DATAMATION

64

August

COBOL

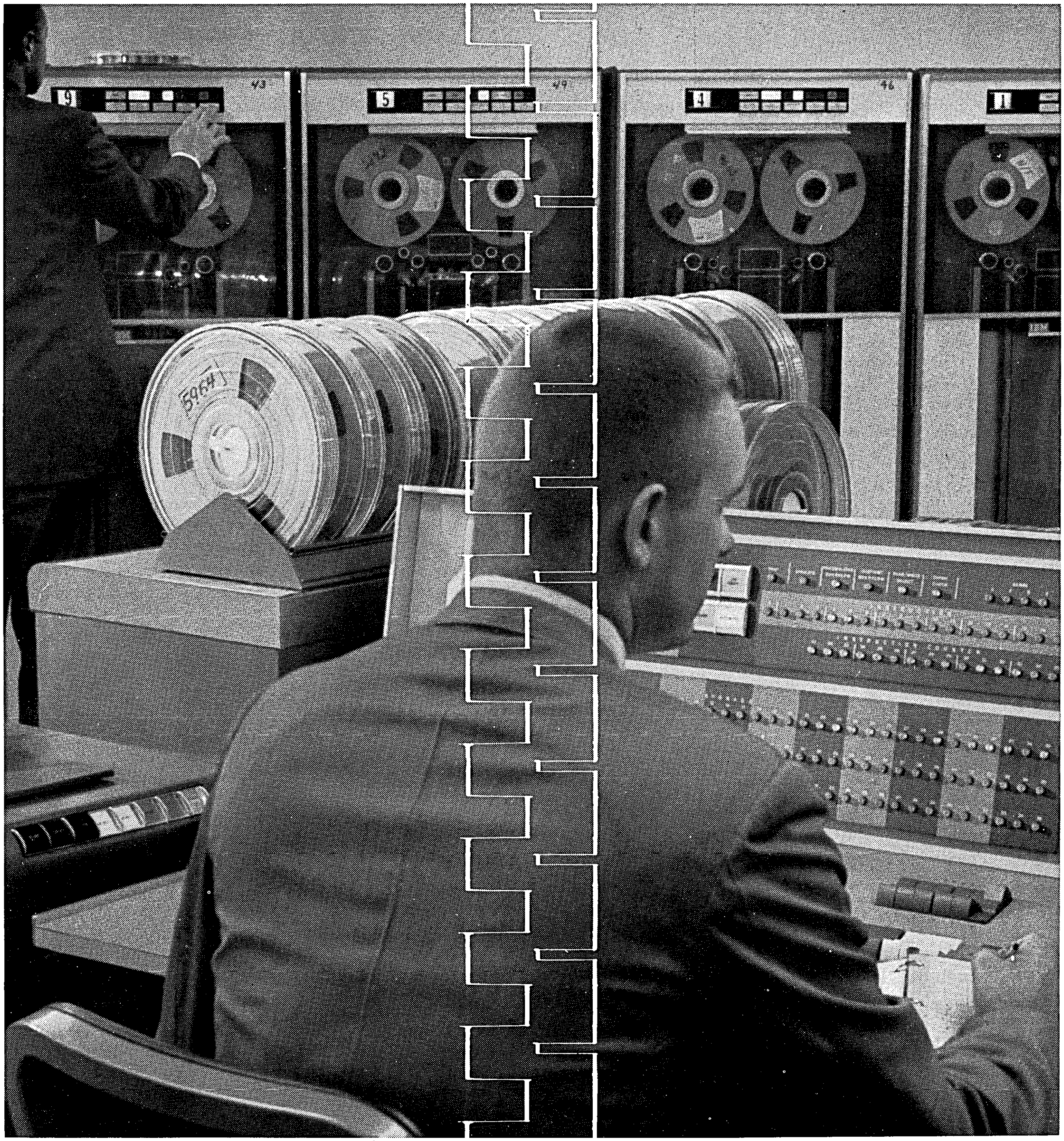
FORTRAN

NPL

IMPLICIT

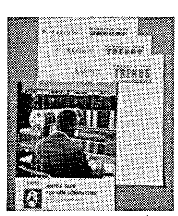
programming languages

PHOTO COURTESY OF LOCKHEED CORP. © AMPEX CORP. 1963



Now: who's got news for everyone with an IBM computer system? AMPEX

The news is inside an eight page booklet. It tells the what, the why and the how of Ampex computer tape—the tape that provides superior performance in IBM computer systems. If you think you might find the booklet helpful, just write and ask for it. Also, we'll put your name on our mailing list and regularly send you our informative periodical, "Tape Trends." It's a good way to keep abreast of the fast changing tape technology. In it, the latest tape developments are clearly explained by Ampex tape

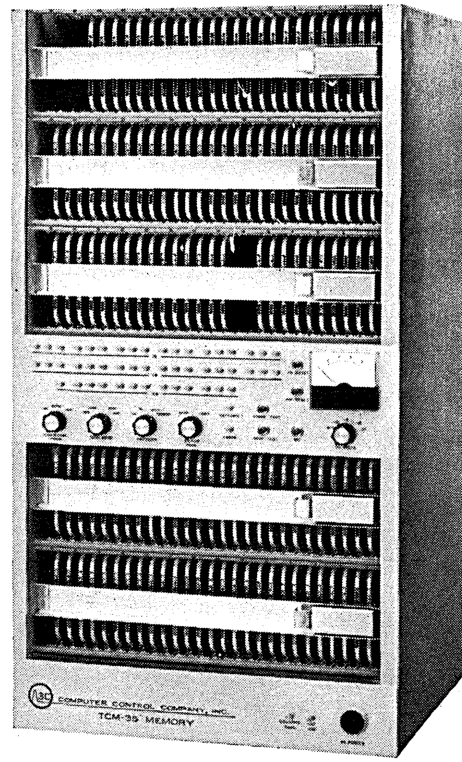


AMPEX

experts—the same experts who application-engineer Ampex tape to your system. This is just one of the many ways we assist you in obtaining maximum system efficiency. In addition to engineering the tape to your system, Ampex digitally checks each reel from end to end, and guarantees its performance. Write for free booklet, "Ampex Tape for IBM Computers," and your copies of "Tape Trends." Ampex Corporation, Redwood City, California. Sales and service engineers throughout the world.

CIRCLE 1 ON READER CARD

NEW
1.4-2 μSEC
CORE MEMORY



General Purpose Front Access TCM-35

New TCM-35 family of coincident current core memories offers 1.4 to 2 μsec full cycle times, 1.2 μsec maximum half cycle time, and 0.75 μsec maximum access time. Capacities up to 8,192 words; word lengths to 36 bits. All silicon circuitry. Advanced system design. Extremely compact packaging. Low input power. System height is 24¼ inches to 35 inches, including self-contained power supply, depending upon storage capacity and options. TCM-35, like the 5 microsecond TCM-32, has pull-out, front-of-rack access. Operating temperature is from 0°C to +50°C with broad margins. 3C quality is built in. Modular design allows custom selection of desired performance and interface characteristics from an extensive list of standard options at volume production prices. Write for TCM-35 brochure (for 5 μsec memories, ask for TCM-32 brochure).

BASIC TCM-35 FEATURES*

Choice of Input/Output
Logic Levels and Polarity

OPERATING MODES
Clear/Write
Read/Restore
Load
Unload
Read/Modify/Write

CONTROL INPUTS
Address Register Clear
Information Register Clear
Full Cycle/Half Cycle
Hold Address

CONTROL OUTPUTS
Memory Busy
Information Available
End of Cycle

NON-DESTRUCTIVE START-UP/SHUT-DOWN
BUILT-IN MARGINAL TEST

* no extra cost

VOLTAGE FAILURE SENSING
BUILT-IN COOLING SYSTEM

STANDARD OPTIONS**

INFORMATION REGISTER CONTROL
Shift Input
Count
Partition (up to 4 zones)

ADDRESS REGISTER CONTROL
Sequential
Sequential Up/Down
Sequential Interlace
Shift Input
Marker Pulses (up to 3)

DATA OUTPUT
Address Register

MEMORY CLEAR
MEMORY TESTER
VOLTMETER
REGISTER DISPLAYS
400 CPS POWER INPUT

** at extra cost

3C DISTRICT SALES OFFICES: NEEDHAM, MASS.; SYRACUSE, N.Y.; COMMACK, L.I., N.Y.; LEVITTOWN, PA.; CLEVELAND, OHIO; SILVER SPRING, MD.; DES PLAINES, ILL.; ORLANDO, FLA.; ALBUQUERQUE, N.M.; PALO ALTO, CALIF.; LOS ANGELES, CALIF.; HOUSTON, TEX.; HUNTSVILLE, ALA.

COMPUTER CONTROL COMPANY, INC.
OLD CONNECTICUT PATH • FRAMINGHAM, MASSACHUSETTS

CIRCLE 4 ON READER CARD

WHEN SHOULD YOU SEND AND

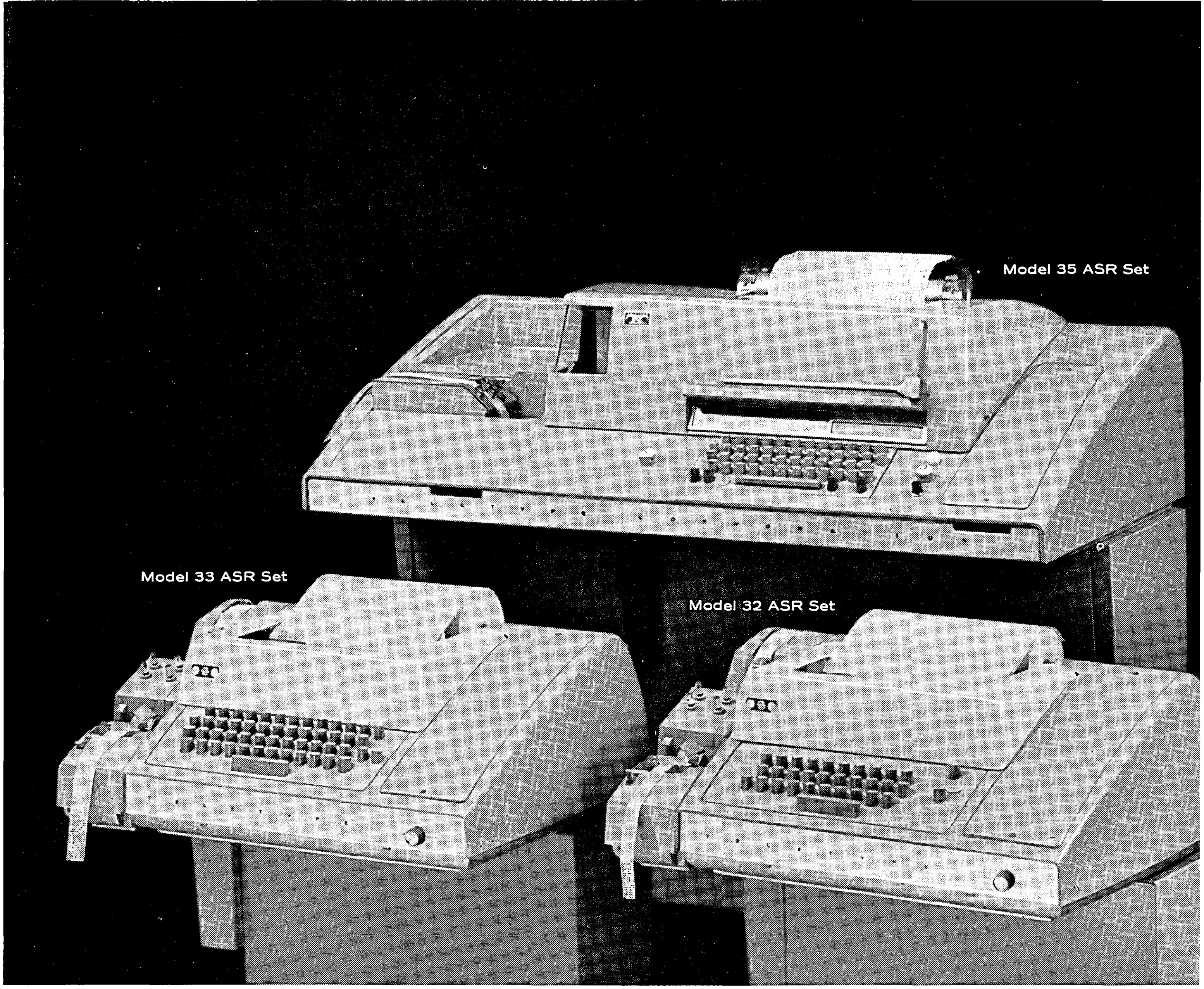
Many firms are stalled behind a communications barrier caused by increased paperwork. Thus, simplifying and speeding up communications will help make operations more efficient and profitable. Firms, using Teletype page printers to handle routine communications, could speed things up considerably by adding the versatility of automatic transmission with punched paper tape.

Three advantages of automatic send-receive sets. Operators can punch messages and data off-line providing errorless tape for later transmission. Messages and data can be collected from many sources at different times and transmitted on-line more economically at maximum speed in one continuous message. Punched tape makes it possible to store messages or basic data that can be used again and even combined with variable data to save retyping.

Three models to choose from. There are three different Teletype automatic send-receive sets to satisfy your communication needs. The Models 32 and 33 are the most economical to use where traffic ranges from normal to light. The Model 35 is designed for handling a much larger volume of messages and data, as well as offering increased versatility for on-line and off-line communications.

Communicates with business machines. Actually, punched paper tape prepared on a Teletype automatic send-receive set can do much to automate communication procedures. These ASR sets "speak" the same language as many business machines and computers because they operate on an 8-level code, which conforms to the newly approved American Standard Code for information interchange.

How are Teletype ASR sets used? Teletype ASR sets speed up order processing by reducing errors and duplication of paperwork between departments; improve warehousing, purchasing procedures, and distribution methods; and link business machines with existing channels of communication. For example:



RECEIVE AUTOMATICALLY?

Steel manufacturers use a computer to analyze samples of high alloy steel during production runs to insure maintaining the proper percentage of elements. Teletype ASR sets provide rapid transmission of this information from the computer location to the steel making shops.

Auto manufacturers use ASR sets as well as other Teletype machines in their operations to alert each assembly point, insuring that proper equipment and accessories arrive where needed, when needed.

Other firms are using Teletype ASR sets in departments throughout their company to send information or problems to a centralized computer, with answers sent back at maximum speeds.

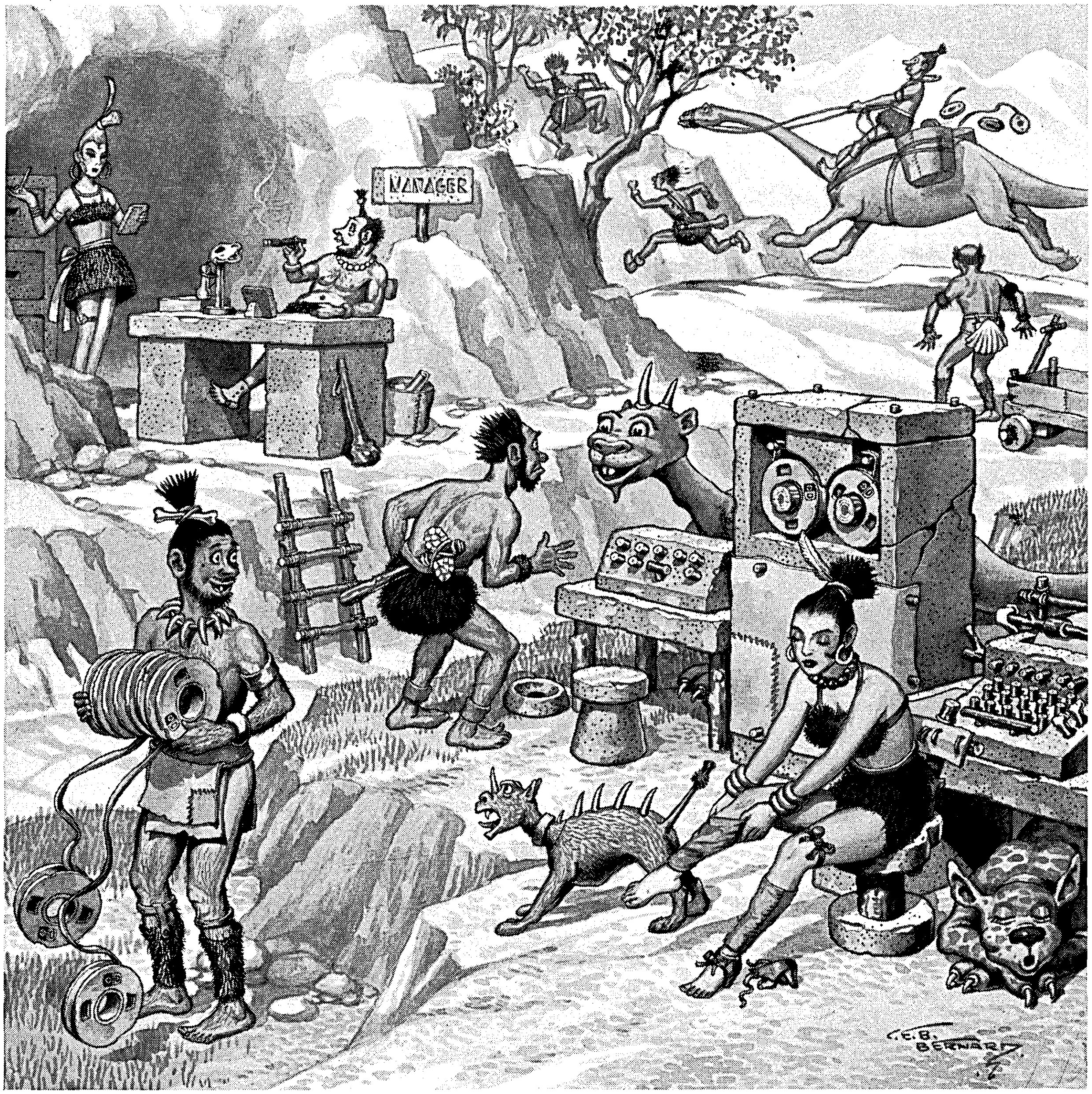
Thus, you can see how essential this kind of equipment is to fast, real-time data communications—and why it's made for the Bell System and others who require dependable communications at the lowest possible cost.

When should you send and receive automatically? The answer more than likely is right now! Continually expanding companies need faster, more efficient communications to control the growing flow of paperwork. Write for a brochure telling how Teletype equipment not only speeds the flow of messages but also plays an important role as terminal equipment in integrated data processing systems. Teletype Corporation, Dept. 81H, 5555 Touhy Ave., Skokie, Illinois 60078.

CIRCLE 5 ON READER CARD

TELETYPE

AN OFF-BIT HISTORY OF MAGNETIC TAPE...one of a series by Computape



As most any student of history will tell you, credit for developing the first successful technique for magnetizing computer tape must go to Pulchritudinous Paula Piltdown, whose sure-fire method is exhibited here. Merely by adjusting her seams she exerted sufficient directional magnetic force to turn a man's head at 56 paces — and permanently magnetize all the tape he was carrying at the same time.

In the 546,312 years since, no one has devised a method for magnetizing tape that's half as much fun.

But Computape has one that's even surer-fire.

First, we clean the Mylar backing of the tape itself. Then, we apply a primer coat. *Carefully.* (To just less than one one-millionth of an inch, to be exact.)

Then we apply Computape's exclusive, extra-heavy duty magnetic coating. *Carefully.* (To a tolerance of 25 millionths of an inch to be exact.)

Then we test. Every inch of every reel — whether it's 7, 8, 9, 10, 16 channel or full-width. *Carefully.* (Any defect large enough to cause even a 50% drop in signal of a *single bit* is cause for rejection, in fact.)

The result: Computape — computer tape so carefully made it gives you 556, or 800, or (if you want it) 1000 bits per inch. For the life of the tape.

Now. If Computape can write that kind of computer tape history, shouldn't you be using it?

*DuPont trademark

COMPUTRON INC.
122 Calvary Street, Waltham, Massachusetts

COMPUTAPE — product of the first company to manufacture magnetic tape for computers and instrumentation, exclusively.

CIRCLE 6 ON READER CARD

DATAMATION 64

the automatic handling of information

volume 10 number 8

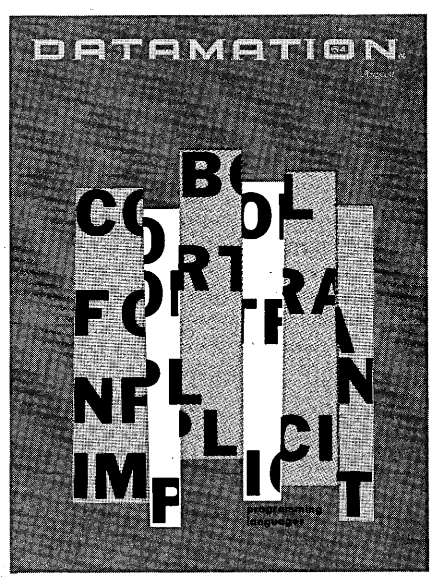
EDITORIAL OFFICES

1830 WEST OLYMPIC BLVD.
LOS ANGELES, CALIF. 90006

Feature Articles

Departments

THIS ISSUE — 47,882 COPIES



Cover

Programming Languages, those awkward but necessary links between man and machine, highlight this month's issue, which features articles on FORTRAN and COBOL. Symbolizing the never-ceasing, ever-changing search for the universal communications link between humans and computers is this month's cover, designed by Art Director Cleve Boutell.

Circulation audited by Business Publications Audit

Member, National Business Publications

DATAMATION is circulated without charge by name and title to individuals who qualify by our standards and are employed by manufacturers or users of automatic information-handling equipment in all branches of business, industry, government and the military. Available to others by subscription at the rate of $15.00 annually; single issues (when available) $1.50. Foreign subscriptions are on a paid basis only at a rate of $25.00 annually. No subscription agency is authorized by us to solicit or take orders for subscriptions.

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

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 Street, New York, New York 10017 (MUrray Hill 7-5180). Editorial offices, 1830 W. Olympic Blvd., Los Angeles, California 90006. Published at Chicago, Ill. Controlled Circulation paid at Columbus, O. Form 3579 to be sent to F. D. Thompson Publications, Inc., 205 W. Wacker Dr., Chicago, Ill. 60606.

Printed by Beslow Associates, Inc.

Command and Control: Why say 100010110 when you mean 278?

Today's military commander lives with frustrations that would have caused von Clausewitz to turn in his sword. His mission is decision-making. His decisions are based on alternate choices derived from the best possible data. The only possible way he can get this data with the speed and accuracy required is through a complex maze of computer systems which do not even speak his language. Who can solve this problem of man/machine communication? Not the hardware manufacturer. Nor the military. It's a job for an independent interpreter.

By Werner L. Frank

The problem of man vs. machine in military command and control systems communication is a problem of conflicting viewpoints. Too many of the men who are responsible for the development of highly sophisticated automatic data processing systems seem to believe that the world must inevitably be recreated in their machines' image. Nowhere is this vanity more prevalent than among those who seek to automate the Military. Their contention seems to be that it is far more logical (and consequently very necessary) for the Military to adapt to their machinery than to attempt to adapt their machinery to the Military. Instead of requiring the machine to communicate with the man, they are requiring the man to communicate with the machine. They are wrong.

WHAT MAKES THE MILITARY TICK?

The very root of the military profession is discipline. Without clear-cut, ascending levels of authority, the military establishment would crumble. What has this to do with military data processing systems? Plenty. Because it has much to do with the role of Commander. In crisis, each man and *each piece of equipment* under the Commander's authority must respond directly and immediately to his will, his needs, his methods, his commands. If he is to realize the full advantage of the computer-centered system at his disposal, he must not be required to use it through a frustrating chain of technicians and codes. He and his staff must be able to communicate with the machine directly — in the language they are accustomed to. To expect the Commander to adapt his requirements and decision-making procedures to the convenience of an electronic command and control system is not only foolish but potentially fatal.

WHAT MAKES THE HARDWARE MANUFACTURER TICK?

The men who design computers and computer-oriented hardware are, by their very nature, machine worshippers. This is as it should be. This is what drives the technology forward. But their near-total dedication to the form and function of the data processing system itself is in sympathy with neither the immediate needs and problems of their users nor the prevailing concept of evolutionary system upgrading. They are not ready to attack the man/machine language problem.

TYPICAL COMMAND AND CONTROL SYSTEM ASSIGNMENTS

Listed here are a few examples of the command and control software projects in which Informatics is engaged:

National Military Command System: programming and analysis in displays and other on-line aspects for the highest level of command and control system in the nation.

Office of Naval Research: two projects involving advanced Naval Tactical Data Systems: (1) system planning and analysis of the country's hardware and software technology for Systems of the 1975 era and (2) development of a design concept for advanced Marine Tactical Data Systems.

Rome Air Development Center: two projects for the Air Force: (1) the development of the programming system for a man/machine system involving an on-line interrogation console and display used in intelligence evaluations and (2) development of an Executive Control Program for a large-scale, multi-computer system for the development of military data processing techniques.

THE BEST SOLUTION: HIRE AN EXPERIENCED INTERPRETER

The only place where maximum value from a military command and control system can be reached lies about half-way between the producers of the hardware and their military customers. Neither of the two is adequately trained or objective enough to go even half of the way. Someone with full comprehension and appreciation of the problems and attitudes of both must stand in the middle. He must have experience with on-line group displays and display consoles and the software techniques that go with them. These include query languages and systems of operational programs responsive to the Commander on *his* terms. There are few firms with such experience. Informatics Inc. is one of them.

WHAT MAKES INFORMATICS INC. TICK?

We at Informatics Inc. represent the user. We work with the manufacturer towards getting more flexible hardware. At the same time, because command and control requires the closest relationship of man to machine, we help the user to employ his present hardware with the least possible deviation from his normal operating procedures.

Our staff is particularly well suited to the analysis and evaluation of command and control communication problems. Twenty members of our senior staff have been directly responsible for the technical direction of successful real time software programs, including some elaborate interrogation-display systems for such applications as intelligence data handling, own-forces evaluation, damage assessment, and weapon assignment. Further, we have developed some basic techniques that allow rapid and efficient tailoring of general-purpose computers and displays to specific military purposes. Informatics Inc., is the *only* independent software firm specializing in solving the real time problems and developing the on-line computer-centered systems of command and control.

If you would like to discuss our approach to command and control communications with the idea in mind of solving some of your own problems or perhaps of joining us in solving other people's problems, give us a call. The number is (202) 244-7102. Ask for Werner Frank, or any other member of our Washington, D.C., staff. We also have literature on our people and capabilities which we will be happy to send you on request. Address Department E, Informatics Inc., 15300 Ventura Boulevard, Sherman Oaks, California.

informatics inc.®
Department E
15300 Ventura Boulevard
Sherman Oaks, California 91403

Please send me your staff-authored article:

Military Command: A Challenge For Information Processing.

Name________________________

Address______________________

____

An equal opportunity employer

IBM reports to the industry

SYSTEM/360 includes universal code capabilities

The IBM SYSTEM/360, designed for a specific eight-bit code, also will accept any character code having *fewer* than eight bits.

This eight-bit structure (the extended binary-coded-decimal interchange code, or EBCDIC) is derived from the six-bit BCDIC. One of the most significant advantages of the eight-bit structure is the packing of two four-bit numerics into one eight-bit byte.

The 7-bit ASCII is to be expanded in SYSTEM/360 to a superset of eight bits. It will preserve the relationships and code properties of the seven-bit form.

Input/output equipment that will handle ASCII includes the 2400 Magnetic Tape units, the 2671 Paper Tape Reader, and the 2822 Paper Tape Reader Control. They will be available starting in April, 1965.

SYSTEM/360's central processing unit will handle any eight-bit character-set, although there are certain restrictions in the decimal arithmetic and editing operations.

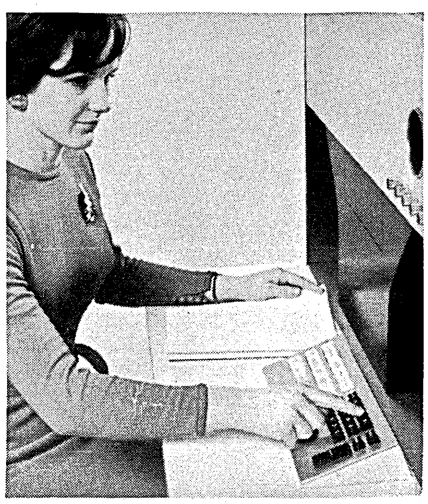
New Optical Scanner reads cash register tapes

IBM's new 1285 Optical Reader reads data from cash register or adding machine tapes directly into any of these computers—1401, 1440, 1460 or SYSTEM/360 (Model 30 or 40).

For all businesses, especially retailing, the 1285 solves the problem of entering raw sales data into a computer while the facts accurately reflect changing market conditions.

The 1285 features a high reading speed of up to 3,000 lines per minute. One reason for this high speed is its solid-logic circuitry—the same as in SYSTEM/360.

Another is a newly-developed "flying-spot" scanning technique, in which a beam of light from a cathode ray tube automatically scans several lines of print without tape movement.



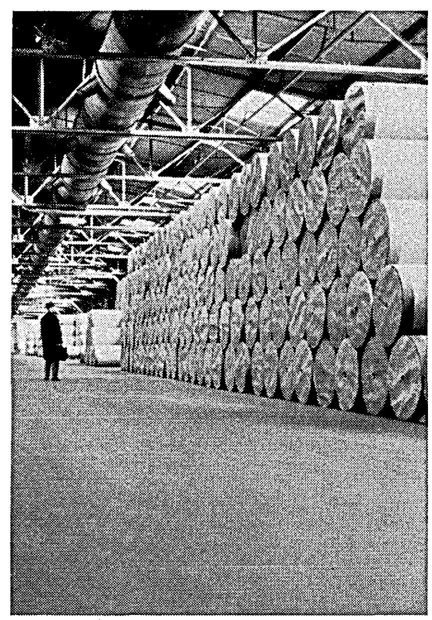
New IMPACT program saves time and money

It's the IBM 1311 IMPACT Computer Program Library for inventory management—an extension of the widely used IMPACT 1401-1405 program.

IMPACT assists the buyer in determining when, what and how much to buy. It helps achieve a more efficient balance between the cost of carrying inventory and the cost of purchasing and receiving. Properly used, it helps improve customer service, while keeping inventory at a minimum.

Library programs include editing, file initialization, estimating and control of joint-replenishment operations.

The IBM IMPACT Program can be used with the IBM 1401/1460-1311 or the 1440-1311 Data Processing Systems.



IBM 1026—new low-cost link to on-line Tele processing

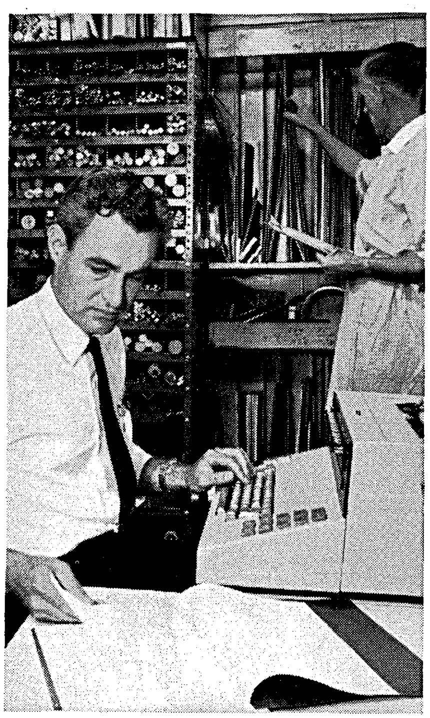
The new IBM 1026 Transmission Control Unit provides any user of a 1240, 1440, 1401 or 1460 with all the advantages of a remote, on-line Tele processing system—at a very low cost.

Large companies can make a modest start toward satisfying their data communications requirements. Expansion to SYSTEM/360 can be made as the need arises.

Smaller companies, too, can use the new 1026 to go on-line now—sooner and more reasonably than ever before.

Each 1026 Control Unit can handle one line, with multiple terminals on that line—either 1030's, 1050's, 1060's or 1070's. Up to four control units may be added to a system.

IOCS support for the IBM 1026 includes polling; addressing and receiving on all lines at the user's discretion; error detection; operation with direct data channel and disk; and time-shared operation of batch jobs with real-time processing.



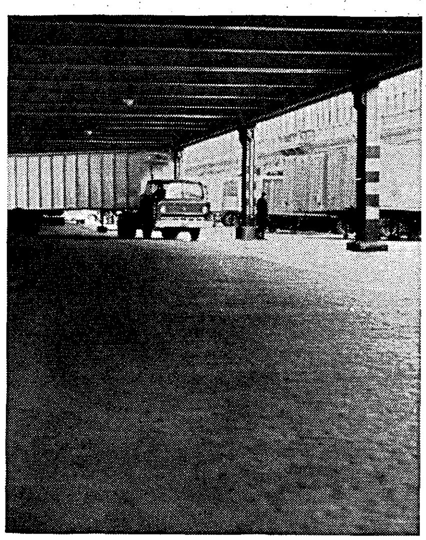
IBM freight program simplifies billing procedures

Revenue-accounting and control in the motor freight industry is now faster and more efficient, through a new IBM 1440 program.

This new Motor Freight Revenue Accounting program edits daily transaction data and prepares the daily transaction register. At the same time, it posts revenue by freight bill number.

It prepares customer statements and simultaneously posts customer accounts-receivable balances. It also edits cash-remittance source data.

Whether the motor-freight carrier uses a centralized statement-and-collection method of billing or a decentralized terminal-level method, doesn't matter. In either case, the program provides effective control of freight bills.



New operating system aids textile-finishing

Textile industry managers can now exercise comprehensive control over all of their operations, by employing the new IBM Management Operating System (MOS) for Textile Finishing.

This is a total information system—a step-by-step guide to better management control.

Textile managers can use it to correlate data associated with any or all of the six basic control areas of textile finishing: expanding the finishing order, applying greige inventory, plant loading, raw-materials requirements planning, scheduling, and operations evaluation.

MOS for Textile Finishing can be used with any IBM computer system that has random-access capacity. It can be used by companies involved in the finishing of any cloth—cotton, wool, or synthetic.

IBM® DATA PROCESSING

CIRCLE 9 ON READER CARD

DATAMATION calendar

● The Fourth International Analogue Computation Meetings will be held September 14-18, 1964, at the Technical College, Brighton, England. Sponsor is the British Computer Society, under the sponsorship of the International Association for Analogue Computation.

● The Northwest Computing Assn. will hold its seventh annual conference, Sept. 17-18. Co-sponsored by the Univ. of Washington Computing Center, the meeting will be held at the Univ. of Washington, Seattle.

● Tutorial introduction to electronic information displays and a course in the use of computer languages and design techniques in programming of business application will be conducted Sept. 28-30 at the American Univ., Wash., D.C.

● The American Documentation Institute will hold its annual meeting Oct. 5-8 in Philadelphia, Pa. The theme is "Parameters of Information Science."

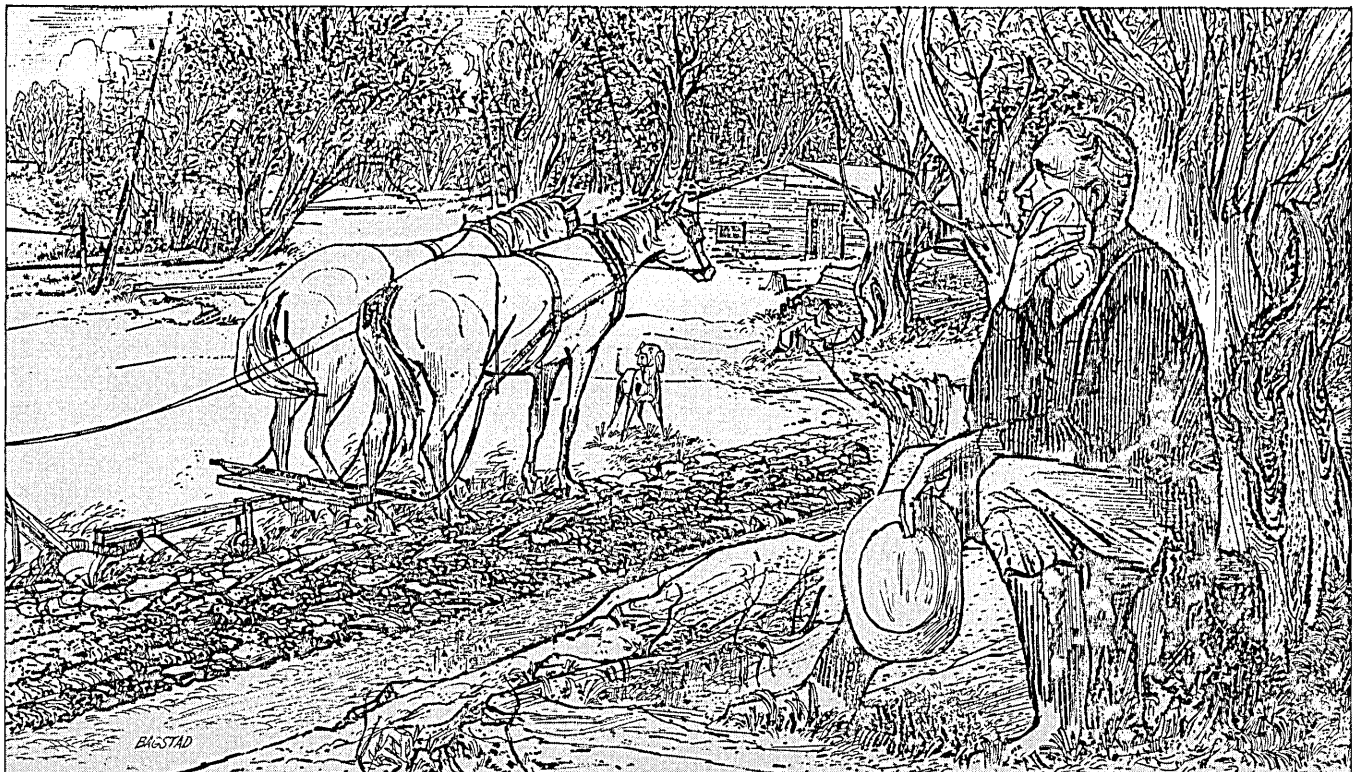
● The California Analysis Center will hold an intensive course on simulation with SIMSCRIPT Oct 5-9 at the Santa Ynez Inn, Pacific Palisades, Calif. Enrollment fee which covers text and materials is $250.

● A national conference on Electronic Information Handling will be held in Pittsburgh, Oct. 7-9. Site of the conference, co-sponsored by the U. of Pittsburgh, Goodyear Aerospace Corp., and Western Michigan Univ., is the Webster Hall Hotel.

● The 17th annual International Systems meeting of the Systems and Procedures Assn. will be held at Philadelphia's Sheraton Hotel, Oct. 12-14.

● The Instrument Society of America will hold its 19th annual Instrumenta-Automation Conference & Exhibit at the New York Coliseum, October 12-15.

● The 6th annual Business Equipment Exposition will be held at the Los Angeles Memorial Sports Arena, Los Angeles, October 19-23.



"...A silent man with patience and industry."

This described William Amery, typical Wisconsin pioneer who settled land near the St. Croix Falls in 1853.

Transportation then was by walking or by horse where the forest paths had been cleared of the virgin pine. Unpainted board shanties afforded shelter as the immigrant families patiently carved a life in the wilderness.

Today, a hundred years later, the wilderness has been transformed into fertile farmland and beautiful homes. Only the "patience and industry" remains as a reminder of the beginning struggle.

Fabri-Tek, at Amery, Wisconsin is pioneering in a different way. In six industrious years Fabri-Tek has become a major supplier of magnetic memories and memory systems.

More than two million cores are being strung on memory planes weekly at the new, modern plant at Amery. Manufacturing capability is expanding at 100 percent a year.

Have you a requirement for magnetic memory planes, stacks, or systems? These people did and came to Fabri-Tek:

Westinghouse
Radiation Inc.
Consolidated Systems Corp.
Telemetrics Inc.
Beckman Instruments
NASA

We have some very interesting material about magnetic memories and memory systems we'd like to send you. Please ask! Write FABRI-TEK Incorporated, Amery, Wisconsin.

FABRI-TEK

EXECUTIVE OFFICES: FOSHAY TOWER, MINNEAPOLIS 2, MINNESOTA

CIRCLE 11 ON READER CARD

letters

ifip, ecma & standards

Sir:

In your June "Business & Science" (p. 17), you make an allusion to ECMA's (European Computer Manufacturers Assn.) activities on programming languages. I am happy to inform your readers that, contrarily (sic) to IFIP policy, ECMA not only did not ignore the U.S. suggestion that ALGOL, FORTRAN, and COBOL be given equal-favour development status, but in factt is already implementing this suggestion.

Our Technical Committees TC 5 (ALGOL), TC 6 (COBOL), and TC 8 (FORTRAN) work on absolutely equal basis. TC 6 and TC 8 keep close contact with the COBOL Committee and X3.4.3., respectively. Numerous papers have been exchanged, comments, questions and suggestions for clarification have been submitted. It is true that we intend to recommend to use the original version as source language; however, you might be interested to hear that at least one company has developed a compiler which accepts as source language English, as well as the translated version (not [yet] into Esperanto!).

TC 5 work on their own. They have produced an ALGOL Subset: ECMALGOL, which has been accepted as an ECMA Standard. This subset is a true subset of ALGOL 60, and it fully contains the alleged IFIP subset. It is therefore expected that those manufacturers interested in providing a compiler for programs written in ALGOL are likely to implement ECMALGOL.

D. Hekimi
Secretary General
European Computer Manufacturers Assn.
Geneva, Switzerland

language comparisons

Sir:

I hope Mr. Shaw's article ("More Instructions . . . Less Work" June, p. 34) is the last machine language vs. problem-oriented language comparison. In the first place, the relevant measures for the cost of a program are development cost and the rapidity with which the resulting program accomplishes its task. Core residence becomes a factor only when the resulting program is slowed by the necessity of accessing additional bulk storage.

As Mr. Shaw demonstrates by example, development costs can be reduced through the use of a sophisticated problem-oriented language. As for execution speed, "almost all" programs have an "inner loop," a small proportion of the code that carries most of the execution. When the "inner loop" is optimized (say, by placing it in a machine-language coded subroutine), no amount of cleverness or effort spent on the remainder of the program will make an appreciable difference in the speed of the resulting program.

In short, one does not compare a MOL and POL; he uses each where appropriate.

W. M. McKeeman
Computer Science Division
Stanford University
Stanford, California

our goof

Sir:

The June issue (p. 76) states that Dr. William Kehl is the director of the Computing Center at this university. The article we submitted stated that Dr. Kehl, director of the Computing Center at the Univ. of Pittsburgh, was the guest speaker at the dedication ceremonies of the center. Please publish a correction.

Rudolf Meyer
Manager, Computing Center
State Univ. of New York at Buffalo
Buffalo, New York

never-ending k's

Sir:

A computer memory of 65,536 words is abbreviated both as a 64K memory and as a 65K memory. The multiplier K represents 1024 in the first case and 1000 in the rounded off second case. An extension of this inconsistency to much larger numbers will result in 16,777,216 being written as 16,384K or 16,777K and possibly as $16K^2$ or $17K^2$. (If $K^2 = M$ then 16M and 17M may be used). The possibilities for confusion are nearly endless. Clearly a standard expression is needed immediately.

Obviously the problem is to find a term that expresses numbers as powers of two since nearly all memories, and many other computer quantities, are even powers of two. I propose the term Binary Orders of Magnitude to be abbreviated as BOOM, boom, B or b. Thus, a 32,768 word memory is called a "15 boom memory" and written "15b." (Binary Magnitudes could be used also but BOOM has a better sound than BM). Two decimal digits can describe all magnitudes likely to be encountered for some time; e.g., $99b=2^{99}>6.3\times10^{29}$. Numbers that cannot be expressed as integral powers of two are easily written in decimal form with the aid of a log log slide rule. The most common such quantities will be 1.25, 1.50 and 1.75 times an even power of two, and the following table is learned quickly:

$1.25=2^{0.322}=.3b$
$1.50=2^{0.585}=.6b$
$1.75=2^{0.807}=.8b$

For Example: $12288=1.5\times8192=1.5\times13b=13.6b$

This notation is simple, logical, unambiguous and easily understood by computer people. Let's stamp out the decimal-derived K from computer literature and reserve it for discussions of professional salaries.

C. W. Glewwe
Univac, Div. of Sperry Rand Corp.
St. Paul, Minnesota

compilation costs

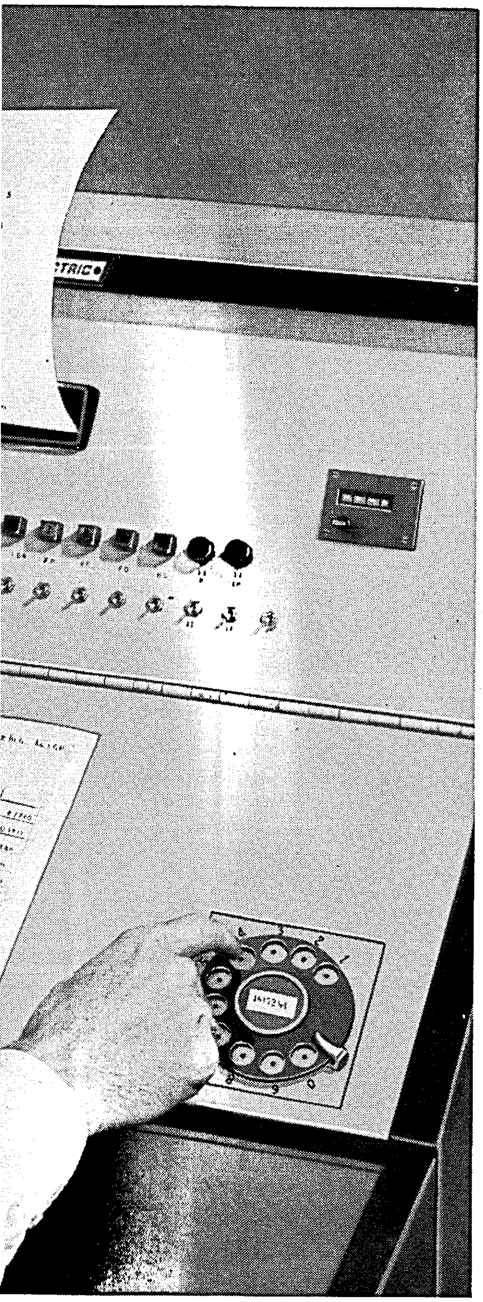
Sir:

Mr. Cowan's article on COBOL ("Is COBOL Getting Cheaper?" June, p. 46) was excellent. I would appreciate a similar article on ALGOL and FORTRAN.

E. Robert Ashworth
Manager, Computing Division
Southern Illinois University
Carbondale, Illinois

So would we. Any takers?

Who in the world would guarantee the manufacture of forms that won't make your high-speed printer stick, stall or stop?



DATAMATION

BUSINESS & SCIENCE

L.A. SOFTWARE FIRM OFFERS IBM FORTRAN COMPILER

Digitek, itsy-bitsy L.A. software firm, tackles gigantic IBM this month with the announcement of a Fortran IV compiler for the 7040 (44) and 7090 (94). To be leased directly to the user, the four-pass compiler is maybe 1/5-1/6 the size of IBM's, and reportedly compiles, on the average, four times as fast as the 94's IBJOB Fortran Version 9. The compiler was designed to really hum on small programs (approx. 200 statements), which they feel constitute the bulk of Fortran jobs.

Formed in May, '61 by three former Hughes Aircraft teammates (Jim Dunlap, Don Ryan, Don Peckham), the privately financed firm began specializing in compilers in early '63, has turned out eight of them since for various manufacturers, ... now has a staff of eight professionals.

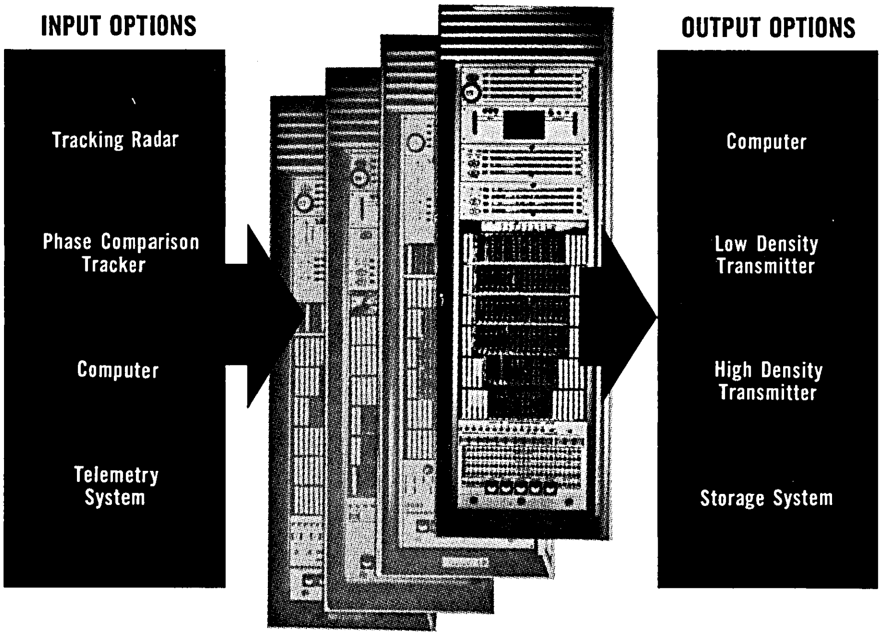
Partially inspired and encouraged by the recent program copyright decision the new compiler -- if successful -- could have some profound implications. Among them: separate pricing of hardware and software by manufacturers. It's analagous to the situation of a few years ago when one firm offered color film and processing in a single package: there was no way to get one without the other. The federal gov't stopped that. Now users may ask why they have to pay for a software system they may not want to use. As the biggest user, Uncle Sam could ask that question with authority.

THERE'S NO BUSINESS LIKE NEW BUSINESS

They're whistling ''Who's Afraid of the Big Bad Wolf'' at Cherry Hill, where RCA/EDP says new biz bookings for the first six months of this year are double those of last year's first half. The company says it has a 9-month backlog, expects to take in $100-million this year ... and go into the black ahead of schedule.

The 3301, thanks to new peripherals and some healthy price cuts (almost 50% on the main frame, 25% average for systems), logged 25 orders, half way to its original marketing target. Even the 301, allegedly over the hill at the age of four, is racking up its second best sales year. Recent 3301 orders include seven (with 14 RACE's) for the Naval Air Stations, and one going into Standard Oil, New Jersey, where it will replace a 7080 and knock out a 360 order. So far, says big, energetic vp Ed McCollister, RCA has lost only one order to the 360 (but won the $ exchange).

Mac is coy about RCA's new product plans, but says the line will embrace the 8-level ASCII code (unlike Honeywell and GE) and nine-level mag tape. Compati-



Typical applications served by Cubic's standard DH-1000 series Digital Distribution Units with "off-the-shelf" input-output options

Standard Cubic input-output options available "off-the-shelf" for your system

CUBIC EXPERIENCE in the design and manufacture of peripheral equipment for a wide variety of data handling systems has resulted in the standardization of input-output options. A wide variety of designs are available "off-the-shelf" to tailor Cubic's basic equipment to the requirements of your specific data-processing system.

THE DH-1000 series DDU (Digital Distribution Unit) shown above is a typical case in point. Any of the input-output combinations shown can be provided by the basic DH-1000. For example, a DDU can be furnished with tracking radar input and any of these four output options: data processor (DH-1011), low-density transmitter (DH-1012), high-density transmitter (DH-1013), or storage system (DH-1014).

EACH OF the other three inputs—phase-comparison tracker, data processor or telemetry receiver—can also be provided with any of the four output options for a total of sixteen possible combinations. Most of these combinations have already been designed and built by Cubic. Complete specifications are available.

OFF-THE-SHELF availability of Cubic peripheral equipment is by no means limited to Digital Distribution Units. Typical of other sub-systems available with "off-the-shelf" input-output options are: equipment for computer interface (DH-5600 series), digital data transmission (DH-6900 series), sensor acquisition (DH-6600 series) and display conversion (DH-7300 series). For more information on how Cubic peripheral equipment can help solve your data handling problems, write to Cubic, Dept. C-180, San Diego, Calif. 92123.

CUBIC CORPORATION
SAN DIEGO, CALIFORNIA 92123
INDUSTRIAL DIV.

CIRCLE 16 ON READER CARD

New Computation Cen
Real-Time Simula
Ground-Floor Opportunity
Long-Range Fut

And the moon, t

The moon? The Apollo Mission S
program—exciting, stimulating, cha
You'll participate in it at Link's new-
nation's newest—computation cente
also contribute to other sophisticat
putation programs centered to vang
entific projects. You'll be a staff me
the company that is the acknowledge
in simulation. The first all-digital s
bears the Link name. So do the si
for the new Boeing 727 and the A
C-135B. Link's Mark I digital comput
basic flight simulation computer in
Mission simulators. Simulation prog
essential to all advanced scientific
Apollo is a program you can plan
around. A future at Link is just that—

SENIOR AND JUNIOR PROGRAMM
write real-time scientific programs f
simulation. Assignments involve app
chanics, space propulsion, celestial
ics and navigation, and life support
BS in mathematics plus 1-5 years of s
programming experience required.

PROGRAMMING PROJECT LEADE
provide engineer/mathematical anal
tems coordination and leadership
systems engineering and the co
programming activity. BS in engine
mathematics and 5 years of comp
ented experience required.
Submit résumé in strict confid
William A. Schell.

LINK GROUP
GENERAL PRECISION INC
PLEASANTVILLE, NEW YORK

An equal opportunity employ

CIRCLE 77 ON READER CARD

bility with the old line will not be a major design criterion, sezzee, but will be solved. Anticipating even better times ahead, RCA will add 50-100 new edp field marketeers (half sales, half technical support).

HEAD FOR THE COUCHES ... THE COMPUTERS IS COMING

At MIT, Professor Weizenbaum, looking for a good example of his programming system's ability to manipulate textual and symbolic information, devised a program in which the computer takes on the role of a sympathetic listener to the woes of the man at the console. Called Programmed Therapy, Help and other things, the routine prints out what looks like a parody of non-directive therapy. Some sample exchanges:

Patient: I am very depressed.
Doctor: How long have you been very depressed?
Patient: Ever since my dad starting drinking.
Doctor: Really,your father started drinking?

And later...

Patient: I don't yell at you or anyone for that matter.
Doctor: Do you want to yell at me?
Patient: I can't get angry with anyone.
Doctor: Have you tried?
Patient: Only secretly in my dreams.
Doctor: Really, in your dreams?

The computer intersperses its paraphrases with comments like, ''Aha!,'' ''Hmmm,'' and ''I understand.'' And it will sometimes circle back to a topic once raised but since dropped. One psychiatrist described the machine's behavior as quite similar to that of a first-year resident in psychiatry. It's all very funny...(isn't it?) except some psychology people take it quite seriously.

OFF-SITE COMPUTER DEPTS. WINNING FRIENDS IN DALLAS

A fixed-price, packaged approach to edp is making sense to a number of Dallas firms ... and giving IBM there a splitting headache. Big D's Electronic Data Systems offers to act as a company's computer dept. After a preliminary analysis, a bid is submitted. If accepted, EDS takes on the whole schmear -- analysis, design, programming ... then runs the programs on computers (7070, 7074, two 1410's and three 1401's) on which they've purchased blocks of time. The contracts are long term -- from five to seven years. Any costs above the quoted figure come out of EDS pockets.

This may represent an industry first: a user who knows in advance what he's going to get and how much it will cost him. So far, EDS has seven package deals in operation; one of them had a computer, two had machines on order. Formed two years ago by Ross Perot (who has pulled together a cadre of other ex-IBMers), the company now has 30 professionals. Perot says he's been trying to overstaff -- one man now spends full time recruiting -- in hopes of expanding his company. He's got his eye on five other cities in which to try the packaged approach which, he says, has made EDS Dallas' second largest edp income-producing outfit.

OLD COMPUTERS NEVER DIE ...

Tired old computers are still moving in the computer marketplace. Dow Chemical has purchased a used G-15 from ITT Federal Labs, with the help of the Lamellar

Continued on page 79

How do you speed up program preparation?

Symbolic program debugging and editing on line with Programmed Data Processor-5.

On line debugging with **DDT-5** gives the user dynamic printed program status information. It gives him close control over program execution, preventing bugs from destroying other portions of his program. He can monitor the execution of single instructions or subsections, change instructions or data in any format, and output a corrected program at the end of the debugging session.

The symbolic editor, **SED,** lets him add, delete, change, or insert single lines or sections of his symbolic program, thereby relieving him of the seemingly endless routine of reproducing, listing, and desk-checking his program. The output of the editor is a complete program ready for assembly or compilation.

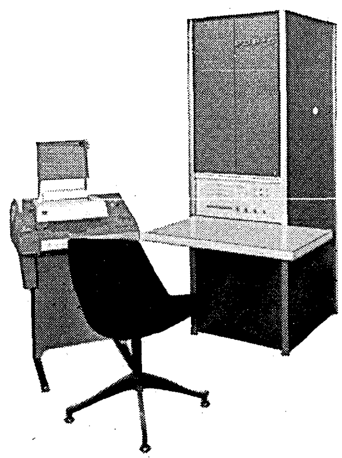
FORTRAN gives the programmer a diagnostic printout during compilation and lets him add, delete, or change statements before continuing to compile. As a result the programmer is not forced to recompile because of the appearance of a minor clerical error in a FORTRAN statement. He can correct a mistake immediately at the console teleprinter and continue compiling.

Other elements in the PDP-5 programming system are a macro assembler, floating-point software, elementary function subroutines in single- and double-precision arithmetic and utility, diagnostic, and maintenance routines.

The general purpose PDP-5 is being used for digital and hybrid computation, nuclear physics experiments, oceanographic research, control of power and steel production systems, reactor monitoring, and flight simulation. It has a six microsecond core memory of from 1024 to 32,768 twelve bit words. It performs 55,555 additions per second, transfers up to 166,000 words per second, and controls 192 I/O devices.

A complete PDP-5 system can be yours for the cost of 50 hours of operating time on large computers. It can be running in your plant producing answers on the day it is delivered.

Delivery time? Two weeks.



digital

EQUIPMENT CORPORATION

MAYNARD, MASSACHUSETTS

Los Angeles • Palo Alto • Pittsburgh Washington, D. C. • Parsippany, N. J. Chicago • Ann Arbor, Mich. • Carleton Place, Ontario • Munich, Germany Reading, England • Sydney, Australia

CIRCLE 17 ON READER CARD

WASHINGTON REPORT

CLEWLOW COMMITTEE REPORT BOUNCES

The first draft of the report being prepared by the Budget Bureau's Clewlow Committee on Government computer procurement and management practises has been returned for further work by the advisory committee. Before giving its o.k., the advisory committee, headed by ex-Georgia Congressman Robert Ramspeck, reportedly wants more information pertaining to two crucial areas; the relative merits of buying or leasing computers, and the advantages and disadvantages of a central agency controlling government computer operations. Originally scheduled for delivery to the President by June 30, it's now unlikely that the report will be ready to go before September. The delay just about kills any chance that the Senate will act favorably in this session on the Brooks Bill, H.R. 5171, which would create a central computer procurement control for Federal agencies.

MAC NOT ONE OF BARRY'S BOYS

Senator Goldwater has sworn to do mighty things at the Department of Defense, should he be elected President, with priority being given to giving Secretary of Defense McNamara the heave-ho. Pentagon observers consider it unlikely, however, that he would attempt to undo most of McNamara's basic management reforms in cost effectiveness and force and budget analysis which are based on computers. Pro- or anti-McNamara, most top military leaders now regard these techniques as a considerable improvement over what went on before. A considerable clutch among the brass, who, though they might admire McNamara's handiwork, would not shed any tears over his departure, feeling strongly as they do about what they consider to be usurped prerogatives. As for the Senator, who is himself a reserve major general, he's known to feel strongly about such matters as the manned bomber and the virtues of untrammeled automation, which may or may not bode well for computer folk.

* * *

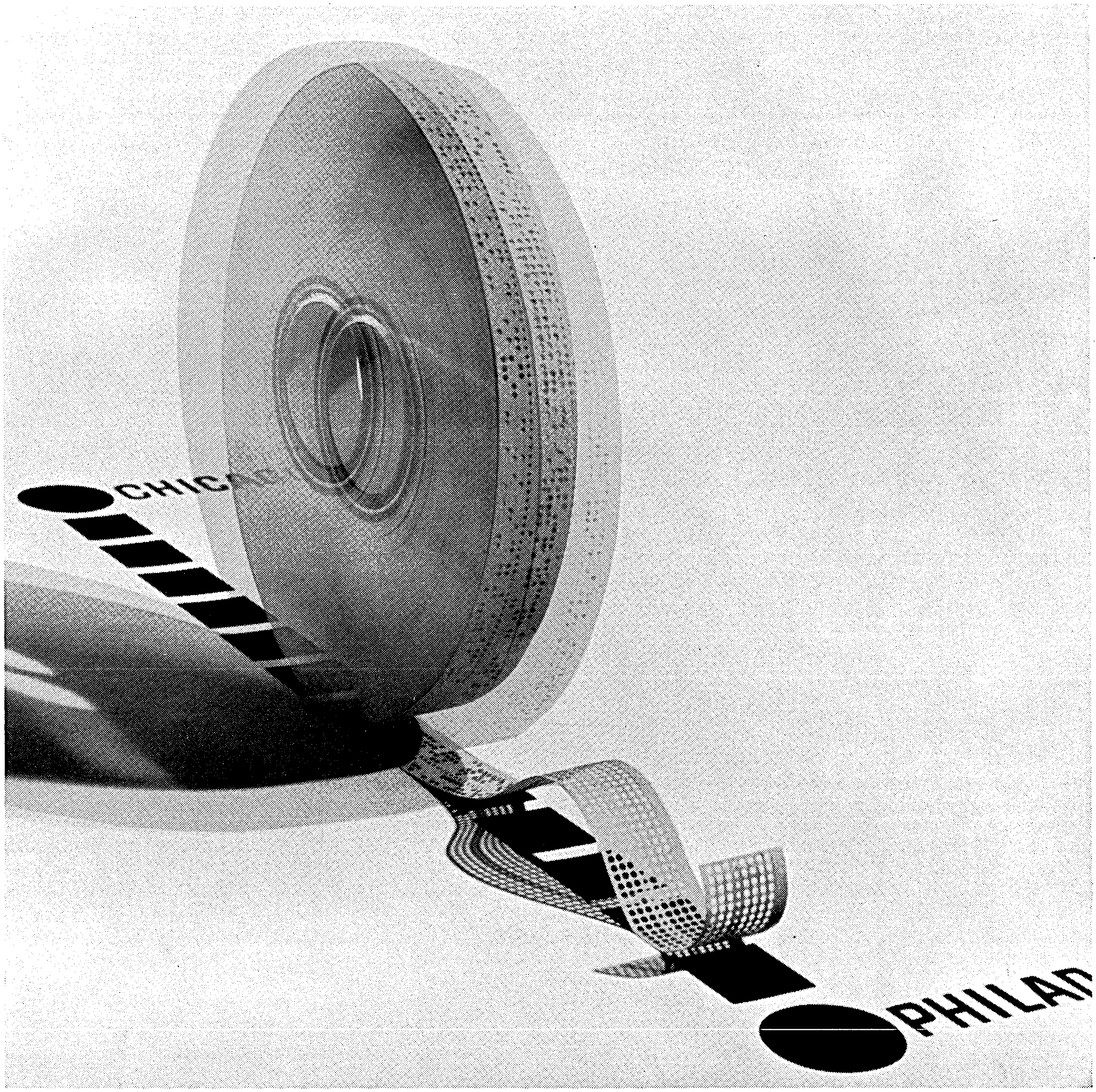
Refuted canard of the month: Senator Goldwater did not plan his election campaign with the aid of an abacus.

IBM 360 BLOODED WITH FAA SELECTION

The first major order by a Government agency for Series 360 computers was placed by the Federal Aviation Agency for prototype Air Traffic Control systems to be installed at its experimental facility in Atlantic City, N. J., and at the ATC center in Jacksonville, Fla. The two 360 configurations, to cost approximately $3 million apiece, are slated for installation some time in 1965.

Selection of the 360's by FAA was a mild surprise; guesses were that quicker availability and proven-in-use records would have favored equipment put forward by other bidders. All told, FAA let contracts totaling $10.4 million for the elaborate ATC systems, for which Burroughs and Raytheon are also providing equipment. The systems will provide scope watchers with much digitized data about the blips (planes) they watch on

Continued on page 91



IN JUST THREE MINUTES THIS PAYROLL CAN TRAVEL FROM PHILADELPHIA TO CHICAGO. FARE: $1.54

This is what DATA-PHONE service has done for business communications.

It gives computers an electronic "voice" that travels via regular telephone lines, at regular telephone rates.

At speeds up to 2500 words per minute, DATA-PHONE service can send anything that can be punched on cards or tape—orders, technical data, waybills, inventories.

Like to know more? Just ask your Bell Telephone Business Office to have our Communications Consultant contact you.

BELL TELEPHONE SYSTEM
Serving you

CIRCLE 18 ON READER CARD

EDITOR'S READOUT

THROUGH GLASSES DARKLY

As your local, friendly salesman can tell you, any one of his prospects or customers can generally be neatly divided into two classes. One could be called Mr. Can Sign—top management—the other Mr. Technical Management.

They are divided along rather classical lines, and each views his own world through a pair of glasses designed to intensify his own set of interests . . . and to screen out other, conflicting viewpoints.

Mr. Can Sign's shades are shaped like dollar signs, while Mr. Technical Management looks through cores strung on a mad spider's web of FORTRAN and COBOL symbols. He lives in a tangled verbal jungle of technical terms, and worries about getting the most out of the mangy kludge which management shoved down his throat.

In short, top management looks at the monthly rental bill; dp management worries about things like throughput and turnaround time. It's an old story, and not a particularly sad one, except that recent events make it appear that the dichotomy of viewpoints between these equally enlightened groups will become more severe.

We're talking about the continuingly accelerating increase in the ratio between machine costs and productivity, sometimes called "bang for the buck."

Certainly in that never-never land of advertising and sales promotion prose, manufacturers are basing their pitches on this basic economic fact of life—cutting dp operating costs through faster, better balanced machines with lower rental figures.

Although he views these claims with somewhat less enthusiastic gullibility than he did a few years ago, top management can't help but be interested in such talk. He sees a throughput/dollar ratio rise from 1:1 to 1.3:1, and he comes to an eminently logical conclusion: we can do the same amount of work, and cut our monthly bill.

The dp manager examines the same set of ratios and comes up with quite a different sort of evalution. He sees all sorts of new things he can do which were too expensive before. He says, give me the dollars. I'll get more work done, find new applications, work toward an integrated system, go for that blue sky that's always just out of reach. I'm the one with the technical know-how, the one who knows how to allocate the dough in the best way.

Top management isn't having any. He's paid a steep learning fee, and he wants some return. Computers are supposed to save money, so here's their chance. Now.

We're not trying to say there's a wrong and a right in this perpetual power struggle. Each company has to figure out for itself what its dp goals are . . . now and for the long range. But maybe the dp manager can come out of his technical jargon jungle to try to learn about his company and its aims. And maybe top management can come out from behind his blinders to learn enough about computers to see how they might help his firm do more than mark time and cut costs.

FORTRAN: COMPATIBILITY & STANDARDIZATION

by W. P. HEISING

Prepared by the chairman of the American Standards Assn. committee charged with the responsibility of developing FORTRAN standards proposals, the following article is intended not as a progress report on the work of that committee, but rather as some conclusions—based on two years' work involving most of the principal American computer manufacturers and user groups—concerning FORTRAN compatibility . . . and what standards can and cannot be accomplished in the FORTRAN area.

Several aspects of FORTRAN history, or programming language standardization and of standardization generally must be kept in mind.

1. FORTRAN, although initially used on the IBM 704, gradually became available for many machines produced by many manufacturers. Two factors are probably primarily responsible for the fact that FORTRAN "just growed." Firstly, FORTRAN was inherently well suited to making programming easier for the non-professional open shop user, and secondly FORTRAN could enormously reduce the reprogramming difficulties in converting from one machine to another. In the early stages of FORTRAN history, compatibility was definitely secondary, so that additions or modifications to the language were made rather light heartedly by new compiler writers on the basis of personal preference. Later, the importance of compatibility considerations was more clearly recognized as a result of the growing investment in existing FORTRAN programs by users, and a definite stabilization of the language set in, although the language continued to grow.

2. Until 1962, when the X3.4.3 FORTRAN standards Committee was formed, there existed no organized channel of information interchange. This is actually quite astonishing considering the very widespread use of FORTRAN. The situation was unlike that of any other widely used common programming language. In order to define FORTRAN standard(s), it was necessary to compare the various existing "FORTRAN" practices. After interchange of FORTRAN programming manuals, it soon became evident that they were all incomplete. Questions could be asked that could not be answered from studying the manuals of individual FORTRAN systems. Fortunately, most of these questions could be answered by someone already familiar with a particular FORTRAN —although some questions arose that required either user tests or compiler authors to go into their own code to determine the answer.

3. The June issue of *Datamation* contained an unattributed statement estimating that the investment in FORTRAN currently is 2½ to $5 billion. That figure seems to me to be much too high. I have not been able to find any survey information containing the type of data on which to base a careful estimate, but the value of FORTRAN programs in use is undoubtedly very substantial. Accordingly, in FORTRAN standardization, the approach called for is not to go off and design a new improved language, but rather to stick to existing practice, although where current usage is conflicting, a definite choice must be made.

4. No American Standard programming language exists; there are no precedents to serve as a guide in this particular area. A crucial question that arose early was: "Can a practical standard be developed that can *guarantee* interchangeability of FORTRAN programs across machines?" The judgment was unanimous—No! The difficulties are twofold: (a) compatibility using a variety of hardware, and (b) improperly written FORTRAN programs.

Compatibility: The expression, "machine independent," language has come into vogue in recent years, and it has seemed to me that it is often used carelessly—as if to imply some fundamental or theoretical distinction between "machine" and "machine independent" programming languages. All so-called "machine independent" programming languages that are really complete are, in fact, the machine language of an actual computer that hasn't yet been built, but such languages could be directly and completely realized in hardware. The fact that such machines haven't been built relates to economic feasibility and performance—thus the distinction between "machine independent" and "machine dependent" languages is practical, not theoretical. In fact, a little thought shows that a compiler *establishes the interdependence* between a "machine independent" language and a machine language. A language that is truly "independent" of a machine is also a "machine irrelevant" language. The real compatibility difficulties arise from the fact that FORTRAN, like other so-called "machine independent" languages, is a deliberately incomplete machine language—in this case, of the "FORTRAN machine." In such areas as precision, range, storage, number and nature of I/O devices, it seems unwise to completely specify the FORTRAN language in order to permit practical, efficient, programmed "realization" of the "FORTRAN machine" by a variety of actual computers. Thus, a FORTRAN program can be dependent for its intended operation on such factors

Mr. Heising is manager of Programming Technology in IBM's Data Systems Div. Joining the firm in 1950, he has been engaged in such areas as automatic coding systems, computer design and use, as well as advanced programming systems planning. He was formerly a radio engineer with RCA Labs, and an instructor in physics. He holds a degree in chemical engineering from Cornell and an MA in physics from Columbia.

which are deliberately left ambiguous in the FORTRAN language, and thus program transferability cannot be guaranteed.

Improper FORTRAN Programs: Strictly speaking, one might take the point of view that the action of a computer on improper FORTRAN programs may be simply left unspecified—as being irrelevant and rightfully outside the scope of any standard. It has been necessary to take this point of view, but only because it appears to be beyond the current state of the programming art to specify "complete" diagnostics and know that goal to be practically realizable. "Improper" FORTRAN programs in specific implementations often do useful things —sometimes unintended by the compiler builder—and deliberate or inadvertent use may be made of such "features," this limiting FORTRAN compatibility. Many of the idiosyncrasies of FORTRAN have arisen directly from such historical accidents. The only bright spot in leaving "improper" FORTRAN programs undefined is that the way is left open for useful additions to be made deliberately to the language by implementors without necessarily being thereby "non-standard." Thus the growth of FORTRAN is allowed for, but on the other hand, a potential source of new incompatibilities remains.

In conclusion, a FORTRAN standard will not bring the millenium; it will *facilitate* but not *guarantee* interchangeability of programs between machines. Nevertheless, it is believed that a useful purpose will be served by identifying and describing much FORTRAN folklore, and thereby reducing the incompatibilities arising inadvertently through misunderstanding. However, many conversion problems will still remain, and cries of dismay over "inadequate FORTRAN diagnostics" will be heard through the land for some time to come. ■

THE VARIOUS FORTRANS

by HENRY OSWALD

FORTRAN, the "universal" programming language, is indeed universal, or at least, nearly so. Of 43 FORTRAN compilers, 16 are listed in the accompanying comparison matrix. There is no doubt that more programmers can write and more machines can compile it than any other programming language. Moreover, on the surface, it appears that all these are the same—i.e., that each will accept a FORTRAN program written for any other. This is not true. We will demonstrate, by means of the FORTRAN Language Characteristic/Compiler Comparison Matrix, some of the variability in the language and some consequent incompatibilities.

It is widely held among programmers and other computer knowledgeable people that FORTRAN is a means to compatibility among different machines. It is generally accepted and often stated that *any* FORTRAN program written for any particular machine can, with only "minor" changes (usually in I/O statements), be compiled on some other machine and produce running code. The statement is true, but only for *some* programs and over a limited set of compilers. If several FORTRAN compilers admit differences in source language, then only programs that employ those language features that represent the intersection of the languages can be compatible. Even if the intersection of language features is used, compatibility is not assured, for built-in features, e.g., maximum number range, over which the user has no control, can destroy compatibility.

Basically all FORTRAN look alike. All have arithmetic replacement statements, the DO, IF, subscripts. Real and Integer constants and variables, GO TO, FORMAT, I/O statements, etc. The form of each statement is, for the most part, universal. The differences lie behind this facade of uniformity. The real differences are subtle (often truly minor) but there are many such differences, and their total impact tells the story of the universality or compatibility of FORTRAN.

About a year ago, the problems of programmers in the use of manuals, particularly the extraction of information from manuals, led the author to consider the creation of Infograph R (Fig. 1). Infograph makes use of the quick-find features of the mechanical list finder to eliminate the necessity of thumbing through a conventional manual. Since the amount of space available in an Infograph is limited, whatever information that is to be displayed must be concise, brief and readily assimilable. It was decided to develop a FORTRAN Infograph since such a device might appeal to a large

(Text continues on page 28)

Mr. Oswald is president of Celestron Assoc. Inc., a software firm in Yonkers, N.Y. He is both a physicist and system analyst with extensive programming experience in both applications and machines. He is also a consultant in the development of software for a mass memory-centered processor to be used in non-numeric language processing. He holds a BS and PhD in physics from New York Univ.

	IBM 7090-94	UNIVAC 1107	UNIVAC III	CDC 3600				
Total Number of Statements Possible	46/35	46/35	41/30	61/44	38/25	32/19	31/17	25/14
Number of I/O Statements	11	11	11	17	13	13	14	11
Character Set[1]	Standard Compiler	Standard plus $'	Standard plus $'	Standard plus $	Standard	Standard	Standard	Standard
Integer Const/Variable								
# Digits	10	10	6	14	5	5	10	1 to k k=4 to 10
Range	$0—2^{35}$	$0—2^{35}$	$0—10^{6}$	$0—2^{47}$	$0—2^{17}$	$0—2^{17}$	$0—10^{10}$	$0—10^{K}$
Range As Subscript Or Index	Mod 2^{15}	Mod 2^{16}	—	—	Mod 2^{17}	Mod 2^{15}	—	—
Real Const/Variable								
# Digits	8	8	10	10	8	8	8	1 to f f=2 to 28
Range	$10^{-38}—10^{38}$	$10^{-38}—10^{38}$	$10^{-51}—10^{49}$	$10^{-308}—10^{308}$	$10^{-38}—10^{38}$	$10^{-38}—10^{38}$	$10^{-99}—10^{99}$	$10^{-100}—10^{99}—10^{99-f}$
Number of Characters in a Name	6	6	6	8	6	6	6	6
Subscripts—maximum number	3	7	3	3	3	3	3	2
Operations (in order of execution)								
Function Reference	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
		unary minus (—)						
*and /	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
+ and —	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
.LT., .LE., .EQ., .NE., .GT., .GE.	Yes	Yes	Yes	Yes	No	No	No	No
Logical NOT	Yes	Yes	Yes	Yes	No	No	No	No
Logical AND	Yes	Yes	Yes	Yes	No	No	No	No
Logical OR	Yes	Yes	Yes	Yes	No	No	No	No
Boolean Masking NOT	No	No	No	Yes	Yes (—) modal	Yes (—) modal	No	No
Boolean Masking AND	No	No	No	Yes	Yes (*) punch	Yes (*) punch	No	No
Boolean Masking OR	No	No	No	Yes	Yes (+) B	Yes (+) B	No	No
Expressions								
Arithmetic	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Logical	Yes	Yes	Yes	Yes	No	No	No	No
Mixed[2]	D&R C&R	C, D, R&I	No	all types Yes, mixed any way	No	No	No	No
Boolean (Masking)	No	No	No	Yes	Yes	Yes	No	No
Assigned GO TO[3]	Yes with list	Yes, list may be omitted	Yes with list	Yes, list may be omitted	Yes with list	Yes with list	Yes with list	No
DO Statement								
Form of Index	IV	IV	IV	IV	IV	IV	IV	IV
Form of Parameters	IV	IV	IV	IV	IV	IV	IV	IV
If initial value > final value	DO executed once	executed in reverse with i decremental	executed once	Not executed	executed once	executed once	executed once	executed once
Pause	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Stop	Exit to monitor	Exit to executive	Exit to Boss III	Yes/exit to COOP	Yes, exit to monitor	Yes	Yes	Yes
End	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Format Arguments								
nP	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Ow	Yes	Yes	Yes	Yes	Yes	Yes	No	No
wH	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Aw	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes
wX	Yes	Yes	Yes	Yes	Yes	No	No	Yes
Lw	No	No	No	Yes TRUE=1 FALSE=0	No	No	No	No
				Special C=complex				
I/O Statements								
Cards	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
On-line Printer	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Mag Tape	No	No	No	Yes	Yes	Yes	Yes	Yes
Typewriter	No	No	Yes	No	No	No	Yes	No
Paper Tape	No	No	No	No	No	No	No	No
Disc	No	No	No	No	No	No	No	No
Drum	No	No	No	No	No	No	No	No
Simplified FORTRAN IV	Yes	Yes	Yes	No plus extras	No	No	Yes	No
Common/Equivalence Interaction — Equivalence order of variables takes precedence over common	No	No	No	No	Yes	Yes	Not in language	Not in language
Function								
Built-in, Library (#)	55	73	35	35	27	26	28	9
Statement	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Subprogram	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Subroutine								
Data Statement/Block Data SR	Yes	Yes	Yes	different form	No	No	No	No
Type Declaration of Variables	Yes	Yes	Yes	Yes	No	No	No	No
Statement ID	Number	Number	Number	Number	Number	Number	Number	Number
number range	0≤n≤32, 767	0≤n≤32, 767	0≤n≤32, 767	0≤n≤99,999	0≤n≤32,767	0≤n≤32,767	0≤n≤99,999	0≤n≤99,999
name?	No	No	No	No	No	No	No	No
FORTRAN Deck Ordering[4]	simple	non-simple	non-simple	extraordinary	simple	simple	simple	non-simple
Special Features: EXIT DUMP PDUMP Subroutines		Normal/Abnormal Functions and Subroutines; Internal Subprogram; Different Form of Subscripts; Special RETURN PARAMETER	NONE	Any number of statements per line separated by $; Multiple Replacement Statement; ENTRY to provide multiple SR entry; I/O IF Statements; BUFFER IN/OUT; ENCODE/DECODE; Special Types; Internal SR	Modal Punch for Complex, Double, Boolean, External; Chaining; Source Language Debugging	Modal Punch; Frequency	Arrays by Row; Fortran IV I/O Statements; No DO Lists in Tape I/O; Mixed Languages	Variable Number Representation; 80 Column, Read Punch; 100/132 Character Print Line; Restriction On Tape Read

FORTRAN Language Characteristic	IBM 1620	CDC 1604-A	CDC 160A	ALTAC III Philco 2000	AUTOMATH H 800 H 1800	SDS 900 (910, 920)	PB 440	RCA 301
Total Number of Statements Possible	23/15	46/30	36/21	39/25	42/21	37/22	38/22	35/20
Number of I/O Statements	8	16	15	14	21	15	16 ?	15
Character Set[1]	Standard	Standard plus $	Standard	Standard plus; $	Standard plus $	Standard	Standard?	Standard plus $
Integer Const/Variable								
# Digits	1 to k k=4 to 10	14	7	4	5	7	7	7
Range	$0-10^K$	$0-2^{47}$	$0-2^{22}$	$0-2^{15}$	$0-2^{15}$	$0-2^{23}$	$0-2^{23}$	$0-10^7$
Range As Subscript	—	—	—	—	—	—	—	—
Or Index	—	—	—	—	—	—	—	—
Real Const/Variable								
# Digits	1 to f f=2 to 28	10	8	?	11	11	?	8
Range	$10^{-100}-10^{99}$—f or	$10^{-308}-10^{308}$	$10^{-33}-10^{31}$ $10^{-38}-10^{38}$	$10^{-600}-10^{600}$	$10^{-77}-10^{76}$	$10^{-77}-10^{77}$	$10^{-38}-10^{38}$	$10^{-100}-10^{100}$
Number of Characters in a Name	6	8	6	7	6	8	80	6
Subscripts—maximum number	3	3	3	4	3	any number	any number	3
Operations (in order of execution)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Function Reference								
**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
* and /	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
+ and —	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
.LT., .LE., .EQ., .NE., .GT., .GE.	No	No	No	Yes different forms	No	No	No	No
Logical NOT	No	No	No	No	No	No	No	No
Logical AND	No	No	No	No	No	No	No	No
Logical OR	No	No	No	No	No	No	No	No
Boolean Masking NOT	No	Yes modal	—(No) modal	No	(—) Yes modal	No	No	No
Boolean Masking AND	No	Yes punch	** (left shift)	No	(*) Yes punch	No	No	No
Boolean Masking OR	No	Yes B	*(AND) punch B + (inclusive OR) / (exclusive OR)	No	(+) Yes B	No	No	No
Expressions								
Arithmetic	Yes	Yes		Yes	Yes	Yes	Yes	Yes
Logical	No	No	Yes	No	No	No	No	No
Mixed[2]	No	No	No	Yes R&I mixed any way	No	Yes R&I mixed any way	Yes R&I mixed any way	No
Boolean (Masking)	No	Yes	Yes	No	Yes	No	No	No
Assigned GO TO[3]	No	Yes with list	Yes list **must** be omitted	Yes list **may** be omitted	Yes with list	Yes with list	Yes list may be omitted	Yes with list
DO Statement								
Form of Index	IV	IV	IV	IV	IV	any variable	IV	IV
Form of Parameters	IV	IV	IV	IV	IV	any expression	any expression	IV
If initial value>final value	executed once	**not** executed	executed once	executed once	executed once	executed once	executed once	executed once
PAUSE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
STOP	Yes	Yes/exit to monitor	Yes	Yes exit to operating system	exit to monitor	Yes, exit to system	Yes	exit to operating system
END	Yes	Yes	Yes, executable as a halt/return	Yes, also COMPLETE	Yes, has restriction	special form		
Format Arguments								
nP	No	Yes	No	Yes	Yes	Yes	Yes	Yes
Ow	No	Yes	Yes	Yes	Yes	No	Yes	No
wH	Yes	Yes	Yes	Yes, extra characters possible	Yes	Yes	?	No
Aw	Yes	Yes	Yes	Yes	Yes	Yes	?	Yes
wX	Yes	Yes	Yes	Yes	Yes, has alternate form	Yes	?	Yes
Lw	No	No	No	No	No	No	Yes	Yes
I/O Statements								
Cards	Yes	Yes	Yes an SR may be specified	Yes	Yes, alternate	Yes	No	No
On-line Printer	No	Yes	Yes	Yes	Yes, alternate	Yes	No	Yes
Mag Tape	No	Yes	Yes in a I/O state-	Yes	Yes	Yes	No	Yes
Typewriter	Yes	No	Yes ment to oper-	No	No	Yes different form	used for I/O	Yes
Paper Tape	Yes	No	Yes ate with any I/O unit	Yes	No		other than PT one	Yes
Disc	No	No	Yes	Yes	No	Yes	Yes	No
Drum	No	No	Yes	No	No	Yes	Yes	No
Simplified FORTRAN IV	No	Yes	No	No	No	No	No	No
Common/Equivalence Interaction — Equivalence order of variables takes precedence over common	Rule to prevent Yes	Yes	Not in language	Yes	Yes	No, may not	No	Yes
Function								
Built-in Library (#)	8	32	9	42	28	?	?	No, **may not**
Statement	Yes	Yes	No	Yes	Yes	Yes	20	19
Subprogram	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Subroutine	Yes	Yes	Yes, compiled with program only	Yes	Yes	Yes	Yes	Yes
Data Statement/Block Data SR Subroutine	No	No	No	No	No	No	Yes	Yes
Data Statement/Block Data SR	No	No	No	No	No	No	No	No
Type Declaration of Variables	No	No	No	No	No	No	No	No
Statement ID	Number	Number	Number	Name or Number	Number	Number	Name or Number	Number
Number Range	$0 \le n \le 99{,}999$	$0 \le n \le 99{,}999$	$0 \le n \le 99{,}999$	$0 \le n \le 99{,}999$	$0 \le n \le 32{,}767$	$0 \le n \le 99{,}999$	?	$0 \le n \le 99{,}999$
name?	No	No	No	Yes, symbolic names not beginning cort	No	No	Yes	No
FORTRAN Deck Ordering[4]	non-simple	extraordinary	non-simple	extraordinary	extraordinary +	non-simple	simple	simple
Special Features EXIT, DUMP, PDUMP Subroutines	Variable Number Representation	I/O IF Statements BUFFER IN/OUT ENCODE/DECODE Internal SRs	Special I/O Subroutines to Communicate with any I/O Unit, Used as Part of A Special I/O Statement.	Statement Names Compound Statements Subscripted Subscripts Any Expression as Subscripts Mixed Type In-Line Assembly Code	Source Program Limitation In-Line Assembly Code Special Control Cards Special I/O Statement	Number and Kind of Subscripts Special Form of DIMENSION Unnumbered Hollerith Free Field Input	80 Character Names Any Expression Subscripts Subscripted Subscripts Statement Names Mixed Type Expression Unambiguous "F" Names Embedded Statement Backward Indexing DO's Dummy Names in	Fortran IV I/O Statements

[1]Standard Character Set: 0-9, A-Z, *, . + — / = ( )
[2]C = Complex; D = Double Precision; R = Real; I = Integer; L = Logical; S = Special
[3]GO TO i(n_1, n_2, ...) the (n_1, n_2, ...) is the List
[4]Simple = one, two or three restrictions; non-simple = specific ordering of statement types required; Extraordinary = ordering and control cards or multi passes required.

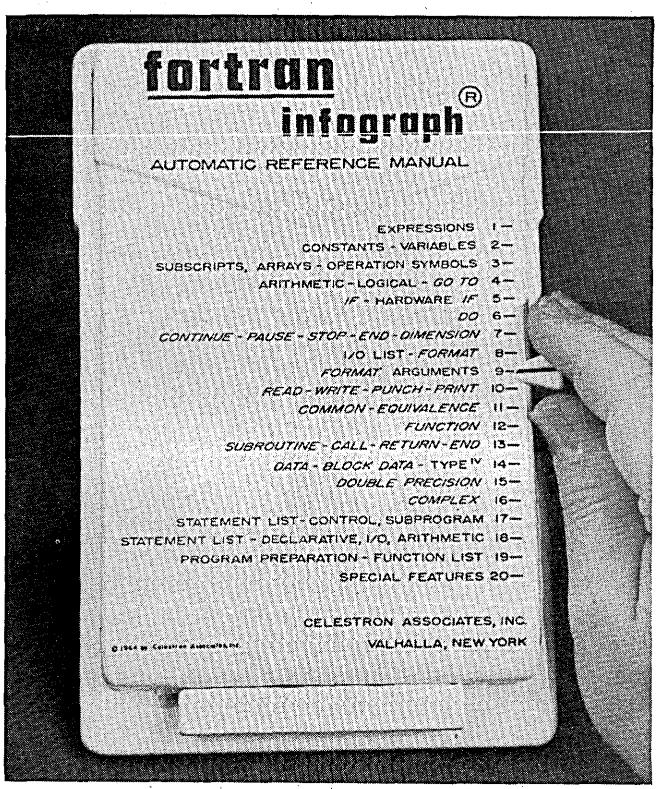
audience. Our first efforts centered around IBM 7090 FORTRAN II. The problem of extracting pertinent data from the manual and restating it in a graphic new format was formidable. It meant abandoning the general FORTRAN manual procedure of explanatory paragraphs followed by examples. Wherever possible we present the general form of a language item, followed by a few sentences or phrases of identification and explanation, followed by a few specific examples. The result is a concise description of the language.

When 7090 FORTRAN II had been successfully rewritten for Infograph, we decided to check several FORTRAN compilers to ascertain whether the Infograph version would be useful to all FORTRAN programmers. It was discovered that the differences were sufficiently important that no single version of the language could possibly be used by all programmers.

Accordingly the project proceded on a much broader basis through a careful analysis of the compilers named in the comparison matrix. Following the analysis, the FORTRAN Infograph was redesigned. Infograph now represents essentially the union of FORTRAN, in that all the basic features are present and provision is made to display special features for particular machines. In addition, a "specializer" is supplied with each Infograph. The specializer permits the user, by making a series of simple modifications keyed to the standard contents, to specialize his Infograph for a particular machine. The FORTRAN Infograph, then, together with the set of specializers, reflects the FORTRAN language now in use.

The FORTRAN Language Characteristic / Compiler Comparison Matrix attempts to illustrate some of the actual differences between the compilers. No attempt was made to include all the variables; in fact, the language characteristics presented were chosen because

It's easy to use the Fortran Infograph. Move the Infograph Subject Selector to the Cover Index line that best describes the Fortran information desired.



they seemed to represent the more important differences without becoming tediously detailed.

Without attempting to discuss each language characteristic in detail, some comment is required to fill in between the lines. For the most part we leave judgment to the reader; we consider that the comparison matrix represents a reportorial effort rather than a critical evaluation.

The following paragraphs each refer to an entry under the language characteristic column of the comparison matrix.

The outstanding compiler under total number of statements is FORTRAN 63, the CDC FORTRAN IV for 1604 and 3600. This particular FORTRAN allows sub and super word variables, I/O buffer definition statements and code conversion statements, among other features. One notion of language content can be had by comparing the number of non I/O statements (separated by a slash from the maximum number). Thus, for example, CDC uses 44 statements on the CDC 1604, while Honeywell uses nine statements on the H290. Both belong to the FORTRAN family.

The character set is remarkably consistent. Variations are either caused by hardware problems or the desire to add a feature (e.g., $).

Representation of numbers is one of the items that displays a great deal of variety. Integers are represented by up to 14 or as few as three decimal digits.These integers can range in value from $0 \leq i < 2^{11}$ up to $0 \leq i < 10^{11}$. Real numbers are represented by as few as two up to as many as 10 decimal digits; the smallest maximum number of digits is six. The range of values for reals varies as follows: $10^{-32} < r < 10^{31}$ on the low end, $10^{-600} < r < 10^{600}$ on the other extreme. From the viewpoint of compatibility such variation can raise havoc with otherwise compatible programs. Consider, for example, the inversion of an ill-formed matrix where small differences between large numbers become significant. A program that runs well on one machine would not run at all when recompiled simply because the number range is exceeded. This sort of non-programmer controlled incompatibility would be missed by "translators" (F-II to F-IV) patterned after LIFT or SIFT.

The PB 440 outclasses all others in the naming department allowing variable names up to 80 characters in length. However, if one wanted to write a program that is compatible across thc board (as far as variable naming goes) one would be limited to four characters.

Double precision and complex arithmetic are very special features usually not available. Note, however, that even where DP is available the number of digits varies.

Subscripts range from a minimum of two to a maximum of any number. In some cases subscripts may be Integer Expressions and in others any Expression. Subscripts may sometimes be subscripted to any level, but usually may not be subscripted at all.

All FORTRAN allow exponentiation, multiplication, division, addition and subtraction. Relational operations may be used to relate arithmetic quantities; however, they are *not* usually available. The logical operators are available only in FORTRAN IV and the Boolean operators are rare. In one case Function reference is *not* available. Again "across the board" compatibility would require limitation to the arithmetic operations (**, *, /, + and −).

The type of expressions possible depends on the operations that are included. Mixed expressions are usually not permitted. The CDC 1604 FORTRAN 63 (F-IV) permits mixed expressions without restriction. The same FORTRAN allows multiple replacement statements (one line with several = operators), each segment of which may be of a different type. Confusion is eliminated by establishing a type hierarchy. Logical expressions deal with single valued variables, i.e., .TRUE., .FALSE. or 1, 0 while Boolean expressions deal with data fields such as, say, a 36-bit machine word.

The ASSIGN and Assigned GO TO often are not available. In a few cases the list of possible successor choices is not necessary.

Logical IFs are found in FORTRAN IV only. There is considerable variation in the availability and interpretation of Hardware IFs. Note that the F-IV subroutines (SSWTCH, etc.) that are to replace Hardware IFs are not uniformly used among F-IV compilers.

The DO statement is identical in many FORTRAN but there are exceptions. The Univac 1107 will count a DO index in reverse if the initial value is greater than the final value. CDC and TRW will skip execution under the same conditions, all the rest will execute the DO loop once, except RPC 4000 which won't allow the condition in the first place. The DO index must be an Integer Variable while the initial, final values and increment may be Integer Variables or constants. The above is true except for SDS 900, RPC 4000 and PB 440. SDS allows the index to be any variable. All three allow the parameters to be any expression.

All versions allow PAUSE. STOP is usually an exit to an operating system (if one exists). END is sometimes *not* required and sometimes performs the function of STOP or RETURN.

The ability to vary the dimensions of an array in a Subroutine is rare.

The Format arguments include those which are most variable. Most FORTRAN allow P scaling. The I/O of Hollerith text with and without processing capability is mostly available. Logical variable I/O is rare.

The I/O statements reflect hardware realities. The magnetic tape statements are used by some of the smaller machines to control unusual I/O devices (e.g., plotter, etc.). A great variety of I/O statements exists, much greater than indicated in the comparison matrix.

FORTRAN II permits EQUIVALENCE to reorder COMMON, for the most part, while FORTRAN IV does not.

Functions are at once the most flexible and rigid elements of FORTRAN. Among built-in and library functions, names vary slightly over several versions. In this department Univac 1107 wins the prize with 73 built-in and library functions. Function naming for statement and subprogram functions seems always to have been a problem. Thus, function name classically must be 4 to n characters ending in F or not if less than 4. The FORTRAN II I, J, etc., first character rule applies. Some F-II versions attempt to simplify the naming rule. F-IV obviates the rule by use of Type and Type Function statements. The CDC 160 and 160A and the TRW 330 allow no statement or subprogram functions, the 160 does not have built-in functions.

The DATA statement and BLOCK DATA subroutine are, of course, available only in F-IV to supply initial data at compile time.

Type declaration is also only available in F-IV. The F-II naming rules (i.e., Integer I, J, K, L, M, N if first character of a name) still hold but may be overridden by a Type declaration.

Statements are usually identifiable by number. The two most frequent maximum statement number values are 32,767 and 99,999.

Several compilers allow symbolic naming of statements. Philco 2000, RPC 4000 and PB 440 allow names. The introduction of statement names has quite an effect on the entire language for all statements that contain references to other statements must allow names rather than numbers.

FORTRAN Deck ordering involves a judgment. Some compilers (e.g., Honeywell "Automath" for H 800, H 1800 and H 400) require not only special ordering of statement categories but the insertion of special control cards and multiple pass compilation. It should be noted that so insignificant a point as the organization of a source deck can mean no compilation when moving to another machine.

Under special features are listed a few of the interesting departures from the "universal" FORTRAN. Features are merely listed without attempt at explanation.

The Comparison Matrix indicates the variability of the language. We have little or no information on compiler efficiency (time to compile), object program efficiency (time to run), hardware configurations, etc. The information presented here was taken directly from the manuals supplied by the various manufacturers.

One is tempted to ask: Why these differences? There is no single cause. Certainly the basic hardware differences make their contribution. Some compilers differ from others because of misinterpretation or a desire to improve upon what exists. It is possible to extend FORTRAN and thereby make it more suitable as a data processing tool. The reading of Hollerith text controlled by $ in SDS 900 or the BUFFER and ENCODE/DECODE statements in CDC FORTRAN 63 are examples. Non-compatibility is also a potential sales tool. Given a class of machines, one competitor can write language specifications that would permit the proper compilation of programs from the competing machines but effectively prevent reverse compatibility. This can be done by adding one or more features that the competitors do not have. Symbolic statement names would be a likely feature for this purpose, some unique form of subscripts would be another; the idea is to supply a feature that programmers will use frequently. The language specification will then call for the union of the competing languages plus the special, non-common features. Thus all competing programs will compile but programs written in the special language will in general, be incompatible with the competing machines. We do not suggest that any of the FORTRAN included in this study were dictated by marketing considerations. Nevertheless several compilers have characteristics that cause them to behave in the manner just described above.

The "universal" programming language, FORTRAN, is indeed universal but not all dialects are the same. We have presented some of the differences in the language and their effect on compatibility of FORTRAN programs. To those engaged in writing codes intended to be compatible with another machine now or in the future, caution is suggested. Such "compatible" programming requires great care and foresight. A practical result of this study has been the FORTRAN Infograph, which together with its detailed specializer should be useful as both a handy, fast and accurate reference for any FORTRAN user, and also as a kind of panorama of the "universal" language.

The author wishes to acknowledge the major contribution of C. L. McCarty, Jr., who together with the author, examined in detail, extracted and organized the basic information from the manufacturer's manuals. The efforts of Mr. McCarty and F. J. Oswald in performance of various functions leading to the Infograph and this paper are also acknowledged. The following companies have cooperated by supplying manuals: IBM, UNIVAC, CDC, Philco, Honeywell, Scientific Data Systems, General Precision, Packard-Bell, RCA, Autonetics, GE and TRW. ■

COMPILING A COBOL QUESTIONNAIRE

by STANLEY M. NAFTALY

COBOL has been in use for over three years now and a steady learning process on the part of compiler writers has been very much in evidence.

In late 1960, two or three compilers were unleashed on an unwary and, at times, over-eager computing public. By early 1962 this first wave of compilers had grown to more than a half dozen. The prime characteristic of this earliest effort, if there was one, was long compile times. For the most part this was due to the manufacturer taking the simple expedient of attaching a preprocessor to his assembly system . . . although the relative inefficiency of the processor itself helped.

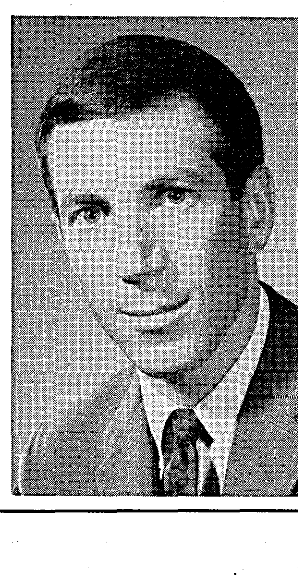
The second phase in compiler development might be termed the "Gee Whiz!" period. Everyone concentrated on compiling as fast as possible—card reader speed being considered a barely acceptable minimum—and the devil take the object code. This period saw the arrival of a plethora of compilers.

Currently we seem to be in a transitional phase. Present compilers are becoming quite sophisticated and efficient. The customer's role in this COBOL development period has up to now been that of using what is available. Suddenly the problem is how to evaluate a compiler system.

In this age of the operating system, it is no longer adequate to evaluate a COBOL compiler using the classic questions of "How many language features does it have?" and "How fast does it compile?

Today the user must consider the supporting software system as well as the compiler itself. It has become almost a truism to say that one cannot evaluate a computer intelligently without considering both the hardware and the software. Similarly, in the area of the COBOL, language, one must evaluate more than the compiler.

Lockheed Corporate offices produced a COBOL questionnaire* which attempts to develop a picture of the compiler within its framework of supporting software and also to measure the efficiency of the computer-com-



Mr. Naftaly, who is a member of the COBOL committee of CODASYL, is the assistant to the director of Systems Planning, Lockheed Aircraft Corp., Burbank, Calif. Prior to that, he was supervisor of IDP Programming and Operations at Space Technology Labs, Redondo Beach, Calif., and training manager for RCA-EDP in Los Angeles. He has taught COBOL at UCLA, and is writing a college text on the language, to be published by Prentice-Hall.

* I wish to thank Don Horigan of Lockheed Missiles and Space Company for all his help with the questionnaire upon which this article is based.

piler-operating system combination. This survey, with explanatory notes, follows.

I Language Features

A. Which elements of COBOL-61 have not been implemented?
   1) Will these features be implemented in the future and, if so, what is the target date for each?
   2) For those required elements that will be implemented, please give reasons for exclusion.

B. Which elements of elective COBOL have been implemented?
   1) If the "Move Corresponding" option, the "Compute" verb, the "Library" function, the "Enter" verb and/or "Segmentation" have not been implemented, please comment on reasons for these omissions and on any plans for future inclusion of these features.

C. Have any elements of COBOL-61 Extended been implemented? If so, which ones?

D. Please list and comment on any other extensions or modifications (present or contemplated) to the language as specified by CODASYL.

II Compiler Listings

A. Are the following produced by a compilation? Please describe the format and content of each.
   1) Source deck listing.
   2) Symbolic and /or object deck listing.

(If the compiler goes through an intermediate phase, we consider it advantageous to have a record of the coding produced in this phase. Some compilers do not give any record of the actual object code, but only of the symbolic).

(a) Buffer area assignments. (Alternate areas.)

(b) Actual memory assignments.

(If the compiler produces "floatable" object code, this listing can show locations relative to some base point).

(c) Compiler-generated subroutines.

(This refers to such things as Input-Output routines, rather than subroutines generated as a direct result of particular source language statements).

(3) Cross reference list.

(a) At source language level.

(An example here would be listing opposite each data element the sequence number of every procedure statement that makes reference to that element. Opposite each branch type procedure statement would be the sequence number of the statement to which control would transfer).

(b) At symbolic or object code level.

(The same techniques mentioned above would apply here in slightly different form. For example, rather than sequence numbers, memory locations might be used. Whether the cross reference is more valuable to the user at the source or object level depends on many things: the speed of compilation, whether the object code is produced in fixed or floatable format, etc. In general, if the system is efficient and complete enough to allow socrce language debugging, cross reference at the source level is far preferable).

4) Error diagnostic listing.

(Further inquiry should be made here concerning the breakdown of errors into classes. Does the compiler recognize minor errors, such as misstatement of the Object Computer name, and correct them (with notification)? Does it also differentiate between errors that preclude running the object deck but allow compilation to continue and those (and there should be very few) that halt a compilation).

B. Please indicate any additional listings which are produced.

C. Are the above listings optional? If so, by what means may they be omitted?

(This question refers primarily to the use of control cards versus operator action. Another question that well may be asked is, does the omission of one or more of these listings from a job speed the compilation at all?)

III Miscellaneous Compiler Features

A. Describe the standard file and/or reel labels supplied by the compiler as to format and checking procedures.

B. Variable data.
   1) May variable size data items and records be handled by COBOL-produced object programs?
      (a) What options, e.g., "SIZE IS integer-1 TO integer-2. . . . .," are utilized to manipulate data-items or records of variable size?
      (b) What options, e.g., "OCCURS integer-1 TO integer-2 TIMES. . . . .," may be used to manipulate tables of variable size?

(The question of variable data must be examined particularly carefully from an object program efficiency standpoint. Because of the hardware design of many machines, these functions can only be included in the compiler at extremely high cost to the user at object time).

C. Program Overlay.
   Is the ability to handle programs which exceed allotted memory through the COBOL Section-Segmentation option?
      (a) If so, how is the priority system implemented?
      (b) If not, what technique is available?

(The user must consider the penalties of having these other techniques use machine or assembly languages).

IV Compiler Operation

(This area is one of the most important in the questionnaire. In order to make efficient use of his computer-software system, the user must have the ability to load a systems input with several source program packages and let the operating system compile and/or load and test these programs without operator intervention).

A. Does the compiler operate under a systems monitor?
   1) What other systems are available under the monitor, e.g.. FORTRAN IV, Sort, Assembly System, etc.?
   2) Describe the monitor's functions briefly.

B. Is there a job monitor providing for stacked compilations?
   1) Are compile only, compile and load, and compile and execute options available?
   2) Describe biefly any test aids for compile and execute with stacked compilations.
      (a) Test data generator and fan out.

(The compiler must be able to analyze errors that have occurred in a compilation in order to determine if, in fact, a program can be executed. If so, the next step is for the test data generator to get the next part of the source package (the test data and its file allocations) from the system input and to properly structure these data and "fan" them out to the appropriate input units. If the computer is equipped with a large random device, this task may be accomplished by setting up all files on the random device and making the I/O portion of the

object program run in an interpretive mode. This technique is much more complex, but it precludes the necessity of disturbing tapes, etc., between compilations. At the close of a compile and execute test, the operating system must see to the production of whatever debugging listings are considered necessary—memory dump, tape prints, etc.)

(b) Dynamic sampler.

(This is a routine for use in the nastier debugging situations. Essentially it monitors the running object program and prints out the contents of specified locations or files at designated intervals. Other versions allow for the inclusion of a procedure trace as well).

(c) Automatic core and tape dumps.

(d) Others?

C. Hardware Requirements.

1) What are the minimum machine requirements for compilation, i.e., number of tape units, core size, card reader, printer, etc.?
   (a) Are options available for source input from mag tape or cards, etc.?
   (b) Are options available for on-line or off-line listings, etc.?
2) Can programs be compiled for and run on another machine type? If so, what types?
3) Was this compiler system originally written for one machine and now running on a newer, more powerful machine?
   (a) For what machine was it originally written?
   (b) What are your plans for upgrading the compiler system to take advantage of the newer machine's features?

V Compiler Efficiency

(For this extremely important and complex topic, the questionnaire can serve only as a beginning. Therefore, the answers to the questions, while important, should be considered as a guideline at most. Every effort should be made to construct and run benchmark problems in order to gain a more precise measure of compiling efficiency).

A. Compile Time

1) What is the speed of compilation in terms of number of COBOL statements per minute?

(This differs from COBOL cards in that a statement can easily occupy more than one card).

2) Are any formulas available to calculate compile time for a given program?
3) What is the approximate compile time (from COBOL to object code) for a program with two files (about 40 level number entries) and 30 procedure statements?
4) What is the approximate compile time (from COBOL to object code) for a program with four files (about 100 level number entries) and 200 procedure statements?

(The answers to 3 and 4 should specify whether or not card reading and listing times have been included).

5) Do symbol dictionaries fluctuate dynamically during compilation? That is, if one dictionary overflows while others are only half full, will the overflow be absorbed by change of allocations or will the compilation be discontinued?
6) Does the compiler provide for individually compiled program segments being combined at object time?
7) Is it possible to combine program segments in several languages (e.g., COBOL, FORTRAN, assembly system) at compile or object time?

(Questions 6 and 7 revolve around a level of the operating system and the object time executive more than around the compiler itself).

8) Does the compiler go through an assembly system phase? Please explain the general compilation technique.

B. Object Time.

1) What is the mix, in the object program, of subroutines and macros versus coding which is the product of analytical generators?

(By analytical generators is meant the technique of producing object code which is specifically tailored to the characteristics of the particular data involved. This is in contrast to the practice of inserting generalized subroutines to perform certain functions).

2) Is the Input-Output system an object-time interpretative one, or is it compiled as an integral part of the program?
3) Will the object program overlap read, write, compute? To what extent is overlap accomplished?
4) Will an object program (of a typical file-update type) be tape bound?
5) Does the object program run under some type of a monitor or executive system? What functions does this monitor perform? Please describe the monitor briefly.
6) Is the object code produced in a fixed or relocatable format?
7) Is the number and location of input-output buffer areas (alternate areas) fixed or can it vary with memory availability?
8) Does the input-output system operate in the locate or transmit mode? That is, when a data record is read into a buffer area within memory, does it remain in its original location unless specifically moved in some way by the programmer (locate mode) or is it automatically transferred for processing to a file area (transmit mode)?
9) If the I/O system operates in the locate mode, what means are used to modify addresses of statements accessing the record (e.g., index registers)?
10) Can the program configuration (e.g., tape and/or channel assignments) be altered at object time without recompiling? If so, how?
11) Please give some measure of the size of COBOL-produced object programs.

(Again, the user must realize that answers to these questions can be approximations at best. It is felt, however, that these questions help to round out the questionnaire and that the answers can be useful).

(a) What is the minimum amount of memory taken by an object program, not including generated code (i.e., taken by the input-output system, subroutines, etc.)?

(b) What is the "average" amount?

VI Compiler Support

A. Availability.

1) Date of full compiler release to the field.
2) How many installations are presently using COBOL system? To what degree?

B. Support

1) By whom was the compiler written?
2) By whom will it be maintained?

3) What procedures have been established for reporting bugs, inefficiencies, etc., in the compiler system?
4) Are any COBOL users groups in existence?

C. Documentation (present or planned). Please describe the content of each manual briefly and give availability date.
   1) What manuals are available for programmer use?
      (a) Programmers' reference manual?
      (b) COBOL system and object program operating manuals?
      (c) Programming Aids manual? That is, a manual that, for each verb and data description option, shows how much coding is generated.
      (d) Compiling Aids manual? That is, a manual containing check lists for pre- and post-compilation.
      (e) Error print out analysis and explanation manual?
      (f) Other.
   2) What system manuals are available?
      (a) Listing and/or flow-chart of compiler?
      (b) Manual describing the various subroutines, analyzers, and generators in the compiler?

VII Miscellaneous

A. Please comment on any other feature of the compiler or its operating system which you feel bears mention.

B. Please state any future plans for inclusion of additional language features, modification of the compiler to effect more efficient compilation and/or object programs, inclusion of preprocessors such as D-TAB, or additions to the testing system.

C. In all of the above questions, where an answer is not currently available, please give a date for availability of that answer and state reasons for current unavailability.

D. Our current plans call for sending out two COBOL source programs for necessary modification and compilation. Full instructions will accompany these programs.

(The user should request (with the benchmark problems) a complete record of all changes made to the source program).

The data processing customer who uses this type of a questionnaire will find that some computer manufacturers are "most reluctant" to answer. One example is a reply we received telling us that, "The answers to all these questions are in our manuals. We would be happy to send you a set of these manuals." Several manufacturers indicated that they had not considered many of these points in building their compiler systems. This is illustrative, in my opinion, of a lack of knowledge on the part of the compiler writer of the realistic uses of the COBOL language and other software. The only way that this gap can be closed is for the manufacturer to work more closely with the user in the design of software. This is being done today on the COBOL Committee of CODASYL, but there it is at the language specification level.

This questionnaire technique has been chosen for several reasons. We feel that a questionnaire is truly more than the sum of its parts. Any one question will obtain an isolated answer while a series of well structured, related questions prompts considerable digging on the part of the respondent. Sometimes this effort even creates an attitude of "take a completely new look." A prime example of the latter case is that one computer manufacturer told us that he had spent three weeks in answering our questionnaire (for four different machines), but that he had learned a great deal about his own software. So much, in fact, that he has since published a manual entitled, *Comparative Study of____COBOL Compilers*, and has made this manual available to all of his marketing and support personnel, as well as to customers and prospects. The manual is based almost entirely on the Lockheed questionnaire.

However good a questionnaire may be it should be used only as a basis for further study. Each user must think through the evaluation problem carefully in terms of his own situation. In addition, the answers to the various questions must be weighted and the new picture that emerges re-evaluated.

It is time then for the user to refine his evaluation techniques. This can only be done by the expenditure of effort in first, the analysis of his data processing job and second, the determination of what facilities are important to him in a higher level language.

The hue and cry of the computer manufacturer today is the "Total System." Nowhere is this concept more important than in the evaluation of computers. And nowhere is it more imperative that the user require the manufacturer to demonstrate that his "total systems" concept extends to software. ■

FORTRAN VS. COBOL

for business dp

by M. D. FIMPLE

Because FORTRAN was developed basically as a mathematically-oriented language, its use in business data processing has, from its inception, been the subject of considerable skepticism on the part of some users. We at Sandia are no exception. Prior to the installation of an IBM 7090, which we have been using for most of our business data processing since 1961, several programming languages were studied and evaluated for use on existing and anticipated problems.

In 1963 when COBOL became available for the IBM 7090, this language was also evaluated for our use. The methodology and results of this test are reported here. But because a standard against which COBOL must be compared exists in the presently-implemented FORTRAN programs, it may be of interest to trace the developments which preceded the tests.

In 1961 when our 7090 was installed we decided to use 9PAC on several high input/output type problems and to use FORTRAN on the remainder of the workload. FORTRAN* was put into use at the time, mainly because it was a well-developed language which was easy to learn and which enabled the programmer to control the logical flow of a problem. The proportion of jobs performed with FORTRAN has since increased, and no new applications of 9PAC are anticipated.

FORTRAN II could not stand alone as a data-processing language. There were two possible routes to making FORTRAN efficient in this type of application. ments of data processing.[1] This second route was chosen mainly because it enabled us to use the standard FOR- First, the compiler could be modified to accept certain statements required in data processing and to produce efficient data-processing code. Second (and this was the route taken by Sandia), the subroutine feature of FORTRAN II could be used to provide all the require- TRAN Compiler and Monitor System maintained by IBM.

The FORTRAN Buffer package produced at the Western Data Processing Center (UCLA)[2] was used initially on a few problems. Although our experience with the WDPC package was favorable, its use was

A programmer with the Sandia Corp., Albuquerque, N.M., Mr. Fimple heads the firm's Statistical Programming Project. He was responsible for the writing of WRAP (Weighted Regression Analysis Program), a multiple regression program. He has been associated with Stromberg-Carlson, Aeronautical Radio Inc., and Bell Aircraft, and is currently chairman of the SHARE committee on regression analysis. He holds an MBA in statistics from the U. of Buffalo.

*Fortran II, Version 2

ON MAY 8, 1964 AN INDUSTRY WAS BORN!

"The Copyright Office . . .has reversed a long-standing policy and has registered the first copyright on computer programs."

New York Times
May 8, 1964

Copyright protection makes inevitable the extensive development and mass marketing of proprietary computer programs. A great variety of absolutely top quality programs will become available to the computer user at very low cost. The myriad activities of promotion, distribution, education and service to support these programs will follow. Capitalistic development of the proprietary program industry is an excellent solution to many of today's programming problems.

The Digitek Corporation's first proprietary program is described on the following pages.

THE DIGITEK CORPORATION

announces a new dimension in compiler systems

FORTRAN 4D

for the **IBM 7040/44/90/94** Data Processing Systems

Machine Time Savings $30,000/mo.

SYSTEM	SYSTEM TIME	RATIO	EXECUTION TIME	RATIO	TOTAL TIME	RATIO
F4D - SB	555 sec.	.349	316 sec.	.818	871 sec.	.441
F4D - JM	1397 sec.	.880	332 sec.	.860	1729 sec.	.876
IBFTC - V9	1588 sec.	1.000	386 sec.	1.000	1974 sec.	1.000

The first System F4D is now being installed at the Hughes Aircraft Company IBM 7094 facility in Culver City, California. The above times were taken during initial runs in July. Production use starts in September. Details are available upon request.

System F4D

System F4D consists of two subsystems, F4D-JM and F4D-SB. Each subsystem consists of a FORTRAN IV compiler, object time package, and support programs. The F4D-JM compiler directly replaces IBFTC under IBJOB and compiles to IBMAP. F4D-SB operates directly under IBSYS and compiles to relocatable binary.

Any program which will compile and execute correctly under IBSYS on the IBM 7040/44/90/94 will compile and execute correctly on the corresponding F4D system. The F4D language includes the IBM and ASA FORTRAN IV languages as proper subsets.

Lease includes system integration and maintenance.

AUTOMATIC DEBUG

control card option ▫ compiled into object program ▫ program flow identified by source line number ▫ snapshots of changed variables ▫ subscripts checked for size and range ▫ control transfers checked for target ▫ subprogram arguments checked on entry ▫ I/O records counted ▫ I/O formats verified ▫ automatic termination ▫ detailed trace of last 100 statements

PROGRAMMED DEBUG

preprogram package ▫ compiled into object program ▫ activated by line number or variable name ▫ contingent on object time calculations ▫ postmortem dumps

LANGUAGE

mixed type expressions ▫ general expressions as subscripts ▫ no limit on number of subscripts ▫ negative and through zero subscript ranges ▫ backward and through zero DO loops ▫ placement of declaration statements immaterial ▫ symbolic parameterization of declarations ▫ context implied declarations ▫ multiple subprogram entries and returns

DIAGNOSTICS

every statement carried through all diagnostic levels ▫ erroneous constructs indicated on listing by undermarking last character of construct ▫ accurate and concise error messages given for each indication ▫ each occurrence of misused identifiers indicated and local context requirements given ▫ recovery from errors by global analysis ▫ complete diagnostics generated on first computer run ▫ execution always reached

OBJECT CODE EFFICIENCY

redundant calculations eliminated within statements and over statement blocks ▫ loop independent calculations evaluated prior to loop entry ▫ array indexing optimized by recursive address calculation ▫ induction variables materialized only if required ▫ constant arithmetic done at compile time ▫ integer powers evaluated by multiplication ▫ code generators for intrinsic functions

COMPILER EFFICIENCY

the F4D compiler consists of 10,000 instructions ▫ compiling speed varies from 500 statements per minute on the IBM 7040 to 2500 statements per minute on the IBM 7094

DIGITEK CORPORATION
12922 Panama Street
Los Angeles, California

limited because of conflicts between it and our library routines. In early 1963, when IBJOB was delivered by IBM, we were faced with the decision of whether or not to switch to FORTRAN IV (F-IV). At this time a typical problem was selected which was being run in our shop, using the WDPC buffered package. This problem was run using three different object decks, F-I (unbuffered), F-II (buffered) and F-IV, it is not the purpose here to report on this test, but the F-II (buffered) program was so much faster than the other two programs that two decisions were made: (1) We would not use F-IV until (unless) its implementation was greatly improved, and (2) we would initiate a project to overhaul the WDPC buffer package to make it compatible with our data-processing subroutines. This latter project was so successful that since mid-1963, our version of the WDPC package has been used on practically all FORTRAN programs with large amounts of I/O. It has been our experience at Sandia that the use of the WDPC buffer package results in improvements in FORTRAN program running time of 20 to 50 percent over the standard FORTRAN I/O routines.

When Commercial Translator was delivered by IBM, one rather large data-processing problem was performed using this language. Reaction to Commercial Translator was moderately favorable; but when IBM discontinued its development,[6] we, of course, decided not to make further use of it.

When COBOL was made available on the 7090, we started making plans to evaluate it for our use. The most important thing we hoped to gain was increased speed in performing our data-processing problems. Since we had previously observed some rather dramatic differences in running times between various programming systems, we expected that COBOL would give us significantly reduced running times on many of our problems. Another gain from our standpoint in using COBOL would be compatibility with other data-processing installations. Since many installations are using COBOL, other installations are going along for the sake of compatibility. Other things being equal, it would be possible to justify a switch to COBOL on these grounds alone; ultimately, however, we decided that we would also like to know that COBOL† does a better job than FORTRAN before adopting it.

The selection of a test problem for comparing two programming languages is highly subjective. Since we are evaluating the languages with respect to their data-processing ability, we decided that the problem should be restricted to input/output and logical decision making, these being the main components of business data processing.

Selected was a problem which had already been programmed in FORTRAN. This problem involved a large volume of input, a very substantial amount of logical decision making, a large amount of computation, and very little output. The problem was redefined in such a way that the input and logical aspects were unchanged, but the computation was completely eliminated, and the data which was previously used in calculations was written on an output tape, giving a substantial amount of output.

The I/O for this test was done entirely in BCD. It is possible that the use of binary I/O would change these results. We will probably evaluate this aspect of the problem in the near future, but we do not feel that the results will be very different from those which we will report here.

The test problem is given in very broad outline form in the flow chart of Fig. 1. There were 30,000 logical input records. The input was divided into two reels to provide tape switching, and test was run using several physical block lengths as shown in Table I. The output consisted of 239 15-line arrays, or a total of 3,585 lines.

Fig. 1

MACRO FLOW CHART OF TEST PROBLEM

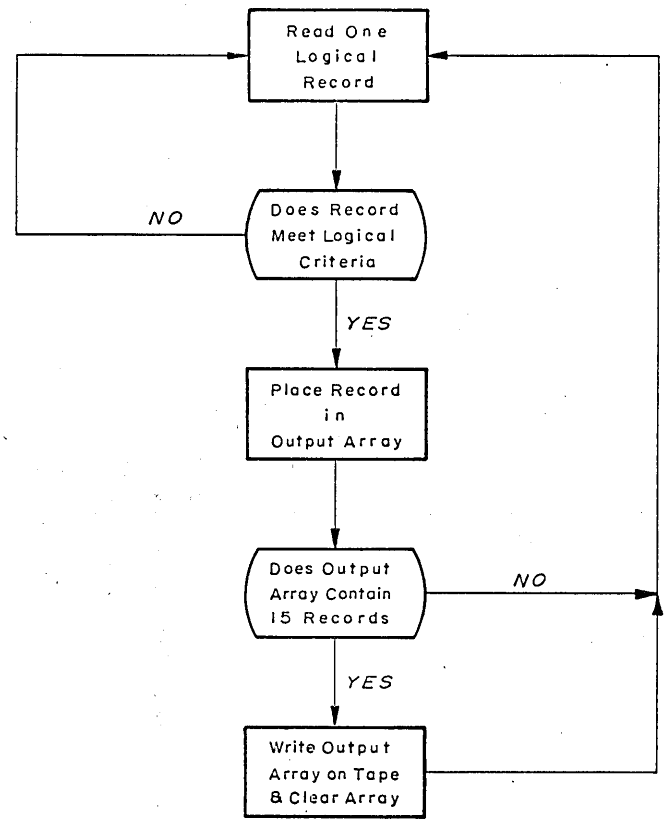


TABLE I

Block Size No. of Cards	No. of Words	FORTRAN II (minutes)	COBOL (minutes)
1	14	5.19	5.55
2	28	3.85	4.19
4	56	3.60	3.92
8	112	3.44	3.78
15	210	3.41	3.74
30	420	3.37	3.73

test findings

Programming Languages

The conclusions concerning the programming languages are largely subjective, and it should be recognized that they represent the impressions obtained by the writer in performing this evaluation.

In 1959, when I first encountered FORTRAN, my only computer experience had been with the Bell Labs Interpretive System for the IBM 650 and very briefly with the 650 SOAP assembler. At that time I was able to learn FORTRAN adequately in about two weeks. Since 1959, I have had experience with several computers and programming systems including Commercial Translator (CT) which is in some ways similar to COBOL. I was able to learn CT well enough in about three weeks

†The version of COBOL evaluated in this report is the version included in Version 2 of IBJOB.

to check out several small test programs. The time required to learn COBOL well enough to check out the test program used here was of the order of six weeks. This longer time requirement was, I feel, partially the result of the inadequate documentation and partially the result of the fact that I was working in near isolation so far as COBOL experience was concerned. Had I not been able to obtain answers to specific questions by calling the IBM analyst, I would probably never have obtained a successful run. The main reason the learning process was so slow, however, was the language itself. The COBOL language is defined by a burdensome number of exacting rules, many of which appear to be unnecessary. The programmer eventually get the impression that a large part of the English language has been set aside as "reserved words." So much extraneous verbiage is used in the language that the program, in this case, required 2.5 time as many cards as did the FORTRAN program.

The conclusions regarding COBOL and FORTRAN can be summarized as follows: COBOL is a verbose, highly restrictive language, difficult to learn and use. FORTRAN, on the other hand, is brief and precise, yet allows the programmer much flexibility in its use and is easily learned and applied.

Documentation

Because of the verbosity and restrictive nature of the COBOL language, good documentation in the form of well organized programming manuals is of primary importance. Unfortunately, such documentation is not available to users, although there are now several COBOL training manuals commercially available.

IBM has provided two manuals[3,4] devoted exclusively to COBOL, and other information is contained in the IBJOB Processor Manual.[5] These manuals were, however, written as reference manuals and are not suited for training purposes, partly because they present almost no programming examples. While it is possible to learn to use COBOL from these manuals, the process is extremely difficult.

The documentation (programmers manuals) on FORTRAN II is not much better, but it has never been a real problem because of the relative simplicity of the language.

Diagnostics

The diagnostics in the COBOL compiler are of about the same quality as those in the FORTRAN II compiler.

Object Code

The object code produced by COBOL takes more core than does the equivalent FORTRAN program—almost 20% more for the program studied. In addition, it appears that the COBOL main program does less of the actual work than does the FORTRAN main program, since the COBOL program accomplishes much of its job through calls to systems subroutines.

The symbolic listing of the COBOL object code is difficult to follow, since it uses compiler-assigned names for all variables instead of those assigned by the programmer. This feature is almost certain to increase debug times significantly.

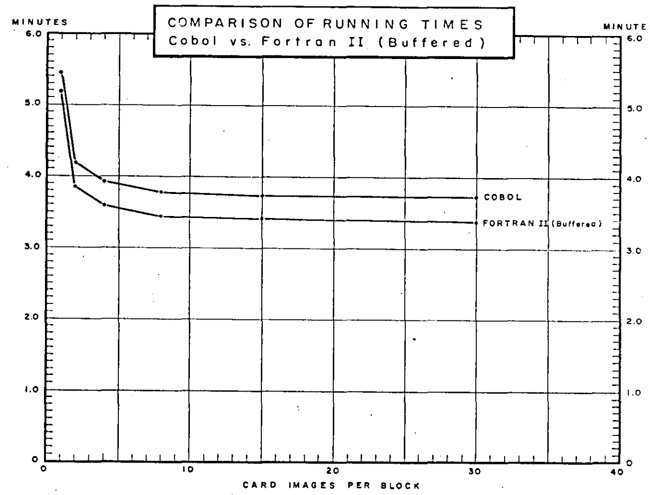
Running Times

The most surprising thing about the test was that COBOL failed to give any improvement in running times for this problem. Inspection of Table I will show that the FORTRAN II program gave slightly better results for all the block lengths tested. Fig. 2 gives a graphic picture of this comparison.

Language Environment

We are using an IBM 7090, which is a binary word machine. FORTRAN was designed to run efficiently on this type of machine. COBOL. it would appear, was designed to run efficiently on a character-oriented machine. It is the feeling here that the selection of language is more dependent on the type of machine than on the type of problem. We have observed that FORTRAN on the 7090 does at least as good a job of data processing as does COBOL, while being much easier to use. We would not expect FORTRAN to compete with COBOL on a character-oriented computer. Possibly future decisions by language-designing committees should be influenced by whether more data processing is to be done on binary or character machines.

Fig. 2



recommendations

The only possible justification for a switch to COBOL at this time would be for the sake of compatibility with other data-processing installations. Such a change would, for the present at least, entail a large training cost. Because of improvements which can reasonably be expected in the language, implementation, and documentation, this cost might be significantly lower in the future. Therefore, the following recommendations have been made to Sandia Corporation data processing management:

1. Continue to use and develop FORTRAN II as a data processing language.
2. Periodically re-evaluate COBOL, expanding the evaluation to include binary input/output and a greater variety of problems.
3. Make the results of such studies available to other computer users. If nothing else, this will help to provide stimulus for improvement of COBOL.

We presently have a good, proven data processing system at our disposal enabling us to concentrate on the solution of other problems without having to nurse COBOL through its infancy. If COBOL does eventually evolve into a superior language we will then make use of it.

REFERENCES

1. Robbins, D. K., "FORTRAN for Business Data Processing," *Comm. of ACM*, July 1962.
2. WD-F-IOF, SHARE Distribution 978, September 1960.
3. "General Information Manual, COBOL," IBM Form F28-8053-1, 1960.
4. "IBJOB Processor, Part 5: COBOL Compiler (IBCBC)," IBM Form J28-6260.
5. "IBJOB Processor," IBM Form C28-6275-1.
6. Rosen, Saul, "Programming Systems and Languages, A Historical Survey," *Proceedings, Spring Joint Computer Conference*, 1964.

CENTRALIZING COMPUTER STUDIES

Every year, under the auspices of the RAND Corporation—famed non-profit think factory—a group of outspoken, articulate and experienced computing professionals sit down for an all-day discussion of topics deemed critical to their profession.

In the past *Datamation* has published lengthy condensations of these conversations, usually divided into two parts. In the interest of order and brevity (this year's final edited transcript added up to 208 double-spaced pages), we have this year decided to isolate for our annual report the headiest and most vital topic for condensation. In the judgment of some of the RAND Symposium prime movers, that topic was the establishment of a U.S. Computer Institute. Among the participants:

Robert Patrick (chairman), consultant, Northridge, Calif.
Jack Granholm, Informatics Inc., Sherman Oaks, Calif.
Jack Roseman, CEIR Inc., Arlington, Va.
Bertram Herzog, Ford Motor Co., Dearborn, Mich.
Brad MacKenzie, Burroughs Corp., Pasadena, Calif.
Paul Armer, The RAND Corp., Santa Monica, Calif.
Herbert Bright, Philco Computers, Willow Grove, Pa.
Richard Hamming, Bell Telephone Laboratories, Murray Hill, N.J.
Frank Wagner, Informatics Inc., Sherman Oaks, Calif.

The possibility and advisability of establishing a national computing institute in the United States was one of several topics discussed at the last RAND Symposium, held last fall at the RAND Corp., Santa Monica, Calif. Any group of knowledgeable people could have a field day with this topic, and this group of computer experts was no exception. Several questions immediately come to mind:

- What would such an institute concern itself with?
- How would it be organized?
- How would it be financed?
- How do we get started?
- Who needs it?

What is required at this point in the history of such a proposal is not so much a tidy, prefabricated package ready to put into action, as an idea free-for-all. Which is what the RAND Symposium supplied. Certainly there was no shortage of provocative thoughts. Perhaps the only missing ingredient: opposition to the idea of an institute.

Assuming the much-needed role of organizer, Hamming delineated what he thought were proper functions of the institute. "I'm going to give a list of things that a National Institute (government-financed) might do.

"Item 1 is **hardware**, and I use NACA* as an example. At various stages of the history, the people in NACA worked on the development of airplanes, airfoils, and so on. I see no reason why an institute should not concern itself with the design of components. I don't think they *should* necessarily build computers at this stage.

"Item 2 is **standards**, similar to the work done by the National Bureau of Standards. This involves the making . . . distribution . . . promulgation . . . and certification of standards. For example, if there were specifications for a standard FORTRAN, the institute could devise programs to test whether a given compiler fits the standard. Again, there would be no guarantees involved.

"Item 3. The institute could subcontract *research* in much the same way that the National Institutes of Health does. The tasks might be to locate individuals who have good ideas relative to computing, and then to finance them.

"Item 4 is **publications** . . . the publication of books, pamphlets, etc., particularly oriented toward education. I think they should look for things that indicate how computers can be used to foster education. This should not only be directed to education on computers, but education of other subjects aided by computers.

"Item 5 is **symposia.** The institute should sponsor symposia in various areas at various time. A situation arises where a particular subject has come to a critical point and it is wise to get a large body of people together to talk at one time and produce a detailed symposium discussion (much as the one on large-scale memories several years ago).

"For Item 6 I have **special areas.** One that I might cite as an example is language translation. Another example might be the problems in the Internal Revenue Service in producing information retrieval systems somewhat similar to that used by the airlines for reservations. This might be a situation which you might not expect commercial manufacturers to develop. There were enough people interested in the airlines reservations problem to get the commercial boys interested in working on it.

"**Control theory and automation** is my Item 7. There is a large body of knowledge on the theory of control using Nyquist diagrams. . . There is, however, no such body of theory in the area of command and control. This is obviously a field for research.

"Item 8 . . . is up-to-date **surveys** of current state of the art and predictions for the future. What are your resources now? What will be the effect of some development on unemployment?

"The final item on my list is **social consequences.** The federal government obviously needs a source to turn to as it has to pass more and more laws affecting social welfare: how to control automation; unemployment and its consequent dislocations. We need an unbiased source of information (not the manufacturers) within the government to furnish information.

"These are all possible topics," Hamming continued. "To propose that an institute attack them all at once is too much. Even to have all of these items in one place is not necessarily a good idea. It might be better to have them scattered. Still, these are some of the things that a National Institute of Computers might take on."

Additions to and modifications of these functions soon arose. Roseman: ". . . I feel that if they did nothing but handle that ninth item (social consequences), it would be worth it." Hamming: "I agree, but I would venture to say that to try to attack number 9 without a good many of the others would be really working in a vacuum. . ." Armer: "I'd like to make a small addition to number 4 (publications), and that is that the institute should make an attempt to go out and get those texts that we were talking about earlier. Another suggestion . . .

Fig. 1. Institute Functions

Area	Current Media
1. Hardware (a la NACA)	IBM, Bell Labs, Texas Instruments
2. Standards (a la NBS)	Underwriters Lab
3. Supported Research	Universities, Foundations
4. Publication & Requested Texts	ACM Journal
5. Symposia	JCC's
6. Special Areas	
7. Control Theory & Automation	
8. Surveys	
9. Social Consequences	
10. Measures of Performance	

*National Advisory Committee for Aeronautics, since renamed the National Aeronautics and Space Administration (NASA).

the institute [should] go about getting some appropriate metrics for the field. There should be metrics with respect to measuring hardware and people." MacKenzie: "With regard to the first point (hardware), I would be more interested in hardware organization than components, although both are important."

Referring to Fig. 1, Patrick explained, "I took the liberty of putting a second column on the blackboard, showing where some of these things are already being done in some form or another. What a centralized agency will do, in the form we've been discussing, is to have the federal government underwrite the research for the littler manufacturers."

Hamming: "Let's take Item 3. If the universities don't support research, then the government supports research through the universities – so that Item 3 is a litttle bit wrong. I'm thinking of the government supporting research through the universities through contracts – through the people who have the ideas."

"The analogy that we have there with NACA (see Fig. 1) is really very good," chimed in Wagner. "NACA has never produced any hardware that really competed with industry. Even when they produced aerodynamic designs for the B-70, the designs were not complete, but they plowed up an awful lot of ground and helped to turn under a lot of things that were useless. The end result was that all the competitors for the B-70 project had available to them all these ideas, and out of those developed the practical plans. I think the same sort of thing could be done with computer organization and in computer component research . . ."

source of financial support

However much agreement may be reached on the necessity for such a national institute, and what it would concern itself with, there still arises the need for funds to establish and maintain it. The natural or acquired instinct today: look to the federal governement. But then one thinks of the bureaucracy, and tries other approaches.

"One thing we might do is try to build a little fire under the AFIPS organization," Wagner commented. "Individually and collectively maybe we ought to tell AFIPS that this is a very important subject and they should drop everything else and get on with it. All the other things they are trying to do might fall into line if they would work on this problem first."

Granholm: "It sounds like the hard way to get anything done."

Hamming: "Don't bother to ask ACM to do it, either."

Similar sentiments were expressed about BEMA (Business Equipment Manufacturers Assn). Another source of funds, suggested by Wagner, was through the establishment of a quasi-governmental organization, such as the Triborough Bridge Authority in New York. "It's conceivable," Wagner said, "that you could get the government [to levy] a tax on sales and rental of [computer] equipment, and then turn the money over to an outfit operated on a taut-ship basis not ordinarily found in government organizations."

That leaves foundations as the largest private source of financial support. Said Bright: "If there are good reasons why the government should not be the primary supporter of such an institue (and I can think of several), then perhaps the Rockefeller and Ford and other foundations that have large sums of money for furtherance of human knowledge and welfare are the appropriate sources of support for such an activity."

Minutes later, this discussion followed:

Wagner: "An individual cannot go to the Rockefeller Foundation and ask for $500,000. He can't even get in the door. But I think perhaps the president of AFIPS could. He represents three societies with 100,000 reputable scientists behind him."

Armer: "His conservative Board of Directors won't let him."

Wagner: "Then let's do something about getting an *un*conservative Board of Directors."

Herzog: "I don't think that these foundations are as unapproachable as you make out. If a collection of people got together and decided that this was a legitimate piece of business to be in, some foundation people would be very happy to support it.

Hamming: "As a matter of fact, if you want a proposal. . .you could go to the foundations and ask for money for a study. You could state what needs to be done and probably get 25 or 50 thousand dollars."

operational aspects

And so it continued, with ideas from people with a variety of backgrounds and experiences. In fact, some of them had received grants from foundations which enabled them to make a particular study. It was this actual-experience aspect of the staffing of a computing institute that troubled Wagner. "A little earlier today, there was a statement tossed about that caused everyone to shake his head. Someone said that the higher-level languages, and to some extent the processors of those languages, are made by people who never had occasion to use them. I am concerned with the ivory-tower aspects of the staff of such an institute—namely, that they would be solving problems that they would never have to face. One way to attack this problem, it seems to me, would be to establish that some fraction of the staff was continuously rotated in and out with the men in the front line trenches. Perhaps you could [get organizations] like RAND to give a man a sabbatical leave for one or two years to work at the institute and to bring to the institute continually fresh points of view."

Showing equal concern was Patrick. "I was just musing about Item 3 (supported research)," he said. "It's not real clear-cut to me just what research should be supported and where we would get the all-encompassing wisdom to know."

Hamming: "My suggestion was to do this by analogy with the National Institutes of Health. People come with ideas and you examine the idea and the man. If he's a real hot man and the idea looks like it should be supported, then you give him the money to do it. You don't try to initiate the research yourself."

Patrick: "It was the 'it ought to be done' part that bothered me."

Hamming: "You get a review board of good men. This is the way all the foundations do it. They have a panel of good men who evaluate projects. They call on outside consultants to help. . .To be sure, a lot depends on the quality of the evaluators."

To be sure, also, if a U.S. Computer Institute ever comes about, it would be a long time in the coming—especially if its areas of activities encompass the scope discussed here. There must be a Parkinson's Law to the effect that the more you include for consideration, the longer final action will take. Even among the participants, no agreement could be reached as to the relative importance of the categories, or which should be left outside the realm of the institute.

At the conclusion of this discussion, a show of hands was requested of those who thought "something should be done toward establishing such an institute." Eight or 10 hands (from among the 18 present) were counted.

ACM NATIONAL CONFERENCE

by HOWARD BROMBERG, Conference Chairman

The goal of the 19th National Conference of the Association for Computing Machinery is in all probability not much more distinguishing, antagonistic or incongruous than the goal of any other scientific meeting, i.e., the dissemination of good and timely information. We attempted to report on various aspects of the computing world and, in so doing, strived to maintain a relationship with the realities of the business world as well as with the scientific environment.

Our approach to this goal was made all the more easy because of the wording of the Constitution of the ACM. In particular, it states the purposes of the association to be to advance the sciences and arts of information processing and to promote the free interchange of information about the sciences and arts of information processing. Consequently, the General Conference Committee stressed the importance of the technical program as the means to achieve the purposes of the society. We felt that only through such emphasis would we be able to reflect appropriately the professional attitude both of the society and of its members.

With the support of both the ACM Editorial Board and the ACM National Meetings Committee, the ACM Council passed a resolution to change the basic structure of its national meetings. This resolution allowed for the publication of complete proceedings of the papers, as well as for the refereeing of each individual contribution. With this as our tool, the General Conference Committee went to work.

It was decided from the outset that all papers would go through a refereeing process. That is, papers which were contributed, solicited and even invited, would have to be submitted in full for the scrutiny of the referees. At the same time we attempted to introduce the "Hotseat" technique wherever practicable. This technique used the referees as part of a panel of experts who would comment upon each paper within a particular session immediately after its delivery, and would question the author on various points. The choice of the referees was made because they would be individuals most familiar with not only the particular paper, but also the general topic. Referees were chosen from the list supplied by the ACM Editorial Board.

As was expected there has been a great deal of criticism concerning our insistence upon the submission of full papers, as well as the refereeing requirement. As a result we probably did not receive as many papers as previous conferences, where only 100 to 500-word abstracts were required. We think, however, that we have made a contribution to the membership by automatically eliminating those submissions which an author himself felt not worthy of full explication in a complete paper. In addition, we hope that the keen eyes of the referees will expose the appearance of any charlatans, cutpurses and delinquents. Those papers that were submitted in full were refereed by at least two experts. We found that approximately 55% of all the papers received by the Technical Program Committee were rejected. Each author of a rejected paper received a personal letter from the Technical Program Chairman, stating why his paper was rejected. To date I know of no author in that category who has complained, fretted or "wafted a sigh from Indus to the pole."

We spared no expense in minimizing the frills normally associated with meetings of this type. There is no plenary session, for example, and the General Conference Chairman makes no opening remarks. Neither does the Governor, the Mayor, or any public official welcome the attendees.

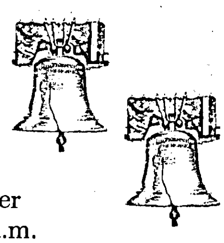
What we *did* was to keep the meeting confined to a full three days with the first technical session starting at 9:00 a.m. on the first day of the Conference. This allows the Monday preceding and the Friday following to be available either for travel or for additional committees to meet. The evenings were, as far as possible, left open for other technical or social meetings. Ladies' programs, social events and field trips were not dwelled upon to any great extent. Perhaps you will find that the conference is a terrific bust. If that be the case, you must complain loudly and appear at the Council Meeting in an attempt to redirect the emphasis of future conferences.

I think the foundations upon which we built this conference were very sound. I think the Technical Program, which may at first glance look to be a bit meager, (i.e., no six-way parallelism) reflects the professional attitude of the contributing members of the community. I think that running a conference is an extremely onerous and trying avocation, and that those individuals who gave of their time to serve on the arrangements committees, as well as those who made the effort to write papers, deserve the compliments of the Society. Finally, I think that everyone should attend.

THE EXHIBITORS . . .

	Booths
AT&T	410-420-430-440
California Computer Products, Inc.	740-750
C-E-I-R, Inc.	550-560
Commerce Clearing House, Inc.	720
DATAMATION	320
Digital Equipment Corp.	130-140
Electronic Associates, Inc.	310
IBM	150-160-170
ITT	490
Nielson Publishing Co., Inc.	340
Pastoriza Electronics Inc.	530
Prentice-Hall, Inc.	400
RCA	470-480
The Service Bureau Corporation	500-510
Soroban Engineering, Inc.	100-110-120
Teletype Corporation	450-460
John Wiley & Sons, Inc.	520

TECHNICAL SESSIONS . . .



Real-Time Systems

J. J. Donegan, Chairman
Goddard Space Flight Center
Tues., Aug. 25, 9:00-10:30 a.m.

The computer field today is expanding spectacularly on many fronts. Associated with this development is a similar development in the art of digital data transmission. Today we move digital data at tremendous rates over long distances routinely. It is estimated that by 1975 half of the income at AT&T will be derived from data transmission. These advances have spawned large automatic real-time computer systems, where computers are used as the heart of information processing systems with tentacles which reach out all over the world. It is self evident that these systems can be used to automate, control, regulate, coordinate, standardize, teach, inform, manage, systematize etc., all kinds of processes on worldwide scales. In the next 10 years, it is anticipated that the on-line, real-time, communication computer network will have a major effect on business activity and organizational concepts.

The art of development of large real-time automatic computer program systems is in its infancy. Such systems require unusual investments in computer programming talent as well as computer system engineering talent, not to mention money. Implementation, management and operation of these systems pose new problems on a scale not encountered before.

Since we are in the early growth cycle of such systems and since such systems will occupy the attention of a large segment of programming talent and computer associated systems personnel, it is important that we study existing systems with a view toward extracting as much knowledge and experience from these developments and implementations as possible, to be used in future development of large systems.

It is the purpose of this session to examine three of these systems, two existing (and operating) systems and one new system under development. Since the field of space exploration is also in its infancy, it appears appropriate that these three systems deal with space projects. However it should not be inferred that these large real-time computing systems are peculiar to the space field. There are the airline reservation systems, SAGE, communication switching centers and the many Department of Defense Command and Control Systems, to mention a few.

In the first paper a description of the experiences with the Goddard computing system, a currently operating on-line real-time system is examined. It is the computing system which provided prime computing support for all the National Aeronautics and Space Administration manned spaceflight missions to date, including the Redstone flights and flights of Shepard, Grissom, Glenn, Carpenter, Schirra and Cooper. Recently it has been used for prime computing support for the Saturn SA-5 and SA-6 flights, the first Gemini GT-1 flight, the Centaur flights AC-2 and AC-3, etc.

The second paper describes a computer-centered, automated, worldwide network testing system called CADFISS. Since this testing philosophy may be applicable to any computer controlled data or information processing network, it is presented as a currently operating system.

A third paper describes the Manned Spacecraft Center's RTCC system which will be used to provide prime computing support for Gemini Rendezvous and Apollo missions. It is a system under development.

Invited papers I

I. E. Block, Chairman
Auerbach Corporation
Tues., Aug. 25, 11:00 a.m.-12:30 p.m.

"A Bull's Eye View of Management and Engineering Systems" by Anthony G. Oettinger is a refreshing and realistic look at the computer in the management environment. The author sharply contrasts current information processing systems which generate masses of data presumably for management, with the role which properly designed systems could play as a catalyst to management planning and decision making at all levels. He comments that "the computer profession has shown a remarkable unprofessional head-in-the-sand attitude bordering on solipsism," and in the rush to prepare "sophisticated and advanced programs . . . , fundamental questions remain unasked and unanswered."

As an example of his targets the author cites PERT. He recognizes that the process of preparing a PERT network enforces explicit planning which is beneficial to any activity. On the other hand, while the many processes through which the data must pass eventually lead to condensed data which receives the attention of top management, it is generally months since the input data has been collected, at which time the input data is no longer valid and the real PERT network is changed. To find out what is really going on the manager telephones his subordinates to obtain their current estimates of progress and problems.

What is needed is a revision of attitudes and objectives. The designer must conceive systems in which the data produced will assist all levels of management and supervision to communicate and make available the information really needed for planning, managing, and decision making. Perhaps the author's point of view is best summarized by his statement, "Information and knowledge are not synonymous and a good subordinate is a better aid to understanding than raw data."

"The Standardization of Programming Languages" by Franz L. Alt discusses the pros and cons of standardization and surveys the activities and organization of the various committees and groups which have been charged with the standardization of common programming languages. Agreement to standardize is not universal—"What seems best to the user does not seem best to the producer of computers and what is best for one particular company is contrary to the interests of all other companies." Moreover standardization tends to stifle the development of new and improved languages. If premature the penalty can be heavy.

On the other hand, standardization seems to be desirable to stem the proliferation of languages which has set in during the past few years, to further the exchange of programs between computing laboratories, and to reduce the number of compilers which are required with each new computer.

Finally the author discusses the activities of other organizations in the development of standards, including ACM, IFIP, AFIPS, BEMA, ECMA, and ISO, and he relates them to the ASA work.

Compilers and Programming Languages

Edward L. Manderfield, Chairman
North American Aviation Corporation
Tues., Aug. 25, 4:00-5:30 p.m.

This session is sponsored by the Programming Languages Committee of the Joint Users Group. This is the only session on programming languages and compilers at this ACM convention.

The paper on FORTRAN by Norman Moraff of Westinghouse Baltimore concerns useful extensions of the language to business and scientific problems, contained in a package called "BEEF."

A paper by Rex Franciotti of IBM and Marjorie Leitzke of Oak Ridge National Laboratory summarizes the organization of the SHARE ALGOL Compiler, a large co-operative users project.

Dr. Ned Chapin will discuss the implementation of IPL-V, the oldest and best established of the list-processing languages, on a small computer, the IBM 1620; this contribution from the 1620 users group should be of great interest to the university users in particular, since it extends the use of list processing to the small college computing centers.

Two papers deal with some up-to-the-minute developments in the "Recursive Descent" method of syntax-directed compiler writing. These papers are "Meta-II" by Val Schorre of UCLA and "Meta-III" by Schneider and Johnson, also of the UCLA computing facility.

They are concerned with automation of compiler writing, reducing the writing of compilers to a matter of weeks. These methods require simply the specification of the syntax, semantics, and pragmatics of a translator by means of tranformational equations bearing some resemblance to Bacchus Normal Form equations. This technique, as it flowers, promises to improve greatly on the costly conventional methods which require from many months to several years for compiler construction.

Pattern Recognition

Peter M. Kelley, Chairman
Philco Corporation
Tues., Aug. 25, 4:00-5:30 p.m.

The pattern recognition session at the 19th Annual ACM Conference presents papers which, in this relatively new field, cover the spectrum from basic theory to actual applications.

R. V. Smith of IBM discusses the possibility of applying measure concepts as developed for information theory to recognition problems.

Vossler and Branston of Cornell Aeronautical Laboratory discuss the related problem of using context as an aid in character recognition and applied their concepts in a computer program which used textual material from books and newspapers.

Prather and Uhr of the System Development Corporation have created a learning program for recognition purposes. This program learns, forgets, uses small flexible subtemplates, and a flexible spatial relationship between different portions of the pattern all to achieve recognition.

The work which is characterized by emphasis upon actual applications is that reported on in two papers—one from Douglas Aircraft and the other from the Philco Corporation. The

Douglas Aircraft paper written by Joseph, Viglione, and Wolf describes techniques which have been developed for interpreting cloud cover photographs transmitted from meteorological satellites. The complexity of the machine required to do the task is described. The results are based upon experimental data gained from over 10,000 measurements.

The Philco paper by Kanal and Randall describes techniques for designing machines for the sorting of aerial photographs. The techniques are based on well accepted statistical concepts. The nature of the data does not allow for the direct application of classical methods of multivariate discriminant analysis; rather, modifications of classical methods are used. Excellent results were obtained on a design simulation which was carried out on a digital computer with the aid of a special input-output device which converts imagery to computer language.

Numerical Analysis—Related Topics

David M. Young, Jr., Chairman
The University of Texas
Wed., Aug. 26, 9:00-10:30 a.m.

There are four papers scheduled for presentation at the Numerical Analysis Session:

The first paper contains a comparison of "three iteration methods for computing approximately a steady-state solution of the diffusion equation, in a pie-shaped region of the plane, using a finite mesh based on the $r-\theta$ coordinate system. The methods are successive line overrelaxation along curves of constant radius, successive line relaxation along radii and alternating-direction-implicit iteration. It is concluded that the first-named of these three methods is superior, in that it is expected to converge faster for most problems."

The second paper contains a discussion of the generalized inverse A^+ of an arbitrary m by n complex matrix A of rank $r \leq m \leq n$, and a presentation of a procedure for computing A^+ which consists of a variant of the gradient projection method.

The third paper is a survey of several procedures available for nonlinear optimization. It includes several examples which show how the procedures may be applied.

The fourth paper contains a discussion of an iterative matrix method for determining the size of modular numbers which includes some previous methods as special cases.

Operating Systems

George H. Mealy, Chairman
IBM Corporation
Wed., Aug. 26, 9:00-10:30 a.m.

Operating systems have been known, in one form or another, at least since the 1953 MIT Summer Session, at which point C. W. Adams and his associates had developed a utility system for the WHIRLWIND computer. The terms "automatic programming" and "automatic operator programs" were both coined at about that time—the twin problems of saving programmer effort and machine time were recognized almost simultaneously. Strangely enough, although the machines themselves were the scarce commodity in the early days it was automatic programming that received the widest acclaim and attention, despite fears by many that programs "written in English" or any other narrative language would waste machine time because a good coder could out-optimize any compiler. Even though the number of homegrown operating systems almost outnumbered the number of compilers about five years later, the literature is still mostly silent on the subject of operating systems. Possibly the original coined terms were to blame—it is clearly more challenging to replace a programmer than to replace an operator!

We are, of course, in no danger of replacing either resource; good operators, like good programmers, will probably always be worth their weight in gold. We are finally realizing, in fact, that polarization of the universe into programming systems vs. operating systems is as faulty cosmology as drawing sharp lines between commercial and business data processing or between "real-time" systems and all others. Similarly, we are beginning to see that the problems presented by teleprocessing, remote computing, multiprogramming, batched job processing, data collection and inquiry processing, multiprocessing, etc. are basically common, rather than distinct, problems. We are even beginning to think in terms of accommodating many, if not all, of these processing modes within the structure of a single operating system.

The proper point of view in considering an operating system is not that it merely replaces or assists the machine operator. Rather, it forms part of the environment within which a program operates, just as the machine itself and the installation's operating staff does. Perhaps the most significant insight in this respect was gained by Holt and Turanski some five years ago—a program really operates on an Extended Machine, which is the physical machine itself together with a programmatic extension (the operating system) which provides services such as interrupt handling, storage management, and a variety of translation and control functions. The programmer sees not just the bare machine itself but also (and very often only) the extension of the machine and its operating environment provided by the operating system.

This is the point of view developed in the final paper of the session. The philosophical tenets described in that paper have been fundamental in the design of several recent operating systems. Another paper describes the organization of a large multi-computer operating system for a command and control application. The remaining paper is a case study of a large scientific installation which, on the basis of measurements and simulations reported in the paper, recently abandoned its prior main computer and separate peripheral computer configurations in favor of a simple type of multiprocessor configuration. The papers represent a cross-section of our current concerns, both mundane and philosophical, in the area of operating systems.

The purpose of the session, however, is not so much to present three papers as it is to examine, with the aid of six expert witnesses, what operating systems are *really* about and where we are headed.

Programmer Training

Gloria M. Silvern, Chairman
North American Aviation, Inc.
Thurs., Aug. 27, 9:00-10:30 a.m.

The Programmer Training session, sponsored by the ACM Special Interest Committee on Digital Computer Programmer Training, promises to be extremely interesting as well as informative and valuable to all those concerned with instructing programmers.

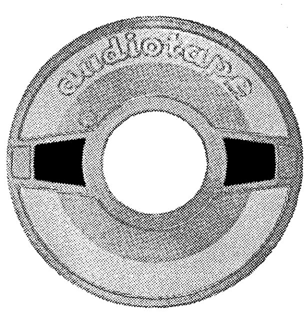
One of the newest media used for teaching programming is described and evaluated in "FORTRAN IV on Videotape—An Experiment in Televised Programmer Training," by Walter T. Mara, University of California at Davis. A televised FORTRAN course was recorded on videotape during the summer of 1963 and has been used since for all FORTRAN instruction taking place on campus. The television medium seems to be effective and well-suited for training programmers. In addition to describing course philosophy and content, videotaping production techniques and methods of conducting televised classes a kinescope of selected portions of the videotapes will be shown and discussed during this presentation. Trainers in business, industry, government and university computing centers, as well as educators in schools and universities, will not want to miss this opportunity to learn first-hand about this exciting new medium of closed-circuit television.

Another method of instruction which has recently achieved great popularity is programmed learning, which is based on presenting information to the learner in a carefully-constructed sequence of small incremental bits known as "steps." A similar way to apply the principles of learning in small pieces or steps is to construct and use a problem set. This is a sequence of problems whose solution on the computer helps the learner to gain progressive capability in computer language, programming fundamentals and programming practice. While not attempting to provide a recipe for construction of such a problem set, "A Discussion of a Problem Set for Instruction in Digital

(Con't. on page 51)

"In the mad, mad, mad, mad world of movies, Computer Audiotape plays an important role,"

says Mr. John Fitzgerald, Data Processing Manager for United Artists Corporation **UA**

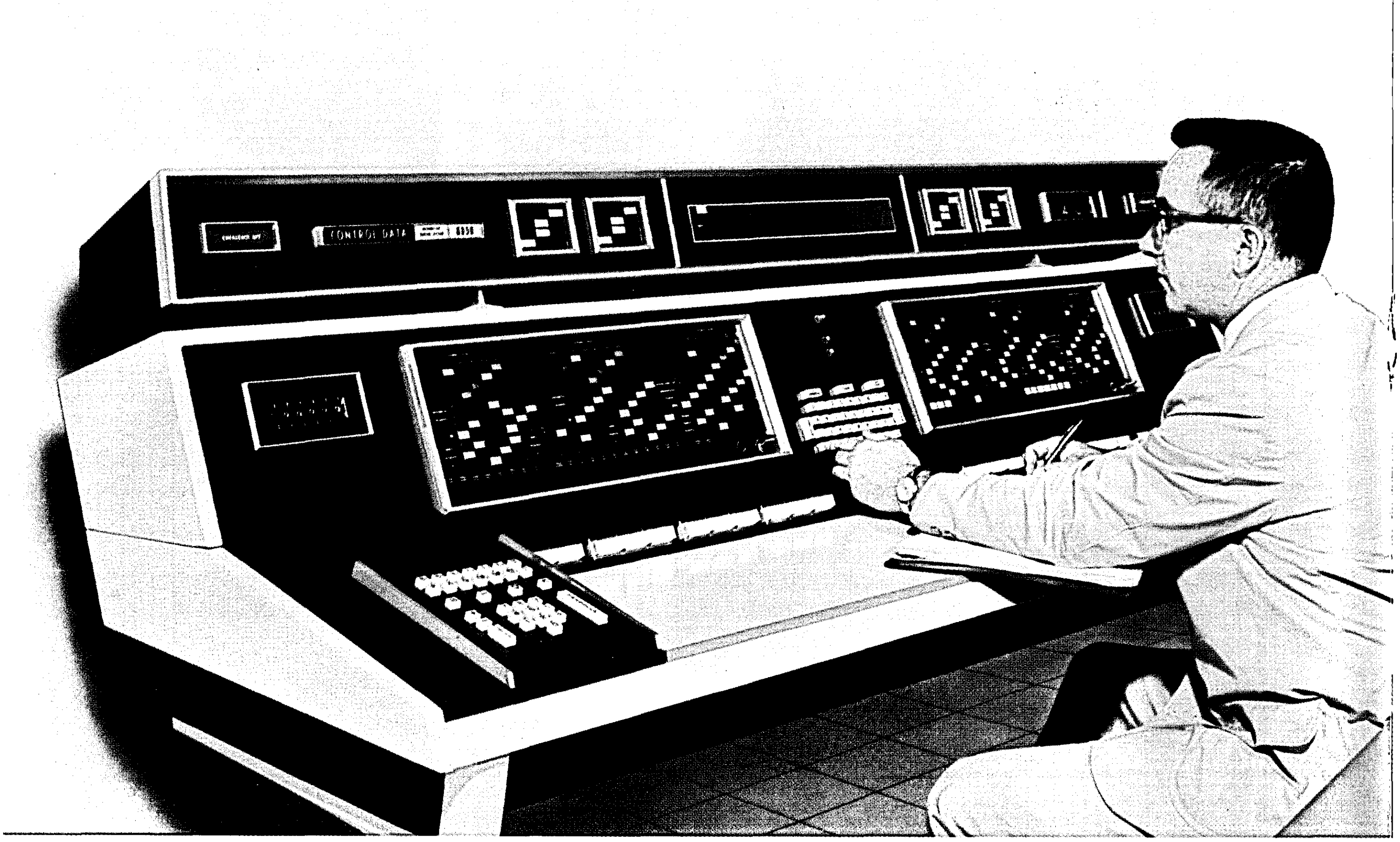


"In the colorful motion picture business even accounting is unique. For example, here at United Artists we use an IBM 1401 Computer for the sole purpose of processing producers' settlement statements. United Artists circulates as many as 1,000 films throughout the world at any given time. Our computer prepares detailed financial statements for each of these films. To do this job, we use Computer Audiotape. We first tried it two years ago, and it worked out so well we've often recommended it to other companies. As a matter of fact, we now use it exclusively."

United Artists Corporation is another prominent firm that consistently specifies Computer Audiotape. You can "test run" Computer Audiotape on your computer. "Test run" it on your own equipment, at no cost and at your convenience. For complete details write to Audio Devices, Dept. DT.

AUDIO DEVICES, INC., 235 East 42 Street, New York, N. Y. 10017

CIRCLE 21 ON READER CARD



Two computers are operated from this Control Data Corporation Console which uses six DATA•PANEL Information Displays.

NEW

TEC LITE

D A T A • P A N E L

SOLVES FIVE INFORMATION DISPLAY PROBLEMS AT LOW COST!

Designers of computer consoles, display panels and status boards now have a strikingly beautiful, yet truly practical way to answer problems of visual impact, operator accuracy, appearance and display versatility. DATA•PANEL offers a totally new approach to custom designing and building information displays at a cost usually far below that for standard panels using conventional individual indicators.

TEC-LITE DATA•PANEL gives the designer almost unlimited freedom in size, style, color and arrangement of messages and indications within the display area. *Only when illuminated* are alpha-numeric messages and symbols visible in color behind planes of glare-free black glass. DATA•PANEL eliminates rows of ever-present individual indicators which restrict design freedom.

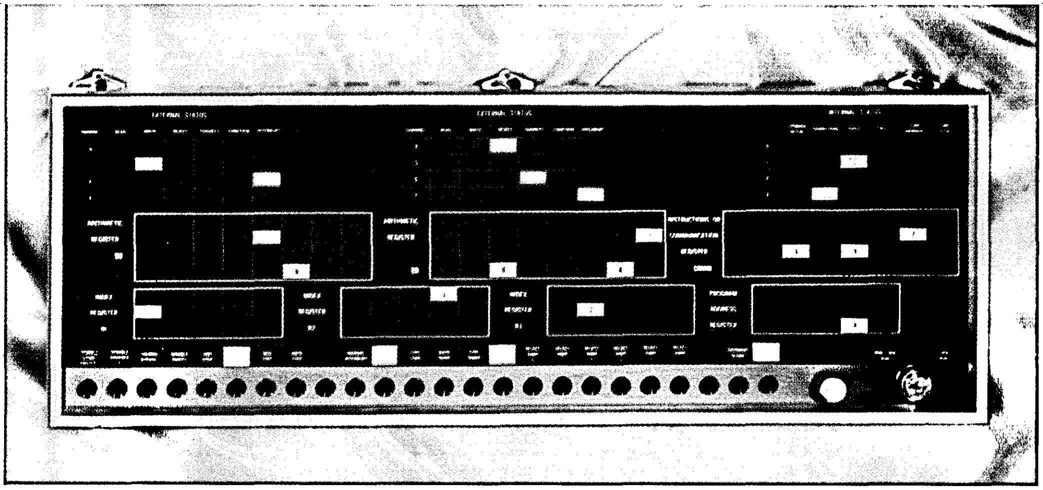
Operator accuracy increases sharply with the use of DATA•PANEL because only the important ON indications can be seen and read. OFF indications are invisible until illuminated and the operator is not confused by the presence of non-indicating indicators.

TEC-LITE DATA•PANEL is also electrically and mechanically versatile and is custom designed to fit the particular characteristics of any computing or control system. DATA•PANEL is designed and built by Transistor Electronics Corporation, originator and world's largest manufacturer of transistor controlled indicating devices. Send for 8-page brochure with design specification forms for more information on DATA•PANEL.

1 APPEARANCE—An entirely new look is given computer consoles and display panels by the distinctive and dramatic appearance of DATA•PANEL. Information is displayed with clarity, brilliance and readability never before available to designers. Indications, readouts, complete legends stand out emphatically, in full color behind smooth planes of glare-free black glass. As modern in concept as the systems it serves, DATA•PANEL is the key to clean, handsome console styling.

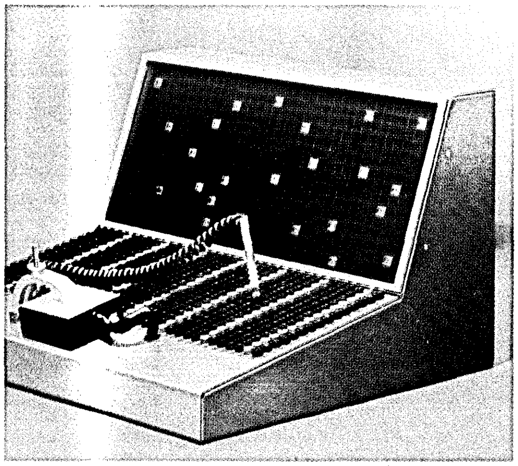
Status boards can be made in a variety of sizes using the DATA•PANEL Concept.

2 DISPLAY VERSATILITY—There are no restrictions, within practical limits, to the overall size of DATA•PANEL, or to the shape, color, size or arrangement of alpha-numeric messages, indications or digital readouts. Legends (which are photographically reproduced) may be as long or as large as required and unlike methods used with conventional indicator lights, are not limited by size of lens cap or cost of panel engraving. Each legend or symbol appears within a DATA•Module, which may be of many sizes or shapes.



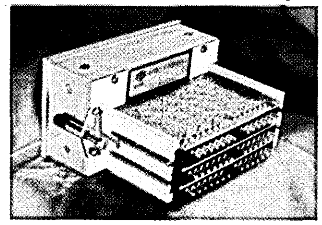
DATA•PANEL indications are visible only when illuminated. Permanently visible legends and grids are provided for operator orientation. Courtesy Control Data Corporation.

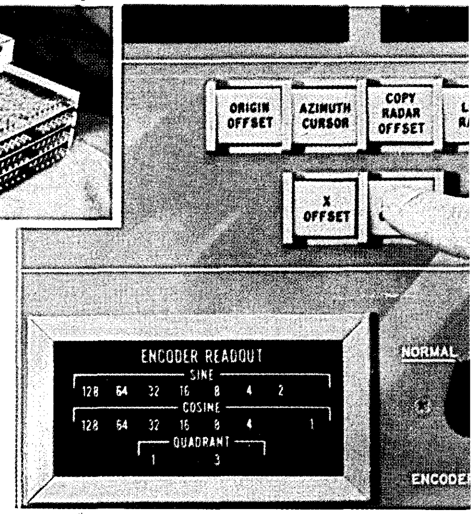
[3] OPERATOR ACCURACY—
DATA•PANEL legends and indications in the OFF condition are totally invisible until illuminated. Operators, therefore, are not distracted or confused by the presence of ambiguous, non-indicating indicators, or, for example, by red lensed indicators that merely turn redder. Permanently visible grid lines and legends can be provided for visual orientation. Operators working with a visually clean panel suffer less fatigue and, consequently, make fewer errors.



DATA•PANEL, with 540 lamp display is used in Module Test Device at Litton Data Systems Division.

DATA•PANEL, 2½" high x 4½" wide, was custom designed for Litton Data Systems Division.

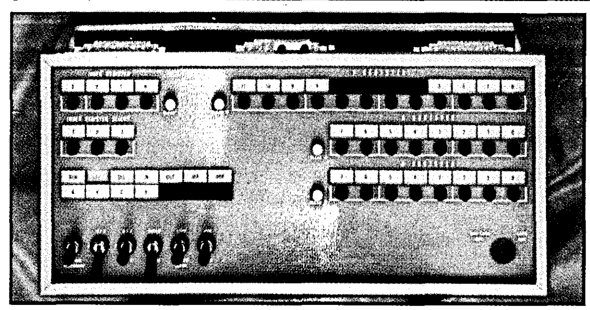




[4] CUSTOM DESIGN—
Information display areas become an integral, complementary element of console styling when DATA•PANEL is used. Its extremely flexible visual and mechanical parameters give designers freedom never before available in display techniques. Built to designers' specifications, DATA•PANEL is a complete, self-contained display unit ready to mount. If you prefer, TEC-LITE DATA•PANEL industrial designers will provide console styling and display layout based on your requirements.

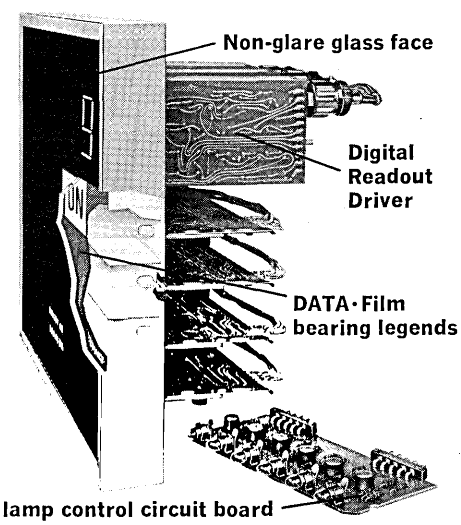
[5] CUTS DISPLAY COSTS—Even though built to your design and offering unequalled appearance, DATA•PANEL usually costs less than conventional display panels using individually mounted indicators. DATA•PANEL costs less because it eliminates these costs: metal panel fabrication and finishing; hot stamping or engraving legends; mounting and wiring of many individual indicators. When transistor controlled lamps are specified, DATA•PANEL costs less because many lamp control circuits are placed on a single printed circuit board, common connections are used and simple lamp mounting devices employed. Eliminated are many expensive parts such as lenses, lamp sockets, nuts, bodies and molded terminal assemblies required for each individual-type indicator. DATA•PANEL production costs are reduced by standard tooling, hardware and component mounting techniques.

CUSTOM SWITCHES—
TEC-LITE DATA•Switches extend console design freedom to important control functions, too. They offer a variety of electrical, mechanical and visual options. Buttons may be produced in configurations appropriate to panel design. Conventional or transistor controlled indicators can be incorporated with, but isolated from, the switch to combine indication and control functions for space conservation and operator efficiency.



DATA•PANEL and switches are combined in this display-control assembly built for Control Data Corporation.

DATA•PANEL SOLVES ELECTRICAL-MECHANICAL PROBLEMS
DATA•PANEL is extremely flexible both electrically and mechanically. Replaceable incandescent or neon lamps are used and may be selected to operate from a wide range of supply voltages or controlled by solid state circuitry which is a part of the assembly. High current drain problems encountered with incandescent lamps are solved by transistorized circuitry that switches lamps ON and OFF with low current level signals usually found in solid state systems. High voltage problems inherent in neon lamps are confined to DATA•PANEL by use of self contained circuitry that operates from low level logic signals. Neon display tube, segmented, projection and other alpha-numeric readout devices can be mounted behind the glass panel as an integral part of the assembly. Mechanically, DATA•PANEL and its DATA•Modules may be of any practical size. The self-contained display panel can be mounted flush with adjacent surfaces or recessed below and at angles to surrounding surfaces. Rack mounting is also available.

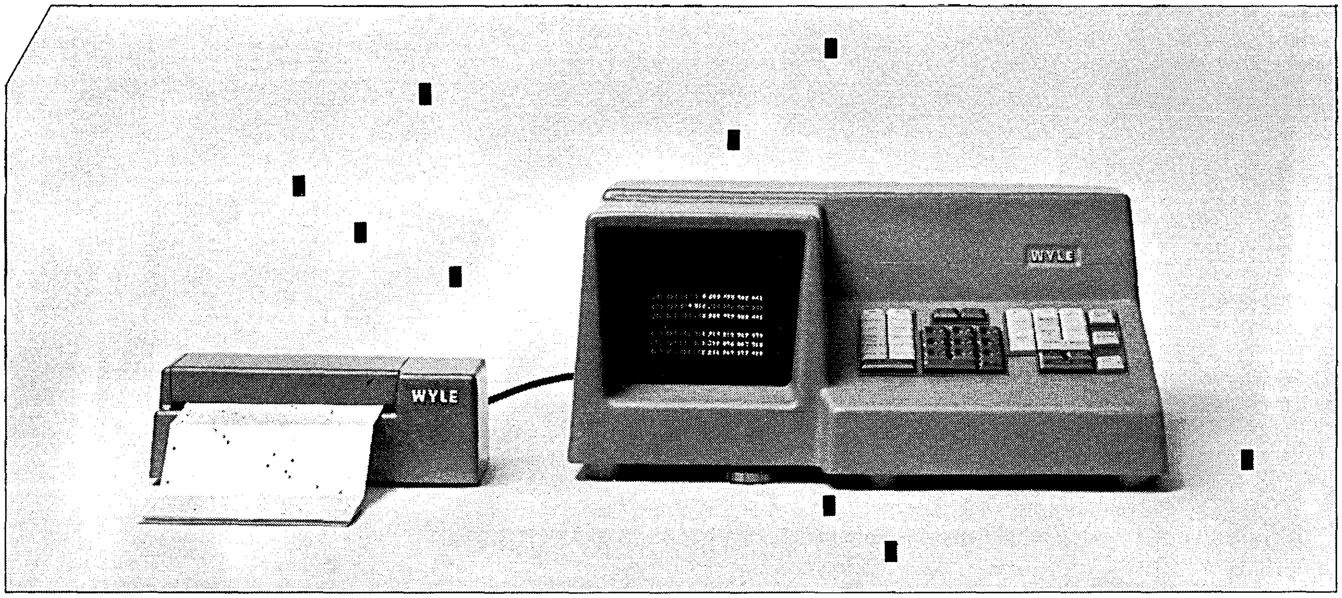


TEC LITE INDICATING DEVICES

Transistor Electronics Corporation

Dept I, Box 6191 • Minneapolis, Minnesota 55424 • Phone (612)-941-1100

INDICATE 20 ON READER CARD

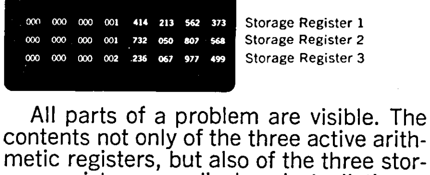


More than a calculator-Almost a computer ...and now with automatic input-$4350

THE WYLE *SCIENTIFIC* IS THE FIRST DESK-TOP COMPUTATIONAL CENTER designed specifically for the solution of complex scientific and engineering problems. Almost as simple to operate as an adding machine, it provides arithmetic capability, efficiency and speed never before available in a calculator.

WITH AUTOMATED DATA ENTRY, it eliminates all the tedious, wasted time of multi-step repetitive arithmetic problem solving. You enter only the variables manually. All the repetitive procedures are run off automatically from a prepared program library of simple punchcards fed into the calculator reader. And you don't have to be a programmer or need additional equipment to prepare your own input library. You simply punch in your instructions by hand on a Wyle stored-program card, which has the calculator keyboard reproduced on it. With this automated input added to the Wyle *Scientific* versatility, you will solve complex problems at speed approaching that of a computer.

NEVER BEFORE CAPABILITIES LIKE THESE
The contents of all registers are displayed, on an eight-inch cathode ray tube, as indicated in the following diagram.



All parts of a problem are visible. The contents not only of the three active arithmetic registers, but also of the three storage registers are displayed at all times. Numbers entered from the keyboard are seen as they are entered and can be verified before use.

Transcription errors are eliminated through complete versatility of transfer from any register to any other without loss of desired data.

All registers handle 24-digit numbers.

Decimal points are entered the same as digits, using an eleventh key, and all input and answers are correctly aligned with decimal point on the output display.

Automatic square root is provided, as is single entry squaring and multiple subtotals.

The calculator has plug-in compatibility with auxiliary input-output devices including printers, paper tape equipment, and other EDP equipment.

Its operation can be learned in minutes, and it functions with the speed, quiet, and reliability of its solid state design.

These capabilities, combined with automatic entry, for the first time fill the technical and economic gap between calculators and computers.

$3950 for basic calculator
(You can add automatic input later)
$4350 complete with automatic input

For further information, write Dept. J, Products Division, Wyle Laboratories, El Segundo, California. Or telephone (213) ORegon 8-4251.

WYLE LABORATORIES

CIRCLE 22 ON READER CARD

Computer Programming," by Dr. Elliott I. Organick, University of Houston, analyzes a set of eight problems used successfully for several years during intensive eight-week seminar for engineering professors. Most programming instructors would agree that the sequence and contents of the problems used in a programming course are critical factors in the success of the course.

The recent emphasis on computing and information processing has caused a number of schools and universities to introduce courses on computer-based mathematics and programming into their curriculums. Summer courses in universities also help train high school and college teachers as well as students. One of these programs, conducted at the University of Pennsylvania, is described and discussed in "Computer and Information Sciences Program for High School Students."* A pilot program was held in 1962, followed by a full scale program in the summer of 1963 and a more extensive program in 1964 in which the participants include high school teachers as well as students.

Following each paper will be a question and discussion period. A panel of selected experts will lead off the discussions, after which the audience will be encouraged to participate.**

Formal Manipulation of Mathematical Expressions

T. E. Cheatham, Jr., Co-chairman
Computer Associates, Inc.
Frank Engel, Jr., Co-chairman
Harvard University
Thurs., Aug. 27, 9:00-10:30 a.m.

This session will offer a single paper on an experimental compiler and language developed expressly for the purpose of formal manipulation of mathematical formulae. Whereas FORTRAN and other algebraic languages facilitate the programming of the numeric calculations of algebraic expressions, FORMAC deals with the variables, coefficients, and symbols themselves rather than their numeric values.

From the early Univac days of Karamanian's program for analytic differentiation, through Newell and Simon's general problem solver and Wang's efforts toward mechanical mathematics, there have been numerous attempts to extend the mathematical capabilities of machines. In a recent survey, Miss Sammet reported that more published work in this area has appeared in the last year than in the preceding 15 years. Polynomial algebra and formal integration in addition to differentiation have become popular. FORMAC as an extension of the FORTRAN system provides the opportunity for further experimentation in mechanization of more general mathematical analysis, and provides insights into needs for future developments.

Leading the discussion of probable impact and evaluation of this paper will be a panel including Professor A. J. Perlis, who has been closely associated with algebraic compiler development through IT, GAT, ALGOL, etc.; Dr. Donald L. Shell who developed the CAGE operating system for the 704, and headed up the SHARE SOS system development; and the chairmen representing wide and varied user experience.

Panel: Computer Science Curriculums

William F. Atchison, Chairman
Georgia Institute of Technology
Thurs., Aug. 27, 11:00 a.m.-12:30 p.m.

During the past several years, the use of computers and the study of information processing has spread rapidly in each of the areas—management science, the behavioral sciences, and the biological sciences. Although these are very diverse fields, the computer specialist may view them as a relatively homogeneous area in which the dominant computational issues are those in which statistical and file processing techniques are of prime importance, and in which complex system models are more relevant than simple mathematical structures.

In broader perspective, the use of mathematical models, statistical analysis, and digital computation are growing rapidly in these fields. A new educational synthesis of mathematics, statistics, and information processing may emerge as the proper preparation for professional work and research.

The members of the panel will describe trends in digital computation and educational problems in their respective fields. They will indicate possible educational approaches and the prospective roll of computer science as it emerges as a distinct educational area, alongside mathematics and statistics.

Information Retrieval

Benjamin F. Cheydleur, Chairman
Philco Corporation
Thurs., Aug. 27, 11:00 a.m.-12:30 p.m.

The first paper by Charles D. Parsons of Phillips Petroleum Company describes the concepts and some programming features of a going industrial information retrieval system, called SIR. Noteworthy as a system that is operational via a large scale computer while possessing an architecture that is simple, "SIR" bears the imprint of the practical thinking of H. P. Luhn, past president of the American Documentation Institute, and pioneer of many information retrieval advances in the direction of automation (for example, the famous KWIC or Key-Word-In-Context System).

Moving from the foregoing system, in which attention is centered on the servicing of queries which are formatted as a concatenation of descriptors, the session next features a concern for a system in which automatic description of information is controlled by profiles of interest. In this second paper, by A. B. Barnes, *et al*, the famous SDI system, as revised to incorporate a more timely feedback between the user and the system, is described under the title "SDI-5, An Advanced System for Selective Dissemination of Information." Here attention is partly turned to such bread and butter operations as the preparation of notices by computer. Modular programming, including such features as vocabulary control, provides a base for specialized and improved service in this already impressive operation. The system provides service to scientific and administrative personnel alike in the IBM Advanced Systems Development Division.

The final paper by Professor Gerard Salton reports practical progress all along the line in harnessing together a battery of advanced techniques, for the formation analysis, in the Harvard Computation Laboratory. In "A Document Retrieval System for Man-Machine Interaction," the concern with the key problem of on-line reactivity is supported by marshalling a number of proven program techniques for associative sensitivity in the manipulation of information files, including alphabetic dictionary operation, concept hierarchy control, text segmentation with concept associations derived from co-occurrences within sentences, syntactic processing, document correlation. This total system, called SMART, when fully mobilized, should constitute an effective base for advanced automated information retrieval research. The system is undoubtedly most interesting as providing a variety of non-mathematical "computing" processes which can be brought to serve man's more sophisticated needs.

EDP Standardization 1964

John A. Gosden, Chairman
Auerbach Corporation
Thurs., Aug. 27, 2:00-3:30 p.m.

As the EDP industry has grown at an amazing pace it has developed many diverse practices and components, some more successful than others. This has led to considerable problems in the exchange of ideas, equipment and programs among users. As a result there have developed many *de facto* standards ranging from collaboration on software in users groups such as SHARE to compatible tape formats among many manufacturers without formal collaboration.

The development of official standards is a slower process which started in the late 1950's and is now beginning to generate results.

The effective establishment and general adherence to standards will only come about if the standards are good ones and acceptable to the community affected. The members of the ACM are a significant part of that community and it is to their interest to contribute to the quality of any standards that affect them.

In this session the panel will briefly review the process of standardization, the current status of standards being pro-

* by D. Ashler, A. van Dam and D. Prener, University of Pennsylvania.
** The panelists will be Marvin M. Wofsey, The American University, SICODCPT Secretary; Dr. Ned Chapin, Consultant; and Dr. Gloria M. Silvern, session and SICODCPT Chairman.

duced and how the ACM is involved. This will be an introduction to a general discussion in which the many facets of standardization will be discussed.

Computers and Communications

A. E. Miller, Chairman
Auerbach Corporation
Thurs., Aug. 27, 4:00-5:30 p.m.

Although the theoretical foundations of the communications field and the computer field have much in common, the vast majority of practitioners in each of these fields know little of the body of knowledge or practices developed by the other. Within the past few years, the development and implementation of communications systems using computer-type devices have started to break down this independence of the two fields. The session on "Computers and Communications" has been developed to aid in this exchange of information by presenting to the computing community, in a coherent fashion, a summary of the use of computers in relation to communications systems.

The first paper discusses the use of computers in planning and managing a communications network as well as the use of computer-type devices as the nodal points of the network itself. Depending on the communications network, the nodes of the network can perform a "circuit-switching" function, a "message-switching" function, or a combination of both. The second paper presents the fundamentals of the use of stored program devices for circuit switching, while the final paper describes the fundamentals of the use of electronic computers for message switching.

Administration of University Computing Centers

S. D. Conte, Chairman
Purdue University
Thurs., Aug. 27, 4:00-5:30 p.m.

Computing has now reached a level of development at universities where the Computing Center is as essential as the library. The importance of the computer as a research tool has for some years been well established. Its use for instructional purposes is today becoming equally well accepted. There is growing acceptance of the philosophy that every student of science and engineering, and many students in the behavioral, social and management sciences must be aware of the role that computers can play in research, and must have first hand experience in preparing and programming problems for computers. Acceptance of this philosophy implies that universities must provide facilities capable of handling large numbers of students—perhaps several thousand each semester. The processing of hundreds of jobs each day creates severe organizational problems. How does the center physically handle the problem of getting cards punched, submitting decks, processing jobs and returning output? How are student problems to be checked for errors and graded? How does one find enough computer time to run several hundred jobs a day with the slow compilers currently available on most computer systems?

As the volume of work increases the Center will inevitably be faced with a shortage of computer time, at which time it will be necessary to impose restrictions on computer usage. A priority system will have to be initiated, computer time per job will have to be limited, pages of output will have to be restricted. What policies should the university adopt with respect to priorities, acceptance of jobs, and rate-charging?

The questions raised above will be discussed at this panel session. The four speakers at this session represent schools which have considerable experience in the use of computers for instructional purposes John Hamblen from the University of Southern Illinois will discuss and make recommendations on university policy concerning priorities and rates. Bernard Galler will describe submission and processing procedures currently in use at the University of Michigan. George Forsythe from Stanford University will discuss the role of automatic grading programs. Saul Rosen from Purdue University will discuss the place of specially designed compilers to allow efficient processing of typical student jobs, and he will describe some compilers of this type currently in use or being developed.

Invited Papers II

M. V. Wilkes, Chairman
The University Mathematical Laboratory
Cambridge, England
Wed., Aug. 26, 11:00 a.m.-12:30 p.m.

Prof. Wilkes has been a leader in the advancement of digital computing techniques since he participated in the development of EDSAC, the first-successfully-operated true stored-program computer. One of the earliest comprehensive programming texts, of which he is a co-author, is still used widely in the U. S. as a reference work. He has contributed recently to the development of the techniques of list processing and symbol manipulation.

Because of his remarkable lucidity, as well as his intimate knowledge of the current status of work in this area, Prof. Wilkes was invited to address a plenary session of ACM'64. He will present a combined tutorial-and-survey paper on list processing and its application.

Computers have long been in general use for solving numerical problems and pioneering interest has now switched to their use for nonnumerical work, that is, for manipulating symbols. Examples are compiling, studies in artificial intelligence, layout problems, etc. List-processing was a breakthrough in symbol manipulation since it provided a flexible way of organizing the computer memory. The paper explains in a tutorial manner what goes on in the computer memory when list-processing operations are performed, and takes as an example the formal differentiation of an algebraic expression written in Polish notation.

Unlike the procedure for most sessions at this meeting, it is planned that live questions will be accepted from the floor after the prepared lecture.

Demonstration of MAC Time-Sharing System

F. J. Corbato, Chairman
MIT
Wed., Aug. 26, 8:00 p.m.

Multiple console, shared digital computers represent a new concept in computer design, in which emphasis is placed on ease of user communication with the machine, and rapid machine response to commands. A complete description of machine response characteristics must include the statistical behavior brought about by queues in the system. The numerical solution of a complex queueing model is one way of predicting the statistical response characteristics, and models incorporating a good deal of realism can be rapidly solved. To illustrate this viewpoint in a simple way, a queueing model has been developed for a multi-console computer with a multiple shared memory. The relation of the model to a physical system is described in detail, and the numerical solutions discussed.

The panel will be informal with discussion ranging from the philosophy and appropriateness of general purpose multiple-console systems to the implementation problems which arise with contemporary equipment. The panel members have been selected on the basis of actively participating in the development of general purpose multiple console systems. Each speaker will be asked to state his views briefly. Comments on these views will be heard from the discussants followed by questions from the audience if time remains.

In the informal evening session, there will be live demonstrations of several general purpose time-sharing console systems using computers located in different parts of the country. Each demonstration on an impromptu basis will be allowed to display the use of his console for a portion of the session. Audience participation will be encouraged. Beyond demonstrating technical feasibility and current performance of time-sharing systems, discussions will be directed toward the different features which can be implemented in such time-sharing systems.

TESTING REAL-TIME SYSTEMS

Part 2: levels of testing

by ROBERT V. HEAD

Last month, the author covered the development and management of testing procedures for real-time systems. As he noted, " 'System testing' here means whatever work must be performed by the system development staff from the completion of programming until the system achieves operational status. This is not limited to checkout of the various real-time programs; it extends also to the necessary checkout of the equipment subsystems and to the integrated testing of combinations of equipment and programs."

levels of real-time testing

The system development staff has planning responsibility for the identification and definition of the levels of testing to which the real-time system must be subjected before it can be stated with confidence that an acceptable standard of error-free performance can be attained. The structuring of the system test will, of course, vary from one system to another depending upon such factors as size of the programming effort, number of equipment subsystems to be interconnected, and degree of reliability expected from the operational system. It should be useful, however, to describe generally the testing that a real-time system might go through, with emphasis on the application programs. The emphasis is placed here because these programs constitute the major portions of the system which must be subjected to testing. The figure illustrates levels of application program testing.

1 simulated environment

At the beginning of testing, it is neither necessary nor desirable to undertake application program debugging with the real-time equipment or even with its software extension, the control program. And as a practical matter, it is often the case that the equipment has not been manufactured, let alone delivered, and the control program not yet tested at the time that the first application programs become available for test.

During this initial stage, input to the programs in the form of inquiries is simulated, i.e., obtained from punched cards or magnetic tape rather than the remote terminals. Similarly, requests for file records are simulated by pulling data in from tape rather than random access storage.

Often then, initial testing is performed using a special simulator program which duplicates, more or less, the functions of both the hardware and the control program. In this testing mode, which is usually run on some non-real-time computer of the same family as the one to be installed, the execution of a control program macro instruction by an application program causes the control program simulator to produce substantially the same result as would the control program and to return control to the application program just as though the system were functioning under real-time conditions. A "Read File Record" macro, for instance, might result in the needed data being read in from tape rather than from a disc file by the control program simulator so that it looks as if it came in from a certain location on the disc file.

In addition to serving as a substitute for the control program and the real-time hardware before these are available, the simulator provides a special debugging facility not found in the control program itself. For example, a macro trace feature, upon completion of an application program test shot, produces a record of all the control program macro instructions executed by the application program in the course of the test. Many useful debugging features can be built into the control program simulator, since there is no urgent necessity of completing the execution of an application program with the same rapidity with which it must be executed in a true real-time environment. The simulator may, in order to provide extensive debugging capability, not return control to an application program for several seconds after execution of a macro, whereas the control program must in an operating environment return control within milliseconds. The extra time may be used by the simulator for tracing, dumping registers and memory locations, etc.

Testing in this simulated environment usually divides itself into at least two distinguishable levels through which each program must pass. One ground rule in establishing these and other testing levels is that *if a program chronically fails at a given level, it should be "demoted" to the previous level for additional testing.*

1.1 simulated unit testing

Here each individual program or program segment is tested as an individual unit. Frequently unit testing is considered part of the programming phase, with the individual who wrote the program responsible for unit testing it through use of the control program simulator.

1.2 simulated package testing

Whenever a sufficient number of related programs or program segments has successfully undergone unit testing, they may be combined into a cohesive and logically related entity of larger size, sometimes called a package. In order to display an airline passenger's record, a dozen or more programs or program segments may be needed to process the various types of inquiries, some of these representing exception paths in the Display Record Program (such as "No Match on Name"), others representing the main line ("Edit Record for Display"). All these segments, having been found to work satisfactorily as individual units, must be tested to assure that they work correctly together. Data must be passed across segment boundaries, and segments must call and transfer control to each other. Normally, all the program segments needed for a package are not written by the same programmer and as a consequence suffer from misunderstandings, overlooked relationships, and other logical and clerical incon-

sistencies. It may be also that not all the program segments needed to form a package are covered by one program specification, in which case misunderstandings are to be expected in both the specification and programming phases.

2 testing with the control program

Those program packages that successfully pass the first level of testing—in which not only communication with the external world is simulated but communication with the file subsystem and with the control program as well—are ready to be tested in conjunction with the control program. Testing at this level is distinguished by the addition of the control program to the test environment. This can satisfy the dual purpose of debugging both the application programs and the control program if the latter is not a pretested, proven piece of coding. At this level, the application programs take a long step closer to final operating reality by being exercised with the control program and with the central processing unit.

Testing here exemplifies another important axiom of real-time testing: *take only one major new step at a time, holding all other environmental factors constant.* Thus, the testing does not yet include "live" input generated at the terminals. This will come later, after the initial tie-in with the control program has been successfully accomplished. For the moment, it is a sufficient step to go from the control program simulator to the control program itself while continuing to use simulated input of the same type used at the previous level. Indeed, it may be the very same input as was used previously.

2.1 package retesting

Every package which survived testing with the control program simulator must be subjected to another test to detect errors in the packages not discovered by the simulator. The complexity and duration of this package retesting using the control program depends upon the degree of sophistication built into the control program simulator, i.e., the extent to which it truly reflects the logic and restrictions of the control program and the real-time equipment.

2.2 multiple package testing

Here, in a multiprogrammed system, groups of packages which have negotiated the preceding sublevel are run concurrently for the first time in a multiprogrammed manner. Even though the input source continues to be simulated, the entries are now fed into the system not one at a time, as was appropriate to the testing of a single package at a time, but in groups which represent a heterogeneous mixture of transaction types. Whereas before, when testing an airline package like "Sell," a variety of Sell actions with different flight numbers and dates, differing numbers in party, etc., might have been fed in to be run against the Sell package of programs, now not only Sell actions, but Waitlist, Cancel, Display Passenger Record, and many more can be admitted as their corresponding processing packages are added to the system.

3 subsystem testing

As soon as multiple package testing progresses to the point where all packages have been proven to work within a multiprogrammed environment, the application programs are ready for a full-fledged subsystem test. Depending upon the equipment configuration, this level may also consist of sublevels.

3.1 multiple package testing with multiplexor input

If a communication multiplexor is part of the system equipment, this would be the time at which such a device and its stored program might be integrated into the system testing. Heretofore, the simulated input for test purposes was probably fed directly into the central processing unit from one or more tape drives. This was done in order to isolate program errors as belonging to either the packages under test or to the control program, without introducing the more complex possibilities of the error being lodged in the program of some other computer entirely, namely the multiplexor. The time must come, however, when the system development staff, satisfied with the performance of the control and application programs of the central processing unit, starts to admit data generated by the multiplexor, accepting the possibility of this device as a source of error. This broadening of the equipment base for further testing still does not mean that live input is to be admitted to the system. It simply means that the communication multiplexor becomes the source of input, which is either generated by a special program in the multiplexor or pulled in from tape units attached to the multiplexor.

3.2 high volume testing

Here the external input, though still not live, is admitted under considerably less control than before. In earlier testing, incoming entries, while mixed to contain a variety of action code types, were received in a predetermined and known sequence. This was done for two reasons, because: 1) not all packages were available, some not yet having graduated to the subsystem test level; and 2) it was desirable for purposes of analyzing test results to know exactly the kinds and sequences of actions being processed by the available packages. Now these barriers can be lowered by removal of some of the limitations on the input actions. The multiplexor and central processing unit programs are inundated with actions possessing these characteristics:

1) Any package may be called, as all are assumed to be checked out and all possible entries can be processed.
2) The mix and sequence of entries are unknown, just as in productive operation.
3) The volume of input is not controlled, and so the programs have to compensate for overload conditions.

It is a tenet of system testing that the entire test phase is characterized by levels in which more *programs and equipment components are admitted gradually into the test and the volume of test entries and test file records gradually expands.* At the conclusion of subsystem testing, the equipment and programs should be able to deal with the kind of volume that must be processed during the maximum operating peaks projected over the life of the system. One of the biggest testing problems of the system development staff lies in generating or otherwise obtaining test data in sufficient quantities to satisfy this requirement.

4 system testing

The subsystem testing as just described has to do primarily with the testing of the application programs as a major subsystem. Testing of all the other subsystems should take place concurrently to speed the day when all pieces are put together for the grand finale—a final, integrated test of the entire system. At this level, two separate steps may be taken . . . the first involving data generated at the terminals and the second a completely integrated final test.

4.1 testing with terminal input

The application programs, having reached the final system testing stage, are required to cope with input generated at the terminals (which have, presumably, been linked to the central processing site and debugged from an equipment standpoint before this time). In actuality, the lines of demarcation drawn here between the various testing levels and sublevels are not always so clear cut. One level will blend into another so that testing of two kinds goes on simultaneously. The processing of terminal originated actions may, therefore, have begun quite early in the testing phase. However, one fact that differentiates the earlier from the later stages of system testing is that live terminal input is *not* necessary to perform initial testing and may, if introduced in quantity and for other than experimentation, be useless or even detrimental. During the final testing stages, though, the transition from artificial to live test data must be made, in order to flush out error cases not detected by using simulated input. Thus, while terminals may be used as test data sources almost from the beginning, they *must* be used and used very heavily during the final stages in order to provide entries in a mix, a sequence, and a format and content that closely resemble operating conditions.

4.2 final system testing

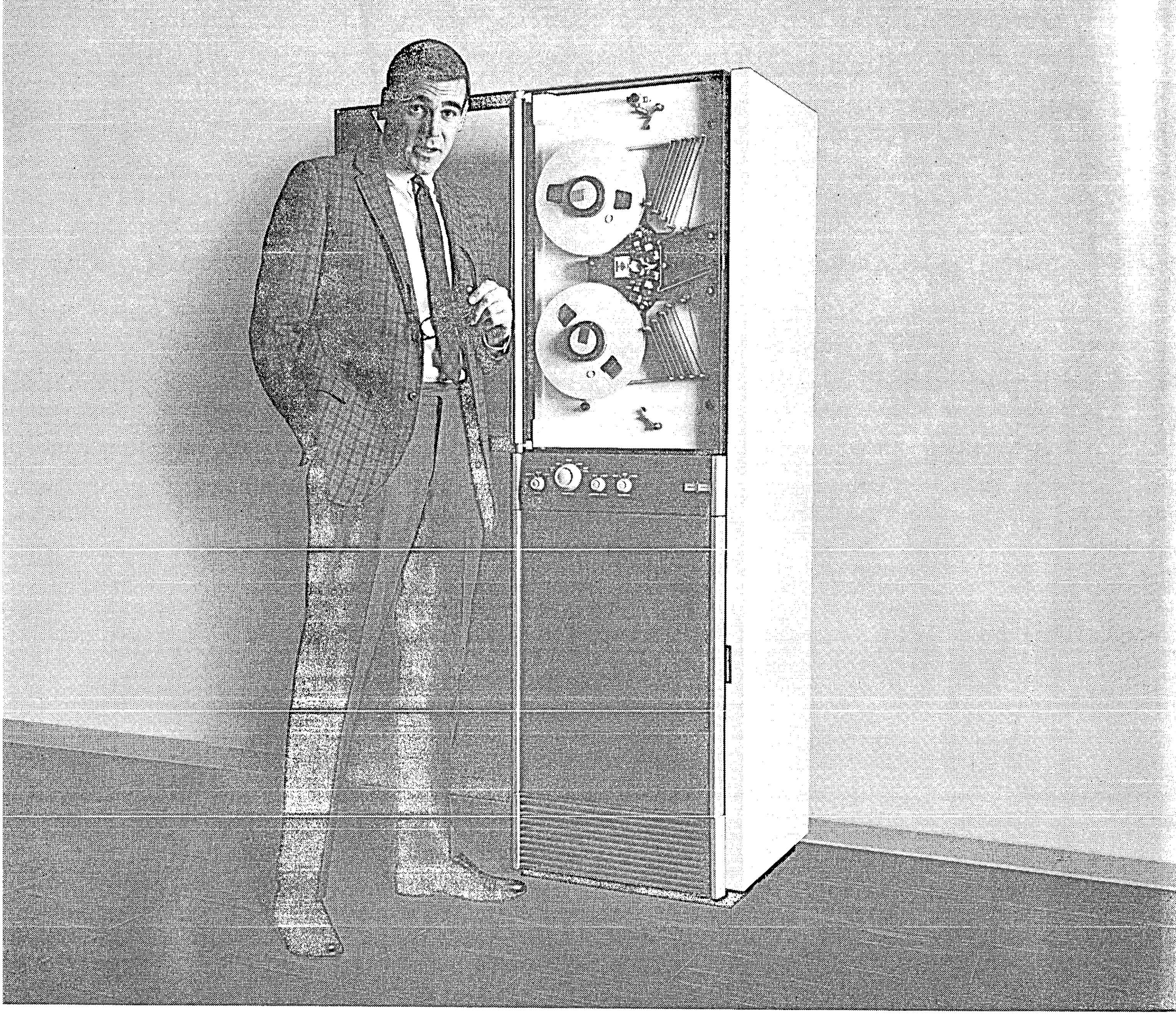
In the more complex systems, such as those of the duplex variety, the real-time portion of the system has to be tied into the off-line portion to assure that the two interact correctly at their points of conjunction. Switchover programs must be fully tested at this phase and assurance gained about the proper functioning of the off-line control program and batch processing programs when they are subjected to interruption by the real-time system. What should emerge at the conclusion of this final system test is a system capable of operating in a sufficiently reliable manner to permit conversion and productive operation.

5 acceptance testing

Heretofore, we have been speaking about the imposition of quality control over error detection and elimination to make certain that the product of the system development staff is operational and that the system will not be hampered by bugs which continually force it into a condition of marginal performance. There is another aspect of system quality control, however. This has to do with evaluating not *whether* the system is capable of performing productively but, assuming that it has been made more or less bug free, how *well* it is performing when measured against the functional or performance requirements. Sometimes an acceptance test is contractually required to provide the user with assurance on this vital question.

The occurrence of program errors may cause the system to fail an acceptance test. But their absence constitutes no guarantee that performance criteria concerning response time and system availability will be met. Even where an acceptance test is not provided for in the contract, there is a tacit condition of acceptance: a successful system must be able to perform the productive work of the application in a manner resembling that originally intended by the system designers, regardless of whether there is a formal acceptance test.

There is a considerable difference between system testing as it has been discussed in this paper and acceptance testing. The successful system test provides an error-free system as the basis for further acceptance criteria to be applied either before or after conversion. ■



THE COMPATIBLES —600...

large-scale hardware from Phoenix

by EDWARD K. YASAKI, Assistant Editor

Broadening its product line and joining what it terms a "trend toward larger computers with communication to remote users," General Electric's Computer Dept. announced late last month its family of two multiprocessing computers—the Compatibles-600. The 625 and 635 bring to eight the number of general-purpose mainframes offered by the company. They range from the small-scale 200's through the medium-sized 400's, and now the large-scale 600's.

Said to be suitable for business, scientific, and real-time computation, the 625 and 635 have two- and one-usec cycle times, respectively. Otherwise, they share hardware features: 36-bit binary words and six-bit BCD characters, 32 to 256K words of core, eight index registers, and 170 instructions. Comparative speeds in micro-seconds are as follows:

	625	635
Access time	2.0	1.0
Fixed-point add	3.0	1.8
multiply	7.0	7.0
divide	14.5	14.2
Floating point add	3.0	2.7
multiply	6.0	5.9
divide	14.5	14.2

Hardware and sotfware specifications were developed simultaneously, and, last September, the complete software system reportedly had been designed before any programs were written. A view of the software design criteria may give some insight into the system concept. Among these criteria: fast turnaround time (via on-line media conversion and multiprocessing), low overhead time (using mass random access storage—at least one drum goes with each system), real-time capabilities, system integration (an operating supervisor controls all programs, and is said to require less than 1% of running-time), and modular software.

This entire software package—operating and I/O supervisors, COBOL '61 Extended, FORTRAN IV, and more—will be delivered with the first machine. Conversion packages will include "GIFT"—Ge Internal FORTRAN (II to IV) Translator—a 9PAC translator to COBOL, and a 7090 simulator.

the time-sharing system

The 600 will also have sequential and threaded list processing capabilities.

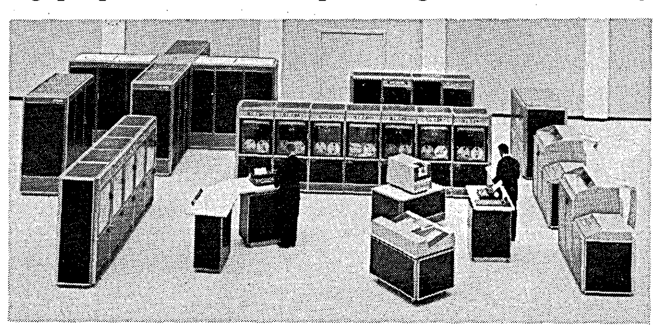
In the area of applications software, a data communication package will go with the first machine. But scheduled for later delivery are APT (for numerical control of machine tools), linear programming, a mathematical library, and PERT/COST.

The 600 is a buss-organized, memory-oriented machine. Every position of core is directly addressable by processors, I/O controllers, and remote users.

Time-sharing of memory by an unlimited number of programs is reported, as is the dynamic relocation of programs: when one program is completed, remaining programs in core are compacted. Then the next program, selected on the basis of a priority scheme and the availability of sufficient core, is brought in from the job stack. (There is, as yet, no program segmentation; whole programs must be handled). An automatic memory protection feature enables programmers to write and execute programs without regard for the memory environment.

Such functions as tape-to-card, tape-to-printer, remote inquiry, and communication transfers are performed concurrently with on-line computation. In addition to having one program compute while another outputs data, two or more programs can process I/O data.

The processor is capable of operating in either of two modes: master or slave. When in the master mode, no restrictions are imposed on a processor program. In the slave mode, a program is prevented from accessing any file-protected memory locations, from directly accessing peripheral devices or special registers, or exercising



control functions of any kind. The operating system runs in the master mode, and user programs in the slave.

the market

Heading the list of prospects for the 600's are those presently using 7090's and 94's (the 635 reportedly has the equivalent internal speed of a 7094 II which, however, lacks time-sharing capabilities). GE is also hunting out those casting flirting eyes at IBM's System/360, mods 50 through 70. Of course, all manufacturers are stalking the 360 game.

Internally, there's also a large market, if only the numerous divisions and departments can be sold. Presently under study is the establishment of computer-centers in Schenectady, Syracuse, Boston, Philadelphia, and Evendale, Ohio, to service the company's plants and offices. GE-Syracuse, alone, has some five computers that might be replaced by a multi-600 system.

First external delivery is slated for April '64, and the firm expresses undisguised glee at the fact that this is about a year earlier than the availability of a 360.

The basic 635 system with 32K words of core, a drum, eight tape drives, two printers, a card reader and punch, is priced at approximately two megabucks, and will rent for some $45K. ■

...AND ITS GENESIS

With first delivery scheduled for the end of this year (to GE's Telecommunications and Information Processing Dept. in Schenectady, N.Y.) and a two-system multiprocessor due to be operating in Phoenix about the same time, the 600 was fast aborning. The go/no go point was reached in April of '63, the former won out, and a design team was formed. Included in the staff were mainframe, peripheral and peripheral control people, software designers, and users and computer management personnel from within the widespread GE organization. (The firm is one of the largest users, if not *the* largest, outside the federal government, and it is estimated that more than half the programs in the SHARE library were contributed by GE. If we may add a historical note, the first commercial application of a computer is said to have been by GE in Louisville, back in 1954. That was a Univac machine).

This shortened gestation period is explained in great part by the fact that the 600's circuitry is borrowed from the firm's military computer, the M-236 (three or four installed, dating back to November '62) and its upgraded version, the 2360, which was never built. Other design features and, principally, the I/O are from a large-scale computer designed and abandoned earlier by the Computer Dept. This design reportedly resembles IBM's System/360.

As for the software, the company makes no bones about the fact that it studied many existing systems, including the "page" concept of ICT-Ferranti's Atlas computer, and selected what it thought were the best features. A 7090 simulator of the 600 was designed and used, but hardware/software checkout is presently being performed on the first two prototype models running on the floor.

Several hundred people were recruited for the design staff. Some have transferred to the Computer Dept. in Phoenix, others served in consulting capacities. Even now, five GE departments are developing software packages under subcontract, an activity headed by James A. Porter, although the operating system and I/O routines are emanating from Phoenix. The latter activity is managed by Russell C. McGee, and the scientific automatic programming by William J. Heffner.

Heading the entire programming effort is Edgar R. Vance, Jr. Before joining the Computer Dept. in 1962, Vance was manager of systems and special applications for the Atomic Power Equipment Dept. (APED) in San Jose, Calif. His responsibilities included systems programming, computer control of nuclear reactors, and N/C programming.

Responsible for both system and hardware design is John F. Couleur, formerly manager of military computer design for the company's Heavy Military Electronics Dept. in Syracuse, N.Y. Under his aegis were the M-236 and Mistram computers.

Balancing this conglomeration of personnel is the project head, Dr. John W. Weil, who joined the department in May of '63 to become manager of the Systems and Processors Operation. As such, he will be responsible for the design, development, and engineering of all dp and data communications equipment. A physicist, Weil had been associated with the Knolls Atomic Power and Vallecitos Atomic Laboratories before being placed in charge of the computer facility at APED in San Jose. ■

Solution by simulation

HOW TO SIMULATE A "WAY-OUT" PROBLEM ON A DESIGNER'S DESK

A simplified method of studying models of airborne systems (whether STOL, SST or anything between) is being used successfully by aerospace engineering groups. Small scale analog computers, compact enough to fit on a desk top, have simulated the various control and propulsion elements of a growing list of complex airborne devices. They have also provided answers to problems in stress and vibration analysis and heat-transfer studies.

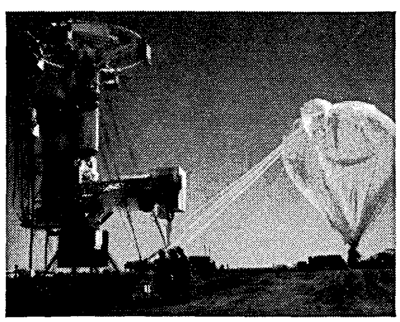
Having "built" a computer model of the system under test, the designer can make it react in a manner similar to the dynamic behavior of the real system in flight. Analog simulation permits dynamic analysis under varying operating conditions—applied simply by changing coefficient potentiometer settings.

Balloons in Aerospace

Even balloon-borne telescopes enter the aerospace realm, as in the design study made at Perkin-Elmer Corporation for Stratoscope II. The system is lifted by balloon to 80,000 feet, a level free of atmospheric turbulence. Stratoscope II provides photographic and spectrometric coverage of planets and galaxies. The star-tracking guidance system is designed to yield 0.02 arc—second tracking accuracy.

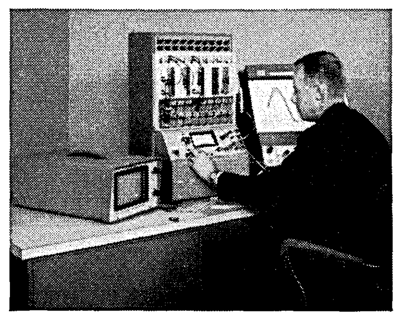
The pointing servos, simulated on EAI desktop analog computers, are highly non-linear, having a sensor with a linear zone equal to 1/600 of its operative range and a total d-c loop gain of over 3,000,000. According to Perkin-Elmer engineers the computers were extremely reliable and "their predicted servo response correlated very closely with measured performance in the actual system."

Perkin-Elmer designers also simulated the astronomer's manual radio-control, coarse pointing system for the telescope and the optical sensors of the guidance system. By using simulation, the engineers obtained considerable data which confirmed the design concept.



Tools for Model Building

EAI has two fully transistorized desk-top computers which offer simple operation and programming. They require no special power or environment. The TR-48, a 48-amplifier computer, is capable of solving complex problems involving high-speed repetitive operation and iterative techniques. It is a sophisticated unit operable by a researcher or systems engineer with a minimum of training. The smaller and highly portable TR-20, the ideal starting computer, is also a successful research tool, capable of solving many problems formerly associated with large-scale computers.



A free computer operations course is provided with every TR-20 and TR-48 purchase at one of the company's computation centers. In addition, EAI offers tuition courses in areas throughout the country (send for details and date of next course).

Every new owner of an EAI computer automatically becomes a participant in the growing EAI Applications Library, and is eligible to receive simulation studies pertaining to all categories of research and design.

Send for detailed literature on EAI TR-20 and TR-48 analog computers, or arrange for a problem-solving demonstration in your plant or office—at no charge.

EAI® ELECTRONIC ASSOCIATES, INC., *Long Branch, New Jersey*

ADVANCED SYSTEMS ANALYSIS AND COMPUTATION SERVICES/ANALOG COMPUTERS/HYBRID ANALOG-DIGITAL COMPUTATION EQUIPMENT/SIMULATION SYSTEMS/ SCIENTIFIC AND LABORATORY INSTRUMENTS/INDUSTRIAL PROCESS CONTROL SYSTEMS/PHOTOGRAMMETRIC EQUIPMENT/RANGE INSTRUMENTATION SYSTEMS/TEST AND CHECK-OUT SYSTEMS/MILITARY AND INDUSTRIAL RESEARCH AND DEVELOPMENT SERVICES/FIELD ENGINEERING AND EQUIPMENT MAINTENANCE SERVICES.

CIRCLE 26 ON READER CARD

THE NCR 315 RCM

A business/scientific computer with an 800-nanosecond cycle time is the latest addition to the 315 family by the National Cash Register Co., Dayton, Ohio. The 315 RMC (Rod Memory Computer) dispenses with ferrite cores and uses, instead, cylindrical, "thin-film," rod-like storage elements.

Maximum capacity is 240K (4-bit) decimal digits or 160K (6-bit) alphanumeric characters. Average access time is 267 nanoseconds per digit . . . six-digit add time is 7.2 usec . . . and 12 x 12-digit multiply time is 200 usec. Floating point arithmetic includes add, subtract, multiply, divide, and normalize.

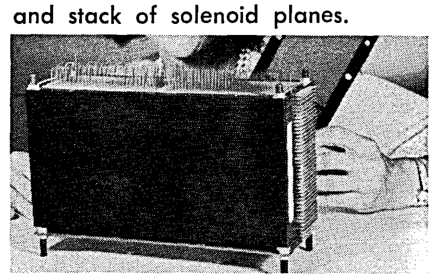
The memory element is a hair-thin, beryllium-copper wire electroplated with a nickel-iron magnetic film. Electrical pulses sent through microscopic coils in the stack magnetize the film in predetermined directions and at predetermined locations. Information can then be read out by sensing the magnetic state of the cylindrical thin film at any given location in the array. Density of the stack is 512 bits per cubic inch—with a 4,000 per cubic inch model reportedly in the lab, with associated integrated circuitry.

With built-in floating point, the RMC will have a FORTRAN compiler, as well as COBOL and NCR's BEST program generator.

Also introduced were higher performance peripherals, including a 120KC tape drive, 1,000 lpm printer, CRAM units with capacities increased to eight and 16 million characters, and a communication controller which enables linkage of up to 100 remote I/O units.

Primary application of the RMC is expected in the area of on-line systems—in banks, for instance. The 315 RMC with 60K characters of memory will rent for $6K monthly. First delivery: mid-'65. ■

Memory module consists of wire rods and stack of solenoid planes.



THE HONEYWELL 2200

The H-2200, which has a one-usec cycle time and a 500-nanosecond scratch pad memory, is an upgraded, enlarged H-200 by Honeywell EDP, Wellesley Hills, Mass. Other features: time-sharing by up to three programs, 15 index registers, and 16-128K (6-bit) characters of core. Eight I/O functions can be performed simultaneously with central processing.

Just as the number of index registers can be expanded to 30, there can also be 32-64 three-character control registers. Half are assigned housekeeping functions, the remainder as a scratchpad for temporary storage or, say, indirect addressing. The full cycle time of the cores is 500 nanoseconds, with access in 250 nsec.

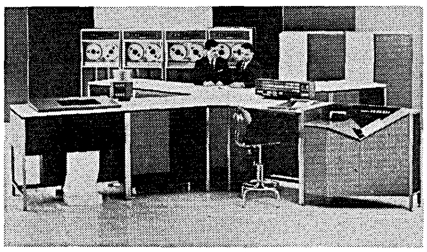
Either four or eight I/O channels are available, each handling 16 or 32 peripheral devices; this makes a total of 256 units that can be on-line. Full peripheral-compatibility among the H-200, 300, and 2200 is achieved with eight new "family interfaces," which also permit core-to-core transfer among the mainframes —including the 800 and 1800. The five peripherals presently in the catalog have been supplemented with 42 more, including an 800-cpm card reader and 19 different communications units.

The company has also announced a doubling of the memory of its 200 and 300—to 64K and 256K, respectively.

The 2200 is program-compatible not only with the 200, but also (thanks to "Liberator") with IBM's 1400's and the 7010. Other software: COBOL '61 and AUTOMATH (FORTRAN II).

Monthly rental for a 48K H-2200 with eight tape drives, printer, and card reader and punch is $10K. First delivery is scheduled for late '65. ■

Another X-shaped computer makes its debut, this the H-2200.



. . . AT LAST — THE FIRST TRUE HYBRID COMPUTER

A completely new kind of signal processor, the Adage AMBILOG™ 200 computer is designed from the ground up to exploit the best of both analog and digital techniques. It combines parallel hybrid arithmetic with stored-program sequential operation.

High processing speeds (often ten times faster than comparably-priced conventional machines) and extensive input/output for both analog *and* digital data make AMBILOG 200 ideal for data acquisition, simulation, and information display.

PARALLEL HYBRID ARITHMETIC

AMBILOG 200 achieves high processing speeds through parallel organization of hybrid computing elements all operating simultaneously on analog and digital operands. Word length of digital operands is 15 bits; analog accuracy is .01%. A fast 30-bit digital accumulator augments the 15-bit hybrid arithmetic.

A typical configuration performs 12 additions, 2 multiplications and 1 division in ten microseconds.

Hybrid arithmetic, A/D and D/A conversion, comparison, and signal routing and conditioning are all carried out under direct control of the stored program.

SEQUENTIAL STORED-PROGRAM OPERATION

Fifteen and 30-bit data words are transferred at high speeds to and from memory and all other parts of the AMBILOG 200 under flexible stored-program control. Core memory word length is 30 bits; cycle time is 2 microseconds. Memory sizes up to 32,768 words are available, all directly addressable. Digital I/O devices include punched tape, typewriter, magnetic tape, and direct data channels. A unique multiple priority interrupt system permits complete servicing of interrupts in as little as 3 microseconds, and facilitates interconnection for multi-processor installations.

PROGRAMMING

Instruction word length is 30 bits, permitting simultaneous control of memory, source and destination selection, word rotation, and Boolean logical operations. The order structure includes provision for recursive indirect addressing and indexing, a number of conditional and unconditional jumps, and program traps.

The AMBILOG 200 Symbolic Translator permits programming in symbolic source language and is easily extended to accept any problem-oriented source language.

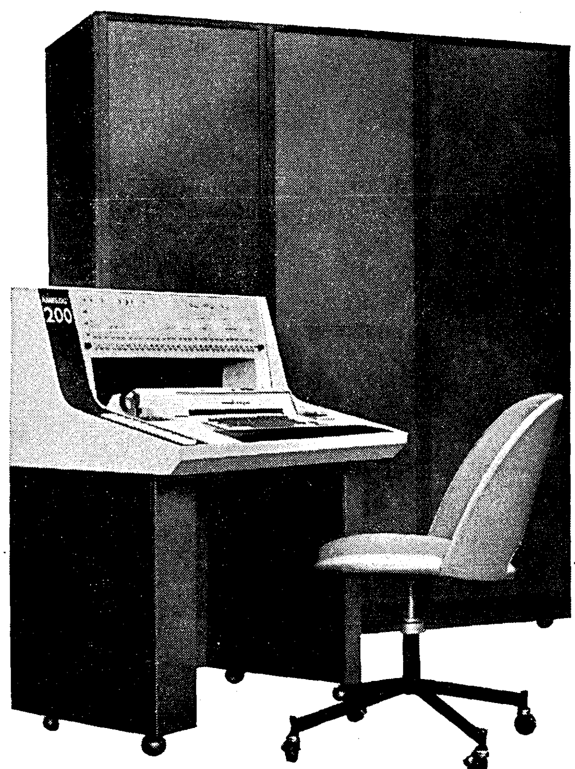
Software support includes programmer and maintenance training, installation and maintenance services, system programs, standard subroutines, and complete documentation.

BROAD APPLICATION

Flexibility, economy, and real-time processing power make AMBILOG 200 a "natural" for

Biomedical monitoring
Seismic data processing
Sonar and radar signal enhancement
Test stand instrumentation
Programmed automatic checkout
Factory test and inspection
Communications research
Process control
Space-vehicle simulation
Simulator trainers
Telemetry receiving stations
Visual display systems

For more facts, contact
I. R. Schwartz, Vice President
617-UN 4-6620.



Adage, Inc. welcomes employment inquiries from professional engineers.

Adage INC

Main Office and Factory
292 Main Street
Cambridge, Massachusetts 02142
West Coast Plant
1145 East Ash Avenue,
Fullerton, California 92631

NEWS BRIEFS

RAND CORP. DEVELOPS GRAPHIC INPUT DEVICE

A graphical computer input device, presently on-line to the time-shared Johnniac at the RAND Corp., Santa Monica. Calif., was discussed recently by Tom Ellis, speaking before the L.A. IEEE Computer Group. The RAND Tablet consists of a "pen" which is tracked as it moves over a page-size surface. This surface contains a thousand etched conducting lines in each of its x and y dimensions, the point of the pen being continuously determined by capacitive effects.

A CRT displays the entire pattern being scribed, and a microswitch in the point of the pen is activated when computer cognizance is desired. This minimal hardware control feature has been extended by "drawing" additional switches in a software-defined region of the tablet. One use of the tablet has been for touching up displayed problem solution data. Some of the current work involves its use for on-line program flowcharting.

Considerable interest in the tablet reportedly has been generated, both by sponsors and researchers in man-machine systems. RAND is committed to build (reluctantly) about 10 units, Ellis said, after which it is expected that industry would produce them. A paper on the system reportedly will be given at the '64 FJCC in San Francisco.

COMPUTER TO BE PLACED IN (MOCK) WITNESS CHAIR

The question, "Are computer records legal evidence?" will be aired in a mock trial—written by Roy N. Freed—to be held at the annual meeting of the American Bar Assn. The trial will involve securing the benefits of the business records rule for computer records, subpoenaing computer records in machine language, circumstantial proof of computer action where no specific records were kept, and a statistical analysis made with a computer.

The fictitious anti-trust trial will try to aid corporate lawyers in deciding whether the types of records desired by management in new computer systems will provide satisfactory evidence in the event of litigation Among transactions involved will be a conspiracy to allocate marketing territories, carried out by a computer without the direct participation of human-folk.

FALL JOINT COMPUTER CONFERENCE SESSIONS SET

An experimental session on "Very High Speed Computers, 1964: The Manufacturers' Point of View" will run for all three days of the Fall Joint Computer Conference, which will be held Oct. 27-29 at the Civic Auditorium in San Francisco, Calif. Beginning the afternoon of opening day, teams of authors representing leading computer developers and manufacturers will describe new systems with which they are currently involved. Session chairman is Dr. Sidney Fernbach of the Univ. of California's computer activities at Livermore, Calif.

There will be 14 technical sessions at which 54 papers (from among more than 150 submitted) will be presented. Following are the sessions and (in parentheses) the chairmen:

Oct. 27—"Very High Speed Computers" (Dr. Fernbach); "Programs and Programming Techniques" (Lewis C. Clapp, Bolt Beranek & Newman Inc.); "Expansion of Functional Memories" (Gordon S. Mitchell, National Security Agency).

Oct. 28—"Very High Speed Computers" (continued); "Machine Organization" (Dr. Harwood G. Kolsky, Stanford Univ.); "Management Applications of Simulation" (Prof. Peter R. Winters, Stanford Univ.);"Digital Software for Analog Computations" (Ralph Belluardo, United Aircraft Corp.); "Input and Output of Graphics" (Donn B. Parker, Control Data Corp.); "Mass Memory" (Harold E. Eden, IBM Corp.).

Oct. 29—"Very High Speed Computers" (continued); "Time-Sharing Systems" (Prof. David C. Evans, Univ. of Calif.); "Non-Numerical Processing" (Prof. Herbert M. Teager, MIT); "Hybrid/Analog Computation—Methods and Techniques" (Dr. Robert Vichnevetsky, Electronic Assoc. Inc); "Computations for Space Programs" (Dr. James H. Turnock Jr., IBM Corp.); "Logical Design and Hardware" (Dr. Donald A. Baumann, Electronic Assoc. Inc.); "Hybrid/Analog Computation – Applications and Hardware" (William J. Quirk, Boeing Co.).

USC STUDIES SHOCK VICTIMS WITH ON-LINE MONITORING

Placing "shock" patients on-line to a computer. researchers at the School of Medicine, Univ. of So. California, have found a way of evaluating the severity of a patient's condition and have improved methods of treatment for several types of shock. Up to three patients can be treated in a special ward of L.A. County General hospital, where they are placed on-line to a 20K IBM 1620, and where an oscilloscope gives continuous reading of blood pressure, EKG, and heart output. Other data are output on a plotter.

The study was established under a $423K grant from the John A. Hartford Foundation, and the IBM system acquired under a five-year, $870K grant from the National Heart Institute of the NIH.

$537K NSF GRANTS BOOST TWO COMPUTER CENTERS

Computation centers of two Chicago-area universities were given financial boosts recently by grants from the National Science Foundation. A matching funds grant of $312K has been made to Northwestern Univ., Evanston, toward construction of its $800K center. It will be known as the Vogelback Computing Center, after a Chicago consulting industrial engineer. Scheduled for October completion, it will serve the entire university.

The computation center at the Illinois Institute of Technology, Chicago, is being expanded and new hardware installed through a $225K grant. Replacing an IBM 1620 is a 32K 7040 and 1401. In addition to offering computer science courses—at least one is mandatory for all science and engineering majors—the university also holds Saturday and summer introductory courses for local high school teachers and students. In the last academic year, more than 2,300 students and 170 teachers completed the courses.

DELTA AIRLINES INSTALLS COLLINS SWITCHING SYSTEM

Delta Airlines is operating what is reported to be the largest privately-owned on-line teletype message switching system . . . connected to more computers than any other computer system. It was designed and installed by Collins Radio Co.; it's their seventh switching system.

Presently handling some 80,000 messages daily, the 256-line system is reportedly capable of switching 240,000. Each line has one station, obviating the necessity of interrogating a station for messages. And if a receiving station is low on paper, messages are queued until the paper supply is replenished, then automatically forwarded. There is also automatic error acknowledgement to a message sender, who must correct his own error replacing manual correction and re-entry at the message center.

Central hardware includes a dual C-8401 system with Bryant discs capable of holding eight hours of traffic at peak load. The system reportedly reduces handling costs from 15 cents per message to nine cents. Still being worked on: three levels of priorities on messages. Completed target date is January '65.

● An order for eight DDP-24 van-mounted computers has been placed by Melpar Inc. with Computer Control Co. Inc. The 24's will be a part of four 2F64A Weapon Systems Trainers—two each—being built by Melpar to train Navy helicopter pilots in anti-submarine warfare.

● A special purpose digital computer, HAVOC (Histogram Average Ogive Circulator), has been delivered by Digital Equipment Corp., Maynard, Mass., to the Eye and Ear

Hospital of the Univ. of Pittsbrrgh. To be used on-line for studying hearing disorders, it is similar to two previous versions designed and constructed of DEC's modules at the Central Institute for the Deaf in St. Louis, Mo. The HAVOC's accept neuroelectrical signals, average away random noise, calculate histograms and ogives of signal amplitude, latency and interval, and display and plots the results.

● For some 1.7 megabucks, Benson-Lehner Corp. of Los Angeles has been purchased by United Gas Corp., Shreveport, La., subject to the approval of B-L stockholders. When effected, B-L's manufacturing, sales and service facilities will be run as a subsidiary of UGC Instruments Inc., a wholly-owned but independently operating subsidiary of UGC. Included will be B-L's facilities in Southampton, England, and Paris, France.

● A Toronto engineering firm has given CEIR Inc. the rights to market seven 7090 programs for the detailing (specifying for fabrication) of steel reinforcing rods used in construction. The software reportedly saves detailers from 30-60% in cost, and includes the analysis, specification, and bookkeeping aspects of the task.

● A thin-film memory system with a 500-nanosecond cycle time and 1K (50-bit) words has been delivered by Fabri-Tek Inc., Minneapolis, to the Naval Air Development Center in Johnsville, Pa. Access time is 200 nsec; with 100 nsec used for indexing, it is equivalent to a 400-nsec read-restore cycle. The system is a modification of the firm's production 300-nsec FFM-202 series.

● Another price reduction brings the cost of a PB250 with Flexowriter down to $26,300 . . . a table model with 3,856 words to $33K. The latter is a 17% drop, according to Packard Bell Computer, Santa Ana, Calif. The firm, which attributes lower prices to production efficiencies and write-offs of engineering costs, reports more than 150 PB250 installations.

● A mobile command-control computer system has been delivered to the Army's CCIS-70 Project at Fort Huachuca, Ariz., by RCA's Communications Systems Div., Camden, N.J. The computer, which outputs a map display of updated logistical and tactical data, has two 16K (38-bit) banks of memory with a cycle time of 5.5 usec, and provision for 63 peripherals.

● A system to digitize photographs for computer-analysis is being prepared by the David W. Mann Co., Burlington, Mass., a division of Geophysics Corp. of America. The system will measure the position (in two coordinates) and the relative densities of micron-sized images while scanning photographic plates at up to 625 mm per minute. Readout is onto mag tape. The Air Force Systems Command has ordered the system under a $220K contract. Three earlier models of the Mann Microdensitometer were delivered last year to MIT's Lincoln Lab to study missile re-entry photos.

● A one-megabuck hybrid computing system has been ordered from Electronics Assoc. Inc., Long Branch, N.J., by General Dynamics' Fort Worth, Texas, division. The HYDAC 2000 with three 231R-V analogs will simulate performance of the USAF/Navy F-111 supersonic jet fighter being developed by GD.

● A 2.5-megabuck follow-on contract for software to be used in command operations control centers has been awarded by the Office of Naval Research to Planning Research Corp., Los Angeles.

● Machine time on a 32K CDC 3600/160-A is being offered by the Computer Lab of Michigan State Univ., East Lansing, Mich. The system includes eight tapes, 1,200-cpm reader, and 1,000-lpm printer.

● Reportedly the first multifont optical reader to be placed into regular commercial use has been installed by Fireman's Fund Insurance Co., San Francisco. The $400K system, by Recognition Equipment Inc. of Dallas, Texas, does not require special type fonts.

● Report generators presently operating on the IBM/1400's will run on forthcoming System/360's, a translator

NEWS BRIEFS . . .

for that task to be made available by the firm. It will not require the optional stored-logic unit. The company previously has announced an optional feature on mods 30 and 40 to run 1401, 1440, and 1460 programs.

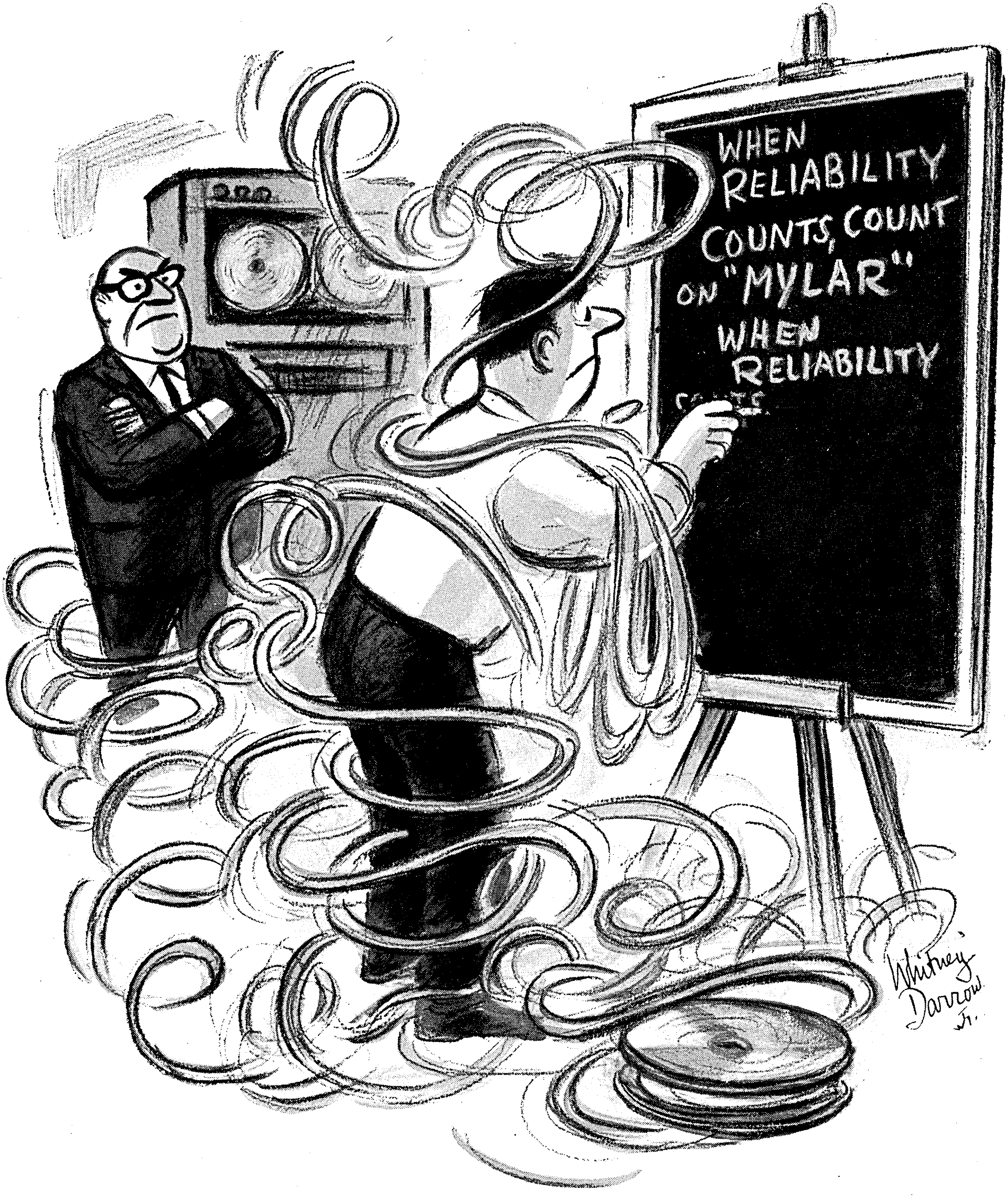
● The ubiquitous Univac 1004 is having its ups and downs aboard a nuclear submarine. Riding in the USS Proteus, the hardware is central to a Ballistic Missile Submarine Tender's dp section, grinding out supply and inventory data. Other 1004's due for three other craft. Crew members are also receiving programmed instructions on the use and maintenance of the processor.

● The analog-conversion equipment firm of Adcom Corp., Chatsworth, Calif., has been purchased for an undiclosed amount by Control Data Corp., Minneapolis. Final transfer is subject to Adcom stockholder approval. The latter's customers reportedly included MIT, Sandia, and Lawrence Radiation Lab.

● With an investment of some $43-million, GE has invaded the European edp market. The dollars represent GE's investment in three new companies which will assume the activities of Compagnie Des Machines Bull, French computer manufacturer. For its dough, GE buys 49 percent of Societe Industrielle Bull General Electric; 51 percent of Compagnie Bull General Electric, a marketing organization; and 49 percent of Societe De Promotion Commerciale Bull (market research and development). The target: an estimated $3-billion sales coming from an anticipated increase from 1K to 10K Common market computer installations by 1970.

● RCA has sold its second service bureau. The buyer: Central National Bank, Chicago. The equipment, including a 501 and peripheral 301, has an original value of over 1 megabuck. Personnel were included in the transfer.

● Two Boston mutual savings banks will share the use of an on-line dp center which will make use of two NCR 315's and 52 on-line teller machines. Valued at $1,250,000, the system will be installed in late '65, will be operated by a jointly owned subsidiary, Savings Bank Service Corp.


WHEN RELIABILITY COUNTS, COUNT ON "MYLAR"
WHEN RELIABILITY
Whitney Darrow Jr.

DATAMATION'S FEATURE INDEX

JANUARY-JUNE, 1964

(Part 2)

MAY

Operating Systems pp. 26-42

An introductory article by T. B. Steel, Jr. describes the evolving function of operating systems since their inception—from the elimination of idle computer time through more sophisticated system control. Benefits and disadvantages of operating systems are discussed. Three additional articles describe the operating systems for the Honeywell 800/1800, the IBM 1410/7010, and the Burroughs B 5000.

The AF's Electronic Systems Division p. 46

by Robert B. Forest

Goals, scope of activity, organization of divisions, and evaluation of a bid are some of the topics discussed in this four-page interview with the chief of the USAF EDP Equipment Office, Col. Edward McCloy.

Computing in the Secondary Schools p. 77

by Fred Gruenberger

The author discusses the role of computers in the high school curriculum. He gives the following reasons for introducing this innovation. a) There is an increasing pace of technology; b) Students at this stage of development are more receptive because of fewer misconceptions about computers; and c) Computers can be used as a teaching device in such areas as mathematics.

JUNE

Design Automation pp. 25-34

An article by E. H. Warshawsky provides an introduction to the use of computers in complex design problems, discussing advantages, reconciliation of long- and short-term goals, hardware and financing. In a second article, David Holstein describes the automated design of transformers.

More Instructions . . . Less Work p. 34

by Christopher J. Shaw

The author maintains that the number of instructions is an inadequate and misleading measure of programming productivity. He points out the differences in cost for programming in machine-oriented and procedure-oriented languages . . . as well as the relative efficiency of programs coded in the two languages.

Conversational Teaching Machine p. 38

by Wallace Feurzeig

Traces the evolution of a computer-based teaching system —from a simple guessing game to a medical diagnosis. Highlight of the system is the emphasis upon "Socratic" techniques: given a basic vocabulary, the student asks for information, makes tentative diagnoses, and is "prodded" by the computer to take a new tack, investigate additional evidence, etc.

Is COBOL Getting Cheaper? p. 46

by Royden A. Cowan

Report of a study of comparative COBOL compilation costs for medium- and large-scale computers. Evidence indicates a trend toward more COBOL compiling power per dollar for medium-scale machines which is missing in larger computer systems.

Program Change Procedures p. 51

by William A. Stewart and John E. Crnkovich

Description of the organization and procedures for making, documenting and controlling program design changes, with special emphasis on military command and control systems.

A Short History of Computer Programming to Date or A Programmer Strikes Back

There once was a group of ambitious young men
Who thought to extend their inadequate ken
And so they embarked on a perilous scheme
To learn to converse with a "Thinking Machine"

And so, they began from a logical start
From first rudiments, and they learned them by heart
And soon they discovered, thru practice and pain
That they could converse with this monstrous "Brain"

And lo' what a wonder, and what a surprise
It answered their questions and told them no lies
Except now and then when a bit went astray
But this wasn't often and caused no dismay

This happy relation went on for a time
And all went quite well and their life was sublime
But then some insidious merchant of fear
Came forward and planted a bug in their ear

Why must we descend to the level of dolts
To learn to communicate with mere nuts and bolts,
For men of our mettle and men of our mien
Should not have to Kow-Tow to any machine

Let's build us a program that knows how we think
To act as a buffer, provide us a link
And so they gave birth to this grand novel plan
And even decided to call it FORTRAN

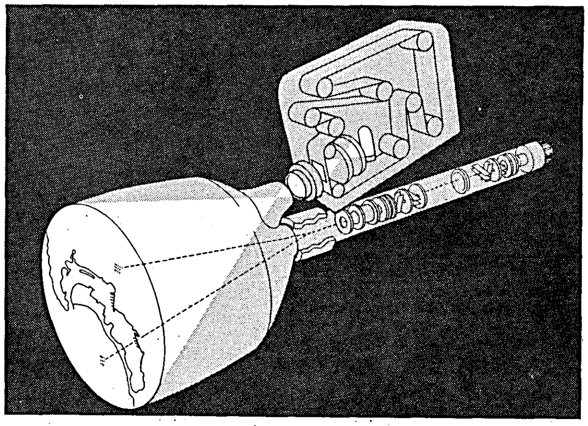
And in a short time there emerged from the group
A program devised to eliminate stoop
But many did scurry, and many did flee
From the need to comply to this conformity

The first was erratic, too crude and too new
More rules were developed, and like Topsy it grew
And as it developed and used more of core
They all started stooping a little bit more

A missed punctuation became a great crime
And each compilation took more and more time
A horrible nightmare to try troubleshoot
Just keep recompiling, for who gives a hoot

Where once stood a proud and unconquerable soul
There now stands a coder without core control
A victim of FORTRAN and all that it means
This new greedy monster controls the machines.

F. S. Ingrassia



New "rear window" display console saves computer time

A unique new display console has been developed which frees the computer from time-consuming production of repetitive material. It is the new S-C 1090 which combines simultaneous cathode ray presentations and film frames on the face of the same tube.

A prime advantage of the new display console is that valuable computer and dynamic display time is not wasted on infrequently changing background data. Maps, business or engineering forms, etc., may be projected on the face of the tube from the inside, in color or black and white, using the built-in film projector. Changing information is superimposed on this image by a CHARACTRON® Shaped Beam Tube. Specific film frames can be selected manually or automatically by the computer.

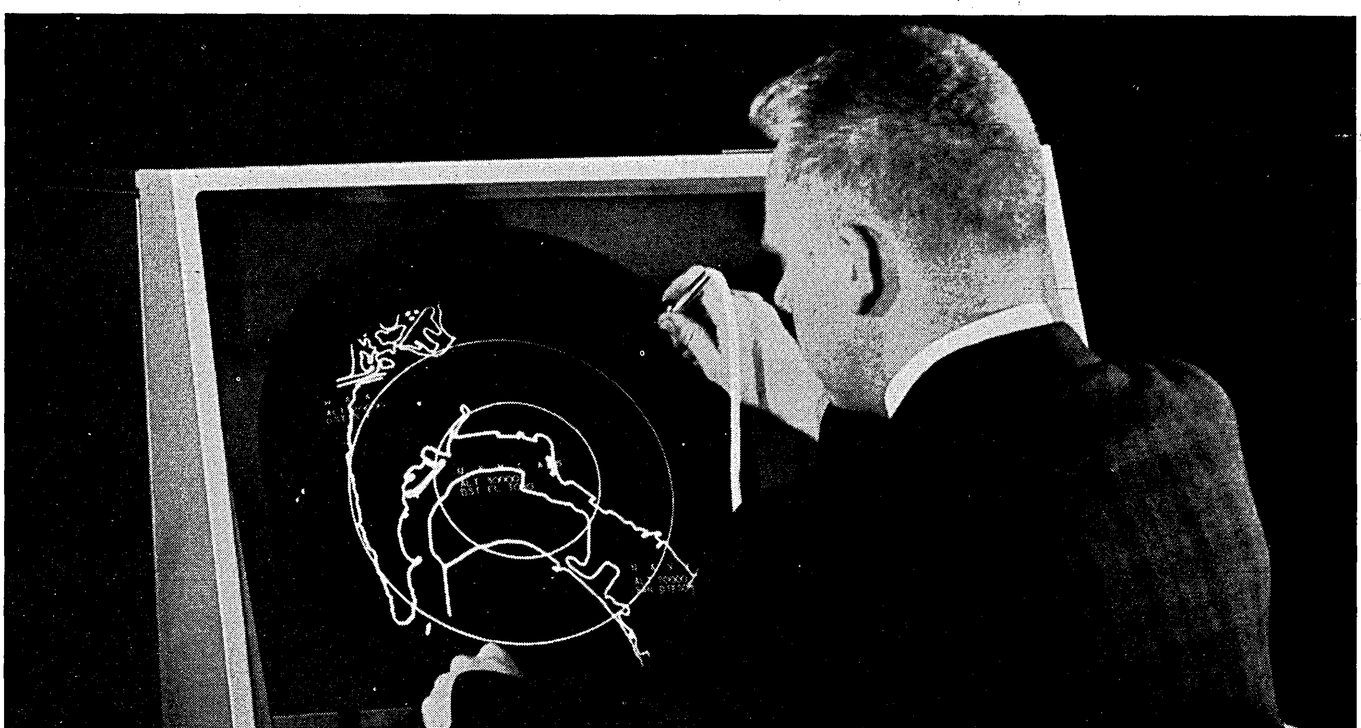
The new development uses a "rear window" tube, so called because the filmed data is projected through a small window located next to the cathode ray gun. The film image is projected onto the inner phosphor-covered surface of the tube from the back and is easily visible from the outside.

The special CHARACTRON Shaped Beam Tube forms alphanumeric or symbolic characters for display on the face of the tube at high speeds. A metal matrix placed within the neck of these tubes produces characters of great clarity. A bright, high resolution spot writing beam is also available to display data from analog inputs simultaneously.

In a typical application, such as tactical air operations, various maps of the tactical area can be produced on film and projected on the screen of the S-C 1090 Console. The computer is then free to present only dynamic data such as movement of aircraft with associated descriptive information.

In business or engineering applications, forms may be projected onto the tube face and filled in with alphanumeric data by the character generator. This compact film projector is offered as a custom option on the standard S-C 1090 Direct View Display Console. For additional information, write to Dept. E-29, General Dynamics | Electronics, Post Office Box 127, San Diego, California 92112.

GENERAL DYNAMICS | ELECTRONICS GD SAN DIEGO



CIRCLE 35 ON READER CARD

Continued from page 19 . . .

Corp. of America. The price: around $12,500, reportedly about ½ of what Control Data asks for it. Lamellar, out of Pacific Palisades, Calif., has other G-15's and peripherals ... the CA-2 card input unit ($2500), the DA-1 Digital Analyzer ($1200), and G-15 modules, selling for one-third the CDC asking price. Lamellar also offers reconditioned Ampex FR 300 tape units with new heads at the bargain basement rates of $3885 for an IBM 7-channel unit, and $2890 for a 1-inch, 16-channel version. Headless as-is transports are available for $1695 or $1895 reconditioned. The company's used gear grab bag also includes some Anelex printer mechanisms, 72 and 120 columns. Lamellar says it will be offering additional second-hand small machines and peripherals soon.

RUMORS AND RAW RANDOM DATA

Don Madden, AFIPS chairman and most recently with IBM, is moving to ACM, where he will become the Association's first executive secretary. Madden will report to the ACM officers, and work out of NYC ... The world's newest and smallest computer society (ELPA) has been formed. It's the Estonian Lady Programmers Association, consisting of two ladies named Liia and Mai Liis, who met through an automatic programming course last summer at the U. of Michigan ... Rumors are that the '68 IFIP meeting will be in Tokyo ... The COBOL maintenance committee has approved the use of outside contract help to update and maintain the beleaguered language, hopes to have a proposed standard for COBOL '65 ready by the first of next year ... Working independently of Ivan Sutherland, John Gilmore of Chas. Adams & Assoc., has come up with a Sketchpad-like device. Using a light pen and a CRT, it is used primarily in drafting ... It looks as if Livermore's Lawrence Radiation Lab will add another machine to its imposing arsenal -- a PDP-6 with remote consoles and multi-computer connections ... ACM and DPMA are continuing their merger talks, which will undoubtedly come to nought, although greater cooperation and coordination may result ... A Houston consulting firm -- Kinotrol -- has a $350K hybrid (a Computer Systems 5800 analog and a 160-A) which it says will run circles around a 7094 on simulation work. System highlight: either machine can get information from the other at any time ... IBM is sponsoring a joint committee composed of IBM, SHARE and GUIDE representatives to tackle the ticklish problem of conversion to the 360. Two members of the committee, Elmer Stonehill, Marathon Oil (GUIDE), and Monroe Fein, IIT Research Institute (SHARE), also head up similar committees within their respective organizations ... University Computing Co., Dallas outfit which rescued CEIR'S L.A. 1604 and built a cooperative SMU and private service bureau around it, will be opening another center in Tulsa shortly ... Control Data, gleefully reporting $61-million worth of post-360 orders in June, will round out their line soon with the 3300 and 3800. The 3300 will take its place in the 31-32-3300 subseries; 3600 users (there are 20 now) will be able to install 3800 modules -- 1/0, arithmetic and memory -- on site to achieve ''economical upgrading.'' CDC says there will be major software system programming compatibility for the whole range.

people IN DATAMATION

■ Daniel Teichroew has accepted a position as head of the Division of Organizational Sciences at the Case Institute. He was formerly professor of Management at Stanford Univ.

■ The appointment of William R. Hoover as vp was announced by Computer Sciences Corp., Los Angeles, Calif. He joined CSC in January as manager of the L.A. Div.

■ Dr. Don Mittleman was appointed professor of computing science and director of the computing center at the Univ. of Notre Dame, Ind. He was formerly associated with the Bureau of Standards.

■ G. Stanley Johnson has been elevated from executive vp to president of Computron Inc., Waltham, Mass. He has been with the company since 1961.

■ Dr. Mervin Muller assumed the directorship of the computing center at the University of Wisconsin. He was formerly with IBM where he was manager of project WELD.

■ Two vp's have been named in IBM's DP Div., White Plains, N.Y. Richard C. Warren has been appointed vp, management controls and John R. Opel has succeeded him as vp marketing.

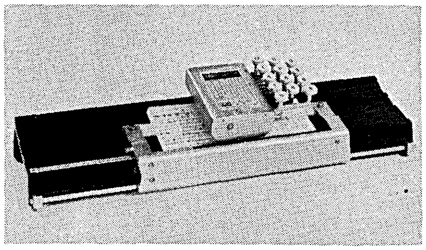
■ Dr. Donald L. Farr has been named head, computers and information processing department at the San Bernardino, Calif. operations of Aerospace Corp.

■ Prof. Gerard P. Weeg is the new director of the computer center at the Univ. of Iowa. He succeeds Prof. John P. Dolch, who is now director of research in the center.

■ William R. Divine, comptroller for the Southern Railway System in Washington, D. C. and the past executive vp of the society, has been named president of the Society for Advancement of Management.

NEW PRODUCTS

card punch
Model 80-A is an improvement over portable mod 80. Among new features: improved spacing mechanism, more rugged punch assembly, transparent chip box, and indicator strip.



Eighty-column model can be adapted to shorter cards. ELECTRO MECHANICS CORP., High Point, N.C. For information:
CIRCLE 200 ON READER CARD

tv display
Alphanumeric display generator uses conventional tv set, accepting six teletype messages of 180 characters each and displaying two messages simultaneously. System is all microelectronic. NORDEN DIV., UNITED AIRCRAFT CORP., Norwalk, Conn. For information:
CIRCLE 201 ON READER CARD

disc memory computer
The 395 reportedly operates 40% faster than conventional accounting machines, uses computer addresses and instructions, and features a 120-word disc file. Access rate is 29 per second. It falls between the Compu-Tronic and the 390 computer. NATIONAL CASH REGISTER CO., Dayton, Ohio. For information:
CIRCLE 202 ON READER CARD

desk-top analog
The TELSIM 6500 analog computer has 10 operational amplifiers, four of which are hooked up as integrators. Others may be used as integrators but without Hold and initial condition feature. Also: two patchboards, 120 points each, and 20 pots. Price $495, including postage in U.S. TESLA RESEARCH FOUNDATION, Phoenix, Ariz. For information:
CIRCLE 203 ON READER CARD

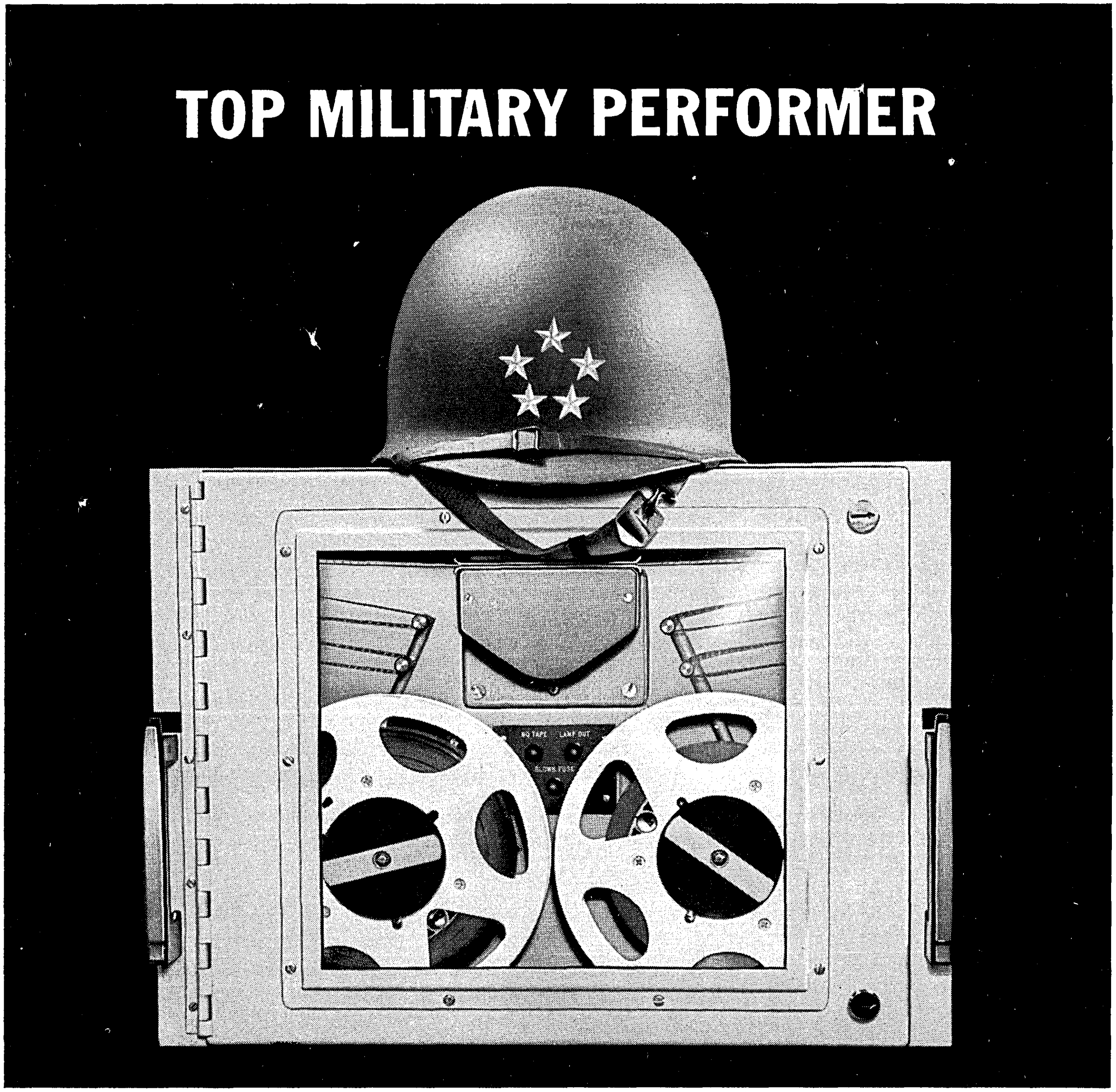
features of fortran
FORTRAN Infograph contains language programming details of 49 F-II and F-IV compilers, lists all statements with brief symbol definition. and has detailed differences among implementations. Device measures 6 x 8 inches. CELESTRON ASSOC., Yonkers, N.Y. For information:
CIRCLE 204 ON READER CARD

paper tape splicer
Operable hand-held or with its base, this unit splices 5, 6, 7, and 8-channel tapes of paper, oiled paper, and Mylar. Six feed-hold pins align tape. ROBINS DATA DEVICES INC.. Flushing, N.Y. For information:
CIRCLE 205 ON READER CARD

gp computer
The 205 is the fourth and lowest-priced of the 200 family, with which it is compatible, and is a card or paper-tape system. Minimum core is 4,000 (20-bit) words, and it adds 14,000 five-digit numbers per second. Floating point hardware is optional. GENERAL ELECTRIC COMPUTER DEPT., Phoenix, Ariz. For information:
CIRCLE 206 ON READER CARD

data collection
Capable of handling up to 128 input stations, the 8010 can feed data to a computer or record them on mag tape. Central unit is the Teleprogrammer, a small, stored program control processor. CONTROL DATA CORP., Minneapolis, Minn. For information:
CIRCLE 207 ON READER CARD

hybrid creator
A digital computer package which reportedly can be made hybrid by ready linkage to one of several ana-



The new fully-militarized 500 RM Tape Reader

At last! . . . a reader conceived, designed and built specifically for military applications — the tape reader that meets all applicable MIL specs including requirements for low temperature operation.

It's Photocircuits' new high-speed 500 RM . . . the fully-militarized photoelectric reader that gives you reading speeds up to 1,000 char/sec. and 8″ reels for maximum data storage. The new 500 RM's exclusive printed-motor direct-capstan drive eliminates completely all the brake and pinch-roller problems typical of today's hybrid or conventional units — giving you accuracy plus reliability unequalled by any ordinary tape reader.

Details and specifications on the fully-militarized 500 RM are yours for the asking . . . or, if commercial/industrial applications are your interest, ask about Photocircuits' new high-speed 500 R. Write today to:

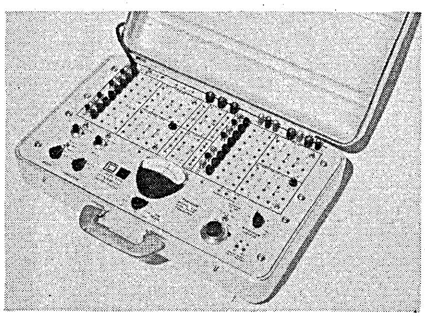
Photocircuits
CORPORATION
TAPE READER DIVISION
GLEN COVE, NEW YORK

CIRCLE 40 ON READER CARD

logs in HYCOMP 250. Consists of a PB 250, Flexowriter, and 64 channels of a-d and d-a conversion. Housing is on one mobile rack. PACKARD BELL COMPUTER, Santa Ana, Calif. For information:
CIRCLE 208 ON READER CARD

portable analog

Model 600 is a six-amplifier, ±10 volt analog computer in an aluminum attache case. Operation can be with real-time, slow-time, or repetitive computation with variable, compat-



ible initial conditions. In graph-plot mode, output is in form for recording on graph paper. COMPUTER DYNAMICS INC., Torrington, Conn. For information:
CIRCLE 209 ON READER CARD

pocket calculator

The Fowler calculator in pocket-watch size, is said to be three times more accurate than a standard slide rule, with multiplication and division accuracy to a minimum of four siginificant digits. Front and back dials are synchronized and have nine concentrically-arranged scales for functions including reciprocals, logs, sq rts, cube rts, natural and log sines and log tangents. Price: $15. SCIENTIFIC EDUCATIONAL PRODUCTS CORP., 30 E. 42 St., New York City. For information:
CIRCLE 210 ON READER CARD

mag tape truck

The 7142 has room for 64 reels and a 24 x 30 inch table on top. Wire dividers are adjustable for thickness of reels, and casters are 4 inch swivel-mounted. STEELCASE INC., Grand Rapids, Mich. For information:
CIRCLE 211 ON READER CARD

collating printouts

The Prebursting Merger, used with the firm's tab card burster, stacks in numerical sequence continuous tab cards and forms that have been processed two-wide. Speed of the two

units is synchronized. STANDARD REGISTER CO., Dayton, Ohio. For information:
CIRCLE 212 ON READER CARD

data acquisition

The DY-2010J can measure up to 200 three-wire analog signals and records digitized readings on an incremental mag tape unit. Its integrating digital voltmeter features 300% overranging and optional six-digit display and autoranging. DY MEC DIV., HEWLETT-PACKARD CO., Palo Alto, Calif. For information:
CIRCLE 213 ON READER CARD

message scrambler

DataGuard scrambles and decodes electronically teletypewriter messages, and has a removable coder/decoder unit that fits in the pocket. Using a random code, it scrambles a word in a different way each time it is repeated in a message. DATACOMMUNICATIONS INC., Moorestown, N.J. For information:
CIRCLE 214 ON READER CARD

transmitting punched holes

Models D511 and D512 are punched card transmitter terminals, handling 80-column cards at 100 and 400 cpm, or whatever Data-Phone and telephone lines will take. Error checks are performed. DIGITRONICS CORP., Albertson, N.Y. For information:
CIRCLE 215 ON READER CARD

data entry keyboard

The 1092 consists of 160 keys adaptable to users' computer-entry needs, provided there's a 1400-series or 360 at other end. Signals sent by the unit can be changed in seconds with a keyboard overlay, representing a completely different set of facts. For output data: the 1053 printer. A smaller version, with 100 keys, is the 1093. IBM DATA PROCESSING DIV., White Plains, N.Y. For information:
CIRCLE 216 ON READER CARD

data acquisition

MOBIDAC III process 50 channels of signals to ±5 millivolts, seven channels in ±5 volt range, and four channels of digital data. Maximum recording rate is 20,000 channels per second with a 12-bit word and 556 bpi, or 13-bit BCD word and 800 bpi. Mag tape unit is Ampex TM-4 (37½ or 75 ips), others can be supplied. SYSTEMS ENGINEERING

LABORATORIES INC., Fort Lauderdale, Fla. For information:
CIRCLE 217 ON READER CARD

tape degausser

The D1250 handles reels to 14 inches in diameter, tape widths to two inches. Erasure time: 130 seconds. It produces a field of 1,450 gausses, and provides complete shielding. MAGNUSONICS INDUSTRIES INC., Farmingdale, L.I., N.Y. For information:
CIRCLE 218 ON READER CARD

tape punch

Model PKP-164 is a photoelectric unit that generates six-level code. It has 63 alphanumeric keys and space bar, solenoid-actuated punch, and own power supply. Operates with oiled and non-oiled paper, Mylar, aluminized Mylar, and vulcanized fiber tape. INVAC CORP., Waltham, Mass. For information:
CIRCLE 219 ON READER CARD

display

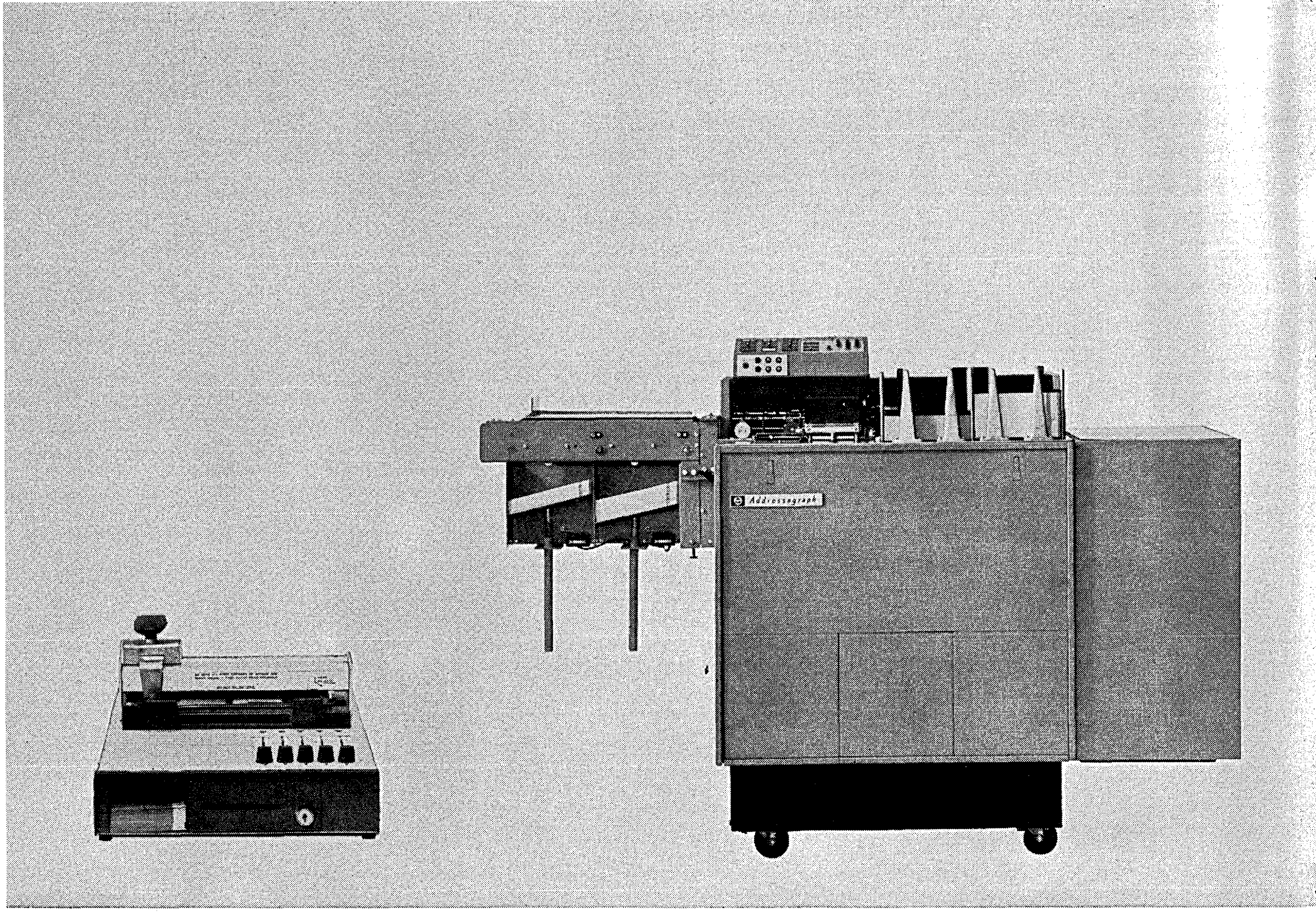
The 340 is an incremental CRT display with an incremental plotting rate of 1.5 usec per point and a random point plotting rate of 35 usec per point. It also operates in a vector and optional character modes. With built-in control and power supplies, only logic level inputs are reportedly necessary for operation. X-y coordinates number 1,024 each in a 9⅜-inch sq. space. DIGITAL EQUIPMENT CORP., Maynard, Mass. For information:
CIRCLE 220 ON READER CARD

paper tape reader

Type 422 is a ruggedized unit with magnetic disc drive and photoelectric sensing of 5, 6, 7 and 8-level tapes at 300 and 600 cps. It accelerates to the lower speed in one millisecond, and stops "on character;" at 600 cps, it stops before the character following the stop character. FERRANTI ELECTRONICS, Industry St., Toronto, Ontario, Canada. For information:
CIRCLE 221 ON READER CARD

mag tape

A new family of analogue magnetic tapes has been developed by Eastman Kodak Co. To be marketed by Consolidated Electrodynamics Corp., Pasadena, the tapes are available in four types, for recording and reproducing at 100, 300, and 600 kilocycles and at 1.5 megacycles. CONSOLIDATED ELECTRODYNAMICS CORP., 360 Sierra Madre Villa, Pasadena, Calif. For information:
CIRCLE 222 ON READER CARD



NEW LITERATURE

MAG TAPE RECORDERS: Three data sheets cover the Model E tape recorder, Model C instrumentation type recorder and the KS-1 digital conversion recording system. KINELOGIC CORPORATION, 29 S. Pasadena Ave., Pasadena, Calif. For copy:
CIRCLE 130 ON READER CARD

FORMAT CONTROL: System, capable of changing raw data to computer format and communicating between remotely located computers, is illustrated in a seven-page pamphlet. INDIANA GENERAL CORPORATION, Keasbey, N. J. For copy:
CIRCLE 131 ON READER CARD

CDC 8090: Six-page brochure discusses application, programming aids, guide to operation, and summary of features of computer system, designed for on-line and general purpose operations. CONTROL DATA CORP., 8100 34 Ave. S., Minneapolis, Minn. For copy:
CIRCLE 132 ON READER CARD

PREVENTIVE MAINTENANCE: Brochure outlines magnetic tape rehabilitation for computer installations, instrumentation tape installations and smaller tape libraries. Also, points out advantages of preventive maintenance to large and small-scale tape users, and features a typical rehabilitation system. GENERAL KINETICS INCORPORATED, 2611 Shirlington Rd., Arlington, Va. For copy:
CIRCLE 133 ON READER CARD

SHORT FORM CATALOG: Describes peripheral equipment for edp including specifications of magnetic tape transports and systems, militarized perforated tape readers and spoolers, read/write amplifiers, switching amplifiers, electronic power supplies, manual controls, magnetic record playback heads, and accessory equipment. POTTER INSTRUMENT CO., INC., 151 Sunnyside Blvd., Plainview, L.I., N. Y. For copy:
CIRCLE 134 ON READER CARD

HYBRID LINKAGE: General system description, system block diagrams and application notes of the ADAGE 770 are given in a 4-page brochure. The ADAGE 770 provides conversion and control hardware necessary to link a digital computer and an analog computer to form an integrated hybrid computing ssytem. ADAGE, INC. 292 Main St., Cambridge, Mass. For copy:
CIRCLE 135 ON READER CARD

HANDBOOK OF MATHEMATICAL FUNCTIONS: Text, tables, graphs and bibliographies of the more important mathematical functions are compiled in 1060 pages. NBS Applied Math. Series 55 is available for $6.50 from the U.S. GOV'T. PRINTING OFFICE, Wash., D.C.

PRODUCT CATALOG: Illustrated 28-page catalog gives information on ribbons, carbons, duplicating and data processing supplies. Index allows reference to individual items and product applications. COLUMBIA RIBBON & CARBON MFG. CO., INC., Glen Cove, N.Y. For copy:
CIRCLE 136 ON READER CARD

PERT: 25-page booklet updates previous studies of the PERT technique in management, and includes general uses, approach, initial planning, operating values, and current uses and problems of the system. BOOZ, ALLEN APPLIED RESEARCH INC., 135 S. LaSalle St., Chicago, Ill. For copy:
CIRCLE 137 ON READER CARD

RESEARCH SERVICES: Four-page folder describing information research services covers information sorting techniques, punched card coding, invert-ed-term-matrix, IBM sorting and computer, reproduction methods and dissemination methods. IIT RESEARCH INSTITUTE, 10 W. 35 St., Chicago, Ill. For copy:
CIRCLE 138 ON READER CARD

MAGNETIC TAPE: Factors which lead to performance deterioration and end of magnetic tape life and various types of tape failure attributable to both normal use and accidental damage are discussed in this 8-page article. MEMOREX CORPORATION, 1180 Shulman Ave., Santa Clara, Calif. For copy:
CIRCLE 139 ON READER CARD

DP ACCESSORIES: Catalog covers splicing, editing systems and accessories for punched tapes, magnetic tapes and edge-punched cards. A listing of replacement reels and storage containers for magnetic and punched paper tapes and chemical care kits are also shown. ROBINS DATA DEVICES, INC., Flushing, N.Y. For copy:
CIRCLE 140 ON READER CARD

H200 TEXT BOOK: Designed for readers with little or no previous computer experience, this 280-page book contains seven lessons ranging from dp principles to programming languages. Available for $4.50. HONEYWELL EDP, Merchandising department, 60 Walnut St., Wellesley Hills, Mass.

PDP-6: A condensed description of the hardware and software that enables time-shared operations on PDP-6 is offered in an 8-page brochure. PDP-6 is a new general purpose digital computing system designed for scientific data processing. DIGITAL EQUIPMENT CORP., Maynard, Mass. For copy:
CIRCLE 141 ON READER CARD

PRINTER PRICES: High speed printer ribbons (speeds to over 1,000 lines a minute) developed for long service and easy operation, are listed according to order number and prices. ANELEX CORPORATION, 150 Causeway St., Boston, Mass. For copy:
CIRCLE 142 ON READER CARD

ON-LINE TELEMETRY DP FOR AF: The dp system, built for the Atlantic Missile Range, is described in terms of general telemetry complex and organization, real-time executive control program and a specific example of the

system's operation. BECKMAN INSTRUMENTS, INC. 2500 Harbor Blvd. Fullerton, Calif. For copy:
CIRCLE 143 ON READER CARD

SEMINARS & COURSES: Day and evening courses scheduled by the firm's newly-formed Institute for Advanced Technology are listed in catalog covering 13 subjects in management and computer sciences, and engineering. CEIR Inc., 1200 Jefferson Davis Highway, Arlington, Va. 22202. For copy:
CIRCLE 144 ON READER CARD

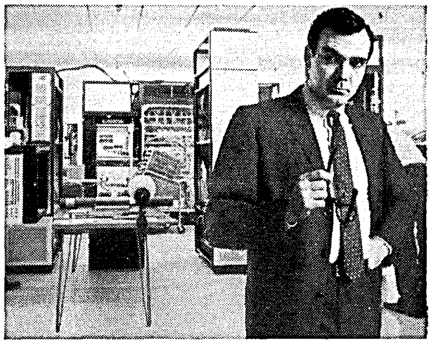
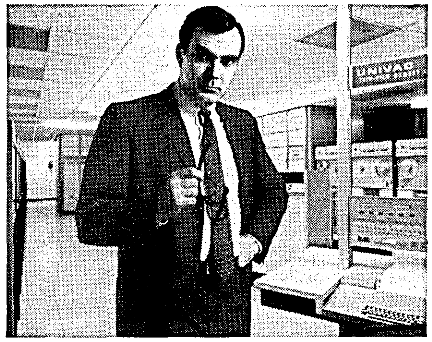
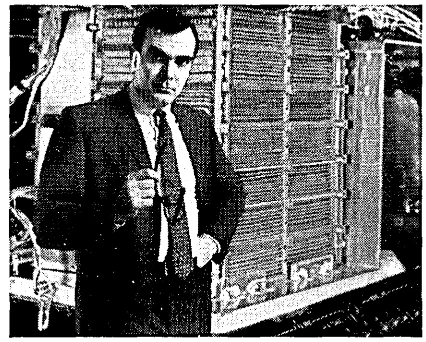
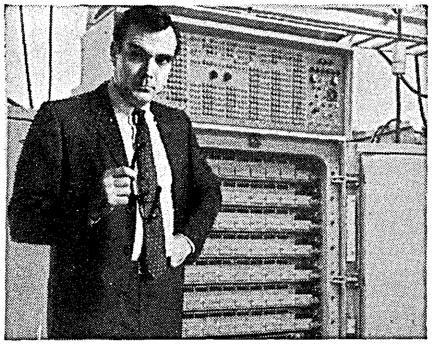
CDC 8080: Computer-controlled typesetting system, capable of setting 10,000 lines per hour, including display advertising, is outlined in brochure. CONTROL DATA CORP., 8100 34 Ave. S., Minneapolis, Minn. For copy:
CIRCLE 145 ON READER CARD

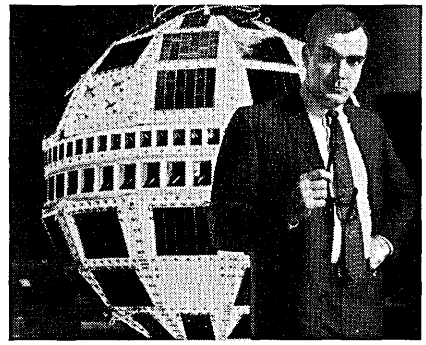
OPTICS: Computer program used to design optical systems is described in folder. It summarizes DIDOC, the mathematical program developed to extend the optical designer's capabilities. IIT RESEARCH INSTITUTE, 10 W. 35 St., Chicago, Ill. For copy:
CIRCLE 146 ON READER CARD

PERSONNEL RECORD: System for the IBM 1401 and H200, accepting all categories of information useful in maintaining accurate data files, is featured in pamphlet. COMPUTER SCIENCES CORPORATION, 650 N. Sepulveda Blvd., El Segundo, Calif. For copy:
CIRCLE 147 ON READER CARD

FERRITE MEMORY: 8½" x 11" reference guide, which summarizes data for ferrite memory cores, is punched and and tabbed for insertion into technical data binders. RCA, MEMORY PRODUCTS OPERATION, 64 "A" St., Needham Heights, Mass. For copy:
CIRCLE 148 ON READER CARD

PRINTER: Brochure describes off-line and multi-function printing system having capabilities of a high-speed alphanumeric printer and graphic plotter, with multiple copy output in either mode. POTTER INSTRUMENT COMPANY, INC. 151 Sunnyside Blvd., Plainview, L. I., N.Y. For copy:
CIRCLE 149 ON READER CARD



AT UNIVAC NO PROGRAMMER IS TYPECAST

He can't be. There are too many diverse contracts in the house to permit stagnation to exist.

Here at UNIVAC-Twin Cities men can move around all over the programming map. For example: into Phase III of Nike work, onto real-time software systems relating to aircraft and missile tracking, into target discrimination, intercept programming, missile guidance, command and control systems, simulation preliminary to hardware design, missile and aircraft flight and computer simulation for design evaluation. Much of this work involves multi-processors. The areas of mechanized design and compilers and language processors are receiving special attention.

The professional environment is complete in every respect. Systems programmers work on their own machines. Development engineers have their own. Software men have a chance to see that hardware design limitations are corrected. Creative interplay between hardware and software R & D activities is a matter of course. The technical content of the work borders on the outer limits of today's knowledge of computer technology. And, programmers can move around geographically as well as professionally.

All in all, UNIVAC-Twin Cities probably ranks as unequalled in industry for its all around environment for the programmer.

PROGRAMMING SUPERVISORS To plan, organize and supervise programming projects, formulate techniques and procedures of programming systems. BS or MS in Math or Science with various combinations of experience in the display engineering activities and 5-10 years' experience.

APPLICATIONS PROGRAMMERS To define, analyze and design solutions to problems, and translate methods developed into computer techniques. BS or MS in Math or Engineering with 3-5 years' large-scale data processing applications experience.

COMMAND AND CONTROL PROGRAMMERS To design real-time information retrieval computer program for AF Intelligence and Command Control Computer Systems. Requires BS in Math, or Science with 3-5 years' sound programming experience.

INTEGRATED PROGRAMMING SYSTEMS Requires BS or MS in Math, Statistics or EE and 2-4 years' experience in programming large-scale digital computers. Must know generative and operational elements and be familiar with auxiliary memory devices.

SYSTEMS PROGRAMMERS To develop large-scale software packages. Requires BS in Math and 2 years' experience in digital computer programming including symbol manipulation, input-output or basic utility routines.

RADAR SYSTEMS PROGRAMMERS BS or MS in Math or Engineering with 2-5 years' experience in systems checkout, radar control, I/O routines, simulation, dynamic radar tests, or executive control to work on advanced real-time systems.

LIBRARY SYSTEMS PROGRAMMERS BS in Math or Science and 2 or more years' experience in assembler-compiler development, simulators (computers, radar/missile), range safety, input/output, mathematical subroutines, or executive control systems.

For more information about these or other openings, or to apply, send your resume to Mr. R. K. Patterson, Employment Mgr., Dept. H-12, Univac Division of Sperry Rand Corp., Univac Park, St. Paul 16, Minn. An Equal Opportunity Employer.

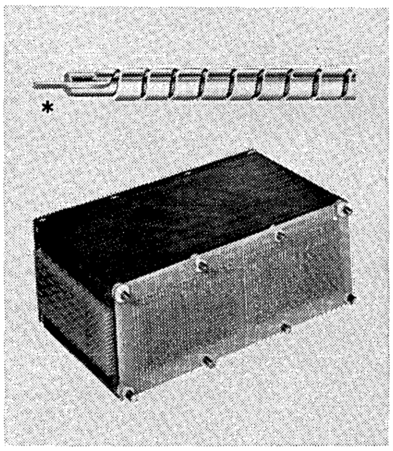
UNIVAC

DIVISION OF SPERRY RAND CORPORATION

NCR is proud to announce . . .

the new 315 RMC (rod memory computer)

The first commercial computer with all thin-film main memory



*Magnified 10 times.

A MAJOR ADVANCE

The new 315 RMC is a major advance in computer technology. Its entire memory of up to 240,000 digits is composed of thin cylindrical wire rods plated with a magnetic thin film. The 315 RMC has an incredible cycle speed of 800 nanoseconds (800 billionths of a second).

COMPATIBLE WITH ALL 315 HARDWARE

The 315 RMC is uniquely versatile. Though cycle speed is 8 times faster than the 315, it is designed to be completely compatible with all existing 315s and all 315 peripheral equipment. NCR users, both present and future, can easily move up to a Rod Memory Computer when additional capabilities are required.

COMPATIBLE WITH ALL 315 SOFTWARE

The command and logic structure of the 315 RMC is identical with all 315s. No re-programming is required. All 315 programs and software, including NEAT and COBOL, may be used "as is." For new applications, BEST, NCR's recently announced program generator, reduces programming time by as much as 50%.

ALSO NEW! FASTER PERIPHERAL EQUIPMENT

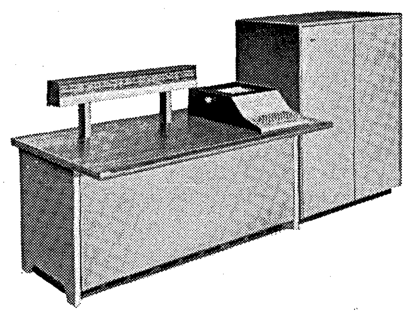
Now available for the new 315 RMC – **and all 315s,** a new line of faster, more efficient peripherals:

- New, faster tape drives; 66KC conversion of data from other computers; 120KC for direct processing
- New 1,000 line-per-minute printer
- New 250 CPM Card Punch
- New 321 Data Communications Controller for expanded on-line and remote inquiry capability
- New built-in floating point arithmetic for scientific applications
- New High Capacity CRAM III (Card Random Access Memory) provides up to 16,000,000 characters of random access storage in each CRAM cartridge.

COMPATIBLE WITH OUR USERS' SYSTEMS

All NCR current and future users benefit from this remarkable new development. The 315 RMC is an important scientific breakthrough – a significant addition to NCR's 315 family. It dramatically extends the life and capabilities of all 315 installations. With a 315, your system can grow as you grow – and you can move up to a high-speed, Rod Memory Computer without paying the penalty in time and money that progress in automation usually costs.

Deliveries of the 315 RMC begin in mid '65. For more complete information, we urge you to send for the booklet describing our new thin-film computer. Write The National Cash Register Company, Dayton, Ohio 45409.



NCR

BE SURE TO VISIT THE NCR PAVILION AT THE NEW YORK WORLD'S FAIR. THE NATIONAL CASH REGISTER COMPANY ®

Continued from page 21 . . .

their screens. With the new equipment it's anticipated one monitor will be able to keep his eye on from ten to fifteen blips, compared to a maximum of five or so under the old system. It's part of a master FAA plan to maintain blanket radar surveillance of all U.S. commercial air traffic. By 1969 the agency plans to have such systems installed at all its 22 ATC centers, with IBM the probable source for the requisite computing gear. Nice business at $3 million a whack.

One possible hitch in FAA's plans: Money. The Atlantic City-bound equipment has been bought and paid for, but the system destined for Jacksonville has been held up pending Congressional authorization of the necessary funds. In a fit of miserliness earlier this year, the House chopped FAA's R & D budget for fiscal 1965 from $42 million to $21 million. It's hoped--confidently by FAA executives and fervently by the IBM salesman on the account--that the Senate will restore most of the axed funds.

YOUNG BLOOD AT ARPA

One of the whizziest kids of them all, Dr. Ivan Sutherland, who concocted Sketchpad while at MIT's Lincoln Laboratory, has taken over the Information Processing directorate long held at DOD's Advanced Research Projects Agency by Dr. J.C.R. Licklider, who's left for a research post with IBM in Yorktown, New York. Sutherland, only 26, takes over a program which is both funding and formulating perhaps the highest-toned computer research occurring anywhere. He'll have a decisive voice in determining ARPA sponsorship, how much and for what, in a program that currently encompasses $13.5 million in grants to ten research centers for way-out work on computers, including the well-known Project MAC at MIT. Sutherland's responsibilities at ARPA won't be quite as broad as Licklider's whose Behavioral Sciences cap was given to Dr. Lee Huff, a greybeard of 31 being recalled from Southeast Asia, but he is expected to blaze the trail for new ARPA thrusts in man-machine communications, time-sharing, programming languages and memory media. Though officially still on Army duty (due for discharge in January) Sutherland has been working on released time with ARPA for the last several months, getting the feel of the reins.

NETWORKS TURNING NEGATIVE ON ELECTION HIP-SHOOTING

The projection by computer of victorious candidates on election night immediately after poll closing, by now almost a hallowed national tradition, has nonetheless aroused the ire of many who consider these projections to have an undesireable influence upon voters in the Far West still waiting to cast their ballots. California Senatorial candidate Pierre Salinger, among others, has called for a moratorium on such projections come next fall. There's some indication the three networks may be willing to go along with this proposal. By holding off on a computer forecast until 11 p.m., EST, there's little chance of a faux pas winner being picked, especially embarrassing since the overlooked candidate has to be the next president of the United States. The see-saw Rockefeller-Goldwater battle in the California primary raised sweat on many an executive brow on those networks which went out on a limb early for Barry. For prudence' sake, if for no other reason, the networks would like to forego very early projection of winners. Odds are, if the networks can come to some agreement among themselves, the prophetic voice of the computer will be muted until at least 11 p.m. EST November 3.

NCR TOTAL EDP SYSTEMS

Programming Research
Sales Assistance
Customer Programming Support
Site Preparation
Systems Analysis

Current expansion and future planning in the EDP area have created the following opportunities for qualified personnel who are ready to take a step forward in their career now, or within the next seven months.

Programming Research

College education plus two years or more programming experience with magnetic tape systems. Challenging opportunity in new and diverse problems in the commercial, industrial, and scientific areas.

Sales Assistance

Education in the business, scientific, or engineering areas is desired with 2 years' experience in the EDP field. Long range interest in the sales area is essential.

Customer Programming Support

Two years' experience is required in programming and related systems analysis with medium to large scale magnetic tape systems. Customer assistance and high interest in debugging would be essential areas of competence.

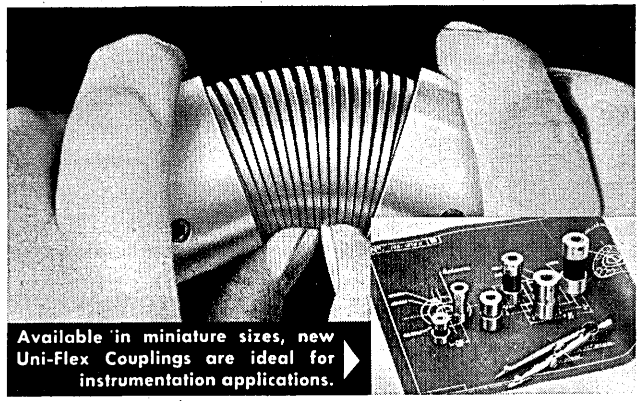
Other available opportunities exist in the area of SYSTEMS ANALYSIS, *and* SITE PREPARATION ENGINEERING.

These positions are in Dayton, Ohio and offer professionally rewarding challenges with a computer manufacturer in a growth position. Confidential inquiries may be addressed to:

T. F. WADE, Technical Placement
The National Cash Register Company
Main & K Streets, Dayton 9, Ohio

An Equal Opportunity Employer

CIRCLE 86 ON READER CARD



New all-steel coupling out-flexes all others!

This Uni-Flex Coupling by Lovejoy embodies a flexing center of 3 triple-thread square-wire springs. This unique construction combines real strength . . . with superior flexibility. The result? Unmatched protection of drive and driven machinery against shock, vibration and misalignment (angular and parallel).

Exclusive Uni-Flex design offers such other benefits as: ■ Durable one-piece construction (with no wearing parts) ■ Simple straight-edge alignment ■ Freedom from maintenance (no lubrication) ■ Adaptability to suit varying drive requirements.

Send for Catalog UF-64.

Lovejoy

LOVEJOY FLEXIBLE COUPLING COMPANY
4949 West Lake Street • Chicago 44, Illinois • Dept. D-84

CIRCLE 46 ON READER CARD

In Today's Data Processing Market . . .

WHAT IS YOUR TRUE WORTH?

You'll never know unless you contact the nation's largest data processing placement service.
Cadillac Associates represents all of the country's leading data processing companies. We presently have over 3,000 positions coast to coast that should be filled immediately.
With Cadillac, you are represented by men conversant in your discipline. And remember, our placement service is yours in ABSOLUTE CONFIDENCE, WITHOUT OBLIGATION AND FREE OF CHARGE. Client companies pay all costs.

SERVICE/SYSTEMS SALES **$12,000**
Excellent co., midwest.
MANAGER, CMPUTER CENTER .... **to $15,000**
Midwest location. Manager — director staff — tape and tab.
PROGRAMMERS .... **$ 8-14,000**
Excellent positions — scientific — comm.
MANAGER — COMPUTER SYSTEMS .... **to $13,000**
Feasibility studies, acctg. oriented.
CHIEF, OPERATIONS ANALYST .... **to $28,000 +**
Applied mathematics.
JR. SYSTEMS ANALYSTS .... **to $10,200**
1401 Systems, programming random access.
REAL TIME PROGRAMMERS .... **$ 9-13,000**
Several locations.
APPLIED MATHEMATICS .... **O P E N**
Engineering and scientic applications.
PROGRAMMERS (EE's) .... **to $14,500**
Symbolic and machine language (real time).
EDP LEAD ANALYSTS .... **$ 13,500**
Manufacturing processing.

May we help you in absolute confidence and without obligation? For further information on the above or other positions, send us a short resume of your background.
Lon D. Barton, President

CADILLAC ASSOCIATES, INC.
29 E. Madison Bldg., Chicago 2, Illinois
Financial 6-9400

"Where More Electronics Executives Find Their Positions Than Anywhere Else in the World."

In California: 120 Montgomery Street YUkon 1-6880, San Francisco
3275 Wilshire Blvd., DU 5-9111, Los Angeles

CIRCLE 82 ON READER CARD

COMPUTER SPECIALISTS

HARDWARE

DEVELOPMENT PROJECT ENGINEERS AND SUPERVISORS: BSEE required with a minimum of 5 years' related experience. These assignments are on state-of-the art development projects within the Advanced Design Department in:

(1) High Speed Digital Circuits

(2) Extremely High Performance Core Memories

These projects will determine the operating characteristics of FUTURE Control Data general purpose computers. Successful performance of assignment will require a high degree of creativity and the ability to manage complex technical activities. MINNEAPOLIS LOCATIONS.

SENIOR LOGIC DESIGNERS: BSEE with a minimum of 3 years' experience designing systems for "fail safe" and "reliable" operations, in addition to an ability to analyze an entire system and construct it with a minimum of logic units. A knowledge of low power dissipation and small physical size transistor circuits for aerospace use is desirable. MINNEAPOLIS LOCATIONS.

PACKAGING ENGINEERS: BSME with a minimum of 2 years' experience in packaging and encapsulation of electronic components suitable for use in space vehicles. Assignments will require substantial vibration, shock and stress analysis experience. MINNEAPOLIS LOCATIONS.

COMPONENTS ENGINEERS: BSEE with experience in circuit checkout and testing in addition to a knowledge of conventional electrical and/or mechanical component parameters. Assignments will be in testing, evaluating and generating procurement specifications and specification control drawings for components used in space programs. MINNEAPOLIS LOCATIONS.

SOFTWARE

SALES SUPPORT ANALYSTS: Consult with CONTROL DATA'S hardware Customers to analyze their problems for computer applications in both pre- and post-sale situations. Experience required in scientific or business data processing programming for medium or large-scale computers. Scientific experience preferred. Positions located at NATIONWIDE SALES OFFICES.

DATA CENTER SALESMEN: Data processing sales experience required. Will sell Data Center computer time and programming services to customers. Must have thorough knowledge of computer applications. Positions located in LOS ANGELES, PALO ALTO, WASHINGTON, D.C., MINNEAPOLIS, HOUSTON, AND LONG ISLAND, NEW YORK.

PROGRAMMING APPLICATIONS INSTRUCTORS: Teach beginning and advanced programming to both CONTROL DATA employees and customer personnel. Must be experienced in scientific or business data processing programming. Scientific experience preferred. Positions located in MINNEAPOLIS, LOS ANGELES, WASHINGTON, D.C. AND PALO ALTO.

PROGRAMMER ANALYSTS: Analyze Data Center Customer problems for computer applications. In addition, you will be involved in sales support work and the preparation of programming proposals. Positions located in LOS ANGELES, PALO ALTO, WASHINGTON, D.C., MINNEAPOLIS, HOUSTON, AND LONG ISLAND, NEW YORK.

COMPUTER SALES ENGINEERS: Digital computer experience in sales engineering and/or appliations programming to sell CONTROL DATA computers and related industrial product lines. Positions located at NATIONWIDE SALES OFFICES.

TO ASSURE PROMPT REVIEW OF YOUR QUALIFICATIONS AND INTERESTS, PLEASE SEND RESUME TO ONE OF THE FOLLOWING AREA STAFFING REPRESENTATIVES:

PALO ALTO:	LOS ANGELES:	WASHINGTON, D.C.:	NORTHEAST:	MIDWEST-SOUTH:
D. J. MORAN, 3330 HILLVIEW, PALO ALTO, CALIF.	P. A. WEBER, 5630 ARBOR VITAE, LOS ANGELES, CALIF.	J. S. FETTIG, 11428 ROCKVILLE PIKE, ROCKVILLE, MD.	J. S. REDSECKER, 295 NORTHERN BLVD., GREAT NECK, N. Y.	T. E. OLDHAM, 8100 34TH AVE. SO., MINNEAPOLIS, MINN.

NOTE: When sending resumes, be sure to mention position or positions of interest as listed above. ATTENTION ACM CONFERENCE DELEGATES: If you plan to be in Philadelphia August 25-28 for the ACM Conference, write T. E. Oldham, Control Data Corporation, 8100 34th Ave. So., Minneapolis, Minn., now to arrange an interview with members of CONTROL DATA'S technical staff.

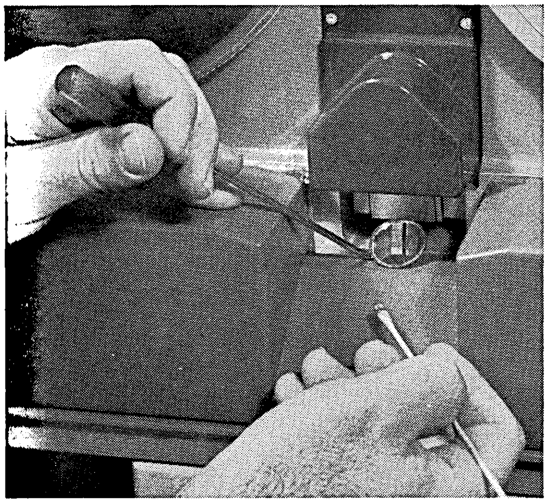
CONTROL DATA
CORPORATION

An Equal Opportunity Employer

CIRCLE 79 ON READER CARD

4 REASONS FOR BUYING ONLY PREMIUM TAPE

(Memorex, of course!)

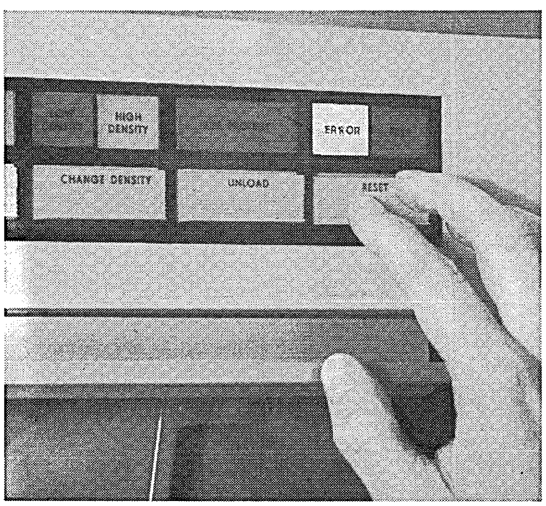


Reduced maintenance down-time. Premium tape increases head life and reduces head replacement. Its better adhesion of oxide coating and tougher, smoother coating surface minimize operating interruptions resulting from oxide build-up on heads and guides.

Memorex magnetic tape is premium tape.

Longer life. Premium tape minimizes tape stripping. It assures error-free performance long after inferior tape breaks down and becomes loaded with dropouts. The more severe the use, the more the economies afforded by premium tape's tougher, smoother coating.

Memorex magnetic tape is premium tape.



Greater reliability. Premium tape remains error free — pass after pass, reel after reel — and provides greater security of data in demanding routines. Despite the somewhat higher price of premium tape, few users can afford to miss the economy inherent in its greater reliability.

Memorex magnetic tape is premium tape.

No rejects. Premium tape provides freedom from rejects because it is always read-pass perfect — reel after reel. Its price is higher, perhaps. But its effective cost is less because premium tape delivers machine-time savings by eliminating pre-testing and maximizing error-free operation.

Memorex magnetic tape is premium tape.

Memorex tape is premium tape. No need to pre-check it. You can place Memorex computer tape directly in service — reel after reel.

Memorex certification means what it says: Memorex computer tape <u>is</u> error-free. Extra care, extra steps and scrupulous attention to every detail make it that way. We know the importance to you of having a tape you can depend on.

Are you on our mailing list to receive the Memorex Monograph Series of informative technical literature? Write 1172 Shulman Ave., Santa Clara, California

MEMOREX

PRECISION MAGNETIC TAPE

CIRCLE 2 ON READER CARD