NASA TECHNICAL
MEMORANDUM

NASA TM X-73,223

NASA TM-73,223

FLIGHT TEST RESULTS OF THE STRAPDOWN HEXAD INERTIAL REFERENCE UNIT (SIRU)

VOLUME II: TEST REPORT

Ronald J. Hruby and William S. Bjorkman

Ames Research Center
Moffett Field, Calif. 94035

July 1977

The individuals who contributed to the preparation of this report are:

Ames Research Center
Ronald J. Hruby, Test Engineer

Analytical Mechanics Associates
William S. Bjorkman
Frederick I. Mann
Stanley F. Schmidt

Charles Stark Draper Laboratory
Robert A. Booth
David Brown
Thomas T. Chien
Jerold P. Gilmore
Richard McKern
Howard Musoff
Thomas P. Schamp
Thomas K. Shuck

Stanford University Digital Systems Laboratory
Roy C. Ogus
Peter A. Thompson

University of Southern Colorado
Ralph A. Carestia
Robert B. Jenkins

FLIGHT TEST RESULTS OF THE STRAPDOWN HEXAD INERTIAL REFERENCE UNIT (SIRU)

VOLUME II - TEST REPORT

Ames Research Center

SUMMARY

A flight test program was conducted under sponsorship of NASA-Ames Research Center to evaluate the performance of the redundant, modularized, fault-tolerant Strapdown Inertial Reference Unit (SIRU) in a short-haul aircraft environment. The SIRU system tested was originally developed as an advanced navigation system for post-Apollo programs. The principal flight test objectives included assessment of:

1. SIRU performance as an unaided inertial navigation system.
2. Capability of the system redundancy management software to detect sensor failures.
3. Flight performance of the SIRU dual-computer configuration

The results of 15 separate flights showed that during cruise the error growth was less than 3 n. mi./hr. During terminal area maneuvers, the errors were less than 5 n. mi./hr Additional analysis, design, and testing would reduce these errors. Updating the inertial data with Distance Measuring Equipment (DME) readings would enable the SIRU system to provide a position accuracy better than 0.2 n. mi.

During the flight tests, no gyro failures smaller than 1.5°/hr were detected because the SIRU software detection threshold was adjusted upward to prevent excessive false alarms. These levels were approximately twenty times larger than gyro drift detected in the laboratory. Thus, the SIRU software could detect and isolate only those gyro failures that were "hard failures" from a navigation point of view. The isolatable failures are "soft failures" from the point of view of flight control applications. The fault detection and isolation algorithms require more research for the overall SIRU concepts to be acceptable for operational aircraft use.

The SIRU dual-computer system provided both reliable and correct information. However, several design improvements were suggested to improve the overall reliability and maintainability of a future system.

INTRODUCTION

The performance goals for economically viable short-haul aircraft require sophisticated powered-lift and aerodynamic high-lift devices to increase payloads and reduce takeoff and landing distances The application of these high-lift devices, coupled with increasingly complicated power-control and vehicle-control techniques, has produced:

1. A proliferation of control sensors, actuators and display devices for stabilizing and flying the aircraft in its various flight modes.

2. Steep, curved, and time-constrained precision flight paths that require redundant high-performance, high-reliability navigation and guidance systems.

The net effect is that short-haul aircraft carry many electronic devices and sensors that, though similar in function, service different subsystems. Because these devices are generally built by different manufacturers, the costs of procuring, developing, integrating, and maintaining the navigation, guidance, and control subsystems are compounded.

Current avionics systems use numerous sensors, actuators, and display devices. These systems often overlap in function, thereby providing a limited degree of redundancy in control and navigation reliability. Redundancy, when it exists at all, exists most often in the form of hardware duplication. For example, reliability of commercial aircraft inertial navigation has been obtained by having identical triplicate Inertial Navigation Systems (INS).

More cost-effective approaches, other than use of multiple individual avionics hardware units, are available for improving avionics system reliability. Cost studies (ref. 1) show a significant potential for improvement in both reliability and cost if integrated avionics concepts were based on strapdown technology. These concepts use the inherent redundancy of the strapdown guidance, navigation and control sensors, advances in sensors and semiconductor technology, and advances in redundant avionics algorithms. They also use the computer to identify faults and to aid in removing failed sensors by software reconfiguration. This integrated approach eliminates unnecessarily redundant hardware. The SIRU is such a system.

The SIRU system is an experimental, integrated, modularized, fault-tolerant avionics system originally developed for space applications by the Charles Stark Draper Laboratory under contract NAS9-8242 with NASA Johnson Space Center. The original mechanization was augmented under contract NAS2-7439 with NASA-Ames Research Center to install a flightworthy, redundant navigation system in the NASA Convair 340 (CV-340) aircraft. The SIRU flight test program was undertaken by NASA/Ames to evaluate SIRU in an aircraft flight environment. The primary goal of the SIRU flight test program was to develop a performance baseline of an integrated guidance, navigation, and flight control system for short-haul aircraft.

The SIRU flight test system is a free-inertial (unaided, dead-reckoning) navigation system consisting of a strapdown hexad inertial sensor array (fig. 1) and dual Honeywell H316 computers with a digital tape recorder (fig. 2). The sensors consist of six single-degree-of-freedom integrating gyros and six linear accelerometers. The six sensing axes of the inertial sensor array lie along the normals to six nonparallel faces of a dodecahedron.

Each axis contains one integrating rate gyro module (IRIG 18 Model B) and one linear accelerometer module (16PM PIP MOD B). The symmetric dodecahedral sensing axis orientation is depicted in figure 3.

The SIRU dodecahedral sensor configuration and the redundancy management software permit the navigation system to function when any three gyros and any three accelerometers are operational. When sensor failures are detected and the faulty sensor identified (isolated), outputs of the faulty sensors are excluded by system software from the computation of position, velocity, and attitude. The system continues to function with the remaining sensors.

The SIRU system includes prealigned, normalized, interchangeable assemblies in which each instrument is integrated with its own torque loop and temperature controller. The remaining electronics are located in interchangeable modules, by axis, in the electronics assembly section. The modular feature facilitates system maintenance repair and replacement of line units.

SIRU was continuously exercised in a laboratory environment from July 1970 through July 1974, prior to the adaptation of the SIRU instrument package for flight tests Fifteen flight tests of the SIRU system were made in NASA's CV-340 aircraft between May 20, 1975 and September 24, 1975. The flight test system had a digital data tape recorder and individual plasma information displays. The installation of SIRU in the CV-340 is illustrated in figure 4.

To assess SIRU's navigational performance, the flight test program used several aircraft position references. In addition to the two Nike-Hercules tracking radars at the Ames Research Center Crows Landing Naval Auxiliary Landing Field test facility, the test program used (a) multiple station DME range data from a digitally tuned DME receiver, (b) optical waypoint position fixes taken on landmarks with a driftmeter and recorded by camera, and (c) surveyed position benchmarks (latitude and longitude) used as external references. Barometric and radar altitude were also measured and recorded in flight. During SIRU's free-inertial navigation computations, the position references at Moffett Field and Crows Landing were used for initialization only.

This report, Volume II, is a compilation of the flight test and laboratory objectives, procedures, and results It is organized as follows:

1. The next section defines the specific flight test objectives in more detail. These include investigations of the inertial navigation accuracy, failure detection and isolation algorithm, and the dual-computer mechanization.

2. The third section describes the flight test program including (a) the flight plan, (b) the reference position measurement system, (c) the SIRU operational procedures, and (d) the data analysis algorithms.

3. The fourth section presents a summary of individual flight test results, the flight test assessment of the SIRU navigation and failure detection and isolation capability, and apparent system limitations.

4. The fifth section presents results from a laboratory evaluation of the SIRU dual-computer system.

5. The final section summarizes the flight test program and makes recommendations concerning future research and applications.

Volume III of the flight test report consists of seven appendixes which provide technical details that support the material included in this volume.

FLIGHT TEST OBJECTIVES

The broad goal of the SIRU flight test program was to continue the investigation of the application of strapdown inertial technology and redundancy management techniques with the objective of achieving a low-cost, highly reliable, integrated flight avionics system for short-haul air transportation. This goal was to be attained by flight testing the SIRU system in the NASA-Ames Convair 340 (CV-340) aircraft and by making additional laboratory investigations.

The principal flight test objectives included assessment of:

1. SIRU's performance as a skewed-sensor-free-inertial navigation system.

2. The system's ability to detect and isolate sensor failures with redundancy management software.

3. The performance of the SIRU system's redundant parallel dual computer configuration and its compatibility with the hexad sensor redundancy arrangement.

The flight test results were examined with specialized post-flight software (refs. 2 and 3) to assess the SIRU system's performance.

Inertial Navigation Accuracy

Inertial sensors can be used for smoothing (low-frequency filtering) of radio guidance and navigation signals on short-haul flights. These signals may be smoothed by using aircraft attitude reference sensors (vertical gyro, directional gyro), body-mounted rate sensors, and body-mounted accelerometers. The SIRU system concept has the potential of replacing all the above sensors with derived outputs from its inertial grade sensors, and of providing both guidance and navigation smoothing and free-inertial navigation (which is not available from the standard sensor set).

As the only inertial sensor system aboard the aircraft, SIRU would perform four essential functions:

1. Inertial sensing for flight control
2. Inertial sensing for radio guidance data (such as ILS) smoothing
3. Inertial smoothing to complement radio navigation data
4. Dead-reckoning, free-inertial navigation

Tasks 1 and 2 are flight-critical and require redundant fail-operational capability.

For short-haul aircraft operations, free-inertial navigation can provide the following services, which are useful but usually not available:

1. Short-term area navigation in the case of radio failure
2. Great circle route navigation, which replaces supplementary electronic systems needed with radio navigation
3. Missed-approach guidance backup in the terminal area under instrument flight rules
4. Wind shear and gust monitoring (this may become a flight-critical requirement, but presently is not)
5. Primary navigation in the absence of radio aids

The navigation performance requirement for these services is generally accepted to be position error growth of less than 1-3 n. mi./hr for all aircraft operation maneuvers, including errors resulting from operation of the algorithms.

The SIRU system was the first hexad free-inertial navigator to be evaluated in flight for the above functions. The purpose of the skewed sensor geometry was to provide optimum redundancy management and fail-operational capability. A principal flight test objective was to demonstrate that this skewed sensor geometry could provide free-inertial navigation performance comparable with typical orthogonal sensor geometry.

In skewed sensor geometry, sensor outputs are combined to form a set of three orthogonal outputs, which are used in navigation. While cross-coupling of sensor output is required with nominally orthogonal geometries to account for known sensor misalignments and nonparallel coordinate frames, the amount of cross-coupling required is greater with skewed geometries. This large dependency upon combining sensor outputs could conceivably amplify the deleterious effects of misalignment, measurement quantization, modeling, and compensation errors on navigational performance. The hope was that advantages of the redundant-sensor aspect of SIRU would balance out any possible negative skewed-geometry aspects. In fact, the concept of using multiple redundant skewed sensors could conceivably provide better performance than an orthogonal triad sensor configuration with comparable sensors (ref. 4). Thus, the objective of evaluating navigational accuracy was partially directed at making this evaluation.

Extra attention to the software was required because information from six skewed redundant sensors had to be combined to render the orthogonal measures. The increased complexity of the skewed, redundant system increased the likelihood of software mistakes and dictated greater sophistication and care in programming. The flight test program was also designed to prove that such problems associated with skewed sensor geometries need not degrade navigational performance.

The SIRU concept was originally designed to meet the operational, electrical, mechanical, environmental, and mission interface requirements of the Apollo spacecraft. Some of these requirements were distinctly different than those for operation in a propeller-driven aircraft, such as the CV-340. These differences are compared in table 1. The many physical differences from the original design-operating environment could affect the flight test outcome in both navigation accuracy and failure-detection performance.

The following points associated with navigation accuracy considerations were to be measured or assessed:

1. Improvement in navigation performance (if any) through the utilization of six sensors for measuring three orthogonal components (angular rate or specific force).

2. The adequacy of the calibration and alignment techniques developed for SIRU, and the identification of needed calibration and alignment improvements.

The flight test program data were analyzed to address these points.

SIRU Failure Detection and Isolation Algorithm Performance

One primary goal of the SIRU project was to advance the state of the art in redundancy management algorithms. Therefore, the second principal flight test objective was the evaluation of the SIRU computer algorithms that detect and isolate (remove) failed instruments (failure detection and isolation). For operational utilization, these algorithms must function without causing a serious degradation to the inertial navigation performance, or without generating deleterious signals (false alarms) when the system is used for flight control purposes.

Two types of failure detection and isolation algorithms were implemented in the SIRU software. These are referred to as the total squared error algorithm and the statistical failure detection, isolation, classification, and recompensation algorithm

The total squared error algorithm operates on the compensated instrument data to estimate the difference (error, $\hat{E}_a$) between what one sensor axis actually reads, A, and what that measurement is estimated to be, $\hat{A}$, based on the other five sensor measurements. Geometric details and the theory behind this algorithm are found in reference 5.

A fault is known to exist (detected) when the sum of the squares (total squared error) of the six individual errors ($\hat{E}_A$ - $\hat{E}_F$) exceeds a preselected threshold, called the maximum allowable squared error. The fault is localized to a specific measurement axis (isolated) by examining the time history of each individual error and finding the maximum. The maximum allowable squared error threshold is set by examining the fault-free values of the total squared error, including effects of quantization and environmental noise. A maximum allowable squared error set too low causes false alarms, and too high a value produces undetected failures.

The total squared error algorithm was the only one used for monitoring the accelerometers because it could theoretically detect and isolate those accelerometer degradations that would cause appreciable navigation errors. The total squared error algorithm was also used to monitor the gyros to detect and isolate "hard" failures whose magnitudes theoretically exceeded 0.75°/hr. The intent of the statistical failure detection, isolation, classification and recompensation algorithm was to detect and isolate gyro failures from below 0.75°/hr down to approximately the environmental noise expected in the error equations.

Additional functions of the statistical failure detection, isolation, classification, and recompensation algorithm were to classify and calibrate the gyro failure as a jump in bias, ramp in bias, noise variance increase, or false alarm. The degraded sensor could then be reinstated for use by recompensation. This algorithm is based on using a sequential probability ratio test and is discussed fully in reference 6.

The fault isolation methods had two different constraints:

1. Simultaneous failures could not be isolated.

2. Failures of equal amplitude (the second occurring during the isolation period of the first) could result in the loss of failure identification of both sensors.

It was assumed that these two conditions would not occur operationally.

One specific subobjective of the failure detection and isolation algorithm evaluation was to establish the failure detection time versus the failure level for sequential failures in flight operating conditions Laboratory measurements indicated that the failure detection and isolation algorithms had the capability to detect, isolate, and recover from gyro drift failures as small as 0.068°/hr, a value commensurate with the IRIG 18 MOD B gyro's random drift Theoretical considerations showed that the time for detection and isolation decreased as the failure level increased. During the flight tests, simulated attitude rate failures were introduced into the navigation computer during cruise conditions and terminal area maneuvers representative of short haul aircraft. The lengths of time required to detect, isolate, and recover from the simulated sensor failure were observed and recorded. This permitted a comparison to be made between the actual failure detection time of the implemented failure detection and isolation algorithms and laboratory performance.

Theoretical evaluation of the SIRU system showed that navigation performance was degraded by the output of a failed sensor during the time interval from inception of the failure to detection and isolation of the failure by the failure detection and isolation algorithms. After isolation of the failure, the navigation performance continued to be degraded only by the attitude or velocity error that developed during the detection interval. The second subobjective was to evaluate this degradation in navigation performance when sensor failures were introduced. This is discussed further in volume III.

Aircraft are subject to small high-frequency oscillations that cannot be measured exactly by a sampled-data system. Digital sampling of these oscillations may introduce errors into redundancy management and navigation and attitude calculations. The third specific subobjective of the failure detection and isolation algorithm performance assessment was to evaluate the effects of these digital sampling errors during aircraft operation.

The statistical failure detection, isolation, classification, and recompensation algorithm was designed to supplement the deterministic total squared error algorithm in order to improve the level of sensitivity for sensor fault isolation and to provide an on-line recalibration capability. When a sensor exhibited normal statistical characteristics but its scale factor or bias deviated from the calibrated value, the statistical failure detection, isolation, classification, and recompensation algorithm identified the sensor and provided new compensation values of bias or scale factor. The fourth specific subobjective in assessing failure detection and isolation algorithm performance was to evaluate its usefulness in flight instrument recalibration for aircraft operations.

Redundant Dual Computer Mechanization Assessment

The dual computer configuration selected for the flight test program provided redundancy compatible with the redundant SIRU hardware and software systems. Dual ruggedized Honeywell H316 computers were selected to provide this capability. The H316 computer assembly language used for the flight test implementation was compatible with all existing laboratory SIRU software, negating the need for recoding. The ruggedized hardware design was flightworthy.

The dual computers were mechanized to run in a "prime/backup" mode. The prime data were used as the valid system output unless the prime channel showed a failure or the test engineer forced a switchover. Each computer ran an identical software program, and the two computers were synchronized by software checkpoints where comparisons were made and go-ahead signals were issued.

Data were acquired by the computers from the SIRU multiplexer and the airborne reference system. The data were then processed and output to the redundancy management hardware. Selftest programs were run in parallel to the normal processing, and the results were transmitted with the regular output data for determination of the prime computer status by an arbiter. The arbiter provided voting to switch automatically from the "prime designated" to "back-up designated" computer.

The final principal objective of the flight test program was to evaluate the performance of the dual, synchronized-with-arbiter, redundant SIRU computer system mechanization under operational aircraft conditions. This objective included laboratory evaluation. Redundancy management of the two computers by dual transfer boxes, dual receivers, and the arbiter is illustrated in figure 5.

Figure 6 shows the dual computer and digital tape recorder control interface box and arbiter control panel. The dual plasma displays are shown in figure 2.

FLIGHT TEST PROGRAM DESCRIPTION

The flight test program was designed to evaluate navigation, redundancy management and dual computer operation. It included both short (<1.5 hr) and long (>3 hr) flight periods, straight and triangular segments, curved paths, and conventional terminal area maneuvers. This chapter describes the SIRU flight test program, including:

1 SIRU flight test plan
2. A description of the CV-340 aircraft position reference system
3. SIRU operational procedures used to prepare and operate the system during the test period
4. Data analysis algorithms used to reduce and analyze the flight test data using the Ames Research Center IBM-360 or CDC-7600 computer

Flight Test Plan

The SIRU flight test program began May 20, 1975 and ended on September 24, 1975. Several acceptance flights were made earlier at Hanscom Field, Massachusetts. Acceptance test results are reported in reference 7. Ames Research Center's tests began and ended at Moffett Field Crows Landing Naval Auxiliary Landing Field was the central calibrated waypoint for every flight test. All flights were made in California's San Joaquin and Salinas Valleys. The largest segmented course included waypoints at Sacramento, Modesto, Salinas, Moffett Field, and Oakland. The most commonly flown path was from Crows Landing, Modesto, to Castle Air Force Base, and back to Crows Landing. Figure 7 shows the general areas used for flight test and the landmarks (waypoints) used for visual calibration of the reference system.

A total of 15 test flights were made, with 3 flights devoted entirely to navigation performance. Cruise portions of the flights were of sufficient length to record one or more complete Schuler periods. Early flights were used for navigation, SIRU calibration, and system adjustments. Portions of each flight were utilized for testing the failure detection and isolation algorithms. During September, several test flights were made near Crows Landing to provide performance data during turns and takeoff and landing maneuvers. Most flights were made below 3048 m (10,000 ft). Flight tests which evaluated terminal area maneuvers followed the flight profile illustrated in figure 8.

Flight tests made in late July, August, and early September included cases where the two flight computers were operated independently from the same sensor outputs. During these flights, the dual computer system was used primarily to evaluate the SIRU system's sensor redundancy management rather than to test dual computer redundancy management.

Flight test patterns were of two types: enroute (Moffett Field-to-Crows Landing or Crows Landing-to-Moffett Field) and approach (in the general vicinity of Crows Landing). Enroute patterns are illustrated in figure 9 (take off from Moffett Field, turn to fly over the Moffett Field DME for a tape-mark, fly over the San Jose DME for a tape-mark (twice), fly over Lick Observatory for a visual mark, then overfly the Crows Landing DME for several tape-marks before landing). The same sequence in reverse order was used on returning to Moffett Field

A typical flight sequence originating at Crows Landing is shown in figure 10. In this flight, the CV-340 left Crows Landing, crossed the runway at Stockton, turned southeast, proceeded to Castle Air Force Base, then flew to Merced, and finally back to Crows Landing.

Aircraft Position Reference System

The continuous position system used during the SIRU flight tests to track the CV-340 aircraft consisted of two primary references:

1. A modified Nike-Hercules radar tracking system located at Crows Landing
2. A six-channel multiple DME receiver system, designed by Sierra Research Corporation, mounted within the aircraft

The modified Nike-Hercules radar provided improved resolution through the use of 19-bit range and angle digital shaft encoders. No atmospheric refraction correction was provided. A transponder aboard the CV-340 was used to improve angle tracking.

The DME receiver system provided range information from up to six DME or TACAN stations. The system utilized a fast-switching DME receiver which was programmed to automatically switch through each of six selectable DME or TACAN frequencies. Range lock-up time was 1 sec maximum and range output resolution was 18.5 m (0.01 n. mi.). Output range information was tagged with station frequency and receiver-clock time for identification.

Time-referenced photographs of airport reference benchmarks were also taken from the CV-340 aircraft periodically throughout each flight to provide a third basic waypoint reference. These waypoints are indicated in figure 7. The photographs were taken from the aircraft by a camera mounted on a military standard driftmeter installed in the underside of the aircraft fuselage. The positions of the driftmeter and DME receiver system within the CV-340 are illustrated in figure 11. The timepoint at which each photograph was taken was recorded in the SIRU digital tape recorder in order to correlate with radar, SIRU, and DME position data. The number of photographs per flight varied from 8 to 16, depending upon the length of flight. The driftmeter's level gyro was inoperable during most of the flight tests, causing uncertainty about the aircraft's attitude. As an example of this uncertainty, a 0.5° vertical misalignment of the camera at an altitude of 3050 m (1000 ft) introduces a position error of 26.2 m (86 ft).

Figure 12 is a chart indicating the relative position of the DME stations, radar system, and reference airports utilized during the flight tests A list of the DME stations and their locations is printed in table 2. The range and bearing of each station is given with respect to the TACAN's coordinates at Crows Landing. The acquisition altitude given is the minimum altitude required at Crows Landing in order to receive a signal from the DME station.

In order to provide accurate start and terminal area position fixes, the United States Geological Survey was commissioned to obtain reference position benchmarks and azimuth lines at both Moffett Field and Crows Landing The benchmarks were located to within ±4.0 arcsec of position (standard deviation). Tables 3a and 3b list the latitude, longitude, and elevation of the applicable benchmarks surveyed for Moffett Field and Crows Landing, respectively. Figures 13 and 14 present maps illustrating the location of the benchmarks with respect to reference buildings and landmarks at Moffett Field and Crows Landing, respectively.

Flight Test Operating Procedures

Because of limited calendar time available for the flight test portion of the SIRU program, a concerted effort was made to record a large amount of data during a variety of test flights. Each flight had raw and calibrated data recorded with engines off, and up to 0.5 hr of raw and calibrated data recorded with engines on prior to taxi and flight In this manner, a sufficient amount of preflight data was recorded for preflight test calculation of inertial sensor compensation values for use during flight The data also provided postflight confirmation of performance changes occurring during the test period.

All the flight tests were generally conducted with the following ten sequential events.

1. Equipment turnon and warmup using ground power with the CV-340 aircraft located at ground benchmark (reference point) #A. Align aircraft position with respect to benchmark prior to azimuth calibration.

2 Level SIRU mechanical platform and align SIRU azimuth using the porro prism theodolite and the calibrated azimuth line.

3. Start aircraft engines and remain in position up to 0.5 hr.

4. Put SIRU in fine align.

5. When computed instrument errors have stabilized for bias calibration, turn on digital tape recorder and shift to navigate mode.

6. Commence taxi for takeoff, hold at Moffett Field runway warmup parking area to confirm operate status

7. Fly from Moffett Field to Crows Landing. If temperatures on the ground at Crows Landing are prohibitive to the continuous instrument operation of SIRU, go to 9.

8. Land at Crows Landing and taxi to benchmark #L. Put a tape-mark on the tape at touchdown and at the end of straight-line deceleration The nose wheel follows as closely as possible to the white line during all take-offs and landings Continue recording for about 20 min. The tape will have 40-50 min of recorded data at this time. At this point a supplementary alignment may be performed if necessary. Continuous data tape recording is made. Go to 10.

9. Initiate Crows Landing test flight pattern. Land afterwards if environmental conditions permit and go to 8 Otherwise, go to 10.

10. Fly from Crows Landing to Moffett Field. Record data. Land and taxi to benchmark #A. Continue to navigate and record for about 5 min after aircraft motion has ceased.

In all navigation flight tests, a full sensor calibration was made prior to flight. Sensor calibration procedures had the following sequential events:

1. The SIRU instrument package was turned on long enough beforehand to remove transients and never turned off until the end of the entire calibration and flight test sequence. The calibrated azimuth line was signed by theodolite through a porro prism and the mechanical platform was bubble-leveled before the start of the sequence.

2. SIRU was put in a calibration mode.

3. After approximately 30 min in one position, the SIRU package was slowly and mechanically rotated to a new position.

4. Step 3 was repeated three more times for a total of five positions.

5. The H316 computer determined bias and g-sensitive elements and prepared a paper tape for SIRU.

6. SIRU was reprogrammed for navigation, and a new calibration matrix was entered via the paper tape.

7. The instrument package was not turned off before completion of flight tests. Fine alignment for the test was begun.

Forced failures were introduced in most of the flights to test the failure detection and isolation algorithms. However, if these forced failures in any way compromised the ability to obtain recorded flight data, they were abandoned. Complete recording of all raw data was a requirement for postflight reconstruction. About 1 hr of continuous data was recorded on one 1097 m (3600 ft) magnetic tape when both computers were on. This limited the flight time for tests when postflight reconstruction was desired. The allowed recording duration doubled when only one computer's output was recorded.

The eight flight test data sources (peripherals) interfaced with the SIRU system are depicted in figure 15. Readings from each peripheral were under H316 program control and were normally checked every 25 msec (every SIRU data input); that is, if new data were ready, they were read within a maximum of 25 msec.

Some of the peripherals (e.g., DME) may have had more than one input at a single service. In this case the device was serviced until all data had been read, which may have prevented other peripherals from being serviced. The time code generator was updated every 2 sec. Data from the barometric altimeter and radar altimeters were updated every second.

Each receiver (of computer A or computer B), upon receipt of data to be recorded, issued a request to have its data stored on tape. The request was stored and presented to an eight-state sequencing operator. The operator selected which request (A or B) was to be serviced first, and then generated three enable pulses to the selected receiver. The data appeared on tape as three bytes from one channel or the other, but did not necessarily alternate from one to the other.

The arbiter examined the eight-bit diagnostic word on every output. It searched for any failure conditions, displayed the status of the diagnostics, and determined which system (A or B) should be prime. When a failure occurred, it was indicated by the appropriate fail light and channel-fail indication The channel-fail indication caused the computers to exchange prime status, after which the prior status, for certain failures, could only be restored manually.

Flight Test Data Processing

The SIRU Flight Test Program produced 46 magnetic tape reels of recorded test data. Thirty-four of the reels were produced by the SIRU Navigation System, the other 12 by the radar tracking facility at Crows Landing. The SIRU tapes contain recorded external reference data (DME, altimeters, time code generator, position fix times) as well as SIRU flight system data (navigation variables, attitude, failure data, sensor data, status information). The radar tapes contain time-tagged range, elevation, and azimuth as measured by each of two radars during those time intervals when the CV-340 aircraft was tracked. The SIRU tapes' data were produced and recorded through the SIRU navigation system's two H316 computers, while the radar tapes' data were generated by a PDP 1145 computer.

Data processing for the flight tests was performed on the NASA-Ames IBM 360/67 and CDC 7600 computers and at Draper Laboratory on the IBM 360/75. Differences in word lengths and word structures among the various computers necessitated considerable programming to extract the desired data from the tapes. Because the CDC 7600 was not operational at Ames at the beginning of the flight test program, the data reduction programs were programmed for Ames' IBM 360/67. These programs were later converted to permit data reduction on the CDC 7600.

SIRU data processing- The phases of SIRU test data processing at Ames are illustrated in figure 16. Programs are shown as rectangular blocks in the figure, while tapes and disk data files are denoted by circles and small ellipses, respectively.

Phase I produced a navigation data file and could also produce a gyro and accelerometer data file or raw data file by input option. The gyro and accelerometer data file contained the derived triad sensor data used in SIRU's attitude and navigation computations The derived triad data were the result of compensation, failure isolation, and mapping of the six raw measurements into three components. The raw data file contained uncompensated measurement data from SIRU's six gyros and six accelerometers. Phase I produced a printed tabulation of the flight and the reduced data files and included the tape copying process necessary for parallel-flight test data processing by Draper Laboratory.

Phase II required the navigation data file of Phase I as input. Program RADAR read and interpreted the radar tape (for flights with radar coverage) and produced a radar-derived trajectory file and a merged SIRU and radar data file. Program RADAR also computed and tabulated residuals between radar and SIRU estimates of position. Recorded time code generator times from the two sources (i.e., airplane and test facility) were used to synchronize the data. Program DMERES used a least-squares differential correction technique to derive a trajectory from recorded DME range data and wrote the trajectory as a DME file

Phase III featured the SIRU Navigation Analysis Program (SNAP) which computed an aided-inertial trajectory. This program also estimated likely error sources from the DME and/or the radar data (ref. 3). SNAP used DME and baro-altimeter data from the navigation data file and radar data from the merged data file. It also used equivalent-triad gyro and accelerometer data from the gyro and accelerometer data file. The aided-inertial (reference) trajectory was written on the smoothed navigation data file. Navigation and attitude residuals between the smoothed and SIRU estimates were also tabulated. Navigation residuals from each flight test are presented in Volume III of this report.

Plotting programs were used in Phase IV of the data reduction process to produce latitude-versus-longitude trajectory plots and to position residual strip-charts. Trajectory data from computer A's navigation data file could be plotted with trajectory data from computer B's navigation data. Program STRIP computed (and plotted versus time) north, east, and root-sum-square position residuals between any two trajectory files. Hard-copy plots were produced from computer-generated tapes at Ames Research Center's central computer facility. Ground track plots from each of the flight tests are presented in Volume III.

The 16-bit data words from each SIRU flight computer were recorded on tape together with eight bits of computer status information (which included a computer identity bit). Words from computers A and B were usually interspersed in fixed-length records on the tape and had to be sorted by using the computer identity bit. Each record contained 241-1/3 24-bit data/status words Three sequential records had to be read to get an integer number of data/status words into the ground processing computer. Recorded SIRU data were grouped on the tape (for each computer) by type (i.e., navigation data, sensor data, keyboard input, status). Each type was identified by a code word. Data types were recorded at differing rates (i.e., navigation at 1 Hz, sensor data usually at 20 Hz). Several different procedures were used to interpret and convert the data to usable form because several different scalings and bit structures were used in recording the data.

Radar data processing- All of the radar tapes from the SIRU Flight Test Program contained recorded data from both radars located at Crows Landing. The tapes were written by a PDP 1145 computer and, like SIRU flight tapes, their data required considerable software processing before being used in calculations on the IBM 360/67 or CDC 7600 computers.

Radar data were usually recorded at 20 Hz, although there were some exceptions. Every data frame was time-tagged with Universal Mean Time (UMT) as read to 0.0001 sec from a time code generator at the test facility. SIRU's navigation data were recorded at 1 Hz, and recorded UMT was included from a time code generator carried on board the CV-340. Program RADAR used only the 1 Hz radar data (i.e., every twentieth frame if radars were recorded at 20 Hz, synchronized as closely as possible to SIRU's recorded navigation times). SIRU's technique for sampling the time code generator gave a resolution of 25 msec to SIRU's recorded time. The recorded radar data were cycled until SIRU and radar times agreed to within 25 msec. The data were then sampled at 1 Hz to stay in synchronization with SIRU. The synchronization uncertainty could amount to a position uncertainty of about 3 m if the CV-340 aircraft speed was taken to be 120 m/sec.

The measured radar data were presented on the tape in raw counts, using nineteen bits (parts of two 16-bit words) to express each measurement. The measurements were biased by "given" calibration or zero values before being scaled to appropriate units (i.e., feet or meters and radians). The calibration values varied from flight to flight and were not readily obtained from the radar tape Rather, they were transmitted by hand from readings taken at the test facility. A shortcoming of the radar tape recording procedure was that ranges "overflowed" every 2^{19} counts (~21 n. mi.) which often required external information to resolve the ambiguity.

Program RADAR used (on option) either target-tracking radar or missile-tracking radar range/azimuth/elevation data to derive the position information which was written on the radar data file. The position derived from the radar data was compared with the SIRU system's indicated position to produce a tabulated position residual history. In this comparison, radar-derived altitude above mean sea level was compared with barometric altitude. The merged data file was simply a navigation data file with measured range, elevation, and azimuth from the two radars appended.

DME Data Processing- DME range data were transferred from the SIRU flight tape to the navigation data file by program QUICK. Program DMERES then read the navigation data file and derived position (i.e., latitude and longitude) from the range data.

Program DMERES obtained DME frequencies and ranges for each recorded SIRU time from the navigation data file. DME frequencies were compared with a stored list of candidate DME frequencies. Table 2 lists the stored DME data with the DME's range and bearing from Crows Landing and with the acquisition altitude there. When the DME's frequency was identified from this list, the corresponding stored DME location was used in Program DMERES's position calculations. When the frequency, could not be found in the list, the corresponding measured range was omitted from further position calculations.

The problem of nonconcurrency of measurements was treated by calculation of range-rate (using SIRU's velocity in the calculation) and prediction of range from the most recent time the range from a particular DME was refreshed. Range was considered to be refreshed when it changed in value or when the DME's frequency changed. A zero value for range was recorded when no "lock-on" occurred.

Program DMERES did not try to form a position solution with less than two range measurements. When two or more range measurements were active, Program DMERES performed an iterative differential correction calculation to adjust the position so that the sum of the squared range residuals was minimized. The resulting position was written out as a function of time. Rounding off the apparent time of the measurement range to the nearest whole second introduced a potential position error of 100 m.

SIRU FLIGHT TEST RESULTS

Flight tests of SIRU were performed on 15 separate days, beginning May 20, 1975 and concluding September 24, 1975. Some of these tests included multiple segments, and each segment produced a magnetic tape of recorded SIRU test data. In most cases, SIRU was fine-aligned between segments, so that each segment could be considered a separate flight test. Thirty-four segments were flown. One of the tapes could not be read by the software on the Ames computers. The other 33 segments totaled about 36 hr in navigation, 26 hr of which were spent in flight.

Individual Flight Commentary and Description

In the following discussion, comments are offered concerning the progress made during each of the flights. More detailed descriptions of each individual flight including the test objective, computer and inertial component configuration, failures scheduled, flight description, durations from the flight plan, test times (warm up, fine alignment, navigation, and flight), general comments, and ground track plots are presented in appendix A (Volume III) The first several flights were intended to remove problems and finalize the test procedures.

Test Flight No. 1. (5/20) This was flown from Moffett Field to Crows Landing by way of San Jose Airport and Lick Observatory on Mt. Hamilton without new sensor calibration after delivery from Hanscom Field, Massachusetts. Only the output of the B computer was recorded during the first two flights because of observed anomalies of the A computer.

Test Flight No 2. (5/30) The flight was to include a segment from Crows Landing to Bakersfield This segment was modified to Crows Landing-Los Banos-Fresno-Crows Landing when the observed navigation error became too large. The flight continued to Moffett Field without landing at Crows Landing because of high cabin temperatures expected on the ground. (The air conditioning system in the CV-340 had proven inadquate earlier.) Three tapes of radar data recorded at 100 Hz were provided.

Test Flight No. 3 (6/16) By the third flight the problem with computer A had been corrected, but the multiple unscheduled sensor failures and poor navigation experienced on the first flights continued without explanation The radar at Crows Landing was inoperable for this third flight because of computer problems.

Test Flight No. 4. (6/18) Because of the air conditioning problem, the fourth flight test was made at night, when it was cooler. This 4-hr flight test was intended to exhibit SIRU's navigation capabilities. Because of procedural problems, SIRU exhibited a 34 n. mi. maximum position error, thus negating the original intent of the tests The radar computers were still having problems, so the only radar data consisted of several pages of hand-recorded readings laboriously taken by the radar operators during the flight. These have never been processed.

Test Flight No 5 (6/25). By the fifth flight, radar readings were again recorded on tape, although the actual recording rate differed from that specified on the tape's format record. The time code generator was initialized incorrectly by 1 hr for this flight, which further complicated the SIRU/radar data synchronization process. Unscheduled sensor failures continued to show up, and navigational performance was poor, even though long calibration and alignment sequences were carried out before each flight.

Test Flight No. 6. (7/14). Longitude for the sixth flight was initialized in error by 1° This flight, which had no radar coverage, was again plagued by unscheduled sensor failures, and it exhibited very poor navigational performance

Before the seventh flight, several modifications were incorporated into SIRU which resulted in generally improved navigation and fewer unscheduled sensor failures The modifications included using "old" accelerometer compensation values obtained the year before in the Draper Laboratory tests rather than values obtained in the preflight calibration procedure. Based on data from the earlier flights, the failure detection threshold was modified to include a variable detection tolerance level for the gyros. The detection level was increased, and the dynamic portion was revised to accommodate the total squared error increase during turning maneuvers. The statistical failure detection, isolation, classification, and recompensation algorithm was also removed from operation in the navigation mode. The flight schedule did not permit reinstatement of the statistical failure detection, isolation, classification, and recompensation algorithm during the remaining flights. However, volume III presents an analysis of the data using this algorithm with recommended corrective action.

Test Flight No. 7. (7/17): The seventh flight showed marked improvement in both navigation and failure indication. Navigation error decreased by an order of magnitude over previous flight tests, and no unscheduled sensor failures were encountered. In addition, radar coverage was again available (for overflight), although Crows Landing was closed to civilian landings and takeoffs. The first scheduled inflight gyro failures were successfully detected and isolated during the return segment of this flight.

Test Flight No. 8. (7/24): The eighth test was again successful, being characterized by relatively small navigation errors, no unscheduled sensor failures, and proper detection and isolation of scheduled gyro and accelerometer failures. The scheduled failures were inserted into only the B-computer's compensation logic during the third segment of this flight, thereby enabling, for the first time, a comparison between "unfailed-SIRU" and "failed-SIRU" navigation. This comparison showed the relative navigation error buildup due to the use of the failed sensor's data between the time the failure takes place and the time the failure is isolated. Good radar coverage was obtained for this flight and on through to the end of the program. An unfortunate aspect of the eighth flight was that only the first of three segments was recorded until completion of the segment. The other two segments ran out of tape. Only 731.5 m (2400 ft) tapes were available for this flight, and the tape moves almost 1 ft per second of real time when both computers are recording at full rate.

Test Flight No. 9. (7/29): The ninth flight again exhibited unscheduled failures and degraded navigation. Some scheduled gyro failures were, nevertheless, correctly detected and isolated. The first segment was marred by the fact that the tape recorder was not started until 1100 sec into navigation, at which time the aircraft was in the air over San Jose.

Test Flight No. 10. (8/22): This flight suffered computer and/or tape recording malfunctions as well as several unscheduled sensor failures. Scheduled catastrophic gyro failures in the second and third segments were successfully detected and isolated; however, the first and second segments of the flight showed fairly good and very good navigation, respectively. During the third segment (Crows Landing to Bakersfield), the B computer recorded navigational data at 20 Hz much of the time. Only a small portion of the tape for the return from Bakersfield was readable. There were many computer failures indicated on the display panel for this segment Some intentional computer failures were also injected on this flight. Computer failures were observed on no other flights in the program

Test Flight No. 11. (8/29)· This flight showed no unscheduled sensor failures while successfully isolating nearly-simultaneous scheduled failures in gyros A and B Navigation was not very good on the Moffett Field-to-Crows Landing segment of this flight. It was good on the second segment which consisted of flying patterns at Crows Landing SIRU's position was reset and the velocity was zeroed in midnavigation on the second segment of this flight, resulting in a navigational discontinuity.

Test Flight No. 12. (9/05): This flight was the last flight involving scheduled failures. These scheduled failures were successfully detected and isolated, although some unscheduled failures were indicated as well. Navigation was relatively good.

Test Flight No. 13 (9/10) This flight featured multi-maneuver patterns at Crows Landing with no scheduled failures and with little regard for navigational performance (which still turned out to be fair). This flight was almost free of unscheduled failures. SIRU was fine-aligned before the flight but was otherwise kept in the navigation mode throughout the flight. Tapes were changed while the aircraft was stationary at Crows Landing.

Test Flight No. 14. (9/18). This was a repeat of Flight 13, except that SIRU was fine-aligned before the flight and also before returning to Moffett Field.

Test Flight No. 15. (9/24): The final flight was the longest of the series. It exhibited good navigation performance and there were no indicated sensor failures.

Figure 17 depicts navigation mode duration, in-flight duration, and radar coverage periods for the 15 flight tests. The flight date appears on the left side of the figure. Each flight segment is represented by an unbroken horizontal line Later segments are shown relative to the zero time of the first segment for each multi-segment flight. Navigation duration for any segment is indicated by the length of the segment's unbroken line. The darkened portion of the line indicates time in flight, with "T" indicating take-off and "L" indicating landing. Radar coverage periods are indicated by parenthesized line segments just above the navigation segments. It may be observed that 12 of the 15 flights had at least some radar coverage. DME coverage, although of poorer accuracy than the radar coverage, was present in almost unbroken fashion for every flight test segment.

Table 4 summarizes characteristics of the SIRU flight tests. Navigation mode and flight duration (in seconds) are shown, after a brief description of the route flown by the CV-340 Radar intersection is the total common time interval for which there are data on both the SIRU and radar tapes. "Marks" denotes the number of position fixes taken and recorded "Computers" identifies which computers were recorded and whether they differed when both were recorded. "Rate" denotes whether raw sensor data were recorded at the high rate (H) (20 Hz) or low (L) (1 Hz) rate.

Maximum position error is shown in nautical miles and, in most cases, refers to the DME-derived position, which is assumed to be correct. The maximum error usually (but not always) occurs at the end of the recorded segment. It should be noted that some tests were flown without particular concern for navigation accuracy and that operational mistakes are included with the errors.

Detected failures are listed in the last two columns. Failure, in the context of this report, means that the performance of a given instrument, (gyro or accelerometer), exceeded a "predetermined" limit that had been established in the failure detection software. The predetermined limit is set by the need to minimize false alarms during operation. Failure levels are determined by sensor random errors, uncompensated modeling errors, digital system noise and temporary or permanent shifts in the sensor operating points Unscheduled failures are those which were unintentional; scheduled failures were deliberately injected by miscompensating the sensors at some point in flight. The notation "G (FCBDEA)," for example, indicates that every gyro failed at some time during the test with F being the first to fail. Accelerometer failures are indicated, for example, by the notation A (D), which means that only accelerometer D failed on the segment. Only two failures of either type of sensor can be indicated by SIRU at any one time.

General Navigation Accuracy Results

Free inertial navigation performance- Appendix A (Volume III) shows the ground track plots of the 33 recoverable flight test segments. Each plot shows SIRU's best ground track estimate as a solid curve. The dotted paths show aircraft ground tracks computed from DME or radar data or SIRU's other computer. A summary of the test conditions and calibration is included

Appendix B (Volume III) shows the navigation position error (residuals) histories for the same 33 recoverable flight test segments. These residuals are based upon DME-derived or radar-derived position versus time. All residuals are plotted in nautical miles.

The SIRU system's navigation accuracy was evaluated by comparing the computed position with an independently derived position from one of the external position references during the same computer frame time. Except for occasional intervals of bad data, the external position references were in close agreement throughout the SIRU test program.

The position differences (in feet) shown in table 5 were computed from data obtained from the three position references. The differences between the radar position and the position computed from the time-referenced driftmeter photographs are labeled "R-M." The differences between the DME-derived position and the driftmeter photograph position are labeled "D-M." Finally, the differences between the radar-derived position and the DME-derived position are labeled "R-D." An average residual for each of the three position measurement difference methods is given in table 6. These errors show close agreement, with the radar-derived and driftmeter-derived measurements being closest

Discrepancies among the references were small compared to the SIRU system's navigation errors In each instance where the system's B-computer differed from its A computer (by the insertion of scheduled failures into B and not into A), the navigation error estimate from the A computer, which was more likely to be accurate, was used in performance calculations.

Table 7 summarizes the SIRU system's navigation performance during the flight test program. The maximum position error (residual) appears after the flight and navigation duration columns in the table. This maximum error does not necessarily occur at the end of the navigation segment because the navigation error does not grow linearly with time. Again, it should be noted that obtaining good navigational accuracy was not an objective for some flights, so calibration and alignment procedures were relaxed. In an attempt to represent the navigation error as a simple function of time, these errors were fitted by the least-squares method to the following three functions:

1. A constant (the average, shown in column 4 of table 7)
2. A straight line through the origin (slope is shown in column 5)
3. A straight line A + Bt (with A and B shown in columns 6 and 7)

A popular single-number characterization of navigation system accuracy is, "x nautical miles/hour " This implies that navigation errors are proportional to time The slope of a fitted straight line through the origin (column 5 of table 7) serves as a single-number accuracy characterization for the SIRU flight test program. Flight segments 9/10A, 9/10B, 9/10C, and 9/10D were navigationally contiguous, as were segments 9/18A, 9/18B, and 9/18C. Segment 9/18D was fine-aligned and reinitialized The table entries for maximum error on these multi-segment flights display the local maxima (i.e., each segment considered alone), which accounts for segment 9/18C showing a smaller maximum than 9/18B The curve fits shown for each flight segment were determined from all data up to and including that segment. For example, the straight-line error slope shown for segment 9/10C (5.37 n. mi./hr) is based on segments 9/10A, 9/10B, and 9/10C.

Figure 18 displays the maximum navigation error (column 3 of table 7) versus flight date. The letters (A, B, C, and D) which appear in the figure alongside the plotted error values denote the flight segment on which the plotted error occurred Figure 19 displays the resulting navigation error slope (column 5 of table 7) versus flight date These figures show

1 Generally improved navigational performance with time into the test program

2. Wide variations in navigation performance between segments of particular flights

On the basis of figure 19, the SIRU system's navigation accuracy rating at the end of the test program was about 3 to 4 n. mi./hr, although flight 9/24 indicated 1.02 n. mi./hr. The wide variations between same-day segments were due to alignment uncertainties and human error, because bias and scale factor sensor output corrections were common among these segments. The variations were also caused by the effects of different maneuvers on dynamic sensor compensations.

Navigation error due to fine alignment angular motion- Navigation accuracy is limited by the precision in alignment of the computational coordinate frame relative to the physical coordinate frame at the instant that the navigation mode begins. The SIRU fine-alignment algorithm (variable gain gyrocompassing) was performance-verified in the laboratory under static conditions and in the presence of small sinusoidal oscillations about the vertical system axis. However, alignment performance in the presence of larger angular motion about the horizontal axes (causing accelerometer outputs to vary) was not investigated prior to flight testing.

The fine-alignment algorithm generated commanded rotation rates about the Down, East, and South axes to bring the attitude quaternion into proper alignment. Earth rate compensation was included in the commanded rates. Base motion isolation was implemented with standard gyro information. The East and South axes commanded rates should converge to zero and to the South component of earth rate, respectively. The Down axis commanded rate should converge to the Down earth rate component.

The CV-340 was subject to unpredictable angular motion about its center of gravity during the SIRU fine-alignment mode. Thus it was of interest to determine if this random motion had a deleterious effect on subsequent navigational performance. An experiment was conducted to determine the effects of this angular motion on later navigation error propagation. This experiment consisted of a single special test designed to observe navigation performance during alignment in the aircraft. The fine-align operating program was modified to record the commanded rotation rates to bring the attitude quaternion into proper alignment. It was not possible, however, to make a precise evaluation of fine-alignment navigation errors from the recorded flight test data because raw navigation data (for reconstruction purposes) and the algorithm command signals were not recorded. At times, the quaternion was also observed (as well as recorded) during alignment. Because the quaternion tracks base motion while responding to commanded rotation signals, it alone could not be used as a measure of alignment performance. These data did indicate the possible presence of convergence errors; however, during the test the down commanded rate did not converge on the down component of earth rate as it should have. The negative excursion caused a 3-mrad change in the algorithm's estimate of true North and a 0.07 mrad change in the level estimate of East. The control loops were in a steady-state condition for about 1500 sec.

This evidence, together with failure detection and isolation algorithm performance data showing that all gyros and accelerometers were online and reasonably stable throughout the test, indicated that the SIRU base-motion isolation was not good enough. Angular motion did not exceed 0.5° about any axis during this special test, yet the SIRU fine-alignment was undesirably disturbed. One possible reason is that miscompensated accelerometer data caused alignment errors in the gyro data when acceleration-sensitive attitude changes occurred Unless the average East and South angular velocities converge to zero (as they ideally would), the alignment algorithm would begin generating an erroneous attitude correction signal resulting in steady-state "level-definition" and "north-definition" errors.

The magnitude of the navigation errors resulting from alignment algorithm problems is unknown because compensation values and environmental conditions varied throughout the flight tests and were not recorded.

Navigation error from gyro misalignment- Gyro scale factor and misalignment compensation values cannot be readily determined when the strapdown system is mounted in the aircraft This fact presents fundamental problems when installing a replacement gyro, for instance Laboratory compensation data must be transferred to the operating environment with an uncertain level of confidence.

Errors from these terms (particularly misalignments) show up most noticeably when the aircraft makes large changes in heading. Navigation accuracy may be impaired by the typically large (180°) turns which occur on the taxiway prior to takeoff and in the air shortly afterwards

The error equation residuals from the deterministic total squared error failure detection algorithm proved to be of some help in gauging the magnitude of these dynamic gyro errors. In one case, a badly miscompensated gyro was greatly improved by observing squared gyro-error magnitudes and adjusting a particular misalignment value to substantially decrease this error magnitude over a 180° turn. Specifically, it was found that abnormally high total squared errors were present during large turns of flight tests in June and the first half of July, 1975 Turns of 180° produced total squared errors from 70 to 90 pulses squared (44 arcsec/pulse). These errors corresponded to an acquired system attitude error in the vicinity of 400 arcsec, or about 600 ppm

Ground experiments led to making a 0.25-mrad correction to the C gyro (nominally horizontal input axis) misalignment term in the direction of the nominal vertical axis Subsequent flight tests (July 24 to termination) had uniformly lower total squared errors of from 20 to 30 pulses squared for 180° turns. This corresponded to an acquired system attitude error of about 200 arcsec, or 300 ppm. This improvement in total squared errors was expected after making the correction in misalignment compensation.

Further trial and error improvements to dynamic compensation terms were not attempted. The redundant nature of the SIRU should permit a selfcorrecting misalignment calibrator to be implemented which would yield much smaller attitude errors during typical flight test maneuvers.

A simple example shows the effect of misalignment errors on navigation accuracy. Assume that an aircraft with a perfectly initialized alignment of the SIRU system makes a 180° turn on the runway prior to takeoff. Assume all six gyros are perfectly compensated except one nominally horizontal gyro (C-gyro) which is misaligned toward the vertical by a small amount, δ radians. The 180° turn has a zero component along the C-axis, but a total rotation ($\delta \cdot 180°$) is erroneously registered. This translates into a body-axis computational misalignment of ($180° \cdot \delta/2$); this can be viewed as an initial level error. This initial level error starts a position error propagating with an 84-min (Schuler) period and with an angular magnitude (relative to the earth's center) equal to the magnitude of the system's level-error. The position error magnitude, in nautical miles, is approximately:

$$|\bar{\varepsilon}(t)| = \frac{60 \text{ n mi.}}{\text{degree}} \left(180° \cdot \frac{\delta}{2}\right)\left[1 - \cos\left(\frac{2\pi t}{84 \text{ min}}\right)\right]$$

Peak position error magnitude (at $t = 42$ min) is

$$|\bar{\varepsilon}_{\text{peak}}| = \frac{60 \text{ n. mi.}}{\text{degree}} (180° \cdot \delta)$$

A misalignment error of $\delta = 0.250$ mrad in this example would then yield a peak position error of 2.7 n. mi.

Navigation errors from failed gyros during failure detection and isolation algorithm tests- A hard attitude gyro failure causes an angular error increment (misalignment) in the inertial system's attitude reference. This misalignment causes navigational errors similar to other misalignments Also, the strapdown system is subject to further alignment errors when large angular motions of the aircraft occur.

The failure detection and isolation total squared error algorithm which monitored the gyros had an error threshold (maximum allowable squared error) set to a base level of 40 pulses squared (44 arcsec per gyro pulse) for flight tests in July, August, and September. A dynamic increase in the maximum allowable squared error of up to 120 pulses squared was programmed for changes in body attitude of 180° (or more) during any total squared error accumulation period (2 to 4 min).

If all but one of the gyros was well compensated, then the attitude error deviation caused by gyro failure would be similar to the velocity error deviation caused by accelerometer failure. Gyro errors occur due to both bias and misalignment errors, so the error was modeled simply as an angular error, $\theta_E(t)$, from one bad gyro. For a nonrotating system, the maximum allowable squared error was 40 pulses squared and maximum θ_E was 0.95 mrad. The maximum body attitude error was one-half that of the failed gyro, or 0.475 mrad. When large angular motions occurred, the maximum allowable squared error was four times its static value. Hence, the maximum body attitude error from a single misaligned gyro was twice that of a static error.

Navigation errors from a failed horizontal input axis gyro were most significant earlier in the flight. For example, the resulting Schuler-period position error (for 0.475 mrad level error) reached a first peak magnitude at

43 min and was 3.28 n. mi. This value occurred only for static navigation or low speed flight at a constant heading Heading changes increased the error

A comprehensive example from a flight test on September 5, 1975, demonstrates the effect a failed gyro had on navigation performance. This flight (9/5C) was a loop flight out of Crows Landing with a navigation mode duration of 39 min. Both A and B computers were initialized and aligned together. The F gyro had an unscheduled failure during alignment and was kept offline in both computers for the remainder of the test. A programmed 6°/hr bias error was inserted into the E gyro compensation of the B computer after the flight was underway. The A computer provided a reference. E and F gyros were in the aircraft y-z plane (wing-vertical) so the unscheduled F gyro failure made the system's navigation performance more sensitive to the scheduled E gyro failure than would otherwise be the case.

Figure 20 shows the reference computer's computed heading (truncated at 360°) for the period of navigation The maximum allowable squared error value shown in figure 21 for computer A was calculated on the basis of angular motion observed over the total squared error accumulation period The plots in figures 22 and 23 (for computers A and B) show how the total squared error forcing function was used to increase the maximum allowable squared error value up to the maximum of 160 pulses squared.

The A computer is seen to have an increased total squared error during the turns, but not enough to incur a failure indication. The E gyro scheduled failure was inserted in the B computer shortly after 2700 sec and the E gyro was taken offline about 50 sec later. The total squared error at removal was changing between 40 and 160 pulses squared because the maximum allowable squared error reverted toward its static value at the end of a 2-min accumulation period. At 2700 sec, the gyro fail status of computer B changed from F only to both E and F offline At 3500 sec, E gyro's bias error was removed and gyro fail status reverted to "F-failed-only "

The resulting attitude of the B computational body frame compared to the A body frame is shown in figures 24-29. These figures are complicated somewhat by the fact that the aircraft made a 90° turn while E gyro was degraded, but before it was taken offline. The scheduled failure caused an initial "Δ tilt about y" and "Δ azimuth" that reached a maximum of 2 2 arcmin and 1.4 arcmin, respectively.

Because of aircraft motion, the navigation position error does not fit a simple form. Figure 27 shows a maximum "speed" residual (difference in computers B and A) of 9.7 n. mi./hr. The longitude and latitude errors continued to increase after scheduled failure removal because of heading error, as shown in figures 28 and 29.

Assessment of inertially smoothed radio navigation- An analysis of the SIRU's potential for inertially smoothed radio navigation (i e., aided-inertial navigation) indicated that a DME/baro/inertial SIRU could easily maintain a navigation accuracy of a few hundred meters. These indications were obtained by performing the necessary calculations during postflight using flight test data.

The SIRU Navigation Analysis Program software was modified to enable computation of inertially smoothed radio navigation data using DME data for position fixes. The implemented scheme used a batch update in which incremental state changes were computed and put into the state estimate only once per minute. Figure 30 shows the position residual history between the inertially-smoothed DME-derived position and the time-averaged radar-derived position for SIRU flight test no. 9/05B. The top curve displays the northward residual, the middle curve displays the eastward residual, and the bottom curve displays the root-sum-square residual. The segmented appearance is caused by the discrete updating process of the Kalman filter. The maximum root-sum-square position error was about 0.2 n. mi. (~400 m) as compared to the free-inertial navigation maximum error of ~3.7 n. mi. on the same flight. Ground "fixes" were processed as measurements at both ends of the trajectory to use the information that the aircraft was stationary before and after flight.

DME ranges, which had no recoverable recorded time tags, were assumed to be measured at even seconds of navigation time when the range value was observed to change The DME measurement weight in the filter was obtained empirically by observing DME range residuals on radar data.

It may be possible to improve the failure detection and isolation capabilities of redundant systems such as the SIRU by external aiding. A Kalman filter using external reference data can be formulated to solve for sensor biases and to predict sensor measurements for use in failure detection and isolation calculations. This represents an interesting and worthwhile extension of this work.

Assessment of application to short-haul flight control- The concept of serving both navigation and flight control functions with a single (redundant) sensor package is appealing, especially from total system cost considerations. The SIRU was flight tested as the sensing package for navigation, but not for flight control. However, some sensor data from the flight test program were analyzed for integrated short-haul flight control applications.

Figure 31 shows equivalent-triad gyro and accelerometer data histories from the static, engines-running portion of flight 9/18B. The top three plots show measured angular increments sampled at 20/sec versus time. The lower three plots show accelerometer output versus time. The scattering (digital noise) observed in these plots is primarily caused by quantization of the sensors' output signals. The gyro resolution used for the SIRU was 44 arcsec/bit (0.0002 rad/bit), while the accelerometer resolution was 0.04 m/sec/bit. The scattering was caused by the mapping of six sensor outputs into three orthogonal components when each of the six individual integrating sensors could change output only by quantized steps. The range of this scattering could be reduced by improving the resolution of a bit.

The 1-sec average of one of the triad measurements is more representative of the real input than any one sample. In navigation use, the running time average of the sensor outputs to produce position and velocity increments provides smoothing prior to the calculation of attitude and navigation quantities. In flight control applications, the signals would require filtering to remove the quantization noise depicted in figure 31. With filtering, a flight control servo loop could probably use such signals.

Figures 32, 33, and 34 show equivalent triad sensor outputs (gyro and accelerometer) for takeoff, maneuvering cruise, and landing Figure 32 shows the takeoff phase, and figure 33 shows maneuvering cruise. Figure 34 suggests that the digital noise from the 44-arcsec quantization (without filtering) would prohibit the use of such data in flight control systems.

Another consideration is that an input gyro measurement with an upper limit of 1 rad/sec full-scale is too low for many flight control applications. However, scale changing for higher full-scale capacity without increasing word length would increase quantization values. These, in turn, would adversely affect the digital noise. Current technology can provide 1 6-arcsec quantization at 2 rad/sec full-scale. This should be a design goal of future strapdown test systems.

The SIRU's experimentally measured performance in failure detection and isolation would qualify the SIRU for flight control application, although near-simultaneous accelerometer bias-shifts took excessive time to detect and isolate. The levels of gyro failures which SIRU successfully isolated were in the noise level of conventional rate gyros. Isolating accelerometer bias shifts of 0.001 g in 1.5 min is adequate performance for short-haul flight control reliability.

Failure Detection and Isolation (FDI) Algorithm Performance

Discussion- The deterministic SIRU failure detection algorithm was based upon computing the sum total of the squared error of each inertial sensor type and comparing it against a dynamically changing threshold called maximum allowable squared error When the total squared error exceeded the threshold (maximum allowable squared error) by a certain ratio, a sensor fault was indicated.

The total squared error for each of the six inertial sensor sets is normally less than the internally defined maximum allowable squared error. The maximum allowable squared error values were extracted from the "normal" total squared error values observed in early flights. The minimum values of the maximum allowable squared error were determined by the sensor quantization. Each maximum allowable squared error was modified dynamically for a desired level of sensor error detection. Setting the maximum allowable squared error too low could lead to predictable but false "failures" during certain aircraft maneuvers. Setting the maximum allowable squared error too high could lead to undetected but real sensor failures The selected maximum allowable squared error values were a function of quantization, digital system noise, and aircraft maneuvers.

The total squared error failure detection algorithm was executed at each inertial (gyro or accelerometer) data update point. This theoretically enabled "failed" instruments to be taken offline without causing excessive perturbation to the computed aircraft attitude or velocity. The SIRU update rate was 20/sec.

Failure isolation consisted of computing the individual squared error of each sensor. The squared error for each instrument was formed by comparing a sensor's compensated output data with the redundant estimate of its output created from the other (similar) five sensors. These errors were summed over 2-min time intervals, compared, and the largest error sum used to indicate the faulted sensor. Usable values of total squared error and maximum allowable squared error depend upon accurate calibration and stability of the sensor's output. If the sensor output is stable, then its output accuracy is mainly dependent on the correctness of software compensation values and the completeness of the sensor error model. Likewise, the correctness of the fault detection algorithm output is dependent upon the calibrated compensation values and error models used to adjust each sensor's output signal. These compensation values and error models (particularly the angular rate-dependent terms), were known with a lesser degree of accuracy in the aircraft environment than in the laboratory where a precision rate table was used to measure them.

Strapdown accelerometers experienced only minor uncalibrated variations in specific force for a typical CV-340 flight as compared to the static situation. Their readings were very stable, and their calibration numbers remained nearly constant for over 1 yr. For the first flights, they were recalibrated prior to each flight For Flight 7/17 and to the end of the program, the static calibration numbers the laboratory measured a year earlier were used. Therefore, the accelerometer maximum allowable squared error was set at a constant value for the entire alignment/navigation flight test duration after Flight 7/17

The integrating rate gyros experienced major uncalibrated variations in angular velocity in the CV-340 flight tests as compared to the static case (11,000°/hr compared to 15°/hr). Gyros are less easily calibrated than accelerometers. Also, the compensation terms are nonlinear with high rates of change so the gyro maximum allowable squared error was set as a constant plus an angular change-dependent value. This angular change-dependent value was set proportional to the square of the net angular rotation that the system had undergone while accumulating the total squared error. This maximum allowable squared error value conformed reasonably well to typical gyro total squared errors during unfailed-instrument flight tests.

Scheduled sensor "failures" were imposed on 11 of the flight test segments by purposely miscompensating a sensor's null bias. Table 8 summarizes the results of the recorded in-flight scheduled failure tests of the failure detection and isolation capability. The level of failure shown is the amount by which the null bias of the sensor was shifted in the test. The levels shown in the table are from Draper Laboratory flight records which were not recorded on the flight tape.

Unscheduled Sensor Failures. The following is a discussion of the unscheduled inertial sensor axes failures which occurred during the flight tests. The conclusions as to causes of the failures are based upon the correlation of the removal of specific adverse operating conditions with the lack of recurrence of specific sensing axes failures. Table 9 is a tabulation of these indicated failures and the adverse conditions which apparently caused them.

The 43 unscheduled failures that occurred in Flights 1-6 could be attributed to improper sensor compensation. Failures which occurred after Flight 6 (7/17 through 9/24) could be attributed to thermal conditions. In two cases, the adverse condition consisted of cold air from the aircraft air conditioning jets blowing directly on the inertial component modules. This caused low temperatures in the C accelerometer and E attitude gyro. Another adverse condition was the high cabin air temperature (>88° F) which affected the E and F gyros and the E accelerometer in 11 different instances. The cold air problem was solved by distributing the air flow

The high cabin air temperature problem was never satisfactorily solved. It most frequently occurred after sitting on the ground in hot weather The aircraft's air conditioning did not have the ability to maintain the cabin air temperature below 88° F on the ground. No high-temperature-induced failures were encountered after reaching cruising altitude.

The SIRU initial failure maximum allowable squared error threshold was initially specified to be one and a half times the standard deviation of the total squared gyro error (total squared error) which was derived from benign laboratory tests. During SIRU flight tests, it was observed that the gyro noise level was greatly increased in the flight environment as compared with the laboratory environment. Therefore, after Flight 7 (7/17 through 9/24) the implemented failure threshold was updated to match the preceding flight dynamic data.

In Flights 1 (5/20) through 5 (6/25), the attitude gyro sensing total squared error limits were set at a maximum allowable squared error of 0 76°/hr,^{2} with a dynamic increase of 0.81°/hr.^{2} For Flights 6 and 7 (7/14 and 7/17), the limits were increased to 1.14°/hr.^{2} These limits were again increased to 1.14°/hr^2 and 1.98°/hr^2 from Flight 8 until the end of the program. The statistical failure detection, isolation, classification and recompensation algorithm was removed after Flight 7 because its operation was correlated with a high level of false alarms

The unscheduled failures, on an individual flight basis, were as follows:

Test Flight No. 1. (5/20): Failures - C and F Gyros. This was a shakedown flight without the 2-hr preflight calibration and warmup required for thermal stability. The primary reason for the failures was insufficient warmup time. This theory is supported by the fact that failures occurred immediately after going into the fine alignment mode and the fact that a C accelerometer low temperature fail signal was detected.

Test Flight No. 2. (5/30): Failures - All Gyros, A Accelerometer. This flight was preceded by an aided single position calibration. The resultant compensation had all of the gyro negative scale factor slopes accidentally zeroed, the wrong sign was entered for the A accelerometer bias, and the wrong value entered for the negative bias for the D gyro.

Test Flight No. 2 (5/30B): Failures - ACEF Gyros, A Accelerometer. In addition to the conditions that existed in the previous test, the cabin air temperature was 100° F.

Test Flight No. 3. (6/16A): Failures - ABCF Gyros, D Accelerometer. The accelerometer biases were being changed for each flight from results of an aided single or multiple position calibration which utilized a bubble level with a defective mount. The apparent misalignment introduced by the inaccurate accelerometer biases along with the low statistical failure detection, isolation, classification and recompensation algorithm limits accounted for the failures.

Test Flight No. 3. (6/16B): Failures - All Gyros, E Accelerometer. In addition to the conditions that existed for the previous test (6/16A), the cabin air temperature was 93° F. This caused the E accelerometer to fail.

Test Flight No. 4. (6/18): Failure - All Gyros, D Accelerometer. In addition to the adverse conditions stated for Flight 6/16A, this flight was preceded by an aided five-position calibration which utilized a wrong null bias drift for the D gyro. Its drift equivalent was 0.4°/hr which corrupted the other compensation parameter values derived from this calibration. These adverse factors accounted for all the failures.

Test Flight No. 5. (6/25): Failures - AECD Gyros, BE Accelerometer. All of the adverse conditions stated for Flight 4 (6/18) were also present for this flight and account for the failures. The C and D gyro failures were correlated with aircraft turn maneuvers and could be attributed to both the improper accelerometer biases and the low statistical failure detection, isolation, classification, and recompensation algorithm limits which were 0.76°/hr maximum allowable squared error with a maximum dynamic increase of 0.81°/hr^2.

Test Flight No. 6. (7/14): Failures - ABCDF Gyros. All of the adverse conditions stated for Flight 5 (6/25) were present for this flight test and account for the failures. Again, there was a correlation between the C gyro failure and the aircraft turn maneuvers.

Test Flight No. 9. (7/29A): Failures - BEF Gyros. The E gyro failure resulted from cold air blowing directly on the E gyro module. The C accelerometer module also indicated a temperature fail condition.

> Note: After the blowing cold air problem was recognized, the cold air was redistributed away from the sensor modules.

Test Flight No. 9. (7/29B): Failure - E Gyro. The E attitude gyro failed during fine alignment and was caused by blowing cold air as in Flight 9 (7/29A)

Test Flight No. 10. (8/22B): Failures - E Gyro, E Accelerometer. These failures were caused by high cabin air temperatures while sitting on the ground at Crow's Landing.

Test Flight No. 10. (8/22C) Failures - E and F Gyro, E Accelerometer Same as Flight 8/22B.

Test Flight No. 10. (8/22D): Failure - E Gyro, E Accelerometer These failures were caused by high cabin air temperature while sitting on the ground at Bakersfield.

Test Flight No. 12. (9/05B). Failure - F Gyro This failure was caused by high cabin air temperature.

Test Flight No. 12. (9/05C): Failure - F Gyro. Same as Flight 9/05B.

Test Flight No. 12. (9/05D): Failures - EF Gyros Same as Flight 9/05B.

Test Flight No. 14. (9/18D). Failures - EF Gyro, E Accelerometer. These failures were caused by high cabin air temperature.

The failure detection and isolation algorithm identified deficiencies in gyro dynamic modeling by indicating a "failure" in the sensor system. As indicated, it was necessary to increase the operating failure threshold (maximum allowable squared error) to be compatible with system dynamics in the flight environment. This defective dynamic error compensation was illustrated by the data from Flight 14 (9/18B) which was analyzed in detail

The standard deviations of the six gyro parity equation residuals from Flight 14 are presented in table 10. The parity residuals were sampled at 30-sec intervals. Because of the limited duration of this flight, only 14 data points were available to be used in computation for the 3°/sec counterclockwise turn, 16 data points for the 3°/sec clockwise turn, and 6 data points for the level flights. The maximum values for the standard deviations were:

Turns	Level Flight
2.88°/hr	1.01°/hr

For a comparison, two data sets from dynamic tests conducted in the (Draper) laboratory environment are presented: one set from slew tests (5°/sec) and from oscillatory tests.

The parity residual standard deviation for slew (turn) tests was computed to be 0.24°/hr. The laboratory test data were sampled at 2 min intervals. The standard deviations corresponding to 30-sec sample intervals for the laboratory test data are found to be:

Lab Slew Data	Lab Oscillatory Data
0.48°/hr	0.2°/hr

A comparison of the flight data for turning (2.88°/hr) and the slew test data (0.48°/hr) shows an increase of the standard deviation in flight test data by a factor of 6. A comparison of the level flight data (1.01°/hr) with the laboratory oscillatory tests (0.2°/hr) shows improvement might be expected if improved dynamic models were used for the aircraft environment.

SIRU Hardware and Software Limitations

During flight tests in the CV-340 aircraft, certain limitations were experienced with respect to the SIRU hardware and software. Many of the limitations were the result of the original SIRU design goal for a post-Apollo advanced spacecraft inertial measurement unit.

Hardware- The 18 IRIG MOD D integrating rate gyro was designed for continuous operation in a spacecraft environment. The orientation of the output axes was optimized for the space mission, and therefore this orientation was not optimum for an aircraft environment. Because of the operational safety requirement to turn the system off when unattended, gyro parameter shifts were encountered across cooldown and powerdown which necessitated frequent calibrations. The quantization level of 44 arcsec was adequate for long term space missions with periodic external updates, but was not adequate for the short-term high-vibrational environment of the CV-340. The rate limitation of 1 rad/sec, again, is no problem in a spacecraft, but does represent a limitation in an aircraft. A 2-hr warmup requirement prior to a 2-hr calibration is no problem in the long countdown of a spacecraft but is definitely unacceptable for an operational aircraft (appendix G, Volume III).

The 16PM PIP specific force sensor (accelerometer) and its torque loop were originally designed for the Poseidon missile. This accelerometer's performance was adequate in all respects in the aircraft environment and therefore posed no limitation. Its calibration was stable over a period of 1 yr.

Figure 35 shows both the theoretical and experimental failure detection and isolation time of the attitude sensors as a function of failure magnitude Also indicated in figure 35 are flight control application requirements of which inertially smoothed landing guidance requires the shortest failure identification and removal time.

The torque rebalance loops used in SIRU (dictated by Apollo Technology) were tailored to 44 arcsec angular resolution for the original spacecraft application. It is apparent from figure 35 that this size is too large for aircraft operations. Also included in figure 35 is a theoretical 1.6-arcsec resolution (available in current strapdown INS technology) plot of attitude failure rate versus failure identification and removal time This increased resolution would provide considerable improvement over the 44-arcsec resolution results.

The SIRU sensor pallet mounting fixture was designed to permit calibration of the SIRU sensors in the aircraft. It provided positioning of the inertial frame package to four cardinal points (90° rotations) about the system Z axis and one 90° rotation to put the Z axis in the vertical or horizontal plane. This permitted placing the ±X, ±Y, and +Z axes down individually for multiple position calibration. The design did not allow a -Z axis down orientation, however. This missing sixth position could have improved the calibration accuracy.

A pair of wedge rings was provided to permit leveling the system prior to the aided single- or multiple-position test. The sensor level was measured by a 10-arcsec bubble mounted on the SIRU frame in a recessed well so that it could be rotated 360°. The 10-arcsec bubble mounting surface had been precision machined with respect to the system reference cube. With repeated operations, its aluminum surface became worn and the repeatability of the measured accelerometer biases was uncertain. The bubble continued to be used in leveling the system, but the accelerometer biases were not changed.

The SIRU sensor pallet cooling system was limited to an ambient air temperature range of +40° to +88° F. The inertial component temperature controllers could not control the gyro and accelerometer temperatures outside this range. Lack of control caused sensor parameter shifts which would be detected by the failure detection and isolation software as a sensor transient failure.

The aircraft air conditioning system was not functioning properly in the early part of the flight test program and caused ambient air temperatures to exceed 90° F. The system was never adequate when the aircraft was sitting on the ground and the outside air temperature exceeded 90° F Operation of the air conditioning (which was always needed at Crows Landing) required that one engine be running. This limited the accuracy of the calibration. The inadequate ground cooling system used to correct this problem caused overcooling of some inertial components (when the cabin temperature was normal) because the outlets of the aircraft air conditioning were exhausting cold air directly onto these inertial components. This over-cooling caused uncompensated temperature gradients in the sensors, and this triggered a transient failure report from the failure detection and isolation algorithm. This condition was corrected once the source of the problem had been isolated.

Software- Postflight analyses indicated that the base motion isolation software mechanization did not perform as well as expected and was the cause of excessive errors during fine-alignment

Using computed body velocity at the end of each 1-sec navigation update to approximate the average velocity over that update caused acceleration induced position errors as large as 52 m (170 ft) for 180° turns. These averaged to zero for straight line acceleration and deceleration. (Volume III discusses this error in detail.)

The statistical failure detection and isolation algorithm threshold limits were specified to be one and a half times the standard deviation of the gyro error derived from the laboratory test environment. Flight test experience showed that the gyro errors were greatly increased as compared to the laboratory dynamic test data. The original limits were 0.06°/hr for static conditions and a maximum dynamic increase of 0.12°/hr to give a maximum threshold of 0.18°/hr These were changed to 0.48°/hr (static) with a 0.96°/hr maximum dynamic increase to give a maximum threshold of 1.54°/hr. These increases were derived more intuitively than analytically and proved to be inadequate as evidenced by the large number of "false" failures. The software was then modified to remove the statistical failure detection, isolation, classification, and recompensation algorithm in the navigation and flight modes (after Flight 7). The failure detection software was then totally dependent upon the total squared error algorithm.

The total squared error algorithm failure detection thresholds (maximum allowable squared error) were initially determined on the basis of laboratory tests without compensation for aircraft dynamics. The total squared error threshold was both mission and environment sensitive. The initial gyro total squared error maximum allowable squared error was $(0.76°/\text{hr})^2$ with a maximum dynamic increase of $(0.81°/\text{hr})^2$ to produce a maximum threshold of $(1.57°/\text{hr})^2$. Early flight test data indicated false gyro "failures" were being triggered during aircraft turn maneuvers because these limits were low. The limits were raised to $(1.14°/\text{hr})^2$ (static) and $(1.98°/\text{hr})^2$ for maximum dynamic increase giving a maximum threshold of $(3.12°/\text{hr})^2$. Also, the accelerometers, initially set at $(0.4\ \text{cm/sec})^2$ for first fail maximum allowable squared error and $(0.13\ \text{cm/sec})^2$ for second fail maximum allowable squared error, were reset to $(0.28)^2$ and $(0.25\ \text{cm/sec})^2$, respectively. These changes removed the false failure indications during aircraft maneuvers.

In summary, the tests showed that the failure detection software worked, but not well. The greatest improvement could be obtained by better dynamic modeling of the operating environment and its effect on normal sensor output.

LABORATORY EVALUATION OF THE DUAL COMPUTER SYSTEM

The SIRU dual computer system was analyzed at the Stanford University Digital Systems Laboratory. Emphasis was placed upon reliability and completeness of fault detection. In this chapter, some considerations are first presented relative to assessing computer-system reliability. An analysis of the dual redundant system's selftest effectiveness, arbiter function performance, and preflight testing abilities are then discussed. This is followed by a description of suggested changes which would improve the overall reliability of a future system.

Basic Design Considerations

In order to evaluate SIRU's reliability, some definitions first had to be established. In n-modular redundant computer systems, a voter module decides the correct final output based on all the information from the redundant modules. If the voter is nonfaulty, then the final output would be correct even though some of the redundant modules could be faulty.

In the SIRU implementation, there was no hardware voter; instead, there existed an arbiter which formed an educated guess as to which (if any) of the two computers was faulty. The arbiter used error detection techniques which did not inspect the actual output data. There was no switching of data as in a hardware voter The human operator observed both the output displays and the arbiter opinion Thus, the "voter" for the SIRU system was actually the operator, despite the automatic prime select mode.

For the purpose of this evaluation, it was decided to assume that the arbiter acted like an n-modular redundant voter unit (i.e., as if the arbiter output controlled a fault-free switch, gating only one of the two data outputs to a single display). A system failure occurred when incorrect output data was selected as correct by the arbiter. This allowed the results to be more directly compared to other simplex and n-modular redundant systems.

The basic design philosophy of the SIRU computer system appeared to be a sound one. A dual computer configuration was an effective way to achieve an increase in reliability because efficient usage was made of hardware resources. For example, in a triplicated system with voters, system failure occurs when two or more channels have failed. Thus, one good channel may still exist but cannot be used unless the system is able to degrade to the simplex configuration after the second failure. On the other hand, in a dual configuration with an arbiter, system failure could occur only when both modules have failed; all modules are used

A dual computer system should be designed so that one failure will not fail the entire function. This implies the presence of not only good error detection, but also a high degree of isolation between the two channels so that errors will not propagate from the faulty computer to the good one. The SIRU computer system achieved very effective electrical isolation between the various modules by using opto-isolators at all points of physical contact This assured that a fault in one module had almost zero probability of adversely affecting any other module.

The propagation of error through data and flag signals also had to be considered. The method proposed by Ressler (ref. 8) of transferring data between the two computers ensured that the nonfaulty computer would never receive contaminated data when only one fault had occurred in the system. It was not possible for the faulty computer to "lock-up" or halt the nonfaulty computer. Because status data were only received through a self-request, the nonfaulty computer would never accept bad data from the other computer.

In order to use both computers, the arbiter had to be able to decide which (if either) of the two computers was faulty. Due to the complicated and extensive error-detection facilities built into the system, the arbiter was sure to make a correct decision, given the occurrence of a single permanent fault. Here, "single fault" refers to the more mundane and expected mechanical and electrical faults.

It is always possible, when analyzing a dual system such as the SIRU design, to postulate some unusual fault consisting of multiple changes or events and to demonstrate that the arbiter would make an incorrect choice in that situation. Discussing such events is meaningless unless the probability of their occurrence is significant with respect to more simple faults. In the case of the SIRU, the large number of components in both the computers and the error-detection circuitry increased the failure rate for the system as a whole. Thus, a complete evaluation of the system included aspects concerned with multiple faults, such as recovery and arbiter functions. The SIRU system was clearly not secure from failure due to a conjunction of two or more simple faults.

Self-Test Routines

The SIRU computer system was designed to reduce the probability that one fault would fail the mission, this would happen if one computer failed and the arbiter made an incorrect decision. To evaluate the SIRU dual computer system, a class of faults to be detected by SIRU had to be specified. This fault set was a function of how the various subsystems were implemented in hardware. (For example, the fault set for a standard core memory might not be suited to the type of error detection required for a semiconductor memory. Therefore, statements about the SIRU hardware systems might not apply to alternate hardware implementations of the SIRU system.)

The Honeywell 316 computers used in the SIRU flight test readiness system had central processors built almost completely with NAND gates on small-scale integration chips. The memories had 12K 16-bit words of core almost entirely taken up by navigation programs, and did not use parity checking. The self-diagnosis for each processor consisted of reading its own data word from the T-box register, executing a series of instructions meant to test the control logic and data paths, and computing checksums on those portions of memory whose contents should not change. There was a significant probability that the software self-tests would never detect a computer failure because the fault would not allow correct execution of the self-test programs.

Figure 36 depicts the SIRU processing cycle in which each computer passed through states A, B, C, D, E between each SIRU interrupt. For a real fault, the program flow would be different. For example, if the program-counter clock line was stuck at 1, the program would repeatedly execute the same instruction as long as the fault was active After the fault became inactive, the self-tests would be positive, and the cross-checking would be negative, even though the output data would be incorrect as illustrated in figure 37. The self-test program would not find this type of error because when it was finally executed, the fault would no longer exist.

Clearly, if the fault set to be detected were the set of all single "stuck-at" faults on the pinouts of the small-scale integration chips or on address and data bus lines, then the self-test programs would not be effective diagnostic tools because of the anomaly which could occur due to a transient fault. These diagnostics were executed only once for every SIRU interrupt cycle. The rest of the time, about 12K words of navigation programs were being executed. Thus, between the time that a fault occurred and the time when the diagnostics should be executed, the memory and central processor unit were being exercised with a very wide variety of instruction types, addresses, data and test conditions.

If a single stuck-at fault existed in memory, it would more likely be the fault of an address decoder or a single bit of data. The fault would affect memory references over a large number of addresses and perhaps the entire address space

There was a small probability that a fault which would allow the program to progress through all the navigation software to the entry point of the test routine would occur causing the diagnostic routines to execute correctly to completion and to send the correct result to the T-box. A more realistic assumption would be that the program counter would randomly skip through memory until a "steady-state" would be reached in which control continuously loops through a set of random memory locations.

In the SIRU system where both instructions and data were in unprotected memory, the loss of control due to a simple memory fault could result in the destruction of program areas including the sections reserved for self-diagnosis, cross-diagnosis, interprocessor communication and recovery software. For fault detection only, as is the case in the SIRU software, the destruction of kernel software would be desirable because the arbiter then has more definite indications of a failure in that processor

An important observation made of the SIRU system which applies to reconfigurable systems was that damage to software from hardware faults should be limited so that only navigation status locations need be transferred during recovery Also, a high probability should exist that all recovery software remain intact. For this reason, it would be better to implement the fixed program memory in an operational SIRU derivative computer system using read-only memories and a random-access memory (read/write) for all temporary and status locations.

A similar problem existed with simple stuck-at faults occurring in the main registers, the instruction decoder, and the arithmetic and logic unit. The Honeywell 316 used the same gates and data paths for both the program counter modification and all accumulator operations Thus, any fault which would cause errors in the data calculations would likely affect the program counter in the same way (This problem would apply even more to current microprocessors in which the program counter and assorted address registers are in the same physical loop and use the same arithmetic units as all the accumulators.) The program execution of diagnostic software which would test the required circuitry would very often not take place. Redesigning the system to have more independent subunits might alleviate this problem, as long as the number of components is kept within the bounds imposed by power, space, cost and total failure rate

This evaluation makes clear the problems of using stored-program self-diagnostics in a typical minicomputer or microprocessor.

.Arbiter Function

Because there are so many components in both the SIRU computers and the transmission channel, it was necessary to consider the behavior of the arbiter when the system had multiple faults. In actual operation, many faults would be transient so the manner in which the arbiter "remembers" previous error detection would be very important. In the SIRU flight test system, all the error detection bits feeding the prime-select circuitry were not latched from cycle to cycle. The error detectors all fed the "A-Fail," "B-Fail," and "system-Fail" latches, which did not affect the prime-select circuitry and could only be reset by the operator. The cross-opinion bits were displayed to the operator; they were not latched between cycles and did not affect the prime-select. The operator could manually select which computer was to be prime.

In certain situations, the operator had to supplement the action of the arbiter to form the optimal system response to the situation. For example, suppose computer A had an internal transient fault which contaminated its status data and recovery was not implemented. The operator would observe the "A-Fail," and both disagreement indicators would light up. It would be possible that the transient fault was of short enough duration that the changes at the panel were not observed until after the fault became inactive If the operator did not switch prime manually to B, then a future transient error in the B transmission channel or error detector could cause the arbiter to choose A as prime, and thus the system would fail. If the operator switched prime to B manually, then the system could only fail when a fault occurred in the B computer, or a permanent fault occurred in the B transmission channel.

On the other hand, suppose that a transient error occurred in the A transmission channel. Now the operator would observe the "A-Fail" indicator lit, but there would be no disagreement between the two computers. In this case, the operator should only reset the "A-Fail" light, and should not switch to B manually, because the A channel would still be providing correct data, and it would decrease the reliability not to use it after that time. It would be important that the operator reset the "A-Fail" light, because otherwise he would not be able to distinguish later between a transient failure in the A computer (as described above) and a loss of communication between the two processors without wasting valuable time to visually compare the displays. A loss of communication between the processors would appear as both computers indicating disagreement. However, neither would have latched the fail indicator. If recovery were implemented and good status data successfully restored to a failed computer, the arbiter panel would appear the same as it would in the case of an intermittent transmission channel fault.

It would be useful to automatically reset the fail indicators when both cross-opinion bits show agreement so that the operator did not have to decide when a failed channel or computer had recovered. It would also be useful to use the "A-Fail" and "B-Fail" latch outputs as additional inputs to the prime select in anticipation of the manual selection by the operator. The operator should retain a complete set of manual override controls so that faults arising in the arbiter, error detectors, and inter-computer communications would not fail the computer system.

The T-box register was well designed so that it was not possible for a single fault to send incorrect data to both the other computer and the display while sending correct data back to the same processor when it tested its input/output. Thus, the operator would always be able to distinguish an intermittent channel fault from errors caused by incorrect status words within the computer by observing the opinion bits (as long as at least one complete computer and channel would be operational and correct).

Preflight System Testing

One problem associated with SIRU's extensive computer error detection was the inability to adequately test it during preflight maintenance. The central arbiter circuitry could be tested if the error detectors which send data to it could be manipulated so as to provide a complete set of arbiter inputs. This was not possible unless all the error detection circuitry would be fully tested. If an error detector failed so as to incorrectly indicate an error in the computer or channel it was checking, then preflight maintenance would continue until the fault in the error detector was located and repaired. But if the error detector failed so as to constantly indicate "no error," then normal maintenance would not reveal any faults, and the system would be cleared for operation with a fault. Consequently, the single fault assumption would no longer apply, and it would be possible that one more fault would fail the mission.

It is necessary to design a future system with the ability to thoroughly test all error detectors and the central arbiter circuitry, either by using self-checking techniques or by exercising all components with auxiliary diagnostic equipment. This was possible in the SIRU implementation for the watchword checker, the self-diagnostics, and the computer time-out generator. It was not possible for the parity generation, bit counts, and clock active signals. Table 11 summarizes the testable and untestable SIRU subsystems.

A practical problem related to the SIRU computer system was the large number of one-shot multivibrators employed for both time-out checking and clock generation. These devices are inherently less stable than crystal generators. Using such a large number of these devices significantly increases the probability of a timing failure. The problem of testing the multivibrators for time delay drift was also very acute, particularly since the existing system had no facility to inject specific signals to the one-shot inputs and directly observe their outputs. Some testing could be done if it were possible to dynamically change the behavior of parts of the circuitry which were being checked by the one-shots.

The method of testing depends partly on the available inputs and outputs of the system. Because the correct operation of each error-detection device must be verified, the direct output of each device should be available either as a panel light or as an outlet to a maintenance jack. If the signals from several error-detection devices are merged into one "fail" signal and are not available separately, then the tests must be designed to activate only one error detector at a time, so that the result would not be masked by other values.

The SIRU flight test system was designed in such a way that the parity checker, the bit count and the clock-active error detectors could not be adequately tested. The parity checker could be tested if an auxiliary input from the maintenance equipment were exclusively ORed with the output of the parity generator in each T-box. The test software would have to change each bit input one at a time to exercise both parity trees, and the extra input would exercise the final output of the receiver parity checker.

If each T-box and receiver had a single crystal-controlled clock instead of numerous one-shot multivibrators, the system would be much easier to test. Dynamic tests would be required only for the single clock output, rather than the inputs and outputs of each one-shot. The clock would feed a set of simple one-chip sequential machines which could be tested at a functional level only. Therefore, they would not require a variety of difficult time delay tests.

Another way to test these error detectors would be to vary incrementally the time delays of the circuits being checked and to measure the "error window" detected by the error detector For example, the computer could send opinions to the time-out generator of the T-box at a variety of different time delays from the SIRU interrupt. The auxiliary test equipment would record the time-out error indicator at the arbiter to know what range of time delay was being marked as erroneous. This would be the best method if it could be implemented. However, the SIRU design did not allow easy incremental change of any time delay except that of the computer outputs. Also, there was the problem of not being able to gauge the incremental time delays correctly without disconnecting a large part of the circuit under test.

Another possibility would be to use self-testing error detectors as described in reference 9. This could only apply to the strictly combinational error detectors such as the parity generators.

Reliability Improvements

The design of the SIRU dual computer system was basically a good one from the standpoint of both correctness and reliability. Its strong points were the intermodule isolation, the intercomputer communication protocol, the wide variety of error detectors, and the effectiveness of the arbiter function. The SIRU computer arrangement also had several weak points which did not affect the normal operation of the system, but do suggest design improvements which could improve the overall reliability and maintainability of a future system. These include improvements in the preflight test facilities, an extension of the arbiter function, self-tests as applied to failure recovery, the computer software structures, and the use of different hardware components.

Computer failure recovery- The question of computer recovery after a transient fault is an important one because the aircraft environment causes many intermittent failures due to gusts, mechanical shock, vibration, radio interference, heat gradients, and power failures. If the duration of transient faults is small compared to the mean time between faults, then an effective recovery process can greatly improve the system reliability.

In order for the recovery process to work properly, there must be no fault in the recovering computer, the computer must know that it needs to recover, and it must be able to communicate with a correct computer. If read-only memories were used, then the absence of a fault would imply that the recovery software was intact. The communication protocol proposed in reference 8 would allow the transfer of state information from the correct to the incorrect computer. Here the status transfer would be just an extension of the cross-checking procedure, and very little overhead would be required in software to implement recovery. Again, if the programs were stored in read-only memories, only a small number of words need to be transferred.

The major difficulty in the proposed method of recovery would be that the computer would probably never know when it is faulty or when it has incorrect status words. As explained earlier, most actual faults would not allow the correct execution of self-diagnostics or recovery software, so that while the fault was active, the computer would act in a random fashion. When the fault becomes inactive, the self-tests would have no fault to find even though the status data were contaminated, and the computer would continue its normal processing cycle never realizing that it should attempt recovery.

If each computer can detect its own failures, then the system reliability would increase. There are several ways to accomplish this without relying on special software or extensive additions to the hardware. The arbiter output could be sent back to the processors, perhaps through triply modular redundant transmission channels, indicating the cross opinions and the latch "A-Fail," "B-Fail" signals. This would not be an ideal solution because the arbiter could always distinguish between faults in the computer and faults in the transmission channel Thus, a correct computer with an intermittently bad channel might load status words from a normally operating computer which has incorrect navigation data, thereby causing a system failure.

A better solution would be to have hardware detection at each computer. One possibility would be to use self-checking processors similar to those discussed by Wakerly (ref. 9), which could achieve a high degree of error detection with only a 30% increase of hardware. Such a processor would supply data to the transmission channel in a form already suitable for use by error-detecting codes. It would have the added advantage of using state-of-the-art large-scale integration technology to make the system more compact and reliable.

A more efficient method would use the fact that error detection need not be immediate and, therefore, could wait until the fault causes abnormal operation at some point in the program before the opinions were sent to the T-box. If the fault would not allow the diagnostics to execute, then the timeout generator in the T-box would probably be set to indicate an error. If the timeout flag were reset only by a signal of data agreement from the other computer, then the faulty computer could read its own flag (after its transient fault becomes inactive). The faulted computer would know that it should attempt recovery as long as the other computer says it has incorrect output. Because each computer could only reset (and not set) the flag of the other computer, a faulty computer could never force a nonfaulty computer to accept its incorrect status words. This minimum hardware solution complements any software diagnostics by making use of the excellent error detection already built into the SIRU system.

Software structure- The software supplied with the SIRU flight test system was adequate in the sense that it seemed to work correctly. All programs were written in Honeywell 316 assembler language and assembled by the standard Honeywell software.

Despite the fact that all the programs were probably correct (no errors were found), it is not unreasonable to assume that errors exist in a program which has not been proven correct. This would especially apply to programs such as the SIRU navigation software because the involved mathematical routines could not be tested by applying all possible inputs and comparing the outputs with known correct values. Unexpected situations could occur when a particular combination of input values corresponds to a singularity in the formulae being manipulated. This would cause the same incorrect result in both computers. Proving the correctness of the software would guarantee that the dual computer system reliability was not a function of the inputs to each computer.

There are many studies (ref. 10) which deal with the details of proving programs correct. The task would be simplified by using a higher level language with block structures and preferably without "GOTO's." The program would be proven correct and then compiled with a compiler which has been proved correct. Although this degree of proof is admittedly not possible with the state-of-the-art software engineering, an evaluation of the SIRU system reliability would not be complete without its consideration.

Component selection- Since failure rates are proportional to the total number of components in the system, it is important to use as few components as possible. This policy always involves a tradeoff between the failure rate and the effect of a single failure, as discussed previously. In the SIRU system, the effect of a fault within the processor had no great significance if recovery were not implemented. The computer would always continue with incorrect status data, and every fault would completely fail that processor. Because recovery could occur while a fault existed and abnormal operation of the processor could be detected by the processor itself, then the effect of an active fault was not significant when recovery was implemented. This assumes that all the recovery software was stored in read-only memories.

It is, therefore, feasible to use large-scale integration components in the future design of a SIRU derivative system. The use of microprocessors instead of an off-the-shelf computer would significantly decrease the overall cost, space, power, weight and failure rate of the system, but would probably limit the speed and accuracy of computation. The number of bits required for high accuracy could be obtained by using a byte-sliced chip organization at the cost of less speed.

As the failure rate of the processor is reduced, the number of components in the transmission channel and arbiter circuitry have a more significant effect on the system failure rate. At some point, it would increase the reliability to use a triplicated or n-modular redundant configuration with a much smaller degree of error detection and arbitration The arbiter itself could be implemented with a microprocessor so that it could more intelligently analyze fewer inputs. The arbiter could be triplicated to reduce the probability of single-point failure (ref. 8). The current large-scale integration technology makes such architectures desirable from the standpoint of reliability. Various methods for fault-tolerant synchronization of n-modular redundant microprocessors and methods of selective voting to achieve higher reliability are discussed in reference 9.

The SIRU flight test system had a direct data line from the sensors to the computers and a complex error-checked transmission line from the computers to the operator display and arbiter. If large-scale integration components were used in much of the system, the processors would be small enough to be placed at the display instead of at the sensors, thus eliminating the need for the transmission line between the processor and the arbiter. The arbiter would not have to be concerned with transmission channel errors.

Unfortunately, this type of design introduces other serious problems. Assuming the sensors could be located at the display, dual transmission lines would be required from the sensors to the computers. If the computer received erroneous data, then the navigation status of the computer would be incorrect until it could recover nonfaulty data from the other computer. If the computers were at the sensors, then a transmission channel fault would not affect the status, and if a transient fault became inactive, the display would continue to receive accurate data. Thus, if the computers were located at the display and if one computer were incorrect, a fault in the channel of the other computer would cause it to fail. The fault would not fail the other computer if they were located at the sensor. Consequently, it appears that this aspect of the design should not be changed in a future SIRU-derivative navigation system which uses large-scale integration components.

CONCLUDING REMARKS

The principal objectives of the SIRU flight test program were to make the following assessments:

1. Determine SIRU's basic performance as a skewed-sensor free-inertial navigation system as evidenced by the navigational accuracy achieved during the flight test.

2. Determine the capability of the SIRU failure detection and isolation algorithms to detect and isolate sensor failures

3. Judge the operating performance of the SIRU parallel dual redundant computer configuration and its compatibility with the hexad sensor arrangement.

Conclusions made from the program test results as applied to each of these objectives follow.

Navigation System Performance

Test results showed that the SIRU inertial system drift accuracy is 1-3 n. mi /hr, which is generally acceptable for short-haul aircraft operations. Navigation performance during terminal area maneuvers (including degradation in the presence of scheduled and unscheduled failures) produced higher error buildup. However, even without preflight calibration, these error rates did not exceed 5 n. mi./hr. In summary, it was demonstrated that a system like SIRU can provide the inertial navigation capability necessary for short-haul applications. However, additional analysis, aircraft operations, oriented software design, and further testing are required to minimize dynamic errors resulting from uncompensated instrument misalignments and scale factor errors. Inertial sensors with long term calibration stability are essential for strapdown applications to short haul aircraft systems

SIRU's potential for providing inertially smoothed radio navigation (i.e., aided-inertial navigation) indicated that a DME/baro/inertial system based on SIRU could easily maintain a navigation accuracy of a few hundred meters. For one flight, the maximum aided-inertial RSS position error was about 400 m (0.2 n. mi.) compared to a free-inertial maximum error of 3 7 n. mi. on the same flight.

The SIRU gyro resolution was 44 arcsec/bit (0.0002 rad/bit), and the accelerometer resolution was 0.04 m/sec/bit. Data analysis showed that the quantization-derived digital noise from the 44-arcsec/bit resolution (without filtering software) would prohibit use of such data in short haul flight control systems. Also, SIRU's gyro measurement capacity of 1 rad/sec full-scale is too low for many flight control applications. Scale changing to achieve higher full scale without increasing the word length would increase the quantization error. Thus, substantial modification to both hardware and software of the SIRU would be required to meet flight control sensing requirements.

Failure Detection and Isolation Algorithm Performance

The SIRU flight experiments showed that achieving a 0.1°/hr failure detection and isolation level is more difficult than originally projected for aircraft operations. The failure detection and isolation algorithms were not able to detect gyro failures smaller than 1.5°/hr because the detection threshold had to be adjusted to a higher value to avoid false alarms. This threshold was a factor of ~20 times larger than the drift errors detected in the laboratory environment. This practical threshold determined in flight indicated that the existing SIRU system can detect and isolate only "hard failures" (i.e., hard from an inertial navigation point-of-view). However, this detection level is adequate for providing redundant operations for flight control purposes.

The analysis of the flight results was directed towards comparing the experimentally measured failure isolation time with the theoretical isolation time. This comparison is related to the increase in digital system noise derived from motion dynamics. Its assessment provides technical insight for assessing other redundancy management strategies for short-haul avionics.

These analyses have shown that the existing SIRU hardware and software, which was designed as a post-Apollo advanced navigation system, has a number of limitations in short-haul applications. The results point to areas where additional design and verification effort are required before a redundant strapdown system could be considered for operational usage. The fault detection and isolation capability is the area in need of the most work if the SIRU concepts are to become applicable for short-haul aircraft. Other fault detection and isolation concepts need to be studied and compared to the SIRU system's redundancy management scheme, which is based upon the use of the parity residual.

Dual Computer System Reliability

The design of the SIRU dual computer system was basically a good one from the standpoints of both correctness and reliability. Its strong points were the intermodule isolation, the intercomputer communication protocol, the wide variety of error detectors, and the effectiveness of the arbiter function. The SIRU computer arrangement also had several weak points which did not affect the normal operation of the system, but do suggest design improvements which could improve the overall reliability and maintainability of a future operating system. These include the self-tests as applied to failure recovery, the computer software structures, the preflight test facilities, the use of different hardware components, and an extension of the arbiter function.

The major difficulty with the SIRU method of failure recovery is that the computer would probably never know that it has incorrect status words as a result of an intermittent fault When a fault is active, the computer would act in a random fashion. Most faults would not allow correct execution of self-diagnostics or recovery software. When the fault becomes inactive, there is no fault to find, and the computer would continue without realizing that it should attempt recovery.

The SIRU software was adequate in the sense that it seemed to work correctly for a single failure. However, it is reasonable to assume that imperfect algorithms exist in all computer programs. This would especially apply to complex programs such as the SIRU software because of its involved mathematical routines which cannot be tested by applying all possible inputs and comparing the outputs with known correct values in a finite time period For example, the SIRU preflight test system was designed in such a way that the parity checker, the bit count, and the clock-active error detectors could not be adequately tested. (Details of the need for preflight testing ability and how system improvements could be made are discussed in the previous chapter.) An imperfect calculation could be interpreted as a single failure which would be indistinguishable from an actual failure. However, a hardware failure would probably cause multiple computer failures which could not be traced to the failed computer by the arbiter.

For a very simple computer system, the SIRU arbiter concept might be satisfactory, but its suitability for the redundant computational problem of the SIRU system is doubtful.

REFERENCES*

1. Eberlein, A. J.; and Savage, P. G.: Strapdown Cost Trend Study and Forecast. NASA CR-137585.

2. Bjorkman, William S.: Final Report for Development of Aided Inertial System Computer Programs. (AMA Report No. 74-37, Analytical Mechanics Associates, Inc.) NASA CR-137641, 1974.

3. Bjorkman, William S.: SIRU Flight Test Summary. AMA Report No. 76-1, Analytical Mechanics Associates, Inc., January 1976.

4. Booth, Robert A.; Gilmore, Jerold P.; and Hruby, Ronald J.: Test Results and Flight Preparation for the Strapdown Redundant Inertial System. Institute of Navigation Annual Meeting, San Diego, California, June 1974.

5 Gilmore, J. P.; and Cooper, R. J.. SIRU Development — Final Report. Volume 1, System Development, CSDL R-747, Charles Stark Draper Laboratory, July 1973.

6. Musoff, Howard: SIRU Utilization, Volume 1, Development and Test Evaluation, CSDL R-746, Charles Stark Draper Laboratory, March 1974.

7. Booth, Robert A ; and Shuck, Thomas L.: SIRU Flight Test Readiness Final Report. CSDL R-934, Charles Stark Draper Laboratory, 1976.

8. Ressler, Bruce E.: Design of a Dual Computer Configuration for Redundant Operation. CSDL T-586, Charles Stark Draper Laboratory, June 1973.

9. Wakerly, J. F.; and McCluskey, E. J.: Design of Low-Cost General-Purpose Self-Diagnosing Computers. Proceedings of the IFIP Congress, 1974, Stockholm, Sweden, 1976, pp. 108-111.

10. King, J. C.: Proving Programs Correct. IEEE Trans. Computers vol. C-20, November 1971, pp. 1331-1336.

*Other reports in this series are:
Flight Test Results of the Strapdown Hexad Inertial Reference Unit (SIRU).
Volume I - Executive Summary, TM X-73,163.
Volume III - Appendices, TM X-73,224.

TABLE 1.- INERTIAL NAVIGATION SYSTEM REQUIREMENTS FOR SPACE-CRAFT AND AIRCRAFT APPLICATION

Physical characteristic	Apollo	CV-340
Length of operation	Weeks	One to four hours
Number of operations	Once without cooldown or power down	Repeated on and off
Preflight calibration	Unlimited time in quiet environment	Hurried in nonquiet environment
Flight dynamics	1-D, straight line; relatively low vibration	3-D, maneuvering; high vibration
Thermal environment	Space radiator, stable, 70° F	Aircraft ambient air· 110° F; cabin temperature: 40°-88° F

TABLE 2.- DME STATION LOCATIONS FOR SIRU FLIGHT TEST PROGRAM

DME	Lat,	Long,	Alt,	Freq,	Range,	Bearing,	Acq-alt,
name	deg	deg	ft	Hz	n. mi.	deg	ft
Modesto	37.62	-120.95	90.	108.4	14.70	19.08	242.0
Porterville	35.91	-119.02	500.	109.2	135.12	121.04	15772.6
Crows Landing	37.40	-121.11	170.	110.2	.45	-158.43	-28.8
Woodside	37.39	-122.28	2270.	111.4	56.11	-101.01	651.3
San Francisco	37.61	-122.37	10.	111.6	61.68	-88.16	3490.8
Los Banos	36.71	-120.77	2107.	112.6	44.72	149.07	-199.4
Fresno	36.88	-119.80	100.	112.9	70.08	106.27	4379.0
San Jose	37.36	-121.92	48.	114.1	39.42	-104.09	1465.4
Linden	38.07	-121.00	260.	114.8	39.95	-2.98	1290.5
Sacramento	38.44	-121.55	5.	115.2	65.26	-28.78	3897.3
Bakersfield	35.48	-119.09	550.	115.4	151.18	129.21	19788.4
Friant	37.10	-119.59	2380.	115.6	74.81	93.77	2703.6
Stockton	37.83	-121.17	40.	116.0	25.37	-16.89	669.5
Gorman	34.80	-118.86	4500.	116.1	190.87	134.36	27849.8
Oakland	37.72	-122.22	30.	116.8	56.42	-80.36	2922.1
Avenal	35.64	-119.97	710.	117.1	119.13	142.26	11969.1
Salinas	36.66	-121.60	77.	117.8	50.83	-162.05	2346.0
Fellows	35.09	-119.85	5000.	117.5	151.77	145.98	15496.2
Moffett Field	37.43	-122.05	4.	117.6	45.39	-98.36	1956.9

TABLE 3a.- MOFFETT FIELD POSITION BENCHMARKS

Benchmark	Latitude	Longitude	Elevation
A	37° 24' 57"	122° 03' 17"	13.31 ft
B	37° 24' 56"	122° 03' 20"	13.80 ft
C	37° 24' 58"	122° 03' 25"	41.70 ft
D	37° 24' 58"	122° 03' 22"	13.45 ft

TABLE 3b.- CROWS LANDING POSITION BENCHMARKS

Benchmark	Latitude	Longitude	Elevation
AP	37° 24' 48"	121° 06' 26"	141.05 ft
L	37° 24' 47"	121° 06' 17"	138.53 ft
R	37° 24' 04"	121° 06' 36"	164.71 ft
MTR	37° 25' 06"	121° 06' 11"	129.59 ft
N	37° 25' 30"	121° 06' 17"	127.46 ft
P	37° 25' 22"	121° 06' 19"	129.43 ft
M	37° 24' 55"	121° 06' 16"	134.97 ft
TTR	37° 25' 03"	121° 06' 08"	129.87 ft

TABLE 4 - FLIGHT TEST SUMMARY

Flight data	Route	Navigation duration, sec	Flight duration, sec	Radar intersection, sec	Marks	Computers	Rate	Maximum position error, n mi	Detected failures Unscheduled (a)	Detected failures Scheduled (a)
5/20	Moffett/Crows	5170	2230	1655	6	B	L	5 93	G(CF)	--
5/30A	Moffett/Crows	2380	1285	0	9	B	H	12 82	G(ΓCBDEA), A(A)	--
5/30B	Crows/Los Banos/Moffett	5980	4390	2295	23	B	H,L	41 02	G(EFABCD)	--
6/16A	Moffett/Crows	3074	1608	0	11	A=B	L	23 08	G(FCEABD), A(AD)	--
6/16B	Crows/Moffett	2321	1670	0	8	A=B	L	3 55	G(ABFCED), A(E)	--
6/18	Moffett/Sac /Moffett	15303	14185	0	30	A	L	33 99	G(EACBF), A(D)	--
6/25	Crows/Crows	3405	2450	3282	10	A=B	H	12 94	G(CDBA)A(D)	--
7/14	Moffett/Stockton/Moffett	6005	4323	0	11	B	H	35 59	G(BDACF)	--
7/17A	Moffett/Stockton	5956	4000	1020	14	A=B	H	4 40	--	--
7/17B	Stockton/Moffett	4155	3925	3085	12	A	H	2 10	--	G(CA)
7/24A	Moffett/Crows	2400	1295	354	6	A=B	H	2 79	--	--
7/24B	Crows/Stockton/Crows	2555	2163	2447	5	A=B	H	1 23	--	A(CE)
7/24C	Crows/Stockton/Crows	2587	2241	2417	6	A≠B	H	4 04	--	A(CA),A(C)
7/29A	Moffett/Stockton/Crows	3971	3439	3174	9	A≠B	H	14 06	G(FBE)	G(CD)
7/29B	Crows/Moffett	2426	1392	1469	8	A≠B	H	7 13	G(E)	G(C)[b]
8/22A	Moffett/Crows	2630	1181	553	7	A=B	H	2 93	--	--
8/22B	Crows/Crows	2110	646	2038	6	A≠B	H	36[c]	G(E),A(E)	G(AB)
8/22C	Crows/Bakersfield	4082	3422	2566	5	A≠B	VH,L	11 16	G(EF),A(E)	G(AB)
8/22D	Bakersfield/Moffett	--	4320	--	6	A	H	--	G(E),A(A)	--
8/29A	Moffett/Crows	2620	1202	850	6	A=B	H	7 65	--	G(BA)
8/29B	Crows/Crows/Crows	2501	553	2316	5	A≠B	H	3[c]	--	G(BA)
9/05A	Moffett/Crows	2200	1231	0	6	A≠B	H	2 50	G(A)	G(FE)
9/05B	Crows/Crows	3067	1990	1774	2	A≠B	H	3 72	G(F)	A(BA)
9/05C	Crows/Crows	2331	1498	1404	2	A≠B	H	1 86	G(F)	G(E)
9/05D	Crows/Moffett	2053	1369	1497	6	A≠B	H	3 00	G(EF)	A(DC)
9/10A	Moffett/Crows	3750	2467	1758	4	A=B	H	3 97	--	--
9/10B	Crows/Crows(A cont'd)	2527	1794	2365	5	A=B	H	10 19	G(B)	--
9/10C	Crows/Crows(B cont'd)	2652	1952	2459	6	A=B	H	19 36	--	--
9/10D	Crows/Moffett(C cont'd)	1859	1187	0	4	A=B	H	20 14	--	--
9/18A	Moffett/Crows	3860	2890	2173	4	A=B	H	6 22	--	--
9/18B	Crows/Crows(A cont'd)	3045	2415	3045	5	A=B	H	7 62	--	--
9/18C	Crows/Crows(B cont'd)	2690	2467	2690	5	A=B	H	6 88	--	--
9/18D	Crows/Moffett	1600	1091	804	5	A=B	H	2 91	G(ΓE),A(E)	--
9/24	Moffett/Baker/Moffett	15819	14293	8928	21	A=B	L	7 8	--	--

[a] It should be noted that the large number of failures through most of the early flights were caused by low TSE limits and procedural problems During later flights, some unscheduled failures were detected and resolution of the cause not determined G(ΓCBDEA) indicates gyros Γ, C, B, D, E, and A failed in that order A(A) indicates accelerometer A failed, etc

[b] Small failure undetected (below detection limit)

[c] Position and velocity were reset during nagivation

TABLE 5.- POSITION REFERENCE DATA

				Residual radar-photograph	Residual DME-photograph	Residual radar-DME
Test flt.no.	Mo.	Day	Mark no.	R-M	D-M	R-D
Crows Landing						
5	6	25	1	423.825	975.927	622.658
5	6	25	2	320.361	550.117	604.741
5	6	25	3	520.697	497.351	711.678
5	6	25	4	361.289	1093.61	733.728
5	6	25	5	128.034	634.744	744.033
5	6	25	7	54.7524	709.752	677.333
5	6	25	8	226.632	656.635	776.282
7	7	17	15	106.305	684.344	612.039
7	7	17	17	122.033	1246.06	1278.73
9	7	29	4	280.788	1012	775.809
9	7	29	8	246.434	853.89	730.937
15	9	24	2	613.069	940.726	735.407
15	9	24	6	574.165	610.515	1053.21
Stockton Metropolitan Airport						
7	7	17	16	439.415	327.143	760.912
9	7	29	5	847.235	55.236	901.919
5	9	24	5	654.978	994.773	779.891
Modesto City-County Airport						
7	7	17	13	499.61	562.744	284.105
9	7	29	6	439.118	507.555	372.682
15	9	24	7	597.118	734.54	879.79
15	9	24	15	1215.7	1576.1	362.534
Castle Air Force Base						
7	7	17	14	368.718	209.895	567.57
9	7	29	7	776.962	448.343	599.089
15	9	24	8	612.854	567.516	513.086
15	9	24	14	1004.99	2083.71	1138.42

TABLE 6.- AVERAGE RESIDUALS BETWEEN POSITION REFERENCES AS MEASURED DURING THE FLIGHT

	Radar-photograph	DME-photograph	Radar-DME
	R-M	D-M	R-D
Average residual (ft)	476.46	722.42	717.36

TABLE 7.- SIRU NAVIGATION PERFORMANCE

Flight	Nav. duration (sec)	Maximum position error (n. mi.)	Average position error (n. mi.)	Average slope (n. mi./hr)	Straight line (A+Bt) A (n. mi.)	Straight line (A+Bt) B (n. mi./hr)
5/20	5171	5.93	3.19	4.61	-0.02	4.65
5/30A	2380	12.82	4.47	9.80	-12.31	34.56
5/30B	5980	41.02	17.76	20.77	-6.56	26.59
6/16A	3074	23.08	12.30	29.61	-2.95	34.73
6/16B	2321	3.55	.81	2.70	-.15	3.06
6/18	15303	33.99	12.60	5.49	4.39	3.90
6/25	3405	12.94	7.63	15.58	1.07	13.86
7/14	6005	35.59	15.65	19.79	-3.77	23.27
7/17A	4888	4.40	2.52	2.94	.53	2.44
7/17B	4155	2.10	1.02	1.22	.45	.77
7/24A	2400	2.79	1.21	3.64	-.01	3.67
7/24B	2555+	1.23+	.36	.97	.04	.88
7/24C	2587+	4.04+	1.48	4.35	-.62	5.63
7/29A	4900	14.06	8.53	10.50	-1.88	12.50
7/29B	2426	7.13	1.76	6.14	-1.32	9.09
8/22A	2630	2.93	1.38	3.94	-.24	4.44
8/22B	2110	.36	.31	.84	.27	.14
8/22C	4082	11.16	2.22	4.81	-.92	6.20
8/22D	?	?	?	?	?	?
8/29A	2620	7.65	2.10	6.65	-1.48	9.71
8/29B[a]	2501[a]	.3[a]	.15	.36	.12	.11
9/05A	2200	2.50	1.06	3.56	-.10	3.81
9/05B	3067	3.72	1.05	2.73	-.53	3.66
9/05C	2013	1.86	.60	2.25	-.16	2.68
9/05D	2053	3.00	.73	4.13	-.57	6.22
9/10A	3733	3.97	1.19	2.36	-.20	2.65
9/10B	6685	10.19	4.40	5.32	-1.92	6.86
9/10C	9620	19.36	6.70	5.37	-1.66	6.29
9/10D	12130	20.14	8.16	5.15	-.85	5.53
9/18A	3860	6.22	2.22	4.52	-.96	5.85
9/18B	7345	7.62	4.00	3.85	.34	3.61
9/18C	10700	6.88	4.33	2.59	2.03	1.57
9/18D	1600	2.91	.95	4.79	-.54	6.61
9/24	15819	7.80	3.92	1.59	1.68	1.02

[a] Position reset 1300 sec.

TABLE 8.- SCHEDULED FAILURE DETECTION/ISOLATION TEST RESULTS

Flight segment	Flight cond (a)	Failed sensor	Failure level	Time in sec	Time detected sec	Undetected duration, sec	Error, δAng, δV
7/17B	SL	C gyro	5°/hr	807	866	59	0.082°
	M	A gyro	10°/hr	1267	1300	33	.092°
7/24B	SL	C accel.	2 cm/sec^2	1628	1648	20	.40 m/sec
	SL	E accel	3 cm/sec^2	2102	2123	21	.63 m/sec
7/24C	SL	C gyro	5°/hr	1392	1443	51	.071°
	SL	C accel	2 cm/sec^2	1599	1619	20	.40 m/sec
	M,SL	A gyro	3°/hr	1808	1925	117	.097°
7/29A	M,SL	C gyro	3.5°/hr	1308	1394	86	.084°
	SL	D gyro	3°/hr	1433	1568	135	.112°
7/29B	M	C gyro(A)	0 5°/hr	607	---	---	---
	M	C gyro(B)	1.5°/hr	659	902	243	.101°
8/22B	SL	A gyro	6167°/hr	495	495	≤1	(.086°, 1.7°)
	SL	B gyro	6167°/hr	516	516	≤1	(.086°, 1.7°)
	M	A gyro	6167°/hr	1664	1664	≤1	(.086°, 1.7°)
	M	B gyro	6167°/hr	1682	1682	≤1	(.086°, 1.7°)
8/29B	SL,M	A gyro	6°/hr	675	775	100	.167°
	SL	B gyro	24°/hr	680	691	11	.073°
9/05A	SL	E gyro	6°/hr	1205	1254	49	.082°
	SL	F gyro	24°/hr	1211	1222	11	.073°
9/05B	M	A accel.	1 cm/sec^2	765	863	98	.98 m/sec
	M	B accel.	4 cm/sec^2	773	863	90	3.6 m/sec
9/05C	M	E gyro	6°/hr	595	645	50	.083°
9/05D	M	C accel.	1 cm/sec^2	488	572	84	.84 m/sec
	M	D accel.	4 cm/sec^2	500	572	72	2.88 m/sec

[a] SL straight-and-level flight; M. maneuvering flight.

TABLE 9.- SUMMARY OF UNSCHEDULED SENSING AXES FAILURES

Flight and failure identification			Estimated reasons for failure						
Test no.	Attitude sensing axes failures (denoted by -R-)	Acceleration sensing axes failures (denoted by -A-)	Insufficient warm-up time	Temperature gradients due to blowing cold air	Ambient temperatures in excess of 88°F	Defective compensation due to human error	Defective compensation due to calibration procedure	Defective compensation due to poor error model	Undetermined
5/20	D F		-R-	-R-		-R-	-R-		
5/20A	ABCDEF	A				-R-A-	-R-A-		
5/30B	A C EF				-R-	-R-	-R-		
6/16A	ABC F	AD				-R-	-R-A-	-A-	
6/16B	ABCDEF	E			-A-	-R-	-R-A-		
6/18	ABCDEF	D				-R-	-R-A-	-A-	
6/25	A CDE	D				-R-A-	-R-A-	-R-	
7/14	ABCD F					-R-	-R-	-R-	
7/17A								Statistical failure detection and isolation removed from navigation program	
7/17B									
7/24A									
7/24B									
7/24C									
7/29A	B EF			-R-					-R-
7/29B	E			-R-					
8/22A									
8/22B	E	E			-R-A-				
8/22C	F	E			-R-A-				
8/22D	E	E			-R-A-				
8/29									
9/05A	A								
9/05B	F				-R-				
9/05C	F				-R-				
9/05D	EF				-R-				
9/10A									
9/10B									
9/10C									
9/10D									
9/18A									
9/18B									
9/18C									
9/18D	EF	E			-R-A-				
9/24									

TABLE 10.- TOTAL PARITY ERROR RESIDUAL STANDARD DEVIATIONS FROM FLIGHT 14 (9/18B)

Flight mode	Gyro parity residuals (standard deviation (°/hr))					
	σ_A	σ_B	σ_C	σ_D	σ_E	σ_F
3°/sec (CCW)	1.04	2.28	0.96	0.70	2.56	2.88
3°/sec (CW)	1.24	1.99	.88	.76	1.96	2.10
Level flight	0.36	0.69	1.01	0.68	0.33	0.72

TABLE 11.- SIRU COMPUTER PREFLIGHT SYSTEM TESTABILITY

Subsystem	Comments
Testable:	
Watchword checker	Software can send various "faulty" watchwords to the receiver to test a match for each bit, one bit at a time.
Self-diagnostics	Software can continuously execute self-diagnostics, with random interrupts from the auxiliary test equipment which performs an action similar to the effects of the fault being diagnosed.
Time-out generator (for computer)	Software can send opinions to the T-box at various time delays, testing the "window" of the time-out. This can also be used to gauge the accuracy of the time-out delay elements, but only if the clock generator of the computer is determined to be within frequency bounds when tested by the auxiliary test equipment.
Not testable:	
Parity checker	It is possible to exercise the T-box parity tree by changing all the input bits one at a time, but there is no facility to change the parity bit sent to the receiver.
Bit count	It is not possible to cause the T-box counter to act in a faulty way, nor can the transmitted clock be tested by circuitry other than the (perhaps faulty) receiver counter.
Clock active	There is no way to incrementally vary the clock time delays or inject various unusual pulses on the clock lines.

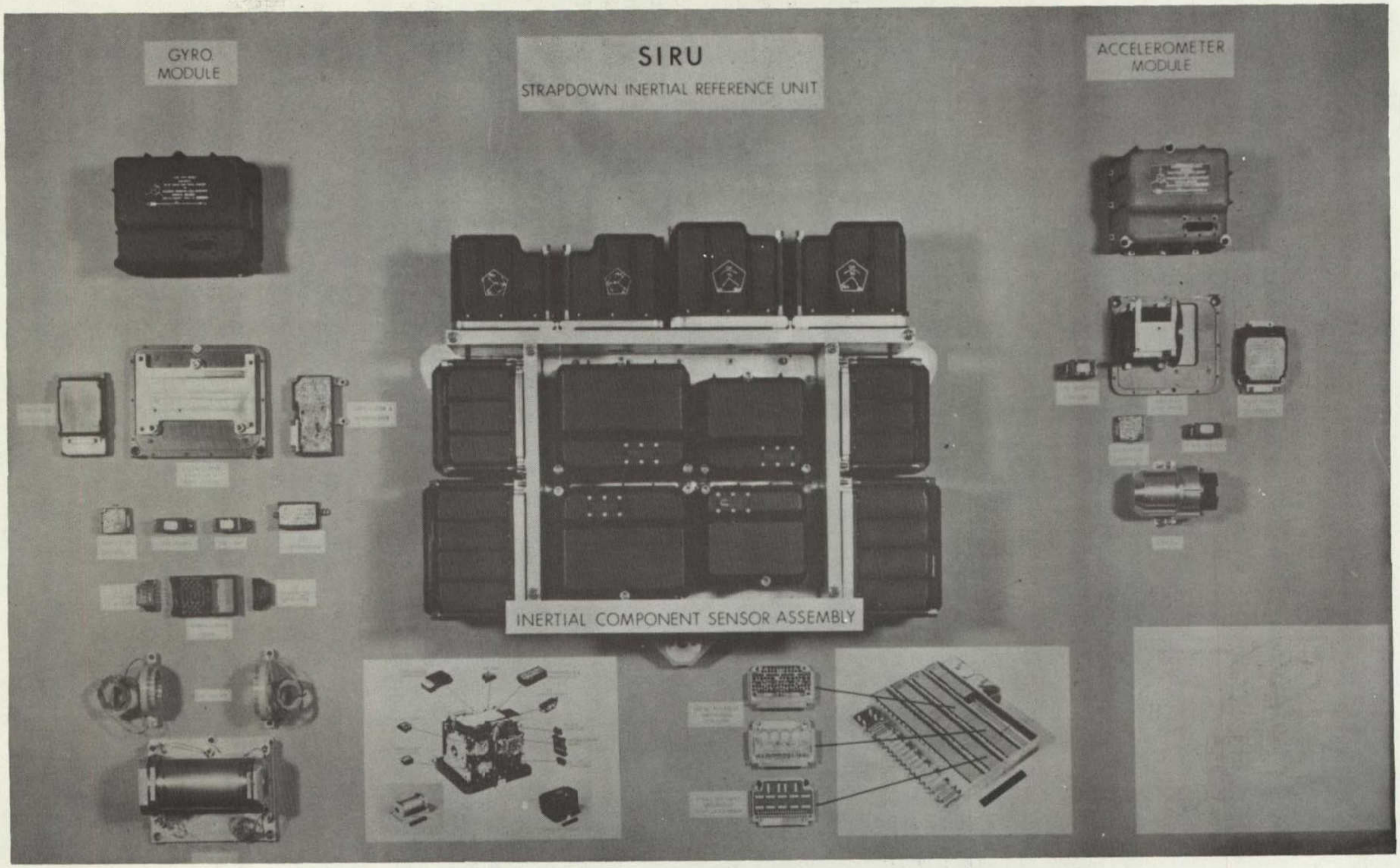

GYRO MODULE
SIRU
STRAPDOWN INERTIAL REFERENCE UNIT
ACCELEROMETER MODULE
INERTIAL COMPONENT SENSOR ASSEMBLY

Figure 1.- SIRU inertial component sensor assembly.

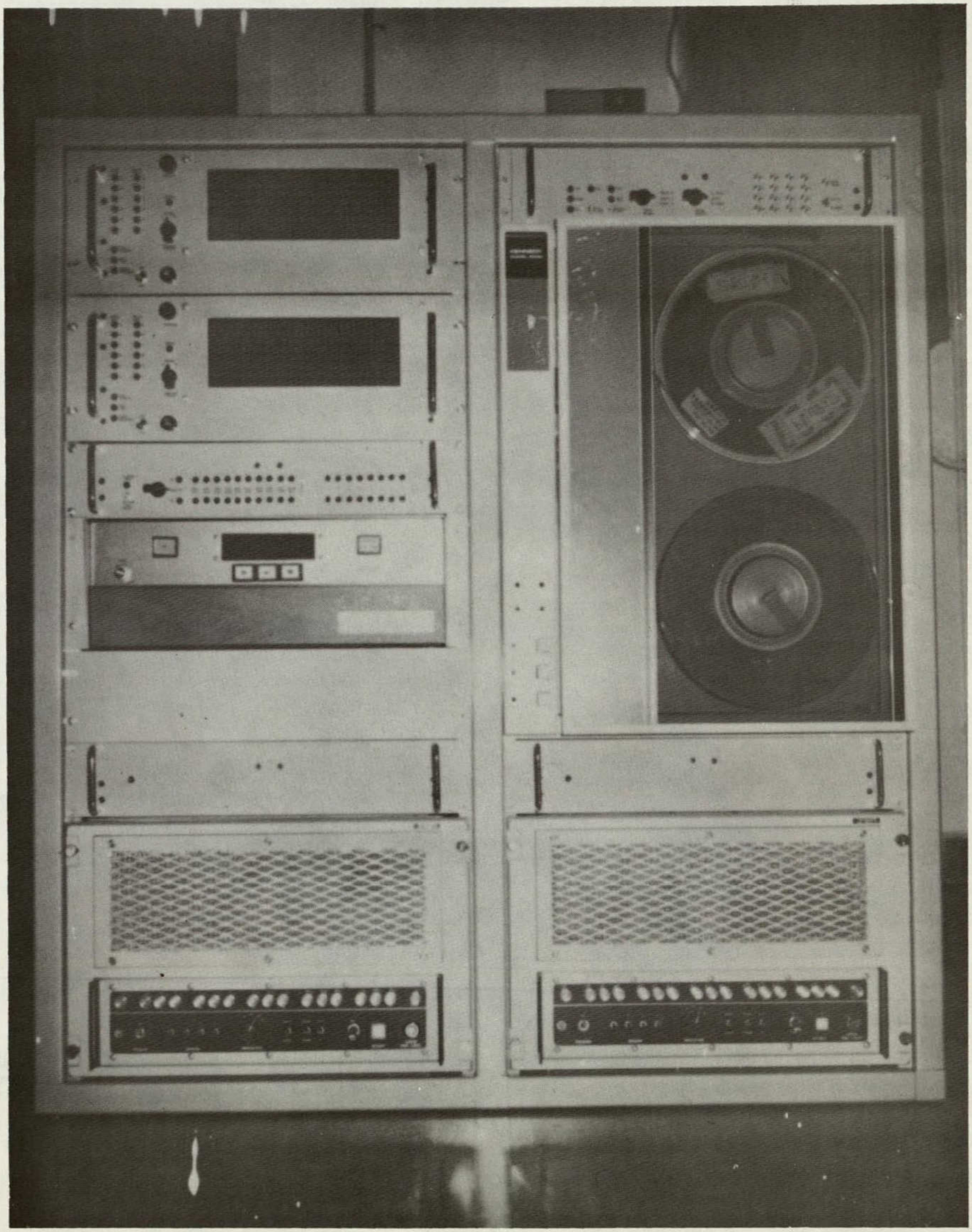
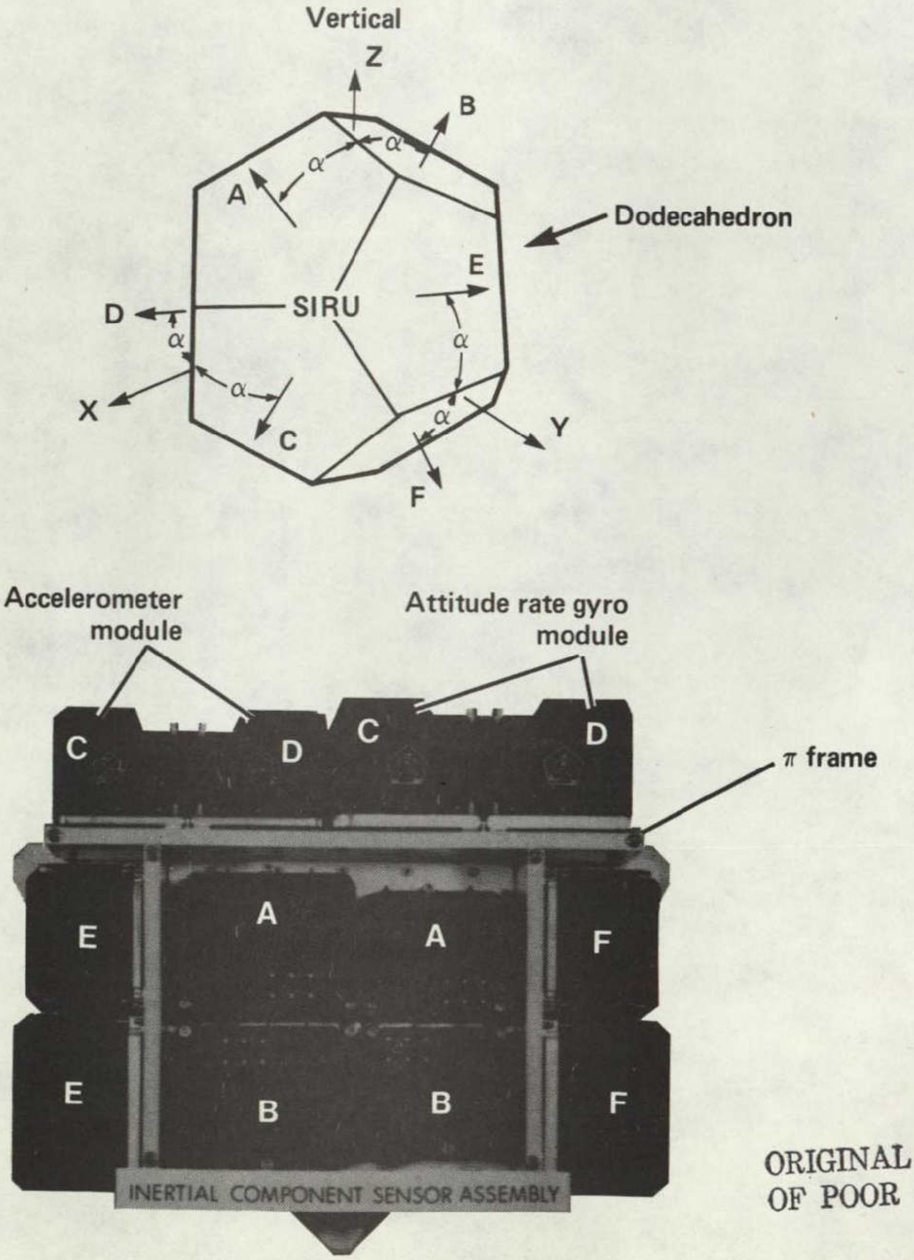


Figure 2.- SIRU computer console.



SIRU
strapdown inertial reference unit

Figure 3.- Dodecahedron SIRU instrument configuration.

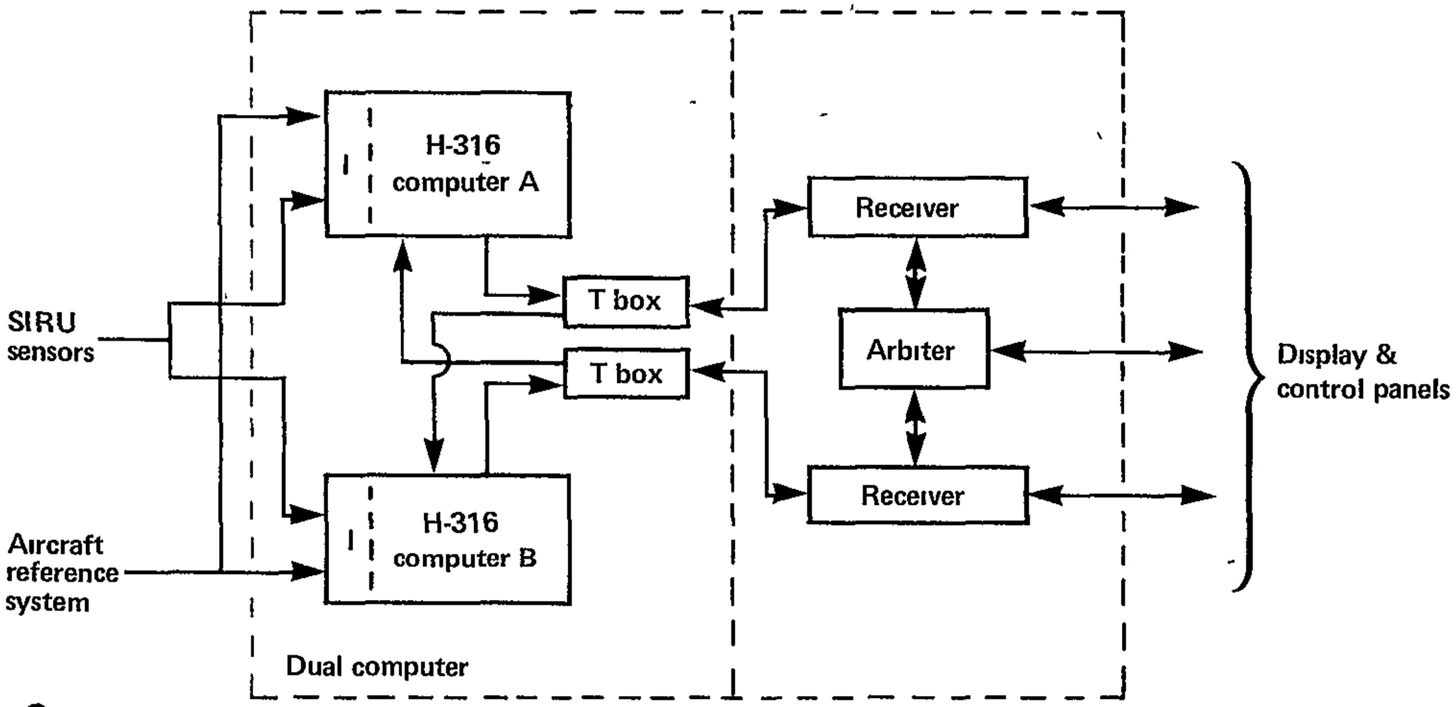


Figure 5.- Dual computer system configuration.

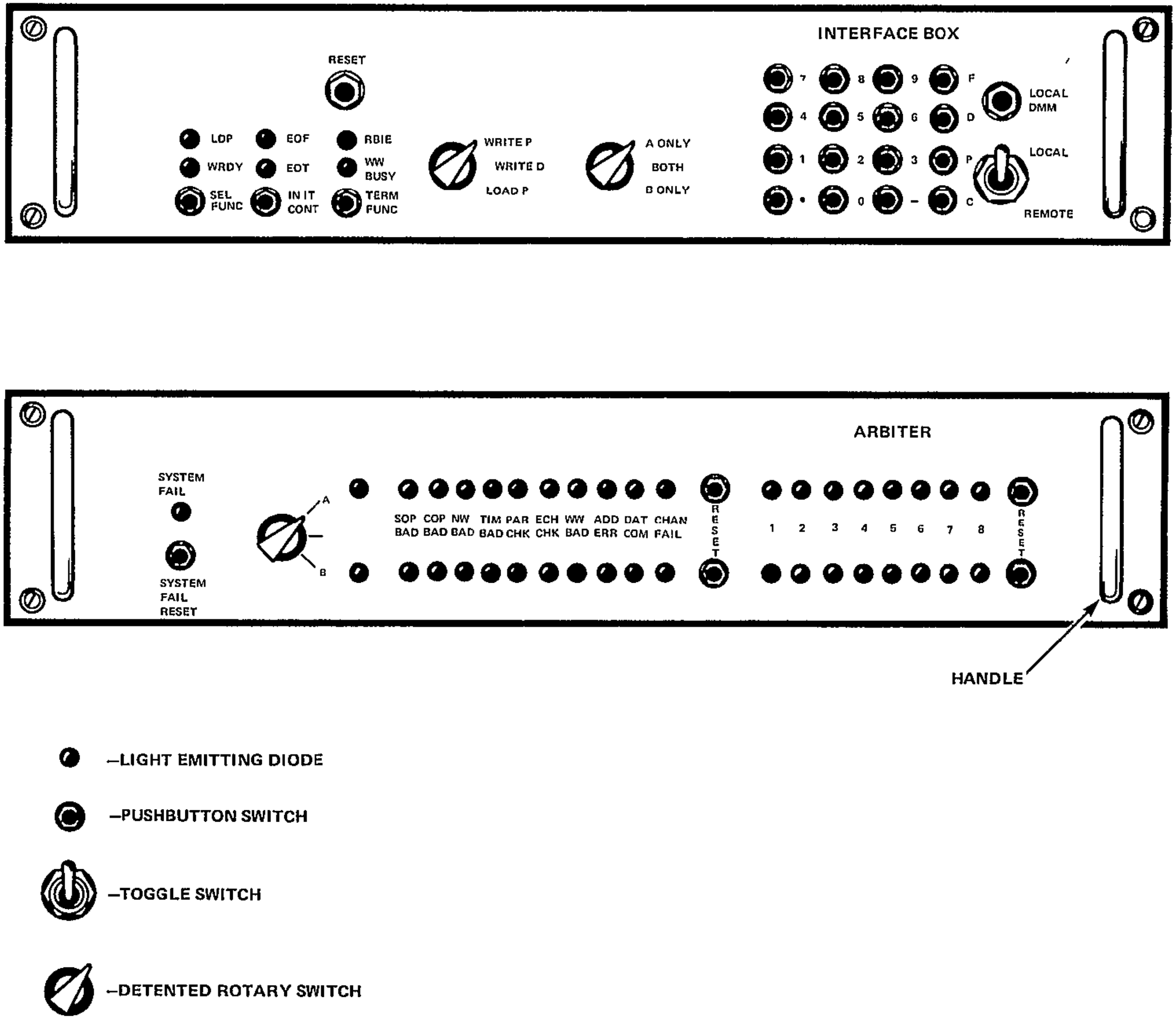
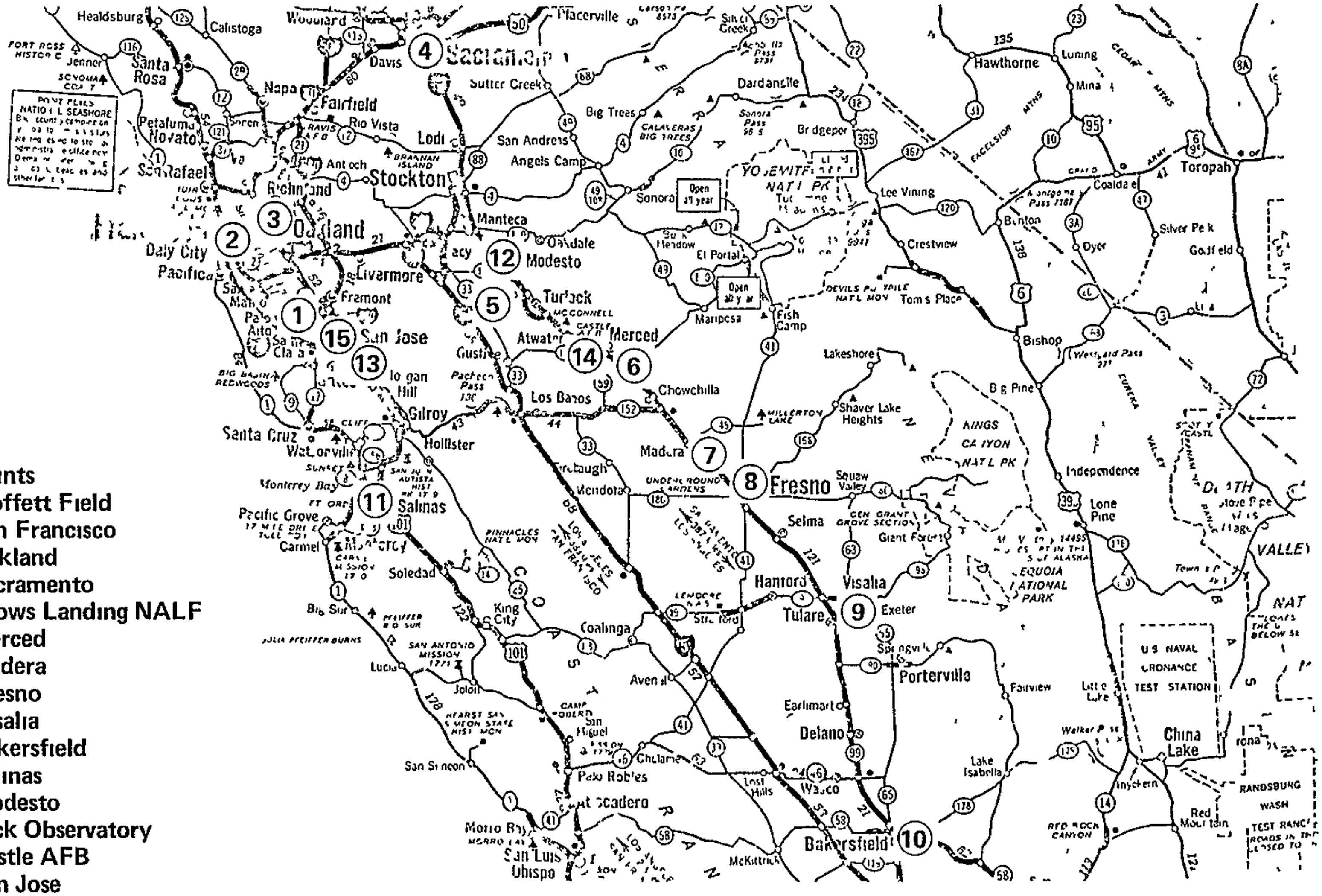
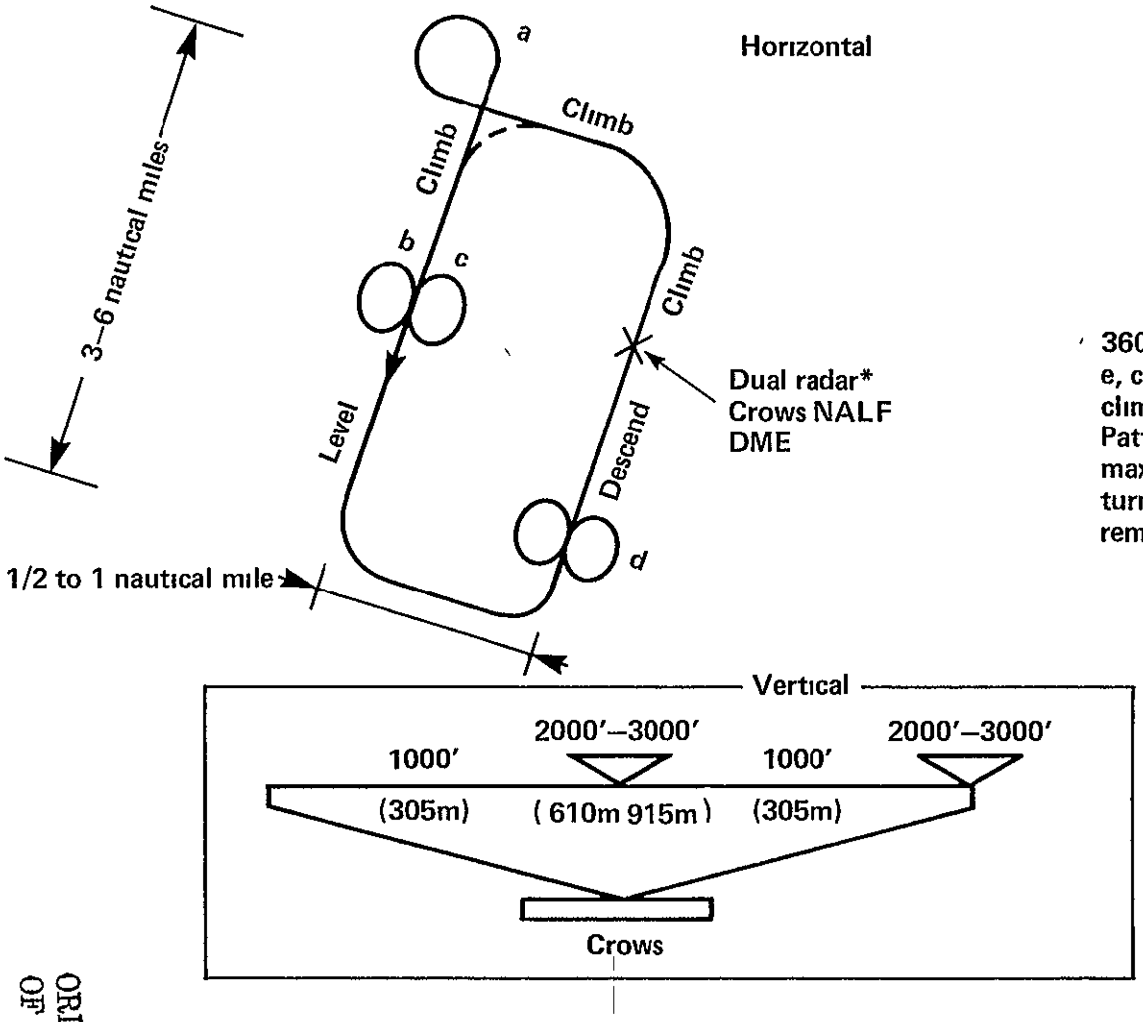


Figure 6.- Control panels for dual computers with arbiter.



Waypoints
1—Moffett Field
2—San Francisco
3—Oakland
4—Sacramento
5—Crows Landing NALF
6—Merced
7—Madera
8—Fresno
9—Visalia
10—Bakersfield
11—Salinas
12—Modesto
13—Lick Observatory
14—Castle AFB
15—San Jose

Figure 7.- Visual landmarks for the CV-340 flight test.



*360° turn at a, b, c, d, e, could either be level, climbing, or descending Pattern altitude 1000′ max-bank angle for turns was 45° Pattern remained in radar view

Figure 8.- Terminal area pattern near Crows NALF.

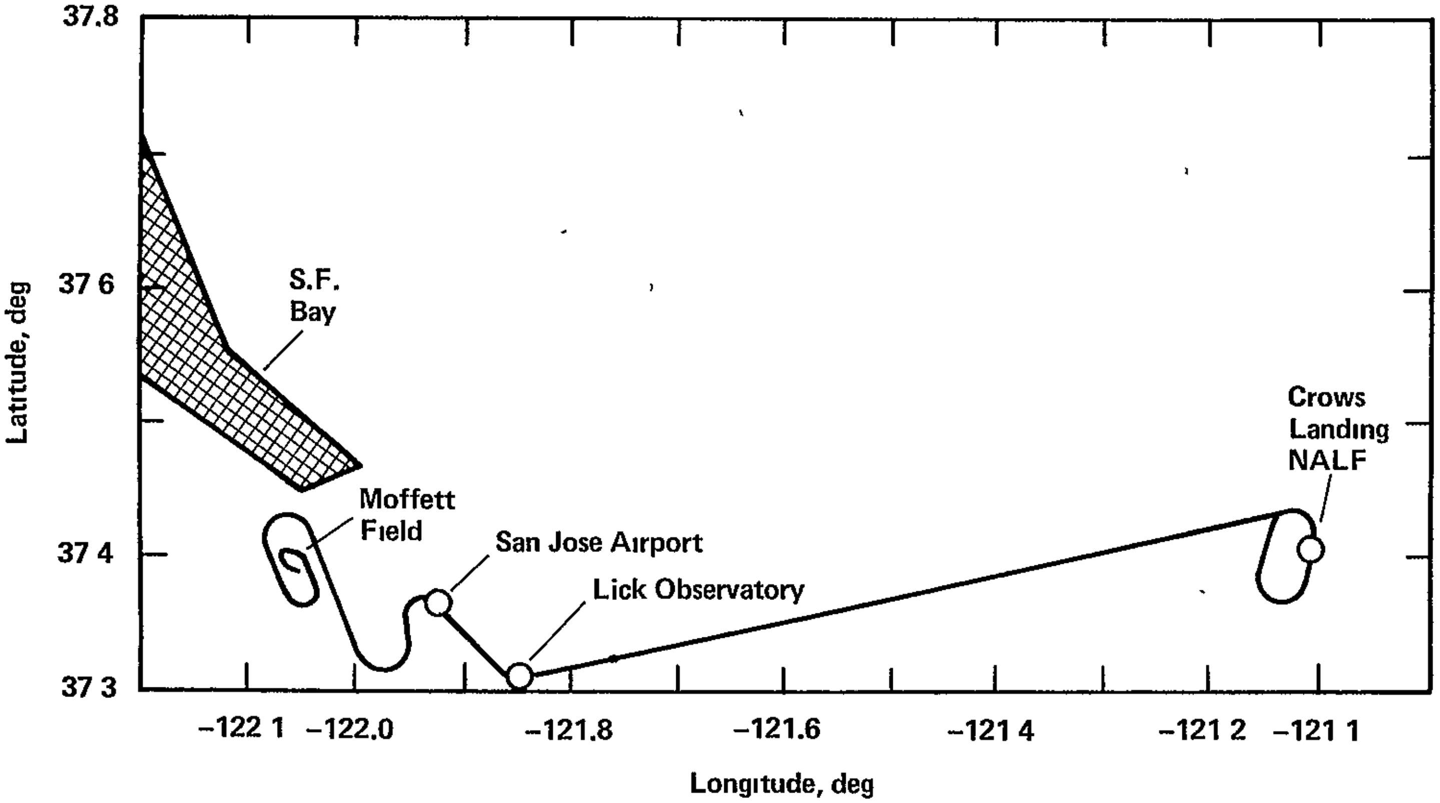


Figure 9.- Flight sequence — Moffett Field to/from Crows Landing NALF.

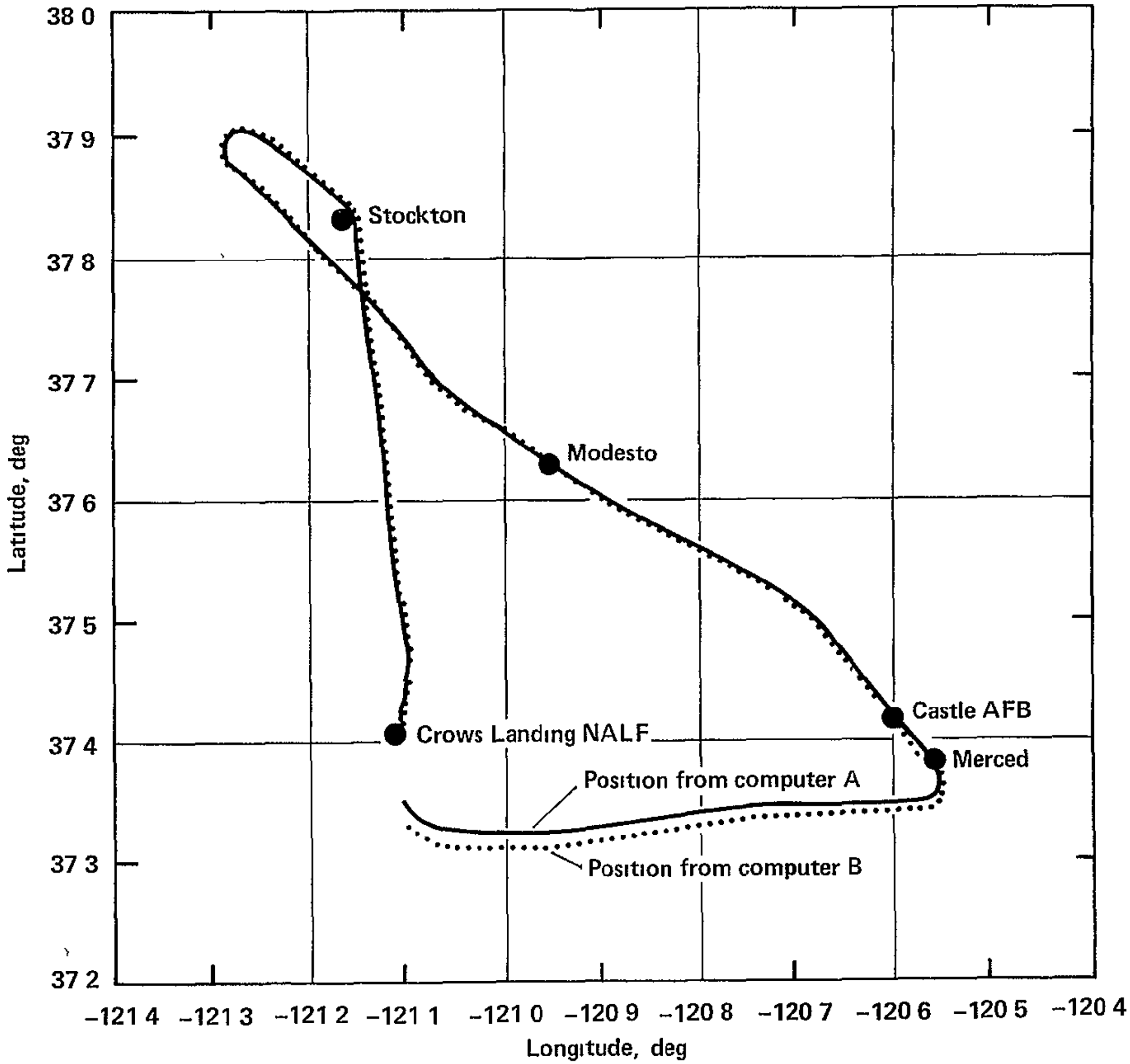


Figure 10.- Typical flight sequence originating at Crows Landing NALF

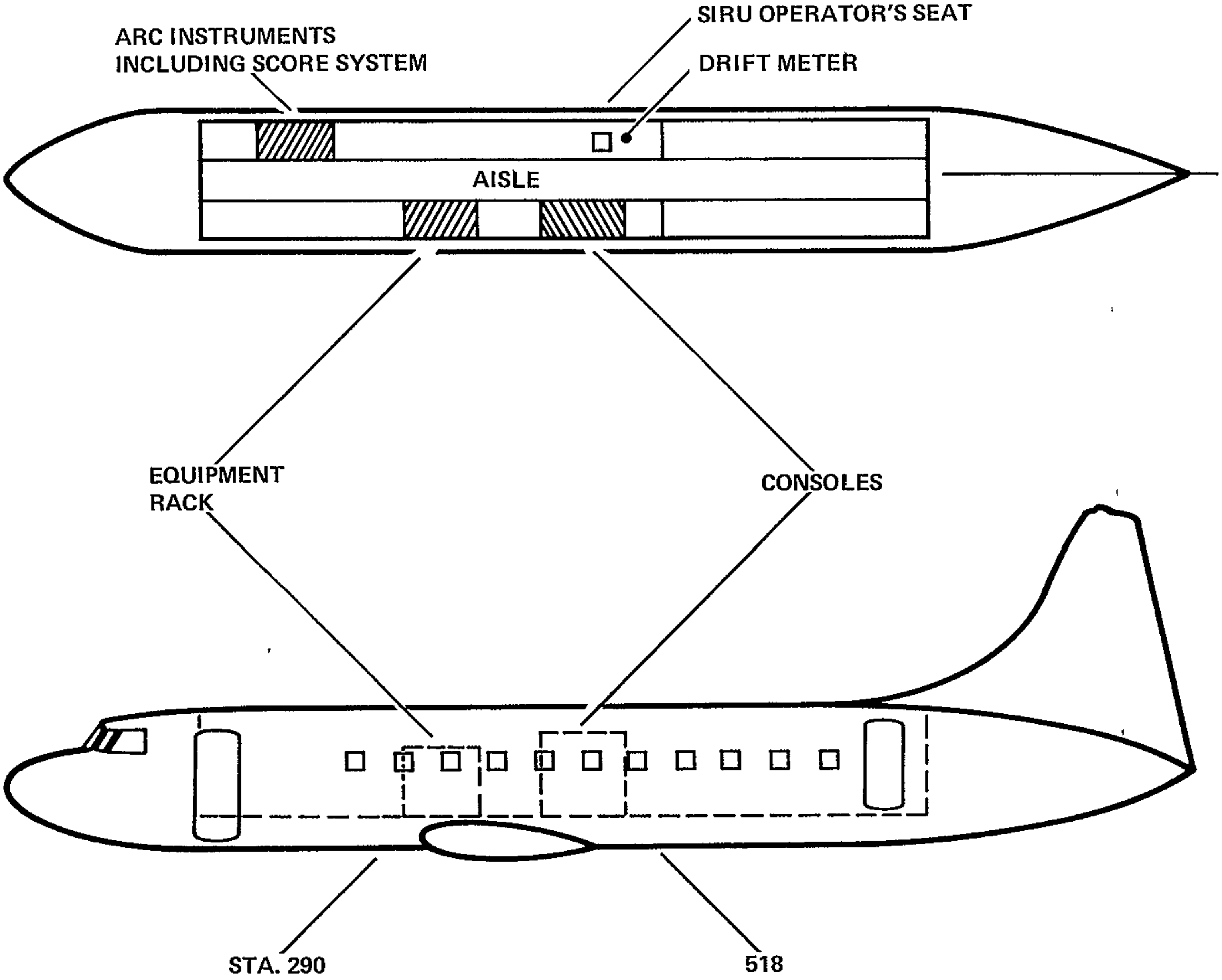


Figure 11.- Location of external reference equipment in the CV-340 aircraft.

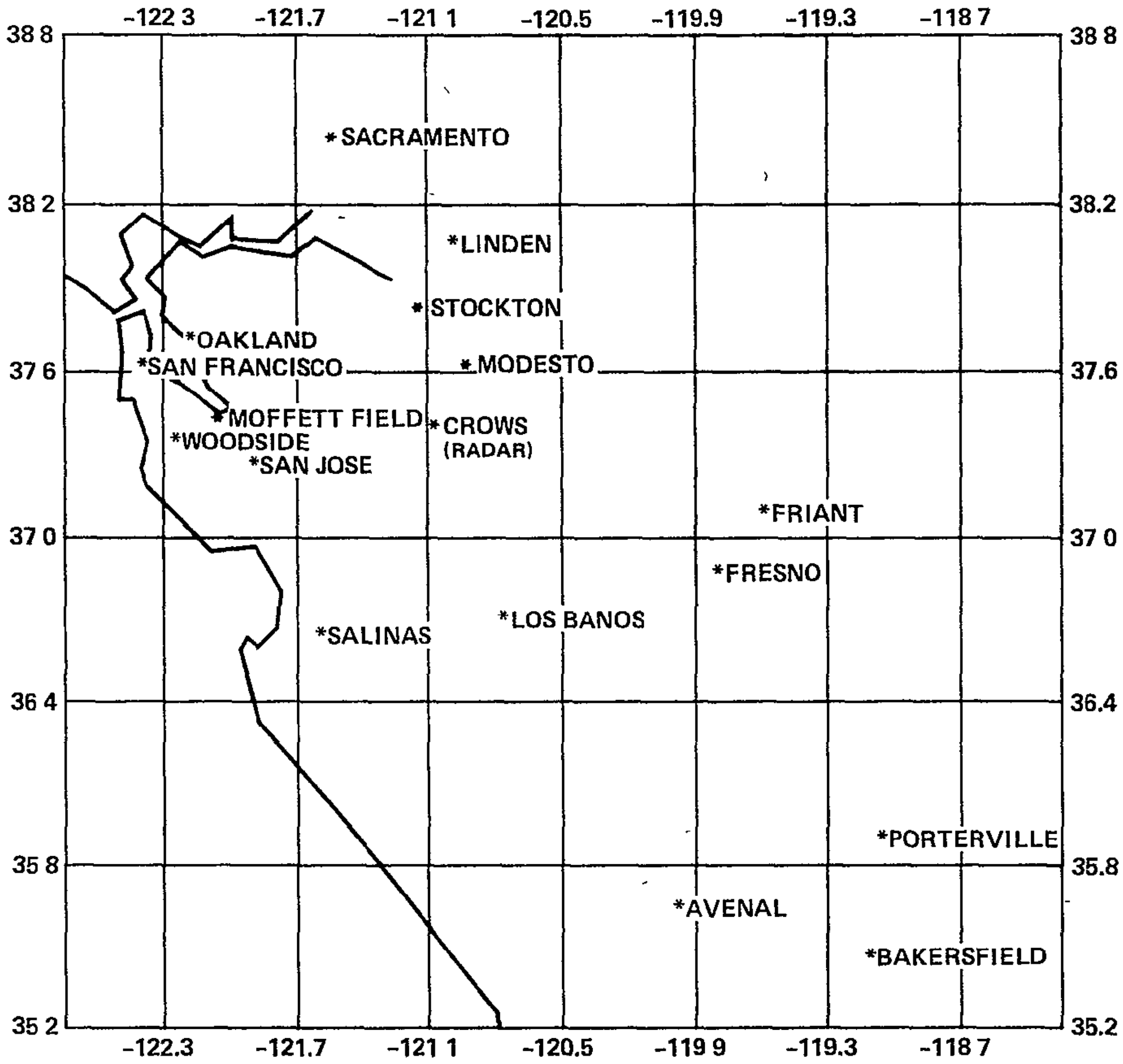


Fıgure 12.- Posıtıon reference locatıons.

ORIGINAL PAGE IS
OF POOR QUALITY

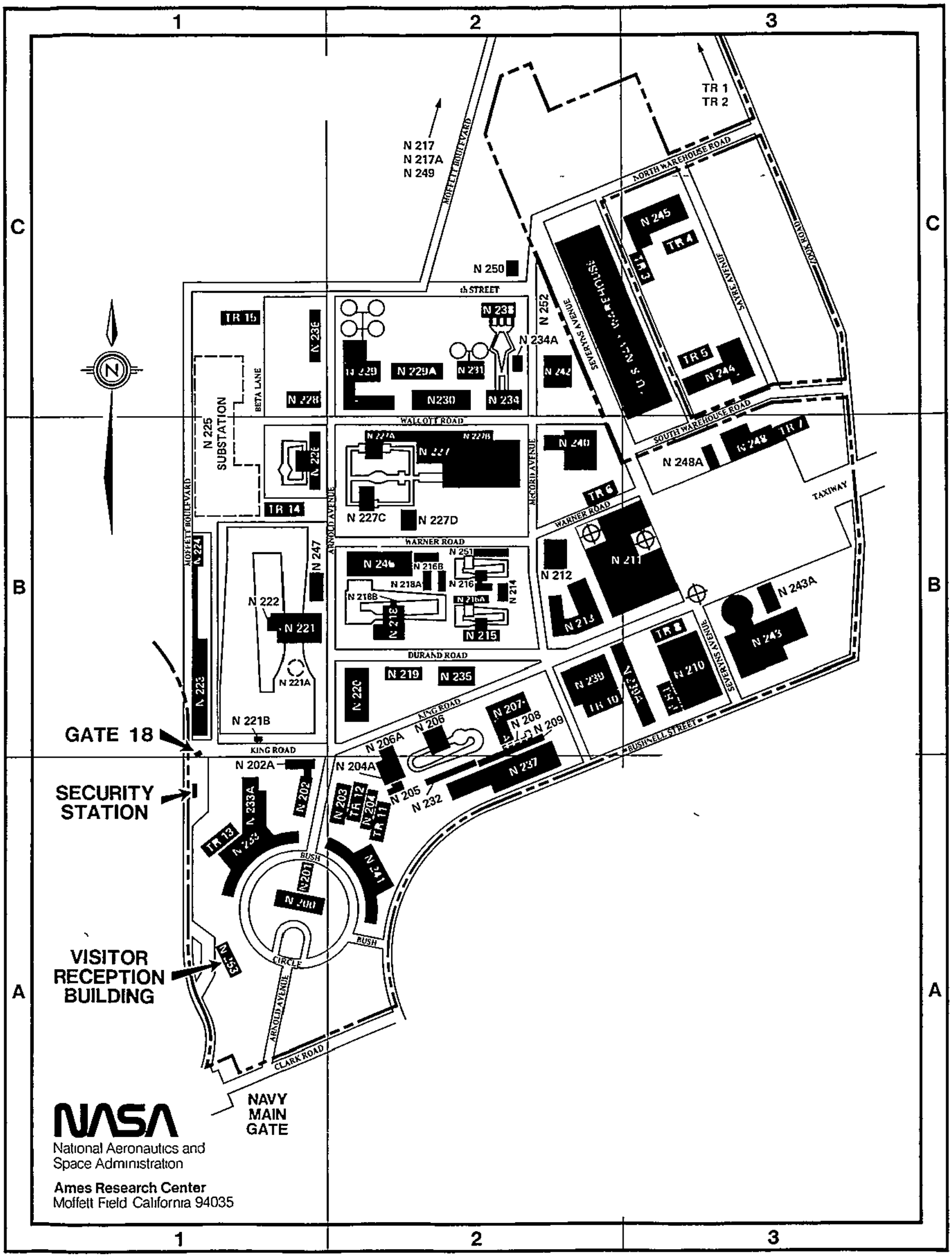


Figure 13.- Benchmark locations at Moffett Field.

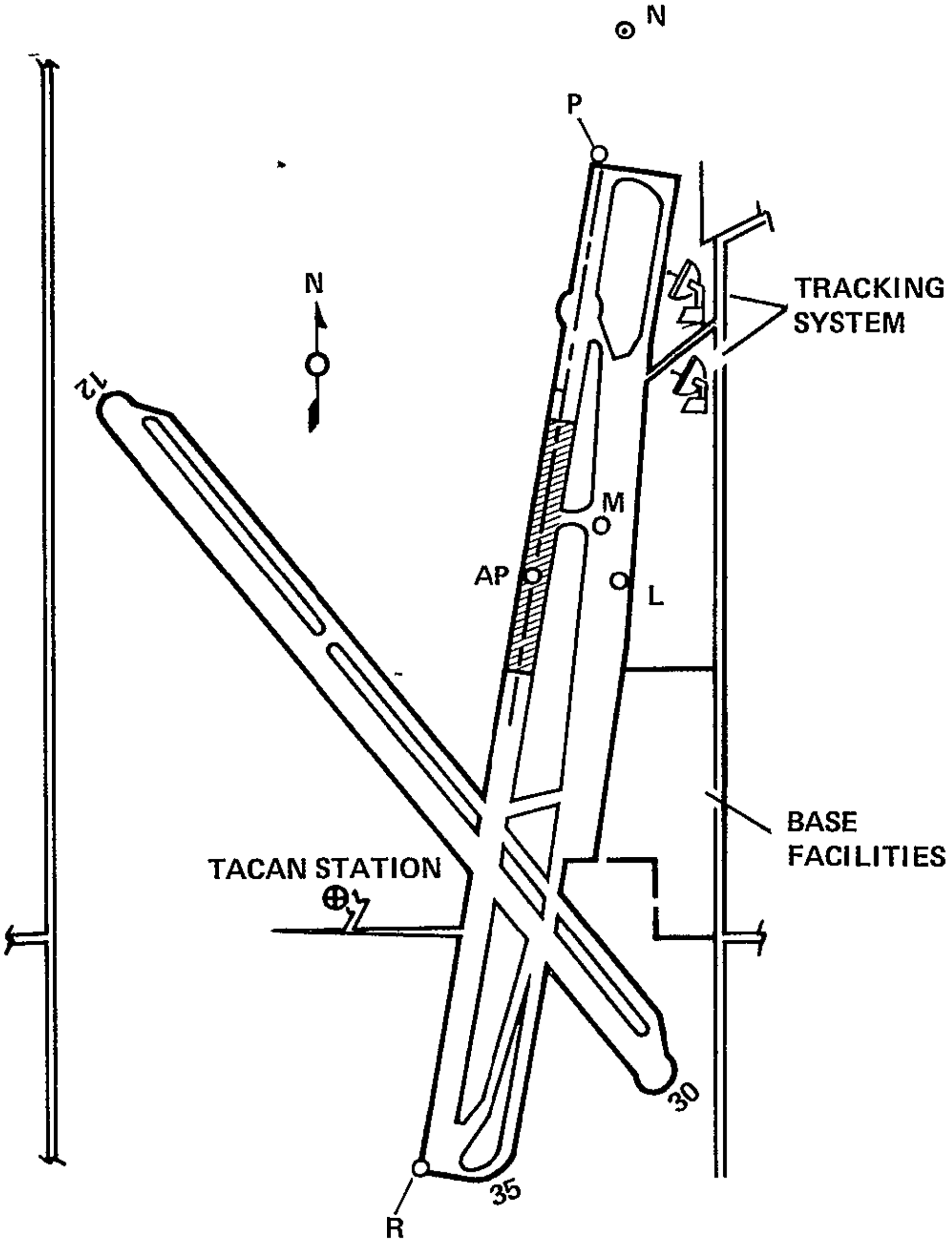
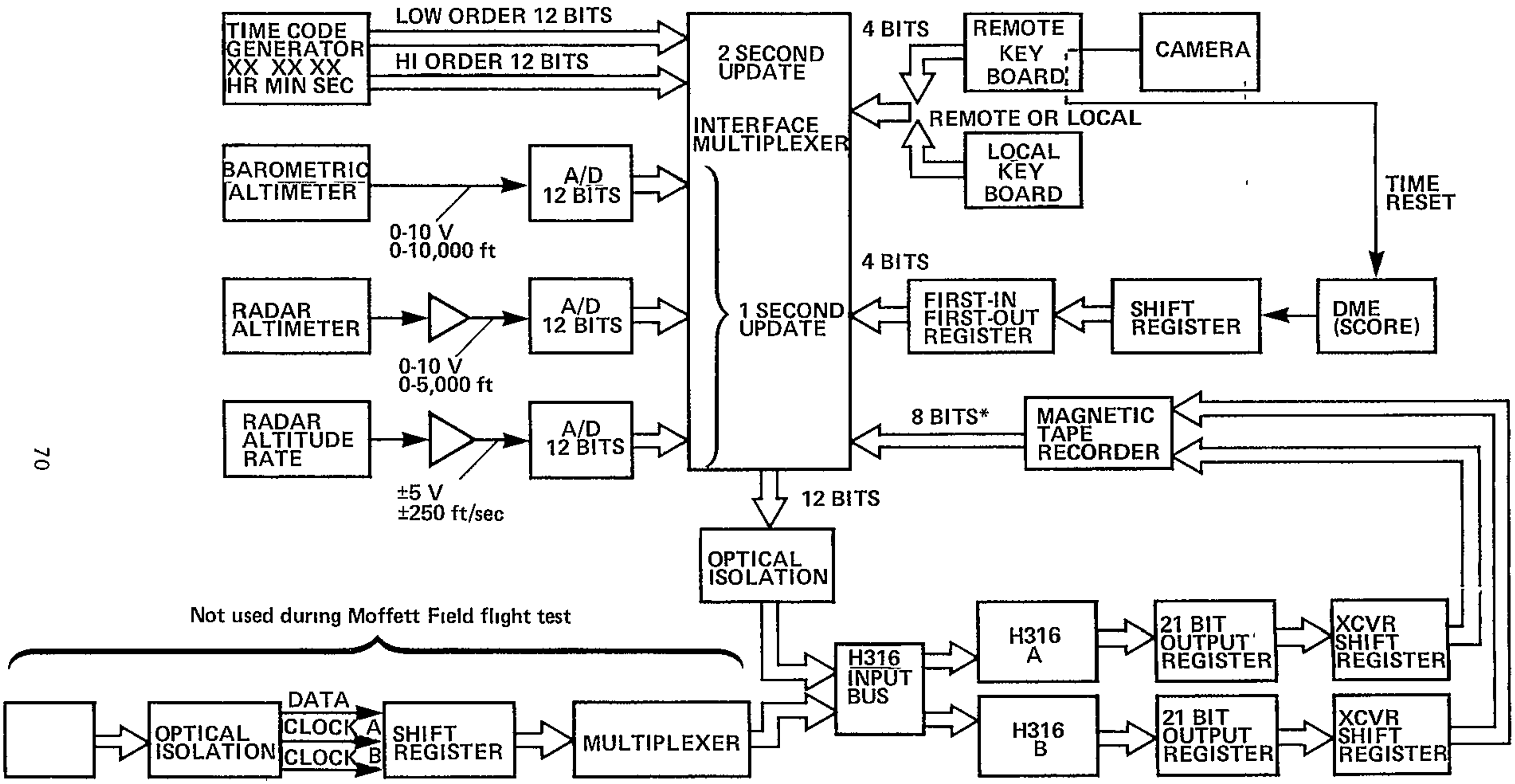
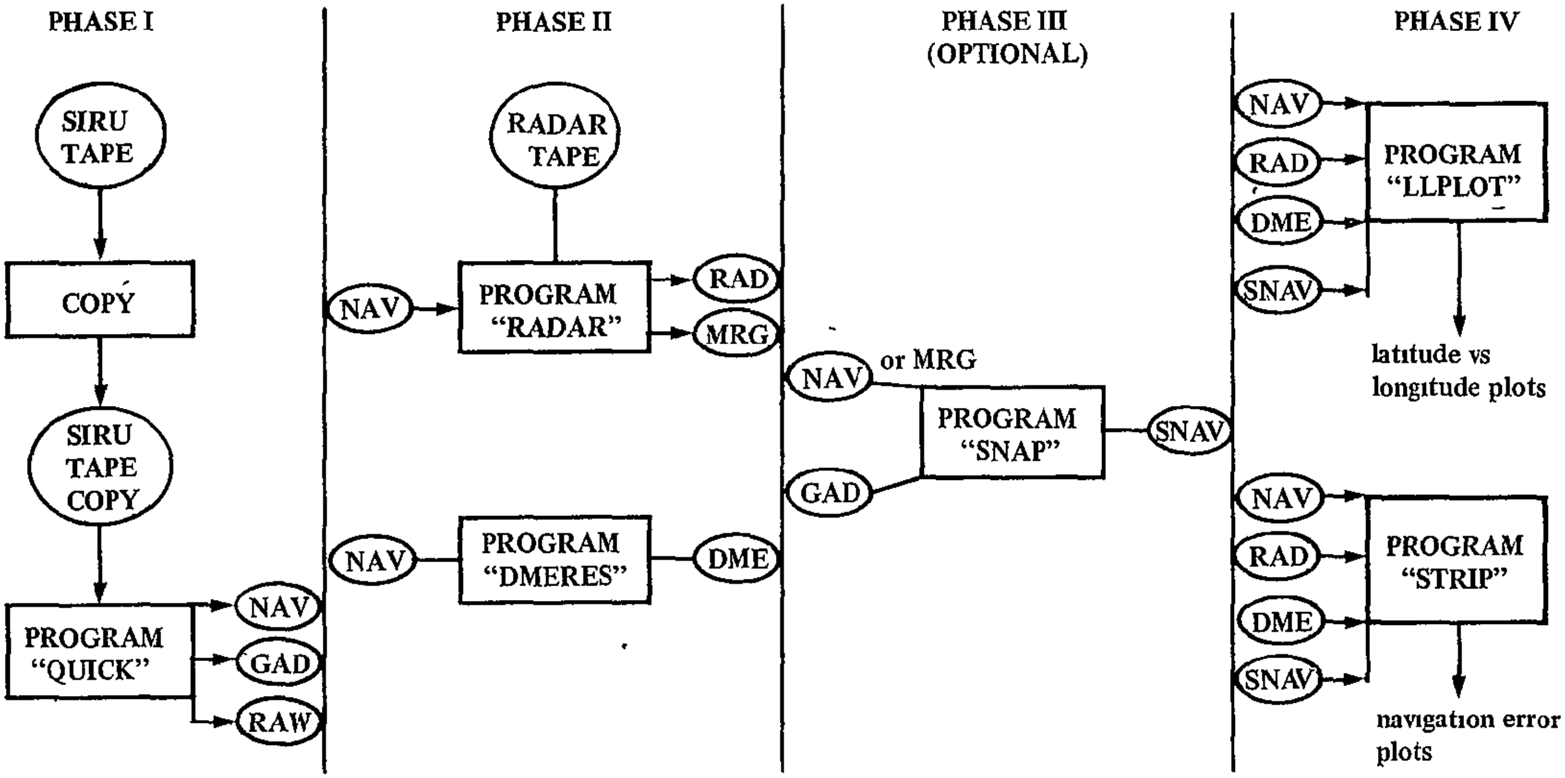


Figure 14.- Benchmark locations at Crows Landing.



*For loading programs only — requires dedicated computer

Figure 15.- SIRU peripherals and multiplexer



Legend

NAV navigation data file
RAD=gyro and accelerometer data file
RAW=raw data file
RAD=radar data file
MRG=merged radar and navigation data file
DME=trajectory data file derived from Distance Measuring Equipment
SNAV=smoothed navigation data file

Figure 16.- SIRU flight test data processing schematic.

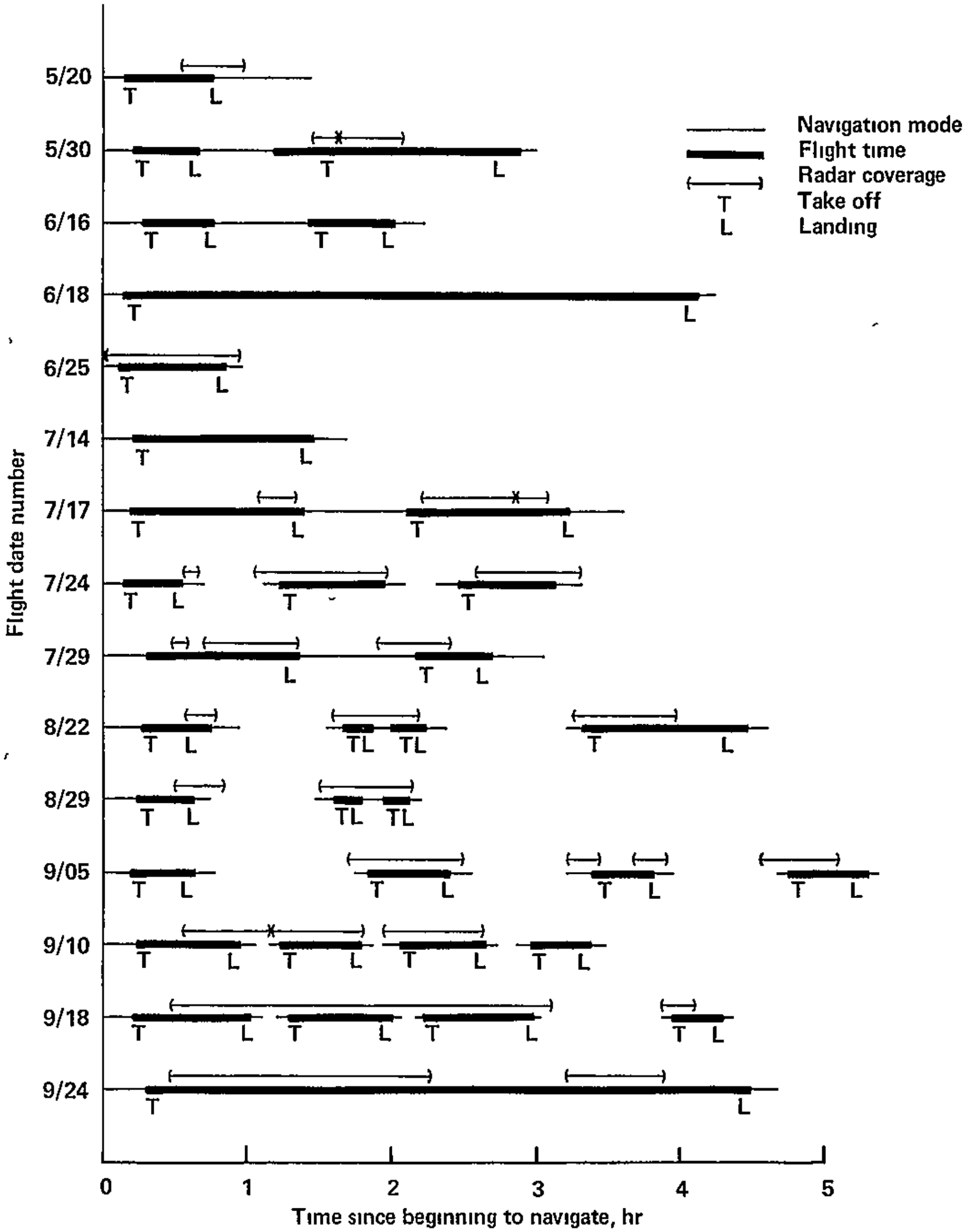
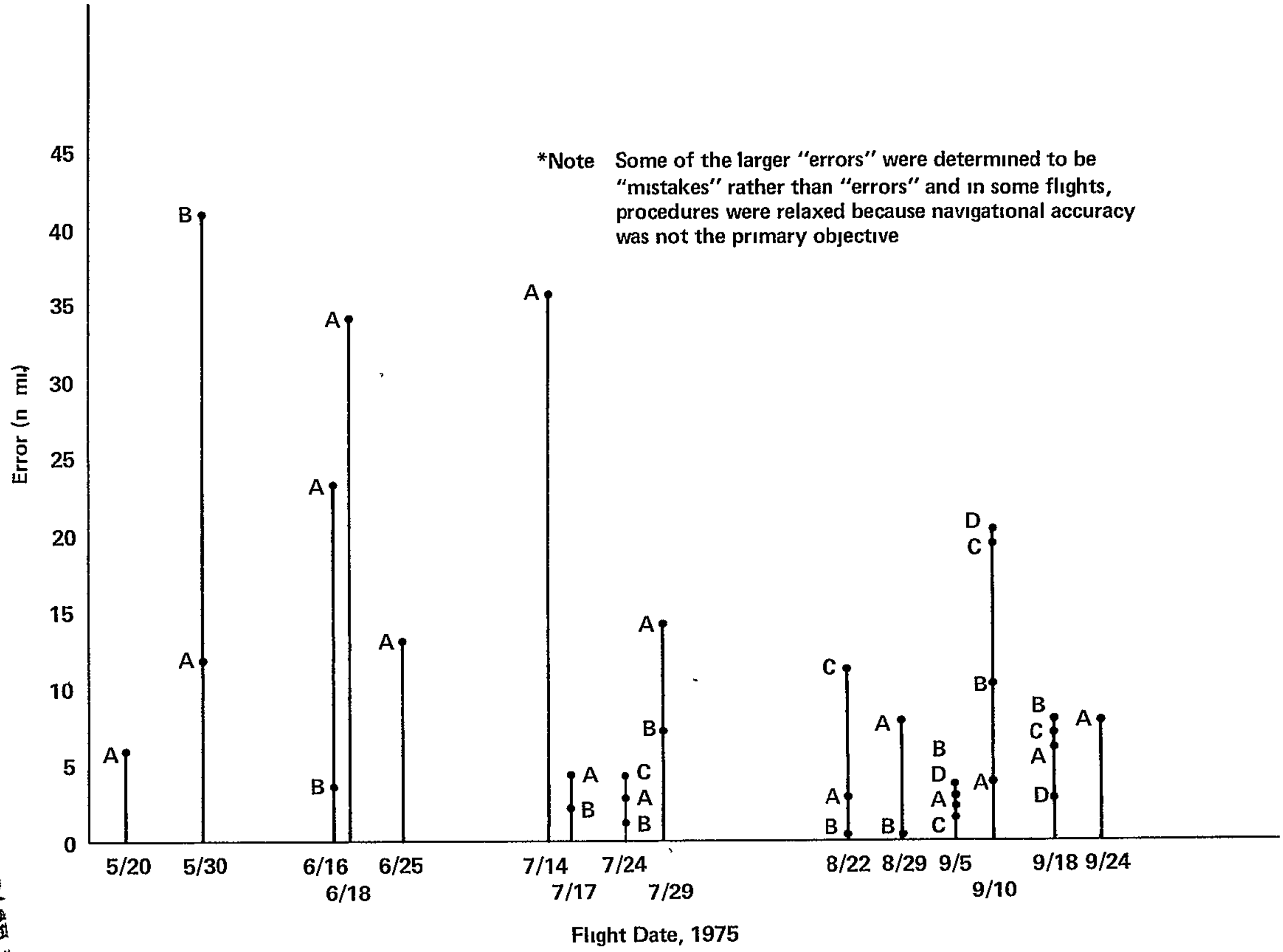
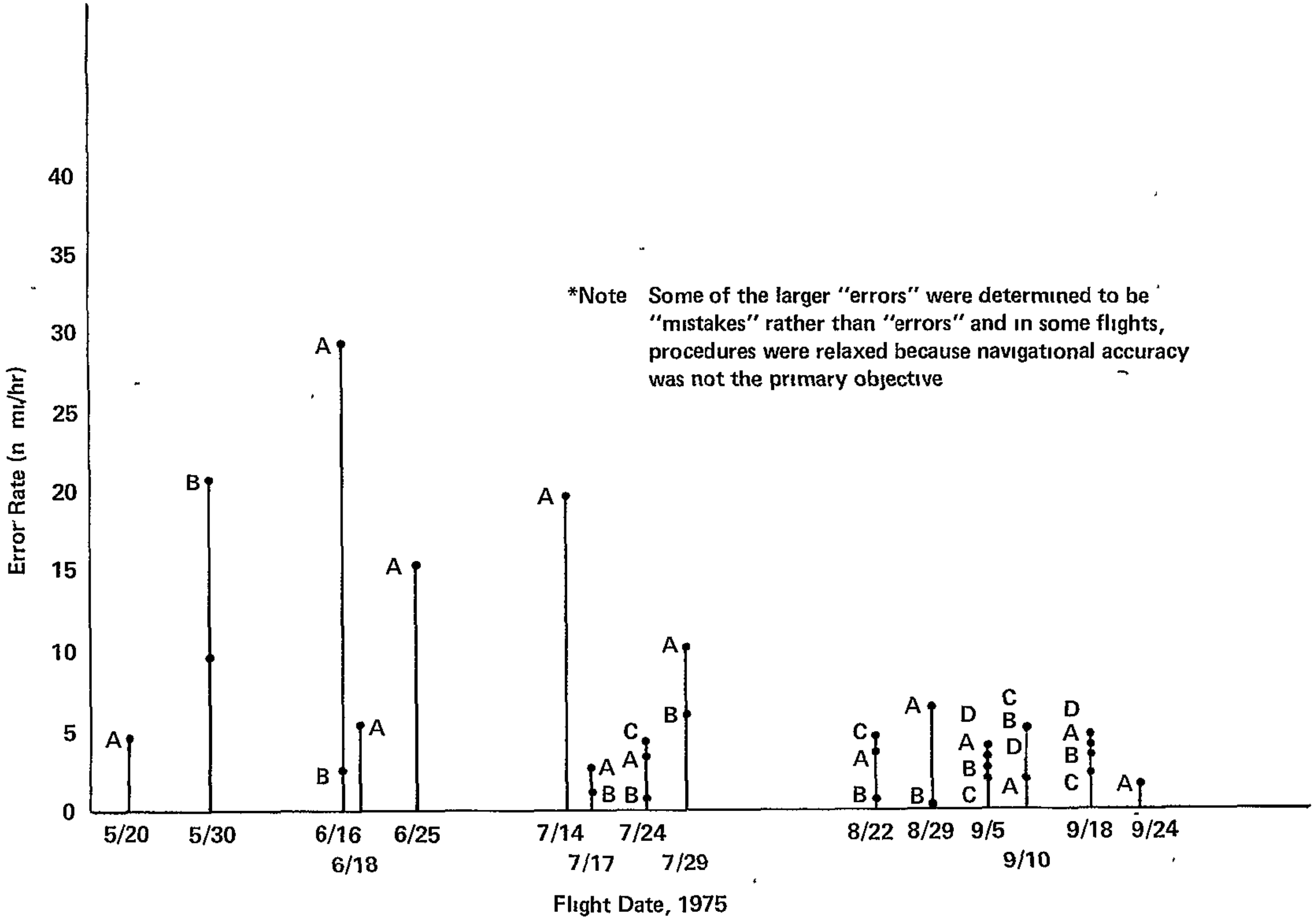


Figure 17 - SIRU flight duration and radar coverage



*Note Some of the larger "errors" were determined to be "mistakes" rather than "errors" and in some flights, procedures were relaxed because navigational accuracy was not the primary objective

Figure 18.- SIRU maximum navigation error vs flight date.



*Note Some of the larger "errors" were determined to be "mistakes" rather than "errors" and in some flights, procedures were relaxed because navigational accuracy was not the primary objective

Figure 19.- SIRU navigational error rate vs flight date.

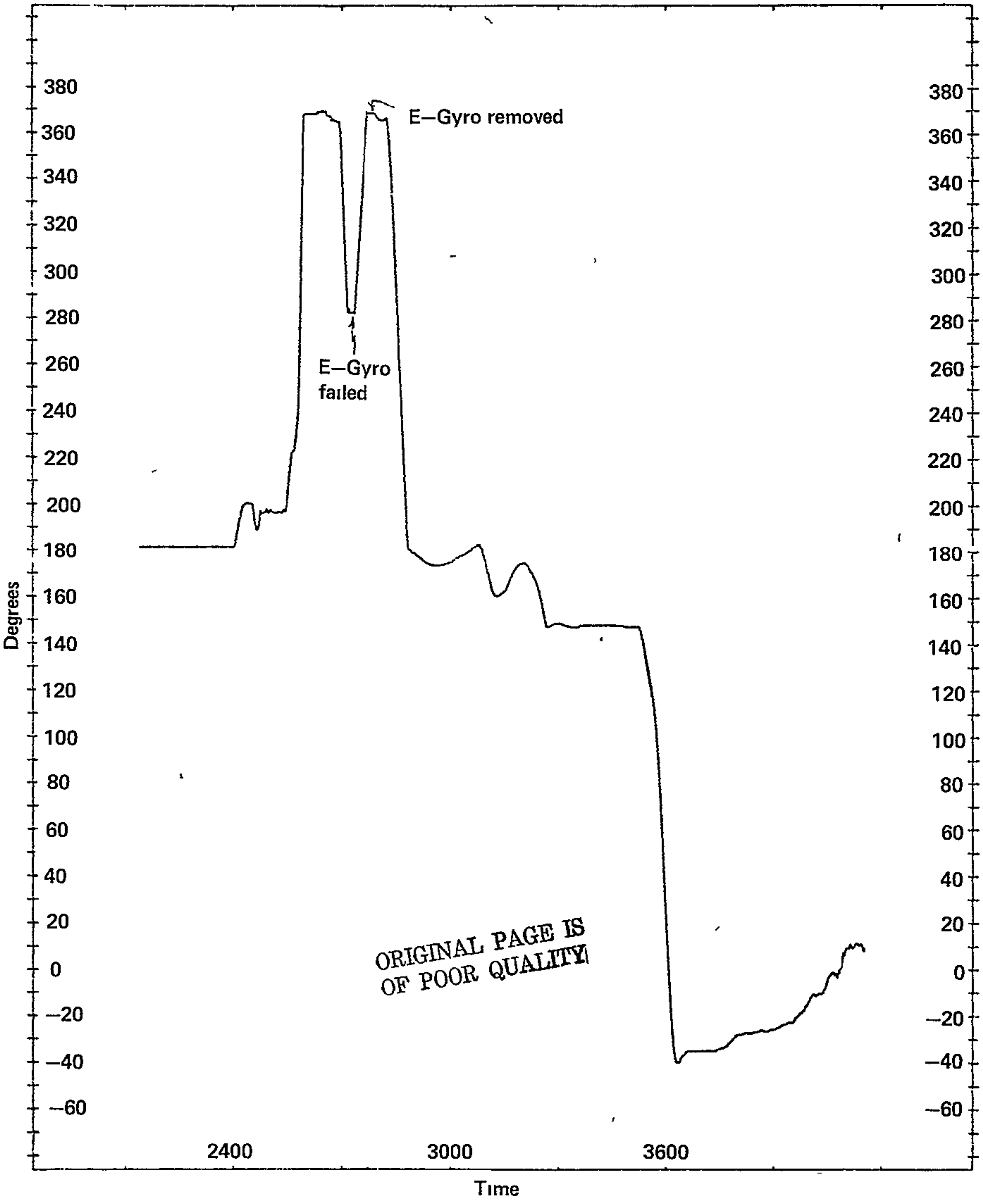


Figure 20.- Aircraft heading vs time, according to the reference computer (A), flight 9/05C.

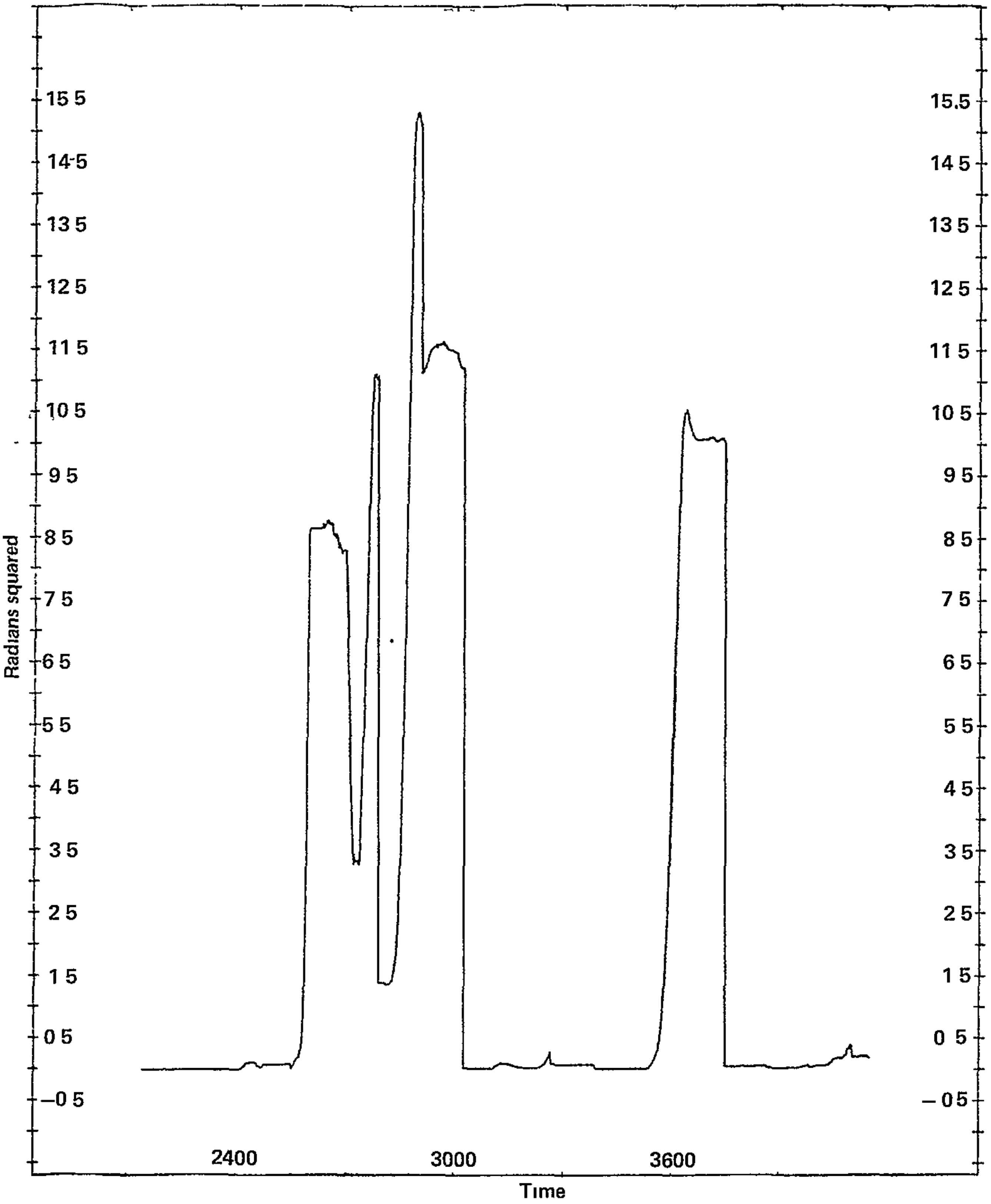


Fıgure 21.- Maxımum allowable squared error (MASE) for gyros vs tıme, 'flıght 9/05C

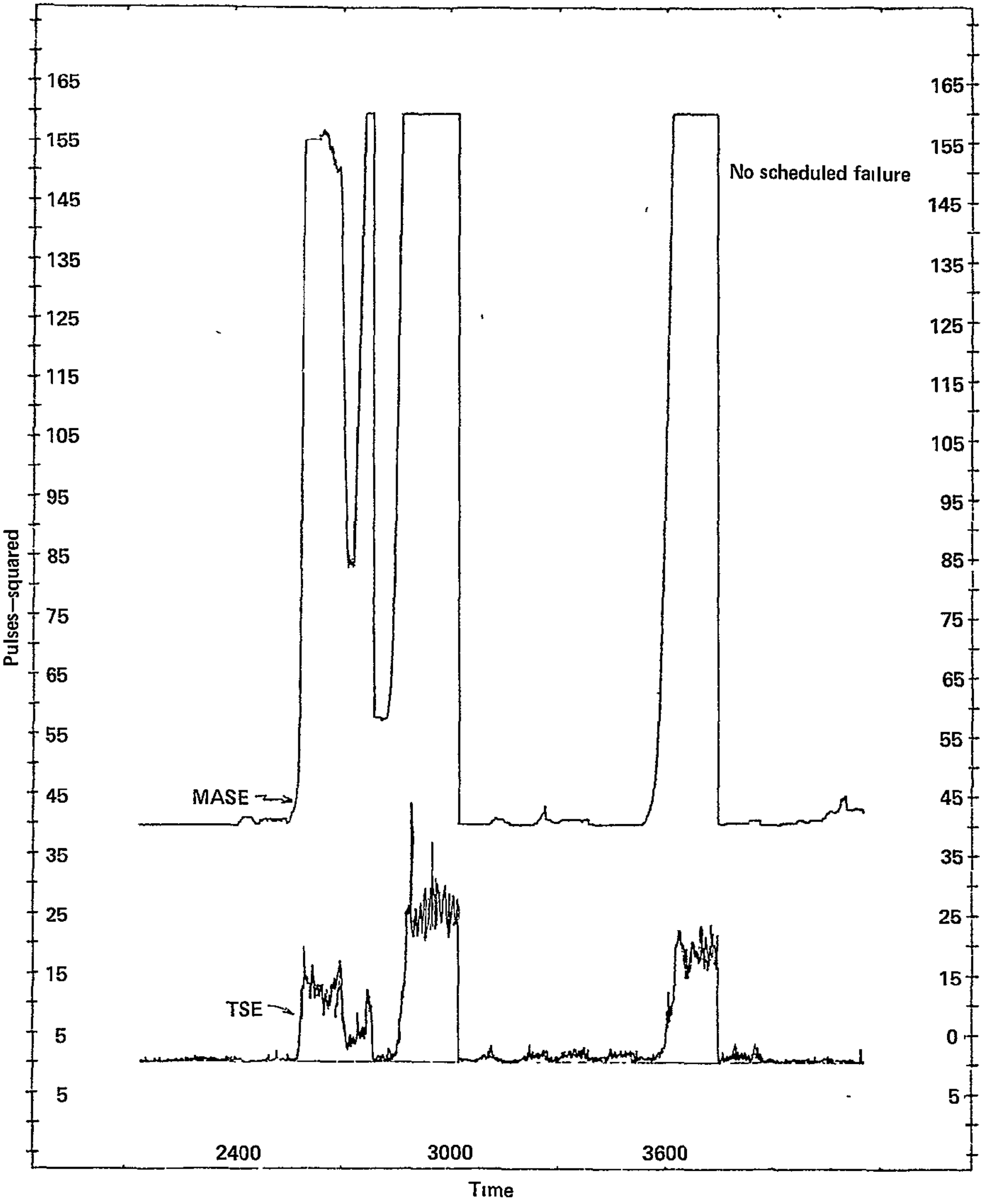


Figure 22 - Maximum allowable squared error (MASE) and total squared error (TSE) for gyros vs time, reference computer (A), flight 9/05C.

ORIGINAL PAGE IS
OF POOR QUALITY

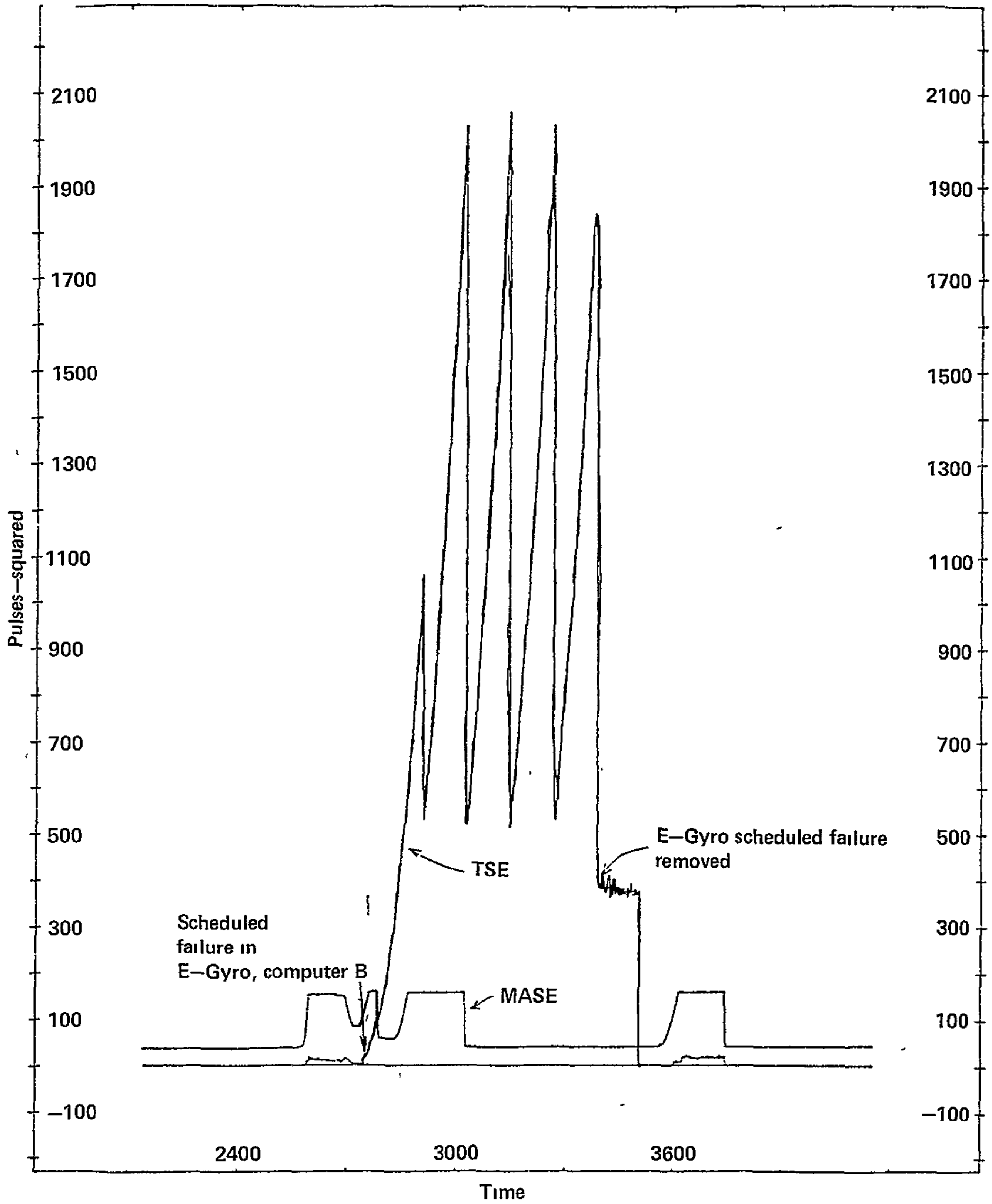
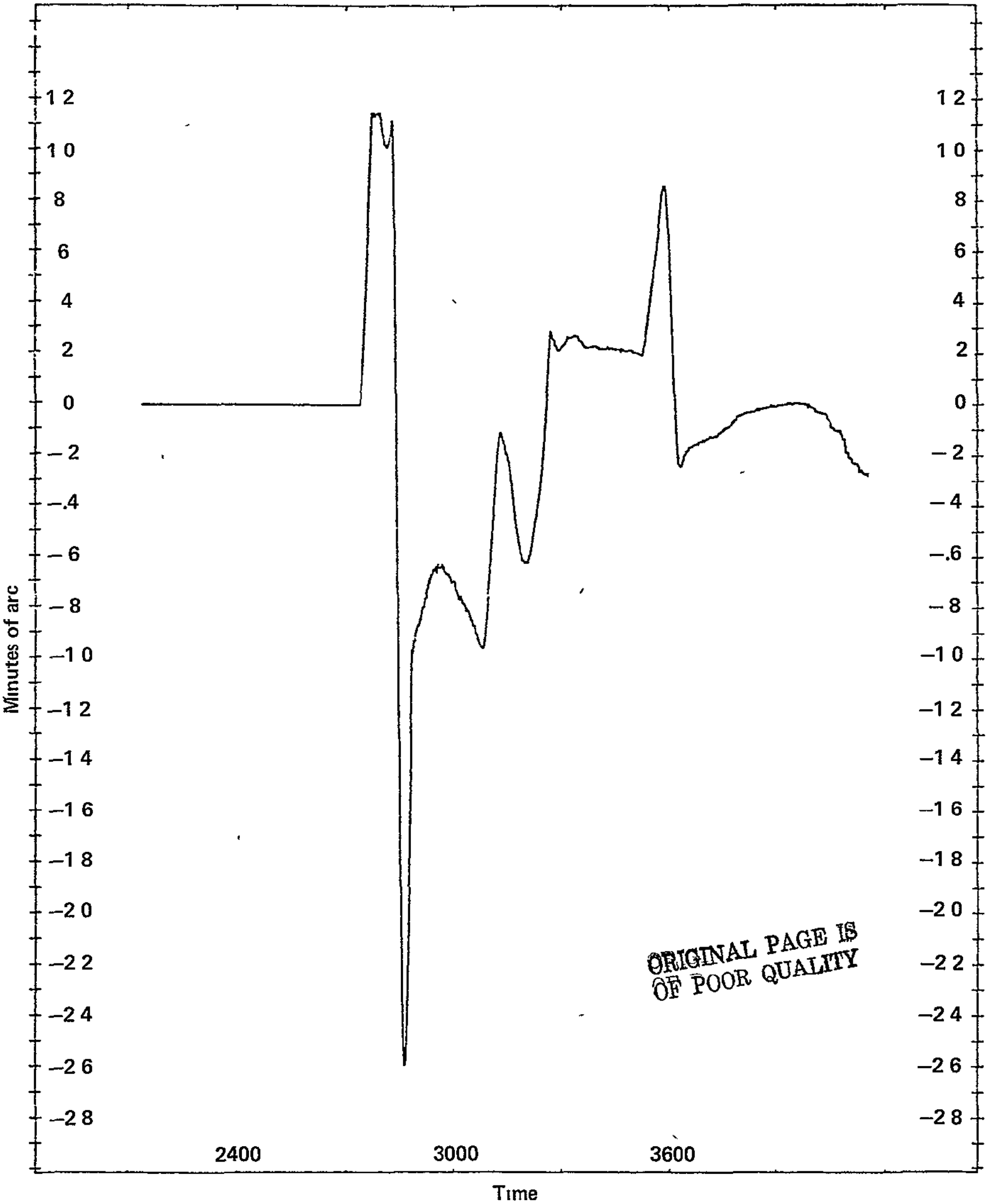


Fıgure 23.- Maxımum allowable squared error (MASE) and total squared error (TSE) for gyros vs tıme, faılure test computer (B), flıght 9/05C.



ORIGINAL PAGE IS OF POOR QUALITY

Figure 24.- Roll angle error due to a scheduled-and-removed E-gyro failure, flight 9/05C.

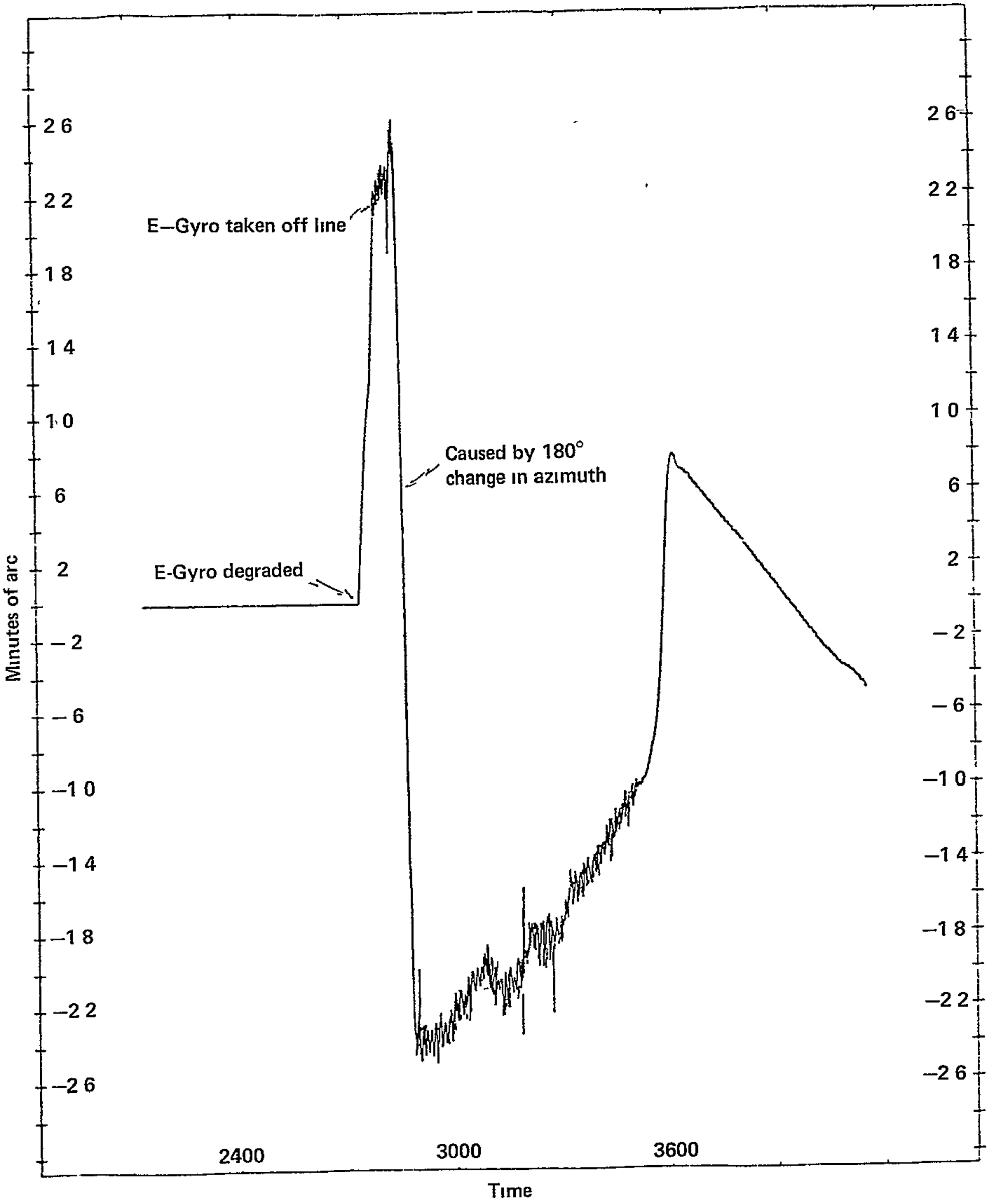


Figure 25.- Pitch angle error due to a scheduled-and-removed E-gyro failure, flight 9/05C.

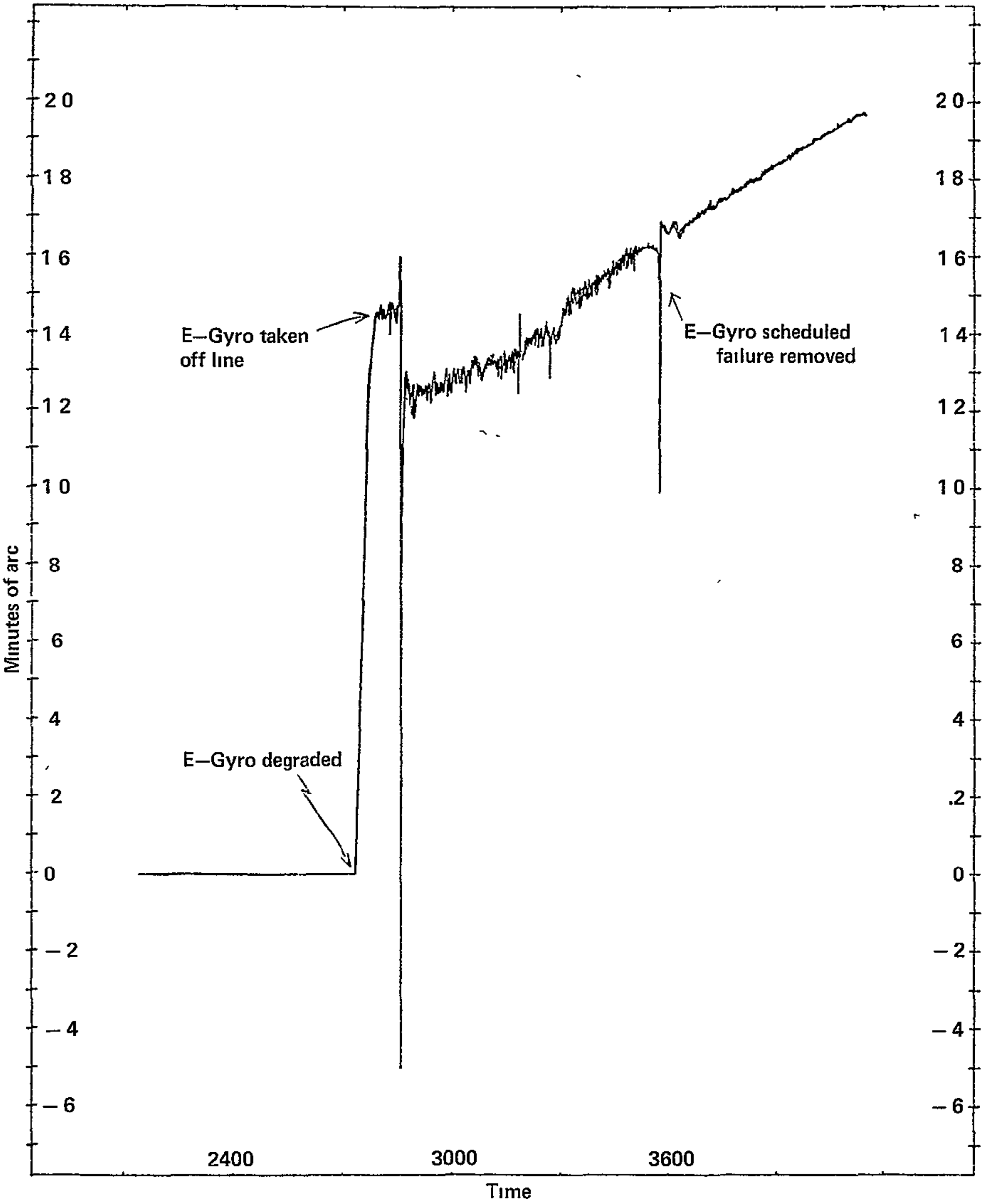


Figure 26.- Yaw angle error due to a scheduled-and-removed E-gyro failure, flight 9/05C.

ORIGINAL PAGE IS OF POOR QUALITY

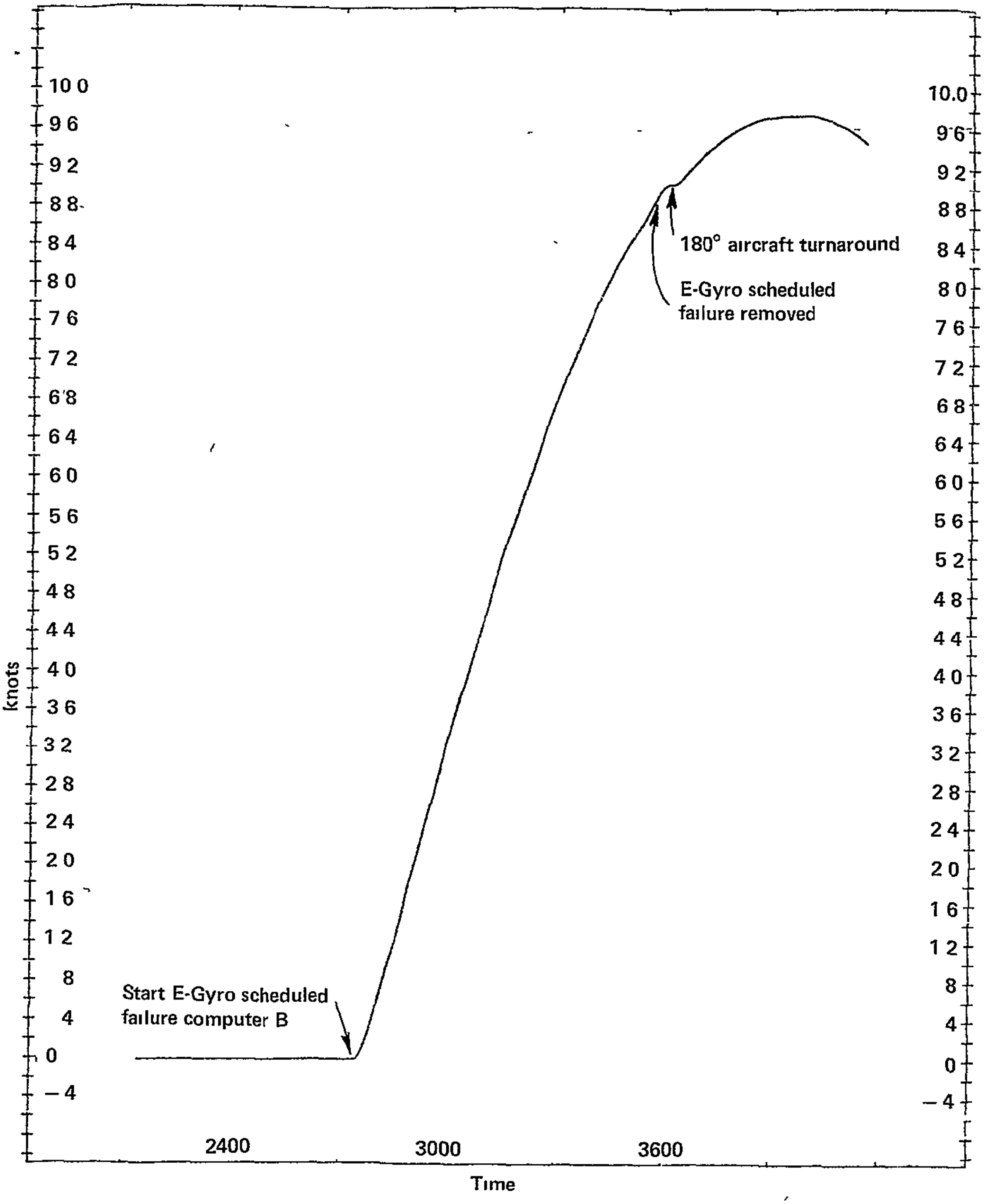


Figure 27.- Speed error due to a scheduled-and-removed E-gyro failure, flight 9/05C.

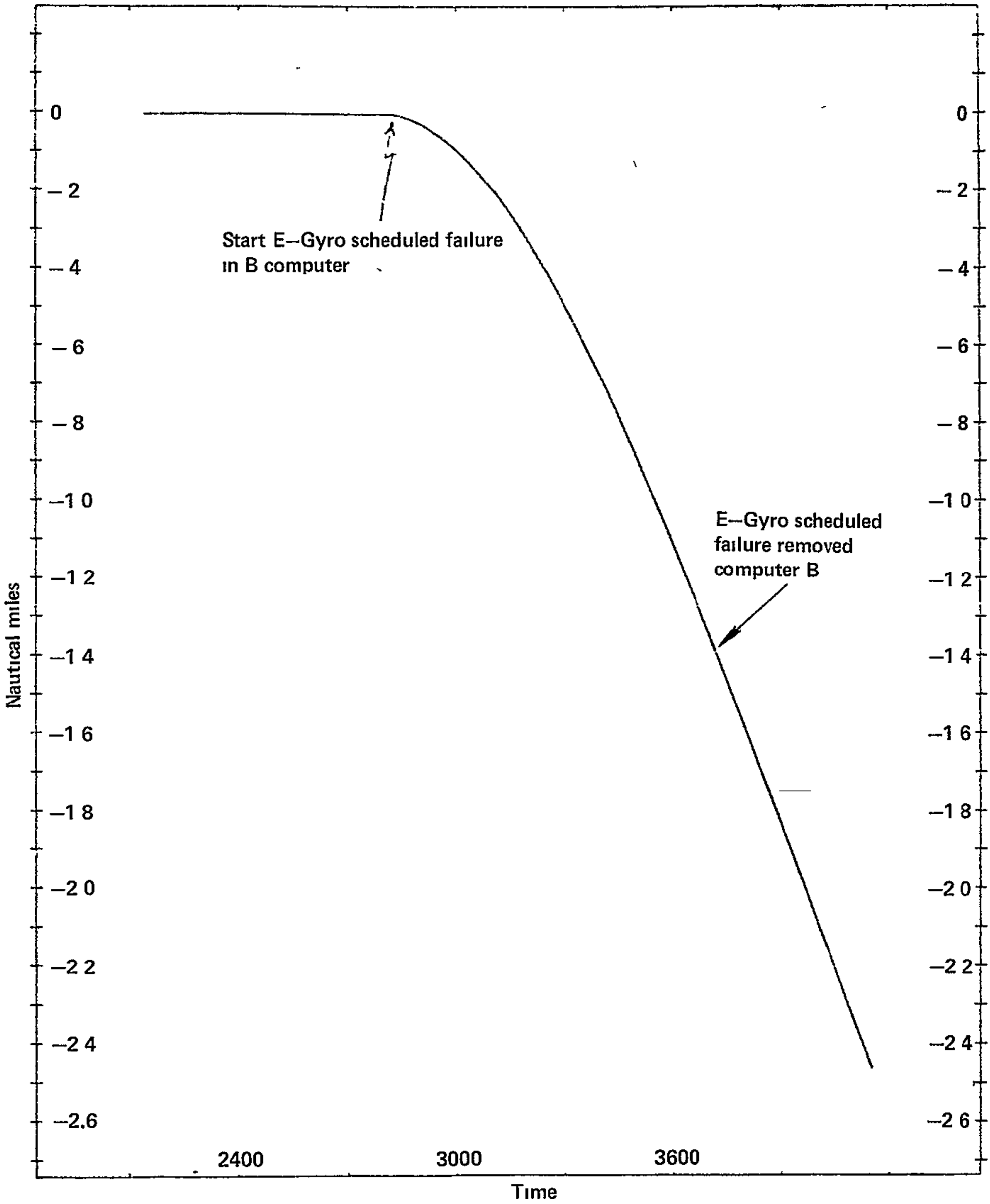


Figure 28.- Latitude error due to a scheduled-and-removed E-gyro failure, flight 9/05C.

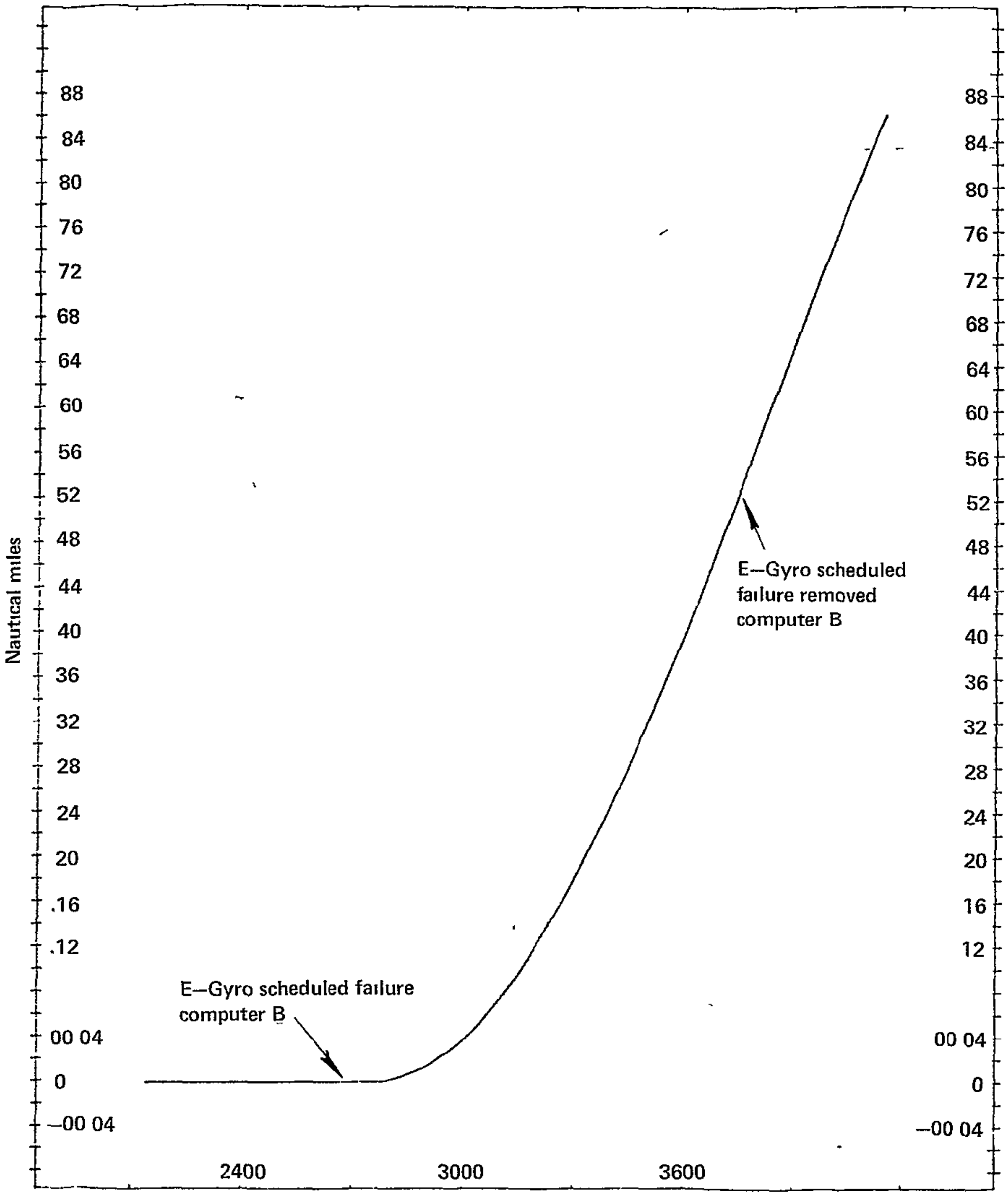


Figure 29.- Longitude error due to a scheduled-and-removed E-gyro failure, flight 9/05C

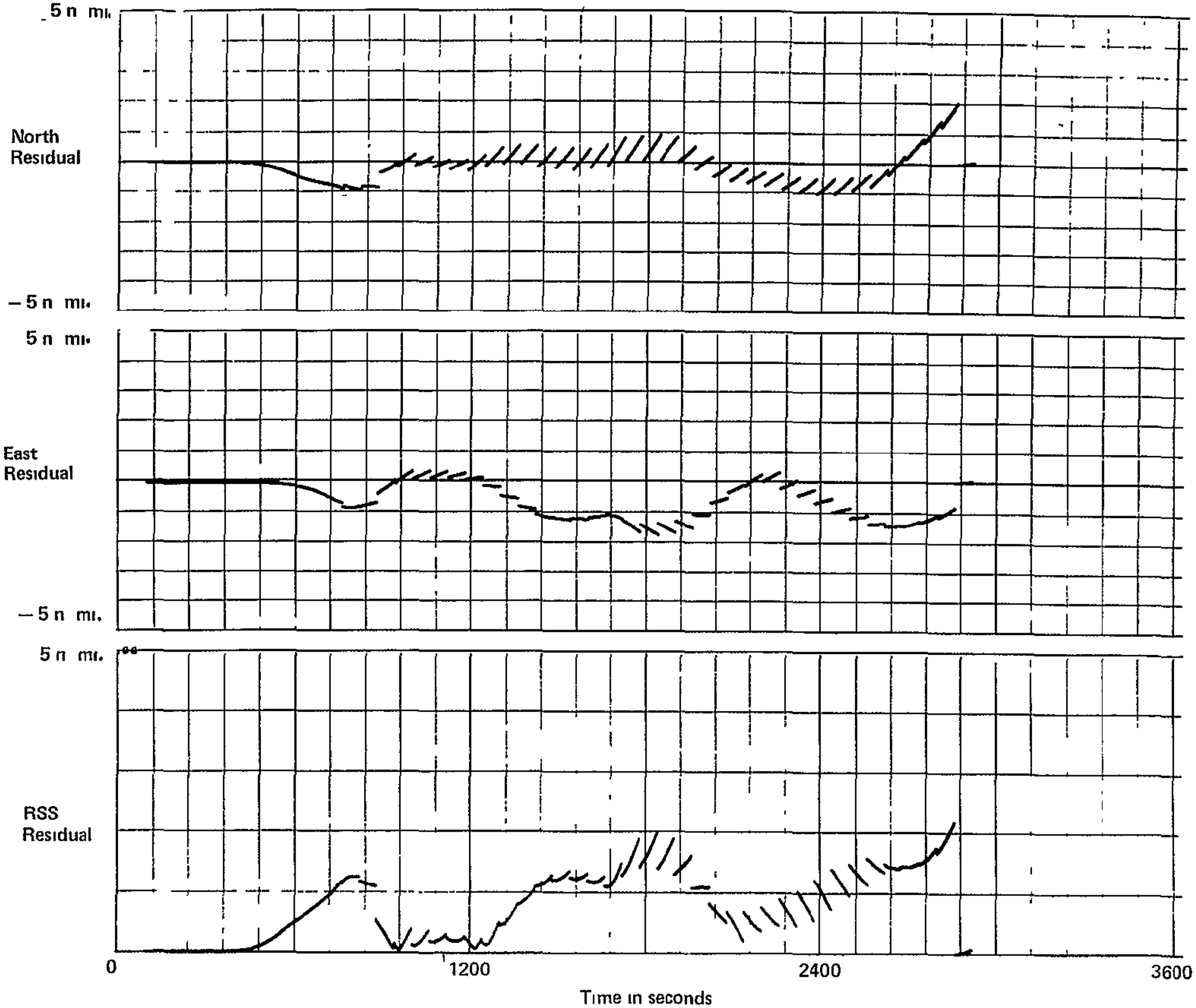


Figure 30 – Position residual history for a DME-aided-inertial trajectory relative to a radar-reference trajectory, flight 9/05C

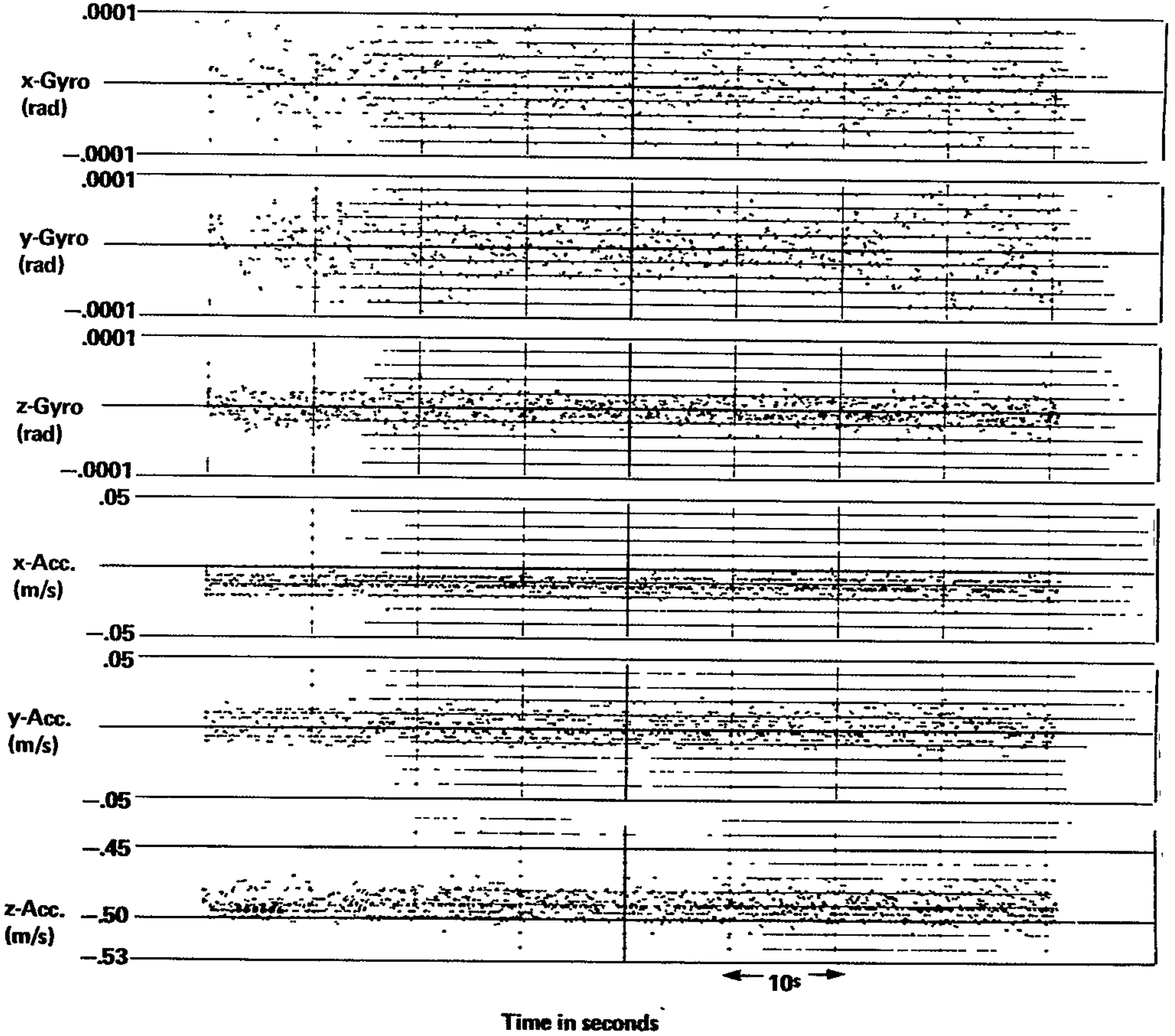


Figure 31.- Equivalent-triad sensor data histories for a stationary, engines-on portion of flight 9/18B.

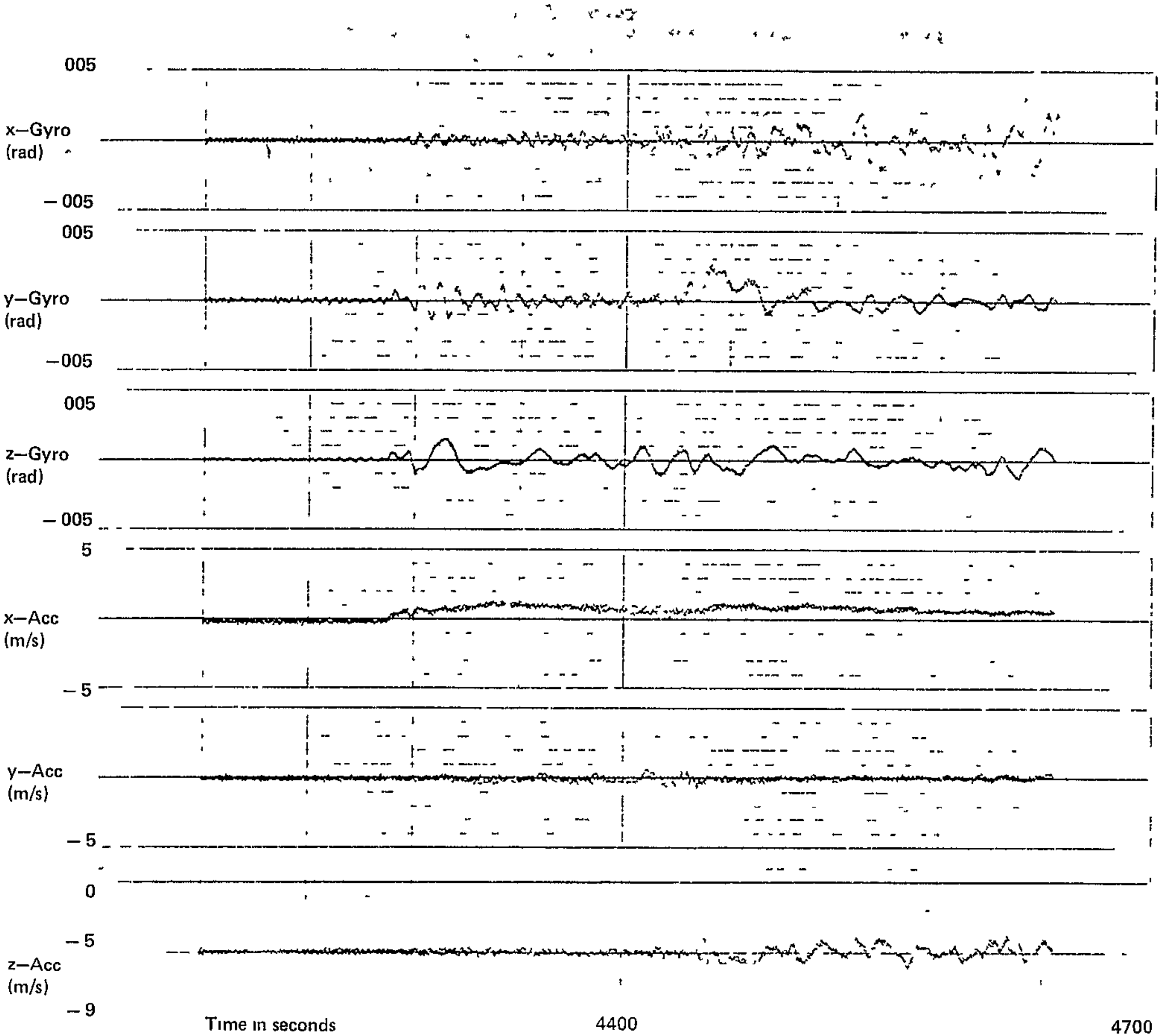


Figure 32 - Equivalent-triad sensor data histories for the takeoff portion of flight 9/18B

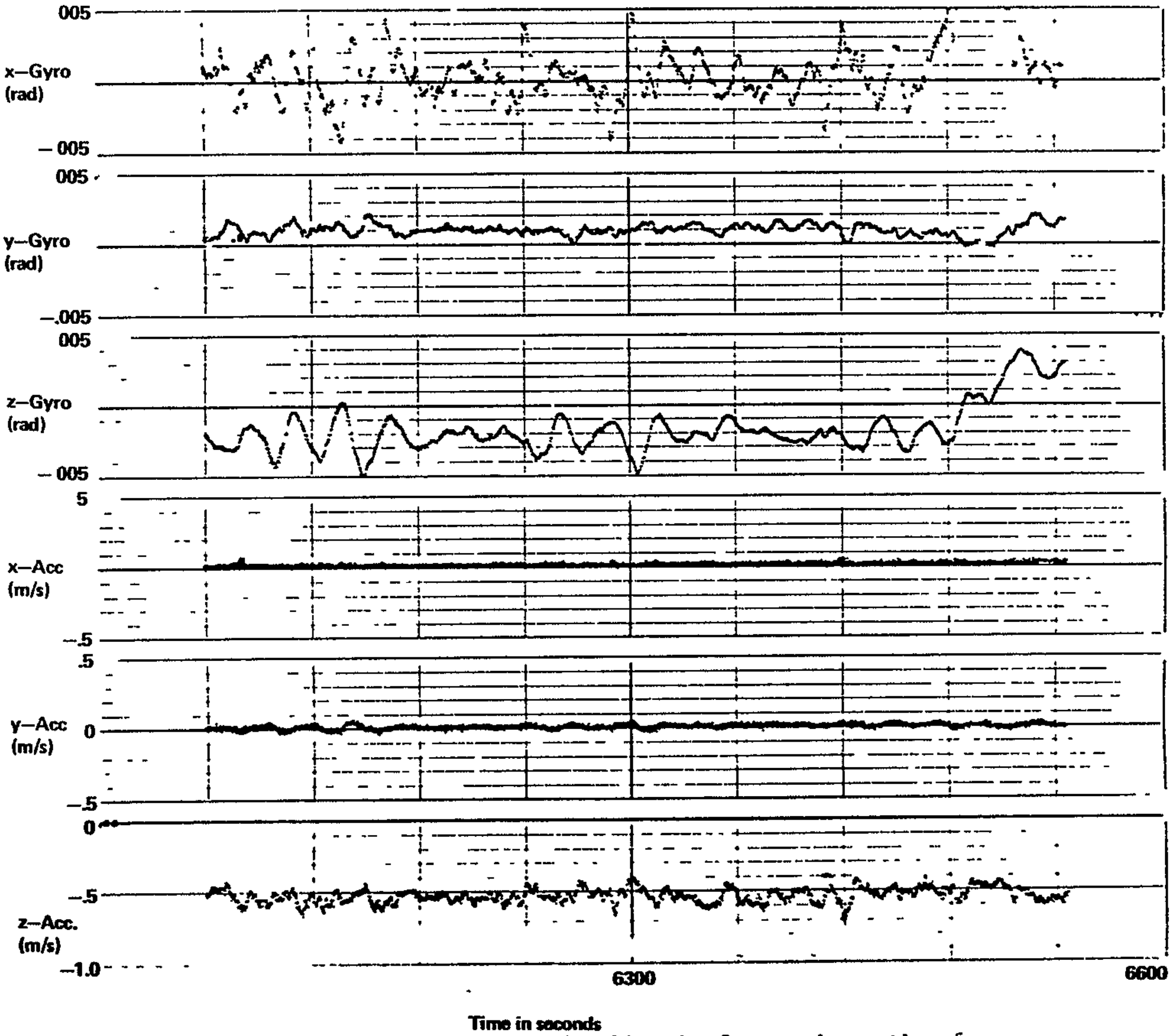


Figure 33.- Equivalent-triad sensor data histories for a cruise portion of flight 9/18B.

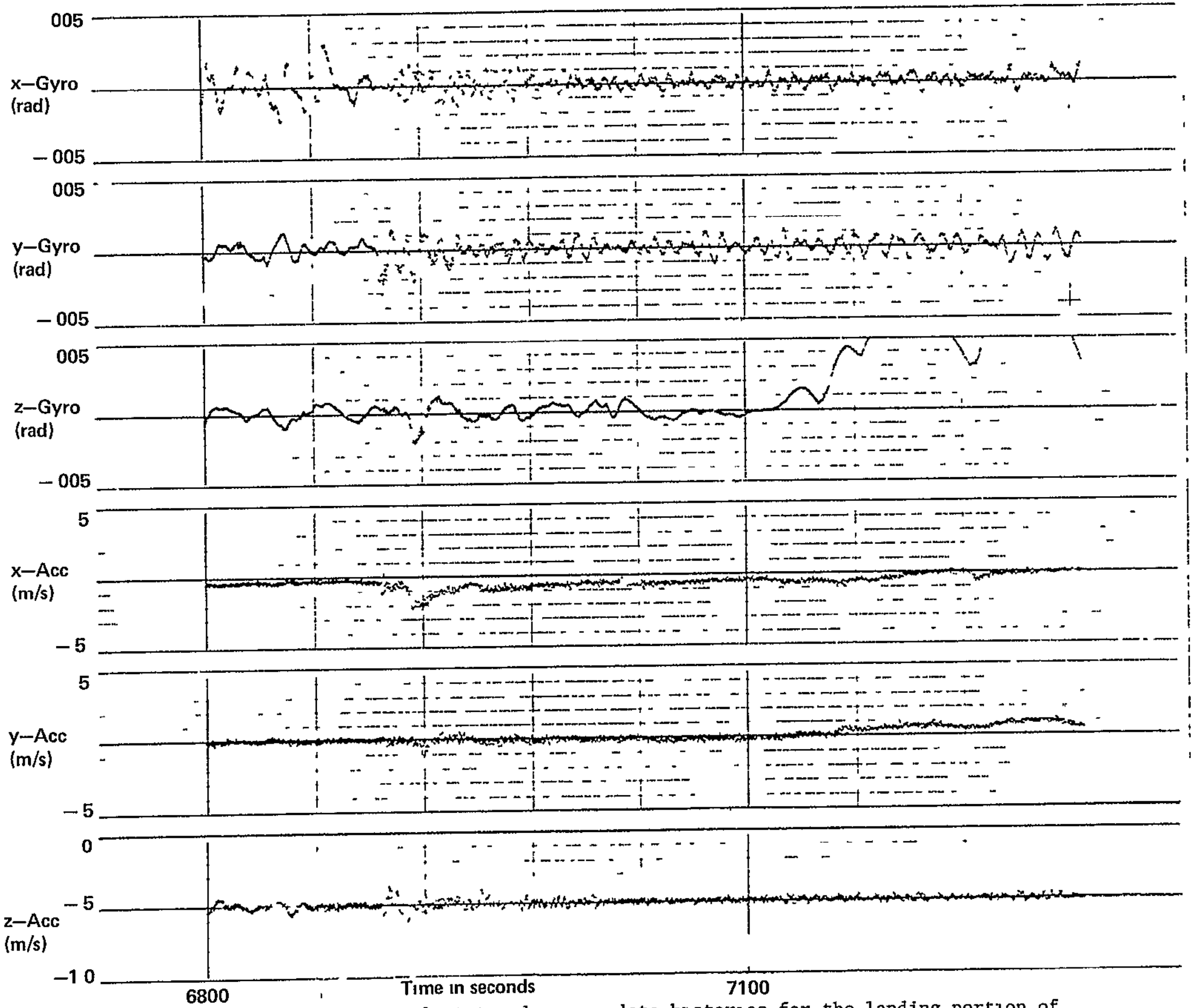


Figure 34 - Equıvalent-trıad sensor data hıstorıes for the landing portıon of flıght 9/18B

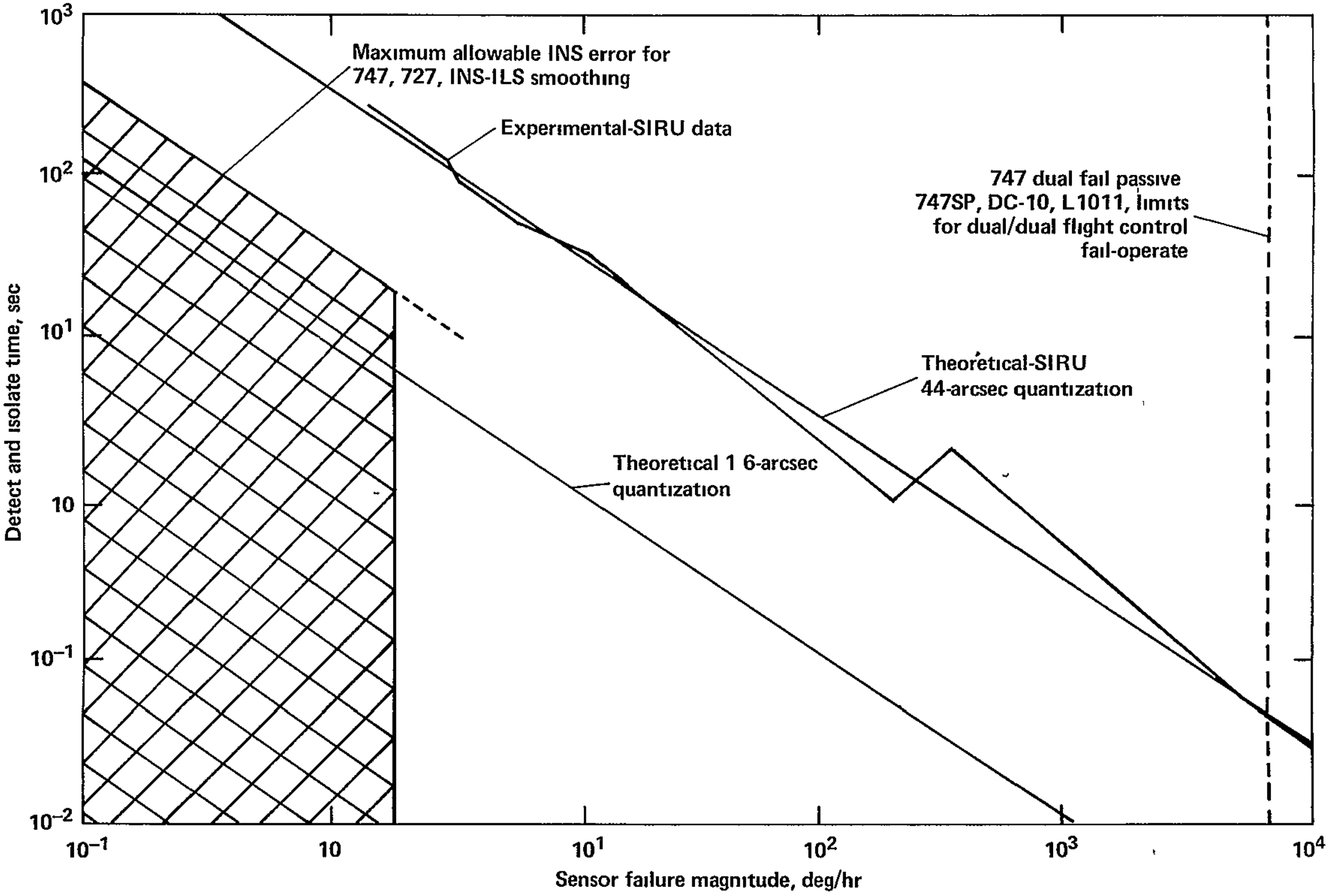


Fıgure 35 - Theoretıcal and experımental faılure detectıon and ısolatıon tıme as a functıon of faılure magnıtude

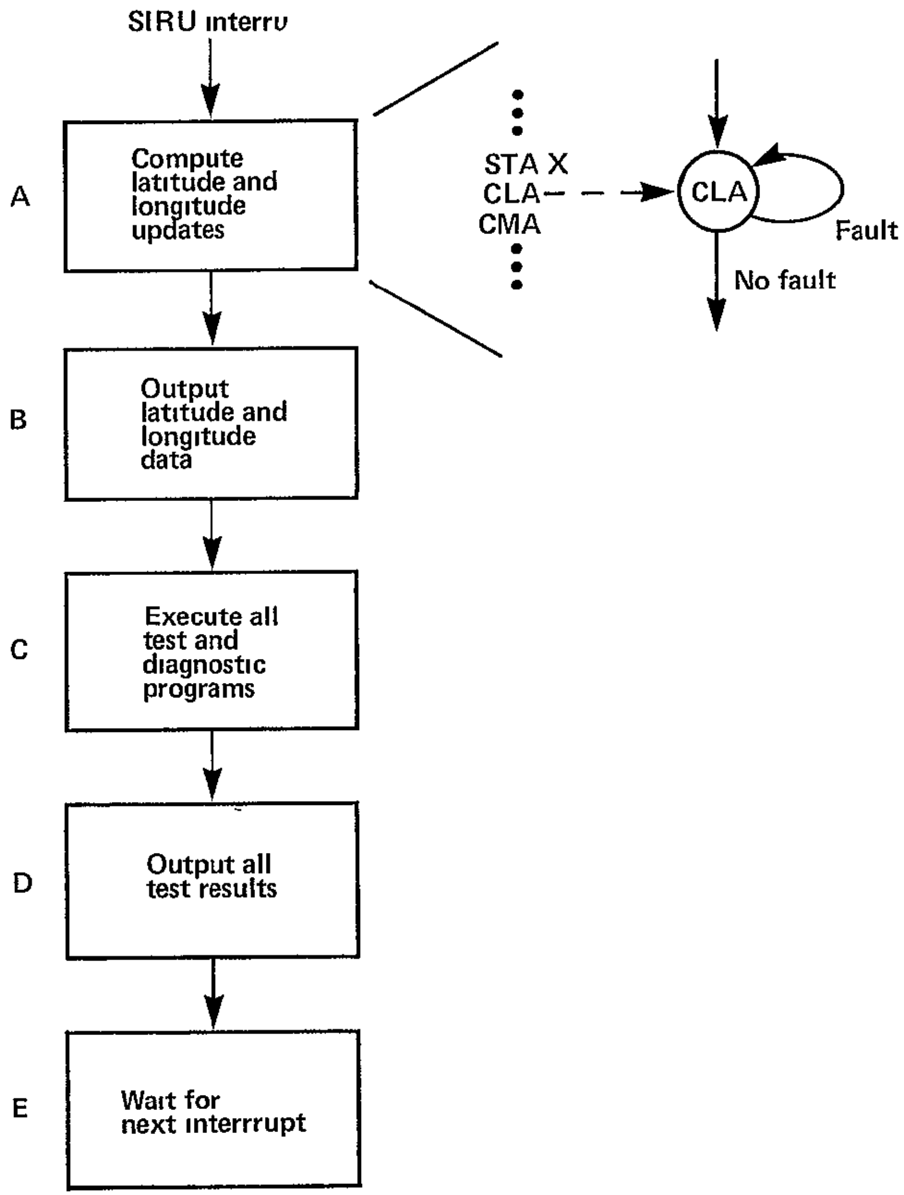

SIRU interru
A
Compute latitude and longitude updates
B
Output latitude and longitude data
C
Execute all test and diagnostic programs
D
Output all test results
E
Wait for next interrrupt
STA X
CLA
CMA
CLA
Fault
No fault

Figure 36.- SIRU software states.

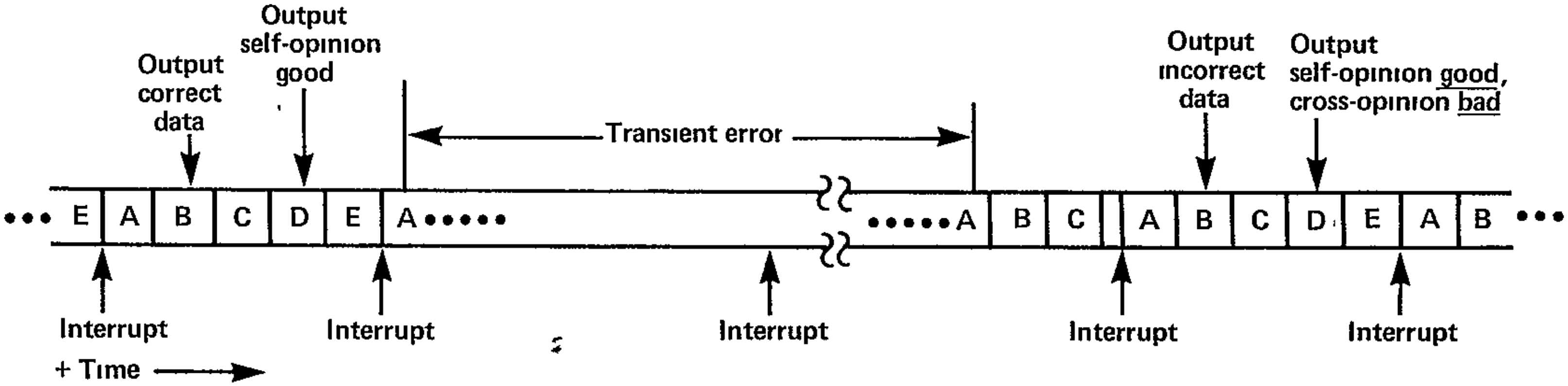


Figure 37.- Software behavior during a transient fault.

1 Report No NASA TM X-73,223	2 Government Accession No	3 Recipient s Catalog No

4 Title and Subtitle FLIGHT TEST RESULTS OF THE STRAPDOWN HEXAD INERTIAL REFERENCE UNIT (SIRU) VOLUME II: TEST REPORT	5 Report Date 6 Performing Organization Code A-6973
7 Author(s) Ronald J. Hruby and William S Bjorkman*	8 Performing Organization Report No
9 Performing Organization Name and Address Ames Research Center Moffett Field, Calif. 94035	10 Work Unit No 513-53-05 11 Contract or Grant No
12 Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D. C. 20546	13 Type of Report and Period Covered Technical Memorandum 14 Sponsoring Agency Code

15 Supplementary Notes

*Analytical Mechanics Associates, Inc.
Mountain View, California 94040

16 Abstract

Results of flight tests of the Strapdown Inertial Reference Unit (SIRU) navigation system are presented. The fault-tolerant SIRU navigation system features a redundant inertial sensor unit and dual computers. System software provides for detection and isolation of inertial sensor failures and continued operation in the event of failures. Flight test results include assessments of the system's navigational performance and fault tolerance

This volume (II) contains a detailed description of the flight test program and the observed performance of the SIRU inertial navigation system. Performance shortcomings are analyzed. Companion volumes to this one are·

Volume I Flight Test Summary
Volume III· Appendixes

17 Key Words (Suggested by Author(s)) Strapdown inertial navigation Redundancy management, fault-tolerance Aircraft navigation	18 Distribution Statement Unlimited STAR Category - 04

19 Security Classif (of this report) Unclassified	20 Security Classif (of this page) Unclassified	21 No of Pages 96	22 Price* $5.00

*For sale by the National Technical Information Service Springfield, Virginia 22161