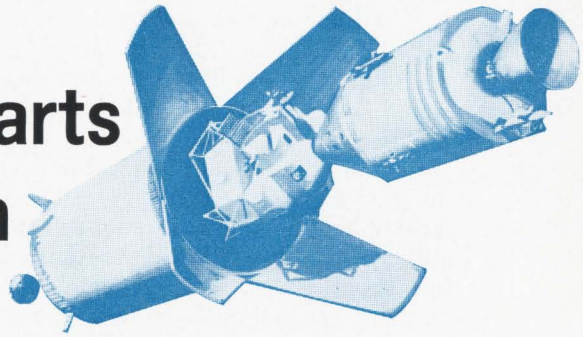
Space electronics

Electronic navigator charts man's path to the moon



Apollo, our boldest step into space, will be guided by a computer that is both elementary and advanced. Its memory holds a fixed program but its applications are diverse and flexible

By Albert L. Hopkins

Instrumentation Laboratory, Massachusetts Institute of Technology, Cambridge, Mass.

When man embarks on his first truly extraterrestrial adventure—a trip to the moon—he will be guided by a little box not much bigger than an unabridged dictionary, but much more versatile. The box is the Apollo guidance computer, which will determine whether the astronauts are on course and help them turn corners when the time comes. Prototypes of the integrated-circuit computer are now being tested on the ground.

The computer is part of the guidance and navigation system of the Apollo Command Module and the Lunar Module. The system performs different tasks during free-fall and powered flight.

In the free-fall phases—earth and lunar orbit, lunar descent coasting, and coasting through space between the earth and the moon—the system must determine the state of the vehicle—for example, orbital elements, position or velocity. These are part of an open-loop navigation process. The system is capable of performing the navigation itself but ground-based tracking stations duplicate the process simultaneously whenever possible.

Powered phases—boost, injection, ascent and entry—are short and critical. They require automatic steering for stability and accuracy, while the system directly controls the translation and rotation of the vehicle. These are closed-loop guidance functions. Each powered maneuver changes the velocity, as determined during navigation phases, until the guidance sensors indicate that the desired velocity is reached.

The guidance and navigation (G&N) system was designed by the Instrumentation Laboratory of the Massachusetts Institute of Technology and produced jointly by the AC Electronics division of the General Motors Corp., Kollsman Instrument Corp., a subsidiary of Standard Kollsman Industries, Inc., and the Raytheon Co. Its three major parts are:

- A three-gimbal inertial platform with three floated integrating gyros, each with a single degree of freedom, and three pulsed integrating pendulum accelerometers. The gyros and accelerometers continuously indicate changes in angular orientation (attitude) and translational velocity.
- An optical angle measuring unit—a sextant and scanning telescope in the Command Module, an alignment telescope in the Lunar Module.
- The guidance computer.

Some of the principal computer activities are correlated with navigation and guidance functions in the table on page 110. Critical phases are in color. These categories are interdependent because they share data, subroutines and interface activity. The computer's program has sections for executing dozens of jobs, but can do only about 10 at a time.

How it all began

The Apollo G&N system concept originated almost eight years ago, during a study for an unmanned space vehicle which would photograph Mars and return to Earth. The study concluded that the vehicle would require a navigation and guidance computer with inertial and optical sensors. The

The author

Albert L. Hopkins is the assistant director of MIT's Instrumentation Laboratory, where he specializes in the development of computers for inertial guidance and navigation systems.

computer logic and erasable memory would be built with core-transistor circuitry and the program would be stored in a wired-in read-only memory, called a core rope. Shortly afterward, MIT began designing the Apollo guidance computer with the same kind of circuitry plus a coincident-current ferrite-core erasable memory.

By late 1962, quantity production of monolithic integrated circuits had begun. The computer was redesigned to take advantage of the greater reliability, high speed, small size and packaging uniformity of IC's.[1] The direct-coupled transistor logic (DCTL) NOR gate was chosen as the basic circuit largely because of its simplicity. The designers chose only a single logic circuit type—a three-input NOR element, consisting of three transistors and four resistors on a monolithic silicon chip. The computer contains over 5,000 such circuits.

The decision to use the single logic circuit paid dividends for the Apollo project because microcircuit vendors could supply Apollo NOR gates some time before they overcame problems in producing more complex circuits. A family of specialized circuits might have saved space, but space was deemed less important than reducing the problems of specification, manufacture, and test of these relatively new circuits.[2]

It had long been known that all Boolean logic functions could be implemented with a single circuit [see "Single-circuit design," p. 115]. Several systems—MIT's Polaris guidance computer, a digital differential analyzer, for one—used a discrete-component NOR gate as its sole logic element type. But the concept has been applied rarely in general-purpose computers because specialized circuits required using fewer discrete components.

The core rope concept[3] was retained, because of the high density and the reliability advantages of nondestructive memory. The memory's main disadvantage is that it can be loaded only at the time of manufacture, but this was tolerable because program changes in any given computer are infrequent. However, each program must be completely debugged before assembling the rope.

When the decision was made to have a separate Lunar Module distinct from the Command Module, still another version of the G&N system was needed for both spacecraft. It was called Block 2 [Electronics, Dec. 12, p. 111]. The initial G&N system (Block 1) was put into production for the developmental command modules now being flown. The Block 2 guidance computer has twice the over-all speed of the Block 1, and has 50% more memory and consumes less power, with little change in size and weight.

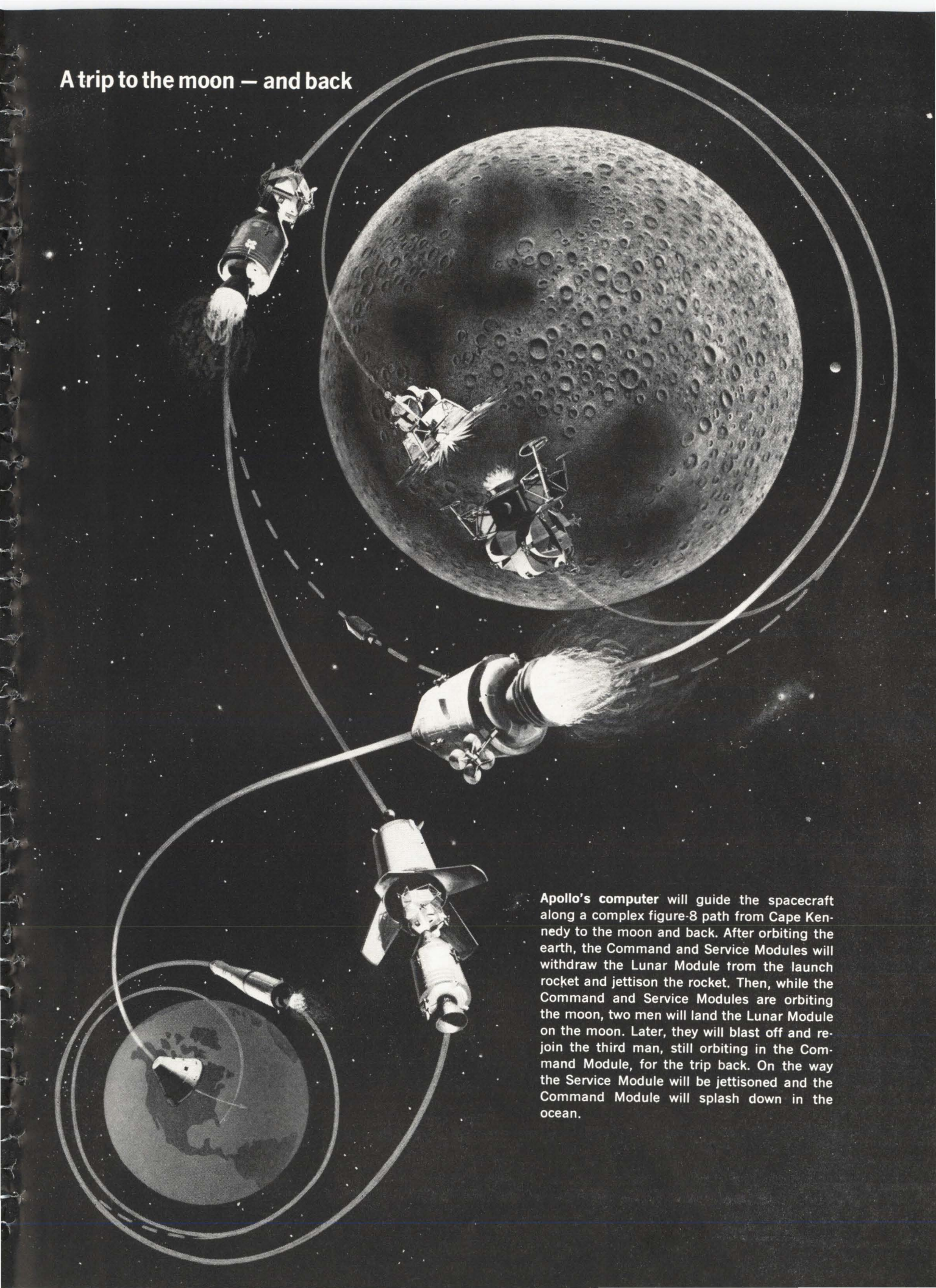
What's in the computer

Today's typical guidance computer is about a cubic foot in volume, and can do about the same job as large-scale central processors of ten years ago. The newer, larger, earth-based data processors can spread a work load over a period of time to balance

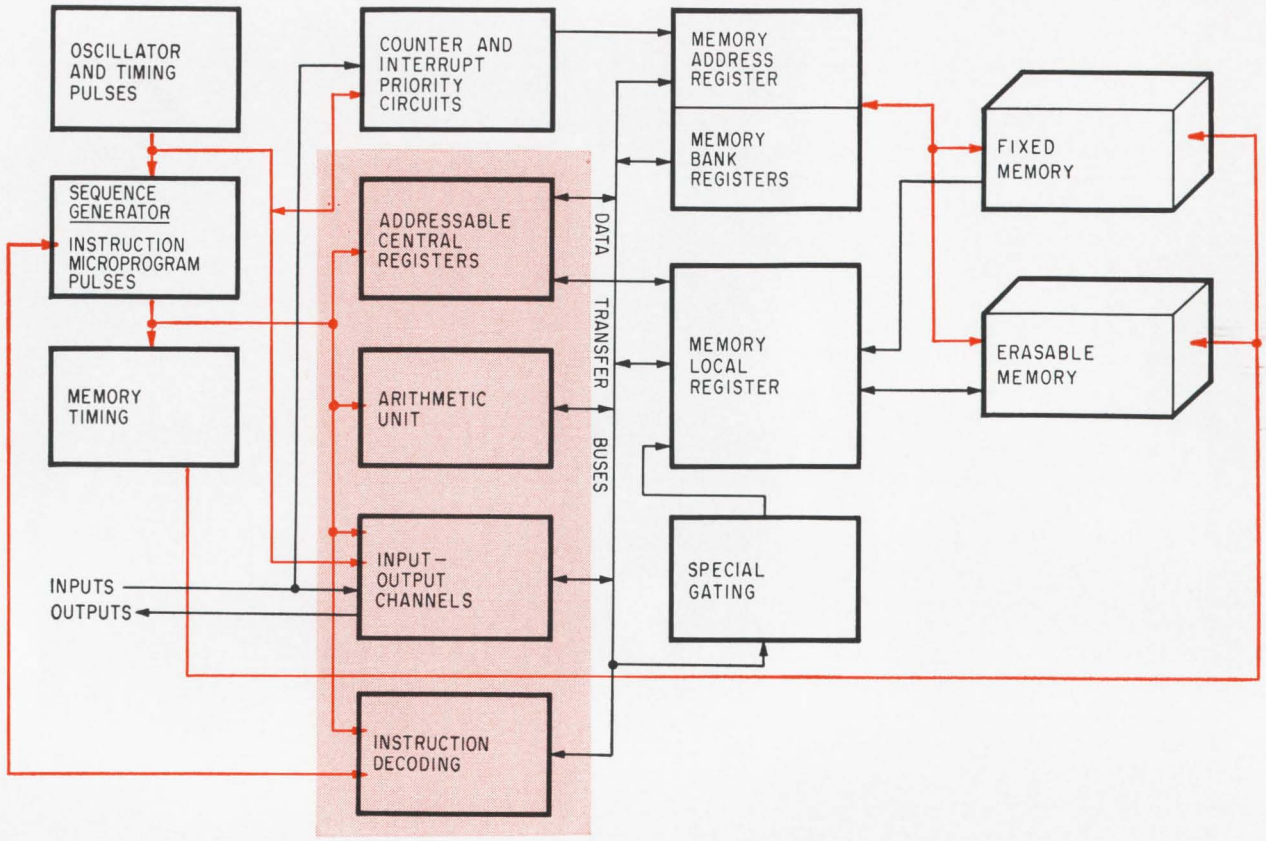
Phases of the moon shot and G&N activity

Phases	Alignment and system test	Inertial guidance	Optics sightings	Attitude control and small translations	Thrust control	Radar	Manual steering	Navigation measurements, calculations & display	Uplink navigation parameters
Prelaunch	*	*	*						*
First stage		*						*	
Second stage		*						*	
Third stage		*						*	
Earth orbit	*		*					*	*
Injection toward moon		*						*	
Module transposition				*			*		
Coasting toward moon	*		*	*				*	*
Midcourse corrections		*			*			*	
Lunar orbit insertion		*			*			*	
Lunar orbit	*		*					*	*
Lunar module descent injection		*		*	*			*	
LM descent coast	*			*				*	
LM powered descent		*		*	*	*		*	
LM final approach		*		*	*	*	*	*	
LM landing		*		*	*	*	*		
LM prelaunch	*		*			*			*
LM ascent		*		*	*			*	
Midcourse rendevous		*		*	*	*		*	
Terminal rendevous				*			*		
Injection toward earth		*			*			*	*
Coasting toward earth	*		*	*				*	*
Entry		*		*				*	

A trip to the moon — and back



Apollo's computer will guide the spacecraft along a complex figure-8 path from Cape Kennedy to the moon and back. After orbiting the earth, the Command and Service Modules will withdraw the Lunar Module from the launch rocket and jettison the rocket. Then, while the Command and Service Modules are orbiting the moon, two men will land the Lunar Module on the moon. Later, they will blast off and rejoin the third man, still orbiting in the Command Module, for the trip back. On the way the Service Module will be jettisoned and the Command Module will splash down in the ocean.



Organization of the Apollo computer shows the central processor (in tint block), the data paths (black lines) and control paths (colored lines). The data-transfer buses are the computer's backbone.

internal computing and input-output activity. For example, they can keep its various facilities busy by time-sharing. A guidance computer's central processor is also time-shared among numerous jobs, but must react to a large demand for input-output service in milliseconds rather than minutes. The Apollo computer can execute about 40,000 instructions per second.

It will be installed in the lower equipment bay of the Command Module, where the navigator will stand after his couch has been stowed. In the Lunar Module it will be behind the astronauts as they stand at their control stations. The crew can give commands to the computer and supply data and decisions when the computer requires it. The computer, in turn, supplies the crew with piloting data. The crew-computer interface is embodied in the display and keyboard unit, described later in this article. Interfacing circuits also connect the computer with other spacecraft systems.

Design of the computer

In addition to the memories, the computer has two other principal sections. The central section includes an adder, an instruction decoder, a memory address decoder and seven addressable registers. The sequence generator provides the basic memory timing and sequences of control pulses for each instruction; it also contains the interrupt priority circuitry.

The backbone of the computer is a set of 16 data-transfer buses, which interconnect the registers. Instructions can address registers in either memory, but the program cannot change the contents of the fixed part. Each memory word has 15 information bits and a parity bit. Data is stored as 14-bit words with a sign; instruction words have three order-code bits and 12 address-code bits.

Every instruction lasts an integral number of memory cycles. The normal sequence of instructions can be broken by a number of involuntary sequences, which are not under normal program control. These are triggered either by external events—an astronaut's entering data from the keyboard, for instance—or by certain overflows within the computer, such as a sum becoming too large for the register containing it. The breaks in sequence may occur between any two program steps and may be divided into two categories: counter incrementing and program interruption.

The shorter the better

Most guidance computers have word lengths of about 24 bits. The Apollo guidance computer is unique in having a 16-bit word. This short length was made possible by a decision to perform navigation calculations with multiple-precision arithmetic, in which a single quantity may occupy more than one computer word.

A guidance computer should have the shortest possible words to keep the circuitry simple, small and fast. Shorter words cut down on the number of memory sense amplifiers needed, for instance. One is needed for each bit in a word. These high-

gain class A amplifiers are considerably harder to operate with wide margins of temperature, voltage or input signal than NOR gates are. Also, the time required for carry propagation in a parallel adder is proportional to word length.

The Apollo computer's data words are generally of two classes; those for navigational computations, and those for control. The first class requires a precision of 27 to 32 bits ($10^{8\pm1}$). The second class of variables can usually be represented with 15 bits. The precision required for navigational variables can be satisfied by using two words. Even if word lengths were 15 to 28 bits, two words still would have been required.

To accommodate both positive and negative numbers, the logic designer must choose among at least three possible number representations: one's complement, two's complement, and sign-and-magnitude. For a binary number such as 1110001011 (equivalent to +907 in decimal notation) the one's complement is simply the complement of each individual bit, or 0001110100; the two's complement is the same as the one's complement except that a 1 is added to the right-most bit position, or 0001110101. These complements represent −907. In sign-and-magnitude notation a bit is added to represent the sign, usually 0 for positive and 1 for negative; then +907 = 01110001011 and −907 = 11110001011.

The Apollo number system is a modified one's-complement scheme using two sign bits for overflow representation in the adder. This combines the advantages of the other two systems while sidestepping their disadvantages. The extra sign bit is available because the parity bit does not pass through the adder. In the conventional one's-complement representation, an overflow alters the sign bit. In the modified scheme the sign is not altered because of the double-bit representation. An independent sign representation for multiple-precision numbers is then possible. During operations such as multiplication, two's-complement arithmetic is used.

Sign-and-magnitude representation was rejected although it is the most straightforward for human inspection. However, it requires either a magnitude comparison or conversion to one of the complement representations when positive and negative numbers are added. In straight one's-complement notation, sign reversal is easy. However, zero is ambiguously represented by a word of all 0's or of all 1's, and addition may require an extra operation called end-around-carry. Two's complement notation avoids this and is convenient for input conversions from devices such as pattern generators, encoders, or scalers; but sign reversal is awkward.

Since the computer often would be using many multiple precision words for navigation, this was considered in the choice of number system and in the organization of the instruction set. Multiple-precision representation commonly requires the sign bits of all component words to agree; nevertheless, in the Apollo computer the signs of the components can be different [see "Multiple-precision arithmetic," p. 116].

Computer characteristics

Word transfers	Parallel
Word length	16 bits=15 data +1 parity
Number system	Modified one's complement
Memory cycle time	11.7 μsec
Fixed memory	36,864 words
Erasable memory	2,048 words
Normal instructions	34
Involuntary instructions (interrupt, increment, etc.)	10
Interrupt options	10
Addition time	23.4 μsec
Multiplication time: 14 x 14 bits	46.8 μsec
Double-precision addition time	35.1 μsec
Increment time	11.7 μsec
Number of counters	29
Power consumption	Less than 100 watts (including two DSKY's)
Weight	58 pounds (computer only)
Size	1.0 cubic foot (computer only)

The algorithm for double-precision multiplication is directly applicable to numbers with independent signs. A special double-precision add instruction simplifies the treatment of interflow. Independent signs are not permitted in double-precision division; both operands must be positive and the leftmost non-sign bit of the divisor must be 1.

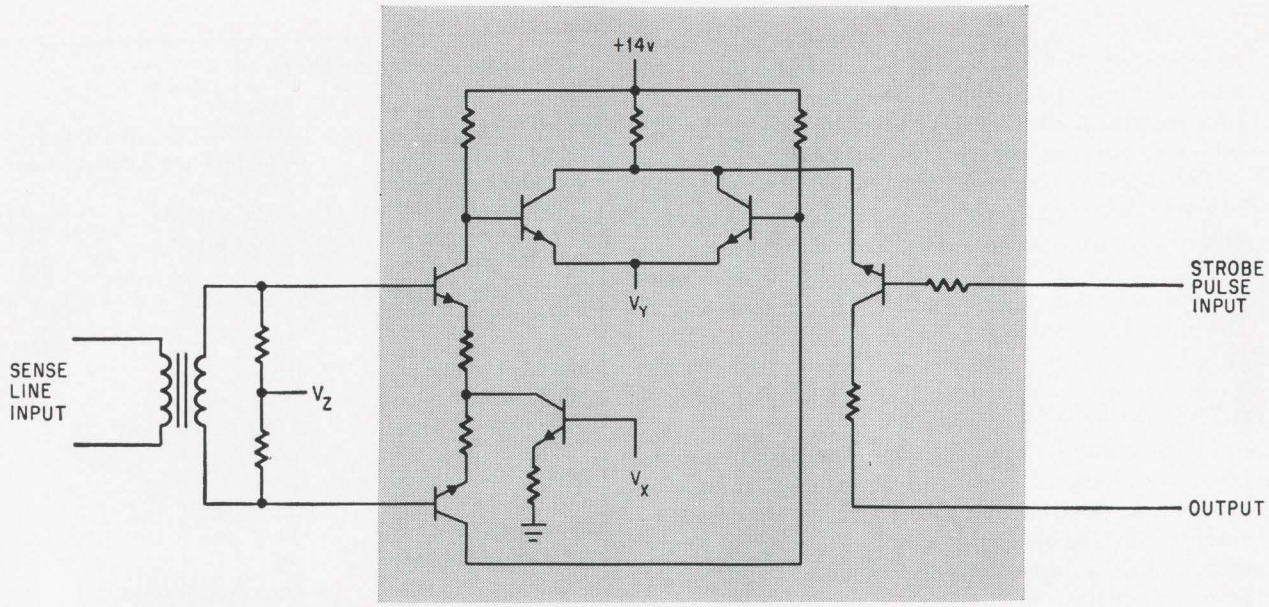
Two memories

The 2,048-word erasable memory is a conventional coincident-current, ferrite-core array. Nothing would be gained by using a faster memory organization, since the 36,864-word fixed memory requires a relatively long cycle. Both have a cycle time of 12 microseconds.

The array is wired in 32-by-64 planes which are folded to fit into a 9-cubic-inch module. A double-ended transistor switching network generates bi-directional currents in each selection wire. One wire in 32 is selected by switch circuits in an 8-by-4 array, and one wire in 64 is selected in an 8-by-8 array. The outputs of these silicon transistor drive circuits vary with temperature so that they match the requirements of the lithium-ferrite cores from 0°C to 70°C.

The integrated-circuit sense amplifiers are considerably more complex than the integrated NOR gates, but there are only 32 per computer. Each amplifier can be carefully selected. None have failed in several million device-hours of operation, although sense amplifiers have historically been the weak link of computer memory systems.

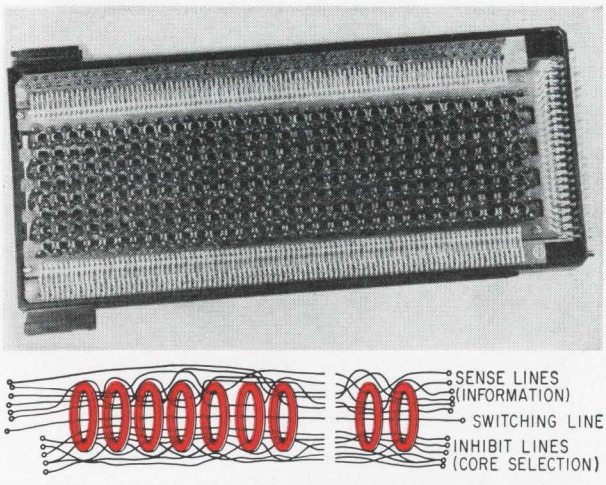
Large fixed memories are rarely used in ground computers, but guidance computer designers favor



Sense amplifier circuit amplifies signals from both the fixed and the erasable memory in the Apollo computer. All components inside the tinted block are in an integrated circuit. Each computer has 32 of these amplifiers.

them. They occupy little space and are highly reliable. Fixed program and data limit alteration of mission plans, but help assure that the computer program is identical through all phases of testing and in flight. It also permits recovery from temporary malfunctions which would alter the contents of an erasable memory—as happened, for instance, during the flight of Gemini 4.

Early models of the MIT core rope memory actually resembled lengths of rope. Large numbers of wires thread through or bypass toroidal cores, as shown in sketch below, to create 1's and 0's.[5] Address decoding is also partly wired in. As a result, its cycle time is longer than that of some other transformer memories whose address decoding is external. However, bit density is extremely high—approximately 2,000 bits per cubic inch, including all driving and sensing circuits, interconnections, and packaging hardware.



Wires through a core store a 1 in a core-rope memory while wires bypassing the core store a 0. Magnetic switching of a core, triggered by currents in the switching and inhibit lines, generates pulses in those sense wires that thread that core; 16 wires at a time are gated to sense amplifiers to read one computer word. Each of the six core-rope modules in the Apollo computer contains 512 cores and 192 wires.

High density is achieved by storing a large number of bits in each magnetic core. The total number of bits is the number of cores multiplied by the number of wires. Each of the Apollo memory's six modules contains 512 cores and 192 wires, or 98,304 bits of information.

A link to ground

Computer words flow over prelaunch and in-flight radio links between the computer and ground control. The upward and downward data rates are different; therefore the mechanizations differ considerably.

The downlink rate is 50 words or 800 bits per second. During one memory cycle the interface stores a full 16-bit word in a flip-flop register; then, upon command, it seems the bits serially in a burst to the communications system of the spacecraft.

Each bit received on the uplink requires a memory cycle; the maximum rate is 160 bits per second. Transfers from the computer to spacecraft display units also require one memory cycle per bit, as does data from the radar measurement subsystem. In these serial pulse trains two adjacent pulses differ in weight by a factor of two and positional notation is employed.

Information is also transferred incrementally. A sequence of pulses is transmitted over a single channel, but each pulse represents the same value, or weight. An incremental receiver counts pulses to form a word, where a serial receiver shifts pulses to form a word. The incremental transfer of information permits high precision and standardization in analog data transmission. For example, the

Single-circuit design

Computer logic circuits are all variations on three basic logic functions: AND, OR and NOT. These three basic functions in turn can be implemented with a single circuit, the NOR gate, shown at right; it has the logical function:

$$D = \overline{A + B + C}$$

read "NOT (A or B or C)." All four variables may have either of two voltage levels. If A, B, and C are all at the more negative level, the transistors are cut off and D is at the more positive level. But if one or more inputs rise, the corresponding transistor turns on and D drops to its more negative level.

When B and C are tied to ground, D is the inverse of A and the circuit is an inverter, or NOT circuit. If the output of the circuit shown is the input to such an inverter, then the inverter's output is:

$$E = \overline{D} = \overline{\overline{A + B + C}} = A + B + C$$

and the combination of two circuits is an OR gate. Likewise if A, B and C are the outputs of such inverters, whose inputs are F, G and H, then

$$D = \overline{A+B+C} = \overline{\overline{F}+\overline{G}+\overline{H}} = \overline{\overline{F}} \cdot \overline{\overline{G}} \cdot \overline{\overline{H}} = FGH$$

read F and G and H; the four circuits form an AND gate. Logically the inverse of an OR function is the AND function of the individual inverted variables.

Two-way functions simply require tying the unused input to ground. For functions of four or more variables, the outputs of additional circuits are connected to each other; only one of the circuits can be connected to the +4 volt supply. The ability to drive other logic circuits is increased by connecting NOR gates in parallel.

Complex logic functions are built up of alternate layers of AND and OR gates. Successive layers of NOR gates can be considered alternately as OR gates with positive inputs and AND gates with negative inputs. Thus the function:

$$E = D_1 \cdot D_2 \cdot D_3$$
$$= (A_1+B_1+C_1) \cdot (A_2+B_2+C_2) \cdot (A_3+B_3+C_3)$$

can be realized with exactly four NOR gates. The reason is that if

$$D = \overline{A + B + C}, \text{ then } \overline{D} = A + B + C$$

and that

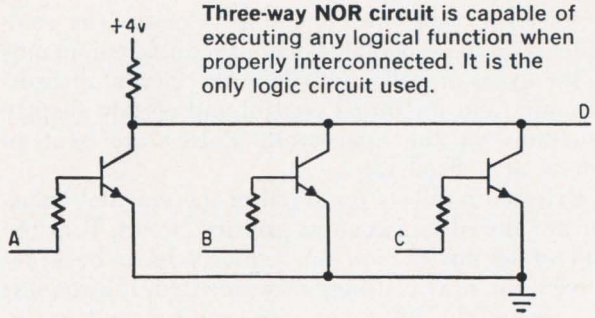
$$D = \overline{A + B + C} = \overline{A} \cdot \overline{B} \cdot \overline{C}$$

Therefore,

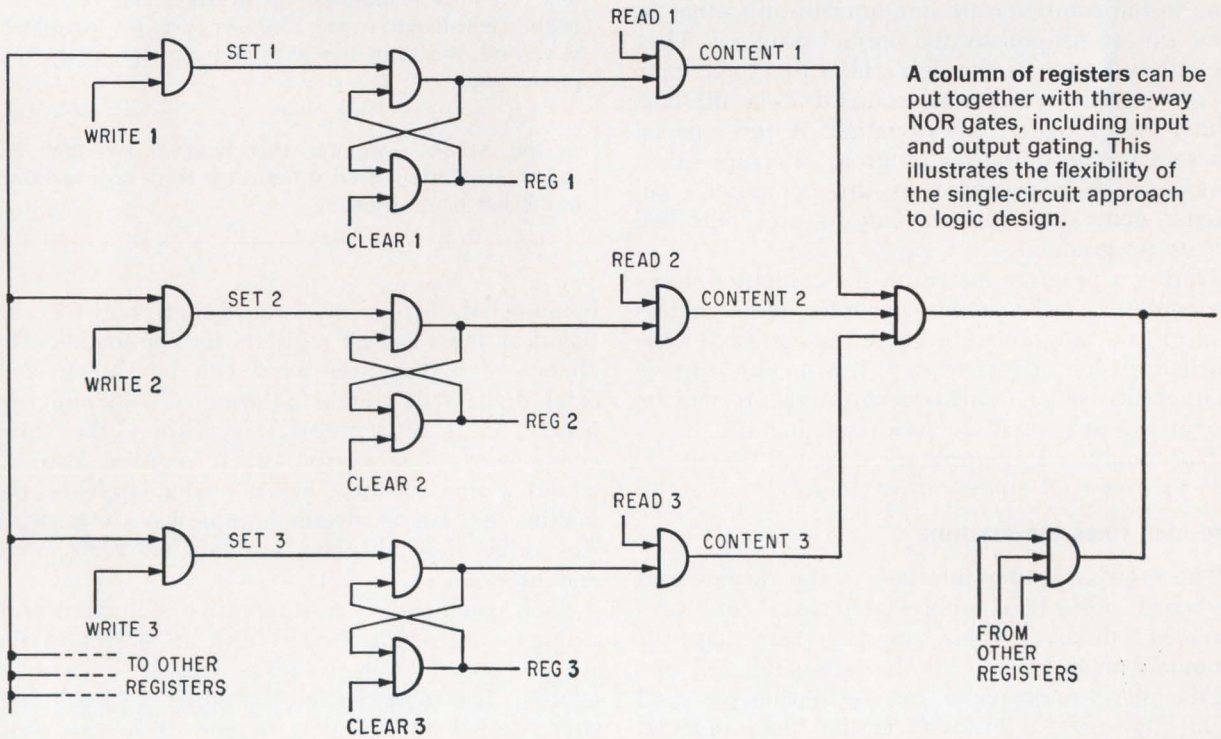
$$E = \overline{D_1} \cdot \overline{D_2} \cdot \overline{D_3}$$
$$= (A_1+B_1+C_1) \cdot (A_2+B_2+C_2) \cdot (A_3+B_3+C_3)$$

If the intermediate variables D are needed, they must be inverted.

For example, the block diagram shows clearly how NOR gates can be interconnected to form a register or a single bit position in each of several registers without using multiple NOR gates for each logic function. Two gates cross connected form a latch or flip-flop; both the input and the output can be gated, or the ungated ON or OFF latch output is available. A six-way OR is at the extreme right.



Three-way NOR circuit is capable of executing any logical function when properly interconnected. It is the only logic circuit used.



A column of registers can be put together with three-way NOR gates, including input and output gating. This illustrates the flexibility of the single-circuit approach to logic design.

Apollo accelerometers are incremental, producing a pulse for each unit change in velocity. The computer uses incremental transfer for angle commands to the gyros and the gimbals in the inertial measuring unit, and for thrust control and certain display functions in the spacecraft. Pulses are sent in bursts at a fixed rate.

External requests for serial or incremental transfer are stored in a counter priority circuit. If at the end of an instruction no requests have been received, the next instruction is executed. If a request is present, the program is interrupted and an incrementing memory cycle is executed. Each counter is a specific word in erasable memory. During the incrementing cycle the computer reads the word, increments or shifts it, and stores the result in the original location. Overflows from one counter may be inputs to another. The computer processes all such requests before proceeding to the next instruction in the original program. This type of interrupt enters synchronous incremental or serial information into the working erasable memory.

Since counter words are in the erasable memory, any program always has ready access to them. Each increment or shift requires a memory cycle. To avoid taking too much time from guidance and navigation calculations, the aggregate counting rate is limited. This sometimes requires a logic circuit between the interface and the priority circuit to prevent the input pulse rate from exceeding a chosen level.

Commands or feedback for discrete actions are given by discrete signals—individual binary digits or small groups. These actions close switches, change mission phases, fire jets, start displays and control many other actions. Most discrete inputs to the computer are non-interrupting—that is, they are acted upon by the normal program. They are either d-c inputs through a filter to a logic gate, or a-c signals transformer-coupled to a flip-flop, which is reset after interrogation. A few special discrete inputs initiate a program interrupt—they announce their presence to the computer's sequence generator. Discrete outputs are controlled by the program.

During a program interrupt, the computer stores the contents of the program counter and transfers control to a program subroutine that processes the particular type of interrupt. When an interrupt is being processed no subsequent interrupts will be accepted, but all will be processed in turn.

The computer is the primary source of timing signals to about 20 spacecraft systems.

The man rules the machine

The crew-computer interface is the display and keyboard unit. It is abbreviated DSKY and pronounced "diskey." There are two DSKY's in the command module, one for the navigator and one on the pilot's main panel, for use during powered flight. One DSKY is installed in the lunar module.

Above the DSKY's keyboard is a numeric electroluminescent display of three 5-digit registers for data and three 2-digit registers for commands. The 15 bits of a computer word can be shown as 5 octal digits—the digits 0 through 7—in one register. The three registers can display the three components of a vector. Each register also includes a sign position; when a sign appears, the number is read as decimal; otherwise it is octal. The other digit displays are labeled verb, noun and program.

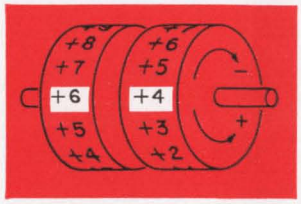
Each command is an imperative statement consisting of an action (verb) and an object of the action (noun). For example, "display" "time to ignition" is a request from the crew; or "load" "desired gimbal angles" is a request from the computer. These statements are expressed numerically;

Multiple-precision arithmetic

Independent signs arise naturally in multiple-precision addition and subtraction. Forcing signs to be identical would be costly because every operation may require sign reconciliation. For example, in decimal notation,

$$(+64) + (-46) = (+18)$$

Suppose the addition is done in one-digit decimal registers that indicate both positive and negative numbers in sign-and-magnitude form, like the mechanical registers below. In multiple precision, the left-hand register carries ten times the weight of the right-hand register, so that +64 would become (+6, +4) in the two registers, and −46 would become (−4, −6). The two registers operate independently so that the operation is



$$(+6, +4) + (-4, -6) = (+2, -2)$$

Considering the weight of the left-hand register, the result is clearly +20, −2. The sum of these is +18, but the signs are mixed unless corrected.

The same result can be obtained if operand signs are mixed. For example, +64 can be represented by either (+6, +4) or (+7, −6), so

$$(+7, -6) + (-5, +4) = (+2, -2)$$

The result again has mixed signs. On the other hand,

$$(+6, +4) + (-5, +4) = (+1, +8)$$

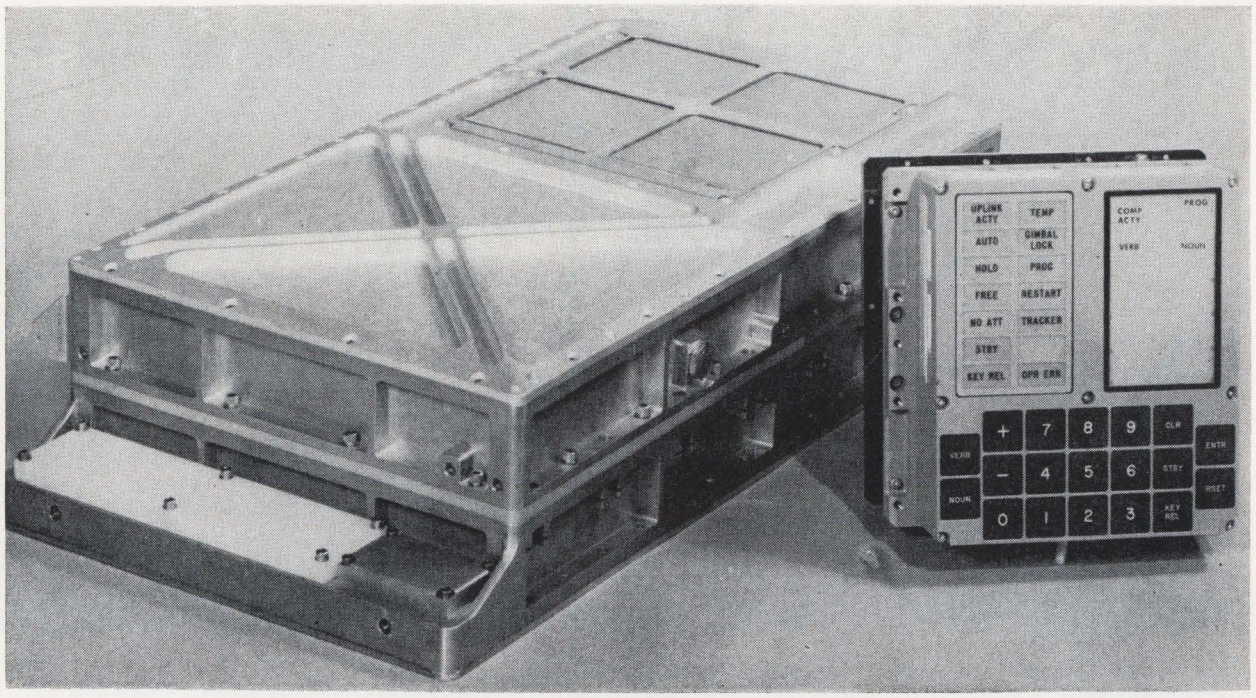
so that here correction is unnecessary. Still another possibility exists if the problem is stated:

$$(+7, -6) + (-4, -6) = (+3, -12)$$

The answer is still correct, but since one-digit registers contain the numbers an overflow has occurred in the right-hand register. In this case, the overflow is carried over into the left-hand register, with the proper sign:

$$(+3, -12) = (+2, -2)$$

In the Apollo computer this kind of overflow is called an interflow to distinguish it from an overflow in the left-hand register.



Apollo guidance computer, viewed from the rear, and its display and keyboard unit.

the crew looks up the verbs and nouns in a glossary. As a crewman keys in statements, they appear in the 2-digit registers of the display where they can be checked for accuracy. When the computer makes a request, the verb and noun display flashes on and off to indicate the computer origin of the command, and to attract the crew's attention.

Key code inputs interrupt the computer program. The keyboard's 19 pushbuttons include the 10 decimal digits, plus and minus signs, and 7 auxiliary keys. The 19 key functions are encoded into five binary signals by a diode matrix mounted directly behind the keyboard.

Program lights tell the operator what major programs the computer is running. The DSKY also has discrete alarm and condition lamps, a condition light reset key, and a key with which the operator relinquishes his use of the display lights to the computer—a function which permits the operator to decide whether his command has a higher priority than the computer's request. If a keyboard entry sequence is in progress when the computer program has a request or a result of display, a condition lamp notifies the operator of the fact. When he is ready to let the computer use the display, he depresses the release key.

Software

Guidance and navigation programs are executed under control of three special programs that ease the task of the human programer and insure that priorities are observed in the computer. These special programs are the interpreter, the executive, and the waitlist.

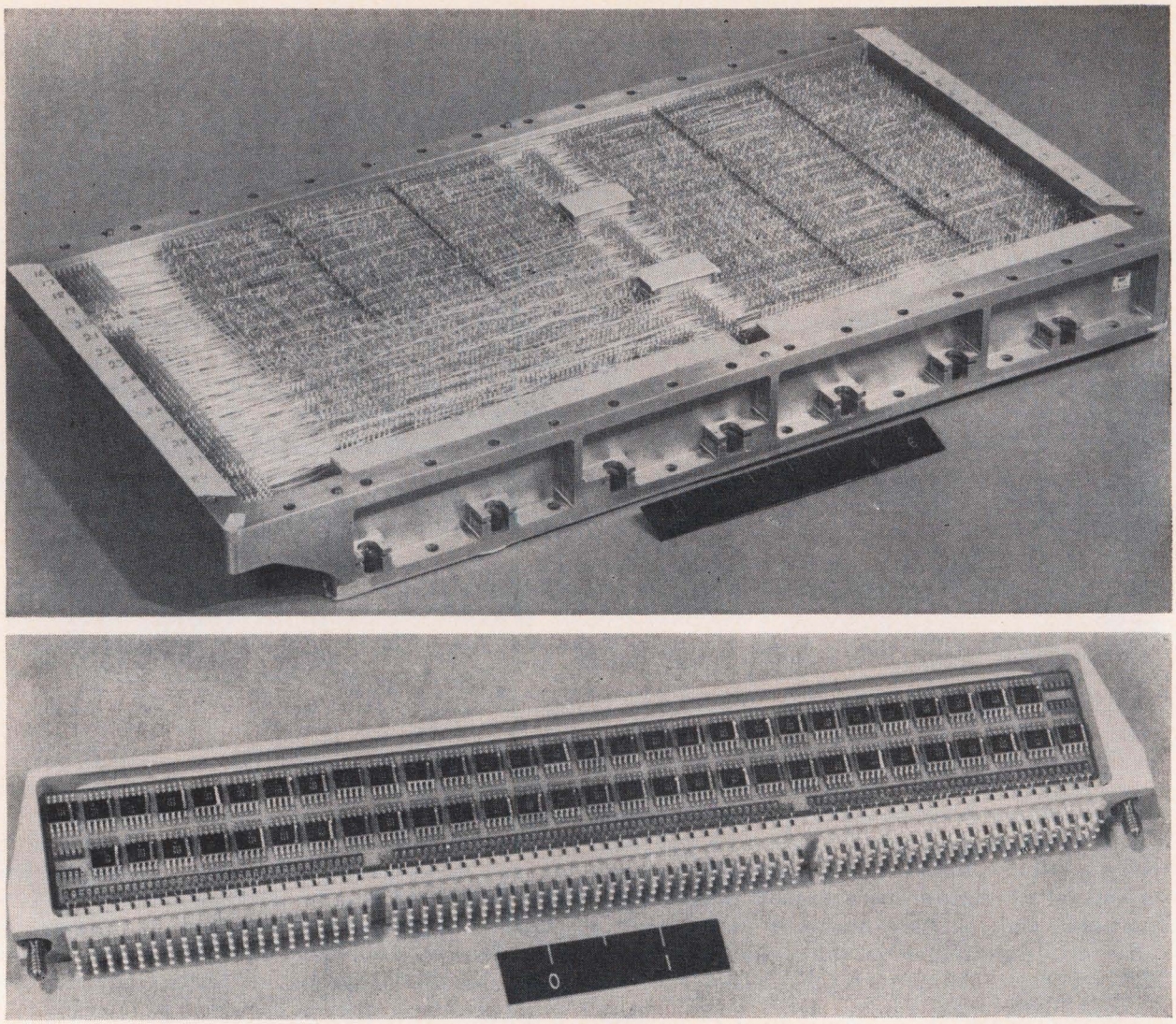
Most of the guidance and navigation programs are written in a special notation and stored directly in the fixed memory. The interpreter translates the notations into a sequence of subroutine linkages, which in turn execute the program. A program executed with the interpreter takes more time than would a machine-language program written with an automatic compiler; however, it occupies less storage space and saves the time of writing a program in basic machine language. The interpreter occupies only a few hundred words in fixed storage and requires only one instruction for relatively complex operations, whereas compiled programs would need several instructions.

All programs except interrupt routines are controlled by the executive routine. Executive-controlled programs are called "jobs" to distinguish them from "tasks," controlled by the waitlist. The routine controls job priority, permits time-sharing of erasable storage, and displays computer activity on the DSKY. As one job runs, the executive checks at least once every 20 milliseconds to see if another job of higher priority is waiting to be executed. If so, control is transferred to the new job until the first job again becomes the one with highest priority.

When a job must wait until an external event occurs, such as the attainment of a specific velocity in powered flight, the executive may suspend it, or "put it to sleep." The job's temporary storage is left intact, so the executive can "awaken" the job when the event occurs.

When a job is finished the executive transfers control to a terminating sequence which releases the job's temporary storage to be used by another job. Approximately ten jobs may be scheduled for execution at one time.

Waitlist tasks are run in the interrupt mode, and must be no more than four milliseconds long. A longer program could cause excessive delay in



Logic module contains 120 monolithic integrated-circuit flatpacks, 60 on each side. These modules plug into a tray, whose wrapped wires interconnect the modules. A second tray holds the memory and its drive circuits.

other interrupts waiting to be serviced, since one interrupt program inhibits all others while it is running.

If the waitlist is to initiate a lengthy computation, it will call the executive routine so that the computation is performed as a job during non-interrupted time.

Modules and trays

Signal interconnections—the major mechanical problem in guidance computers—take up approximately three-quarters of the computer volume. The IC flatpacks are interconnected with multilayer boards, and wrapped wires connect the modules.

A basic design goal was small modules that could be easily installed and removed, for the sake of producibility, testing, diagnosis and maintenance. Another goal was to keep the computer's volume small. The goals were attained with 24 modules each containing two independent groups of 60 flatpacks with 72 connector pins. Two gates are on each flatpack; flatpacks and pins are connected to the board by spot welding.

All modules plug into two carriers called trays. One tray holds the logic, power supply and interface modules, and the other contains the memory, oscillator and alarm modules. The trays are magnesium alloy frames into which the modules are inserted by jacking screws. These provide a good thermal path between modules and tray, which in turn is screwed to a cold plate.

References

1. R.L. Alonso, Hugh Blair-Smith and A.L. Hopkins, "Some aspects of the logical design of a control computer: a case study," IEEE Transactions on Electronic Computers, Vol. EC-12, No. 5, Dec. 1963, p. 687.
2. Jayne Partridge, L.D. Hanley and E.C. Hall, "Progress report on attainable reliability of integrated circuits for systems application," Symposium on Microelectronics and Large Systems, Spartan Books, 1965, p. 77.
3. P. Kuttner, "The rope memory—a permanent storage device," Proceedings, Fall Joint Computer Conference, American Federation of Information Processing Societies, Vol. 24, p. 45; Spartan Books, 1963.
4. A.L. Hopkins, "Guidance computer design," Space Navigation, Guidance and Control, J.E. Miller, editor, Technivision, Ltd., Maidenhead, England, 1966.
5. R.L. Alonso and J.H. Laning, "Design principles for a general control computer," Institute of Aeronautical Sciences, New York, S.M. Fairchild Publication Fund Paper FF-29, April 1960.