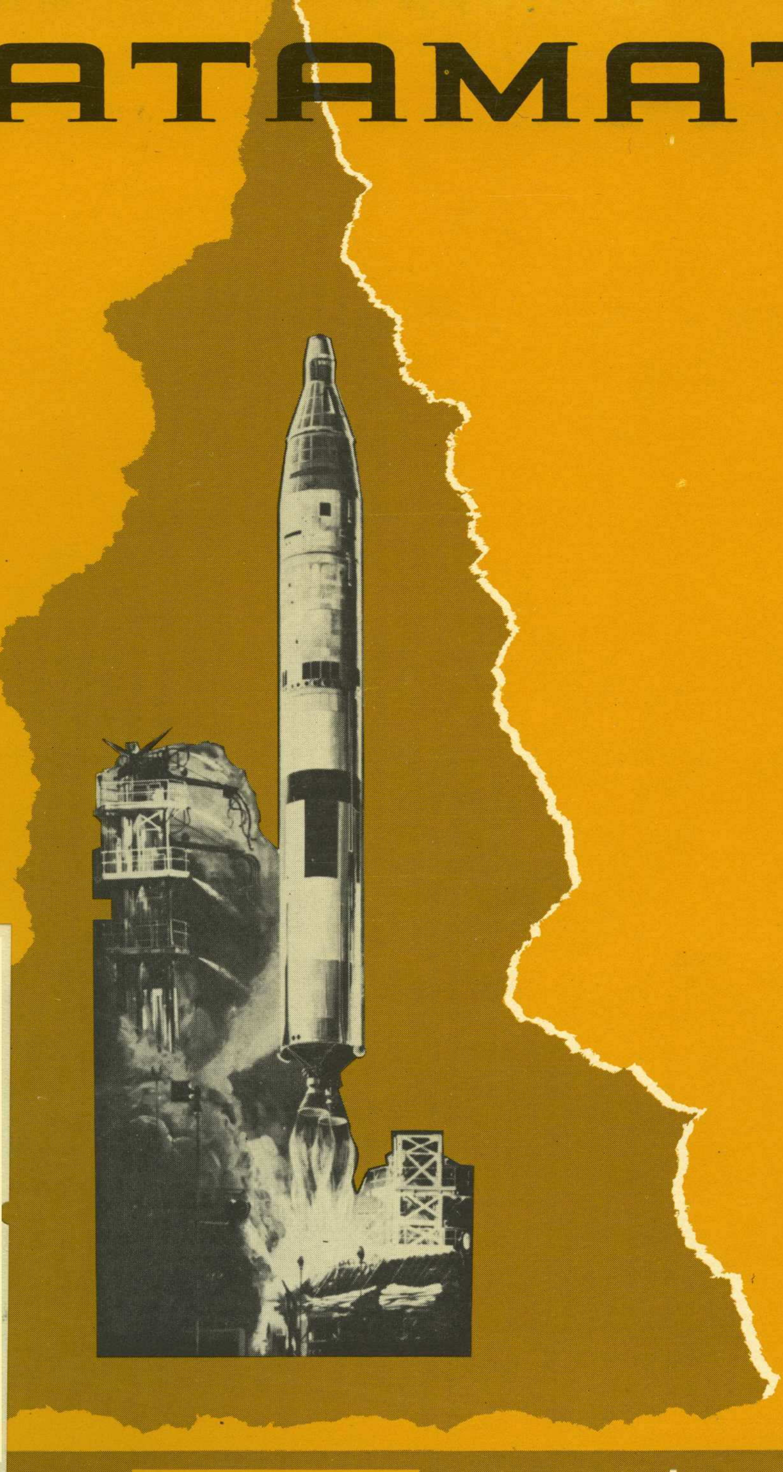
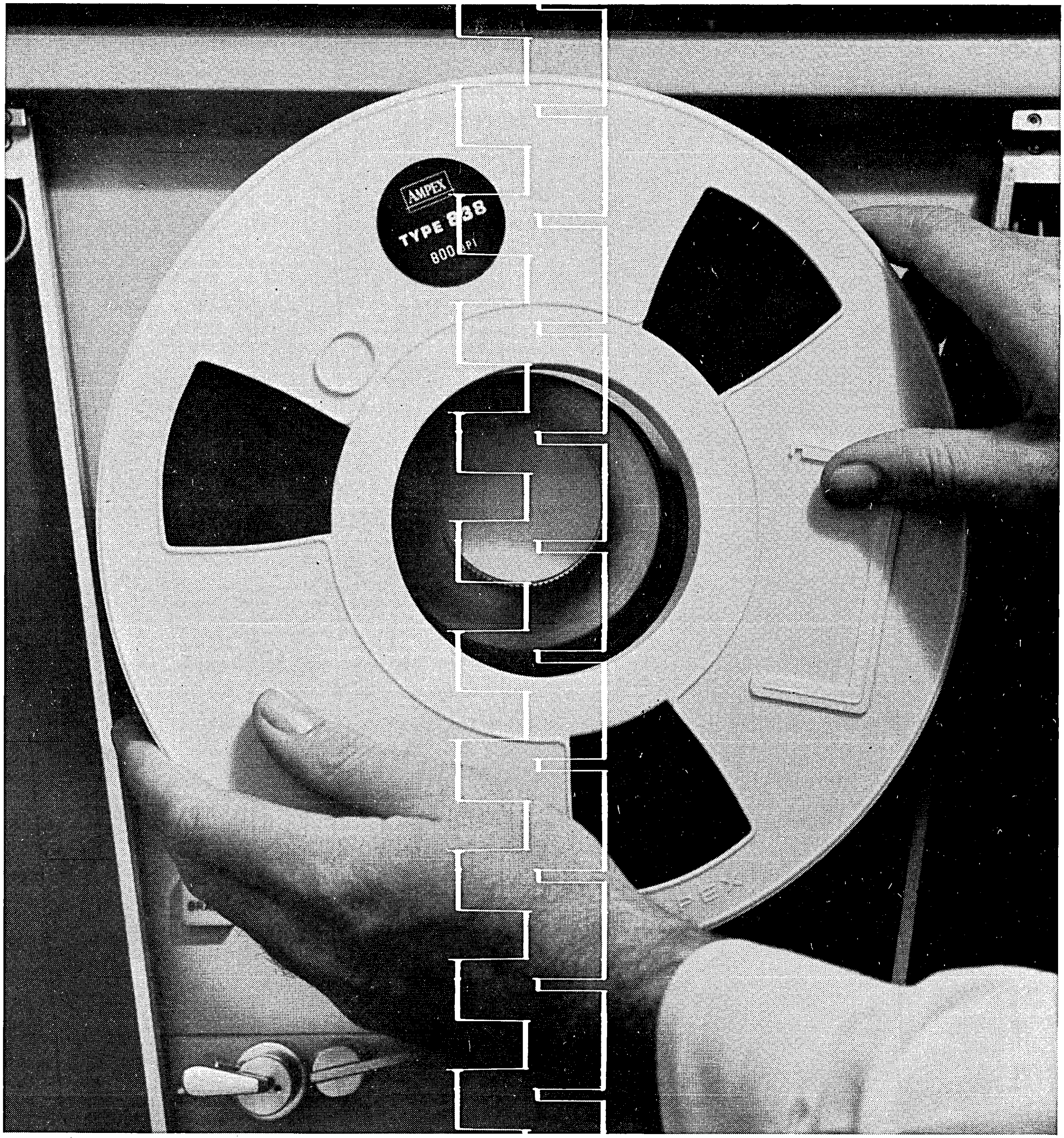
DATAMATION

63

May



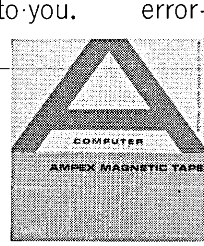
programming on-line systems



Who wants you to test their computer tape before you buy it?

AMPEX

We think we've got a computer tape that's next to perfect. And we'd like to prove it. We'll bring you a tape to test out on your own system—at no obligation to you. (All Ampex computer tapes are fully compatible with IBM systems.) And each reel you buy is guaranteed to meet specifications on the first read pass—or we'll replace the tape. Here's why we can be this confident: Before we pack a tape for shipping, we digitally check it from end to end on a system compatible with the one on which it will be used. That way we know it's completely free from even one permanent error. And our tapes stay error-free longer. An exclusive Ferro-Sheen* process, together with an improved binder, keeps the surface smoother; reduces headwear and oxide build-up. For more information write the only company providing recorders, tapes, core memory devices for every application: Ampex Corp., 934 Charter Street, Redwood City, Calif. Worldwide sales and service.



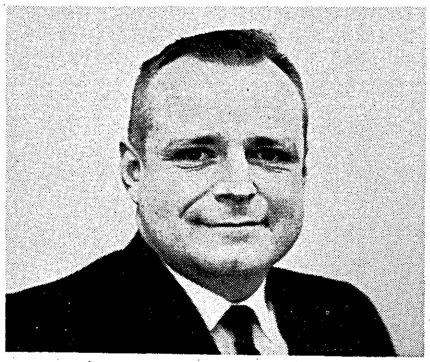
AMPEX

*TM Ampex Corp.

CIRCLE 1 ON READER CARD

SDS's Borgers Discusses Software Packages

Los Angeles—According to Emil Borgers, Manager of Programming for Scientific Data Systems, Inc., "the 'comprehensive software packages' available from computer manufacturers seldom live up to advertising claims. This is particularly evident with smaller computers where the 'package' may consist of little more than a simple octal loader, a sketchy assembler and a few subroutines based on reference book equations."



BORGERS

The SDS spokesman continued, "right from the start, we were determined to lick the software problem before we put the first SDS computer out the door. We ran our hardware and software programs in parallel, with each program dictating design criteria for the other. That way—refusing to simply tack software development onto the tail-end of our computer design program—we were able to offer the most efficient and comprehensive software package in the small computer field, right off the bat. That's why the $98,000 cost of an SDS 920 includes a four-level package of utility programs, assembly programs, mathematical subroutines and Fortran II, plus a continued program of refinement, improvement and software extension."

Borgers stressed the SDS Fortran capability: "Fortran II is a subset in our SDS 900-Series compiler. The SDS 920 is the only computer that, without magnetic tape units, can process Fortran II programs in one pass. Right now, considerably less than a year after our first computer was installed, SDS customers are solving a wide range of production problems using our Fortran. Comparative tests have already proved it to be a superior Fortran processor. And our unique diagnostic capability, efficiency and compilation speed are impossible to achieve with any other computer near our size."

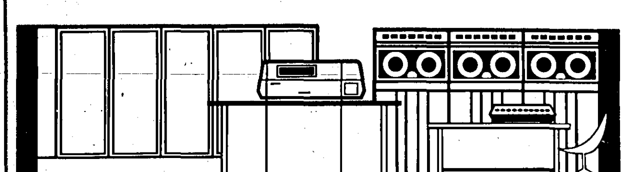
Borgers concluded, "We at SDS feel that we've provided the efficient type of programs that users want but are frequently left to write for themselves."

"We take our software seriously."

NOTE: Complete detailed literature on the SDS software package is available, on request from Dept. S, 1649 Seventeenth Street, Santa Monica, California.

Meet the new welterweight champ: the Honeywell 1400

This new big brother to the Honeywell 400 can lick anything anywhere near its own weight. The fastest, dual-purpose intermediate computer around, its 6.5 microsecond access time means speed on the order of 14,000 binary additions per second. Optional floating-point arithmetic and multiply-divide instructions make it a dual-purpose workhorse equally proficient at scientific as well as business data processing tasks.



More memory, more simultaneity

The memory for the new system is available in modules of 4,096 words up to a maximum of 16,384 words. In comparing this with other computers, it is important to remember that a Honeywell word is 12 decimal digits or 8 alphabetic characters, and 16,384 words are equivalent to 196,608 decimal digits or 131,072 alphabetic characters.

A key feature of the Honeywell 1400 is its ability to double-up on many operations. It can read cards, compute, and print at the same time. Or it can read on one tape while writing on another with both moving at full speed. And two high-speed printers can be operated simultaneously.

Handles more tape units

The ability to control up to eight magnetic tape units is standard with the Honeywell 1400. Control of an additional eight is optional. Three models of tape units — economy, standard and high density — are available, with transfer rates of 48,000, 96,000, and 133,000 decimal digits per second, respectively. All tape units feature Honeywell's famous vacuum-actuated tape transport design and Orthotronic Control, a unique automatic error detection-correction system.

Meets any peripheral requirements

Peripheral equipment includes a 900-line-per-minute high speed printer which can be operated on-line or off-line, and print storage and control options to permit simultaneous printing by two printers; a 650 card per minute card reader; and a reader/punch that reads 850 or punches 250 cards per minute. A card storage option permits card reading or punching concurrent with other operations.

Paper tape equipment, disc storage units, optical scanning and Orthoscanning devices and communications control units are also available for use with the Honeywell 1400.

Features EASY programming

The Honeywell 1400 central processor has many features that make it easy to program. These include: three-address instructions, choice of binary, alphanumeric or decimal information, program interrupt, three index registers, edit instructions and masking operations. Programming aids for the Honeywell 1400 are compatible with the Honeywell 400 system. EASY assembly language, COBOL, and the AUTOMATH scientific (algebraic) compiler form the basic software package. Programs for the Honeywell 1400 can be run and tested on Honeywell 400 or large-scale Honeywell 800 computers.

Has speed and precision to spare

The use of unique three-address instruction enhances the Honeywell 1400's high internal operating speeds. Typical add-time for a single three-address instruction is 77 microseconds. The floating point arithmetic unit uses a 12 digit number system, which includes a sign (one digit), exponent (two digits) and mantissa (nine digits).

Rounds out the Honeywell line-up of champions

The 1400 is the latest addition to the fast growing line of Honeywell computers. Purchase price, depending on size and configuration, ranges from $450,000 to $900,000; corresponding monthly rentals range from $10,000 to $20,000. Delivery time is nine months. For a complete run-down on the new 1400 — or, for that matter, the Honeywell 400, Honeywell 800 or Honeywell 1800 — just contact any Honeywell EDP sales office or write to Honeywell EDP, Wellesley Hills 81, Mass. In Canada, Toronto 17, Ontario.

Honeywell

ELECTRONIC DATA PROCESSING

CIRCLE 5 ON READER CARD

DATAMATION

the automatic handling of information

volume 9, number 5

EDITORIAL OFFICES

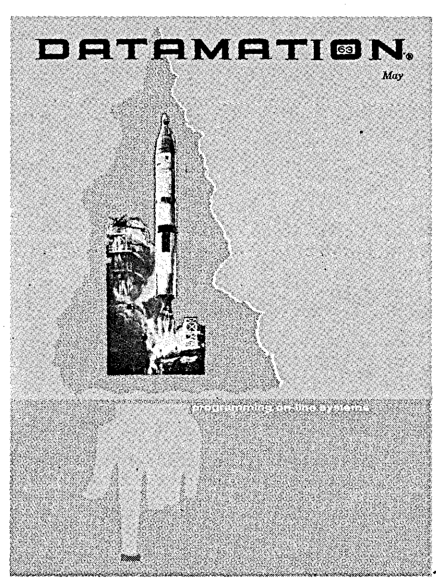
1830 WEST OLYMPIC BLVD.
LOS ANGELES 6, CALIF.

Microfilm copies of **DATAMATION** may be obtained from University Microfilms, Inc., 313 No. First St., Ann Arbor, Mich.

Feature Articles

Departments

THIS ISSUE—46,298 COPIES



Cover

Programming aspects of on-line systems with response times ranging typically from 250 usec to three minutes are examined this month in the first of a two-part series. One of many such real-time situations is portrayed on the cover, designed by Art Director Cleve Boutell.

Circulation audited by Business Publications Audit — BPA

Member, National Business Publications — NBP

DATAMATION is published monthly on or about the tenth day of every month by F. D. Thompson Publications, Inc., Frank D. Thompson, president. Executive, Circulation and Advertising offices, 141 East 44th St., New York 17, N.Y. (MUrray Hill 7-5180). Editorial offices, 1830 W Olympic Blvd., Los Angeles, 6. Published at Chicago, Ill. Controlled Circulation paid at Columbus, O. Form 3579 to be sent to F. D. Thompson Publications, Inc., 201 N. Wells St., Chicago 6, Ill.

Printed by Beslow Associates, Inc.

important DATES

• A National Telemetering Conference will be held May 20-22 at the Hilton Hotel, Albuquerque, N.M. Sponsors are the AIEE, ARS, IAS, IRE, and ISA.

• A COINS (COmputer and INFormation Sciences) symposium on Learning, Adaptation, and Control in Information Systems will be held June 17-18 at Northwestern Univ., Evanston, Ill., which is a co-sponsor with the Office of Naval Research, Information Systems branch.

• The Fourth Joint Automatic Control Conference will be held at the Univ. of Minnesota, Minneapolis, on June 19-21. Sponsors are the American Institute of Chemical Engineers, IEEE, and American Society of Mechanical Engineers.

• The annual International Data Processing Conference and Business Exposition, sponsored by the Data Processing Management Association, will be held June 25-28, at Cobo Hall, Detroit, Michigan.

• The sixth annual Summer Conference, sponsored by the Northwest Computing Assoc., will be held August 8-9 at the Pacific Science Center, Seattle, Wash.

• The Western Electronics Show and Conference (WESCON) will be held August 20-23 at the Cow Palace, San Francisco, Calif. Sponsors are Region 6, WEMA, and all PGs.

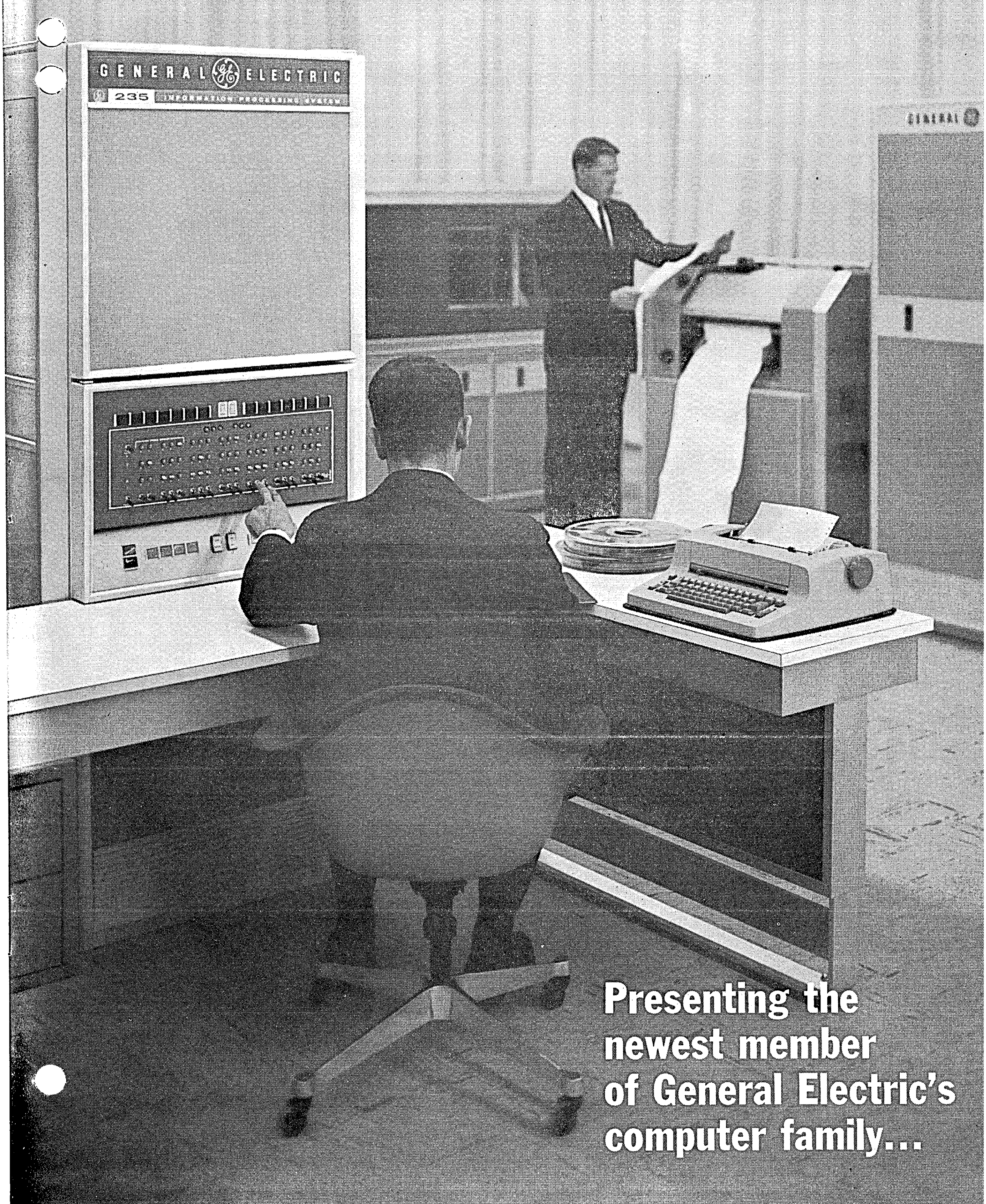
• The 1963 ACM National Conference will be held Aug. 28, 29, and 30th in Denver, Colorado.

• The second Institute on Electronic Information Display Systems will be held September 16-20 at The American Univ., Washington, D.C.

• The 1963 Fall Joint Computer Conference will be held in the Las Vegas, Nev., Convention Center, Nov. 12-14.

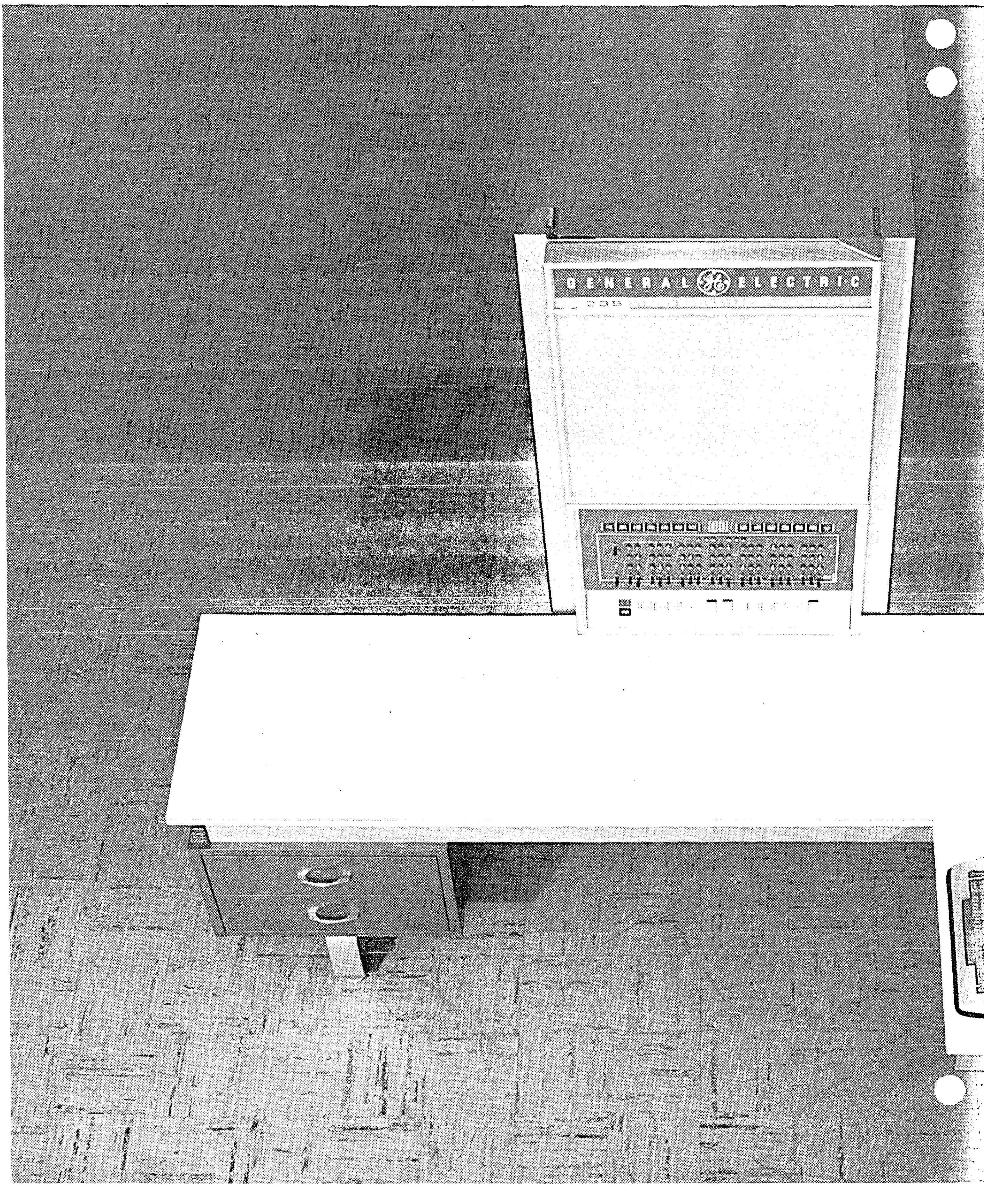
• The 1964 Spring Joint Computer Conference will be held at the Washington Hilton Hotel, Washington, D.C., May 26-28.

Computer progress at General Electric



Presenting the newest member of General Electric's computer family...

Now the powerful new



letters

attitudes on positivism

Sir:

I have just finished reading the first part of Paul Armer's article "Attitudes Toward Intelligent Machines" which appeared in the March issue of DATAMATION. I was (prior to reading the article) one of the negativists of whom he speaks. I thoroughly enjoyed being dissuaded from my position by his arguments and look forward to reading the remainder of the article.

Ann Solomon
System Development Corp.
Santa Monica, Calif.

Sir:

While the mass of serfs, in medieval times, toiled prodigiously to earn their daily crust, a number of pundits were engaged in endless debate anent the burning question of their day,—to wit: how many angels can dance simultaneously on the point of a needle?

For almost two decades, I have been toiling prodigiously to stuff the ravenous maw of the Automatic Computer and could barely find the time to peruse the seemingly endless debate in which my betters are engaged anent the burning question of today,—to wit: is a pile of wires, crystals, and tubes able to think?

Inasmuch as lovely trees must be mangled into pulp in order to carry that deathless prose, I pray that my brief comment on the discussion will help to curtail this ruthless carnage:

If you must have machines that think,
Because your gray stuff's on the blink,
ACs manufactured are in vain,
For only God can make a brain.

Ida Rhodes
National Bureau of Standards
Washington, D.C.

Sir:

The article by Mr. Paul Armer in your March issue "Attitudes Toward Intelligent Machines" is a contribution which should do much in putting into proper perspective the pointless efforts of those who would endeavour to down-grade the present and future potentialities of the great computers. I am definitely a "Positivist" within the meaning of Mr. Armer's definition.

However, I feel he exhibits a narrowness of thinking in the statement "A bird isn't sustained in the air by the hand of God—natural laws govern its flight." Who, does he think, put the natural laws there which would sustain the bird in flight? Certainly no man, and no creation of man!

We humans are not inventors, we are only discoverers of possibilities which have been in existence for longer than any of us can comprehend.

A. G. MARJERISON
Canadian Pacific
Montreal, Quebec

Sir:
Paul Armer seems to be putting Descartes before the horse in his discussion of intelligent machines. "It am, therefore it thinks."

ED CONTI
UNIVAC
San Diego, Cal.

Irish displeasure

Sir:
Regarding the St. Patrick's Day article by J. Granholm in your March issue, what an idiotic, empty, not-funny piece of writing for you to carry in an erstwhile good publication!!! Besides a first-grader's attempt to be "clever" why bring good St. Patrick's name in and slush it over???

This edition should have been 92 pages long.

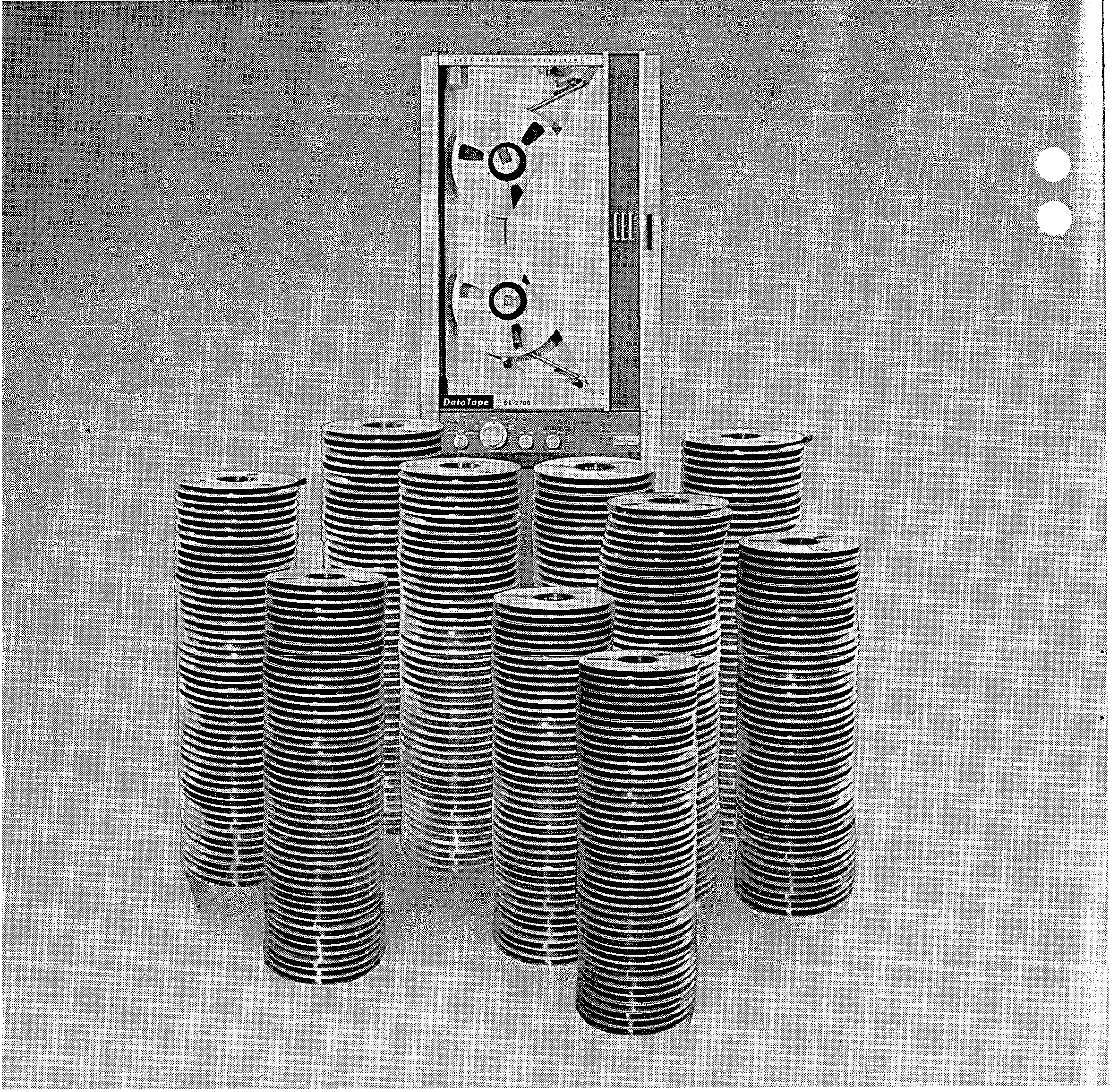
CHARLES HOOPER
Townson, Md.

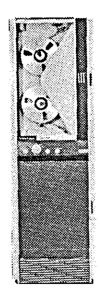
(Editor's Note: Actually, St. Patrick was shown the article prior to publication and thought it was well above the first grade level.)

oops!

Sir:
I have been a reader of DATAMATION for some time and it especially hurts to see an error of omission in your usually accurate journal. Your news brief about the award to Stanford University for a computer based teaching lab says that there is but one other installation of that kind, and that is at the System Development Corporation in Santa Monica. I think that if your news sources had checked John Coulson's (of SDC) 1962 book entitled *Programmed Learning and Computerized Instruction* (Wiley, 1962), they would have discovered that a number of other installations of this kind are active. The University of Illinois has not one but two such laboratories. Bolt, Beranek and Newman of Cambridge, Massachusetts have been working in this field, and last, and I hope not least, our group at IBM Research has an eight-terminal system operating in a fully multiprogrammed fashion with a curriculum of three courses.

WILLIAM R. UTTAL
IBM
Yorktown Heights, N.Y.

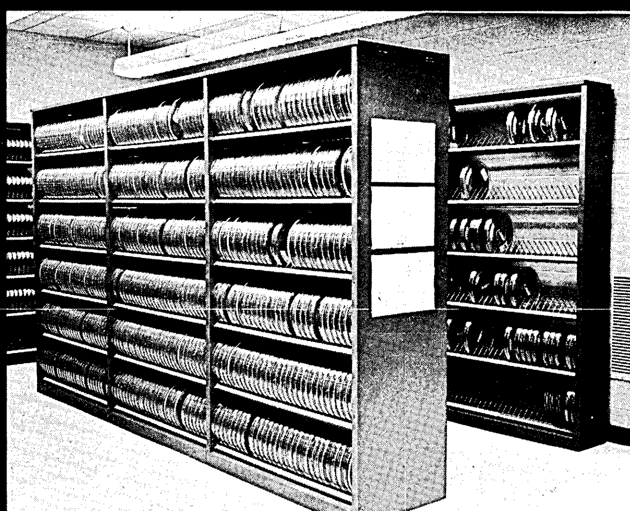

DataTape



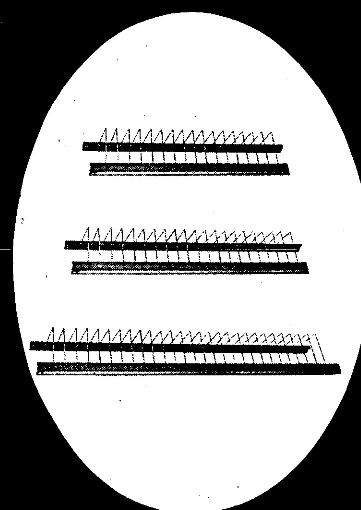
DESIGN

makes the difference*

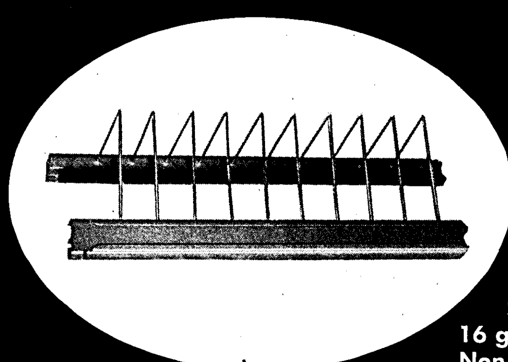
REEL LIBRARY UNITS GREATEST NUMBER OF REELS PER SQUARE FOOT



DESIGNED TO MAKE THE MOST
OF YOUR STORAGE AREA!!!



GREATEST FLEXIBILITY
2½ - 3 - 4 ft. sections
to accommodate any room size



STURDY CONSTRUCTION
16 ga. bars for NO-SAG-SUPPORT
Non-Weld 11 ga. Wire Separators

*ALL Systems Metal Products are unconditionally guaranteed for FIVE YEARS, under normal usage.
Your SYSTEMS Representative can show you the difference in full detail.

DIV. SYSTEMS MANUFACTURING CORPORATION

SYSTEMS SALES COMPANY

EXECUTIVE OFFICES and PLANT: 13 BROAD ST., BINGHAMTON, N.Y.

— *Finest Quality Data Processing Accessories Since 1945* —

CIRCLE 12 ON READER CARD

BUSINESS & SCIENCE

AIR FORCE ORDERS 30 301s LESS COBOL

RCA's largest domestic computer order to date is a government contract for the installation of 30 301 systems without COBOL. The hardware will replace more than 550 punch card systems at 11 air materiel centers of the Air Force Logistical Command (AFLC). "The AFLC is a 100 per cent supporter of the COBOL effort," says Jack Jones of the Command at Wright-Patterson AFB, Dayton, Ohio. Within one year, he adds, expansion of the 10K-memory systems is anticipated; COBOL will be utilized on the expanded hardware as applications grow and as conversion of systems is completed. "One benefit of COBOL is that it's forcing people to learn to write compilers," Jones said.

Current production rate of the 301 is one per day; as of the end of March, 233 had been installed, and 237 orders are in house.

The forthcoming 401, which will not be announced before the middle of the year, will have microferrite scratchpad memories with a cycle time of 300 nanoseconds. Main memory will be an adaptation from the 601. A gp computer with provisions for communications applications, it will have high speed circuitry from the company's military computers.

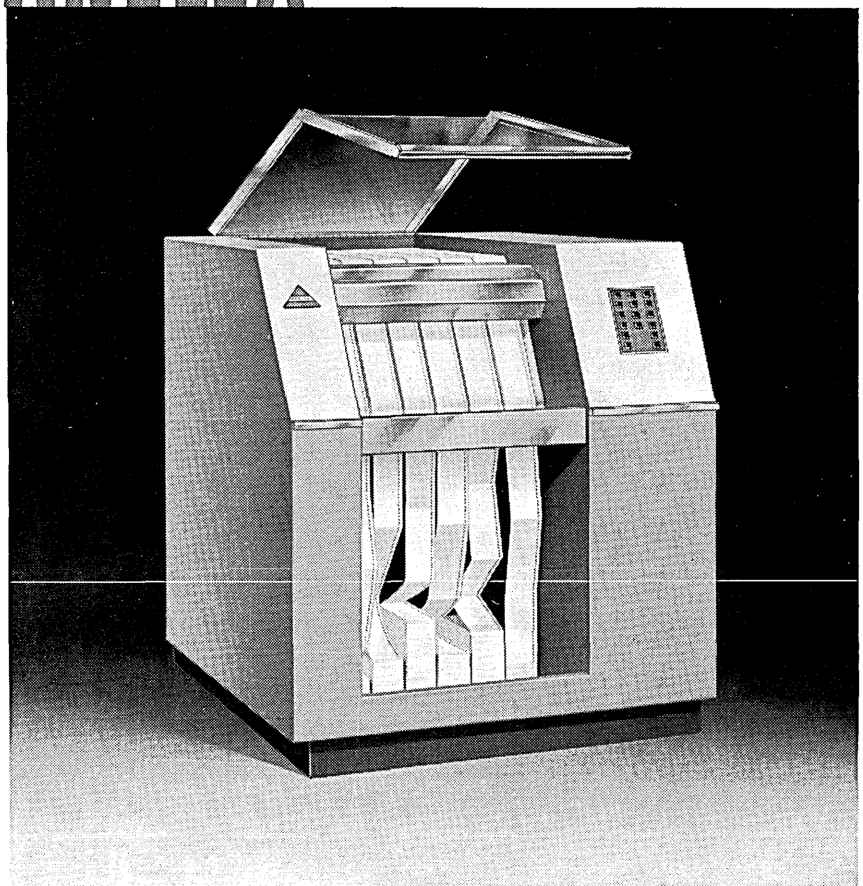
GOVERNMENT MOVES TOWARD PURCHASING COMPUTERS

Computer manufacturers, for some time, have been attempting to sell more equipment, rather than awaiting returns on their investment in the form of monthly rental payments. Indeed, a plethora of annual reports decry the drop in profits on an increased dollar-volume of business. Reversing this trend has become all-important, and where better to begin pushing sales than with the largest single customer: the federal government?

Following a hearing of the House subcommittee on Census and Government Statistics on June 5, 1959, the General Accounting Office and BuBudget were requested to make a lease vs. purchase study of hardware by federal agencies. A resulting BuBudget circular of October 1961 recommended purchasing if rental charges over a six-year period exceed purchase and maintenance costs.

Unfortunately, implementation of such a policy has been impeded by two factors: (1) Lack of a centralized overseer. Each agency decides more-or-less on its own what, when, and how to acquire equipment, guided by its budget restrictions. (2) Budget-review methods. As presently practiced, agencies are discouraged from

NEW FROM ANELEX



MULTIPLE TAPE LISTER SYSTEM

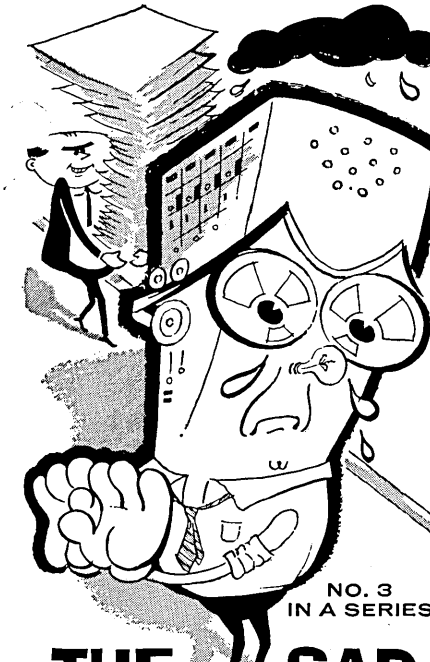
For proving and clearing operations in demand deposit accounting in banks. Completely integrated system . . . six 24-column high speed listers, control electronics and buffers in a single, soundproofed housing. Compatible with almost any type of MICR Sorter-Reader. Prints a master list, four transit lists and one miscellaneous items list in a single pass through the Sorter-Reader. Prints at rates up to 2000 lines per minute.

Further information is contained in the new 4-page Bulletin, "Anelex Multiple Tape Lister Systems", available on request.

ANELEX CORPORATION
SYMBOL OF QUALITY

ANELEX®
155 CAUSEWAY STREET, BOSTON 14, MASSACHUSETTS

CIRCLE 15 ON READER CARD



NO. 3
IN A SERIES

THE SAD COMPUTER

I am a sad computer. I suffer from a panacea complex. People think I'm the sovereign remedy for all business ills. I like to think of myself as the key to automation, but in all honesty, my success depends on the effectiveness of the consultants, systems analysts, programmers and data processing specialists.

The McDonnell Automation Center has been doing a lot of work educating people in the ways of computers. From the Automation Center a businessman can find out if and where automation is justified, and if it is, the type of program to follow. They don't have any computers to sell; just guidance and assistance for virtually any automation task. Automation Center specialists can help you design, install and program a complete system of your own, or they can institute an effective automation program utilizing their facilities at low by-the-hour rates.

Here comes my boss again, both hands stuffed with problems. He'll fume with irascibility when he finds out I can't solve them after he paid so much for me. I wish he'd write or call the McDonnell Automation Center.

MCDONNELL AUTOMATION CENTER
Division of MCDONNELL AIRCRAFT
St. Louis, Missouri
Phone: PErshing 1-2121

CIRCLE 38 ON READER CARD

purchasing expensive hardware by the more careful scrutiny which such expenditures receive.

As a result, two separate actions were taken recently to establish a sole administrator of the procurement and utilization of all computers by the government. An office with this authority, directly under the President, has been recommended by the Comptroller General of the U.S. In a report to Congress, "Study of Financial Advantages of Purchasing Over Leasing of EDP Equipment in the Federal Government," dated March 1963, it is recommended that the government pursue the policy of purchasing equipment through a revolving fund arrangement.

According to the authors, "The detailed cost comparisons of 16 different electronic machine models, which constituted the principal part of our study, indicate potential savings of about $148 million over a five-year period. These significant possible savings apply to only 523 of approximately 1,000 electronic data processing systems installed or planned for installation on a lease basis by June 30, 1963. For additional use of the 523 machines after five years, there would be further savings at the rate of over $100 million annually." Fears of obsolescence of hardware are diminished, the report states, because of the ability of this office to transfer equipment among agencies as needs vary.

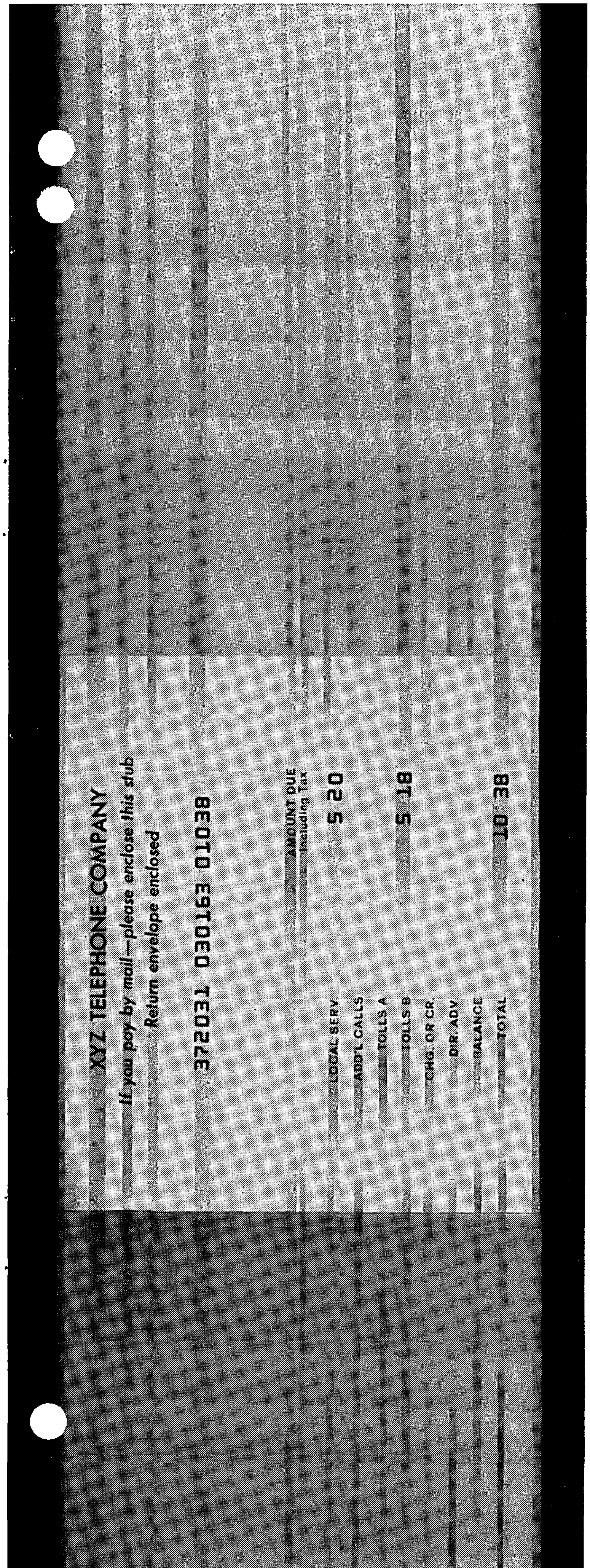
Utilizing the results of this study, Rep. Jack Brooks, Texas, introduced a bill which would empower the administrator of the General Services Administration with the same authority. In both arrangements, the decision of purchasing vs. leasing as well as the logistics would be centralized outside the using agency.

FANATICS, IRRESPONSIBLES, AND ASSORTED I.R. DREAMERS

Current prognosticators on the role of computers in communications and information retrieval were branded as "fanatics" by Dr. John R. Pierce of Bell Telephone Labs, Murray Hill, N.J. Speaking at dedication ceremonies of Crerar Library, Illinois Institute of Technology, Chicago, Dr. Pierce said, "I believe that someday, somehow, things will be done which resemble in a superficial degree, if not deeply, the things that fanatics have promised and badgered us about, but have failed to produce." Among undelivered goods, he says, is a computer which understands spoken English and the relations between words, and translates languages with the facility and econony of humans.

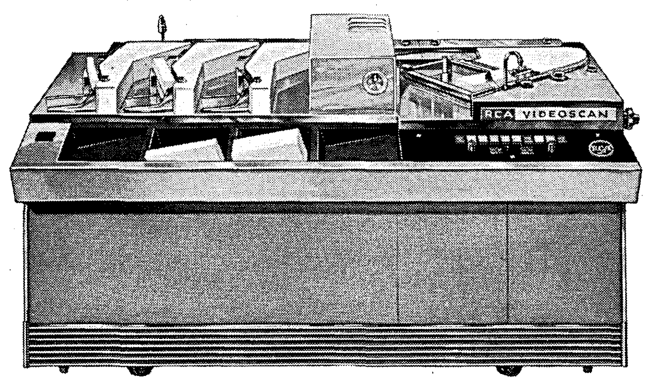
"If I have painted a rather gloomy picture of the relations of computers to libraries," he said, "I have done so deliberately to counteract the sadly fanciful picture that has been painted by some irresponsible people." He acknowledged hardware utilization for preparing abstracts and indices, and predicted their role in data and facsimile transmission in libraries.

"There've been lots of grandiose words about the role of computers in libraries and the information retrieval area, but there is no step-by-step approach, only magnificent plans. Lacking are viable inventions, things that will work with present equipment...The real, workable thing is missing."



NOW...WHIP 'EM THROUGH... 90 THOUSAND DOCUMENTS PER HOUR!

RCA's new Videoscan couples TV with EDP for ultra-reliable optical character reading



A new development in electronic data processing . . . devoted to the concept that "time is money" . . . has been born of RCA's far-reaching electronic and television research.

The name is Videoscan. The speed it provides is three times that of conventional electromechanical scanning methods. The benefits it brings include: reading smudged, creased, or marred documents without interruption; processing turnaround documents of varying sizes and thicknesses without a falter.

Videoscan marries television recognition techniques with electronic computation. It thus provides unprecedented speed, reliability and versatility . . . complementing the efficient RCA 301 computer system. For documentary evidence, please call . . .

RCA ELECTRONIC DATA PROCESSING, CHERRY HILL, N.J.

RCA®

The Most Trusted Name in Electronics

CIRCLE 50 ON READER CARD

EDITOR'S READOUT

A NEW EDITOR FOR DATAMATION

With this issue, Robert B. Forest assumes the position of Editor of DATAMATION, replacing Harold Bergstein, who has served in that capacity for the past two years, and earlier as Managing Editor.

Bob Forest brings to his new position a varied background in information processing as technical writer, editor, public relations executive and free-lance writer for the past seven years. He holds an M.A. in English from the University of Chicago.

Hal Bergstein, who has served this magazine and the industry so well, is resigning to accept a public relations post. To him go our best wishes for success.

We know that the industry as a whole will continue to give our new editor the cooperation which has enabled DATAMATION to maintain its vigorous, timely and comprehensive coverage of the important and dynamic information processing community.

—GARDNER F. LANDON
Publisher

a user's view

CORRECTING OBFUSCATIONS BY ORDAINED LINGUISTS*

by STANLEY M. NAFTALY, Integrated Data Processing, Space Technology Laboratories, El Segundo, Calif.

One of the newest parlor games seems to consist of sitting around in a circle (*inner*, of course) and debunking COBOL. I have heard everything blamed on this language but the recent recession in Monaco.

My intent is to break the spell of the "I've never really used it but I have a feeling it won't work" syndrome, by evaluating the language realistically and attempting to place the COBOL issue in proper perspective.

My facts and figures derive from experience at Space Technology Labs in using COBOL as a business data processing language over the past two years. As a measure of use, we have written, compiled, debugged and installed over four hundred production programs in COBOL in this period. In addition, I have taught COBOL on the job to experienced and novice programmers, to supervisors of customer areas within the company, and to management. Also, for three years I have conducted a UCLA course in COBOL. Among the students have been programmers, supervisors, compiler writers, analysts, laymen, a lawyer, and undergraduates. I have just finished a three week Saturday morning series with high school students.

From these combined experiences, I believe that COBOL can be a valuable, moneymaking tool in a production data processing shop, as well as a provocative subject for symposia.

I do not believe that COBOL is the ultimate language. I do believe that it is an evolutionary step in the development of commercial languages.

Granted, COBOL has shortcomings which affect its efficiency. Some of these deficiencies are inherent in the design of the language, but as I will point out, many of them are more closely associated with the compiler or the programmer.

I have made a compilation of criticisms of COBOL, which, while by no means complete, does cover the major points.

partial hits and near misses

The statement is made that the English Narrative form of COBOL makes fast, efficient compilers impossible. This is in part true. A free-form language using variable length words and entries is, of necessity, more time consuming to compile. The patent remedy for this complaint is a symbolic language. Certainly this would do the job. The data division of COBOL approaches the symbolic form. However, I feel that another format is still better: the fixed form narrative. Let us keep English words in order to preserve the ease with which the language may be taught to non-technical people and to avoid reducing its usefulness as a communication tool. There would be little difference in compiling speed between a symbolic language and a fixed-form language without noise words. Compiling time is consumed in variable scans locating operator words, not in deciphering them once found.

While the narrative form can and does have an effect on compiling times, it need not affect the efficiency of the object program.

The cry that COBOL is too verbose possesses some validity. Very little readability or communication would be lost by eliminating some of the verbal frosting and this would improve compile times.

Compilers could be much simpler if some of the compound procedure options were eliminated. PERFORM—VARYING—UNTIL, could be accomplished by writing PERFORM, ADD, IF. The object program, in most cases, would be virtually the same.

The Maintenance Committee is presently firming the specifications for a subset of COBOL-61 to be called COMPACT. This language is intended for use on small computers or to provide faster compilations on large computers. Toward these ends, many things have been done to reduce both the verbosity of the language and the number of options available to the programmer. Examples of these changes are: elimination of optional words, reduction in maximum size of data names to ten characters, and elimination of some compound procedure statements.

It is also asserted that COBOL forces the programmer to write too much. This is a corollary of the verbosity accusation. A data division entry using only LEVEL NUMBER, DATA NAME, PICTURE, and, if appropriate, USAGE and SYNCHRONIZED, would save a great deal of writing. This suggestion applies also to the too-many-options argument. The same thing holds true, although to a lesser extent, in the procedure division.

It is true, and regrettable, that source language debugging is impractical for the COBOL user. Compilation times must be improved and techniques implemented for compiling and testing in one pass.

COBOL is also condemned for being a programmer's language. Right again. Bad programmers write bad programs, in any language. But this facet of the subject has been overemphasized. Years of experience are not needed to write good COBOL programs. An understanding of programming concepts, (update, sub-routine, end of file, Tape Labels, segmentation, etc.), is needed. This knowledge can be at a more general level than with assembly systems, but it must be there. While there is also validity in the corollary complaint that a COBOL programmer must know a lot about the machine for which he is writing, this too has been overstated. Good COBOL manuals point out many of the factors which affect object program efficiency. Other such factors are the responsibility of the librarian. More will be said of the library's function later.

*Based in part on material for a COBOL textbook authored by Mr. Naftaly, to be published by Prentice Hall in 1964.

COBOL publicity is often attacked as laughable. It often is laughable, having about the same percentage of validity as some of the allegations leveled against the language itself. Incidentally, what has this situation to do with the merits of COBOL as a programming tool?

Adoption of COBOL by DOD (whether explicit or implicit) has tended to reduce research in new techniques by computer manufacturers. In support of previous statements on COBOL's evolutionary nature it is obvious that DOD never intended COBOL as the "ultimate language." Look at the organization of CODASYL. The express purpose of the Development Committee was, and is, "to consider the next generation of computer languages. . . ."[1] Certainly the economics of the computer business dictate that if manufacturer X must produce four COBOL compilers, he is probably not going to undertake any other major commercial language efforts. Again, COBOL is an evolutionary language. No one expects it to be the only language around ten years from now.

Let us consider some weaknesses that come to light in the use of the language.

In its present form and as currently used (or misused) COBOL has a tendency to lull the programmer into becoming sloppy. There are too many ways to save a little pencil lead by getting fancy. For example, there is a tendency to PERFORM segments of the program which are comprised of only a few lines of coding. This, of course, produces larger, slower object programs and, in the case of this example, increases chances of coming up with a very messy debugging problem. Here the blame does not rest on the language, but on the way in which it is used. It is possible to write very bad COBOL programs that work. Fortunately, it is also possible and practical to write very good COBOL programs. The range of efficiency of COBOL object programs is as large as that for assembly systems or symbolic languages.

At present certain necessary functions are not provided for in COBOL. Provision is made for the description of tables of data; however, there is no search verb. In addition, there is no efficient manner of setting and retaining table indices. The DATA NAME SUBSCRIPT option is fine in theory, but very slow in practice.

This is currently an implicit, tape oriented language. There is no provision for the use of mass storage or random access devices. The COBOL Maintenance Committee is currently considering extensions to the language in the areas of internal tables and mass storage.

Most of the above criticisms are at least partially valid. Usually, they are voiced in a manner which leads the reader to believe that no one in his right mind could ever accept the idea of COBOL, let alone use it.

Even within the acknowledged limitations of the language, compile times can be greatly improved. To state it another way, the severity of COBOL's limitations depends to a great extent on the design of the compiler. As an example, a program compiled on two machines of roughly equivalent speed took 25 minutes to compile on one, and an hour and 45 minutes on the other, although the second machine had tapes three times as fast as the first. Admittedly, this is an extreme example, but it illustrates the point. One of the newest COBOL-61 compilers is attaining average compiling speeds of two minutes, and big programs are going through in three minutes. Another processor is compiling programs at the top speed of the card reader, 800 cards per minute. Here again the point is that much depends on the design of the compiler, and, of course, on the power of the computer. Most of the faster compilers are not cascading through an assembly system on the way from COBOL to machine language. When manufacturers simply tack preprocessors onto the front of existing assemblers, it is rather senseless for the user to compare COBOL compile times with assembly times. This is not to blame the compiler or the manufacturer for all problems. Nor is it to minimize the importance of these problems to the user. Some thought should be given, though, to the sources of these problems to determine how many of them are really attributable to the language, how many to the compiler, and how many to the user.

not even close

A remarkable number of criticisms of COBOL have little validity. One of the most prevalent is that the language is not and cannot be common. If commonness means the ability to take a source program deck prepared for one computer and drop it blissfully into the card reader of another computer, the point is valid. It is regrettable that much of the propaganda on the subject has intimated, if not blatantly asserted that this is the case. If commonality is measured by the much more realistic yardstick of savings in conversion from one machine to another, COBOL is very compatible. Tests show that converting a COBOL source program from one computer to another involves only 10 to 40% of the time required to write the original program in COBOL. It should be stressed that this time is a percentage of actual "coding" time, and not the total time required to analyze, design and code. If the two machines are very similar in design and have the same collating sequence, the original coding time might be reduced by 95%. The 40% figure assumes that the original machine is of variable word length design, has only one internal mode (e.g. BCD) and that the new machine is a fixed word type with more than one internal mode. In a case of this type the Data Division would have to be screened very carefully for proper insertion of USAGE and SYNCHRONIZED clauses. In any case, the file conversion would require far more effort than would the change over of programs.

A saving of from 60 to 90 or even 95% of the effort required to reprogram illustrates very clearly that while COBOL may not be "magically" compatible, it certainly meets the realistic test – it saves the user money.

We occasionally hear that commonality is not good for us, anyway. We are counseled that great gains can be made by reprogramming everything from time to time. While this statement may be valid enough in theory, when approached from a realistic point of view, it is reduced to absurdity. Most commercial data processing organizations are constantly fighting to keep up with new work and revisions. Incidentally, because of the changes that are an inherent part of business data processing most programs are rewritten, at least partially, during their life span. Besides, the elimination of a major portion of the conversion reprogramming effort will allow more time for housecleaning under less pressure. I suspect that the Blue Sky of the Riviera might be good for a lot of us, but can we afford it?

The next claim usually heard is that even if a program runs about three years before being scrapped, COBOL doesn't save us any money. Any program that runs for three years is bound to have been changed several times along the way, probably *after* the original programmer has gone. The time saved on these changes, the ability to make them without spending weeks trying to figure out what's going on in the innermost bowels of the program, can save more than the program originally cost.

The efficiency of COBOL object programs very often

[1] Solomon J. Pollack, "CODASYL, COBOL and DETAB-X," *Datamation*, February, 1963, p. 60.

is criticized. Inefficient object programs are often given as an obvious and sufficient cause for pulling the language off the market. Here again, is a case of a deficiency of the compiler mistaken for a deficiency of the language. One of the most succinct statements on this subject is, "People won't use COBOL because the compilers are bad".[2] There is much confusion between language deficiencies and compiler problems. The area of object program efficiency is one of the most fecund areas for this confusion. Some compilers are extremely slow and others produce grossly inefficient object programs. Indeed, one or two are guilty of both sins. There are, however, several extremely efficient compilers on the market and more coming all the time. For a change the industry seems to have profited by its early mistakes. COBOL contains within its structure all specifications necessary to produce highly efficient object programs, both in terms of size and speed. Many compilers do not exploit this potential.. When editing is called for, the insertion of a generalized macro or subroutine must turn out oversized and slow programs. The preponderance of blame for inefficient object programs lies with the compiler. Part of the remaining guilt belongs to the user. A knowledge of the "compiler's ways" is invaluable to the user and can account for differences of as much as 40% in running time, and even more in object program size.

This is not to suggest that every programmer should have this depth of knowledge. It is to suggest that programming standards are very important in an installation using COBOL. Manufacturers' manuals must be precise, informative, and cover the points of relative efficiency. One manufacturer issues a manual that shows the number of machine instructions generated for every verb option in the Procedure Division. These figures are shown for different data descriptions and for the various options in the Environment Division. We have consistently obtained programs that are 80-85% as efficient in terms of speed as machine language. At any rate, they are 80-85% as efficient as we have been able to write in machine language with "good average" programmers. In terms of size of object program, the COBOL programs average 10-15% larger than their machine language counterparts. The user shouldn't be too interested in comparing COBOL object programs with the hand coded work of expert programmers, unless he happens to have a shop full of these experts, and unless they can write awfully fast in machine or assembly languages. In the absence of either of these conditions, the expert programmer is not going to turn out enough runs to make much of a dent in the workload of a shop big enough to support him. Moreover, he is going to be constantly making changes in his programs. Lack of efficiency in object programs is certainly a problem to the user if it exists. It needn't.

Before an object program can be evaluated as to its efficiency it must be debugged. It is surprising to hear that COBOL extends debugging time and makes checking out a program more difficult. We have found that debugging averages approximately 20% of our experience with assembly systems. One of the major reasons for this saving is the documentation produced by a compilation. Again, this differs from compiler to compiler. The user who gets, as we do, "extras" like Data Division cross references showing every use in the Procedure Division of all items in the Data Division, and a transfer trace of the Procedure Division, undoubtedly pays for these extras in compiler time. When it comes time to debug a program, however, these extras more than pay for themselves. The mere presence of this documentation does not, of course, ensure ease of checkout. You cannot legislate good programming. Effectiveness of the compilation listings depends on the cooperation of the programmer in making them meaningful.[3]

The diagnostics performed at compile time are another major factor contributing to COBOL's debugging ease. DOD specifications for COBOL compilers stipulate only a few types of diagnostics that must be performed. These tests are largely up to the implementor. Certainly, too many or too extensive diagnostics can cost time during compilation, but, held to a realistic quantity, they save time during debugging. In the compilers that go through an assembly system, diagnostics should be designed so that rarely does an error appear during the assembly phase. That is, errors should be caught during the COBOL phase of the compilation.

Some of the additional debugging benefits present in COBOL are not unique to this language, but are present to some extent in any library-type assembly system. Among these benefits are standard I-O processing subroutines with standard linkages, routines for check point and rerun, file labeling, etc. The presence of these routines, "canned" or generator-type, means that the user need not concern himself with control of batched, variable record files, end of reel – end of file tests, or with the linkage for these functions.

An item of major importance in the area of compiler generated routines is that of SECTIONING. The ability to overlay program segments at object time can mean the difference between an efficient system and one which is little more than a series of electronic tab runs. This is true particularly when dealing with a computer of limited memory capacity. The SECTION option in COBOL assures not only the ability to segment, but the ability to go from one program section to another without special entrance and exit points.

There are ways in which the COBOL programmer can produce an impressive debugging problem. The use of nested PERFORM statements can, if carried to extremes, generate a logic labyrinth of major scope. The same is true of programs with 30 or more overlapped ALTER statements. These abuses reflect neither a language nor a compiler problem, but a programmer problem.

There are techniques that programmers can use to aid in debugging. Punching from a procedure flow chart is one of these. Rather than drawing a detailed flow chart and coding from it, the programmer prepares a COBOL procedure chart. This type of chart takes approximately 15% longer to draw than does the standard detail flow chart, but there are several benefits to be derived from this technique. The overall time spent in charting and coding is somewhat reduced. More important, this practice eliminates the non-creative, clerical process of translating a detail chart to a coding form. The incidence of clerical errors by programmers is highest in the non-creative phases of problem solution. Elimination of this step does away with most of these errors.

The use of this technique requires that certain keypunch personnel be given training in COBOL notation. Familiarity with general statement format allows the keypuncher to supply necessary punctuation and, surprisingly, to spot many clerical errors.

Further, during debugging, a COBOL chart helps provide the visibility necessary to further speed check-out.

Debugging COBOL programs, while not yet econom-

[2]Harry Cantrell, Business Data Processing Hall of Discussion, ACM National Conference, Syracuse New York, September 6, 1962.

[3] Lawrence D. Provisor, "Some Aspects of Automatic Programming as Viewed by Different Groups", (Unpublished Master's thesis, Graduate School of Business, University of Southern California, 1963).

ical in the source language, is far easier and faster than debugging in assembly systems. Problems encountered in run check-out are more often logic problems than clerical problems.

The contention that COBOL is not a good communications language should be answered. COBOL can be taught readily, on a communications rather than programming level, to customer areas within a company. Twenty hours are usually more than enough for this instruction, including the time necessary for a very short course in computer fundamentals. The customer, usually at the supervisory level, can then work closely and profitably with data processing personnel assigned to his area. The structure of the Data Division is such that file and report formats can be easily understood by the non-technical person, affording him the ability to concisely and precisely describe data relationships. An understanding of existing programs sometimes becomes necessary for the customer and can be conveyed easily at a general level.

COBOL is an excellent communications language, provided the proper training is administered. It must be remembered that COBOL is a language for *business* data processing not scientific or engineering computation. By their nature, business problems do not lend themselves to as concise a description as do scientific problems. In describing business problems to the non-technical person the English Notation and Narrative form of the language are invaluable.

We are also told that COBOL is a terrible language for open shop use. The idea of an open shop for business data processing seems ill-advised. The typical business program is one that runs repetitively for a period of time, and is then revised. Certainly an open shop approach to this situation would be inefficient. For the many one-time jobs that must be run, Load and Go packages such as extracts and report generators are used, not individually written programs. Saying COBOL is not good for open shop operation is somewhat like asserting that it's wasteful to simulate a tab on a 7090.

COBOL detractors insist that coding is a small part of the actual cost of programming. This is true, no matter what system is used for programming. However, it does not follow that it is not worthwhile to trim coding costs. COBOL is not automatic programming (nothing available today is) but automatic coding. As such, it can cut the cost of coding by 50% and can significantly reduce the costs of problem definition, analysis, and systems implementation. Savings of this magnitude are certainly not to be ignored.

Another brickbat hurled at COBOL is that the equipment manufacturer raises the rental prices of his computers to cover the cost of developing COBOL compilers; therefore, the customer pays for the privilege of using the language. Far from refuting this allegation, I will ask one question. When has any software worthy of the name been free?

teaching COBOL

Many statements have been made concerning the relative ease of teaching and of learning COBOL. It has been alleged that FORTRAN is far easier to learn. This may very well be the case if a student knows mathematics, because scientific notation is far more concise than English. However, COBOL is a great deal easier to teach to the non-technical person. More and more organizations are making use of the systems analyst as a COBOL programmer. More and more people in the customer areas of a company (materiel, accounting, communications, etc.) are finding it necessary to learn the language of data processing. Finally, more and more Business Administration students are taking courses in data processing as part of their curriculum. COBOL is an excellent medium for teaching all of these people. Organization of the language, narrative form, and English Notation all contribute to the ease and speed with which COBOL can be taught.

The group of high school students mentioned earlier picked up the intent of COBOL quite readily. The language was used to teach them computer fundamentals as well as how to solve simple business data processing problems.

The problems encountered in teaching COBOL as a programming language are not problems of mechanics, but problems occasioned by lack of understanding, of the intent of the language. The "hardened bit programmer" is outraged by his lack of direct control of the machine. The scientific programmer does not always understand the emphasis on file control or the concept of a three-way-IF statement. On the other hand, the experienced commercial programmer, particularly if he has used an assembly system, almost teaches himself. The analyst finds the language very satisfactory for his work and learns it easily. The interested layman, once he is taught something of computer fundamentals, learns COBOL most readily.

Teaching COBOL as a programming language is not necessarily faster than teaching other languages. The savings arise from the ability of the student to become productive more quickly The learning curve observed may be expressed in terms of the increase in efficiency of the object program rather than the historical "how long does it take to get it to work?"

use and abuse

We have all heard of installations which have tested COBOL and found it wanting. While many tests are valid and yield credible results, too many are questionable and yield near-worthless results. These latter tests usually are of two types. The first involves rewriting an existing program in COBOL and then comparing object programs as to speed and size. The fallacy here is that the original program was written in assembly system A and the detail flowchart from that program was used as a basis for the COBOL version. That is, the original flow chart was simply transliterated into COBOL statements with no regard for unsuitability to COBOL of the run design. Efficient COBOL object programs require that analysis and design of the program be made with the language in mind. Valid results can hardly be expected from this kind of test.

The second fallacious test is one that is a little more difficult to identify, Two programmers of equal ability and experience are given a problem definition. Each is to design and write the program involved, one in COBOL and one in an assembly system. Usually there is a fairly wide disparity in the efficiency ratings of the two programs produced. Upon close examination it is found that though both men have about the same amount of experience, that experience is in the assembly system involved. The COBOL programmer had to write his program in a language which he had just learned. If valid results are desired, the user must be very careful in designing comparative tests.

In order to gain the maximum benefit from COBOL, certain functions should be provided for by the user. The establishment of firm programming standards is essential. Relative efficiency of source and object programs, degree of debugging assistance provided by documentation, and the amount of compatibility provided for the future all depend in large part upon the quality of these programming standards.

One aspect of COBOL programming standards deserves particular mention. This is the use of the library. Howard Bromberg[4] has covered this point very well, but it bears repeating here.

The COBOL library may contain entries to the Environment, Data, and Procedure Divisions. Use of the library can help eliminate a great deal of writing for the programmer. Standard entries such as the Source Computer Section of the Environment Division can be called in toto from the library. Proper use of the library can assure good documentation by providing standard File and Record Descriptions with meaningful data names. These standard data descriptions also ease the modification of programs by improving the communications involved. Cleaner and more efficient Procedure Divisions can be assured by using library entries containing sub-routines in wide use. In evaluating the usefulness of the library it must be remembered that the relative efficiency of the object program is determined to a large extent by the Data Division entries.

One of the most experienced programmers should be responsible for the library. This man must arm himself with an extensive knowledge of the hardware being used and of the compiler itself. In addition, he should have as much general knowledge as possible of applications. In most cases an installation will have a relatively small number of files and by careful analysis of the uses of these files, an efficient library can be initiated and maintained.

an organizational tool

The concept of COBOL can be used by the data processing manager as an organizational tool. Use of experienced analysts as programmers is encouraged by the format of the language and its notation. This approach eliminates the communications problem inherent in passing a program from the analyst to the programmer and permits one man to nurture a job from inception, through analysis of existing procedures, design of the proposed system, selling of the system to the customer (who works closely with the programmer-analyst), flow charting, coding, compilation, debugging, and production documentation.

This type of organization requires careful selection of personnel, but enables the user to select analysts with broad experience in administrative systems design rather than machine language programmers. The use of COBOL makes it possible for these people to become productive programmers quickly, and eventually very efficient programmers.

A computer-programming expert becomes essential in a department of this type. This man is the watchdog of efficiency and, until the analyst-programmer gains experience, his debugging mentor. In this arrangement the need for complete programming standards becomes apparent. Since the analyst is used in order to take advantage of his knowledge of applications, the importance of the library as a built-in efficiency booster again becomes obvious.

summary and conclusions

Looking back on the criticisms examined, we see that they are almost all concerned with time. The question is, what sort of time? In this context, time may be considered in four classes.

Programming time can be significantly reduced by the realistic use of COBOL.

Compiling time is affected to an extent by the form and structure of the language. However, experience is showing that vast improvements can be made within the scope of these limitations if compilers continue to be improved.

Object program time is wholly dependent on the compiler and the programmer.

The fourth time class is the one with which management is most concerned, response time. The ability to implement programs rapidly, and to respond to the demands of management quickly and economically is, after all, the most important test of a successful Data Processing Department. Here again, viewed realistically and implemented properly, COBOL has a great deal to offer the user.

COBOL is a step in the evolution of programming languages. It represents a workable and realistic approach to compatibility. It serves well as a communications language and is easy to learn.

Much of the criticism leveled at COBOL arises out of an "if its not perfect, I don't want it" attitude. The business data processing user must not confuse philosophy with reality. ■

[4] Howard Bromberg, "Compilers—Where They are Today," *Business Automation*, December 1962, pp. 25, 32.

people IN DATAMATION

■ Walter L. Anderson, executive VP, General Kinetics Inc., Arlington, Va., has been reelected chairman, Professional Technical Group on Electronic Computers, IEEE. Largest of the institute's 30 technical groups, the group is responsible for publication of technical material relating to computers.

■ Harold Bergstein has joined Computer Sciences Corp. as Director, Public Relations and Advertising. He was formerly editor of DATAMATION.

■ Walter P. Czeropski Jr., has been appointed program manager of Motorola's Random Access Discrete Address program. He was formerly chief engineer, Communications Laboratory, Motorola Military Electronic Div.

■ Dr. Stanley Gill has been named to the newly-created, part-time chair as professor of computation, Manchester University's College of Science and Technology, England. He retains his position as head of the Advanced Applications Group, Ferranti Ltd. He will assist users of the university's Atlas which will be available to the college through a data link system.

■ Five additions to the programming staff of Systems Programming Corp., Santa Ana, Calif., are Tom Morris and George Connell from Nortronics, Andre Voskressensky from Douglas, Charles Milton from Litton, and Alvano Dean from Bendix.

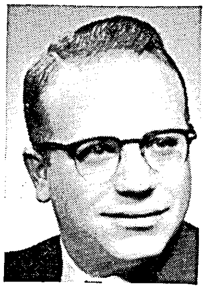
■ John Fedako, formerly with Univac, has joined National Computer Analysts Inc., Princeton, N.J. as project engineer.

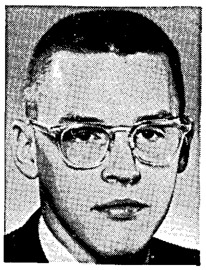
■ Dr. Hans Zassenhaus, professor of mathematics, has been named director of the Univ. of Notre Dame's three megabuck computing center. He joined the faculty in 1959, and has been conducting basic research in the geometry of numbers. Named assistant director for scientific affairs was Dr. Louis Pierce, assoc. professor of chemistry.

PROGRAMMING ON-LINE SYSTEMS

part one: a survey of problem characteristics

by W. L. FRANK, W. H. GARDNER, and G. L. STOCK,
Informatics, Inc., Culver City, Calif.





On-line data processing systems have recently become an important area of interest to digital computer applications specialists. The development of digital transmission systems and the availability of faster bulk data storage devices have stimulated this new kind of data processing concept in which information is entered into the data processor *as it is generated*, and outputs are requested from the computer *as they are required*, and, in fact, limited to that information needed at the moment.

In the context of this discussion, "real time" refers to information handling having the above characteristics and with the additional requirement that certain functional operations meet well-specified timing limitations.

Two important types of systems are discussed. The first is one in which the response times are measured in milliseconds. Such systems are automatic and many are closed loop, since the timing requirements preclude the intervention of man. Examples of such systems are process control applications or radar tracking and recording systems.

A second important area is a computer system to which are connected a number of interrogation and display devices – establishing a man/machine relationship. Here not only the hardware characteristics are important, but also the communication devices and the communication language by which man interacts with the system. Examples in this area are military command and control systems and commercial reservation systems.

After citing the problem characteristics which typify these application areas, the programming implications are discussed. Since timing and control are the most critical aspects of this problem, detailed consideration is given to hardware features which bear on the efficacy of the programming system.

application areas

Until recently the major interest in real time systems has been limited to three principal areas: a) military command and control applications [1, 2, 3], emphasizing the man/machine interaction, b) commercial process control applications [4] where computers have been integrated into a closed loop system, and c) guidance and control applications [5, 6] aboard missiles and satellites.

During the past year interest in on-line systems has accelerated rapidly and present application areas touch many government and business facets including:

(1) Space flight and tracking [7]
(2) Simulation [8]
(3) Management Analysis and Control [9, 10]
(4) Airline reservations [11, 12, 13, 14]
(5) Banking systems [15, 16]
(6) Automatic checkout [17, 18]
(7) Merchandising [19]
(8) Teaching [20, 21]
(9) Hospital Automation [22, 23]
(10) Information Retrieval (Medical, Law, etc.) [24]

Interestingly enough, the on-line concept is also becoming important in the general scientific and engineering area. Whereas these original users of data processing equipment influenced the separation of programmers and users from the actual equipment, a current trend is developing which brings them together again. Two important reasons are given. The first concerns the question of better service to the engineer or analyst. By providing remotely located input/output stations connected to a large data processor, it is expected that turnaround time for many programs will be decreased. Users may also view a time shared, central processor as their own [25, 26]. The second reason is that the on-line system will serve in the experimental laboratory as a practical tool in the solving of complicated analytical problems and in the real time conducting and evaluating of experiments. The investigator instructs the machine to perform various computations based upon the analyst's insight. Based upon the computed results presented to him in graphic or numerical form he either continues along his current line of thought or else tries an alternate approach by selecting different problem parameters [22, 27].

problem characteristics

There are various aspects which distinguish on-line programming applications from other types of computer applications. The more important characteristics—those which exercise greater influence in system design and programming—are as follows:

Real Time

The most obvious and most important characteristic of an on-line system is that it must operate in real time. The system must respond to external events within times which are consistent with the occurrences of the events themselves and with respect to the desired behavior. This implies three time aspects: a) the interval between events, b) response time—the actual elapsed time between an event and the completion of all the processing triggered by the event, c) frame or cycle time—a programming convenience which is some integral multiple of the interval between events. In computers controlling rapid, continuous processes (e.g., missile control systems), the concept of response time must be thought of in terms of the "frequency response" of servo theory (i.e., amplitude variation and phase lag of dependent variables vs frequency of independent variables). Since digital computers, by definition, are sampled-data devices, the sampling rate, and implicitly the response time, must be a function of the frequency of the sampled data. The subject of sampling theory and servo mechanisms is, however, beyond the scope of this paper.

All of these aspects must be considered in designing and programming an on-line system. At one extreme the interval between events depends upon the response time and can be controlled to some extent by the program as for example in the case of a reservation system. At the other extreme the interval between events is determined by a clock or comparable device, as in the case of radar data sampling, and the response must be programmed so as not to exceed certain well defined system specifications; otherwise there is a loss of data.

Interval between events — A spectrum of typical on-line systems showing the intervals between events for selected applications is illustrated in Figure 1. The examples cited reflect applications with which the authors are familiar, and the times shown are application and not computer dependent. The requirements establishing these values were as follows:

a) Sample up to 20 channels of radar read-outs at a maximum rate of 200 times per second for each channel.
b) Generate the instantaneous impact point for range safety every 50 ms.
c) Accept a processing request from ticket agents in an airline reservation system at 10 per second maximum.
d) Accept on-line typewriter input at a typing rate of 60 words per minute maximum.
e) Accept an input function request (such as request for a prestored display) from an on-line man/machine communication device every five seconds.
f) Compute a differential equation solution in an on-line computing application approximately every three minutes.

The fully automatic systems or functions may have representation at any point of this spectrum but they are usually found in the microseconds-to-seconds range. On the other hand, in the case of man/machine systems, the interval between events initiated by a human operator is measured in human terms, i.e., in the fractions of seconds-to-minutes range. Of course, in the case of multiple operators, as in the case of the reservation systems, the interval between effective events may be much less. In addition, in fully automatic systems, the events often occur at fairly uniform or predictable intervals while in the other case the frequency is more of a statistical estimate which is a parameter employed in the system design.

Response time — The response time spectrum for these same selected applications also appears in Figure 1. These are the response times for the actual computers used for these applications. The use of a larger and/or faster computer for the same application would in most cases result in a shorter response time. However, the response times illustrated can be considered to be typical requirements for the indicated applications.

It is significant to note that the ratio of response-time to interval-between-events for the cited applications are respectively:

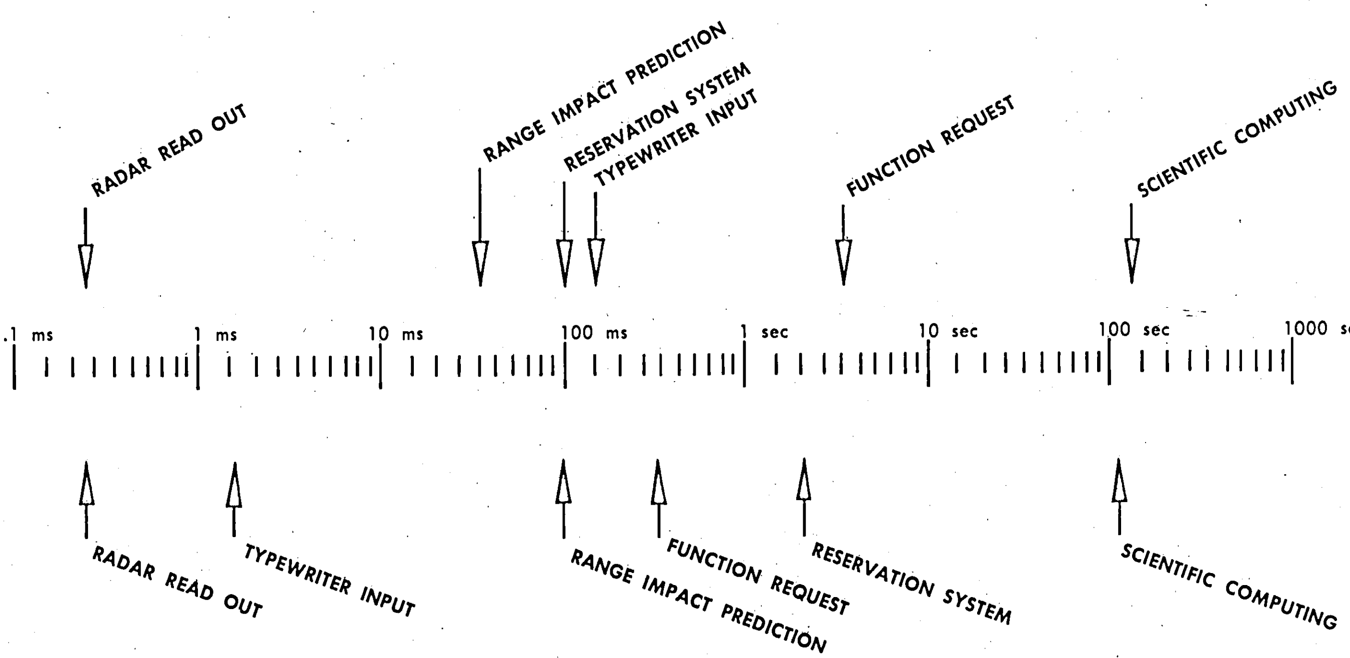
	Response Time	Interval	Ratio
Radar readouts	250 us	250 us	1.00
Impact points	100 ms	50 ms	2.00
Reservations	3 sec	100 ms	30.00
Typing	2 ms	200 ms	.01
Function requests	500 ms	5 sec	.10
Scientific Computing	2 min	3 min	.66

If the internal processor capacity is tied up throughout the response time for a single event, then those cases with ratios larger than one cannot be satisfied by a single computer. On the other hand, if some of the processing requirement involves multi-computers or buffered input/output, then the response times can be shared over a number of events and cases for ratios larger than one may be satisfied. This is the case for the second and third examples above. For the ratios less than one it is possible for a single computer to accommodate a number of transactions at least as large as the reciprocal of this ratio.

In addition to the computer-dependent response time there is also the maximum allowable response time. This time is a system design specification and its value is usually less than or equal to the time between events. It is then a programming requirement to insure that the response

Fig. 1

APPLICATION SPECTRUM OF INTERVALS BETWEEN EVENTS FOR TYPICAL ON-LINE SYSTEMS



RESPONSE TIME SPECTRUM FOR TYPICAL ON-LINE SYSTEMS time is less than or equal to the maximum allowable response time. Typically this is achieved by selecting an appropriately matched computer system.

In a fully automatic system with a high frequency of events, the maximum allowable response time usually corresponds to the interval between events. In these types of applications the interval between events is specified independently of the program and failure to meet the maximum allowable response time implies loss of data or inadequate frequency response. This, of course, affects the amount of processing that can be performed and in extreme cases the system may be limited to merely recording the incoming data.

In the man/machine system there are two kinds of response times to be considered. The first involves entry of information by an operator, under program guidance, to make up a complete message or action request. The second is the processor response to that message.

An example of entering of information is the step-by-step composition of a message by use of a typewriter keyboard. Typically, each key generates one or more characters of information which are collected, operated upon, and stored by the computer. Pressing a key may also cause a message from the computer to be generated which provides guidance for the next step or which requires the entry of additional information. In any event, this mode of operation must permit the operator to enter data at his own speed. This is typically specified to be five characters per second maximum. Therefore the maximum allowable response time is 200 ms. This does not usually place any burden on the program since the processing of the individual entries requires only a few milliseconds of computer time for processors in the 10-15 us memory cycle range. The response time may become a factor when multiple communication devices must be serviced by a single processor within the same 200 ms interval.

The second aspect of response is the fulfillment of a particular request by the processor. The total action may involve:

Request validation
Information retrieval
Information transformation or computation
Data formatting
Output generation

Depending upon the specific application, the total response time required of the system for these actions may range from a few seconds in a reservation system to several minutes in the case of a library search or solution of a differential equation. In many of these cases the total response time is determined not by the frequency of requests but rather by the desire to respond to the request as rapidly as possible, but within reason.

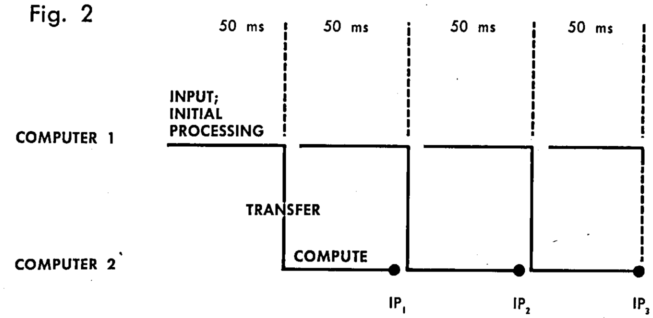
Examples of on-line applications where the response time is greater than the interval between events are:

a) Range Impact Prediction—
50 ms interval between events, 100 ms response time

b) Reservation System—
100 ms interval between events, 3 second response time

In the first case two computers can be employed in tandem. One computer reads in the data and performs initial processing (coordinate transformations and smoothing) for approximately 50 ms and then transfers the results to the second computer where the trajectory equations are computed. Meanwhile, the first computer is processing the next input. This process is illustrated in Figure 2. The 100 ms total response time is divided between the two computers and each computer maintains a 50 ms response time. This is analogous to parallel processing techniques employed in other on-line applications where the buffer time for one event is shared with the processor time for another event. Thus, in many cases,

Fig. 2



IMPACT PREDICTION OPERATION

the system is able to process the events as they occur, without losing data, by overlapping response times for events.

In the second example noted above, the reservation system, the 100 ms interval between events is for peak load conditions. The average interval time is much larger. The incoming events must be placed in a queue and then processed in the order received or according to some other priority scheme. Under these conditions the system may not be able to achieve the 3 sec response time even using parallel processing techniques. However, the delay in the response is not catastrophic since no data is lost and the additional time does not significantly affect system performance.

Special Timing

The special nature of the on-line device interface and the possibly required electromechanical responses may impose special timing requirements on the computer program. This depends, of course, upon how much of the detailed bookkeeping and control is actually committed to hardware. Some representative items which must be considered are:

a) Refresh—in a man/machine communication display console, if automatic buffers do not refresh the volatile cathode ray tube (CRT) display, then the computer must re-transmit the information sufficiently often, e.g., every 20-25 milliseconds in order to prevent flicker.

b) Scanning—it may be necessary to insure the clearing out of special on-line device registers sufficiently often if data entry is to continue at the appropriate speed.

c) Outputs—special devices may require output data that must meet specified timings. This may be true, for example, of electromechanically driven devices where start/stop problems may arise.

The actual implementation of these response requirements will depend upon the availability of suitable interrupt features, an active or passive clock, and sufficiently long buffers. If these features are not available, programming must take their place at the expense of possibly affecting total system efficiency. In this regard an important point is that timing may not always be critical and cycles can be skipped now and then. For example, a CRT image would not suffer from occasional misses at refreshing, or from a 5-10 millisecond delay in a cycle. However, while such random perturbations will not adversely affect the viewer, periodic misses or a slow down of the cycle will be noticed.

Many Stations

Typically, an on-line system will have a number of transaction stations operating simultaneously. The actual number or maximum number employed depends upon the speed and size of the processor and its ability to keep up with events. In general, the number of transaction stations per processor can increase as the event frequency per station decreases.

Multiple stations will be competing for servicing from the time shared processor. It is therefore necessary to make proper programming allowance for assuring equal servicing for all stations, or service which is a function of the priority assigned to each station.

Examples of numbers of stations that are typically employed are:

a) SAGE: 100
b) Banking: 25
c) Airline Reservation: 500-1000

Independent Functions

In addition to the fact that there are many stations being employed, it is also true that individual functions or tasks will usually vary from one station to the next. The stations may be engaged in different functions or may be at different stages of operating on the same function. In addition, they may be accessing the same files. For example, in one banking system sixteen different operations are defined, any of which may be initiated by any of the stations at any time. This places the requirement upon the designer to determine which functions can be handled simultaneously and which cannot. Thus, two essentially simultaneous transactions executed by independent functions upon the same record may invalidate both transactions as, for example, when one function updates one aspect of a record while the other function is updating another aspect of the same record.

The degree to which time sharing of the processor is implemented will also affect the number of independent functions which can proceed simultaneously. For example, internal memory size may be a limiting factor restricting the number of active programs to a small number.

Random Transactions

Processing of events which are asynchronous, unscheduled, and often unpredictable in their time of arrival and transmission is a characteristic of most on-line systems.

In man/machine systems the majority of the transmissions between the processor and the real time devices are of this type. The man controls the system and proceeds at his own pace, performing some functions in rapid succession and others more slowly because of the need to analyze and determine the next step to perform.

There can also be random transactions in a completely automatic system, such as responding to an equipment malfunction or to a programmed event in an automatic checkout system.

Because of these random transactions, interrupt techniques and/or frequent scanning of input lines is required of the program.

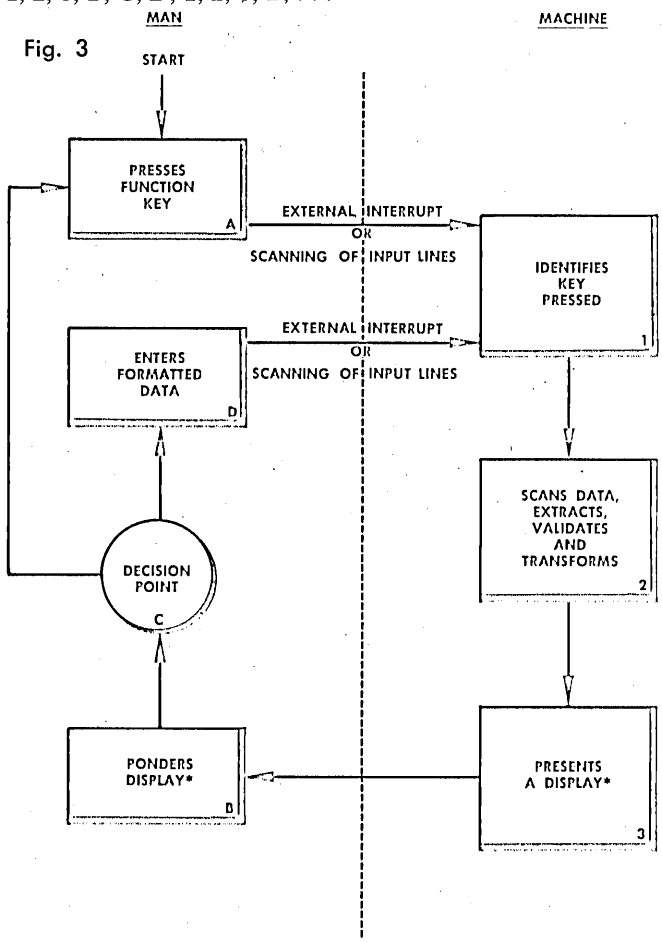
Many Short Demands

The on-line system is usually subjected to many short duration demands in the microseconds-to-seconds range. In the automatic system where the demands occur at intervals in the microseconds range, practically all of the processor's capacity must be devoted to servicing the demands. In the other cases, as typified by the man-machine system, the demands are many in a human sense, i.e., they occur at intervals in the seconds range. The response to the demand, however, is usually measured in milliseconds. Thus, there is excess capacity in the processor which can be used to service additional on-line devices or which can be used to perform other time shared tasks.

Consider, for example, an alpha-numeric entry by an operator via an electronic typewriter station (typewriter keyboard and CRT viewer). The operator would depress a key to indicate his desire for service. The computer would respond with a ready signal (display) and/or present a message (display) requesting the operator to select an entry format type. The operator would now enter a selected code—say two characters—and depress the "end of message" key. The selected format would be displayed and the operator would enter his data characters. The sequence performed by man and machine up to, and including, the first data character entered into the selected format is, using the steps shown in Figure 3:

A, 1, 2, 3, B, C, D, 1, 2, 3, B, C, D, 1, 2, 3, B, C, A, 1, 2, 3, B, C, D, 1, 2, 3, B, . . .

Fig. 3



MAN/MACHINE INTERACTION

*INCLUDES CRT, TYPED COPY, LIGHTS, ETC.

The total internal processing time for the above actions may be on the order of 10 milliseconds for a medium speed computer. In addition, up to several hundred milliseconds may be required for accessing and presenting displays if they are needed and auxiliary storage is used. Since the total elapsed time for the entry in the above example may be on the order of five seconds, it can be seen that very little of the total computer capacity has been used. If the I/O transfers are buffered then the references to auxiliary storage will not be additive with the processing time. On this basis, a single station would utilize 10/5,000 or .2 percent of computer processor capacity and k stations would require .2k percent of capacity. It is interesting to note that for larger and faster computers the denominator (the operator time) remains the same whereas the numerator decreases. It is concluded that a single processor is capable of servicing many stations and still have large capacity left over for other functions.

Time Sharing

In all cases where there is a low frequency of events coupled with a high response time there will be a considerable amount of processor capacity available for tasks other than the routine servicing required by the on-line devices. Because of the amount of processor capacity available between events, it is desirable to overlap functions by means of time sharing, and perhaps by multiprogramming, in order to achieve efficiency. A good part of this capacity, if not all, may be required to actually execute the functions which may be initiated by completed messages or requests. In addition there will be other computations related to the total problem, or for that matter secondary in nature, which will use up excess capacity; for example, performing equipment diagnostics during idle time.

In determining the extent to which time sharing can be implemented, careful consideration must be given to such factors as:

a) The amount of processor memory required for permanent routines. These are routines which, due to their frequency of use, must be in primary storage at all times; for example, executive control and interrupt handling routines.

b) Segmenting of operational routines. The feasibility of segmenting operational routines must be determined. This may be required because of a limited amount of working memory available after the space requirements for the permanent routines have been accounted for.

c) Type of mass storage available for programs. The amount of storage and the random access and transfer times must be considered. For example, the use of magnetic tapes with their serial access capability may make time sharing impractical for some applitions.

Auxiliary Storage

The requirement for rapid response and the servicing of many stations, each actively spread over a long period of time relative to the effective computing rate, leads to the need for large capacity auxiliary storage. This is especially true for man/machine systems which must have access to many individual programs, working displays, and the system data base. While this points to the need for random access, bulk storage such as drums or discs, it is still possible by appropriate program design, to be reconciled to magnetic tapes. However, the use of magnetic tapes will almost invariably result in longer response times and/or higher system costs, since meeting specified response times may require assignment of individual tape transports to each transaction station.

Large Data Base

This characteristic is not necessarily typical of the automatic, closed-loop systems since these function mainly as data collectors requiring a relatively small amount of retrieval. However, systems with on-line man/machine communication capabilities invariably are systems involving large amounts of information in a data base or through a generating function. Otherwise, the on-line devices can hardly be economically justified. Since man and his judgment are also involved in the operation of the system, there is the requirement that this data be randomly accessible. Hence file design and storage media are important considerations in system design. It is often necessary to make tradeoffs between file designs which allow the most efficient file maintenance and designs which allow the most efficient data handling by the human operator.

Simultaneous File Access

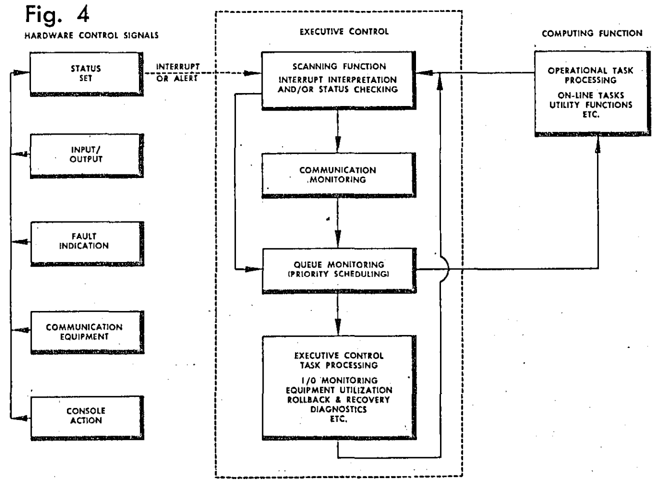
This characteristic is closely associated with the large data base aspect and is more typical of man/machine systems than closed-loop automatic systems. The point to be considered here is that, for the most part, the multi-functions and multi-users may be operating on a single file. Hence, it is necessary to maintain file order and, most important, file integrity. The latter is a problem since one function may be extracting information from a record at the same time another function is modifying the record. Furthermore, these functions may have been initiated either automatically or by human operators.

executive control programming requirements

The problem characteristics described in the previous section impose a number of requirements on the Executive Control programming system. Chief among these is that the system must have continual cognizance and control over the operating environment both from the point of view of sequencing the jobs and meeting the timing requirements. This requires executive control over and above that required in the conventional data processing environment. It furthermore implies an interrupt and priority handling capability commensurate with the hardware which has evolved in modern real time computers.

The Executive Control in an on-line programming system serves as the interface between the real world and the computer. As the timing requirements become more stringent, i.e., as the application appears further to the left on the real time spectra of Figure 1, the demands for automatic (program) control become greater. The complexity of the Executive Control is also a function of the number of unrelated or independent tasks which the system is required to perform. However, to a greater or lesser extent, the modern hardware design simplifies the automation of such things as priority, interrupt, I/O, etc.

Fig. 4



EXECUTIVE CONTROL DESIGN AND FUNCTIONS

A conceptualized diagram of the Executive Function appears in Figure 4. The normal procedure is for the Operational Tasks to be executed under the auspices of the Executive. Upon a hardware signal or completion of a task control is returned to the Executive. These signals may be due to:

a) Readiness of and I/O channel
b) Completion of an I/O transfer
c) Data transfer fault condition
d) External signal
e) Special internal machine status changes

Upon receipt of the signal, the first job of the Executive is to check status in order to determine if special communication monitoring is required or if the standard queue monitoring should be initiated. As can be seen, Executive Control task processing is treated in exactly the same way as operational task processing, thus making it possible to modify the Executive Control without redesigning the whole system.

The principal functions which the Executive Control must handle will be discussed in Part Two, to be published next month. ■

BIBLIOGRAPHY

1. Bauer, W. F., and W. L. Frank. "DODDAC—An Integrated System for Data Processing, Interrogation, and Display". Proc. of the EJCC, December 1961.
2. "Computers in Command and Control". Institute for Defense Analysis, Technical Report Number 61-1Z, November 1961.
3. "Where the Computer Fits in Command and Control". Armed Forces Management, July 1962.
4. "First Commercial Closed-Loop Computer Control System—A Review". Auto. Contr., Vol. 16, No. 2, February 1962.
5. Lewis, T. B., "Digital Computer Equipment for an Advanced Bombing, Navigation and Missile Guidance Subsystem for the B-70 Air Vehicle". Proc. of the IRE, Vol. 49, January 1961.
6. Truxal, J. G., "Computers in Automatic Control Systems". Proc. of the IRE, Vol. 49, January 1961.
7. Gass, S. I., et al, "Project Mercury Real-Time Computational and Data-Flow System". Proc. of the EJCC, December 1961.
8. Wilson, A. N., "Use of a Combined Analog-Digital System for Re-entry Vehicle Flight Simulation". Proc. of the EJCC, December 1961.
9. Vazsonyi, A., "An On-Line Management System Using English Language". Proc. of the WJCC, May 1961.
10. Pardee, D. R., "Real-Time Management Control at the Hughes Aircraft Company". Proc. of the WJCC, May 1961.
11. "United Books Flights on Instamatic". Management and Business Automation, June 1961.
12. Sampson, D. K., V. E. Herzfeld, and C. W. Fritze. "The UNIVAC Airlines Reservation System, A Special-Purpose Application of a General-Purpose Computer". Proc. of the EJCC, December 1958.
13. "A Survey on Airline Reservation Systems". Datamation, June 1962.
14. Perry, M. N., and W. R. Plugge. "American Airlines 'SABRE' Electronic Reservation System". Proc. of the WJCC, May 1961.
15. "Savings Bank Automation with Real Time Computers", Data Processing, Vol. 3, No. 12.
16. Shannon, W., "Instant Service at the Teller's Fingertips". Journal of American Bankers Association, September 1962.
17. Heskin, J., "The Saturn Automatic Checkout System". Proc. of the EJCC, December 1961.
18. Dahnke, J. W., "Computer-directed Checkout for NASA's Biggest Booster". Contr. Eng., Vol. 9, No. 8, August 1962.
19. "Data Processing for Hotels". The Cornell Hotel and Restaurant Administration Quarterly, August 1962.
20. Licklider, J. and W. Clark. "On-line Man-Computer Communication." Proc. of the WJCC, May 1962.
21. Issue of Computers and Automation, February 1962.
22. Stark, L., R. Payne, and Y. Okabe. "On-line Digital Computer for Measurement of a Neurological Control System". Comm. of the ACM, Vol. 5, No. 11.
23. Roach, C. J., M. H. Ellestad, and R. B. Lake. "Medical Data Processing and Computer Automated Hospitals." Datamation, June 1962.
24. "Western Reserve Up-dates Information Retrieval Center". Business Automation, July 1961.
25. Corbato, F. J., et al. "An Experimental Time Sharing System". Proc. of the WJCC, May 1962.
26. MIT, Report of the Long Range Computation Study Group, April 1961.
27. Cullen, G. J. and R. W. Huff. "Solution of Non-linear Integral Equations Using On-line Computer Control". Proc. of the WJCC, May 1962.

COMPUTER TRAINING & EDUCATION

a reflection on progress

by FRED GRUENBERGER, The RAND Corp., Santa Monica, Calif.

The world of computers is most obviously a big world, and getting far bigger. For any measure applied to it, the second derivative is positive. For example, taking top speeds as a rough index of computing power, the total computing capability of the U. S. will at least double in the next 12 months. months. This doubling will be due almost entirely to mass production—a production breakthrough leading to more speed per dollar would probably increase the rate of growth.

It is dangerous to extrapolate from such figures, but it is fascinating to conjecture on the long range implications of a rate of growth which is so high. Either the present users of computers are going to step up their consumption by large factors in the next decade, or the number of users will greatly increase, or both. My personal inference is that the number of persons who have regular access to a million executed instructions is likely to increase by a factor of a thousand or so within 10 years. All of these people must be at least trained, if not educated. The question is, what's so different about this educational problem from the problem of training and educating masses of people in any other skill or technology?

Part of the answer lies in the differences between the computing world and all others. Let me list some of the things that are different about the computing world.

1. I have already touched on the extremely high growth rate, which is probably exceeded by no other technology.

2. The computing fraternity exhibits a degree of cooperation among its users which is unique. We are all familiar with the cooperative programming groups that have sprung up around machine types. Compared to other fields, much less information is kept proprietary in computer work. In fact, just the converse is the norm; individual users race to fill central libraries with better versions of utility routines, interpreters, and even new FORTRAN compilers.

This cooperation extends to machine time, too. Every prudent user arranges for backup of his machines on a reciprocal basis.

It is difficult to think of corresponding examples of either type of cooperation in any other industry.

3. Every computer user can obtain, on delivery of his late-model machine, a detailed description of just how obsolete it is. This is due partly to the long lead time involved in installing a machine, coupled with the dynamic improvement in hardware and software. We always have dangled in front of us how much better off we will be in two years. Among both users and manufacturers, the complacent person is more than half dead. The point here is that our industry tends to be livelier than others.

Consider the case of a plant manager who has (a) just replaced his truck fleet with the latest model trucks, and (b) just replaced his old computer with the latest model. In the one case he can feel reasonably assured that a portion of his plant is now operating at optimum efficiency; in the other case, he is apt to have doubts. For one thing, he knows of better equipment already available (it is two years since the computer was ordered); for another, he has good reason to suspect that his computer people don't know how to get maximum use out of the machine.

4. Since computers are the ultimate in precision instruments, they can be described with precision; moreover, the effectiveness of the hardware/software combination can frequently be estimated for any job or mix of jobs. This is not to say that we see any evidence that this is done; merely that it is possible to a greater degree in computing than in other industries.

5. The computer offers vast power to an individual to a degree not found in any other technology. An individual cannot operate a B-52 by himself; if he could, he is seldom in a position to make effective use of 15 seconds, worth of its capability. But John Q. Citizen can buy—anywhere—that million executed instructions. The going rate is about one dollar, and falling. Computing power seems to be a fluid commodity available to all.

present-day status

If computing is different from other technologies (and the list of differences could be extended), then I think it follows that its training problems differ also, if only in size. The manufacturers and users are already aware of having an acute training and education problem on their hands; I suspect that the problem is growing a great deal faster than any possible solutions so far considered. I am told that the number of IBM 1401s on order approximates the total number of computers now in the world. Considering that there are lots of other machine types on order, the implication to me is that the real problems in training and education are still ahead of us. The movement of computers into the secondary schools, for example, is just beginning. That movement is likely to have a steep wave-front and hence a significant shock-wave.

Is anything being done to solve the problems before they become really acute? Of course *something* is being done. But I contend that the efforts are woefully small, in view of the magnitude of the pending task. Those who need computing knowledge will get it (or what appears to be a fair facsimile). They won't know what they *aren't* getting, and they will proceed to use computers anyway. I think we are proposing to bail out the ship with a teaspoon. Perhaps this is a better metaphor: we are gearing up to change the tire while going down the freeway at 70 mph. There ought to be a better way, and since we computing people keep telling each other how smart we are (an added item to that list of things that are different about computing), we ought to look for it. We may find it by accident, but that's a rather slim hope.

Several years ago I addressed the Northwest Computing Association on the subject of computing in the universities. At that time, I divided up the domain of the university computing center into four parts: within-campus computing service, outside (i.e., contract) work, training, and research. I suggested then that one might be hard pressed to name more than six universities that were doing anything significant in the last two categories (training and research). The absolute number is higher today (perhaps as many as 10), but the percentage hasn't changed much. It's probably gone down.

What little is being done centers around industry. The computer manufacturers are in the forefront; they must train their own people as well as many of the users. Many industrial users, of course, are up to their ears in training new people on the job.

the problem

What is so difficult about training and/or educating masses of people in computer technology? It's hard to say, but I can ask a great many more questions than I can answer.

The first question anyone has to face is this: what is it we want to teach? Do we want to teach how to get answers? Or do we want to teach how to solve problems? Or do we want to teach computing? Or computers? Or numerical analysis? Or what? *These are all different subjects.* Do we want to teach them all, in a one-semester course?

Some of the answers I hear, when I raise these questions, strike me as illogical. Take, for example, the person who wants to cover many of the fields listed above, and who proposes to do it by starting with a magic language (FORTRAN, NELIAC, MAD, etc.) "so the students will have something to work with right away." It sounds so plausible, especially if you say it fast.

But the person who says this didn't learn by that route himself. Moreover, we can see plenty of examples of where that approach failed—the graduate seems to be somewhat lacking in the basics. Now, I am the first to attack the line of reasoning that says "we did it this way, and it worked, and therefore it must be the correct way (or *a* correct way)." On the other hand, I'm inclined to defend the reasoning that says "we did it this way and it didn't work, and therefore it's probably not a correct way."

Let me raise several other questions. Will the students have access to a computer while they learn? Will they be exposed to a vast number of facts and concepts rapidly in the hope that a fair percentage will rub off? Will they "learn while doing?" (This is sometimes a euphemism for "we can't wait to do a good job, since we're desperate to get today's jobs done.") Will they have a crash course, perhaps followed by apprentice training at the feet of some expert?

I'm merely suggesting that there is room for discussion as to how the job should be done. All too often we seem to reach what seem to be, at the time, quite plausible answers by rationalization. For example, in the early days of the game there were many courses in computing given at places which had no computer. A bit later, courses built around a hypothetical machine became popular. And for a time it was argued vehemently that access to a wired calculator would suffice. I submit that in all three of these situations, the real learning atmosphere began with the subsequent exposure to a real computer, and we had simply fooled ourselves. To be sure, we were forced into this rationalization by the lack of available equipment at the time.

Even today, the argument is advanced seriously that it is vital, in teaching computing, to advance to symbolic assembly language as soon as possible. This is fine—on a binary machine. The argument weakens somewhat when one postulates a computer for which the (decimal) machine language is acceptable. Notice that I am not arguing against symbolic assemblers (or even high level narrative compilers)—I am simply questioning the reasoning by which we conclude what ought to be done.

If the problem is large and complex (and I think it is) and little has been done to attack it systematically (and I believe that's true), then I think we should be on guard against an influx of charlatans, who generally have answers for everything.

For example, I recenlty examined a "model" curriculum for a junior college course in data processing. In the 15-weeks' course, the students were to be studying punched card sorters in the third week and monitor systems in the fifth week.

In another case, a course outline stated that ". . . reference to number systems other than digital" would not be included. That may have been a typo; I don't know. The point is that I hardly believe we are yet ready to freeze a curriculum, particularly at the secondary school level. But in the absence of any massive attack on the problems involved in mass training, the quacks will take over, or try to. We might reflect on what they've already done with "teaching machines."

is there hope?

Sure there is. The best thing we have going for us is the computer itself. It is what makes the problem unique; it also helps solve it. A computer is a superb teacher, all by itself. If I had to start from scratch, I'd rather have a computer and no teacher than the best teacher and no computer. In this tradeoff, the computer is constant, so given any computer and even a poor teacher, we're not in too bad shape.

The 5% law is also in our side. This is the principle that says that after you've learned one machine, the next one requires only 5% more effort. A corollary to this is that while learning that first machine you were really spending most of your time learning fundamentals and very little time actually learning the machine.

Better equipment keeps appearing on the market. One has only to compare today's machines with the best of a few years back to realize the tremendous increase in the computer-per-buck ratio. This ratio is bound to increase even more. To put it another way, computing power is certain to be extended to more and more people—which gives more chance to explore ways to teach the subject. Computing-on-the-end-of-a-wire is almost upon us. The new equipment is not only better in the sense of more for the dollar, but is easier to use.

Systematic computer training, both in industry and in our academic institutions, is bound to improve. The improvement could be less of a hit or miss proposition, it seems to me, with better planning of our goals. ■

COMPUTER CHARACTERISTICS... AN ADDENDUM

process control, DOD and foreign hardware featured

Each November for the past three years, DATAMATION has reprinted in full the current edition of the Computer Characteristics Quarterly, compiled and published by Charles W. Adams Associates, Inc. as an adjunct to its principal services of developing techniques, systems and programs for man-machine communication and on-line data processing.

Until recently the Quarterly specifically omitted computers manufactured outside the United States, those intended primarily for use in process control, message switching and other specialized applications, and systems developed for the Department of Defense and other U.S. Government agencies. In recognition, however, of the increasing importance of real-time systems and the mounting interest in computers produced abroad, the December 1962 issue of the Quarterly was expanded to include this information in three new sections. These are reprinted on the following pages with the permission of Adams Associates.

The appearance of question marks and stars indicate the absence or uncertainty of data at this time, a shortcoming which, we have been assured, will be overcome in subsequent issues. Moreover, in the military and process control sections, price information has been omitted since it would be almost meaningless without elaborate discussion of what part of the overall system is involved. Relatively detailed data on real-time, input-output channel capacity has been included because of the critical importance of this factor in such systems.

Readers who wish to reference the salient features of virtually all the general purpose, digital computers produced in the free world can do so by sending $10 for a one-year subscription to Adams Associates, Bedford, Massachusetts. The Quarterly appears in two formats: a pocket-sized and an 8½" x 11" booklet.

SECTION II

PROCESS CONTROL COMPUTERS

	First Delivery Month and Year	*Processor Speed* Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	*Internal Storage* Capacity in Words / Type	*Logic* Word Size / Instr. Addresses	*Magnetic Tape* Thousands of Characters per Second / Buffering	Maximum Units Attachable	*Random Access File* Capacity / Access Time in Milliseconds	*Peripheral Devices* Cards per Minute In — Out	Paper Tape Characters per Second In — Out	Printer Lines per Minute	*Input-Output Channels* Number and Type	Direction	Transfer Rate	*Other Features* Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter
DAYSTROM 46	/58	220	10	4-16K core	21b[E]	83	4	—	12	300	300	1p[Q]	I & O	100K	√	1	—	—	I/O
				16-100K drum	1	RC, WC		—	12	110									
	E. Double-precision arithmetic available. Q. Channel with direct access to memory.																		
DAYSTROM 136	8/61	28	4	4-16K core	25b[E]	83	4	—	12	300	300	1p	I or O	890K	√	64	√	—	I/O
				16-172K drum	1	RC, WC		—	12	110									
	E. Double-precision arithmetic available. Note. UNICOMP (Universal Compiler for Process Control) available 4/63. System designed to work in vans and under extreme temperatures.																		
DAYSTROM 636	8/63	17	4	4-32K core	15b[E]	83	4	—	60	300[N]	300[P]	1p[Q]	I or O	890K	√	2	√	—	I/O
				16-262K drum	1	RC, WC		—	12	110									
	E. Double-precision arithmetic instructions included along with variable-field selection and partial operand instructions. N. Lower speed paper-tape equipment available. Eleven column lines. Input-output channel has direct access to the core memory UNICOMP (Universal Compiler for Process Control) available 4/63. System designed to operate in vans and at extreme temperatures.																		
COLLINS DATA CENTRAL ★	?	10	5	4-65K core[D]	1	?	4	40M[K]	?	350	600	1sp	I & O	1.6K	√[T]	?	?	?	?
		6			1	MRWC			?	110		1sp	I & O	2.4K					
	D. System features programmed logic through a separate "programmed logic unit storage" of 512 to 1,024 words of one-microsecond cycle time, non-destructive. K. Disc storage of 20 million bits per disc and 12 discs per unit. T. Priority processing.																		
GENERAL ELECTRIC 312 ☆	?	196	?	8-56K drum[D]	20b	—	—	—	—	60	—	1	I & O	3K	?	?	?	?	?
					1	—		—	—	60									
	D. 640-word fast track.																		
GENERAL ELECTRIC 412 ☆	?	40	?	4-8K core	20b	—	—	—	40	100	—	1	I & O	3K	?	?	?	?	?
					1	—		—	—	100									
GENERAL PRECISION LIBRATROL 1000 ★	/60	1000	17000	8-16K drum[D]	32b	—	—	—	—	500	300	1s[Q]	I	2K	—	1	—	—	I/O
					2	—		—	—	60		1p	O	2K					
	D. Drum offers 128 words in dual access tracks of four- to six-microsecond access time and eight words with a two-microsecond maximum access time. Q. Input system features a 200-microsecond analog to digital converter yielding 11-bit words. Control outputs are 10-bit words. Note. This is the industrial control version of the RPC 4000.																		
HONEYWELL 290 ☆	/60	140	20	1-4K core	186	15	2	—	—	110	—	?	?	?	√	√[U]	—	√	I/O
				8-256K drum	1	RC, WC		—	—	60									
	U: Any core location may be used as an index register.																		
IBM 1710 ☆	2/62	560[B]	20	20-60K core	1d	—	—	15M[K]	250	150	—	?	?	?	√	—	√	√	I/O
					2	—		250	150	15									
	B. Add time assumes a five-character field. K. Up to five 1311 disc drives with interchangeable packs of three million characters each. Note. This is the industrial control version of the IBM 1620 computer.																		
ITT 025 ☆	/59	16	8	16-65K core[D]	32b	15	16	1.3M[K]	250	—	900	86p	I & O	2.4K	√	256	√	—	—
					1	RW		9.6	100	—		32p	I & O	4.8K					
												1p	I & O	.6M					
												1p	I & O	.6M					
	D. 256-word index and 256-word program storage included with six-microsecond access time. Up to sixteen drums of 16,000 words each.																		
ITT 525-VADE ☆	3/63	6	2	4-32K core[D]	32b	15	8	—	—	300	8	16s	I & O	8M	√	1	—	—	I/O
					1	RW		—	—	10		128s	I & O	?					
	D. Six-microsecond commands use micro-logic mode and require three memory cycles.																		
RAMO-WOOLDRIDGE TRW 300 ★	?	780	?	8-15K drum	18b[E]	7.5	8	—	—	60	—	1024p	I	25K	?	?	?	?	?
					2	—		—	—	60		128p	O	600					
	E. Two words per instruction.																		
RAMO-WOOLDRIDGE TRW 330 ★	?	2600	?	7-130K drum	28b	—	—	—	?	?	?	1	I	1680	√	3	?	?	?
					1	—		—	?	?		1	O	100K					
RAMO-WOOLDRIDGE TRW 340 ★	?	16	8	4-16K core	28b	—	—	—	?	?	?	1	I	1680	?	?	?	?	?
				8-131K drum	1	—		—	?	?		1	O						

☆Information supplied or confirmed by manufacturer but not reviewed in detail by publisher.
★Incomplete information compiled from various sources but not confirmed by manufacturer.

SECTION III

DOD & OTHER U. S. GOVERNMENT SYSTEMS

	First Delivery Month and Year	*Processor Speed* Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	*Internal Storage* Capacity in Words	Type	*Logic* Word Size Instr. Addresses	*Magnetic Tape* Thousands of Characters per Second	Buffering	Maximum Units Attachable	*Random Access File* Capacity Access Time in Milliseconds	*Peripheral Devices* Cards per Minute In — Out	Paper Tape Characters per Second In — Out	Printer Lines per Minute	*Input-Output Channels* Number and Type	Direction	Transfer Rate	*Other Features* Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter
GENERAL PRECISION LIBRASCOPE L-2010 ★	?	78	10000	4K	disc	30b	—		—	—	—	300	—	1p	I or O	?	—	—	—	—	I/O
						1F		—		—	—	100									
HONEYWELL PICO ☆	?	12	2	3-8K	coreD	24bE	—		—	—	—	—	—	32sp	I	3M	√	—	—	—	—
						1		—		—	—	—	—	48sp	O	3M					
IBM AN/FSQ-7 (SAGE) ★	11/57	12	6	69K	core	32bE	18.6		8	—	150	—	500	—Q	—	—	—	4	—	—	—
				153K	drum	1		MRWC		—	100	—									
IBM AN/FSQ-32 ★	10/60	2.5B	2.5	81-163K	core	48bE	62.5		24	4.4M	200	—	600	32sQ	I	1.3K	—	13	√	√	I/O
						1		MRWC		11	100	—		40s	I	1.6K					
														32sp	I	500					
														25s	O	1.3K					
														8s	O	1.3K					
														25sp	O	500					
PHILCO AN/TYK-4V (COMPAC) ★	?	24	12	4-16K	core	36b	45		8	—	—	300	—	1sp	I & O	45K	?	11	—	—	I/O
						1		RC, WC		—	—	30									
PHILCO BASICPAC ★	11/59	24	12	4-28K	core	38b	45		35J	—	—	30	—	7spQ	O	75-1.2K	√	4	—	—	O
						1		MRWC		—	—	20		6sp	I	75-1.2K					
														1sp	I	75-1.2K					
RAMO WOOLDRIDGE TRW 130 ☆	?	12	6	8-32K	core	15b	15-41		16	—	200	300	150	2p	I or O	1M	√	—U	√V	—	I/O
						0-1F		RWC		—	—	60		1p	I or O	500K					
														1p	I or O	25M					
SYLVANIA 9400 ☆	10/60	8	4	16-32K	core	36b	90		252J	6300MK	2000	1000	900	4p	I or O	740K	√	7	—	√	O
						1		MRWC		53	250	100		1p	I & O	9M					
SYLVANIA AN/MYK-1(U) (MOBIDIC) ☆	/59	16	8	4-8K	core	36b	38		252	315MK	800	1000	900	4p	I or O	237K	?	7	?	?	?
						1		MRWC		53	250	100		1p	I & O	4.6M					
SYLVANIA M-64 ☆	?	10	4	28K	core	24b	45		4	—	200	1000	300	4sp	I	43K	√	2	—	—	O
						1		MRWC		—	100	100		1sp	O	42K					
UNIVAC 1206 ☆	/58	9.6B	8	16-32K	core	30b	25		168J	377MK	600	1500N	600	12p	I or O	.9M	√	7	—	—	I/O
		11.2				1		MRWC		17	150	110	700	2p	I or O	.9M					
UNIVAC TARGET INTERCEPT ★	10/60	10	2.2C	12K	coreD	24b	30		5	—	—	—	—	?	?	?	√	15	—	—	I/O
						1		?		—	—	—	—								

GENERAL PRECISION LIBRASCOPE L-2010: F. Instruction contains address of next instruction to be executed. Note. System designed for mobile operation. Size. Computer area 2 cu. ft., weight 60 lbs., power consumption 500w.

HONEYWELL PICO: D. 3,000 words of biax memory permit non-destructive readout for program storage. E. Three instructions stored in every two memory words. Note. System designed for operation in missiles and airplanes under extreme temperatures. Size. Height 5″, width 13″, depth 10.5″, weight 20 lbs., power consumption 46w.

IBM AN/FSQ-7 (SAGE): E. Accumulator split into two 15-bit plus sign registers. Arithmetic commands operate on either half or full (but not coupled) accumulator. Q. All input-output is buffered through the drum system. Size. Computer floor area 1,508 sq. ft., weight 113 tons, power consumption 1,500 kw.

IBM AN/FSQ-32: B. Instruction look-ahead allows increased internal speed. E. Data can be handled in the arithmetic section as two independent half words of 24 bits each, or as six-bit "bytes". Q. Output data must be buffered through "dator" drum system, which has 139,000 words. Input-output data channel has maximum rate of 1.5 million bits per second. Size. Computer floor area 1,161 sq. ft., weight 90 tons, power consumption 203 kw.

PHILCO AN/TYK-4V (COMPAC): Note. Major component of the U. S. Army automatic data processing system. Designed for mobile operation under extreme temperatures. Size. Computer area 9 cu. ft., weight 200 lbs., power consumption 4 kva.

PHILCO BASICPAC: J. One I/O converter handles five I/O devices. System can have up to seven I/O converters. Q. One communications converter handles up to seven two-way real-time channels. Note. System designed to operate in vans and under extreme temperatures. Size. Height 21″, width 24″, depth 60″, weight 900 lbs., power consumption 2.5 kw.

RAMO WOOLDRIDGE TRW 130: F. Two instructions per word in no-address mode. U, V. Indirect addressing, indexing, and multiple-word-length operations facilitated by micro-programming technique. Note. Designed to operate in vans, ships or airplanes under extreme temperatures. Size. Height 59″, width 20″, depth 16″, weight 530 lbs., power consumption 600w.

SYLVANIA 9400: J. Magnetic tapes read in forward and reverse directions. K. Each disc unit has a capacity of 100 million characters. Size. Computer floor area 360 sq. ft., weight 21,825 lbs., power consumption 20 kva.

SYLVANIA AN/MYK-1(U) (MOBIDIC): K. Each disc file unit stores up to 100 million characters. Note. System designed to work in vans and under extreme temperatures. Height 68″, width 62″, depth 25″, weight 1,350 lbs.

SYLVANIA M-64: Note. System designed for operation in vans or ships and under extreme temperatures. Size. Height 38″, width 7″, depth 22″, weight 230 lbs., power consumption 1,281 w.

UNIVAC 1206: B. 9.6 microseconds is add time for repeat mode only. J. Magnetic types read in forward and reverse directions. K. Each flying head drum unit has a capacity of 3,932,160 BCD characters. N. 300 ch/sec reader available.

UNIVAC TARGET INTERCEPT: C, D. Storage consists of 10,000 words of "permanent" storage with a cycle time of 2.8 microseconds; 2,000 words of "variable" storage with a cycle time of 2.2 microseconds; 15 14-bit words of "reference" memory with a cycle time of 0.9 microseconds; and 48 words of "read-time" memory with a cycle time of 2.8 microseconds. Also, an overlapping instruction repertoire and simultaneous execution of arithmetic and non-arithmetic sequences permit concurrent operations.

☆Information supplied or confirmed by manufacturer but not reviewed in detail by publishers.
★Incomplete information compiled from various sources but not confirmed by manufacturer.

SECTION IV

FOREIGN COMPUTERS

England

Computer		Monthly Rental: Typical / Range	First Delivery: Month and Year	Processor Speed: Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	Internal Storage: Capacity in Words / Type	Logic: Word Size / Instr. Addresses	Magnetic Tape: Thousands of Characters per Second / Buffering	Maximum Units Attachable	Random Access File: Capacity / Access Time in Milliseconds	Peripheral Devices: Cards per Minute In — Out	Paper Tape Characters per Second In — Out	Printer Lines per Minute	Off-line Equipment	Other Features: Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter	Software: Algebraic Compiler	Business Compiler	Notes
AEI-1010	☆	$9,800 (6.7-42)	/60	18	8.5	4K core	44b[F] 1	10-120 MRWC	31	5.3M[L] 1000	400 100	1000 300	1000 3000	—	√	7	√	√	I/O	—	—	F. Two instructions per word. L. Carousel file unit with 5.3 million characters. Additional random access of 100,000-character drum storage with 10-millisecond access time. Also, 50,000 characters of additional core with 14-microsecond access time.
COMPUTER ENGINEERING 102	★	?	/61	?	10	20 core[E] 4-28K drum	32b ?	— —	—	√[K] ?	? ?	200 200	?	?	?	?	?	?	?	?	?	E. Twenty-word core working storage. Optional 4,000 words of core storage instead of drum available. K. Up to seven drums of 4,000 words each.
ELLIOTT 503		$8,300[A] (5.5-26)	4/63	7	3.5	8K core	39b 1	40[K] RWC	16	131K[L] 15	400 300	1000 100	1000	—	√	√[T]	—	√	I/O	6/62[X]	—	A. Prices include 12.5% import duty. This computer is commercially available in the United States. K. Tapes are IBM compatible. L. Slow-speed core storage (50-microsecond cycle time) in blocks of 16,000 words up to a maximum of eight blocks. T. Any storage location may be used as an index register. X. ALGOL '60, FORTRAN /63.
ELLIOTT 803		$3,750[A] (2.4-8.7)	3/61	576	24	4-8K core	39b 1	4.3 None[J]	4	— —	340 100	500 100	300	—	—	√[T]	—	√	O	6/62[X]	—	A. Prices include 12.5% import duty. This computer is commercially available in the United States. J. Search-compute only. T. Any storage location may be used as an index register. X. ALGOL '60.
EMI EMIDEC 1100	★	?	/60	?	?	1K core[E]	35b ?	? ?	16	320K[L]	400 ?	350 ?	600 3000	?	?	?	?	?	?	?	?	E. Sixty-four-word fast diode storage. L. Four drums of 16,000 words each available.
EMI EMIDEC 2400	★	?	/61	?	?	16K core[E]	34b ?	? MRWC	25	?	400 ?	350 ?	600 3000	?	?	?	?	?	?	?	?	E. Sixty-four-word fast diode storage.
ENGLISH ELECTRIC KDF 9	★	?	?	?	?	4-32K core[E]	48b ?	? ?	?	?	400 150	?	900	?	?	?	?	?	?	?	?	E. Fifteen-word fast storage.
ENGLISH ELECTRIC KDN 2	★	?	/61	?	?	.5-4K core	18b ?	— —	—	— —	? ?	20 20	?	?	?	?	?	?	O	?	?	
ENGLISH ELECTRIC KDP 10	★	?	/62	?	?	16-262K core	1a ?	33 ?	62	? ?	400 150	? ?	900	?	?	?	?	?	?	?	?	
FERRANTI ATLAS	★	?	/62	1.1[C]	.3 2 12000	8-262K rod[E] 16-262K core 25-100K drum	48b 1	90 MRWC	32	? ?	600 100	1000 300	3000[Q]	?	√	91	√	√	?	√	√	C. Overlapped core memory banks allow increased internal speed. E. Ferrite rod memory is non-destructive and designed for subroutine storage. Q. 600 lpm printer also available. 3000 lpm device is Xeronic printer.
FERRANTI ORION	★	?	?	64	?	1-16K core	48b 2-3	90 MRWC	64	—[L] 12	600 100	1000 300	1000	?	√[S]	64	√	√	I/O	?	?	L. Random access of 16,000-word drums available. S. Priority processing and automatic time-sharing of several simultaneous independent transfers between drums and core are possible.
FERRANTI SIRIUS	★	?	/60	250	4000	1-10K delay	10d ?	? ?	?	? ?	? ?	300 60	?	?	?	?	?	?	?	?	?	
ICT 1301	☆	$3,500[A]	/61	21	?	2K core	12d ?	?[H] RW	8	96K[L] ?	600 100	? ?	600	?	?	?	?	?	?	?	?	A. Excludes cost of magnetic tape units. H. Ampex FR300 or 400 tape units available. L. One to eight drums, each with a capacity of 12,000 words, may be attached.
ICT 1500	☆	$5,200 (3.3-)	/61	189	7	10-40K core	1a 2	33-66 RC, WC, RW	12	176M[L] 100	600 110	1000 100	1000	?	?	?	?	?	I/O	?	√[Y]	L. Up to two disc file units, each with a capacity of 22, 44, 66, or 88 million alphanumeric characters, are available. Up to six record files, with a capacity of 4.6 million alphanumeric characters, are also available. Y. COBOL. Note. This appears to be a version of the RCA 301.
LEO III	☆	$11,200 (8.4-19.6)	/62	34[C]	13.5	4-32K core	40b[F] 1[G]	22-90 MRWC	32	— —	600 100	1000 110	1000	—	√	12	√	√	O	√[X]	√[Y]	C. Built-in mixed-radix arithmetic and data-handling operations. F. Multi-programming of up to 13 programs, each fully protected by tag reservation. G. Instructions stored two per word. X, Y. CLEO.

☆Information supplied or confirmed by manufacturer but not reviewed in detail by publisher.
★Incomplete information compiled from various sources but not confirmed by manufacturer.

System		Monthly Rental Typical Range	First Delivery Month and Year	Processor Speed Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	Internal Storage Capacity in Words	Type	Logic Word Size Instr. Addresses	Magnetic Tape Thousands of Characters per Second Buffering	Maximum Units Attachable	Random Access File Capacity Access Time in Milliseconds	Peripheral Devices Cards per Minute In—Out	Paper Tape Characters per Second In—Out	Printer Lines per Minute	Off-line Equipment	Other Features Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter	Software Algebraic Compiler	Business Compiler
LEO III/F	☆	$14,000 (11.2-33.6)	/64	4[C] 12	2[D] 6	4-32K	core	40b[F] 1[G]	22-90 MRWC	32	— —	600 100	1000 110	1000	—	√	12	√	√	○	√[X]	√[Y]

C. Built-in mixed-radix arithmetic and data-handling operations. D. Two-microsecond core memory available as option. F. Multi-programming of up to 13 programs, each fully protected by tag reservation. G. Instructions stored two per word. X, Y. CLEO.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
STC STANTEC ZEBRA	★	$2,100	/58	?	?	8K 12	drum fast	33b ?	? ?	32	? ?	? ?	200 10	?	?	?	?	?	?	?	?	?

France

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
BULL GAMMA 10	☆	$2,000 (1.6-24)	/63	200	7	1-4K	core	1a 2	— —	—	— —	300 300	— —	300 600	—	√	—	√	—	—	—	—
BULL GAMMA 30		$7,000 (4-16)	2/62	217	7	10-40K	core	1a 2	10-66 RC, WC, RW	14	176M[L] 100	600 300	1000 100	1000	—	—	—	√	—	I/O	—	/63[Y]

L. Each disc unit has a capacity of 4.6 million characters. Up to six disc units, with an average access time of 4.25 seconds, may be attached. Additionally, up to two Bryant disc file units, each of four modules of 22, 44, 66, or 88 million alphanumeric characters, are available. Y. COBOL. Note. Central processor is a version of the RCA 301.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
BULL GAMMA 30S		$8,800 (6.3-18)	/63	70	7	20-40K	core	1a 2	10-66 RWC	14	177M[L] 100	600 300	1000 100	1000	—	—	3	√	√	I/O	/63[X]	/63[Y]

L. Up to two Bryant disc file units, each of four modules of 22, 44, 66, or 88 million alphanumeric characters, are available; or up to six record files of 4.6 million characters each are also available. X. ALGOL. Y. COBOL. Note. Central processor is a version of the RCA 301 Model 354, 355.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
BULL GAMMA 60	☆	$32,000 (28-68)	/60	200	10	8-32K	core	24b 1-3	21 MRWC	48	512K[L] 10	300 300	800	300[Q]	—	√	—	√	√	I/O	/63[X]	/63[Y]

L. Random access of up to four drum units each with 153,600 characters. Q. Double-feed type line printer. X. ALGOL. Y. COBOL.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
BULL GAMMA MDE (300)	☆	$4,200 (3.7-4.6)	10/59	865	173	64	delay[E]	12d 1	21 RWC	8	196K[L] 11	300 300	— —	300	—	√	—	—	√	—	/57	—

E. Additional delay line storage available. L. Drum storage.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
CIT CITAC 210B	☆	$2,000 (1.6)	/61	150	20	4-32K	core	21b[F] 1	— —	—	72M[L] 100	— —	500[P] 40	—	—	√	7	—	—	—	—[X]	—

F. Two words per instruction. L. Disc units in increments of three million alphanumeric characters. P. Analog input and output systems available. X. CITAC AUTOCODE.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
SEREL 1001	☆	$3,000 (2-6)	/60	20	6	4-32K	core	20b 1	√ MRWC	√	700M[L] 10	√[N] √	√[P] √	√[Q]	—	√	2	√	—	I/O	—	—

L. Random access drums with capacity chosen according to specific requirements. N, P, Q. Any manufactured equipment adaptable.

Germany (West)

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
TELEFUNKEN TR4	★	?	?	11	6	4-32K	core	48b ?	37.5 MRWC	64	? ?	800[M] 250	1000[N] 300	?	?	?	?	?	?	?	?	?

M. 700 cpm reader and 100 cpm punch available. N. 500 cps reader and 150 cps punch available.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
TELEFUNKEN TR5	★	?	?	?	?	64K	core	1d ?	35 ?	8	? ?	800 250	1000 300	?	?	—	252	—	—	—	?	?
ZUSE Z23	☆	$3,700 (2-5.5)	/60	300	18 5000	.2-8K 8K	core drum	40b 2	15 RC, WC	4	— —	120 120	300 150	300	—	√	240	√	—	I/O	—	—
ZUSE Z25	☆	$1,300 (.7-5.6)	—	125	10	1-18K	core[E]	18b 1	15-100 None	8[K]	√[L] 10	200 100	300 150	300	—	√	√	√	—	I/O	—	—

E. Fixed-core storage of 1,000 to 2,000 words is included. K. Additional tape units available as option. L. Random access drum files available.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
ZUSE Z31	☆	$5,500 (2.1-15.8)	/62	420	420	1-10K	core[E]	11d 1	18[H] None	12[K]	2.6M[L] 20	800 300	300 150	1000	—	—	10	√	—	I/O	—	—

E. Fixed-core storage of 2,600 words is included. H. Transfer rate is for four-bit decimal characters. K. Additional tape units are available as option. L. Additional drums of 66,000 BCD characters each are available.

Japan

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
FUJI FACOM 222	★	?	10/60	160	?	2-10K	core	7d 1	15 ?	10	700K[L] ?	500 200	400 170	300 500	?	?	99	?	—	—	?	?

L. Random access is drum.

System		Monthly Rental	First Delivery	Add Time	Cycle Time	Capacity	Type	Word Size / Instr. Addr.	Mag. Tape / Buffering	Max Units	Random Access / Access Time	Cards	Paper Tape	Printer	Off-line	Prog. Interrupt	Index Reg.	Indirect Addr.	Floating-point	Console Typewriter	Algebraic Compiler	Business Compiler
FUJI FACOM 241	★	?	6/61	250	?	1-4K	core	7d 1	15 ?	10	? ?	500 200	400 170	300 500	?	?	4	?	—	—	?	?

☆Information supplied or confirmed by manufacturer but not reviewed in detail by publisher.
★Incomplete information compiled from various sources but not confirmed by manufacturer.

	Monthly Rental Typical Range	*First Delivery* Month and Year	*Processor Speed* Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	*Internal Storage* Capacity in Words - Type	*Logic* Word Size / Instr. Addresses	*Magnetic Tape* Thousands of Characters per Second / Buffering	Maximum Units Attachable	*Random Access File* Capacity / Access Time in Milliseconds	*Peripheral Devices* Cards per Minute In—Out	Paper Tape Characters per Second In—Out	Printer Lines per Minute	Off-line Equipment	*Other Features* Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter	*Software* Algebraic Compiler	Business Compiler
HITACHI HITAC 103 ★	?	/60	400	10	1K core 8K drum	48b 1	2 ?	8	? ?	—	200 8	300	?	?	0	?	√	I/O	?	?
HITACHI HITAC 201 ★	?	6/61	4000	3300	9K core 4K drum	12d 1	.2 ?	4	48K[L] ?	—	200 6	120	?	?	8	?	—	I/O	?	?
	L. Random access is drum.																			
HITACHI HITAC 301 ★	?	5/59	3000	3000	.2K core 2K drum	12d 1	2 ?	10	108K[L] ?	200 100	200 8	300	?	?	0	?	√	O	?	?
	L. Random access is drum.																			
HITACHI HITAC 3010 ★	?	?	300	70	40K core 4K disc	? 2	1.3 ?	?	? ?	? ?	1000 100	1000	?	?	?	?	?	?	?	?
MATSUSHITA MADIC IIA ☆	$1,000 (1.2-1.7)	/61	1000	11	4K drum	33b 1[G]	.4 None	2	— —	— —	200[P] 70	—	—	—	2	—	√	I/O	/63[X]	—
	G. The last address indicates the address of the next instruction. P. 1,000-conversion-per-second analog to digital converter available. X. ALGOL.																			
MATSUSHITA MADIC IIIB ☆	$5,000 (2-7.5)	/63	540	10	4K core	36b 1	15 RWC	8[K]	— —	400 100	400[P] 100	500	—	√	64	√	√	I/O	/63[X]	—
	K. Tapes are IBM compatible. P. 1,000-conversion-per-second analog to digital converter available. X. ALGOL.																			
MITSUBISHI MELCOM 1101F ★	?	3/60	160	7800	4K drum	32b 2	.025 ?	?	? ?	? ?	600 20	—	?	?	7	?	√	I/O	?	?
NIPPON ELECTRIC NEAC 2203 ★	?	5/59	3300	700 8300 20000	.2K core 2K drum 100K drum	12d 1	8 ?	10	? ?	200 100	200 50	200 350	—	?	3	?	√	—	?	?
NIPPON ELECTRIC NEAC 2204 ★	?	12/61	1500	13000	.1K core 3K drum	12d 3	? ?	6	? ?	200 100	200 50	200	—	?	14	?	—	—	?	?
NIPPON ELECTRIC NEAC 2205 ★	?	3/61	2900	12000	3-6K drum	10d 1	.4 ?	4	— —	200 100	100 50	200	?	?	3	?	—	—	?	?
NIPPON ELECTRIC NEAC 2206 ★	?	3/62	100	50	4-10K core 120K drum	12d 1	90 —	20	? ?	600 250	600 50	900	?	?	54	?	√	—	?	?
OKI ELECTRIC 5090D ☆	?	10/61	400	10	4K core	12d 1	10 ?	6	— —	500 150	200 65	500	—	—	1	—	√	I/O	√[X]	—
	X. OKI PAL, OKI ART.																			
TOSHIBA TOSBAC 3100 ★	?	9/61	420	7000	5-10K drum	12d ?	6 RW	10	—	400 100	400 70	200	?	?	3	?	√	?	?	?
TOSHIBA TOSBAC 3200 ★	?	?	240	?	1-5K drum	6d ?	—	—	—	—	400 10	?	?	?	?	?	?	?	?	?
TOSHIBA TOSBAC 4200 ★	?	12/61	420	200	2-40K core	1d 2	6 ?	10	— —	400 100	400 70	200	?	?	6	?	—	I/O	?	?

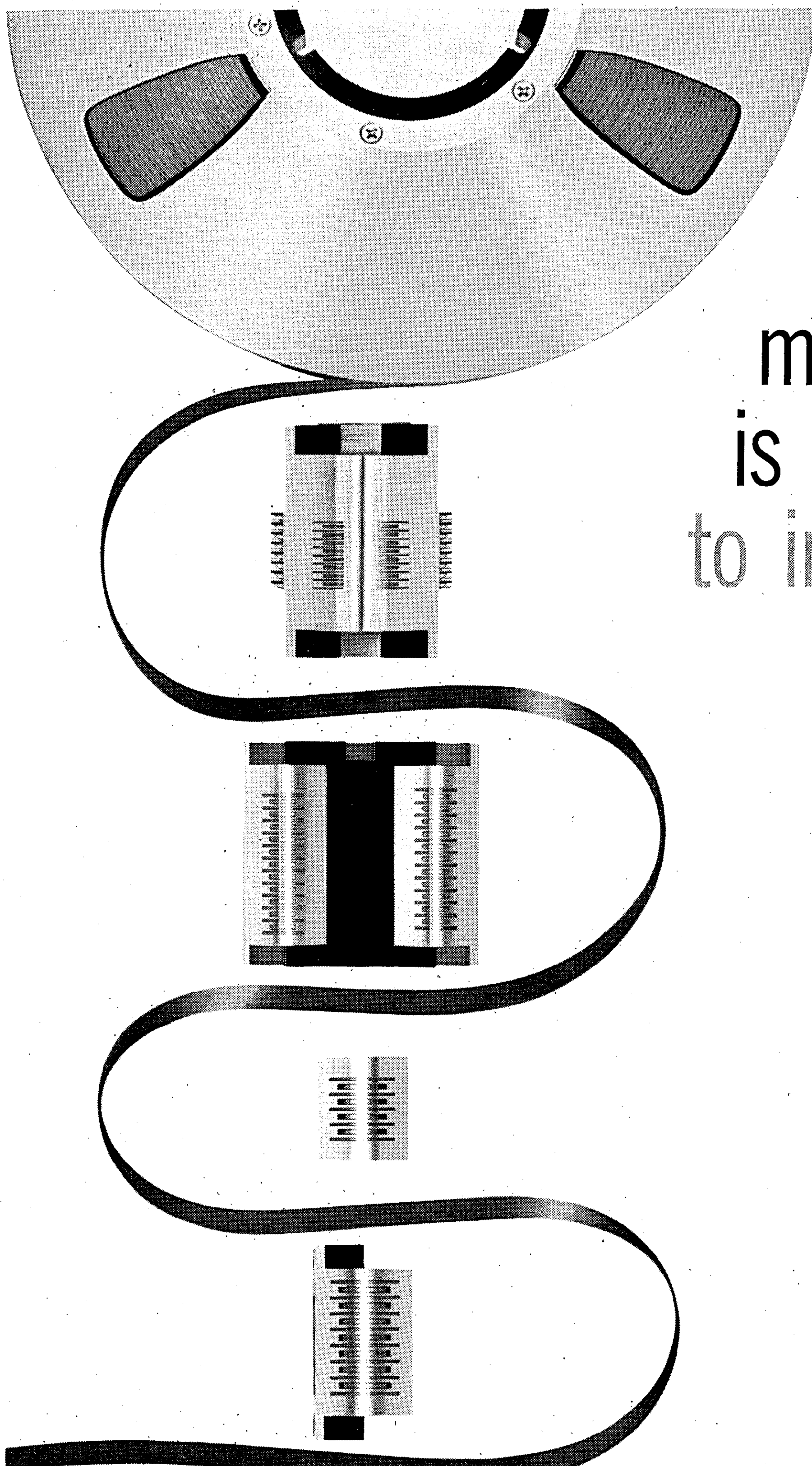
Sweden

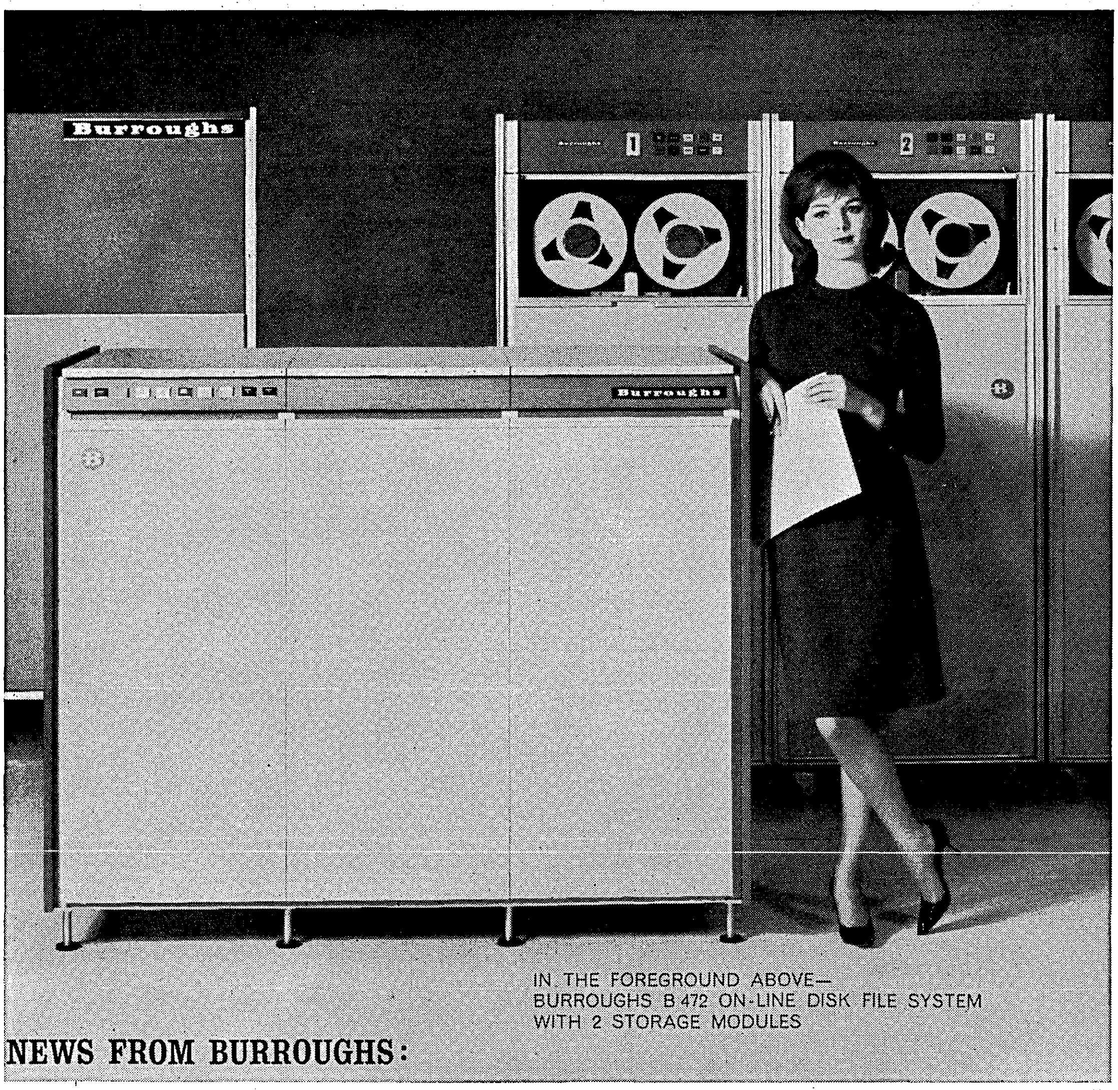
	Monthly Rental Typical Range	*First Delivery* Month and Year	*Processor Speed* Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	*Internal Storage* Capacity in Words - Type	*Logic* Word Size / Instr. Addresses	*Magnetic Tape* Thousands of Characters per Second / Buffering	Maximum Units Attachable	*Random Access File* Capacity / Access Time in Milliseconds	*Peripheral Devices* Cards per Minute In—Out	Paper Tape Characters per Second In—Out	Printer Lines per Minute	Off-line Equipment	*Other Features* Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter	*Software* Algebraic Compiler	Business Compiler
SAAB D21	$10,000 (4-20)	/62	9.6	4.8	65K core	24b 1	48 MRWC	256	—[L] —	1200 120	1000[P] 150	1000	—	√	√	—	√	I/O	—[X]	—[Y]
	L. Various random access stores can be connected. P. Analog conversion equipment available. X. DAC 11/62, ALGOL 9/63, GENIOUS 12/63. Y. DAC 11/62, COBOL 9/64, GENIOUS 12/63, SORT GENERATOR 6/63.																			

The Netherlands

	Monthly Rental Typical Range	*First Delivery* Month and Year	*Processor Speed* Complete Add Time in Microseconds	Storage Cycle Time in Microseconds	*Internal Storage* Capacity in Words - Type	*Logic* Word Size / Instr. Addresses	*Magnetic Tape* Thousands of Characters per Second / Buffering	Maximum Units Attachable	*Random Access File* Capacity / Access Time in Milliseconds	*Peripheral Devices* Cards per Minute In—Out	Paper Tape Characters per Second In—Out	Printer Lines per Minute	Off-line Equipment	*Other Features* Program Interrupt	Index Registers	Indirect Addressing	Floating-point Arith.	Console Typewriter	*Software* Algebraic Compiler	Business Compiler
ELECTROLOGICA X1	$8,300 (2.7-26)	/58	64	16	.5-32K core	27b 1	30 MRWC	16	31M[L] 15	700 120	1000 300	600	same	√	1	√	√	I/O	6/60[X]	—
	L. Up to eight drums, each with a capacity of 524,288 words. X. ZEBRA, ALGOL '60.																			

☆Information supplied or confirmed by manufacturer but not reviewed in detail by publisher.
★Incomplete information compiled from various sources but not confirmed by manufacturer.





Burroughs—TM

A SIGNIFICANT advance in random access DATA PROCESSING

Burroughs introduces a new type of random access magnetic disk file which—with its data communications and inquiry devices—makes practical real time applications possible.

In addition, it allows significant advances in memory capacity, reliability and economy.

The new file eliminates hydraulic or mechanical read/write head positioners and gives each track of information an electronic head of its own. This head-per-track principle allows an average access speed of 20 milliseconds—five times faster than the best of its type now available.

Instant availability of data eliminates the need for artificial file organization techniques—and the programing and use problems which they impose.

Read/write head assemblies self-adjust to maintain a uniform, precision-controlled air gap between head and disk. Retraction is automatic as disks decelerate. Result: a completely reliable system.

A totally new disk-plating technique allows high packing density. Per-module capacity is 9.6 million alphanumeric characters. Thus data storage can be "built up" in small economical increments to a maximum of nearly one billion characters. Modules can be added to your system as needed.

The new disk file greatly extends the application range of our B 5000 and B 200 series, while giving both systems additional immunity to obsolescence. For additional details, write us at Detroit 32, Michigan. Remember: our business is computation for all businesses of all sizes, everywhere.

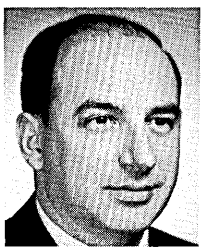
Burroughs Corporation

CIRCLE 18 ON READER CARD

prospects & problems

AUTOMATIC PROGRAM TRANSLATION

by ASCHER OPLER, Computer Usage Corp., New York, N.Y.



After a long period of disrepute, translation between computer codes has emerged as a promising technique. Four or five years ago, any discussion of automatic translation tended to be entirely negative. Today translation is one of the most discussed topics in the computer world; and considerable optimism is expressed concerning its future.

evidence of activity

In the past four years a number of projects have been directed toward translation. Most of the major manufacturers have been involved in developmental translator work. Several universities are actively studying this field. Last year two papers [8, 9] on the subject were presented at international conferences, [7, 8] and several unpublished manuscripts describing translation techniques were widely circulated. Activity has also increased in related program areas, including decompilers and special translators which operate in conjunction with source language compilers. Table 1 gives examples of translation pairs that have been considered, studied or implemented.

translation's growing importance

The increasing pace of activity in this field is a response to a definite need. In this era of repeated replacement of computing equipment, the task of converting programs

Partial List of Actual and Proposed Translators
Table 1

	FROM	TO	REFERENCE
I	PROGRAMMING LANGUAGES		
	FORTRAN II	FORTRAN IV	1
	COBOL 60	COBOL 61	2
	COMMERCIAL TRANSLATOR	COBOL 61	3
	FORTRAN	ALGOL	3
	AIMACO	COBOL 61	3
II	SYMBOLIC PROGRAMMING		
	SOAP (IBM 650)	USS — 80	5
	SAP (IBM 704)	CODAP (CDC 1604)	4
	FAP (IBM 7090)	COMPASS (CDC 3600)	4
	FAP (IBM 7090)	SLEUTH (UNIVAC 1107)	3
	FAP (IBM 7090)	TAC (Philco 212)	3
	AUTOCODER (IBM 1401)	AUTOCODER (IBM 1410)	3
	AUTOCODER (IBM 705)	AUTOCODER (IBM 1410)	6
III	PROGRAM CODE		
	Ferranti MERCURY	Ferranti ORION	7
	IBM 705	Honeywell 800	4
	IBM 705	IBM 7074	8
	IBM 705	RCA 501	4
IV	COMPILER OUTPUT		
	AIMACO 1105	UNIVAC -1107	4
	FORTRAN 7090	IBM 7030	4
V	DECOMPILERS and DISASSEMBLERS		
	CDC 1604	NELIAC	9
	IBM 704	SAP	10

from one machine to another looms as a major problem. As the magnitude of the stockpile of programs requiring conversion increases, it becomes quite apparent that there are not enough programmers in the world to do the reprogramming.

One obvious solution to the transition problem is the use of standard common languages (e.g., COBOL, FORTRAN, ALGOL) which are sufficiently machine independent that source programs can be easily converted to run on the new computer. However, only a modest portion of applications have been written in common languages. Even where these source languages have been used, their insufficiencies have caused a fairly widespread use of either interspersed hand coding or frequent calls to special machine-language subroutines.

Although the increased pace of computer replacement is the principal factor in spurring on translation development, translation would also be useful in converting programs from one contemporary species of computer to another. Furthermore, translation also offers a potential method for converting the output of an efficient compiler to coding for more than one object machine.

An entirely different reason for developing translators arises in the *logical design* of new computers. The requirement for compatibility with predecessors handicaps the design of each new machine. One way in which designers could be free to develop radically new designs (which might not be hardware compatible with predecessors) would be to develop new machines and translators concurrently so that, when the new computer appears, translation programs would be available to convert existing programs to run on the new machine.

types of translation

Before translation can be discussed in detail, some basic definitions must be given. By translation we mean a process which converts a program in one "language" to a program in another "language" in a single operation. By analogy to mechanical language translation, we will call these source language and target language.

The word "language" has been used in many contexts in the computing sciences. For this discussion it is sufficient to distinguish high-level machine-independent source languages, intermediate level symbolic assembly languages and hardware level machine languages. If we consider three levels of language and two computers (source computer and target computer) we see that there are many possible processes which may be named "translation". Figure 2 shows six significant translation routes.

1. **Translation between source languages.** An excellent example of successful translation between closely related languages is the SIFT program described in a recent issue[1] of DATAMATION.
2. **Translation between the symbolic assembly languages.**
3. **Translation between the machine languages.**
4. **Compilation** of a source language to an object language and/or **assembly,** of a symbolic language to machine language. These are the best known types of translation. Indeed, many of these processors have the word "translation" in their name or acronym.
5. **Decompilation** from machine language to source language and **disassembly** from machine language to symbolic language. These have been reported[9,10] and used for some time in limited areas.
6. **Compilation - Translation - Assembly.** Many source language compilers produce an intermediate symbolic assembly language as a stage of their processing. This assembly language is susceptible to translation using the methods developed for Type 2 above. In effect, compilation becomes a three-stage process—compilation to symbolic language on the source computer, translation to the symbolic language of the target computer and assembly on the target computer.

The remainder of this article will concentrate on translation Types 2 and 3.

the "impossibility" of translation

Many arguments against translation have been put forth. The following section lists just a few of these.

1. The problem of translation can be approached from the viewpoint of computability theory. For example, a recent paper by Ginsberg and Rose[11] presents a discussion on the recursive undecidability of whether a program could be constructed to convert rigorously one computer code to another. An examination of the internal codes of two non-identical computers from the viewpoint of Turing machine theory shows the improbability that successive sequential states of two computers can be made equivalent. In general, those concerned with the theoretical basis of this problem believe it either unsolvable or undecidable.

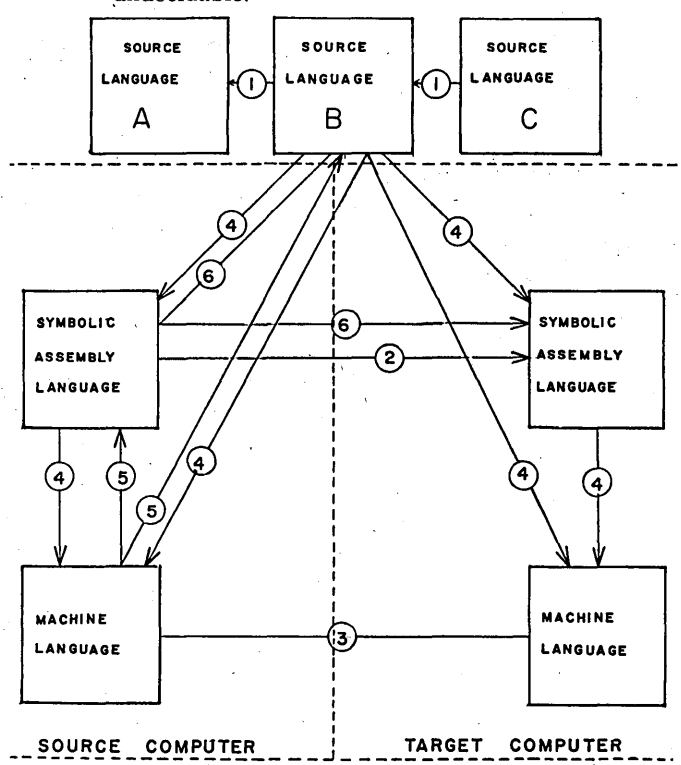


Fig 2· SOME TRANSLATION PATHS

2. In von Neumann machines, which currently dominate the field, the internal memory contains bits of information whose function is undetermined until they are fetched to a register for interpretation and use. The information, in a typical modern machine, might be fetched to an instruction register, to several instruction registers, to one or more arithmetic registers, to one or more input/output control registers or to any number of registers with special functions. Until the bits are fetched, it is not clear what their function is. As a matter of fact, these bits may

have no function, multiple functions (static) and multiple functions (dynamic).

Given a program as the total complex of information it can readily be seen that the *functional information is not available*. However, in order to translate properly to equivalent activity in the target computer, the function(s) of every bit must be known. Thus, proper translation in a von Neumann machine implies complete knowledge of programmer intent as embodied in the functional nature of his coding.

3. A program rarely exists in isolation. Most programs read data and operate on it. In commercial data processing, as well as in many scientific applications, the data plays a critical role in determining the selection of logical paths to be followed by the program and the sequence of such actions. A more complex type of "data" exists in the form of control cards which are used to specialize a general program for a specific purpose.

   This data-program interaction increases the difficulty of translation. The target program resulting from translation must respond to data in an equivalent, if not identical, manner. Without examination of the data, analysis of the program can rarely lead to complete information.

4. Every computer has its idiosyncrasies, and programmers learn to cope with these. Even the most straightforward program for one computer contains many commands written to handle special hardware features of this computer. Commands that are so used pose extreme difficulty for the translator. For example, a command to "End around shift a two's complement" is rather difficult to translate effectively on an alphanumeric character-oriented computer lacking arithmetic registers.

   Added to this complexity is the use of "programmer tricks" in which the virtuoso demonstrates his mastery over the subtleties of the equipment. Now the translation difficulties become compounded.

5. All of the manipulations performed in a computer are not purely logical ones implicit in the circuitry. In particular, one must consider time delays, action of the mechanical components and activity centered around interrupts, multiprocessing, etc.

the "feasibility" of translation

If translation appears technically impossible, it does not, paradoxically, mean that it is not technically feasible. The apparent paradox is resolved when we abandon the goal of completely automatic, perfect translation that causes the target computer to carry out all manipulations programmed for the source computer. If this were the goal, translation could be readily dismissed. However, the goal generally sought is a processor which will produce translations which will cause *approximately equivalent operations on the target computer which may be completed and polished by skilled programmers*. Few of the schemes brought forth have aimed at fully automatic translation, and none have claimed perfect translation as their goal.

The surmounting of many of the difficulties concerned with getting a second computer to perform equivalent operations to a first computer has been achieved repeatedly. The existence of dozens of practical *simulators* demonstrates the feasibility of transforming one computer's manipulations to another.

Confidence in the practicality of translation can be gained by regarding it, on the one hand, as a technique for using the knowledge gained as a program is simulated in order to prevent subsequent simulations from naively repeating most of the instruction interpretation and, on the other hand, as a reprogramming aid which examines symbolic assembly programs written for the source computer looking for statements that may be painlessly translated to the target computer.

It should be emphasized that the existing programs which are candidates for translation are primarily computer applications in the scientific area and, to a lesser extent, in the data processing area. No one has seriously advocated that translation operate on compilers, monitors, loaders, programs that play "Jingle Bells", etc. They do aim, in general, at application programs of moderate length and minimum complexity.

symbolic translation versus code translation

There is a great difference between translators that operate directly on programs written in symbolic assembly language and translators that operate directly on the machine language programs. The former operate on the principle that the symbolic language carries the information required for translation while the latter works on the principle that only the machine code itself is valid enough to serve as a base for translation. The former is primarily applicable to short programs that are relatively free of modification, data dependency and programmer tricks. It also assumes some similarity in logical design between the source and target computers. The code translators make no such assumptions and can operate between machines of greatly disparate logical structures.

The author has available several short program sections which illustrate the problems of symbolic and code translation.

symbolic translation

Here the translation is definitely at a programming level. The intent is not to reproduce the detailed manipulation carried out by the source computer, but rather to translate the programmer's intent. The assumption is that, if translation is to be carried out, the original programmer is not available to reprogram his work for the target computer. Reading his symbolic program into a translator is a poor but, perhaps, adequate substitute. The symbolic locations representing data, commands, work area and tables are frequently clear and adequate for translation as written.

Still there are many difficult problems in this area. Since one hundred per cent translation is not the aim of symbolic translators, the most important function they perform is to *separate* the lines of symbolic coding that are difficult to translate from the lines that are easy to translate. A clearing of a register, an unconditional branch, a testing of the sign of a register are relatively easy to translate. Input/output operations, normalization, block transfer and bit addressing are generally difficult to translate symbolically. Instructions of the latter group will typically translate to a subroutine call (the subroutine is similar to that used in interpretive simulation). The success of line-by-line translation hinges on the similarity of the word structure of the two machines. Referring to Table 1, it may be seen that symbolic pairs are listed for ten-digit decimal machines; character addressable data processing machines; and floating binary, indexed scientific computers. Only the presence of considerable similarity permits symbolic translation to succeed.

An entirely different problem arises with the mapping of the whole program from one core storage to the other. Unless the two machines have similar storage size and command packing, considerable difficulty will arise. Examples of such complication are the mapping of one-command-per-word machines to two-command-per-word machines and in the mapping of two-address to one-address computers. Symbolic programs which use relative addressing, symbols relative to "the current contents of the location counter," and symbolic instructions to modify other symbolic instructions all harass the translator designer.

Of great assistance in symbolic translator design are memory maps, symbol tables, interspersion of original Remarks cards and translator-generated Remarks cards in the target symbolic program. Since complete translation will rarely be obtained, it is most important to provide guidelines for those whose task it will be to patch and debug the translated program.

code translation

These processors aim at a much wider segment of programs than symbolic programs and hope to translate a much higher percentage of the source program to the target program.

One such translator designed in 1961 by the author and co-workers (at IBM and CUC)[8] has been described in considerable detail. The only information required by this translator is the program operating instructions and a representative data-set. The translator first carries out a process closely related to simulation of the source program using the test data-set. While the simulated execution takes place, a detailed recording of the status of the principal registers is made. (This has been likened to photographing the console with a fast movie camera.) Following the completion of the execution, the translator analyzes the recording, building many cross reference tables and studying, among others, the input/output usage, program modification, functional use of storage and logical flow of the program. The second phase ends with the construction of a special symbol table whose symbols are synthesized to contain functional and logical information concerning each significant word in the program. (This has been likened to the detailed frame-by-frame analysis of the film previously taken.) The translator now proceeds to separate the original program into blocks according to functional complexity. Those blocks which may be readily translated are so processed. Those for which insufficient information was obtained or the analysis indicated to be excessively troublesome are flagged for simulation at object time. Both input/output and modification blocks are given special treatment. Finally the translator takes all segments of the target program, assigns proper memory locations and ties all parts together for execution on the target machine.

While this translator was not carried to completion, the design and extensive analysis performed revealed that the method was technically feasible, that the object programs would run considerably faster than simulated ones, but slower than hand-programmed versions of the source programs. The presence of simulation subroutines in the object program required that the target computer be substantially larger than the source computer.

J. H. Gunn[7] has described a code translator in which the source computer is a Ferranti MERCURY and the target computer a Ferranti ORION. A considerable portion of Gunn's analysis is devoted to the remapping of quite dissimilar storage media.

Code translators require many times the implementation effort of symbolic translators. Their advantage lies in giving a more fully automatic translation.

target program efficiency

Translation is being developed because of practical requirements occurring in the transition from one computer to another. Interpretive simulation of the source computer on the target computer has been a successful method for avoiding the reprogramming of the numerous small programs which occupy a small fraction of the total workload of a computer installation. Reprogramming is the most desirable transition technique, but reprogrammers are scarcer than programmers and can only be used on the most critical programs requiring the larger percentage of the installation's workload. Recompilation will obviously suffice to convert those programs written primarily in common source languages.

In order for translation to carry its load, a translator must produce a rather efficient target program. Furthermore, this must be done with minimum intervention on the part of skilled programmers, the translated programs must be readily maintained and the target program used without too many data file interconversions.

Symbolic translation will generally give the most efficient object program because human intervention is relied upon heavily; and therefore human ingenuity can be applied. By contrast, the efficiency of programs produced by code translation is *extremely variable*. The object efficiency depends strongly on the nature of the program undergoing translation. Thus, the same code translator system may produce a tenfold drop in target program efficiency between processing straightforward, clean programs and processing complex, tightly-programmed, highly modified, data-dependent programs. Code translators, as a matter of fact, will turn out to be excellent tools for measuring programming style and quality.

the need: less guessing and more translating

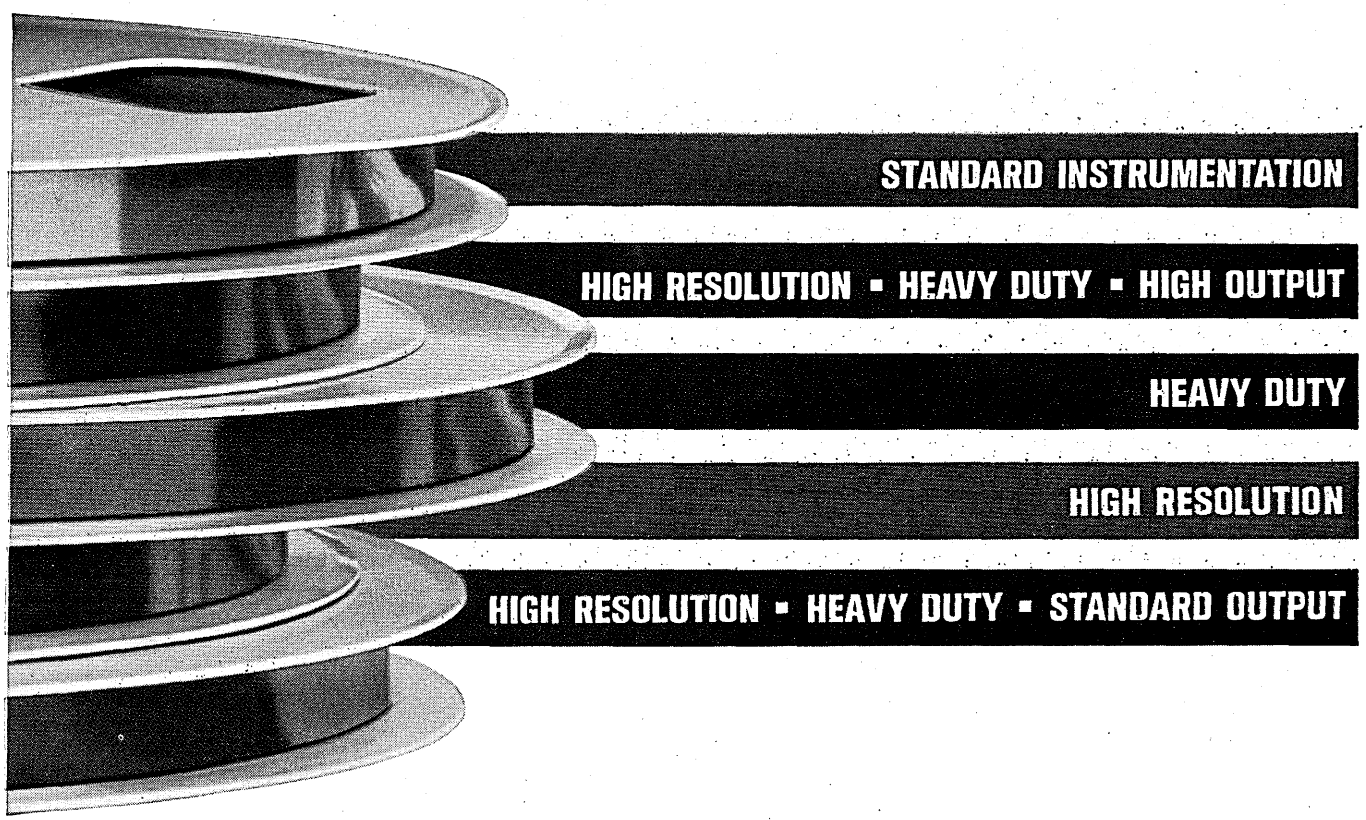
Translation has been very slow in coming. Experience has been pitifully small, while conjecture and interest remain very high. The next few years will definitely see the introduction of a number of translators. The considerable investment of energy, brains and money will at least produce enough working translators that we may begin to evaluate their usefulness. It is only in day-to-day effective use that translation can take its place with other recognized automatic programming techniques. ■

REFERENCES

1 DATAMATION, March 1963.
2 Reported in *Communications* ACM, 6, (1963) #2, Page 56.
3 Believed to have been seriously considered by one or more organizations—not necessarily computer manufacturers.
4 Believed to have been the object of a study or implementation project.
5 "SOAP II to UNIVAC Solid State 80 Translator", Staff, University of Arizona Numerical Analysis Laboratory, Tucson, Arizona.
6 Computer Sciences Corporation, 660 Madison Avenue, New York 21, N.Y.
7 Gunn, "Problems in Program Interchangeability", *Symbolic Languages in Data Processing*, 777-790, Gordon and Beach Science Publishers, New York, 1962.
8 Opler, Farbman, Heit, King, O'Connor, Goldfinger, Landow, Ogle and Slesinger, "Automatic Translation of Programs from One Computer to Another", *Proceedings of IFIP Congress 62*, North Holland Publishing Company, Amsterdam, 1963.
9 Halstead, "Machine-Independent Computer Programming", Chapter 11, Spartan Books, Washington, D.C., 1962.
10 For example, see PK DSMB for the IBM 704. This is SHARE Distribution 158, September 5, 1956.
11 Ginsberg and Rose, *Journal ACM*, 10 (1963), 29-47.

Your instrumentation tape supplier should be a complete source.

Soundcraft is!

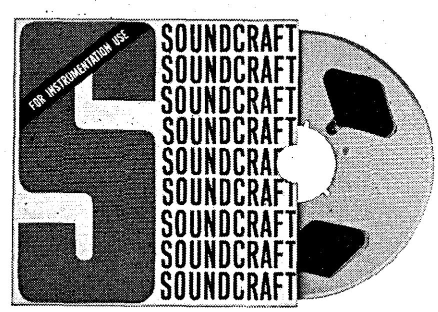


With Soundcraft Instrumentation Tapes, you can meet every data recording need. Sixteen combinations of coatings and base materials are available to fulfill the recording time, resolution, wear-life and output characteristics required by your data recording equipment.

Common to each instrumentation tape is Soundcraft's technical competency. This is demonstrated by the unique oxides and binder systems which are engineered to provide superior physical and magnetic qualities for all recording applications. The Soundcraft plant is one of the newest and best-equipped in the industry—where advanced manufacturing processes and control methods produce superior instrumentation tapes. Reliable data recording is assured.

In military and space facilities . . . among industry leaders . . . Soundcraft Instrumentation Tapes are meeting the most exacting performance standards. They can meet yours, too.

Soundcraft Instrumentation Tapes are available from stock for immediate delivery. Write for literature.



REEVES **SOUNDCRAFT** CORP.

Main Office: Great Pasture Rd., Danbury, Conn. • New York: 10 E. 52nd St. • Chicago: 28 E. Jackson Blvd. • Los Angeles: 342 N. LaBrea • Canadian Reps.: Toronto • Vancouver

CIRCLE 19 ON READER CARD

(*Advertisement*)

New generation of Shaped-Beam Tubes makes

by: A. H. Wisdom,
Manager of Research and Engineering for Data Products

CHARACTRON® Shaped-Beam Tubes produced nearly 10 years ago, many of which are still being used in the display consoles of the SAGE program, have achieved 20,000 hours or more of reliable performance. Today's CHARACTRON Tube represents a new generation of development, offering dozens of major improvements over the original tube. The principle, however, remains essentially the same.

HOW IT WORKS

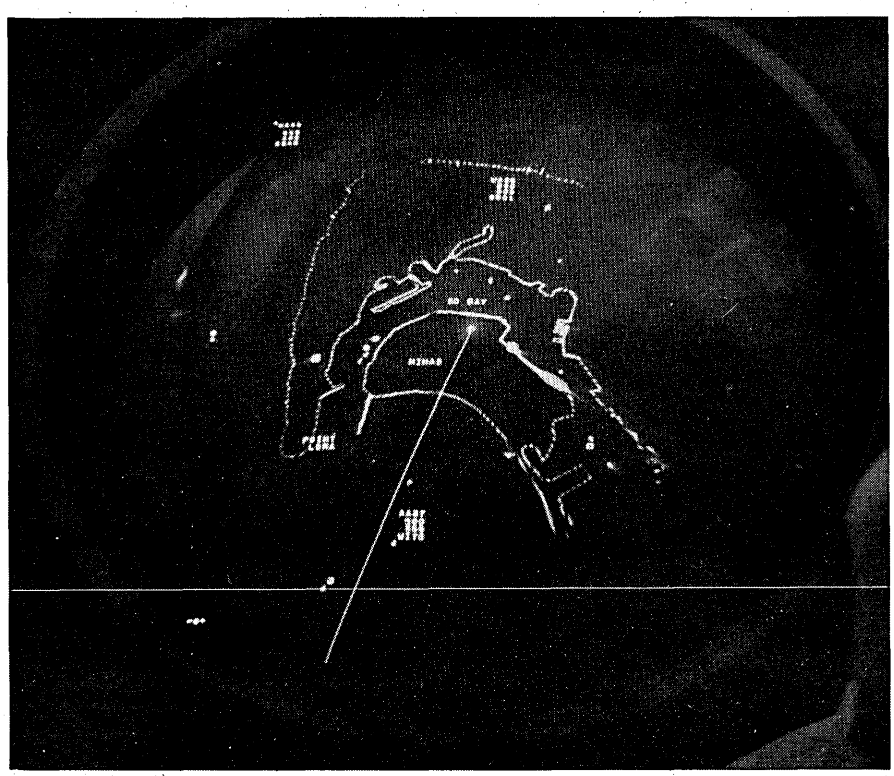
Heart of the CHARACTRON Shaped-Beam Tube is the stencil-like matrix, a thin disc with alphanumeric and symbolic characters etched through it. This matrix is placed within the neck of the tube, in front of an electron gun. The stream of electrons emitted from the gun is extruded through a selected character in the matrix. When the beam impinges on the phosphor-coated face of the tube, the character is reproduced.

The standard matrix carries 64 characters. However, matrices have been made with 88, 128, and 132 characters. Coupled with new variable character size capabilities, the CHARACTRON Tube offers a wide latitude in symbol generation. The beam is passed through one of the characters by applying the proper voltage to the selection plates. Electrostatic reference plates and/or magnetic deflection are then used to position the beam at any tube face location. In more compact tubes, the entire matrix is flooded with electrons generating a complete array of characters, while only the desired character is allowed to pass through a small masking aperture. A small diameter beam can be used to display data from analog inputs simultaneously with the characters.

NEW GENERATION OF TUBES

Today's CHARACTRON Tube is *not* the same tube built ten years ago. While all major improvements cannot be discussed, following are some of the more significant:

Earlier tubes had some deformation of the characters at the screen edge. The modern tube is sharp to the edge, with much greater resolution. New bright phosphors have been developed including a pastel green which eliminates spot size variation or "blooming". When necessary, tube length can now be dramatically decreased. A tube 25 in. long now achieves the same results once requiring a tube 45 in. long.



Time-share version of CHARACTRON Shaped-Beam Tube.

SPEED

Many optimistic goals have been claimed regarding the speed of character writing tubes. Frequently, however, these claims do not delineate the time required for the positioning of these generated characters but simply state the time necessary for generation. It is a simple matter to blink a character at tremendous speeds at the same place on the tube face. Generating different characters and positioning them in different places on the tube is something else. The shaped-beam principle generates characters in a period of time independent of the complexity of the character. Complex symbols can be generated as simply as a dot. With high speed circuitry, selection can be accomplished at rates equivalent to oscilloscope deflection frequencies.

For example, characters could easily be generated at a million each second. However, today's magnetic deflection yokes require a minimum 5 to 8 micro-seconds to settle the magnetic domain in the core and this is the limiting factor in positioning speeds. Using a high speed selection system and allowing five to ten micro-seconds for unblanking, CHARACTRON Tubes can provide realistic writing rates of 50,000 characters per second or more, even using random deflection. Electrostatic deflection tubes now under development promise writing speeds of up to 200,000 characters per second.

ECONOMICS

A CHARACTRON Tube by itself appears to be relatively expensive, but a system using this tube can economically justify itself easily. This is true because the CHARACTRON Tube replaces both the necessary character generator and much of the circuitry required by other systems.

In recent models, alignment procedures have been simplified, and the tube holds alignment longer than other character writing systems. A CHARACTRON Tube can be set up by an experienced man in less than one hour. Tubes are available in a wide range of phosphors with practically any desired color or degree of persistence. Resolution of 1800 TV lines can

advanced techniques possible in data display

be provided, the only limitation being the grain size of the phosphor. CHARACTRON Tubes are no more fragile than any other cathode ray tube. They have been exposed to a 32G shock for 52 milliseconds without harm, and can take just about any shock that does not fracture the glass. In one application, the tube was used in a portable battlefield display console.

PICTURE WINDOW TUBE

Frequently, it is necessary to continuously repeat certain data on the face of a tube while changing other data.

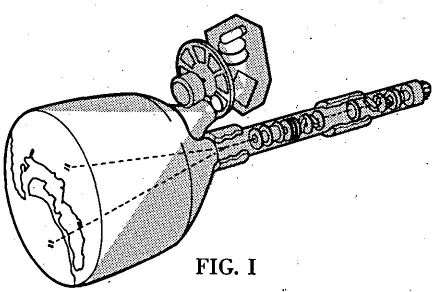


FIG. I

This may be done easily with a new development called the "picture window" concept. In the "window" tube, changing data from computer, radar, or communications link, is presented in the usual manner. Repetitious data are projected through the "window" onto the faceplate using a slide or film projector (Figure I). In a typical application, a geographic map of the area is projected on the face while the computer presents changing data. As the area under surveillance changes, the operator pushes a button to select another map. In another application, business or engineering forms are projected on the tube and filled in with data from the computer. Included in this option is a recording camera. By means of a beam splitting half-silvered mirror, the camera maintains optical access to the entire tube face. A button actuated solenoid operates the camera, recording all data being displayed.

TIME-SHARE TUBE

A new "time-share" version of the tube produces alphanumeric data and at the same time performs beam writing to draw curves and vectors. In the drawing mode, electrons pass through a special large aperture so that none of the beam is blocked. Brighter beam drawings result. The name "time-share" is derived from the fact that both the alphanumeric and drawing mode share the beam from one cathode for part of the time. This tube is ideal for applications such as long range radar where the antenna may turn at a relatively low speed of six times a minute.

TWO-GUN TUBE

On short range radar requiring high rotation speeds of perhaps 25 times a minute and many hits on small targets to build up an image, there may not be enough time left for forming alphanumeric symbols. With these applications, a two-gun tube (Figure II) is suggested. This tube retains the beam shaping electron gun for producing characters and employs another gun to accomplish the video writing. This second gun, when coupled with video driving circuitry, can be used to generate high resolution TV images including scan converter readout or

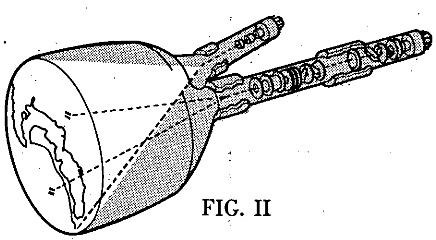


FIG. II

raw radar data. These images, of course, can occur at the same time as and without any effect on the alphanumeric data supplied from the shaped-beam gun.

SYSTEMS

In addition to offering the CHARACTRON Tube as a display or recording character generator, General Dynamics|Electronics has a number of custom and standard display, printing and film recording systems which utilize the tube.

Custom installations include direct-view consoles as well as film recorders which automatically process and project large-screen displays for group viewing.

The S-C 1090 Display console (Figure III) presents alphanumeric, symbolic and graphic data from computers

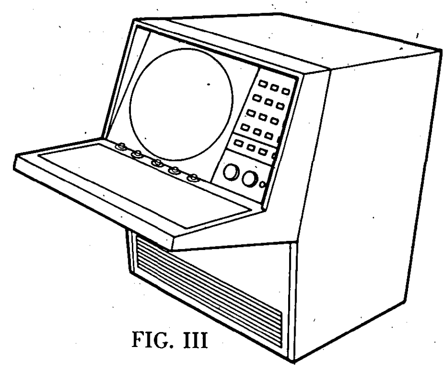


FIG. III

or other sources. It is a complete, "off-the-shelf" display unit. Optional equipment includes internal test routine, input register, level converters, internal storage of complete display frame, vector generator, expansion and off-centering, category selection and various data channel buffers. The console is 66 in. long, 32½ in. wide, and 47 in. high. It is recommended for a variety of applications, including command and control systems, air traffic control, computer readout and data display for any automated process.

The S-C 4020 records the output of large scale computers on film and/or paper at equivalent speeds. Combinations of drawings and alphanumeric data may be recorded in fractions of a second.

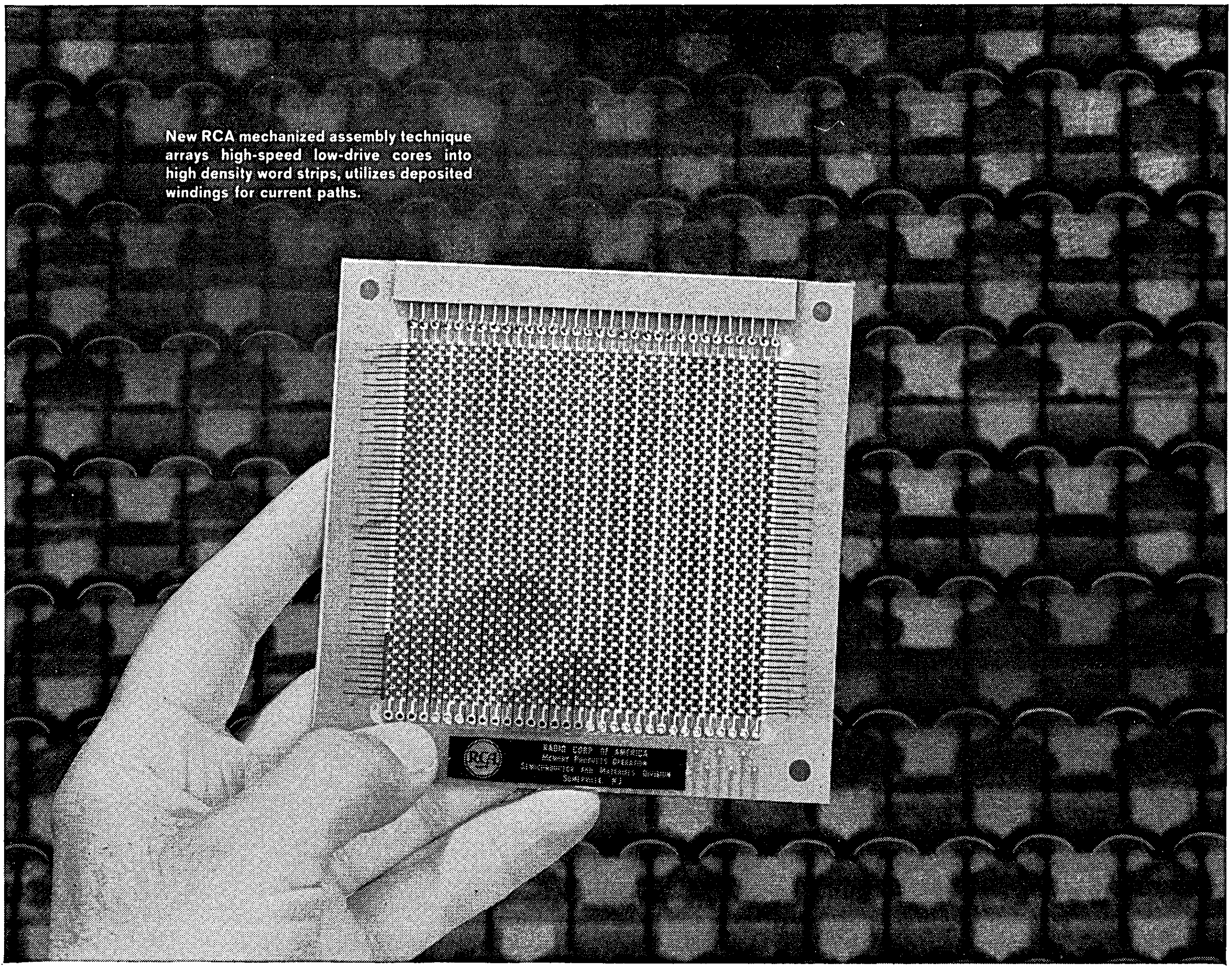
The S-C 3070 provides high-speed asynchronous printing without impact on paper for communications or computer output applications.

WRITE FOR MORE INFORMATION

For technical information on the S-C 1090 Display, the S-C 4020 Computer Recorder, the S-C 3070 Electronic Printers, or the new generation of CHARACTRON Shaped-Beam Tubes, write to General Dynamics|Electronics, Department D-25, P. O. Box 127, San Diego 12, California.

GENERAL DYNAMICS | ELECTRONICS SAN DIEGO

CIRCLE 20 ON READER CARD



This New RCA Microferrite Memory Stack

Completes A Full Cycle In 300 Nanoseconds With Only 350 ma Drive

Now, a major advance in Ferrite Stack Design and Construction by RCA makes 65-Nanosecond Switching a reality.

Here is the industry's first commercially available Microferrite Memory Stack with complete read/write cycle time of 300 nanoseconds at drive current levels below 350 ma—bit outputs of 50 mv.

This revolutionary two-core-per-bit word-address system bypasses today's experimental memory techniques by using proved, reliable ferrite cores in a high-density array of advanced design. Check these important benefits:

- **High Packing Density**... 1,000 to 2,000 bits per cubic inch.
- **Superior Stability and Ruggedness**... Printed wiring assures positive, rigid contact to each core. Planes designed to meet Military Mechanical and Environmental Specifications.
- **Precision Uniformity**... Mechanized fabrication eliminates many hand-assembly variables.
- **Outstanding Reliability**... Mechanized production techniques permit more precise control of each fabrication step—produce a rugged, high-reliability structure.
- **Broad Capacity Range**... Available in 32 word x 32 bit size, and in any multiple of this size.
- **Plug-In-Convenience**... Each stack incorporates standard plug connections for fast, easy installation.
- **Complete Service**... Whatever your requirements, custom or RCA standard, your local RCA Semiconductor and Materials Division Field Representative is prepared to provide a completely coordinated application service for all RCA Computer-Memory Products. Call him today at your nearby RCA Field Office.

For complete technical information on new RCA Microferrite Memory Stacks, write department FD5, RCA Memory Products Department, 64 "A" Street, Needham Heights 94, Mass.

TENTATIVE DATA
TYPICAL DRIVE REQUIREMENTS AT 25°C

	Amplitude (ma)	Rise Time (nsec)	Duration (nsec)
Read Pulse	350	30	100
Partial Write Pulse	250	20	45
Digit Pulse	70	15	85

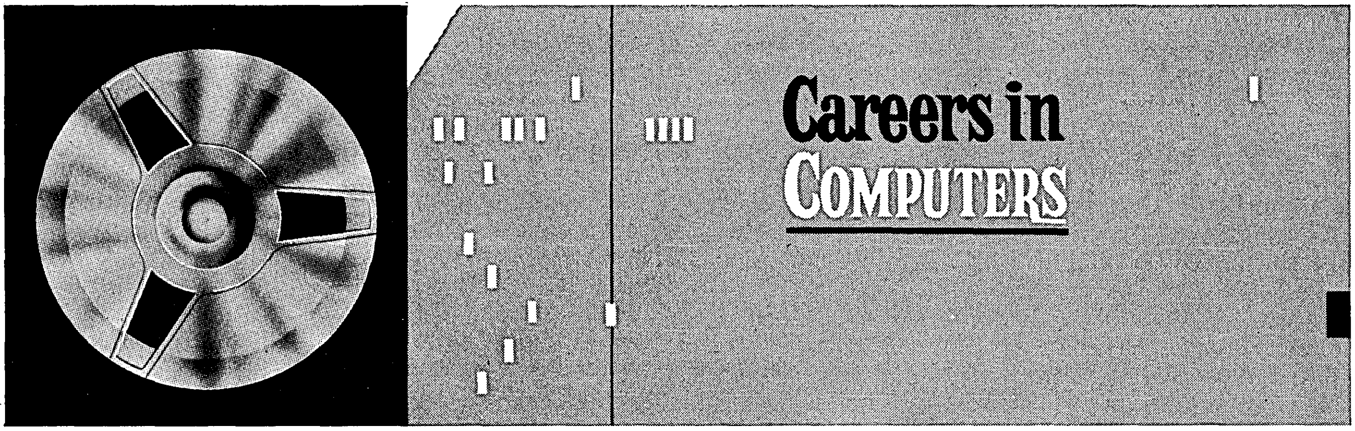
BIT OUTPUT (Two-Core/Bit Word-Address)

		Undisturbed '1' (mv)	Undisturbed '0' (mv)
Bit Out-Puts	Amplitude Sensing	60	12
	BiPolar Sensing	+50	−50
Bit Switching Time	70 nsec		

RCA SEMICONDUCTOR & MATERIALS DIVISION FIELD OFFICES ...**EAST:** 32 Green Street, Newark 2, N. J., HU 5-3900 • 731 James St., Room 402, Syracuse 3, N. Y., GR 4-5591 • 605 Marlton Pike, Erlton, N. J., HA 8-4802 • Greater Baltimore Area, 1725 "K" St., N.W., Washington 6, D. C., FE 7-8500 • **NORTHEAST:** 64 "A" St., Needham Heights 94, Mass., HI 4-7200 • **SOUTHEAST:** 1520 Edgewater Dr., Suite No. 1, Orlando, Fla., GA 4-3768 • **CENTRAL:** Suite 1154, Merchandise Mart Plaza, Chicago 54, Ill., 527-2900 • 2132 East 52nd St., Indianapolis 5, Ind., CL 1-1405 • 5805 Excelsior Blvd., Minneapolis 15, Minn., WE 9-0676 • 714 New Center Bldg., Detroit 2, Mich., TR 5-5600 • 7905 Carpenter Freeway, Dallas 7, Texas, ME 1-9720 • Continental Terrace Bldg., 2785 North Speer Blvd., Room 346, Denver 11, Colo., 477-1688 • **WEST:** 6801 E. Washington Blvd., Los Angeles 22, Calif., RA 3-8361 • 1838 El Camino Real, Burlingame, Calif., OX 7-1620 • 2250 First Avenue, South, Seattle 4, Wash., MA 2-8816 • **GOVERNMENT:** 224 North Wilkinson Street, Dayton 2, Ohio, BA 6-2366 • 1725 "K" Street, N.W., Washington 6, D. C., FE 7-8500 • **RCA INTERNATIONAL DIVISION:** 30 Rockefeller Plaza, New York 20, New York, Cable Address: RADIOINTER, New York

THE MOST TRUSTED NAME IN ELECTRONICS

CIRCLE 22 ON READER CARD



Senior Programmers
Analyze and program computer solution of scientific and data processing problems in guidance, process control, trajectory analysis, statistical problems in fire control, signal processing, numerical analysis, inventory and production control. B.S.E.E., Math., or Physics.

Information Systems Engineers
Numerical analysis, error analysis and mathematical model construction in the synthesis of weapons and information processing systems. Design and specification of data processing systems programs. M.S. in Math or Physics.

Computer Programmers
For real time programming analysis and development. To develop compilers, problem oriented computer language and advanced programming systems.

Commercial Programmers
To develop business programs using detail block diagrams and symbolic language. Will test and debug individual programs and modify existing programs as necessary.

Applied Mathematicians
To work on the application of computers to complex numerical problems. M.S., Ph.D.

Computer Sales Engineers
Digital computer experience in sales, engineering or applications programming.

Operations Research Scientist
M.S., Ph.D., physical or behavioral science background and OR orientation and experiience required. To work in areas of war gaming, analysis of weapon and logistic systems, design and application of simulation models.

Systems Analysts
Experienced in the study of customer's problems and individual data processing needs who can suggest economical and practical system configuration for a particular application.

Logical Design Engineers
To work on the proposal, specification, and logical design of EDP systems. Two or more years of related experience is required in digital systems planning and specification, detail logical design, or systems design.

Circuit Design Engineers
To work on design of transistor and/or magnetic circuit for computers.

Systems Engineers
To develop and analyze complete digital computer systems requirements, from source information to final display, control, printed and/or other output. Will coordinate logical, circuit, and mechanical design effort on new systems and develop technical performance and cost information.

Digital Computer Engineers
Air-borne system and detail digital computer experience required. Must possess intimate knowledge of digital computer techniques as applied to closed loop logic control systems and methods of test and computer systems verification.

Contact us during the Spring Joint Computer Conference at the Sheraton-Cadillac Hotel, Detroit, Michigan, May 21, 22 and 23. Client companies will be interviewing at our Suite.

Professionals whose backgrounds are in the above areas, please submit complete resume with salary requirements and geographical preference to either our Boston or New York office. Our fees and relocation expenses are paid for by client companies.

DATAMAN*®
ASSOCIATES
PERSONNEL CONSULTANTS

120 Boylston St., Boston, Mass. — 473-5858   30 East 42nd St., New York, New York — MU7-6330

CIRCLE 90 ON READER CARD   CIRCLE 91 ON READER CARD

As the concluding segment in DATAMATION's triplet of taped interviews with leading consultants, the following session was conducted prior to publication of the CSC and CUC stories. Readers may reference preceeding interviews with Fletcher Jones and Elmer Kubie appearing in the March and April issues. Order of appearance was based on alphabetical arrangement of corporate designations.

THE CEIR VIEW

an interview
Dr. Herbert Robinson & George Dick

Q: I'd like to begin by asking for a capsule history of CEIR.

ROBINSON: We started business in March, 1954 (we are 9 years old), working in econometrics, statistics, and large mathematical economic models. Within 18 months, we found we needed computers to run the models and get answers to practical problems. So we installed an IBM 650 in 1956 but, within about a year, found this was inadequate to handle the size of the problems we were called upon to solve for our clients. We moved to the IBM 704 in late '57, and this started our career in the computer technology side of the business.

Naturally, with a large machine like that, we had to build capabilities in programming, systems analysis, computer systems design, and so on. Accordingly, we rapidly enlarged our capabilities to cover a very wide range of professional services, especially those required to exploit the computers.

Our purpose was not just to be in the computer business but to provide the broadest spectrum of problem-solving professional services, using the computer as our most powerful tool – at the same time satisfying a real need in computer services for those people who could not afford to program, or could not afford machines themselves. Our business grew; we put in an IBM 709 as a prelude to the IBM 7090, and some 2½ years ago we began to expand from a local Washington organization to a national and international one. We made some acquisitions, and established completely new CEIR centers. The cities involved were Boston, New York, Houston, Los Angeles, San Francisco, London, Paris and, finally Mexico City.

This ambitious expansion was accomplished in a very short time. It meant, in many instances, starting from nothing and creating an operating replica, we hoped, of the complex we had in Washington. We ended up with 7090's in all these locations except Mexico, where we have a 1401. As you might imagine, such expansion was expensive; if you write off all the preparatory and start-up costs – don't capitalize any of them – then obviously the company goes deeply into the red in order to put itself in business in these locations.

The program was not actually completed until the summer of '62, and since then we've concentrated on making this international complex work efficiently. That is our present preoccupation.

In every location we intend to enlarge considerably the professional base. We feel this is the key to the future of the company. The machines are there as a tool; at this moment our own use is often inadequate and we sell machine time simply to gain the required revenue, but this is not the main purpose of the operation. And as we see the future, we will have operations that are comparable to our Washington services. Over 50 per cent of machine capacity there is used on our own projects not just programming, but problems completely solved for our customers. This is our concept. It is summed up in our motto I plus A(I), "intellect plus the amplifiers of intellect." This is the keynote of our company – a professional rather than just a programming and computer services business.

Q: Could you give us some idea of the growth of the company in terms of volume of business in the first year, and in the last fiscal year?

ROBINSON: Yes. The first year we did business at the rate of $240,000 a year, and in 1962 we did roughly $17 million.

Q: What about the capital assets of the company?

ROBINSON: The total assets the end of the first year were $37,000, and at the end of 1962, $15,600,000.

Q: And a comparison in terms of professional personnel? This would exclude secretarial and clerical help.

ROBINSON: I would say — this has to be an estimate — that we had about 20 professional personnel at the end of the first year but we now have over 400. In CEIR we consider our marketing and computer operation staffs must be highly professional too, which increases our full-time complement to about 600. Some 160 part-time specialists complete the present professional roster.

Q: Have there been major personnel changes recently within CEIR? If so could you explain these changes?

ROBINSON: Yes. As I mentioned earlier, once we had completed our expansion program, we felt we had to concentrate on making the whole complex work efficiently. We sought expert outside advice of management consultants, and reorganized the company. For example, at corporate headquarters we have brought in Mr. Dick as executive vice president. This position was long overdue, and Mr. Dick is responsible for the entire operating management of the company.

Q: Could we include a brief resume of your background, Mr. Dick?

ROBINSON: In addition, we have brought in Mr. Bob Holland, vice president for finance and administration. Mr. Holland also has a distinguished career in this kind of business.

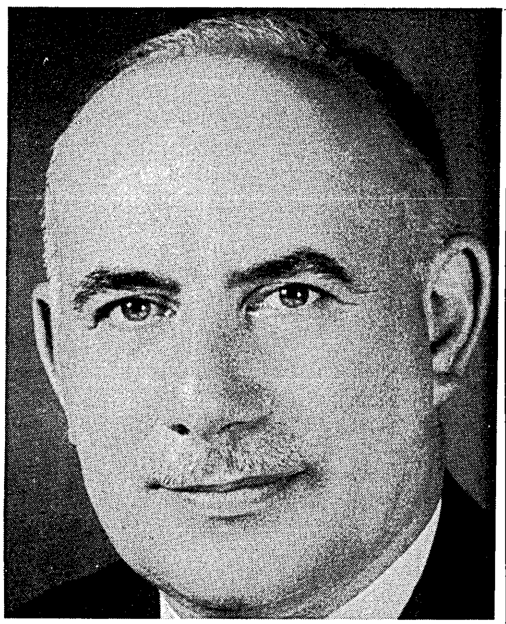
DICK: Bob has been in the industry since the early '50s, just at the beginning of the computer outbreak, we might say. His last position involved responsibility for industrial marketing for RCA. To continue, we brought in Meade Camp as vice president-marketing. Most recently he was responsible for scientific marketing at RCA. He too has a long background, from the early '50s, in data processing industry management.

ROBINSON: I think we should mention that our main interest was in Bob Holland's heavy experience over the years in the financial side of the business.

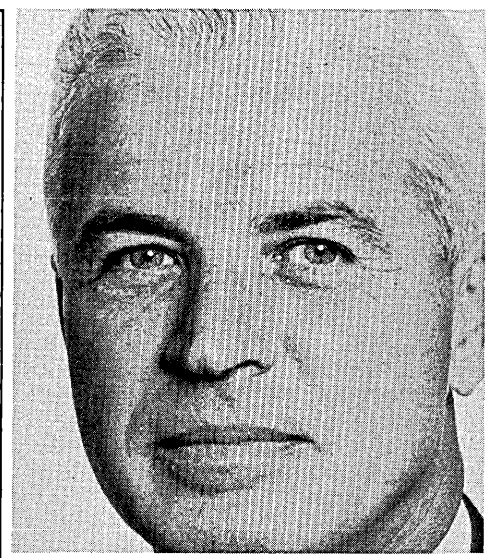
DICK: The background of all three of us is important in this respect: while each of us has in a sense a specialty, we are all marketing oriented and have had extensive marketing experience. This was sought purposely to balance the heavy professional orientation of almost everyone else in CEIR. Now we're trying to blend these two essential orientations throughout the company.

Q: Have these additions served as replacements for people who have left?

ROBINSON: Two principals have left. Of course, we must go back to the beginning of the company. Among the founders of the company were Mal Catlin who died in 1960; Rudy Johnson, who retired when he reached age 65; Bill Eaton, physicist, who filled many roles, one of which was marketing although he was not a marketing specialist; and Al Reimers who also filled many roles, especially in finance and administration. The last two decided, at the time of this reorganization, to withdraw



Dr. Herbert Robinson



George Dick

DICK: Yes. I have been mainly associated with the data processing industry for almost 25 years. I was with IBM for 17 years in various marketing and corporate management positions. I was also with American Mutual as vice president of marketing. I am a CPCU (Chartered Property and Casualty Underwriter) in that industry. Following that I joined RCA as divisional vice president and general manager of the new Commercial Systems Department. Basically, I had the responsibility for marketing, engineering and manufacturing the 501, 301 and 601. For the past year, I've been doing private consulting in the management field. During this particular period I was invited to join CEIR as executive vice president. It was an important decision for me and for CEIR to make.

from an active role in the company. (Dr. Eaton, incidentally, has since been appointed Deputy Assistant Secretary of Commerce for Science and Technology.) We have now filled the positions in corporate headquarters with specialists in each corporate function.

DICK: And it is not our intention to add. As a matter of fact, we have considerably reduced the corporate staff.

Q: By what number or percentage?

ROBINSON: Let's say, in dollars, 50 per cent. Probably in numbers, by a third. We feel we now have a streamlined, vital corporate team. Our main objective now is to build the properly balanced teams in the regions to perform our mission successfully and profitably.

Q: My understanding was that one or two regional offices

have been eliminated. Is that correct?
DICK: Not exactly. May I explain why we have closed Houston as an operating center (it still remains a sales office). Naturally, with the rapid expansion the company pursued during the past couple years, a lot of things could happen. It's possible, for example, to be premature in the market. I think this was the case in Houston. We closed the Houston center on March 1. Philadelphia is very definitely a sales and professional CEIR location. It has not, so far, had a computer. The main direction of CEIR is problem solving, which is not necessarily tied to computer installations. We feel we can expand throughout the world, for that matter, and only move in our own computers where it is profitable to do so.
Q: Are there changes in any of the other regional offices? Offices closing or opening? Will the overseas offices remain as they are? And, aside from Houston, are there any other major changes that are in the offing?
DICK: We're constantly on the alert for new opportunities both in the U. S. and overseas.
ROBINSON: I think it's true to say, basically, that we've had a pretty good batting average on the locations we selected. The Houston one was not far from being viable, but we decided, as George said, that it was just a little premature.
Q: What happened to the equipment you had in Houston?
DICK: We moved the Houston machine to Boston, where there was a rented machine which will go back to IBM. That's the 7090. The 1401 went to our San Francisco center, also to replace a rented machine.
Q: Have there been any other reductions of hardware in any of the other branch offices that you presently have?
DICK: No.
Q: I understand that the Los Angeles office has planned, certainly as far as your annual report indicates, to remove its 1604/160-A.
DICK: That is true. The decision was made some time ago.
Q: And I also understand that there is some plan to give up the 7090 in Los Angeles.
DICK: No. Both the 7090 and the 1604 are presently operating in our Center on Wilshire Boulevard. Of course, it is entirely consistent with our basic philosophy that we will, whenever profitable and practicable, share our equipment with a compatible partner and such a move is quite possible in Los Angeles. This would be similar to our arrangement with Union Carbide in New York where we are able to render the full range of our services yet enjoy the financial benefits of such an arrangement.
Q: Do you have plans for additional equipment at the present time?
DICK: We always are ready to act promptly in response to opportunity or project needs.
ROBINSON: We expect, if I may elaborate here, that while we're now concentrating on our present locations, opportunities may well arise which are attractive and profitable enough that we would go ahead immediately. We are not by any means saying we're stopping anything new.
Q: Aside from corporate headquarters personnel, in terms of operating personnel in the field on a professional level, has there been any reduction and if so, roughly what percent?
DICK: There has been about ten per cent, temporarily, in an effort to tighten up the ship.
Q: I understand from your annual report that you are removing the Los Angeles 1604. Is this because of incompatibility of programming, or what?
DICK: I would say it's a function of timing as much as anything else. It is not a reflection on the equipment. You well know there are many manufacturers of good equipment. It just happens that we weren't in a position to sell the kind and quantity of applications on this machine necessary to make it profitable to CEIR. There's no question that when you have extra time available on a machine, you need a population of users of that type of equipment and there are not that many 1604 users as yet in the area.
Q: Specifically, then, you feel that it would have been difficult, or was difficult, to persuade customers to put their programs on a 1604.
DICK: At this particular time, yes.
ROBINSON: It seems there is a certain obstacle or barrier to simply taking FORTRAN, say, and recompiling and running on another machine. It isn't quite that simple, and even with a very attractive price it seems we've not succeeded in getting the interest our marketing people expected.
DICK: It can well be CEIR's inability to solve this particular marketing problem quickly enough.
Q: Has there been any decision on the disposition of the machine? It was your machine, and it was partially written off. Have you decided where the machine will go?
DICK: Not yet, but we have some interested prospects.
Q: Could we have your reasons for the disposition of the STRETCH?
ROBINSON: I think the decision came when our technical people analyzed the capabilities of the newly announced 7094 vs. the STRETCH, and the costs and programming required. We came to the conclusion that economically we would be just as well off with the 94's—and we could have them in many more locations. I think this was a sound decision. We took perhaps six weeks of very thorough soul searching and analysis before we made the decision. It was a very painful one for us. We had invested a tremendous amount in preparation for the STRETCH. We reached what we felt was a reasonable settlement with IBM. We reached the decision quite apart from any multi-processing considerations concerning the STRETCH. We found that it was ruled out on other considerations even before we got to that question.
Q: A final question on STRETCH concerns the point that you brought up, namely the very reasonable settlement which was reached with IBM. It was our understanding that the cancellation penalty was $250,000 per machine.
ROBINSON: Your figures are not accurate. All I should say is that both of us felt we had suffered considerably from the venture, we had serious negotiations for quite a while, and we hammered out an equitable settlement for both of us.
Q: It has been claimed that one of CEIR's problems has been an overcommitment to machinery and an undercommitment to manpower. Do you feel this is correct?
ROBINSON: The implication is inaccurate. Some have charged CEIR with being "machine-happy" but this is completely untrue. However, we must admit that to a casual observer, equipping ourselves with massive machine support could have created this impression. As we said at the beginning of this interview, our objective has always been I+A(I) – the emphasis being on "intellect". Our only mistake, although a serious one from a financial viewpoint, was in assuming a greater ability to muster enough problem-solving business that generated in-house machine use quickly enough to justify some of the equipment installed. Having excess time forced a stepped-up computer service marketing effort, which many misinterpreted as a policy shift. This was not so. In order to quickly get back to the main purpose of CEIR, we are sharing our equipment where there is currently excess capacity, as we mentioned before.
DICK: I am constantly amazed, as a newcomer to CEIR,

A. how quickly most of the centers have developed a strong acceptance in the market place.
Q: Do you think the market can absorb your existing machine capacity within a reasonable time?
ROBINSON: We are making real progress developing a market among small users, thereby broadening our base and reducing the risk of large fluctuations.
Q: Are there any other changes planned—aside from the personnel and equipment—such as the rate structure? Will there be any lowering of rates or changes in the presentation of rates, and departure from the prime shift concept?
DICK: Basically there will not be any lowering of rates, but there may well be some changes in the pricing of machine service. Prime time varies from coast to coast, and we must be careful that pricing is fair to national customers.
ROBINSON: We in the business are far from having solved all the problems of pricing this service. It's a complex problem. We're continuously examining our ideas and experience and are open-minded about what we do in the pricing area. We realize we still have to find the right formula.
Q: As long as we're on the question of rates, Computer Sciences has recently announced a departure from the prime shift concept. I wonder if you would care to comment on their concept?
ROBINSON: Yes. I'm familiar with that. It has some attractions and we will watch carefully how it works out.
Q: Would you comment on the brokerage business such as Computer Usage has, and whether you feel that this might have some virtues?
ROBINSON: It is a valuable activity for those companies which have excess machine time available and we feel the machine brokers are providing a useful service in the industry.
Q: Is there any change in CEIR's policy of offering the facilities of the company for solution of the problem with the machine used as one tool for that solution?
DICK: No. The order of emphasis within CEIR is (1) professional problem solving, (2) the sale of what we call proprietary programs – PERT, RAMPS, Electric Utility Applications, LP 90/94, Mediametrics, Time Series Analysis – management tools that we feel are profitable to us and to our customers and which also generate machine time, (3) the red-carpet computer services for those who have their problems programmed, and who, with our assistance, get their problems organized and solved using our equipment.
ROBINSON: But I should stress that the last service is not just button-pushing. It is, as George said, complete red-carpet treatment for the professional man. It is a professional service.
Q: I wonder if I can ask for a general comment on the field of computer consulting. What do you see is happening or what perhaps might happen in terms of the growth of the field?
ROBINSON: Do you mean consulting in systems design?
Q: Consulting in any fashion that would be related to electronic data processing, either in the preparation of software, the sale of time, or the design of the system.
ROBINSON: I feel that this is a growing field, undoubtedly, especially with manufacturers pushing the computer deeper and deeper into the economy. This is going to create a tremendous number of users with problems. These will be people with less EDP sophistication – to whom consultants can be invaluable. The service would include selection of equipment configuration, and the right kind of system – really a management system – to use data processing techniques as a tool in their business. We think it's very rapidly growing and a promising field, one that will increasingly use our services.
Q: Do you have an estimate as to the size of the field in terms of gross volume of business?
ROBINSON: If you took a snapshot today of those involved, I mean services such as we are, not captive groups, I think I would put it at $75-100 million a year at present as a rough guess.
Q: What might it increase to by 1968?
ROBINSON: I think it could more than double.
Q: What about the size of firms in this field? CEIR is one of the largest and has spawned a considerable number of smaller firms, and there are many other small firms presently in the field. Do you feel this will be a continuing trend?
ROBINSON: Well, I think that we must distinguish here between small firms that are programming-oriented and the type of problem-solving business we are building. I think our business contemplates a much broader range of professional services, a totality of skills needed to solve problems, and I don't think this can be done by a very small firm. I think you need the large firm to have the required specialties.
Q: Do you think, however, there is a permanent position in the field for a large number of small firms in programming?
ROBINSON: In programming? Definitely. In fact, we welcome this. While we sometimes feel their competition, they often become our customers. They are the professionals we serve with our red-carpet computer service. I would say there is a definite place for the highly talented man who is also an entrepreneur. We're getting individual companies with very fine talents, the market for whose services flows out of their own specialties.
DICK: I'd like to suggest this though: I don't think there has yet occurred the shakeout that happened in the electronics business, for example – a lot of the electronic firms that were started by people with strong entrepreneurial desires and a specialty. They were engineers usually, and as they grew they found that their problems no longer could be handled between Joe and Bill and Tom. They became a going business and they were not equipped with the management skills to handle the complexities of a larger company. Our industry has not yet seen this, but I think it will.
Q: There seems to be a very heavy interest in the time plus cost contract as opposed to the fixed-fee contract. I wonder if you have any opinions in this area?
ROBINSON: I think the reason for the popularity at the present time of the time and material contract is that in this new industry so much of the work is still developmental, and it is difficult to establish a fixed price. The successful companies of the future will be those using fixed price who have developed much better techniques for defining the problem and estimating the cost of the job, and who also employ sound contract administration as the job progresses.
Q: Some people feel there is a problem in the consulting and service field – that of the lack of competence by many firms, perhaps many of the classically accountant-type firms which have gone into the computing area. I wonder if you would have any comment on this?
DICK: This can be true. We have heard of some actual instances where the basic problem has been lack of competence but these are isolated instances and certainly not typical of the responsible reputable firms.
ROBINSON: I would like to add that increasingly, as the complexity of the consulting business grows, the necessity

arises to be completely immersed in customers' problems. It becomes necessary to have a great deal of experience in actually executing jobs, actually doing the data processing. I think increasingly it is becoming true that you can't have a consultant who isn't in that kind of work every day: many of the jobs demand this kind of complete immersion in day to day operations to produce the right kind of product.

DICK: Surface consulting, I think, is on its way out. It will have to be consulting in depth.

ROBINSON . . . . . . and this will grow to become more and more apparent as the consumer becomes more sophisticated. Some of the mystery of EDP is disappearing – losing its magic nature – and this inexperienced practitioner that no one understands is now becoming understood and is losing some of his weight with management.

Q: What tests would you prescribe for the user to judge competence?

ROBINSON: Well, one of the best questions is: Can you give me an example of something you've actually done, and engineered, and made work? There's nothing like a happy customer to be your best salesman.

Q: How about measuring professional capability of employees by seeing resumes? Does this have any value?

ROBINSON: I think it's valuable but it's very partial because you will see a man with a wonderful background on paper who, in practice, doesn't do a good practical job; likewise, you can see a man whose resume doesn't seem outstanding, and yet he's got just what is needed to produce a practical problem solution. Too many contracts are awarded on the basis of individual resumes and not enough on achievements of the organization.

Q: One of the problems that concerns us a great deal is what has been referred to as a Tower of Babel in languages – there are too many languages. I wonder if you would comment on this. Do you feel that this will change in due course or is this going to be a continuing problem for the next 10 years?

ROBINSON: Well, I think it's bound to improve because I think there must be some underlying, ineluctable logic that would ultimately prevail; the reason you've got so many is that it is a rapidly developing field and we haven't yet answered all the problems. Once we've solved the problems, I think automatically you'll get a convergence.

Q: Use of the COBOL language and compiler has been hailed as a possibility for compatibility. Do you feel this is a real possibility?

DICK: I'm no programmer or authority on these languages, but I have been in the computer business for a long time and I feel that, looking at the subject from the manufacturers' point of view, in order to be able to compete with each other, they've got to keep coming up with the best and strongest hardware. This tends to hinder the compatibility we seek in a common language. We don't know enough about the software side of the business, in my opinion, to accommodate all the **present** hardware, let alone what is in the pipe line of machine design. I think it will be a long time before we have program systems that are both economic and widely compatible.

Q: Do you feel that the problems that CEIR has had, as outlined in its latest annual report, can quickly be solved?

ROBINSON: Of course they can. During the past several months, we have already solved many of our most difficult problems—however, I must admit, at considerable short-term cost. For example, we have incurred heavy penalties in order to be relieved of long-term leases entered into in anticipation of STRETCH and other earlier planned expansions. We have decided to concentrate for the time being on solving our remaining problems in order to provide a firm platform for our future progress.

Q: Is there any one area that you would definitely like to touch on that I haven't brought up? About the company, or some expression on the field?

DICK: Being new at CEIR, I still feel I have an objective point of view. CEIR is a romantic business. Anytime you talk about solving problems, particularly a lot of the frontier and esoteric type of problems we do solve, it's something for the man on the street to get excited about. This is what happened; there was some overexcitement from a stock market point of view. But this does not detract from the fact that it is fundamentally a sound business, going in the direction which I personally feel has a tremendous future. Any time you are helping people, so long as you don't price yourself out of the market, it seems to me that you have a going business if you perform successfully for your clients. We're dedicated to doing both of these – pricing so that the customer can get a reasonable solution at a reasonable price, and performing. These two things will build CEIR's value to its employees, to its clients and to its stockholders.

ROBINSON: Yes, I feel that the company has done something basically that is very unique. It has tried, in the space of three or four years, to multiply itself 15 times, to become national and international overnight, in a completely new area of technology where there was no fund of experience, no signposts, nothing to indicate the wisdom of the policy decisions that had to be made. And I hope that, in retrospect, it will be said that we had the right vision at the right time of a golden opportunity, that we did marshall enough capital to see us through the decisive period, and that we emerged finally with the rewards that were anticipated. I believe it will be said that CEIR took a bold, dramatic but also wisely calculated risk in undertaking a most worthy and productive endeavor.

Q: Finally, Dr. Robinson, what problem would you consider to be the most important that the industry faces at this time?

ROBINSON: In one word – education! First, our problem is to educate our customers in the true potentials and challenges of the tools and techniques of our new technology – the full scope of **what** can be accomplished economically and socially by this powerful combination of intellect and the amplifiers of intellect. They must also be educated in **how** to apply this combination to their own fields of endeavor. The heart of this problem is to educate top and middle management in industry and government sufficiently in our technology that they can use the tremendous resources now available in our industry to the greatest advantage in solving their practical problems. They must be educated to embrace and exploit these new concepts rather than to fear them, and to understand the exciting new horizons opened up by the new technology in almost every aspect of their management life.

We must also realize what is perhaps of even greater importance, the problem of educating ourselves. We fall victim to narcissism when we become too impressed and elated with our own technical achievements. Our shortcomings are still tremendous. We must educate ourselves in **communication** with the layman and the managers; we must educate ourselves in **adaptation** to the practical problems with which we deal – to adapt our technology to solving the problem rather than distorting the problem to match our technology; we must educate ourselves in finding a **discipline** of work and thought to truly take our place as an important new working member of the complex of trades, skills and professions that constitute the productive apparatus of the economy; and finally we must educate ourselves in our own **technology** to raise it to the level of a science that is recognized by the universities and the lay world as being on the same plane as physics, chemistry and mathematics. ■

In appreciation of the acceptor's primer given in R. W. Rector's article, "A Guide to Testmanship" in *Datamation's* December 1962 issue, the following manufacturer's primer is presented.

TESTMANSHIP REVISITED

a manufacturer's primer

by KIRK WHITMAN, Ebasco Service, New York, N.Y.

The acceptance test is in some respects anticlimactic. It was preceded by years of what was optimistically considered development work but without a product. This was followed by months of having a product but lacking a ready application. The next step was a feasibility study yielding a ready application but without a sale. Finally after months of persuasive marketing (and guided by the engineering technique known as brute force) a sale was consummated. It then became the design engineers' and programmers' task to invent or develop a system which fulfilled the specifications of the purchase order. After several more months, the machine, called a computer, was built. Externally it looked like a row of beautifully painted gymnasium lockers. It contained miles of colorful spaghetti-like wire and rows of circuit cards, each of which had a collage of miniature electrical parts. The circuit cards indecisively stood on their end like coins which were neither head nor tail. Systemwise, the entire conglomeration surprisingly represented a supreme effort in logic.

The time and anxieties thus involved can give impetus to shipping the computer as quickly as possible. The only possible obstacle might be the so called "unacceptable acceptance test".

facing the test

No two acceptance tests ever run the same! One reason is that there always seems to be some bugs which wait until this time to show up. This is really not difficult to understand when you consider that most of the testing which the manufacturer does as part of his quality control concerns components rather than the system. However, the acceptance test is primarily interested in the system and in the computer functions which usually involve combinations of components. Thus, this may be the first opportunity for the bug to be uncovered. While this may seem to be an excuse for bugs being present, don't mention this to the acceptor who is liable to become disconcerted in finding that there isn't 100% quality control. The impression should be created that even after large scale system checkouts, there may be a certain rare combination not yet checked and this is the type bug which may be seen. The manufacturer's engineer should also be prepared to take the following diagnostic steps:

1. Check the test setup. This is expected of him and

besides he may discover something.

2. Diagnose it as a programming error. This will not seem serious to the acceptor for everyone knows programming errors are simple to detect and to correct. This will also bring credit on the engineer for having software knowledgeability.
3. If, however, a programmer is present, diagnose it as a hardware problem. Programmers are notoriously reluctant to admit a programming error. This is probably encouraged by two factors: 1) Few people concerned besides programmers have the wherewithal to catch and prove programming errors and 2) there may be sufficient opportunity for the programmer to correct the program without anyone realizing what he is doing. While the hardware engineers are checking out their equipment to find the bug, the programmer may insert changes into the computer. The next time the test is run, the bug has disappeared to the relief of all concerned.

To the already charged atmosphere, the manufacturer, while maintaining absolute confidence, should subtly raise the customer's suspense in a manner to cause him to emotionally root for the success of the machine. The acceptor must never be able to forget that DELIVERY WILL SLIP IF THE MACHINE IS NOT ACCEPTED. The full impact of this idea can only be achieved if the computer sales personnel, over the previous several months, have conscientiously generated excitement in the customer's organization for the many special plant operations the computer can serve, especially before and during trial operation of the plant. This will insure sufficient pressure from all sides so that no time is wasted in getting the computer accepted and installed before the plant has its trial operation.

The manufacturer's engineer should bear in mind that THE ACCEPTOR MAY KNOW MORE ABOUT THE MACHINE'S WEAKNESSES THAN MANY OF THE MANUFACTURER'S PEOPLE. This is a result of the pre-purchase order melee in which the user is bombarded with the competitors' advice and research about the other bidders' machines. It is best not to awaken these now dimming memories. Thus certain key phrases should be avoided. Some of these are:

1. 99 per cent availability
2. on-line assembling and compiling (and the efficiency thereof)
3. on-line debugging
4. computer payout period

These phrases of course vary from one individual to another so the engineer might be wise not to use any computer oriented cliches. The ultimate would be not mentioning computers at all, but would be difficult in this situation.

defining the tests

There are certain tests which the acceptor may think of which the manufacturer would prefer not to do. In order to counter any trend to thinking in this direction, the manufacturer prepares a description of the tests which usually is a modest guide to what the computer should be able to do. If the test procedure is submitted sufficiently in advance, it will act as an effective thought stifler. Given proper circulation through the customer's company, it will considerably relieve the acceptor of responsibility of approving the ground rules. If handled properly, the customer will not even realize that he has the right of modifying the tests. Should this subtle phase be bungled, the manufacturer may face as acceptance test with an addition such as the following:

"Test to failure under conditions of extreme limits of voltage, frequency and temperature".

This is the other extreme to the manufacturer's plan of testing only to the limits of the product specifications. (A compromise of testing to determine some safe margin beyond the product specs would probably be a more informative and equitable test).

mutual dependency

Computer manufacturing people pride themselves on knowing infinitely more about the computer system than outsiders such as the customer. This is only a half truth. There is no doubt that they are masters of knowledge concerning their computer, being that they designed, built, programmed and tested it. However, in speaking of computer systems the application knowledge becomes just as important for successful operation. It is here that the customer (with his consultants) can be the expert by a country mile. The manufacturer must try to keep up with application theory presented to him in the form of performance calculations and other equations. Sometimes he is able to comment on this theory due to knowledge from previous jobs but rarely is he able to initiate it in the sophisticated process and power generation fields. He is completely out of the picture in the inevitable refinements to the equations which occur in the field as new operating information is correlated. This limitation of knowledge is an observation rather than a criticism. Costs of "computer systems" would increase substantially if the manufacturer were expected to do a system study in depth for each application.

Emphasis should be placed by both the manufacturer and customer upon making available such information as will help the understanding of each other's concepts and problems. The fact that this will improve the job is a truism that hardly needs stating. This philosophy should carry down to even the smallest hardware details. It is probably inconceivable to the manufacturer that the following obvious mistakes could occur in a computer installation, but is it really impossible and couldn't the consequences be crippling to the machine. Does your information preclude the occurrence of:

1. The washroom door swinging open to hit the computer cabinet and cause a parity error.
2. The chiller piping being connected to the water fountain.
3. The unused space in the Input-Output cabinet being used to store hard hats and wet raingear.

The above sections have attempted to lightheartedly give the customer's view of some real and some imaginary acceptance situations. It is hoped that the following paragraph will be seriously considered as being basic to acceptance testing.

significance of the test

The unwarranted acceptance of a poor machine can result in consequences several times more serious than rejection of the machine at the tests. It would be, in fact, a Pyrrhic victory. Manufacturers are becoming aware of the costly nature of extensive hardware and programming modifications in the field. They readily admit that a field change is at least several times more expensive than the same factory operation. It is therefore of mutual advantage that every effort be made in the factory to expose and correct lurking problems. The acceptance test is actually the manufacturer's last chance to catch the problem on his home grounds. With this philosophy the acceptance test can become known for the special technique that it is: a quality control stage rather than a meaningless ritual. ■

THE AMERICAN WAY

a comment on economic gambits

by ROBERT L. PATRICK, Computer Specialist, Northridge, Calif.

Almost daily one hears such remarks as "*Why don't they ever come out with a mainframe that's really new?*"

"*Well, we'll get together a sub-committee, draw up some specs, and get the computer manufacturer to implement it.*"

"*He must be doing all right. Look at the facility he has.*"

If these three statements are analyzed to determine what they have in common one concludes that none of the speakers has even the basic understanding of the free enterprise system. At the risk of being pedantic, a short note on how our great country works is in order.

First, some old cliches need to be drawn out and dusted off.

> "*The free enterprise system is based on the right of an entrepreneur to exploit a situation for profit or loss.*"

Many otherwise educated individuals ignore the "or loss" phrase. A person may not affect unfair competition or misleading advertising, but fair competition is allowed. It seems that many of our leading individuals have never stopped to apply the scientific method to the world around them and ask, "Why?" Why do the Department of Commerce statistics indicate a high failure rate among young companies? Why do dividends fluctuate? What is the reason for growth stocks selling at high prices when they have never shown a profit? Why are some companies classed as blue-chip? These are all related to some basic premises of the system.

First, it should be obvious that

> "*You can't sell something that the buyer doesn't want to buy.*"

Every day companies try to reverse this basic law. The sale of iceboxes to Eskimos is an example. Computer time offered where no shortage exists is another. Advertisements and salesmen try to convince a reluctant buyer that he should sign on the dotted line. "Reluctant" is the key word. This implies that the guy is a potential customer. If the guy is not even a *potential* customer, they are wasting their time.

Another axiom that shouldn't need stating, but evidently does, is:

> "*You can't sell to a customer who lacks the ability to pay.*"

Perhaps "can't" is too strong. Shouldn't is more appropriate. The whole purpose of the game is to recover more than your investment and hence to show a net after taxes. This is impossible if the customer is unable to pay. Some of our own major corporations let their commission salesmen get too excited and sign up loss business. (Salesmen are traditionally paid for the signed up sale and not for a profitable deal. Some magnificent in-house battles have been due to this defect.)

Another variant on this same theme is to take in trade something the buyer has surplus in return for a service given. This is another bit of confused thinking. If programming and coding services are supplied, somebody is paying for them one way or the other. These services are not free to the supplier. Barter, be it in programming services or machine time, is something to be engaged in only if the recipient has an immediate sale for the goods received. Both programming services and machine time are perishable and cannot be stored.

If the guy is willing to buy, and has the ability to pay, then a deal *can* be made. Even then there is no promise of a sale; just a probability greater than zero. For a deal to be consummated it must, in the final outcome, be beneficial to both parties. Many companies overlook this important aspect and unpleasantries result. Sometimes lean and hungry companies "buy" a deal by offering uneconomic conditions under the presumption that follow-on sales will be sufficient to show a profit. This is mortgaging the future for the present, a risky thing to do. This puts the seller at the mercy of the buyer. What if he continues to drive hard bargains? What if his future need evaporates? What if you are not successful in the pursuit of the work and he won't tolerate you at any price? What if your special friend gets transferred and the replacement doesn't recognize his predecessor's obligation? These things do happen. This is the best way I know to do several million dollars worth of volume and show a loss.

By the same token, the buyer should be hesitant about pressing too hard. Without an equitable deal the seller cannot continue to exist and will not support the equipment as it should be. On the other hand, a software contract too hard driven will cause the vendor to divert a significant portion of his energies toward trying to escape with his skin—energy which should go into the product. Even in the era of the Federal Supply Schedule (computereese for Fair Trade) sufficient latitude exists for squeezing too hard. The line between free and paid software, the date rental begins, and how we account for second shift are all areas where deals can be made. If all is well and good, if the sales personnel can counter all of the arguments involving intangibles, the deal finally hinges on price. Clearly it is important to know how prices are set.

"Only in the rare case of the sole supplier is the price related to cost."

That statement may appear strange to some, but it bears examination. If a company obtained a patent on a new device, they would total up their predicted costs for sales and manufacturing, add in some for profit, and recover their initial development costs by adding in a pro-rata share after they predicted the market volume. Clearly the procedure for setting an initial price is iterative: the size of the market affects the price; the size of the price affects the market. Most companies are careful not to set too high a markup on a new product as this invites competition. Henry Ford was the first successful operations researcher. He observed that the optimum strategy for maximizing profit was a low markup, a high volume, and a standard product line (any color you wish as long as it is black).

Once the pioneer has set the price, done the advertising, and established the market (or gone broke) the game changes.

"In an established market, the competition sets the price."

How many times have businessmen overlooked this fundamental. Perhaps that sentence bears re-reading. Rephrased, it says:

"You sell goods or services for what you can get."

This makes no mention of profit, cost, or worth. The price is determined, ultimately, by the buyer, *not* the seller. The game goes this way. After the product specs are available, the performance is predicted. The performance of the competitor's offering is also predicted. Based on what is already in the field, and whether you own it or not, a selling price is chosen (chosen, not computed). As before, the development cost is included as a lump. The manufacturing costs per unit are computed, and sales expenses are included as a function of sales activity (volume of units). Corporate policy usually dictates the profit margin of so much per unit. After all of this is done, a linear equation in one unknown (number of units) is written. This is made equal to the expected competitive price. The expression is then solved for the unknown and the number of units is found. This quantity has several names, but the automobile companies call it the standard volume. If the manufacturer sells just the standard volume he makes the profit margin dictated by his management. Any less, and the profit picture loses its rosy hue. Any more, and the stockholders benefit magnificently. Plot it for yourself and see.

After the competitive price is set, the product will be killed if the selling and manufacturing costs exceed the market value. The endeavor may also be doomed if it looks like the standard volume exceeds the reasonable expected share that the manufacturer has previously enjoyed. Sometimes such products get to see the light of day anyway. This is attributable to the marketing crew losing their objectivity. When they stop assessing the competition and start believing their own news releases, a sad spiral usually begins.

We can now see why it has been said that

"It takes money to make money."

If we assumed that engineering costs are roughly proportional to the component count, then it takes approximately the same engineering to manufacture an IBM 1401, an RCA 301, a Solid State 80, or an H400. The components themselves cost about the same. The software is required to be competitive and is not a function of volume. The selling expense is sensitive to volume, but in favor of the higher volumes. Just predict for yourself what happens to your profit margin if your sales are 600 versus sales of 8000! Clearly this is a game where only men can play. Now to the subject of software. Just what was its original raison d' être?

The purpose of software is to sell machines. It's just that simple. Some customers mistakenly attribute the manufacturer with the elemental concern for other than his own welfare. Nothing could be farther from the truth. If the manufacturer appears magnanimous, it is because he sees this as being the best way to maximize his long run profit. The software is supplied to make machines more appealing to the multitudes and get the production volume up! It is a technique to get more machines of this vintage delivered before the competitor's next model inhibits further sales.

A manufacturer spends some risk capital in order that they can exceed the standard volume by the greatest amount, thereby maximizing the profit on a particular model. A secondary benefit also results. The more you sell, the more potential repeat customers there are. Now this is a very serious business. There is no guarantee to any player, no more than the tables at Las Vegas guarantee a win. Clearly one should bet only what he can afford to lose, exert his best efforts, and pray. If he has a good product, the customers will find it out no matter how stupid his salesmen are. This is a game like cricket; there are rules which all gentlemen are supposed to follow. If you don't, you may get away with it, but most probably you will be found out. Perhaps an analogy with ice hockey might be better: stay within the rules, protect yourself, and play to win.

If the above sounds too trite, the following quote from a leading standards light may throw some fuel on the fire. He said, "But it also depends on keeping the conversion costs to each adopting group low, by permitting that group to retain almost intact the system it already has. Since no particular group should be isolated from the main community, and since no one group may be favored over others, it is not difficult for the result of such discussions to be a practical or even a logical impossibility." Clearly this man does not understand what the standards activity is. Standards committees are the forum where the aforementioned gentlemen agree on the rules by which the game will be played. To be sure, the customer will get equity in the long run, for this is the only way for all to prosper. Recognize the standards mission for what it is now and is going to be. The big boys are sitting down and deciding the rules for the competition. Don't let the Justice Department think for a moment that we are dividing up the market so that each retains his present "fair" share and we just won't isolate him from the rest of the community. YOU JUST WATCH! If a man can't compete, he withdraws from competition or gets hurt. In a situation with one overwhelming competitor, he exercises some self restraint or big government steps in. This is sometimes worse. There is already some indication that BIG GOVERNMENT is fooling with the free market to the advantage of the marginal producer. I pity the government official or civil servant who gets caught spending tax money for the second best just to stimulate competition.

There are some that seem to think that the manufacturers have an infinite supply of money. Many of the users groups entertain this demented idea from time to time. At the time a manufacturer sets his price and determines his standard volume, he establishes budgets for all activities. There is a budget for the mandatory basic software, and there are additional software monies to satisfy special demands from large segments of the user population. If the users make their desires known, this extra money will be allocated to fulfill their requests. As long as all of the budget money is not allocated, the manufacturer *appears* to give in rather easily. As soon as all of the money is allocated, the manufacturer suddenly becomes quite firm and resists demands made upon him. Clearly he is protecting his profit margin. If the user persists in making demands, then adjustments are made. The adjustment is usually made to the previous commitments. The quality of the product or the quantity of the work suffers. This is no more nor less than what should be expected. The user has achieved equity: if the manufacturer responded to additional demands, he would not be fulfilling his responsibility to his stockholders.

Other observations can be made about the major manufacturers which appear to be universal. Clearly the attitude of a major manufacturer is very similar to a defensive boxer. He must always be cocked and ready to counterpunch, but he never (or almost never) exhibits the initiative in an *established* field. His attitude would be characterized by

> *"Always be second, never be first."*

If any wild experimentation is to take place, this will be brought about by the smaller manufacturers who have nothing to lose and everything to gain. The major manufacturer will not be guilty of obsoleting his own equipment in the field.

But, on the other hand, if competition forces the return of obsolete equipment, the major manufacturers will leave no stone unturned to guarantee that those established customers stay in the family. Watch the antics of the principals yourself and see if this doesn't explain their behavior.

It appears that the computer field is reaching the end of the salary and rental spiral. An old axiom states

> *"The charges for direct salaries and supplies approximately equal the first shift rental of the computer in an average installation."*

Thus as the size of our computers keeps increasing and the rentals paid to the manufacturers keep climbing upward, the salaries of the installation manager and the programmers follow. The users groups have placed a premium on having at least a minimum hardware configuration so that they can communicate programs with each other. Some cold-blooded installation managers are discovering that they seldom interchange programs with anyone else anyway. This awareness is partially brought upon by what is happening to the finances of the parent company.

If the gross business of the parent company is fixed, or if the total size of the work force being served by the computer installation is fixed, there comes a time when additional computer equipment cannot be justified. When the computer was first installed, it was justified on the basis of work untackled, competitive position, or straight economics. After the first computer was installed, it was replaced by a larger additional computer. The total budget for computing increased monotonically. If this trend were allowed to continue to its limit, then there would be no money left for the primary activity of the firm: the computing shop would use the entire gross income of the parent company.

Clearly this will not be allowed. Furthermore, the cutoff occurs somewhat short of such a magnificent empire. Corporate managements are beginning to wonder what the "proper" allocation of resources should be, and how much of their precious resources should go for computing. When the computer shop approaches this fair share of the corporate gross, then the computer shop will cease to grow. There are some indications that many managements feel that the time has arrived.

Historically the user groups moved as a body from machine to machine as soon as the manufacturer could deliver them. The user groups are beginning to fractionate. Eventually the time will come when each corporation will install as much computer as it needs and can afford, independent of what the other firms around it may be acquiring. There is some indication that all 7090s may not *ever* become 7094s merely due to the onset of this situation.

In closing, a comment on installment buying. Some ill-informed people confuse installment purchases with ownership of an item. An installment contract is a commitment to pay so much money for an object, be it a computer or an automobile, over a stipulated period of time. As is well known, the buyer does not own the device until the contractual obligation is fulfilled and, at that time, the value of his purchase is worth merely what he can obtain for it in another sale. Sometimes the value of a used computer is so low that there is no market whatsoever and the equity, after the installment contract is retired, is merely salvage.

The prime assumption in installment buying is that the buyer is financially solvent and will stay that way long enough to meet his obligations. Furthermore, it is assumed that he will need the device longer than the payout period. In short, the manufacturer shoulders the additional burden of financing, and runs the risk of having to take back the second-hand article and re-sell it.

When a user rents a computer, he frankly admits that he cannot anticipate enough load to purchase, he lacks the finances to purchase, or he declines to be encumbered for more than a set period. Thus, the deal is remade repeatedly until something disturbs the delicate balance. That something might be competition, the workload, or some alternate technique for accomplishing the function. When a man walks right up and buys, he is making some profound predictions for the future. Sometimes he can make these predictions come true, but sometimes the market changes in some unpredicted way and he goes down the drain. Such are the risks of the American Way. ■

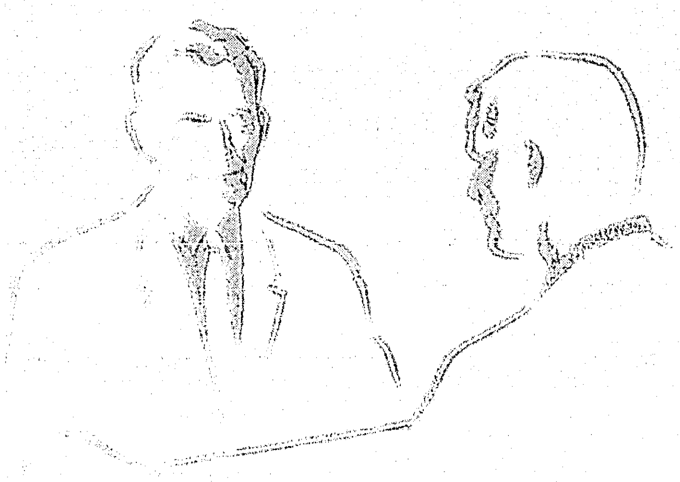
Recently we surveyed a number of engineers who use Tally Mark 20 Series systems to automatically duplicate and verify paper tapes for a variety of computer applications. Here's the unabridged feedback:

ASK THE MAN WHO RUNS ONE

(A Tally Paper Tape Verifying/Duplicating System)

Tally Is Fast. Operation at 60 characters per second was cited as the No. 1 reason for choosing the Mark 20. (New Tally Mark 20 systems operate at up to 75 characters/second. See announcement elsewhere on this page.)

Tally Is Error-Free. "Never mispunched yet," one man said. Admittedly this is an exception. However, all those interviewed said they were able to produce error-free tapes, and with little trouble. One engineer said that to him error-free meant that you could not possibly mispunch. This is not true, of course. But duplicated tapes are error-free, and this is why: in Mark 20 systems two tapes are verified and a third generated. If two tapes do not compare, duplication is halted automatically before an error is punched. Then the third tape is verified against the master. Otherwise, as one computer programmer noted, he would have blown a missile or two by now.

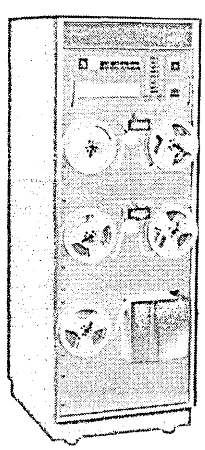


Tally Is Reliable. The worst case of trouble reported was three minor servicings needed in a year's time. One system operated flawlessly for 12 hours a day for one month. One man had to straighten out his relays before he was happy. Nobody had any serious complaints.

Tally Is Economical. Several of those interviewed noted that Tally offered them the best price/performance ratio. Actually, Tally is by far the lowest-priced 75 character/second tape duplicating and verifying system.

Tally Is Flexible. The users found such features as forward and reverse operation, character insertion, and 10 or 15 key input for original tape preparation provided system flexibility and simplified operation.

Summing up, all said they were well-satisfied with their Tally systems. If you'd like to check for yourself, we'll be happy to supply the names of the users we called.



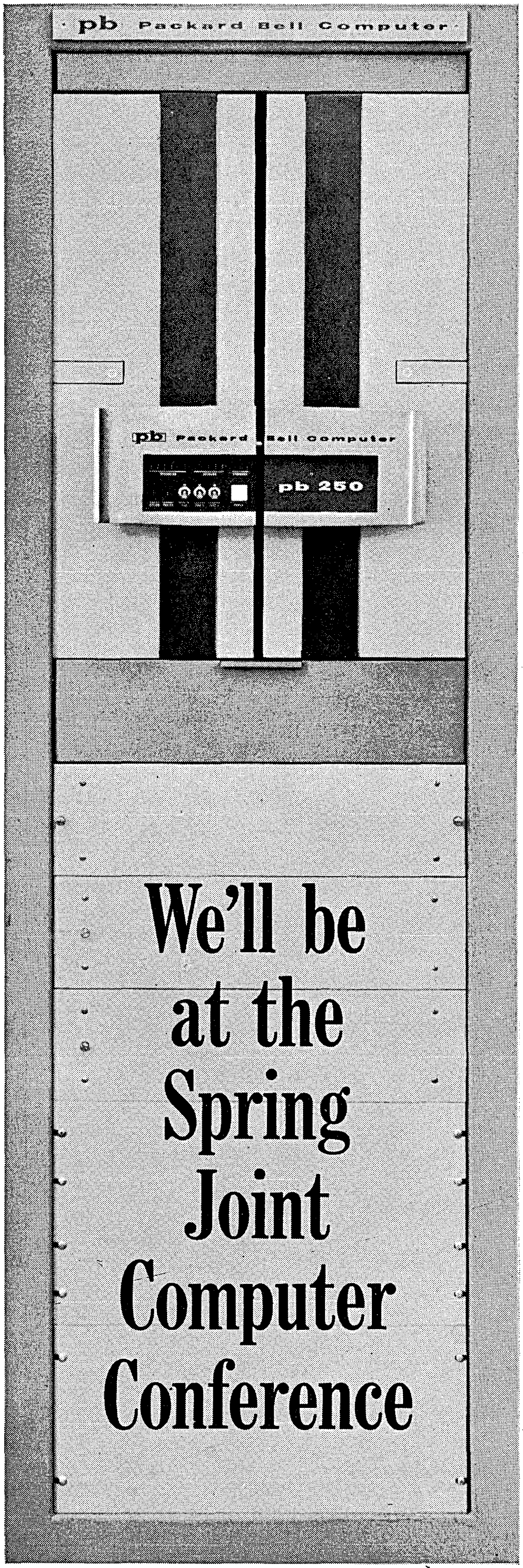
AND NOW, THE NEW MARK 20 SERIES

The new Mark 20 Series offers even greater performance. Solid state logic permits system speeds limited only by perforators, readers and other input/output devices. System speed is now 75 characters/second with a new perforator. New options include parity check; skip blank and/or delete codes; code recognition; keyboard or auxiliary input buffer, and an auxiliary output buffer.

For details and prices on new Mark 20 Series systems, call your nearby Tally engineering representative or write us directly. Address Tally Register Corporation, 1310 Mercer St., Seattle 9, Washington. Phone: (206) MAin 4-0760.

TALLY®

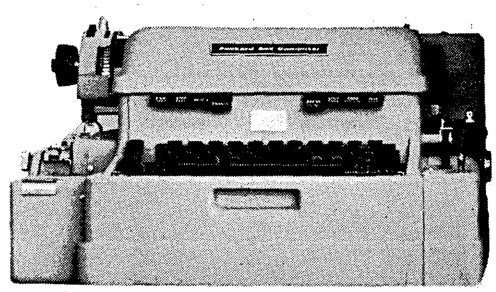
CIRCLE 24 ON READER CARD



Cost Breakthrough for Computer Control from PACKARD BELL

PB250 Computer
2320 word memory
Flexowriter with paper tape reader and punch. A complete control computer for $33,000—saving the systems designer $7,000.

PB250 Computer
3856 word memory
Flexowriter with paper tape reader and punch. A complete control computer with expanded memory for $38,400—saving the systems designer $4,700.



These new PB250 prices mean that you can now have a solid-state microsecond speed computer ready for operation at up to $15,000 or 30% less than any comparable computer on the market today.

Nearly 150 PB250 digital computers have found successful application in a variety of system applications. Including: automatic checkout • data logging • radar antenna, astronomical telescope and wind tunnel control • data acquisition • telemetry and tracking. Now, through volume production and improved manufacturing techniques, the systems designer can have the PB250's versatility and reliability at significant new savings.

For complete details on peripheral equipment, leasing and other configurations see your Packard Bell representative or write for Data File C15-7.

pb Packard Bell Computer

A division of Packard Bell Electronics
1905 Armacost Ave. • Los Angeles 25, Calif. • GRanite 8-0051

CIRCLE 27 ON READER CARD

"Our purpose is to apply the CONTROL DATA® 3600 with a software library . . . THAT MATCHES THE COMMANDING POSITION THE 3600 HOLDS IN THE COMPUTER INDUSTRY.

"Control Data's programming staff is developing a broad range of programming systems for the **Control Data** 3600 Computer with two basic goals in mind: to provide a solid base in higher level programming languages (FORTRAN, COBOL, ALGOL) and to broaden applications programming that will effectively solve specific problems. For example, in advanced linear programming, nuclear codes, and other industrial applications.

Dr. Clair E. Miller, Director of Applications, Control Data Corp.

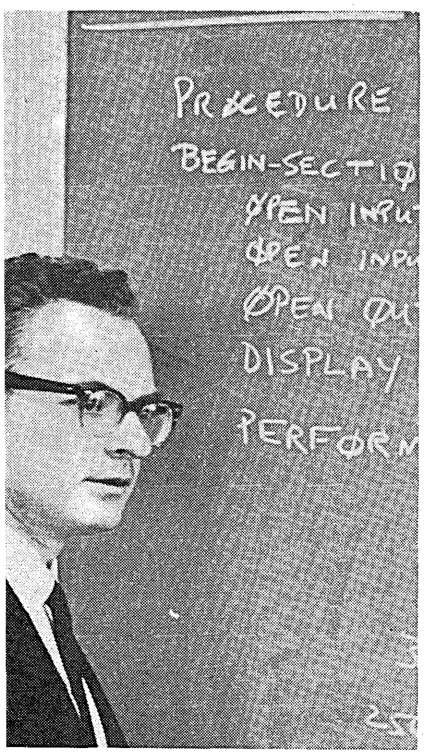
"In compilers, FORTRAN '63 incorporates all the advantages of earlier FORTRAN'S plus important extensions beyond the most advanced competitive systems. These refinements are: high object code efficiency without sacrificing speed of compilation; new statements allowing data manipulation and input/output transmission of long strings of information; accommodations allowing the user to write his own arithmetic beyond the usual types, (real, integer, complex, double logical) e.g., writing in BCD or triple precision; complete mixed-mode arithmetic within a statement with real, complex, double precision variables all in one statement; all subscripts may be any expression, and subscripting to any level provides for optimum use of the massive 3600 memory.

"Control Data's COBOL and ALGOL development is aimed at providing the widest versatility of the 3600 for use in scientific applications and data processing. The 3600 COBOL implements a larger number of elective features to take advantage of the high internal speed and more versatile I/O provisions of the 3600.

"The ALGOL 60 compiler for the 3600 will provide a flexible, efficient programming system for solving the problems of the research scientist and engineer.

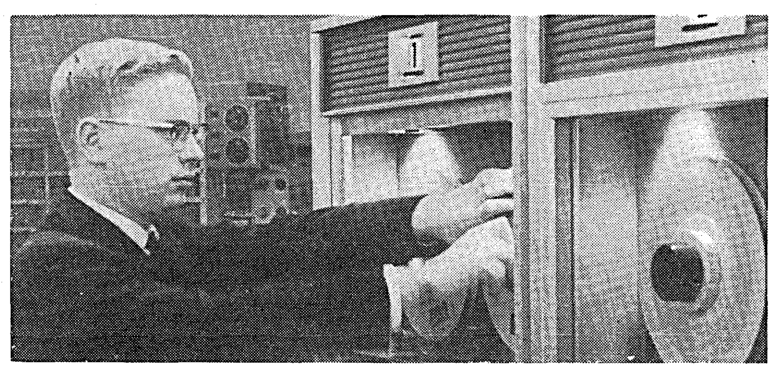
"The complete operating system for the 3600 is called SCOPE. It simplifies programming and operation of the 3600 and at the same time retains job processing speeds. Compilations, assemblies, scientific problems and data processing applications are handled with equal facility. The significant advantages of SCOPE include sequential job processing permitting any number of runs; convenient I/O and internal/external interrupt control; automatic I/O equipment assignment; program and library subroutine loading; incorporation of debugging aids at run time; complete library maintenance; and a loader that links together routines written in all of the 3600 source languages so that they may be run as a single program.

"Control Data's software development is focused squarely on applications programming. For exam-



Bill Rosenstein explains how the 3600 Cobol implements a large number of elective features.

C. T. Casale at the 3600 Console.



Control Data 606 magnetic tape handler used with the 3600.

Part of task group assigned to optimization programs is developing available "public domain" system to allow both linear and non-linear optimization.

Drs. Clair Miller and Richard Zemlin are shown discussing advanced 3600 programming goals.

ple, a task group is developing large, problem-oriented programs in fields related to nuclear physics in cooperation with present users in this field. This group is developing a basic set of nuclear codes, modifying some existing reactor codes for more effective application using FORTRAN '63, creating others for specific nuclear requirements, and engaging in original development jointly with key nuclear groups.

"With the growing interest in linear programming, Control Data is also developing optimization programs for large-scale computer systems, particularly the 3600. For example, CDM-3, a linear programming code of advanced design, includes a valuable feature that allows non-linear programming. Using the 'separable programming' technique developed in the oil industry and heavily used by several major oil companies, Control Data has produced an available 'public domain' programming system that allows both linear and non-linear optimization.

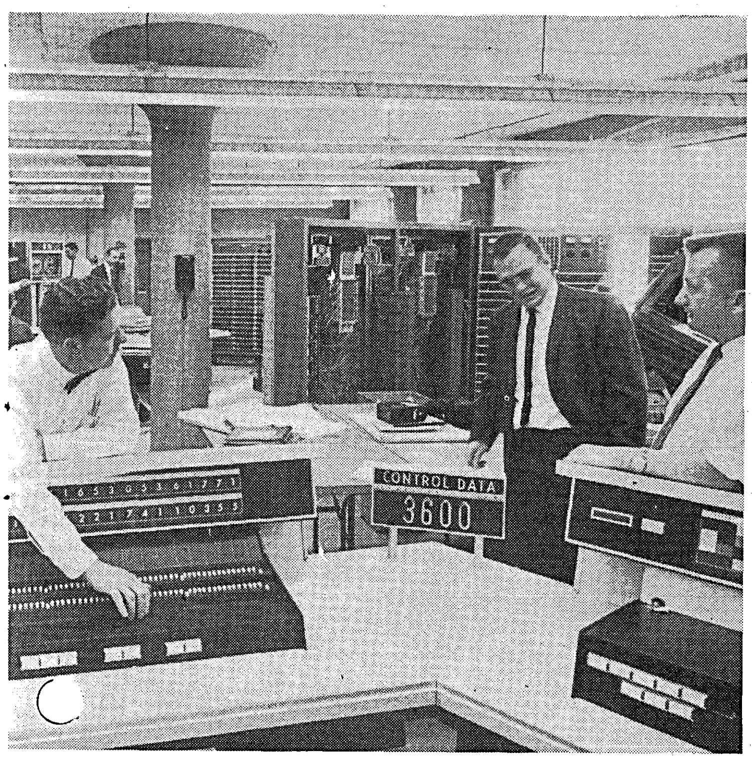
"Control Data's intention, already evident in developed software tailored to the user's problem, is to continue close coordination with a wide range of user applications while supporting them with a strong base of system programming—for problems arising in the military, scientific and commercial communities."

Offices: Albuquerque • Beverly Hills • Birmingham • Boston • Chicago • Cleveland • Dallas • Dayton • Denver • Detroit • Honolulu • Houston • Huntsville • Ithaca • Los Altos • Minneapolis • Newark • Norfolk • Orlando • Palo Alto • San Diego • San Francisco • Washington, D.C. • Wilmington.

CONTROL DATA

CORPORATION

8100 34th Ave. So., Minneapolis 20, Minn



3600 Computer System

CIRCLE 29 ON READER CARD

COMPUTER PROGRAMMERS

REAL TIME SYSTEMS DESIGN AND IMPLEMENTATION

MITRE is expanding its effort on the design and development of computer programs for critical experiments in the area of large-scale computer-based command and control systems. Opportunities exist to plan and implement such systems on the 7030 STRETCH computer within the System Design Laboratory.

Programmers experienced and interested in the following areas should apply:

- **Real Time System Design**
- **Information Storage and Retrieval**
- **Problem-oriented Languages**
- **Systems Programming**

Recent college graduates with high scholastic achievements and an interest in helping us develop these fields are also invited to apply.

Inquiries may be directed in confidence to: Vice President — Technical Operations, The MITRE Corporation, Post Office Box 208, Dept. MM9, Bedford, Massachusetts.

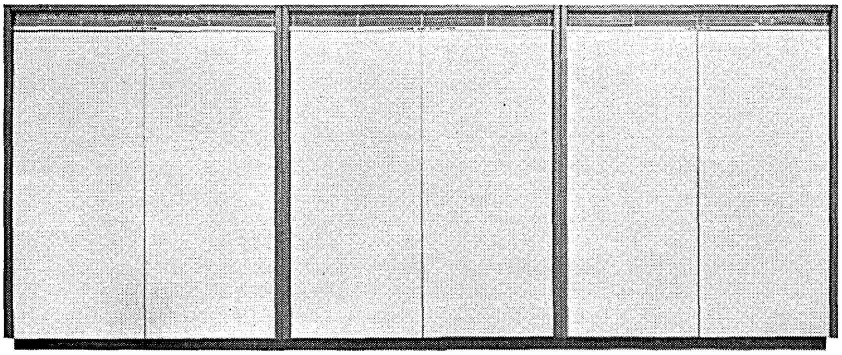
THE MITRE CORPORATION

An Equal Opportunity Employer

MITRE, an independent nonprofit corporation, working with — not in competition with — industry, serves as technical advisor to the Air Force Electronic Systems Division, and is chartered to work for such other Government agencies as the Federal Aviation Agency.

CIRCLE 79 ON READER CARD

ANOTHER
NEW
PRODUCT
FROM
DAYSTROM



636

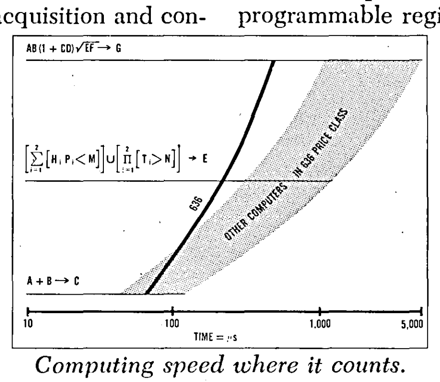
the full-size computer that doesn't limit your problem-solving capability

Here is the general purpose computer specifically designed to meet the challenge of today's complex computing applications. In fact, the more complex the problem – the more you need the 636.

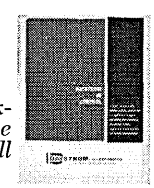
If you have ever outgrown a desk-top or medium-size computer, you already realize how complex even the apparently "simple" jobs can become. A case in point: Daystrom's last 15 on-line computer applications averaged 22,000 words – considerably more than the maximum capacity of smaller computers in the 636 price class. For on-line applications, the 636 can be expanded to 294,912 words, more than ten times this average.

Optimally designed for real-time data acquisition and control, the 636 also has the greatest, most economical expandability for off-line use in its class. Consider the 636 for any of these applications: on-line monitoring and control . . . off-line conversion . . . engineering and scientific problem-solving . . . telemetering . . . any application requiring a full-size computer.

Compare these sample 636 features to any other computer in its price class. ✱ Unlimited number of elapsed time counters. The 636 utilizes any number of memory cells as elapsed time counters without program intervention. ✱ Wide range of instructions – 131 including partial operand, square root, Gray-to-binary, 45 branches, 15 Boolean algebraic logic manipulations. ✱ Direct access to memory. Direct communication between the core memory and peripheral devices gives advantages of multiple computer installations. For example, the following functions can be executed in parallel with the normal program *without any loss of computer time:* random event counting . . . elapsed time counting . . . reading and writing on the auxiliary drum . . . reading and writing on magnetic tape . . . acceptance of digitized data up to 880,000 bits/sec. . . . direct drive of output devices and displays at the same rate. ✱ Ten programmable registers, including two additive index registers and an operand address register. ✱ Sorting of an infinite number of events on a priority interrupt basis. ✱ Expandable memory: core up to 32,768 words – auxiliary drum to 262,144 – tape up to 32 tape handling units. ✱ **Basic 636:** $95,000.



Computing speed where it counts.



Get this comprehensive booklet on the 636 by circling the reader service number, or call your local Daystrom office.

DAYSTROM, INCORPORATED
CONTROL SYSTEMS DIVISION
Miramar Road, La Jolla, California • Telephone: 454-0421 Area Code 714

CIRCLE 30 ON READER CARD

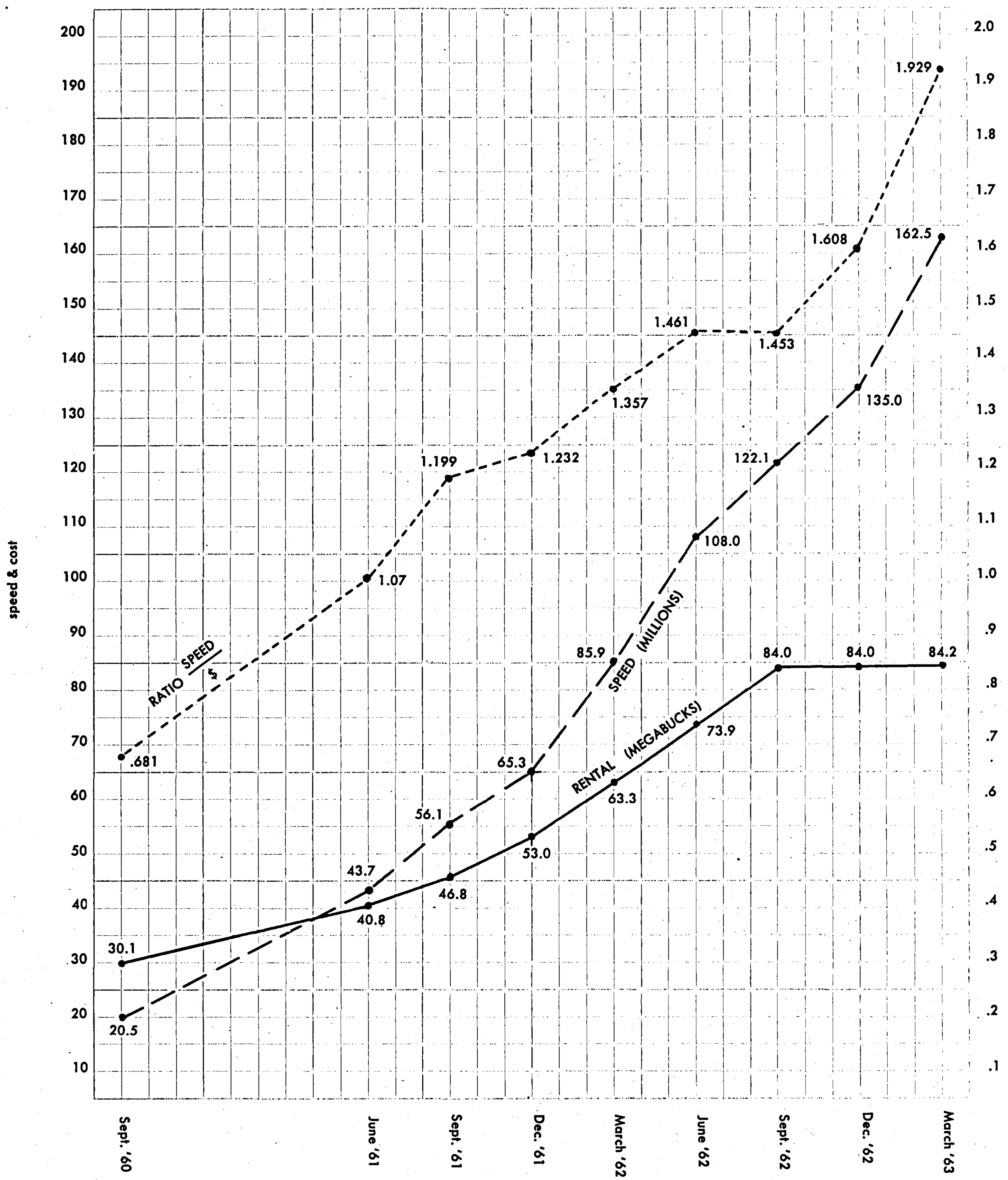
DATAMATION'S QUARTERLY INDEX OF COMPUTING

With compute speed increasing and rental costs remaining constant, the computer-per-buck ratio has risen sharply during the first quarter of 1963. Two factors contribute to this rising computing index: sales of small to medium range hardware at a faster rate than the replacement of vacuum tube devices, and the addition of large computers to the constantly-revised basis of the statistics.

The number of ops/sec rose 20.3 per cent over the previous quarter, from 135.0-162.5, attributable to both the readjusted statistical base and continuing installations of 7070s and 90s, a combined total now exceeding 500.

Concurrently, however, the cost index rose a negligible 0.2 of a point, remaining level at 84.2 megabucks for the second consecutive quarter—during which the speed index increased by 24.8 per cent.

The resultant ratio of computing power per dollar, which represents the quotient of the speed and operations per dollar indices, is established at 1.929. The 20 per cent increase is the largest during the history of this study.



NEWS BRIEFS

HONEYWELL INSTALLS 100TH COMPUTER

Honeywell's 100th installation is a 400 at MCA Inc., Hollywood, Calif., for use by the firm's Revue Productions and Universal Pictures Co. Applications include payroll, inventory control, picture cost accounting, PERT in real estate development, and film distribution.

The configuration includes 3K words of core memory, five mag tape units, printer, and card reader-punch. The installation raises to more than 100 megabucks the value of Honeywell computers in operation.

GPL OFFERS CLOCK TO METER USAGE, SET RATE

Utilization of a clock to record time in use and set computer rental rates on the LGP-30 has been announced by the Commercial Computer Div., General Precision Inc., Glendale, Calif. It is called the Meter Plan.

Rates per hour decrease with the amount of time used each month. After the first 50 hours minimum at $11 per hour, the next 50 hours are charged at $9 per hour; usage in excess of 100 hours is at $7 per hour. If the flat monthly rental is more advantageous, the user can convert to the standard contract.

CIRCLE 100 ON READER CARD

WHITE PLAINS FIRM INSTALLS CDC 1604-A

United Nuclear Corp., White Plains, N.Y., has installed a Control Data 1604-A for scientific research applications. The configuration includes two 1607 mag tape units and a 1,000 lpm printer.

The system will be utilized in the analysis of the penetration of neutrons or gammas through duct shields and to track the particles through spherical shields of varying physical compositions.

IBM REPORTS TRANSISTOR RESEARCH

An optical transistor which can speed high-frequency signal transmission by utilizing light rather than electric current has been devised by Richard F. Rutz of IBM's Watson Research Center The gallium arsenide device, capitalizing on the greater speed of light, overcomes the necessity for thin-base transistors previously required to shorten the time required for electric charges to move through it. Early experimental devices have switched signals in 10 nanoseconds, with some power loss, in an inverting circuit.

IOWA TO AUTOMATE SCHOOL CLERICAL TASKS

The use of computers for clerical tasks in elementary and secondary schools on a statewide basis has been announced by Iowa State Univ. and the state Dept. of Public Instruction. The program, called UPDATE (Unlimited Potential Data through Automation Technology in Education), will include workshops and research on common school problems.

Six school districts have been selected as pilot schools to test methods and assess services required by schools. Training of personnel has begun with one course presently being offered at Iowa State; workshops are scheduled throughout the state this summer.

During the pilot year, uniform methods of record-keeping will be inaugurated to facilitate integration of the system. The first computer programs projected are in the scheduling of high school classes; in addition to student and teacher assignments, other dp functions will include accounting, budgeting, inventory control, payroll, and bus routes. Further services for administrators will include enrollment projection, building utilization analyses, and financial projections.

As each service is developed, personnel from pilot schools will set up the program at other schools, and additional pilot schools designated as staffing for the program is completed. Director of the UPDATE program is Prof. Robert Marker, associate director of the SUI center.

Although Iowa claims to be the first state undertaking this work, California has had a pilot project completed and presently has several committees working toward an integrated system of data processing in education.

UNIVAC READIES 1107s IN GERMANY

A 2½ megabuck Univac 1107 system has been delivered to the West German government for a classified application. The configuration includes 32K words of memory, 10 Uniservo IIIA mag tape units, card and paper tape systems, printer, and a 786K-word drum. Another 1107 is scheduled for delivery this summer to the Ministry of Culture at Baden-Wurttemberg, Stuttgart, Germany.

AUTONETICS EXPLORES TEACHING TECHNOLOGIES

A nationwide series of management seminars in advanced teaching technologies, conducted by Basic Systems Inc., New York, N.Y., is being held by Autonetics Div., North American Aviation, Los Angeles, Calif.

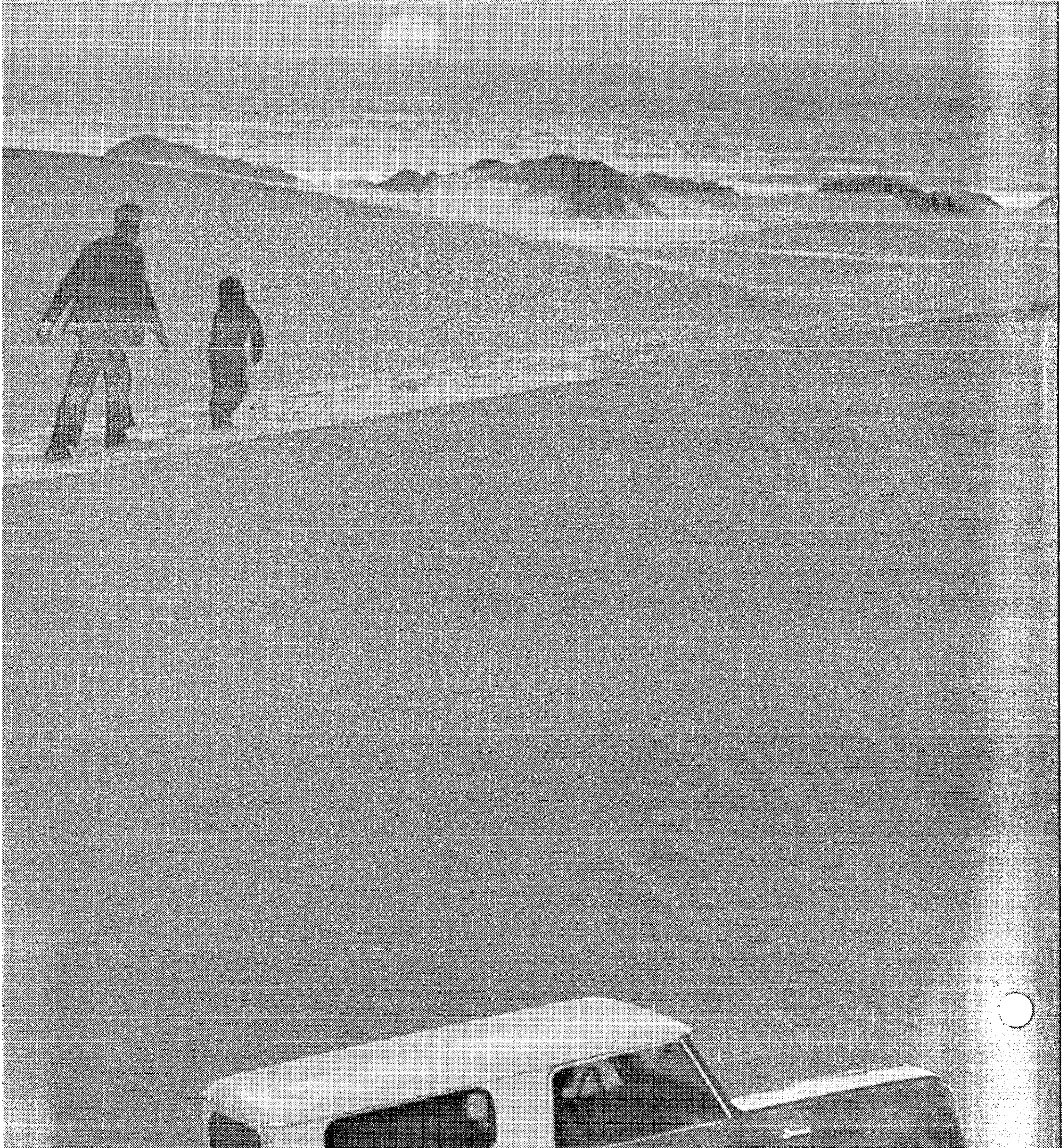
Areas of learning theory being covered include training analysis, principles and practice of programmed instruction, behavioral design, role of learner-centered teaching equipment, and development and installation of integrated training systems. Project administration in the design of programmed courses, and management orientation in the economics of automated and self-teaching systems are being emphasized.

Other Basic Systems seminars on the subject, open to industrial managers and training personnel, are scheduled this year in Chicago and at Brigham Young Univ., Provo, Utah.

DOD TO DEVELOP STANDARD MATH VOCABULARY

A conference for the development of a specialized mathematics vocabulary was called in April by the Armed Services Technical Information Agency (ASTIA). It was part of a program to revise specialized vocabularies in various scientific fields and improve the flow of scientific and technical information within the defense establishment.

Broad categories in the mathematics



microthesaurus (specialized vocabulary) include algebra, approximations, calculus, function theory, geometry, mathematical analysis and logic, number theory, operations research methods and tools, topology, and group theory.

Other ASTIA conferences to improve its document-indexing capability include the areas of semi-conductor devices, lasers and masers, and bionics.

PRIVATE COMPUTER COLLEGE TO OPEN IN ENGLAND

A private computer college in North Wales is being established by Computer Consultants Ltd., Middlesex, England. Beginning with two and three-week introductory courses, three-month curriculums are scheduled to begin in late summer. In 1964, a one-year course is anticipated; it will cover ADP equipment, programming, mathematics for dp, accounting, and statistics.

The school is intended for students in the 22-35 age group, sponsored by their employers. Fees for short courses will be five guineas ($14.70) per working day; room and board will be 15 guineas ($44.10) per week.

● An International Telephone and Telegraph 7300 ADX data switching system will be installed at NASA's Marshall Space Flight center, Huntsville, Ala. It is the sixth ADX installation. The 7300 will connect various divisions and project offices within NASA with computers in a data management center, as well as switching data between computers.

● The second price reduction in as many months has been announced by Packard Bell on its 250 series. A rack-mounted model with 3K words has been cut $4.7K, to $38.4K. Prices also were dropped on 256-word memory modules. Lease prices have not been affected.

CIRCLE 101 ON READER CARD

● A price reduction for the TRW 130 (AN/UYK-1), from $83,500 to $73,430, has been announced. Lease price for a configuration with computer, controller, tape reader-punch I/O, and typewriter I/O is $2,680. FORTRAN reportedly is being developed for off-line applications. Introduced in late 1961, 65 130s have been installed, and 50 are on order.

CIRCLE 102 ON READER CARD

● A door prize PACE TR-10 analog computer has gone to the Univ. of California, Berkeley. Given away as part of the commemoration ceremony at Electronic Associates Inc.'s Los Angeles computation center, the TR-10 was won by Warren Hull, Northrup Corp. His choice of the recipient college was his alma mater.

CIRCLE 103 ON READER CARD

● A larger memory drum, optical paper tape reader, and a magnetic card device for the Monrobot XI have been announced by the Monroe Calculating Machine Co., Inc., Orange, N.J. Monro-Cards are reusable magnetic cards holding up to 1,500 alphanumeric characters; reading speed of the device is 34 cps. It rents for $290, sells for $8.5K.

CIRCLE 104 ON READER CARD

● A Control Data 8050 message and data switching system is being built for the Crucible Steel Co. of America, Pittsburgh, Pa. The one-half-megabuck system will include two 160-As, one of which will monitor and back-up the other while also performing other computation functions. The message traffic, customer orders and inquiries, will involve the company's service centers and warehouses, and the dp center where orders are processed.

● A 160-A has also been delivered to the Univ. of Wisconsin's Psychology Dept. for on-line experiments in the area of delayed sensory perception with human subjects. It will be utilized as a controlled delay system to study effects on learning and behavior with visual and auditory inputs such as handwriting and speech.

● A $382K contract for the design and development of a digital dp system for Project Gemini has been received by Radiation Inc., Melbourne, Fla., from McDonnell Aircraft. The system will be capable of converting Pulse Code Modulation signals to digital format on mag tape for input into a computer.

● An air force contract for a third year's research in thin films has been awarded to LFE Electronics, Boston, Mass., for its Cambridge Research Laboratories, Bedford, Mass. LFE is conducting similar research for the Office of Naval Research. Studies in thin ferromagnetic films have been undertaken to control the motion of the minute domain walls that can be electrically generated to perform switching, logic, and storage.

TODAY'S COMPUTAPE IS BETTER THAN EVER—AND IT HASN'T EVER CHANGED A BIT

556 or 800 bits per inch. No dropout. That's a real guarantee of reliability. *That's a reel of Computape* — product of the first company to manufacture computer and instrumentation tape *exclusively*. Investigate new Computape today. Better still — *immediately*

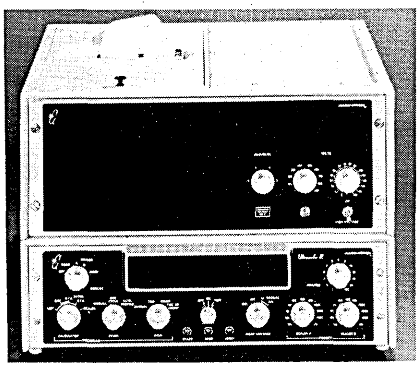
COMPUTRON INC.
122 Calvary Street, Waltham, Massachusetts

CIRCLE 33 ON READER CARD

NEW PRODUCTS

ultrascaler II

This read-out and programming system, designed primarily for automatic sample changers, combines one or two decade scalers, an electronic timer, a regulated high voltage supply and

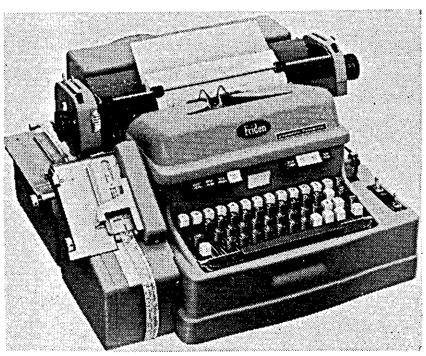


permanent data read-out. Eleven preset counts, up to 1,000,000, can be selected for each scaler, and the timer offers 13 preset time intervals. NUCLEAR-CHICAGO CORP., 359 E. Howard Ave., Des Plaines, Ill. For information:

CIRCLE 200 ON READER CARD

5-channel flexowriter

These automatic writing machines have been redesigned and standardized to incorporate new auxiliary component facilities including connections for tape and edge-punched



card reading units. Material processed on this new Flexowriter may be transmitted to distant locations over existing wire communications equipment. FRIDEN, INC., 97 Humboldt St., Rochester 2, N.Y. For information:

CIRCLE 201 ON READER CARD

"computer" of average transients

The CAT 400B is a multi-purpose, on-line device designed for detecting, isolating and measuring repetitive transient signals occurring in as finite a time as 78 usec. The 400B provides an in-internal range of 31.25 milliseconds to 16 seconds. MNEMOTRON, 202 Mamaroneck Ave., White Plains, N.Y. For information:

CIRCLE 202 ON READER CARD

sorter-comparator

The Class 406 is compatible with both electronic and tabulating systems and is able to sort, verify and perform sequence selection routines on 80-column punched cards, numerically or alphabetically, at a speed of 1,000 cards per minute. Up to 12 columns of punched data can be read and compared in a single pass of the cards at full sorting speed by the Class 406. The machine is priced at $6,600. NATIONAL CASH REGISTER CO., Dayton 9, Ohio. For information:

CIRCLE 203 ON READER CARD

bidirectional reader/spooler

The punched tape reader and spooler of model 110 are two separate assemblies, with both chassis mounting to standard RETMA panels. The 110 is able to step at asynchronous rates up to 120 cps in the character read mode. In the continuous read mode, the data rate is 400 cps. COOK ELECTRIC CO., DATA-STOR DIV., 8100 Monticello Ave., Skokie, Ill. For information:

CIRCLE 204 ON READER CARD

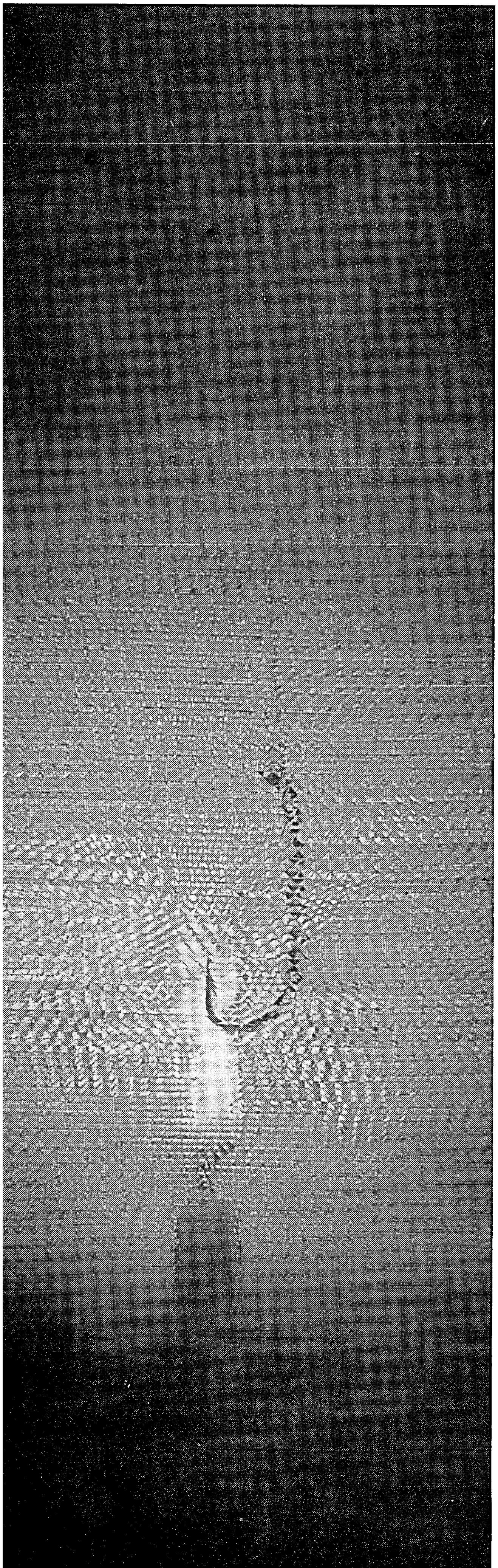
digivisor

This one-plane, on-line digital read-out device operates on a meter movement principle and is able to accept binary, analog, or decimal input directly from low signal voltage. The digivisor provides a 0 to 9 display with a maximum power requirement of 150 millivolts at 250 microamps for full scale deflection from 0 to 9. The unit is priced at $45. INDUSTRIAL ELECTRONIC ENGINEERS, INC., 5528 Vineland Avenue, North Hollywood, Calif. For information:

CIRCLE 205 ON READER CARD

digital recording and computing

This instrumentation system is able to monitor the function of a stress analysis machine, automatically detecting and computing yield point and ultimate strength. The system performs



One of a series briefly describing GM's research in depth

Fishing Btu's from fluid streams

A heat exchanger can catch escaping thermal energy. For example, it more than doubles the efficiency of gas turbines by recovering exhaust heat. But predicting its performance is a subtle and often perplexing art.

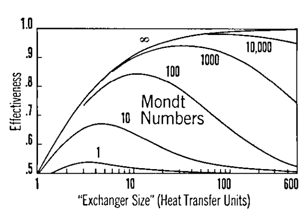
To put heat exchanger design on a more scientific basis, our thermally inclined colleagues at the Laboratories have been developing—and experimentally verifying — better ways of analyzing its behavior. One result is a new conduction parameter for the thermodynamicist's kit of tools. Dubbed the Mondt number by fellow staff members, it neatly explains why increasing the size of a heat exchanger does not necessarily improve effectiveness (see graph). The reason: longitudinal conduction losses may more than offset the expected gain in performance.

Or say you're interested in thermal distortions and stresses of metals, or in sealing the leakage of counterflowing gases. We are. Another analytical advance has enabled us to compute the transient and steady-state temperature distribution in the solid and in both fluid streams. With it most any heat exchanger can be simulated on a digital computer.

These fundamental studies are aiding the understanding of heat transfer—in theory and in engineering practice. They illustrate General Motors' constant search for a better way—through research in depth.

General Motors Research Laboratories

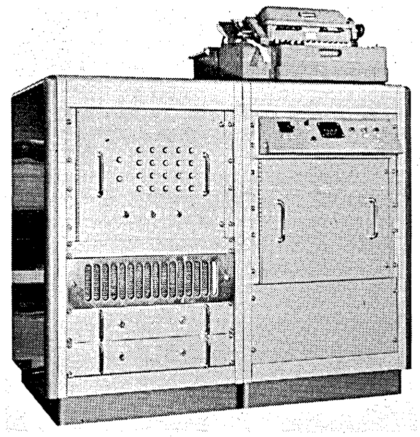
Warren, Michigan



The Mondt number is the ratio of convective heat transfer rate to conductive rate.

View through an experimental heat exchanger matrix.

CIRCLE 35 ON READER CARD



arithmetic computations and records parameter data and calculated data in pounds-per-square-inch on an electric typewriter and a tape punch. DATEX CORP., 1307 S. Myrtle Ave., Monrovia, Calif. For information:
CIRCLE 206 ON READER CARD

tape spooler

The TS-405 high-speed tape spooler is able to wind tape at speeds up to 50 ips for photoelectric and other high-speed tape readers. It is bi-directional and automatically tracks tape in either the forward or reverse direction. The TS-405 is priced at $960. ELECTRONIC ENGINEERING CO. OF CALIF., AUTOMATION DIV., Box 58, Santa Ana, Calif. For information:
CIRCLE 207 ON READER CARD

desk top calculator

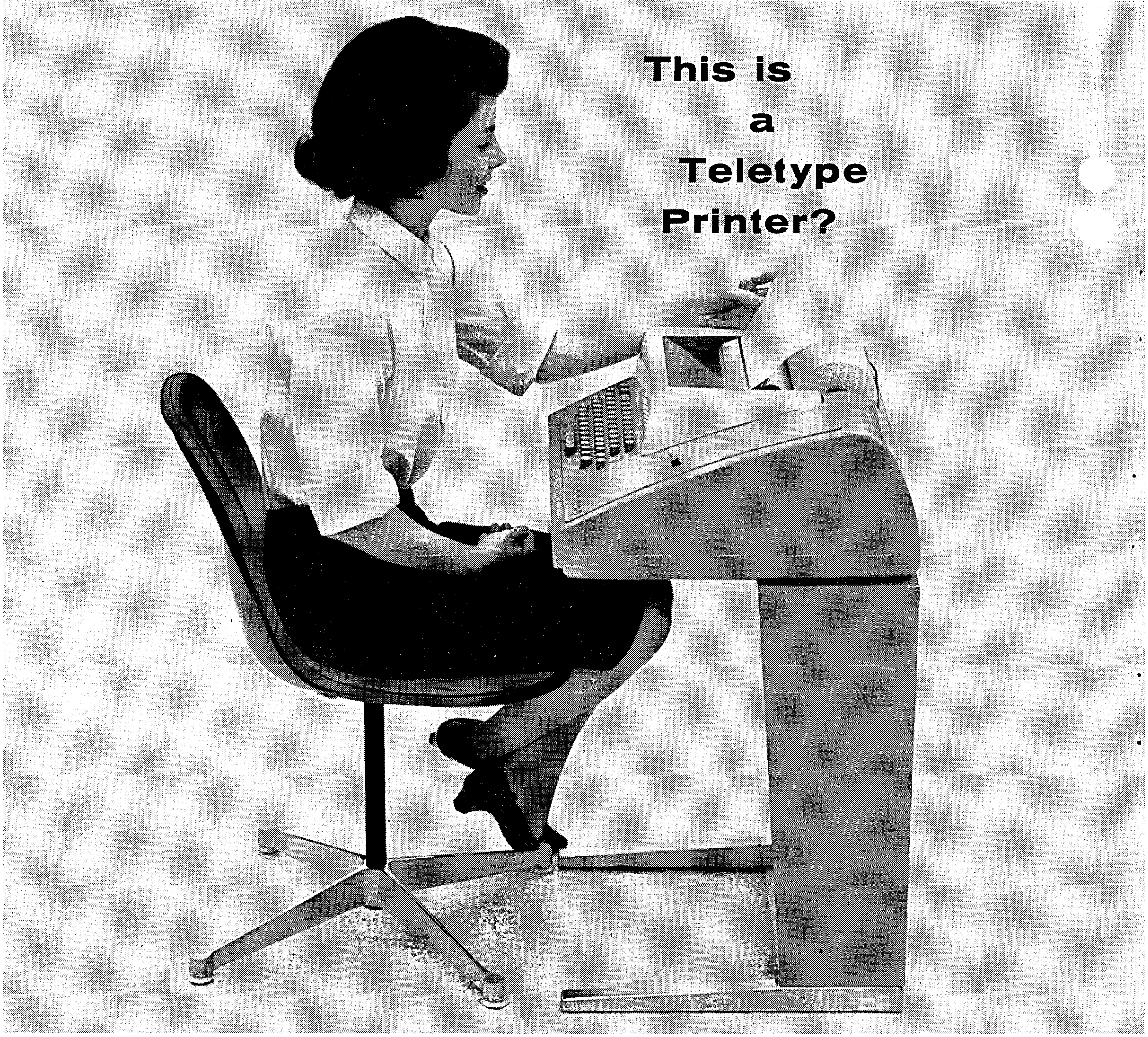
The IQ-213 features memory dials and completely automatic division and multi-factor multiplication. The calculator programs itself automatically and there is no need to reset constants on the keyboard. LITTON INDUSTRIES, BUSINESS MACHINES GROUP, 555 Mitchell St., Orange, N. J. For information:
CIRCLE 208 ON READER CARD

process control

The 412-M modular process control computer system has a central processor of digital, binary, fixed point with transistorized circuity. Magnetic core storage handles 4K 20 bit words. The system is capable of handling over 100 commands and has a one word operating time of 20 microseconds. GENERAL ELECTRIC CO., PROCESS COMPUTER SEC., Phoenix, Arizona. For information:
CIRCLE 209 ON READER CARD

digiac 3050

This semi-automatic logic and programming computer trainer permits the student to learn programming



Right! It's the Model 33, a completely new printer made by Teletype Corporation.

If you're wondering what happened to the more familiar one — we're still building that, too.

You see, we designed the Model 33 to fit a new set of requirements. It has a 4-row keyboard that eliminates shifting for figures and common punctuation marks. This saves operator time, cuts errors and means that every typist in your office can be a competent operator with only a few minutes' instruction.

The Model 33 uses a new 8-level message and data communications code. The 8-levels make it compatible with many computers and data handling systems.

Styling — while it's just part of what's new — is smart, functional, and as modern as a man-made satellite. Features include lighted push-button controls, easy paper insertion, and automatic station identification.

There are three models in this compact, economical "33" group — the send-receive printer with keyboard, the receive-only printer without keyboard, and the automatic send-receive set with self-contained tape punch and reader.

For users who require a 5-level communications code, there is a new Model 32 group with similar styling and economy.

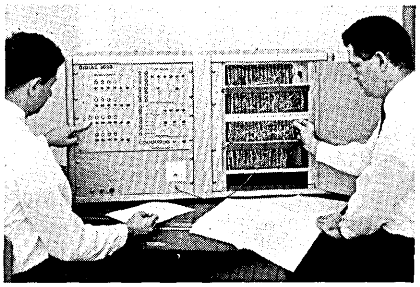
We would like to send you more information on this new Teletype message and data communications equipment. Contact: Teletype Corp., Dept. 81E, 5555 Touhy Avenue, Skokie, Illinois.

This equipment is made for the Bell System and others who require dependable communications at the lowest possible cost.

TELETYPE®
CORPORATION SUBSIDIARY OF *Western Electric Company* INC.

CIRCLE 37 ON READER CARD

NEW PRODUCTS . . .



techniques by actual usage. The 3050 contains a control unit and an arithmetic unit, which is a parallel 4-bit adder using Diode Logic, NAND, Transistor Logic and NOR. The unit is priced at $2,495. DIGITAL ELECTRONICS, INC., 2200 Shames Drive, Westbury, L. I., N.Y. For information:

CIRCLE 210 ON READER CARD

digital clocks

Elimination of noise problems is achieved with the use of this new standard feature for digital clocks, called RF filtering. CHRONO-LOG CORP., 2583 West Chester Pike, Broomall, Penna. For information:

CIRCLE 211 ON READER CARD

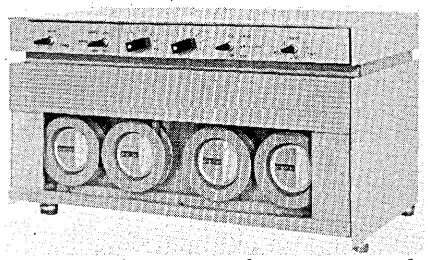
Selectric typewriter

The Selectric input-output typewriter, designed for use in conjunction with dp systems, measurement recording devices, automatic control mechanisms and in computer engineering, is able to automatically type at a rate of 15½ characters per second. Prices begin at $1,350. IBM CORP., ELECTRIC TYPEWRITER DIV., 545 Madison Ave., New York, N. Y. For information:

CIRCLE 212 ON READER CARD

mag tape system

The Micro Tape 555 features phase recording and a permanent timing tract, and can read, write and search



at a speed of 80 inches per second. Total storage of the 555 is 4 megabits per reel. DIGITAL EQUIPMENT CORP., Maynard, Mass. For information:

CIRCLE 213 ON READER CARD

display system

The transistorized D-200 provides continuous visual representation of data from a high speed computer. The D-200 may be driven by tape inputs or any general purpose digital computer. The system consists of an input buffer unit, display programmer unit, character generator, line generator and 12-inch screen monitor. DATRONICS ENGINEERS, INC.

CIRCLE 214 ON READER CARD

control computer

The Digicom/micro control computer has a basic memory of 2048 24-bit words including sign, expandable, add-substract time of 78 microseconds and standard 19-inch relay rack mounting 12¼″ high. Air conditioning is not necessary. The computer sells for approximately $20,000. DIGINAMICS CORP., 2525 E. Franklin Ave., Minneapolis 6, Minn. For information:

CIRCLE 215 ON READER CARD

accounting unit

The auxiliary accounting keyboard has been developed to provide adding and listing facilities, payroll data preparation for tape input to a computer and assembly of many types of operating and statistical data for the Flexowriter. FRIDEN, INC., 97 Humboldt St., Rochester 2, N. Y. For information:

CIRCLE 216 ON READER CARD

RAPID GROWTH CONTINUES AT CONTROL DATA

Many opportunities exist in several key professions

COMPUTER DIVISION
PALO ALTO AND NATIONWIDE LOCATIONS

SYSTEMS PROGRAMMING: Participate in the development of advanced programming systems, including compiler development, monitor and executive routines and language analysis. Digital computer experience and B.S. degree are required.

INSTALLATION SUPPORT: Technical representative at customer site. Responsibilities include orientation, training, programmer consultation and systems analysis for large-scale CONTROL DATA® 3600 and 1604 computer customers. Various locations in continental United States. Large-scale computer experience and B.S. degree are required.

TECHNICAL APPLICATIONS: Technical programming for applications such as nuclear reactors, general simulation, information retrieval, linear and non-linear programming.

PLEASE RUSH RESUME TO M. D. WALTER, **CONTROL DATA CORPORATION, 3330 HILLVIEW AVENUE, PALO ALTO, CALIF.**

GOVERNMENT SYSTEMS DIVISION
MINNEAPOLIS LOCATION

GOVERNMENT SYSTEMS DIVISION, because of rapidly expanding activity in program development for space and military applications, has long-range opportunities for programmer analysts and mathematicians with skills in the following areas:

REAL TIME PROGRAMMING FOR WEAPONS SYSTEMS CONTROL / DATA SYSTEMS ANALYSIS / SOFTWARE SYSTEMS DESIGN AND IMPLEMENTATION FOR REAL-TIME PROBLEMS/MAINTENANCE AND DIAGNOSTIC PROGRAMMING

A minimum of two years' scientific digital computer programming experience is required.

PLEASE RUSH RESUME TO E. C. THODE, **CONTROL DATA CORPORATION, 8100 34th AVENUE SO., MINNEAPOLIS 20, MINN.**

DATA CENTERS DIVISION
PALO ALTO, WASH., D.C., LONG ISLAND, MINNEAPOLIS AND OTHER LOCATIONS

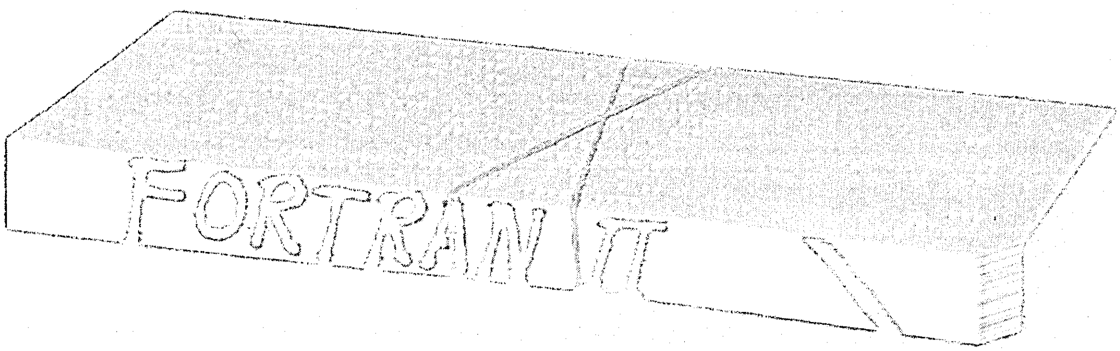
PROGRAMMER ANALYSTS: Immediate openings exist in the above locations. Analyze customer problems for computer application. Also be involved in sales support work and the preparation of programming proposals. A minimum of two years' experience with large-scale computer systems is required.

PLEASE RUSH RESUME TO J. C. CASSIDY, **CONTROL DATA CORPORATION, 8100 34th AVENUE SO., MINNEAPOLIS 20, MINN.**

CONTROL DATA
CORPORATION

An Equal Opportunity Employer

breakthrough in compilers . . .



1 COMPILER ☐ 83 CARDS ☐ 2981 INSTRUCTIONS

A new method of compiler construction has been developed *and tested* by the DIGITEK CORPORATION. This method drastically reduces the time and cost of compiler production. A case history:

☐ SDS 900 FORTRAN II ☐ IBM compatible ☐ all statements ☐ one pass ☐ 4K core ☐ no auxiliary memory ☐ room for 300 symbols and labels ☐ seven times faster than comparable compilers ☐ efficient machine code ☐ backward DO loops ☐ nested subscripts ☐ mixed mode expressions ☐ new computer ☐ minimal machine configuration holds entire compiler ☐ no reloading between compiles ☐ good diagnostics ☐ fixed price contract ☐ delivered on time

This method is applicable to any programming language for any computer. FORTRAN IV and JOVIAL systems are under construction. Research is being conducted in the areas of functional hierarchies, communication stacks predictive scans, status control and tree algorithms. Compiler customers include SDS, HUGHES, and DAYSTROM.

Prices start at $30,000, delivery times start at six months.

DIGITEK CORPORATION

Compiler Systems

1626 CENTINELA AVENUE INGLEWOOD, CALIF. PHONE: 213-678-2854

DATAMATION

COMPONENT PRODUCTS

logic modules

The 1000 series digital logic modules feature switching speeds to 10 megacycles, operating temperature range from −55° to +125°C, are transistorized, and interchangeable. SOLID STATE ELECTRONICS CORP., 15431 Rayen St., Sepulveda, Calif. For information:

CIRCLE 217 ON READER CARD

modular core memories

Modular Isodrive Memory Series are available in sizes from eight bits to 80 bits and can operate from −30°C to ±80°C without special heating for current compensating circuits. ELECTRONIC MEMORIES, INC., 9430 Bellanca Ave., Los Angeles 45, Calif. For information:

CIRCLE 218 ON READER CARD

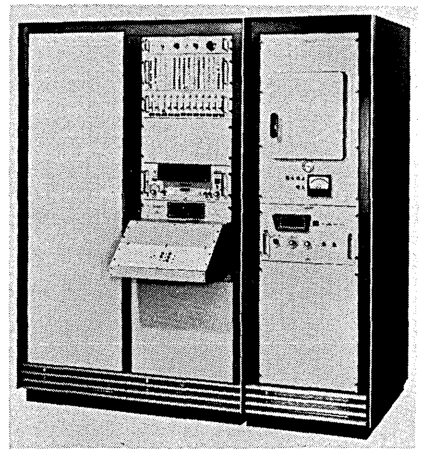
data acquisition filters

These bandpass filters exhibit constant time delay with respect to frequency and may be custom designed to meet exacting specifications. ORTHO FILTER CORP., 7 Paterson St., Paterson 1, N.J. For information:

CIRCLE 219 ON READER CARD

transistor tester

TACT (Transistor and Component Tester) provides test parameters up to 500 volts and 25 amperes. Pulse duration can be varied from 100 to 500 microseconds, repetition rate from 2 to 100 pps. Basic TACT prices begin at $40,000. TEXAS INSTRUMENTS INC., INDUSTRIAL PRODUCTS GROUP, 3609 Buffalo Speedway, Houston 6, Texas. For information:



CIRCLE 220 ON READER CARD

DATAMATION

NEW LITERATURE

AUDIO-VISUAL AIDS: This 24-page catalog includes a compilation of over 150 films available in data processing, systems and automation fields. Short descriptions, running times, sources and fees are included. Cost of the catalog is 25¢. DPMA INTERNATIONAL HEADQUARTERS, 524 Busse Highway, Park Ridge, Ill.

LP-1: This 22 page booklet offers a complete description of a linear programming system for the IBM 1401/1410. Topics discussed include operation, running time, data input, monitoring features, input error checking, and options for the LP-1. CENTURY, INC., 52 Vanderbilt Ave. New York 17, N.Y. For copy:
CIRCLE 130 ON READER CARD

PERT PROGRAMMING AIDS: An illustrated brochure on a magnetic visual system includes information on the development of various shaped magnetic elements, arrows, and programming boards useful in the establishment and use of a program network. METHODS RESEARCH CORP., 103 Willow Ave., Staten Island 5, N.Y. For copy:
CIRCLE 131 ON READER CARD

PERT/CPM PROGRAMMED INSTRUCTION: This portfolio for self instruction on PERT includes built-in sliding mask for step-by-step presentation. This 700-step program is priced at $27. ENTELEK, INC., 42 Pleasant St., Newburyport, Mass.

420 OPTICAL READER: This 20-page booklet explains the operation and lists the specifications of the 420. Features of the reader are offered. NATIONAL CASH REGISTER CO., PRODUCT INFORMATION, Dayton 9, Ohio. For copy:
CIRCLE 132 ON READER CARD

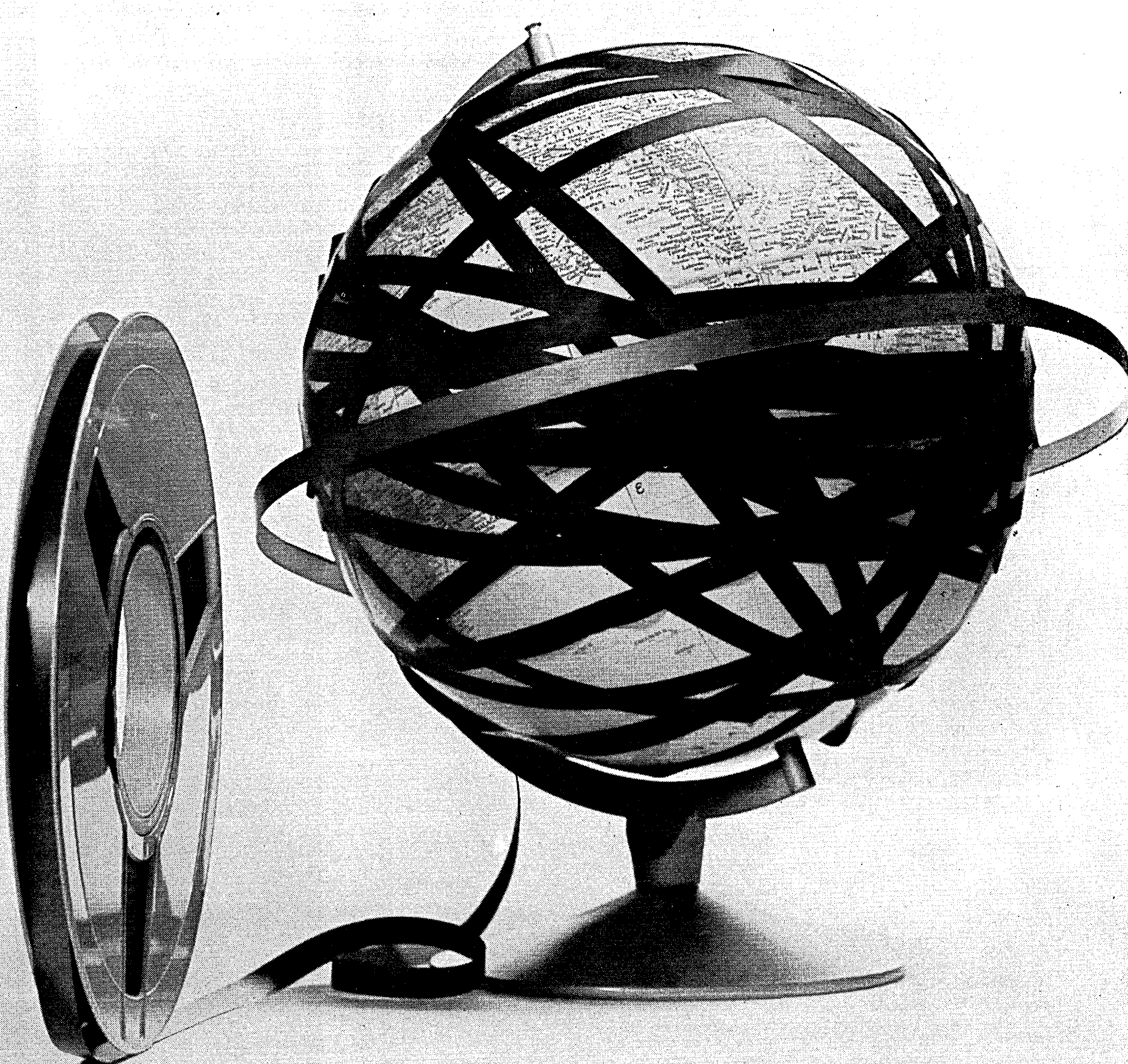
MULTIPLE TAPE LISTER SYSTEM: Features, a general description and specifications of this system are offered in a four-page bulletin. ANELEX CORP., 150 Causeway St., Boston 14, Mass. For copy:
CIRCLE 133 ON READER CARD

TRANSDATA 944: A descriptive summary of this microfilm printer-plotter includes typical applications, performance highlights, description, optional features and general characteristics. Illustrations are offered. BENSON-LEHNER CORP., 1860 Franklin St., Santa Monica, Calif. For copy:
CIRCLE 134 ON READER CARD

MEMORY TEST SYSTEMS: An illustrated catalog describes memory core testers for research and production applications, memory testers for coincident current and word address planes and stacks, and memory exercises for testing complete memory systems. DIGITAL EQUIPMENT CORP., 146 Main St., Maynard Mass. For copy:
CIRCLE 135 ON READER CARD

DATA REDUCTION: This six-page folder on the GADRS-4 contains information on the basic control console, program control, available outputs and input mechanisms and illustrations. GERBER SCIENTIFIC INSTRUMENT CO., P. O. Box 305, Hartford, Conn. For copy:
CIRCLE 136 ON READER CARD

SEQUENTIAL CARD LISTING SYSTEM: This 56-page catalog on the Compos-O-Line includes operating instructions, methods of card preparation, line composition devices, card handling procedures, camera operation, and film and negative processing. FRIDEN, INC., 97 Humboldt St., Rochester 2, N.Y. For copy:
CIRCLE 137 ON READER CARD



"Join with us and rule the world"

Computers are replacing the physical endeavors of mankind in kaleidoscopic fashion. . . Undeniably, ours is more and more a cerebral society . . . You see, the last frontier is no longer geographical; rather, it is intellectual . . . once pioneered to the ultimate, all else will fall into place . . . So that in the context of manual capabilities, *people may very well be on the way out* . . . But one area of human endeavor remains inviolate to the pioneer. We refer to computer problem-solving and its inseparable corollary: computer programming . . . Therefore, of critical importance to those of you with computer programming experience is your choice of the company with whom you elect to travel during this journey into the land of the gray . . . Before making a final decision, consider Computer Concepts, Inc., one of the nation's fastest growing computer consulting firms . . . With us you will be free to stretch your creative limbs in such areas of information processing as machine translation, computer efficiency studies, computer applications programming, management information systems and operations research . . . Salaries range to $20,000 annually, relocation expenses paid. Send resume to: **COMPUTER CONCEPTS, INCORPORATED,** 1012 14th Street, N.W., Washington, D.C.— Branch offices located in Los Angeles, California, and New York City.

CIRCLE 81 ON READER CARD

DATA PROCESSING

Computing Engineers for research and development of mathematical models. Degree in math or physics with two years of experience.

Applied or Utility Programmers to program integrated data processing systems applied to EDPM 709-7090-1401. Degree in math or accounting or equivalent in directly related experience.

Please write to:

NORTH AMERICAN AVIATION, INC.
The Professional & Technical Employment Office
Box DAMA-596
4300 East Fifth Avenue
Columbus 16, Ohio

Attn: H. Keever, Manager

An equal opportunity employer.

COLUMBUS DIVISION

NORTH AMERICAN AVIATION

FAIRCHILD-STRATOS

Aircraft-Missiles Division

has immediate openings in SCIENTIFIC COMPUTER PROGRAMMING

This aggressive, dynamic organization is moving boldly forward with important aerospace programs. Anticipated changeover to an IBM 7040 Computer, and an increased backlog of space projects has created immediate openings for scientific programmers on both the senior and intermediate level.

Computer problems to be worked in the following areas:

Aerodynamics	Stability and Control
Aircraft Performance	Structural Dynamics
Flight Dynamics	Thermodynamics
Information Retrieval	Trajectories
Numerical Analysis	

Requirements for these positions include at least two years experience in the analysis of engineering problems and programming for a large scale computer. A degree in mathematics, engineering, or some related field is required.

In addition, an opportunity exists for a

PROGRAMMING SYSTEMS MAN

To maintain the 7040 Monitor System and develop programming standards. In addition to a general background in computing this man should be familiar with the design of monitors, compilers, and assemblers.

These positions offer full benefits program . . . liberal relocation allowances . . . unusually attractive compensation policy. For information on a personal interview, forward background data including salary requirements to:

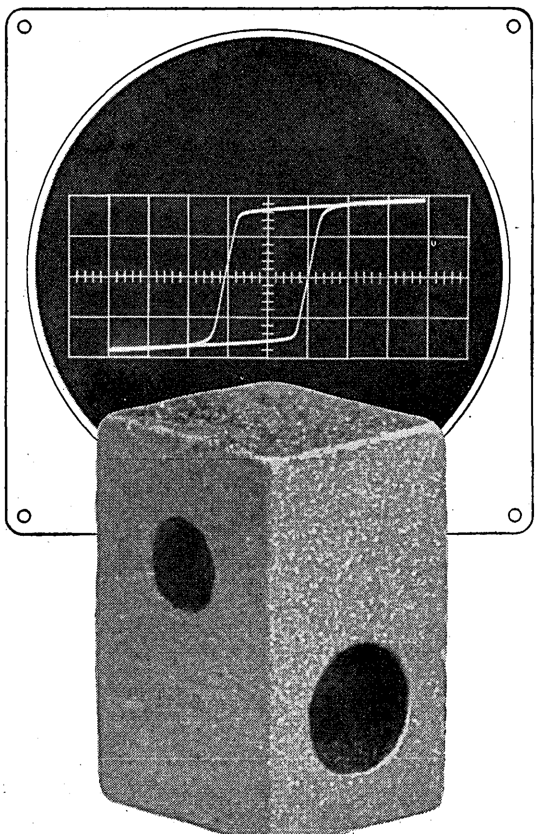
MR. C. A. WEBB, Manager, Professional Relations Department A-5

FAIRCHILD STRATOS

AIRCRAFT—MISSILES DIVISION

Hagerstown, Maryland
An Equal Opportunity Employer

CIRCLE 82 ON READER CARD



2 megacycle non-destructive readout from BIAX® memories

Now—increase system programming versatility and gain the advantages of micro-programming techniques with the new 2 megacycle BIAX memory. Two of the industry's newest systems, the Collins C-8401 Data Central and the Packard-Bell PB-440 Computer use BIAX memories to achieve advanced levels of performance.

Advanced BIAX memories and memory stacks are available in both non-destructive read/write and instruction memory versions. Early delivery is offered on standard modules which range from 128 words to 1024 words—up to 48 bits per word. Custom memories can also be supplied.

BIAX memories offer the designer the unique combination of multi-megacycle random read cycling, true non-destructive readout insuring reliability and permanence of vital storage, and extremely low operating power levels. These characteristics can significantly enhance the performance of both ground-based and aerospace data systems.

BIAX memories find ready application in program, control sequence and statistical storage; associative memories; automatic checkout equipment; digital simulators and training devices; character generators; and process control systems.

For technical brochure, specific application information, or price and delivery details, direct inquiries to:

Marketing Manager, BIAX Memory Systems

®Registered Trademark

AERONUTRONIC DIVISION *Ford Motor Company* GENERAL PRODUCTS GROUP

FORD ROAD, NEWPORT BEACH, CALIFORNIA

See the new 2 megacycle BIAX memory in operation at the SJCC in Detroit, May 21, 22, 23 – Booth Number 220

CIRCLE 46 ON READER CARD

For EAI CIRCLE 47 ON READER CARD


HYDAC 2000:
The first computer capable of real-time simulation of complete missile and space vehicle systems. Write for the facts today.
EAI
ELECTRONIC ASSOCIATES, INC. Long Branch, New Jersey

THE 235 — THIRD OF THE COMPATIBLES

a new entry from G.E.

Third in G.E.'s line of program-and-peripheral-compatible general purpose computers is the 235, announced late last month. The new hardware features a dual Controller Selector which the manufacturer claims can double system peripheral performance.

Basic features of the 235 include: code compatibility with the 215 and 225; 5.93 usec memory cycle time; core memory of 4, 8 or 16K 20-bit words; priority control of accesses to memory with optional dual controller; GECOM, WIZ and FORTRAN software; automatic program interrupt permitting concurrent operation of two or more unrelated programs (optional); automatic address modification.

The 235 is a single address, binary machine with decimal capabilities. The instruction repertoire includes more than 300 defined instructions. Both floating point and decimal arithmetic are available as optional features.

Execution times, including access:

Fixed point add (binary or decimal): 11.8 microseconds
Floating point add (normalized or unnormalized): 29.6 - 35.6 microseconds

The basic GE-235 system consists of three functional modules: Central Processor; Core Memory; and Priority Control Section. The input/output system is compatible with the 215 and 225 input/output systems. As with the two other systems, the GE-235 peripherals are fully buffered and can be operated concurrently.

The 235 system was designed to use the existing 225 peripheral set to allow system upgrading without major installation changes.

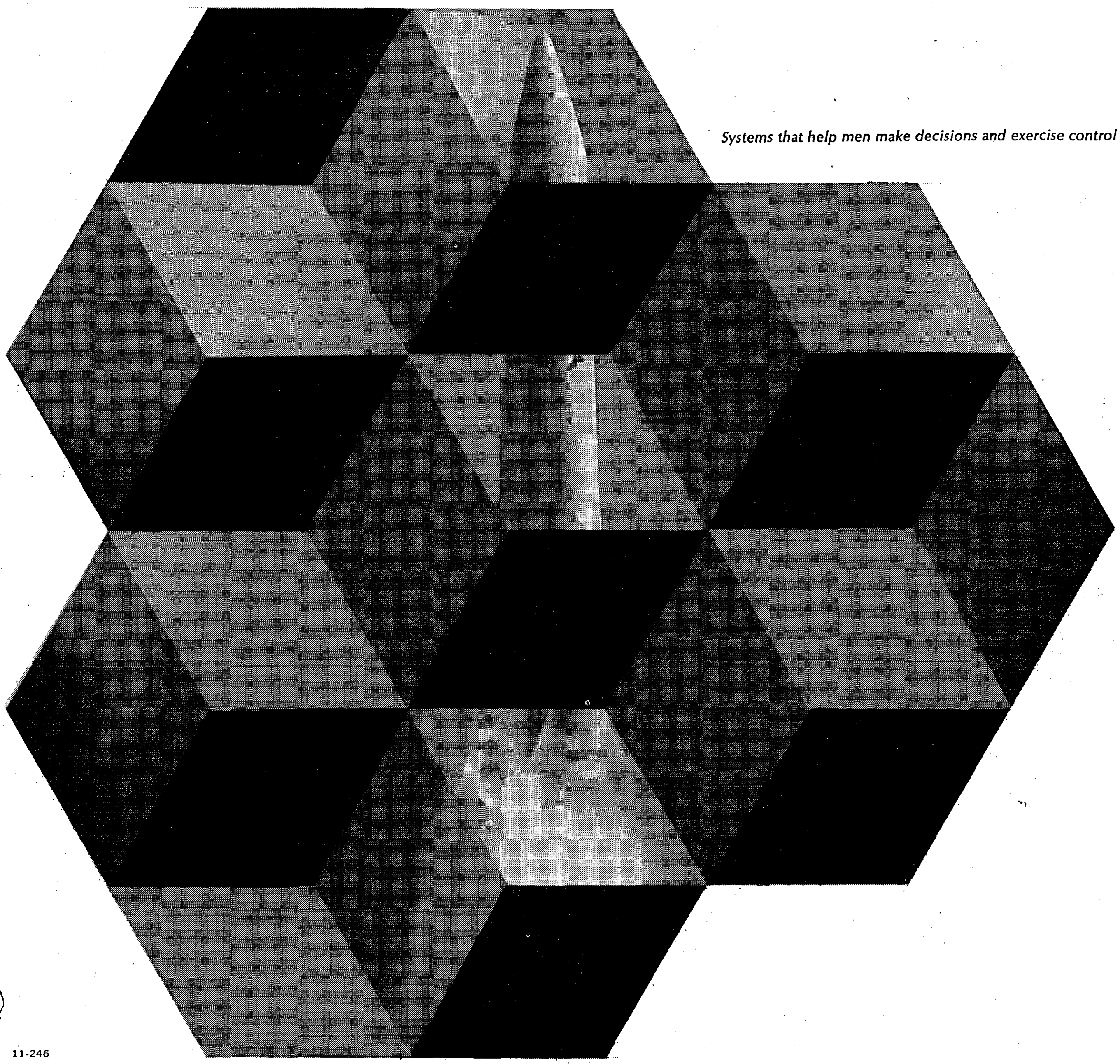
In order to share a common peripheral set and achieve greater simultaneity, the peripheral controllers are divided into three priority groups. Each group operates on an independent 17.8 microsecond input/output cycle for data and instruction transfer. The three memory priority groups are phased in such a manner that after every three memory cycles each priority group has had the availability of one memory cycle.

One memory priority group serves the Card Reader and Card Punch Controller; the other two memory priority groups each serve separate peripheral Controller Selector busses. All memory cycles which are not in use by the peripherals are available to the Central Processor.

The Central Processor performs all the computing and logical functions of the system, including address modification and programmed initiation of input/output operations. The 235 operates in the parallel binary mode. Optional decimal circuitry permits decimal addition and subtraction; the optional Auxiliary Arithmetic Unit permits direct operation on normalized and unnormalized floating point operands and on double-precision fixed-point operands.

The 235 instruction repertoire is bit compatible with the GE-215 and -225 machine language. Most instructions are executed in 2 or 3 word times; that is, in 11.8 or 17.8 microseconds. ■

CIRCLE 105 ON READER CARD



ADVERTISERS' INDEX

SPEED is many things

Actual working speed—meaningful speed—in a computer is a combination of many factors. Consider our ASI-210. Its short 2-microsecond total memory cycle is important, but even more important are the design features that make it possible to use this speed in normal operations.

ASI-210 is proving in actual use that its overall computer speed is unsurpassed in its size and price range. We planned it that way by building in adequate memory capacity, convenient instructions, a full complement of arithmetic commands, fast and easy communication with peripheral devices — plus user-oriented software.

The result is an exceedingly useful and versatile computer. It is available at an eminently reasonable price. We'd like to tell you more about our ASI-210, and we'd like to tell you about new resources and new capabilities that have become available through our recent acquisition by Electro-Mechanical Research, Inc., Sarasota, Fla.

Write for our ASI-210 bulletin, or ask an ASI representative to tell you how the newly combined technical resources of ASI and EMR can solve your specific computation problem. Contact Advanced Scientific Instruments, 5249 Hanson Court, Minneapolis 22, Minn.

ASI-210 FEATURES: Completely solid state, 2-microsecond total memory cycle, parallel operation, 21-bit words, 4096-word randomly addressable core memory expandable in modules to 32,768 words, .6-microsecond add time, 50-microsecond multiply time, high speed I/O simultaneous with computation.

Trapped program interrupt, indirect addressing, multiple index registers, built-in muliply and divide, Fortran compiler, assembler and complete diagnostic routines, paper tape system included.

Power consumption under 1750 watts, standard 110/220 volt 60 cycle AC. No special temperature or humidity control needed. ASI equipments are IBM compatible.

PERIPHERAL EQUIPMENT: Options include magnetic tape units, typewriters, punch card units, line printers, additional assembly registers, converters and plotting equipment.

EXCITING OPPORTUNITIES for computer-oriented professionals continue to exist in our rapidly growing organization. Write or call today.

ASI

MORE COMPUTATION PER DOLLAR