Computer-Controlled Steering of the Apollo Spacecraft

Frederick H. Martin* and Richard H. Battin†
Massachusetts Institute of Technology, Cambridge, Mass.

The digital guidance computer is the central control element in the Apollo control, guidance, and navigation system. Efficient operation of the guidance computer during any mission phase requires the performance of many different functions occurring at approximately the same time. Some examples are the processing of input data in the form of velocity increments, gimbal angles, system status signals, astronaut keyboard commands, and ground commands, and producing output such as steering commands, control of mode and caution lamps, digital display updating, and digital telemetry transmission. To illustrate the diversity of requirements with which this computer must cope, a specific phase of the Apollo mission is described in detail, i.e., the control of the spacecraft to accomplish a powered maneuver.

Navigation and Guidance

FOR position and velocity determination (navigation) the Apollo system includes inertial instruments capable of measuring thrust accelerations along three mutually orthogonal axes which are nonrotating. A computer performs accurate integrations and gravity calculations on a real-time basis. Incremental outputs from inertially stabilized integrating accelerometers, together with components of gravitational acceleration computed as functions of inertial position in a feedback loop, are summed to give the components of inertial velocity.

The gravity calculations may be performed in a straightforward manner. In Fig. 1, the equations of motion for a vehicle moving in a spherical gravitational field are given together with a simple computation algorithm by means of which position and velocity are obtained as a first-order difference equation calculation. Since velocity is updated by means of the average effective gravity over the interval of one time step, this technique has been termed the "average g" method.

The task of providing steering commands (guidance) for major thrusting maneuvers is a boundary value problem subject to a variety of constraints of which fuel conservation, vehicle maneuverability, and time are examples. Explicit solutions to the problem of guidance during periods of major thrusting require relatively complex calculations to be performed in flight on a time-critical basis.

Many of the major orbital transfer maneuvers can be accomplished conceptually by a single impulsive velocity change. For these cases an instantaneous velocity-to-be-gained vector based on conic orbits can be defined and the vehicle steered to null this vector. Refer to Fig. 2 and let a vector $\mathbf{v}_r$ be defined, corresponding to the present vehicle location $\mathbf{r}$, as the instantaneous velocity required to satisfy a set of stated mission objectives. The velocity difference $\mathbf{v}_g$ between $\mathbf{v}_r$ and the present vehicle velocity $\mathbf{v}$ is then the instantaneous velocity-to-be-gained.

Two convenient guidance laws will assure that all three components of $\mathbf{v}_g$ are simultaneously driven to zero: 1) we may orient the vehicle to align the thrust acceleration vector $\mathbf{a}_T$ with the direction of $\mathbf{v}_g$ or 2) since a convenient expression can be developed for $\dot{\mathbf{v}}_g$, we may direct $\mathbf{a}_T$ to cause $\dot{\mathbf{v}}_g$ to be parallel to $\mathbf{v}_g$ and oppositely directed. (If $\mathbf{a}_T$ is not sufficiently large it may not be possible to align the vector $\mathbf{v}_g$ with $\dot{\mathbf{v}}_g$, but with typical chemical rockets for which the burning time is relatively short, no difficulty has been encountered with this guidance logic.) A combination of these two techniques leads to a highly efficient steering law that compares favorably with calculus of variations optimum solutions.[1] The scalar mixing parameter γ is chosen empirically to maximize fuel economy during this maneuver. A constant value of γ is usually sufficient for a particular mission phase; however, if required, it may be allowed to vary as a function of some convenient system variable.

A functional diagram illustrating the computation of the error signal required for control purposes is shown in Fig. 3. The position, velocity, and gravitation vectors are computed as described previously. The required impulsive velocity needed to achieve mission objectives is determined as a function of the position vector and used to calculate $\mathbf{v}_g$. (Convenient formulas for many targeting problems are given by Battin.[2]) Numerical differentiation of the required velocity vector and the accelerometer outputs, using values stored from the previous sample time, provides two important ingredients of the error signals. When properly scaled, the system output is a vector rate of command whose magnitude is proportional to the small angular difference between the actual and commanded thrust acceleration vectors and whose direction defines the direction of vehicle rotation required to null this error. Near the end of the maneuver, when v_g is small, cross-product steering is terminated, the vehicle holds a constant attitude, and engine cutoff is made on the basis of the magnitude of the $\mathbf{v}_g$ vector.

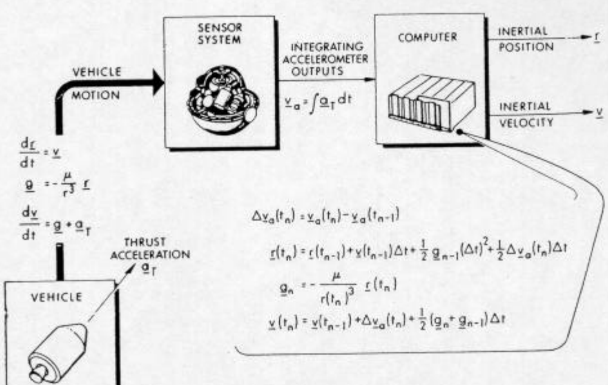

Fig. 1 Position and velocity computation.

Presented as Paper 67-603 at the AIAA Guidance, Control and Flight Conference, Huntsville, Ala., August 14–16, 1967; submitted September 1, 1967; revision received December 18, 1967.

* Assistant Director, Instrumentation Laboratory.
† Associate Director, Instrumentation Laboratory. Associate Fellow AIAA.

Inertial Measurement Unit

For control, guidance, and navigation during an accelerated maneuver, the Apollo system uses inertially stabilized instruments to measure thrust acceleration, an optical device to determine the specific orientation of this physical coordinate system, and a guidance computer to perform the numerical functions of data processing and exercising control over the vehicle orientation and rocket engines. Our primary concern in this paper is with the use of the computer in its central role as a control and processing element of the guidance system. Therefore, we shall give only a brief description of the overall system hardware.[3]

The Apollo Inertial Measurement Unit (IMU), as shown schematically in Fig. 4, is a three-degree-of-freedom gimbal system utilizing integrating gyroscopes to detect angular deviations of the stable member with respect to inertial space, and to provide, along with their servo electronics, the establishment of a nonrotating member. On this stable member, in an orthogonal triad, are three accelerometers that are single-degree-of-freedom pendulums with a digital pulse restraining system. Angle information as to the orientation of the computing coordinate frame with respect to the navigation base is derived from a two-speed resolver system mounted on each axis of the IMU. This information is visually displayed to the navigator through a ball indicating system with resolvers servo-controlled to follow the IMU resolvers. The same resolver system by means of a Coupling Data Unit (CDU) provides to the computer quantized angle increments corresponding to changes in gimbal angles. The CDU couples angle information to and from the guidance computer, performing both analog to digital and digital to analog conversion.

IMU Alignment

Since the Apollo IMU is normally not functioning during the long coasting periods, in-flight inertial system alignment against star references has been provided before the start of each accelerated phase of the mission. Since the trajectory and the thrust or drag lie fairly close to a fixed plane, the inner gimbal axis is aligned approximately perpendicular to this plane. All required large maneuvers result mostly in inner gimbal motion, so that the difficulty of approaching gimbal lock associated with large middle gimbal angles is avoided. Finally, because large roll maneuvers are frequently required, the outer gimbal axis is mounted along or near the roll axis.

A rigid structure mounted to the spacecraft, called the navigation base, provides a common mounting structure for a star alignment telescope and the base of the IMU gimbal system. Precision angle transducers on each of the axes of the telescope and on each of the axes of the IMU gimbals permit the indicated angles to be processed in the AGC to generate the star direction components in inertial system stable-member coordinates.

IMU alignment is normally performed in two stages—coarse and fine. The first step in coarse alignment is to provide the computer with a reasonably accurate knowledge of spacecraft attitude. For this purpose, the navigator sights sequentially two stars using the Scanning Telescope (SCT), which is a single-power, wide-field-of-view instrument. The star image is sensed by the navigator, who uses an optics hand controller to command the SCT prism so as to center the star on the reticle. By depressing a mark button when satisfactory tracking is achieved, the navigator signals the computer to read the SCT angles as transmitted by the optics CDU's. A second star direction at a reasonably large angle from the first is similarly measured. The navigator identifies the stars to the computer through the computer keyboard and the spacecraft attitude in three dimensions is thus determined. This orientation is maintained by an attitude hold autopilot.

In the second step of coarse alignment, the computer determines the desired IMU gimbal angles based upon its knowledge of spacecraft attitude and the guidance maneuver that will be next performed. These desired angles are sent to the IMU through the CDU to be matched by the IMU gimbal servos in response to error signals developed on the angle transducers on each gimbal axis. The IMU gimbal servos are then driven by the gyro stabilization error signals to hold the achieved orientation.

For fine alignment, two star directions are again measured by the navigator. However, this time he uses the Sextant (SXT), which is a 28-power, narrow-field-of-view instrument, in order to achieve necessary accuracy. When appropriately signalled, the computer simultaneously reads the SXT and IMU angles being transmitted via the CDU's. With these

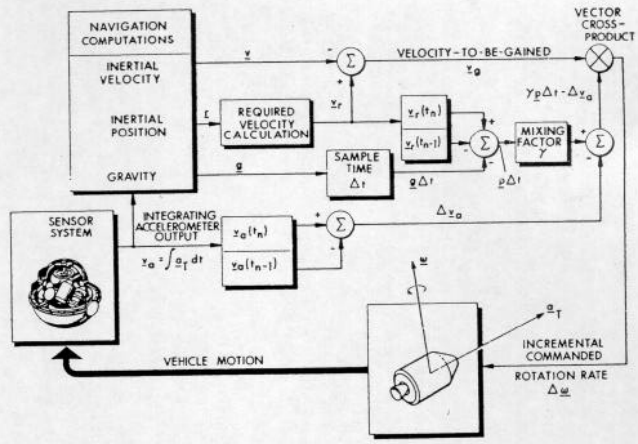

Fig. 3 Velocity-to-be-gained steering.

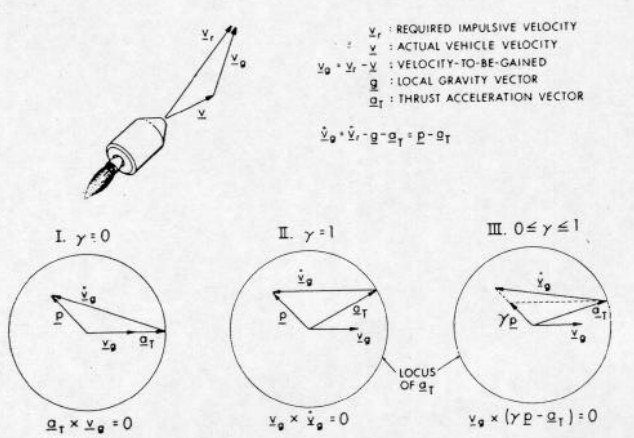

Fig. 2 Velocity-to-be-gained methods.

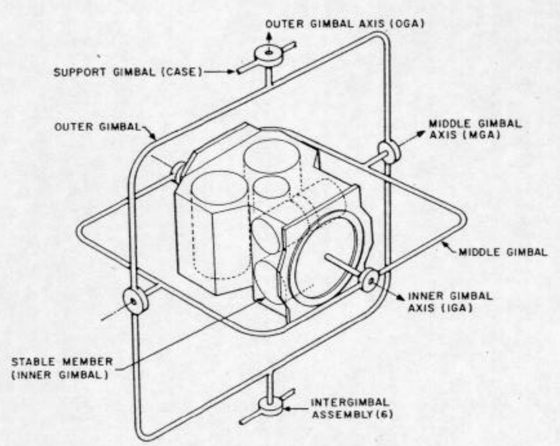

Fig. 4 IMU schematic diagram.

data the computer determines star directions in IMU stable member coordinates from which the spatial orientation of the IMU being held by gyro control can be calculated. From a knowledge of the desired attitude, the computer then determines the existing IMU attitude error and meters out the necessary number of gyro torquing pulses to precess the gyros and the IMU to correct the IMU alignment error.

Apollo Guidance Computer (AGC)

The AGC is designed to handle a relatively large and diverse set of onboard data processing and control functions. Some of the special requirements for this computer include 1) real-time solution of several problems simultaneously on a priority basis, 2) efficient two-way communication with the navigator, 3) capability of ground control through radio links, and 4) multiple signal interfaces of both a discrete and continuously variable type. The memory section has a cycle time of 12 μsec and consists of a fixed (read only) portion of 36,864 words together with an erasable portion of 2048 words. Each word in memory is 16 bits long (15 data bits and an odd parity bit). Data words are stored as signed, 14-bit words using a one's complement convention. Instruction words consist of 3 order-code and 12 address-code bits. Because of the short word length, the address portion does not always determine uniquely the address of a memory word. The ambiguity is removed by auxiliary addresses contained in so-called bank registers which are under program control.

The data words used in the AGC may be divided roughly into two classes: those used for mathematical computations and those used in the control of various subsystems. The latter class can almost always be represented with 15 bits, whereas the former requires double precision arithmetic.

The sequence-generator portion of the AGC provides the basic memory timing and the sequences of control pulses (microprograms) which constitute instructions. It also contains the priority-interrupt circuitry and a scaling network which provides various pulse frequencies used by the computer and the rest of the guidance system. A number of "involuntary" sequences, not under normal program control, may break into the normal sequence of instructions; these are triggered either by external events or by certain overflows within the AGC and are used for counter incrementing and program interruption.

Counter incrementing may take place between any two instructions. External requests for incrementing a counter are stored in a counter priority circuit. At the end of every instruction a test is made to see if any incrementing requests exist. If not, the next instruction is executed directly. If a request is present, an incrementing memory cycle is executed to read the word stored in the counter register, increment or shift it, and store the results back in the same location. This type of interrupt provides for asynchronous incremental or serial entry of information into the working erasable memory at the expense of increasing the time required for normal program steps in direct proportion to the amount of counter activity present at any given time.

Program interruption also occurs between program steps and consists of storing the contents of the program counter together with transfer of control to a location fixed for each interrupt option. Interrupting programs may not be interrupted, but interrupt requests are not lost and are processed as soon as the earlier interrupted program is resumed.

The three bits reserved for instruction codes can provide only eight possible operations. This number is extended through the use of so-called partial codes and an extend instruction. Partial codes exploit the fact that a wider variety of instructions are applicable to erasable than to fixed memory. Since erasable memory addresses are short, the instruction field of a word may be correspondingly lengthened. The extend instruction allows the instruction set of the AGC to be doubled by signalling that the following instruction code is to have an alternate interpretation.

Input-Output Interfaces

Information transfer between the AGC and other subsystems is of various kinds. In one form, entire computer words are transferred into and out of the computer to provide prelaunch and in-flight radio links maintained between the computer and ground control. For the conversion of gimbal angles and optics angles, an intermediate transformation to incremental form is made in the CDU's, the input to which are electrical resolvers. The Apollo accelerometers are incremental by nature, producing a pulse output to the computer for each unit change in velocity. Incremental transfer is also used for angle commands from the computer to the gyros and the CDU's and for thrust control and certain display functions in the spacecraft. Pulses are sent in groups or "bursts" at a fixed rate.

Discrete signals are individual or small groups of binary digits used for switch closures, mission phase changes, jet firings, display initiations, and many other similar controlled events. Discrete inputs to the computer may be noninterrupting signals that can be interrogated by input-output channel instructions. Interrupting inputs cause the transfer of control to a particular address. When the appropriate action has been taken, the original program is then resumed at the point where it was interrupted. Programs operated during interrupt never exceed a few milliseconds in running time.

Incremental output transmission is made by placing a number in an output counter register which is then decremented at a fixed rate of 3200 pps. Output pulses are generated concurrently until the number reaches zero. Pulse bursts are used in this way for gyro torquing.

In another form of output transmission the number in the register controls the switching of a set of precision resistors in an operational amplifier network such that the amplifier output is proportional to that number. These analog signals are available as voltages for driving such equipment as attitude displays and steering gimbals for the rocket engine.

Display and Keyboard

The display and keyboard (DSKY), shown in Fig. 5, serves as the communication medium between the computer and the navigator. The principal part of the display is the set of three registers, each containing five decimal digits so that an AGC word of 15 bits can be displayed in one register by five octal digits. Three registers are used because of the frequent need to display the three components of a vector. Digits are entered into the computer from a keyboard of 19 push buttons including the 10 decimal digits, plus and minus, and a number of auxiliary items. Each key depression causes a computer interrupt that initiates a request to the computer's executive program to process the character at the earliest opportunity.

The display also has digit displays labeled verb, noun, and program, and the keyboard has keys labeled verb, noun, enter, and clear as well as three others. Commands and requests are made in sentences, each with an object and an action, such as "display velocity" or "load desired angle." The DSKY has a vocabulary of 100 actions, or verbs and 100 objects, or nouns. When the operator depresses the verb key followed by two decimal digit keys, the desired verb is entered into the computer, where it is stored and also sent back to the DSKY to be displayed in the verb lights. The desired noun is entered using the noun key, and it is displayed in the noun lights. Then depression of the enter key initiates action on the command. When the computer requests action from the operator, a verb and a noun are displayed in the lights and a relay is closed which causes the verb and noun lights to flash on and off in order to attract the operator's attention.

Utility Programs

Most of the AGC programs relevant to guidance and navigation are written in a pseudocode notation for economy of storage. This notation is encoded and stored in the AGC as a list of data words. An "interpreter" program translates this list into a sequence of subroutine linkages. A pseudocode program consists of a string of operators and addresses with two 7-bit operators stored in one AGC word. Thus, the instruction set is expanded into a comprehensive mathematical language, which includes matrix and vector operations, using numbers of 28 bits and sign.

All AGC programs operate under control of the "executive" routine except those which are executed in the interrupt mode. Executive-controlled programs are called "jobs" as distinct from so-called "tasks," which are controlled by the "waitlist" routine and completed during interrupt time. The executive routine controls priority of jobs and permits time-sharing of erasable storage. A job usually is initiated during interrupt by a task program or a keyboard program. It is specified by its starting address and another number which gives it a priority ranking. As the job runs, it periodically checks to see if another job of higher priority is waiting to be executed. If so, control is transferred away until the first job again becomes the one with highest priority. No more than 20 msec are permitted between these periodic priority checks. When a job is awaiting the occurrence of a certain external event, it may be suspended or "put to sleep." The job's temporary storage is left intact through the period of inactivity. When the anticipated event occurs, the job is "awakened" by transfer of control to an address that may be different from its starting address. If a job of higher priority is in progress, the awakening will be postponed. When a job is finished, it transfers control to a terminating sequence which releases its temporary storage to be used by another job. Approximately ten jobs may be scheduled for execution or be in partial stages of completion at any one time.

The waitlist routine provides timing control for other program sections. Waitlist tasks are run in the interrupt mode and are restricted to a few milliseconds duration. If an interrupt program were to be longer, it could cause an excessive delay in other interrupts waiting to be serviced since one interrupt program inhibits all others until it calls for resumption of the main program. The waitlist program derives its timing from one of the counter registers in the AGC. The counter priority state that controls this counter is driven by a periodic pulse train from the computer's clock and scalar such that it is incremented every 10 msec. When the counter overflows, the interrupt occurs which calls the waitlist program. Before the interrupting program resumes the normal program, it presets the counter so as to overflow after a desired number of 10-msec periods up to a limit of 12,000 for a maximum delay of 2 min. If the waitlist is required to initiate a lengthy computation, the task will make an executive routine call, so that the computation is performed as a job during noninterrupted time.

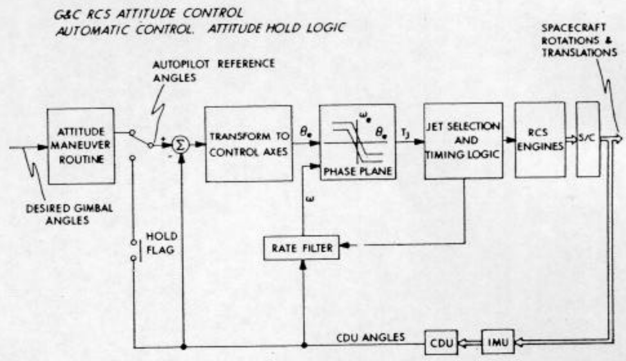

Fig. 6 Coasting flight autopilot.

Digital Autopilot Control

Two of the flight control problems associated with major powered maneuvers of the Apollo spacecraft are discussed in this section. The first is control of the attitude of the vehicle while coasting in free space and the second is powered flight control of the attitude and flightpath of the vehicle while thrusting.

Coasting Flight Control

During periods of coasting flight when there are no significant forces acting on the spacecraft, control over the vehicle involves attitude control only, with a reaction control system (RCS). The primary requirement for the RCS prior to thrusting is to maneuver the vehicle to its proper thrusting orientation. The RCS is also used for vernier translation control. The rocket pulses can be as short as 14 msec. Sixteen of these engines are mounted on the sides of the service module in quadruple sets at four locations. They are normally fired in pairs to produce control couples. A variety of operational modes can be selected by the crew including attitude-hold and rate-command modes in addition to direct actuation of the reaction jets by the pilot through a three-axis hand controller.

In the primary mode, the AGC operates the jet solenoid valve drivers directly based on attitude error and attitude error rate information. As illustrated in Fig. 6, the attitude reference is obtained directly from the CDU's whereas attitude rate must be computed in the AGC. As a function of attitude error and attitude error rate, nonlinear switching functions in the computer are employed to generate the necessary firing time intervals. A jet-selection logic is then used to select the individual jets to be fired. Simultaneously, attitude error signals are transmitted via the CDU digital-to-analog converters to drive the ball attitude indicator.

Central to the operation of the RCS digital autopilot is the angular-rate estimator. Derivation of the body rates of the spacecraft from the gimbal angles is complicated by the effects of CDU angle quantization and the effects of body bending modes of the vehicle. The accuracy of estimating body rates is effected by the desire for fuel economy and the system capability of holding attitude errors within a selected deadband.

Automatic attitude maneuvers are implemented by exactly the same logic as that used in attitude hold. However, they are based on a specific rate command and moving reference

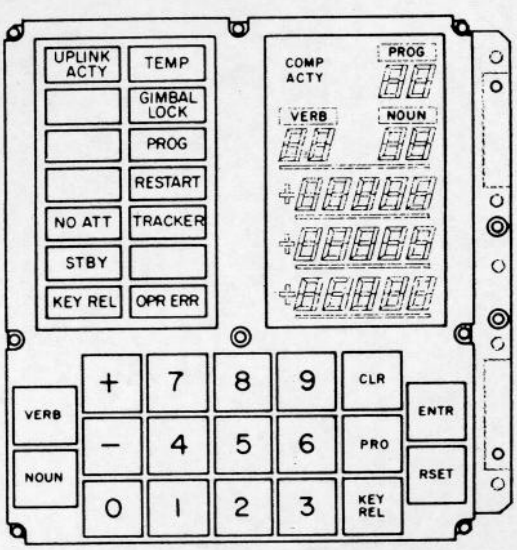

Fig. 5 Display and keyboard.

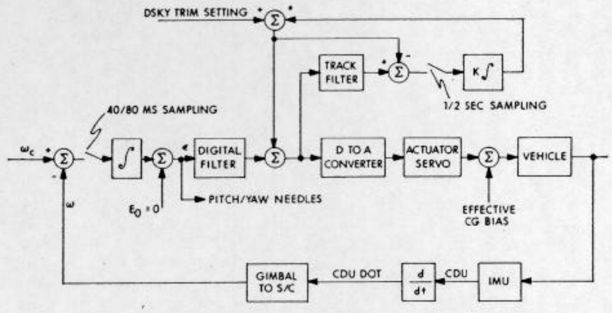

Fig. 7 Powered flight autopilot.

attitude. The generation of the moving desired attitude is performed in two stages. First, on the basis of current attitude and desired attitude, as supplied by either the crew or an AGC program, the attitude maneuvering routine determines the axis about which a single rotation will achieve the desired reorientation. The routine determines if the calculated maneuver will drive the IMU through gimbal lock. If so, the rotation axis is readjusted so that the IMU will skim the gimbal-lock zone when the x-axis of the spacecraft is properly pointed. A final roll is usually necessary to complete the maneuver. In either case, the result of the calculation is a direction in space about which to rotate the vehicle together with an angle of rotation. A transformation matrix is then computed representing a rotation about the computed vector through an incremental angle equal to the selected maneuver rate times the iteration cycle period.

The second stage in the calculation is to develop a transformation matrix equal to the product of the matrix that relates initial spacecraft and stable member axes and the previously determined incremental rotation matrix. This transformation is updated once per computation cycle. Desired CDU angles are also developed at each iteration cycle, compared with actual CDU angles, and the difference transformed into attitude errors. In order to minimize discontinuity, the desired CDU angles are interpolated between computation cycles.

Powered Flight Control

The powered flight control system orients the vehicle thrust acceleration vector $\mathbf{a}_T$ in response to commands generated by the guidance system. Since $\mathbf{a}_T$ is, on the average, oriented in the vicinity of the longitudinal axis of the vehicle, the powered flight control problem is also primarily one of attitude control. Immediately prior to initiation of a major burn using the Apollo Service Propulsion System (SPS) the RCS holds the initial thrusting attitude in a narrow deadband. The preselected trim, which points the engine bell through the vehicle center of gravity, has been commanded and confirmed by the crew and the crew has accepted the thrusting parameters including time of ignition. At a prescribed time before ignition, the navigator initiates ullage. At ignition time the RCS digital autopilot releases control of the vehicle to a thrust vector control (TVC) autopilot which then controls engine excursions. Ullage is terminated by the major program when successful thrust buildup is sensed.

During the thrust period the cross-product steer law commands a vehicle rate proportional to the angular separation of $\mathbf{v}_g$ and $\mathbf{a}_T$ (or $-\dot{\mathbf{v}}_g$). The autopilot responds in pitch, yaw, and roll to three independent control signals generated in accordance with the attitude errors. The implementation of roll control is achieved by reaction jets whose firing is controlled by phase-plane and jet-selection logic similar in concept to the RCS autopilot. The regulation of the outer gimbal angle to within $\pm 5°$ of a preset value is sufficient to prevent adverse cross-coupling between the pitch and yaw channels.

The TVC pitch/yaw autopilot programs must fulfill the primary requirement of vehicle stabilization in conjunction with the external guidance loop to provide satisfactorily small velocity pointing errors at thrust cutoff. The autopilot programs must also limit excursions in vehicle attitude and thrust vector in such a way as to minimize propellant usage, gimbal-servo clutch wear, and crew malaise. Figure 7 shows a schematic block diagram of one channel (pitch and yaw) which can serve as a basis for describing the autopilot. For simplification it is assumed that the rate command ω_c has already been transformed into body coordinates. The role of the AGC in this autopilot is to perform a dynamic filtering operation on the sampled attitude error ϵ as required to generate a suitable sampled command to the gimbal-servo. A digital-to-analog converter changes the gimbal-servo-command to an analog voltage, which is held for each sample period.

One of the major obstacles to be overcome by the digital autopilot is the effect of an initial error in the alignment of the thrust vector through the vehicle center of gravity. The misalignment will cause a nonzero attitude error because the digital filter has a finite gain at zero frequency. It is important to note that although this steady-state error produces a nonzero display to the navigator on the attitude error needles, it does not generally imply a steady-state pointing error in the velocity of the guided vehicle since the external guidance loop will act to readjust the attitude command signal to the autopilot to compensate for this attitude error. However, the pointing error could be appreciable during short thrusting periods when there is insufficient time for the external guidance loop to react to the thrust vector misalignment before thrust cutoff. By the addition of a tracking filter, the position of the center of gravity is estimated every $\frac{1}{2}$ sec and a bias signal fed back to provide actuator trimming. The digital filter, whose primary responsibility is the shaping of signals in order to avoid the addition of energy into the slosh and bending modes, is thus relieved of maintaining an actuator offset signal. After the transient period the attitude errors and needle display will settle on the null positions. At 4 sec prior to engine cutoff the TVC autopilot maintains constant attitude, damping out the rates through most of the tail-off transient. At 2.5 sec following the engine-off command, TVC operation ceases and the RCS autopilot is initiated in the wide deadband mode.

Computer Operations for a Powered Maneuver

The Apollo mission program is divided into functional sub-programs by designating major modes by program numbers corresponding to the various onboard computational capabilities. Generally speaking, the required sequences for the flight to the moon encompass prelaunch, boost, navigation, targetting, powered maneuvers, stable member alignments, and entry. The sub-programs within these eight categories are identified, respectively, by the numbered intervals: P01-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, and P60-P67. The navigator selects, by DSKY entry, the program needed to perform the particular mission phase at hand. In addition to specific programs, a variety of special algorithms are available, again at navigator selection, which display useful information concerning the state of the vehicle in space. For example, the apogee and perigee or latitude, longitude, and altitude can be computed.

Computer Program Description

The program P40 is selected by a keyboard entry at least 5 min before the estimated time of ignition whenever the service module propulsion system is used to effect a change of orbit. The precise direction of thrust initiation and the method of steering depend on previously calculated target parameters. The targeting problem of arriving at a given point in space at a specified time is solved in P34. This program is entered prior to P40 and allows the ignition time and the transfer time interval to the target point to be loaded as DSKY inputs. The required impulsive velocity change, resulting perigee, and the expected middle gimbal angle are displayed so that the crew may evaluate, in advance, the propellant usage and alignment adequacy for the maneuver as well as the final orbit safe perigee margin. Upon leaving P34, the program stores away, for later use, the crew approved times of ignition and transfer and the computed "offset target vector" to be used in the conic calculations for steering. The transition to P40 by DSKY input occurs at a time in advance of ignition sufficient to accomplish the busy crew checklist that precedes any major thrusting maneuver.

The primary purpose of program P40 is to control the guidance, navigation, and control system during countdown, ignition, thrusting, and thrust termination of an SPS maneuver. Before P40 can be selected, certain data must be provided for use by the control autopilots. These data are rate, deadband, and jet usage, as well as SPS engine gimbal turn estimates, and are loaded in a prescribed way as DSKY inputs. The CSM weight, inertias, and engine torque, quantities also required by the autopilots, are computed onboard and tracked during the thrusting maneuver. Their values are always available for inspection and readjustment.

After the vehicle and engine parameters have been satisfactorily set, the AGC then computes the initial thrust direction and the initial value of the velocity-to-by-gained vector $\mathbf{v}_g$. The three components of $\mathbf{v}_g$ are displayed in local vertical coordinates to the navigator, who has the option to abort the program if he detects any gross errors in the computation. From the initial thrust direction and engine bell trim angles, the AGC computes the preferred IMU orientation with the x-axis of the stable member in the direction of the computed attitude for thrust initiation. From this the gimbal angles are computed and displayed which would result if the present IMU orientation were held and the vehicle maneuvered to the preferred orientation, i.e., wings level and heads up as seen from the couch looking out along the x-axis. If the displayed middle gimbal angle exceeds 45°, the navigator will select an IMU realignment program. Following the realignment procedures, the navigator may again select program P40.

Assuming that the IMU is properly aligned, the AGC will extrapolate the position and velocity to a time 30 sec prior to the predicted time of ignition and then select a routine to cause the vehicle to be maneuvered to the desired thrust attitude under computer control. After completion of the maneuver, the AGC requests the navigator to perform the engine gimbal drive test as a safety measure which he does via a keyboard entry. Following the test the gimbal is trimmed.

The computer now checks the time remaining before engine ignition. If it is less than 45 sec, an alarm is flashed to the navigator who then has the option of aborting the program. On the other hand, with sufficient time to go, the AGC will display the time-to-ignition in minutes and seconds, v_g during the thrusting maneuver, and the measured change in velocity to the nearest 0.1 fps. This last quantity will, of course, be zero until ullage is started.

P40 next initiates a call to the waitlist program to begin the average g integration calculation at 30 sec before ignition. The navigator readies the service propulsion system by setting a main panel switch. Upon receipt of the $+\ x$ translation hand controller discrete, the computer commands attitude jets on to begin the ullage maneuver. The navigator monitors the velocity change on the DSKY to insure sufficient ullage is occurring. At 5 sec prior to ignition, the computer signals the navigator to enable the main engine on. This is the final opportunity to abort the propulsion maneuver.

If the decision is made to continue, the computer commands the engine on when the time-to-ignition reaches zero and immediately changes the autopilot mode from coasting flight to powered flight. Ullage is terminated as soon as the velocity change monitor detects that the main engine has, indeed, ignited. The DSKY register, which has been displaying time-to-ignition, is changed to display predicted time-to-go to engine cutoff. During the burn the navigator monitors the DSKY registers to insure that the time-to-go and v_g are actually decreasing and monitors the attitude ball indicator to insure that the attitude error and attitude rates are within acceptable tolerances. When the computer determines that the targeting conditions have been met, an engine-off signal will be sent to the SPS and attitude control again returned to the coasting flight autopilot. At the end of the burn, the DSKY will display the residual values of $\mathbf{v}_g$ in spacecraft coordinates. The navigator now has the option of manually trimming these velocity components by exercising the translation and rotation hand controllers. In either case, when the maneuver is complete, the AGC will determine the new spacecraft orbital parameters, display them to the navigator, and request the navigator to select either the idling program or some other major mission program.

Computer Program Mechanization

Any of the mission programs may be viewed as a chain of computational routines linked together by logical coding, which sets and resets appropriate bits for flags, controls timing and sequencing, and produces DSKY displays. The numbered mission programs are initiated as jobs of specified priority. The job allows all types of interrupts to occur and reestablishes itself after the interrupt period. In fact, during its execution, the job itself must periodically check to determine if a higher-priority job is waiting.

The combination of several jobs with varying priorities, controlled in time by waitlist tasks and DSKY inputs, gives the effect of many computer activities being carried on almost simultaneously. In addition to this structure, other interrupt activities can proceed in the background. The Time 4 counter interrupt routine (T4RUPT) is initiated whenever the Time 4 counter overflows. Normally this counter is set to overflow every 120 msec. Every time this occurs, the T4RUPT routine is initiated and one or more of the following functions is performed: 1) sampling and verification of the IMU mode of operation including turn-on, 2) monitoring the telemetry rates, 3) sampling of malfunction indications from the IMU, and 4) control of the relays of the DSKY for display of information, for commanding IMU and other spacecraft, modes, and for control of indicator panel illumination. The Time 4 counter monitors the entire system in search of malfunctions. The Time 5 and Time 6 interrupts usually control digital autopilot timing and jet firing.

During a thrust maneuver, the computer may spend 20 to 25% of its time in the interrupt mode. For example, the TVC autopilot is cycled every 40 msec and remains in interrupt for 8 msec in each cycle. Additional interrupts which consume time are KEYRUPTS from the depression of DSKY buttons, MARKRUPTS from optics usage, UPRUPTS and DOWNRUPTS signifying up and down telemetry activity, and interrupts generated by the hand controllers.

The preceding description is meant to serve as an aid in understanding how the procedures of P40 are implemented. For more detailed information, consider the illustration in Fig. 8, where the various jobs and tasks are shown in graphical form. Program P40 is shown as a job of priority 13. During this job, the initial thrust direction and initial value of $\mathbf{v}_g$ are calculated and displayed in local vertical coordinates. The flashing display calls for navigator approval and P40 is "put to sleep." Upon awakening, the wings-level attitude is computed and again displayed asking for DSKY action. When the program is signalled to proceed using Verb 33 Enter, P40 is suspended while the RCS autopilot directs the vehicle to the thrusting attitude by means of a new job of priority 23. The autopilot cycles at 100 msec timed by the Time 5 clock. When the maneuver is complete, the naviga-

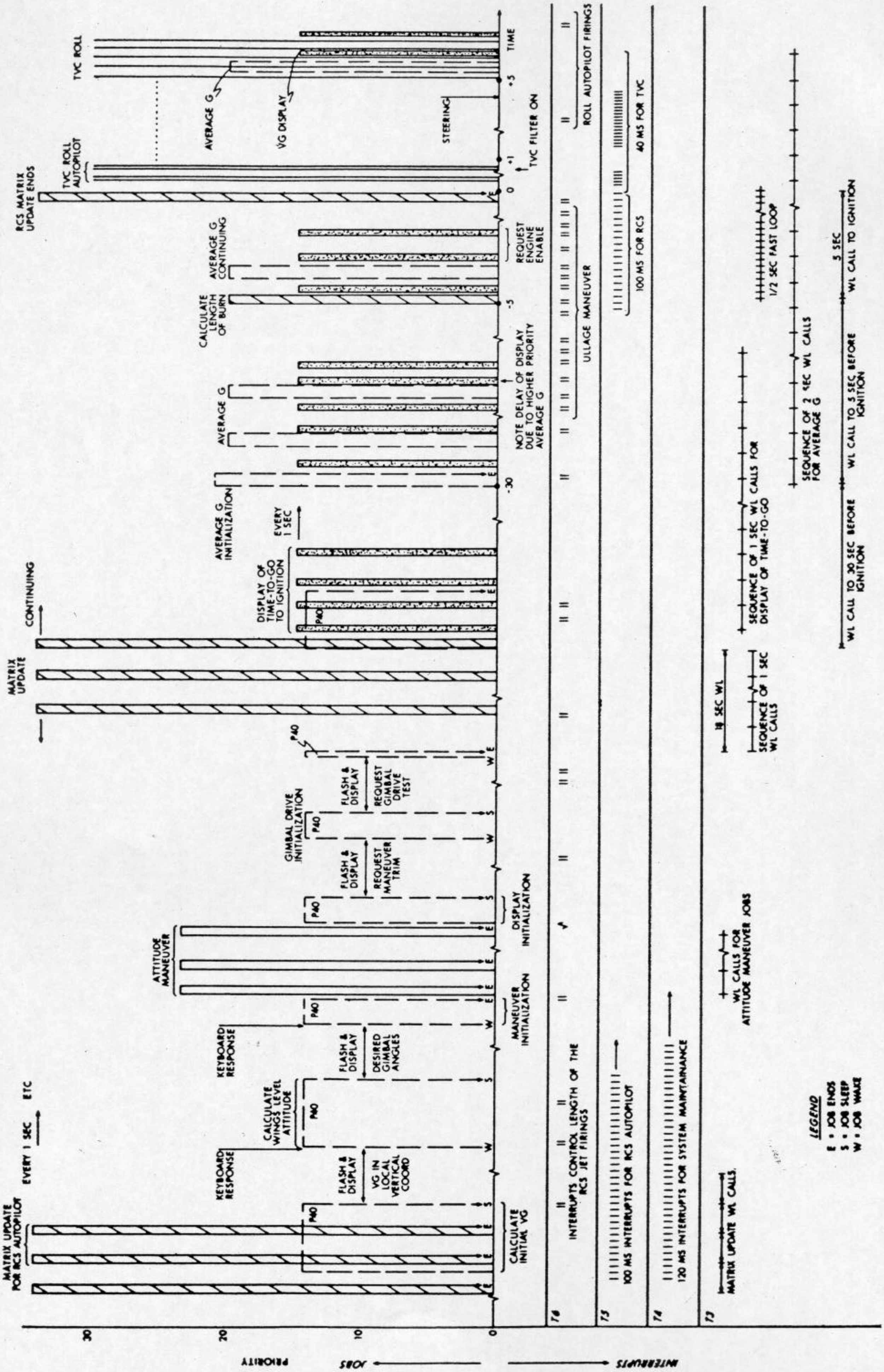

Fig. 8 Graphical description of program P40.

tor signals his approval and P40 again becomes the active program. The next step is a request by the computer to perform the engine gimbal drive test. If this option is selected by the navigator, a waitlist call is placed for an immediate interrupt that will then occur within 10 msec. The test proceeds as a sequence of waitlist tasks called once each second. In the meantine, P40 independently uses the waitlist as a delaying action to provide sufficient time for the test to be completed.

As the time of ignition approaches, P40 establishes a 1-sec waitlist loop to count down to the firing time. This task, in turn, requests a job of priority 14 to display the time-to-ignition. P40 then extrapolates the position and velocity vectors to the time 30 sec prior to ignition and places a waitlist call for the average g integration and steering equation computation to begin at that time. During the extrapolation, which is performed with priority 13, time-to-go display, having priority 14, is periodically updated. P40 now ends as a job leaving the display function as the only computer activity except, of course, the interrupt controlled RCS autopilot.

At 30 sec before ignition the average g integration is initialized by a job of priority 21 and a waitlist task is established to occur 5 sec before ignition. In addition, a 2-sec waitlist loop is begun to read the accelerometers. Each waitlist task in the cycle requests a job to perform the average g integration with priority 20. The reading of the accelerometers, together with the integration, continues every 2 sec throughout the maneuver until the program is terminated and a new program number selected.

At 5 sec to go, a waitlist task is requested to occur at ignition. In addition, a fast reading accelerometer loop, based on $\frac{1}{2}$-sec waitlist calls, is initiated to monitor the main engine thrust buildup. The length of burn is now calculated by a job of priority 20. Although this has the same priority as average g integration, no problem results since sufficient time exists for both jobs to be completed. During the final 5-sec period, the time-to-go display job of priority 14 changes the display in order to flash a request for the navigator to enable the engine. A positive response brings the time-to-go back to the DSKY and all is prepared for ignition.

At ignition time, the engine bit is set on and control of the vehicle is transferred from the RCS to the TVC autopilot. Thereafter and until shut down, the TVC autopilot will operate repetitively on 40-msec Time 5 interrupts. For most of the burn, until the time-to-go to shutdown reaches -4 sec, the system remains in this guided mode. The timed events immediately before and after shutdown are not different conceptually from those described previously and will not be elaborated on here.

References

[1] Martin, F. H., "Closed-Loop Near-Optimum Steering for a Class of Space Missions," *AIAA Journal*, Vol. 4, No. 11, Nov. 1966, pp. 1920–1927.

[2] Battin, R. H., *Astronautical Guidance*, McGraw-Hill, New York, 1964.

[3] Miller, J. E., "Space Navigation Guidance and Control," 1966, Technivision Ltd., Maidenhead, England.