ARTICLE NO. 78-1239R

G-80-001

Modular Strapdown Guidance Unit with Embedded Microprocessors

30010
30019
40006

Jerold P. Gilmore*
The Charles Stark Draper Laboratory, Inc., Cambridge, Mass.

The Low-Cost Inertial Guidance System (LCIGS) is a modular strapdown implementation of attitude (gyro) and velocity (accelerometer) axes which permits the interchangeable use of different manufacturer's instruments without affecting the system's electronic or mechanical interfaces or processing software. This design flexibility is made possible by the use of microprocessors for processing and control. The microprocessors are embedded in each module and five are used: one per accelerometer triad, one each per gyro module, and one in the service module. The processors effect on-line digital torquing control of the gyros, active instrument error model compensation, including modeling for temperature sensitivity effects, temperature control, self-testing, etc. Adaptation of processing and calibration algorithms to accommodate for instrument changes or sensed environmental variations is achieved through the use of an alterable read-only data base that may be updated by the LCIGS support equipment as required at calibrations or upon an instrument replacement. This data base is accessed by the microprocessors and used to compute coefficient corrections for the processing algorithms. The system architecture is presented and the microprocessor software partitioning and functions are described.

Introduction

THIS paper describes a strapdown guidance unit which uses embedded microprocessors for instrument control and data processing. It is designed for use in tactical air-to-surface standoff missiles, but is also applicable to a broad spectrum of avionic implementations.

The name chosen for this development is the Low-Cost Inertial Guidance System (LCIGS). This name underscores one of the program's primary objectives, that is, development of a strapdown inertial reference unit that can be competitively produced at a production unit cost (PUC) target goal of $10,000 in FY1976 dollars in quantities of 2000 systems per buy. To address this goal, the design is configured as a modular implementation of sensing axes, attitude (gyro), and velocity (accelerometers). The axes are configured about mature available single-degree-of-freedom (SDF) instruments. Each sensing axis design is such that different SDF manufacturer's instruments, of the same generic class, can be interchangeably used without affecting the electronic or mechanical interface or processing structure. Thus, if a vendor's proprietary instrument meets a basic normalized specification, it may be used interchangeably in the system by all LCIGS manufacturers. The intent is that all manufacturers will be able to compete for production at every procurement cycle, including sparing.

The modularity is extended within each sensing axis so that the electronics is also functionally partitioned into replaceable electronic modules with corresponding interface specifications. This partitioning will permit graceful modernization and technology growth through the phase-in of new components or modules that meet the common interface design requirements. Changes will not impact the system configuration. Thus, performance or reliability improvements, via instrument or electronics changes, can be readily incorporated and obsolescent parts problems circumvented. The standardized modular interchangeability feature minimizes system life cycle costs (LCC). This design flexibility is in large measure made possible by the use of microprocessors, embedded within the modules, for data processing and control.

A block diagram of the system is shown in Fig. 1. Five microprocessors are used: one each per gyro module, one per accelerometer triad (velocity reference module), and one in the service module. The respective microprocessors perform instrument control and error compensation, data processing, and formatting functions. Adaptation of the control and compensation algorithms to the specific instruments is achieved by using coefficient variables that are stored in the service module processor's data base, an electrically alterable read-only memory (EAROM). The EAROM is loaded using the test support equipment (PSE) and may be altered as required at system calibrations or when instruments are replaced. (Subsequent sections describe the processor architecture and software processing features.)

LCC and rapid operational deployment goals were major design drivers; a 10-yr inventory life with extended storage periods was specified. Testing requirements have been minimized and concepts that permit rapid checkout and calibration have been developed. Current ownership cost estimates correspond to 10% of the PUC, and 30-min calibration capability appears to be realizable. The calibration and checkout accommodate large initial bias errors while permitting relatively loose tolerances on test fixturization. The PSE, which includes a minicomputer, effects the calibration processing and also commands the embedded microprocessors to perform self-test diagnostics automatically.

Design Features

For the baseline design, the LCIGS configuration has been scoped to be compatible with use in the guidance adapter section of the USAF modular glide bomb/GBU-15 configuration. Angular rate and acceleration dynamic range design requirements of ± 150 deg/s and $\pm 10\ g$, primarily for the preservation of the in-flight alignment while attached to the launch aircraft (i.e., across aircraft evasionary maneuvers), were also defined. Data requirements $\Delta\theta$ and ΔV

Presented as Paper 78-1239 at the AIAA Guidance and Control Conference, Palo Alto, Calif., Aug. 7-9, 1978; submitted Oct. 31, 1978; revision received June 11, 1979.

Index categories: LV/M Guidance; Sensor Systems; Computer Technology.

*Division Leader, Inertial Subsystem Division.

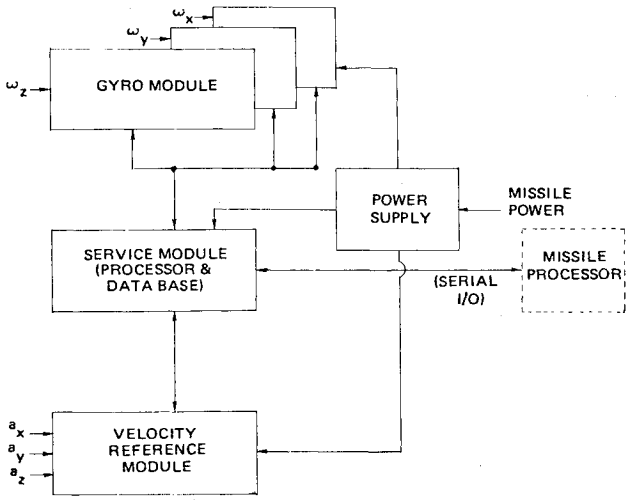

Fig. 1 LCIGS block diagram.

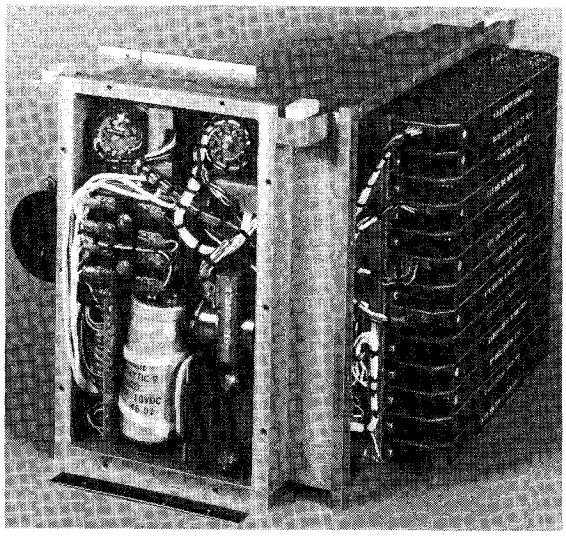

Fig. 2a LCIGS system (with covers removed).

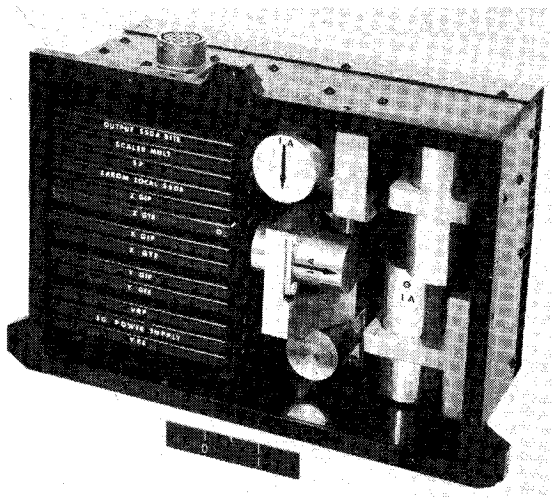

Fig. 2b Hybridized model.

of ~3 s and 0.03 ft/s per pulse, respectively, with an interface iteration rate of 100/s were selected to be compatible with the Global Positioning System and radiometric correlator aided-navigation and autopilot operations. A serial interface with a two-way communication capability is provided.

The GBU-15 physical space envelope limitations necessitated the L-shaped brassboard LCIGS assembly shown in Fig. 2a. The frame utilizes a low-cost aluminum casting, which requires a minimum of final machining.

The electronic cards are printed circuit assemblies that incorporate integral heat sinks. The instruments are mounted in normalization assemblies. Consistent with the previously stated program intent, as many as six gyros (Timex IG10, Honeywell GG1111, etc.) and two accelerometers (Sundstrand QA 1200 and Systron 4851) are candidates that could be used interchangeably in this system. The system weight is 23.25 lb; the volume is 450 in.^{3}. An additional 56 in.^{3} were required for the cooling fins and blower provisions. The March 1978 PUC for this assembly corresponds to $10,969 in FY76 dollars.[1]

An alternate design, based on the use of hybrid electronic packaging, is shown in Fig. 2b. It preserved all the LCIGS design characteristics and was investigated to provide a basis for those applications in which space and weight are critical. The corresponding volume and weight are 166 in.^{3} and 8.7 lb, respectively. If cooling provisions are needed, an additional 36 in.^{3} and 1.1 lb would be required. The corresponding PUC for the hybridized version is estimated at $17,000.

System Architecture

The detailed system mechanization diagram shown in Fig. 3 depicts the LCIGS distributed processing architecture. A microprocessor operates through interfacing conversion electronics with the instruments in the respective velocity reference module (VRM) and the gyro modules (GM's) and communicates with the system service processor (SP) under its executive control. In the VRM, the translational motion sensed by the accelerometers, whose outputs are analog voltages proportional to specific force, is digitized by voltage-to-frequency converters (V/F's). The resultant digitized incremental velocity data (ΔV) is collected, accumulated, compensated for errors (bias and scale factor), and formatted by the VRM processor.

In the GM, the processor operates in conjunction with the gyro torquing electronics to effect closed-loop digital pulse-width torquing of the gyro. The processing algorithm operates on the basis of the digitized gyro signal generator's output and commands the torquing electronics to issue variable width pulses in a ternary or binary mode. The resultant torque commands correspond to an integer number of angular rotational measurement increments ($\Delta\theta$). The $\Delta\theta$ data across data transmission updates is accumulated in the processor, where it is compensated for scale factor (SF) and bias errors and formatted for transmission.

The processed $\Delta\theta$ and ΔV data is synchronously transmitted across a two-way serial internal bus structure to the SP where the data is collected and formatted. Data transmission is under SP polling and gating control.

The SP communicates with the instrument processors to initialize and synchronize the system upon turn-ons or system resets. It also provides updates of the SF and bias coefficients that are used in the instrument processor compensation routines. These updates are determined by the SP by accessing the thermal sensitivity data stored in its alterable data base (EAROM) and computing coefficient corrections based on the sensed temperatures. The EAROM also contains variable data, which permits the interchangeable use of different instruments.

The SP communicates with the missile system processor across a two-way serial data interface (500 kHz bit rate). The output I/O structure is shown in Fig. 4. A variable length block transfer protocol is used in which "Flag" (a pulse) and "LCIGS Sending" (LCBLOK, a level) signal that the LCIGS output must have priority servicing. The missile processor must respond after each "Flag" by clocking out two LCIGS bytes plus parity with a CTS signal. Block transmissions are sent to LCIGS by signaling with the "Request to Send" and

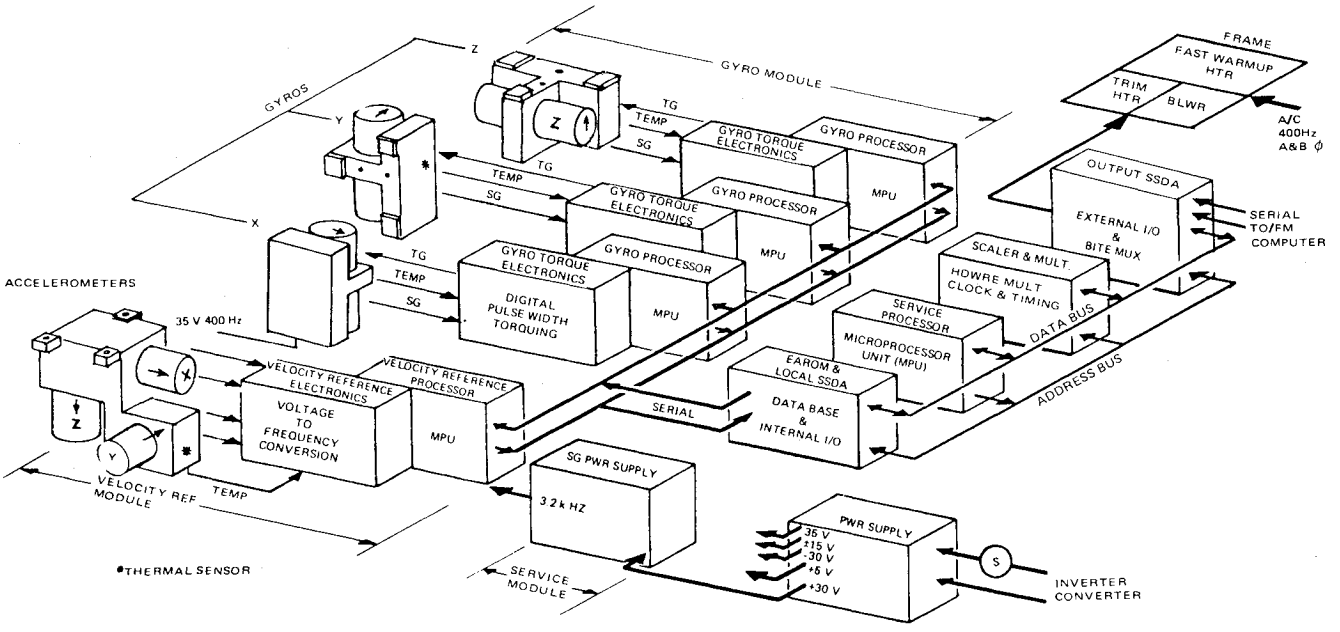

Fig. 3 System mechanization.

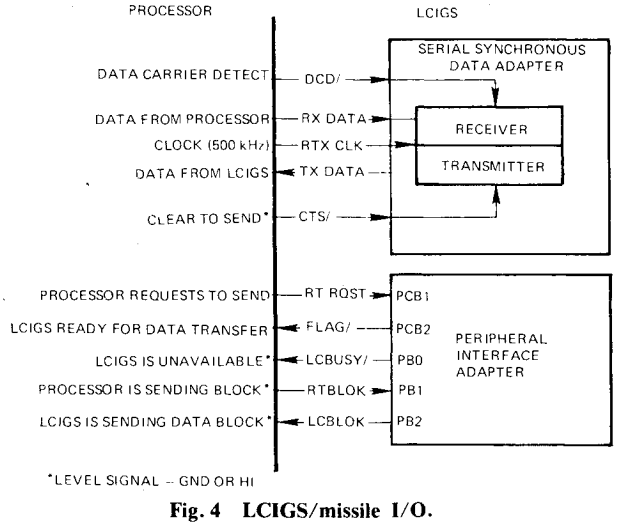

Fig. 4 LCIGS/missile I/O.

the "Sending Data Block" levels. Two bytes may be clocked into LCIGS with the Data Carrier Detect (DCD) signal after each LCIGS "Flag."

Output transmissions from LCIGS include the preprocessed and formatted sensor data, $\Delta\theta_{x,y,z}$ and $\Delta V_{x,y,z}$, and other specific LCIGS parameters (e.g., gyro g-sensitive drifts, and/or output-axis coupling, anisoinertia coefficients, etc.) as required. Status information, such as LCIGS "Ready" or that a reset has occurred, is also sent when applicable. Missile processor messages may request specific LCIGS data (e.g., memory status) and it can initiate reset. The PSE functions through this I/O, augmented by additional functions, so that it can alter the EAROM load and effect diagnostic tests, etc.

A summary of the various processing tasks that are allocated to each of the different embedded microprocessors is shown in Fig. 5. When compared to a traditional customized digital logic design, this processing implementation has enhanced automated system testing and eliminated the need for special-purpose BITE. Further, microprocessor costs are competitive with traditional designs, and high-volume industrial usage is assured.[2] High-volume cost reduction trends are already yielding PUC reductions.

Processor Structure

All of the processors are configured with the family of Motorola M6800 LSI microcomputer components. High-volume industrial usage, second source availability, and military environment qualification considerations were major factors in the selection of the M6800 family. Additionally, its system-oriented architectural features were compatible with the hardware interfacing inferred in the mechanization shown in Fig. 3 and the software tasks structure of Fig. 5.[3]

Although not unique to the M6800 family, the interface functions provided by the peripheral interface adapter (PIA) and the serial synchronous data adapter (SSDA) components facilitated the LCIGS I/O architectural design. These peripherals, when used, simply correspond to memory locations on the MPU address and data bus; they are progammable from the bus and their real-time status is accessed through the bus. These provisions simplified interfacing and eliminated the need for I/O instructions. Another highly desirable feature is that the 6800 family buffers are compatible with standard TTL load driving and only a +5 V power supply was required.

Figure 6 depicts the processor organization used in the instrument modules. The specific interfacing shown is used in the GM. The VRM processor is identical; interfacing differences are effected by the individual software utilization choices and system interconnection provisions.

The MPU, a M6800CL, functions as a bidirectional bus-oriented general-purpose processor. Eight-bit parallel processing is implemented and a 16-bit address bus is available (65k bytes of addressing). The processor is capable of directly interfacing with eight peripheral parts and one TTL load on the bus at a 1-MHz cycle clock rate. As shown, seven peripherals (including 2 RAM and 3 PROM components) are used in the instrument processor. The SP is buffered since 12 peripheral devices are interfaced with its bus. The MPU contains: an arithmetic logic unit (ALU), two 8-bit accumulators, a 16-bit index register, a 2-byte stack pointer, and a 16-bit program counter.

The processor minimum execution time is 2 μs. The instruction set (72) provides a variable length capability that aids in optimizing memory and processing time utilization. Three interrupts are available: reset, nonmaskable (NMI), and peripheral interrupt request (IRQ). The reset, which is software vectored, is used for LCIGS initialization at power

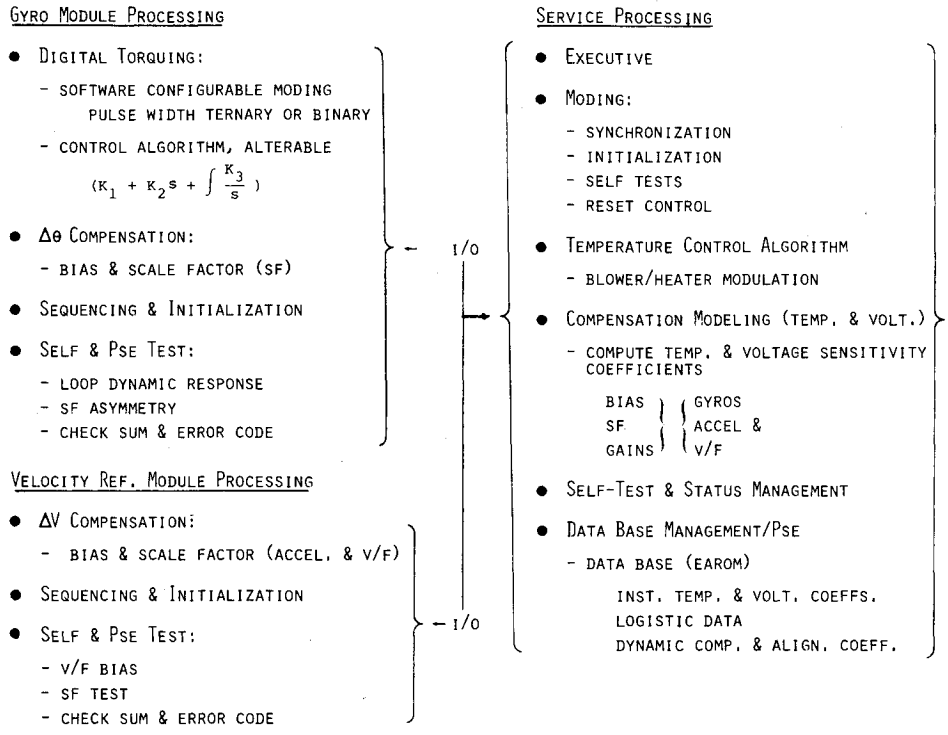

Fig. 5 Microprocessor tasks.

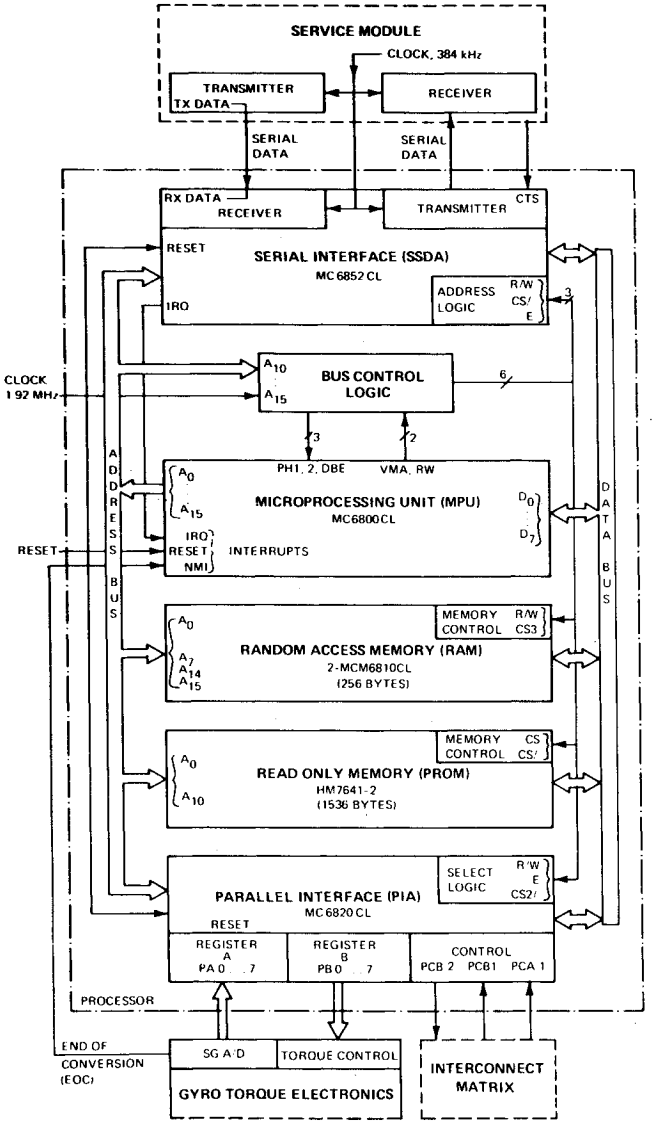

Fig. 6 Processor organization.

turn-ons, or if unacceptable system power excitation conditions occur, or when commanded by the higher-level missile management processor. The NMI interrupt is used in the different processors as a basic internal program cycle initiator. It also serves to provide a basis for synchronization of all the processors in the system. The IRQ functions as the input message processing interrupt (SSDA receiver).

Programmable read-only fusible link memory is used. These memory elements are programmed, as applicable, to contain the dedicated routines required for the different VRM and GM functions. The RAM is used for the processing task variables. The special-purpose bus control logic shown was added primarily to enable the peripheral expansion capabilities that are required in the service processor configuration. The SSDA provides for serial synchronous communications between the instrument processor and the service module. It permits simultaneous transmission and reception with 3 bytes of FIFO buffering in each direction and includes automatic generation and detection of parity. The PIA provides parallel interfacing capabilities and two 8-bit bidirectional channels in which each bit may be independently configured as either an input or an output. In the GM, the two channels are dedicated to an input and an output, respectively. In the VRM, the channels are mixed and 12 bits are allocated for reading the ΔV accumulators, while the other lines are used for accumulator control.

The service processor configuration corresponds to an expansion of Fig. 6 by the addition of several peripheral devices. The RAM and PROM are increased, and an EAROM is added. Two SSDA's are required—one for internal and the other for external data communication (Fig. 4). Two PIA's are used. One provides control functions for the EAROM and the polling and gating control of each of the GM and VRM SSDA's. The second PIA is used for data transfer and control of an A/D multiplexer in the service module (temperature monitoring), and provides the external communication control functions shown in Fig. 4.

Gyroscope Module (GM)

The gyro, installed in an aluminum mounting block, is interconnected to the torque electronics (GTE) card, which interconnects via the system "motherboard" to the instrument processor. The gyro input axis is approximately

perpendicular to the plane defined by the pads on the mounting block. These pads mate with surfaces on the LCIGS frame assembly. Three different sets of mating surfaces on the frame define an orthogonal angular rate-sensing triad. Temperature sensors and electronics are mounted on the block to provide for temperature compensation processing and instrument normalization (i.e., torquer and motor tuning, etc.).

The GTE interfaces with the gyro signal and torque generator. It amplifies, filters, and digitizes the gyro signal generator (SG) output signal (peak sampling at 3.2 kHz) with an A/D. The A/D output is read by the instrument processor via PIA data register B. The processor computes a torque control command in each sampling cycle which is transmitted via PIA register B to the GTE.

The GTE, in response to the torque command, applies a precise amplitude-controlled current to the gyro permanent-magnet torque generator. The decoding logic configures the GTE in ternary (PWT) or binary pulse width (PWB) torquing operation and meters the pulse width and polarity in accordance with the processor's moding and $\Delta\theta$ data command. The maximum pulse width in each cycle corresponds to approximately 312 μs with an equivalent width resolution of 60 to 1.

The different mode options are provided so that different instruments may be controlled in a manner best suited to their specific design sensitivities. For example, PWB control maintains constant power on the gyro torquer, and scale factor asymmetry (ΔSF) yields an equivalent bias. However, ΔSF instability results in large bias instabilities. PWT avoids the ΔSF instability problem, but applies variable power, dependent upon the sensed angular rate. The mode that is used by a processor is based on the data stored in the EAROM.

The gyro processor performs the control-loop algorithm and compensates the accumulated $\Delta\theta$ torque count for bias and scale factor errors, including thermal variations, and formats the corrected data for transmission. The compensation processing is similar to the VRM described in the next section, except that independent gyro and electronics modeling is not required since closed-loop operation is used in the GM. The coefficients used in compensation processing are altered in the processor by updates from the service processor (SP). The updates are determined by the SP from stored EAROM data sensitivity coefficients and sensed instrument and electonics temperatures.

The processor functionally implements a control algorithm of the form

$$T = K_1\theta + K_2\dot{\theta} + K_3\int\theta dt \qquad (1)$$

where T is the torque command to the GTE and corresponds to the SG-sensed gyro motion, and K_1, K_2, and K_3 correspond to the proportional, rate, and integral gain coefficients. The rate term is used to compensate for the lags associated with the gyro time constants and computational delays of the digital processing. The integrating function minimizes gyro gimbal offsets with rate inputs, which reduces errors due to cross-axis coupling, etc.

The design achieves a nominal loop bandwidth of 80 Hz. The step response rise time is approximately 3.75×10^{-3} s, and the overshoot is less than 20%. An average gyro float hang-off, 80 arc-sec, exists at an angular acceleration of 5 rad/s^2. Figure 7 is a plot of a GM step response, as obtained from reading the processor A/D register and torque command data at 3.2 kHz. The control point is offset 40 A/D bits (240 arc-sec) and then returned to the SG null point. The step response shown is one of the self-test provisions. Others include SF asymmetry, A/D resolution, processor check sum, and error code testing. These test functions are embodied in the processing software structure.

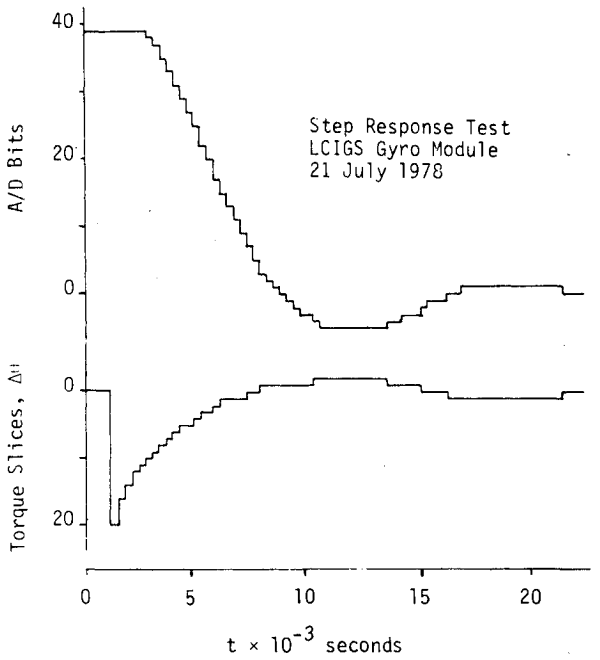

Fig. 7 Step response test results.

The use of identical interchangeable microprocessors for the GM requires that they be able to identify their own axis system assignments since communications with the SP is across a "party" line. Similarly, software phasing must be included so that the modules' NMI cycle rate and the SP's data transfer rate can be synchronized. Finally, for transient resets, recovery provisions must assure that the closed-loop torquing operations are resumed rapidly. Thus, the gyro processor automatically reverts to a default PWT operating mode until system reinitialization is completed. The GM memory capacity and current MPU utilization levels are shown in Table 1.

Velocity Reference Module (VRM)

The VRM consists of an accelerometer triad assembly, an electronics assembly, and an instrument processor. Power and timing are provided by the service module.

The accelerometer triad assembly consists of three accelerometers mounted with their axes nominally orthogonal in an aluminum block. A temperature sensor is mounted on the block for use in temperature compensation. The accelerometer is a gas-filled hinged pendulum with an integral analog torque-to-balance loop.

The accelerometer loop output, an analog dc voltage proportional to acceleration, is fed into voltage-to-frequency (V/F) converters in the velocity reference electronics (VRE), scaled to a nominal 0.5 Vdc/1000 cm/s^2. The V/F output frequency is proportional to acceleration, and each output frequency pulse corresponds to 1 cm/s.

A bipolar V/F design was configured to achieve the required bias stability. Three V/F converters and their associated data buffers are mounted on the VRE printed circuit card.

The V/F output is defined by

$$f_{\text{out}} = (K_{+,-})(V_{\text{out}} + V_{0s}) \qquad (2)$$

where the gain, K_+ or K_-, is dependent on the polarity (a bipolar design characteristic) of the total input voltage, $V_{\text{out}} + V_{0s}$.

$$V_{\text{out}} = K_{1a}(K_{0a} + a_{\text{in}}) \qquad (3)$$

Table 1 Gyro processor utilization[a]

Memory	Capacity	Utilization
RAM	256	143
ROM	1536	1433

[a] Time available utilized = 85%

Table 2 VRM processor utilization[a]

Memory	Capacity	Utilization
RAM	256	188
ROM	1536	1326

[a] Time available utilized = 58%

Table 3 Service processor utilization[a]

Memory	Capacity	Utilization
RAM (8 bit-bytes)	1024	269
ROM (8 bit-bytes)	4096	2361
EAROM (8 bit bytes)	1024	730

[a] Time available utilized = 70%

where K_{Ia} is the accelerometer scale factor, K_{0a} is the accelerometer bias, a_{in} is the sensed acceleration input, and V_{os} is the offset voltage of the V/F converter. The V/F bias (K_{0VF}) corresponds to

$$K_{0VF} = (K_{+,-})(V_{os}) \tag{4}$$

The VRM processor accumulates the V/F count in the VRE buffers every 625 μs and compensates this data for bias and scale factor errors. Since the V/F functions as an open-loop digitizer, the processor compensation routines must correct for the V/F bias (V_{os}) and its plus and minus scale factors, as well as the accelerometer bias and scale factor. All of these terms are temperature sensitive, and the thermal modeling must also include them.

The velocity reference processor computes a corrected velocity, ΔV_{comp}, of the form

$$\Delta V_{comp} = [(1 + \Delta SF_c)\Delta V_r] + B_c \Delta t \tag{5}$$

where ΔV_r is the raw accumulated V/F output velocity change data in an update interval, B_c is the bias correction in velocity during the update interval, ΔSF_c is the scale factor correction, and Δt is the time between updates. The memory capacity and current MPU utilization levels are shown in Table 2.

Service Module Processor (SP)

A major function of the SP is to effect the thermal compensation modeling. For example, the B_c and SF coefficients in Eq. (5) are computed by the SP using the parameters stored in its EAROM and the sensed temperature data that is multiplexed and digitized in the service module. The computations are achieved using the hardware multiply provisions in the service module and are of the form

$$B_c = -[K_{0a} + K_{0VF}] \tag{6}$$

$$\Delta SF_c = 1/(1 + \Delta K_I) - 1 \tag{7}$$

where

$$\Delta K_I = [(1 + \Delta K_{Ia})(1 + \Delta \pm K_{IVF})] - 1$$

and $\pm K_{IVF}$ represents the plus and minus V/F scale factors.

The coefficients, K_0 and K_I, are corrected for temperature sensitivity using a two-slope characterization, if required. The processor determines the individual K_0 and K_I terms for the accelerometer and V/F using processing structure of the form

$$K_{(\ )} = K_{(\ )} + (K_{(\ )}/\Delta T)(T - T_N) \tag{8}$$

where K are the coefficients corresponding to the nominal operating system temperature and $(K_{(\ )}/\Delta T)$ corresponds to the applicable slope segment thermal sensitivity. The B_c and SF_c coefficients are then determined using Eqs. (6) and (7). The SP computes new correction coefficients for all of the modules, and updates all of the modules once every 20 s. These computations also include modeling to account for thermal transients, the thermal lag between sensor readings, and the instrument responses to environment temperature changes.

The bias and scale factor parameters may be determined at module or system level testing. The parameters are loaded into the EAROM via PSE. The EAROM serves as the data base for the LCIGS system. In addition to instrument parameters, compensation parameters, torque moding (PWT or PWB), control loop gains and dynamic terms (g-sensitive and OA coupling coefficients, etc.), information defining instrument type, SN, installation and last test date, etc., may be stored for logistic purposes. Data base management is effected by using the PSE. A magnetic stripe card function is included in the PSE. The stripe card accompanies the LCIGS, and its data duplicates the EAROM load. The card "current data" is used by the PSE to verify the EAROM load. It is updated at each calibration or repair by using the PSE. Card file status may also be compiled for inventory control.

In addition to the compensation processing function, the SP also implements a coarse temperature control function using aircraft power during captive flight. The SP implements a control algorithm using the thermal sensor data and time-modulates the application of power via solid-state relays to either a trim heater or a blower. Since coolant sources are not available and battery power cannot be spared for heating in free fall, thermal modeling is also required. The coarse control provision narrows the thermal variation region which optimizes modeling during captive flight.

In addition to the application tasks just described, the SP performs the major system executive and moding functions described in the next section.

The SP memory capacity and current MPU utilization levels are shown in Table 3.

Executive Software

The executive is responsible for all I/O procedures and internal task scheduling. As noted previously, the executive functions, in order of priority, correspond to the system reset, nonmaskable interrupt (NMI), and interrupt request processing routine (IRQ).

In the instrument processor, the reset routine initializes all variables stored in RAM, and configures I/O devices as required. It leaves the processor in a well-defined default mode that can only be altered by command from the SP.

The NMI, common to all processors, provides intersystem synchronization.

The IRQ routine is an input message processor. The SP broadcasts various commands and data to the instrument processor. Some examples of these messages are: send compensated inertial data; send raw inertial data; send self-test data; receive this bias correction; receive this scale factor correction; and perform self-test program. The IRQ interprets these messages and configures the executive to perform the requested command.

The SP executive also contains reset, NMI, and IRQ programs, and provides for low-priority task schedule control.

The reset program must initialize RAM and I/O devices, but it also has the responsibility of insuring that all processors are synchronized correctly. A power-up reset is a special case, where the SP must command all gyros to be torqued into a preferred stop.

The most important responsibility of the SP NMI is to gather the inertial data from the four sensor processors and pass this information along to the missile processor. The NMI program must also maintain the real-time clocks, read selected temperature and voltage multiplexed channels, and provide system temperature control.

The IRQ program performs message control functions similar to those described for the instrument processor. The low-priority tasks are initiated either by a message interpreted by the IRQ or from another active low-priority task. The subroutines used to execute task control (Fig. 8) are called CMDPRO, ADDJOB, and JOBCTL. The CMDPRO subroutine interprets messages received from IRQ, selects the application task, and adds it to the task queue by using the ADDJOB subroutine. The tasks in the queue are activated in sequence by the JOBCTL program. A listing of the application tasks is also shown in Fig. 8. For example, the BITEJB task corresponds to the compensation modeling function, Eqs. (6-8).

The task "INERT" is initiated on command from the missile processor. The command is interpreted as "Start inertial data processing and transmission." The task performs the transfers of initial compensation coefficients to the instrument processors and the missile processor. The instrument processors are then instructed to transmit compensated inertial data. The final responsibility of INERT is to schedule the BITEJB task.

"PIPELN" will transfer, without interpretation, a command message directed to an instrument processor. This feature is primarily used by the PSE during system test. It provides the PSE with flexible control over the instrument modules.

The electrically alterable read-only memory (EAROM) is updated by the PSE (e.g., after a system calibration). The task "LDEAR" provides the capability of selectively loading blocks of EAROM.

The DUMP task is used by the PSE to selectively interrogate any block of memory, whether it be RAM, ROM, or EAROM. For example, this feature is used with LDEAR to read, update, and restore EAROM after a calibration.

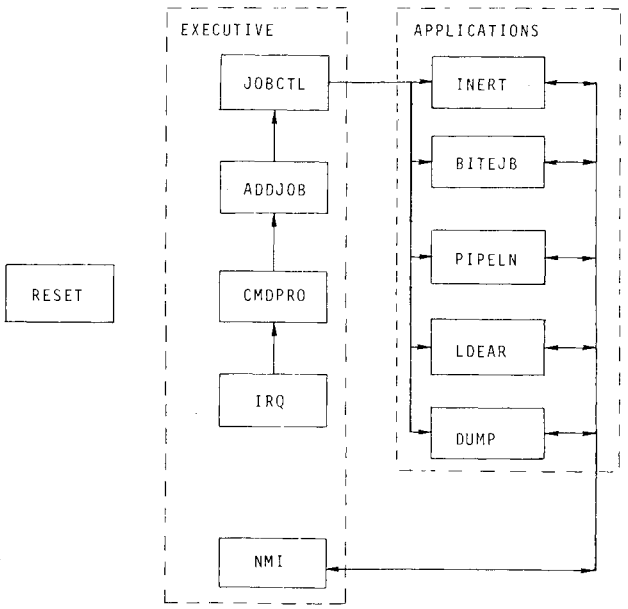

Fig. 8 Executive applications.

Expanded Processing

The expansion of the LCIGS processing capabilities to include cross-axis compensation, attitude and velocity algorithms, and navigation functions is feasible. Several alternatives are possible. An additional processor, which is also augmented by a hardware multiply function interfaced with the SP output port, can be mechanized, or a more powerful microprocessor can be used to replace the SP, or a more extensive hardware multiply function can be added.

The current SP could be replaced with the recently announced M68000, a 16-bit MPU that contains hardware multiply functions, and a control bus that is designed to connect directly to the standard M6800 peripheral chips used in the LCIGS instrument processors.

The major real-time burden associated with the expansion of LCIGS processing capability is in the implementation of the attitude and velocity algorithms. Traditionally, a third-order quaternion[4] transformation has been used in minicomputer implementations. Typically, the attitude and velocity algorithms require numerous multiplications and operate at an iteration rate of 100/s. This represents a duty cycle problem in the more limited microprocessor. Recognizing this limitation, a partitioned algorithm was developed and demonstrated using a 1 MHz M6800 MPU and a peripheral 32×32 hardware multiplier. The multiplier implemented a serial algorithm at an 8 MHz clock rate. The partitioned algorithm implemented a fast and slow loop processing mechanization based on the technique introduced by Bortz.[5] The fast loop computes a term called the rotation vector, ϕ_n, which is related to the quaternion, q. The slow loop utilizes this rotation vector to compute an incremental Δq that updates the quaternion's solution (a similar approach is used in body velocity processing):

$$\phi_n = \phi_{n-1} + \Delta\theta_n + [\tfrac{1}{2}\phi_{n-1} + (1/12)\Delta\phi_n] \times \Delta\theta_N \tag{9}$$

$$\Delta q_m = \left(\cos\frac{\phi_m}{2}, \frac{\phi_m}{\phi_m}\sin\frac{\phi_m}{2}\right) \quad \phi_m \triangleq |\phi_m| \tag{10}$$

$$q = q_{m-1}\Delta q_m \tag{11}$$

In Eq. (9), $\Delta\theta_n$ is the incremental body angle accumulated over the minor cycle. At each slow-loop cycle m, the Δq is generated and the ϕ_n reset.

Tests were conducted with both the partitioned attitude velocity algorithms at fast- and slow-loop operating rates of 100/s and 20/s, respectively. For the attitude algorithm, for example, a total of 1420 multiplications per second were required (900 in the fast loop and 520 in the slow loop), which corresponded to more than a 50% reduction over the traditional 100/s quaternion implementation. In the test implementation, the attitude algorithm utilized approximately 27% of M6800 real time (the 100/s loop used 19% and the 20/s loop used 8%). The velocity algorithm utilized approximately 20.5% of real time (15% in the 100/s loop and 5.5% in the 20/s loop). The coning performance of the partitioned processing implementation was equivalent to that achieved in a third-order 100/s minicomputer implementation. The use of this partitioned implementation permitted processing expansions with adequate real-time margin. The additional 3000 bytes of memory projected for this expansion are well within the MPU's architectural capabilities. The phase-in of a M68000 microprocessor, when it becomes available, would obviate the need for the 32×32 hardware multiplier peripheral.

Conclusion

The LCIGS distributed processing implementation has provided a unique level of design and application flexibility. It has permitted functional hardware partitioning with in-

strument interchangeability and enables considerable growth potential. The dedicated processing also allows partitioning of software to manageable subsets of "firmware" that are isolated from mission- or weapon-related changes. Further, the microprocessors have enhanced testing by permitting microscopic checkout capabilities; e.g., the processors perform their own test control and data acquisition.

Microprocessor implementations do present constraints for the software designer. Because of the microcomputer limitations, the designer must effect tradeoffs between duty cycle and memory utilization. Often table-lookups, in lieu of conventional processing, become a required time-saving measure. Similarly, compromises are required that replace software multiplication with simple shifting operations.

Finally, program development software for microprocessors has not yet reached the general-purpose computer maturity level. For example, cross-assemblers with provisions for independent relocatable subroutine generation and linking are not universally available. Thus, independent development of programs is practically impossible and intensive coordination between programmers is required.

Acknowledgments

The work reported in this paper was sponsored in part by the U.S. Air Force Armament Laboratory of the U.S. Air Force Systems Command under Contract F08635-76-C-0306.

The author acknowledges the assistance of R. Cooper, D. Dunn, R. McKern, K. Vincent, T. Schamp, H. Musoff, and J. Publicover.

References

[1] Gilmore, G.P., "Low Cost Inertial Guidance System Design Concept and Operating Features," The Charles Stark Draper Laboratory, Inc., Cambridge, Mass., C-5072, March 1978.

[2] "Motorola Wins Microprocessor Contract from GM," *Electronics*, Vol. 50, No. 2, Jan. 1977, p. 31.

[3] "M6800 Microcomputer System Design Data," Motorola, Inc., Phoenix, Az., 1976.

[4] McKern, R.A., "A Study of Transformation Algorithms for Use in a Digital Computer," Massachusetts Institute of Technology Instrumentation Laboratory (The Charles Stark Draper Laboratory, Inc.), Cambridge, Mass., T-493, Jan. 1968.

[5] Bortz, J.E., "New Mathematical Formulation for Strapdown Inertial Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-7, Jan. 1971, pp. 61-66.