteristic of the tunnel diode in Fig. 4. The more popular term for *biasing characteristic* is *load line;* however, since this circuit will have other loads, *biasing characteristic* is preferred.

When an input is applied to the stage of Fig. 1, the biasing characteristic shifts to a position indicated by curve b in Fig. 4. This causes switching along the trajectory indicated by the dotted lines. The switching cycle is divided into a number of regions as indicated in Fig. 4. The tunnel resistor provides a desirable biasing characteristic, as it is relatively flat when the tunnel diode switches over the peak; it then drops sharply to permit monostable operation. Since the voltage required to obtain this biasing characteristic is only 250 mv, the power dissipation is relatively low (Table I).

TUNNEL-DIODE OR GATE

To perform a logical *or* function, a circuit must produce an output when any one of its inputs is activated. This is easily accomplished by providing more than one input to the basic monostable stage of Fig. 1. A current into any one of the inputs causes the tunnel diode to fire and thus accomplish an *or*.

A complete circuit diagram of a tunnel-diode *or* gate is shown in Fig. 5. It consists of two monostable stages cascaded to provide the required current amplification. (TD_1, TDR_1, R_1, and L_1 are elements of the first stage, while TD_2, TDR_2, R_2, and L_2 are elements of the second stage.) Because of the requirements of increased fan-in, fan-out and speed, new techniques had to be employed in this circuit. Two of the most important circuit techniques now utilized in the *or* gate and the other gates are *trimming* and *transmission line terminating*. Trimming is done with R_1 and R_2; transmission line termination is accomplished by a network consisting of TR_1 and TR_2.

Trimming

To increase both fan-out and operation speed, TD_1 and TD_2 (Fig. 5) required being biased closer to their peaks; however, worst-case tolerance conditions do not permit this. To cancel some of these tolerance variations, a current source of several milliamperes is added to each stage. In Fig. 5, the current sources for the first and second stages are provided by R_1 and R_2 connected to +3 volts. The resistance of R_1 and R_2 can be increased by trimming (mechanically removing some of the conductive material). After the gate is assembled, R_1 and R_2 are trimmed in accordance with a specified procedure to bias each stage to the desired amount below its tunnel-diode peak. In addition to its essential functions, trimming also provides the desirable characteristic of making the electrical performance more uniform. This results in an additional increase of speed in the entire system.

Terminating Network

The purpose of the terminating network is to eliminate reflections between gates. The operation of this network may be understood by referring to Fig. 6, which shows an output stage of an *or* gate driving the input stage of an *or* gate via a transmission line. When TD_2 fires, it supplies a current to TD_1 via TR_o, TR_1, and TR_2. Because TR_1 is biased, during this time it absorbs no current. When TD_1 fires, V_6 increases to about 500 mv. Assuming temporarily that TR_1 is not in the circuit, the increase in V_6 *prevents* the flow of I_{in}, which is reflected back to TR_o. Depending on the length of the transmission line and the pulse width produced by TD_2, this current may once again be reflected. By this time, the current pulse is traveling towards the input stage and if that stage had recovered from its firing cycle, this reflection could cause undesired firing of TD_1. With TR_1 in the circuit, current I_{in} is absorbed by TR_1 when TD_1 fires. Thus the flow of I_{in} is practically not interrupted, and consequently no significant reflection is generated. The characteristics of TR_1, TR_1, and TR_2 are chosen such that the transmission line will be approximately terminated at all times.

TABLE 1—Gate Properties and Performance

OR GATE:	
Fan-in	5
Fan-out	6
Delay (Total delay of the gate from input to output, measured at 300-mv points.)	0.27 nsec min; 0.6 nsec max
Repetition rate (Maximum repetition rate at which pulses may be applied.)	300 Mc
DC Power Dissipation	57 mw
AND GATE:	
Fan-in	6 (5 pulse and 1 level*)
Fan-out	3
Delay (Total delay of the gate from input to output, measured at 300-mv points.)	0.65 nsec; 1.00 nsec max
Repetition rate (Maximum repetition rate at which pulses may be applied.)	300 Mc
T_1: (Waiting time between application of a pulse and application of a level to an *and*-gate when switching is desired.)	0.13 nsec
T_2: (Same as T_1 except switching not desired.)	1.00 nsec
T_3: (Waiting time between application of a pulse and removal of a level from an *and*-gate when switching is desired.)	0.62 nsec
T_4: (Same as T_3, except switching not desired.)	0.13 nsec
DC Power Dissipation	137 mw
BISTABLE CIRCUIT:	
Fan-in set	4
Fan-in reset	4
Fan-out	3 + Console
Set Delay	0.13 nsec min; 0.65 nsec max
Reset Delay	0.23 nsec min; 1.1 nsec max
Set-Reset Wait (Waiting time between application of a reset pulse to a bistable gate.)	1.6 nsec
Reset-Set Wait (Waiting time between application of a reset pulse and the application of a set pulse to a bistable gate.)	2.0 nsec
Repetition Rate (Maximum repetition rate at which pulses may be applied.)	150 Mc
DC Power Dissipation	74 mw

*Performs an *and* operation between the level and any one or more of the five pulse inputs.

TUNNEL-DIODE *AND* GATE

A schematic diagram of the *and* gate is shown in Fig. 7. To understand the operation of this gate, it is divided into three stages: pulse buffer stage, *and* stage, and output stage. All of these operate in a monostable mode and are biased with tunnel resistors, as in the *or* gate.

A pulse input, after being reshaped by the pulse buffer stage, is applied to TD_2 of the *and* stage. Because TD_2 is returned to a positive potential, under quiescent conditions TR_4 and TR_5 are conducting currents the amplitudes of which are determined by the magnitude of R_3 and R_4, respectively. If either the pulse or the level is low, the *and* stage is inhibited from firing because of the conduction of either TR_4 or TR_5. The presence of a level cuts off TR_5, while an output from the pulse buffer stage cuts off TR_4. This causes the entire current from TDR_2 to flow into and fire TD_2.

This type of *and* gate has the advantage of not requiring tight control between the amplitude of its pulse and level inputs. The only requirement is that the inputs (V_6 and V_L) be less than 100 mv when they are low and greater than 470 mv when they are high.

Output Stage

The output stage is coupled to the *and* stage via the reverse direction of TR_6, the characteristic of which is shown in Fig. 3. Coupling in this manner is necessary in order that the quiescent voltage difference (approximately 250 mv) between the *and* stage and output stage be absorbed with negligible DC leakage current between the two stages.

The purpose of R_5 is to provide the proper biasing characteristic for the *and* stage and efficient coupling between the *and* and output stages.

Pulse Buffer Stage

This stage serves several important functions: 1) It converts the pulse input impedance of the *and* gate to that of the other gates. This prevents the *and* gate from imposing severe requirements upon the preceding gate, alleviating a reduction in fan-out of all gates. 2) It makes the pulse input impedance of the *and* gate compatible with the termination network used in the *or* gate. 3) The output of the buffer stage is relatively

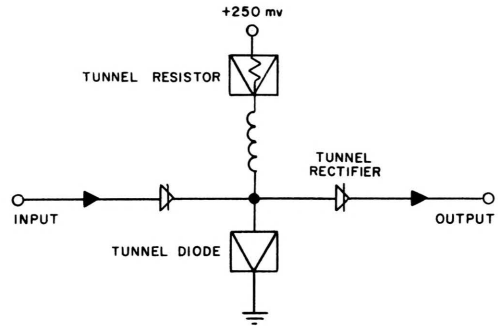


Fig. 1—Basic monostable stage.

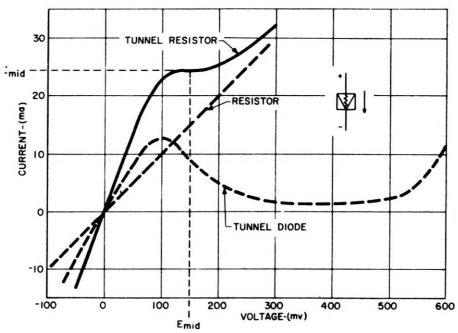


Fig. 2—Tunnel-resistor characteristic.

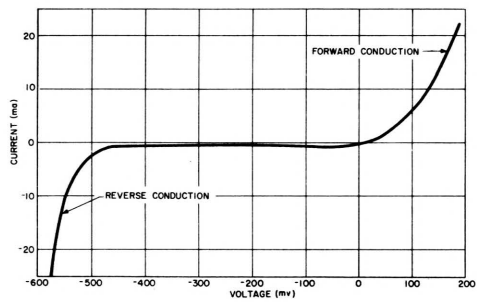


Fig. 3—Tunnel-rectifier characteristic.

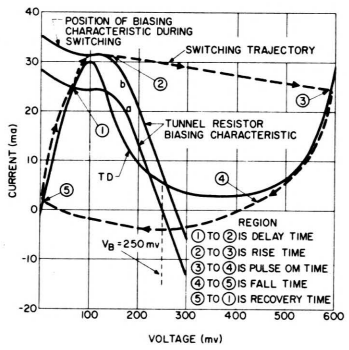


Fig. 4—Monostable stage, tunnel-resistor biasing.

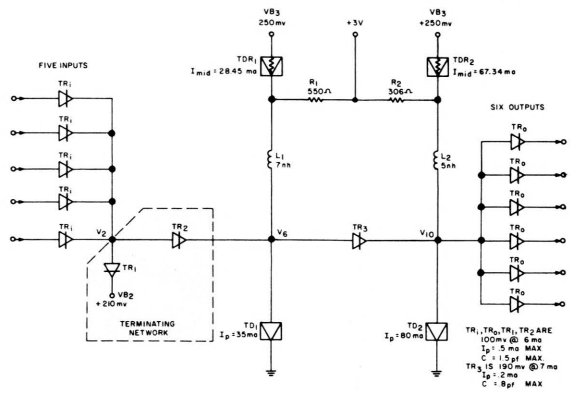


Fig. 5—Tunnel diode or gate.

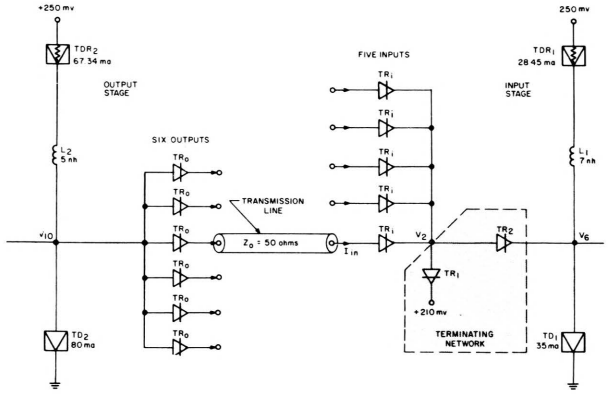


Fig. 6—Interconnection of gates.

insensitive to input pulse width variations. Thus, under all conditions of input, the buffer stage generates a narrow pulse of constant width. This reduces the waiting period for applying a level after the application of a pulse.

In this gate, trimming is employed whenever possible; R_1, R_2, and R_6 are the trimming resistors for the buffer, *and*, and output stages, respectively.

TUNNEL-DIODE BISTABLE CIRCUIT

The bistable circuit consists of a set amplifier, inverter driver, inverter, and bistable unit (Fig. 8).

The bistable unit stores a *0* when it is in the low state and a *1* when it is in the high state. The other units are monostable, and when activated, their function is to set or reset the bistable unit.

The bistable unit (Fig. 9) consists of TD_4, V_{B4}, and V_{B6} and R_4. Bistable action is obtained in conjunction with the *and*-gate loading (Fig. 10). After the curve of TD_4 is inverted and returned to 550 mv, its characteristic shifts from the first into the fourth quadrant. The current biasing characteristic of R_4, in conjunction with that of the *and*-gate input characteristic, results in a bistable load line, as indicated.

Setting

The set amplifier is a monostable stage identical to the first stage of an *or* gate. When activated by a set pulse, the set amplifier supplies an amplified current pulse to the bistable unit which is switched to the high state along with the indicated trajectory of Fig. 10.

Resetting

The inverter driver (TD_2 and TDR_2) operates like a monostable stage, while the inverter (TD_3 and TDR_3), is an inverted monostable stage.

When a reset input is applied, TD_2 switches to the high state. This causes the current I_2 in TD_2 and TDR_2 to decrease. A decrease in I_2 acts as a negative current input to TD_3, triggering the inverter to produce a negative pulse. The negative pulse causes reverse conduction in TR_6 which diverts some of the current from TD_4. (An approximate characteristic for TR_6 is shown in Fig. 3.) A current flow out of TD_4 causes the load line of Fig. 10 to move down such that the intersection at point B disappears. This causes the bistable unit to switch to the low state along the indicated trajectory.

In the bistable circuit, R_1 and R_2 trim the set amplifier and inverter driver, respectively; R_3 boosts V_2 and V_6 to values required for proper operation.

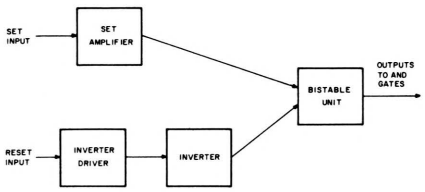


Fig. 8—Bistable circuit.

DESIGN CONSIDERATIONS

The three circuits were designed so that after having been trimmed to overcome some of the initial tolerance variation, they could operate under worst-case conditions of component and power-supply variations. This design resulted in certain rules to be adhered to by the logic designer using these circuits. Using these circuits and restrictions, a number of trial logic designs were made for general and special purpose computers. These designs indicated that the rules could be adhered to with very little or no sacrifice in system performance.

The choice of circuit parameters was based on DC worst-case analysis, and dynamic simulation on the RCA 301.

CIRCUIT INTERCONNECTIONS

All interconnections are made with 50-ohm miniature coaxial transmission lines for monostable circuits and 31.5-ohm lines for bistable to *and*-gate connections. The choice of 50-ohm lines for monostable circuits was based on obtaining optimum coupling efficiency between gates. The choice of 31.5-ohm lines for bistable circuit coupling was based on a compromise between speed of level build up and reliability.

PERFORMANCE

Table I summarizes circuit properties and performance.

Fig. 11 shows a disassembled wafer and a typical assembled wafer. The

Fig. 7—Tunnel diode *and* gate.

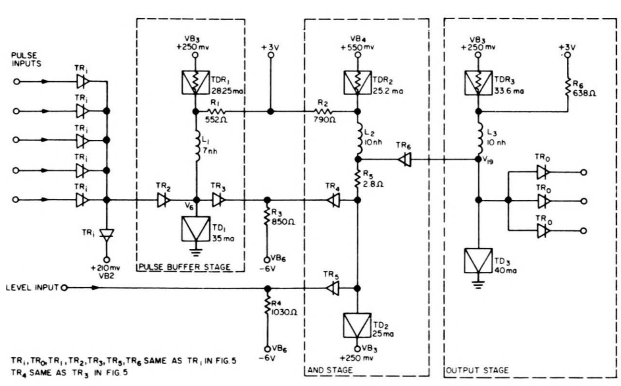
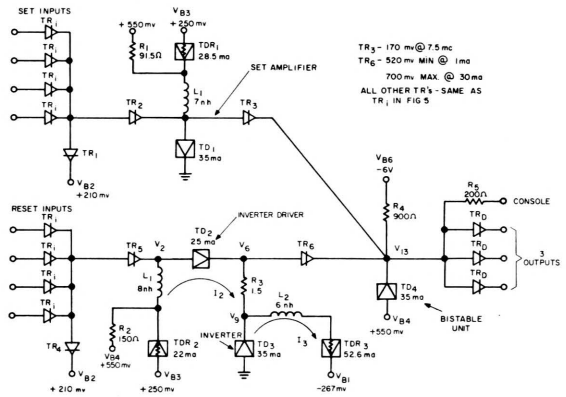


Fig. 9—Tunnel diode bistable circuit.



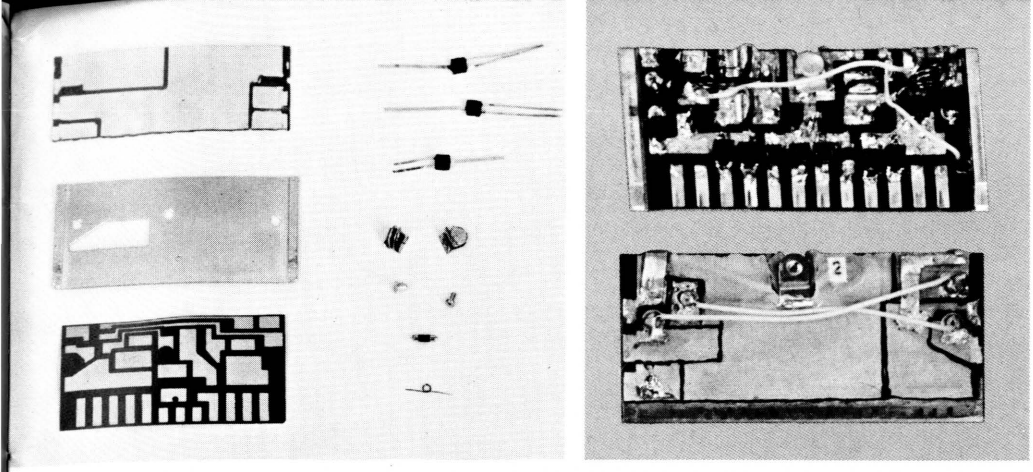


Fig. 11—Disassembled wafer, and an assembled bistable wafer (front and rear). Wafers are about 1½" long.

wafer has 12 signal connections and 6 power-supply connections. The signal connections are on top of the wafer; power is supplied to the back of the wafer. Each power supply connection is shunted by a high-dielectric pad which serves as a capacitance to absorb transients due to circuit switching. In the circuit layout, care was taken whenever possible to keep stray inductances and capacitances to a minimum. In several cases, the tunnel diode had to be mounted in a flat position in order to reduce the stray inductance to a tolerable value. This was especially true with the output of the *or* gate.

Circuits of this kind were constructed and trimmed in accordance with a trimming procedure established for each circuit. The circuits were then tested for electrical performance, which indicated reliable operation and good agreement with theoretical and computed results.

Typical output waveforms from these circuits are shown in Figs. 12, 13, and 14.

Using 40 of these gates, a 5-bit shift-register–counter was constructed and has been operated reliably at 300 Mc. The *shift mode* utilizes two bistable circuits per bit of storage and two *and* gates to transfer information from one stage to the next. In the *count mode*, the same two bistable circuits and two *and* gates are used per bit, resulting in five triggerable flip-flops, which are connected serially to function as a counter.

Fig. 10—Bistable switching.

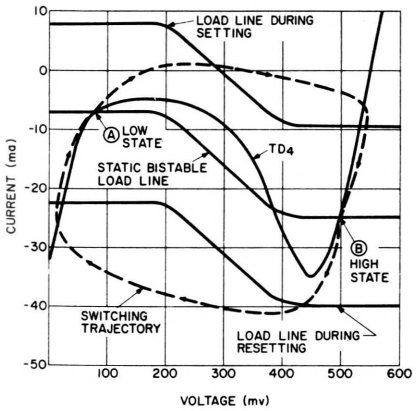


Fig. 15a shows the output of one of the bistables in ring shifting for various bit patterns. Fig. 15b shows the states of the first through the fifth bits when counting is performed.

CONCLUSION

Theoretical analysis and experiments have shown that the tunnel-diode logic circuits presented here are logically complete and suitable for the construction of large-scale, high-speed digital computers. The 40-gate model constructed with these circuits demonstrated the most commonly performed logic operations in digital computers. Among these were shifting and counting at a 300-Mc rate with an average delay per logic level of 0.5 nsec. Logic designers have made trial designs for general-purpose and special-purpose computers using these circuits, which were found to be effective and versatile logic building blocks.

ACKNOWLEDGEMENTS

The work described here is the product of a joint effort by the Logic Circuit Group of the EDP Advanced Development Section. Credit is due the following individuals for their part in this work: R. H. Bergman, for coordinating this project and for his ideas which lead to this circuit approach; E. C. Cornish, for circuit analysis and laboratory development of the individual circuits and the 40-gate subsystem; C. R. Pendred and D. Durr for dynamic simulation of the circuits on the RCA 301 Computer; I. Abeyta and H. Ur for laboratory circuit development; and W. J. Lipinski for DC tolerance analysis.

Some of the basic ideas which lead to circuit trimming were contributed by S. T. Jolly. The mechanical engineering of the wafers and assembly was done by M. E. Ecker. The power distribution system was designed by H. V. Rangachar.

The author also wishes to express his gratitude to the colleagues at Pennsauken, Somerville, and Princeton for their support and cooperation. Particular thanks are due to R. K. Lockhart and J. N. Marshall for their support.

Fig. 12—*Or* gate output voltage waveform: vert., about 300 mv above baset of waveform; horiz., about 3 nsec peak to peak. (Reference grid lost during photography.)

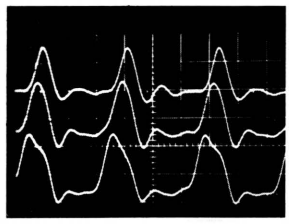


Fig. 13—*And* gate voltage waveforms. Top to bottom: output stage, *and* stage, buffer stage. Vert.: 200 mv/div.; horiz., 1 nsec/div.

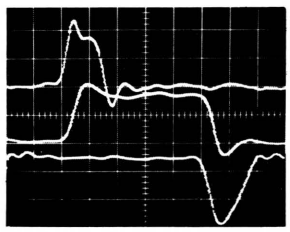


Fig. 14—Bistable circuit voltage waveforms: top to bottom, set amplifier, bistable output and inverter. Vert., 200 ma/div.; horiz., 1 nsec/div.

BIBLIOGRAPHY

1. R. H. Bergman, M. Cooperman, H. Ur, "High-Speed Logic Circuits Using Tunnel Diodes," *RCA Review*, June 1962, Vol. XXIII, No. 2. *Also see* their paper "High-Speed Tunnel-Diode Logic Circuits"
2. R. H. Bergman, "Tunnel Diode Logic Circuits," *Trans. IRE-PGEC*, Vol. EC-9, p. 430, December 1960.
3. H. S. Sommers, "Tunnel Diodes as High Frequency Devices," *Proc. IRE*, Vol. 47, p. 1201, July 1959.
4. M. H. Lewin, "Negative-Resistance Elements as Digital Computer Components," *Proc. Eastern Joint Computer Conf.*, No. 16, p. 15, December 1959.
5. M. Cooperman, "300-Mc Tunnel Diode Logic Circuits," *Proc. Pacific Computer Conference*, March 14-15, 1963.

Fig. 15a—Ring shifting at 300 Mc: vert., 300 mv/div.; horiz., 10 nsec/div.

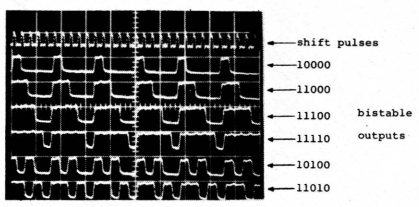
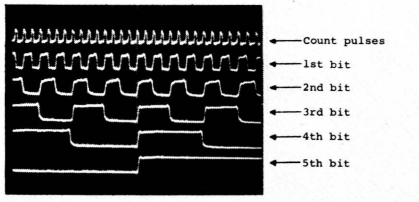


Fig. 15b—Counting at 300 Mc: vert., 300 mv/div.; horiz., 10 nsec/div.



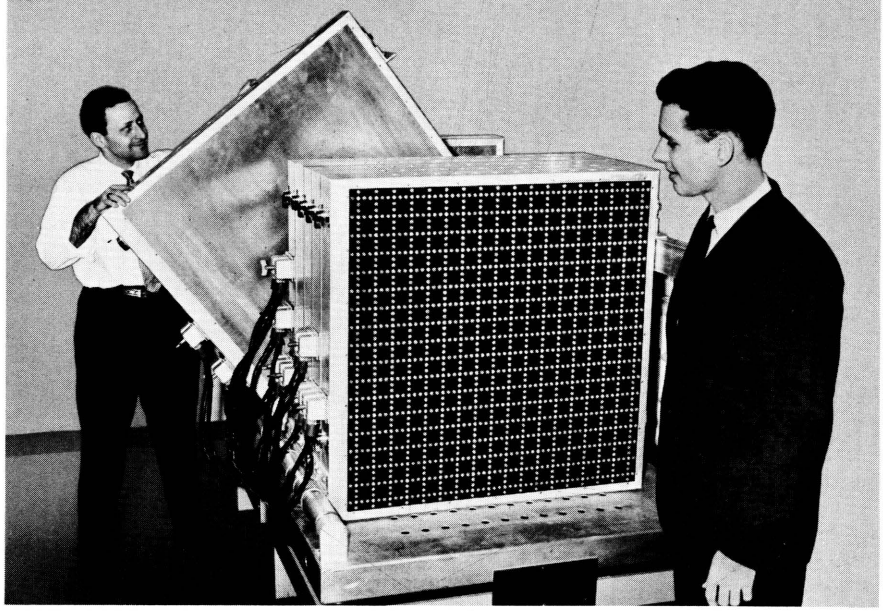


Fig. 2a — Retina model (view from direction A of Fig. 2b). M. Herscher. co-author, is pulling out one of the logic panels, while co-author T. P. Kelley observes.

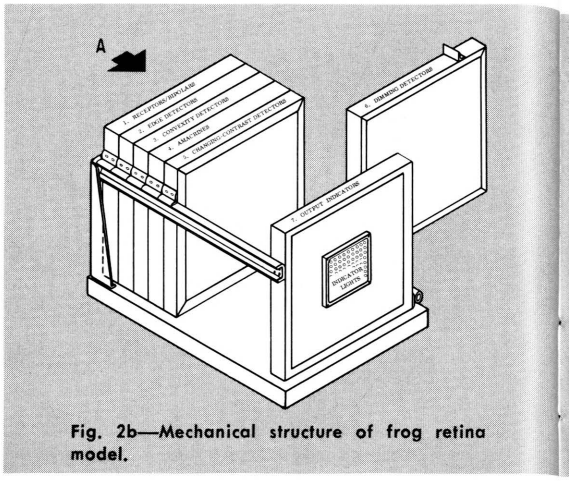


Fig. 2b—Mechanical structure of frog retina model.

FUNCTIONAL ELECTRONIC MODEL OF THE FROG RETINA

A functional electronic model of the frog retina has been constructed at DEP Applied Research which simulates many qualitative and quantitative properties of the visual system of the frog. Because of the unique construction techniques employed, intermediate logic outputs may be visually observed. The model abstracts visual features by using deterministic parallel processing and overlapping responsive-receptive fields — two features that are novel in the field of visual-pattern recognition. This equipment should provide a useful research tool for further studies in pattern recognition, feature abstraction, and parallel processing. The model, while itself capable of only limited performance, may well be a significant first step toward the design of complex equipment for surveillance, reconnaissance, and vehicular guidance.

M. B. HERSCHER and T. P. KELLEY
Applied Research
DEP, Camden, N.J.

THE FUNCTIONAL ELECTRONIC MODEL of the frog retina described herein is based on the physiological work of Lettvin and others[1,2,3], who measured the electrical signals that the frog retina was sending to the colliculus (the brain "mapping" center). These investigators distinguished four classes of optic-nerve fibers; each class carries different information regarding a specific feature of the image on the retina. Their studies revealed that the frog retina performs a sorting operation in which four classes of properties are abstracted: 1) edges, 2) moving convexities, 3) contrast changes and 4) dimming.

Fig. 1 shows schematically some of the qualitative properties of the feature-abstraction processes performed by the frog retina. The circle at the left (labeled *input*) contains four small figures representing images which might be presented to the retina; the arrow indicates the direction in which the images are assumed to be moving. Circles I through IV illustrate how these images are "mapped" into the colliculus. Circle I shows that the edge detectors abstract both stationary and moving edges. Circle II shows the features abstracted by the moving-convexity detectors; since only dark convexities moving toward the center of the field of view are detected, only the trailing edge of Image 1 and the entire outline of Image 2 are abstracted. In abstracting changes of contrast, the leading and trailing edges of all moving dark images are abstracted as shown in Circle III. Circle IV shows that the dimming detector abstracts only the leading edges of moving dark images.

In biological systems, the size of the *responsive-receptive field* (RRF) associated with the visual-image information-reduction process is an important consideration. The relative size of the RRF is proportional to the number of receptors projected onto one of the ganglion cells in any class. This projection is a result of a many-to-one transformation which occurs in the retina system. In the visual system of the frog, there is a specific size for the RRF which is associated with each of the four classes of feature-abstracting ganglion cells. The size of each receptive field increases as the ganglion class increases from 1 to 4.

The frog retina model which has been constructed possesses many of the specific feature-abstraction properties found in the frog. In addition, the model simulates other physiological properties such as the incorporation of overlapping receptive fields for each class of ganglion cell, and the preservation of the relative RRF size for each class of ganglion. The model also has a general physical correspondence to the anatomy of the frog retina; i.e., processing takes place in various neural layers. Since the ganglion cells of the frog are sensitive to both the speed and the size of the image presented to the retina, this sensitivity also is incorporated in the functional model.

GENERAL DESCRIPTION OF RETINA MODEL

This functional model employs seven processing layers which are mounted vertically as illustrated in Fig. 2. Fig. 2b shows where each logic function is located, while Fig 2a is a photograph of the actual retina model. The six layers perform parallel processing on the information presented to the retina. The seventh layer contains the output indicators and power distribution system for the model. Each of the six active processing planes may be removed from the main frame for testing or servicing. Fig. 2 shows layer 6 in this position. The interconnections for the first four processing layers are made by light transmission from neon lamps in the back of one layer to photoconductors in the front of

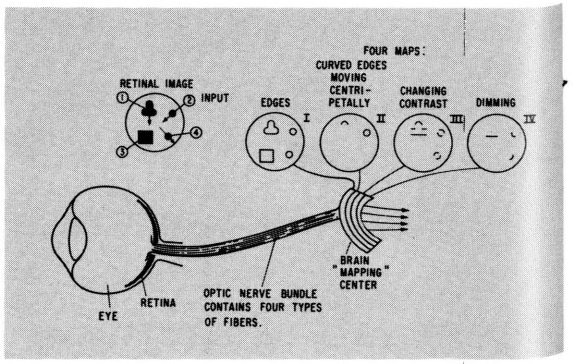


Fig. 1 — Schematic of frog visual system.

M. B. HERSCHER received his BSEE in 1953 and his MS in 1959 from Drexel Institute of Technology, and is presently enrolled in the PhD program at the University of Pennsylvania. His experience includes work at the Frankford Arsenal on electronic fuse circuitry. In 1953 he joined the Circuit Development Group in RCA. He spent 21 months as an officer in the U.S. Army Signal Corps, where he worked on transistor circuits as applied to frequency control, and returned to RCA in 1956. Since then, he has worked on development of high-power transistor audio amplifiers, transistor-magnetic pulse circuitry, nonlinear-amplitude coding devices, and semiconductor linear mixers. Recently, he was project engineer on a study of intelligence information processing and a transistor magnetic video recorder. Currently, he is a Project Engineer responsible for applied research on visual pattern recognition. He has taught several transistor circuit theory courses at RCA and the Fort Monmouth Signal Labs. He has published papers and is a coauthor of the 2nd edition of the "Handbook of Semiconductor Electronics."

T. P. KELLEY received his BS in Physics from St. Joseph's College, Philadelphia, Pa. in June 1961. He is currently studying for an MSEE at the University of Pennsylvania. Since joining RCA, Mr. Kelley has made contributions in many fields, including radar receivers, photovoltaic cells, and digital transmission. More recently, he has been conducting pattern-recognition studies. Specifically, he has studied the application of parallel logic schemes to pattern recognition problems, and the implementation of parallel logic through the use of light-emitting and light-sensing devices. This recent work has led directly to the design and construction of the functional electronic model of the frog retina. Mr. Kelley is a member of the Franklin Institute.

the following layer. In this manner, light provides the majority of the synaptic connections of the model. The neon lamps and photoconductors serve also as the principal circuit components for the threshold logic.

The output of the ganglion cells are available in two forms: 1) relay contact closures which provide binary electrical outputs, and 2) incandescent lamps which provide a two-dimensional display of the four features abstracted. The Class 1 and Class 2 indicator lamps provide binary visual outputs, while the Class 3 and Class 4 lamps provide analog outputs.

The following guidelines were used in the design and fabrication of the model.

1) A construction technique was selected which would permit individual logic functions to be fabricated on individual layers. This technique greatly simplifies the wiring and interconnection of components.
2) All components in each layer were made readily accessible, thereby permitting easy repair.
3) The logic for the various feature-abstraction functions was isolated to simplify the testing procedures.
4) Neon-lamp photoconductor interconnections were used between the first four processing layers to permit all circuits on each layer to be individually tested by observing the visual outputs.
5) The model was designed so that individual layers of logic may be modified, and/or replaced. In addition, logic changes can be made in some of the present layers by simple changes of wiring.
6) The model was designed to simulate the anatomy of the frog by using logic processing layers and by preserving the relative location of each ganglion class within these layers.

SYSTEM LOGIC

The system logic required to accomplish the feature-abstraction operations performed by the four classes of ganglia of the frog retina model is presented in this section. This logic is a modification of the logic originally described by E. E. Loebner[4].

Edge-Detection Neuron (Class 1)

The edge-detection neuron abstracts edges from the input image. In order to detect an edge, the existence and position of horizontal and vertical contrasts on the receptor matrix are detected by the bipolar neurons. Then, if a sufficient number of bipolar neurons of the same directionality are activated, an edge is presumed to exist in the receptive field of the Class 1 ganglion cell.

The basic contrast decisions are made by the bipolar neurons. These neurons are located in the first layer of the retina model and consist of two photoconductive receptors connected to two neon lamps in a balanced-bridge circuit. When contrast exists between the paired-input receptors, an appropriate neon lamp (bipolar output) is activated, thereby identifying the location and direction of contrast. Each bipolar neuron, then, provides two directional outputs per pair of input receptors. Since 1,296 input receptors are employed, there are 648 bipolar neurons which provide 1,296 outputs.

The activation of particular combinations of directional bipolar outputs results in the detection of an edge of minimum length. This detection is accomplished by summing four adjacent bipolar outputs which possess the same contrast directionality into a majority-of-four decision-maker as shown in Fig. 3. This majority logic is located in the edge-detection layer, where additional photoconductive cells receive the directional contrast information from the neon-lamp outputs of the receptor-bipolar layer. Thus, an edge will be de-

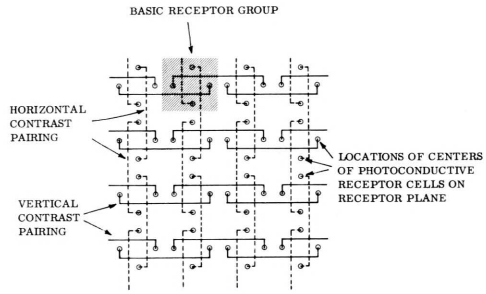


Fig. 3 — Portion of receptor matrix. This shows the bipolar pairing of receptors (Class 4 receptors are omitted in this figure).

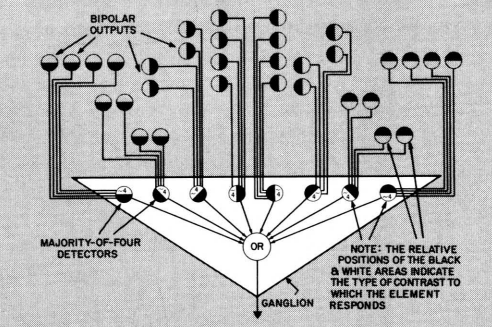


Fig. 4 — Edge-detecting ganglion cell (Class 1).

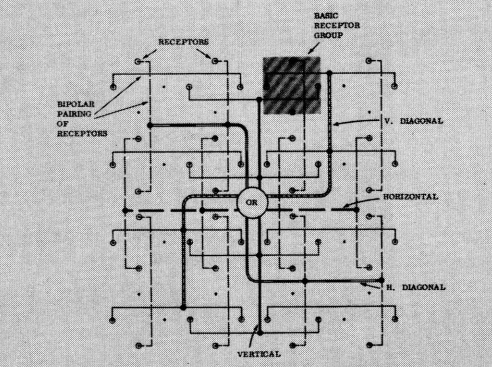


Fig. 5 — Complete layout of a typical Class 1 ganglion cell.

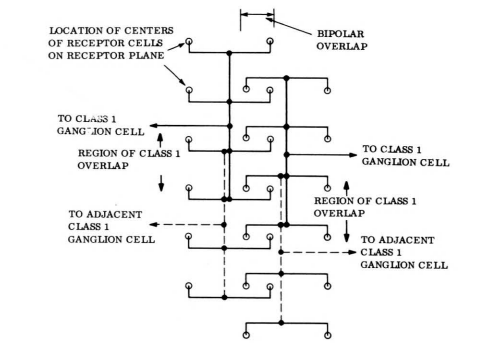


Fig. 6 — Portion of receptor matrix. This shows the bipolar pairing of receptors and summations of bipolars into majority-of-four detectors for *vertical* edge detection (other summations omitted for clarity).

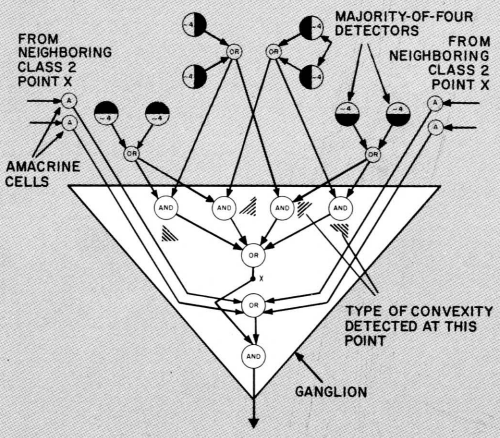


Fig. 7 — Tracked convexity ganglion cell (Class 2).

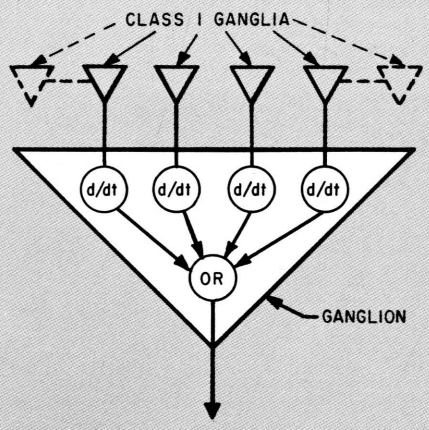


Fig. 9 — Changing contrast ganglion cell (Class 3).

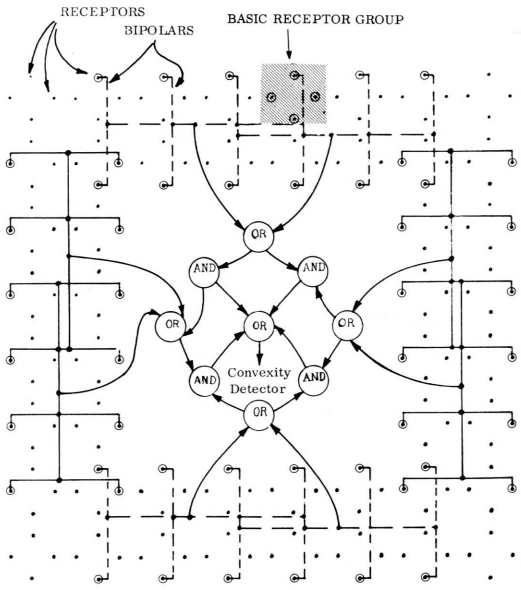


Fig. 8 — Portion of receptor plane showing logic connections for Class 2 ganglion cell.

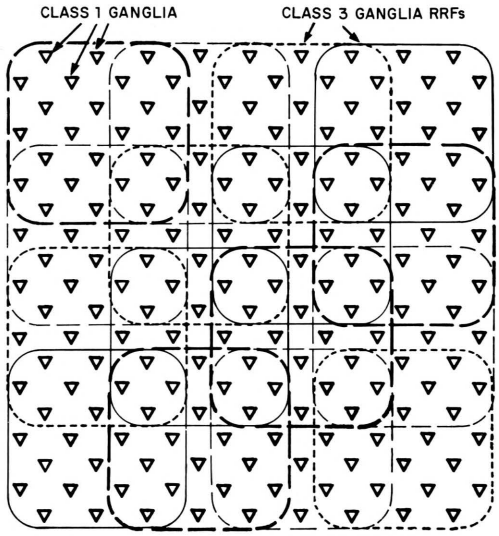


Fig. 10 — Class 3 ganglion cell overlap. This shows the overlap of Class 3 ganglia (squares), as well as the particular Class 1 ganglia (triangles) combined to form a Class 3 ganglion cell.

tected when an edge focused on the receptor plane is long enough to cause an output from at least three of four adjacent bipolars.

As stated above, the bipolar neurons determine the existence of horizontal and vertical contrasts only. It might seem that this restriction limits the detection of edges to only horizontal and vertical types. (Actually there is a range about the major axes of approximately $\pm 35°$ in which the presence of edges may be detected.) It is not necessary, however, to confine the bipolar neuron summation to adjacent horizontal or vertical bipolars. By summing several nonadjacent bipolar neurons of similar directionality, it is possible also to abstract diagonal edges. This additional summation permits the detection of all the possible orientations of edges appearing on the receptor matrix. Fig. 4 shows diagrammatically how eight connections of the four types of bipolar neuron outputs (two horizontal and two vertical types) are summed in an *or* gate to provide an output for the Class 1 ganglion cell. Thus a Class 1 ganglion output indicates the presence of an edge in a given location, irrespective of its orientation. The directional summations which make up the inputs for a Class 1 ganglion cell are shown in Fig. 5. This spatial arrangement leads to a Class 1 RRF consisting of 4 x 4 receptor-pairs.

An important feature of the model is the incorporation of overlap at various processing levels. For the Class 1 ganglia, overlap is present in the pairing of receptors and in the summation of bipolar neurons for the majority-of-four detectors. Each majority detector overlaps its neighbor by two bipolars. This feature, together with the requirement that only three of four inputs are necessary to activate the majority detector ensures that there are no insensitive spaces between neighboring majority detectors. In addition, this overlap also provides a form of spatial redundancy which reduces the possibility that a single malfunction will cause an error in the output. The ganglion cell overlap, as well as the receptor-pair overlap, is shown in Fig. 6. The overlap of ganglia results in a total of 180 Class 1 outputs for the retina model.

Moving-Convexity Neuron (Class 2)

The moving-convexity neuron abstracts dark convexities moving in a given direction. Inputs to the Class 2 ganglia are derived from the majority-of-four detectors of the Class 1 ganglion, as shown in Fig. 7. These inputs are obtained by sensing the neon-lamp outputs in the sec-

ond layer with photoconductive cells located in the third layer. Two overlapping majority-of-four detectors (which have the same directional contrast) feed an *or* gate.

A similar operation is performed on the output of six other bipolar neurons which are sensitive to contrast in a direction perpendicular to the first group of bipolar neurons. The second group of neurons also must have the proper spatial location with reference to the first group in order to be combined in an *and* gate, and thereby indicate the presence of a dark convexity. The output signals of the four possible combinations of convexities then are combined in an *or* gate to denote the detection of any convexity within the field of view of a particular Class 2 ganglion.

Fig. 8 shows the logic connections for a Class 2 ganglion cell, together with the relative spatial locations of the receptors. The RRF of a Class 2 ganglion cell for a frog is approximately 50 percent larger than that of the Class 1 ganglion cell. Therefore, on a relative basis, the RRF for the Class 2 ganglia of the model has six receptor-pairs on a side, as shown in Fig. 8.

In order to obtain an output from a Class 2 ganglion cell, it is necessary that a dark, convex object move across the input-receptor matrix in a given direction. The neural mechanism which provides tracking of lateral motion in the model is the *amacrine cell.*

Fig. 7 shows the location of the amacrine cells in the Class 2 ganglia logic. It is the function of the amacrine cells to provide a short-term memory of the presence of a convexity within the RRF of a convexity detector. The output (neon lamps) from the convexity detectors located in the third layer activate the inputs (photoconductive cells) to the amacrine circuits which are located in the fourth layer. The output of an activated amacrine is fed to specific adjacent Class 2 ganglion cells, enabling the *and* gate of these neurons to pass an output when a convexity is present in their receptive fields. Thus, a convexity will be tracked if it moves from one receptive field to another within a certain velocity range. For the Class 2 ganglia, a large amount of overlap is provided between the receptive fields of adjacent neurons. The model provides 90 output signals for the Class 2 ganglion cells.

Changing-Contrast Neuron (Class 3)

The changing-contrast neuron abstracts changes of contrast of relatively large dark objects appearing in its receptive field. As shown in Fig. 9, the Class 3 neuron of the model obtains its inputs from the Class 1 ganglia outputs derived previously. Many of these Class 1 outputs are summed in overlapping areas, then differentiated and fed through an *or* gate. Through the summation process, a large RRF for each Class 3 ganglion cell is obtained. Fig. 10 shows the arrangement of Class 1 outputs, represented by triangles. The 16 Class 3 ganglion cells are represented by large squares. This particular type of summation was chosen in order to obtain the proper receptive-field size for the Class 3 ganglia.

Since it is desirable to obtain analog information from the Class 3 ganglion cell, the *or* function is obtained by the linear addition of the transient voltages. The analog output therefore is a measure of the size and speed of the input image. Sixteen output signals from Class 3 ganglia are provided in the model.

Dimming Neuron (Class 4)

The dimming neuron abstracts changes in dimming appearing in its receptive field. The information for this neuron is most readily obtained by using separate receptors located on the input (receptor-bipolar) layer. These receptors take the central position in the basic block of receptors as shown in Fig. 11. Since the RRF of the dimming neuron is very large, many Class 4 receptors are summed before additional logic operations are performed. This summation is shown schematically in Fig. 12. Dimming is obtained by differentiating the summed receptor potentials (in order to detect motion of the input image) and then generating an *off* response to denote that dimming occurs. The *off* response is an analog output which simulates the envelope of the integrated physiological *off* response. The large overlapping receptive fields shown in Fig. 13 yield five independent output indications for the Class 4 ganglia.

SUMMARY

The functional model of the frog retina which has been constructed incorporates the four basic feature-abstraction functions found in the visual system of the frog.

The 4,580 logic operations contained in the model require the use of over 32,000 individual circuit components, including 3,793 photoconductive cells and 2,652 neon lamps. More than 2,000 neon-lamp-photoconductor pairs provide "light connections" between processing layers.

Because of the unique construction techniques employed, intermediate logic outputs may be visually observed. The model abstracts visual features by using deterministic parallel processing and overlapping responsive-receptive fields. The inclusion of these two features are novel in the field of visual-pattern recognition. This equipment therefore should provide a useful research tool for further studies in pattern recognition, feature abstraction, and parallel processing. The model, while itself capable of only limited performance, may well be a significant first step toward the design of complex equipment for surveillance, reconnaissance, and vehicular guidance.

ACKNOWLEDGEMENT

This work was suported by the U.S. Air Force under Contract AF33 (616)-7734; the work was administered under the direction of the Electronic Techniques Laboratory of the Wright Air Development Division.

BIBLIOGRAPHY

1. J. Y. Lettvin, H. R. Maturana, W. S. McCulloch, and W. H. Pitts, "What the Frog's Eye Tells the Frog's Brain," *Proc. IRE*, 1959, *47*, 1940-1951.
2. H. R. Maturana, J. Y. Lettvin, W. H. Pitts, and W. S. McCulloch, "Anatomy and Physiology of Vision in the Frog," *J. of Gen. Physiol.*, 1960 42 Suppl., 129-175.
3. J. Y. Lettvin, H. R. Maturana, W. H. Pitts, W. S. McCulloch, "Two Remarks on the Visual System of the Frog," *Sensory Communications*, M.I.T. Press, 1961.
4. E. E. Loebner, *Image Processing and Functional Retina Synthesis*, (1960 Bionics Symposium), WADD Technical Report 60-600, 309-338.

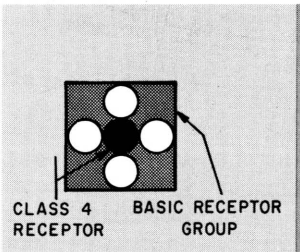


Fig. 11—Location of Class 4 receptors in basic receptor group.

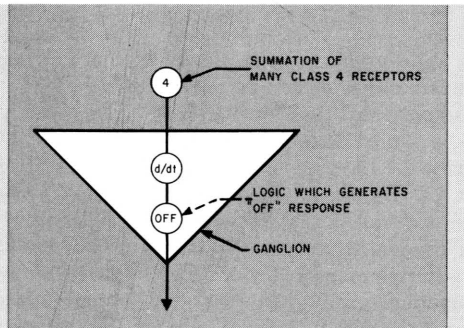


Fig. 12 — Dimming ganglion cell (Class 4).

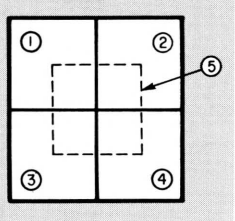


Fig. 13—Overlapping RRF's for the 5 Class 4 ganglia.

AUTOMATIC TEXT PROCESSING

A number of character-recognition devices are under development which hopefully will transcribe hard-copy text material to a form useable by a computer. Increasing attention is now being given to the problem of what the computer can do with this tremendous data base. The goals of such computer processing of natural-language text include automatic retrieval of data, automatic indexing, automatic formatting of data for subsequent logical and manipulative processing, and so on. A number of possible techniques might be suggested to achieve these goals: fast scanning techniques for searching text for occurrences of words in particular combinations; word counting techniques which will extract word lists in order to approximate an indexing function; word association techniques to group words which have some semantic relationship; grammatical analysis techniques which will make explicit the functions of the words in sentences so that levels of information and word relationships can be determined. This paper discusses each of these techniques.

W. D. CLIMENSON
Data Systems Center
Bethesda, Md.

WILLIAM D. CLIMENSON received his BSEE magna cum laude from Duke University in 1953 and his MSEE from the University of Pennsylvania in 1958. He joined RCA in 1957 and worked on Project ACSI-MATIC concentrating on natural language information to be handled by the computer. He set up automatic syntax analysis programs, and has done basic language control research into linguistics and computer-language problems. As Leader, Language Analysis and Documentation Handling Group in the Data Systems Center, Mr. Climenson has ancillary responsibilities for document handling and medical data-processing programs. He is a member of ACM and Vice-Chairman of its Special Interest Group on Information Retrieval. He is the author of several papers in his field.

OVER THE PAST ten years, considerable attention has been given to the challenging task of reading printed text by machine. A number of specialized devices geared to a well-controlled reading environment are already in operation to process bank checks, credit-card purchases, and public-utility accounts. Progress is being made in the development of devices to read typewritten or printed text, and even handwritten material.

Certainly one of the goals of this development is the ability to automatically transcribe data available in reports, books, and periodicals to a form which can be accepted by a computer. The question which one must immediately raise is: *What can the computer do with these new sources of data?* Some of the goals are:

automatic language translation
automatic retrieval of the stored texts
automatic abstracting of texts
automatic extracting of text portions in a form suitable for manipulating and correlating data
cryptoanalysis

But such ambitious goals can only be met with equally sophisticated and advanced computer processing techniques. It is curious that when such tasks are suggested as appropriate to computers, one usually hears either of two extreme reactions: "impossible" or "obviously feasible." It is hoped that the reader will be left with the impression that the current state of the art is somewhere between these extremes; that the computer can be productive as an assistant in the tasks normally expected of a text processing function.

COMPUTERS AND FORMATS

Before some promising computer techniques for dealing with raw text are discussed, it would be useful to consider the problem as one of formatting. If we view the capabilities of the computer in a deterministic sense, we must use terms such as *record* and *item descriptions*, *parameters*, and *bounds* when we describe a task to be performed; i.e., we expect data to obey certain format rules. Otherwise we could not predict the outcome of computer operations. Thus, a major task in data processing is defining and adjusting formats so that data can be recorded and processed (or rejected) in a consistent fashion. In short, natural languages such as English are quite *un*natural to the computer and must be manipulated to conform with the machine characteristics if any productive work is to be done.

Thus we are faced with somewhat of a paradox: we want to process unformatted data directly with a machine which can only accept formatted data. The answer must lie in a sort of bootstrapping operation, starting with a minimum of format information which is available in natural language orthography:

a limited character set.
a limited punctuation set (including word, sentence, paragraph, and document delimiters).
a large, but finite, number of legal combinations of symbols.

While we take comfort in the fact that the combinatorial possibilities are finite, the central problem is to recognize an organization in the seemingly chaotic nature of natural language. Can we state what is *legal* in the language? If so, is the set of rules for forming utterances (which we call a grammar) of a character and size which the limited capacity of a computer can handle?

TEXT ANALYSIS

We might start by matching the simpler characteristics of the language with the elementary functions which a computer can perform, build on the insight that such an analysis provides, and recognize and test possible operational applications as we proceed.

First, we can certainly compare letter sequences, count matches (or mismatches), and order the results. We could do this at the word or word sequence levels, and suggest, for example, that words which occur a sufficient number of times can be interpreted as descriptive of a document's content. We can tack on to this process a list of reference words which permits us 1) to discard common "function" words (such as *in, the, for*), and 2) upgrade the importance of some words which we know to be indicative of content regardless of the number of times they occur.

At first glance this notion, as simple as it is, seems to provide striking results. Authors obviously select words in a nonrandom fashion, and word-frequency distributions tend to exhibit clustering around significant words. But, on the other hand, good writing style sometimes demands the use of synonyms, and a given word spelling can have two or more unrelated meanings. This will degrade the value of word-frequency lists. We could overcome this difficulty to some degree by storing synonym and homograph word lists, and "respelling" texts so that synonyms become indistinguishable and homographs are assigned subscripts prior to the application of counting techniques.

This basic word-look-up, word-counting approach (with many variations) has been tried for three applications

which require a document description or characterization:

automatic indexing, where a list of words ordered by their significance is used.

automatic abstracting, where sentences which best satisfy some statistical criterion are extracted from the document.

direct text searching for information, where sentences containing selected key words are extracted and examined.

None of these applications has been very successful; to the author's knowledge, these techniques have never been advanced to an operational stage. There are two basic reasons for this: English orthography simply does not contain sufficient information which can be used directly for such intellectual tasks as indexing and abstracting, and the "respelling" of words in text which was suggested above requires a satisfactory solution for a more difficult problem, that of semantics.

THE SEMANTIC PROBLEM

An appreciation of the rich reservoir of concepts which English allows us to communicate can be gained by playing a game with *Roget's Thesaurus:* pick two words at random from the thesaurus index. The object of the game is to get from one word to the other, using the cross references available in the thesaurus categories, with as few intermediate words as possible. For example, we can trace a path from *pacify* to *punish* as follows:

pacify	723.4
harmonize	23.8
adjust	27.7
compensate	30.4
repay	807.9
retaliate	718.2
punish	972.5

Any two adjacent words could be considered as synonyms, but a different aspect is selected to determine the next synonym in the chain. This suggests that if we could isolate these aspects of meaning as a list of elementary concepts such as *movement, size, relation* and the like, we could synthesize any complex concept we need, and more important, we could state explicitly how two concepts are related. In principle, this sounds like a good way to work on the problem of nailing down the definitions of words, and therefore suggests an approach to a vocabulary format for computer processing of text.

Unfortunately, we can see that the solution is still as far away as before when we realize that there would be more than one way to synthesize a concept; i.e., the rules for synthesizing concepts constitute a sort of language grammar which has all the characteristics of the natural language we are trying to control. This is the problem which has been plaguing investigators concerned with automatic language translation (and the entire field of linguistic analysis, for that matter).

With such a pessimistic introduction, the reader may begin to wonder if we know anything at all about the language we use for communicating so effectively. We do, in fact know quite a bit about natural language, particularly its structure. By *know*, we mean that we have been able to state formal rules for constructing sentences, for example. Secondly, more sophisticated mathematical notions, when applied as carefully as they are defined, have been shown to be useful. The remainder of this paper will be concerned with these linguistic and statistical techniques.

AUTOMATIC SYNTAX ANALYSIS AND ITS APPLICATIONS

A description of sentence structure is fundamental to any linguistic approach to text processing. If it were possible to determine the relationships of words in sentences, we could systematically develop the necessary processes needed to analyze sequences of sentences, paragraphs, and documents. This philosophy has motivated an investigation into natural-language structure by the Language Analysis Group at the DEP Data Systems Center in Bethesda, Md. Considerable progress has been made in automatic "parsing" or "diagramming" of English sentences. An RCA 501 program has been written which accepts declarative sentences, marks the limits of phrases and clauses, and displays the results.

An example of the printed output for two sample sentences is given in Fig. 1. The sentence is read vertically, one word per line, with indention to indicate nesting of phrases and clauses. The symbols on the extreme right show the limits of phrases (called first-order strings): *N*, noun phrases; *V*, verb phrases; *A*, adjuncts (prepositional or adverbial phrases). The prefixed numbers indicate a phrase count. For example, in the second sentence the sixth adjunct is "*by a quick and decisive victory*," which contains the eighth noun phrase "*a quick and decisive victory.*" Similarly, the symbols in the center column indicate the limits of clauses or degenerate clauses (called second-order strings). Three types of second-order strings are recognized: *MN*, main string in independent clause; *NL*, nounlike (entire strings which behave as nouns in a larger string); *IG*, ignorable (strings which do not play a vital syntactic role in the larger string). In addition, suffix symbols are used to indicate the beginning of a second-order string subject *(S)*, the limits of the verb *(V)*, and the end of the object *(O)*.

The steps needed to perform this analysis are shown in Fig. 2 and described below:

1) After initial screening, the words in the sentence are replaced by the syntactic class(es) of the word. These correspond in a rough way to the conventional parts-of-speech.
2) Words assigned to more than one

Fig. 1—Example of printed output for two sample sentences (1 and 2).

SENTENCE 1

```
    WHILE                1 1G
    THE                  1 1GS          1 N
    MONSOON                             1 N

       WHICH                    2 1G
       HAD                      2 1GV  1 V
       JUST                                    1 A
       STARTED                  2 1GO  1 V

,
    TURNED               1 1GV          2 V
    #KOREA'S                            2 N
    DIRT                                2 N
    AND

       GRAVEL                           3 N
       ROADS                            3 N
       INTO                             2 A
          BOGS           1 1GO          2 A  4 N

,
  THE              1 MNS                5 N
  ENEMY                                 5 N

  MANAGED          1 MNV                3 V
  TO
  FERRY            1 MNV                3 V

  HIS                                   6 N
  ARMOR                                 6 N
  ACROSS                                3 A
  THE                                            7 N
  #HAN                                  3 A      7 N
  AND
        ADVANCE                         8 N
        INTO                            4 A
           THE                                    9 N
           OUTSKIRTS                    4 A      9 N
        OF                              5 A
           #SUWON   1 MNO               5 A     10 N
```

SENTENCE 2

```
    IN                   1 1G
    DRAWING                             1 V
    UP
       THEIR                            1 N
       TIMETABLE         1 1GO          1 N

,
  THE              1 MNS                2 N
  COMMUNISTS                            2 N
  APPARENTLY                            1 A

  ASSUMED          1 MNV                2 V

    THAT                 1 NL
    THE                  1 NLS          3 N
    INITIAL
    ADVANTAGE                           3 N
    OF                                  2 A
    A                                          4 N
    STRONG
    SURPRISE
    ATTACK                              2 A    4 N
    IN                                  3 A
    GREAT                                      5 N
    FORCE                               3 A    5 N
    AGAINST                             4 A
    THE                                        6 N
    #REPUBLIC                           4 A    6 N
    OF                                  5 A
    #KOREA                              5 A    7 N
    WOULD                1 NLV          3 V  PASSIVE
    BE
    FOLLOWED             1 NLV          3 V
    BY                                  6 A
    A                                          8 N
    QUICK
    AND
    DECISIVE
    VICTORY                             6 A    8 N
    FOR                                 7 A
    THE                                        9 N
    FORCES                              7 A    9 N
    OF                                  8 A
    COMMUNISM    1 MNO   1 NLO          8 A   10 N
```

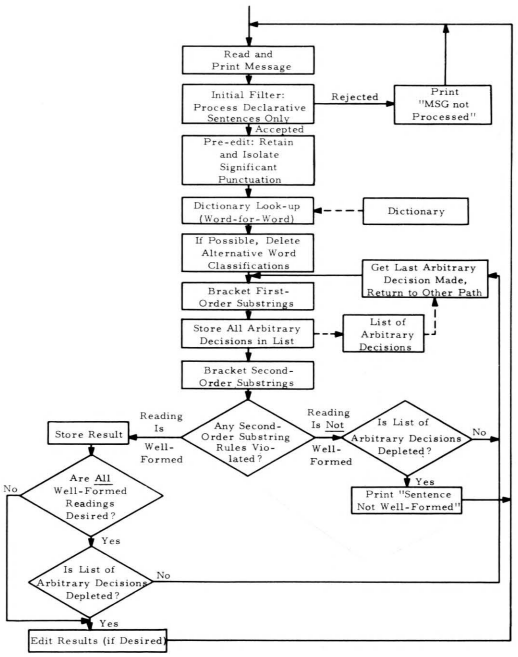


Fig. 2—Basic program steps in syntax analysis.

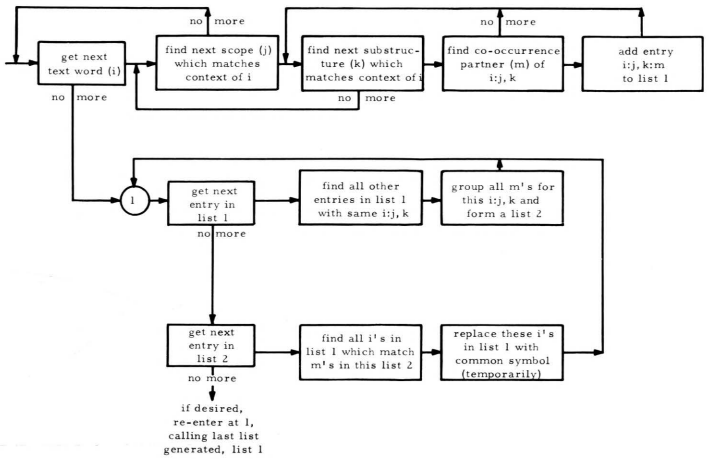


Fig. 4—Generation of a document thesaurus.

class (such as *surprise*, which can be a noun or verb) are examined to see if their specific function in this particular context can be resolved.

3) The first-order strings (phrases) are bracketed. This is done by a pass through the sentence for each of the three phrase types, searching for allowable word sequences.
4) On a final pass through the sentence, second-order strings are isolated. During this pass, the program searches for allowable sequences of first-order strings (rather than of individual words) to form clauses.
5) Because some multiclass words could still remain unresolved after step 2, the search for first-order strings may include some arbitrary decisions. Rather than stop the analysis at these points and seek to resolve the difficulty, the program chooses one logical path and notes in passing that the decision may be wrong. Thus when the second-order analysis is complete, so-called "well-formedness" criteria may not have been met (e.g. a clause may lack a needed noun phrase). If this happens, the analysis returns to the point where the last arbitrary decision was made and follows a new path.

The program, while imperfect, can provide a "good" reading in about 80 percent of the sentences expected in normal written discourse, and it seems feasible to expect that the program could be refined to an operational status. If this is so, how can the results of such an analysis be used? A number of interesting possibilities are suggested in the following paragraphs.

Sentence Condensation

First, we can hypothesize that there is a correlation between the position of a word or phrase in a sentence and its significance to the document. To test this hypothesis, a sentence-condensation program was written for the RCA 501 which uses the syntactic information available in sentences as the sole criterion for reducing the length of the document. Machine implementation of a limited set of rules applied to this data produced a reduction to about 35 percent of original document length. An example of such a condensation is shown in Fig. 3, which was produced from a section of a book on the history of the Korean War. This type of intermediate use of automatic syntax analysis shows promise, but needs to be investigated further.

Thesaurus Generation

A second application of automatic syntax analysis is in the development of a specialized thesaurus. The principal difficulty with a semantic analysis even in a limited subject field is that the meaning

THE ATTACKING COMMUNIST DIVISIONS MOVED SWIFTLY ACROSS THE 38TH PARALLEL TOWARD SEOUL, FIFTY MILES TO THE SOUTH, AN BY 28 JUNE 1950, THREE DAYS AFTER THE BEGINNING OF THE NORTH KOREAN INVASION, THE CAPITAL HAD FALLEN INTO ENEMY HANDS. THE ELEMENT OF TACTICALSURPRISE AND THE NORTH KOREANS' OVERWELMING SUPERIORITY IN WEAPONS CURSHED ORGANIZED ROK RESISTANCE IN THE VICINITY OF THE PARALLEL. NORTH KOREAN TANK COLUMNS GROUND FORWARD UNSCATHED AGAINST THE INEFFECTIVE REPUBLIC OF KOREA FORCES WHICH LACKED TANKS AND ADEQUATE ANTITANK WEAPONS. THE WILD EXODUS OF REFUGEES FROM SEOUL SWELLED THE POPULATION OF THE TWON OF SUWON, WHICH LAY A FEW MILES BELOW THE CAPITAL. INDIVDUAL ROK SOLDIERS DISPLAYED A WILL TO FIGHT, BUT COMMUNICATIONS WERE IN A STATE OF CHAOS, AND AMMUNITION OF EVERY KIND WAS RUNNING LOW. ON 27 JUNE THE SEAT OF THE SOUTH KOREAN GOVERNMENT WAS TEMPORARILY MOVED FROM SEOUL TO TAEJON BELOW THE HAN AND KUM RIVERS. ALTHOUGH U.S. BOMBERS ATTACKED P'YONGYANG AND TARGETS NEARER THE FRONT, ENEMY PLANES FREELY STRAFED THE SUWON AIRSTRIP WHICH WAS NEAR THE HEADQUARTERS OF BRIG. GEN. JOHN H. CHURCH, THE COMMANDER OF ADCOM. WHILE THE MONSOON, WHICH HAD JUST STARTED, TURNED KOREA'S DIRT AND GRAVEL ROADS INTO BOGS, THE ENEMY MANAGED TO FERRY HIS ARMOR ACROSS THE HAN AND ADVANCE INTO THE OUTSKIRTS OF SUWON. RUSSIAN T34 TANKS, RUSSIAN VEHICLES, RUSSIAN COMBAT PLANES, AND RUSSIAN AUTOMATIC WEAPONS WERE USED AGAINST THE PEOPLE OF SOUTH KOREA. THOUSANDS OF NORTH KOREAN SOLDIERS WITH YEARS OF SERVICE IN THE OTHER COMMUNIST ARMIES PROVIDED THE HARD CORE OF THE INVADING TROOPS. THERE CAN BE NO DOUBT THAT SOVIET ADVISERS PLAYED AN OVERWELMING PART IN PLANNING THE OPERATION. IN DRAWING UP THEIR TIMETABLE, THE COMMUNISTS APPARENTLY ASSUMED THAT THE INITIAL ADVANTAGE OF A STRONG SURPRISE ATTACK IN GREAT FORCE AGAINST THE REPUBLIC OF KOREA WOULD BE FOLLOWED BY A QUICK AND DECISIVE VICTORY FOR THE FORCES OF COMMUNISM.

Fig. 3a—Original text to be processed.

Fig. 3b—Text condensation.

ATTACKING DIVISIONS MOVED ACROSS 38TH PARALLEL TOWARD SEOUL; THREE DAYS AFTER NORTH KOREAN INVASION BEGINNING, CAPITAL HAD FALLEN INTO HANDS. SURPRISE ELEMENT, OVERWHELMING SUPERIORITY CRUSHED ROK RESISTANCE IN PARALLEL VICINITY. NORTH KOREAN GROUND FORWARD AGAINST KOREA FORCES REPUBLIC WHICH LACKED TANKS, ADEQUATE WEAPONS. REFUGEES WILD EXODUS SWELLED SUWON TOWN POPULATION. INDIVIDUAL ROK SOLDIERS DISPLAYED WILL; COMMUNICATIONS WERE IN CHAOS STATE; EVERY KIND AMMUNITION WAS RUNNING LOW. SOUTH KOREAN GOVERNMENT SEAT MOVED FROM SEOUL TO TAEJON. PLANES STRAFED SUWON AIRSTRIP. ENEMY MANAGED TO FERRY ARMOR. RUSSIAN PLANES; AUTOMATIC WEAPONS USED AGAINST SOUTH KOREA PEOPLE. THOUSANDS OF NORTH KOREAN SOLDIERS WITH SERVICE YEARS PROVIDED TROOPS HARD CORE. COMMUNISTS ASSUMED THAT ATTACK INITIAL ADVANTAGE IN FORCE AGAINST KOREA REPUBLIC FOLLOWED BY VICTORY FOR COMMUNISM FORCES.

of a word depends on the *context* in which it is used. While context is a very elusive concept, a structural analysis at the sentence level can be used as an elementary approach to its formal definition.

The particular approach suggested here avoids the pitfalls of setting up a single thesaurus for the entire language; instead it is hoped that small, document-oriented thesauri can be constructed and that some of these may be eventually joined into larger organizations of vocabulary. If this is possible, the "respelling" of words in a text which was mentioned earlier could become quite useful.

Assuming that a syntax analysis has been performed on a text, we could proceed as follows (see Fig. 4).

1) Search the text for all occurrences of a given word. For each occurrence note the structure (context) in which it is used. This must be done with knowledge that these structures are described at various levels of precision. Those levels of structure which can be defined formally we will call "substructures;" the higher, less defined portions of text we will call "scopes."
2) Attempt to match word occurrences in the same type of structure, starting at the most precisely defined level.
3) Where a match is discovered within some limits of scope and substructure, find for each word its "co-occurrence partner" among the other units of the substructure (e.g. the co-occurrence partner of the head of a noun phrase might be the object of a succeeding preposition). List these co-occurrence partners together.
4) Replace all occurrences of all the words on such lists by a single, common symbol (word).
5) Reenter the process, this time using the text in its "normalized" form, and repeat the procedure until some criterion is satisfied.

While this procedure has not been mechanized, manual simulation has shown that some significant semantic implications can be drawn from the results. Note that there was no mention of word meaning in the above; the procedure makes use of the author's structuring of his ideas to determine when and where words can be considered synonyms *in this text.*

Paragraph Analysis

Paragraph Analysis, a third application of automatic syntax analysis, is really an extension of it. That is, we would like to be able to state "well-formedness" rules for units of text larger than the sentence. Because the variety of ways to form sentences into paragraphs is greater than that of putting words in sentences, we could expect a paragraph "grammar" to be less precise.

A paragraph, in this analysis, is similar to the orthographically marked paragraph in text in that both are concerned with unity of topic. But there is one important difference: Paragraph analysis considers a set of paragraph markings as its *output*, not its input. Therefore we are free to use as complex a paragraph structure as necessary to explicitly mark the relationships between sentences (just as we marked the relationships between words in syntax analysis).

The relationship we are concerned with is one of dependency, which is manifested in two ways:

If some one element of a paragraph structure is deleted, other elements which depend on it must also be deleted.

When certain dependencies are established, it is possible to treat whole groups of sentences as a single element in subsequent analysis.

Some simple examples of sentence dependency rules are given below (these can operate only after a syntax analysis on the text has been performed):

1) A sentence (clause) containing a noun phrase with *this* at the left depends on the first preceding clause which contains a noun phrase with the same word at its head (usually the right most word in the noun phrase).
2) A clause introduced by a word in a certain class of conjunctions such as *although*, depends on the preceding clause.
3) A clause containing *its* plus noun phrase depends on the next following clause which has the same noun phrase head.

In addition to rules such as these, paragraph analysis makes use of rules to organize the discovered dependencies, and partition the document into paragraphs. These take the form of topological and linguistic well-formedness criteria.

The availability of structural data above the sentence level makes feasible a number of useful tasks in processing textual material. A few possibilities are listed below.

The notion of sentence condensation can be extended to permit deletion of entire clauses or sentences in order to systematically reduce the size of a document.

The resultant paragraph structure can be examined to determine the minimum text unit which can be considered informationally independent, therefore indexable.

The previously mentioned method of generating a thesaurus could be enhanced since the paragraph structure adds precision to the notion of context.

Transformations

An important aspect of structural linguistics is the recognition and explicit marking of equivalent or similar structures. Considerable work in this area at the sentence level has been done under the name "transformational analysis." This analysis provides the means for reducing the number of structures possible in the language in a systematic fashion. Thus transforming or rewriting a sentence is to the language's structure as respelling of a text (from a thesaurus) is to its vocabulary.

If both of these could be realized at the level of context required, we would be within reach of our formatting goal. Context is again mentioned here because care must be exercised not only in determining the types of transformations to be applied to the entire language, but where and when to apply them in the local environment of the text. For example, we can specify a passive-voice to active-voice transformation:

His buddy carried John.
$\rightarrow$*John was carried by his buddy.*

But if we met the latter sequence out of context, we would not know the transformation to use to answer the question: Who does the *his* refer to?

This type of processing and a description of it are very important for two basic reasons. First, it is as fundamental as syntax analysis in any attempt to describe language. Secondly, all of the applications in linguistic analysis mentioned previously could be more manageable if a previous (or a concurrent) transformational analysis was performed.

Statistical Analysis

As mentioned previously, statistical techniques applied directly to natural language texts cannot promise more than rudimentary results. But it should be obvious that if something close to a "well-controlled" sample can be found, statistical techniques will offer much in the analysis of documents.

The purpose of the following discussion is not how the needed control of data might be achieved (perhaps manually or with the use of the suggested linguistic techniques); rather we will assume that a sufficient degree of text normalization has been obtained.

Counting

In addition to the counting techniques

suggested earlier, mention should be made of a method for "normalizing" word counts by using the ratio of their *relative* frequency in the document to their relative frequency in the file. This notion would attempt to characterize a document by determining how it differs from the remainder of the file. Thus, words which were equally common (or rare) in the file and in the document would be ignored. Words which occurred more in one document than would be expected (as predicted by the file statistics) would be selected as significant.

Correlation

Correlation techniques are pertinent to text processing in three ways:

The retrieval of data which is only implicitly stored in the file.

The amplification of requests for data.

The characterization of the file as a whole.

In each of these we are getting farther away from the text itself and working, in most cases, with document descriptor lists.

Basically, we are interested in determining numeric values for word (or descriptor) relationships. For example, starting with a group of selected words and their occurrences in file documents we can generate a word-document matrix, with each cell containing N_{ij}, the number of occurrences of word i in document j. Using conventional correlation techniques we can develop correlation coefficients for each word-pair. Similarly we could obtain a measure of correlation between two index terms using the number of documents indexed by both terms, and the number of documents indexed by each term (without regard for the other). Other parameters have been used as well.

The value of correlating terms can be seen if reference is again made to the semantic problem and context. That is, we can legitimately ask questions about the relationships of terms within the context of the file or group of data we are working with (and not the entire universe of concepts). For example, the effectiveness of a document-retrieval system can be enhanced considerably if in addition to retrieving documents, the system retrieves information *about the file* to help the requestor formulate his query. Specifically, the original terms in a request can be correlated with all other terms used in the file to determine which new ones (the requestor neglected to use) ought to be appended to the request. These new terms might be checked against the file for additional correlations.

Secondly, correlation techniques are important in measuring the dynamic elements of the data to be processed. New trends and aspects of a particular subject area can be detected. One method of extending correlation techniques in the area of file characterization is discussed below.

Factor Analysis

It is often quite difficult to analyze a matrix of correlation coefficients because of the complex relationships which one must examine in order to determine, say, the overlap of two pairs of elements. The problem of interpreting correlation matrices has been tackled by psychologists who have developed a technique called "factor analysis" to assist in the task of finding common elements in the variables of a testing situation. The technique has been carried over to the document-retrieval field. In this application, factor analysis has been used to extract "factors" from a descriptor correlation matrix. A factor consists of a list of related descriptors each with a numeric weight (or loading), which are extracted by a matrix reduction process.

The various factors extracted from the matrix (lists of terms) can be interpreted and assigned names. They should (and do) show the gross classes of data in a document file and relationships of the terms used in document descriptions. Dynamic document classification might be possible using this technique.

SUMMARY

It is hoped that techniques borrowed from psychology, graph theory, linguistics and so on will develop into useable tools in text analysis. Some progress has been made already. Because of the tremendous increase in technical and administrative documentation and the anticipated availability of reading machines more progress *must* be made.

There are two basic reasons why research in automatic text processing has produced disappointing results thus far:

1) The potential customer wants to see operational results in a relatively short time. This has directed the researcher's attention to *ad hoc* and obvious techniques which seem to promise some payoff quickly.
2) The researcher himself is either impatient to see useable results, or at the other extreme, is pursuing relevant tasks but at a leisurely academic pace, far removed from the realities of the problem at hand.

As a result of these two factors, we have seen a number of simple techniques worked and reworked by different people at considerable cost with little success.

It seems that an approach closer to that which we might call "goal-oriented linguistic-theoretic" is needed. This requires considerable discipline and patience from both the investigator and his sponsor, but could be more successful in the long run.

One could point to numerous examples of research in text processing supported on this basis. Three projects being carried on at the DEP Data Systems Center might be mentioned:

1) *Fact Correlation Study*, performed for the Rome Air Development Center, which supplied to the Air Force an applied research plan in text processing.
2) *Advanced Recognition Techniques Study*, performed for the U. S. Army Electronics Materiel Agency, which is concerned with the use of contextual clues in improving the performance of character readers.
3) *Fact Correlation, Phase IIa*, for Rome Air Development Center, which is concerned with a benchmark program in concept correlation; specifically, the problem of automatic reassembly of paragraphs into the documents from which they were extracted.

BIBLIOGRAPHY

A paper of this type normally demands a rather complete bibliography. In lieu of this, a list of basic references is included whose topics and internal references should lead to the desired information.

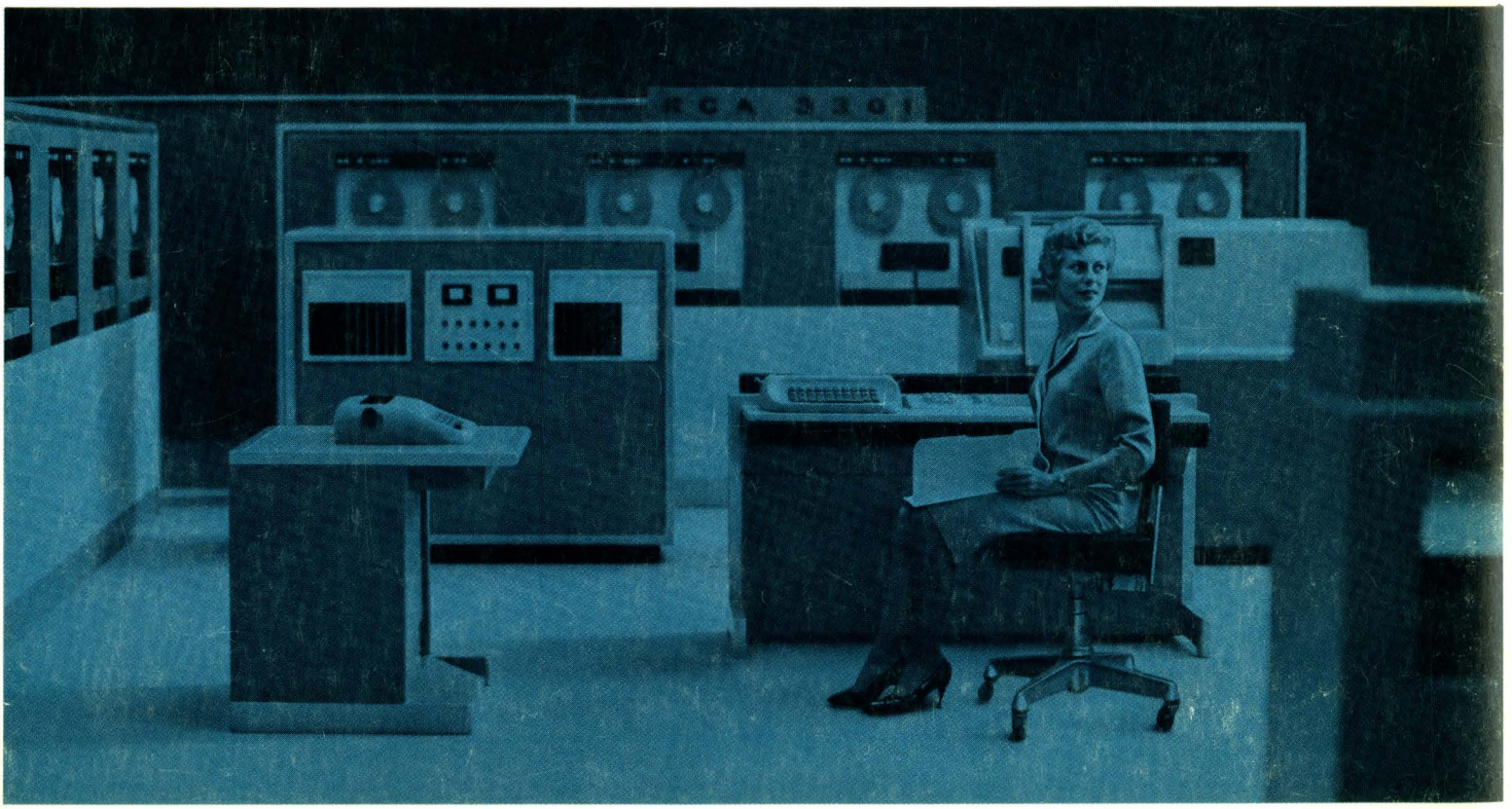
1. *Current Research and Development in Scientific Documentation*, Office of Science Information Service, National Science Foundation, numbered bulletins published semi-annually.
2. *Foreign Developments in Machine Translation*, Joint Publications Research Service, Office of Technical Services, U. S. Department of Commerce, numbered translated reports.
3. "Machine Indexing, Progress and Problems", *Proceedings of the Third Institute on Information Storage and Retrieval*, The American University, February 1961.
4. *Mechanical Translation*, Massachusetts Institute of Technology, published quarterly.
5. *Organization of Information for Storage and Retrospective Search*, Preprints of papers for the International Conference on Scientific Information, Area V., National Academy of Sciences, November 1958.
6. *Transformations and Discourse Analysis Projects*, Department of Linguistics, University of Pennsylvania, numbered reports.
7. S. R. Snodey, "Information Retrieval Systems and Technology—A Literature Survey", *IEEE Transactions on EWS*, (in press; to appear in Fall 1963); A comprehensive bibliography of world literature on information retrieval for the period 1957-61.

For further information, phone or write a nearby RCA EDP Sales Office:

- ATLANTA, Suite 1201, Georgia Power Bldg. 270 Peachtree St., 525-6547
- BOSTON, 886 Washington Street, Dedham, DA 6-8350
- CHICAGO, Room 220, Morton Salt Bldg., 110 N. Wacker Dr., ST 2-0700
- CLEVELAND, 1600 Keith Bldg., 1621 Euclid Ave., CH 1-3450
- DALLAS, 7901 Carpenter Freeway, ME 1-3050
- DENVER, 2401 East Second Avenue, 399-1460
- DETROIT, Room A3-300, Southfield Office Plaza Bldg., 17000 W. Eight Mile Road, Southfield, 356-6150
- HARTFORD, 80 Farmington Ave., JA 7-4143
- LOS ANGELES, RCA Bldg., 1550 N. Vine St., HO 6-4101
- MIAMI, Gulf Building, 95 Merrick Way, Coral Gables, 445-5487
- NEW YORK CITY (Downtown) 45 Wall St., (Uptown) 1250 Avenue of Americas: MU 9-7200 (both offices)
- PHILADELPHIA, Suite 1909 2 Penn Center Plaza, LO 8-8150
- PITTSBURGH, 222 Four Gateway Center, CO 1-1080
- SAN FRANCISCO 343 Sansome St., YU 1-5600
- SEATTLE, 1111 Washington Bldg., 1325 Fourth Ave., MA 2-4234
- ST. LOUIS, 7710 Carondelet Ave., Clayton, PA 6-5322
- SYRACUSE, Room 302-303, State Tower Bldg., GR 4-5337
- WASHINGTON, 1725 "K" St., N. W., FE 7-8500

RCA ELECTRONIC DATA PROCESSING RCA-CHERRY HILL CAMDEN 8, N. J.

The Most Trusted Name in Electronics



Form No. 80-32-234 Rev.

Printed in U.S.A.