Flight Test of a Digital Guidance and Control System in a DC-10 Aircraft

S. S. Osder*, D. C. Mossman†, and B. T. Devlin‡
Sperry Flight Systems, Phoenix, Ariz.

A digital guidance and control system flight tested in a DC-10 aircraft in the summer of 1974 confirmed that system's monitoring techniques, certifiable control law performance (including automatic landing), and automated maintenance management concepts. It also verified compatibility of the system architecture with the aircraft's guidance/navigation sensors and redundant electrohydraulic flight control actuators. System organization, in terms of redundancy and software-hardware monitoring, is described. Computer memory and time requirements for the various control modes and other software functions are summarized. Flight results associated with autoland performance and the system's recognition and responses to inserted failures during autoland mode engagement are described.

Introduction

In the summer of 1974, a digital autopilot was flight-tested in a DC-10 aircraft in order to verify system design concepts in a realistic operational environment. The primary areas of interest to be demonstrated were: the compatibility of the digital system architecture with the existing DC-10's redundant sensor and surface actuator configuration; the performance of control law and monitoring software throughout the aircraft's flight regime, with emphasis on the autoland function; and the automation of the total system checkout, with all fault isolation and reporting integrated within the software.

The system, designated as the Digital Flight Guidance and Control System (DFGCS), was mechanized from off-the-shelf equipment that had been used since 1971 in several digital autopilot and digital avionics applications, principally under NASA sponsorship.[1,2] The system was integrated into the existing DC-10 avionics wiring and flight control redundancy concept. The basic DC-10 redundancy architecture involves the so-called dual-dual or quadruplex fail-operative configuration to meet the Category III autoland requirements. In this configuration, each half of the system can operate in the autonomous, fail-passive mode, and when both fail-passive halves are operating (with a small amount of safe cross channel communication), fail-operative performance is attained. The digital flight guidance and control system evaluated in the DC-10 flight tests represented one of these autonomous fail-passive halves of a total fail-operative system. Its unique architectural difference from its predecessor dual-dual analog configuration is that a single, general purpose digital computer provides 100% fail-passive monitoring and fault isolation. This is in contrast to the need for two computers in each half of the conventional dual-dual quadruplex configuration.

Although the system was also flown in the high altitude, higher speed cruise modes, the emphasis of the flight test was on take-off, go-around, approach, and landing modes including automatic ground roll-out. These are the modes in which the critical monitoring, fault isolation, and related safety of flight requirements are most demanding. This paper provides a description of how all sensors, signal acquisition, and distribution circuitry and actuators are completely monitored to the point where faults are isolated, reported, and automatically recorded in computer memory for subsequent maintenance action. The test program was constrained to a short schedule during which a total number of seven flights was accomplished. Of these, three were considered developmental in which minor software changes were made to cope with peculiarities in the wiring interfaces and aircraft dynamics not accurately represented in prior simulator validation tests. The final four flights were considered demonstration flights from which system performance could be evaluated. Specific results from these final four evaluation flights regarding autoland touchdown dispersions and failure detection of inserted sensor and actuator faults are described in this paper.

Presented as Paper 75-567 at the AIAA Digital Avionics System Conference, Boston, Mass., April 2-4, 1975; submitted April 3, 1975; revision received Nov. 4, 1975.

Index categories: Aircraft Handling, Stability and Control; Aircraft Navigation, Communication and Traffic Control; Computer Technology and Computer Simulation Techniques.

*Engineering Manager, Advanced Systems and Avionics Department. Member AIAA.

†Principal Engineer, Advanced Systems and Avionics Department.

‡Staff Engineer, Advanced Systems and Avionics Department.

System Description

System Architecture

The complete fail-operative version of the system organization is illustrated in Fig. 1. The two channels operate autonomously except for a small amount of buffered serial communication needed for mode compatibility and other status reports. The two halves of the total, fail-operative system are designated as channel 1 and channel 2. 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*. 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.

This system architecture is compatible with the basic quad redundant hydro-mechanical surface actuation systems found in recent wide-body transport aircraft such as the L-1011 and DC-10. Specifically for the DC-10 application, each fail-passive half of the fail-operative system is given direct access to one inboard and one opposite outboard surface actuator, but indirectly drives the other two surfaces by mechanical coupling through the manual control links and feel system. Authorities are force limited by the reaction of the autopilot actuator against the feel system and/or manual overpower forces.

The flight test system described in this paper was one autonomous half of the total system depicted in Fig. 1. As

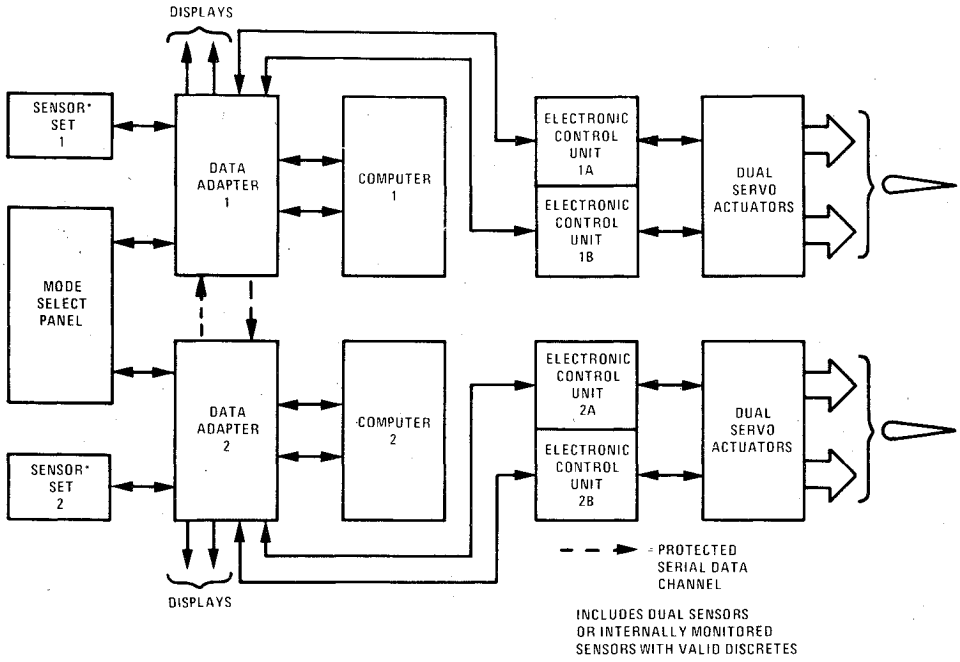

Fig. 1 Dual fail-operative system architecture.

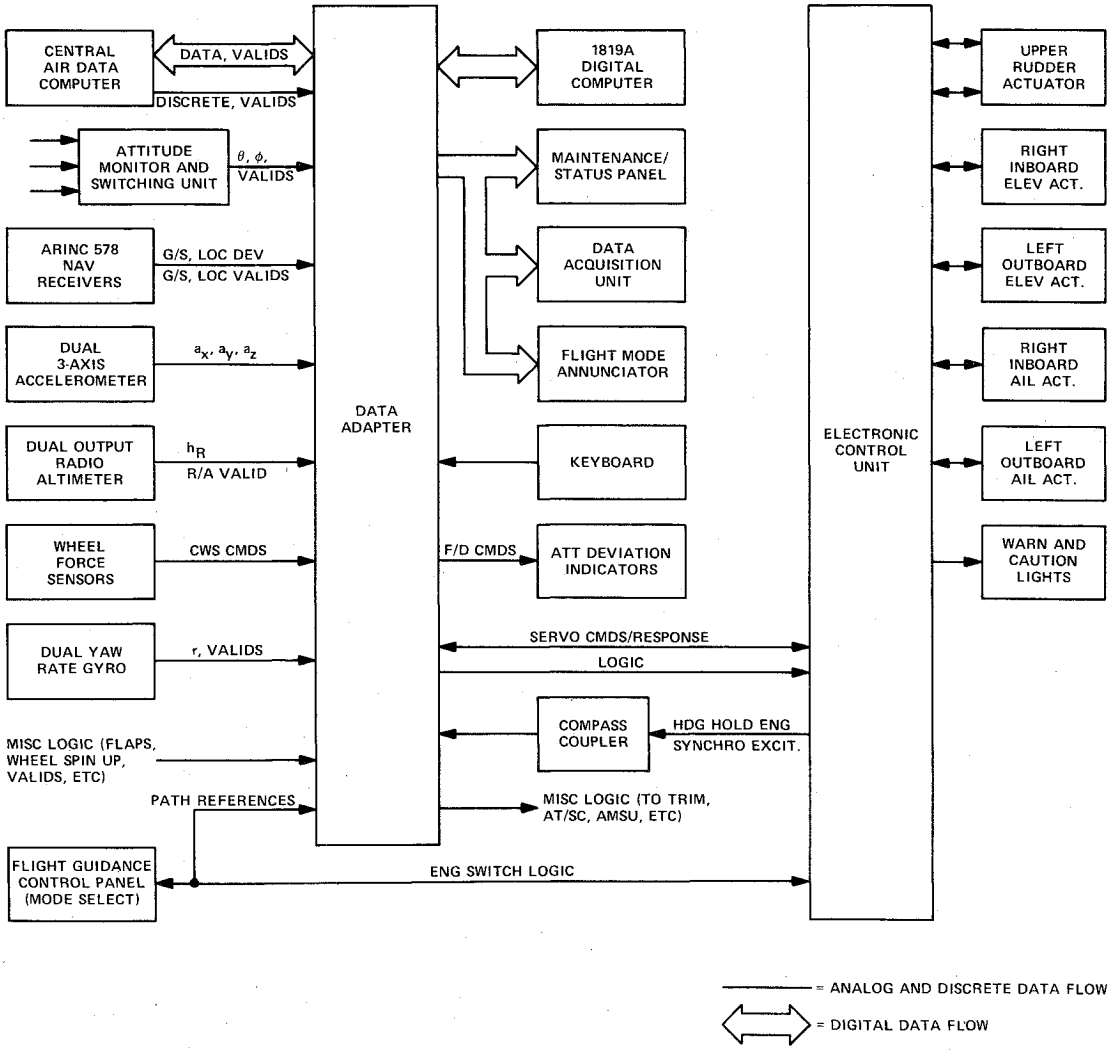

Fig. 2 System block diagram for flight tested functions.

shown, it comprised three main electronic elements: the general purpose digital computer; the data adapter or signal acquisition, conversion and distribution terminal that interfaced the computer with the hardware world; and the electronic control unit that drove the servo actuators and other power loads. The rationale for partitioning the system in this manner has been described in Refs. 1 and 2.

A more detailed block diagram of the system, emphasizing the specific internal and external interfaces, is shown in Fig. 2. Note that this diagram includes only those interfaces involved in the flight evaluations, but the system was mechanized to include the usual complement of automatic guidance modes with interfaces to the appropriate navigation devices and subsystems. Also, the basic system design included the autothrottle/speed command and engine thrust rating computations, but to minimize aircraft wiring and instrument panel modifications, these latter functions were provided by the existing DC-10 subsystems during the flight evaluations.

Equipment Summary

Figure 3 is a photograph of the pallet containing the electronic assemblies used in the flight test. The pallet also included a maintenance/status panel and associated keyboard for communication with the computer, various equipment valid indicators, power switches, and failure insertion switches. The pallet was wired into the DC-10 cabling system through the connectors for the existing DC-10 flight guidance equipment racks. Three of the boxes represent the basic complement of electronic Line Replaceable Units (LRUs) of the system. They are the data adapter (upper right), electronic control unit (upper left), and digital computer (lower right). The box in the lower left is a special power supply added to condition power for various auxiliary panels and is not required for the basic DFGCS. The Sperry 1819A digital computer was used for the flight test, but subsequent work on this system has used the 1819B computer which is twice as fast as the 1819A, one-half the size of the 1819A, but has an upward compatible software system with the 1819A.

The maintenance/status panel and associated keyboard provided English language communications with the total DFGCS. It was the command and display center for a totally automated preflight system checkout (using a program resident in the DFGCS computer) and the means by which the system could be interrogated to present alphanumeric message

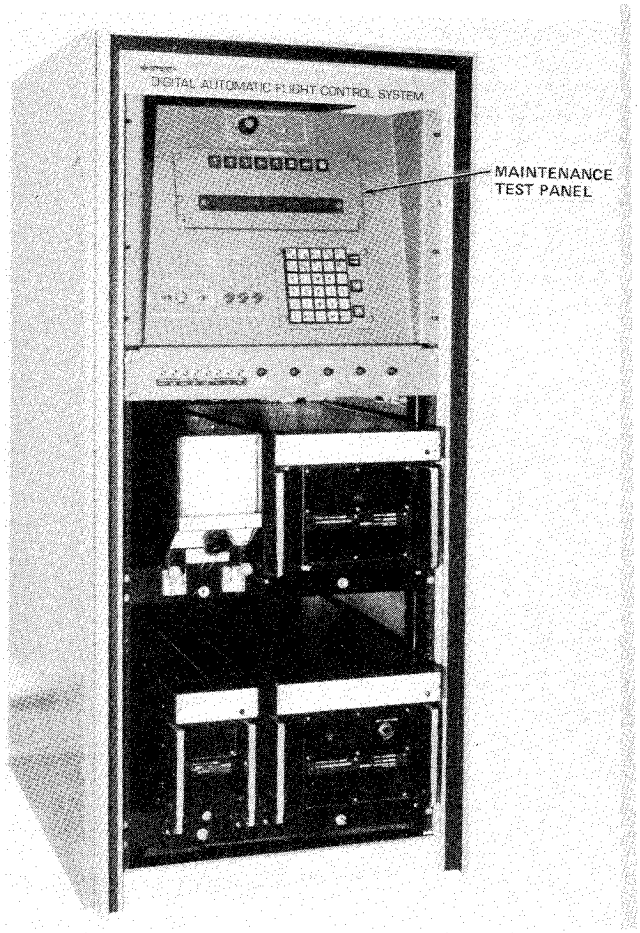

Fig. 3 Airborne equipment pallet.

Table 1 Summary of DFGCS control functions

	Functions		
Control axis	Autopilot	Flight director	Yaw damper
Pitch	CWS/ATT hold alt hold glideslope go-around flare nose lowering [a]altitude preselect [a]altitude alert [a]flight path angle and vertical speed select and hold [a]airspeed hold [a]Mach hold	alt hold glideslope go-around [a]altitude preselect [a]vertical speed, etc. [a]airspeed, Mach	
Roll	CWS/ATT hold/HDG hold HDG select localizer, align, (de-crab) go-around [a]lateral guidance (VOR, INS, area NAV coupling)	HDG select localizer go-around [a]lateral guidance	
Yaw	align (de-crab) rollout		yaw damper turn coordination

[a]Included in basis system design but not flight tested.

summaries of faults which had been identified either during the ground checkout, or had occurred and had been recorded and diagnosed during the flight. A detailed description of the automated testing and failure diagnostic system is given in Ref. 3. The keyboard and maintenance/status panel were also the means by which system parameters (gains, time constants, thresholds, etc.) could be adjusted during flight test.

Other components of interest in the DFGCS are the Flight Mode Annunciator (FMA) and the redundant 3-axis accelerometer package. The FMA was a general purpose alphanumeric display, mounted for expediency above the center of the instrument glare shield. It decoded serial data words transmitted by the computer to display the arm and engage lateral and pitch guidance mode status. Although accelerometers are conventional in contemporary autopilot systems, they are used here to provide a precise, short term inertial contribution to position and velocity state estimation algorithms which include the complete body axis to inertial frame transformation equations.

Summary of Functions and Associated Software

The DFGCS was programmed to perform the complete fail-passive autopilot, flight director, and yaw damper functions. This included preflight tests, maintenance/diagnostic programs, control law computations, logic and monitoring computations, and various display functions. Table 1 is a summary of the functions performed by the system.

A simplified representation of the system's computation flow is shown in Fig. 4. This illustration reveals a hierarchical computation structure significantly different from that found in contemporary analog autopilot systems. Sensor data is screened through the validity and monitoring and state estimation algorithms to produce velocity and position estimates for the guidance equations. This is in contrast to analog systems where filtering and control law computations are implicit in common mechanizations, leading to excessive complexity when fault isolation to distinguish between sensor and electronic failures is attempted. Figure 4 also illustrates the interaction between the monitoring and automated test

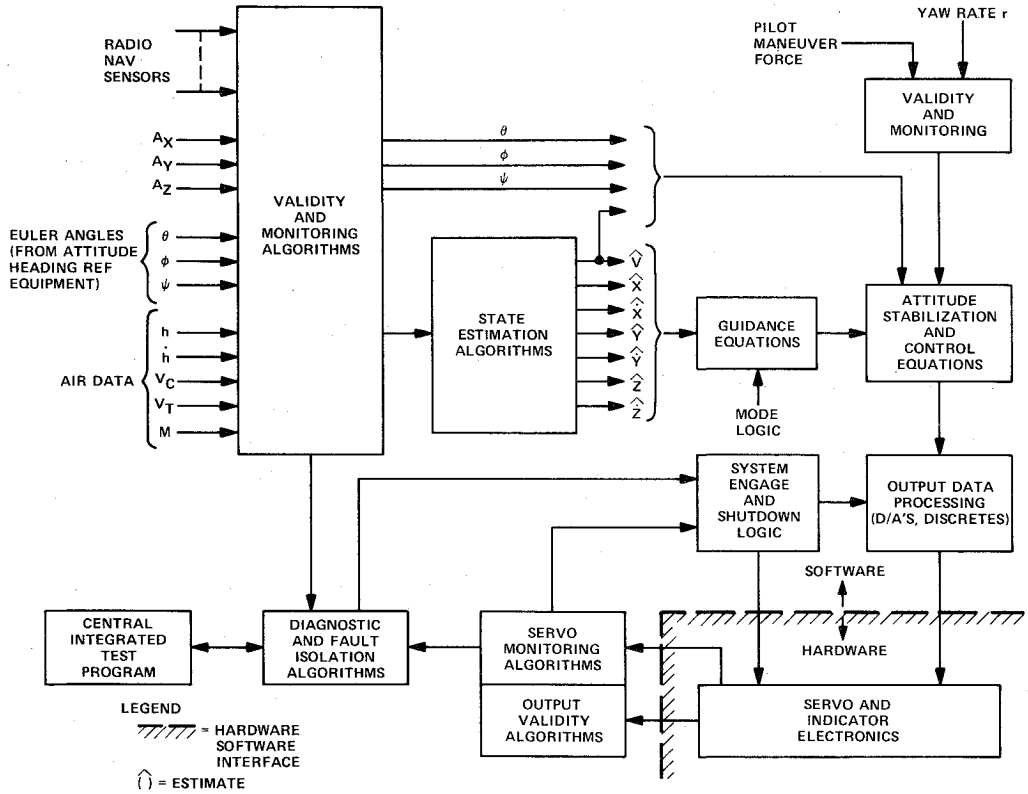

Fig. 4 Simplified computation flow.

Table 2 Program memory utilization summary (18 bit words)

Function	Memory locations: Flight test program in 1819A	Memory locations: Subsequent optimized program in 1819B
automated test maintenance	2,210	2,046
diagnostics		
data acquisition	265	198
main timing executive	350	350
guidance and control (including dual computation)	6,618	3,644
input conversion and scaling	781	701
input discretes	1,045	820
displays	1,124	802
monitoring	1,790	1,090
Total	14,183	9,651

functions since the essence of the automated test is to exercise the existing in-flight monitoring and failure diagnostic algorithms.

The flight test program was implemented using 14,183 words of the 1819A computer's 16K core memory. (This 16K is in addition to about 1K of semiconductor memory used for computer BITE and loading routines.) Table 2 is a summary of memory utilization for various functions. This table also includes a comparative summary of an optimized program that was designed subsequent to the flight test for the same function in the improved 1819B computer. The optimized program involved the elimination of some of the research flexibility incorporated in the original software, improvement in monitoring efficiency, and exploitation of the 1819B computer's improved repertoire. A detailed breakdown of the memory utilization is given in Table 3.

Table 3 covers the original program implementation in the 1819A computer. The worst case, from the viewpoint of computational speed, occurs during glideslope and localizer track with flareout armed with a failure state existing. The 1819A computer's time utilization for this situation was .819 sec/sec (with an iteration rate of 20 per sec or 50 millisec frame time). The same measurement for the program converted to run on the 1819B computer was .378 sec/sec. Table 4 shows an instruction mix and time breakdown for all the major computational blocks with the 1819B computer controlling the

Table 3 Program memory utilization (18 bit words)

Function	Program	Constants	Storage	Subtotal	Total
Monitoring					1,790
Inputs	223	105	200	528	
Servos	250	20	50	320	
End Around	40	--	--	40	
Outputs	500	--	--	500	
Miscellaneous	15	--	--	15	
Failure Reporting	55	332	--	387	
Displays	840	160	124	1,124	1,124
Input Discretes					1,045
Packing and Processing	300	--	45	345	
Mode Engage Interlock	200	--	--	200	
Miscellaneous Valid Processing	500	--	--	500	
Input Conversion and Scaling					781
Input Signal Screening	140	--	32	172	
A/D	21	130	130	281	
Serial	160	30	310	500	
Data Acquisition (for Flight Test)	65	100	100	265	265
Automated Test and Maintenance Diagnostic	1,210	1,000	--	2,210	2,210
Main Timing Executive	280	35	35	350	350
Guidance and Control (Including Redundant Computations)					6,618
*Arithmetic Routines	115	126	9	250	
*Computation Executive	250	--	--	250	
*Initialization	80	--	--	80	
*Roll Inner Loop	100	--	--	100	
*Pitch Inner Loop	80	--	--	80	
*Yaw Damper	65	--	--	65	
*Roll Flight Director	45	--	--	45	
*Pitch Flight Director	55	--	--	55	
*Roll CWS (Control Wheel Steering)	150	--	--	150	
*Pitch CWS	100	--	--	100	
*Low Pass Filters (25)	9	25	25	59	
*High Pass Filters (15)	10	15	30	55	
*Integrators (30)	10	30	60	100	
*Pitch Engage/Arm Computations	500	100	100	700	
*Roll Engage/Arm Computations	430	100	100	630	
*Transformations and Complementary Filters	220	--	--	220	
Air Data Computations	70	--	--	70	
Speed Schedule	110	--	--	110	
Output Storage	18	--	--	18	
Grand Total					14,183

*Redundant

Table 4 1819B computer instruction mix/computation speed summary (DFGCS in glideslope/localizer track mode)

Operations / Computation Routines	Enters	Stores	Adds	Subtracts	Multiplies	Divides	Complements	Reg. Exchange	Reg. Transfers	Calls and Jumps	Compares	Conditional Jumps and Skips	Logicals	Absolute Value	Index Select	Page Select	Flag Instructions	I/O Instructions	Shifts/Rotates n-bits	Total Shift Count	Time/Execution milliseconds	Total Operations
A/D Input Conversion and Scaling	237	105	-	-	52	-	-	-	-	1	-	156	-	-	104	104	-	-	52	312	1.90	811
Sensor Comparison Monitoring	83	103	50	36	36	-	-	24	36	25	25	60	5	36	-	-	-	-	57	257	1.48	576
Serial Input Conversion Scaling and Monitoring	42	28	17	4	1	-	6	-	15	36	7	50	14	-	3	22	-	7	3	12	.49	225
Input Discrete Packing and Processing	41	7	-	-	-	-	5	-	2	12	15	39	8	-	1	10	-	-	49	152	.39	189
D/A, A/D End Around Monitoring	38	26	12	-	-	-	-	-	-	1	-	24	-	12	-	24	-	-	48	360	.43	185
Guidance and Control (One of Two)	384	271	178	88	112	39	24	-	65	211	63	296	48	2	28	-	60	-	83	646	4.79	1952
Output Monitor and Conversion	136	59	20	12	11	-	25	-	11	87	71	177	36	-	-	-	-	-	1	16	1.39	646
Valid Processing	76	1	7	-	-	-	1	-	-	46	-	48	34	-	-	-	30	-	-	-	.48	243
Servo Model Monitoring	27	12	14	10	4	-	-	-	12	13	4	21	1	12	-	-	-	-	4	68	.29	134
Total of Shown	1064	612	298	150	216	39	61	24	141	432	185	871	146	62	136	160	90	7	297	1823	11.64	4991
Total/Cycle (Includes Dual Computation)	1448	883	476	238	328	78	85	24	206	643	248	1167	194	64	164	160	150	7	380	2469	16.43	6943

glideslope and localizer track modes, but in a nonfailure detected state (and consequently not worst case). The frame time is 50 millisec which has been found to be adequate for transport flight control applications. It is seen from this table that the 1819B computer is operating at 6943 operations in 16.43 millisec or 422,580 operations per sec for the instruction mix shown.

Monitoring

Several different monitoring techniques were used to achieve 100% failure detection and thereby permit safe shutdown of the fail-passive channel. The following is a summary of the fault detection techniques that were employed: 1) processing of sensor valid discretes; 2) sensor data validity and reasonableness checking algorithms; 3) sensor data voting and comparison monitoring-variable thresholds dependent upon aircraft state, signal amplitude, and signal duration; 4) redundant computations internal to the computer using separate computer memory banks and comparison checks of results; 5) memory check sums for nonredundant programs; 6) end around I/O checking-all outputs are fed back to the computer via the input conversion sections and verified against the specified output; 7) test words continuously checked for all intrasystem communications; 8) model and comparison monitoring of servo actuator responses; 9) software executive continuously verifies that the required sequence of software tasks is accomplished each 50-millisec iteration period; 10) external (to computer), dual hardware monitors examine the computer's output for a required dynamic signal pattern (computer heartbeat)—any computer failure that will prevent the execution of the specified program will cause the pattern to cease.

Input and Computation Monitoring

A simplified representation of data and computation flow is illustrated in Fig. 5. This figure emphasizes the handling of data from one specific sensor pair whose signals are designated X_{iA} and X_{iB} where A and B represent the physical separation of data into the A and B internal subchannels. Note that the A and B channelization begins prior to the A/D converter at the data adapter connector pins. After conversion and transmission to the computer, the A and B raw sensor measurements are stored in two separate memory locations (LOC A and LOC B). The necessary filtering and scaling is done in the software and then a comparison between the A and B numbers is made. This comparison is made in the software element designated comparison threshold, but it is noted that this threshold expands or contracts as a function of the estimated value of the parameter, $\hat{X}$ (obtained from previous iteration averages of subchannel A and B values), the maneuvering state of the aircraft and the mode logic. If the difference signal is below the threshold level, both sensor

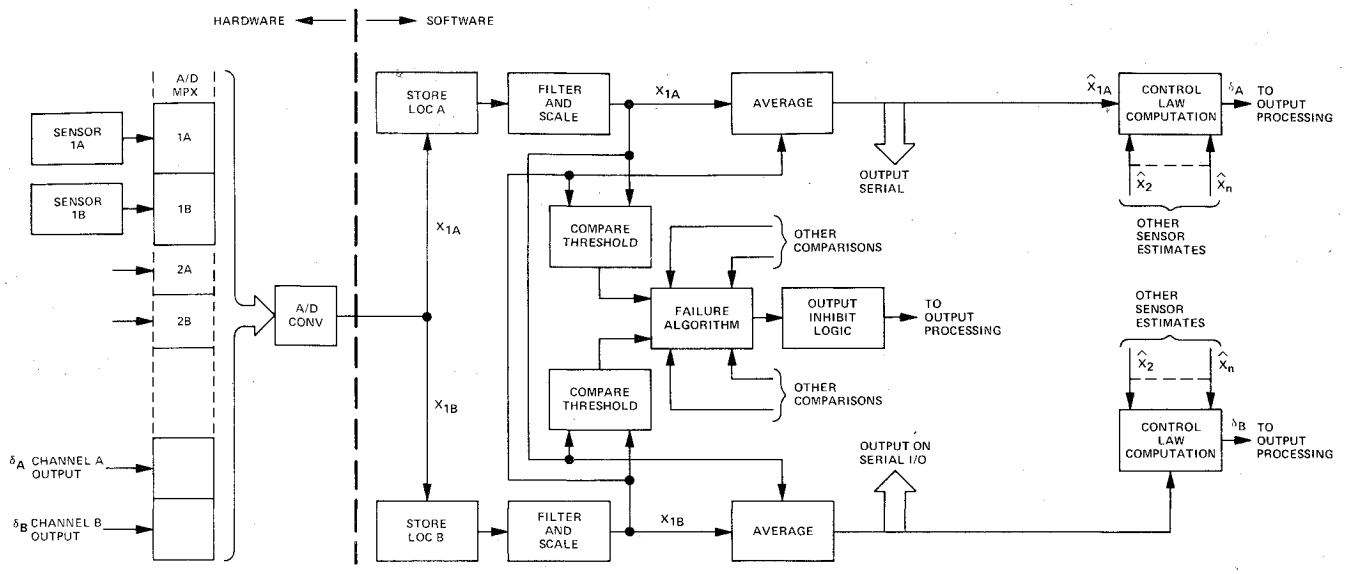

Fig. 5 Input and computation flow.

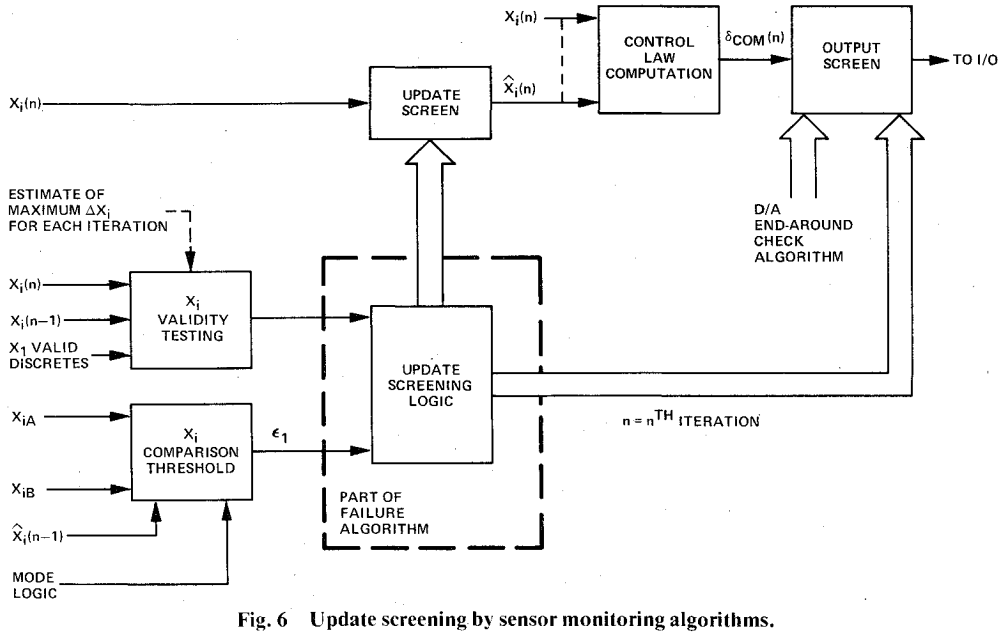

Fig. 6 Update screening by sensor monitoring algorithms.

signals are used as inputs to each of the dual computations. In each channel, the average of the two signals is computed and used as the effective sensor input signal. When the differense signal exceeds the threshold level, the most recent valid signals are used for the DFGCS computation. The sensor error flag, which triggers a shutdown routine further downstream and also identifies the failed sensor is not set until the difference signal has remained above the threshold level for a variable period of time.

Stated mathematically (using continuous, frequency domain notation for convenience, the sensor monitoring algorithm is

$$X_i = 1/2\ (X_{iA} + X_{iB}) \left(\frac{1}{\tau S + 1}\right) \tag{1}$$

$$\epsilon_i = \text{Error magnitude} = |X_{iA} - X_{iB}| \frac{1}{\tau S + 1} \tag{2}$$

$$\Delta_i = X_i \text{Threshold} = \text{Max}\,[K_1,\ (K_2 + K_3 |\hat{X}_i|)\,] \tag{3}$$

when

$$\epsilon_i \geq \Delta_i,\ \epsilon^* = \epsilon_i \tag{4}$$

when

$$\epsilon_i < \Delta_i,\ \epsilon^* = 0 \tag{5}$$

The response to the state, $\epsilon_i \geq \Delta_i$ is: new values of X_i are screened from the control law computations; and failure computation proceeds as follows

$$\int_0^{t_F} \epsilon_i^* dt \geq T_{M_i} \ ? \tag{6}$$

where T_{M_i} is a time magnitude threshold for sensor i. As a specific example, in the DFGCS, the constants used for localizer signal monitoring were: $k_1 = 7.725\ \mu$A, $k_2 = 0$, $k_3 = .1$, $\tau = .1$ sec, and $T_M = 14.6775\ \mu$A-sec. These constants were empirically established from known gradient and null tolerances specified for each specific sensor. This very tight threshold is possible in the case of the localizer receiver because both signals are derived from a common receiver with dual output channels. Note also that the filter time constant τ

used in the monitor algorithms is not necessarily the same value used in creating an averaged estimate for control law computation.

Comparison monitors are not the only means used to diagnose and screen out faulty sensor data. The general approach to the screening algorithms is illustrated in Fig. 6. This figure shows that sensor valids as well as a sensor's change-of-state characteristics between sample intervals also contribute to the input data validity checking. Each sensor is treated in accordance with our prior knowledge of its expected signal characteristics. Thus, a localizer or glideslope signal is constrained to maximum changes between computation iterations, and consistent departures from these maxima are used to indicate anomalous performance. The advantage of this update screening technique is that a detected discrepancy in the input data is not allowed to cause an erroneous output command, but also does not require a disengagement which could be a nuisance disengagement if the discrepancy were the result of some noise process.

End-Around Checking

The elements of the end-around checking are illustrated in Fig. 5. The output commands of each subchannel *A* and *B* are fed back to specific *A/D* multiplexer switches so that the computer can examine whether its commanded output has been achieved by the data adapter's *D/A* converters. Two types of malfunctions can cause a failure to verify a *D/A* output against the computer's commanded output value: a *D/A* failure or an *A/D* failure. If the *A/D* converter had failed, a large number of discrepancies would have resulted since that converter is multiplexed for all analog inputs. If only one discrepancy occurs in the end-around check algorithm, then it is generally indicative of a specific *D/A* circuit failure. The fault isolation algorithm will produce the proper diagnostic, and in the case of a *D/A* failure, only the faulted function may be shutdown.

Computer Executive and Hardware Monitor

The computer system verification function generates 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 interwoven throughout the entire program, so 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 ECU where it is compared with a correct signal pattern. A difference in these signals will cause the DFGCS 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 hardware monitor is insensitive to software changes provided the system iteration rate is unchanged.

The computer system verification function serves principally to detect massive computer failures and does not allow shutdown of partial DFGCS 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 Fig. 7. In this figure, the concept of an executive program which generates a task list as a function of the mode 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. Note that individual routines such as the Guidance and Control block on Fig. 7 contain their own sub-executives which also employ task generation lists and task completion indicators. Following the real-time interrupt that controls the program iteration rate, the first subroutine performed is a check to determine that all specified tasks were performed. This validation routine also checks dual computation comparisons, status of memory check sums against the correct value, and other indications of computation anomalies. Recognition of a computation failure causes a jump to a failure response routine. This jump will also result in the disruption in the generation of the dynamic signal pattern or computer heart-

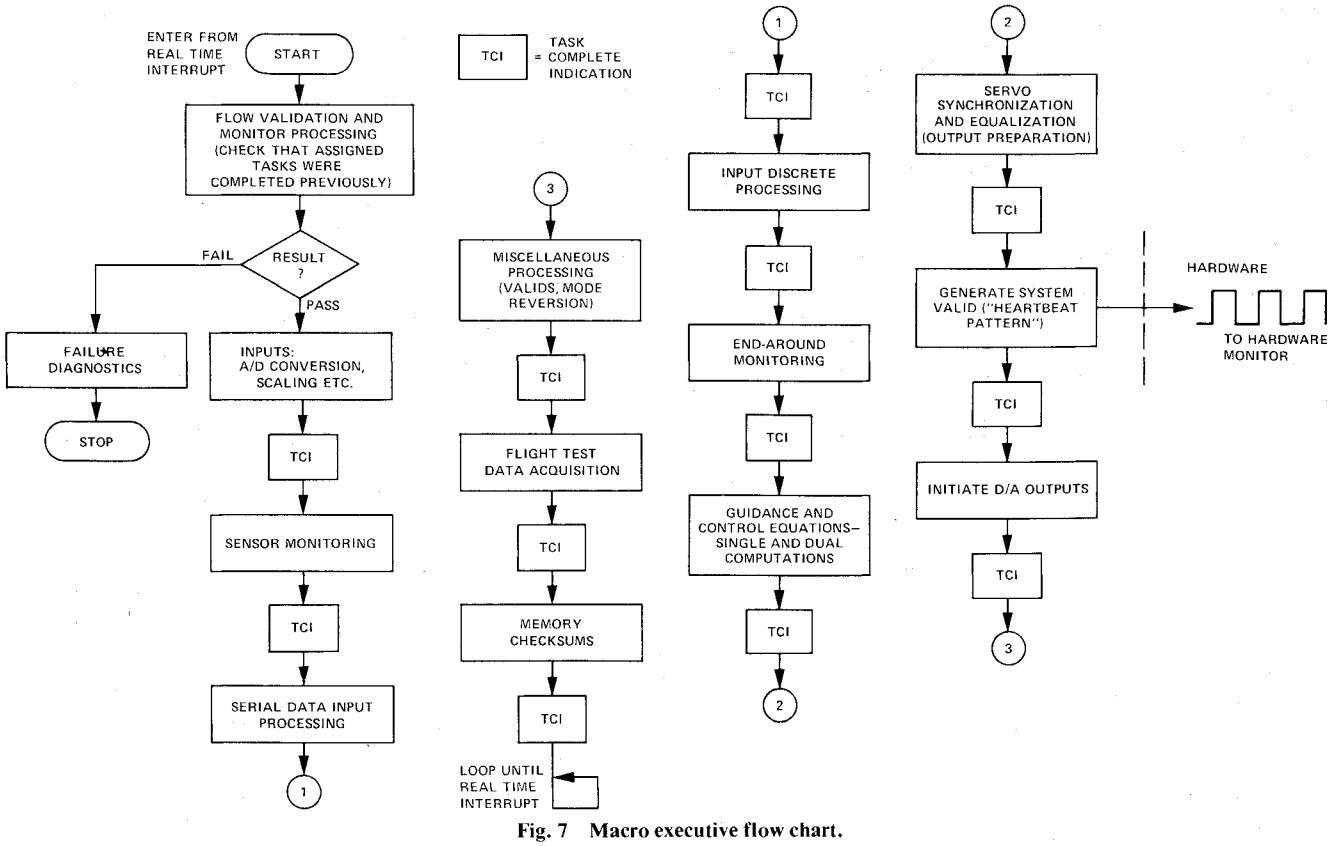

Fig. 7 Macro executive flow chart.

Table 5 Failure insertion summary

Failure insertion (10-sec duration)	Trip	Remarks
open localizer *A* (1000 ft)	no	insufficient error (LOC track)
open glideslope *A* (1000 ft)	no	insufficient error (G/S track)
open pitch attitude *A* (1000 ft)	yes	tripped in one second (G/S track)
open roll attitude *A* (1000 ft)	no	insufficient error (LOC track)
open roll attitude *A* (1500 ft)	yes	tripped in one second (LOC capture
open longitudinal acceleration (1000 ft)	yes	tripped due to G/S beam bend (G/S track)
open lateral acceleration *A* (1000 ft)	no	insufficient error (LOC track)
open normal acceleration *A* (1000 ft)	no	insufficient error (G/S track/LOC track)
shorted elevator RAM *A* LVDT (1000 ft)	yes	tripped less than two seconds
shorted aileron RAM *A* LVDT (1000 ft)	yes	tripped less than two seconds
shortd rudder RAM *A* LVDT (1000 ft)	yes	yaw damper inoperative light in about one second
failed No. 1 hydraulics (100 ft)	yes	elevator servo fail at flare–manual landing OK
failed No. 3 hydraulics (50 ft)	yes	elevator servo fail at 10 ft–manual takeover OK

beat. In this case, both the software and hardware monitors will detect the failure, but the hardware monitor will require a few cycles of incorrect output before it will respond.

Failures of the digital computer's logic circuitry associated with the execution of specific instructions are the types that 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 instruction. 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 detects a fixed state on its output heartbeat rather than the required dynamic pattern. It thereby indicates a system shutdown by commanding a computer power-down and interruption of power to *D/A* outputs.

Flight Test Results

The flight test program consisted of three development flights and four demonstration flights. Automatic approach and landing development were conducted at Palmdale, and demonstrations were at Yuma, Palmdale, and Stockton. Additional data were obtained by conducting an automatic approach and landing on the return to Long Beach after each flight.

Demonstration flights 2 and 3 were used to gather autoland touchdown dispersion data for worst case cg conditions, and to demonstrate the effect of high altitude failure insertions. Demonstrations flight 4 was used primarily for low altitude failure insertion and off-nominal condition testing.

Failure Insertion Summary

The results of the failure insertion testing are summarized in Table 5. As shown in the table, simulated passively failed sensors were not detected unless an error was present during the 10-sec interval that the failure was applied. This result is consistent with past experience in analog systems. No physical tracer signals or similar techniques were used to detect the simulated open failures. The detection technique was software comparison monitoring between the *A* and *B* channels. The worst case situation was represented by these tests because the failures were inserted after very tight, zero error tracking of the ILS beams had been established. Moreover, they were inserted in regions where the beams were unusually straight. In prior simulator studies using ILS beam bend models, the beam noise was found to be a sufficient stimulus to the variable threshold, time dependent comparison monitor algorithms. In the limited flight time available, experimentation with detection thresholds was not warranted and the question of passive failure detection obviously remains to be resolved.

Table 6 Off-nominal performance summary

Condition	Result
flaps 35→50 at 1000 ft	no deviation from beam
1.3 V_s + 15 landing, flaps 35 (overspeed)	good landing ($\dot{h} = -3.2$, $X = 694$)
1.3 V_s + 0 landing, flaps 50 (underspeed)	good landing ($\dot{h} = -3.5$, $X = +212$)
pitch trim inoperative landing	good landing ($\dot{h} = -4$, $X = +20$)
pitch trim inoperative, one deg. aircraft nose down mistrim	good landing ($\dot{h} = -2.4$, $X = +597$)
upwind course split at align (yaw rate reversion mode)	good landing ($Y = 10$L) - ground roll OK
downwind course split at align (yaw rate reversion mode)	good landing ($Y = 0$) - pilot disconnect at ground roll engage

Two aspects of the passive failure problem require additional analysis and verification before a design decision can be made. The first involves the dual signal voting technique. In the flight test system, the ILS beam signal used for guidance law computation was the average of channels *A* and *B*. Thus, opening a sensor channel caused the system to operate at half gain as long as the error was near zero. The system can perform quite adequately at half gain in all but the worst wind shear conditions. If the voting technique were changed for critical modes to minimum value rather than average of channel *A* and *B*, then an open failure would result in zero gain, with a resulting buildup error which is more easily detected by the comparison monitor algorithms.

The second aspect involves probability of failure considerations. The condition that eluded the monitor was an open wire failure which had to occur after the beam capture maneuver on a quiet beam and in nonturbulent, nonwind shear conditions. In such conditions, a good landing could have been accomplished in the half gain state, but does the probability of this specific set of circumstances warrant additional failure detection features? This is the essence of the autoland safety of flight problem, and in the specific circumstances illustrated by these flight test results, the final design solution awaits the additional analyses.

Off-Nominal Performance Summary

The results of the off-nominal performance are shown in Table 6. All results were excellent although complete evaluation of the ground roll reversion mode was not accomplished. In this mode, the loss of the heading reference for runway alignment causes a computation of pseudo-heading from the yaw rate gyros. At the initiation of ground

roll, the pilot disconnected. Subsequent analysis of the flight recordings did not indicate any incorrect control commands. The reversion mode had not previously been checked and since this was the last flight of the program, the existence of any problems in the reversion mode ground rollout guidance equation has not been confirmed.

Autoland Dispersion Summary

Autoland dispersion data for worst case cg locations were recorded on demonstration flights 2 and 3 at Palmdale and Stockton. An aft cg location existed on flight 2 and forward cg on flight 3. A summary of the results from these two flights is presented in Table 7. The flareout control law was set for a firm nominal touchdown vertical speed and no attempt was made to adjust the flareout law for other nominal final sink rates. It is noted, however, that touchdown sink rates of 4 to 5 ft/sec in the DC-10 aircraft appear quite soft and rates of about 3.5 ft/sec can barely be sensed in the passenger compartment. Standard deviations are pessimistic since only worst case cg locations were exercised, and the Palmdale and

Table 7 Autoland dispersion data for demonstration flights 2 and 3

Flight	Approach	Runway	GW (10^3 lb)	cg (%)	Headwind (kt)	Crosswind (kt)	$X_{T/D}-X_{G/S}$ (ft)	$\dot{h}_{T/D}$ (ft/sec)	$Y_{T/D}$ (ft)
2	1	PMD/25	366.1	26.0	- 5.0	+ 6.7	+151	-3.1	- 8.1
2	2	PMD/25	361.0	25.8	- 1.6	+ 4.7	+342	-4.0	- 6.0
2	3	PMD/25	357.2	25.6	- 3.8	+ 4.7	+494	-3.2	+ 3.5
2	4	PMD/25	353.1	25.2	- 5.0	+ 6.2	+497	-3.6	+11.2
2	5	PMD/25	347.3	24.7	- 3.1	+ 3.9	+ 19	-2.4	+ 7.5
2	6	PMD/25	344.3	24.5	- 4.3	+ 2.6	+274	-4.5	- 7.6
2	8	SCK/29	325.1	23.8	+ 3.3	+ 3.8	-191	-3.7	-24.3
2	9	SCK/29	323.3	23.7	+ 3.5	+ 1.9	+ 93	-4.0	-10.5
2	10	SCK/29	316.4	23.4	+ 1.3	+ 1.5	+364	-2.0	-14.2
2	11	SCK/29	312.5	23.0	+ 5.2	+ 2.9	+539	-3.5	- 7.4
2	12	SCK/29	308.0	23.0	+ 5.4	+ 4.4	- 84	-2.7	+14.0
2	13	SCK/29	303.4	23.0	+ 2.6	+ 4.3	- 84	-3.8	+12.0
3	3	PMD/25	353.4	13.2	+14.1	- 5.1	- 16	-4.3	+ .4
3	4	PMD/25	348.7	13.2	+19.9	- 3.5	-420	-7.5	-14.6
3	6	PMD/25	339.5	12.5	+16.9	- 6.2	-378	-4.5	- 5.0
3	7	PMD/25	335.0	12.2	+14.9	- 2.6	+ 64	-2.7	+15.9
3	8	PMD/25	330.0	11.8	+15.3	-12.9	-283	-2.8	+24.3
3	9	PMD/25	325.8	11.4	+18.8	- 6.9	-120	-3.9	+11.9
3	10	PMD/25	321.1	11.0	+17.3	-10	-304	-2.8	-17.9
3	11	PMD/25	316.8	10.8	+15.3	-12.9	-416	-3.5	+ 5.4
3	12	PMD/25	312.8	10.5	+15.3	-12.9	-261	-4.0	+16.3
					Mean		+13.3 (+35)*	-3.6 (-3.45)*	-1.3
					STD Deviation		304 (295)*	1.1 (.7)*	12.8

*Asterisked values are results if the worst case landing is removed from the data (approach 4, flight 3)

Stockton beams are not of Category II quality. Nevertheless, the results are well within the FAA certification requirements.[4] The landing on approach 4 at Palmdale on flight 3 was short and firm (about 7.5 ft/sec), apparently due to environmental conditions; however, the flight test recordings indicated that the elevator surfaces were near their authority limits in the nose up direction from 20 ft to touchdown. The recordings indicate that a higher flare error gain would have helped in this case, since it would have commanded full authority at a higher altitude. The conclusion reached from reviewing the complete flight test data was that a higher gain on low altitude glideslope and on the flare error gain was desirable to improve the consistency of the flare entry and of the touchdown dispersions.

References

[1] Hansen, Q. M., Young, L. S., Rouse, W. E., and Osder, S. S., "Development of STOLAND, A Versatile Navigation, Guidance and Control System," AIAA 4th Aircraft Design, Flight Test and Operation Meeting, AIAA Paper 72-789 (also NASA TMX-62, 183), August 1972.

[2] Rouse, W. E., "A Universal Digital Autopilot and Integrated Avionics System," WESCON Technical Papers, Vol. 16, 1972.

[3] Devlin, B. T., Mossman, D. C., and Urling, H. R., "Automated Avionics System Checkout and Monitoring in a Flight Test Environment," Society of Flight Test Engineers, Fifth Annual Symposium Proceedings, 1974.

[4] FAA Advisory Circular on Automatic Landing Systems, AC 20-57A, Jan. 1971, Paragraph 5.b.3.