

REC-1063

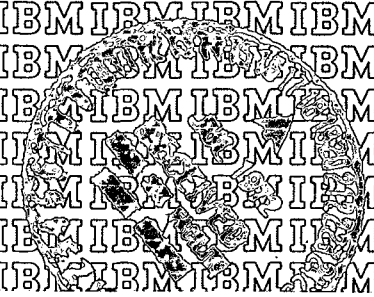
(NASA-CR-123824) APOLLO STUDY REPORT,
VOLUME 1 (International Business Machines
Corp.) 1 Oct. 1963 188 p

N72-75110

Unclas
00/99 36428

CR-123824 Apollo Study Report Volume I

Technical Library, Belmont, MA



Apollo Study Report
Volume I

Contract No. NAS 8-5296

CLASSIFICATION AND CONTENTS

APPROVED BY: mm Dickson

PROJECT OFFICE APPROVAL: RE Kuehn

IBM NO.: 63-928-129

Prepared for the
GEORGE C. MARSHALL SPACE FLIGHT CENTER
Huntsville, Alabama

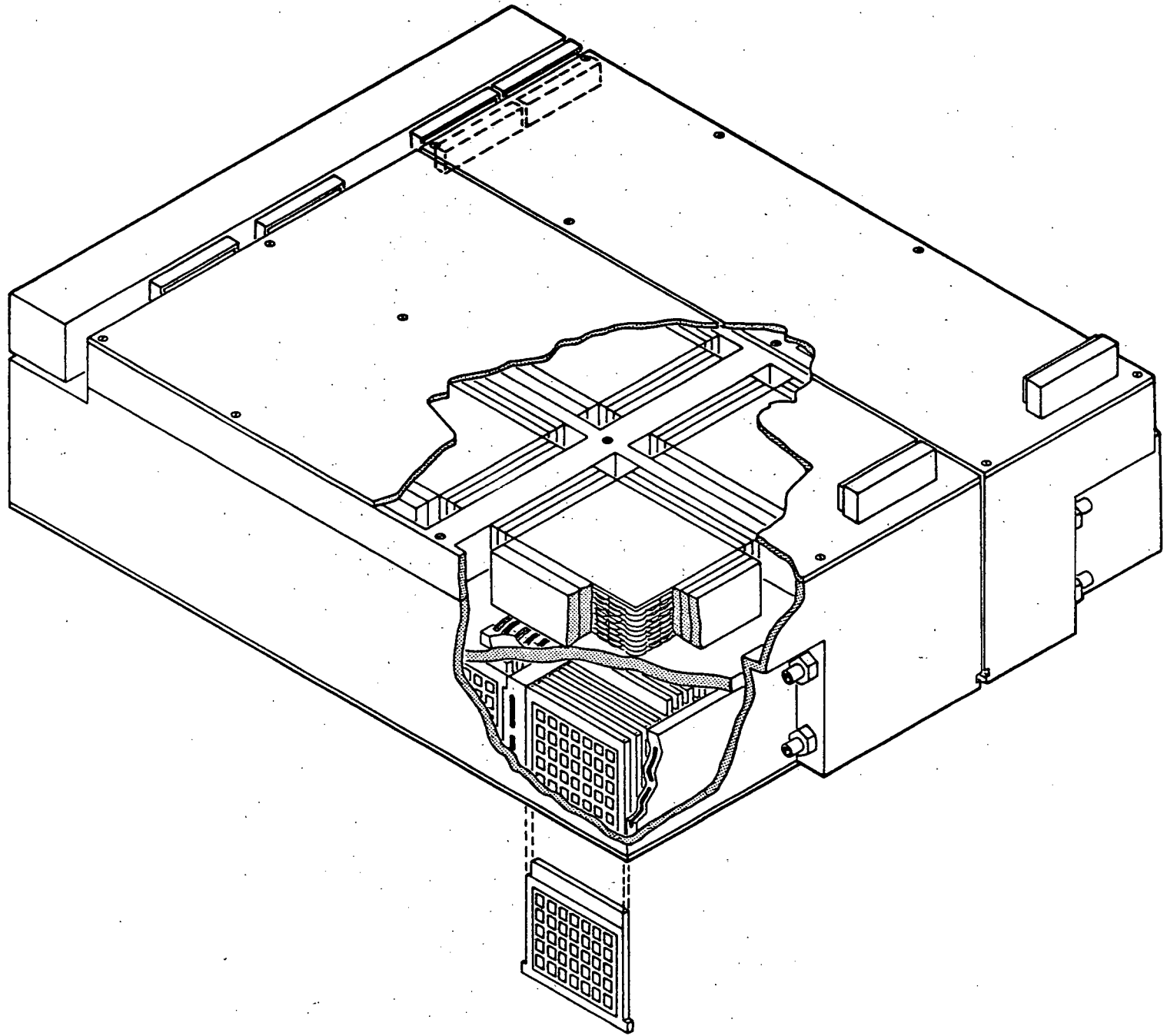
IBM Space Guidance Center, Owego, New York

FOREWORD

The feasibility of adapting the Saturn V Guidance Computer, Data Adapter, and Laboratory Test Equipment to the Apollo application was studied by IBM under NASA Contract NAS 8-5296.

This report presents the results of the study effort. In order to publish this report as soon as possible, it is being submitted without prior NASA review. However, discussions with NASA personnel contributed significantly to the compilation of this report. Volume I describes equipment that will successfully fulfill the requirements of the Apollo mission. Volume II describes the equipment currently being developed for Saturn V. The essential similarities of Apollo and Saturn V equipment can provide significant time and dollar savings during the Apollo development program.

III/IV



Backup Apollo Guidance Computer

CONTENTS

Section		Page
I	INTRODUCTION	1-1
II	APOLLO GUIDANCE COMPUTER	2-1
	A. GENERAL	2-2
	B. CENTRAL PROCESSOR ORGANIZATION	2-7
	1. FUNCTIONAL DESCRIPTION	2-7
	2. COMPUTER CONTROL	2-12
	3. PROGRAMMING CONSIDERATIONS	2-16
	4. ARITHMETIC ELEMENTS	2-18
	5. MEMORY	2-22
	C. DATA ADAPTER ORGANIZATION	2-28
	1. GENERAL	2-28
	2. TIMING	2-28
	3. DOWNLINK	2-32
	4. INTERRUPT	2-33
	5. DISCRETE OUTPUTS	2-33
	6. DISCRETE INPUTS	2-33
	7. ADDRESSING	2-33
	8. REDUNDANCY IN THE APOLLO DATA ADAPTER	2-35
	D. PROPOSED PACKAGING FOR APOLLO GUIDANCE COMPUTER	2-36
	1. GENERAL	2-36
	2. SATURN V PACKAGING REVIEW	2-36
	3. EQUIPMENT PACKAGING APPROACHES FOR APOLLO	2-41

Contents (cont)

Section	Page
II (cont)	E. CIRCUITS 2-50 1. GENERAL 2-50 2. APOLLO POWER SUPPLIES 2-50 3. TRANSFORMER INPUT CIRCUIT 2-54 4. TRANSFORMER OUTPUT CIRCUIT 2-56 5. APOLLO OSCILLATOR 2-57 F. IMPACT OF IDENTICAL CENTRAL PROCESSOR FOR SATURN V AND APOLLO 2-59
III	RELIABILITY 3-1 A. GENERAL 3-2 B. METHOD OF ANALYSIS 3-4 C. SYSTEM MODEL 3-9 D. SUBSYSTEM ANALYSES 3-12 1. CENTRAL PROCESSOR BINARY LOGIC 3-12 2. MEMORY 3-16 3. POWER SUPPLY 3-24 4. DA BINARY LOGIC 3-26 5. DA INPUT-OUTPUT CIRCUITRY 3-27 6. CHRONOMETER 3-29 E. FAILURE RATES AND K-FACTORS 3-30 1. COMPONENT-PART FAILURE RATES AND FAILURE MODES 3-30 2. K-FACTORS 3-37
IV	SYSTEMS AND PROGRAMMING 4-1 A. STUDY GOALS 4-2 B. INPUT/OUTPUT PROGRAMMING 4-4 1. GENERAL 4-4 2. DOWNLINK 4-4

CONTENTS

Section	Page
IV (cont)	<ul style="list-style-type: none"> 4. CDU (CONTROL AND DISPLAY UNIT) 4-8 5. END PULSE FAILURE 4-8 6. INTERRUPT BOOKKEEPING 4-9 7. OTHER I/O COMPUTATIONS 4-9
C.	COMPARISON OF STORAGE AND SPEED OF AGC-4 AND SATURN V 4-9
D.	DOUBLE REGISTER 4-10
E.	SOFTWARE 4-12 <ul style="list-style-type: none"> 1. GENERAL 4-12 2. ASSEMBLER 4-12 3. SIMULATOR 4-13 4. LOGIC SIMULATOR 4-13
F.	POTENTIAL HARDWARE CHANGES 4-14
V	LABORATORY TEST EQUIPMENT (LTE) 5-1
A.	INTRODUCTION AND PHILOSOPHY OF LTE SUPPORT TO AGC HARDWARE 5-2 <ul style="list-style-type: none"> 1. CENTRAL PROCESSOR (CP) HARDWARE TESTS 5-2 2. DATA ADAPTER (DA) HARDWARE TESTS 5-2 3. POWER SUPPLY (PS) HARDWARE TESTS 5-3 4. AGC HARDWARE INTEGRATION TESTS 5-3 5. LTE SUPPORT TO AGC SOFTWARE OPERATIONS 5-3 6. GENERAL DESIGN AND PACKAGING TECHNIQUES FOR LTE 5-8 7. TESTING TECHNIQUES 5-12
B.	SATURN V AEROSPACE COMPUTER MANUAL EXERCISER (ACME) 5-12 <ul style="list-style-type: none"> 1. INSTALLATION CONFIGURATION 5-12 2. TESTS TO BE PERFORMED ON THE CP USING ACME 5-12 3. MACHINE ORGANIZATION 5-14 4. ACME DESIGN GOALS 5-16

CONTENTS

Section		Page
V (cont)	C. APOLLO DATA ADAPTER TESTER AND MONITOR (ALAMO)	5-16
	1. INSTALLATION CONFIGURATION	5-16
	2. TESTS TO BE PERFORMED ON THE DATA ADAPTER USING ALAMO	5-16
	3. MACHINE ORGANIZATION	5-18
	4. ALAMO DESIGN GOALS	5-18
	D. APOLLO GUIDANCE COMPUTER EVALUATION EQUIPMENT (APOGEE)	5-20
	1. INSTALLATION CONFIGURATION	5-20
	2. TESTS TO BE PERFORMED ON THE AGC SYSTEM USING APOGEE	5-20
	3. MACHINE ORGANIZATION	5-20
	4. APOGEE DESIGN GOALS	5-20
VI	PROGRAM PLAN	6-1
	Appendix A — SYSTEM FAILURE RATES	A-1
	Appendix B — "RANDOM NUMBER" RELIABILITY CAL- CULATION	B-1

ILLUSTRATIONS

Figure		Page
I-1	Unique Structural Approach	1-9
I-2	Common Central Processor Structural Approach	1-10
II-1	Apollo Guidance and Control Block Diagram	2-5
II-2	Central Processor Block Diagram	2-8
II-3	Computer Timing	2-11
II-4	MPY-DIV Timing Chart	2-15
II-5	Self-Correcting Duplex Toroid Memory System	2-23
II-6	Coordinate Half-Select Current Error Detection	2-25
II-7	Simplex Memory Module	2-26
II-8	Apollo Data Adapter Block Diagram	2-30
II-9	ULD Layout	2-38
II-10	MIB Top Land Pattern for 70-ULD Page	2-39
II-11	Page Assembly	2-40
II-12	Volume Allotted for Apollo CP and DA	2-43
II-13	Proposed Memory Package	2-43
II-14	Volume Allocated for Power Supply	2-44
II-15	Apollo Power Supply	2-45
II-16	Central Processor and Data Adapter in Individual Housings	2-46
II-17	Single-Housing Central Processor and Data Adapter	2-49
II-18	Bias Circuits	2-52
II-19	Transformer Coupled Input Circuit	2-55
II-20	Transformer Coupled Output Circuit	2-56
II-21	Transformer Coupled Output Logic	2-57
III-1	Simplex Module Failure Rate	3-7
III-2	System Model	3-11
III-3	Processor Binary Logic Reliability Model	3-13
III-4	Power Supply Configuration (simplex module)	3-23
III-5	Output Driver Configuration	3-27
IV-1	Apollo/LEM Downlink, Display, and CDU Math Flow Diagram	4-7
IV-2	Apollo/LEM Display and CDU Subroutines	4-8

Illustrations (cont)

Figure		Page
V-1	Conceptual Flow Diagram for The Test Operations on the Proposed Apollo Guidance Computer (AGC) Central Processor (CP) Hardware Tests	5-4
V-2	Typical SMS Circuit Card	5-9
V-3	Example of LTE Using SMS Technology	5-11
V-4	ACME Configuration	5-13
V-5	ACME Block Diagram	5-15
V-6	ALAMO Configuration	5-17
V-7	ALAMO Block Diagram	5-19
V-8	APOGEE Configuration	5-21
V-9	APOGEE Block Diagram	5-22
VI-1	System Definition and Software Schedules	6-5
VI-2	Phase II Prime Hardware Schedules	6-6
VI-3	Phase II LTE Schedules	6-7/8
A-1	Mission Profile (Processor and DA Chronometer for Entire Mission	A-3

TABLES

Table		Page
II-1	Apollo Guidance Computer Characteristics	2-3
II-2	Data and Instruction Word Format	2-9
II-3	Operation Code Map	2-12
II-4	DL Register Availability	2-29
II-5	Power Requirements for the Real Time Clock and Associated Circuitry	2-51
II-6	Component Requirements for Duplexed Power Converters	2-52
II-7	Module Check Control	2-53
II-8	Central Processor/Data Adapter Current Requirements and Voltage Tolerances	2-54
III-1	Component-Part Failure Rates	3-31
III-2	Total Component Part Operating Hour Experience	3-33
III-3	Component-Part Failure Rates Observed on the Titan Missile Guidance Computer Program	3-34
III-4	K-Factors	3-38
A-I	One Channel of Processor TMR Logic Component Failure Rates	A-4
A-II	Timing Generator Module Component Failure Rates	A-5
A-III	Hardware Failure Rates and Probability of Open and Short Circuits	A-6
A-IV	Memory EI Driver Component Failure Rates	A-8
A-V	Memory Circuit Failure Rates	A-9
A-VI	Simplex Memory Phase Reliabilities	A-10
A-VII	Failure Rates of Memory Circuits Having Nonde- tectable Failure Modes	A-12
A-VIII	Nonredundant Power Converter Component Failure Rates	A-13
A-IX	Nonredundant Direct-Current Amplifier Component Failure Rates	A-14
A-X	Data Adapter Simplex Channel Component Failure Rates	A-15
A-XI	Simplex Output Driver Channel Component Failure Rates	A-15
A-XII	Simplex Accumulator Channel Component Failure Rates	A-16

Section I
INTRODUCTION

Section I

INTRODUCTION

The IBM Company has been under contract to NASA since June 1961 to develop guidance computer equipment and technology. This report presents the results of a study performed by the IBM Space Guidance Center in which the applicability of Saturn V guidance computer equipment to the Apollo spacecraft program was examined. The study was authorized by the Marshall Space Flight Center in a directive dated 1 August 1963. It has been directed toward:

- (a) Identifying those modifications to Saturn V guidance computer required to permit its use in the Apollo spacecraft.
- (b) Defining a data adapter that will interface the Saturn V central processor with the Apollo guidance/navigation system.

Several ground rules were applied to the study in order to complete the task within the allocated resources:

- (a) The resultant design should minimize the form factor and installation impact on the Apollo spacecraft.
- (b) The Apollo data adapter design should necessitate no modification of the present Apollo computer - guidance system electrical interface.
- (c) The resultant design should minimize the change impact on the Saturn V equipment consistent with (a) and (b) above.
- (d) Six Saturn V memory modules are required for the Apollo guidance computer. This memory capacity provides duplex storage of 12,288 data words, 24,576 instruction words, or any equivalent combination of data and instructions.
- (e) The present Saturn V central processor logic need not be altered.

Meetings have been held with MIT, NAA, MSC, and MSFC to derive the data essential to fulfilling the purpose of the IBM effort. Additional data will be needed for the final optimization of the back-up Apollo equipment in the

absence of the constraints suggested by the ground rules listed above. This study, however, clearly confirms the feasibility of employing the Saturn V computer in the Apollo spacecraft. Since study results encourage the proposed Apollo application, this report includes a plan for implementing a back-up development program based on Saturn V technology.

The Saturn V guidance computer is characterized by several factors particularly significant to the issue treated in this study:

- (a) The computer design employs a modular memory concept in which from one to eight memory modules, each containing the required memory electronics, may be provisioned. This feature permits the machine to be adjusted to a special application and enables four memory modules to be designated for Saturn V and six memory modules for Apollo without design impact.
- (b) The guidance computer is functionally and mechanically subdivided into a central processor and data adapter; the data adapter accommodates computer input-output functions. Consequently, the Saturn V data adapter can be replaced with an Apollo-compatible unit, reducing the design impact of the Apollo guidance system requirement.
- (c) The Saturn V equipment design uses triple-modular-redundant (TMR) organization and duplex modular redundant memory and power supply elements. This redundancy feature is retained in the proposed Apollo equipment configuration and permits the realization of high mission success probability. More detailed characteristics of the Saturn V - Apollo central processor and the Apollo data adapter are found in Section II of this report.

The equipment identified for the Apollo application allows for a machine commonality between the Saturn V guidance computer and the Apollo guidance computer. The proposed back-up approach employs common central processor units and similar data adapter units. It employs common central processor laboratory test equipment and similar data adapter laboratory test equipment. Based upon the significant level of commonality which exists between the two programs, the following benefits will favorably influence the Apollo back-up program and lead to an over-all increased effectiveness in both NASA programs:-

(a) Common Stock Room:

A common stock room reduces the requirements for space, manpower, and paperwork and will reduce the program impact of the availability of critical components.

(b) Engineering Support to Fabrication:

Since manpower is available on current program, requirements will be drastically reduced.

(c) Basic Design Utilization:

Funds required for design development will be at a minimum.

(d) Single Production Line:

The Apollo back-up program will benefit from experience gained on the present production line in such areas as scrap and rework.

(e) Personnel:

Personnel will have been trained and certified in the production of subassemblies for the current program, thereby eliminating duplication of effort and retraining costs.

(f) Common Factory Tools and Test Equipment:

Tools and test equipment have been designed during the current program.

(g) Certification of Component Parts:

All common components will already have been certified under the current program.

(h) Cost Savings in Parts Procurement:

Increased quantities in each order will reduce the costs of a follow-on program.

(i) Certification of Processes:

Processes will have been certified under the current program.

(j) Set-up Time:

Larger quantity orders will reduce the piece price per component for in-house fabrication.

(k) Reduced Specification and Drawing Preparation:

Common specifications and drawings will not have to be re-generated.

(l) NASA Specifications:

NASA specifications and requirements are already established and in use on the current program. Costs will be greatly reduced in their application to the back-up program.

(m) Emergency Spare Support:

Larger quantities of hardware will enable a very quick response for subassembly spare back-up.

(n) Lower Residual In-Plant Inventory:

Inventory costs on the Apollo back-up program will be reduced through the utilization of inventories on both programs.

(o) Common Software:

Assembler and simulator programs being written under the current contract will be directly applicable to the Apollo back-up program.

(p) Reliability Enhancement:

More machines will have been built before the first flights, so more data will be available to make reliability improvements.

As part of the Saturn V development program, the projected equipment reliability has been carefully analyzed. Reliability engineering effort has been applied to the design at the component, subassembly, and system level. Component field performance data, supplemented by laboratory life test data, have been conservatively extrapolated to represent the new component-environment situation suggested by the Saturn V application. Redundant equipment design has been validated by analysis and simulation work. Similar analyses carried out for the proposed Apollo equipment configuration identify a predicted probability of mission success for the Apollo guidance computer of 0.09953 for the 336-hour mission. A non-redundant computer would require a 46,000-hour mean-time-to-failure to afford an equal probability of mission success. It is significant to note that the analysis has been predicated upon an over-all reduction in failure rates during standby operation of a conservatively-identified factor of two. IBM is confident that the proposed equipment will meet the reliability requirements of the Apollo spacecraft. Results of the Apollo reliability analysis are presented in Section III of this report.

The characteristics of the Apollo guidance computer are summarized below:

Apollo Guidance Computer Characteristics

Type:	Stored program, general purpose, serial fixed point, binary
Clock:	512 kilobits per second, 2.048-megacycle clock
Speed:	Add-subtract and multiply-divide simultaneously:
Add Time:	82 microseconds, 26-bit accuracy
Multiply Time:	328 microseconds, 24-bit accuracy
Divide Time:	656 microseconds, 24-bit accuracy
Storage Capacity (Duplex):	12,288 26-bit words plus two parity bits. Memory can be divided between program and data as desired, typically: 2,000 data words (25 bits and sign) 20,576 instructions (each 13 bits)
Inputs:	35 discrete inputs, UPLINK word (16 bits) three accelerometer pulse rates, three sets of gimbal angle pulses, radar range (15 bits) Tracker angles x and y (13 bits) Real time 26 bits, optic angles x and y (13 bits)
Outputs:	Pulse rates: 1.024 megacycles to 1.5 cps 22 discrete outputs, DOWNLINK word (16 bits) Optic or thrust control pulse rate Six quantities to radar (13-bit register) One pulse rate to CDU or gyros

Cooling:	Circulating coolant (integral cooling) or cold plate
Reliability Estimate:	0.9953 based on mission length of 336 hours with 84% standby operation
Weight:	Central processor and data adapter - 88 pounds Power supply - 30 pounds
Volume:	Central processor and data adapter - 1.9 cu. ft. Power supply - 0.4 cu. ft.
Power:	Central processor and data adapter - 210 watts Power supply - 150 watts Standby power - 43 watts

The redundant central processor and data adapter units are packaged to fit in the command module space now allocated to the Apollo guidance computer. The power supply unit has been packaged to permit its installation in an available adjacent space behind other guidance and navigation equipment. Equipment characteristics are presented in greater detail in Section II of this report.

It is important to recognize the potential impact of the Apollo program upon the basic Saturn V program. Two Saturn V modifications, identified during this study, have already been directed by MSFC:

- (a) Basic clock frequency has been modified from 2.0 to 2.048 megacycles to permit proper interface between the Apollo data adapter and other guidance and navigation subsystems.
- (b) A minor wiring change has been made within the Saturn central processor to permit a standby mode implementation uniquely required for the Apollo application.

Two additional Saturn V modifications need yet be considered:

- (a) Increase clock stability from 25 parts per million to 1 part per million.
- (b) Modify the form factor of the computer memory module for compatibility with the Apollo guidance computer form factor.

Neither of these changes can be conveniently implemented in the first Saturn V computer breadboard without schedule modification, but changes can be broken into a subsequently delivered unit without schedule implication. Although break-in point can best be defined when the back-up Apollo program is implemented, break-in well before system qualification is clearly feasible. The Apollo back-up program proposed in this study implies no Saturn V schedule impact and no significant technical compromises in order to achieve the high degree of commonality identified earlier in this section.

This report identifies two basic technological questions whose resolution should be undertaken only after a full understanding of their significance has been gained by both Apollo and Saturn V program groups at MSC and MSFC. These two basic alternatives, both related to the structural design of the Saturn V and Apollo equipment, are:

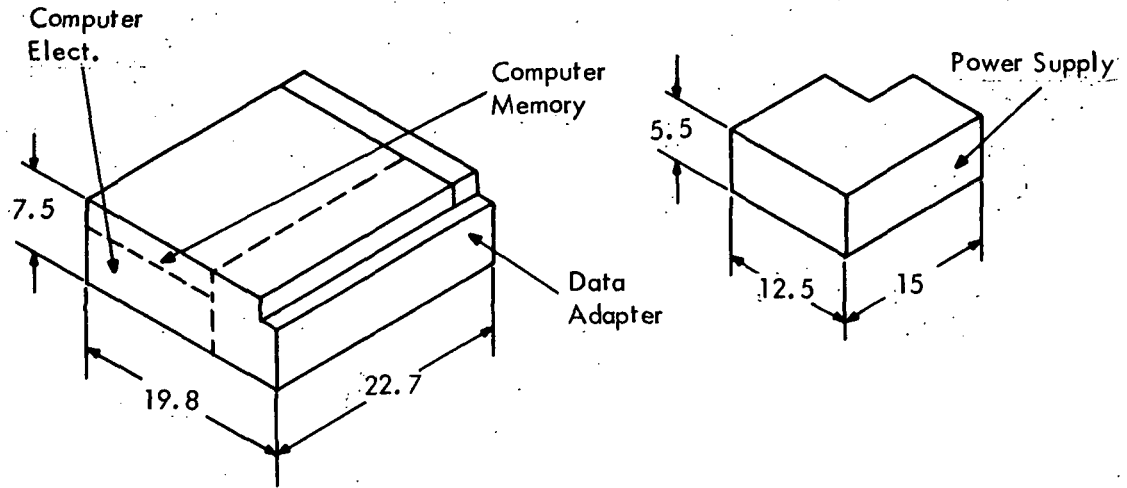
- (a) Whether to use cold plate cooling or integral cooling within the guidance computer equipment.
- (b) Whether to use a common computer structure for both the Apollo and Saturn V applications or a unique structural design for each program.

Either approach to each of these alternatives is technically feasible. Development costs, qualification costs, and production costs are essentially invariant with the alternatives presented. Figures I-1 and I-2 illustrate both the Apollo and Saturn V installations based upon the choices presented in alternative (b). *

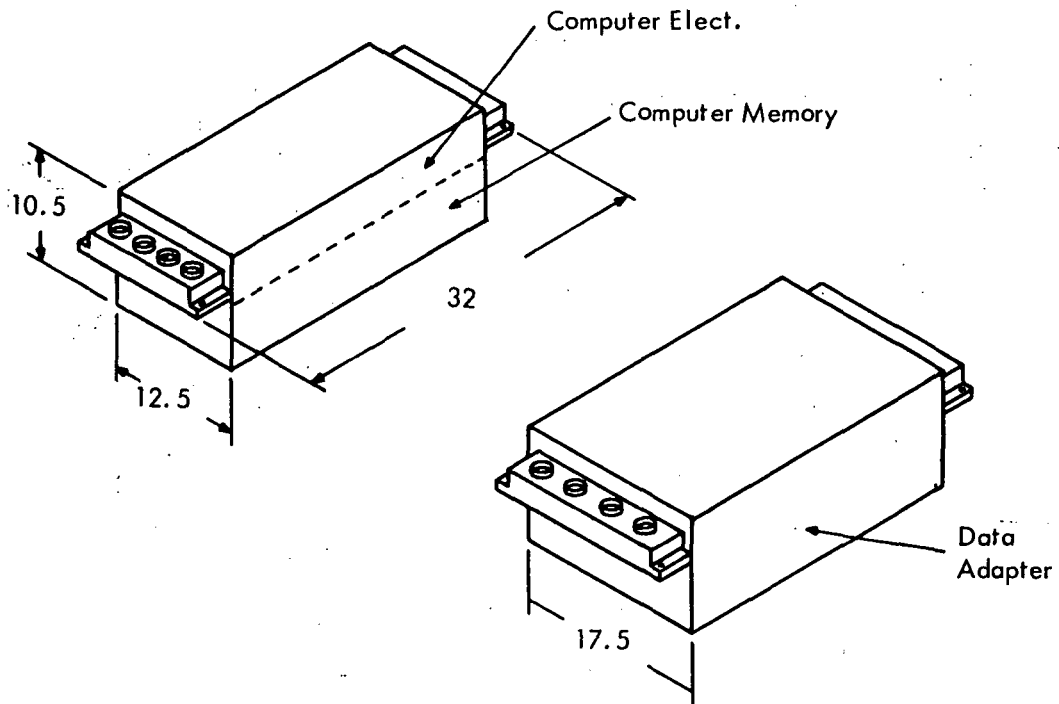
The advantages of the unique-structure approach are:

- (a) Minimum redesign of Saturn V equipment.
- (b) Reduced Apollo electrical and coolant interconnection design problems.
- (c) Connector compatibility with other subsystems in the Apollo guidance and navigation and the Saturn V instrument unit.
- (d) Structures tailored to unique vibration, environmental, and coolant systems.
- (e) Slightly reduced installation weight in Apollo.

* An additional alternative is a combination of Figures I-1(b) and I-2(a).

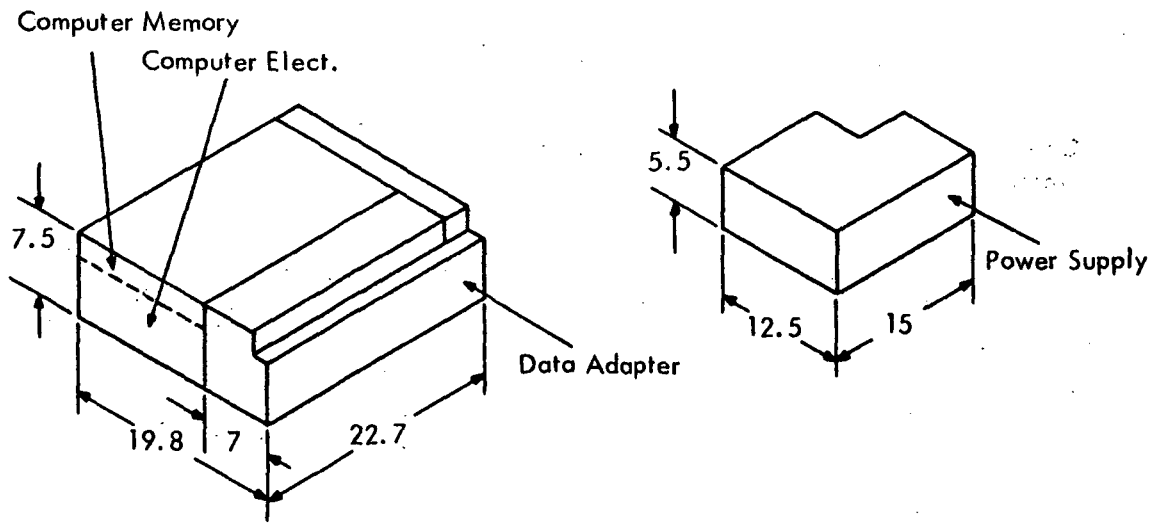


A. APOLLO INSTALLATION

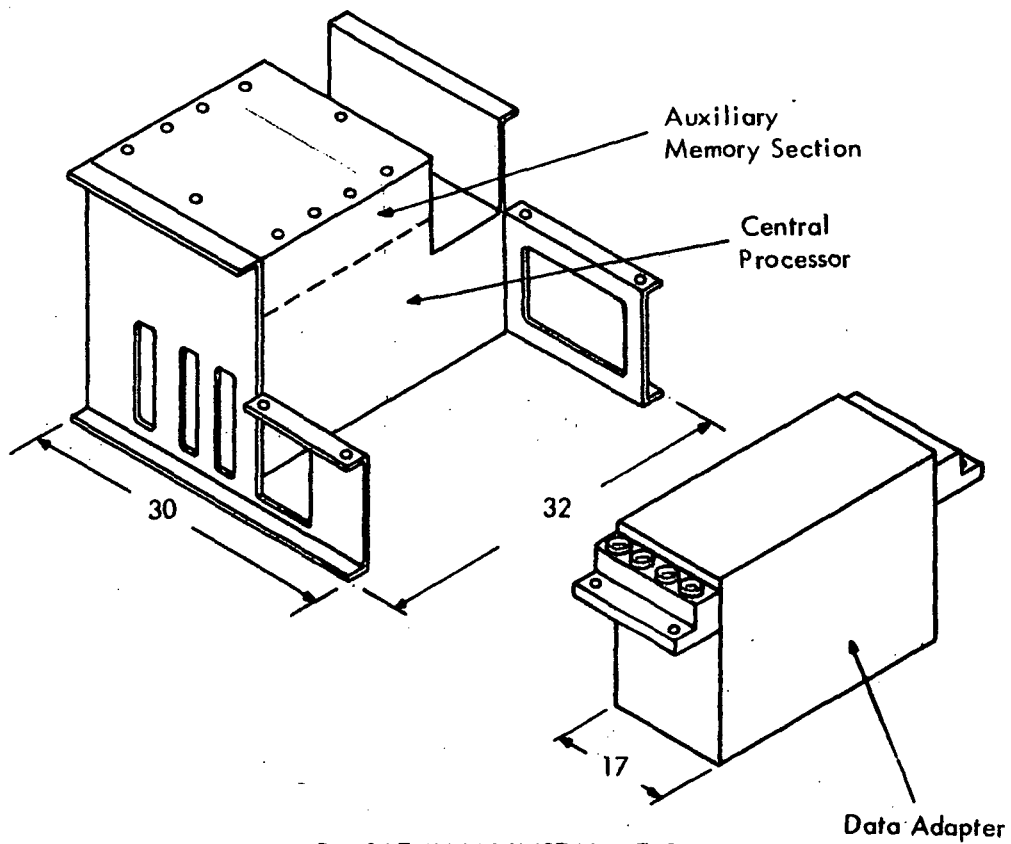


B. SATURN V INSTALLATION

Figure I-1. Unique Structural Approach



A. APOLLO INSTALLATION



B. SATURN V INSTALLATION

Figure I-2. Common Central Processor Structural Approach

The common structure approach provides:

- (a) Potential reduction in Atlantic Missile Range (AMR) systems spares support requirements.
- (b) Common application of Saturn V and Apollo central processor laboratory test equipment at certain installations.
- (c) Improved control on future design changes that might reduce design commonality.

The integral cooling approach provides:

- (a) Minimum redesign of Saturn equipment.
- (b) Reduced component temperature with resultant improved reliability.
- (c) Reduced installed system weight.
- (d) Reduced mounting area requirement in Saturn instrument unit.

The cold plate cooling alternative provides:

- (a) Minimum redesign of Apollo command module.
- (b) Eliminating the Apollo installation constraints of the integral cooling approach.

IBM recommends that all interested parties jointly consider these basic structural design alternatives at an early date in order to derive a conclusion and avoid unnecessary cost and schedule impact in the present development program.

Systems application and programming studies carried out within this study are reported in Section IV. Although it is indicated that the present Apollo computer and proposed back-up computer are different in data processing detailed characteristics, it is established that the machines are basically equivalent in their capability to treat with the overall Apollo guidance problem. Minor logic modifications now under consideration in the Saturn V development program may permit improvement in the machine capability for both the Saturn V and Apollo applications, particularly in the area of sub-routine linkage.

This study has included basic considerations of laboratory test equipment necessary both for factory build and acceptance test support, and also for field system support before integration of the guidance computer into the over-all guidance and navigation system. Equipment requirements for the Apollo program parallel those associated with the present Saturn V program. Utilization of present Saturn V equipment with modification to permit treatment of the unique Apollo input-output characteristics is feasible. The results of the test equipment considerations are presented in Section V.

Based upon preliminary discussions with MSC, a three-phase Apollo back-up program plan has been outlined. This report concludes the Phase I study effort. A 15-month Phase II development program plan is discussed in Section VI of this report. It identifies the fabrication of two production prototype Apollo guidance computers and related laboratory test equipment for delivery during the last quarter of 1964. If an IBM back-up program is to be meaningful to the over-all Apollo effort, immediate implementation of the Phase II program is recommended. The Phase III production program can be developed to permit delivery of flight-quality equipment in 1965. Such an effort would augment delivery of 15 Saturn V flight-quality equipments now scheduled for 1965 delivery. Tooling, test equipment, and long-lead software requisite to responsive equipment delivery in the Phase III production effort has been projected within the Phase II development program.

From the several viewpoints from which the applicability of Saturn V equipment to the Apollo spacecraft program has been considered, it should be concluded that implementation of the Phase II back-up development program is technically feasible and basically attractive from the standpoint of resources commitment.

SECTION II
APOLLO GUIDANCE COMPUTER

SECTION II

APOLLO GUIDANCE COMPUTER

A. GENERAL

The computer that IBM recommends for the Apollo back-up application is very similar to the one presently being designed for the Saturn V program. It is a serial computer which uses a random-access magnetic core memory and microminiature packaging techniques developed under the Advanced Saturn Technology program. Triple modular redundancy in the central processor and multiple duplex memory modules will be used for high reliability. Glass delay lines will also be used for the serial arithmetic register and for the storage of the instruction counter. Table II-1 summarizes the characteristics of the recommended computer.

Figure II-1 shows how the Apollo Guidance Computer interfaces with the rest of the guidance and control system. The computer is composed of three major subassemblies: power supply, central processor, and data adapter. The power supply would be located in a space adjacent to the central processor - data adapter and would supply the d-c voltages required by the computer. Two packaging schemes are being considered for the central processor and data adapter. One scheme would provide for two separate units; the other would combine both subassemblies into one unit. The relative merits of each scheme are considered in Part II-D. The central processor provides the instruction and constant storage as well as the arithmetic processing functions. The data adapter interfaces with the remainder of guidance and navigation equipment. Velocity increments are received from the inertial measuring unit in response to control pulses which are sent from the data adapter. Steering angles are sent to the coupling display unit while attitude angle information is received. The computer generates engine cut-off commands which are sent to the spacecraft. Communication between the computer and the astronauts is accomplished through the display and control unit. This unit is not considered a part of the back-up computer. Angular and range information are sent to the data adapter under the command of control lines which originate in the data adapter.

A special requirement of the Apollo Guidance Computer is operation during a standby mode. This mode, which conserves spacecraft power, requires only the time-keeping capability of the computer. The impact of this requirement is discussed in more detail in Part II-F.

Table II-1

APOLLO GUIDANCE COMPUTER CHARACTERISTICS

Type	Stored program, general purpose, serial fixed point, binary
Clock	512 kilobits per second, 2.048 mc clock
Speed	Add-subtract and multiply-divide simultaneously:
Add Time, Accuracy	82 usec, 26 bits
Multiply Time, Accuracy	328 usec, 24 bits
Divide Time, Accuracy	656 usec, 24 bits
Storage capacity (Duplex)	12,288 26-bit words plus two parity bits. Memory can be divided between program and data as desired, typically: 2,000 data words (25 bits and sign) 20,576 instructions (each 13 bits)
Input	35 discrete inputs UPLINK word (16 bits) 3- accelerometer pulse rates 3- sets of gimbal angle pulses Radar range (15 bits) Tracker angles X and Y (13 bits) Real Time (26 bits) Optic angles X and Y (13 bits)
Outputs	Pulse rates 1.024 mc to 1.5 cps 22 discrete outputs DOWNLINK word (16 bits) Optics or Thurst Control pulse rate 6- Quantities to Radar 13-bit register 1- Pulse rate to CDU or Gyros

Table II-1. Apollo Guidance Computer Characteristics (cont)

Cooling	Circulating coolant (integral cooling or cold plate
Reliability Estimate	0.9953 based on mission length of 336 hrs. with 84 percent standby operation
Weight	Central Processor and data adapter - 88 lbs Power supply - 30 lbs
Volume	Central processor and data adapter - 1.9 cubic feet Power supply - 0.4 cubic feet
Power	Central processor and data adapter - 210 watts Power supply - 150 watts Standby Power - 43 watts

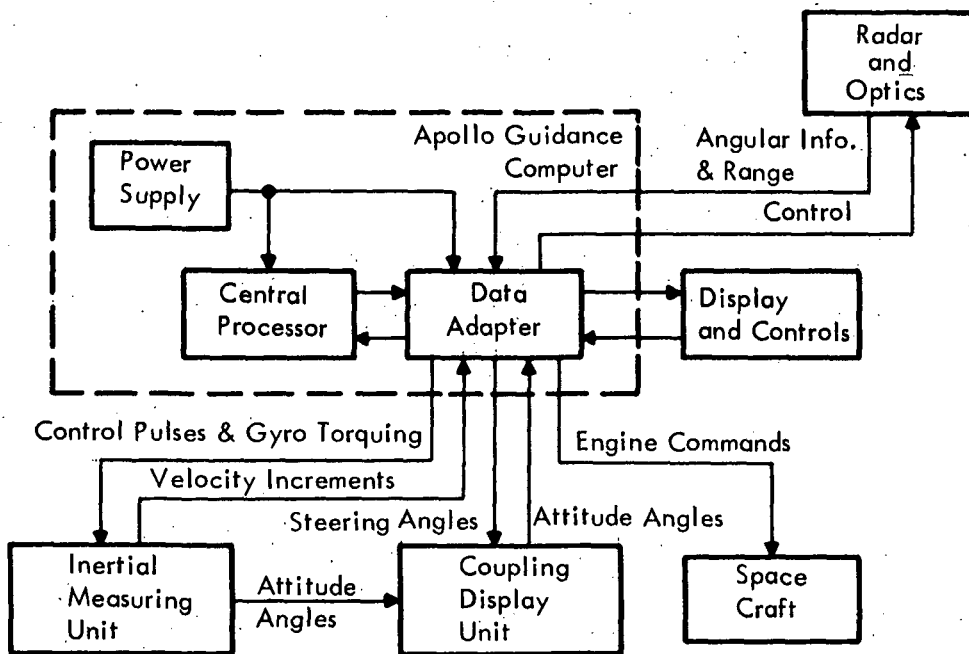


Figure II-1. Apollo Guidance and Control Block Diagram

The back-up Apollo Guidance Computer would provide general purpose computing capability which is characterized by high internal computing speed and a variable-capacity random-access core memory. The internal arithmetic structure would employ both adder and multiplier units which may operate concurrently with a single program control unit. This arrangement provides operating speeds up to 40 percent greater than for a more conventional nonconcurrent organization at essentially no additional cost in components. The programmer would have the option of selecting either a concurrent or nonconcurrent type of multiply. If the number of instructions is to be optimized, then the nonconcurrent multiply instruction would be selected. However, if the cycle time is to be optimized, then the concurrent mode of multiply would be chosen.

Memory words are 28 bits in length (including two parity bits). The memory is arranged so that one data word or two instructions may occupy one 28-bit memory word. The memory element uses a module that consists of an array of fourteen planes, each plane containing 64×128 cores. This memory module contains 4096, 28-bit simplex words and also includes the driving and sensing circuits. The six memory modules estimated for the Apollo mission would provide for 12,288 duplexed, 28-bit words for highly reliable memory operation, or 24,576 simplex words.

Reliability of the central processor would be ensured by the use of triple modular redundancy (TMR). IBM proposes that the central processor be divided into seven modules, which would be triplicated. Redundancy at this level provides reliability superior to the duplex equipment approach and raises fewer design problems than the use of quad components.

Electronic circuits will be mounted on 0.3 in. square wafers on which interconnected wiring and film resistors (cermet) have been deposited by silk-screen printing and subsequent firing operations. These Unit Logic Devices (ULD) will be attached to Multilayer Interconnection Boards (MIB) by use of solder reflow techniques. Each MIB will have a capacity of 35 ULD's. Two MIB's will be bonded back-to-back to a supporting metal frame, and the assembly will comprise a page. Pages are interconnected by back panel multilayer printed circuit boards. The central computer electronics will be packaged on 78 pages. A welded compartmentized structure will house the computer electronics and delay line registers. Memory electronics will be mounted on MIB-type boards where possible. Each memory module will be a self-contained unit with individual timing, control, drive, address, sense, and inhibit circuitry.

The use of TMR permits the subdivision of the central processor into three simplex machines for testing purposes. Significant machine registers are brought out to Laboratory Test Equipment for troubleshooting during ground testing. The maintenance equipment will have the capability of observing register contents by use of panel lights. This equipment will also be able to control the voltage connection of the output inverter and voter circuits in each TMR module. This control will permit the isolation of malfunctions on a simplex level by using test programs and on a fine basis by using module switches, test lights, and maintenance problems.

For purposes of comparison, a description of the Saturn V computer and data adapter have been included as Volume II of the study report.

B. CENTRAL PROCESSOR ORGANIZATION

1. FUNCTIONAL DESCRIPTION

Figure II-2 illustrates the central processor information flow. This simplified block diagram depicts the major data flow paths and associated register level logic. The I/O philosophy is not shown, but is described in Section II-C.

The central processor has been organized as a serial, fixed point, stored program, general purpose machine which processes data using two's complement arithmetic. Two's complement arithmetic obviates the recomplementation cycle required when using sign-plus-magnitude arithmetic. Special algorithms have been developed and implemented for multiplication and division of two's complement numbers. Multiplication is done four bits at a time and division two bits at a time. These algorithms are treated separately in the Arithmetic portion of this section.

A random-access magnetic core memory will be used as the computer storage unit. A serial data rate of 512 kilobits-per-second will be maintained by operating the memory units in a serial-by-byte, parallel-by-bit operating mode. This allows the memory to work with a serial arithmetic unit. The parallel read-write word length of 14 bits includes one parity bit to allow a checking of the memory operations.

Storage external to the memory is located predominantly in the shift register area. The reliability in this area is maximized by using glass delay lines for arithmetic registers and counters. Delay lines are the best choice when transistor count for the various registers is considered.

a. Organization

Each instruction comprises a four-bit operation code and a nine-bit operand address. The nine-bit address allows 512 locations to be directly addressed. The total memory is divided into sectors of 256 words, and contains a residual memory of 256 words. The nine-bit address specifies a location in either the previously selected sector (Data Sector Latches) or in the residual memory. If the operand address bit, R , is a binary 0, then the data will come from the sector specified by the sector register. If R is a 1, the second register is disregarded and the data comes from residual memory. Instructions are addressed from an eight-bit instruction counter which is augmented by a four-bit instruction sector register. Sector memory selection is changed by special instructions which change the contents of the sector register. The sector size is large enough so that this is not a frequent operation.

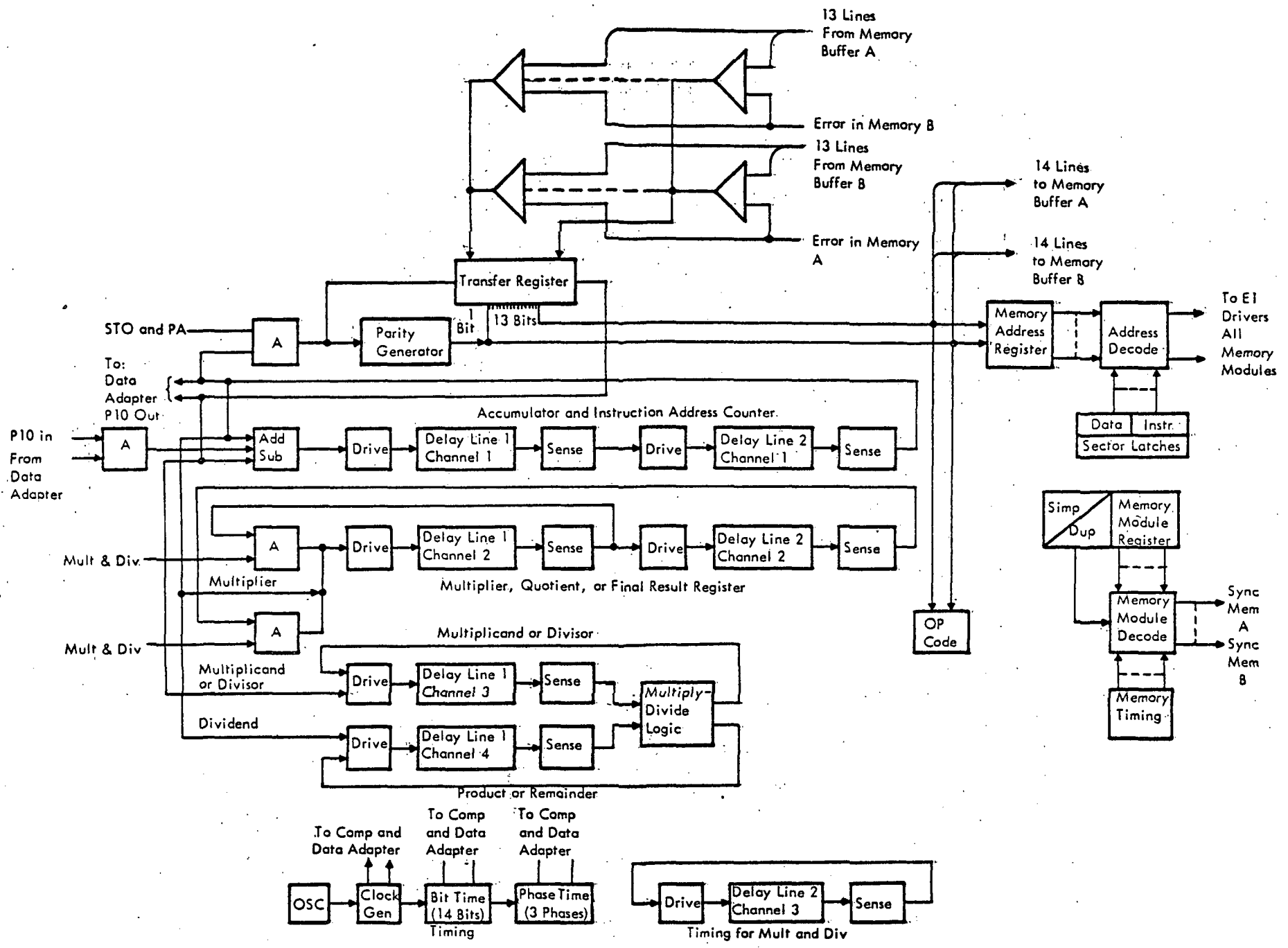


Figure II-2. Central Processor Block Diagram

Data words consist of 26 bits. Instruction words consist of 13 bits and are packed in memory, two instructions per data word. Hence, instructions are described as being stored in syllable one or syllable two of a memory word. Two additional bits are used in the memory to provide parity checking for each of the two syllables. (See Table II-2.)

The computer is programmed by means of single-address instructions. Each instruction specifies an operation and an operand address. Instructions are addressed sequentially from memory under control of the instruction counter; and each time the instruction counter is used, it is incremented by one to develop the address of the next instruction. After the instruction is read from memory and parity checked, the operation code is sent from the transfer register to the OP code register. This is a static register which stores the operation over the duration of the execution cycle.

Table II-2

DATA AND INSTRUCTION WORD FORMAT

Memory Plane	Syllable 2	1	2	-----	13	14
	Syllable 1	15	16	-----	27	28
Data Word	Syllable 2	S	2-1	-----	2-12 _P	
	Syllable 1	2-13	2-14	-----	2-25 _P	
Instruction Word	Syllable 1 or 1 or 2	A8	A7 --- A1	R	OP4	OP3 OP2 OP1 P
S Sign Position					
A8, A7, etc., Operand Address					
R Residual Bit					
OP1, OP2, etc., Operation Codes					
P Parity Bit					

The operand address portion of the instruction is transferred in parallel (9 bits) from the computer's transfer register (TR) to the memory address register. The TR is then cleared. If the operation code is one which requires reading the memory, the contents of the operand address are read 14 bits at a time (including parity) from the memory into the buffer register where a parity check is made. Data bits are then sent in parallel to the TR. This information is then serially transferred to the arithmetic section of the computer. If the operation code is a store (STO) the contents of the accumulator are transferred serially, into the TR and stored in two 14-bit bytes. A parity bit is generated for each byte.

Upon completion of the arithmetic operation, the contents of the instruction counter are transferred serially into the TR. This information is then transferred in parallel (just as the operand address had previously been transferred) into the memory address register. The TR is then cleared and the next instruction is read, thus completing one computer cycle.

The data word is read from the memory address specified by the memory address register and from the sector specified by the sector register. Data from the memory goes directly to the arithmetic section of the computer where it is operated on as directed by the OP code. The arithmetic section contains an add-subtract element, a multiply-divide element, and storage register for the operands. Registers are required for the accumulator, product, quotient, multiplicand, multiplier, remainder and divisor. The add-subtract and the multiply-divide elements operate independently of each other. Therefore, they can be programmed to operate concurrently if desired; i. e., the add-subtract element can do several short operations while the multiply-divide element is in operation. No dividend register is shown in Figure II-2 because it is considered to be the first remainder. As indicated, both multiply and divide require more time for execution than the rest of the computer operations. A special counter is implemented to keep track of the multiple-divide progress and stops the operation when completed. The product-quotient (PQ) register has been assigned an address and is addressable from the operand address of any instruction. The answer will remain in the PQ register until multiply-divide is initiated.

b. Timing

The three levels of computer timing are illustrated in Figure II-3. Basically, the computer is organized around a four-clock system. The width of each clock is approximately 0.4 usec and the pulse repetition frequency is 512 kilocycles. The bit time (four clock pulses) is $1/512$ usec. Fourteen bit times occur in one phase time, resulting in a phase time of 27.34375 usec. Three phase times, P_A , P_B , and P_C are required to perform a complete computer operation cycle. Phase A (P_A) makes up the instruction cycle and phases B and C (P_B and P_C) determine the data cycle.

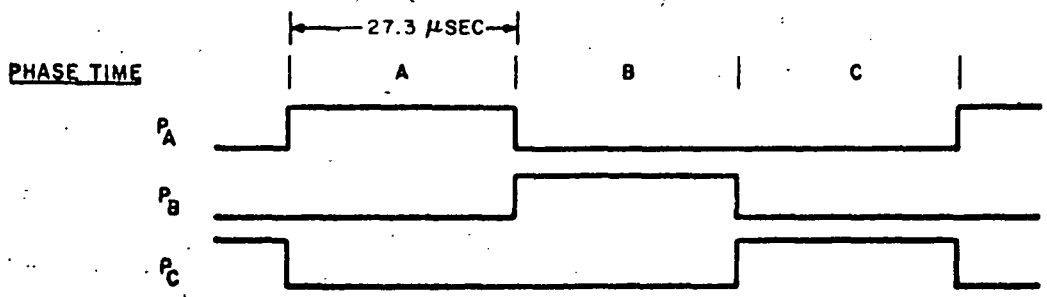
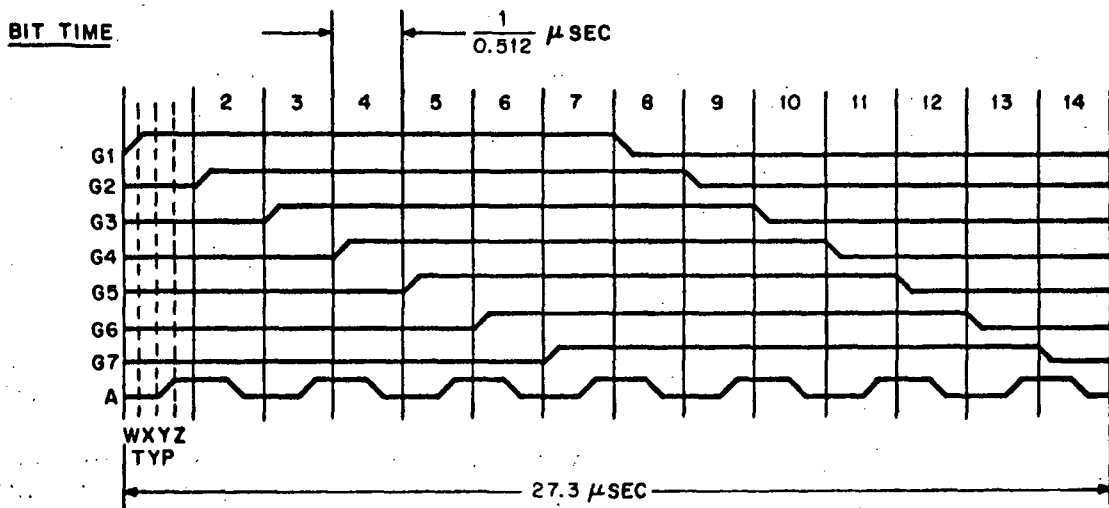
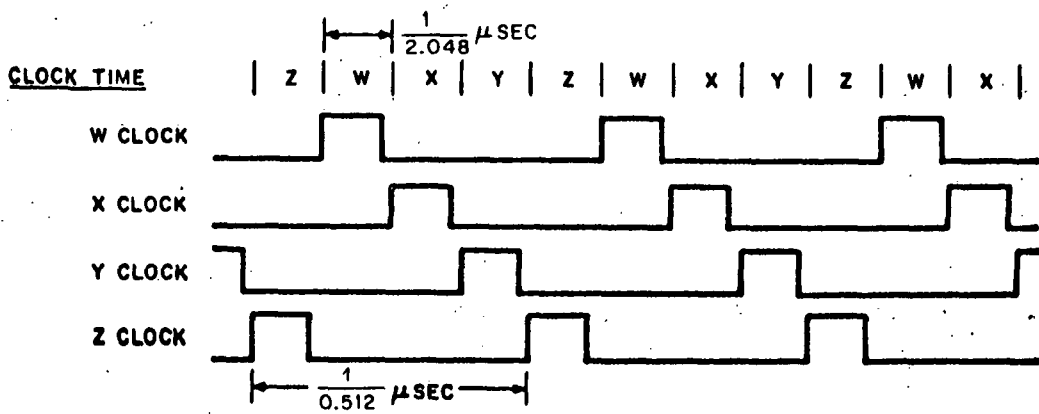


Figure II-3. Computer Timing

2. COMPUTER CONTROL

a. Instruction List

The instruction bit assignment for the operation code is shown in Table II-3.

Table II-3
OPERATION CODE MAP

OP1	OP2				OP3
	MPY		STO	DIV	
	MPH	XOR	CLA	ADD	
	TNZ	TMI	SHF	AND	
	HOP	TRA	PIO	SUB	
	OP4				

HOP
(82 usec)

The contents of the memory address designated by the operand address specify the next instruction address and data sector. Four bits identify the next instruction sector, eight bits are transferred to the instruction address counter, one bit conditions the syllable control, four bits identify the next data sector, three bits identify the next memory module, one bit defines either a simplex or duplex memory operation, and one bit resets the memory error latch when specifying a new memory module.

TRA
(82 usec)

The eight-bit operand address is transferred to the instruction counter. The residual bit in the operand address is used to specify the instruction syllable latch. The sector register remains unchanged.

TMI
(82 usec)

A transfer occurs on the minus accumulator sign. If the sign is positive, the next instruction in sequence is chosen (no branch); if the sign is negative (zero is considered positive), the eight-bit operand address becomes the next instruction address (perform branch), and a TRA operation is executed.

TNZ
(82 usec) A transfer occurs when the accumulator contains a nonzero number. If the accumulator is zero, the next instruction in sequence is chosen; if the accumulator is not zero (either negative or positive), the eight-bit operand address becomes the next instruction address, and a TRA operation is executed.

SHF
(82 usec) The SHF instruction shifts the accumulator contents right or left one or two places as specified by the operand address.

A1 Right Shift 1 A5 Left Shift 1

A2 Right Shift 2 A6 Left Shift 2

AND
(82 usec) The contents of the memory location specified by the operand address are logically AND'ed, bit-by-bit, with the accumulator contents. The result is retained in the accumulator.

CLA
(82 usec) The contents of the location specified by the operand address are transferred to the accumulator.

ADD
(82 usec) The contents of the location specified by the operand address are added to the accumulator contents. The result is retained in the accumulator.

SUB
(82 usec) The contents of the location specified by the operand address are subtracted from the accumulator contents. The result is retained in the accumulator.

STO
(82 usec) The contents of the accumulator are stored in the location specified by the operand address. The contents of the accumulator are retained.

DIV
(656 usec) The contents of the accumulator are divided by the contents of the memory location specified by the operand address. The 24-bit quotient is contained in the product-quotient delay line. Concurrent use of the adder-subtractor element is permitted.

MPY
(328 usec) The contents of the memory location specified by the operand address are multiplied by the accumulator contents. The 24 high-order bits of the multiplier and multiplicand are multiplied together to form a 24-bit product. Concurrent use of the adder-subtractor element is permitted. The product is stored in the product-quotient delay line.

MPH (410 usec)	This is the multiply and hold operation. It is the same as the MPY operation except concurrent use of the adder-subtractor element is not permitted and the product is stored in the accumulator.
XOR (82 usec)	The contents of the memory location specified by the operand address are exclusively OR'd, bit-by-bit, with the contents of the accumulator. The result is retained in the accumulator.
PIO (82 usec)	The low-order address bit, A1, determines whether the operation is an input or output instruction. If A1 = 1, the contents of the input register specified by the operand address are transferred to the accumulator. If A1 = 0, the contents of the memory or accumulator specified by the operand address are transferred to the output register.

b. Multiply and Divide Timing

All operations, except MPY, MPH, and DIV require one operational cycle (82 usec) for execution. Other instructions must be executed concurrently with the MPY and DIV instructions (except MPH). Three instructions can be executed between the initiation of the MPY and the time when the product is available; similarly, seven instructions can be executed between the initiation and the termination of DIV. More one-word-time instructions can be inserted before the product or quotient is addressed if maximum efficiency is not required, since multiplication or division is stopped automatically and the result retained until addressed. Figure II-4 illustrates the timing of the MPY and DIV operations. The MPH instruction cannot operate concurrently with other operations because it inhibits further memory accesses until completed.

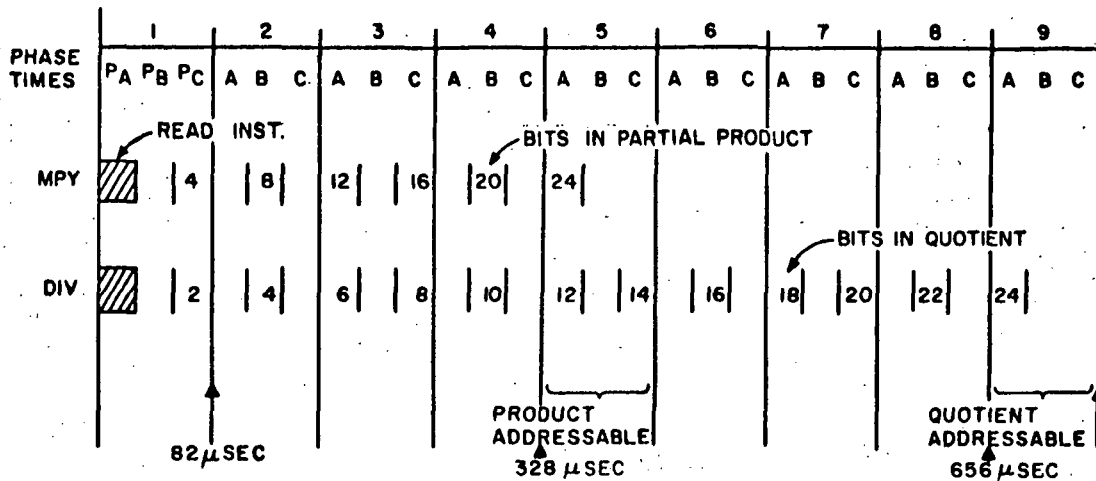


Figure II-4. MPY-DIV Timing Chart

c. Interrupt

A program interrupt feature is provided to aid the input/output processing. An external signal interrupts the computer program and causes a transfer to a subprogram. Interrupt occurs when the instruction in progress is completed. The instruction counter, sector and module registers, and syllable latch are stored automatically in a reserved residual memory location (octal address 777). A HOP constant which designates the start of the subprogram is subsequently retrieved from a second reserved residual memory location (octal address 776). Automatic storage of the accumulator and product-quotient registers is not provided; this must be accomplished by the subprogram. Multiple interrupt protection or interrupts during MPY and DIV operations is provided.

The interrupt signal may be generated by a timed source. The rate at which it is generated will be program controlled by changing the magnitude of a number which is being continually summed. When the summed number reaches a predetermined value, the interrupt signal is generated. The main program can be resumed by addressing the contents of residual memory word 777 with a HOP instruction.

Other external signals such as discrete inputs will also be allowed to cause interrupt. These signals will be useful in causing the I/O subprogram to give immediate attention to an input or output event.

3. PROGRAMMING CONSIDERATIONS

The back-up Apollo Guidance Computer uses a conventional repertoire of arithmetic instructions including add, subtract, multiply, and divide; two multiply instructions are included. MPY requires one-word-time operations in the adder unit during the multiplication process because the instruction counter advances once each word time. This procedure is useful in speeding up the computer operation by permitting simultaneous multiplication and one-word operations. Trial programming has shown a speed increase for a similar configuration of up to 40 percent over a conventional sequential organization. When the program is multiply-limited and a sufficient number of useful one-word operations cannot be located in the portion of the flow diagram being executed, the MPH instruction is used. This instruction inhibits the advance of the instruction counter so no new instructions are read from memory until the MPH operation is completed. This feature conserves program steps relative to an organization which does not contain a MPH-type operation code. Both types of multiply orders permit the increased speed of a concurrent operation without sacrifice in the number of program steps required, and permit a programming trade-off of speed and instructions required.

TRA, TMI, and TNZ instructions provide flexibility in programming unconditional transfers; in branch instructions, through transfer on the contents of the accumulator; and in easy handling of discrete inputs, which are obtained in the accumulator through masking with an AND instruction.

The HOP instruction is used for transfers outside of the sector currently being used. HOP permits jumping to another portion of the flow diagram and to subroutines. To return from a subroutine, the last instruction in the routine is a HOP. The HOP constant causes a return to the original program sequence. Since each use of a subroutine in the program results in return to a different place in the flow diagram, the HOP constant is loaded prior to entering the routine. An automatic program compiler could be used to generate the correct HOP constants.

An exclusive OR operation, XOR, permits the rapid checking of changes in discrete inputs which are grouped into data word inputs. Discrete output words may be generated by masking out the bit to be changed with an AND instruction and by adding the discrete output into the selected position. The product-quotient (P-Q) register can be addressed (by Octal 775) with the operation CLA, ADD, SUB, STO, AND and XOR.

An interrupt feature is provided in the guidance computer to facilitate the timing of input-output operations by causing a transfer to an input-output sub-program. The interrupt signal may be set to interrupt at the highest rate at which any I/O quantity must be handled. The timed interrupt thereby avoids the necessity of keeping track of the time expired since last entering the I/O subprogram. Otherwise many instructions would be required in the various branches of variable length in the flow diagram. An automatic interrupt is also provided to permit certain discrete inputs to cause interrupts. While all applications for this feature have not yet been defined, allowing discretely to interrupt can be used to demand that the program give attention to an important discrete. Communications between the guidance computer and the vehicle telemetry monitoring system can thus be facilitated. The monitor system may be selected by an address code from the computer, and the vehicle parameter to be monitored can be defined over the output line to the DA and stored in a buffer register. When the monitor has acquired the desired parameter, an interrupt can be given, causing the computer I/O sub-program to read the value as an input. This scheme will permit computing to continue while waiting for the monitor system to acquire the parameter.

The sector register permits considerable flexibility in handling data and constants. The instructions indicate whether data is located in the residual sector or the sector referred to by the data sector register. By confining data to the residual register and a limited number of other memory sectors, the changing of the data sector register can be minimized. In this manner, the residual sector is reduced in size and made more readily usable for data which is referred to by instructions stored in many sectors. The small size of each sector, achieved by concentrating instructions rather than both data and instructions in each sector, reduces the size of the instruction word and conserves memory core planes. The programmer is free to move between disjointed parts of the program without frequently changing either instruction or data sector registers. The data sector register is also useful in addressing sets of constants which are stored for use with polynomial injection guidance equations. The instructions necessary to compute the polynomials are stored once while the sets of coefficients for the many different polynomials are each stored in different memory sectors. These coefficients can be readily accessed by use of the data address register, which is set to select a given set of coefficients to evaluate the polynomial. Thus, the polynomial number is set in the sector register and the coefficients are selected.

The separate instructions, and data sector register feature, eliminates the need for indexing since it accomplishes the same end in polynomial evaluation (the chief application of indexing). Hardware and instruction bits are both saved by omitting indexing.

IBM plans to store upper and lower limits for orbital check-out parameters in the two halves of a data word. The monitoring system will relate the address of the parameter to the storage location in memory. A simple, regular sequence of addresses will make programming easy by use of address modification techniques.

4. ARITHMETIC ELEMENTS

The back-up Apollo Guidance Computer has two independent arithmetic elements; the add-subtract element and the multiply-divide element. Although both operate independently, they are serviced by the same program control circuits and may be operated concurrently. Each program cycle time, the add-subtract element can perform any one of the computer instructions, except MPY, MPH, and DIV. During each program cycle time, the results of the simple arithmetic operations are circulated through the accumulator delay line and through the accumulator sync delay line channel to prevent their precessing.

The multiply-divide element uses two channels of a delay line as previously shown in Figure II-2. One channel of the instruction counter delay line is used as a counter to stop the multiply or divide operations. Another channel of the instruction counter delay line is used to sync the product or quotient when the operation is completed. This is controlled automatically by the counter. The product-quotient register is addressable as a residual memory word and has the octal address of 775. The product or quotient can be obtained on any subsequent operation cycle after completion of the multiply or divide, but must be used before initiation of another multiply or divide. The product of the MPH operation is stored in the accumulator.

The recursion formulas for implementing multiply and divide with two's complement numbers are explained in the following paragraphs.

a. Multiply

The multiply element operates in a two-phase cycle, serial-by-four parallel, and requires 15 phase times, including instruction access time. The program initiates a multiply by placing the 24 high-order bits, contained in the memory location specified by the operand address, in the multiplicand delay line. The multiplier delay line contains the 24 high-order bits of the contents of the accumulator. The phase counter terminates a multiply at the proper time following the original MPY or MPH instruction.

The instrumentation of the multiply algorithm requires three delay line channels. Two of the channels contain the partial product and the multiplier.

These channels shift both the partial product and the multiplier four places to the right every two-phase cycle. The third channel contains the multiplicand. The accumulator portion (fourth channel) of this delay line is not involved in the multiply element and can be used concurrently with multiply.

Upon initiation of a multiply, and during every phase time thereafter, the five low-order bits of the multiplier (MR₁ through MR₅) are used to condition latches or thratches. These latches or thratches in turn initiate addition or subtraction of multiples of the multiplicand, to the partial product.

The following algorithm is used for multiply:

$$P_i = 1/16 [P_{(i-1)} + \Delta 1 + \Delta 2]$$

where P_i is the new partial product, and Δ1 and Δ2 are formed according to the following rules:

MR ₁	MR ₂	MR ₃	Δ1	
MR ₃	MR ₄	MR ₅		Δ2
0	0	0	0	0
1	0	0	+2M	+8M
0	1	0	+2M	+8M
1	1	0	+4M	+16M
0	0	1	-4M	-16M
1	0	1	-2M	-8M
0	1	1	-2M	-8M
1	1	1	0	0

M represents the multiplicand. For the first multiplication cycle, P_(i-1) and MR₁ are made zeros.

b. Divide

The divide element operates in a two-phase cycle, serial-by-two-parallel, and requires 27 phase times per divide, including instruction access time. The program initiates a divide by transferring the 26 bits of the addressed memory location (divisor) and the 26 bits of the accumulator (dividend) to the divide element. The phase counter terminates a divide at the proper time following the original divide instruction.

The following algorithm is instrumented as follows to execute divide:

$$Q_i = R_{iS} \cdot DV_S + \overline{R_{iS}} \cdot \overline{DV_S} \quad (1)$$

and

$$R_{i+1} = 2R_i + (1 - 2Q_i) DV \quad (2)$$

where:

$$i = 1, 2, 3, \dots, 24$$

$$Q_i = \text{The } i^{\text{th}} \text{ quotient bit}$$

$$R_{iS} = \text{The sign of the } i^{\text{th}} \text{ remainder}$$

$$DV_S = \text{The sign of the divisor}$$

$$R_i = \text{The } i^{\text{th}} \text{ remainder}$$

$$R_1 = \text{The dividend}$$

$$DV = \text{The divisor}$$

Equation (1) states that the i^{th} quotient bit is equal to a "1" if the sign of the i^{th} remainder is identical to the sign of the divisor. The high-order quotient bit (sign bit) is the only exception to this rule. Q_1 , as determined by equation (1), is used to solve equation (2) but must be complemented before it is stored as the sign bit of the quotient.

The instrumentation of the divide algorithm requires three channels of a delay line. One channel contains the quotient, one the divisor, and one the dividend. These three channels are used during multiply to contain the multiplier, the multiplicand, and the partial product, respectively. The quotient and the remainder channels of the delay line have been lengthened by latches to shift two places to the left each two-phase cycle. The divisor circulates once each two-phase cycle.

In the two's-complement number system, the high-order bit determines the sign of the number. Since this is the last bit read from memory, it is impossible to solve equations (1) or (2) until the entire divisor has been read from memory. However, equations (1) and (2) can have only two possible solutions:

Either

$$Q_i = 1$$

and,

$$R_{(i+1)} = 2R_i - DV$$

or,

$$Q_i = 0$$

and,

$$R_{(i+1)} = 2R_i + DV$$

Both the borrow of $2R_i - DV$ and the carry of $2R_i + DV$ are generated as the dividend and divisor registers are loaded. When the sign bits of these quantities are finally entered into their respective registers, equation (1) is solved for the first quotient bit. If this quotient bit is a one, the borrow is examined to determine the second quotient bit. If the first quotient bit is a zero, the carry is examined to determine the second quotient bit. The following truth table is solved to determine the second quotient bit. If the first quotient bit is a one:

R_1	DV_s	B	$R_{(i+1)_s}$	Q
0	0	0	0	1
0	0	1	1	0
0	1	0	1	1
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

Where,

R_1 = The first remainder bit to the right of the sign bit

DV_s = The divisor sign

B = The borrow into the R_1, DV_s , position

$R_{(i+1)_s}$ = The sign of the new remainder

Q = The quotient bit as determined by comparing DV_s with $R_{(i+1)_s}$ according to equation (2).

$$\begin{aligned} Q &= \bar{R}_1 \bar{D}V_s \cdot \bar{B} + \bar{R}_1 DV_s \bar{B} + R_1 \cdot \bar{D}V_s \cdot B + R_1 \cdot DV_s \cdot B \\ &= \bar{R}_1 \cdot \bar{B} (\bar{D}V_s + DV_s) + R_1 \cdot B (\bar{D}V_s + DV_s) \\ &= \bar{R}_1 \cdot \bar{B} + R_1 \cdot B \end{aligned}$$

The equation used in generating the new remainder , R_{i+2} , is obtained by expanding equation (2).

$$R_{(i+2)} = 2 R_{(i+1)} + (1 - 2 Q_{(i+1)}) DV$$

$$R_{(i+2)} = 2 \left[2R_i + (1 - 2 Q_i) DV \right] + (1 - 2 Q_{(i+1)}) DV$$

$$R_{(i+2)} = 4 R_i + 2 (1 - 2 Q_i) DV + (1 - 2 Q_{(i+1)}) DV$$

As $R_{(i+2)}$ is being generated the next iteration of divide is started by generating, as already described, the borrow and carry for $2 R_{(i+2)} \pm DV$.

5. MEMORY

The memory for the back-up Apollo Guidance Computer uses conventional toroidal cores in a unique self-correcting duplex system. The memory unit consists of six identical 4k-memory modules which may be operated in simplex for increased storage capability or in duplex pairs for high reliability. The basic computer program can be loaded into the instruction and constants sectors of the memory, at electronic speeds, on the ground or just prior to launch. Thereafter, the information content of constants and data can be electrically altered but only under control of the computer program.

The proposed self-correcting duplex system uses an odd parity bit with detection schemes for malfunction indication and correction. In conjunction with this scheme, error-detection circuitry is also used for memory drive current monitoring. Unlike conventional toroid random-access memories, the self-correcting extension of the basic duplex approach permits regeneration of correct information after transients or intermittent failures. Otherwise destructive read-out of the memory could result.

a. Basic Memory System Operation

Figure II-5 shows a simplified block diagram of the computer memory system. The basic configuration consists of a pair of memories providing storage for 8192 14-bit memory words for duplex operation, or 16,384 14-bit memory words for simplex operation. Each of the simplex memories includes independent peripheral instrumentation consisting of timing, control, address drivers, inhibit drivers, sense amplifiers, error detection circuitry and I/O connections to facilitate failure isolation. Computer functions which are common to these simplex units consist of the following:

- Memory address register outputs
- Memory transfer register input-output
- Store gate command
- Read gate command
- Syllable control gates

The computer functions, which are separate for each simplex memory, consist of synchronizing gates which provide the serial data rate of 512 kilobits per second. This data rate is required by the computer to generate a start memory unit command at 125 kilobits per second. These gates also provide the selection of multiple simplex memory units for storage flexibility and permit partial or total duplex operation throughout the mission profile to extend the mean-time-before-failure for long mission times. Each of the simplex units can operate independently of the others or in a duplex manner. The memory modules are divided into two groups; one group consisting of even numbered modules (0-6), and the other consisting of odd numbered modules (1-7). There is a buffer register associated with each group, which is set by the selected modules.

For duplex operation, as shown in Figure II-5, each memory is under control of independent buffer registers when both memories are operating without failure. Both memories are simultaneously read and updated, 14 bits in parallel. A single cycle is required for reading instructions (13 bits plus 1 parity bit per instruction word). Two memory cycles are required for reading and updating data (26 bit plus 2 parity bits). The parallel outputs of the

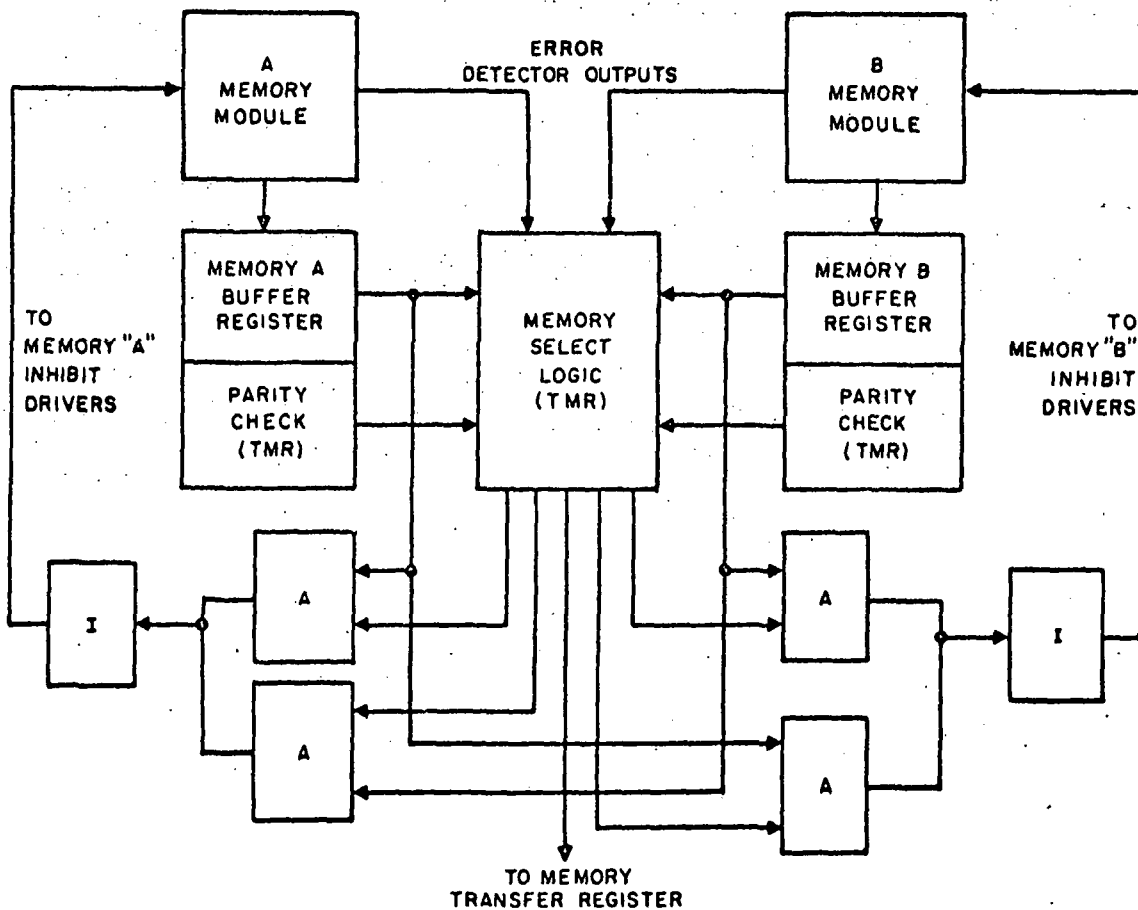


Figure II-5. Self-Correcting Duplex-Toroid Memory System

memory buffer registers are serialized at a 512-kilobit rate by the memory transfer register under control of the memory select logic. Initially, only one buffer register output is used but both buffer register outputs are simultaneously parity checked in parallel. When an error is detected in the memory being used, operation immediately transfers to the other memory. Both memories are then regenerated by the buffer register of the "good" memory, thus correcting transient errors. After the parity-checking and error-detection circuits have verified that the erroneous memory has been corrected, operation returns to the condition where each memory is under control of its own buffer register. Operation is not transferred to the previously errored memory until the "good" memory develops its first error. Consequently, instantaneous switching from one memory output to another permits uninterrupted computer operation until simultaneous failures at the same location in both memories cause complete system failure.

Proper operation of the memory system during read cycles is indicated by each 14-bit word containing an odd number of bits and a logical "one" output of the error detecting circuitry. If either or both of these conditions are violated, operation is transferred to the other memory. During regenerate or store cycles, since parity checking cannot be performed, failure detection is accomplished by the error-detection circuitry only and by parity detection during subsequent read cycles. Intermittent addressing of memory between normal cycles is also detected by the error-detecting circuitry producing a logical "one" output at the improper time. Figure II-6 indicates the system connection of the error-detector circuits for a simplex memory.

The control latch circuits are packaged with the buffer register circuit in the computer. The output latch is in a logical "zero" state for normal operation. If the error-detector output is a logical "zero" at normal cycle times, or a logical "one" at the improper time, the output latch is set to the "one" state indicating an error. Conditions resulting in an error output are as follows:

- Address without voltage source
- Address without current sync.
- No address
- Dual source-single sync address
- Single source-dual sync address

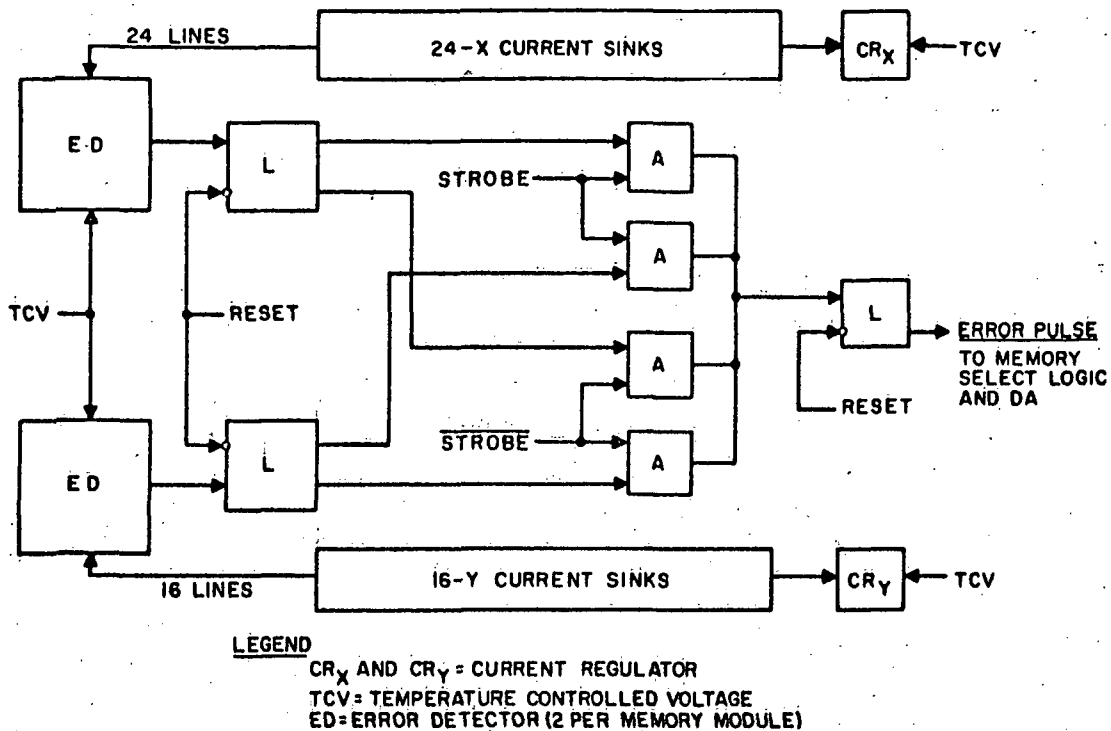


Figure II-6. X-Y Coordinate Half-Select Current Error Detection

b. Out of Time Addressing

Figure II-7 is a more detailed block diagram of a single simplex memory unit. The blocks contained within dashed lines are located in the central computer. All other functional blocks are an integral part of each simplex memory unit. Each memory consists of a 14-bit core array, which represents the two memory syllables. Instructions occupy one syllable and data occupies two syllables. The 4,096 words in each syllable are divided (X and Y) into 16 sectors of 256 words each. Sector 16 is called residual memory. The proper choice of any of the sectors in either simplex unit is determined by the state of the data sector register or instruction sector register. Variables and most constants not associated with the polynomials or orbital check-out will be grouped together and stored in residual or data sectors.

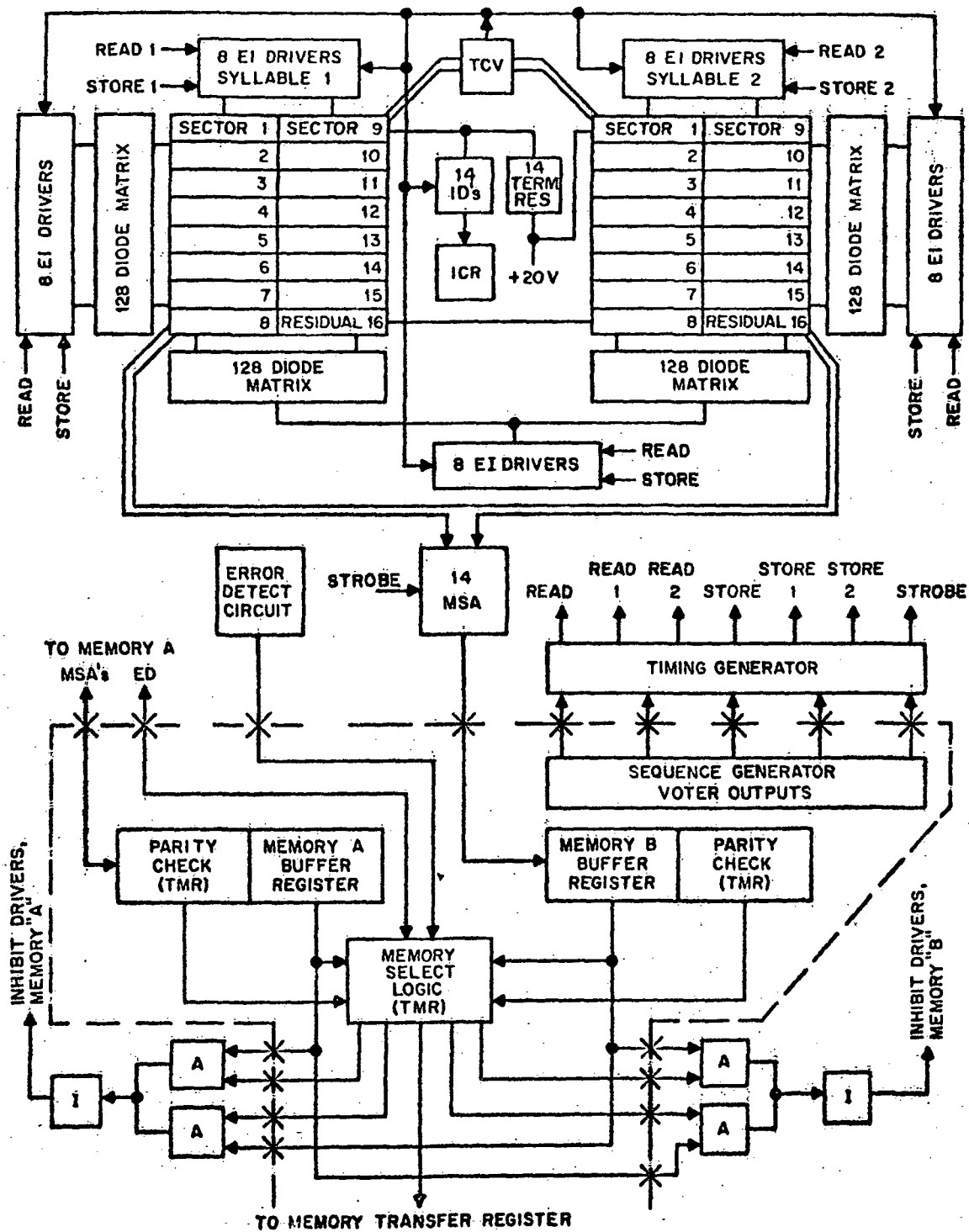


Figure II-7. Simplex Memory Module

(1) Instrumentation

Each memory array is addressed with direct-drive coincident current as shown in Figure II-7. The appropriate X-Y coordinate memory address driver (EI) source and sink gates, and the associated diode decoupling matrices, are selected. Then the Y-coordinate address simultaneously drives both syllables with an independent X-coordinate address. Applying the Y-coordinate address before one of the X-coordinate drives occur causes the delta noise in the nonselected syllable to decay before the selected syllable is read. Thus, the two sense lines of the two syllables can be connected in series. In this manner, the memories are sensed as if they were one 64 by 128 array. Therefore, only 14 memory sense amplifiers are needed instead of 28. An instruction word (13 bits plus parity) is read by addressing one of two syllables, whereas a data word (26-bit plus 2 parity bits) requires addressing of syllables 1 and 2 in sequence.

During the "store" mode, the memory buffer register controls the inhibit drivers. If "0's" are to be retained, the Y-coordinate half-select current is cancelled by inhibit current. As illustrated, each inhibit driver simultaneously inhibits both syllables through a series connection of inhibit lines for the same bit location in each syllable. As a result, each inhibit driver does the work of two.

The coordinate selection current drivers comprise a voltage source, E, and a current sink, I. Each driver is capable of either delivering or accepting current, and functions in conjunction with a like circuit in the opposite ordered group. The instrumentation of a single syllable requires only eight drivers for each order, operated in pairs, to selectively interrogate each of the 4096 word locations. During the application of the Read clock followed by Read 1 or Read 2 clocks, current is delivered from a high-order to a low-order driver. During the application of Store 1 or Store 2 clocks followed by the Store clock, current is delivered from a low-order to a high-order driver. The diode matrix in series with the address driver prevents sneak currents from passing through other nonselected address conductors, and minimizes the effects of displacement currents on current rise time caused by interwiring capacitance.

(2) Temperature Characteristics

The maximum array temperature is limited to 70°C, with a maximum temperature differential across the array of 10°C. These requirements dictate the thermal design of the memory module support structure and the location of the memory address electronics.

C. DATA ADAPTER ORGANIZATION

1. GENERAL

The Apollo Data Adapter (ADA) consists of the input and output circuitry and logic necessary to interface the central processor with the rest of the guidance and navigation equipment. In addition to this hardware, two of the six memory modules are located herein, the remaining four being located in the central processor. Development effort for the ADA will be minimized since the electronic packaging and all circuitry except that servicing the input and output transformers will have been developed under the Saturn V contract.

2. TIMING

The ADA receives timing signals (clocks, bits, and phases) from the central processor (CP). The clocks which occur at a 512-kc rate drive a counter having a 100-cps output as its lowest frequency. This 100-cps signal in turn drives a 26-bit binary counter which is instrumented as two 13-bit words in Channel 3 of the four-channel delay lines.

A countdown of two is performed on the phase times from the computer producing six phase times which are used to time the operation of the delay lines in the ADA.

Delay line registers similar to those used in the Saturn V data adapter provide much of the storage required in the ADA. To prevent the information in the ADA from precessing with respect to the information in the CP, the length of the delay line for the ADA was chosen to be six phases, thus providing 24 13-bit registers. Sixteen of these registers are used to process inputs, five are used to process outputs, and three registers are spares. An odd-even cycle time is generated to provide six phases to time the delay lines.

Since the CP can transmit or receive information to or from the ADA only during phase B or C, the 6-phase ADA delay line loop is made up of one 4-phase delay line and one 2-phase delay line. With proper addressing, any register is available to the CP during a phase B or C. This is illustrated in Table II-4.

Table II-4

DL REGISTER AVAILABILITY

Entering DL No. 1 at Phase	Available at Output of	
	DL No. 1	DL No. 2
A odd	B even	—
B odd	C even	or B odd
C odd	—	C odd
A even	B odd	
B even	C odd	or B even
C even	—	C even

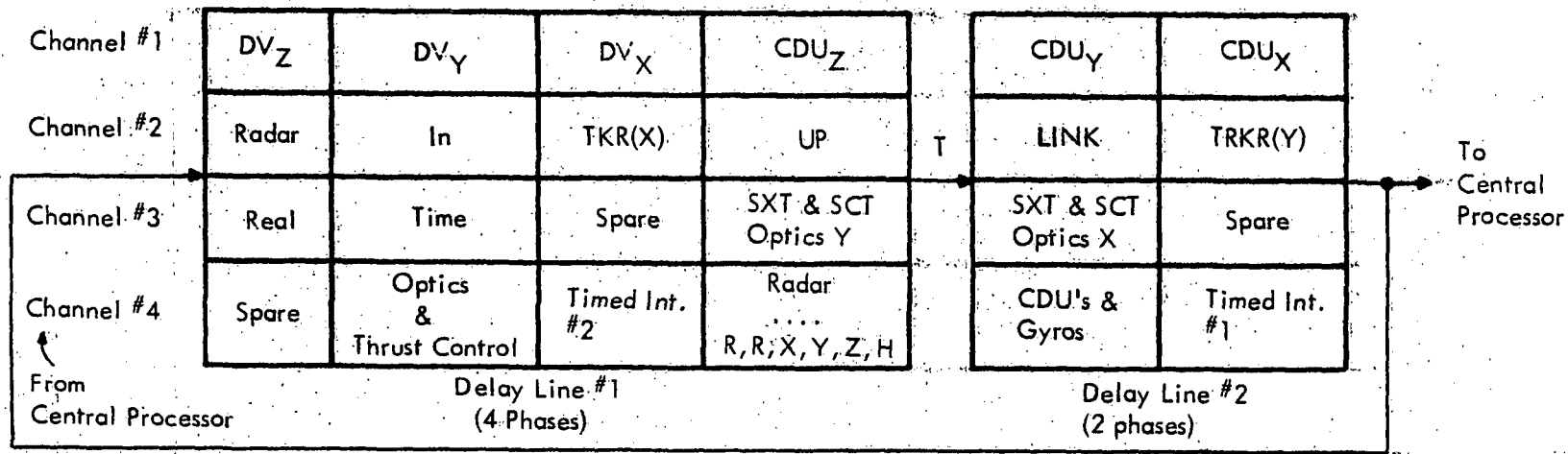
For the CP to address a particular delay line register in the ADA, the programmer must ensure that the register is available at the phase B or C time during the PIO instruction. To do this, one side of the odd-even phase latch has been made addressable as a discrete input. The programmer determines the state of this latch by addressing it with a PIO instruction and then testing the accumulator with either a TMI or a TNZ instruction.

Channels 1, 2 and 3 of the delay lines as shown in the block diagram of the ADA (see Figure II-8) are used to process inputs to the ADA. Channel No. 1 contains the six registers required to accumulate plus or minus pulses from the Pulse Integrating Pendulum Accelerometer (PIPA's) and from the gyros in the CDU. Interrogating pulses from the frequency countdown circuitry are sent to the PIPA's at a 3.2kcps rate. The return plus or minus pulses from the PIPA's are stored in logic devices until processed. These logic devices are then reset and are ready to accept the next PIPA outputs. With a loop length of approximately 164 usec

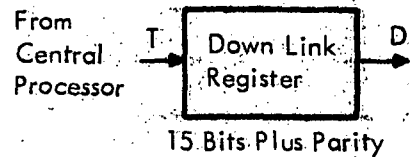
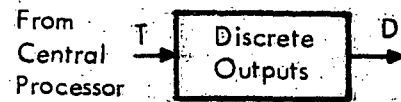
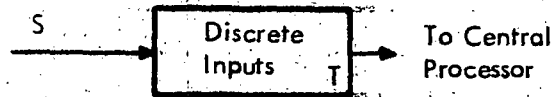
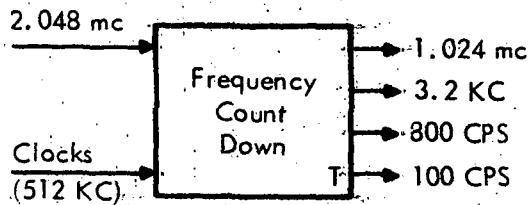
$$\left(\frac{1 \mu \text{ sec}}{512 \text{ bits}} \times \frac{14 \text{ bits}}{\text{phase}} \times \frac{6 \text{ phases}}{\text{loop}} \right) = 164.0625 \mu \text{ sec/loop}$$

the PIPA interrogation frequency could be increased from 3200 times/sec to approximately 6095 times/sec.

The inputs from the gyros in the CDU are treated in the same manner as the inputs from the PIPA. The plus or minus pulses from the gyros may occur asynchronously at any frequency up to a 6095-cps rate.



2-30



T - TMR Cktry.
D - Duplex Cktry.
S - Simplex Cktry.

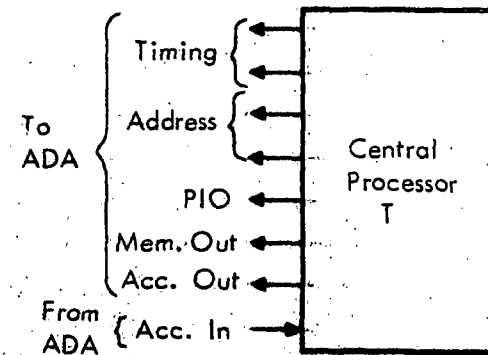


Figure II-8. Apollo Data Adapter Block Diagram.

Channel No. 2 contains the registers associated with UPLINK, radar range and $\Delta x + \Delta y$ pulses from the tracker radar. UPLINK and radar range both contain 15 bits of information and have been assigned two register positions each. The two registers associated with UPLINK may be addressed by the computer at the output of DL No. 1 during odd phases B and C. The two registers associated with radar range may be addressed at the output of DL No. 1 during even phases B and C.

Of the 16 bits of information received as UPLINK, the first bit is always a one. This bit is written into the UPLINK register. The contents of this register are shifted left one bit position each time a new bit — either one or zero — is written in the register. When a one appears in the UPLINK register 16 bit positions to the left of the first UPLINK bit, an interrupt is issued which indicates that the total UPLINK word has been received.

The radar range register is loaded in a manner similar to the loading of the UPLINK register. When radar range information is requested by the CP program, a one is written into the radar range register. This allows pulses to be sent to the radar unit at a 3.2-kcps rate. As each pulse is sent, a bit of information is received from the radar unit. The contents of radar range register are shifted left one bit position and the new bit is written in. This continues until the original one written into the register appears 16 bit positions to the left of its original position. At this time the 15 bits from the radar unit have been received and the pulses to the radar unit are terminated.

The remaining two registers in channel No. 2 are incremented in a manner identical to the registers in channel No. 1.

Channel No. 3 of the delay lines contains two 13-bit registers used to keep track of real time. Two additional registers accumulate the OR of Δx and Δy pulses respectively from the sextant or scanning telescope. The remaining two registers in this channel are spares.

The real time register is incremented by one in the low-order position each time the 100-cps signal from the frequency countdown unit changes from one to zero. This counter, which continues to operate when the CP is in its standby mode, will overflow each 7 days, 18 hours, 24 min. and 48.64 sec. This period is long enough so that no undetectable overflow can occur during the longest standby time.

The two registers associated with the Δx and Δy pulses are incremented in a manner similar to that used in channel No. 1

Channel No. 4 is the only channel on the delay line which may be loaded from the CP. Two of these registers, timed interrupt No. 1 and timed interrupt No. 2, are loaded independently from the computer with a negative quantity which is then counted toward zero one bit once each loop time. When the quantity in either of these registers reaches zero an interrupt signal is issued to the computer. Interrupt No. 2 has an additional function which is explained in the section describing DOWNLINK.

Three other registers in channel No. 4 are each multifunction registers used in sending a predetermined number of pulses to various pieces of external equipment. When a quantity is loaded into any of these registers, that quantity is counted toward zero at a 3.2-kcps rate until the register contents are zero. Each time the quantity is incremented or decremented by one, a plus or minus pulse is sent to the addressed equipment.

One of these registers is associated with the optics and thrust control, another with radar, and the third with the CDU and gyros. The sixth register in this channel is a spare.

3. DOWNLINK

The DOWNLINK register is loaded with 16 bits when addressed by the CP by a PIO instruction. Fifteen of these bits are from the CP. Bit 16 is an odd parity bit generated within the ADA. Another bit from the CP identifies the DOWNLINK word as either the first or not the first of a series of DOWNLINK words. Upon request of the telemetry equipment, 40 bits are serially sent to the telemetry equipment. These bits consist of the data word and parity bit which are sent twice and the identifying bit which is sent eight times. Upon receiving bit 40, the telemetry equipment issues an END PULSE causing an interrupt which informs the CP that DOWNLINK has been completed. These END PULSE's should occur every 20 ms. In the event that they occur too rapidly or too slowly, equipment associated with timed interrupt No. 2 will detect this and issue an interrupt informing the CP of the END PULSE failure. This is accomplished by loading the timed interrupt No. 2 register after the occurrence of an END PULSE with a quantity which will cause timed interrupt No. 2 at some time greater than 20 ms. When an END PULSE occurs, the contents of the timed interrupt No. 2 register are compared with a predetermined wired-in constant. If the contents of the register are greater than the constant, END PULSE's are occurring too rapidly. This conditions logic to prevent END PULSE's from causing interrupts and, by means of a discrete input, informs the CP of END PULSE failure. If, however, the CP is interrupted by a timed interrupt No. 2, the program in the CP will interpret this as either an END PULSE failure or a reduced telemetry rate.

4. INTERRUPT

Each occurrence of a condition which requires interrupting the CP sets a bistable device associated with this condition. The setting of one or more of these bistable devices will interrupt the present CP program except for the following conditions:

- The CP is involved in another interrupt routine.
- The program has inhibited the interrupt signal.
- A MPH, MPY, or a DIV is in progress.

If or when the above conditions do not exist, the CP may be interrupted. As part of the interrupt routine, the interrupt register is interrogated by a PIO instruction. By testing this word, the condition which caused the interrupt is determined and may then be processed.

5. DISCRETE OUTPUTS

Discrete outputs from the CP are issued as a word of 26 bits or less. The set gate on the bi-stable devices storing the state of the discrete outputs have been assigned an address different from the address assigned to reset gate, thus reducing the amount of bookkeeping required in the CP. The reset gate on the bi-stable devices in the interrupt register have been assigned a discrete output address. The CP is required to reset the bi-stable device in the interrupt register which indicated an interrupt.

6. DISCRETE INPUTS

All discrete inputs other than those processed by the delay line registers are serialized into a maximum of 26-bit words. Each word of discrete inputs is assigned an address and may be read into the CP with a PIO instruction. As mentioned in the section dealing with interrupt, the interrupt register is assigned an address and may be read in as a discrete input word.

7. ADDRESSING

The PIO instruction is the only voluntary command link between the CP and the ADA. Eight address bits associated with the PIO instruction have been given the following assignments:

- A1 If A1 = 1, information flow is from the ADA to the CP. If
 A1 = 0, information flow is from the CP to the ADA.

- A2** If A2 = 1, set inhibit interrupt
 If A2 = 0, reset inhibit interrupt
- A3** If A3 = 1, address delay lines
 If A3 = 0, address discrettes

The remaining five bits are used to specify devices within the groups defined by A1 and A3. The table below lists these four groups.

Group No.	A1	A3	
1	0	0	Discrete outputs from CP
2	1	0	Discrete inputs to CP
3	0	1	Load delay line register
4	1	1	Read delay line register

Group No. 1

- A4** If A4 = 1, the set gates of the discrete outputs are addressed.
 If A4 = 0, the reset gates of the discrete outputs are addressed.
- A5** If A5 = 1, the gates addressed by A4 are conditioned by memory output. If A5 = 0, the gates addressed by A4 are conditioned by the accumulator output.
- A6, A7, A8** Address bits A6, A7 and A8 define three subgroups within the area defined by A4 and A5. A one out of three code is used to define the subgroup. Each discrete output gate within a subgroup is conditioned by a particular bit position of the memory or accumulator output.

Group No. 2

Bits A4 thru A8 in a one-out-of-five code define five subgroups within Group No. 2. Discrete inputs within each subgroup are each assigned a particular bit position in a serialized word to the central processor accumulator.

Group No. 3 and 4

- A4** If A4 = 1, select delay line No. 1
 If A4 = 0, select delay line No. 2
- A6, A7** If A6 = 1, information is transferred during Phase B. If A7 = 1, information is transferred during Phase C. If A6 = A7 = 0, information is not transferred. If A6 = A7 = 1, information is transferred during Phase B and C.

Group No. 3

- A5 If A5 = 1, load channel No. 4 from memory. If A5 = 0, load channel No. 4 from the accumulator.
- A8 If A8 = 1, condition latches associated with the registers in channel No. 4 with the contents of the accumulator. If A8 = 0 these latches remain unchanged. The A8 bit is made equal to "one" when loading any of the three multifunction registers in channel No. 4. The three low-order bits in the accumulator in a one-out-of-three code specify which group of latches is to be conditioned. The next three bits in the accumulator — in a binary code — specify the new state of the latches.

Group No. 4

- A5 and A8 A5 and A8 specify which of the three channels is to be read by the CP.

A5	A8	
0	0	Nothing read
0	1	Read channel No. 1
1	0	Read channel No. 2
1	1	Read channel No. 3

8. REDUNDANCY IN THE APOLLO DATA ADAPTER

As shown by the key on the block diagram of the ADA (Figure II-8), some portions of the ADA are simplex, some duplex, and some TMR.

The simplex equipment consists of the transformers through which the ADA receives its inputs. The output from the secondary fans out to gates in the TMR portion of the ADA. Using simplex transformers in this area has the effect of multiplying the reliability of the external equipment by a factor of 0.99996 for each transformer connected to the particular piece of external equipment. During phase 2 of the Apollo program, the possibility of duplexing these input transformers will be considered. The multiplying factor in this case should be approximately $1 - (40.0 \times 10^{-6})^2$.

The duplex equipment consists of the discrete output circuitry. Signals within the TMR portion of the ADA drive two voters which in turn drive two paralleled transformers. The effect of this method of duplexing on the over-all system reliability is discussed in the reliability section of this report.

The remainder of the ADA is TMR. This includes the interface between the ADA and the CP.

D. PROPOSED PACKAGING FOR APOLLO GUIDANCE COMPUTER

1. GENERAL

This feasibility study included as a major effort, determination of physical packages for the guidance computer that would comply with the installation and environmental requirements of the Apollo Command Module. This section of the report addresses those physical constraints imposed by this application and defines the equipment packaging possibilities which are feasible within those constraints.

Through the cooperation of MSFC-Houston, MIT, and North American Aviation Corp. , adequate information regarding the physical and environmental requirements was obtained for the Command Module.

It is readily apparent that general environments, i. e. , vibration, shock, humidity, etc. are of secondary importance since the basic electronic packaging technology being developed for Saturn V equipment has a similar environmental specification.

Of prime importance and consideration are the physical size, cooling, connector requirements imposed by the present Apollo volume allocation, preference for cold-plate cooling, and usage of a specially developed high density (Hughes) connector for equipment interconnection. Consequently, feasible packaging approaches in this report are described in terms of these prime items.

The reader needing a detailed description of the present Saturn V equipment for reference in evaluating the proposed packaging approaches is referred to Vol. II, Sections III and IV.

2. SATURN V PACKAGING REVIEW

The entire equipment, when assembled, is designed to comply with environmental requirements. The equipment will be "semi-sealed" to maintain a slight over-pressure to prevent hydrogen hazards, and humidity conditions of internal condensation. The present weight and volume requirements of this equipment are as follows:

<u>Saturn V</u>	<u>Weight</u>	<u>Volume</u>
4-Memory Computer	80 pounds	2.1 cubic feet
Data Adapter	94 pounds	2.6 cubic feet

A brief review of the Saturn V packaging for readers already familiar with the technology is as follows:

a. Unit Logic Device (ULD)

A ULD is the lowest meaningful electronic assembly containing resistors, conductors, and semi-conductors. (See Figure II-9.) The device is 0.3 in. x 0.3 in. x 0.070 in., and is fabricated by silk-screening and firing techniques to obtain a resistor-conductor pattern on an alumina substrate. After resistor trimming and solder bath processing, sealed semi-conductors fabricated by IBM, using a leadless mounting technique, are connected by a solder re-flow technique and the assembly is then encapsulated.

b. Multilayer Interconnection Board (MIB)

A MIB is a laminated assembly of two-sided circuit boards, each containing etched circuit wiring. Connection is made between layers by plating through drilled holes made in land patterns established in each of the conducting layers.

c. Page

A page is the next higher assembly of electronic packaging using ULD and MIB technologies. Two MIB's having 10 to 12 layers of wiring each and a top land pattern (shown in Figure II-10) are bonded to opposite sides of a flat structural member. At the same time each is connected to half of a 98-pin connector at the lower part of the pattern, and feed-through connections from MIB to MIB (through apertures in the structural plate) are made.

Tested ULD's are positioned singularly on the proper land pattern, and are connected electrically by reflowing land pattern solder using infra-red energy.

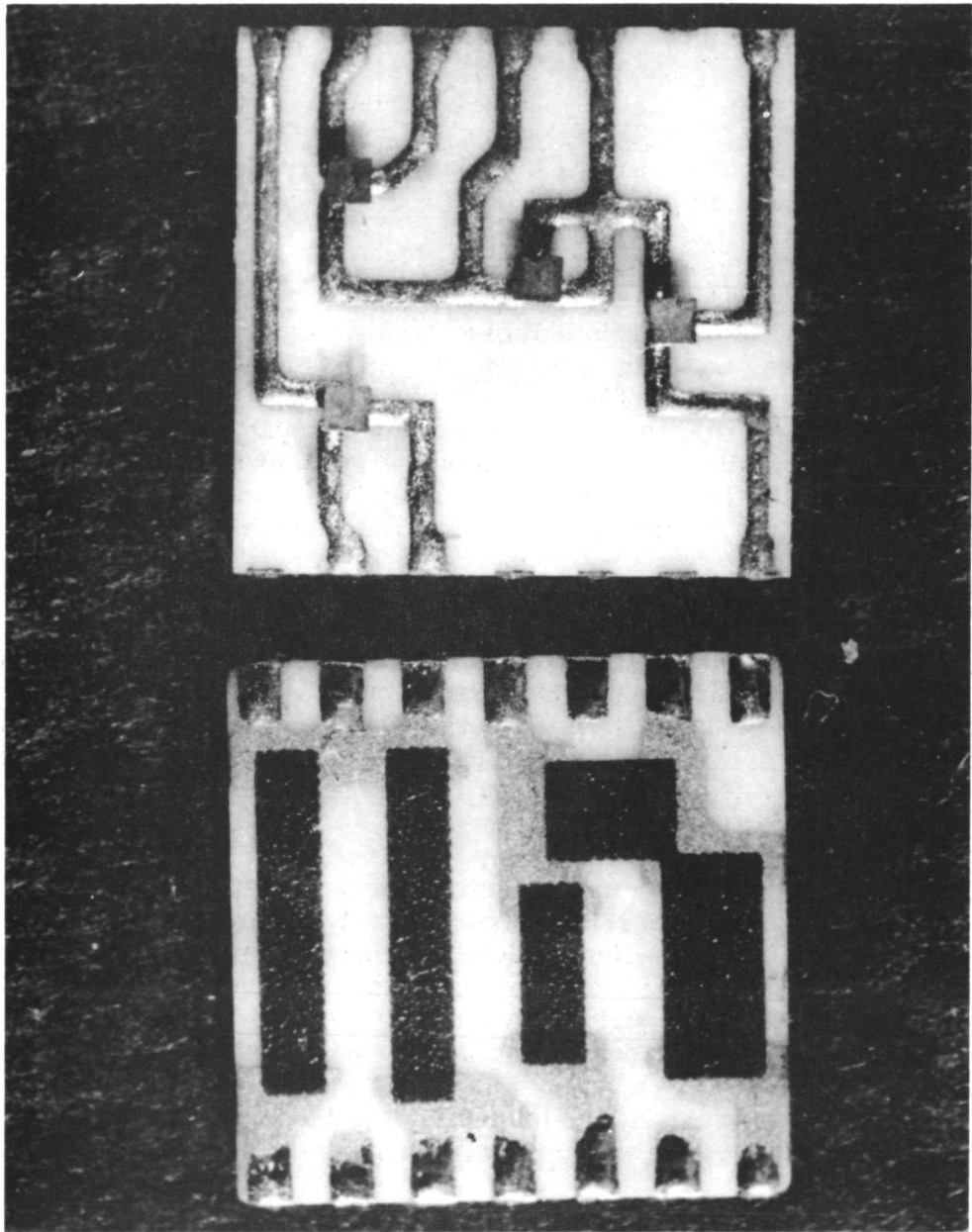


Figure II-9. ULD Layout

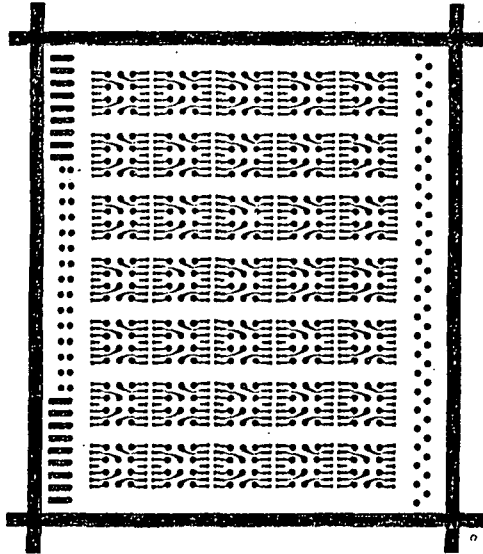


Figure II-10. MIB Top Land Pattern for 70-ULD Page

The result is a pluggable assembly of 70 ULD's, electrically interconnected, and bonded to a structural element which provides rigidity and a thermal conducting path when installed in the equipment. (See Figure II-11.) Since each ULD represents an average capacity of eight components, then one 70-ULD page has a component density of 560 components.

The basic page design also has the flexibility of mounting delay lines, transformers, and discrete components, in total or in part, with ULD's. Pages are plugged into a main interconnection board (referred to as a back panel), capable of mounting eighteen page connectors. This assembly is referred to as a channel. Interconnections between channels are achieved through high density interconnection blocks. Each block contains 144 interconnecting points (12 groups of 12 points) interconnected with tape cables.

d. Memory

Each memory module consists of 14 planes of 8192 cores each, together with associated driving and reading circuitry in ULD and MIB packaging surrounding the array on 4 sides. Connection to the computer is made with one 98-pin connector mounted on the memory distribution board. The present memory module configuration is 5.25 in. x 5.5 in. x 5.75 in. Duplexing of modules for redundancy requires two modules for 4096 words of 28 bits each.

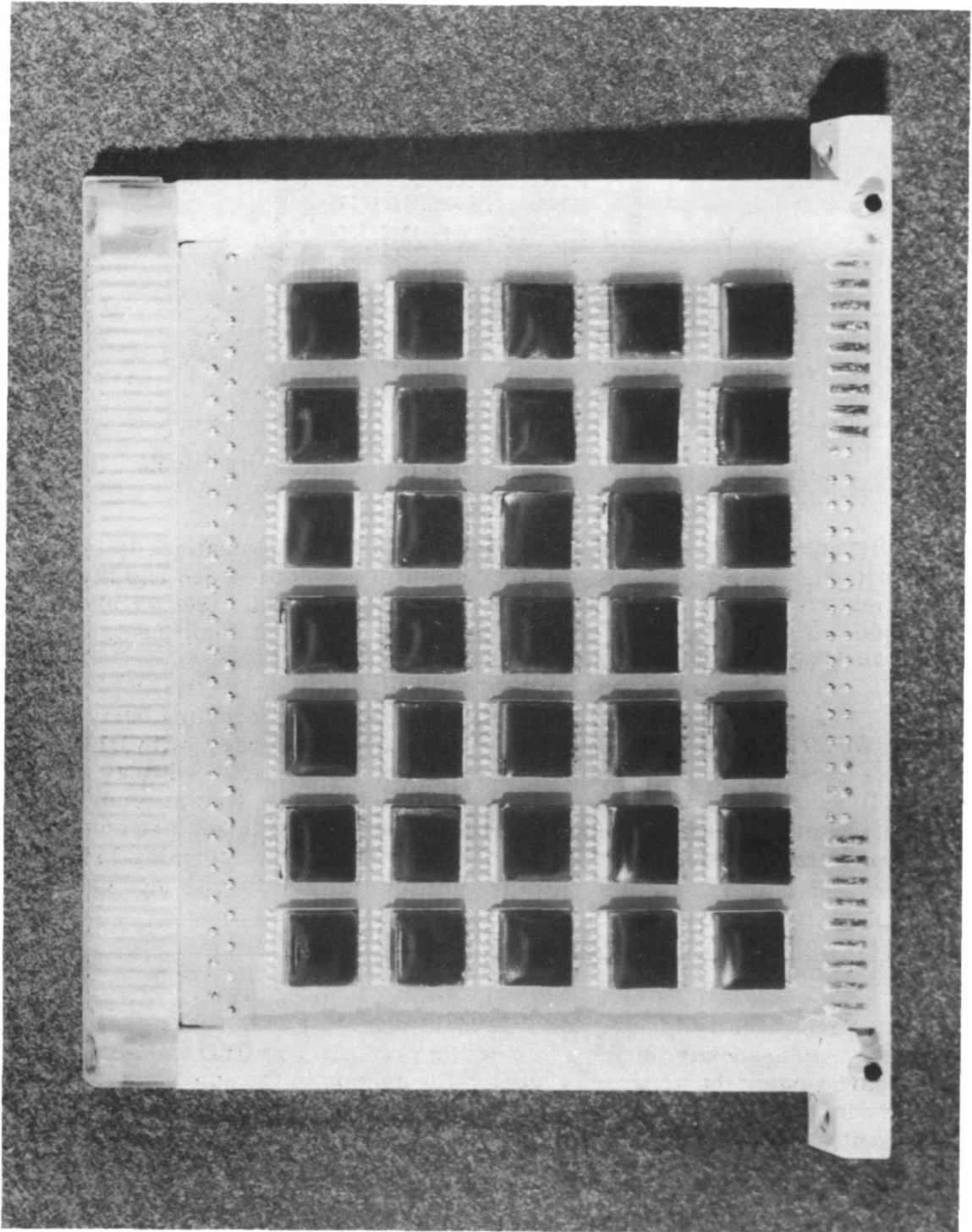


Figure II-11. Page Assembly

e. Structure and Cooling

A structure with integral, or self-contained, cooling is used. The basic structure is a cell-like machined element, fabricated from a solid Magnesium-Lithium billet. As part of the fabrication, self-contained coolant passages are gun-drilled through the walls of this celled structure. Welding is used to seal intersections that occur on outside walls. Coolant is circulated through this structural element from the main system in the Saturn V Instrument Unit through "quick-disconnect" fitting attachments.

Internal heat transfer is effected by conduction from the metallic sides of "Pages" to the cooled structure walls by spring clips to offset vacuum conditions. Other electronics, i. e., memory, is also assembled so that heat generating elements have a short conducting path to the main structure.

f. Interconnection

Internal interconnections between major logic and electronic elements are made by use of etched flat cable techniques. These cables are soldered to the MIB's acting as internal panels or to Bendix "Pigmy" connectors that make up the equipment electrical interface. An out gassing test was performed on the Bendix connector receptacle under a vacuum condition of 1×10^{-4} mm Hg and 100°F. (Condition as per NASA Apollo G&N Spec. ND-1002037.) The weight loss recorded was 0.213 grams for 72 hours of operation. This represents a 0.03 percent total weight loss in comparison to its original weight.

3. EQUIPMENT PACKAGING APPROACHES FOR APOLLO

Within the Command Module, most electronic packages are installed within a central rack structure. For each shelf within this rack structure, a cold plate is provided serving the dual purpose of unit support and ultimate heat sink capability. For those units not requiring in-flight maintenance, integral cooling may be considered as a means of heat sinking. The use of an integral cooling technique also eliminates the need for maintaining pressure against the cold plate thermal interface and reduces the number of thermal interfaces from electrical component to ultimate sink.

For both cold plate and integral cooled systems a maximum liquid temperature, to the CP and DA, of 90°F is expected. A maximum coolant temperature rise of 30°F is expected through these units. The power supply will experience an inlet temperature of approximately 120°F with a negligible temperature rise from inlet to exhaust.

The volume allotted for the CP and DA with the Apollo rack structure is shown in Figure II-12. All electronic apparatus unit structure, electrical and liquid connectors with the exception of the power supply must be housed within this given volume. If integral cooling is used an additional 0.5 in. of depth is realized. The deletion of the present cold plate and its thermal interface material accounts for this gain. To package all the necessary electronic apparatus within this volume, an alteration of the present Saturn V memory module is necessary. The proposed memory package, shown in Figure II-13, consists of a 14-core-plane stack ($64 \times 128 \times 14$ cores) with its associated electronics. Each core plane is considered to be of the dimensions 3 in. \times 5 in. \times 150 in. A complete Apollo memory module with electronics will have the overall dimensions of $5.34 \times 8.25 \times 2.8$ in.

An additional volume is allotted for the power supply (refer to Figure II-14 situated adjacent to the central processor and data adapter. A trade-off study was made to place the DA within this allotted volume. This would burden the rack-mounted cold plate with an additional 80 watts and complicate overall interconnection between processor, power supply, and data adapter. The power supply (shown in Figure II-15) will be attached to a cold plate for cooling purposes. This method of cooling is most desirable since, due to its remote location, making or breaking liquid quick disconnects would be difficult.

Two basic arrangements of the CP and DA are proposed for the Apollo Command Module. The first arrangement considers a CP and DA as separate entities. The second arrangement proposes an integrated unit in which both the CP and DA are housed.

Complying with the installation criteria, discussed previously, Figure II-16 delineates a two-unit configuration which utilizes the allowable rack structure volume.

Physically, the difference between this and the integrated unit is the addition of structure walls required for the two-unit design. However, this feature accentuates the ease of handling concept for insertion and removal within the rack structure. Furthermore, a major advantage may be achieved in providing a common CP both the Saturn V and Apollo installation. In Saturn V, additional brackets would be required, on the C. P. front and rear Figure II-16 to meet the installation dimensions dictated by MFC.

The CP will consist of 78 ULD pages combined into three simplex channels and two voter channels. In addition, four memory modules will be mounted to the CP top structure. Sufficient mounting areas, between the memory package and the processor structure, will be allotted for (a) structural integrity and (b) minimizing temperature gradients between the memory heat generating components and heat sink.

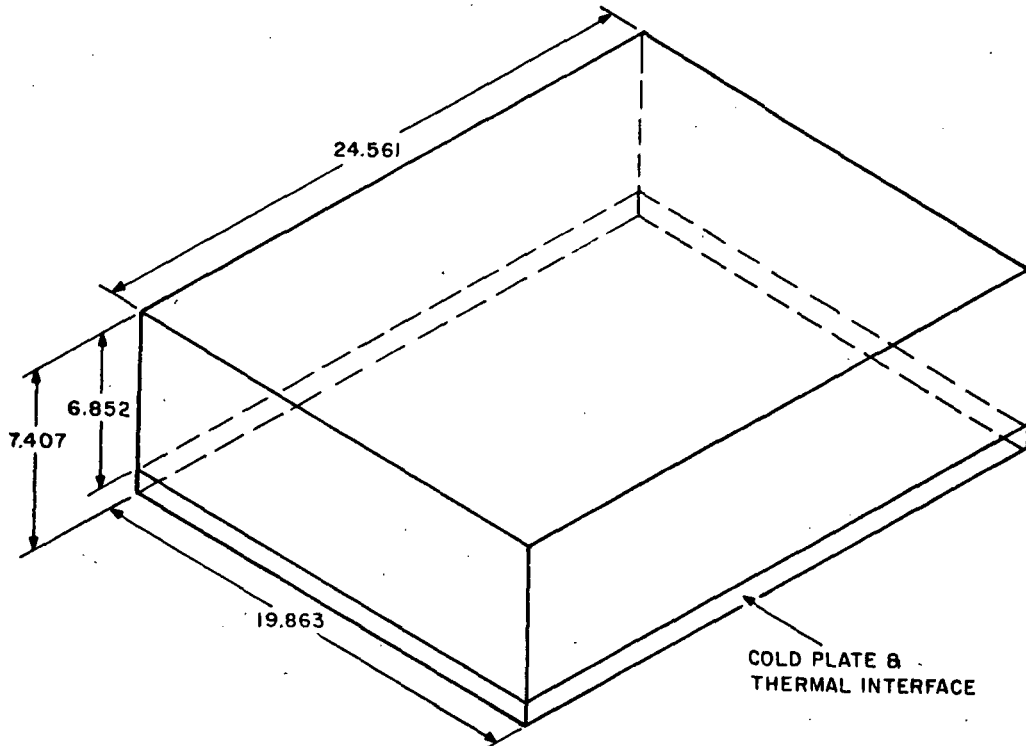


Figure II-12. Volume Allotted for Apollo CP and DA

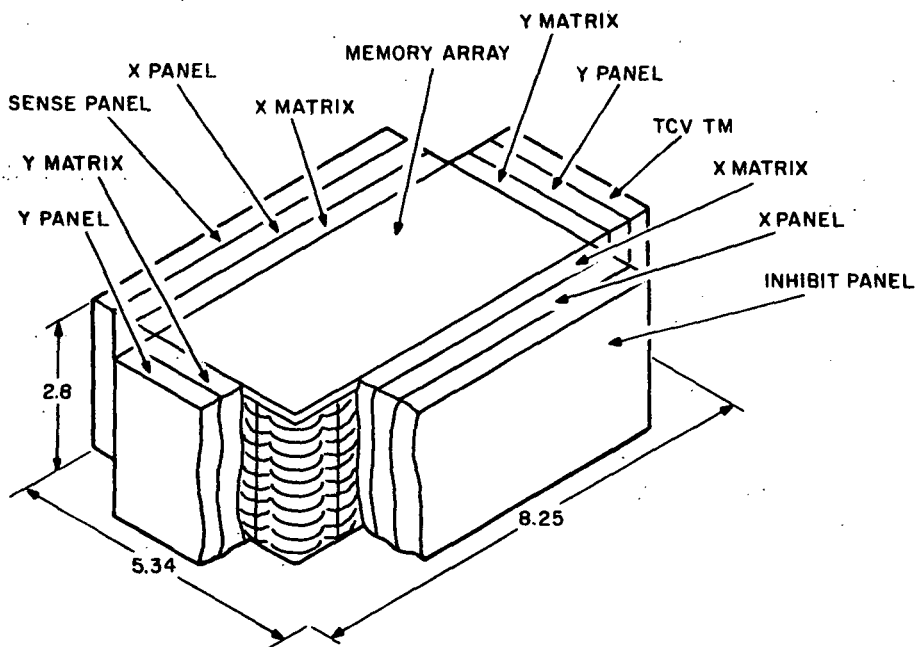


Figure II-13. Proposed Memory Package

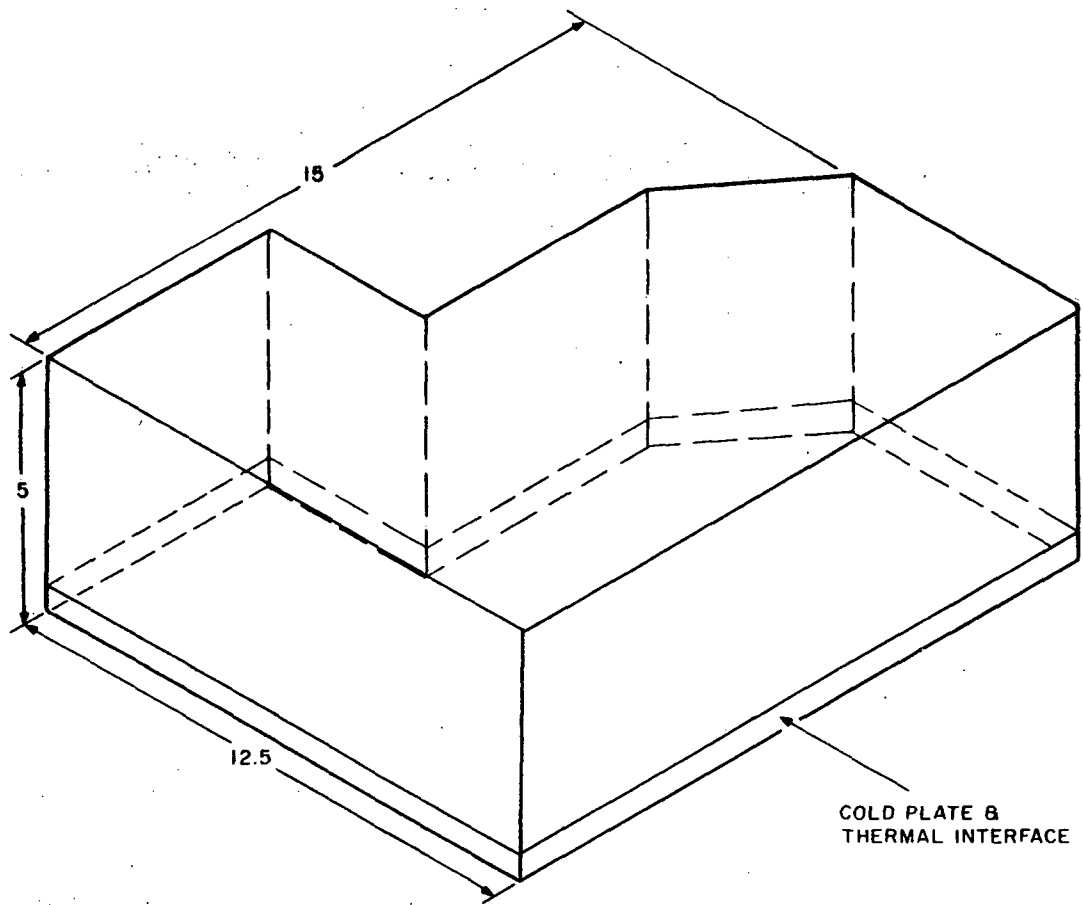


Figure II-14. Volume Allocated for Power Supply

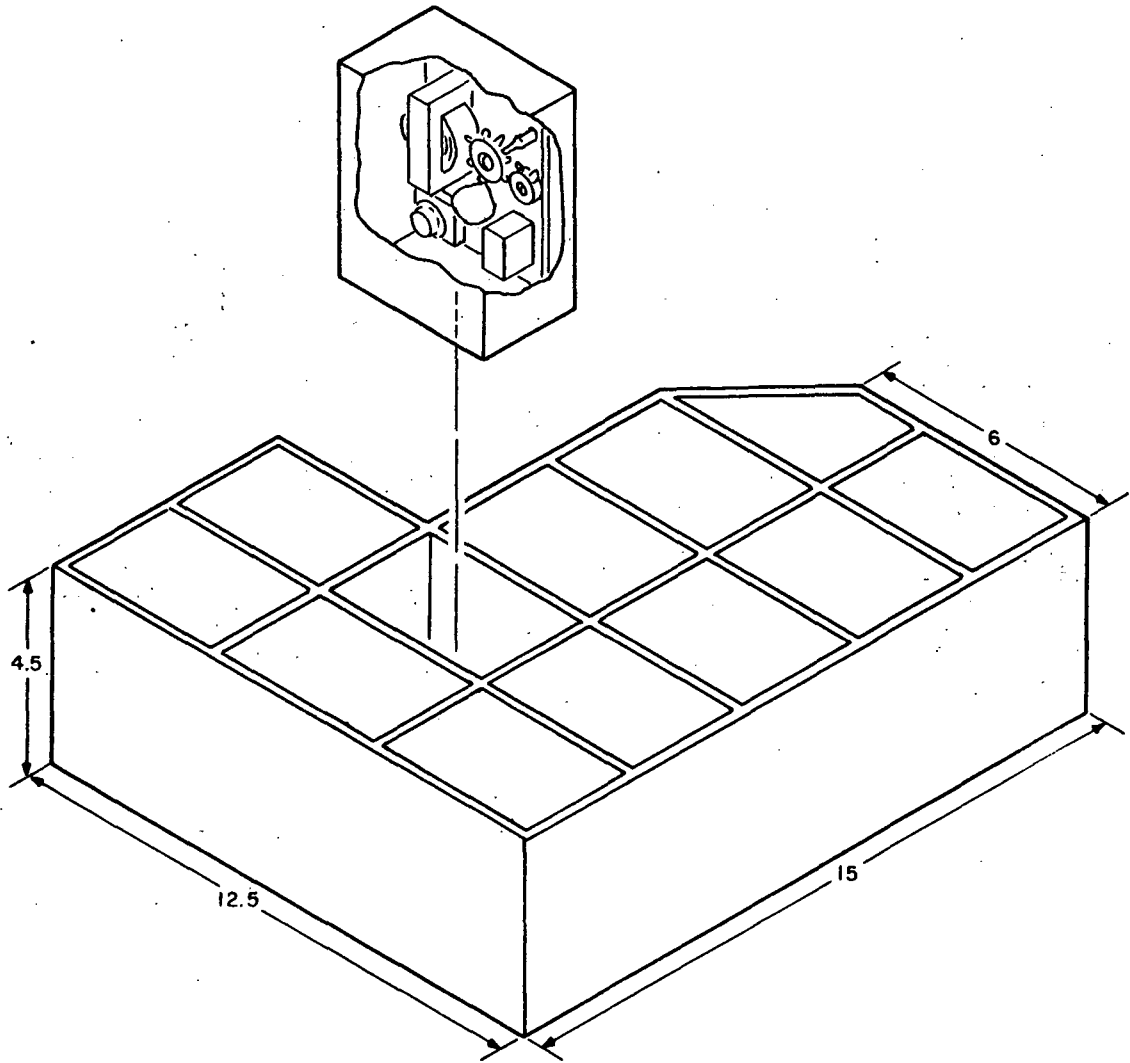


Figure II-15. Apollo Power Supply

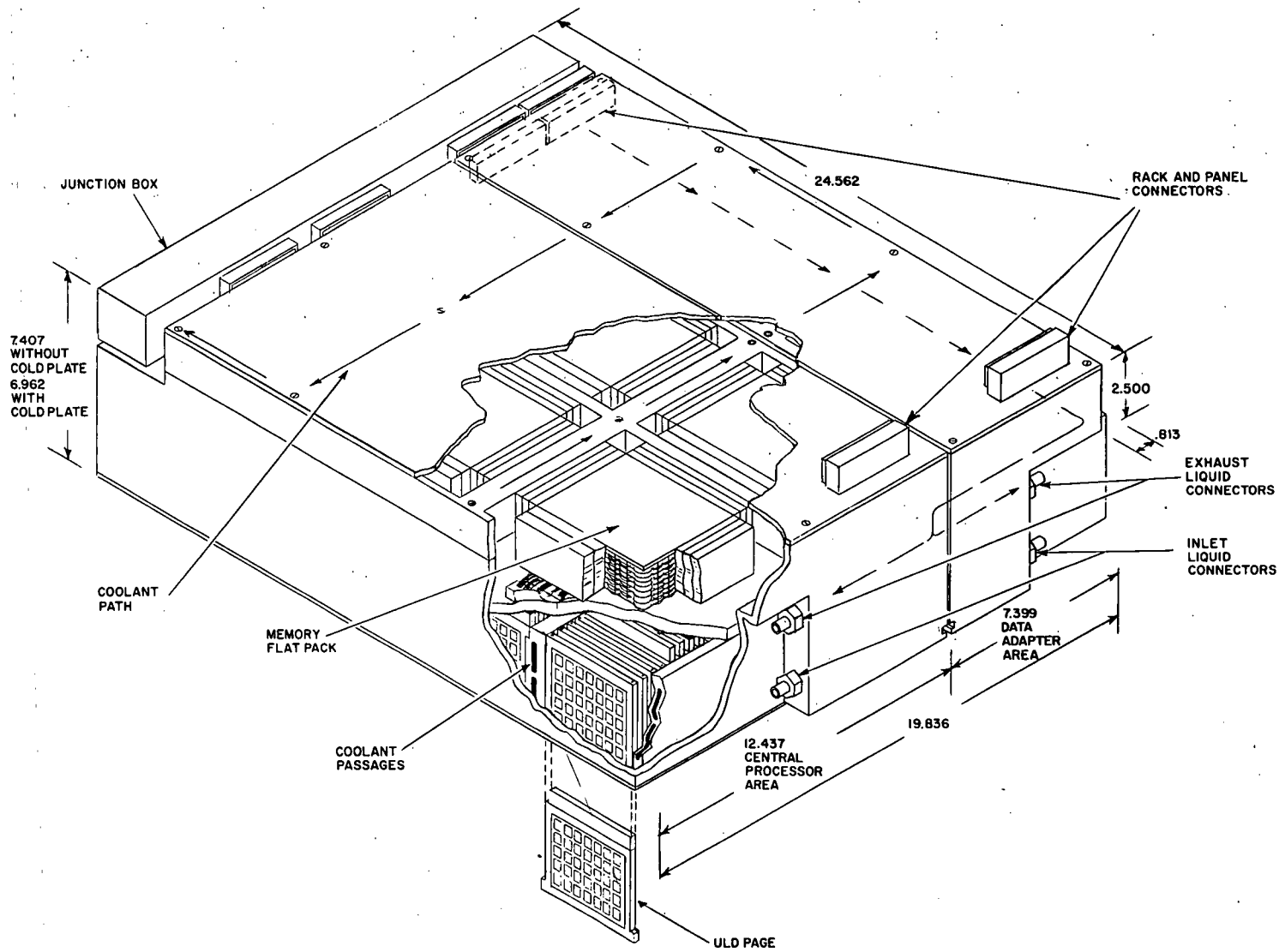


Figure II-16. Central Processor and Data Adapter in Individual Housings

The DA consists of 56 ULD pages with two additional memory modules attached as indicated in the processor design. Delay line modules will be attached on a reference plane identical to the memory. This fully uses all available volume within the DA housing.

Figure II-16 shows a general layout for an integral cooled system. For this configuration, coolant passages are provided within the unit internal structure. The coolant is allowed to flow through the structure passageway, serially, from inlet to exhaust. All component heat generations are transferred from the ULD page metallic frame to a mechanical support attached to the coolant passages. Maximum component operating temperatures for the integral design are 14°C lower than the cold plate cooled design. Memory electronic heat generations are transferred to the array frame mounting pads attached to the internal structure.

For a cold plate cooled system, modifications to Figure II-16 are minor. All coolant passageways, now become solid structure members. These members, along with a surface from every ULD page, mount directly to an interface material provided between the units and cold plate. All unit heat generations are transferred from the page metallic frame to the unit structure and ultimately dissipated to the cold plate. The unit liquid connectors shown in Figure II-16 are deleted since the cold plate acts as the main heat sink.

For both the cold plate and integral unit designs, electrical connections are identical. One main junction box, located to the rear of the rack structure, connects to the Power and Servo Assembly located above. Interconnection between the DA and CP is performed at the front and rear of each unit. The following, is a list of required pin densities to totally accomplish all connections:

<u>Unit</u>	<u>Location</u>	<u>Pin Density</u>	<u>Number Required</u>
Central Processor	Rear	38	1
	Rear	144	1
Data Adapter	Rear	240	2
	Front	38	1
Central Processor	Front	240	1
	Front	240	1
Data Adapter	Front	240	1
	Front	108	1
	Front	88	1

All connectors, referenced above are based on using Hughes Connectors. A family of Hughes connectors is being presently qualified at NAA.

The combination of both a CP and DA into one complete housing is shown in Figure II-17. Again, provisions for a cold plate or integral cooled design may be considered similar to the split unit design. The concept of liquid connections, electrical connections, general heat transfer paths and internal structure are also identical to the split unit. The removal of those structure walls, required in the two-unit design is reflected in a slight growth in DA page capacity. The DA section can contain a maximum of 66 ULD pages in comparison to the 56 mentioned previously.

The main electrical connections are also alleviated to some extent. Those required for the one unit concept are as follows:

<u>Location</u>	<u>Pin Density</u>	<u>Number Required</u>
Front	240	1
Back	240	4

In summary, the common mechanical features which exist between the Saturn V and the proposed Apollo designs are as follows:

- All logic ULD pages are 3.8 in. × 3 in. × 3.8 in. in dimension
- All page centerline spacings are 0.440 in.
- Electrical connections from page to back panel are identical
- Interconnections between back panels are identical
- Both the Saturn V and Apollo CP contain identical layouts of three simplex page channels and two voters page channels
- Page capacity for the Apollo central processor and the Saturn V computer is identical
- Mechanical page attachment to internal structure is identical for Saturn V and Apollo
- Memory module pages are identical.

Those items proposed which are dissimilar are:

- The memory array plane
- The memory array form factor
- The type of external electrical connectors.

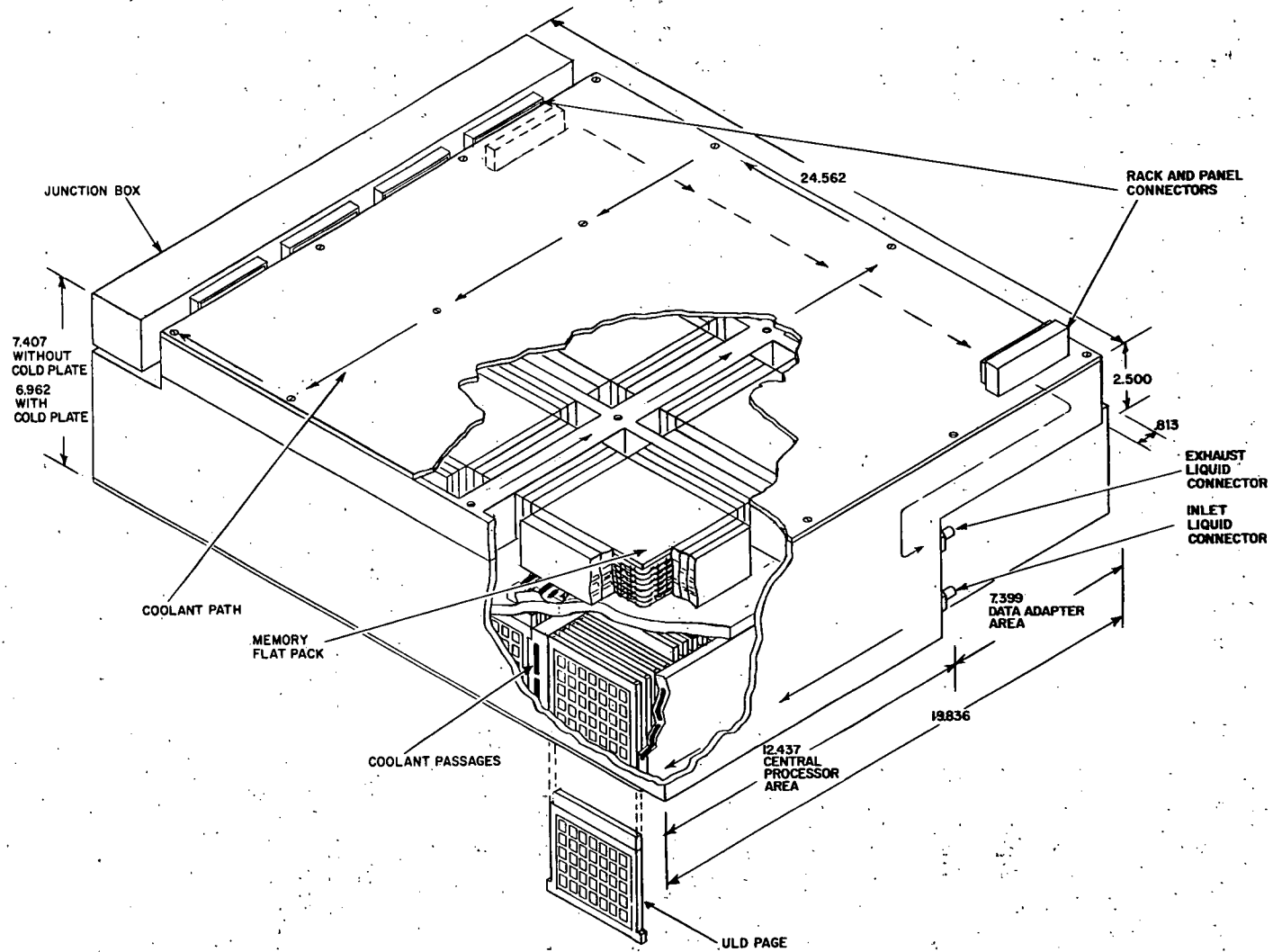


Figure II-17. Single-Housing Central Processor and Data Adapter

A comparison of the maximum expected semiconductor operating temperature between the Saturn V and Apollo units is as follows:

	Inlet Coolant Temp.	Exhaust Coolant Temp.	Maximum Semiconductor Junction Temp.
Saturn V	60°F	65°F	158°F
Apollo (Integral)	90°F	120°F	203°F
Apollo (Cold Plate)	90°F	120°F	228°F

E. CIRCUITS

1. GENERAL

The logic and memory circuits for the Apollo Guidance Computer are exactly the same as those designed for the Saturn V computer. These circuits are described in the Saturn V Computer description which is Volume II of the study report. The circuits unique to the Apollo Guidance Computer will be described here; they include the power supply, oscillator, and input-output transformer circuits.

2. APOLLO POWER SUPPLIES

The d-c power supplies for the Apollo computer and data adapter will be duplexed, pulse-width regulated dc-to-dc converters. The basic requirements will be the same as those for the Saturn V circuitry. In addition, for the portion of circuitry that is required to operate continuously (i. e., the real time clock), an independent, duplexed, multiple-output power supply will be provided. Since the loads imposed on this supply are approximately constant, it is feasible to use a single regulated converter to provide several well-regulated output voltages. The estimated power requirements for the real time clock are given in Table II-5. The use of an independent supply for this application results in maximum circuit efficiency. The multiple-output technique minimizes the number of components required. The total component requirements represent duplexed power converters, each with duplexed feedback amplifiers, plus six additional rectifier-filter-isolation diode circuits. The components are listed in Table II-6.

To determine whether a failure has or has not occurred in any portion of the duplexed power supply, it is necessary to disable each redundant section and check the output voltages or the load operation. The d-c feedback amplifiers may be disconnected via the bias circuits; shown in Figure II-18; marked PSI-2, PSI-3, PS-2 and PS2-3. Of the pair of power converters, one may be forced off by applying a bias to the PS-1 or PS2-1 points. The check-out truth table is shown in Table II-7.

Three connections are required for each of 14 power modules. The total requirements for independent module checking are 42 connections. However, six connections are sufficient to prove that all circuits are operational.

The module switching requirements may presumably be met with direct ground control or with a local decoding matrix and drivers. The requirements are similar to those of TMR channel switching.

Table II-5

POWER REQUIREMENTS FOR THE REAL TIME CLOCK AND ASSOCIATED CIRCUITRY

Circuit Function	Supply Voltages			
	+20V	+12V	+6V	-3V
Clock generator and clocked ANDs	0.005 amp	0.62 amp	0.72 amp	0.21 amp*
Logic and control	-----	-----	1.20 amp	0.11 amp
Voters (36 trios)	-----	0.32 amp	0.56 amp	0.14 amp
Delay line drivers and sense amplifiers	-----	-----	0.01 amp	0.001 amp
Total Current	0.005 amp	0.94 amp	2.49 amp	0.461 amp
Total DC Load Power -----		27.8 watts		
Power Supply Efficiency -----		65 percent		
DC Input Power -----		43 watts		
*Based on 12 Clock Drivers for both the computer and data adapter; 1 Emitter Follower/ 9 Clock Drivers and 2 Emitter Followers/3 Clock Drivers respectively.				

Table II-6

COMPONENT REQUIREMENTS FOR DUPLEXED
POWER CONVERTERS

Power transistors	-----	4
Small signal transistors	-----	28
Transformers	-----	6
Inductors	-----	8
Ceramic capacitors	-----	16
Electrolytic capacitors	-----	12
Power rectifiers	-----	24
Diodes	-----	10
Resistors	-----	92
Zener diodes	-----	4
Potentiometers	-----	4
Total	-----	208

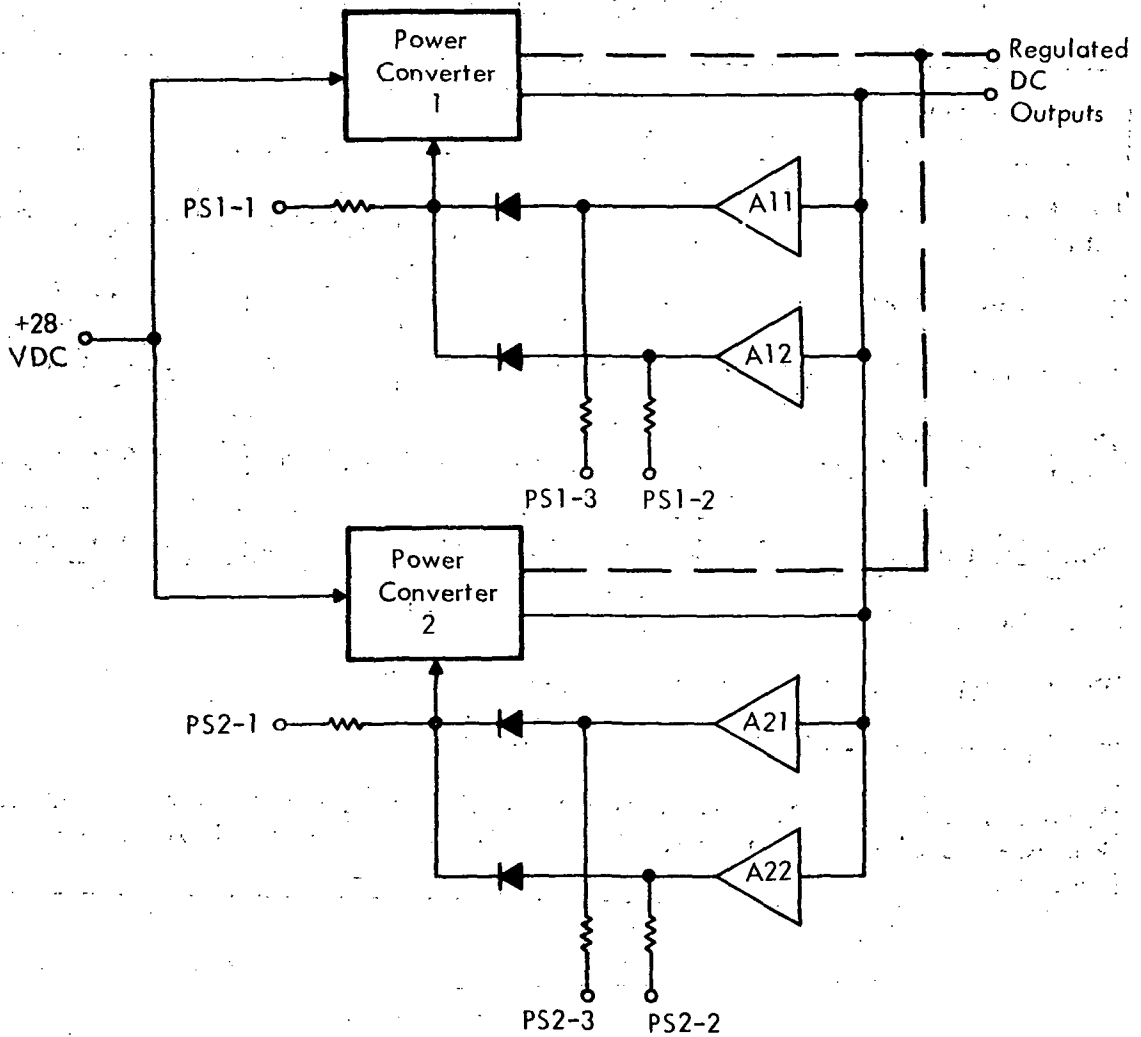


Figure II-18. Bias Circuits

Table II-7

MODULE CHECK CONTROL

Operational Check	PS1-1	PS2-1	PS1-2	PS2-2	PS1-3	PS2-3
none	----	----	----	----	----	----
PS1 OK	----	+20V	----	----	----	----
PS2 OK	+20V	----	----	----	----	----
A11 OK	----	+20V	+20V	----	----	----
A12 OK	----	+20V	----	----	-20V	----
A21 OK	+20V	----	----	-20V	----	----
A22 OK	+20V	----	----	----	----	-20V

For power supplies located externally to the processor/data adapter, one of two conditions must be met. Either (1) the interconnection cable must have a large conductor cross-section to reduce distribution tolerances or (2) the load voltages must be remotely sensed (i. e., at the load). Reliable remote sensing will require two lines per supply voltage or 14 total sense lines. These must be RFI shielded, but may be multiple-conductor cables requiring 16 connections at each end plus the shield bond. Sense current requirements are about 2 ma per amplifier or 8 ma per sense line (with one open line permitted).

The load current requirements and voltage tolerances for the total central processor/data adapter are listed in Table II-8.

Using remote sensing, the voltage line drops can be permitted to go as high as 100 mv. The total line requirements are 16 sense lines plus 53 power and ground lines for a total of 69 wires.

Table II-8

**CENTRAL PROCESSOR/DATA ADAPTER CURRENT REQUIREMENTS
AND VOLTAGE TOLERANCES**

Supply Voltage	Maximum Tolerances	D. Adapter Load Current	Central Processor Load Current	Total Load Current
+20V	+440 mv -490 mv	-----	1.15 amp	1.15 amp
+12V	+240 mv -290 mv	1.2 amp	2.4 amp	3.6 amp
+6V	+120 mv -170 mv	8.5 amp	-----	8.5 amp
+6V	+120 mv -170 mv	-----	13.0 amp	13.0 amp
-3V	+110 mv -60 mv	1.0 amp	2.9 amp	3.9 amp
Total Power ----- 206.9 watts				

3. TRANSFORMER INPUT CIRCUIT

The transformer coupled input circuit is shown in Figure II-19. A simplex input circuit is used. The input line is properly terminated, and a transistor buffer is used to fan out into TMR logic. Redundant configurations for the input transformer circuit will be investigated during Phase II of this program.

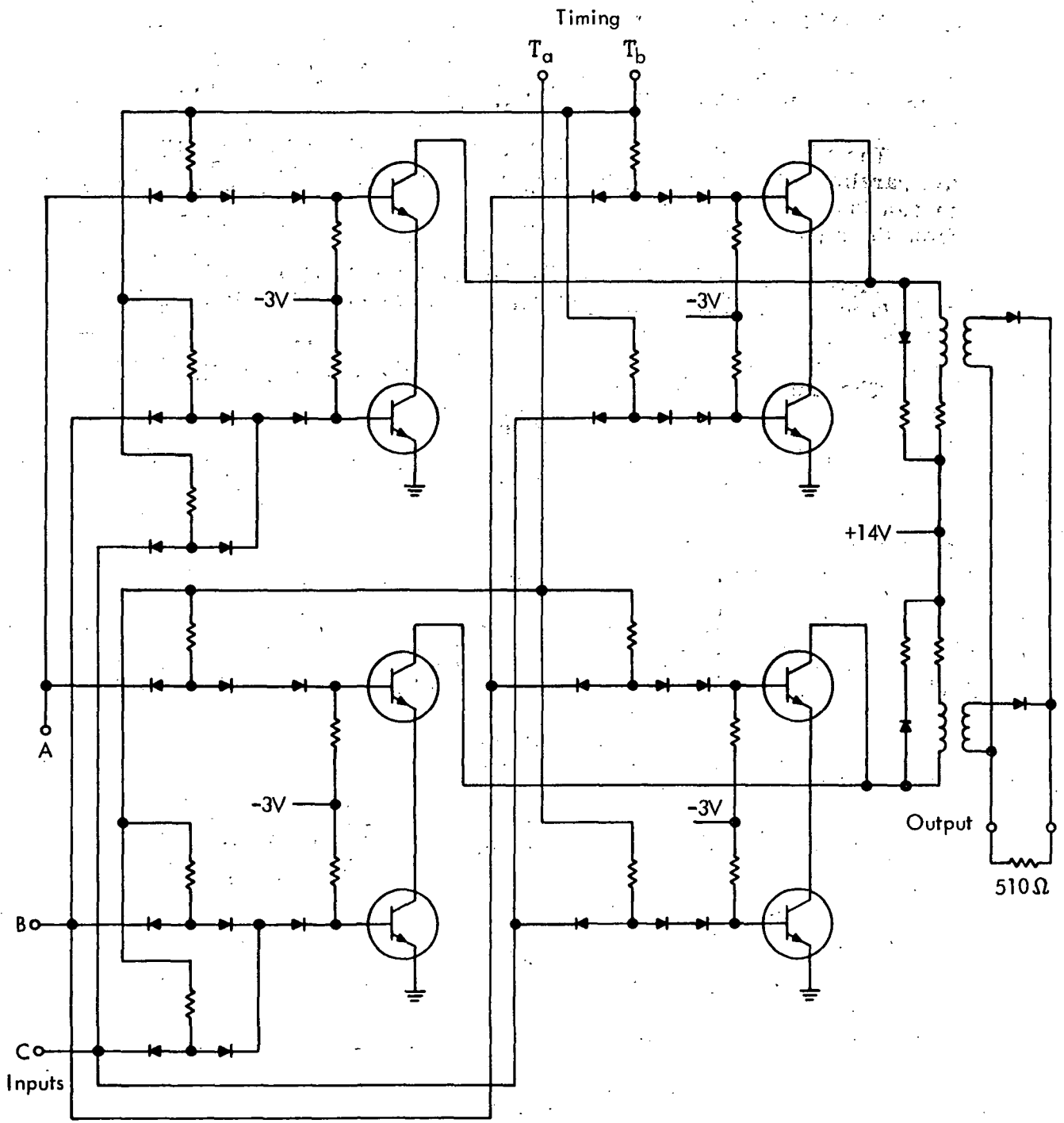


Figure II-19. Transformer Coupled Input Circuit

4. TRANSFORMER OUTPUT CIRCUIT

The transformer coupled output circuit is shown in Figure II-20. The output is a majority function of the three logical inputs A, B, and C.

The circuitry is redundant, (i. e., there are two similar circuits which are paralleled so that their outputs perform a logical OR function). Each circuit is constructed so that no single failure can cause an erroneous output from the redundant pair.

Timing of the output pulses is controlled by the timing signals T_a and T_b . These signals are similar, but are generated separately to provide failure isolation.

The logical structure of the circuit is shown in Figure II-21.

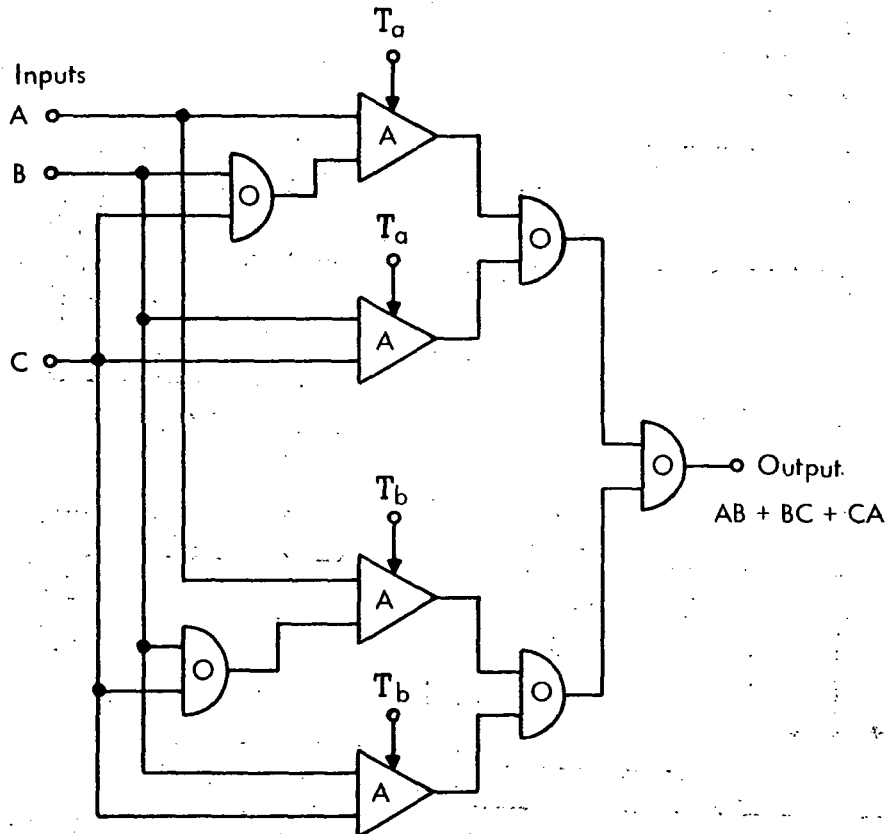


Figure II-20. Transformer Coupled Output Circuit

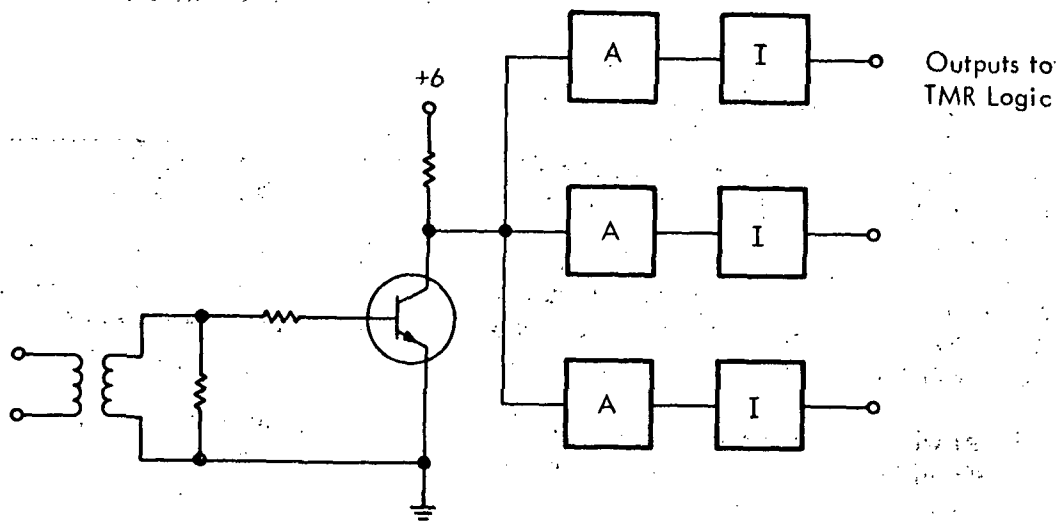


Figure II-21. Transformer Coupled Output Logic

5. APOLLO OSCILLATOR

The Saturn V oscillator is a 2.048 mc crystal controlled oscillator of a modified Pierce configuration with no temperature control or compensation. The oscillator stability is predicted at ± 25 parts in 10^6 with ± 5 parts in 10^6 due to initial tolerance, ± 5 parts in 10^6 due to aging over a two year period, and ± 15 parts in 10^6 due to the 0 to 80°C temperature range.

The Apollo mission requires an oscillator stable to within one part in 10^6 over a 10-day period. The oscillator nominal frequency is to be 2.048 mc. The operating temperature range of 0 to 80°C will remain the same as the present oscillator. Also, with the capability of calibrating out initial frequency differential from nominal, the initial purchase tolerance of ± 5 parts in 10^6 and the two year aging drift tolerance of ± 5 parts in 10^6 may remain the same as the present oscillator. Thus, the most difficult portion of the task will be stabilizing the oscillator against frequency changes due to temperature without a significant reduction in reliability as compared to the reliability of the present Saturn V oscillator (.99986 for 250 hours).

The Apollo oscillator will require accurate temperature compensation or control to meet the stability requirement. Temperature control is not desirable because of the possible high ambient temperature (crystal operation above 80°C would cause faster aging), the warm-up time required, the thermal lag of an oven, the power required by an oven, and the complicated control circuitry required for maintaining an accurate oven temperature. Therefore, compensation for changes in crystal frequency with temperature will be used to stabilize the oscillator. A promising circuit configuration is a bridge oscillator with the temperature compensating elements included in the bridge. The amplifier portion of the oscillator would be a two-stage differential amplifier.

Redundancy will be used to obtain the required stability without a significant reduction in reliability. The bridge oscillator may be made more reliable without reducing accuracy by making the amplifier portion of the oscillator redundant. Another approach to a redundant oscillator scheme is to have two oscillators, the a-c output of one oscillator biasing the amplifier portion of the other oscillator off. Once the first oscillator fails, the second oscillator will provide the a-c output. Such a scheme, however, usually suffers from momentary lack of an output during transition to the standby oscillator. Another scheme would provide three separate oscillators which would be mutually independent with the exception of a mutual reactive element(s) which would provide synchronization of the three outputs. This scheme should provide the highest reliability but may suffer from a loss in accuracy after a failure. The latter scheme appears the most promising and will be investigated first.

The Apollo oscillator design task shall result in a quartz crystal controlled oscillator with the following characteristics:

- Nominal Frequency 2.048 mc
- Reliability 0.9998 for 250 hours
- Temperature Range 0 to 80°C
- Stability
 - Temperature Stability ± 4 parts in 10^7
 - Short Term Aging ± 1 part in 10^7 for 10 days
 - Long Term Aging ± 5 parts in 10^6 for 2 years
 - Initial Tolerance ± 5 parts in 10^6

Because of the unavoidable increase in size of the proposed oscillator over the present configuration, the oscillator may have to be remotely situated from the clock generator and the output converted to a low impedance for transmittal to the clock generator.

F. IMPACT OF IDENTICAL CENTRAL PROCESSORS FOR SATURN V AND APOLLO

If the central processors used by the Saturn V and Apollo programs are to be identical, certain changes must be made. Since the Apollo configuration appears to be the most restrictive case, the bulk of the changes must be made to the Saturn V central processor. The Apollo configuration is limiting because the form factor, weight, power, and system clock accuracy are more critical than in the case of Saturn V. However, in some areas such as the cooling mode, it is possible to go either way; therefore, a choice must be made to assure that the central processors are identical.

For Apollo, a standby mode is required; during this mode only time is updated. To conserve power, all computer functions not related to updating time must be switched off. Since the clock, bit gate, and phase generators are located in the central processor, and since these logic functions are required to update time, provisions must be made to isolate (powerwise) the circuits associated with generating these functions. The change embodying this concept has already been authorized by MSFC on the Saturn V central processor.

Another Apollo mission requirement is that the system clock should neither lose nor gain more than one second during the total mission time. This is a much more stringent requirement than for Saturn V; consequently, a new oscillator section must be developed for Apollo. If the Apollo and Saturn V Computers are to be identical, this new circuit must be incorporated in the Saturn V CP.

The Apollo guidance and navigation system requires certain discrete frequencies from the computer. To generate these frequencies with a minimal amount of hardware, the basic computer oscillator frequency for Saturn V would have to be changed from 2.0 mc to 2.048 mc. Since this small frequency change has a relatively minor effect on the central processor hardware, MSFC has directed IBM to incorporate this change on Saturn V.

Two alternate types of cooling systems have been described in section IID (cold plate and integral cooling). The present Saturn V design uses integral cooling, while the majority of the electronic equipment in Apollo, including the present guidance computer, use cold plate cooling. The proposed Apollo computer can accommodate either type of cooling; however, a decision must be made based on the relative advantages and disadvantages of each type.

For the most part, the Apollo command module uses a Hughes rack and panel connector type, while Saturn V uses Bendix connectors. Since the connectors are not accessible in the Apollo application, a rack and panel type is a necessity. Therefore, IBM feels that the Saturn V central processor could use the Hughes-type connector with possibly, Hughes connectors where the central processor mates with the data adapter and Bendix connectors where the system ties to the data adapter.

Form factor appears to be a less critical item for the Saturn V central processor since adequate space is available in the instrument unit. In contrast, form factor and volume are extremely tight in the Apollo command module. Therefore, it appears to be feasible to attach brackets to the Apollo central processor for mounting between the structural channels in the Saturn instrument unit.

Section III
RELIABILITY

Section III

RELIABILITY

A. GENERAL

The estimated probability of success, or reliability, of the proposed Apollo Guidance Computer described in this report is based on a 336-hr. mission. The reliability estimate is:

$$R = 0.9953$$

The mission profile definition* places the computer in the operate mode for 54 hours and in the standby mode (only time-keeping equipment operating) for the remaining 282 hours.

The Guidance Computer reliability figure represents two major pieces of equipment:

- Central Processor (and power supplies)

$$R = 0.9971$$

- Data Adapter

$$R = 0.9982$$

The predicted reliability of the computer can also be expressed in terms of "effective MTF" (mean-time-to-system failure). "Effective MTF" is defined as the operate mode MTF that would be required for a nonredundant computer to achieve the same reliability as the redundant computer for the mission profile under consideration. For the Apollo Guidance Computer, the "Effective MTF" is 46,000 hours.

Since the recommended computer is a redundant system, the MTF is not equal to the mean-time-between-component failures (MTBF), as is the case for a nonredundant system. The MTBF indicates how long one could expect the computer to sit on the launch pad, or in storage, without a component failure occurring. The MTBF for the computer in the energized state is approximately 650 hours, and approximately 1200 hours while the computer is de-energized.

* The mission profile used for this analysis is shown in Appendix A (Figure A.1). It is derived from a North American Aviation memo No. 454-110-63-112 Guidance and Navigation System 14-Day Power Profile.

IBM feels that the reliability of the guidance computer analyzed herein is such as to allow NASA to achieve a high over-all reliability for the Apollo Mission.

The computer reliability estimate presented here is pessimistic due to the effect of certain assumptions which were made to facilitate the analyses. These assumptions are as follows:

- All three memory modules must complete the entire mission.
- If single component failures are present in both units of a dual-redundant memory, the guidance computer has failed.
- Any failure in the processor clock-driver circuitry causes failure of an entire channel of triple modular redundant binary logic.
- The failure rate of mechanical parts (solder connections, printed circuit boards, tape cables, etc.) is constant whether the computer is energized or unenergized.

The first two assumptions are explained more fully under Subsystem Analyses (Section III D). The third assumption is very pessimistic as will be noted in Section III, D. 1. The last assumption is based on the results of thorough failure analyses performed on defective parts which were removed from other systems presently in production.

These analyses reveal many failure mechanisms which are independent of electrical stress levels, particularly for mechanical parts. However, the quantity of available data is insufficient to attach any statistical confidence to the assertion that the mechanical-part failure rates are constant. In addition, the available data are mostly from the Titan II Missile Guidance Computer which does not use ULD's or multilayer interconnection boards of the type used in the proposed Apollo Guidance Computer. These two components for which the Titan II data are not directly applicable contribute a high percentage of the total computer failure rate. Thus, the failure rates for the mechanical parts used in the Apollo Computer may actually be lower for the unenergized state than they are for the energized state.

On the average, the failure rates employed in this study for the high-usage electronic components (resistors, transistors, diodes, etc.) in the de-energized state are approximately 15 percent of the respective failure rates in the energized state. This fact, coupled with the invariant mechanical-part failure rate previously discussed, results in a de-energized computer failure rate which is approximately 55 percent of the computer failure rate in the energized state.

This slightly less than 2:1 ratio of failure rates in the energized versus de-energized state seems to be rather pessimistic since it is common practice in industry to either assume ratios in the order of 1000:1 or use a failure rate of zero for the de-energized state. However, the available failure data suggests that the 15 percent factor for electronic components is correct. Also, the only justifiable prediction that can be made, at present, for the mechanical components is that the rates are invariant.

By recognizing that the assumption regarding the mechanical components may be pessimistic and that the reliability of the proposed Apollo Guidance Computer is heavily dependent on the de-energized failure rates (since the system is de-energized for approximately 84 percent of the mission duration) the implication of reduced de-energized failure rates is noteworthy. If, as is the common practice, one were to assume that the de-energized system failure rate is zero, the reliability of the Apollo Guidance Computer would be:

$$R = 0.9980$$

One final point of interest is that if a nonredundant version of the proposed Apollo Guidance Computer were to be employed, its reliability for the mission described herein would be approximately 0.8373. Stated another way, the unreliability of the nonredundant computer is 0.1627 as compared to the 0.0047 unreliability of the proposed redundant computer. Thus, an increase in component count by a factor of approximately 3.5 has caused a decrease in unreliability by a factor of 35.

The following subsections contain both a general method of analysis for use when evaluating a redundant system and also the analysis of the specific system described in this report. The detailed component counts and failure rate summations for the various subsystems are contained in Appendix A. Appendix B contains a mathematical justification of the technique used for predicting the reliability of the triple modular redundant logic.

B. METHOD OF ANALYSIS

The following terms will be used quite frequently in this analysis and are defined here for the sake of convenience.

- Reliability - The probability that the equipment being considered will perform properly for a specified period of time under a given set of operating conditions.
- Mutually exclusive events - A set of events are mutually exclusive if the occurrence of one of the events precludes the occurrence of any of the others. If the events within a set are mutually exclusive,

the probability that any one of them will occur is equal to the sum of the probabilities of all the events in the set. For example, the two events "resistor fails" and "resistor works" are mutually exclusive.

- Independent events - Events are independent if the probability of the joint occurrence of all the events is equal to the product of the probabilities of the individual events occurring. Practically speaking, various equipments are independent if a failure in one does not cause failure in the others.
- Failure Rate - The probability that a failure will occur in the next interval of time, provided that a failure has not already occurred, divided by the length of time interval.¹ If many components are placed on test (failures are replaced as they occur), the failure rate represents the percentage of components that fail per time interval.
- Component - "Component" as used herein was the same meaning as "component part".

The first step in the analysis is to divide the system into smaller subsystems. A failure in one subsystem does not cause failures in any others; therefore, the subsystems are independent of each other. For the mission to be a success, all subsystems must operate. The reliability of the system is equal to the probability that all the subsystems perform, and since they are independent, system reliability is determined as follows:

$$R_{\text{sys}} = R_1 \times R_2 \times \dots \times R_n$$

where

R_1, R_2, \dots, R_n = the reliabilities of the various subsystems.

The second step in the analysis is to obtain mathematical expressions for the reliabilities of those subsystems which contain redundancy (i. e., the subsystem circuitry is such that several component part failures can occur and not fail the subsystem). These subsystems are analyzed by first considering that they are combinations of simplex modules (modules containing no redundant elements; one component part failure fails the module). One specifies a set of mutually exclusive events; the occurrence of any of the events will yield subsystem success. Thus, subsystem reliability is the probability of one of the events occurring and since the events are mutually exclusive:

$$R_{\text{sub}} = P(\text{event A}) + P(\text{event B}) + \dots$$

1. Precisely, the limit of this quantity as the length of the time interval approaches zero.

where events A, B, are events such as "simplex module works" or "simplex module fails to a high state." The problem of computing redundant system reliability has therefore been reduced to evaluating probabilities for simplex modules.

Simplex module evaluation is the third step. If one assumes a constant failure rate during the entire mission for each component type in the simplex module, reliability is computed as follows:

$$R = \exp (- \lambda T_M) \quad (1)$$

where

λ = Simplex module failure rate or,

λ = Sum of the failure rates of all components in the module

T_M = Mission Time

If the operating conditions vary significantly during the mission, the assumption of a constant component failure rate is not realistic. Consequently Eq. 1 is not valid. A valid method of obtaining simplex module reliability is to separate the mission into a number of phases. Operating conditions vary from one phase to the next, but are assumed to be fixed within any one phase. (The various operating conditions are stated in Appendix A.) Each component type is assigned a number of failure rates; one for each different operating condition encountered.

The simplex module failure rate for mission phase j is then:

$$\lambda_j = \sum_{i=1}^k n_i \lambda_{i,j} \quad (2)$$

where

n_i = number of type i components in the module.

k = number of different component types in the module

$\lambda_{i,j}$ = failure rate of type i components when exposed to the operating conditions of mission phase j.

The failure rate for the simplex module as a function of time might then look like Figure III-1.

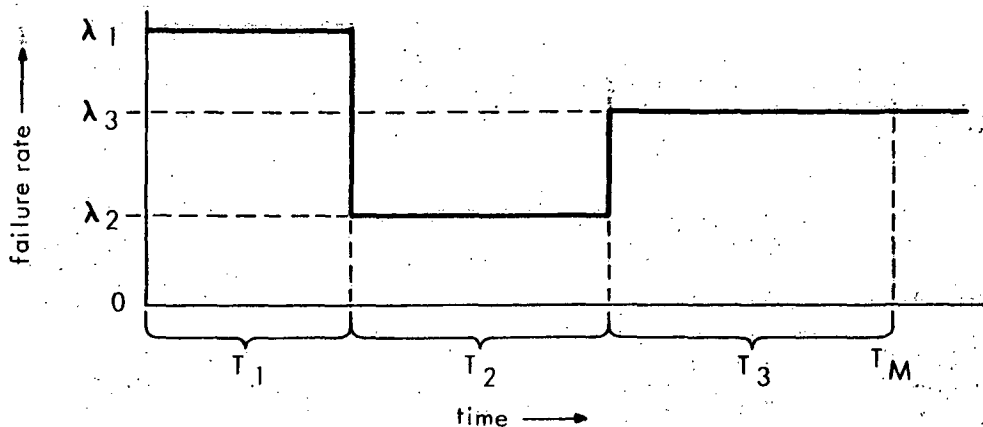


Figure III-1. Simplex Module Failure Rate

T_1 , T_2 , and T_3 would be the time durations for mission phases 1, 2, and 3 respectively.

Generally speaking, the probability of a simplex module performing properly during phase j , providing no failures have occurred prior to the start of phase j , is

$$R_j = \exp(-\lambda_j T_j)$$

where

λ_j = as defined by Eq. 2 (the simplex module failure rate for phase j).

T_j = the time interval for phase j .

For a simplex module to perform properly during the entire mission, it must successfully complete all phases. The probability of working through all the phases is the reliability of the simplex module for the entire mission and, assuming a failure rate as depicted by Figure III-1;

$$R_M = \prod_{j=1}^m R_j$$

or

$$R_M = \exp \left(- \sum_{j=1}^m \lambda_j T_j \right) \quad (3)$$

where

R_M = probability of the simplex module working properly during the entire mission.

λ_j, T_j = as defined above.

m = number of phases into which the mission is divided.

Equation 3 is employed frequently in the analyses that follow. For the sake of convenience, it will be referred to as the "simplex reliability equation."

It is sometimes necessary to compute the probability of a failure occurring. For any one phase, the probability of a module failing is $(1 - R_j)$. If it is necessary to evaluate the probability of a particular mode of failure (i. e., an inverter failing to a "logical one"), one must analyze the circuitry of the module under consideration. Since this varies from situation to situation, the analyses will be described as the need arises. However, once the failure probability of interest has been calculated for each of the phases, one can evaluate the failure probability for the mission by recognizing that mission failure can occur in the following mutually exclusive ways:

- failure in phase 1
- work in phase 1 and failure in phase 2
- work in phases 1 and 2, fail in phase 3
- work in phases 1 through $m - 1$, fail in phase m

The probability of a failure occurring during the mission is the sum of the probabilities for these events:

$$P(f)_M = P(f)_1 + R_1 P(f)_2 + \dots + R_1 R_2 \dots R_{m-1} P(f)_m \quad (4)$$

If the failure of interest is simply failure of the simplex module (without any concern about mode of failure) then,

$$P(f)_j = 1 - R_j$$

and Equation (4) reduces to

$$P(f)_M = 1 - R_M \quad (5)$$

If a particular mode of failure is of interest, each term of Equation (4) must be evaluated. Again for convenience, Equation (4) will be referred to hereafter as the "modal failure equation." For those situations where probability of failure is of interest (no modal considerations) Equation (5) will be employed.

C. SYSTEM MODEL

Figure III-2 shows the 6 subsystems into which the Central Processor and Data Adapter are divided in this analysis. Since the subsystems are independent, and since all of them must operate when needed for the mission to be a success, the reliability of the Guidance Computer in terms of the subsystem is:

$$R_{sys} = R_{cl} \times R_{mem} \times R_{ps} \times R_{dal} \times R_{daio} \times R_{chr}$$

where

R_{sys} = Guidance Computer Reliability

R_{cl} , R_{mem} , R_{ps} , R_{dal} , R_{daio} , and R_{chr} are, respectively, the reliabilities of the processor logic, memory, power supply, data adapter (triple modular redundant, i. e., TMR) logic, data adapter input-output circuitry, and chronometer.

The Central Processor consists of the processor logic, memory, and power supply so that,

$$R_{proc} = R_{ps} \times R_{mem} \times R_{cl}$$

The Data Adapter contains TMR logic, input-output circuitry, and the chronometer.

$$R_{DA} = R_{dal} \times R_{daio} \times R_{chr}$$

Summarizing the results of the analyses contained in the following sections of the reliability portion of this report,

- $R_{cl} = 0.9987$
- $R_{mem} = 0.9984$

- $R_{ps} > 0.99999$
- $R_{dal} = 0.99929$
- $R_{daio} = 0.99944$
- $R_{chr} = 0.99950$

Performing the previous multiplications,

$$R_{proc} = 0.9971$$

$$R_{DA} = 0.9982$$

and

$$R_{sys} = 0.9953$$

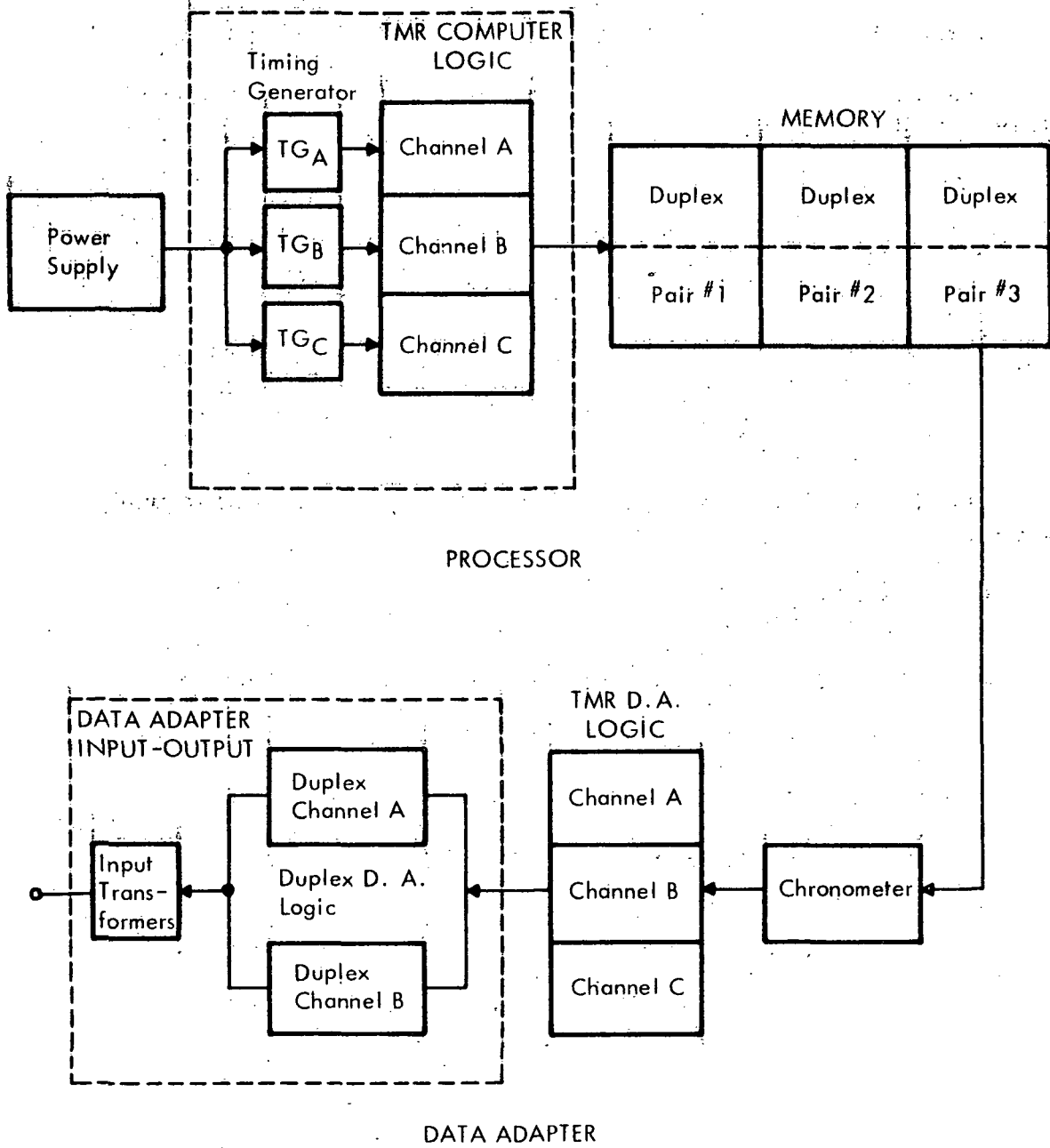


Figure III-2. System Model

D. SUBSYSTEM ANALYSES

1. CENTRAL PROCESSOR BINARY LOGIC

The processor binary logic may be divided into simplex modules as shown in Figure III-3.

Let

R_{tmrl} = reliability of the TMR portion of the binary logic

R_{tg} = reliability of a timing generator module

R_{sl} = reliability of one channel of TMR logic operating as a simplex machine (i. e. , the probability of exactly zero failures in one channel of the TMR logic)

The computer logic can operate successfully in the following mutually exclusive ways:

- All three timing generator modules operate and the TMR logic operates.
- Two timing generator modules work, one fails and the two associated channels of the TMR logic operate as simplex machines. (This event can occur in three ways.)

This rule for success can be expressed mathematically as:

$$R_{cl} = R_{tg}^3 R_{tmrl} + 3R_{tg}^2 (1-R_{tg}) R_{sl}^2 \quad (6)$$

The above expression for the reliability of the processor logic is very pessimistic in that it assumes that all failures in the clock drivers (timing generator module) cause an entire channel of the logic to fail.

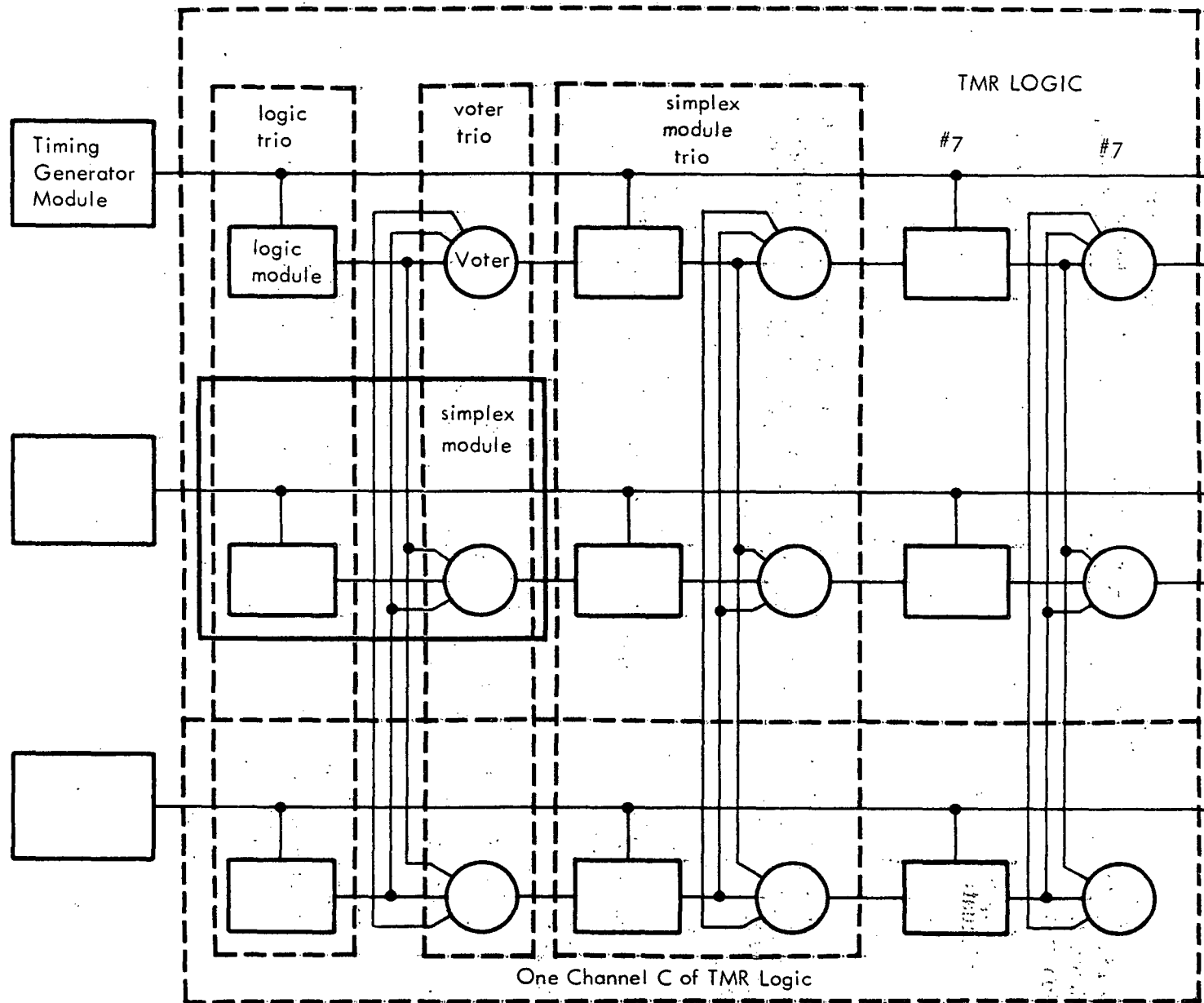


Figure III-3. Processor Binary Logic Reliability Model

Since there are numerous drivers, one for each portion of a simplex channel, failure of a single driver would only fail a portion of the simplex channel. However, the present assumption is made to facilitate the formulation of a mathematical model for the logic.

R_{tg} and R_{sl} are reliabilities for simplex modules and may, therefore, be evaluated with the "simplex reliability equation":

$$R_M = \exp \left(- \sum_{j=1}^m \lambda_j T_j \right)$$

R_{tmrl} describes a redundant (TMR) configuration. One method of computing R_{tmrl} is to recognize that the TMR logic can be divided into simplex module trios. Trio success can occur in the following mutually exclusive ways:

- All three simplex modules operate.
- Two modules work and one fails. (This can occur in three ways.)

Thus

$$R_{trio} \cong (R_{mod})^3 + 3(R_{mod})^2 (1-R_{mod})$$

or, rearranging terms

$$R_{trio} \cong 3R_{mod}^2 - 2R_{mod}^3$$

(The equation is approximate because the events above do not describe all the ways that success can occur. An attempt to describe them all precisely would involve literally thousands of terms. The approximation is slightly pessimistic.)

If there were n independent identical trios,

$$R_{tmrl} = (R_{trio})^n \cong (3R_{mod}^2 - 2R_{mod}^3)^n \quad (7)$$

R_{mod} is the reliability of a simplex module thus allowing evaluation by the "simplex reliability equation."

Equation 7 yields an estimate for the reliability of TMR logic. Unfortunately, it implicitly assumes, among other things, that the relative placement of majority voters has no impact on the TMR network under consideration. Experience has shown, however, that variations in voter placement can have a severe impact on the TMR reliability. Insofar as the Processor logic is quite complex, the optimum voter placement (approximately 160 voters used) is difficult to determine. A program has been written for the IBM 7090 EDPS using a Monte Carlo technique to estimate R_{TMR} . This method has the advantage of simulating the actual logic being designed. This allows for determination of optimum voter placement and evaluation of changes in circuit configuration. Also, many of the assumptions implicit in Equation 7 are eliminated.

The Monte Carlo program performs the following steps to generate the system reliability: (See Appendix B for a more detailed discussion of the Monte Carlo program.)

- **Component Failure Generation** — For each component type, the probability of failure as a function of time is inserted into the 7090 computer. Through a simple routine involving a random number process, a set of failures is generated for each type of component. The number of failures is dependent upon the component failure rate, the number of each type component in the system and the mission time. A second random number process, weighted by the conditional probability of open (K_0) for each type component, is used to determine which of the component failures are opens and which are shorts.
- **Logic Block Failures** — The failed components are randomly assigned to various logic blocks. The program recognizes those logic blocks which contain failed components and decides if the block has failed to a logical one or zero. In cases where a block has more than one failed component giving a possibility of output failure to one or zero, the output state which is most harmful to system performance is assumed.
- **Failure Trace** — Having established which modules have failed and to what state, the program traces the effect of the failed modules on other module inputs and outputs. If the distribution of failed modules is such as to cause two of the three inputs to a majority voter element to fail in the same direction (i. e., logical "one" or "zero") the system has failed.

Each repetition of these three steps is called a "game", and a typical system evaluation might consist of playing five to ten thousand games.

The program finally calculates the reliability of the TMR machine. The equation used is:

$$R = \frac{\text{Number of successful games}}{\text{Total number of games}}$$

In practice, Equation 7 is employed to obtain an estimate of R_{tmr1} in the early stages of the development program when the logic circuitry is not well defined. Once circuit definition becomes relatively fixed the Monte Carlo program is used to optimize voter placement and obtain a more precise estimate of R_{tmr1} .

The logic configuration for the Processor is well defined. Equation 6 is therefore evaluated using values of R_{tmr1} from the Monte Carlo program (R_{sl} and R_{tg} from the simplex reliability equation). The estimates for the processor binary logic are:

$$\begin{aligned} R_{tg} &= 0.9966 \\ R_{sl} &= 0.9573 \\ R_{tmr1} &= 0.9995 \end{aligned}$$

which when inserted into Equation (6) yields

$$R_{cl} = 0.9987$$

As previously mentioned, this figure is pessimistic due to the assumption that a clock driver failure disables an entire channel of logic.

2. MEMORY

To evaluate the dual-redundant memory reliability the following operating conditions are assumed to apply:

- Both nonredundant memories will be operating at time zero (i. e. , both memories will be going through the same program and will be updated).
- Only one nonredundant memory output will be used by the computer starting at time zero.

- If the nonredundant memory whose output is being used by the computer system fails, the failed memory output will be disabled for all words and the computer will use the output of the other nonredundant memory if it has not failed at the same word address.
- If the second nonredundant memory fails, the computer will use the output of the first nonredundant memory if there is no failure at the same word address. Operation will continue in this manner with the computer using one nonredundant memory output until a failure occurs and then switching to the other.
- If a failure occurs in one of the nonredundant memories, the correct output from the working nonredundant memory will be used to regenerate the failed memory (i. e. , following every read there is a regeneration cycle wherein the information is written back into the cores. Normally, with no failures, the output of each nonredundant memory is used to regenerate itself). If a transient failure occurs in one of the memories and some of its information is destroyed due to the failure, the working memory will completely regenerate the memory in which the transient failure occurred.

Use of dual-redundant memories where more than one failure mode is possible at the nonredundant memory output, necessitates the use of some form of failure detection. The failure detection circuitry, in addition to detecting failures, must be capable of identifying the memory in which the failure occurred. To accomplish this, parity checking and half-select current monitoring will be instrumented into each nonredundant memory.

Half-current select monitoring is performed by the Error Detection circuit. This circuit checks for the following:

- Absence or presence of X or Y half-select current and proper timing thereof.
- Presence of half-select current in more than one X or in more than one Y drive line.

This circuit is expected to detect all first failures in the EI drivers, X-Y decoupling, Memory Timing (i. e. , MCD-1 and MCD-2 circuits) and the X-Y connection circuitry.

Parity checking will monitor the memory output for odd parity. Analysis of the memory operation indicates that parity checking will detect all first failures in the Sense Amplifiers, cores, inhibit drivers and memory buffer registers, and variable strobe gate. With odd parity checking in a word consisting of 13 bits plus one parity bit at least one half of the TCV, X-Y Terminating Resistor and Error Detection circuit first failures will be detected. Therefore the parity checking and Error Detection circuitry will detect all first failures in the nonredundant memory with the possible exception of TCV, X-Y Terminating Resistors, and failures in the Error Detecting circuit itself. The parity checking circuit assists or supplements the Error Detection circuit in carrying out its function in that many of the failures detected by the Error Detection circuit will also be detected by parity checking (i. e., an open X or Y connection in the array during a read operation).

It is possible for compensating failures to occur in the Inhibit Drivers, cores, Sense Amplifiers and Memory Buffer Registers which a parity check would not detect. Compensating failures in the above circuits would be those failures where a "0" and "1" bit in a particular word at a given address is changed to a "1" and "0" respectively at the memory output and at the same address. This result requires that at least two or more circuit failures occur in the above circuits (i. e., Sense Amps, Inhibit Drivers, Memory Buffer Register and cores). The probability of compensating failures is a function of the length of time between the first circuit failure and the first interrogation of the circuit following the failure. This probability is very small and will be neglected in the reliability computations.

Assumptions for the memory analysis are:

- The reliability of the parity checking circuitry is 1.0. The output of each nonredundant memory goes to TMR parity checking (i. e., two TMR parity checks). Each set of TMR parity check logic is ultra reliable as compared to its nonredundant memory; thus no significant error will be introduced in the memory analysis by excluding the parity check success and failure events.
- All failure events in the X-Y Terminating Resistors, Temperature Controlled Voltage and Error Detection circuits are classed as nondetectable failures. Nondetectable failures are those failures whose detection by parity checking is a function of chance. For example, a failure which results in more than one word being addressed may by chance

give either correct or incorrect parity (i. e., odd number of "1's" at the memory output). The probability of detecting this type of failure is ≥ 0.5 ; the value 0.5 will be used.

The Error Detection circuit is included in this category since a failure in this circuit may prevent failure detection in other memory circuits. To class all failure modes of these circuits as nondetectable is slightly pessimistic.

- All other memory circuit failures which are not categorized in the preceding assumption are categorized as being detectable failures. These are failures whose detection is not a function of chance. For example, parity checking will detect all failures which result in failure of a single bit in a word.
- The amount of success in the events which say that both memories have failed is zero. Provided that each memory has a failure, the conditional probability of success is approximately 0.2. This factor, weighted with the probability of both nonredundant memories failing, results in a negligible contribution to the total probability of success. This assumption results in a slightly pessimistic reliability number.
- Conditional probability of compensating failures is zero for those circuits having failure events categorized as detectable. This assumption is valid if memory off, or idle, time is not excessive. For missions during which the memories will be idle for significant periods IBM assumes that the idle memories will be exercised periodically to ensure that the probability of compensating failures is small.

In the analyses, memory (A) will be the memory whose output is used by the computer starting at time zero. Memory (B) will be the other non-redundant memory.

Given the preceding assumptions, the following mutually exclusive events will yield success:

- Memory (A) works for the entire mission. Memory (B) can work or fail.
- The first failure in Memory (A) is a nondetectable type failure which by chance is detected by parity check. Memory (B) works for the entire mission.

- o The first failure in Memory (A) is a detectable type failure which is detected (i. e. , parity checking circuitry and Error Detection circuitry works). Memory (B) works for the entire mission.

$$P(\text{first event}) = R_{sm, M}$$

$$P(\text{second event}) = R_{sm, M} \times C(x) \times P(f)_{nd, M}$$

$$P(\text{third event}) = R_{sm, M} \times P(f)_{d, M}$$

where

$R_{sm, M}$ = reliability of one nonredundant memory for the entire mission.

$C(x)$ = probability that a nondetectable failure is by chance detected by parity check. ($C(x) \geq 1/2$ (the value 1/2 is used.)

$P(f)_{nd, M}$ = probability that a nondetectable failure occurs in Memory (A) at some time during the mission.

$P(f)_{d, M}$ = probability that a detectable failure occurs in Memory (A) at some time during the mission.

Since the three events are mutually exclusive, the reliability of one dual-redundant memory is:

$$\begin{aligned} R_{dm} &= R_{sm, M} + R_{sm, M} \cdot P(f)_{nd, M} \cdot C(x) + R_{sm, M} \cdot P(f)_{d, M} \\ &= R_{sm, M} (1 + P(f)_{d, M} + C(x) \cdot P(f)_{nd, M}) \end{aligned} \quad (8)$$

$R_{sm, M}$ is evaluated using the "simplex reliability equation". $P(f)_{nd, M}$ and $P(f)_{d, M}$ are modal failure probabilities and are evaluated using the "modal failure equation" with:

$R_j = R_{sm, j}$ = the probability of a nonredundant memory working through phase j given that it has not failed prior to the start of phase j .

and

$$P(f)_{nd,j} = \frac{\lambda_{nd,j}}{\lambda_{mem,j}} (1 - R_{sm,j})$$

$$P(f)_{d,j} = \frac{\lambda_{d,j}}{\lambda_{mem,j}} (1 - R_{sm,j})$$

where

$P(f)_{nd,j}$ = probability of nondetectable failure in phase j provided that no failures have occurred prior to phase j.

$P(f)_{d,j}$ = probability of detectable failure in phase j provided that no failures have occurred prior to phase j.

$\lambda_{nd,j}$ = failure rate during phase j of circuitry whose failures are categorized as nondetectable.

$\lambda_{d,j}$ = failure rate during phase j of circuits whose failures are categorized as detectable.

$\lambda_{mem,j}$ = total memory failure rate for phase j.

Note:

$$\lambda_{nd,j} + \lambda_{d,j} = \lambda_{mem,j}$$

$\frac{\lambda_{nd,j}}{\lambda_{mem,j}}$ = the probability that the first failure is nondetectable, provided that a first failure occurs in phase j.

$\frac{\lambda_{d,j}}{\lambda_{mem,j}}$ = the probability that the first failure is detectable, provided that a first failure occurs in phase j.

Substituting into the "modal failure equation",

$$P(f)_{d,M} = \frac{\lambda_{d,1}}{\lambda_{mem,1}} (1 - R_{sm,1}) + R_{sm,1} \frac{\lambda_{d,2}}{\lambda_{mem,2}} (1 - R_{sm,2}) \\ + R_{sm,1} R_{sm,2} \frac{\lambda_{d,3}}{\lambda_{mem,3}} (1 - R_{sm,3}) + \dots$$

$$P(f)_{nd, M} = \frac{\lambda_{nd, 1}}{\lambda_{mem, 1}} (1 - R_{sm, 1}) + R_{sm, 1} \frac{\lambda_{nd, 2}}{\lambda_{mem, 2}} (1 - R_{sm, 2}) \\ + R_{sm, 1} R_{sm, 2} \frac{\lambda_{nd, 3}}{\lambda_{mem, 3}} (1 - R_{sm, 3}) + \dots$$

These two values and $R_{sm, M}$ are substituted into Equation 8 to obtain R_{dm} .

R_{dm} is the reliability of one dual-redundant memory. Generally, more than one memory will be used. Assuming that dual memories are independent of each other,

$$R_{mem} = R_{dm1} \times R_{dm2} \times R_{dm3} \times \dots$$

where R_{mem} is total memory reliability and R_{dm1} is the reliability of dual memory module 1 and so on. If the mission is such that dual memory one is used only for the first one-third of the mission, then R_{dm1} should be calculated noting that the "specified time period" is $T_M/3$. Since the memory use is not well known at present, it will be pessimistically assumed that all dual memories must complete the entire mission. Therefore,

$$R_{mem} = (R_{dm})^n$$

with

R_{dm} = reliability of one dual-redundant memory for the entire mission.

n = number of dual memories used in the configuration being studied.

For the present mission profile

$$R_{sm, M} = 0.9849$$

$$P(f)_{d, M} = 0.014410$$

$$P(f)_{nd, M} = 0.000751$$

and, substituting in Equation 8

$$R_{dm} = 0.99946$$

Three memory modules are employed so that

$$R_{mem} = (R_{dm})^3 = 0.9984$$

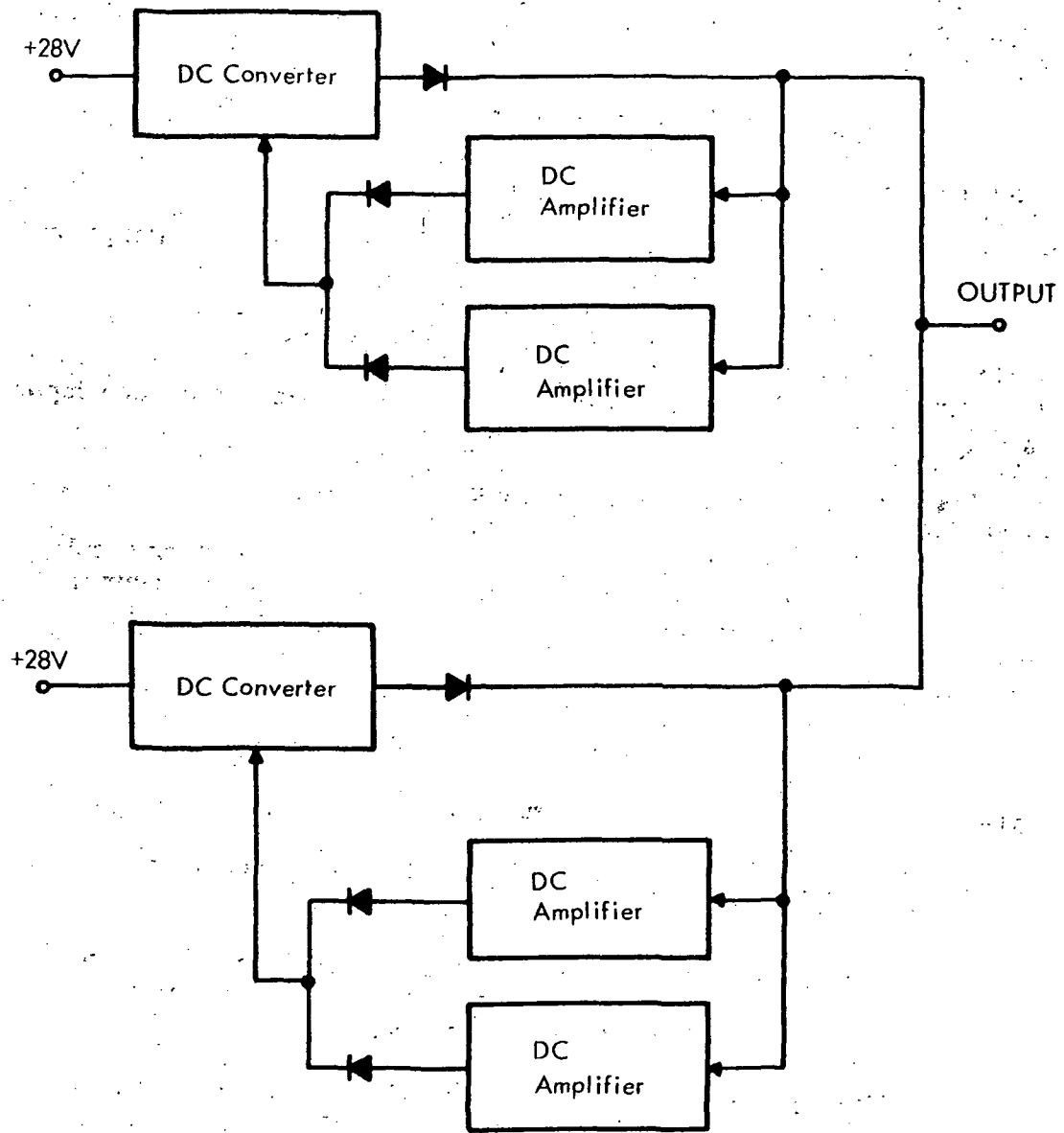


Figure III-4. Power Supply Configuration (simplex module)

3. POWER SUPPLY

The power supplies which furnish the various voltage levels required have identical simplex module configurations as shown by Figure III-4. The supplies are independent of each other and all of them must work. Power supply reliability is

$$R_{ps} = (R_{dup, M})^n$$

where

$R_{dup, M}$ = reliability of one dual-redundant or duplex power supply for the entire mission.

n = number of duplex supplies used.

The following mutually exclusive events yield success for one duplex supply:

- Both supplies work for the entire mission.
- One converter output fails to a down state; the other supply operates for the entire mission. (This event can occur in two ways.)

Thus

$$R_{dup, M} = (R_{ss, M})^2 + 2R_{ss, M} P(f)_{cl, M} \quad (9)$$

where

$R_{ss, M}$ = probability that a single supply operates properly for the entire mission.

$P(f)_{cl, M}$ = probability that a converter output fails low and remains in the failed low state for the duration of the mission.

Each power supply consists of two single supplies in a redundant configuration. The reliability of the duplex supply is given by Equation 9. However, a single supply is not simplex in that it contains redundant feedback amplifiers. So $R_{ss, M}$ and $P(f)_{cl, M}$ still describe redundant configurations. These terms can be evaluated through the usual process of stating mutually

exclusive success events. However, the computations involved are rather laborious. The following considerations lead to simplification:

- All component failure modes in the converter cause the converter output to fail low.
- The converter output fails high if both feedback amplifiers fail low.

Therefore,

$$P(f)_{ch,M} = (P(f)_{al,M})^2$$

where

$P(f)_{ch,M}$ = probability that a converter output fails high at some time during the mission.

$P(f)_{al,M}$ = probability that a d-c amplifier output fails low at some time during the mission.

Two pessimistic assumptions are made at this point, to simplify the analysis:

- The reliability of a single supply, $R_{SS,M}$ is computed with the "simplex reliability equation" assuming that the single supply is a simplex machine (this ignores the fact that the d-c amplifiers are duplexed).
- The most damaging mode of a single supply failure is converter output high (or, as mentioned, both amplifiers down). Therefore, it is assumed that all failures in the d-c amplifier will cause a down failure.

$P(f)_{cl,M}$ is obtained by recognizing that

$$(1 - R_{SS,M}) = P(f)_{ch,M} + P(f)_{cl,M}$$

or, rearranging terms and substituting

$$P(f)_{cl,M} = (1 - R_{SS,M}) - (P(f)_{al,M})^2$$

Due to the second assumption

$$P(f)_{al,M} = 1 - R_{amp,M}$$

where

$R_{amp, M}$ = reliability of a d-c amplifier for the mission.

This yields

$$P(f)_{cl, M} = (1 - R_{SS, M}) - (1 - R_{amp, M})^2$$

Substituting into Equation 9

$$R_{dup, M} = (R_{SS, M})^2 + 2R_{SS, M} \left[(1 - R_{SS, M}) - (1 - R_{amp, M})^2 \right] \quad (10)$$

Application of the "simplex reliability equation" yields

$$R_{SS, M} = 0.99937$$

$$R_{amp, M} = 0.99988$$

Recognizing that four supplies are required and substituting into Equation (10),

$$R_{ps} = (R_{dup, M})^4 > 0.99999$$

Therefore, even with the pessimistic assumptions made in this analysis, the unreliability of the power supply is insignificant when compared to system unreliability.

4. DA BINARY LOGIC

High reliability is achieved for the binary logic portion of the Data Adapter (DA) by using triple modular redundancy. This scheme has already been described in section D1. Although an approximate component count is available for the DA logic, the exact circuit configuration is not available. Therefore, as mentioned in section D1, the reliability is calculated using

$$R_{dal} = (3R_{mod}^2 - 2R_{mod}^3)^n \quad (11)$$

Since the DA can be divided into six functional groups, this equation will be evaluated under the assumption that one simplex channel of the DA consists of six approximately equal (in failure rate) simplex modules. The number so obtained constitutes a lower bound on the true reliability.

$$R_{mod} = 0.9937 \text{ (using the simplex reliability equation).}$$

Substituting into Equation (11)

$$R_{dal} = 0.99929$$

5. DA INPUT-OUTPUT CIRCUITRY

The output portion of the DA contains 46 dual-redundant output drivers. Each of the drivers are essential for mission success and are independent of each other, so

$$R_{out} = (R_{od})^{46}$$

R_{out} = reliability of output portion of DA

R_{od} = reliability of an output driver.

The configuration of an output driver is shown in Figure III-5.

The transformer outputs are combined with an "OR" circuit.

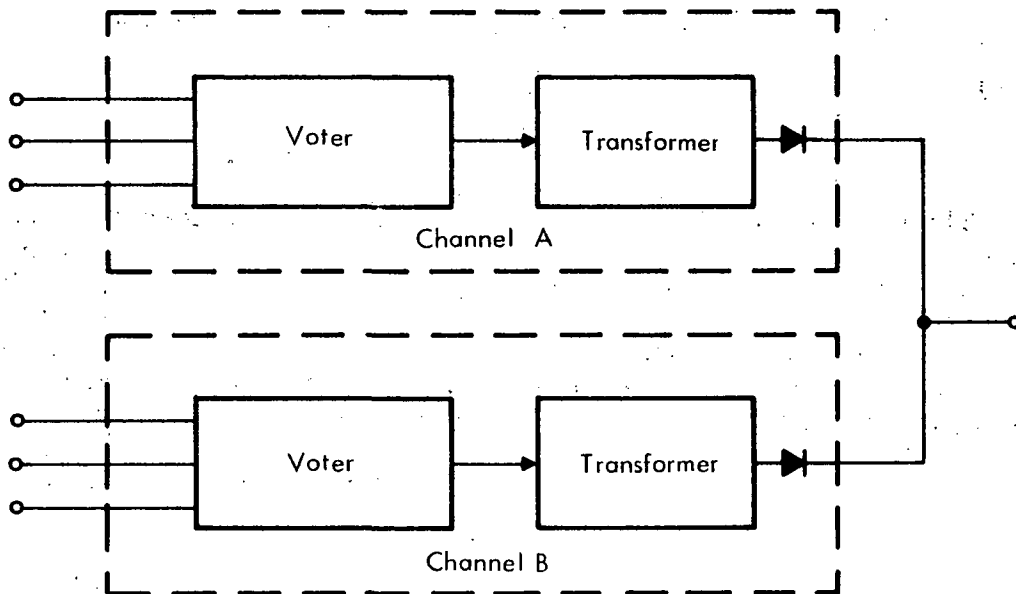


Figure III-5. Output Driver Configuration

Success occurs for this circuit in two mutually exclusive ways:

- Channels A and B work for the entire mission.
- One of the two channels fail, the other works for the entire mission. (This event can occur in two ways.)

The probabilities for these events are:

$$P \text{ (first event)} = (R_{ch})^2$$

$$P \text{ (second event)} = 2R_{ch} (1 - R_{ch})$$

$$R_{ch} = \text{reliability of a channel; evaluated with the "simplex reliability equation".}$$

Summing,

$$R_{od} = 2R_{ch} - R_{ch}^2$$

This analysis is based on the observation that the conditional probability of occurrence is small for those component part failures which would cause an up-level failure at the channel output.

For the mission profile defined herein,

$$R_{ch} = 0.99974 \text{ (using the "simplex reliability equation")}$$

Substituting,

$$R_{out} = (R_{od})^{46} > 0.99999$$

The input portion of the DA consists of 42 pulse transformers. These transformers are fed directly by external equipment (such as accelerometers, resolvers, etc.) whose reliability may not be high relative to the reliability of the transformers. If this is the situation, the over-all space vehicle reliability will not be significantly enhanced by making the transformers redundant. The decision as to whether to make the transformers redundant should be made on the basis of a comparison between the relative failure rate of each transformer and the failure rate of the equipment feeding the transformer. For this analysis, IBM assumes that the transformers will not be redundant.

The reliability of the input portion of the DA is, then, evaluated using the "simplex reliability equation".

$$R_{ixfmr} = 0.99945$$

The reliability of the DA input-output subsystem is then

$$R_{daio} = R_{ixfmr} \times R_{out} = 0.99944$$

6. CHRONOMETER

As opposed to all the subsystems discussed thus far, the chronometer (equipment which keeps track of real time) must be energized for the entire mission. Thus simplex module reliabilities will be calculated using the energized failure rates for the entire mission. The chronometer contains triple modular redundant logic which forms the timing pulses for the Computer and DA, and accumulates real time. It is assumed that a simplex channel of this TMR accumulator logic will be broken into three smaller modules before applying redundancy. Using the "simplex reliability equation" (energized failure rate for entire mission)

$$R_{sa} = 0.9930$$

where

$$R_{sa} = \text{reliability of an accumulator simplex module.}$$

Using Equation 7, the reliability of the TMR accumulator is

$$R_{acc} = (3R_{sa}^2 - 2R_{sa}^3)^3 = 0.99985$$

The chronometer also requires one power supply of the type described in Section D4. Re-evaluating to account for the fact that this supply is energized for the entire mission:

$$R_{ss} = 0.9985$$

$$R_{amp} = 0.99979$$

$$R_{chrps} > 0.99999$$

where

$$R_{ss} = \text{reliability of a single supply}$$

$$R_{amp} = \text{reliability of a feedback amplifier}$$

$$R_{chrps} = \text{reliability of chronometer power supply}$$

To achieve the drift accuracy required of the Apollo oscillator (one part per million for ten days) the oscillator must be temperature compensated. The complexity of the compensated oscillator will most likely dictate that a simplex version is not reliable enough.

The design of a redundant oscillator having the required accuracy is a difficult task. IBM assumed, for this analysis, that the oscillator is of the triple modular redundant configuration. It is also assumed that the failure rate of a simplex oscillator is 30×10^{-6} . Given these assumptions, the oscillator reliability would be 0.99966. (Obtained with Equation (7), $n=1$)

The reliability of the chronometer is then,

$$R_{chr} = R_{acc} \times R_{chrps} \times R_{osc} = 0.99950$$

Since the accumulator, chronometer power supply, and oscillator are independent and must all work for mission success.

E. FAILURE RATES AND K-FACTORS

1. COMPONENT-PART FAILURE RATES AND FAILURE MODES

Table III-1 lists the failure rates used in all the analyses shown in this report. Predicted conditional probabilities of open failure (K_O) are shown for those components where this parameter is significant for the analyses. % E. S. is the percent of rated electrical stress and λ is failure rate in failures per million hours.

The component-part failure rates used by the Space Guidance Center are based primarily on failure rate data obtained from the operation of IBM designed and produced systems. Thus, the essential details regarding these data are known factors and can be used in extrapolating these data for varied conditions or new component part types. The extent of this knowledge is shown (for the principal types of component parts) in billions of component-part operating hours in Table III-2. Detailed failure rate data for some of the component-parts in the Titan Missile Guidance Computer Program are presented in Table III-3. Under this program all removed parts are thoroughly analyzed in a special laboratory facility to learn the precise mechanism of failure. Knowledge thus gained materially aids in extrapolating observed failure rates for other conditions. Consider, for example, the task of extrapolating failure data to obtain failure rates for de-energized component-parts. Here two factors materially aid in this work. First, the results of the thorough failure analysis of removed component-parts reveals which failures were independent of electrical stress (only present during the energized state). These failures would just as likely have occurred during the de-energized state. Consequently, total duration of stress application would be used in conjunction with the observed number of failures of this type to calculate the failure rate for the de-energized condition. Second, a large scale test, now comprising over 1.4 billion component-part hours, is being conducted on systems under storage conditions. This is proving to be a valuable source of failure rates for component-parts under de-energized conditions.

Table III-1

COMPONENT-PART FAILURE RATES

Component Types	Energized			De-energized	
	%ES	λ	K_0	λ	K_0
Transistors					
1. Leadless	≤ 10	0.012	0.38	0.0023	0.56
2. Leadless - matched pair	≤ 10	0.036		0.0048	0.56
3. Sil. planar - in stitch welded can	≤ 10	0.011		0.0036	
	≤ 50	0.017			
4. Same as 3 - matched pair	≤ 10	0.033		0.0076	
	≤ 50	0.051			
5. Sil., alloy, power	≤ 50	0.14	0.1	0.0036	0.36
Diodes					
1. Dual Leadless - half used	≤ 10	0.007	0.46	0.0011	0.59
2. Dual Leadless - both halves used	≤ 10	0.006/ half	0.46	0.0010/ half	0.59
3. Zener discrete	≤ 50	0.06	0.33	0.003	0.4
4. Sil., planar, micro	≤ 10	0.008		0.0008	0.5
5. Sil., power rectifier	≤ 50	0.1	0.1	0.0015	0.3
Resistors					
1. Cerment (ULD type)	≤ 30	0.013	0.72	0.001	0.9
2. Metal film, precision	≤ 30	0.022	0.99	0.003	0.99
3. Molded carbon comp., nonhermetically sealed	≤ 30	0.003	0.99	0.003	0.99
4. Variable trimmer	≤ 10	0.15		0.03	
Capacitors					
1. Glass	≤ 10	0.001	0.99	0.00004	0.99
2. Ceramic	≤ 30	0.015	0.5	0.00005	0.9
3. Tantalum, solid-section	≤ 50	0.06	0.1	0.0014	0.5
Connections					
1. Unit or Page body active pins/pair	—	0.003	1.0	0.003	1.0
	—	0.007	0.8	0.0007	
2. Flow solder	—	0.001	1.0	0.00028	0.99
3. Hand solder, memory frame		0.0005	1.0	0.00036	0.99
4. Solder fillet (ULD)		0.001	1.0	0.001	0.99
5. Core, toroidal, T-38		0.0001		0.0001	
6. Cable, flexible, tape/length		1.0	0.9	1.0	0.9
7. Choke, filter, power		0.12		0.002	

Table III-1. Component-Part Failure Rates (cont)

Component Type	Energized			De-energized	
	%ES	λ	K_0	λ	K_0
Connections (cont)					
8. Choke, R. F.		0.1		0.002	
9. Crystal oscillator		0.5		0.003	
10. Delay line, glass		0.3		0.0025	
11. P. C. strip, memory		0.0001		0.0001	
12. MIB (1 page side)		0.553	0.80	0.553	0.99
13. MIB (back panel)		3.762	0.80	3.762	0.99
14. Signal transformer		0.43	0.99	0.004	0.88
15. Power transformer		0.7	0.99	0.004	0.88
16. Wire, memory (per wire)		0.0001		0.0001	
17. Resistor, minco, temp. -sensing		0.001		0.001	0.95
18. Pulse transformer		0.16			
19. Chip connection - URD/ball		0.0005	1.0	0.0005	1.0
20. Single-sided connection - ULD conductor pattern		0.0001	0.60	0.0001	0.60
21. H-clips including solder- WAL & WAL		0.0005	1.0	0.0005	1.0
22. Connectors		0.689		0.689	
23. Substrates		0.0001	1.0	0.0001	1.0
24. Connections; wrap-around lands, ULD		0.0005	1.0	0.0005	1.0
25. Hand solder-memory address wire		0.0002	0.99	0.0002	0.99
26. Sense or Inhibit		0.00028	0.99	0.00028	0.99
27. Splice		0.00036	0.99	0.00036	0.99
28. AVERAGE Substrate + lands + comp-ball-joints + solder fillet joints		0.0246	0.7	0.0246	0.7
Substrates (ULD)					
1. Inverter		0.0824			
2. Type AA AND		0.0724			
3. Type AB AND		0.0810			
4. TMV		0.1188			
5. VIN		0.0922			
6. HCI		0.0960			
7. CLN		0.0617			
8. CDN		0.0356			

Table III-2

**TOTAL COMPONENT PART
OPERATING HOUR EXPERIENCE**

Name of Component Part	Billions of Component-Part Operating Hours
Capacitors	3.36
Connections, Solder	31.70
Connections, Welds	5.00
Connections, Other	1.30
Connectors	12.60
Cores	15.70
Diodes, Over-all	62.40
Electrical Parts, Miscellaneous	66.00
Resistors, Over-all	15.64
Transistors, Over-all	42.95
Other	3.50
Compilation Over-all Total	260.15

Because the Space Guidance Center has such a large amount of component-part failure rate data and an extensive knowledge of the many factors that pertain to these data, IBM feels that the use of such data results in more realistic reliability predictions than if component-part failure rates of highly questionable applicability were taken from published tables such as MIL-HDBK-217. This handbook was produced by RCA and is essentially the same as Section 8 of the RADC Reliability Notebook which RCA produced for the Air Force. These are updated versions of RCA's TR59-416-1 and their earlier TR-1100. The failure data for these publications was gathered mainly from field experience on a ground-based data link system using vacuum tubes and linear amplifiers. This equipment was designed, for the most part, prior to 1958. A small portion of this failure rate data was obtained from the airborne operation of a system similar to the data link system.

Recently small amounts of additional information, principally on semiconductors, was obtained from life tests conducted by RCA, Battelle Memorial Institute, and various manufacturers. These life test data were usually obtained at maximum rated stress conditions and consequently would shed little light on the failure rate of such devices under normal usage conditions where

Table III-3

**COMPONENT-PART FAILURE RATES OBSERVED ON THE TITAN MISSILE
GUIDANCE COMPUTER PROGRAM**

Name of Component Part	No. of Fail.	Millions of Component-Part Hours	Failures/10 ⁶ Component Hours
Capacitors			
Ceramic	0	30.385	0.0329*
Glass	0	14.047	0.0712*
Mica	0	1.911	0.5233*
Paper	0	0.341	2.9325*
Tantalum	1	11.399	0.0877
Total	1	58.083	0.0172
Diodes			
Si., D.J., Rectifier	0	3.572	0.28 *
Si., D.J., Sw.	0	20.937	0.0476*
Si., P.C., Sw.	0	135.088	0.0074*
Si., D.J. & P.C., Insep. Combnd.	3	201.058	0.0149
Si., Zener	5	68.815	0.0727
Si., Rectifier	0	4.153	0.2408*
Total	8	433.623	0.0184
Relays			
6 Pole, Sw.	8	0.515	15.534
2 Pole, Sw.	0	0.783	1.277 *
Time Delay	1	0.113	8.85
Total	9	1.411	6.378
Resistors			
CC, Fixed, H.S.	2	282.926	0.0071
MF, Fixed, H.S.	1	21.735	0.046
WW, Fixed, H.S.	0	26.799	0.0373*
WW, Fixed, Power	1	1.092	0.916
WW, Fixed, Precision	1	9.191	0.1088
WW, Variable Trimmer	0	7.159	0.1397*
Total	5	348.902	0.0143
Transistors			
Ge, D.M., Sw.	6	52.563	0.1141
Si., A.J., Power	0	2.68	0.373 *
Si., D.M., Power	0	0.64	1.5625*
D.M., Sw.	1	35.671	0.028
Si., G.J., Sw.	0	2.471	0.4047*
Si., Power	0	1.251	0.7994*
Si., Sw.	0	4.65	0.215 *
Total	7	99.926	0.07

*Indicates that no failure has occurred as yet. The value shown is what the rate would be if one had failed.

the stresses are kept to a minimum to obtain the best component-part reliability. MIL-HDBK-217 provides no means of accounting for weld or solder junction failure rates which can be critical factors in the reliability of a system. This handbook also states that the minimum failure rate is $0.001\%/1\text{K hrs}$ (0.01×10^{-6}) for any component part. Failure rates of 0.001×10^{-6} an order of magnitude lower, have been frequently demonstrated with a high degree of confidence. The range of parts covered is also limited and could not be expected to cover special types of component parts such as those contemplated for this equipment. For example, of the 37 different component-part types planned for this equipment, only 18 were included in MIL-HDBK-217. While this represents approximately 50 percent of the part types involved, it probably constitutes less than 1 percent of the total component-part population because these 18 types are low usage items.

It would be impossible to intelligently extrapolate MIL-HDBK-217 failure rates since it lacks necessary background data such as:

- The number of failures observed.
- The number of component-part hours observed.
- The failure mechanisms observed.
- The failure modes observed.
- The definition of a failure.
- The type, vintage, and construction of the component part.
- The amount of burn-in or preaging received.
- The screening given the component part.
- The type of failure analysis made.
- The level of electrical stresses other than dissipative.
- The amount of degradation experienced.
- The level of incoming inspection.
- The quality control level at the manufacturer's plant.

- The degree of cleaning and inspection after storing or welding.
- The circuit type in which the component part was used.
- The protection given to the component part by its circuit and system packaging.
- The thermal factors associated with packaging and operation.
- The length of time the component parts operated.
- The degree of data censoring.
- The level of system development at which data were obtained.
- The amount of debugging time.
- The number of systems produced and from which ones data were obtained.
- The level of the reliability program.
- The application of the system, its environment and other factors.
- The quality and availability of test equipment for field operation.

The handbook provides no means for calculating failure rates for the de-energized state. Thus, the user often assumes that the failure rate is zero during this period. A little reasoning will show that this is fallacious since all failure-inducing stresses except electrical (with usually, a reduction in temperature) are still present in the de-energized state and do actually cause failures.

In summary, IBM feels that the best source of failure rate data is data observed on the Titan Missile Guidance Computer. This source more closely approximates the use conditions expected than any other available data and more detailed characteristics are known (i. e., failure modes, environments, stresses, etc.) The Titan data is then modified to account for differences in usage conditions. The resultant failure rates are then compared with the larger quantities of data available from all sources and appropriate adjustments made if required.

2. K-FACTORS

The confidence level associated with the failure rates for many component parts operated under ground environments is high because much data has been collected for this case. As time progresses, the amount of these data increases as does the degree of refinement in the data collection process. Thus, failure rates for specific types of transistors (operated under certain conditions) are known as opposed to a failure rate for transistors, in general, as was the case just a few years ago. This knowledge is based on many billions of transistor operating hours and is further enhanced by detailed information on the device, its operating condition, and its mechanism of failure (determined through thorough failure analysis of the failed item).

There is no corresponding kind of knowledge regarding the failure rates of component parts during vehicle launch conditions. There is a definite need for such information, but because of the problems associated with obtaining these data, such as: 1) defining the levels and duration of mechanical stresses applied to the component part; 2) the uncertainty as to which part failed and why (since parts are usually not recovered); and 3) for the relatively few vehicles launched (and fully monitored), the data have not been obtained. In lieu of this information at the component part level, several attempts have been made to derive an over-all K-Factor, applied at the system level, which modifies the ground condition system failure rate to account for launch conditions. The approach has been to develop the K-Factor from failure rates (or MTBF's) observed for a number of different systems operated under both ground and launch environments. It is highly questionable that factors obtained in this manner are applicable for predicting the performance of a new system. The proneness to failure under applied mechanical stress would not be the same for the component parts used in the new system nor would be the level of applied mechanical stresses. Consequently, K-Factors were developed for this program on the basis of engineering judgment after considering what effect the estimated mechanical stresses such as vibration, shock, and acoustic noise might have in inducing failures in the component and structural parts employed in this system. As revised data becomes available on the frequencies, levels, and durations of applied mechanical stresses at the part level, re-evaluation of the effects on these parts could be undertaken.

Because approximately 60 percent of the system failure rate is attributable to hardware, and stage burn times are as shown in Table III-4, an average K-factor of 50 is calculated for launch. A K-factor of 15 is used for re-entry (the same value as used by IBM for Gemini analysis).

Table III-4

K-FACTORS

	Electronic Comp. K-Factors	Hardware K-Factors	Time (sec)
Stage 1 burn	36	100	150.0
Stage 2 burn	26	73	400.0
Stage 3 burn	17	47	480.0

Section IV
SYSTEMS AND
PROGRAMMING

Section IV
SYSTEMS AND
PROGRAMMING

A. STUDY GOALS

The goals of this study in the programming and systems area were as follows:

- Determine the feasibility of the Saturn V Guidance Computer performing Apollo guidance problem.
- Compare the speed and capacity of the AGC-4 Apollo Guidance Computer with the IBM Saturn V Guidance Computer.
- Identify the areas in which double precision computations are required.
- Identify and program in detail any high speed I/O requirements.
- Specify the necessary I/O equipment.
- Identify software requirements and outline available or easily modifiable existing programs.

Two basic ground rules were established based upon the following primary considerations:

- Compatibility between IBM computer in Saturn and Apollo.
- Time limitation of study.

The ground rules which resulted from these considerations were:

- No central computer changes in interrupt, word length, accumulator overflow, or memory partitioning.
- The memory capacity of 6 modules was assumed to be sufficient.

As a result of this study, IBM arrived at the following conclusions:

- **Speed** - The AGC 4 is slightly faster except during high I/O pulse activity.
- **Storage** - The effective storage capacity of the Apollo and Saturn computers is roughly equal. The AGC-4 being favored by better sub-routine linkage and shorter data word storage; while the Saturn V leads in addressing efficiency and a larger basic instruction set.
- **Feasibility** - The Saturn V with the data adapter (Section II C.) can functionally perform the Apollo guidance problem.
- **Software** - Three basic programming and debugging tools will be developed. They are:
 - (1) **Assembler**
 - converts symbolic inputs into machine language programs
 - allows definition of macro instructions
 - allows editing feature
 - provides master tape output
 - provides post-processor data for debugging and analysis
 - detects programming errors.
 - (2) **Simulator**
 - provides symbolic outputs
 - allows symbolic program changes
 - checks for errors (overflow, illegal division etc.)
 - (3) **Logic Simulator**
 - provides capability to include switching delays
 - provides capability to include detection of circuit failures
 - allows either functional or detailed logic description of computer logic

B. INPUT/OUTPUT PROGRAMMING

1. GENERAL

The purpose of this section is to detail the high speed programming requirements of the Apollo and LEM input/output areas. The input pulse rate from accelerometer does not constitute a high-speed computer load due to the data adaptor design which independently counts these inputs. The total computer time required for the computations described below, and further detailed in Figures IV-1 and IV-2, is 25 percent for the Apollo mission and 30 percent for the LEM mission. The necessary instructions for saving the registers requires 924 usec. in the Saturn V and 228 usec. in the AGC-4. *This overhead is included in the previous figures.*

The decimal display, downlink, command word and CDU constitute the only high-speed sustained computer load. Figure IV-1 and IV-2 are flow diagrams of the necessary logic and computations involved. The subsequent description will closely follow the contents of Figures IV-1 and IV-2.

2. DOWNLINK

The basic timing of all the high-speed inputs and outputs is set by the downlink since it has the highest rate. The downlink rate is 50 words/second. These words are divided into 10 groups of five words each, the group rate therefore being 10 groups/second. Each word contains 16 bits which is converted by the data adapter into a 40-bit format as follows:

- The 16 bits are transmitted twice.
- A word order bit (WOB) is transmitted eight times; this bit is a "zero" for the first word of each group and a "one" for the other four words.

The first word of each group is an identification word (See Figure IV-1, No. 12-25) denoting the content of the following four words. The identification format consists of:

- The group count denoting content of the third, fourth, and fifth words.
- Tag bits denoting contents of the second word.

K-BIT = 1 (keyboard word)

U-BIT = 1 (uplink word)

T-BIT = 1 (display or command word)

$\overline{\text{K-BIT}} \cdot \overline{\text{U-BIT}} \cdot \overline{\text{T-BIT}} = 1$ (a preselect memory location)

More than one of the tag bits may be set at once; therefore, a priority must be assigned. The K-BIT has top priority, U-BIT second and T-BIT last. As mentioned previously, the word order bit for the first word is set to "zero."

The second word in the group (See Figure IV-1, No. 26-29) consists of keyboard, uplink, display, command word or preselect memory data as specified by the tag bits in the first word. The word order bit is a ONE for words two through five.

The computer programming for the third, fourth and fifth word is identical (See Figure IV-1, No. 29). The group count (G) and word count (N) are used to address modify a "clear and add" (CLA) instruction. This results in selecting one of 30 preselected telemetry words.

Following the downlink activity for the second and fourth word of a group, the CDU control subroutine is entered (See Figure IV-1, No. 30, 33, 34). Following downlink of the fifth word, the display subroutine is entered (See Figure IV-1, No. 30, 31, 32).

3. DISPLAY AND COMMAND WORD

The display and command word activity occurs at 10 samples/second following the fifth word of each downlink group (See Figure IV-2, No. 1-14). The first four bits of the display word denotes address; binary codes 1 through 13 denotes display, codes 14 and 15 denotes command word, and code 0 denotes no activity.

The command words have priority (See Figure IV-2, No. 1-5). Tag bits are set by the main program to determine whether a command word is to be sent out. Bits 5 through 16 of the command word selects 11 relay drivers in the DA, which are automatically reset in approximately 20 msec. when the next downlink interrupt occurs. These relay drivers drive latching relays which require no standby power.

If no command words are ready, the program checks for display words. The display words are stored by the executive program in a first-in/first-out buffer list.

This list operates as follows (See Figure IV-2, No. 6-9). The executive program steps a pointer (PN) each time a quantity is added to the list. When the bottom of the list is reached, the pointer is reset and the next word is stored at the top of the list. The display routine maintains a pointer (PL) which denotes the position of the last quantity displayed. If the two pointers are not equal there are quantities to be displayed. These quantities are displayed at a 10/second rate if no command words are being processed, until the pointers are equal. Each display word contains the address of the display device and two 5-bit digit codes. These bits are processed exactly as the command word; selecting drivers for 20 msec, which set or reset latching relays.

4. CDU (CONTROL AND DISPLAY UNIT)

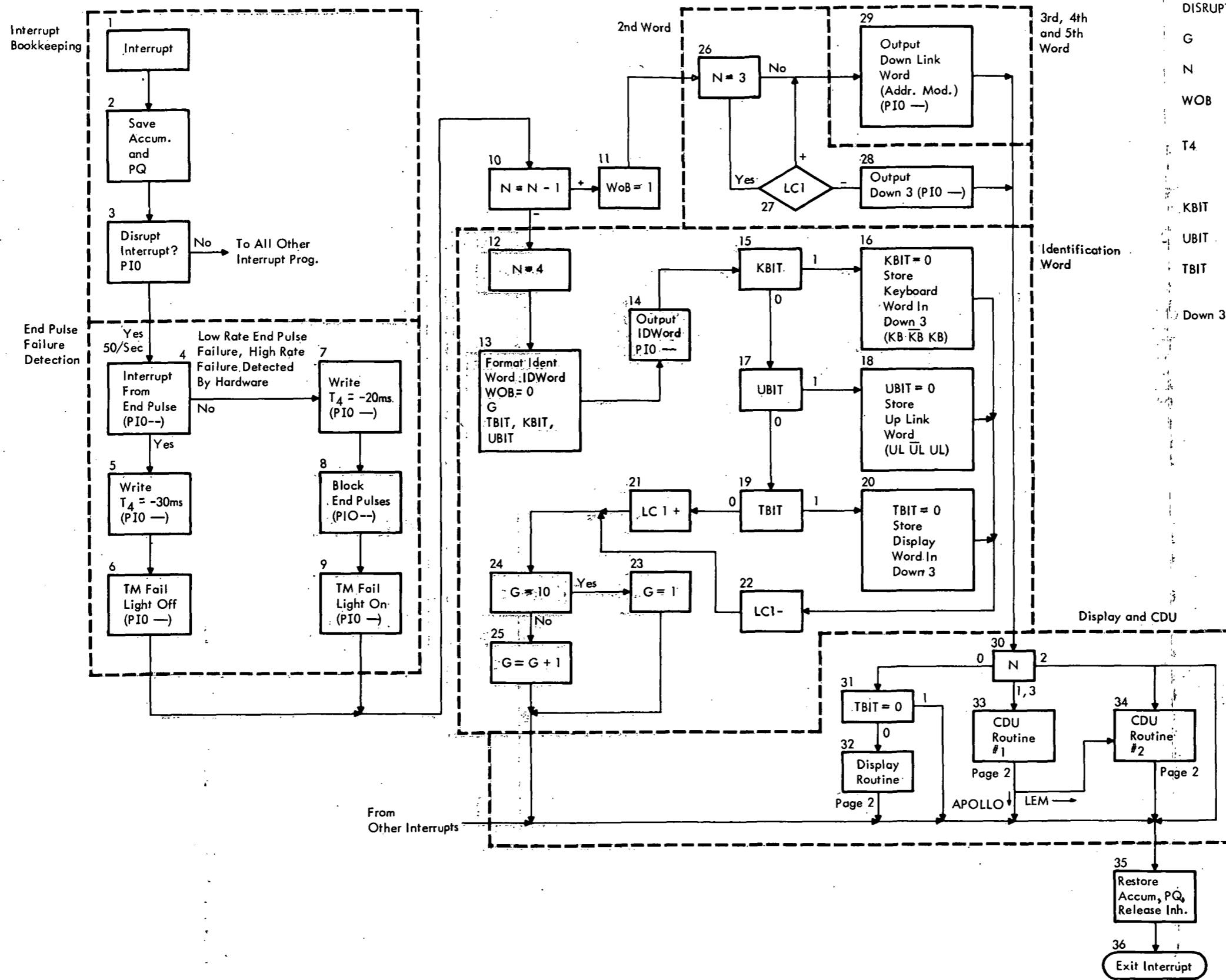
The gimbal angle stepping motors in the CDU require a 20 samples/second rate. This is approximated by performing the CDU subroutine (See Figure IV-2, No. 15-24) following downlink activity of the second and fourth word of each downlink group. The Apollo has three gimbal channels while the LEM has five. The operation of each channel is identical so that description of one will suffice.

The required pulse count (ΔX) is put on a delay line in the data adapter where it is diminished by 1 at a 3.2 KC rate. If pulses still exist on the line when the channel is again sampled, a busy signal will prevent writing on the delay line. This will only occur during the movement through a large angle. Each of the three channels for Apollo will be successively sampled; therefore, any one channel will be serviced every 150 msec.

The executive program provides a desired angle (X_D) to the CDU subroutine. This is differenced with the presently commanded angle (X_P) to determine the present error (ΔX_i). The error (ΔX) is then determined as a linear combination of the present error of the previous error (ΔX_{i-1}). This value is rounded to the nearest integer, and sent to the data adapter. Finally, the present commanded angle is updated by this amount ($X_O = X_O + \Delta X$). In the case of the LEM, two such operations occur during each entry to the CDU subroutine.

5. END PULSE FAILURE

Another requirement during this particular interrupt is the detection of a low rate of interrupt (See Figure IV-1, No. 4-9). The downlink equipment should provide an "end pulse" every 20 msec. to synchronize these computations. If the "end pulse" occurs at too high a rate, the data adapter will sense this and block further end pulses. At this time the T4 counter in the data adapter will provide 50-times-per-second interrupt signals. If "end pulse" comes too slowly, the T4 will also overflow and the program sensing

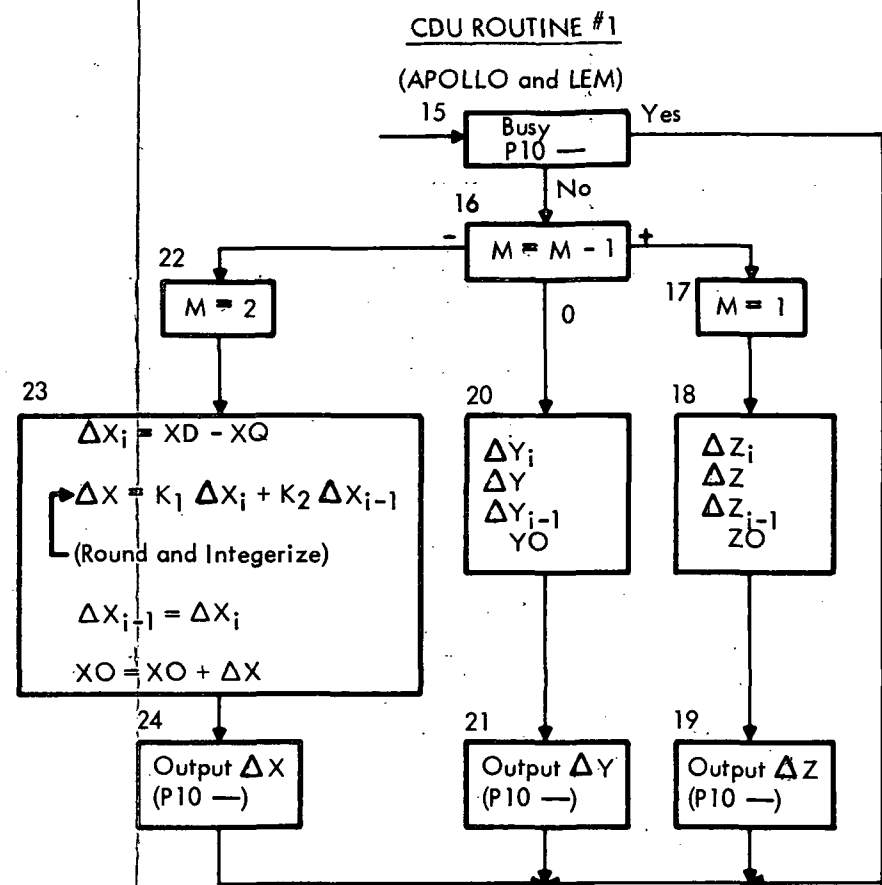
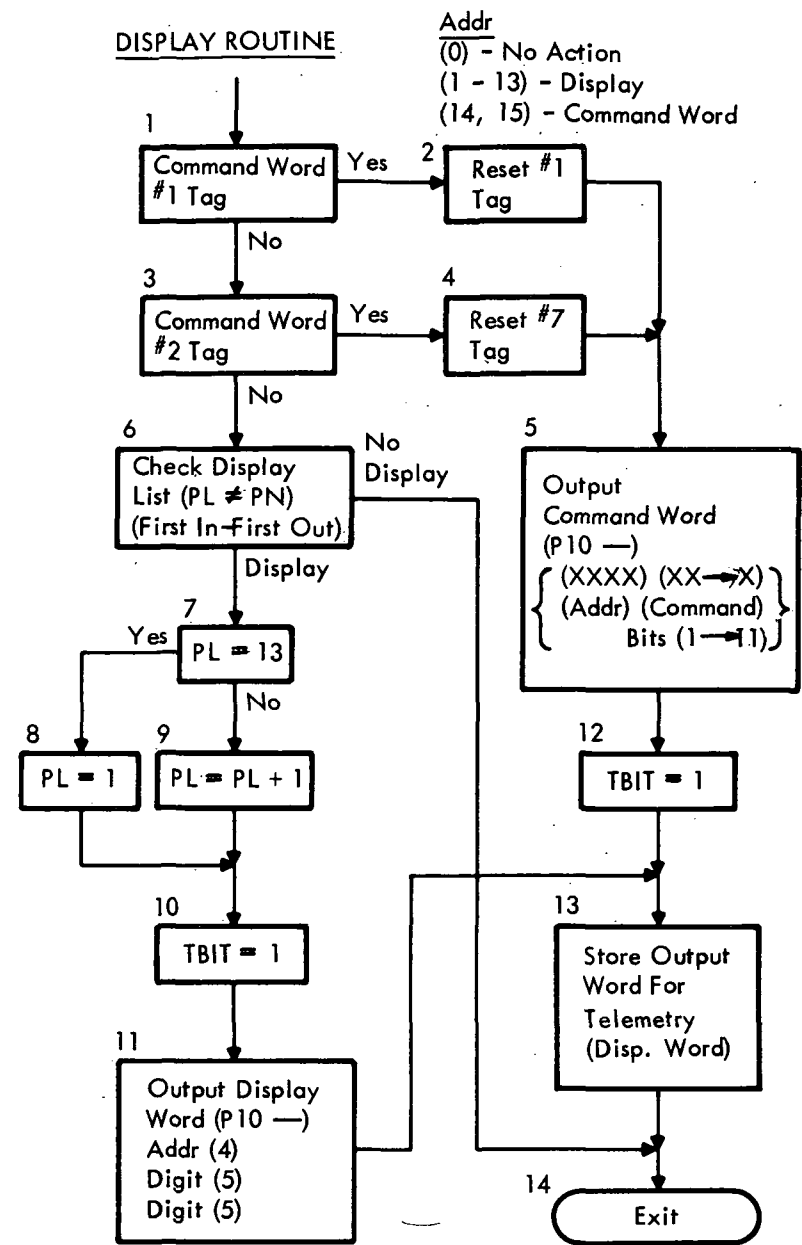


- DISRUPT - Interrupt From End Pulse or T4 Overflow
- G - Group Number (1 Thru 10)
- N - Sub Group Word (0 Thru 4)
- WOB - Word Order Bit (0 For 1st Word or Group, 1 For 2nd - 5th Word of Group)
- T4 - A Counter Which is Used to Detect End Pulse Failure and to Provide 50 Cycle Interrupt if Failure Occurs.
- KBIT - Denotes a Keyboard Insert has been Received
- UBIT - Denotes an Up Link Word has been Received
- TBIT - Denotes a Display or Command Word has been Outputted.
- Down 3 - Temporary Storage for Telemetry of Up Link, Keyboard, Display and Control Words.

DATA ADAPTER FEATURES

1. Programmable Inhibit Interrupt
2. Automatic Inhibit Interrupt When Turning On
3. Automatic Detection of High Rate End Pulse Failure
4. Discrete to Determine Cause of Disrupt (T4 Overflow or End Pulse)
5. Discrete Output to Block End Pulses
6. Separate Interrupts For
 - a. Up Link
 - b. Keyboard
7. A Programmable Timed Interrupt (0 - 80 Sec.)
8. Discrete Input From Counter Output Denoting "BUSY" With Previous Output
9. Automatic Reset of Relay Matrix at Beginning of Disrupt.
10. All Interrupt Bits are Available with One PIO; Disrupt is in the sign bit.

Figure IV-1. Apollo/LEM Downlink, Display, and CDU Math Flow Diagram



- PL - Pointer to Last Display Word Which has been Outputted
- PN - Pointer to Next Display Word to be Outputted
- TBIT - Set to 1 to Allow Display or Command Word to be Sent on Down Link
- M - Logic Control for CDU
- P - Logic Control for CDU
- ΔX_i - Present Gimbal Error
- ΔX - Change Sent to CDU
- ΔX_{i-1} - Previous Gimbal Error
- X_Q - Commanded Position of Gimbal Angle

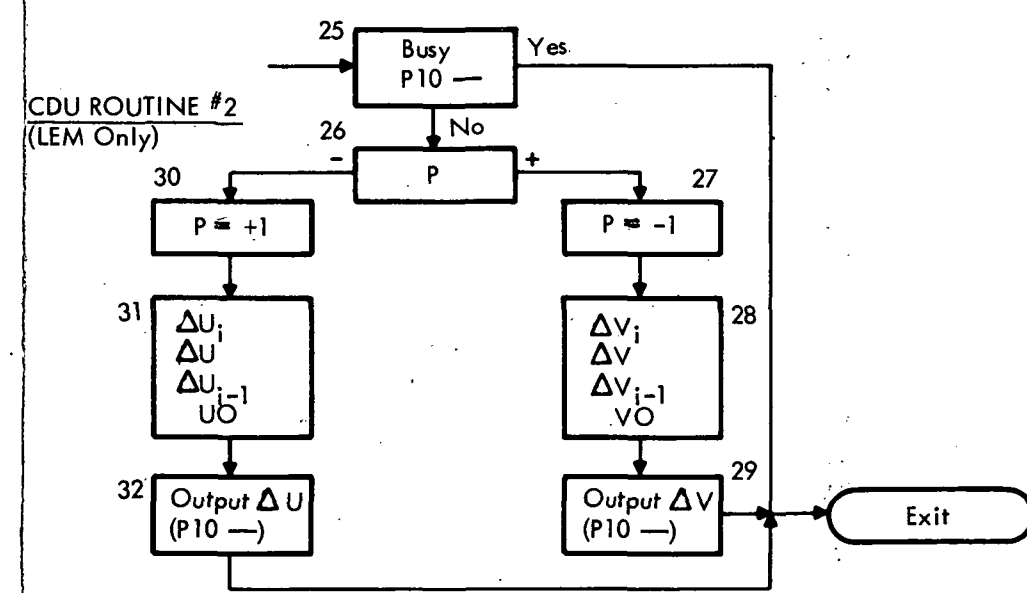


Figure IV-2. Apollo/LEM Display and CDU Subroutines

that this interrupt occurred via a T4 overflow will block further end pulses. The program, in either case, (i. e., high-rate or low-rate failure) will write every 20 msec. into the T4 counter to allow subsequent interrupts to occur 50 times/seconds.

6. INTERRUPT BOOKKEEPING

Also require at this rate is the interrupt bookkeeping (See Figure IV-1, No. 1, 2, 3, & 35). Upon sensing an interrupt, both the accumulator and PQ delay line contents are saved in fixed locations. Following the interrupt computations the accumulator and PQ are restored to their value at the time of the interrupt, and the interrupt inhibit is released.

7. OTHER I/O COMPUTATIONS

No other I/O function presents a significant speed requirement to the computer. The uplink and decimal insert functions occur at a low rate and are processed by separate interrupt routines. The accelerometer pulses previously mentioned are counted independently by the data adapter and need only be read once per navigation or thrust control computation cycle.

C. COMPARISON OF STORAGE AND SPEED OF AGC-4 AND SATURN V

The AGC-4 computer uses an internally programmed interpreter as a means of executing statements in polish form. A program can then be written in the polish form and the interpreter can determine each operation at execution time. This technique saves storage in some cases because more operations can be implemented for the interpreter and because temporary storage instructions are not usually necessary. However, this same technique costs storage in that 7 bits are required for each operation code and every address is a complete address capable of specifying any location in memory. In the Saturn V computer, memory partitioning is used as a storage saving technique. This saves storage because, in general, data can be confined to a small area of the computer memory and therefore long data addresses are not necessary.

In the programming comparison, both computers have certain areas in which they clearly have an advantage. The main advantage of the AGC-4 is its efficiency in linking to small subroutines because these subroutines can be defined as an operation to the interpreter. The Saturn V computer can execute most standard equations in fewer bits than the AGC-4, but subroutine linkage is more difficult than the linkage on the AGC-4. However, because of the more limited operation - code set and the shorter word length, the AGC-4 is forced to make subroutines out of functions which do not require subroutining on the Saturn V. An example of this is double precision. The AGC-4 is required to do most operations in double precision while the Saturn V will have to do a minimum of double precision operations.

The AGC-4 is much slower than the SATURN V when in the interpretive mode but is faster in the standard mode. Most computations, other than I/O, will probably require double precision which would cause the Saturn V computer to operate faster than the AGC-4. However, in the high-speed I/O requirements the AGC-4 has a large speed advantage.

The AGC-4 requires approximately 10 percent of the time to service downlink, display, command words, and CDU. The Saturn V computer requires approximately 25 percent of the time for these same functions. However the counter type inputs, (i. e., the accelerometer pulses) can cause a significant slowdown in the AGC-4 computer at the same time that maximum computation rates are required (i. e., ascent and re-entry). The implementation of the Apollo data adapter (pulse count bufferring on delay lines) is such that computer speed is independent of pulse rate input.

In view of the previous factors, IBM concluded that the AGC-4 is slightly faster than the Saturn V for the Apollo problem. The AGC-4 being approximately 3/1 faster in the I/O area but being slower in the computational area. The storage capacities are roughly equivalent, the AGC-4 being more economical in subroutine linkage and double precision computations, while the Saturn is more efficient for the following reasons or areas:

- (1) nonsubroutine computation
- (2) less double precision computations are required
- (3) a larger basic instruction set

D. DOUBLE REGISTER

Double-register operations can be programmed, if necessary, for the Saturn V without any hardware changes. There are two types of computation such as adding two full double-register words together and storing the result in a third word. The other kind, which would likely be more common is a 1-1/2 register addition. This is used when increment is computed for updating purposes. The scaling of the increment is different from the scaling of the full number, therefore all of the bits of the increment can be maintained and added to the full number as soon as bits ripple into high enough positions. Examples of a double-register addition and a 1-1/2 register addition are presented as follows:

● **DOUBLE REGISTER ADD**

$$Z = X + Y$$

NORMAL SEQUENCE

	CLA	XLO
	ADD	YLO
	STO	ZLO
	ADD	K1
	TMI	OVER*
	CLA	ZERO
**RET	ADD	XHI
	ADD	YHI
	STO	ZHI
	Done	

The low-order portion of the Double Register Quantities are shifted right one place. After addition, a difference in the two high-order places indicates that over flow has occurred. Adding 0100---0 to the sum yields a negative reset if over flow has occurred. If no over flow has occurred the low-order result is correct and the program simply adds the high-order parts.

*OVER	CLA	ZLO
	ERA	K1
	STO	ZLO
	TMI	*+3
	CLA	PLOBIT
	TRA	RET
	CLA	MLOBIT
	TRA	RET**

When control is transferred to "over", the computer must correct the low-order result and set the accumulator zero (all 0's or all 1's). For a positive or negative over flow, control is returned to "RET" for the high order addition.

● **1-1/2 REGISTER ADD**

CLA	XINCR
ADD	CINCR
STO	CINCR
ANA	MASK 1
STO	XINCR

CLA	CINCR
SFT	2
SFT	2
SFT	2
SFT	2
ADD	X
STO	X

CINCR is the computed increment. XINCR is the residue of the previous increments which was too small to be added in. Each time a new increment is computed it is then added to the residue. The portion that is large enough is added to X, after proper scaling, while the rest is kept in XINCR until the next computation of CINCR.

E. SOFTWARE

1. GENERAL

The plans for the Saturn V computer software are to build a flexible assembler with many debugging features; a simulator which will provide several different methods of program checkout, and a logic simulator. These tools are being designed to give the programmer as much assistance as possible without burdening him with details. The ideas used here are the result of experience gained in building and using such tools for the B70, Titan, and Gemini, computers. These same programs, designed for Saturn V, will be directly applicable to the IBM Apollo Guidance Computer.

2. ASSEMBLER

The assembly program will be a two-pass system which will include the following features:

- Editing
- Master Tape System
- MACRO's
- Post Processor Data
- Error Detection

The edit feature allows the programmer to reassemble a program and correct the program by editing correction cards with the master tape. This master tape, produced on a previous assembly, contains all the necessary information about an assembly to permit the following to be obtained at a later date: (1) extra copies of the program listing, (2) extra copies of the paper tape, (3) information necessary for simulator runs, and (4) the required information for future edits and assemblies to produce updated programs.

The macro capability is provided by a subroutine which was previously used on another assembler. Macro instructions are very useful on system computers since standard operations will be performed which require a fixed sequence of instructions with different parameters each time.

The post processor data contains a list of all symbols and the locations assigned to each symbol. In addition, the location of all references to each symbol. In addition, the location of all references to each symbol is provided. This enables the programmer to make informed decisions as to the proper locations for data quantities. In addition, lists of undefined and multiple-defined symbols are printed.

3. SIMULATOR

The simulator loads the assembled program from the assembler's master tape. In addition, the symbol list is loaded. This list allows the simulator to give the programmer symbolic output on the debugging information and allows the programmer to make symbolic corrections to the assembled program. The simulator is controlled by control cards which tell the program to trace, spot trace, or take snapshots.

The simulator also contains checks for programmer errors such as (1) cutting off a MPY or DIV before it is finished, (2) accumulator overflow, (3) illegal division and other such errors.

When a program is checked out using the simulator, the computer program simulator may be tied into a system simulator for complete system simulation.

4. LOGIC SIMULATOR

The development of a high-speed 7090 logic simulator will include the capability of simulating the logical operation of a computer with or without considering switching-circuit delays and/or failures. This simulator will be made up of a group of modules. Each module will represent a well-defined area of the machine. A control program connects and initiates the operation

of the modules in sequence, to enable the simulated computer to perform normal functions (run a program in machine language). At the beginning, most or all of each module will be simulated functionally. As each stage of logical design is completed, that portion will be replaced by logical simulation until all the logic is completed. There are areas which will remain in functional level; e.g., input-output devices, core memory, master oscillator, etc. An important criterion of the simulator is that the operating speed should be reasonably fast. In this way, liberal use of the simulator will be practical.

F. POTENTIAL HARDWARE CHANGES

A few hardware changes have been considered and the impact on programming and hardware has been estimated. One of these is the addition of overflow latches to facilitate double register computation. There may be no necessity for this change if few or no double register computations are required with the Apollo.

The following is an example of Double Register with the additional hardware.

CLA	XLO
ADD	YLO
STO	ZLO
ADD	XHI
ADD	YHI
STO	ZHI

This addition would cause the accumulator to be set to zero, (all 0's or all 1's) depending on the overflow latches after a STO operation. This means that the accumulator is destroyed after STO operations. However, in most cases the quantity in the accumulator is no longer needed once it has been put into memory.

The hardware involved in instrumenting this change would be approximately 39 additional ULD's using triple modular redundancy.

Another proposed change is automatic generation of a HOP constant every time a HOP instruction is executed. This would save time and storage in subroutine linkage, and also ease communication between the memory modules. The HOP constant would be stored in a register which is used only during multiply and divide operations. Upon entering a subroutine, the first instruction would be a STO 774. This would cause the contents of this register to be stored into memory location 774. Return from the subroutine would be via a HOP 774. This is a savings of three instruction times and

two full memory locations for every subroutine linkage. An additional saving would be realized on the interrupt routine. The interrupt overhead would be cut from 924 usec. to 672 usec. thereby increasing computation rate by 1.25 percent since there are 50 interrupts per second. The hardware involved in instrumenting this change is approximately 81 additional ULD's using triple modular redundancy.

Section V

LABORATORY TEST EQUIPMENT (LTE)

Section V

LABORATORY TEST EQUIPMENT (LTE)

A. INTRODUCTION AND PHILOSOPHY OF LTE SUPPORT TO AGC HARDWARE

To support the gradual build/up and check-out of Central Processors (CP) and Data Adaptors (DA) during final electrical assembly, test, sell-off, qualification tests, field demonstrations, and maintenance, certain pieces of support equipment for the back-up AGC will be needed. Figure V-1 a coarse, conceptual flow chart covering certain of these operations, illustrates the flow of events envisioned for this program.

1. **CENTRAL PROCESSOR (CP) HARDWARE TESTS (See Figure V-1, Sheet 1)**

This sheet of the flow diagram starts with the fabrication, assembly, and testing of the ULD's, MIB's, BIB's, frames, harnesses, etc. that make up the AGC. The components and subassemblies used in the gradual buildup of the CP would be tested, at various fabrication and assembly levels, on Factory Test Equipment (FTE) to assure that they meet all of their specifications. After the CP has been built and checked out by stages, and the Hardware Demonstration Program (HDP) has been run successfully, the CP's for delivery would be tested for compliance with the Interface Measurements, Temperature/Altitude, Operative Vibration, and Operative Life portions of the CP test specification. Any CP's intended for Flight Qualification Testing would be injected into the qualification testing operation after passing the Hardware Demonstration Test. The ACME, with suitable environmental chambers/machines, would be used during the Operative Environmental and Flight Qualification tests at the IBM Space Guidance Center. Since the Apollo Guidance Computer Evaluation Equipment (APOGEE) would be designed to perform all ACME functions in addition to its own normal functions, the APOGEE would be used for all testing operations on the CP hardware in the field.

2. **DATA ADAPTER (DA) HARDWARE TESTS (See Figure V-1, Sheet 2)**

Sheet 2 of the flow diagram shows that the logic pages and frame assemblies go through the same type of gradual buildup operation for the DA as was considered for the CP. This includes the following tests: (1) DA Hardware (including memory modules), (2) Interface Measurements, (3) Temperature/Altitude, (4) Operative Vibrations, and (5) Operative Life. The DA's intended for Flight Qualification Tests would be tested in a manner similar to that used for the CP. All DA hardware testing at the IBM Space Guidance

Center would be performed on the Apollo Data Adapter Tester and Monitor (ALAMO), and DA field testing would be performed on APOGEE.

3. **POWER SUPPLY (PS) HARDWARE TESTS** (See Figure V-1, Sheet 3)

Since the PS would, in general, be made of discrete components packaged and connected in a conventional manner, sheet 3 shows the fabrication and assembly of the entire PS unit. There would be some testing of the sub-assemblies that go into the PS, but there would not be the gradual buildup of logic functions as shown in the charts for the CP and DA, since the PS would not be that complex a unit. The PS would undergo Unit Test, Interface Measurements Test, Temperature/Altitude, Vibration, and Life Tests. Flight Qualification models would also undergo the Flight Qualification Tests. The ALAMO would be used to perform these tests at the IBM Space Guidance Center, and APOGEE would perform these tests, where required, in the field.

4. **AGC HARDWARE INTEGRATION TESTS** (See Figure V-1, Sheet 4)

To prove that the various units of the AGC system (CP, DA, and PS) operate properly together, the Hardware Integration Tests as shown on sheet 4 of the flow diagram is performed.

5. **LTE SUPPORT TO AGC SOFTWARE OPERATIONS** (Figure V-1, Sheet 4)

The flow chart includes not only AGC hardware tests, but also tests of the AGC software. At this time, it is envisioned that the AGC software tests would consist of four different stages:

- Operational Flight Program (OFP) evaluation on an IBM 7090 Computer.
- Pseudo-Flight (or Flight Simulation) operations using the actual CP hardware and appropriate LTE. This Pseudo-Flight test would evaluate the operation of certain critical OFP loops while other portions of the OFP were being designed. The CP would use input data stored in its memory as inputs to the program loops under test and check the results of calculations against pre-determined output data, also stored in the CP memory.
- Complete OFP check-out - Part 1. Using only the CP and appropriate LTE, the actual OFP would be checked out. Under this concept, each branch of the OFP would be tested, and the OFP need not be changed from the form which would be in the AGE at the time of launch. In using only the CP, OFP, and LTE, OFP branches which might depend on DA malfunctions can be tested without actually causing DA hardware failures, since the LTE would work only with data inputs and outputs to/from the CP and not those to/from the DA.

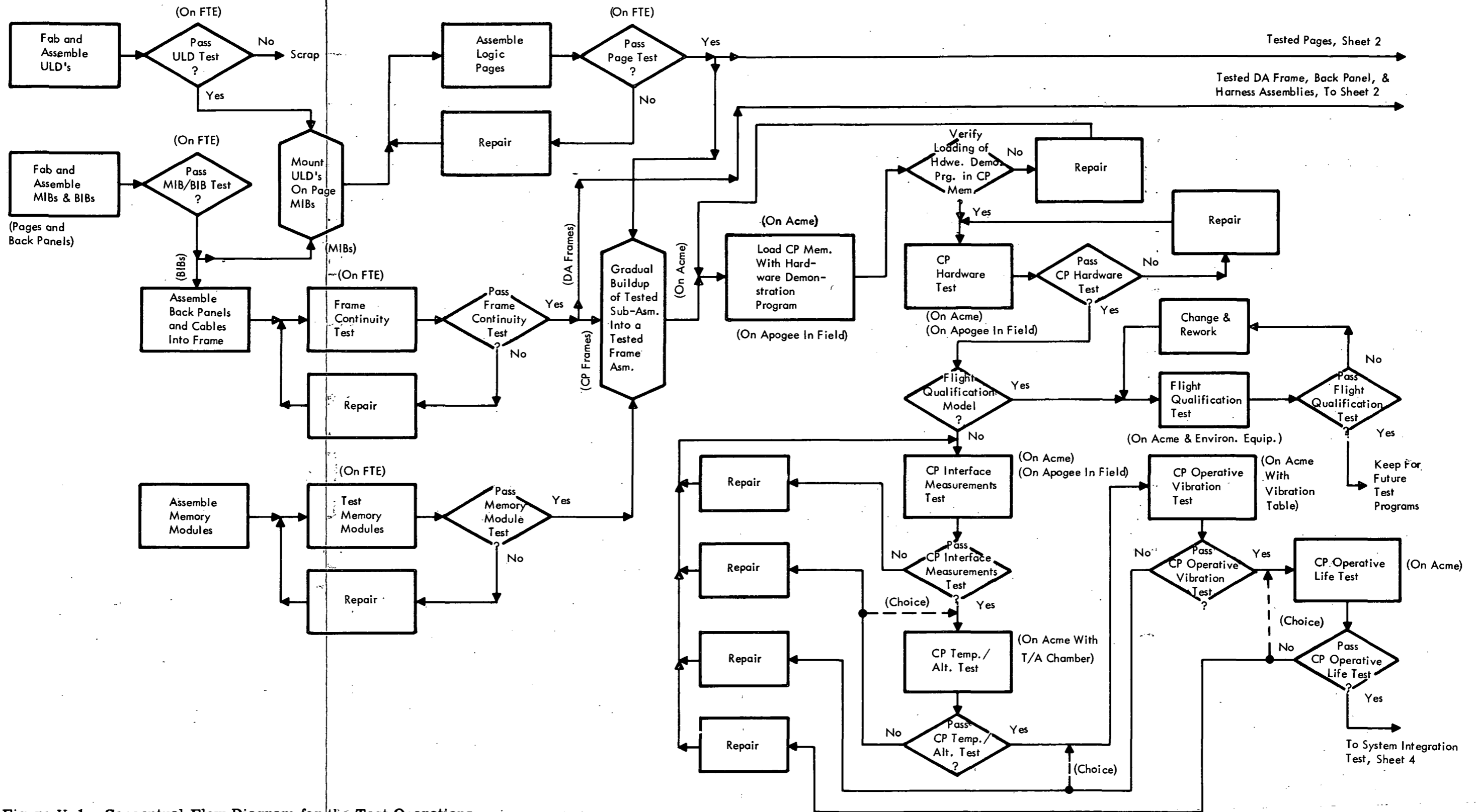


Figure V-1. Conceptual Flow Diagram for the Test Operations on the Proposed Apollo Guidance Computer (AGC) Central Processor (CP) Hardware Tests (Sheet 1 of 4)

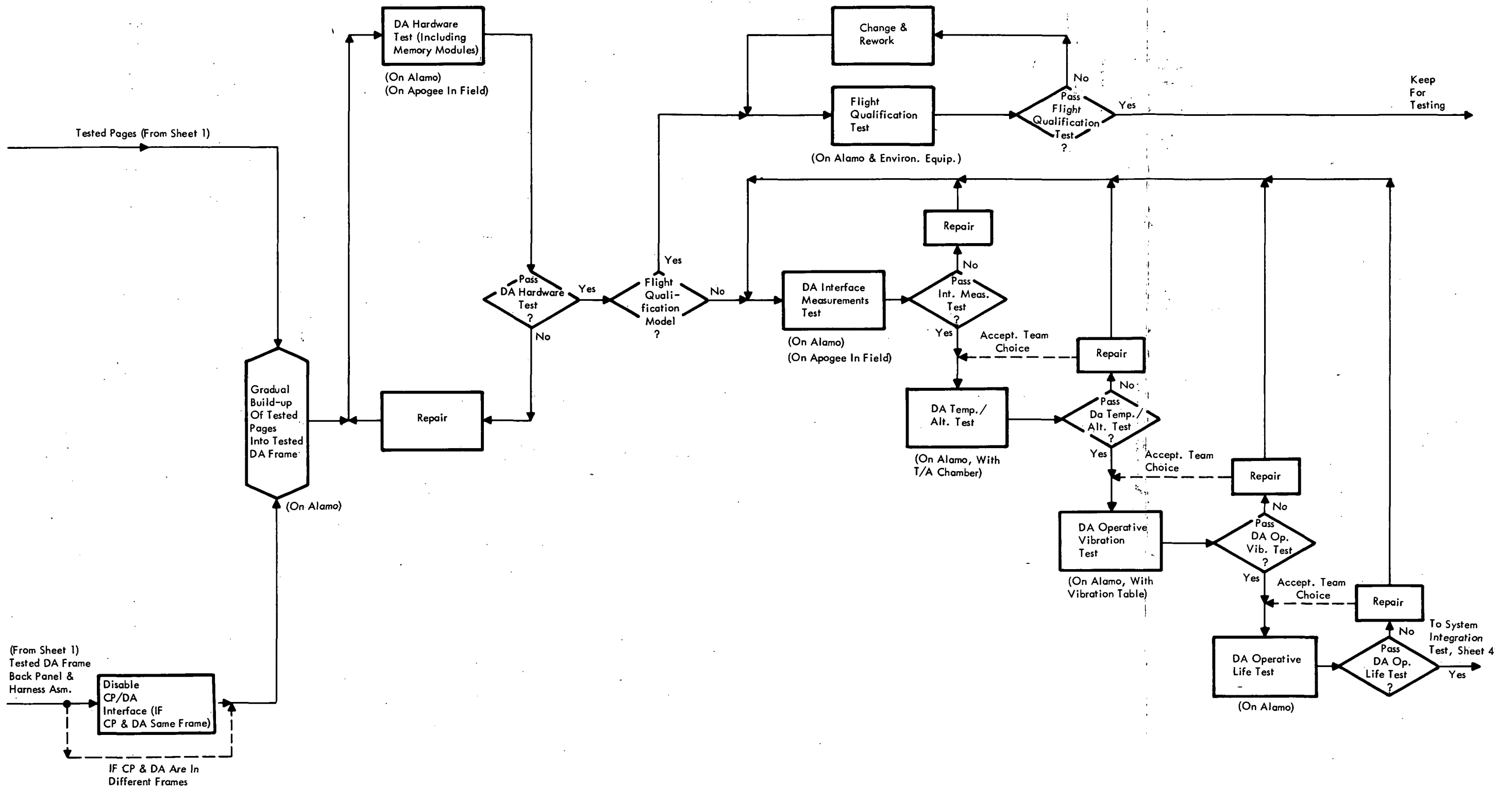


Figure V-1. Conceptual Flow Diagram for the Test Operations on the Proposed Apollo Guidance Computer (AGC) Data Adapter (DA) Hardware Tests (Sheet 2 of 4)

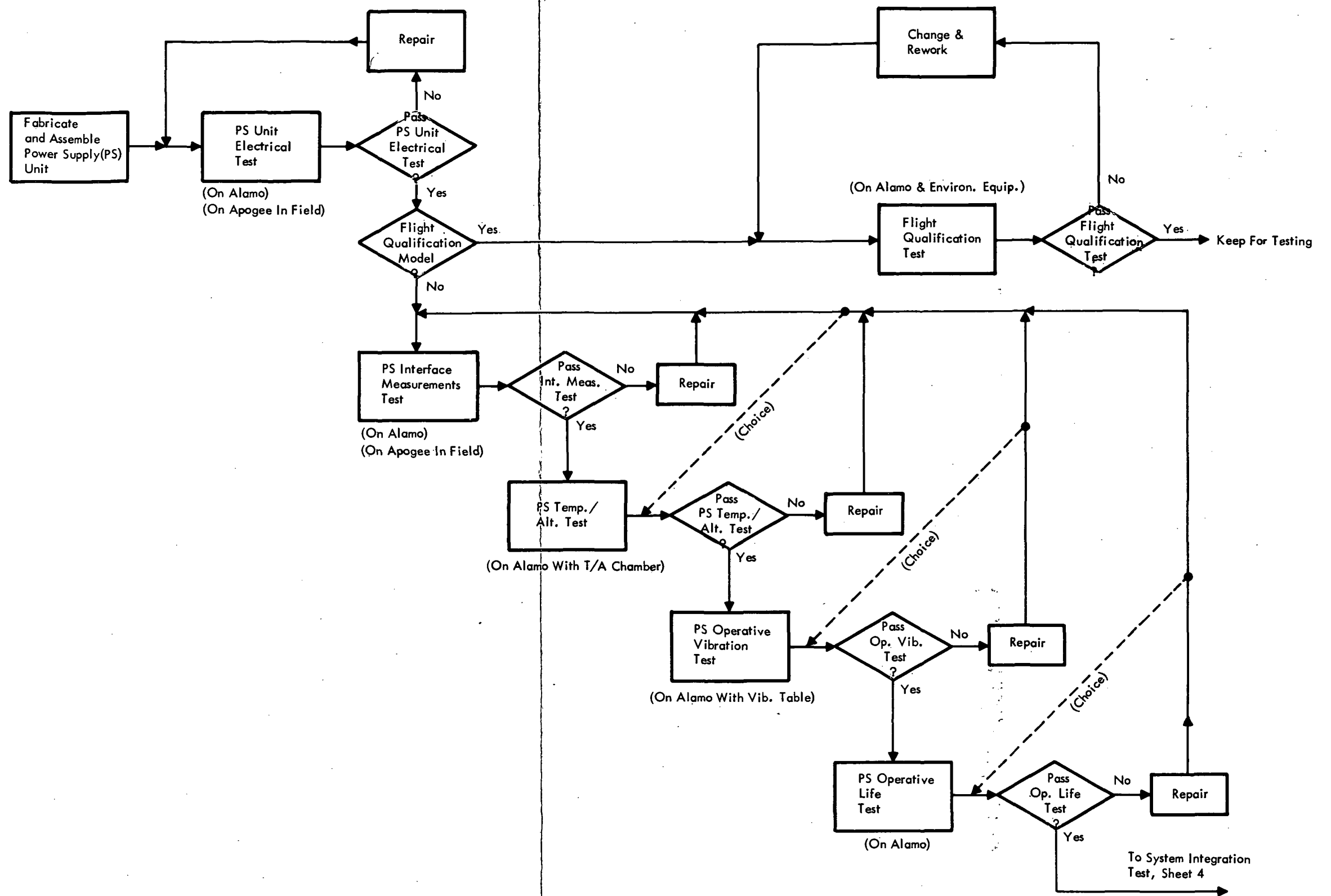


Figure V-1. Conceptual Flow Diagram for the Test Operations on the Proposed Apollo Guidance Computer (AGC) Power Supply (PS) Hardware Tests (Sheet 3 of 4)

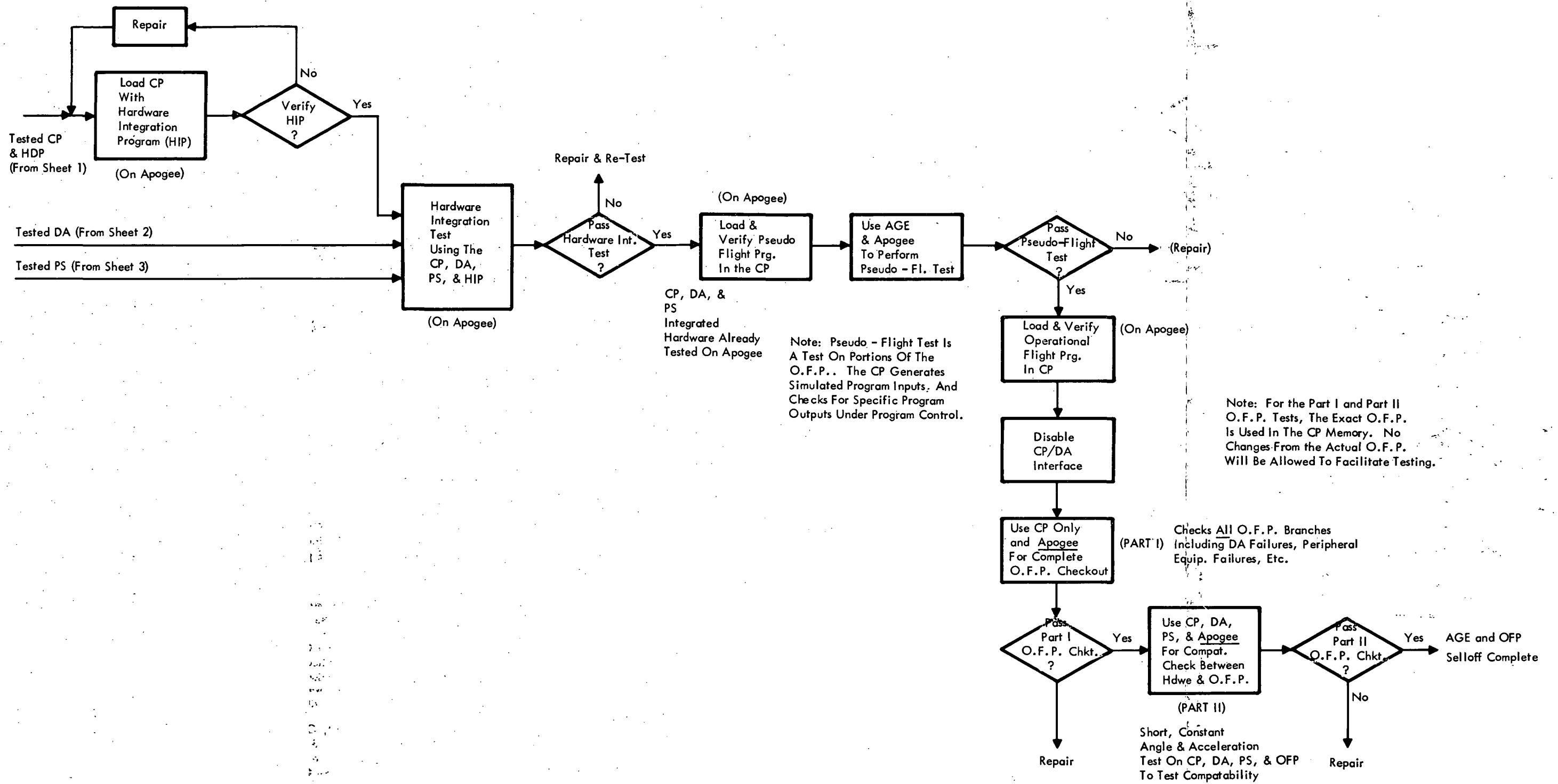


Figure V-1. Conceptual Flow Diagram for the Test Operations on the Proposed Apollo Guidance Computer (AGC) Hardware Integration and Operational Flight Program (OFF) Sheet 4 of 4)

- Complete OFP check-out - Part 2. After the AGC hardware has been completely tested and the AGC OFP has been completely tested on the CP alone, one additional test would be needed to verify proper operation of the AGE hardware/software system; a compatibility demonstration with the CP, DA, Power Supply (PS) and OFP. This would comprise Part 2 of the OFP check-out and would be a relatively short, simple test to demonstrate the required compatibility. It would not be intended that this compatibility demonstration be a simulated flight, since the first part of the OFP test would be composed of essentially several simulated flights with various high and low worst case and nominal data inputs to the CP from the LTE.

6. GENERAL DESIGN AND PACKAGING TECHNIQUES FOR LTE

a. Circuitry Building Block Description

Advanced IBM technology has developed several programs to standardize and expedite the design and manufacture of new data processing systems. Notable among these programs is the Standard Module System (SMS), which utilizes solid state circuits on printed wiring cards, design automation, and modular packaging. The LTE will take full advantage of this technique.

The basic SMS circuit card, a typical example of which is shown in Figure V-2, is being produced in quantity by the IBM Endicott facility. SMS cards have been in use for a number of years and are well proven in reliability and effectiveness.

The family of logic to be used is a Saturated Drift Transistor Diode Logic system providing a N'AND function. The N'AND circuitry provides a complete system of solid state logic for use in intermediate and high speed systems. The high diode/transistor ratios and large signal levels provide excellent noise rejection characteristics. Component cards in this group are capable of performing all necessary logical functions with high reliability and low cost.

b. Economy and Flexibility of Design Procedures

The Automated Logic Design (ALD) techniques presently used by IBM in the design and manufacture of data processing systems would be fully utilized. Basically, ALD provides computer (IBM 7090 and 1401) preparation of final logic drawings and wire lists, as well as a partial logic design check in accordance with a computer program written around the logic circuit ground rules. In addition, ALD prepares tape and punched card inputs to machines which wire-wrap the logical back panels and automatically make

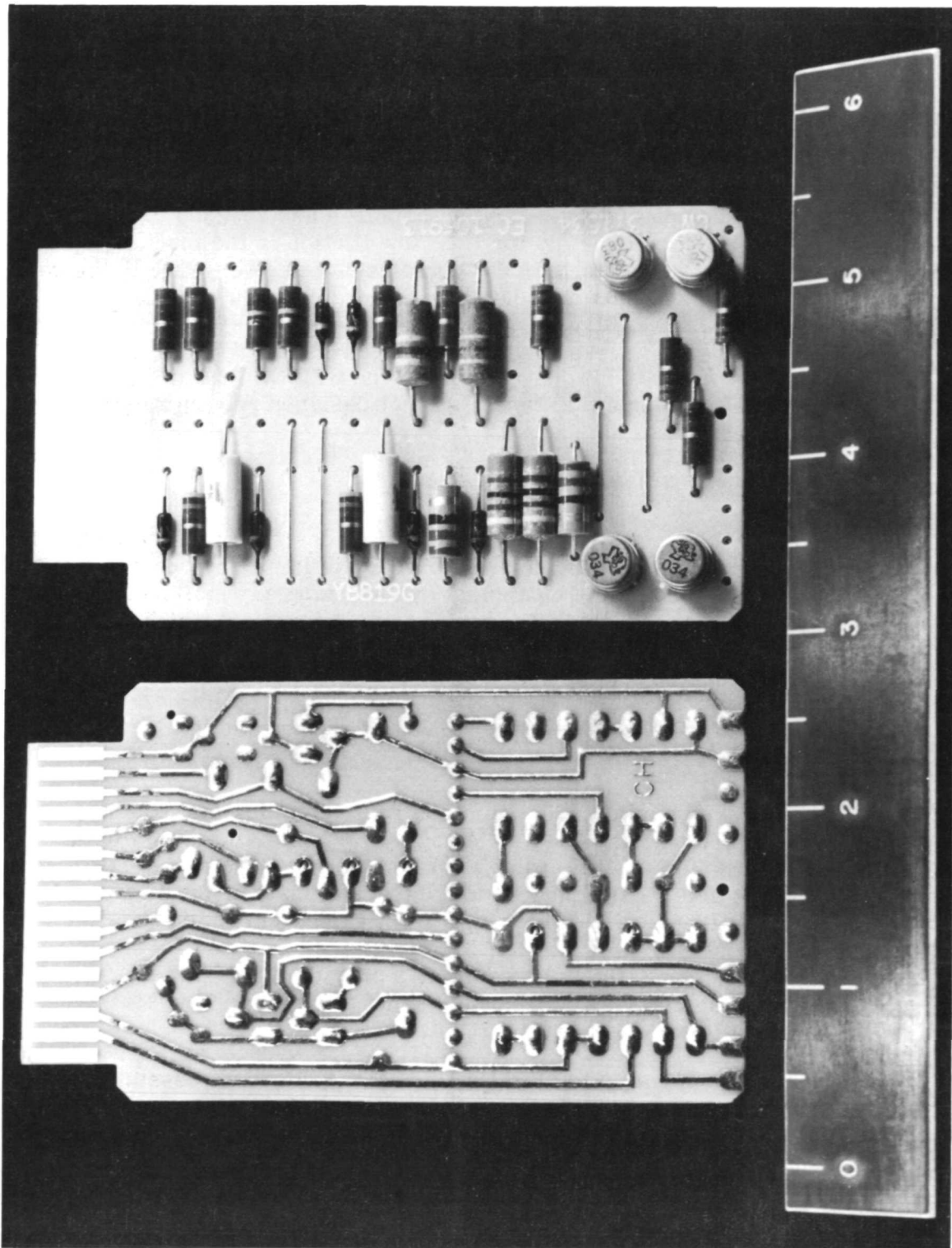


Figure V-2. Typical SMS Circuit Card

complete wiring checks. Thus, the design of the LTE would be as economical as is possible in that all serious electrical mistakes in the logic design of the LTE would be detected by the ALD system and corrected before the logic gates of the LTE are wired. Therefore, only a minimum of design errors need be detected and corrected during the "de-bugging" activity. In addition, very little time elapses between running the final logic configuration run through the ALD computer and receiving the wired and checked logic back panels from IBM Endicott, ready for final installation in the machines. Consequently, the logic design of the LTE could be changed very late in the design phase of the program to accommodate engineering changes resulting from AGC modifications.

Figure V-3 shows LTE designed under the STINGS R&D program. The appearance of this equipment is typical of that designed and built using the IBM SMS packaging system.

c. Design for Ease of LTE Troubleshooting

A "self check" mode of operation would be designed into the LTE. The features of this mode would be included to evaluate improper operation of the LTE and its associated prime equipment by localizing any malfunctions to either the particular piece of LTE, or the prime equipment which it is testing, or to a combination of the two, thus minimizing the time required to repair LTE malfunctions.

d. Impact on LTE Design of Alternate AGC Frame and Cooling Design

Since the design of an LTE liquid cooling system for the Saturn V Booster Computer and Data Adapter is nearly complete, the LTE for the Apollo Program would use this same approach to minimize design costs and time. If the Apollo CP, DA, and PS are integrally cooled, the LTE would furnish liquid at the proper temperature to the inlets and recirculate the liquid from the outlets. However, if the Apollo CP, DA, and PS are to be mounted on a cold plate, the LTE would then provide a cold plate for mounting the prime equipment, and would supply and remove the temperature-controlled liquid to and from the cold plate.

If one common frame is used for the CP and DA, it is possible that the proposed Apollo Data Adapter Tester and Monitor (ALAMO) could be eliminated and all CP, DA, PS, and integration testing done on the Apollo Guidance Computer Evaluation Equipment (APOGEE). This approach would require more APOGEE's to fill in for the eliminated ALAMO's.

5-11

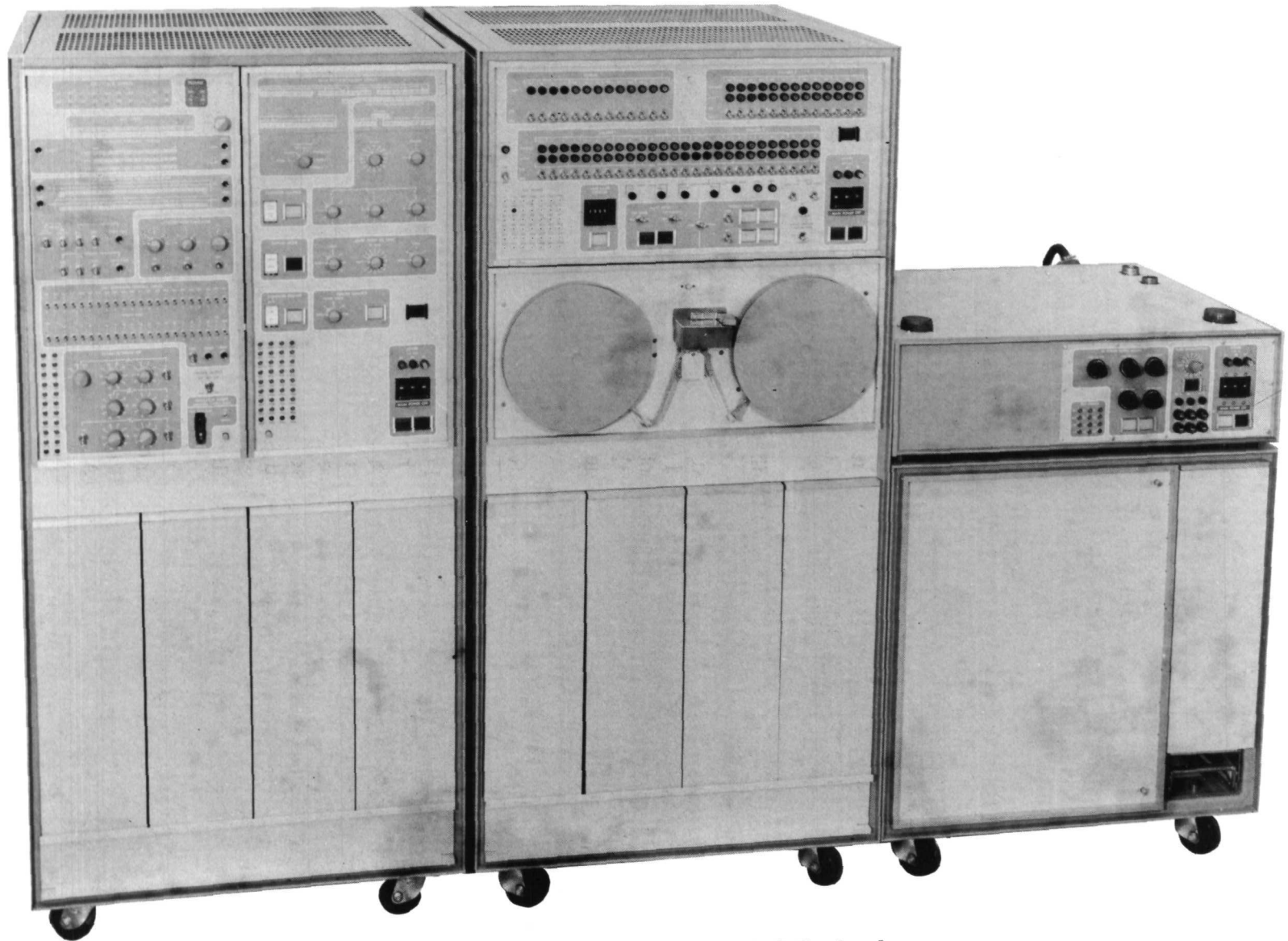


Figure V-3. Example of LTE Using SMS Technology

7. TESTING TECHNIQUES

In general, the testing techniques proposed in this report are identical to those used in the Saturn V Booster Program. This should facilitate compatibility between the two programs, and allow improvements made on one program to be used on the other.

B. SATURN V AEROSPACE COMPUTER MANUAL EXERCISER (ACME)

Detailed information on the ACME presently being designed and built for the Saturn V Booster Program is available in IBM Specification No. 6900007, Advanced-Saturn Computer Manual Exerciser ("ACME"), Design and Performance Specification For. The salient points of this specification are listed here for convenience.

1. INSTALLATION CONFIGURATION

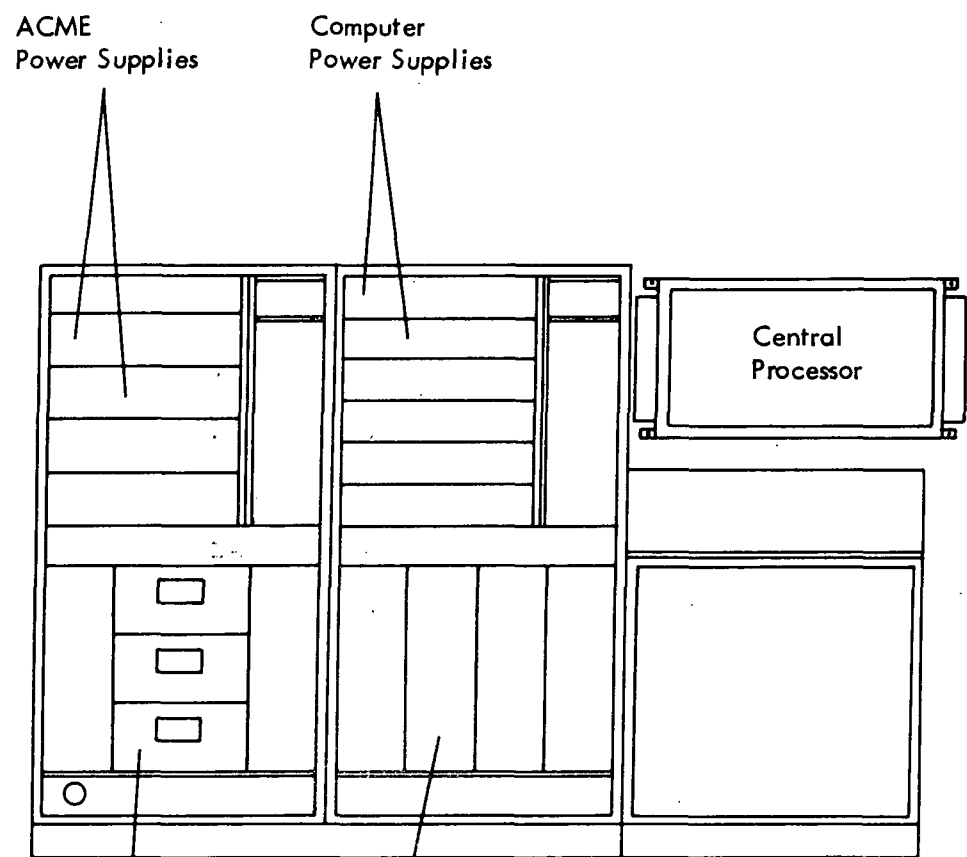
The ACME (Figure V-4) consists of two SMS double "cube" frames bonded together to form the higher of the two ACME modules, and one SMS single "cube" frame containing the cooling system components and mounting hardware for the Saturn V Guidance Computer, which forms the lower of the two ACME modules.

Only a minimum of modification would be needed to enable the Saturn V ACME to test the Apollo CP.

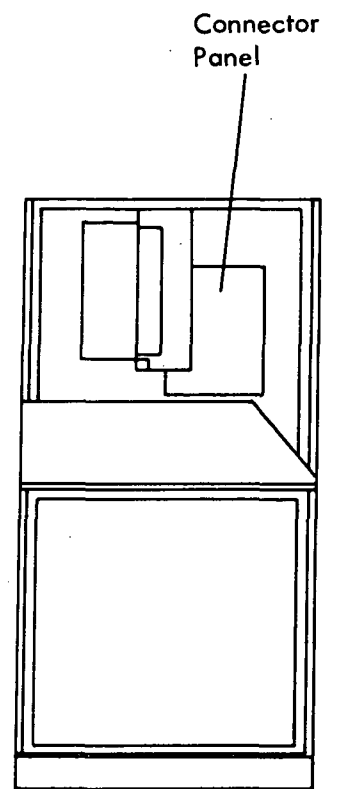
2. TESTS TO BE PERFORMED ON THE CP USING ACME

In general, the ACME would be used to test the CP hardware as follows:

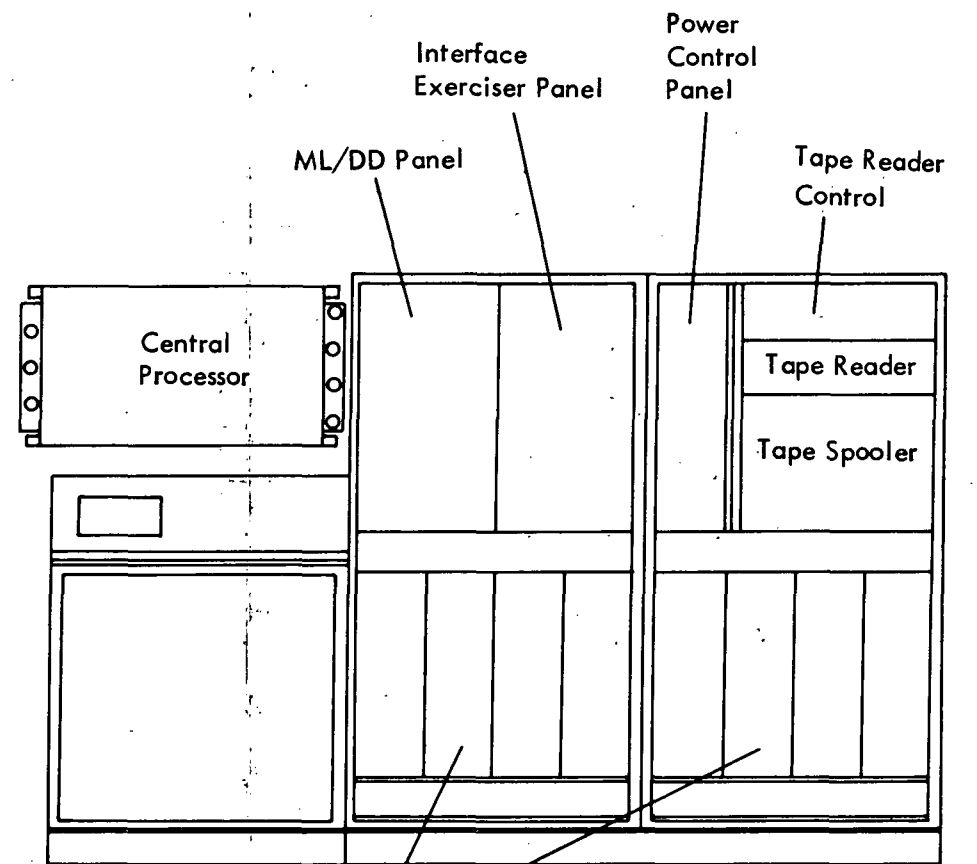
- (a) Gradual build-up and check-out of CP logic pages into a previously tested CP frame, back-panel, and harness assembly.
- (b) Complete pre-acceptance CP hardware test to prove that all CP hardware is present and operating properly.
- (c) Complete CP hardware acceptance test including interface measurements test, temperature/altitude test, operative vibration test, and operative life test.
- (d) CP flight qualification test which would be essentially the same as a normal acceptance test, except that the environments provided would be more demanding.



Rear View



Left End View



Front View

Figure V-4. ACME Configuration

3. MACHINE ORGANIZATION

To a large degree, the operation of ACME would be dependent upon a Hardware Demonstration Program (HDP) stored in the CP memory for use with the ACME. This HDP would be a maximum-exercise program intended to exercise every component in the CP.

Figure V-5 shows an overall block diagram of ACME. The ACME is divided into four basic functional parts: The Memory Core Loader (MCL), the Data Display (DD), the Interface Exercisor (IE), and the Power Control and Distribution System (PCDS).

To load data into the CP memory and verify that this data has, in fact, been loaded, the MCL section of the ACME would control the CP by first halting the CP operation, forcing an STO operation with a given operand address, and storing a 26-bit data word in that operand address location. This operation would continue until all the CP memory modules are loaded. After the desired number of memory modules are loaded, the MCL would force successive CLA operations in the CP, giving appropriate operand addresses each time. The data read from the CP under this mode of operation would be compared to the data previously loaded into the CP. As long as this data was verified, the verification test would be continued until the desired number of memory modules were verified. In case of nonverification, the MCL operation would stop and the data word input to the CP and the data word output from the CP would be displayed visually on the ACME control panel.

The MCL would load and verify one CP duplex memory module (8,192 26-bit words) in about 3.75 minutes. Data for the MCL load and verify operation would be contained on punched paper tape and introduced into the MCL via the Rheem photoelectric tape reader.

Data would be displayed in the data display portion of ACME. It would originate from several CP locations, such as the CP product or negative remainder delay line, CP multiplier or quotient delay line, CP multiplicand or division delay line, CP accumulator, CP transfer register, or from one of two spare data probes which could be placed at random test points throughout the CP. Some of the modes of data display operation would include: a repeat display mode in which a new display of data would be generated each time the CP instruction address would match an instruction address set in on switches on the data display front panel; a single display mode where only one display would be generated at the first match of CP instruction address with the manually selected instruction address after an "enable" switch is depressed; a single step mode to allow the CP program to proceed only one instruction at a time, controlled by an "Advance" switch; an automatic/manual restart mode for returning the CP program to instruction address

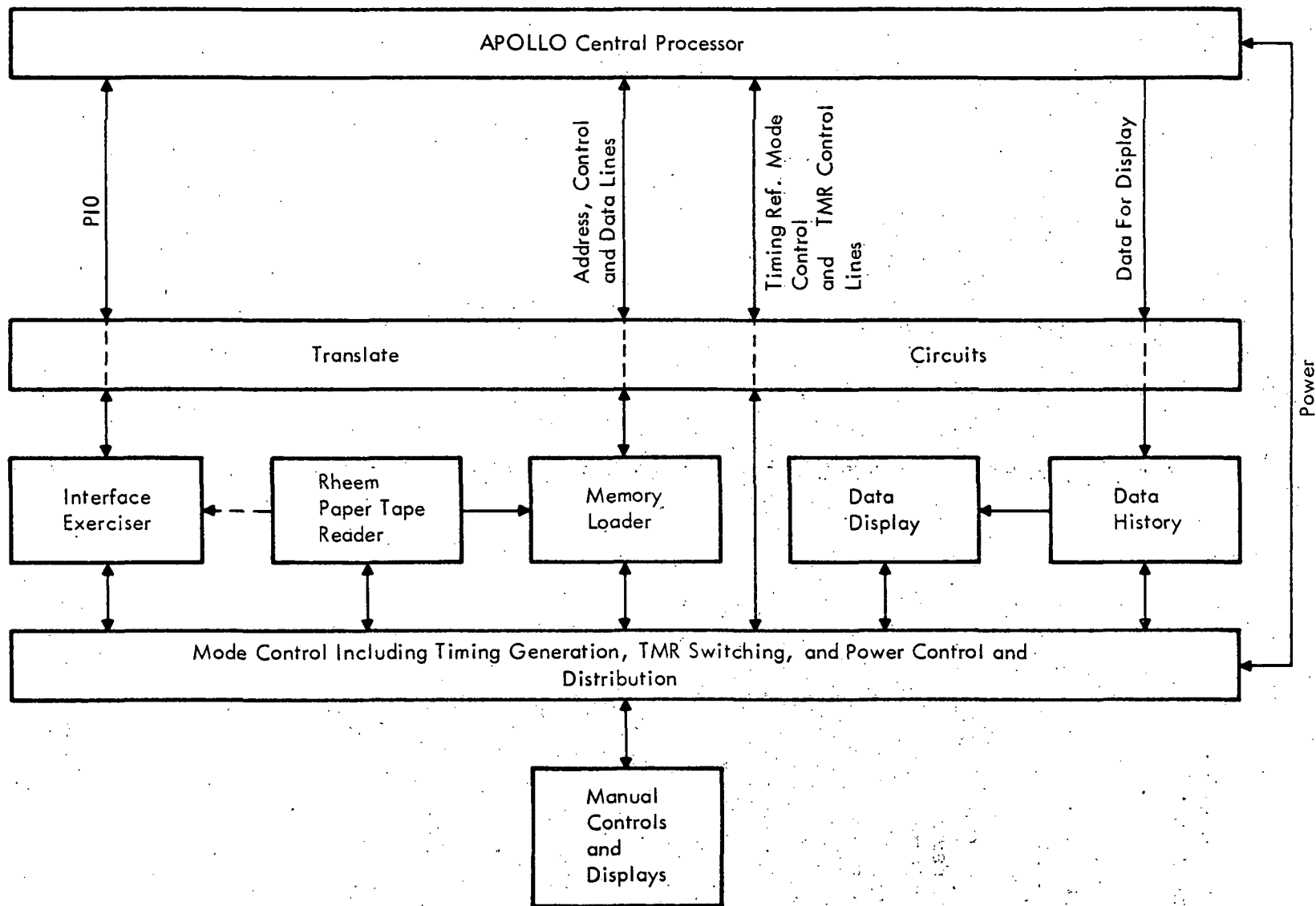


Figure V-5. ACME Block Diagram

“O”; Oscilloscope Sync and Marker modes; and a Lamp Test mode for detecting display lamp failures.

To supply data inputs to the CP and to allow the HDP to run properly, the Interface Exerciser (IE) would receive periodic data outputs from the CP, store the data output in a latch shift register, and feed the data back to the CP as a data input when requested to do so by the CP. Thus, the IE just “exercises the interface” and is under full control of the CP.

The Power Control and Distribution System (PCDS) of the ACME would be designed according to IBM Product Safety Standard No. 0-3-0501-0. Controls would be included for all intended uses of the ACME/CP combination. Ground loops and random noise would be minimized to the greatest possible extent.

4. ACME DESIGN GOALS

Several design goals would be stressed in the adaptation of ACME to the Apollo Program. One would be to enable the ACME to test a large portion of the CP without the necessity of having one or more CP memory modules connected. This would allow a large portion of the CP to be assembled and tested before the CP memory is connected. Another would be to provide the capability to isolate malfunctions manually to a pluggable logic module. This manual malfunction isolation would be accomplished by using pattern recognition techniques from ACME front panel indicators.

C. APOLLO DATA ADAPTER TESTER AND MONITOR (ALAMO)

1. INSTALLATION CONFIGURATION

Figure V-6 shows the installation configuration to be used for the ALAMO. This is the same over-all frame configuration that would be used with ACME, with the addition of one extra low module or “single cube” which would be used to mount an automatic output typewriter of the IBM “Selectric” type.

2. TESTS TO BE PERFORMED ON THE DATA ADAPTER USING ALAMO

All tests using the DA, PS, and ALAMO would be hardware tests, as follows:

- (a) Gradual build-up and check-out of DA logic pages and memory modules into a previously tested DA frame, back panel, and harness assembly.

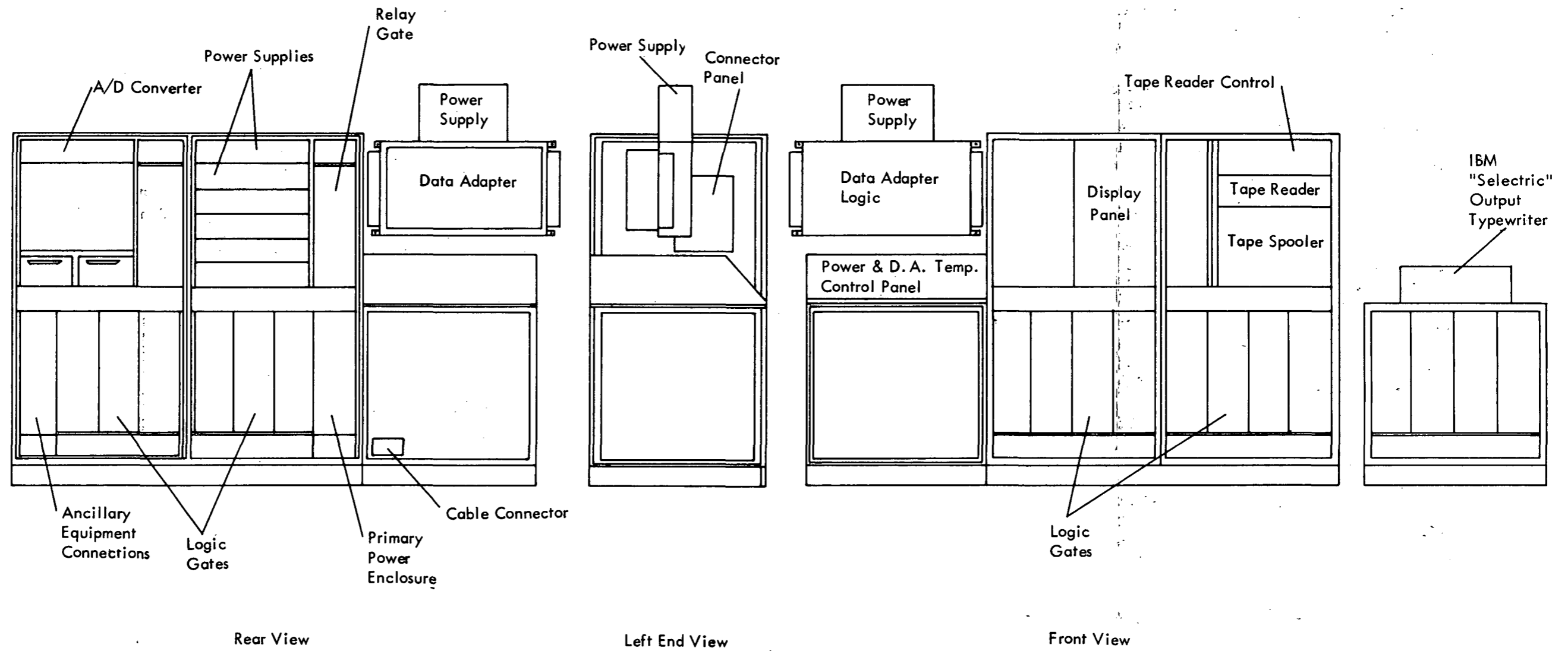


Figure V-6. ALAMO Configuration

- (b) Complete electrical test of the PS unit.
- (c) Complete pre-acceptance DA and PS hardware test, including DA memory modules, to prove that all DA and PS hardware is present and operating properly.
- (d) Complete DA and PS hardware acceptance test including interface measurement test, temperature/altitude test, operative vibration test, and operative life test.
- (e) Complete DA and PS flight qualification tests which would be essentially the same as the normal acceptance tests, except that the environments provided would be more stringent.

3. MACHINE ORGANIZATION

Since the DA is envisioned to undergo many engineering changes because of various mission requirements, the ALAMO would be designed to be as flexible as possible to minimize the cost and availability impact of DA changes. As such, the basic part of the ALAMO would be a highly flexible test processor with associated core memory. It is intended that the bulk of the engineering change activity could be accommodated in the ALAMO by changes in the test processor's stored program, rather than by actual hardware changes in the ALAMO.

Figure V-7 is a block diagram of the ALAMO. The basic functional parts are: the Test Processor, which would be the heart of the machine; the I/O multiplexers, which would control the data transfers to and from the Test Processor; the data displays, which would help in the actual operation of the ALAMO and in isolating malfunctions in either the DA/PS or ALAMO; the manual controls and mode control, timing generator, TMR switching, and power control and distribution which support the entire operation; the paper tape reader input for loading and verifying the ALAMO's memory; and the typewriter for recording acceptance tests, etc.

4. ALAMO DESIGN GOALS

The ALAMO design goal would be to provide as flexible a machine configuration as is practical to minimize the sensitivity of the ALAMO hardware design to DA/PS changes. Also, the malfunction isolation techniques developed in conjunction with ACME would be automated as much as possible in the pattern recognition area. Since the function of the ALAMO would be very similar to that of ADAPT on the Saturn V Booster Program, much of the ADAPT circuitry and hardware design would be used for ALAMO to minimize design time and costs.

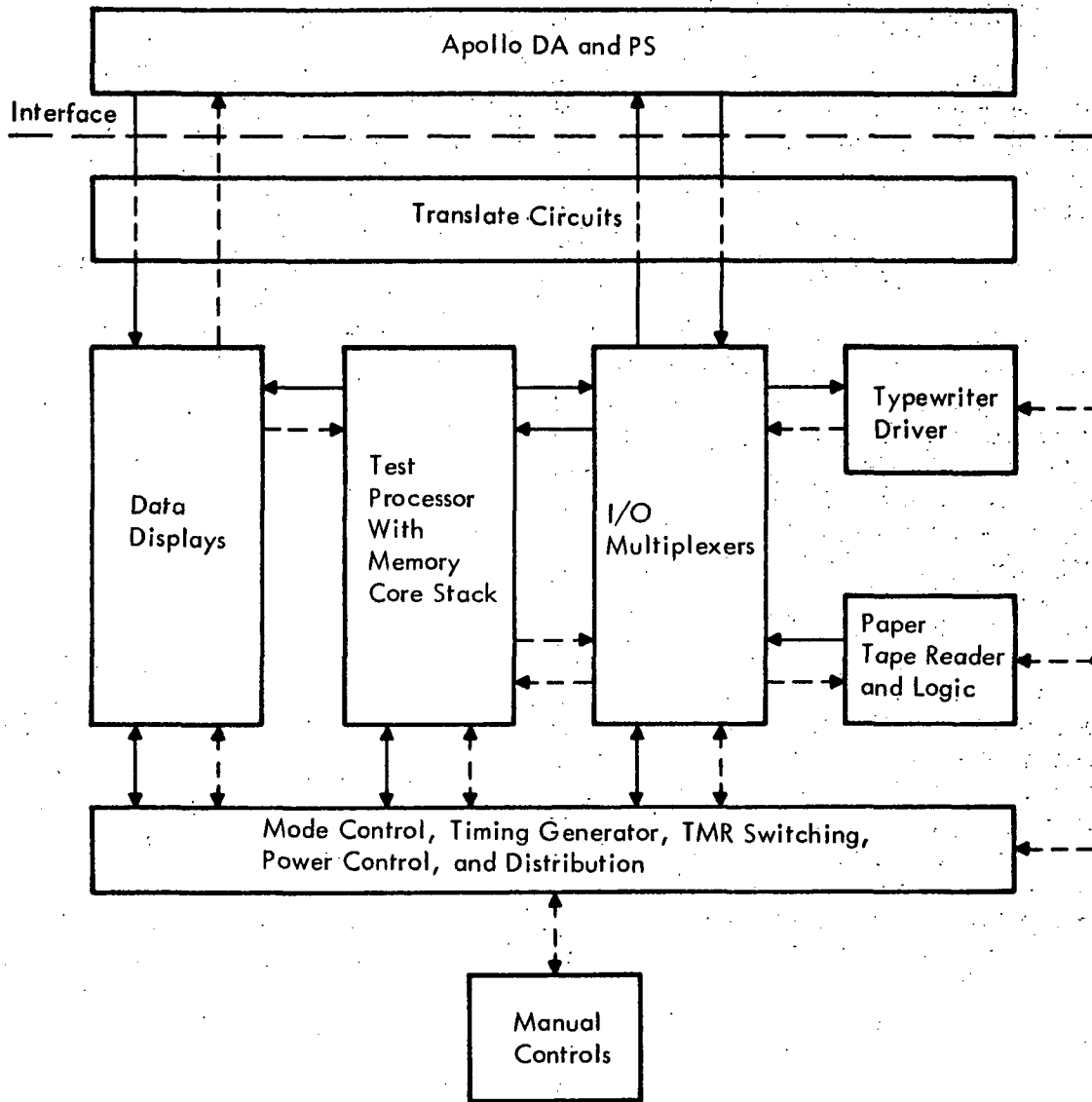


Figure V-7. ALAMO Block Diagram

D. APOLLO GUIDANCE COMPUTER EVALUATION EQUIPMENT (APOGEE)

1. INSTALLATION CONFIGURATION

The APOGEE (Figure V-8) would be made up of four basic modules: one low module made up of two SMS single "cube" frames bonded together, one high module made up of two SMS double "cube" frames bonded together, one high module made up of one SMS double "cube", and one IBM 1443 printer.

2. TESTS TO BE PERFORMED ON THE AGC SYSTEM USING APOGEE

Since the APOGEE would be the piece of LTE most used in the field, it would need to perform the following functions:

- (a) All ACME tests on the CP
- (b) All ALAMO tests on the DA and PS
- (c) All OFP tests (as given in part A5 of this section)

3. MACHINE ORGANIZATION

Figure V-9 is a block diagram of APOGEE functions. The APOGEE would use the same basic Test Processor as the ALAMO to reduce design and fabrication time and costs. The APOGEE block diagram is very nearly the same as the diagram for ALAMO (Figure V-7). Every effort would be made to minimize new hardware design for APOGEE and to build upon experience and hardware accumulated during the ACME and ALAMO efforts.

4. APOGEE DESIGN GOALS

The design goals for the APOGEE would be to implement as much flexibility in machine design as is possible, and to streamline and partially mechanize malfunction isolation pattern recognition.

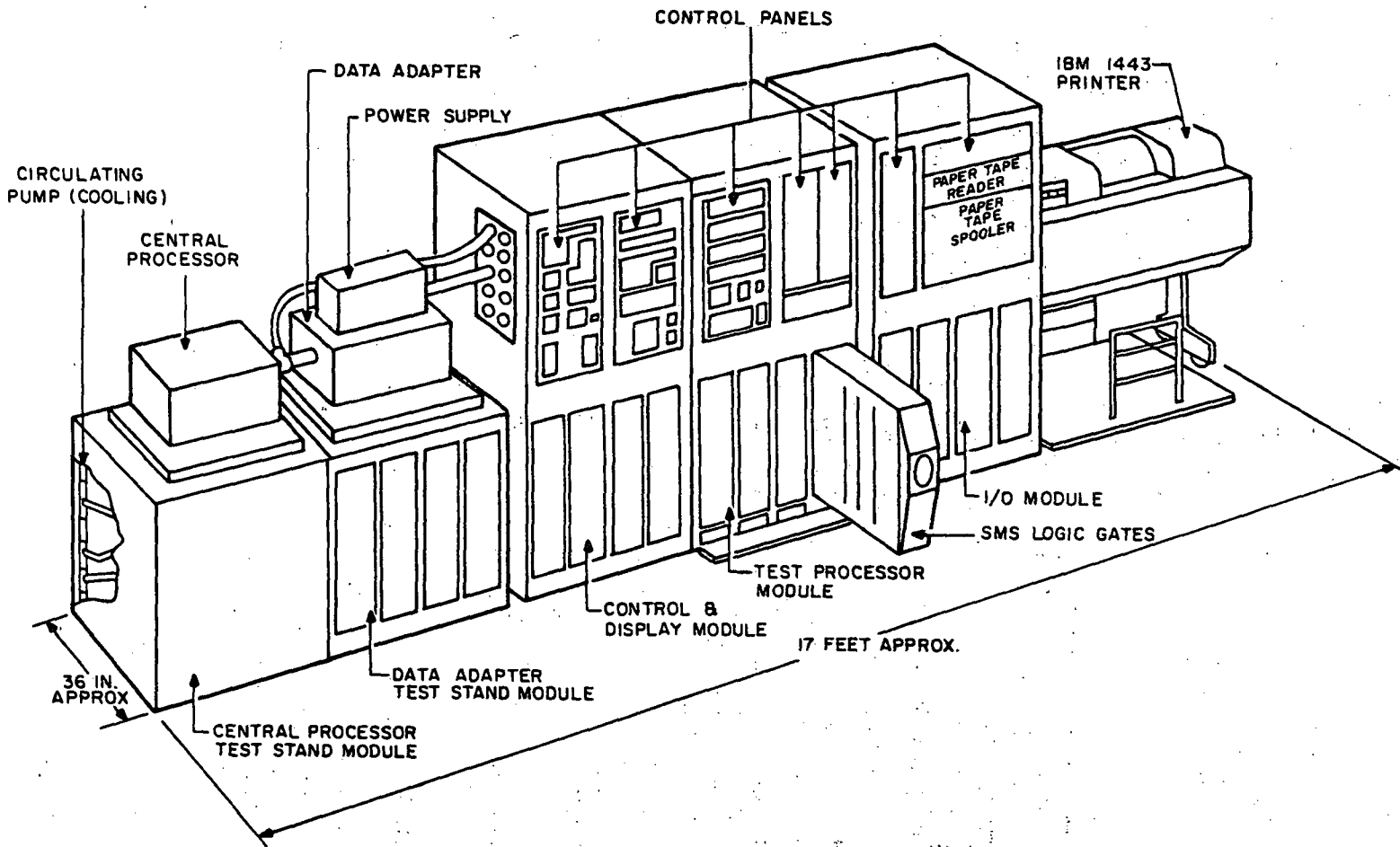


Figure V-8. APOGEE Configuration

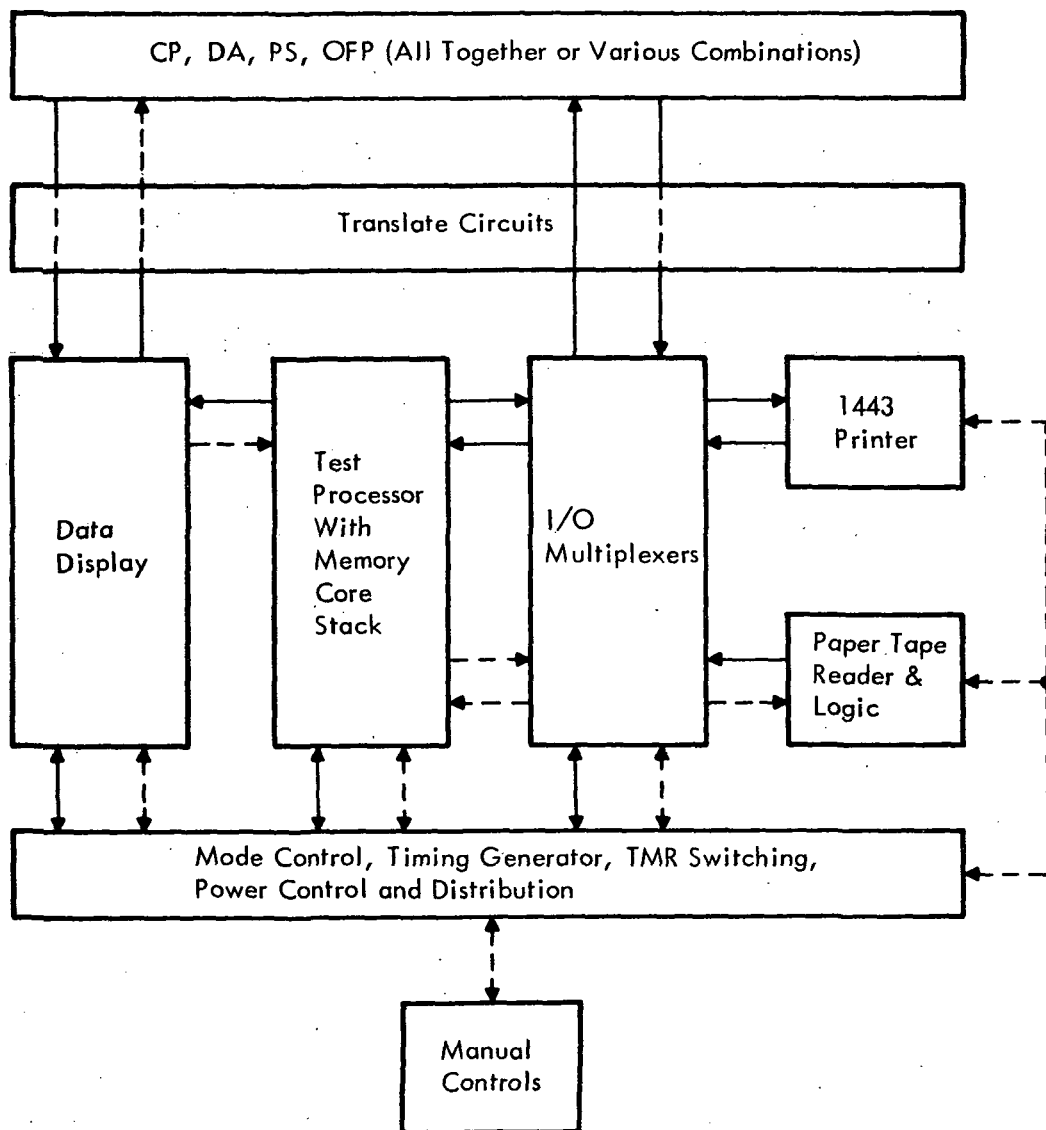


Figure V-9. APOGEE Block Diagram

Section VI
PROGRAM PLAN

Section VI

PROGRAM PLAN

This report discusses the feasibility of utilizing the basic Saturn V guidance computer, presently being developed for MSFC, in the Apollo guidance application. The report also answers the questions regarding the feasibility of utilizing this same technology for the data adapter as well.

This program plan is based on an over-all common application of Saturn V technology for the Saturn V and Apollo programs. An alternate approach to the structural design, discussed elsewhere in the report, suggests the possible use of a unique structure around a common internal computer. This alternative, if adopted, would have a minimal impact upon technology commonality.

MSFC has already experienced the benefits of a program of this type. The ASC-15 computer used in Saturn I was, with slight modification of structure and electronics, common in technology and hardware to the computer used in Titan II. Even though the frame for the ASC-15 computer was modified and electronic modules were added, the Saturn I computer program experienced a 50 percent reduction in procurement and fabrication costs based on the break-in point in the common production line and on cost savings in the procurement of components.

The Apollo back-up program will benefit in a ratio commensurate with the break-in point, but, regardless of break-in point, a substantial savings will be realized due to procurement pricing and the major reduction in development design costs for prime hardware, tools, and test equipment.

The successful accomplishment of such a program, as outlined, is predicated on a common approach to design change. Both sets of hardware must accept or reject design changes, as an entity, to preserve the common approach. IBM would recommend that central processor changes be directed to us by MSFC and ADA changes by MSC. In this manner, program impact will more easily be evaluated.

Based on experience, the advantages of an operation of this type are overwhelming, and the lower cost impact to succeeding programs are extremely significant.

IBM feels that it has the capability to economically perform the requirements of the Apollo back-up program and submits the following proposal for its accomplishment:

Phase I - Study

The report on the study is submitted herein.

Phase II - Development

Phase II has been subdivided into two steps, one covering the period from 1 October 1963 to 31 December 1963, and the other covering the period from 1 January 1964 to 31 December 1964. Detailed scheduling information on systems definition and software, prime hardware, and LTE is shown in Figures VI-1, VI-2, and VI-3 respectively. What will be accomplished during each step is described below.

Step I - 1 October through 31 December 1963

- (a) Saturn V changes for the Apollo-Saturn application will be reconciled with MSFC and MSC to preserve the commonality approach.
- (b) Accomplish preliminary design of the ADA, power supply unit, and LTE to maintain a responsive program.
- (c) Examine the system application and conduct trial programming of the Saturn V computer to confirm system operational adequacy for Apollo.
- (d) Procure long-lead components for hardware requirements to maintain a smooth program flow and minimize any schedule impact.
- (e) Examine further optimization of the mechanical structure during this step based on more recent information received from NAA. In particular, the possibility of increasing the memory capacity from 6 memory modules to 8 will be explored in detail.

Step II - 1 January 1964 through 31 December 1964

- (a) Design and develop an ADA, power supply unit, and LTE. Fabricate the breadboard equipment needed to prove out the design.

- (b) Fabricate two deliverable prototype Apollo back-up systems (CP/DA) and necessary LTE.

1) Allocation

One system (delivered in-house in November 1964) with associated LTE allocated to IBM for qualification testing.

One system (delivered in December 1964) allocated to MIT for system interface testing in the MIT G/N House System.

- (c) Initiate qualification testing of the IBM in-house system and provide support to system interface check-out at MIT.
- (d) Develop a computer check-out test program for use at MIT in interface check-out.
Provide programming effort for operational application.
- (e) Carry out production planning and provide long-lead facility development to permit delivery of flight hardware in 1965. This effort will provide for full production implementation of the Phase III program with a favorable delivery reaction time.

Phase III

Starting in the first quarter of 1965, IBM will be delivering Saturn V equipment to MSFC at the rate of one per month. At that point IBM sees no difficulty in increasing the production to satisfactorily meet the full requirements of the Apollo Program.

	1963			1964												
	O	N	D	J	F	M	A	M	J	J	A	S	O	N	D	
System Definition			→													
Definition and Decisions on Changes to Saturn for Commonality			→													
Circuit and Logic Design of Apollo Data Adapter			→													
Design of Apollo Laboratory Test Equipment			→													
Design and Check Out Hardware Test Program																→
Design and Check Out Interface Test Program																→

Figure VI-1. System Definition and Software Schedules

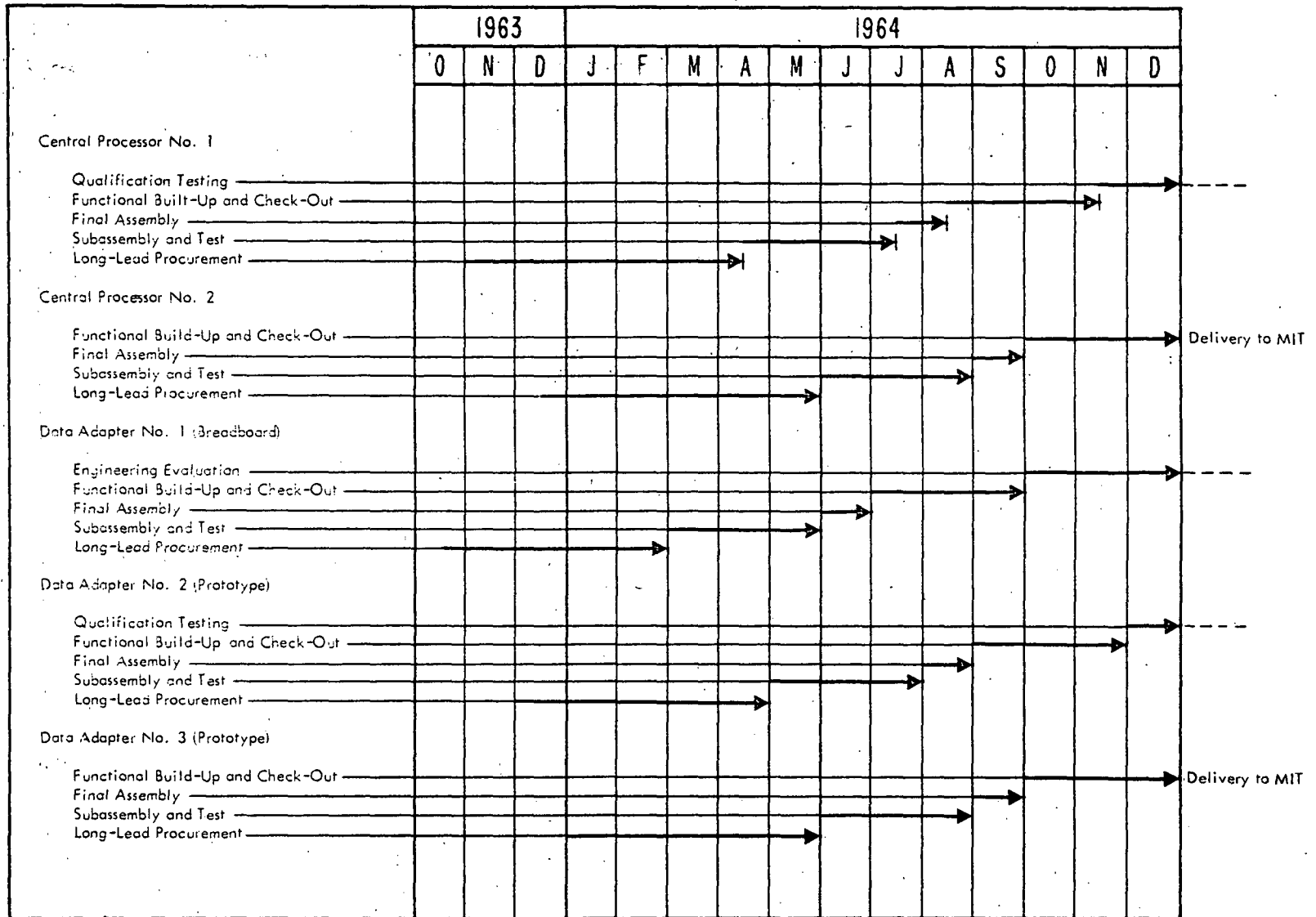


Figure VI-2. Phase II Prime Hardware Schedules

6-7/8

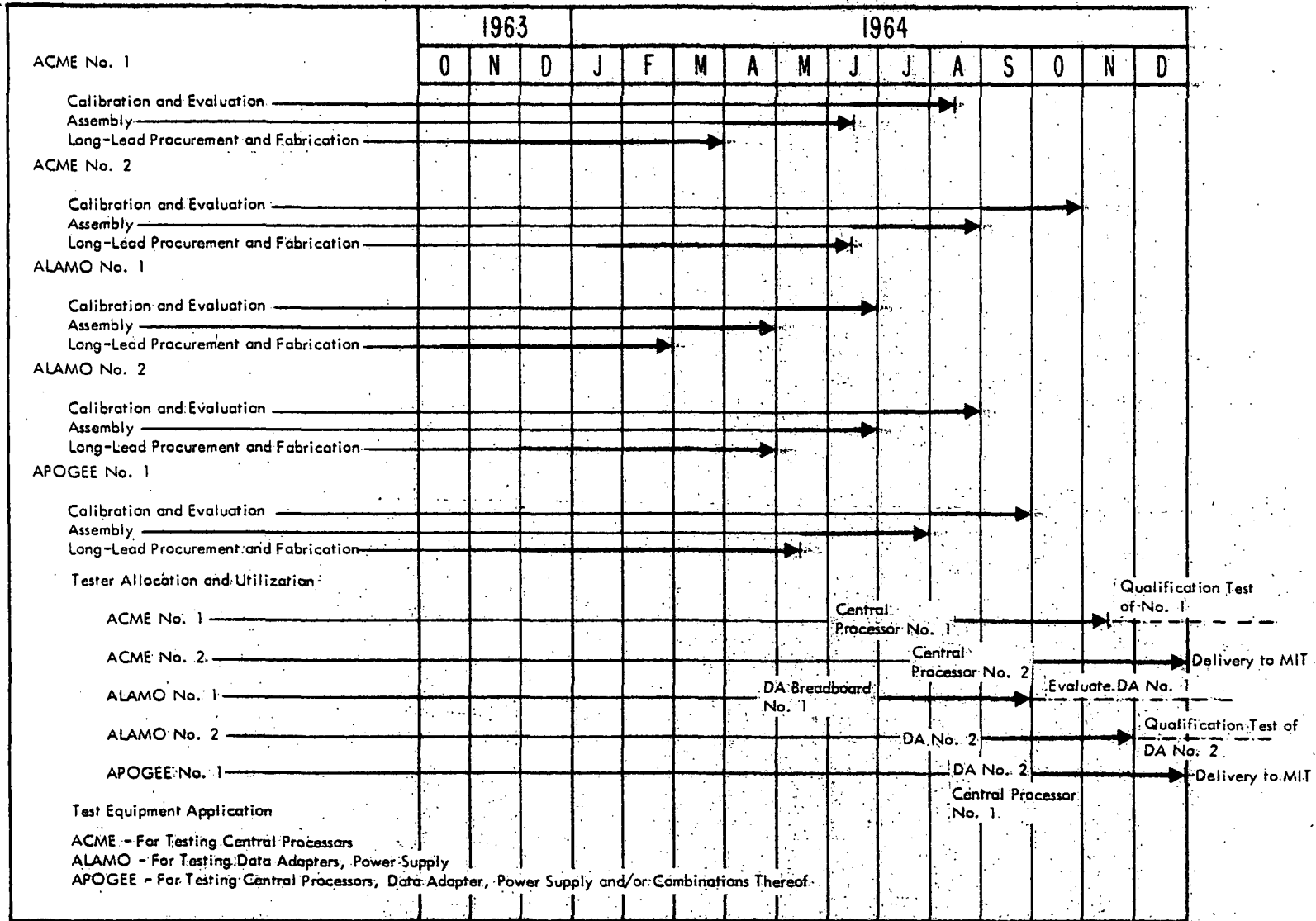


Figure VI-3. Phase II LTE Schedules

Appendix A
SYSTEM FAILURE RATES

Appendix A

SYSTEM FAILURE RATES

A. SIMPLEX MODULE RELIABILITY ESTIMATES

In Section III-C mathematical expressions were derived for the reliabilities of the redundant subsystems. This section compiles the simplex module failure rates and probabilities needed for solution of those expressions. The mission use profile assumed for performing the computations is shown in Figure A-1. Although Figure A-1 does not depict the actual profile, which has the computer switching on and off approximately fifty times, the total "on" time, total "off" time, and relative spacing of the larger "on" intervals are believed to be correct. To consider the actual profile would, for instance, require evaluating approximately 100 terms of the module failure equation.

Even though the mission is broken into 11 phases, only three different usage states exist:

1. Central Processor and Data Adapter energized; insignificant vibration or acoustics.
2. Central Processor and Data Adapter not energized; insignificant vibration or acoustics.
3. Central Processor and Data Adapter energized; significant vibration and/or acoustics.

States one and two are accounted for by assigning each component two failure rates; one for the energized state and one for the nonenergized state. For state three, the simplex module failure rate is obtained by multiplying the energized (state 1) rate by a K-factor.

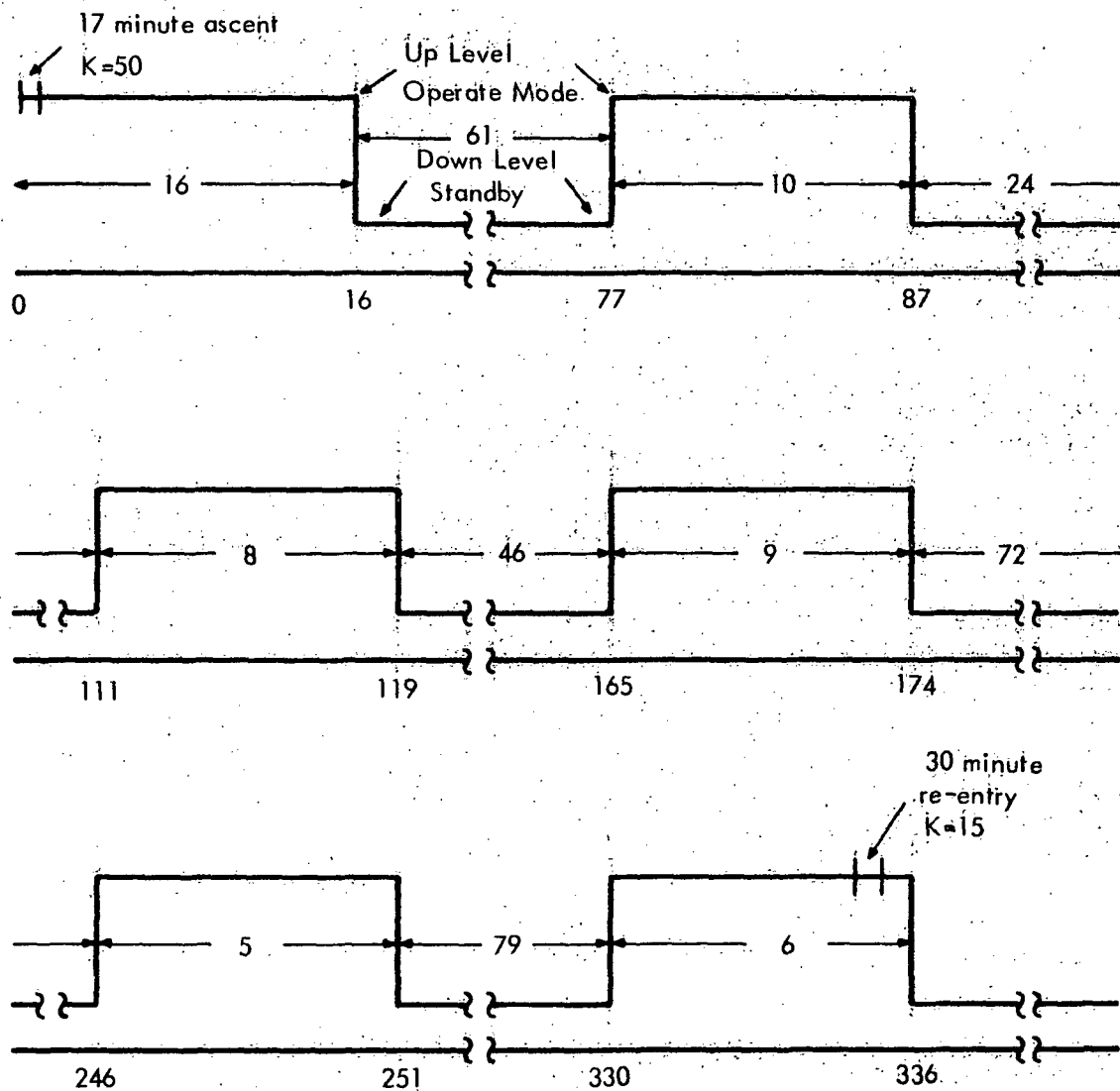
The "simplex reliability equation" then reduces to:

$$R_m = \exp - \left[\lambda_e (T_e + K T_v) + \lambda_{ne} T_{ne} \right] \quad (1)$$

where:

λ_e = energized simplex module failure rate

λ_{ne} = nonenergized simplex module failure rate



Total Time = 336 Hours
 Total On Time = 54 Hours
 % On Time = 16%

Figure A-1. Mission Profile
 (Processor and DA Chronometer for Entire Mission)

- T_e = total time energized - with insignificant vibration and noise
 T_{ne} = total time not energized - with insignificant vibration and noise
 T_v = total time energized - with significant vibration and noise
 (0.27 hour launch; 0.5 hour re-entry)
 K = value of K-factor (K = 50 for launch; K = 15 for re-entry)

B. PROCESSOR LOGIC

Table A-I gives the component failure rates for one channel of processor TMR logic.

Table A-I
ONE CHANNEL OF PROCESSOR TMR LOGIC
COMPONENT FAILURE RATES

Item Component	Quantity	Energized		Unenergized	
		Failure Rates $\lambda \times 10^6$	$n \lambda \times 10^6$	Failure Rates $\lambda \times 10^6$	$n \lambda \times 10^6$
Pages	18	1.106	19.89	1.106	19.89
Flex Cables	40	1.0	40.0	1.0	40.0
Connectors	22	0.689	15.158	0.689	15.158
Back Panel	1-2/3	3.762	6.283	3.762	6.283
Substrates	824	*0.0246	20.27	0.0246	20.27
Delay Lines	2	0.3	0.6	0.0025	0.005
Transistors	680	0.012	8.16	0.0023	1.564
Diodes	4240	0.006	25.44	0.001	4.24
Resistors	2503	0.013	32.539	0.001	2.503
Total			168.34		109.913
R = 0.9573 using equation (1)					
*Substrates: The failure rate of the substance is synthesized from:					

Table A-I. One Channel of Processor TMR Logic Component Failure Rates (cont)

Item Component	Quantity	Energized and Unenergized	
		Failure Rates $\lambda \times 10^6$	$n \lambda \times 10^6$
Clips	12	0.0005	0.0060
Comp. Ball Joints	10	0.0005	0.0050
Solder Fillet Joints	12	0.001	0.0120
Single-Sided Lands	15	0.0001	0.0015
Substrate	1	0.0001	0.0001
Substrate Failure Rate = 0.0246			

The reliability of a timing, generator module (R_{tg} , Section III-C) is obtained using the component failure rates of Table A-II.

Table A-II

TIMING GENERATOR MODULE COMPONENT FAILURE RATES

Component Type	Total Number	Energized		Unenergized	
		$\lambda \times 10^6$	$n \lambda \times 10^6$	$\lambda \times 10^6$	$n \lambda \times 10^6$
Transmitter	224	0.012	2.688	0.0023	0.5152
Resistor	289	0.13	3.757	0.001	0.289
Diode	90	0.006	0.54	0.001	0.09
Pages	2-2/3	1.106	2.4493	1.106	2.9493
Substrates	106	0.0246	2.6076	0.0246	2.6076
Connectors	2-2/3	0.689	1.8371	0.689	1.8371
Total $n\lambda$		14.37		8.28	

$R_{tg} = 0.9966$ using equation (1)

The 7090 EDPS program which obtains the reliability of the TMR logic cannot directly account for hardware such as connections, cables, back panels, etc. Thus, it is necessary to adjust failure rates of electronic components to account for hardware failures. This adjustment is obtained by summing the failure rates of system hardware, and apportioning this summation back into component rates.

Using this approach, the failure rate input applied to the program is dependent upon the following:

- The logical organization of the machine, from which the true component count is derived.
- The physical packaging configuration. Table A-III lists the hardware used in the processor, together with assigned failure rates and conditional probability of open circuits (K_o) and short circuits (K_s) for each type of hardware.

Table A-III

HARDWARE FAILURE RATES AND PROBABILITY OF OPEN AND SHORT CIRCUITS

	Quantity	$\lambda \times 10^6$	K_o	K_s	$n \lambda_o \times 10^6$	$n \lambda_s \times 10^6$	$n \lambda \times 10^6$
Back Panels	5	3.762	0.80	0.20	15.048	3.762	18.810
Flex Cables	120	1.0	0.90	0.10	108.0	12.00	120.0
Pages	54	1.106	0.80	0.20	47.7792	11.9448	59.724
Connectors and Bodies	64	0.003	1.0	0.0	0.192	0.00	0.192
Connector Pins	6272	0.007	0.8	0.20	35.1232	8.7808	43.904
Substrates	2472	0.0245	1.0	0.0	60.564	0.00	60.564
Total Failure Rates					266.71	36.49	303.19

The adjusted failure rates were inserted into the 7090 program and the following reliability of the processor TMR logic was obtained:

$$R_{\text{tmr1}} = 0.9995$$

No decoupling capacitors are included in the component failure rates. Decoupling capacitors failing in the short mode represent a limitation on the system reliability. The TMR processor requires approximately 300 capacitors which are described by:

$$\lambda = 0.06 \times 10^{-6} \quad (\text{It will be pessimistically assumed that the capacitors are energized for the entire mission})$$

$$K_s = 0.90 \quad (\text{Conditional probability of short circuit})$$

If no precautions were taken, the capacitors would of themselves limit the guidance computer to a reliability of no greater than 0.995.

$$\text{i. e. , } R = 0.995 \quad (\text{reliability of system exclusive of decoupling capacitors})$$

The capacitors are therefore made component redundant by placing quad-redundant capacitors at each circuit position where a decoupling capacitor is required. Then,

$$R_{\text{capacitor}} > 0.99999.$$

Therefore:

$$R > 0.99999 \times (\text{Reliability of guidance computer exclusive of decoupling capacitors})$$

The decoupling capacitors are therefore ignored in the module component counts.

C. MEMORY COMPONENT COUNT

Due to the number of different circuits employed, the memory component failure rates are shown for only one typical circuit, the Memory EI Driver. (Table A-IV)

Table A-IV

MEMORY EI DRIVER COMPONENT FAILURE RATES

Component	Number	$\lambda \times 10^6$ Energized	$n \lambda \times 10^6$ Energized	$\lambda \times 10^6$ Nonenergized	$n \lambda \times 10^6$ Nonenergized
ULD TYPE COMPONENTS					
Transistor-Leadless	2	0.012	0.0240	0.0023	0.0046
Diode, 1/2 Dual-Leadless	5	0.007	0.0350	0.0011	0.0055
Diode, Dual-Leadless	2	0.012	0.0240	0.0020	0.0040
Resistor	6	0.013	0.0780	0.001	0.0060
Connection, ULD to MIB	21	0.001	0.0210	0.00028	0.00588
Connection, chip to URD per ball	21	0.0005	0.0105	0.0005	0.0105
Connection, single conductive pattern segment	12	0.0001	0.0012	0.0001	0.0012
Connection, wrap around	30	0.0005	0.0150	0.0005	0.0150
DISCRETE COMPONENTS					
Transistor, TI 1991, Dual NPN, matched pair	1	0.051	0.0510	0.0076	0.0076
Connector, tape cable per conductor	3	0.0156	0.468	0.0156	0.0468
Connector, 98 pin, per pin	1	0.0070	0.0070	0.0007	0.0007
Connector, solder	6	0.001	0.0060	0.00028	0.00168
		Total λ = 0.3195		Total λ = 0.10946	

Memory circuit failure rates are shown in Table A-V. Summing the circuit failure rates allows computation of $R_{sm, M}$

Table A-V

MEMORY CIRCUIT FAILURE RATES

Circuit Type	Number Used In One Simplex Memory	Energized $\lambda \times 10^6$	Energized $n \lambda \times 10^6$	Non Energized $\lambda \times 10^6$	Non Energized $n \lambda \times 10^6$
EI Driver	40	0.3195	12.78	0.10946	4.3784
EI - Serial	1	5.5084	5.5084	3.01072	3.01072
Inhibit Driver	14	0.35430	4.9602	0.10695	1.4973
Inhibit Driver Serial	1	0.9158	0.9158	0.71868	0.71868
Sense Amplifier	14	0.6284	8.7696	0.11698	1.63772
MCD-1	2	0.5812	1.1624	0.11740	0.23480
X-Y Connection	1	9.8880	9.8880	6.66240	6.6624
X-Y Decoupling	1	5.4400	5.4400	0.73984	0.73984
Memory Cores	1	11.4688	11.4688	11.4688	11.4688
VSG	1	0.4713	0.4713	0.09592	0.09592
MCD-2	4	0.4937	1.9748	0.09718	0.38872
MBR	14	0.4187	5.8618	0.09060	1.2684
MBR - Serial	1	0.924	0.9240	0.62430	0.62430
TCV	1	1.2175	1.2175	0.22130	0.2213
X-Y Term Res.	1	4.6080	4.6080	0.68352	0.68352
Error Detection	2	0.2172	0.4344	0.05006	0.10012
Total Failure Rate			76.385	33.73094	

$R_{sm, M} = 0.9849$ Using Equation (1)

Since the "modal failure equation" is to be employed, the simplex memory phase reliabilities must be calculated for all 11 phases. The first and last phases are evaluated with

$$R_j = \exp - \left[\lambda_e (T_e + KT_v) \right] \text{ (Refer to Section I of this Appendix)}$$

all other phases are evaluated with

$$R_j = \exp (- \lambda_j T_j) \text{ (Refer to Volume 1 Section III-A)}$$

Table A-VI summarizes the results of the simplex memory phase reliabilities.

Table A-VI

SIMPLEX MEMORY PHASE RELIABILITIES

Phase Number	Time (hours)	$\lambda \times 10^6$	Phase Reliability
1	16 + 50 (0.27)	76.385	0.99771
2	61	33.731	0.997943
3	10	76.385	0.99236
4	24	33.731	0.999190
5	8	76.385	0.999389
6	46	33.731	0.998448
7	9	76.385	0.999313
8	72	33.731	0.997571
9	5	76.385	0.999618
10	79	33.731	0.997335
11	6 + 15 (0.5)	76.385	0.999007

The "modal failure equation" also requires calculation of

$$P(f)_{nd, j} = \left[\frac{\lambda_{nd, j}}{\lambda_{mem, j}} \right] \times (1 - R_j)$$

and

$$P(f)_{d, j} = \left[\frac{\lambda_{d, j}}{\lambda_{mem, j}} \right] \times (1 - R_j)$$

The last three circuit types of Table A-V are those categorized as having nondetectable failure modes. So,

$$\frac{\lambda_{nd, e}}{\lambda_{mem, e}} = 0.0819 \text{ for phases during which memory is energized}$$

$$\frac{\lambda_{nd, n}}{\lambda_{mem, n}} = 0.0298 \text{ for phases during which memory is not energized}$$

Also,

$$\frac{\lambda_{d, e}}{\lambda_{mem, e}} = 1 - \frac{\lambda_{nd, e}}{\lambda_{mem, e}} = 0.9181 \text{ for phases during which memory is energized}$$

$$\frac{\lambda_{d, n}}{\lambda_{mem, n}} = 1 - \frac{\lambda_{nd, n}}{\lambda_{mem, n}} = 0.9702 \text{ for phases during which memory is not energized}$$

Table A-VII summarizes the failure rates of the memory circuits having nondetectable failure modes.

Extracting values from Tables A-VI and A-VII and inserting them into the "modal failure equation" yields

$$P(f)_{nd, M} = 0.000751$$

$$P(f)_{d, M} = 0.014410$$

Table A-VII

FAILURE RATES OF MEMORY CIRCUITS HAVING
NONDETECTABLE FAILURE MODES

Phase	$1 - R_j$	$\frac{\lambda_{d,j}}{\lambda_{mem,j}}$	$\frac{\lambda_{nd,j}}{\lambda_{mem,j}}$	$P(f)_{d,j}$	$P(f)_{nd,j}$
1	0.00229	0.9181	0.0819	0.00210	0.000188
2	0.002057	0.9702	0.0298	0.0020	0.000061
3	0.000764	0.9181	0.0819	0.00070	0.000063
4	0.00081	0.9702	0.0298	0.00079	0.000024
5	0.000611	0.9181	0.0819	0.00056	0.000050
6	0.001552	0.9702	0.0298	0.00150	0.000046
7	0.000687	0.9181	0.0819	0.00063	0.000056
8	0.002429	0.9702	0.0298	0.00236	0.000072
9	0.000382	0.9181	0.0819	0.00035	0.000031
10	0.002665	0.9702	0.0298	0.00259	0.000079
11	0.000993	0.9181	0.0819	0.00091	0.000081

D. POWER SUPPLY

The power supplies contain two types of simplex modules:

1. The power converter. Its component failure rates are given in Table A-VIII.
2. The d-c feedback amplifier. Its failure rates are given in Table A-IX.

Table A-VIII

NONREDUNDANT POWER CONVERTER COMPONENT FAILURE RATES

Component Type	Number	Energized		Unenergized	
		$\lambda \times 10^6$	$n\lambda \times 10^6$	$\lambda \times 10^6$	$n\lambda \times 10^6$
Q (Power)	2	0.14	0.28	0.0036	0.0072
Q (Non-Power)	6	0.017	0.102	0.0036	0.0216
CR (Power)	2	0.1	0.2	0.0015	0.003
CR (Non-Power)	3	0.008	0.024	0.0008	0.0024
C (Glass)	3	0.001	0.003	0.00004	0.00012
C (Tantalum)	2	0.06	0.12	0.0014	0.0028
R (Carbon Composition)	18	0.003	0.054	0.003	0.054
R (Precision)	2	0.022	0.044	0.003	0.006
L	1	0.12	0.12	0.002	0.002
T (Signal)	2	0.43	0.86	0.004	0.008
T (Power)	1	0.7	0.7	0.004	0.004
Hardware		0.493	0.493	0.493	0.493
Totals		3.0		0.60412	

$$R_{conv} = 0.99961$$

As indicated in Section III-C of the Reliability Analysis, the single supply reliability is to be computed assuming that the dual d-c amplifiers are not redundant. Thus, the failure rates for a single supply are:

$$\lambda_{ss} (\text{energized}) = \lambda_{\text{converter}} (\text{energized}) + 2 \lambda_{\text{d-c amplifier}} (\text{energized})$$

and

$$\lambda_{ss} (\text{nonenergized}) = \lambda_{\text{converter}} (\text{nonenergized}) + 2 \lambda_{\text{d-c amplifier}} (\text{nonenergized})$$

The reliability of a single supply is then

$$R_{ss} = 0.99937 \quad \text{using equation (1)}$$

Table A-IX

NONREDUNDANT DIRECT-CURRENT AMPLIFIER
COMPONENT FAILURE RATES

Component Type	Number	Energized		Unenergized	
		$\lambda \times 10^6$	$n\lambda \times 10^6$	$\lambda \times 10^6$	$n\lambda \times 10^6$
Q	4	0.017	0.068	0.0036	0.0144
CR (Zener)	1	0.060	0.060	0.003	0.003
C (Glass)	1	0.001	0.001	0.00004	0.00004
R (Carbon Composition)	8	0.003	0.024	0.003	0.024
R (Precision)	4	0.022	0.088	0.003	0.012
P (Potentiometer)	1	0.15	0.15	0.03	0.03
Hardware		0.1876	0.1876	0.1876	0.1876
Totals			0.5786		0.27104

$$R_{amp} = 0.99988$$

The expressions of Section III-C also require the reliability of a d-c amplifier:

$$R_{amp} = 0.99988$$

E. DATA ADAPTER BINARY LOGIC

The approximate component failure rates for one simplex channel of the Data Adapter binary logic are shown in Table A-X.

Dividing the channel failure rates by six gives the module failure rates as per Section III-C of the Reliability Analysis.

$$R_{mod} = 0.9937 \text{ using equation (1)}$$

Table A-X

DATA ADAPTER SIMPLEX CHANNEL COMPONENT FAILURE RATES

Item	Quantity	Energized		Unenergized	
		$\lambda \times 10^6$	$n \lambda \times 10^6$	$\lambda \times 10^6$	$n \lambda \times 10^6$
Pages	15-1/2	1.106	17.143	1.106	17.143
Connectors	10	0.689	13.091	0.689	13.091
Back Panel	1-1/3	3.762	5.015	3.762	5.015
Substrates	709	0.0246	17.441	0.0246	17.441
Delay Line	2	0.3	0.6	0.0025	0.005
Transistors	585	0.012	7.020	0.0023	1.345
Diodes	3651	0.006	21.906	0.001	3.651
Resistors	2155	0.013	28.015	0.001	2.155
Flex Cables	34	1.0	34.000	1.0	34.000
		Total $\lambda = 144.231$		Total $\lambda = 93.846$	

λ per module = 24.038 (energized), 15.641 (unenergized)

F. OUTPUT DRIVER COMPONENT FAILURE RATES

The component failure rates for a simplex output driver channel are shown in Table A-XI.

Substituting the values into equation (1):

$$R_{ch} = 0.99974$$

G. ACCUMULATOR COMPONENT FAILURE RATES

The approximate component failure rates for a simplex channel of the accumulator logic are shown in Table A-XII. Since this logic is energized for the entire mission, only energized failure rate summations are obtained. Dividing the total failure rate by three gives the module failure rate as called for in Section III-C. Using equation (1):

$$R_a = 0.9930$$

Table A-XI

**SIMPLEX OUTPUT DRIVER CHANNEL
COMPONENT FAILURE RATES**

Component Type	Number	Energized		Unenergized	
		$\lambda \times 10^6$	$n \lambda \times 10^6$	$\lambda \times 10^6$	$n \lambda \times 10^6$
Transistors	6	0.012	0.072	0.0023	0.0138
Resistors	19	0.013	0.247	0.001	0.019
Diodes	12	0.006	0.072	0.001	0.012
Pulse Trans- formers	1	0.16	0.16	0.004	0.001
Hardware			0.56		0.56
Total $n \lambda \times 10^6$		1.111		0.6088	

Table A-XII

**SIMPLEX ACCUMULATOR CHANNEL
COMPONENT FAILURE RATES**

Component Type	Quantity	Energized	
		$\lambda \times 10^6$	$n \lambda \times 10^6$
Pages	7-1/3	1.106	8.1107
Flex Cables	18	1.0	18.0
Connectors	10	0.689	6.89
Back Panels	1/3	3.762	1.254
Substrates	380	0.0246	9.348
Delay Lines	2	0.3	0.6
Transistors	200	0.012	2.4
Diodes	743	0.006	4.458
Resistors	586	0.013	7.618
Total $n \lambda =$		58.68	

Appendix B

“RANDOM NUMBER” RELIABILITY CALCULATION

Appendix B

"RANDOM NUMBER" RELIABILITY CALCULATION

In Section III. D. 1 it was stated that the Monte Carlo program used a random number process to generate sets of failed components. This process is described herein.

When a constant failure rate for a component is assumed, it implies that the density function for the component is

$$u_c(t) = \exp(-\lambda t)$$

Considering a set of n similar components, the density function for the random variable t (time to first component failure) is

$$u_1(t) = n\lambda \exp(-n\lambda t)$$

The operation of a system can be simulated by finding some experiment that generates a random variable t having the density function $u(t)$, and then performing the experiment to obtain a value t_1 . This then represents the time of first failure for the n components. If t_1 is less than the mission duration T_M , one of the n components is considered to have failed. At time t_1 , then, one component has failed and $n-1$ components have survived. The density function for the random variable t (elapsed time from first failure to second component failure) is

$$u_2(t) = (n-1)\lambda \exp[-(n-1)\lambda t]$$

The experiment is then repeated (modified for $u_2(t)$) to obtain t_2 , and the total elapsed time is $t_1 + t_2$. If $t_1 + t_2$ is less than T_M , a second component has failed. Then the experiment is modified (reducing the component count by one) and t_3 is generated. This process is continued until a t_k is generated such that

$$\sum_{i=1}^k t_i > T_M \quad \left(\sum_{i=1}^{k-1} t_i < T_M \right)$$

When the total elapsed time exceeds T_M , a component has not failed. The number of component failures that were recorded just prior to generating t_k such that

$$\sum_{i=1}^k t_i > T_M$$

is then the set of failures for the n similar components for this simulation or "game." This process is repeated for each different set of components in the system thus producing the total set of failed components for the mission simulation or game.

A problem exists when trying to find an experiment (other than actually testing the system for many hours) that will generate random variables having the needed density functions $u_1(t), u_2(t), \dots, u_k(t)$. It is possible, however, to program a computer such as the IBM 7090 EDPS to generate a random variable r with density function

$$u(r) \equiv 1 \quad 0 \leq r \leq 1$$

$$u(r) \equiv 0 \quad 1 < r < \infty$$

(r , then, is a "random number" taking on values between zero and one).

Consider the function $t = f(r)$. Since t is a function of the random variable r , t is also a random variable possessing a density function $u(t)$. Knowing the density function $u(r)$, it is necessary to determine $u(t)$ given that $t = f(r)$. Although proof is not given here, it can be shown that if

(1) $u(r)$ is defined for $-\infty < r < \infty$ and is continuous except at isolated points, and

(2) $f(r)$ is a differentiable monotonic function, then

$$u(t) = \frac{d}{dt} \int_{-\infty}^{g(t)} u(r) dr \quad (1)$$

where g is the inverse function of f . That is

$$g[f(r)] \equiv r$$

We would then like to find an $f(r)$ that would give an exponential $u(t)$. Consider the function

$$t = f(r) = \frac{-\log_{\epsilon}(1-r)}{n\lambda}$$

Obtaining the inverse function,

$$-n \lambda t = \log_e (1-r)$$

$$1 - \exp (-n \lambda t) = r = g(t)$$

Substituting into Equation (1),

$$\begin{aligned} u(t) &= \frac{d}{dt} \int_{-\infty}^{1 - \exp (-n \lambda t)} 1 dr \\ &= \frac{d}{dt} \int_0^{1 - \exp (-n \lambda t)} 1 dr \quad (\text{since } u(r) \equiv 0 \text{ For } r < 0) \\ &= \frac{d}{dt} [1 - \exp (-n \lambda t)] \\ &= n \lambda \exp (-n \lambda t) \end{aligned}$$

Thus the random variable

$$t_1 = \frac{-\log_e (1-r)}{n \lambda}$$

has the correct density function for the random variable time-to-first component failure.

Similarly,

$$t_2 = \frac{-\log_e (1-r)}{(n-1) \lambda}$$

can be used to generate the time to second component failure.

So, the IBM 7090 EDPS produces the failure set for each type of component by first generating a value for the random variable r (i. e., random number between zero and one). Then t_k is obtained using

$$t_k = \frac{-\log (1-r)}{(n-k+1) \lambda}$$

Once the failure set is generated, a failed component is considered to be in the open state if

$$0 \leq r < K_0$$

where r is another random number between zero and one, and K_0 is the probability that if the component fails, it will fail open. If

$$K_0 \leq r \leq 1$$

the component has shorted.

All components of a given type in the particular system under consideration are tagged with numbers ranging from one through n . The specific number of a failed component (N) is then obtained with

$$N = \text{integer part of } r(n + 1)$$

where, again, r is a random number between zero and one (actually, a value of the previously defined random variable r).

The effects of the component failures are traced to determine whether or not the system fails. If many such games are played, the reliability of the system is estimated by

$$R^* = \frac{\text{number of games in which system did not fail}}{\text{total games played.}}$$

It should be noted that the only assumption made was a constant component failure rate. It is possible to use the same technique to handle the step-function failure rates assumed in this report (such is the case for IBM's Monte Carlo program). Thus, given the component failure rate functions, the Monte Carlo technique can be used to estimate the reliability of an actual system configuration, simplex or redundant, without making any simplifying assumptions.

However, it must be recognized that R^* is a statistical estimate of R , the true but unknown probability of success. The question is, "how good an estimate?"

R^* is random variable with mean value R and variance

$$\frac{R(1 - R)}{N}$$

where N is the number of Monte Carlo games played. If N is large enough to satisfy the inequality

$$N - NR \geq 3\sqrt{NR(1-R)} \quad (2)$$

then R^* can be assumed to have approximately a Normal or Gaussian distribution about its mean, R . A standard Normal random variable is created from R^* by the transformation

$$\frac{R^* - R}{\sqrt{\frac{R(1-R)}{N}}}$$

Given a confidence level γ , a table of "Areas under the standard Normal distribution" is entered to find K_γ , which is the number of standard deviations that must be taken on both sides of the mean to include γ of the area under the curve. Then, the probability is γ that

$$-K_\gamma \leq \frac{R^* - R}{\sqrt{\frac{R(1-R)}{N}}} \leq K_\gamma \quad (3)$$

Solving this equation for R gives:

$$\frac{N}{N + (K_\gamma)^2} \left[R^* + \frac{K_\gamma^2}{2N} - K_\gamma \sqrt{\frac{R^*(1-R^*)}{N} + \left(\frac{K_\gamma}{2N}\right)^2} \right] \leq R \leq \frac{N}{N + (K_\gamma)^2} \left[R^* + \frac{K_\gamma^2}{2N} + K_\gamma \sqrt{\frac{R^*(1-R^*)}{N} + \left(\frac{K_\gamma}{2N}\right)^2} \right] \quad (4)$$

To determine the confidence interval for an N game Monte Carlo simulation, N , R^* , and K_γ are inserted into Equation (4). The upper limit for R from Equation (4) is inserted into Equation (2). If the inequality is satisfied, it can be stated that R has a probability of γ of being in the interval given by Equation (4). A typical simulation yielded $R = 0.9986$ with $N = 10,000$ games. With a confidence level or probability of $0.9 (\gamma)$, solution of Equation (4) yields

$$0.9979 \leq R \leq 0.9991$$

Inserting 0.9991 into Equation 2 satisfies the inequality:

$$9 \geq 3 \sqrt{0.9991 \times 9}$$

It is therefore valid to state that the probability is 0.9 that $0.9979 \leq R \leq 0.9991$.

IBM **IBM** IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM