THE IMPLEMENTATION OF FAIL-OPERATIVE FUNCTIONS IN INTEGRATED DIGITAL AVIONICS SYSTEMS

Stephen S. Osder
Sperry Flight Systems Division

SUMMARY

System architectures which incorporate fail-operative flight guidance functions within a total integrated avionics complex are described. It is shown that the mixture of flight critical and non-flight critical functions within a common computer complex is an efficient solution to the integration of navigation, guidance, flight control, display and flight management. Interfacing subsystems retain autonomous capability to avoid vulnerability to total avionics system shutdown as a result of only a few failures.

INTRODUCTION

The advent of the airborne digital computer in an attractive practical configuration (from the standpoint of cost, size and power) has set the stage for the emergence of a variety of new avionics system architectures. Despite the continuing growth in requirements for navigation, guidance, control and data management functions, the industry is faced with relentless pressures to hold system costs to pre-1970 levels. We require increased system sophistication, but cannot afford increased cost or increased complexity and its concomitant reliability penalty. The solutions appear in new avionics architures that feature a high level of integration and consolidation of functions. Indeed, the trivial answer to any cost trade-off study of competing avionics architectures is the totally integrated system where a single central computer (of sufficient speed) performs all required functions so that the cost of functional growth is measured only by the cost of the memory increment. This solution does not acknowledge the complicating factors of flight critical fail-operative requirements and the related problems of fault isolation and redundnacy management.

The usual approach to defining a system architecture that must provide some fail-operative functions is to separate subsystems into fail-operative and non-fail-operative categories. In this paper it is shown that this type of separation does not result in the most efficient mechanization of the desired function. An alternative integrated system architecture that starts with the requirements for the fail-operative autoland and stabilization and control functions is described. It soon becomes apparent that the majority of information interfaces needed for these fail-operative functions are also used for the other guidance, navigation, display and data management requirements. The system architecture and safety techniques used to mechanize the fail-operative requirements can be made completely compatible with the generally accepted methods of implementing the non-flight critical functions.

Expanding from the fail-operative flight guidance system, additional interfaces are added to achieve the remaining navigation, control and display functions. These additional functions are treated differently in terms of interface hardware and software mechanizations because the rather elaborate monitoring and fault isolation routines for fail-operative performance are not required.

The vulnerability of such integrated systems to the total loss of avionics functions with only two failures, such as the loss of two central computers, must be avoided. Consequently, the system architecture must make provision for continued although degraded operation through the retention of autonomous capability in the various interfacing subsystems. These back-up provisions generally appear as residual hardware functions in contrast to the software functions which are provided by the primary or central integrated mode of operation.

This paper presents a brief rationale for the selection of a totally integrated avionics architecture over two other competing candidates. The organization of the totally integrated system and the techniques for achieving fail-operative performance for flight critical modes are described. The vulnerability to total system shutdown is analyzed, and methods of protecting against that vulnerability are suggested. In general, the practical feasibility of such a totally integrated avionics system appears to be limited only by questions regarding the manageability of the system software.

SYMBOLS AND ABBREVIATIONS

θ	Pitch Attitude	Q	Dynamic Pressure
Φ	Roll	P_s	Static Pressure
ψ	Heading	P_T	Total Pressure $(P_T - P_s) = Q_c$
$F\theta$	Column Force	$P_F(t)$	Probability of failure in time duration t
$F\Phi$	Wheel Force		
A_x, A_y, A_z	Linear body axis accelerations in x, y, z direction	T_T	Total Temperature
		T_s	Static Air Temperature
h	Altitude	V_c	Calibrated Airspeed
M	Mach number	INS	Inertial Navigation System

ILS	Instrument Landing System	MLS	Microwave Landing System
CWS	Control Wheel Steering	MFD	Multi-Function Display
DME	Distance Measuring Equipment		

RATIONALE FOR CANDIDATE SYSTEM ARCHITECTURE SELECTION

Three generic candiate avionics system architectures illustrate the requirements, considerations, and controversies surrounding the selection of an integrated avionics approach for future transport aircraft. These three candidates are:

1) The Federated System -- a combination of new computers for each required class of functions. This is a direct extension of today's technology, but the argument is made that computers are becoming sufficiently inexpensive that we can afford the separate computers of the federated concept. This argument does not address the problem of intercomputer communication and interface complexity.

2) The Integrated System with Separate, Fail-Operative Flight Control Computers -- a major acknowledgment of the need for integration but, nevertheless, it continues to duplicate the majority of sensor interfaces in order to separate the fail-operative guidance functions.

3) The Integrated System with Self-Contained, Fail-Operative Flight Control Functions -- this system involves a minimum of interface duplication.

Trade-off analyses of these three configurations can be performed to prove any desired conclusion merely by applying the desired arbitrary weighting to one or more criteria of interest. Therefore, rather than perform a quantitative trade-off we will illustrate how a single parameter, "the interface complexity," varies with each of the candidate architectures. It is contended that interface complexity is the single most significant factor that influences cost, complexity and reliability of digital systems. When the computation and logic are performed in software, the largest hardware function is the acquisition and distribution of the data required by the computer. If we minimize the scope and complexity of that function, we create the simplest, least expensive and most reliable system. With this viewpoint in mind, we can compare the three candidates with reference to Figures 1, 2 and 3 which illustrate some of the typical interactive elements of the system requirements.

Figure 1, the federated combination of computers, is an extension of the 1970 state of the art where integration exists primarily to the extent of sharing sensor sources through relatively standardized interface mechanizations.

The navigation computer, in this concept, is responsible only for area navigation, receiving navigation sensor and Inertial Navigation System (INS) inputs. The flight control computers retain their traditional autopilot and flight-director modes, including autoland; hence the triplex redundancy for the fail-operative requirement. Note that in all candidate systems, a separate flight control electronics function is shown in order to emphasize the fact that a considerable amount of electronics are required in addition to control law and logic computation. This electronics is associated with servo actuator drives, engage and shutdown controls, power conditioning for transducer excitations, and some signal conditioning. Dual, independent air data computers feed the navigation computers, the flight control computers, and dual EPR/autothrottle computers. Redundant navigation receivers representing the ILS function feed both the flight control (autoland) computers as well as the navigation computers.

This candidate is rejected because it represents the extrapolation of the traditional and presumably unsatisfactory approach to avionics. The problem of unwieldly interconnections and equipment growth is not adequately handled by this configuration. More interfaces are generated, and the number of black boxes grows, as we can readily see in Figure 1.

The second candidate (Figure 2) makes a reasonable attempt at integrating functions and minimizing black boxes and interfaces by using the navigation computer as the new integrating element. That computer complex incorporates all navigation, including air data computation and thrust management/autothrottle computations. It also includes flight path guidance computations other than those associated with autoland. The weakness of this approach is the use of three additional computers and their associated interfaces for the basic autopilot plus autoland guidance functions. The input interfaces required for the flight control computers are: VHF navigation receivers (ILS), air data (h, Q, $\dot{h}$, V_T), attitude and heading, radio altitude, accelerometers (A_z and A_y), and a considerable amount of mode selection logic. All of this information, with the possible exception of radio altitude, is also required in the navigation computer. Moreover, if provision is made for growth to MLS, then the MLS localizer, glide slope and DME will be required interfaces for both the flight control and the navigation computers. What then is the reason for also moving this information to a separate set of flight control computers? It can only be the edict that flight control functions are flight critical, as implied by the fail-operative requirements, while the other functions are not. Hence, if one assumes that fail-operative capability is achieved with a minimum of triplex redundancy, Candidate 2 is a natural conclusion.

The simplest interfacing of sensors is achieved with the third candidate (Figure 3). This system mechanizes the fail-operative autoland functions with two computers. These computers are shown interfacing with a triplex actuator control mechanization, although that interface could readily be quadruplex. Since the autoland architecture does not differ from the system architecture requirements of the non-flight critical navigation functions, those navigation functions are incorporated in the same computer complex. Triplex navigation functions are interfaced with both computers, as in the other candidates, but only one set of interfaces is required. This interface reduction is representative of the significant minimization of electronics and wiring when this level of functional integration is implemented.

Candidate 3 is based on technology advances made in recent years where techniques have been developed that permit 100-percent fail-operative performance with dual digital computers. We define 100-percent fail-operative as follows: If the probability that the best contemporary triplex or quadruplex fail-operative system will respond properly to all failure situations is P_1, and the probability that the dual digital system will respond properly is P_2, then

$$P_2/P_1 \geqslant 1.0$$

In effect, this definition acknowledges that all fail-operative systems have loop-holes in such matters as multiple simultaneous failures, but the recommended dual system is at least as good as the best contemporary system in regard to fail-operative integrity.

If the fail-operative functions are mechanized in dual computers and will meet every stringent safety ground rule for Cat. III certification, why not use the same computers (using non-fail-operative techniques) for the other functions? When we follow this approach, the resultant configuration yields a major reduction in interface complexity and a significant reduction in the number of required black boxes.

SYSTEM ARCHITECTURE, REDUNDANCY AND SUMMARY OF FUNCTIONS

The recommended system organization is illustrated in Figure 4. The dual computational redundancy is represented by the pair of data adapters and computers. The autoland and stabilization and control autopilot functions that must be fail-operative are contained within the elements shown on this block diagram. Moving from left to right on the diagram, this is achieved through the use of appropriate redundancy in the required sensors, special hardware techniques within the data adapter, special software monitoring and data handling routines within the computer, and the necessary redundancy to interface the flight control electronics with the aircraft's electro-hydraulic actuation system. The number of flight control electronic units is shown as n where n may be three channels or four. Whether the control electronics is triplex or quadruplex depends upon the specific aircraft application and its servo actuator/control surface philosophy. All other non-fail-operative sensing and computational functions are performed without these special fail-operative techniques, although very thorough monitoring and fault isolation software routines are included for non-fail-operative as well as for the fail-operative functions.

A data adapter, a computer, and a flight data storage unit (mass storage) make up one computer complex. The data adapter is the computer's hardware interface with the physical world. It isolates the computer from all problems of electronic mechanization so that the computer's only contribution to the system is contained within its software. The data adapter serves as a communications terminal for all data transfers, and as a data conditioning and data conversion center for its computer.

Each computer contains a program for performing all flight control, guidance, navigation, automatic flight planning, air data computation, engine EPR (thrust rating) computation, autothrottle controls and associated display functions. In regard to displays, CRT instruments are recommended for the ADI and HSI. The HSI function is implemented from a Multi-Function Display (MFD) which provides a moving map presentation (or, on pilot selection, a fixed map, moving aircraft display). The computer provides all the electronic map data processing; it receives continuous updates of data from the flight data storage unit, an air-bearing disk memory that provides mass storage of air navigation route logistic data. The computer also contains programs that allow it to perform an automatic central integrated test function that enhances the maintenance management of a major part of the aircraft's avionics equipment. It also presents checklist information on the MFD and includes interactive interfaces with the flight crew through pedestal mounted Control and Display Units (CDUs). These CDUs are normally used for automated flight plan selection and modification; however, their keyboard controls and associated alphanumeric readout (in conjunction with the large data display capability of the MFD), allow a convenient man-computer interface for checklist activity.

As shown in Figure 4, switching controls, activated automatically or by the crew, allow transferring of displays and sensor sources from left side to right side, and vice-versa.

SENSOR SUMMARY

The sensor requirements are covered as general categories in Figure 4. A list of the sensor complement and a discussion of redundancy requirements follows. In the category of stabilization and control, sensors are:

- CWS Force Sensors ($F\theta$, $F\Phi$)
- Yaw Rate* (r)
- Pitch and roll Attitude* (θ, Φ)
- Heading ** (ψ)

*It is recommended that pitch and roll rates be obtained as software-derived rates from the attitude data.

**Heading data free of gimbal errors is desirable because this information is used for coordinate transformations during turning maneuvers in those configurations which are not provided with INS. If ψ is obtained from a conventional 2-degree-of-freedom directional gyro, then a gimbal error correction algorithm is incorporated in the system software.

- Linear Acceleration Triad (A_x, A_y, A_z)
- Flap Position
- Surface Position

The Air Data Sensors are:

- Static Pressure (P_s)
- Total Pressure (P_T)
- Total Temperature (T_T)

(Note that angle of attack (α) may be computed from inertial and baro data.)

An inertial navigator is shown, although for the configurations that do not include an INS, provision is made for inertial smoothing of radio navigation data, using strapdown accelerometers, plus attitude and heading references. When the INS is provided, its velocity-north and velocity-east information is used as the basis of the smoothing algorithm, and the short-term strapdown inertial computations are not needed. The radio NAVAIDS are:

- VOR
- DME
- ILS

although provision is included in the data adapter for interfacing with the future MLS system and hyperbolic radio navigation systems such as OMEGA.

The radio altimeter is required only for the autoland and instrument approach functions. Engine EPR is needed for the autothrottle EPR mode, and throttle servo rate is needed because the throttle servo loop is closed through computer software.

Redundancy of sensors where fail-operative capability is required is approached by using the three techniques illustrated in Figure 5. The first (Figure 5a) feeds each sensor into each of the dual computing channels. A voting, middle-value selection or averaging algorithm is mechanized in the computer software to ensure that both channel 1 and channel 2 use the same estimate of the sensed parameter. Intercomputer communications, via buffered serial data links, inform each computation channel of the estimated value, ($\hat{A}$, $\hat{B}$, $\hat{C}$), and whether a sensor discrepancy or anomaly has been detected. The technique of Figure 5a is the most efficient from the standpoint of sensor equipment minimization, least efficient from the standpoint of interface complexity (and wiring), and somewhat more complex in regard to software complexity when compared to the other candidate sensor configurations.

The second technique (Figure 5b) uses quadruplex sensors arranged in pairs. As in the first case, software voting and averaging are used to isolate faults and equalize the estimates in both channels. The third arrangement (Figure 5c) uses internally monitored sensors that generate their own valids to indicate that the data is usable. The serial data exchanges allow channel equalization. When this method is used, appropriate interfacing techniques are employed to avoid the situation where the valid is received, but the data is lost through an open connector pin.

There are many factors which enter into the selection of configuration 5a, 5b, or 5c for a specific sensor. Some of the considerations are logistic. For example, two sets of dual sensors (5b) may be easier to maintain than three individual sensors (5a). Other factors involve safety guidelines and allowable probability that a failure may be undetected. For example, configuration 5c assumes a self-monitored sensor. Modern radio altimeters fall into this category, but it may be argued that the built in sensor monitoring is not 100 percent effective and a finite probability may exist for an undetected radio altimeter failure in the final phases of an autoland approach. We may respond to a stringent safety guideline regarding radio altimeters by adding a third sensor and using the configuration (5a) approach. However, it can be shown that the validity determination for a given sensor may be augmented within the system's monitoring software where state estimations from other types of sensors may be used to verify a given sensor signal. Thus, for example, a radio altimeter signal may be analyzed with regard to its validity by means of comparisons with baro-inertial estimates of the aircraft's vertical velocity. Hence the 5c sensor configuration may be justified over the 5a configuration.

MONITORING CONCEPT FOR DUAL-FAIL-OPERATIVE FLIGHT GUIDANCE FUNCTIONS

Summary

The two halves of the total, fail-operative Digital Flight Guidance System are designated as channel 1 and channel 2 (Figure 6). Channel 1 has a dual internal structure with the two parts designated as channels A and B. Channel 2's subchannels are also designated as A and B. Both channel 1 and channel 2 are autonomous of each other, and each is capable of operating as a fully monitored fail-passive system. Each channel is designed to detect any discrepancy from normal operation and activate safe shut-down controls if the discrepancy is deemed to constitute a system failure.

There are several different monitoring techniques used to achieve 100-percent failure detection in each computer channel. Unlike analog systems, however, we cannot identify a unique set of malfunctions with each type of monitor. There are very large overlaps in the fault detection routines. Four different monitoring algorithms, for example, may detect one failure. In some cases this overlap is exploited to permit partial shutdowns, and in other cases only a total channel shutdown is permited. The following is a summary of the types of fault detection techniques that are employed:

- Processing of sensor valid discretes

- Sensor data validity and reasonableness checking algorithms

- Sensor data comparison monitoring -- variable thresholds dependent upon aircraft state, signal amplitude and signal duration

- Redundant computations internal to the computer using separate computer memory banks and comparison checks of results

- End around I/O checking -- all outputs are fed back to the computer via the input conversion sections and verified against the specified output

- Test words continuously checked for all intrasystem communications

- Model and comparison monitoring of servo actuator responses

- Software executive continuously verifies that the required sequence of software tasks is accomplished each 50 millisecond iteration period

- External (to computer), dual hardware monitors examine the computer's output for a required dynamic signal pattern -- any computer failure that will prevent the execution of the specified program will cause the pattern to cease.

In addition to the monitoring algorithms, all input signal data are processed so that all redundant control law computations are performed with identical values for all variables. Hence all control output commands must be identical. The servo actuator commands are therefore identical so that servo system monitoring criteria are dependent only upon servo system tolerance. Some cross-channel (between channel 1 and 2) computation equalization is needed, but the amplitude constraint on the amount of equalization is a small percent of the control authority. Cross-channel equalization is needed to correct for small offsets caused by an occasional 50-millisecond time skew between data used in channel 1 and channel 2.

Computer Executive and Hardware Monitor

Descriptions of the input signal screening, monitoring and equalization algorithms are beyond the scope of this paper. The necessary system concepts can be appreciated as extrapolations and improvements over techniques used in contemporary analog systems. However, some additional comment is needed to elaborate on the concept of a 100 percent, self-monitored computer. A computer system verification function is used to generate a prescribed output signal pattern at the end of each iteration cycle only if a checklist of required computation routines has been completely satisfied. The instructions for checking off this list are therefore interwoven throughout the entire program so that if any of the required routines is not properly completed, or if a processor function is faulty, the verification signal pattern will not be properly generated. This verification signal is D/A converted and transmitted to the hardware monitor in the Data Adapter where it is compared with a correct signal pattern. A difference in these signals will cause the computer complex to shut down safely (without servo command transients). Since the verification signal is dynamic and must contain correct timing information to be valid, a failure in the verification signal path to the hardware monitor (such as an open or a hardover) will be detected, as well as timing errors in the computer. The computer system verification function serves principally to detect massive computer failures, and does not allow shutdown of partial computation functions as is possible with the software monitoring functions. Nevertheless, there is a very intimate relationship between the software and hardware monitoring functions. This is shown in a simplified representation in Figure 7. In this figure the concept of an executive program which generates a task list as a function of the status logic is illustrated. With the completion of each of its specified tasks, the program acknowledges that it is ready for the next task by setting a task-completion bit. When the real-time interrupt that controls the program iteration rate occurs, a check is made to determine whether all required tasks were completed. If they were not, the computer software recognizes a computation failure and jumps to a failure response routine. It simultaneously neglects to generate the correct output pattern. In this case both the software and hardware monitors will detect a failure, but the hardware monitor will require a few cycles of incorrect output before it will respond. For simplicity, an output pattern in the form of a 10 Hz square wave is illustrated by Figure 7. In practice, more complex, multilevel patterns have been used.

Failures of the digital computer's logic circuitry associated with the execution of specific instructions will result in the condition just described. The airborne program incorporates techniques which deliberately exercise the instruction repertoire so that failures in repertoire logic will cause the program sequence to get lost -- that is, the program is forced to a wrong address. The result is a program hang-up or loop where it never reaches completion of the specified tasks. The program will recognize the real-time interrupt, and the machine may be capable of executing shutdown instructions. However, a more fundamental computer failure, such as loss of clock or memory read-write circuitry, will leave the computer in a state where it cannot execute any instructions. In that case, the hardware monitor will detect a fixed state on output D rather than the required dynamic pattern on output D of the figure. It will thereby initiate a system shutdown by commanding a computer power-down and interruption of power to D/A output commands. As mentioned earlier, some dual computation paths are also used within the computer primarily to detect failures associated with single-bit malfunctions in storage of data words.

BACKUP CONCEPTS AND RELIABILITY IMPLICATIONS

Summary of Display/Control Functions

A complete description of the cockpit displays and controls and their interfaces with the redundant computer complex is beyond the scope of this paper. However, it is essential that the software-controlled functions be identified so that we can devise an appropriate back-up strategy for the remote possibility of a total computer shutdown.

Referring to the highly schematic cockpit layout shown in Figure 8, consider normal system operation with computer complex No. 1 driving the left set of displays, and computer complex No. 2 driving the right set of displays. The computer/display interconnection may be switched, either automatically in response to failure detections, or manually by pilot selection. The primary flight displays are:

Multifunction Display

The MFDs primary use is to serve as an HSI incorporating a moving-map display. In this configuration, it provides the HSI pictorial representation of the flight situation with regard to course, course deviation, distance to destination and heading. The reference path is drawn as a solid line connecting waypoints. Projecting from the aircraft symbol is a trend vector depicting the aircraft's predicted location up to a software selectable time into the future. Behind the aircraft is a sequence of dots representing the previous position history. Waypoints, airports, airways, landmarks, VORTAC, VOR, VOR/DME stations are normally displayed on the map. The heading tape is at the top, with a digital readout of aircraft heading. Scale factor selection is provided on the MFD control panel located to the right of the MFD. Scales of 1, 5, 20 and 80 nautical miles-per-inch are provided, but these values are obviously completely under software control. When the landing area is reached, if the scale factor is reduced to 1.0 nautical mile-per-inch, then a runway symbol appears, and a useful presentation in the MLS era when accurate terminal DME and wide-angle azimuth to the landing area is available. The MLS accuracy would permit the use of the fine scale map so that navigation accuracy is consistent with map resolution.

On the left side of the MFD display area, various parameters associated with flight plan progress and 4-D guidance (arrival time) status are presented as alphanumeric readouts.

The map is also displayable in the north-up mode (moving aircraft fixed-map display) upon selection at the MFD control panel. Slewing controls move the map up-down and left-right, with the aircraft symbol remaining fixed at its true location on the map. Mode selection at the MFD control panel permits pilot editing of the map content. Other mode-select buttons delete the map and allow the display to list pages of data, such as that associated with route planning or preflight checklists.

Electronic Attitude-Director Indicator

This display presents the basic horizon presentation via instrument interfaces that are completely autonomous of the computer system (not under software control). Also independent of software is a digital radio altitude readout in the upper right window. Indicated airspeed appears in a window at the upper left of the screen, and the system software provides a choice of which parameter one can display in the window at the upper center of the screen. Experimental work has been done where this window was used to display distance to touchdown (during final approach) in nearest .1 nautical mile, or vertical speed in feet-per-minute.

Other information displayed and retractable (figuratively) under software control is listed:

- ILS or Flight Path Window

  Raw data deviation from the ILS flight path or computed position error from area navigation flight paths.

- Flight Path Angle Symbol

- Flight Path Acceleration

- Flight Director Command Bars

- Fast-Slow Indication

- Perspective Runway Symbol (This presentation is used when accurate DME information to the landing site is available, as in MLS systems.)

On the right bezel of the EADI is a set of approach progress annunciators. Modes that are armed illuminate amber, and when engaged they illuminate green.

Radio Altitude, Altitude, Vertical Speed, Airspeed/Mach

These indicators are clustered around the ADI in the conventional manner.

Autopilot Flight Director System Mode Annunciator

The mode annunciator is an electronic display containing four alphanumeric readouts that present the autothrottle mode, vertical guidance mode, lateral guidance mode, and autoland mode. These readouts flash if the mode is being captured, and illuminate steady when the mode is in a "track" phase.

Instrument AFCS/Warning Display

The instrument /AFCS warning display panel provides for annunciation of subsystem failures. A unit is located in the primary viewing area on each side of the instrument panel.

Dual Digital DME and Radio Magnetic Indicator

To the left of the MFD is a basic RMI indicator that has direct interface with the radio receivers and the heading reference systems in order to display bearing to VOR or ADF stations. It also provides dual digital DME readouts through direct digital interfaces with the DME receivers.

ATS/EPR Control Display Panel

This panel, located at the bottom of the center instrument panel, serves as the thrust-rating readout and thrust-mode selector. It also provides the means of engaging the dual autothrottle servos. By selecting either the take-off, maximum continuous, climb, cruise or go-around mode, the computed EPR limit for those modes is displayed in conjunction with the total air temperature. This instrument may also be used to display total and static air temperature and true airspeed.

Mode Select Panel

The Mode Select Panel (MSP) located in the glare-shielded region provides the following control and display capability:

- Dual VHF Nav Receiver frequency readouts (for display of an automatically tuned station) or manual tuning override capability -- located on left and right side of MSP.

- Speed Control mode select and reference readout (airspeed and Mach via pitch or autothrottle control).

- Vertical Guidance mode select and reference readouts. These include flight path angle and/or vertical speed and altitude pre-select displays and controls.

- Autopilot and Flight Director Engage Switches, including flight critical engage switches, turbulence mode control and engage controls for autoland, take-off and go-around.

- Lateral Guidance mode select and reference read-outs. These include heading and course set controls and display redundant navigation sources, plus means for selecting various navigation guidance modes and displays.

Dual Control/Display Units (CDUs)

Dual Control/Display Units (CDUs) are shown on the left and right side of the pedestal. These CDUs are normally used for automatic flight plan selection and modification. However, their general purpose keyboard controls and associated alphanumeric readout (in conjunction with the large data display capability of the MFD), allows a convenient man-computer interface for checklist activity.

Backup Concepts

The integrated system has many of the same reliability hazards as contemporary systems. If all attitude references fail in flight, many of the system functions and modes are disabled. If all of the NAV receivers fail, a different set of functions and modes are disabled. The superior fault isolation and failure assessment capability of the integrated system allows automatic reconfiguring of the navigation and guidance functions into alternate or degraded modes. The crew can also participate in the reconfiguring of the system data flow and displays through control of instrument switching. The fewer black boxes and the improved failure detection, isolation and annunciation capability results in a significant improvement of overall avionics reliability and utility. There is, however, one potential weakness that disturbs the critics of avionics integration. They cite the possibility of losing all avionics functions as a consequence of losing one or two system elements. This criticism must be addressed, and the recommended approach must be justified in terms of system operational capability in all failure situations as well as with quantitative reliability analyses that show overall MTBF improvement.

First it must be emphatically stressed that most failures, including multiple failures in redundant channels, do not wipe out the system. Three questions must be answered. They are:

- What failures can wipe out the system?
- What is the probability of such an occurrence?
- What are the backup provisions in the event of such a failure occurrence?

The answer to the first question is that the loss of both computer complexes (Computer and Data Adapter) will disable the entire system. The projected MTBFs of the computer and data adapter are 5000 hours each. Considering that only one half of single data adapter failures are totally disabling, the probability of total system loss in a 3-hour flight, $P_T(t) = P_F(3)$ is

$$P_F \text{ (3 hours)} = .81 \times 10^{-6}$$

Making allowances for combinations of other multiple failures which would contribute to a total system disability, it can be stated that the probability of total system shutdown in a 3-hour flight is about 10^{-6}. Suppose we are being overly optimistic on the projected MTBF and we only achieve one-half the MTBF values specified. Then the $P_F(3)$ rises to 3.24 x 10^{-6}, or, making provision for other disabling failures, the probability of total system shutdown in a 3-hour flight is about 4 x 10^{-6} (or four shutdowns per million flights).

The response to the third question shows that the backup provisions are sufficient to allow continued instrument flight (although not to a Cat. II level). The following is a summary of these backup provisions:

- Both EADIs present horizon displays independent of the computers, and the attitude references are manually selectable from alternate sources.

- Both DDRMIs present ADF or VOR bearing (selectable) and aircraft heading from selectable data sources. The VOR radials are selected through the Mode Select Panel course-select knobs which contain course-reference synchros.

- Provision can be made for a direct interface between the heading references and the NAV receivers and the MFD so that a course line pointing to the azimuth scale would represent the desired flight path (localizer or VOR radial). The aircraft symbol would be displaced from the course line by the course-deviation signal. Thus the MFD reverts to a residual HSI through the use of direct, hard-wired interfaces to the required sensors.

- Manual tuning of NAV receivers is independent of the computer system. DME data to two stations is coupled directly from the DME receivers to the DME readouts on the DDRMI instruments.

- Both EADIs present radio altitude independent of the computer system. Also, the radio altimeter display is independent of the computer system.

- Raw data ILS (localizer and glide slope deviation) is presented on the EADIs' ILS window symbol. Course deviation from VOR radials can also be presented on this display if a course resolver is incorporated in the course-set controller on the MSP.

- Pneumatic altimeters, airspeed indicators and vertical speed indicators may be located on the center instrument panel. A self-contained horizon instrument may also be located on this panel. Another means of providing backup air data would be the use of a low cost, mini-air data computer having only three outputs: altitude, altitude rate, and airspeed. These three outputs can be encoded to provide the word stream needed to drive all air data instruments, following the selection of the backup air data by an appropriate instrument switching arrangement. The backup air data would also provide the required encoding for the aircraft's altitude-reporting function.

- A backup, redundant, hardware yaw damper (with somewhat degraded capability) is included in the flight control electronics. That yaw damper function is independent of the computer system.

This leads to a final observation regarding logistical problems, and a very significant departure from contemporary practice. It would appear that the consolidation of several flight-critical functions within an integrated system would necessitate the requirement that two computer complexes be designated as reliability "dispatch items" by an operating airline. The provisioning of spares on a short-haul route structure would be resisted by airline maintenance policies. Perhaps the minimization of the total number of black boxes would permit the carrying of the spares aboard the aircraft. With advanced fault isolation and maintenance-management techniques inherent in a sophisticated digital system, it might even be possible to consider in-flight repairs using the on-board spares.

SOFTWARE SUMMARY AND CONCLUDING COMMENTS

The system design is organized into a software module grouping with a master executive program that integrates these various modular routines and performs such tasks as timing, system reconfiguring, backup algorithm selection, and part of the monitoring functions. A list of software modules, the estimated time per iteration in an advanced Sperry computer, typical iteration rate requirements and memory storage estimates are given in Table I. The advanced Sperry computer (designated RMM-1) was designed for application in the post-1975 era, and has some extremely high speed and architecture innovations. Add/subtract times range from 350 to 700 nanoseconds and multiply times, including memory access ranges from 1.15 microseconds to 4.2 microseconds (for a floating point multiply). That computer would be provided with a 32K plated wire NDRO memory for this application, but Table I shows that the memory budget is only 17,800 words (not including the integrated test and pre-flight checklist which would be contained in the mass memory [disk] and transferred to the computer resident memory when required). The mass storage requirement is estimated as 8×10^{-6} bits for worldwide logistic data, or 1×10^{-6} bits for regional data only. The disk capability is 10×10^{-6} bits.

A perusual of Table I shows that the advanced computer would be working at less than 10 percent of its available time to complete the entire computation task. An estimate of the computer load using a more contemporary 1974 machine (Sperry 1819B) indicates that the entire task could be done in 70 percent of that machine's available time with memory (main store) consumption of about 26K words. Thus there do not appear to be any serious questions regarding whether the state of the art in avionics can meet the requirements of this type of system. One nagging question persists. Is the software manageable? That is, can such a software system that encompasses so broad a scope of functions, technical disciplines and organizational responsibilities be developed, verified and configuration-controlled in a typical transport aircraft development environment? Fortunately for the author, that question is easily dodged.

The answer is no, if traditional approaches and relationships between participating parties (airframe manufacturers, avionics equipment manufacturers and airlines) are maintained. However, even those digital system pioneers who have survived to regret slogans such as "there are no problems because it's all in the software," will optimistically answer yes if the development environment and responsibilities can be properly disciplined. There is pessimism, however, that industry can achieve that organization and discipline in the near future.

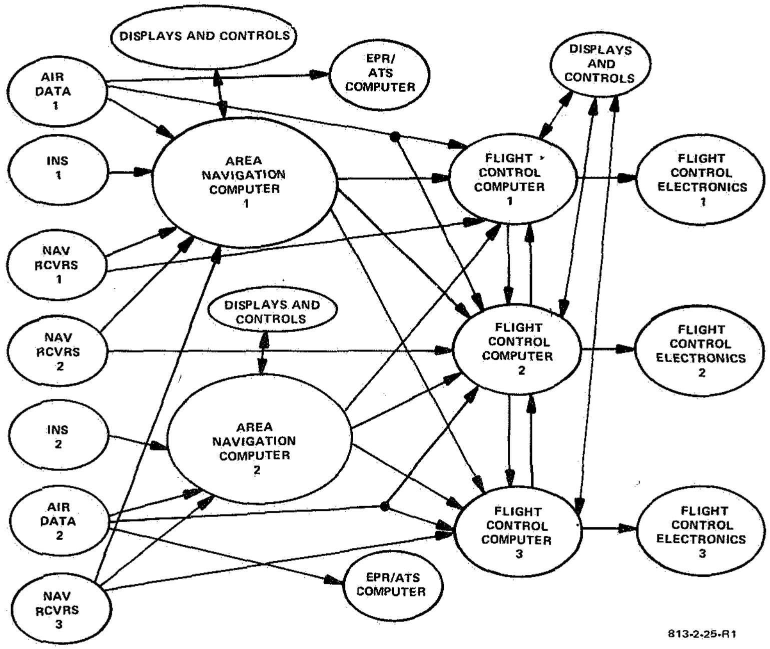

DISPLAYS AND CONTROLS
EPR/ ATS COMPUTER
DISPLAYS AND CONTROLS
AIR DATA 1
INS 1
AREA NAVIGATION COMPUTER 1
FLIGHT CONTROL COMPUTER 1
FLIGHT CONTROL ELECTRONICS 1
NAV RCVRS 1
DISPLAYS AND CONTROLS
NAV RCVRS 2
FLIGHT CONTROL COMPUTER 2
FLIGHT CONTROL ELECTRONICS 2
INS 2
AREA NAVIGATION COMPUTER 2
AIR DATA 2
FLIGHT CONTROL COMPUTER 3
FLIGHT CONTROL ELECTRONICS 3
NAV RCVRS 3
EPR/ATS COMPUTER
813-2-25-R1

Figure 1
Candidate 1, Federated Computer System

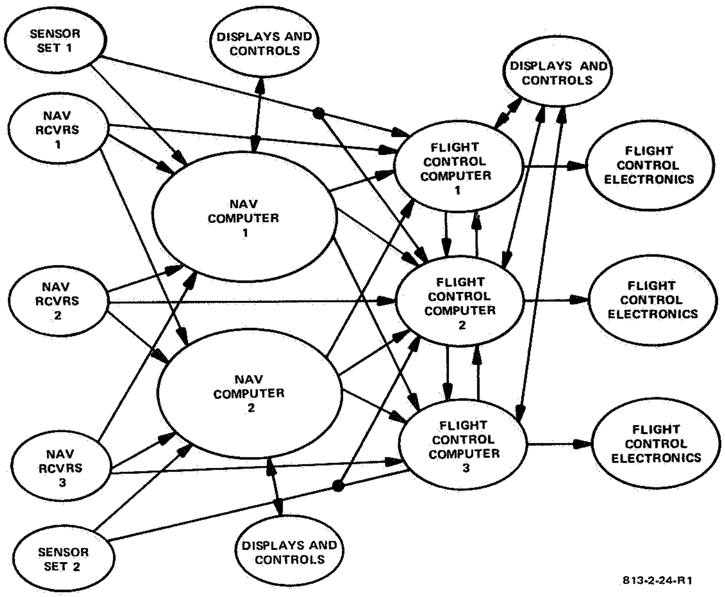


Figure 2
Candidate 2, Dual Navigation Computerization Separate
Fail-Operative Flight Control Computer

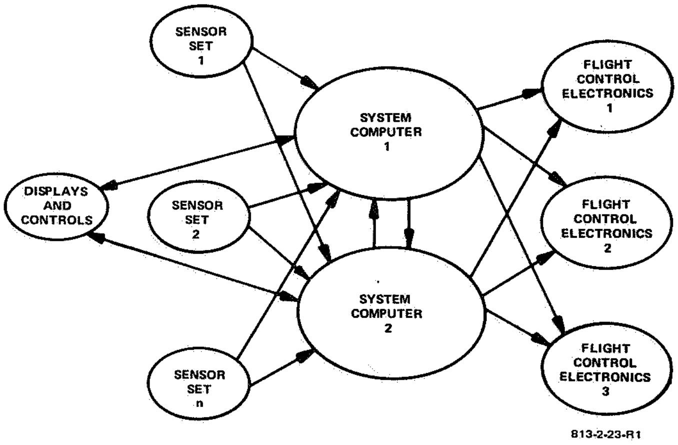


Figure 3
Candidate 3, Integrated Dual Fail-Operative System

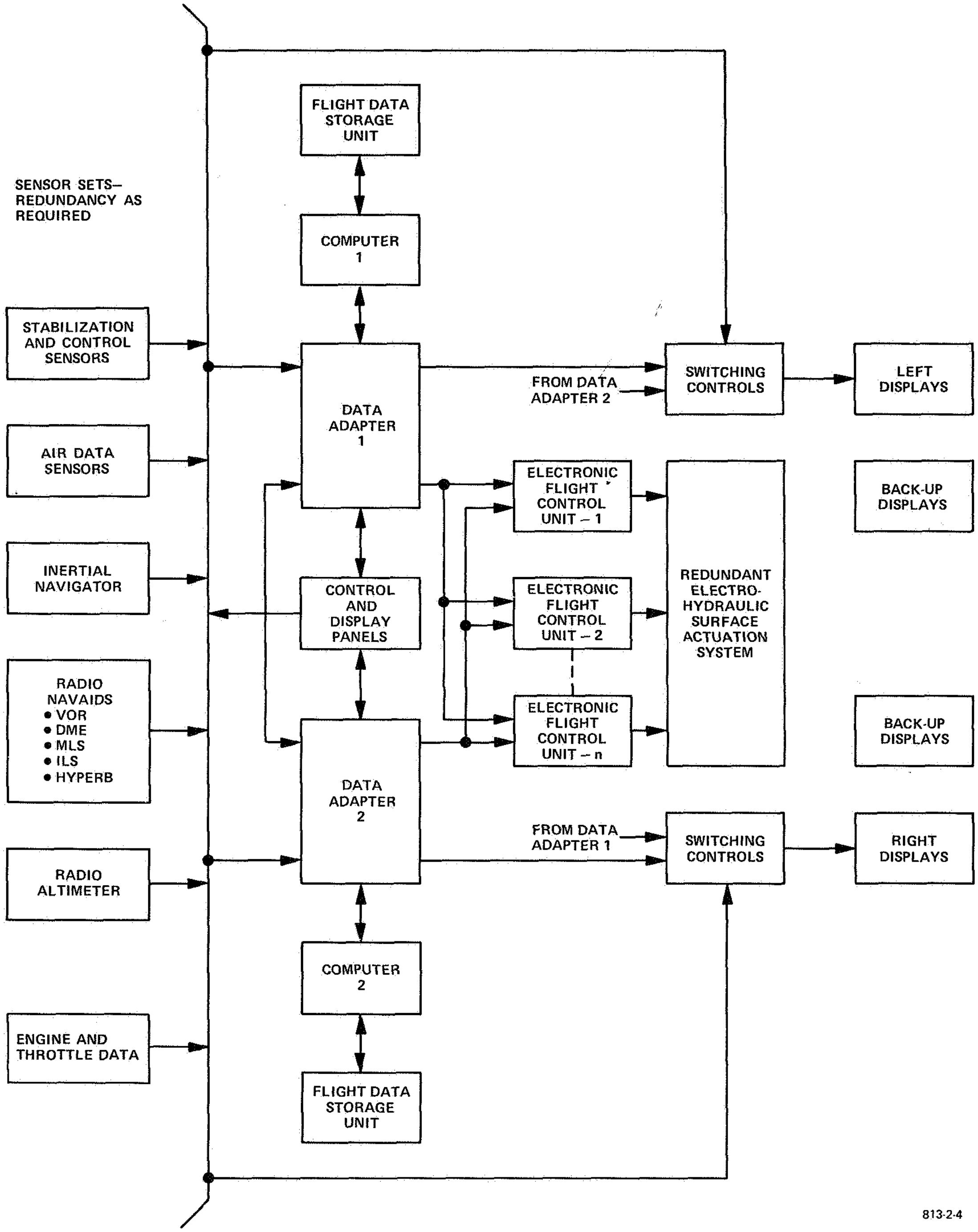
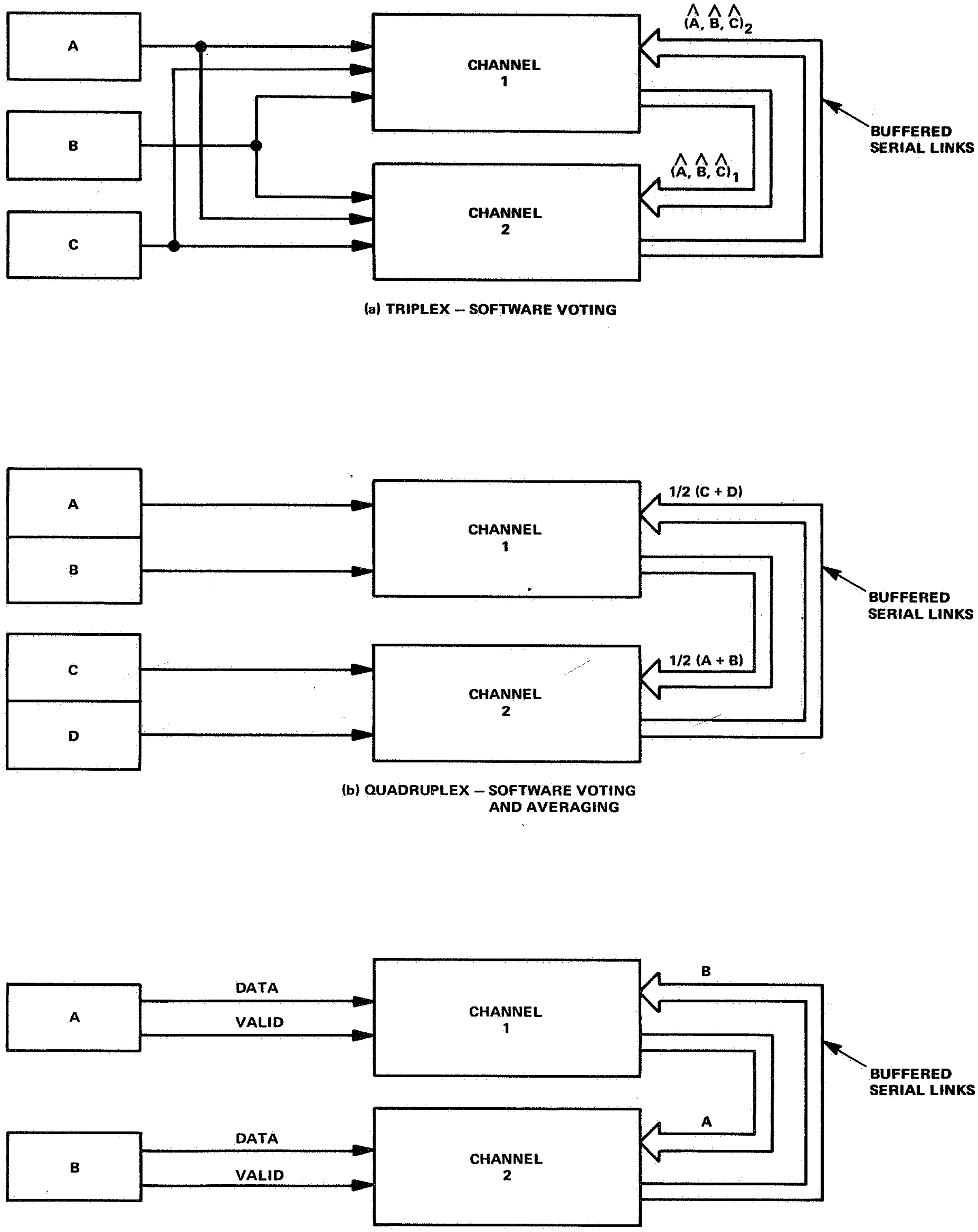


Figure 4
Redundancy Architecture



(a) TRIPLEX – SOFTWARE VOTING

(b) QUADRUPLEX – SOFTWARE VOTING AND AVERAGING

(c) DUAL IN-LINE MONITORING WITH SOFTWARE AVERAGING

Figure 5
Redundancy Schemes for Sensors

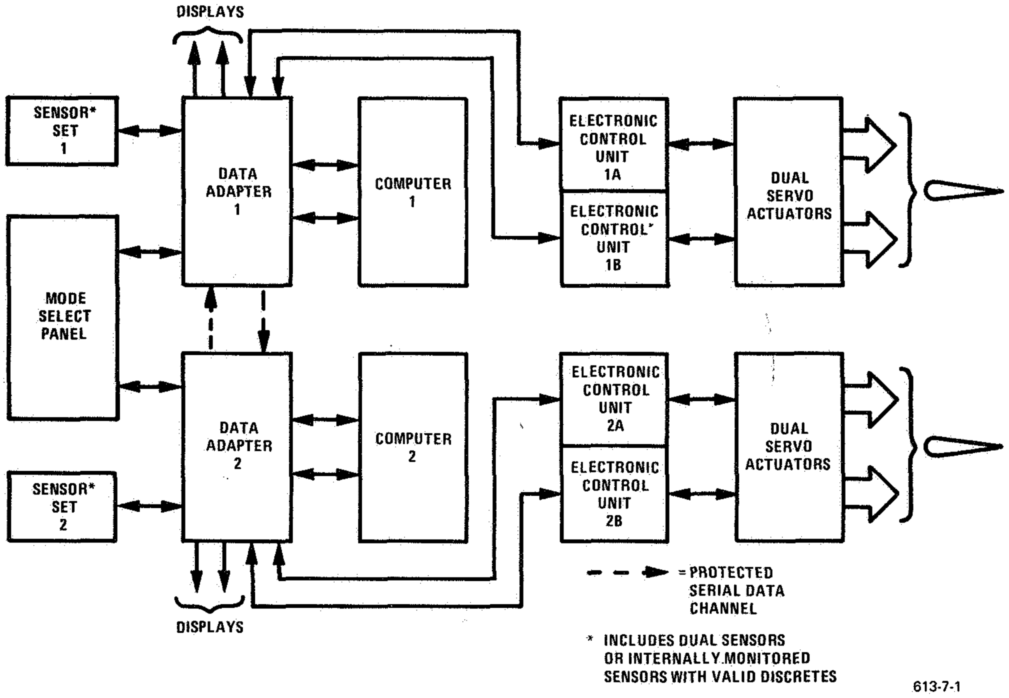

DISPLAYS
SENSOR* SET 1
DATA ADAPTER 1
COMPUTER 1
ELECTRONIC CONTROL UNIT 1A
ELECTRONIC CONTROL UNIT 1B
DUAL SERVO ACTUATORS
MODE SELECT PANEL
DATA ADAPTER 2
COMPUTER 2
ELECTRONIC CONTROL UNIT 2A
ELECTRONIC CONTROL UNIT 2B
DUAL SERVO ACTUATORS
SENSOR* SET 2
DISPLAYS
= PROTECTED SERIAL DATA CHANNEL
* INCLUDES DUAL SENSORS OR INTERNALLY MONITORED SENSORS WITH VALID DISCRETES
613-7-1

Figure 6
Dual Fail-Operative System Architecture

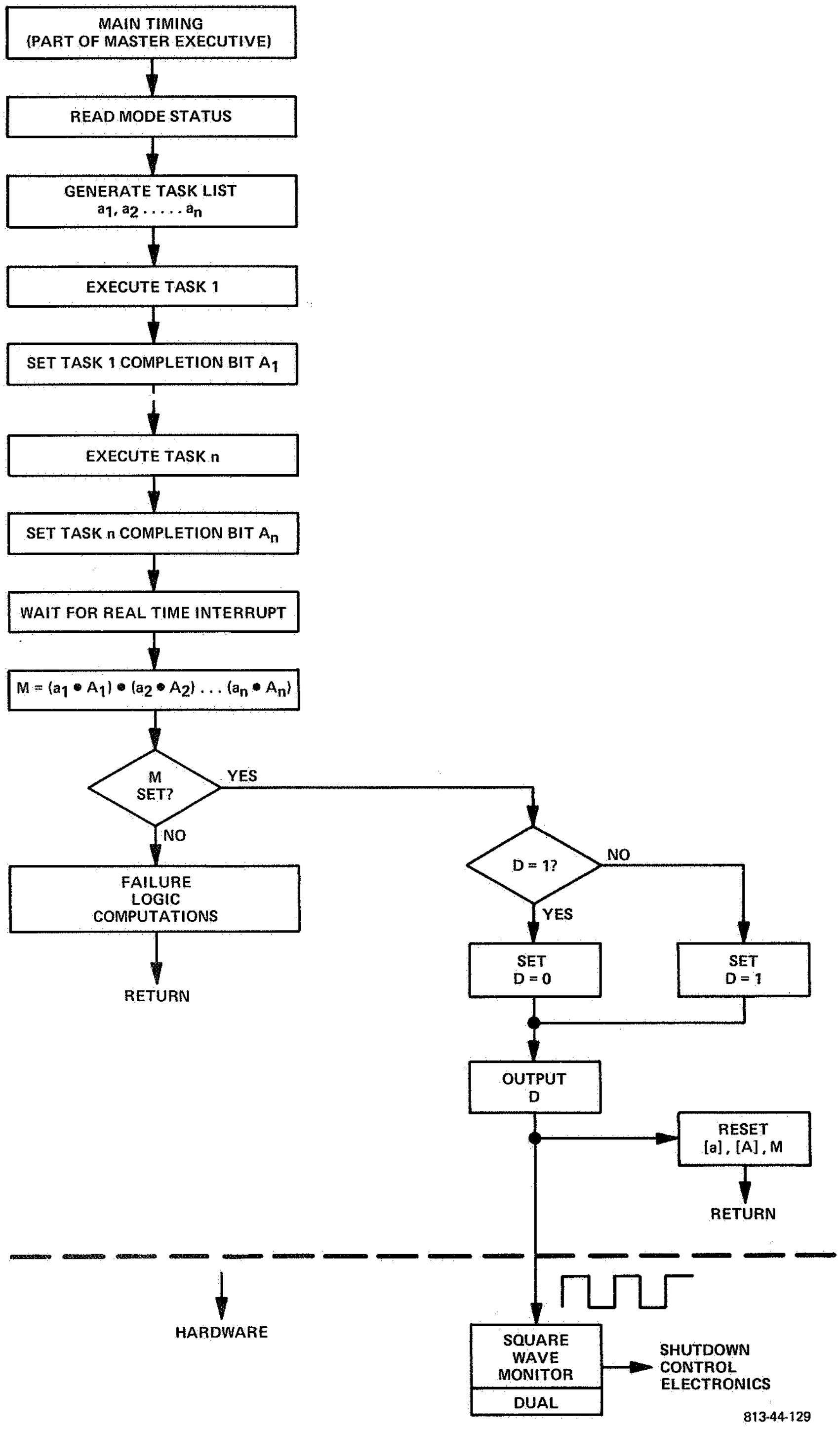

MAIN TIMING
(PART OF MASTER EXECUTIVE)
READ MODE STATUS
GENERATE TASK LIST
$a_1, a_2 \ldots\ldots a_n$
EXECUTE TASK 1
SET TASK 1 COMPLETION BIT A_1
EXECUTE TASK n
SET TASK n COMPLETION BIT A_n
WAIT FOR REAL TIME INTERRUPT
$M = (a_1 \bullet A_1) \bullet (a_2 \bullet A_2) \ldots (a_n \bullet A_n)$
M SET?
YES
NO
FAILURE LOGIC COMPUTATIONS
RETURN
D = 1?
NO
YES
SET D = 0
SET D = 1
OUTPUT D
RESET [a], [A], M
RETURN
HARDWARE
SQUARE WAVE MONITOR
DUAL
SHUTDOWN CONTROL ELECTRONICS
813-44-129

Figure 7
Relationship Between Software Executive Monitor and Hardware Monitor

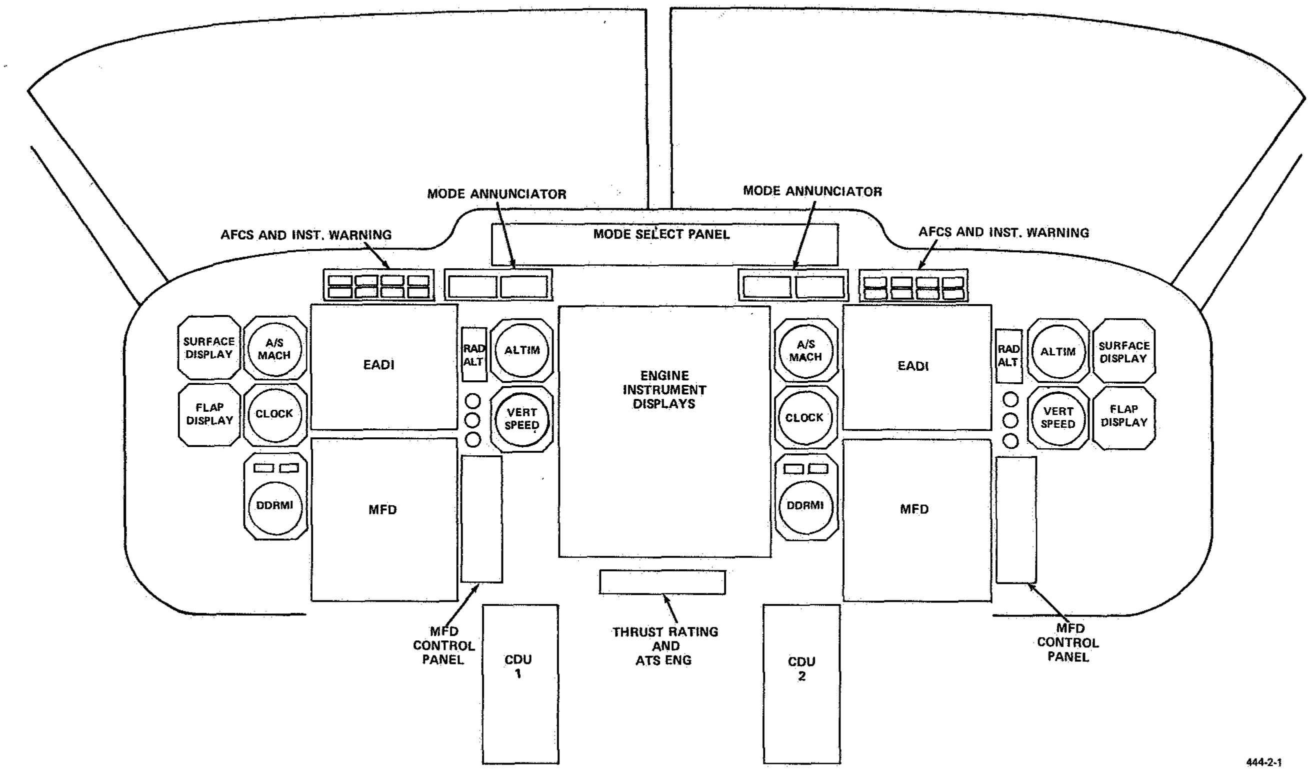


Figure 8
Schematic of Cockpit Display and Control Layout

TABLE I

COMPUTER REQUIREMENTS SUMMARY
(BASED ON RMM-1 COMPUTER)

Function	Typical Time Per Iteration (μ sec)	Required Iteration Rate (per sec)	Memory Storage Requirement (words)
Master Executive	100	1 to 20	1,000
Autopilot/Flight Director Guidance and Stabilization ● Attitude Stabilization ● CWS ● Vertical Guidance ● Lateral Guidance ● Autoland ● Interlocks and Mode Logic ● Panel Communication ● Basic Monitoring	2000	20	5,700
Special Fail-Operative Routines	50 to 700	20	2,000
Navigation ● ρ, θ Nav from Navaids ● Remote Tuning ● State Estimation (filtering) ● Flight Planning (Waypoint Data Processing, Updating, CDU Communication	400	1 to 20	4,000
Air Data Computation $h, \dot{h}, V_c, V_T, M, T_T, T_s, Q_c, P_s$	175	20	800

TABLE I (cont)

COMPUTER REQUIREMENTS SUMMARY
(BASED ON RMM-1 COMPUTER)

Function	Typical Time Per Iteration (μ sec)	Required Iteration Rate (per sec)	Memory Storage Requirement (words)
Autothrottle/Speed Command and Stall Warning (includes α computation)	200	10 to 20	400
EPR/Thrust Rating Computation	125	1 to 5	900
MFD	2,000	1 to 20	3,000
● Communications and Formatting			
● Map Processing			
Integrated Test and Preflight Checklist	-	-	4,000 (Resident in mass storage)
Air Navigation Logistic Data			
● Worldwide	-	-	500,000
● Regional Only	-	-	62,500 (Resident in mass storage)