AD NUMBER
ADB010399
LIMITATION CHANGES
TO: **Approved for public release; distribution is unlimited. Document partially illegible.**
FROM: **Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; 28 JAN 1976. Other requests shall be referred to Office of Naval Research, Attn: Code 211, 875 N. Randolph Street, Arlington, VA 22203-1995. Document partially illegible.**
AUTHORITY
ONR per ltr, 2 Aug 1977

THIS REPORT HAS BEEN DELIMITED
AND CLEARED FOR PUBLIC RELEASE
UNDER DOD DIRECTIVE 5200.20 AND
NO RESTRICTIONS ARE IMPOSED UPON
ITS USE AND DISCLOSURE.

DISTRIBUTION STATEMENT A

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED.

REPORT ONR-CR233-052-2

2

MODULAR DIGITAL MISSILE GUIDANCE

PHASE II REPORT

BRUCE A. HALL
FRANK J. LANGLEY

RAYTHEON COMPANY
MISSILE SYSTEMS DIVISION
BEDFORD MASS., 01730

CONTRACT N00014-75-C-0549
ONR TASK 233-052

28 JANUARY 1976

TECHNICAL REPORT FOR PERIOD 20 JAN 75 - 19 NOV 75

PREPARED FOR THE

OFFICE OF NAVAL RESEARCH ● 800 N. QUINCY ST. ● ARLINGTON ● VA ● 22217

NOTICES

Change of Address

Organizations receiving reports on the initial distribution list should confirm correct address. This list is located at the end of the report. Any change of address or distribution should be conveyed to the Office of Naval Research, Code 211, Washington, D.C. 22217.

Disposition

When this report is no longer needed, it may be transmitted to other authorized organizations. Do not return it to the originator or the monitoring office.

Disclaimer

The findings in this report are not to be construed as an official Department of Defense or Military Department position unless so designated by other official documents.

Reproduction

Reproduction in whole or in part is permitted for any purpose of the United States Government.

B

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ONR-CR233-052-2	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): Modular Digital Missile Guidance, Phase II.	5. TYPE OF REPORT & PERIOD COVERED: Technical Report, 20 Jan - 19 Nov 1975,
	6. PERFORMING ORG. REPORT NUMBER: BR-8738
7. AUTHOR(s): Bruce A./Hall, Frank J./Langley, et al	8. CONTRACT OR GRANT NUMBER(s): N00014-75-C-0549
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Raytheon Company, Missile Systems Division, Bedford, Mass 01730	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research, Vehicle Warfare Technology Code 211, Arlington, Virginia 22217	12. REPORT DATE: 28 Jan[illegible] 1976
	13. NUMBER OF PAGES: 449
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution limited to U.S. Government Agencies only; Test and Evaluation; 28 JAN 76. Other requests for this document must be referred to the Office of Naval Research, Code 211.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Missile Guidance
Digital Control
Computers

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report presents the results of the second phase of a study to investigate the feasibility of a modular digital guidance system for Navy air-to-air missile applications. The studies involved the analysis of functions for digital implementation in all classes of air-to-air missiles and the derivation of computer requirements in terms of throughput memory, architectural features, modularity and compatible

software characteristics.

The functions of: target seeker head control, estimation, guidance and autopilot were addressed in the first study phase and those of seeker signal processing, fuzing, telemetry, test and flight phase/mode control were analyzed in the second study phase reported herein. In addition, simulation analyses of estimation, guidance and autopilot algorithms were performed to determine performance improvement as a function of complexity.

In summary, the studies have shown that modular digital guidance and control is both feasible and effective in improving missile performance and flexibility to counteract changing threat situations and advancing technology. Using a common bus interface, a family of ten major computer function elements, hybrid large-scale-integrated (LSI) circuit macromodules, in various configurations, will support the entire range of air-to-air missile functions. Federated microcomputer systems best match missile function with computer capability, providing desired subsystem autonomy for modular design, manufacture, assembly, test, maintenance and subsequent modification without system disruption. Throughput is driven by radar signal processing which can be accommodated by optimized central processing unit (CPU) macromodules, incorporating either a hardware multiplier or two-point complex transform arithmetic unit. A common higher-order language for system simulation and missile computer code generation, together with structured design and modularity, minimize software cost and risk.

PREFACE

This technical report covers the work performed under Contract No. N00014-75-C-0549 from 20 January 1975 through 19 November 1975. The first draft was submitted by the authors on 30 January 1976.

The purpose of this contract together with the work performed in Phase I was to provide a basis for designing modular air to air missile guidance systems with improved performance, flexibility and growth features compared to traditional analog and early digital implementations.

Cdr. P.R. "Bob" Hite, and Mr. David S. Siegel, Office of Naval Research, Vehicle Technology Programs, Technology Projects Division, Arlington, Va., were the Navy Scientific Officers.

Dr. B.A. Hall, was the Program Manager for Raytheon supported by the following study team members:

F.J. Langley	Lead Engineer Modular Computer Definition
L.J. Casey	Autopilot, Telemetry, Test and Mode Control
T.J. Cullinane	Fuzing
R.J. Fitzgerald J.H. Nickerson	Estimation and Guidance

K.D. Wefald
J.D. Hadad
J.E. Hopson

Seeker Signal Processing

Publication of this report does not constitute Navy approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF TABLES

LIST OF TABLES

LIST OF TABLES

LIST OF TABLES

1. INTRODUCTION

1.1 Background

The design, development and production of missiles to cover a range of presently defined missions with the capability of being upgraded to accomodate changing threat situations and advancing technology without major redesign, stresses the need for more modular guidance and control electronics possessing both physical and electrical flexibility features at lowest cost.

Figure 1 illustrates the functional complement typical of air to air missiles.

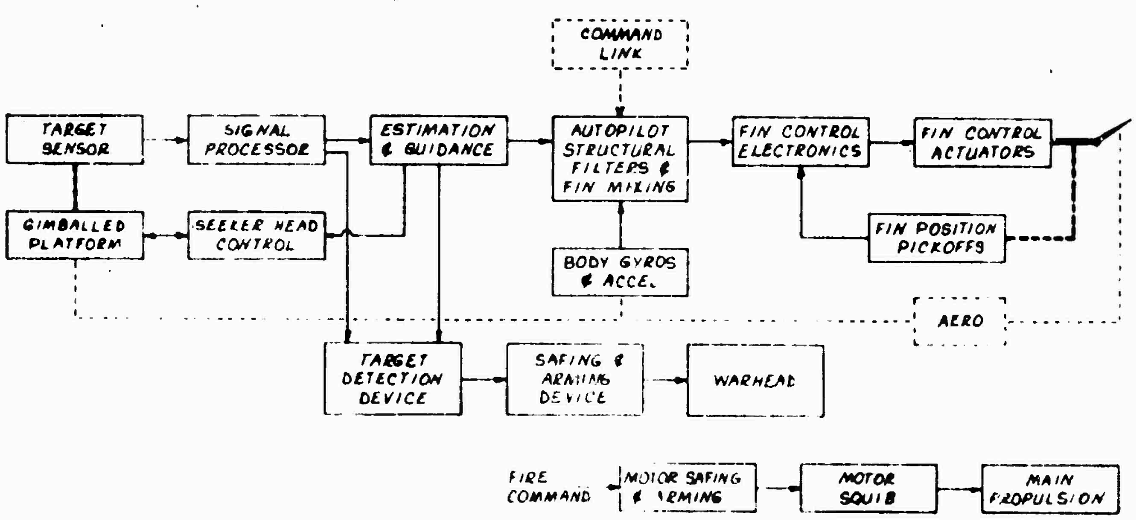


Figure 1, On-Board Missile Guidance and Control System

In the previous study phase, (Ref. R-1), programmable digital techniques were shown to offer improved performance and greater flexibility than the traditional hardwired analog implementations of seeker head control, estimation, guidance and autopilot functions.

To achieve modularity and growth in hardware and software a top-down system study approach has been adopted by first dividing the entire range of air to air missiles into a set of distinguishable generic classes, including upper and lower performance boundaries within each class, then by: defining the major functions and data rates amenable to digital processing, determining their constituent software modules and sizing these in terms of computer throughput and memory requirements.

Such a modular breakdown of on-board missile guidance and control functions together with their associated interfaces, provided the option of configuring and evaluating either single or multiple federated/distributed computer system implementations according to the design constraints of a given missile.

1.2 Objectives and Scope

The objectives and scope of the Phase II study under contract N00014-75-C-0549, as defined in the Statement of Work, are as follows:

To investigate the feasibility of a modular digital guidance system to Navy air-to-air missile applications by:

a) Analyzing for digital implementation in all classes of air-to-air missiles the functions of seeker signal processing for both infrared (IR) and radar type seekers, warhead fuzing, mode control and telemetry.

b) Performing a simulation analysis to confirm computer requirements and relate algorithm complexity to performance improvements for the guidance, estimation, autopilot/control functions as defined under contract N00014-74-C-0056.

c) Updating the computer requirements per generic class of air-to-air missile based upon the results of (a) and (b) above.

The intention of the above being to conclude the major function analysis work started in the first phase, (contract N00014-74-C-0056) thereby defining the total practical digital processing and control requirements for each generic class of missile and providing/identifying continuity and commonality features across the entire spectrum of air-to-air missiles. Further, since the Phase I study resulted in the definition of improved estimation, guidance and autopilot algorithms compared to the more simple analog counterparts, simulations were required to ascertain the degree of performance improvement as a function of complexity from a computer load aspect.

Early in the Phase II study it became evident that greater emphasis should be placed on the definition of compatible computer hardware and software characteristics to achieve more timely visibility in this critical area. In response to a request by the Navy Scientific Officer a revised program plan was developed and presented at ONR. The revised study plan was formally approved by ONR on 29 May 1975, confining the simulation analysis work to a Class II missile with three degrees of freedom and initiating more comprehensive computer hardware and software studies in June 1975.

2. SUMMARY AND CONCULSIONS

As a result of completing the functional analyses, qualifying performance improvements versus digital processing capacity, and defining compatible hardware and software features for effective modular digital guidance and control, the following significant results and conclusions can be stated:

1. Modular, programmable, digital guidance is feasible, affords performance improvements and provides flexibility, modular expansion and system updating without major redesign.
2. A family of ten, major computer function elements, hybrid large-scale-integrated (LSI) circuit macromodules, in various configurations, using a common bus interface, will support the entire range of air-to-air missile functions.
3. Radar sensor signal processing dominates the throughput requirement and can be supported by an "optimised" central processing unit (CPU) macromodule incorporating either hardware multiplier or two-point, complex transform arithmetic unit.
4. Federated/distributed microcomputer systems provide the the best match of missile functions with computer capability, providing desired subsystem autonomy for modular design, manufacture, assembly, test, maintenance and subsequent modification without system disruption.

5. Missile guidance and control systems readily partition into four autonomous and asynchronous functional groups for modular, federated computer systems:

   a. Steering command generation (signal processing, estimation and guidance).

   b. Missile stabilization and control (autopilot and inertial reference).

   c. Seeker stabilization and control (tracking and stabilization)

   d. Support functions (fuzing and telemetry)

6. Serial digital multiplex as defined in MIL-STD-1553, provides an optimum interface between missile subsystems/computers and carrier aircraft avionics.

7. Unified software system using one high-order-language for system simulation and missile computer code generation together with structured design and modularity minimize software cost and risk.

The findings of the individual study tasks are discussed in greater detail in the following paragraphs.

2.1 Functional Analysis

Target Sensors - Digital signal processing and mode control requirements for target sensors of the radar, anti-radiation-missile (ARM) and infra-red (IR) types have been determined.

Semi-active, continuous-wave (SA-CW), semi-active pulse doppler (SA-PD) and active pulse doppler (A-PD) radars use digital processing to the greatest advantage. The inherent optical/analog processing in IR sensors limits digital processing to mode and logic functions. The wide bandwidths and short pulse widths processed in ARM sensors are more efficiently done in the analog domain. Digital mode control is only used for ARM sensors.

Radar digital processing consists of spectrum analysis, via discrete Fourier transform techniques and the fast Fourier transform (FFT) algorithm, detection (thresholding and integration), range, doppler, angle extraction and mode logic. Table 1 summarizes the nominal digital processing requirements. The radar design parameters are specified in Section 4.

TABLE 1

RADAR SENSOR PROCESSING REQUIREMENTS

PARAMETER	SA-CW	RADAR TYPE SA-PD	A-PD
A-D Conversion Rate (KHz)			
Acquisition	25.6	128	256
Track	19.2	32	32
FFT Size (No. of points)	64	64	64
*FFT & PDI Throughput (Mops)	2	10	20
*GP Throughput (Kops)	60	300	950

NOTES: *See Section 7. Table 70 for redistribution of FFT/PDI vs GP throughputs to match performance of state-of-the-art computer architectures.

The doppler resolution process with FFT and post detection integration (PDI) dominate the throughput requirements, requiring 2 to 20 million instructions per second which is supportable by a general-purpose (GP)-type computer architecture using a hardware multiply or two point transform arithmetic module in support/place of the normal GP computer arithmetic and logic unit. Data extraction (range, doppler, angle) mode logic are accomplished in a conventional general-purpose computer configuration which may be shared with other functions such as guidance and estimation.

Fuzing - Three fuze types are identified as appropriate to air-to-air missiles. These are:

TABLE 2

FUZE TYPES vs MISSILE CLASS

FUZE TYPE	MISSILE CLASS I	II	III
Semi-active CW	X	X	
Active Radar		X	X
Active Optical	X		

The greatest impact of digital processing is in the timing of the firing command to the safing and arming (S&A) device. The use of sophisticated timing algorithms using data from the estimator function permits improved fuze performance. The impact on processing is minimal ranging from 1 to 25 Kops and up to 250 words of memory.

Mode Control - Mode control is the selection and execution of a specific set of missile control functions (e.g., seeker control, estimation, guidance, etc) in each mode of missile operation (e.g., test, initialize, launch, target acquisition, midcourse, terminal etc). This type of mode control is more applicable to the single computer missile system where all missile functions must be executed sequentially. A hierarchical structure is used where calls are made downward from the executive to subordinate program modules to select and execute the functions pertinent to the active mode. Individual mode supervisors select all functions and utility program modules involved in the particular mode. Mode control is a minor load on computer throughput and memory. Maximum values are 53 Kops and 630 words.

Telemetry - Telemetry is a data gathering and formatting process. It overlays all other functions and must not interfere with system operations. In a digital missile all data to be telemetered is available in computer memory normally for use by the major functions, if not, it is brought in through the input-output (I/O) interface. Serial transmission using pulse code modulation is normally used. Telemetry data rates, throughput and memory required are indicated in Table 3.

TABLE 3

TELEMETRY REQUIREMENTS

	MISSILE CLASS		
PARAMETER	I	II	III
Serial Bit Rate (kbits/sec)	12	24	40
Throughput (Kops)	16	32	32
Memory (words)	100	110	120

Test - Readiness tests in a digital missile are conducted by test program modules in the missile computer(s) in response to the launch aircraft test inputs via the digital umbilical. Tests are executed as off-line functions without severe timing constraints and with memory requirements being the chief consideration. Testing includes each missile computer, all function programs, I/O interfaces, telemetry and the seeker and missile control servos. In single computer systems, testing requires an operating computer. For federated systems the avionics-missile test command is distributed to each subsystem computer. A total of 12 test modules are defined, divided into 3 categories, computer self test, interface test and subsystem test. Memory requirements are: Class I, 360; Class II, 450; Class III, 700.

2.2 Performance versus Processing

In designing digital missiles information is needed as to the improvements in performance (miss distance, and signal to noise ratio (SNR) for target acquisition) that are achieved through increasing digital processing capacity (computer throughput, sampling rates). Performance processing summary tradeoff results are shown in Figures 2 and 3.

Guidance miss can be reduced by increasing the guidance acceleration command update rate, (f_{GUID}) and/or increasing the complexity of estimation and guidance algorithms used, as indicated in Figure 2, at the expense of increased computer throughput. Relative to the throughput requirements of other functions (signal processing, autopilot), guidance and estimation throughput is modest so that if terminal accuracy is of prime importance, the use of higher guidance sampling frequencies and Kalman filters (Class II & III) are indicated.

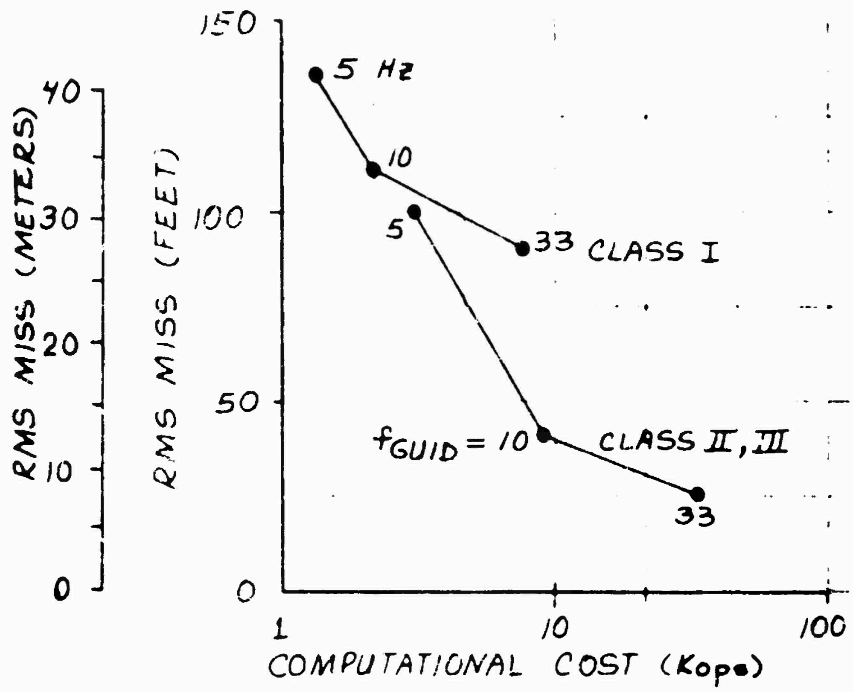


Figure 2 RMS Miss Distance vs Estimation and Guidance Computational Throughput

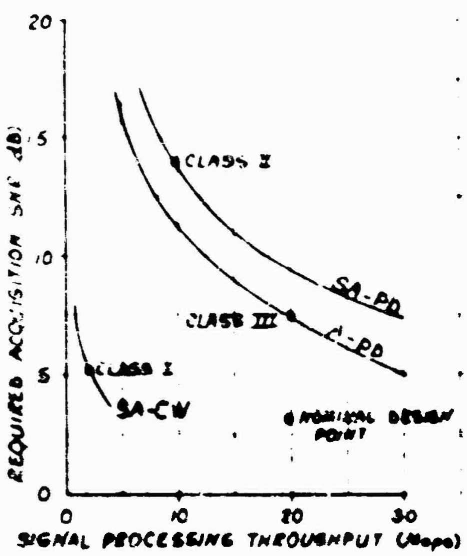


Figure 3 Required SNR for Target Acquisition vs Digital Signal Processing Throughput

Similarly target sensor acquisition performance can be improved through increased digital processing. Figure 3 shows the reduction in SNR required for acquisition achieved by increasing the number of range gates and FFT points used to cover the initial range-doppler target uncertainty. Throughput in the digital signal processing computer elements increases accordingly. The SA-CW sensor requires the least SNR and throughput since it is resolving in doppler only. However pulse-doppler sensors which resolve in both range and doppler perform better in a clutter environment generally so that they are used in Class II and III missiles. Note that the throughput for signal processing is substantially greater (Mops) than for guidance and estimation (Kops) so that firmware algorithms and higher speed arithmetic hardware, i.e., hardware multiply or two-point "butterfly" FFT arithmetic modules are required.

2.3 Computer Requirements

The analysis of functional requirements and tradeoff of performance versus processing load yields a set of computer requirements for the three generic missile classes.

Throughput in terms of thousands (K) or millions (M) of operations per seconds (ops) are summarized in Figure 4. Throughput is computed over the time intervals allowed for each function, being either in a critical (time limited) path or multiplexed over a major computing interval.

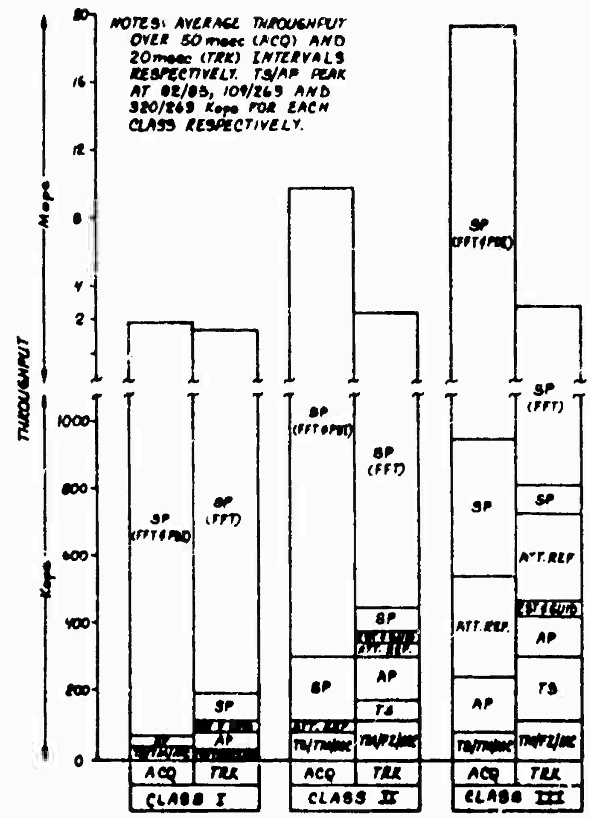


Figure 4 Throughput Requirements by Function and Generic Class.

Signal processing required for target detection, acquisition and track is shown separately from the more general-purpose load, since FFT and PDI would be implemented using CPU macromodules optimised for these functions. By proper allocation of the total allowable computing time delay for steering command generation between the FFT/PDI functions and the other signal processing estimation and guidance computations, a balance in computing load is achieved to match the performance capabilities of more special-purpose versus general-purpose computer configurations. Both the special and general-purpose computational loads are maintained nearly constant over both modes.

Total memory required per class is shown in Figure 5. The major memory driver is radar signal processing involving data buffering for FFT and PDI functions in all classes. Program memory requirements for estimation and autpilot functions are significant for the Class III missile.

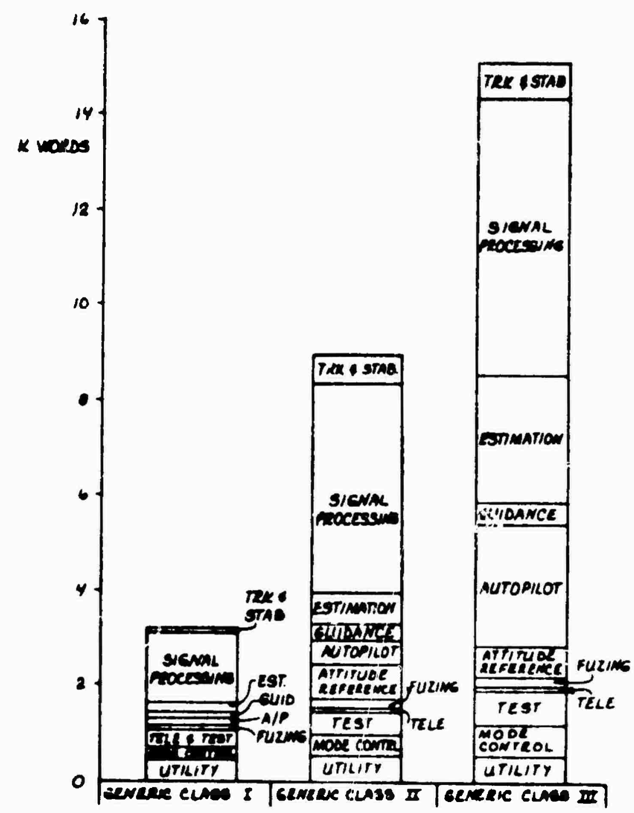


Figure 5 Memory Requirements by Function and Generic Class

In addition to memory and computational throughput, requirements are established on sampling rates, allowable delays and precision (quantization). Table 4 lists the minimum sampling rates.

TABLE 4

DATA SAMPLING/UPDATE RATE REQUIREMENTS

FUNCTION	MISSILE CLASS I	II	III
Signal Processing (1)			
Acquisition (2) (KHz)	25	128	256
Track (2) (KHz)	20	32	32
Seeker Tracking			
Estimation & Guidance (Hz)	10-20	10-30	20-40
Seeker Stabilization (Hz)	250	250-500	500
Autopilot			
Rate Loops (Hz)	250	500	500
Accelerometer Loops (Hz)	125	250	250
Gains (Hz)	5	10-20	20-25
Inertial Reference			
Attitude (Hz)	N/A	100	100-500
Position and Velocity (Hz)	N/A	20	20-250
Aerodata Estimation (Hz)	N/A	20	20-100

NOTES:

(1) Radar sensors.

(2) Composite, multiplexed A-D conversion of all channels.

Two critical computational delay requirements are established. One is the elapsed time from receipt of one burst of data from the target sensor and the update of the guidance command. This time delay is limited to 20-40 msec depending upon missile class and intercept scenario, the shorter times required for Class III, highly maneuvering targets and higher altitudes and closing velocities. The second critical delay is in the closure of the autopilot rate stabilization loop and the seeker stabilization loops not less than 600 sec for Classes II and III and 600 sec for Class I.

Precision of conversion and computation are established by accuracy, dynamic range and stability considerations.

For conversion of analog receiver data generally 8 to 12 bits is required to preserve signal to noise ratio. For estimation, particularly the time variable Kalman Filters, floating-point computations with mantissa lengths of at least 12 bits are required. For autopilot and seeker stabilization loops, 16 bits of fixed-point precision is required to maintain stability and accuracy of compensating network pole and zero locations.

2.4 Computer Hardware and Software

Digital missile functions are fully supported through the use of a modular family of computing elements, called macromodules, which can be combined via a standarized interface to form computers of various capacities and functional capability. Ten basic macromodules defined in Table 5 have been identified from digital missile requirements which cover the spectrum of applications. From these basic modules, a variety of computers can be configured. Six comfigurations which cover the range of missile classes and functions are shown in Figure 6.

In terms of software, the emphasis has shifted from "tight" assembly language coding, used to minimize bulky magnetic core memory space and conserve throughput, to lower software cost for design, coding, verificaton and maintenance/updating. The latter has spurred the need for a commom higher-order language and structured, modular software design, to achieve simplicity and visiblity in the coding process and machine independence for portability and re-use of proven programs and program modules.

Conservation of throughput and memory space has been de-emphasized through the availability of high-density, large-scale-integrated (LSI) semiconductor circuits, which tend to absorb the inefficiencies of HOL-generated programs in terms of size, weight, power and cost penalties.

Figure 7 illustrates a recommended unified approach to missile simulation and tactical software development using a common high-order language and host computer. Code written for simulating guidance and control functions is also used by a cross-compiler to generate object code for the missile computer.

TABLE 5

MACRO-MODULAR MICROCOMPUTER

LSI MACRO-FUNCTION MODULES

MODULE GROUP	LSIC MODULE	NAME	CHARACTERISTICS	MODULAR MICROCOMPUTER APPLICATION
1	RAM-1	Random Access Memory	High-speed ≯ 100 nsec access.	High-speed data storage, single computer systems. Signal processing.
	RAM-2	Random Access Memory	Medium-speed, ≯ 500 nsec access	Medium-speed data storage federated computer systems.
2	PROM-1	Electrically Programmable Read-Only Memory	High-speed, field programmable. 100 nsec access.	High-speed macro/micro program storage, single computer systems.
	PROM-2	Electrically Programmable Read-Only Memory	Medium-Speed, field programmable/ re-programmable 500 nsec access/	Medium-speed macro program storage federated computer systems.
3	CPU-1	Central Processor Unit	Medium-speed, 8-bit byte, 2 sec register-register add, fixed instruction set.	Medium-speed federated computer systems
4	RALU-1	Register Arithmetic and Logic Unit Slice	High-speed, 4-bit slice, general-register, 300 nsec cycle	High-speed, single computer systems, word lengths from 4 to 32 bits.
5	FFT-1	2-Point Complex FFT Arithmetic Unit	High-speed, 8+j8, pipeline, 200 nsec/operation.	High-speed, dedicated FFT processors.
6	HWCU-1	Hardwired Control Unit	High-speed, fixed instruction set.	High-speed, low-parts count, central, federated, or FFT computer systems
7	PCU	Microprogrammed Control Unit	Medium-speed, 512 microinstruction addressing.	Medium-high speed single computer and FFT processor control
8	HMPY-1	Hardware Multiplier	High-speed, parallel multiplier. 200 nsec 8x8 bit.	High-speed gp computers performing FFT functions. Multiply speed improvement for other configs.
9	ADAC-1	Analog-Digital, Digital-Analog Converter	Medium-speed, 12-bit A-D and D-A converter with multiplexed inputs and outputs, 0-100 KHz	Multiplexed, low frequency, guidance and control analog input-output interface
	PDIO-1	Parallel Digital (programmed) Input-Ouput Channel	Low-Speed data/command transfers under program control	Initializing DMAIO and interfacing with SDIO
	SDIO-1	Serial Digital Input-Output Channel	Medium-speed (up to 1Mbps) word serial, bit-serial data/command transfers.	Inter-subsystem/computer, modem and avionics interface
	DMAIO	Direct Memory Access Input-Output Channel	High-speed data transfers to/from main memory at memory cycle rate.	High-speed signal processor I/O and multiplexed stability loop I/O transfers.
10	PLA	Programmable Logic Array	Combinational Logic networks	Dedicated controllers and decoders e.g. missile up-link

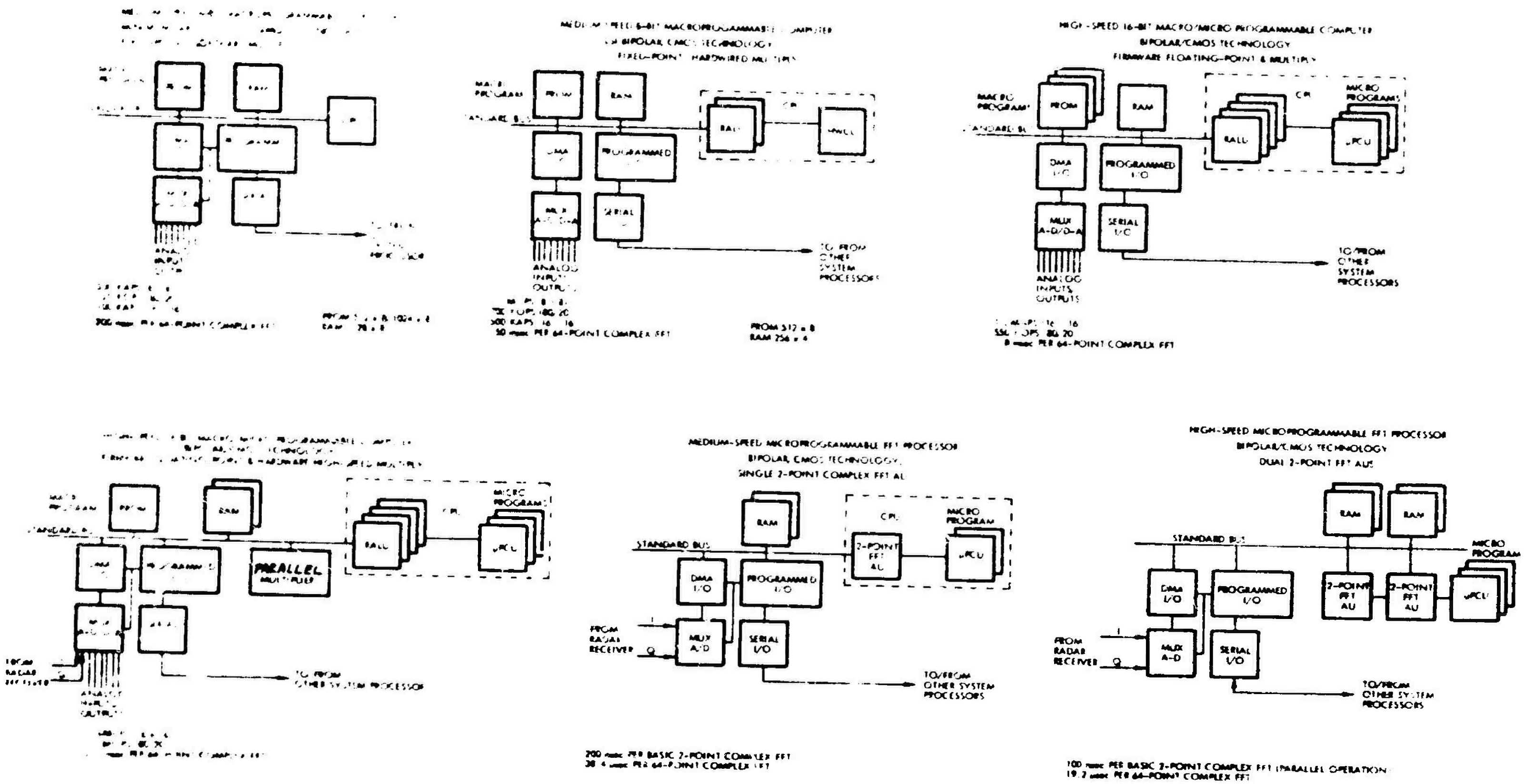
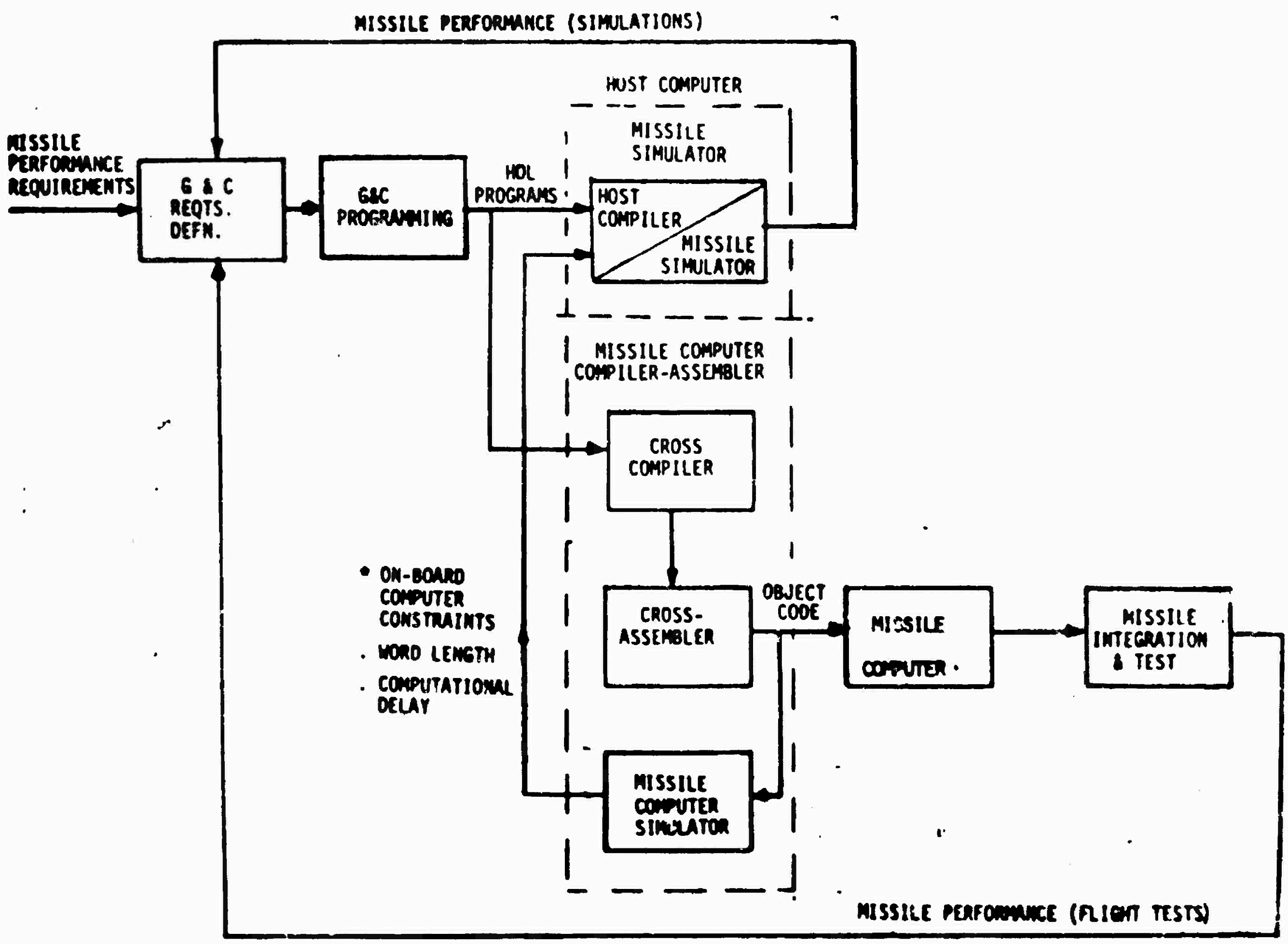


Figure 6 Macro-Modular Microcomputer Family for On-Board Missile Signal Processing and Control



NOTES: * USED FOR 2ND PASS THROUGH MISSILE SIMULATIONS.

Figure 7 Unified Guidance and Control System Software Development Process for Digital Missiles.

3. MODULAR DIGITAL GUIDANCE AND CONTROL

With the rapid improvements in digital technology the concept of missile-borne digital processing and control to achieve improved performance and flexibility in design and growth evolution becomes very attractive. Small, modular digital computers having considerable computing power can perform the variety of functions required of tactical air-to-air missiles more efficiently and effectively than conventional analog technology. Figure 8 shows the type of functions which can be performed digitally as determined by the Phase I and Phase II study programs.

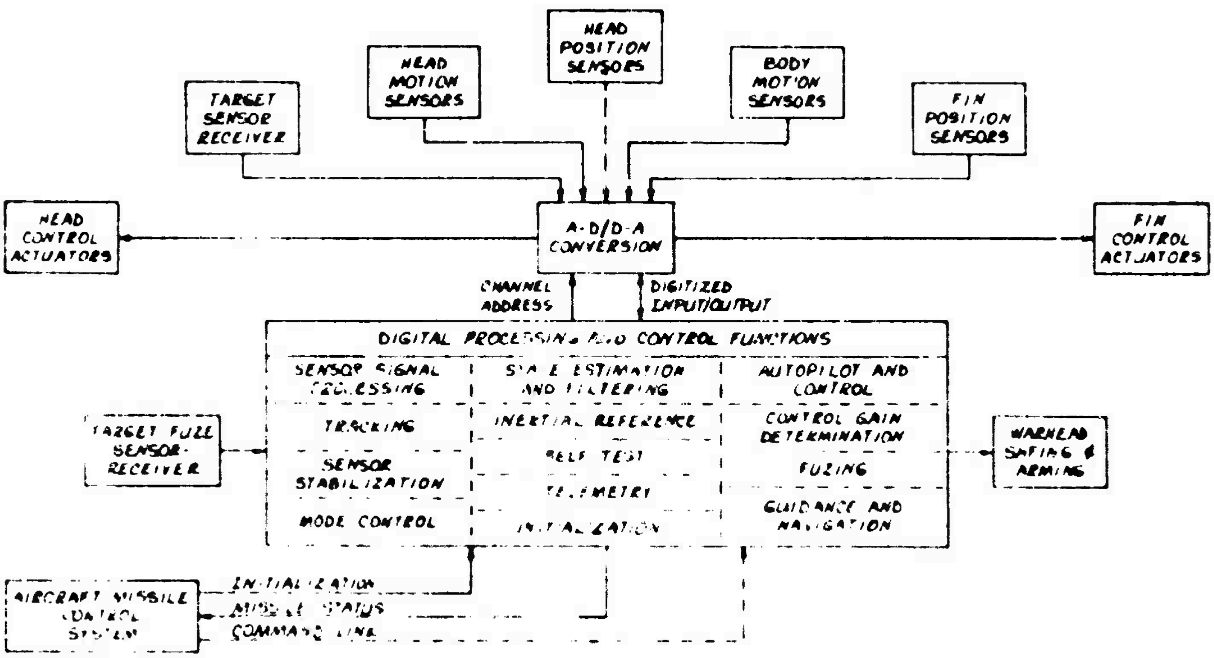


Figure 8 Missile-Borne Digital Processing and Control Functions and System Interfaces

Digital techniques offer flexiblity through the use of on-line software control that can be more rapidly altered to accomodate different missile configurations, sensors, control mechanisms and mission requirements. Arithmetic and logic ability and memory provide a capability to perform certain functions more accurately. Digital techniques can also accomplish other functions such as time variable estimation and guidance, adaptive autopilot control, digital signal processing and electronic counter-counter measures (ECCM) logic which cannot be performed easily or at all with analog techniques.

In addition to improved performance and greater functional capability, the digital implementation of missile guidance and control systems offers the means of providing a flexible, modular approach to missile system design. A single, easily expandable minicomputer or a federation of several microcomputers of standard design can replace much of the present distributed hardware which comprises the missile guidance and control system. As Figure 8 illustrates, a digital computer system can replace seeker head control electronics, processing of baseband signals from target sensors, guidance electronics, fuzing and autopilot control circuitry.

A standardized interface between missile subassemblies/ sections and avionics of the carrying aircraft is a natural characteristic of digital implementations.

Exploitation of the inherent flexibility of digital processing and control leads to the concept of modularity and its practical application to the continual process of technology transfusion into operational missiles and hence the use of a standardized, modular computer family for all classes of air-to-air missiles.

In addition to the inherent advantages of digital processing cited above, multipurpose, modular digital processing in tactical missiles should prove more cost-effective than present hardwired configurations.

The above features of digital missile guidance and control are discussed at greater length in the following paragraphs, with the remaining subsections devoted to: an overview of the major functions, the classification of air-to-air missile types and, lastly, digital system design considerations which are fundamental to both single and federated/distributed computer system implementations.

3.1 Digital vs Analog Systems

In view of the parallel improvements evidenced in both analog and digital circuit technology, the question is raised as to the advantages of digital versus analog system implementations in missile guidance and control.

The following paragraphs summarize the flexibility, growth, interface, performance reliability and cost features of digital versus analog mechanizations for the guidance and control functions analyzed in the Phase I and Phase II studies. It can be seen that, where practicable, (based on the results of function partitioning trade-off studies), programmable digital processing and control meets the overall system design goals more effectively than alternative analog techniques.

3.1.1 System Flexibility and Growth

Design Changes - A modular digital approach accommodates design changes throughout all phases of the weapons system life-cycle. These changes can arise from both redesign during the RDT & E phase and evolutionary growth thereafter. Redesign may be needed to correct deficiencies identified during test, accomodate specification and/or interface changes or to improve producibility. Evolutionary growth provides capability to perform new missions and counter new threats.

Evolutionary Growth - Evolutionary growth also occurs as the fruits of new technologies become available for incorporation into existing missile systems. The availability of new sensors, and better signal processing and guidance techniques can improve the performance of existing weapons. Replacement of conical scan processing with monopulse processing can provide improved acquisition performance and electronic counter measures (ECM) invulnerability as does fast Fourier transform signal processing instead of sweeping velocity gate. Variable bandwidth filtering and optimal guidance provide a significant increase in missile performance against maneuvering targets. Both FFT processing and optimal guidance require digital processing techniques.

Digital techniques reduce the problems resulting from product growth as shown in Table 6. For example, when a new sensor is made available, the conventional design approach requires that not only the sensor be replaced but the processing circuits that support it also be replaced. Replacing the sensor processing circuits requires a complete hardware development cycle, with an equally severe impact on logistics.

Both the original sensor and processing circuits would be thrown away. New test procedures and new or modified test equipment would be required as well as added personnel training. The new analog modules would then have to be reintegrated with the existing analog modules, and each weapon configuration using the new modules would require modification to the interface equipment with the various carrying aircraft.

Using a modular digital approach offers a much simpler and more economical introduction of new technology. Only the sensors would have to be replaced and the computer program memory modules would be reprogrammed requiring only a software development cycle. In terms of logisitics, only the sensors are thrown away since the processing function resides in the new software. It is not likely that modification to the test equipment would be required nor would additional personnel training be required. There would be no impact on integration, either within the various modules or between weapon configuration and the carrier aircraft.

TABLE 6

SYSTEM GROWTH IMPACT

ANALOG VS DIGITAL IMPLEMENTATIONS

CONVENTIONAL DESIGN	MODULAR DIGITAL APPROACH
o REPLACE SENSORS	o REPLACE SENSORS
o REPLACE PROCESSING	o REPROGRAM
o Develop new algorithms	o Develop new algorithms
o Design new circuits	o Design new program modules
o Fabricate new hardware	o Reprogram digital processor
o LOGISTICS	o LOGISTICS
o Throw away sensors and processing	o Throw away sensors
o Design new test procedures	o Update test procedures
o Modify test equipment	
o Train personnel	
o INTEGRATION	o INTEGRATION
o Modify module to module integration	o No impact
o Modify weapon/aircraft integration	

In summary, the flexibility needed to accomodate the above changes is best achieved in a digital implementation using a standardized macro-modular building block approach to digital computer design. Modular, application-oriented, software subroutines would be developed concurrently with the hardware.

3.1.2 System Interfaces

Modular weapons can be launched from significantly different launchers and launch vehicles and contain a wide range of functional options. Analog implementations, while providing modifiable system and matching the weapon to the mission, yield problems of almost unmanageable proportions in integration, training, maintenance and logistics.

Current launch aircraft umbilicals for example, contain discrete analog and digital interfaces with dedicated cockpit controls and displays. Analog integration is inflexible and even if accomplished does not remove the interface burden from future advances in technology.

Analog integration between modules is similarly inflexible and imposes severe constraints when minimizing cost and complexity. Each module must contain additional hardware to satisfy a common interface. The additional hardware adds to the cost of the modules. The management and control of the common interfaces adds to the complexity of the system design.

As technological advances and performance improvements become possible, inflexible common interfaces delay the early use of these advances. Cost tradeoffs have always resulted in making-do with already acquired hardware because of the cost and complexity of adapting to rigid interfaces.

Modular digital techniques can solve the problems outlined above and provide the desired flexibility at a reasonable cost. A single family of digital macrofunction modules will support the requirements of short, medium and long-range air-to-air weapons as well as surface-to-air and air-to-surface missions.

3.1.3 System Performance

Digital processing provides inherent performance capabilities not available in analog circuitry as described in the following paragraphs.

Memory - Modular digital memories of 256/512/1024/2048 8-bit words per large-scale integrated circuit package provide an accurate means of storing programs, real-time data and constants e.g., radome compensation data, missile aerodynamic data, and calibration data. These data may be applicable to all missiles of a given type e.g., aerodynamic data, or may reflect individual component characteristics such as the g-sensitive drift coefficient for each gyro on a missile.

Arithmetic - While addition and subtraction have straightforward analog implementations, multiplication and division require complex circuits which are subject to drift and inaccuracy and are generally avoided if possible. Digital processors have inherent high-speed add/subtract/multiply/divide capabilities which provides the means of generating complex functions such as the trigonometric functions used in coordinate transformations. The arithmetic capability can also be used to generate recursive equation solutions thus providing a digital filtering and estimation capability as well as fast Fourier transform capability.

<u>Accuracy and Dynamic Range</u> - The dynamic range of analog components (e.g., operational amplifiers) is generally limited to 10 in a laboratory environment. Under military environmental extremes this can degrade to 10 or less. A digital processor can provide greater precision and dynamic range through the use of either a large computer word size or a shorter word length and double-precision arithmetic. A 16-bit machine, for example, has a 3x10 dynamic range for single-precision operations and 2x10 for double-precision operations. Calculations in a digital processor are drift free and immune from the usual analog noise. As a result, high accuracy can be achieved in the long term integration operations used in inertial reference functions. This large dynamic range is also needed for range calculations and for the nonlinear matrix equations used in filter calculations. Dynamic range can be made virtually unlimited by the addition of floating-point arithmetic to the processor.

<u>Logic</u> - Logical operations are inherent in a digital processor. In typical analog missiles, arithmetic logic and switching functions are distributed among several circuits, each performing an individual function. A digital processor time-shares one arithmetic and logical circuit/unit under program control.

Enhanced functional capability can be obtained through digital signal processing and control techniques not available generally via other implementations. The specific enhancements in performance obtainable with digital techniques are discussed for each major system function in subsection 3.2.1 through 3.2.8.

3.1.4 System Reliability and Test

The digital implementation of missile functions provides improved reliability compared to analog systems. Digital devices which use saturated logic have inherently higher reliability and noise immunity with less temperature and vibration sensitivity than analog devices. The execution of guidance and control functions in digital processors, using high-density, semiconductor circuits, achieves a significant reduction in parts counts and circuit connections which in turn provides a more reliable system compared to multiple, single-function, analog circuit implementations.

The testing of digital computer systems is simplified by the use of built-in-test routines in each computer memory. Computer and subsystem tests can be performed with fault isolation to both line and shop-replaceable-unit levels.

3.1.5 System Cost

The use of multipurpose digital processors in missile systems should result in lower development, test, production and life cycle cost.

Development Cost - During the development cycle numerous design changes occur. The cost is minimized if the changes are in modular software as opposed to analog hardware. Further savings accrue through the use of a unified common software operating system for the entire modular computer family. Development cost of the processor hardware is reduced by building- up functional capability from a family of macrofunction computer modules.

Production Cost - Reduced production cost can be anticipated from the use of a family of macro-modular digital processors for missiles. Savings accrue from the use of common, large-scale integrated-circuit, macrofunction modules, (hybrid/monolithic), across a broad range of missiles thus taking advantage of the savings inherent in high-volume production. These modules would use mainly MIL-qualified versions of proven commercial semiconductor products which minimizes risk and ensures delivery through multiple procurement sources.

Life Cycle Cost - The higher reliability and inherent standardization of parts in digital missile implementations should result in lower maintenance costs. Similarly, a common software operating system reduces the cost of training operating and maintenance personnel. Both of these factors reduce life-cycle cost.

Further, the availability of a computer inside the missile provides ready access to all sensor and actuator points and greatly simplifies the missile test equipment. The cost of capital equipment and testing of the missile in the factory and in the field will therefore be reduced. The number of physical nominals which need to be adjusted will also be minimal, since many of them have been absorbed into the computer memory.

3.2 Major System Functions/Subsystems

The functions and interfaces which comprise a tactical air-to-air missile guidance and control system are depicted in Figure 5. These functions range from sensor signal processing, sensor tracking and stabilization, filtering and estimating, to guidance and control, fuzing, launch initializations and mode control in the case of single computer systems. Not all of these system functions are necessarily required in all missiles, nor is their degree of complexity or performance the same; however, all should be considered in the process of determining the feasibility and application of digital techniques, together with telemetry and test as supporting functions. A description of each function and the performance enhancement expected from the use of digital techniques is given in the following subsections.

3.2.1 Target Sensors

The target sensor can be optical (television-TV), infrared (IR) or radar (CW or pulse doppler; active, semi-active or passive, in the case of ECM). The purpose of the sensor is to search for, acquire and track the target. In a radar system, for example,

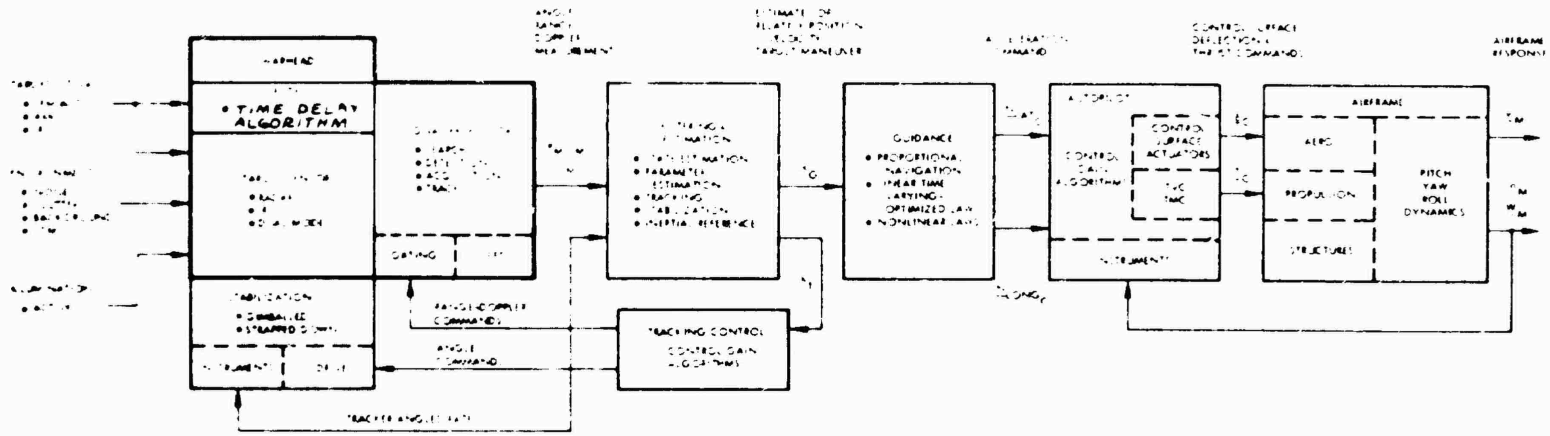


Figure 9 Missile System Functions

the signal processor extracts signals from the sensor to perform search in range, doppler and angle and provides detection through various algorithms such as constant false alarm rate (CFAR), dual threshold, and various "m out of n" return logic. Acquisition in range and doppler may be performed using digital FFT techniques coupled with target selection logic that provides superior performance over analog techniques. Selection of quiet targets in clutter, in the presence of standoff jammers, or with cloud background, and flares in the IR case, and resolution of multiple targets may be considerably enhanced using a digital approach. In the radar case, after target acquisition, tracking is performed in conjunction with filtering and estimation of angle range and doppler, to close these loops. Digital filtering and prediction offer the means to perform range and doppler tracking that can be more precise and more resistant to pull off type deceptive jammers. Extraction of error angle information can be accommodated for IR reticle scan systems involving AM or FM encoding of angle information or IR two-dimensional arrays, or for radar sensors using conical scanning or monopulse techniques.

The data that the sensor provides to the guidance system can consist of boresight error ($\underline{\varepsilon}_M$), range error ($\Delta R_{MT}$) and range rate error ($\Delta \dot{R}_{MT}$) along the line of sight to the target, signal to noise ratio, (SNR), inertial rates ($\underline{\omega}_{SM}$) in the seeker frame and gimbal angles ($\underline{\theta}_{SM}$) relating the seeker frame to missile body coordinates. In some systems this data is restricted to measurements of $\underline{\varepsilon}_M$, $\underline{\omega}_{SM}$, and $\underline{\theta}_{SM}$.

Digital signal processing provides the ability to adapt to varying target signatures, environments and ECM. The logic and processing during search, detection, acquisition and tracking can all be improved with digital technqiues. Digitally implemented logic can provide close to optimum utilization of the missile ECCM features. The overall probability of acquisition can be optimized by varying the search parameters (false alarm rate, threshold), over the search sector according to the probability of target signal presence. CFAR levels can be set digitally for maximizing detection.

FFT processing can provide a wide doppler band display of target and clutter signals. Digital logic can then be used to find and isolate the main clutter signals so that the target signal can be seen within the dynamic range of the analog to digital convertor and the FFT processor. FFT processing can provide a 6-10 db SNR advantage over sweeping velocity gates for the same detection probability.

Digitally implemented processing for IR homing missiles can enhance performance by providing more accurate biasing techniques for intercept forward of the target tail section.

3.2.2 Tracking and Stabilization

Tracking and stabilization of the tracking sensor in a homing missile is a critical control function where it is desired to obtain an accurate measure of the line-of-sight rate while isolating the measurements from body motion. Conventional gimbal servo systems provide this isolation to a certain extent, but in general this approach cannot easily compensate for gimbal cross-coupling effects and torque disturbances which are important in electric drive systems with limited torque output. Sensor gyros, required for stabilization, have error components in their outputs due to missile induced motions around their output axes. These effects can be reduced through the use of digital control. For example, improved isolation can be achieved by correcting for gyro output axis coupling, and by correcting for g sensitive drift using the sorted g sensitive drift coefficient for each seeker head gyro.

Obviously in the case of strapped down sensors, such as conformal body fixed arrays, a digital capability is essential to the stabilization function.

In conjunction with digital filtering and estimation, tracking loops, which are less sensitive to fluctuating receiver signal to noise ratio and mode switch due to ECM, can be implemented. The digital filter increases the track loop bandwidth for high SNR, reduces it for low SNR, and coasts the antenna during data dropouts by maintaining the seeker space rate equal to the current best estimate of LOS rate.

In radar systems, radome refraction slope places significant design constraints on the guidance problem and can cause stability, miss distance and missile maneuverability problems. Negative refraction slopes are destabilizing while positive slopes slow down the missile's response to line-of-sight errors. Of the compensation algorithms available to the digital system, the ability to reduce the impact of this disturbance is probably most important. A compensation algorithm is superior to present biasing techniques and particularly important for wide band RF applications. In addition, design problems encountered with dual mode guidance systems which may have conflicting requirements on dome material should be reduced.

3.2.3 Filtering and Estimation

Perhaps the greatest impact of the availability of an on-board digital computer is the vastly improved estimation capability it provides. With modest computational equipment, it is possible to implement relatively sophisticated estimation algorithms (e.g., Kalman filters), which would be virtually impossible by analog means. This estimation capability yields performance improvements in three ways:

1) More effective guidance and reduced miss distance, resulting from improved knowledge of the relative motion of missile and target;
2) Improved tracking tenacity, through the use of predicted angles, range and/or range rate for seeker pointing and range and/or doppler gate setting; and
3) Capability for estimating auxiliary parameters (e.g., stability derivatives and target properties, which aid in trajectory estimation, autopilot gain setting, and any engagement decisions which may depend on target parameters or behavior.

It is clear, from previous guidance studies, that estimation of target motion (including acceleration) provides a considerable improvement in performance in the maneuvering target case. Target acceleration estimation may thus be considered an essential part of any high-performance homing guidance system.

Filtering and estimation of system states through the use of digital discrete recursive estimators provides information allowing improved guidance laws and autopilot control. The estimator can be as complex as a fully-coupled, 9-state, Kalman filter where estimates of target acceleration $\hat{\underline{a}}_T$ relative position $\hat{R}$, and relative velocity $\hat{\dot{R}}$ can be obtained, or where the computational burden is too severe, or the accuracy of a fully-coupled filter is unnecessary, simpler configurations can be used to provide suboptimal estimates. Further simplification could reduce this function to fixed gain noise filters on boresight error alone.

Application of digital discrete recursive estimators improves angle, range, and doppler track especially in cases where pulsed illumination is used and in cases where blinking jammers cause constant mode switching.

3.2.4 Guidance

Guidance accuracy can be improved over basic laws, such as proportional navigation, by explicitly compensating for target maneuvers using estimator outputs, and compensating for missile autopilot dynamic lag. Digital memory and arithmetic capabilities allow the necessary computation to be performed in a real time mode; whereas, analog technology is generally limited to simpler laws such as proportional navigation. Augmented proportional navigation (APN) compensates for target acceleration and a four-state law (4SL) provides, in addition, compensation for autopilot lags.

For simple missile systems as represented by Class I, traditional proportional navigation (PN) guidance is generally used because of limitations on the available measurement data and computational capacity. For more sophisticated systems, and the associated higher levels of required performance, experience has shown that two important requirements are:

1) Estimation of target acceleration and its use in the guidance law.
2) Some method of compensating for the dominant lag of the autopilot.

When these factors are included in the derivation of the guidance law, considerable improvements in performance are realized. Figure 10 compares performance (RMS miss versus missile g limit) for a typical intercept using three different guidance laws:

1) Proportional Navigation (PN)
2) Augmented Proportional Navigation (APN) which utilizes the estimate of target acceleration n_T
3) Four-State Law (4SL), which adds compensation for autopilot lag

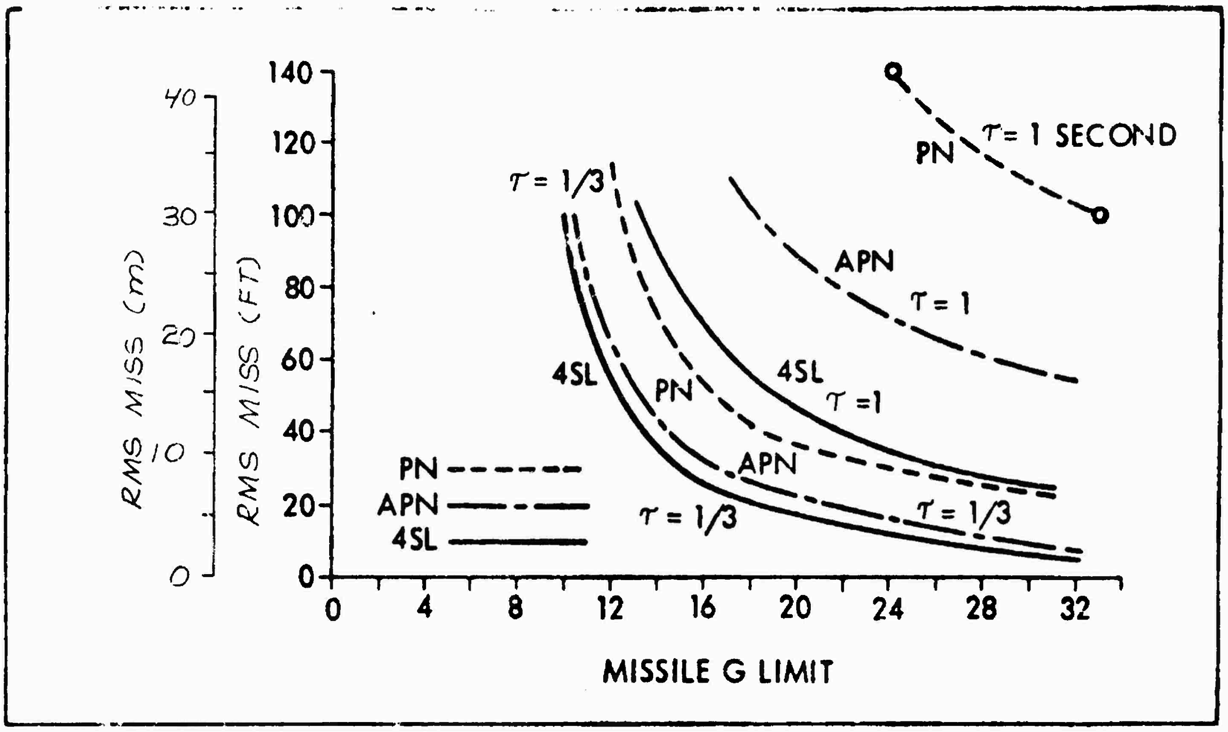


Figure 10 Comparison of Guidance Laws

The case chosen for the comparison is an 8 second flight during which the target initiates a 4g maneuver at random times. Initial heading error is 5 deg and the three components of angular measurement error are assumed to be: glint 5 ft, receiver noise 6 mrad and range-independent noise 1 mrad, (all values rms). Data rate is 10 Hz. The controller is assisted by a three-state Kalman filter estimating y_d (differential position), $\dot{y}_d$ and n_T. The improvement in performance is considerable when the guidance law is compensated for target maneuver and/or autopilot lag.

3.2.5 Autopilot/Control

Control and stabilization of the airframe is a critical function for all missiles. Rapid response to acceleration commands in two axes, while maintaining pitch/yaw damping and roll control is required for satisfactory guidance. The problems associated with airframe control are wide variations in flight conditions (altitude and velocity) and resultant aerodynamic coefficient variation, stability under transient conditions, dynamic range limits and nonlinearities in the actuation system.

The role of the digital computer for missile control goes far beyond digitizing an analog autopilot. With the computational power for adjusting the control, the response of the controlled airframe can be made relatively independent of flight condition and the closed loop poles placed within specified regions. Since autopilot design is constrained by the range of aerodynamic characteristics, the digital design should consider from the outset the estimation, measurement, or storage of aerodynamic parameters.

A more general approach to the control problem is to use sensed performance of the missile to continually change the controller gains during flight. These implicit or explicit adaptive designs do not have simple analog formulations and the digital computer becomes important. Implicit techniques do not estimate the unknown parameters of the missile but generate the control signal directly from sensor information. Explicit techniques generally seek to estimate the unknown parameters and, using this information, treat the control problem as if the aero parameters were known.

Implicit adaptivity is attractive for simple control systems and may have application for those missiles designed for specific missions. Multimode and multimission weapons, however, are more complex and explicit adaptivity is more likely desired. In all cases, however, knowledge of aero parameters is important.

3.2.6 Fuzing

Better fuzing techniques can be digitally implemented using end-game estimates of time-to-go to intercept, missile attitude, expected miss distance, and relative velocity which can be developed in digital filters as inputs to multivariable fuzing time delay algorithms for optical, active and semi-active fuzes. The algorithms can be optimized for different warheads including blast, fragment, and rod types. Aiming instructions for aimable warheads can also be computed.

3.2.7 Logic and Mode Control

In every missile there are a large number of logical decisions which must be made during the course of a flight. Launch logic, target selection, autopilot band switching and fuzing logic are just a few of the logical decisions which must be made in the missile. The hardware logic which makes these various decisions is scattered throughout a typical missile and involves a very substantial fraction of the total electronics and hence, the overall logic diagram of a missile tends to be a rather complex logic tree. In a digital missile system, logic functions can be implemented in a computer by means of decision tables and other extremely simple logic routines for which the computer is ideally suited. Also, the information upon which the decision must be based is available in the computer memory for other purposes. Thus, one of the many effective uses of a computer in a digital missile is the execution of the decision logic.

Mode control applies to the time-multiplexing of digital missile functions in a single computer system. The initiation of a given mode is determined by the master executive program, which performs the decision logic based on the occurrence of external, real-time events and/or the results of guidance and control algorithms executed by the computer and reported to the executive program via the active mode supervisor. As such, mode control ensures the proper function mix at any point in the mission time-line, and the execution of functions in accordance with the data sampling and stability criteria of the system.

3.2.d Telemetry and Test

Telemetry - Telemetry is used primarily in flight-test vehicles to monitor the performance of an all-up system in a true operating environment. In a digital missile, most of the telemetry information is already stored in computer memory and the remaining data can be readily acquired (e.g., airframe stress data, hydraulic pressures, battery voltages). A computer is therefore capable of controlling the entire telemetry function either on a time-shared basis in a single computer system, or continuously through a dedicated processor in a federated/distributed computer system. There are substantial advantages to this method of telemetering over analog systems. First, a separate multiplexer for telemetering is not required. Second, all of the buffer amplifiers and circuitry required to scale signals to proper telemetering levels have been eliminated. Also, the very large number of wires which are required to collect the telemetering information have been eliminated, thus eliminating many sources of pickup and a very large cost reduction because of the reduction of cabling. Interconnection wires always present a major problem in any missile design, and the elimination of wires is extremely important and cannot be over-emphasized. Scaling of the signals can be done in the computer if necessary to preserve telemetering dynamic range. And finally, digital telemetering can provide a more accurate picture of the missile parameters than is usually available with analog telemetering systems.

Test - A digital computer allows considerable pre-flight testing to be performed in a tactically configured missile. Extensive testing of the computer and the individual guidance and control functions and subsystems insures that a test vehicle is fully operational immediately before launch, with the option of providing the same test procedures in an all-up tactical missile to ensure reliability. Digital missiles could also be ground-tested in the same way, by applying primary power but without squibbing batteries, prior to installation on an aircraft.

Test therefore applies to the self-contained testing and maintenance of a missile at Factory, Overhaul and Repair shop at shore depot, Carrier Electronic Workshop, Hangar Deck and Flight Deck maintenance levels.

3.3 Generic Classification

The application of modular digital computers to air-to-air missile systems involves the consideration of a wide range of missions, missile characteristics and engagement environments. Depending on the missile involved, the airframe, guidance modes, control configuration, seeker and available instrumentation vary from relatively simple specific mission designs to highly sophisticated, multimode/multimission applications. In order to determine the feasibility and application of modular digital computers to perform desired missile functions, the configuration and requirements of the Sidewinder, Sparrow and Phoenix families of missiles have been studied and a generic classification of requirements has been established. The classification includes presently operational systems as well as anticipated future systems and was developed from a survey of mission requirements and functional configurations. The definition of the air-to-air missile missions included:

1) Launch envelopes and launch conditions.
2) Guidance modes and the avionics interface.

3) Target types and parameters.

4) Missile sensors and instrumentation.

5) Missile physical properties.

The definition of generic configurations involved a survey of:

1) Functional modes and phase of flight

2) Target sensors including IR, radar and dual mode.

3) Radomes including ranges of IR and radar boresight error slopes.

4) Receiver configuration encompassing conical scan and monopulse radar and reticle/array IR systems.

5) Gimbal drives and configuration including electric versus hydraulic, two or three axes or strapdown.

6) Airframe configurations encompassing wing, tail or canard aerodynamic control or thrust vector control.

7) Aerodynamic characteristics such as linear stability derivative values over the range of flight conditions.

8) Propulsion systems involving single or multiple stages.

9) Instrumentation and tachometers.

3.3.1 Class Definitions

As a result of this study, three generic missile families have been established and, relative to this classification, on board computational requirements can be defined for each class.

The generic families consist of a low cost, specific mission design Dynamics The first level of classification involves a description of guidance mode, interface with the launch aircraft, available instrumentation and control requirements as described by Table 7.

From this classification, the functions which must be provided can be defined and the relative complexity of each function can be assessed. In Class I for example, only one guidance mode is required and the signal processing function need only be concerned with continually processing data from one tracking sensor. Multiple sensors exist in Class III systems, however, and for in-flight hand-over capability, the data from each sensor must be processed simultaneously. The same increasing level of complexity exists for all functions as one progresses from Class I to Class III.

TABLE 7

GENERIC MISSILE FAMILY DEFINITION

FAMILY PARAMETER	MISSILE CLASS I: LOW COST SPECIFIC MISSION DESIGN	MISSILE CLASS II: HIGH PERFORMANCE LIMITED MODE DESIGN	MISSILE CLASS III: HIGH PERFORMANCE MULTI-MISSION MULTI-MODE DESIGN
Envelope			
Range (nmi)	10	30	75
Altitude (Kft/Km)	30/9.1	70/21.3	90/27.4
Guidance Mode	Single mode (no inertial mode)	Single or dual mode (no inertial mode)	Multi-mode (in-flight handover capability)
Avionics Interface	Minimum-guidance lock before launch, no missile parameter initialization	Limited missile parameter initialization alt, V_{MO}, control initialization	Unlimited, based on missile perf. reqts.
Missile Instrumentation	Seeker instrumentation only (simple autopilot)	Seeker instrumentation (2 rate gyros or equiv) Body instruments 3 rate gyros 3 accelerometers	Full Seeker and body instrumentation (strapped-down inertial reference system)
Target Sensor			
Optical:	IR Reticle	IR-Reticle or Array	IR-Array
Radar:	SA-CW	SA-CW, SA-PD, ARM or combinations of two.	Class II plus a A-PD or combinations of 2 or 3.

3.3.2 Parameters

Further quantification of the three generic classes is provided by specifying the parameters and their ranges that apply to each class. Table 8 lists these data for each Class and the various missile elements.

3.3.3 Environmental Inputs

In addition to missile and target properties, natural and man-made environmental inputs are bracketed for each generic class in Table 9.

With these definitions and parameters established for each class, the functional analysis, design approach and computer requirements are developed in the following sections of this report.

TABLE 8

SYSTEM PARAMETERS

PARAMETER	MISSILE CLASS I	II	III
Target Characteristics			
Altitude Range (kft/km)	SL-30/9.1	SL-70/21.3	SL-90/27.4
Speed Range (MACH)	0.5-2	0.5-3.0	0.5-4
Maneuver Level (g)	2-4	3-6	5-8
Missile Launch Data			
Launch Altitude (kft/km)	SL-30/9.1	SL-50/15.2	SL-70/21.3
Speed Range (MACH)	0.5-3.0	1.0-4.5	1.5-6.
Launch Range			
Max (nmi)	10	30	75
Min (ft/m)	1000/304.8	3000/914.4	>5000/1524
Physical Characteristics			
Launch Weight (lb/kg)	100-250	250-500	500-1000
Length (in/cm)	<120/305	<150/381	<180/457
Diameter (in/cm)	5-7/12.7-17.8	8-11/20.3-28	13-16/33-40.6
Configuration	Canard/Wing	Wing/Tail	Tail/ Tail-TVC
Dynamics			
Aero			
Nat. Freq (rps)	6-30	6-20	6-20
Max. Alpha (deg)	20	25	30

TABLE 8 (Continued)

SYSTEM PARAMETERS

PARAMETER	MISSILE CLASS I	II	III
Structural			
Bending Mode (rps)			
First	200-600	150-200	200-500
Second	600-1800	600-800	600-1800
Kinematic			
Angular Rates (deg/sec)			
Pitch/Yaw	≤150	60-100	60-150
Roll	≤500	300-450	300-450
Acceleration			
Angular (deg/sec^2)			
Pitch/Yaw	<2000	<6000	<4000
Roll	<10,000	<20,000	<20,000
Translational (g)	<30	<40	<60
Inertial Instrumentation			
Missile Body	None-1 Roll gyro	3 Rate gyros	3 Rate integrating gyros (RIG)
		2 or 3 accelerometers	3 Acc.
Seeker	Free gyro (Spin Stab.)	2-3 Rate gyros	2-3 RIG

TABLE 8 (Continued)

SYSTEM PARAMETERS

PARAMETER	MISSILE CLASS I	II	III
Target Sensor			
Control			
Track Bandwidth (rps)	10-100	10-50	10-50
Stabilization Bandwidth (rps)	<1000	100-200	100-200
Type	Single Mode	Single or dual mode	Singe dual or multimode
	o IR	o IR	o SAR
	o SAR	o SAR	o SAR/AR
		o SAR/AR	o SAR/ARM/AR
		o SAR/ARM	o SAR/ARM/IR
Radar Parameters			
Antenna Diam. (in/cm)	3-5/7.6-12.7	6-9/15.2-22.8	11-14/27.9-35.5
Beamwidths (deg)	27-17	14-9	7-6 X Band
			5-4 K_u Band
Rqd Acq. Range (nmi)	10	30	20-50
Avionics Designation Accuracy			
Angle (deg)	8	6	4
Range (ft/m)	±1100/335.2	±2300/701.0	±5000/1524.0
Velocity (fps/mps)	±500/152.4	±500/152.4	±500/152.4

TABLE 9

ENVIRONMENTAL INPUTS

INPUT	MISSILE CLASS I	II	III
Clutter (m^2/m^2, db)			
Land	-10/-30	-10/-30	-10/-30
Sea	-20/-40	-20/-40	-20/-40
Rain (mm/hr)	1.0	4.0	4.0
ECM			
Barrage	Yes	Yes	Yes
Spot	No	Yes	Yes
Blinking	No	Yes	Yes
Deceptive	No	No	Yes
IRCM			
Flares (IR)	Yes	Yes	Yes

3.4 Digital System Design

The design of a digital guidance system for a missile involves the translation of functional requirements into algorithms and parameters and the specification of interfaces and timing. This process was discussed in the Phase I Final Report, (Reference R.1), fo the relatively low frequency guidance and control functions addressed. A modular design approach was adopted in which the benefits of the digital approach, i.e., time variable estimations and guidance, and adaptive autopilot control, are retained, while offering a practical design procedure which results in improved performance, relative insensitivity to design model errors and a reasonable real-time computer load. The same approach is applied in this report. The signal processing, fuzing, mode control and telemetry /test functions are defined, algorithms developed and interfacing requirements (data rate, delay, etc.), set and then integrated into a total digital missile guidance system design. The impact of these added functions on system design and digital processing are discussed in this subsection.

3.4.1 Guidance System Design

The general missile guidance system model shown in Figure 11, is an expansion of the Phase I model to include the track signal processing function. This function is highly interactive with estimation, guidance and autopilot control functions in establishing the performance of the guidance system. The computational delay associated with digital extraction of target

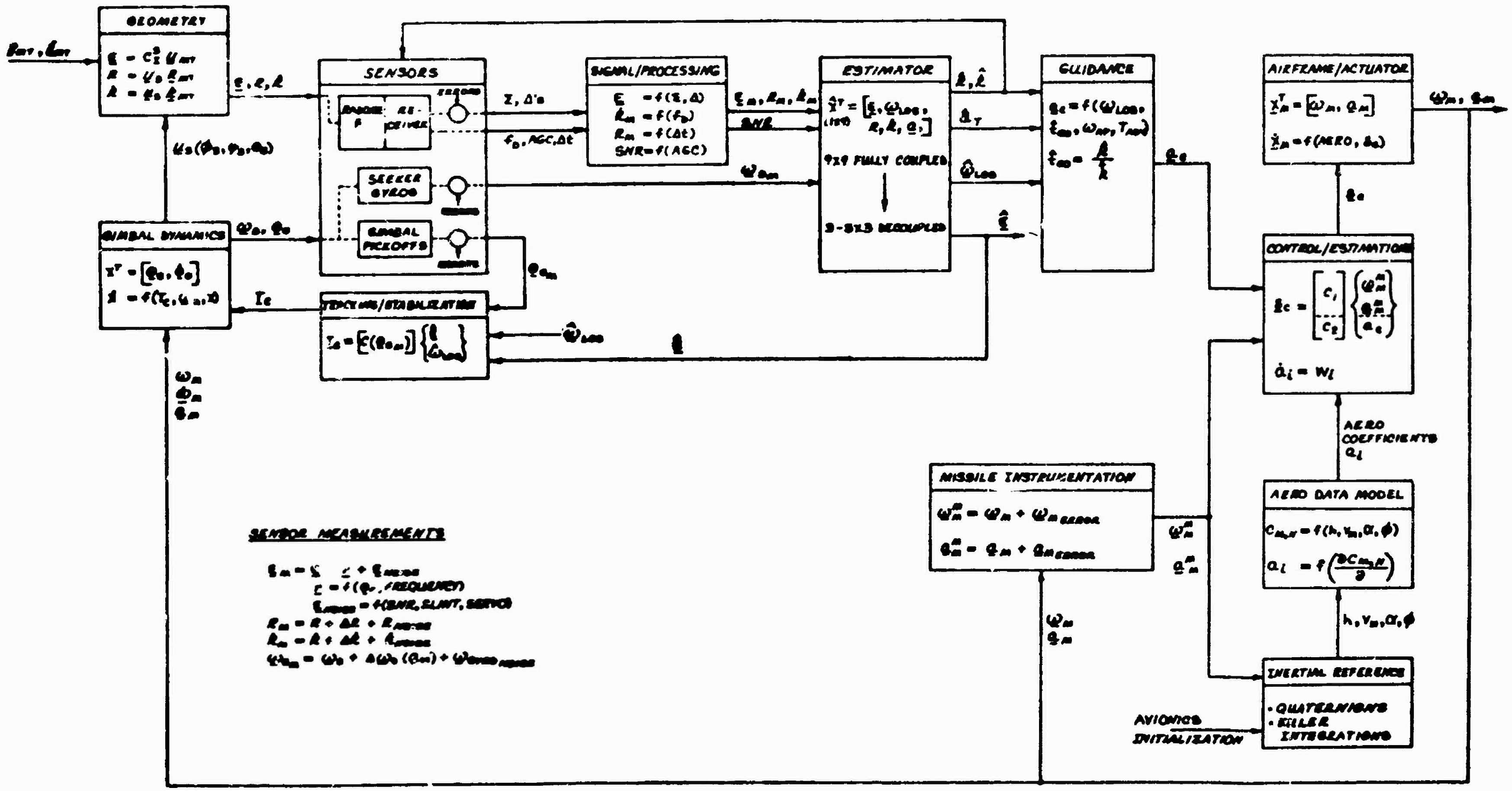


Figure 11 General Missile Guidance System

tracking data directly affects guidance miss distance performance. Bounds on this delay must be set in the context of the total model. On the other hand, target acquisition signal processing, fuzing, and mode control, although having their own requirements on data rate and delay, do not directly interact with missile guidance and control.

The track signal processing function involves the extraction of seeker boresight error, $\underline{\epsilon}_M$, relative range and doppler data and a measure of target tracking signal to noise ratio from the sampled sum and delta channel outputs of the receiver. Cascading this function with the other functional elements adds dynamics (tracking cell bandwidth BW_{sp}) and computational delay, $\underline{\tau}_{sp}$. In order to establish the impact of signal processing delay or guidance, a model of the guidance system which explicitly identifies where sampling and delay occur is required for setting digital processing requirements and assessing performance.

3.4.2 Digital Design Considerations

Along with the benefits of digital processing and control comes the requirement to convert back and forth from the analog to the digital domain. This conversion involves: sampling, quantization, and computational delay. In designing a digital guidance system these processes must be accounted for and requirements placed on the allowable values. The model for analyzing these effects is given in Figure 12. The functions remain the same as those of Figure 11, but the input sampling (analog to digital conversion) output sampling and holding (digital to analog conversion) and computational delay paths are shown explicitly. Quantization also occurs upon conversion.

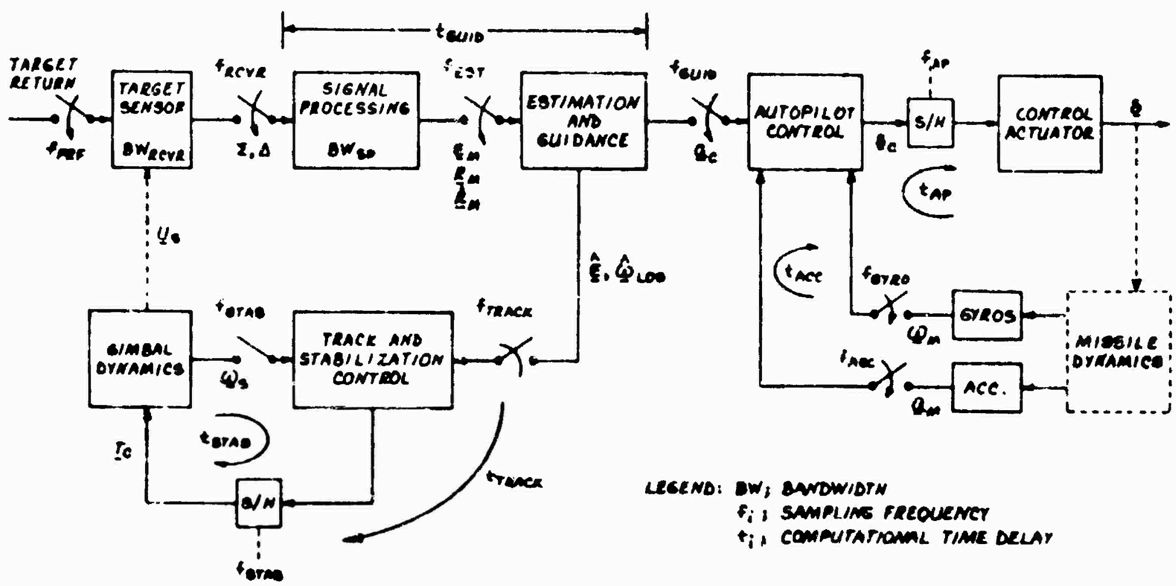


Figure 12 Digital Guidance System

In the Phase I Study, requirements were established on certain conversion parameters, namely,

1) Guidance data rate ($f_{GUID}=f_{EST}=f_{TRK}$) ranging from 5 to 25 Hz depending upon missile class.

2) Autopilot data rate (f_{AP}) ranging from 500 Hz for proper digital structural filtering to 10 Hz for control gain, determinations.

3) Seeker stabilization loop sampling, f_{STAB} at 500 Hz.

To complete the requirements on the digital process, additional conversion parameters must be specified. These include:

1) Autopilot multiple sampling rates f_{ACC}, f_{GYRO}, and computational delay τ_{AP} not considered in the Phase I study.

2) Sensor output sampling, f_{RCVR} and signal processing delay τ_{sp} and the overall delay τ_{GUID}, in generating autopilot acceleration commands from sensor outputs.

The timing relationships which exist among the various elements is shown in Figure 13.

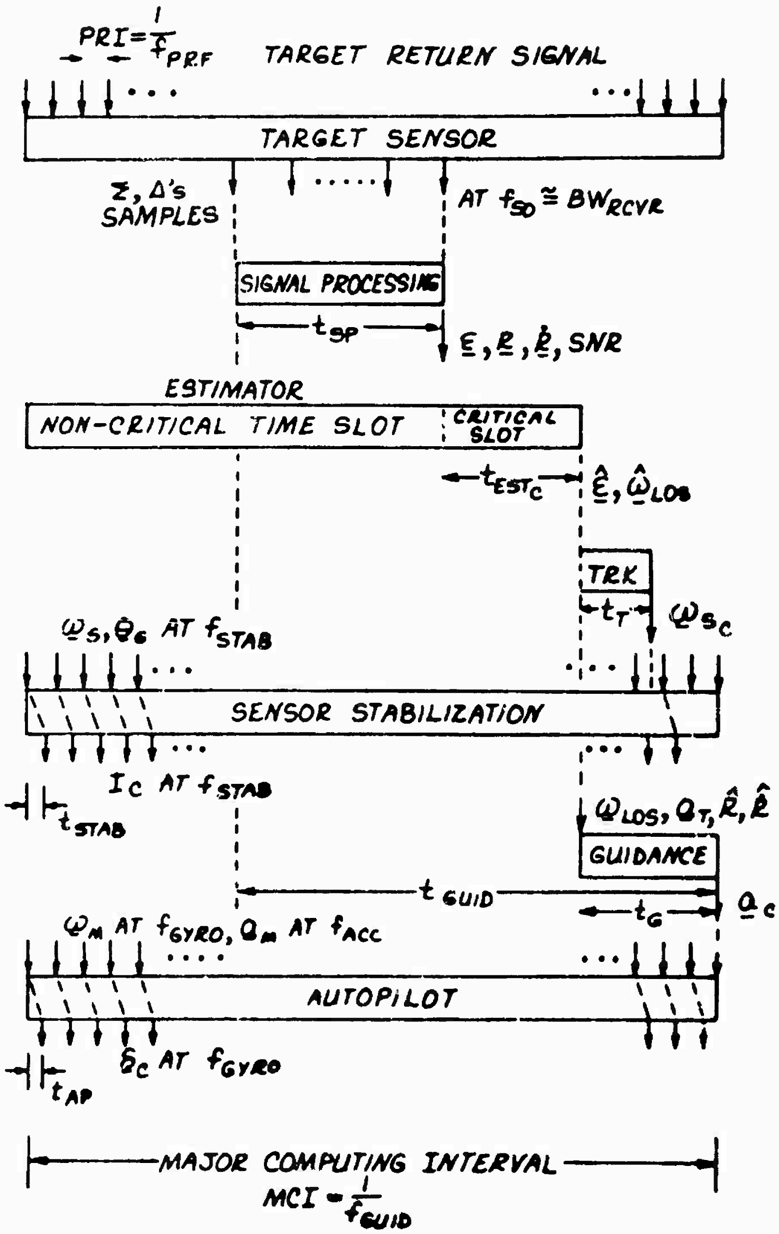


Figure 13 Digital Guidance System Timing

The lowest sampling frequency is generally, guidance, f_{GUID}, so that a major computing interval (MCI) is established by $MCI = 1/f_{GUID}$. MCI ranges from 40 to 200 msec, 100 msec (f_{GUID}=10Hz) being typical. Within the MCI all functions are executed at least once; namely guidance, estimation, track, signal processing or many times; sensor stabilization, f_{STAB}, and autopilot control, f_{GYRO} and f_{ACC}. Information passes from target sensor to signal processing to estimation and guidance. The delay in the computing process is additive, that is the total delay from start of signal processing to update of the command, $\underline{\alpha}_C$ is $\tau_{GUID} = \tau_{sp} + \tau_{EST_C} + \tau_G$. The allowable delay τ_{GUID} is established by miss distance requirements. Section 5 determines the allowable values of τ_G and f_{GUID}. The critical part of the MCI τ_{GUID} is generation of $\underline{\alpha}_C$ from the samples of the receiver output used in generating outputs $\underline{\varepsilon}, \underline{R}$, and $\underline{\dot{R}}$ in an output signal processing bandwidth BW_{sp}. Some computations, such as estimator covariance propagation and gain calculations can be done in the non-critical time slot prior to updating the state estimates when the new $\underline{\varepsilon}, \underline{R}, \underline{\dot{R}}$ data are available.

For the sensor stabilization and autopilot functions sampling rates which are high compared to guidance are required so that sampling at f_{STAB} and f_{GYRO} may proceed independently and asynchronously from guidance. For these wide bandwidth loops, however, computational delay τ_{STAB} and τ_{AP} are important. These requirements are set in Section 5 on autopilot control.

3.4.3 Digital-Analog Conversion Process

At each point in the digital guidance system where conversion from one domain to the other takes place, the effects of sampling , quantization and delay must be evaluated and specifications set. In the Phase I report (Ref. R.1, Section 3.3.2, Digital Digital Controller Design, these effects and design guides were discussed. Figure 14 illustrates the conversion process.

The addition of the signal processing functions to the guidance model and the accounting for computational delays requires that additional conversion requirements be established.

1) Sensor output sampling rate and quantization
2) Computational delay in guidance, and autopilot and seeker stabilization.

Sensor output sampling rate is determined by the Nyquist sampling theorem, that is:

$$f_{RCVR} \geq BW_{RCVR}.$$

This requires that the sum (Σ) and difference (Δ) channels each be sampled at a rate greater than the receiver (roughing filter) bandwidth typical values are

$$f_{RCVR} \geq 10\text{-}20 \text{ KHz}$$

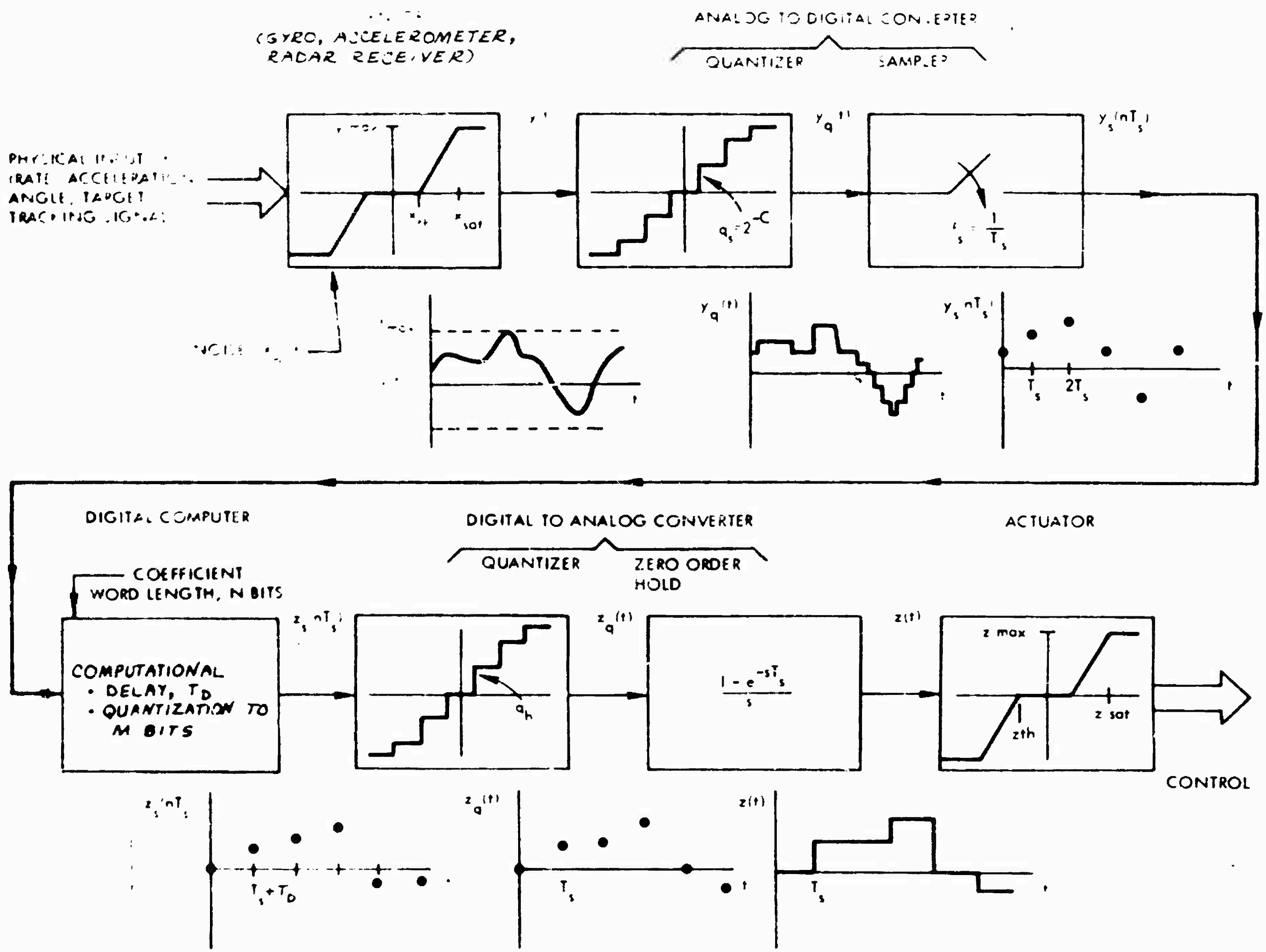


Figure 14 General Digital Processing Sequence

These are:

The number of binary bits required in the conversion is determined by:

$$C+1 \geq \frac{F(db)}{6} + 0.8$$

Where, C is number of magnitude bits

F desired signal to quantization noise ratio.

For typical radar sensors a 40 db ratio is acceptable so that an 8-bit converter on the Σ and Δ channels is usually sufficient.

For autopilot and seeker stabilization the effect of computational delay, τ_{AP} or τ_{STAB}, on loop stability is related through the relationship:

$$\Delta\phi = 2\pi f_{Critical}\left(\frac{\tau s}{2} + \tau\right)$$

Where, $\Delta\phi$ - phase shift due to sampling at $f = \frac{1}{\tau s}$ and delay τ. If loop phase shift is to be limited to say, $\Delta\phi = 0.2$ radians at the critical gain cross over frequency f_{CRIT} then a limit exists on $\frac{\tau s}{2} + \tau$. This is an approximate relationship. Through simulation, more exact requirements are set for autopilot and stabilization in Section 5.

4. DIGITAL MISSILE PROCESSING & CONTROL

This section contains a summary of the guidance and control functions analyzed in the Phase I study and the results of further analyses performed in Phase II covering the remaining on-board missile functions viz: radar, IR, ARM and multimode sensor systems; fuzing; mode control; telemetry and test. The object of these analyses being: to determine the functions suitable for digital implementation to achieve performance improvements and greater flexibility without a severe cost penalty; to define supporting digital algorithms and program modules; and lastly, to determine the computer loads for each digital function in terms of worst-case operation counts and instruction mixes, (i.e. percentage breakdown of add/subtract, multiply/divide and load/store/logical/branch operations), for the computer performance requirements and modular computer definition tasks, (see Section 6 and 7).

As in the Phase I Study, a simple 16-bit, fixed-point, single-register, minicomputer architecture and instruction set was assumed for sizing the program modules, since this type of machine, in addition to establishing worst-case operations counts, provides the logical point of departure for evaluating the merits of more sophisticated computer designs. Similarly, for consistency with Phase I computer load estimating procedure, instruction counts given in the computer requirements tables include only those operations executed in the worst-case/time-critical path through each program module. These instruction counts, together with those of the associated utility sub-routines, have been increased by 30% when converting to equivalent adds. Program memory requirements include the total number of instructions for any given program module also increased by 30%. The 30% increase constitutes an allowance for additional subroutine linkage and other overhead operations which are necessary to achieve a completely operational program. Equivalent adds are as defined in the Phase I study, (Ref. R1, Sect. 4, p. 4-1), i.e., multiply and divide operations are equivalent to 8 add/subtract/load/store instructions.

4.1 Guidance and Control Summary

The functions of: target seeker-head tracking and stabilization; filtering and estimation; guidance and autopilot, were analyzed in the Phase I study and described in Section 4 of the Phase I Final Report (Ref. R.1). Tables 10 through 14, extracted from the latter report, summarize the computer requirements for each of the above missile guidance and control functions to provide continuity with the current work. These computer loads are integrated with the remaining missile functions in Section 6 to establish the composite computer loads for each generic missile class.

TABLE 10

TRACKING AND STABILIZATION

COMPUTER REQUIREMENTS

MODULE	NAME	ADD/ SUB.	MULT/ DIVIDE	LOAD/ STORE	UTILITIES	*EQUIV. ADDS	MEMORY *PROGRAM	DATA RAM	ROM
51.	Basic Tracking & Stabilization	5	4	24	2-u13 2-u14	314	70	22	16
52.	Radome Compensation	2	4	34	1-ATAN 1-ACOS 3-u20 2-u19	762	89	10	343
53.	Head Aim	2	2	4	-	29	14	4	-
54.	Gyro Output Axis & Outer Axis Torque Compensation	10	16	34	2-sin 2-cos	489	99	14	-
55.	Linear 9-state Feedback Control	24	24	72	4-INT	5-7	177	12	-

LEGEND:

*Totals include 30% additional short operations for subroutine l kages and other miscellaneous overhead instructions.

TABLE II

FILTERING & ESTIMATION

COMPUTER REQUIREMENTS

MODULE	NAME	ADD/ SUB.	MULT./ DIVIDE	LOAD/ STORE	UTILITIES	*EQUIV. ADDS	MEMORY *PROGRAM	DATA RAM	DATA ROM
E1	FIXED GAINS								
	Roll Stabilization	9	11	28	1-u1	297			
					1-u2				
	State Propagation	25	23	48	-	334			
	SUBTOTAL	34	34	76	2	531	198	23	-
E2	SWITCHED GAINS								
	Roll Stabilization	9	11	28	1-u1	297			
					1-u2				
	Gain Determination	-	-	24	6-u17	187			
	State Propagation	25	23	48	-	334			
	SUBTOTAL	34	34	100	8	818	260	33	44
E3	DECOUPLED KALMAN								
	Roll Stabilization	9	11	28	1-u1	297			
					1-u2				
	Covariance Propagation	46	69	119	1-u8	780			
	Gain Determination	6	12	18	-	156			
	State Propagation	28	27	55	-	389			
	SUBTOTAL	89	119	220	3	1822	572	58	-
E4	COUPLED KALMAN								
	Roll Stabilization	9	11	28	1-u1	297			
					1-u2				
	Covariance Propagation	369	448	825	1-u7	6308			
					1-u8				
	Gain Determination	8	40	48	-	489			
	State Propagation	52	51	103	-	732			
	SUBTOTAL	438	550	1004	4	7826	2610	71	-

LEGEND

*Totals include 50% additional short operations for subroutine linkages and other miscellaneous overhead instructions.

TABLE 12

GUIDANCE

COMPUTER REQUIREMENTS

MODULE	NAME	ADD/ SUB	MULT/ DIVIDE	LOAD/ SOTRE	UTILITIES	EQUIV. ADDS	MEMORY *PROGRAM	MEMORY DATA
G1	PROPORTIONAL NAVIGATION (PN)	7	13	20	-	131	52	7
G2	FOUR STATE/RANGE-DESENSITIZED						354	40
	Option 1 (4SL)	37	54	105	1-u7 2-u8 1-u12	896		
	Option 2 (RDL)	47	74	127	1-u7 2-u8 1-u12	1146		

LEGEND:

*Totals include 30% additional short operations for subroutine linkages and other miscellaneous overhead instructions.

TABLE 13

AUTOPILOT

COMPUTER REQUIREMENTS

MODULE	NAME	ADD/ SUB.	MULT/ DIVIDE	LOAD/ SOTRE	UTILITIES	*EQUIV. ADDS	MEMORY *PROGRAM	DATA RAM	DATA ROM
A1.	Basic Autopilot	7	4	23	2-Int	183	86	16	-
A2.	Structural Filters And Fin Command Mixing	4	0	13	3-u16	432	34	28	20
A3.	Air Slew Commands	3	3	18	1-u11 1-SQRT 1-ASIN	372	52	11	-
A4.	Slew Autopilot	5	8	29	4-LIM	185	66	16	-
A5.	Roll Command for Induced Roll Reduction	3	2	10	0	38	26	5	-
	CONTROL GAIN DETERMINATION								
A6.	Band Switched Gains	5	0	17	0	35	30	2	-
A7.&A8.	Gain Determination with Aero Estimates Simple Model (Parts a + b)	0	9	29	1-u20 4-u17	284	77	7	30

TABLE 13

AUTOPILOT

COMPUTER REQUIREMENTS

A9.&A10. Variation (Parts a + b)	2	19	73	4-u20 1-u17 6-uI8 1-SQRT 1-DADD	1110	190	13	101
A11&A12. (Fin Effectiveness Model (Parts a + b)	2	19	85	7-u20 2-uI8 5-u19 1-SQRT 1-DADD	1905	223	19	424
A13&A14. Cross Coupling Model (Parts a + b)	6	23	109	7-u20 2-u18 9-u19 1-SQRT 1-DADD	2794	284	19	706
A15&A16. Single Panel Model (Parts a + b)	101	199	514	7-u20 24-u19	8077	958	20	1202

LEGEND:

* Totals include 30% additional short operations for subroutine linkages and other miscellaneous overhead instructions.

TABLE 14

STRAPDOWN INERTIAL REFERENCE

COMPUTER REQUIREMENTS

MODULE	NAME	ADD/ SUB	MULT/ DIVIDE	LOAD/ STORE	UTILITIES	*EQUIVALENT ADDS	MEMORY PROGRAM	DATA RAM	ROM
11.	Missile Attitude Determination	62	55	121	4-Int	743	283	45	-
12.	Missile Velocity Determination	7	12	28	3-Int 1-SQRT 2-Double Add	306	65	20	-
13.	Missile Position Determination	1	3	13	3-Int 1-SQRT 2-Double Add	213	35	20	-
14.	Angle of Attack Determination	7	11	22	2-ATAN	261	55	10	-
15.	Mass and Balance Estimation	4	3	15	1-u20 1-u17	117	30	10	19
16.	Aero Parameter Estimation	0	3	11	1-u20 2-u17	135	26	6	39

LEGEND:

*Totals include 30% additional short operations for subroutine linkages and other miscellaneous overhead instructions.

4.2 RADAR SENSORS

In this subsection, various candidate radar sensor types are identified and their respective performance features reviewed in the context of air-to-air missile applications. Compatible sensor types are then selected and described in more detail from an operational mode viewpoint. Signal processing is then addressed and analog/digital boundaries identified for each sensor type. The form and function of the digital spectrum analyzer as a primary digital signal processing element is discussed followed by the definition and analysis of the total digital signal processing chain of functions culminating in the derivation of computer loads.

4.2.1 Candidate Sensor Types vs Air to Air Missile Requirements

Of the several different types of radar systems defined and developed to date, the following three general classes are candidates for missile radar sensors:

- Pulse
- Doppler continuous-wave (CW)
- Pulse-doppler

These general classes of radar sensors can be operated in a semi-active or active mode. For the semi-active mode the radar system transmitter is in the launch aircraft. The role of the missile radar sensor is to select/acquire the radio frequency energy reflected by the target and process the signal to derive guidance commands. For the active mode the missile seeker incorporates the transmitter.

The operational requirements for air-to-air missiles given in Section 3.2 disqualify both the active and semi active pulse radar sensors due to their inability to reject ground clutter via doppler frequency discrimination, thereby masking low altitude target returns. CW doppler and pulse doppler radar sensors take advantage of the motion of the target using the doppler principle to separate moving targets from fixed ground clutter.

The CW doppler active sensor is not an ideal sensor candidate for missile applications due to the high transmitter leakage into the receiver which in turn degrades the receiver sensitivity. It is possible, by the use of separate receiving and transmitting antennas and by an RF leakage cancelling network (Feed-Thru Nulling) to obtain a receiver sensitivity limited only by receiver input noise. This is currently not practical for missile use.

Based on the foregoing review of operational requirements versus radar sensor capabilities, the following sensor types are selected for further analysis:

SENSOR TYPE	MISSILE CLASS
Semi-active/CW doppler	I, II, III
Semi-active/Pulse doppler	II, III
Active/Pulse doppler	II, III

The distinguishing functional features of these radars are discussed in the following paragraphs using first-level functional block diagrams.

Semi-Active/CW Doppler Radar Sensor - The major functional elements of a semi-active CW doppler (SA-CW) sensor are shown in Figure 15. The sensor is comprised of a rear receiver, front receiver, and signal processor. The function of the rear receiver is to receive and track the illuminator signal as a reference, coherently offset this reference, and provide this offset reference with sufficient spectrum purity and power level to the front receiver monopulse mixers. The front receiver amplifies the doppler shifted illuminator energy reflected from the target in three narrow-band monopulse channels, and coherently translates this information to baseband where it is encoded and fed to the digital signal processor. The function of the digital signal processor includes extraction of target angle and velocity errors, target detection and verification, and generation of a variable frequency reference to the rear receiver to provide AFC tracking of the target.

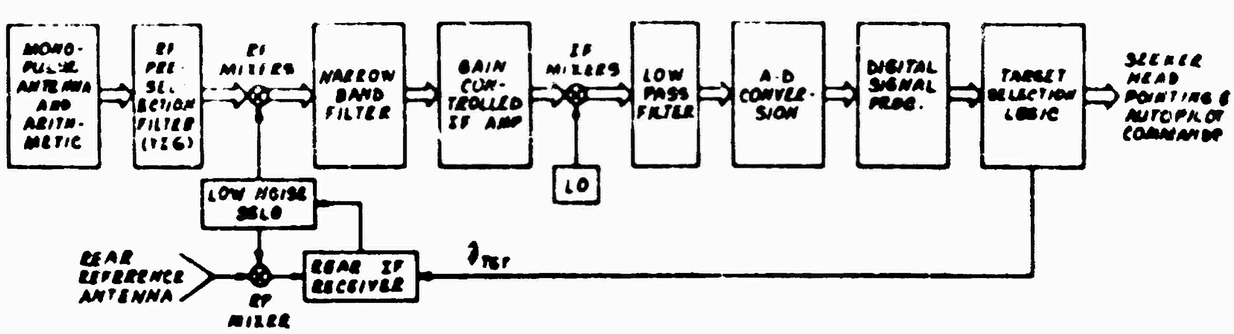


Figure 15 Semi-Active CW Doppler Radar Sensor

Semi Active/Pulse Doppler Radar Sensor - The semi-active pulse doppler (SA-PD) sensor configuration is shown in Figure 16. The addition of a pulse compression line, front range gates and a range gate generator provides the capability of operating with low duty cycle pulsed doppler illuminator energy. The PD waveform provides multiple target range discrimination, and clutter rejection improvement by range discrimination.

Active Pulse Doppler Radar Sensor - Active pulse doppler (A-PD) sensors incorporate the system components shown in Figure 17. The addition of a transmitter provides a self-contained sensor which is independent of the launch aircraft.

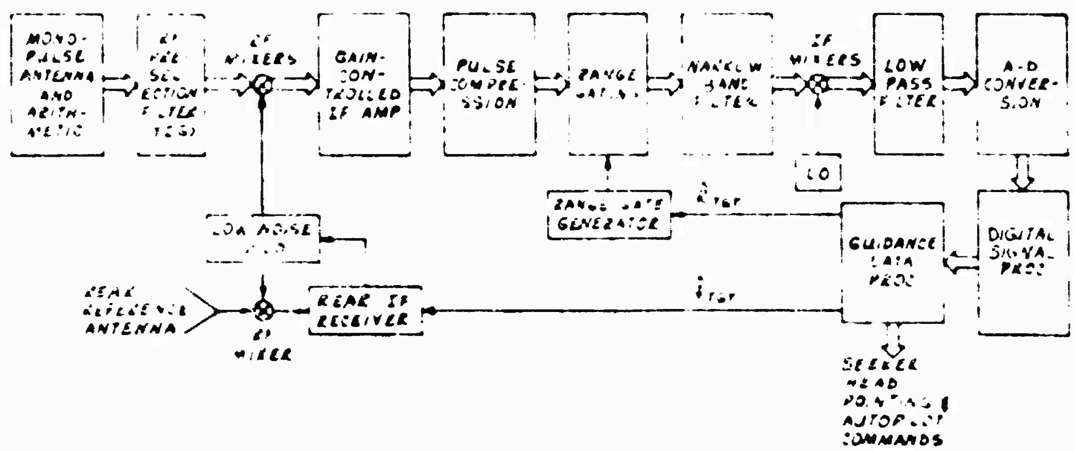


Figure 16 Semi Active Pulse Doppler Radar Sensor

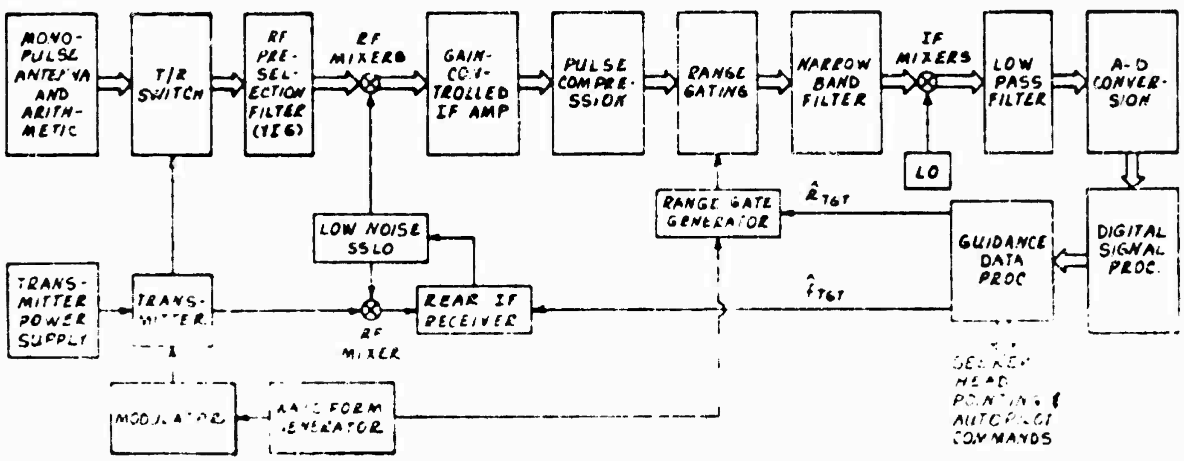


Figure 17 Active Pulse Doppler Radar Sensor

System Design Requirements - To qualify the requirements imposed on digital signal processing, the set of performance requirements and missile parameters given in Section 3.2 have been established for air-to-air missiles employing the three different types of radar sensors for both existing and projected (1980's) missile system parameters.

It is assumed that each interceptor or launch aircraft has an airborne inteceptor (AI) radar. The function of the AI radar is to search for and acquire threat aircraft and initialize the missile for launching. Thus, the missile sensor would be aided by the AI radar in initial target designation. A reasonable requirement to impose upon the AI radar by the missile radar sensor is the provision of missile antenna pointing commands accurate to within the sensor beamwidth to insure locking onto the desired target. The maximum doppler/closing velocity error designation of ± 10% is based primarily upon the missile speed uncertainty. The maximum range (R) uncertainty of ± (500 ft/152.4m + 1%R) is based primarily upon the ability of the AI radar to measure range. Avionics designation accuracies for the three missile types are summarized in Section 3.2, also, since radar sensors must perform in an environment which includes ground clutter, rain, and ECM, environmental parameters and conditions pertinent to each missile class defined in Section 3.2 are inputs to the sensor selections and design process.

The following paragraphs discuss the system design requirements and implications for: target acquisition, target tracking and ECCM.

Target Acquisition - Radar sensor target acquisition is based upon the capability of the AI radar to designate the position of the target prior to launch in the case of Class I and II missiles, or via a command guidance link for Class III/active missile radar sensors. Table 15 lists the sensor performance requirements which impact on target acquisition. These requirements were derived from the information contained in Section 3.2.

Ground clutter caused by microwave energy striking the surface of the earth and re-radiating into the missile sensor can produce degraded target acquisition performance. For approaching

TABLE 15

DERIVED SENSOR REQUIREMENTS

PARAMETER	MISSILE CLASS I	II	III
Dynamic Limits			
Range, max. (nmi)	10	30	20/50
Closing Velocity, min/max (fps)	200/5000	200/7500	200/9500
(mps)	61/1524	61/2286	61/2895
Angle, gimbal (deg)	±60	±60	±60
Search Uncertainity Ranges			
Initial			
Range (ft/m)	NA	±2300/701	±5000/1524
Closing Velocity (fps)	±500	±500	±500
(mps)	±152	±152	±152
Angle (deg)	8	6	6
Frame Time (sec)	0.5	1.0	1.0
Reacquisition			
Range (ft/m)	NA	±1500/457	±500/152
Closing Velocity (fps)	±500	±500	±500
(mps)	±152	±152	±152
Angle (deg)	8	6	6
Frame Time (sec)	0.1/0.2	0.05/0.2	0.05/0.2

TABLE 15 (Continued)

DERIVED SENSOR REQUIREMENTS

PARAMETER	MISSILE CLASS I	II	III
Detection Performance			
Prob. of Det. in one second	0.95	0.95	0.95
Avg. Time between false alarms	1 sec	1 sec	1 sec
Type/Mode			
CW-Semiactive	HAW	HAW	MC/TERM
Pulse Doppler-Semiactive	NA		
Dedicated Illuminator		HAW	MC/TERM
Shared Illuminator			MC
Pulse Doppler-Active	NA	TERM	TERM

LEGEND:

NA - not applicable

MC - midcourse phase

TERM - terminal phase

HAW - homing all the way

targets the three sensors proposed can cope with ground clutter since the target's doppler frequency is outside the clutter doppler frequency region. However, for the tail-chase missile the target doppler must compete with ground clutter. The effect of different waveforms on subclutter visibility or signal-to-clutter ratio (SCR) for the receding target situation has been analyzed in numerous studies. Figure 18 is a representative plot of SCR for a low altitude receding target as a function of missile-to-target range (R_{MT}) for the SA-CW, SA-PD, and A-PD waveforms. The geometry assumed for this plot is a co-altitude AI radar, missile and target, with the AI radar remaining at a constant 20 nmi range from the target. This situation is somewhat unfavorable for the A-PD system since the missile is assumed to be flying at a constant altitude which implies that the clutter power entering the missile is approximately constant throughout the flight. In the SA-CW and SA-PD systems, the clutter power entering the missile decreases as the missile gets further from the AI radar, i.e.:

$$\left.\frac{S}{C}\right|_{A-PD} = \left(\frac{R_{MC}}{R_{MT}}\right)^4 \approx \left(\frac{1}{R_{MT}}\right)^4$$

$$\left.\frac{S}{C}\right|_{\substack{SA-PD \\ SA-CW}} = \left(\frac{R_{IC}}{R_{IT}}\right)^2 \left(\frac{R_{MC}}{R_{MT}}\right)^2 \approx \left(\frac{R_{IC}}{R_{MT}}\right)^2$$

where R_{MC} and R_{IT} are approximately constant throughout the flight.

The SCR for the SA-PD signal processor is improved over the SA-CW signal processor by the ratio of the duty cycle (a 3% duty cycle implies 15db less clutter area seen by the missile).

A high duty cycle is normally employed for A-PD to lower the peak power requirements of the sensor transmitter.

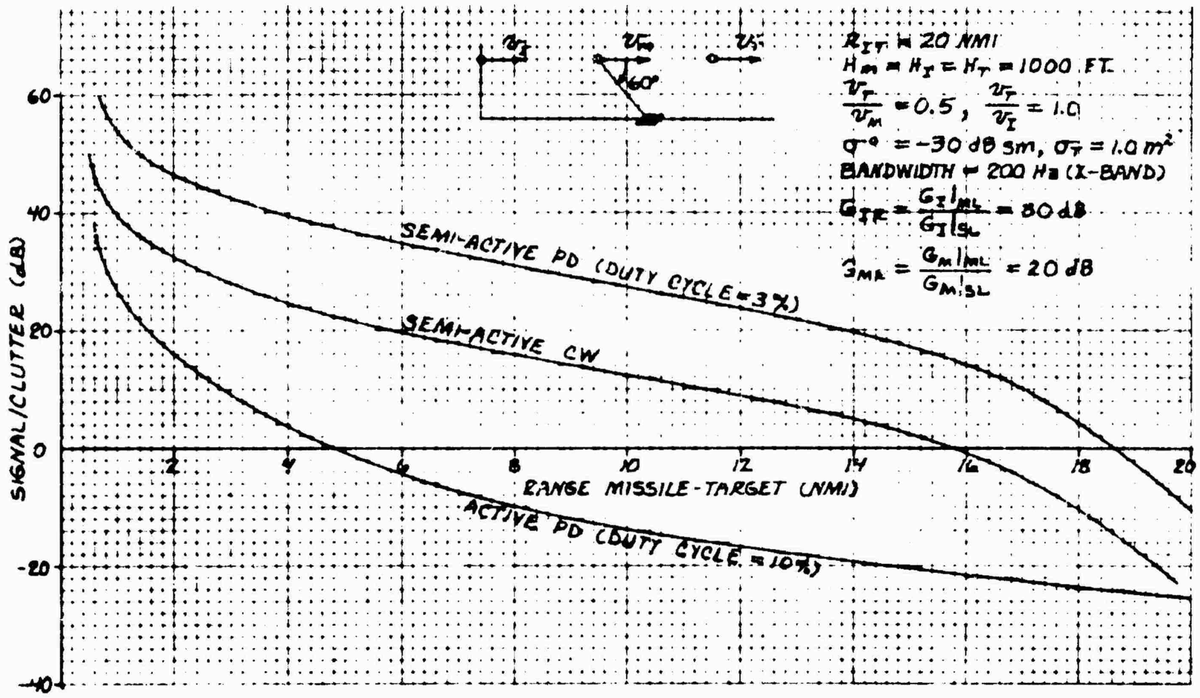


Figure 18 Radar Sensor Signal to Clutter Ratio (SCR) Versus Missile Target Range

A summary of design parameters for the three cnadidate radar sensors is presented in Table 16. Note that the A-PD sensor can be mechanized at X or K_u band. The reason for considering the higher frequency K_u band is that narrower antenna beamwidths and higher gain can be acheived and thus better tracking accuracy and multiple target resolution can be obtained. However, a dual mode SA-PD/A-PD system would want to use the same antenna thus requiring that the A-PD system use the same RF frequency as the SA-PD system. Also, higher transmitter power can be developed at X band than K_u band.

TABLE 16

RADAR SENSOR DESIGN PARAMETERS

PARAMETER	SA-CW	SA-PD	A-PD
		SENSOR TYPE	
Carrier Frequency (band)	X	X	X or K_u
Dynamic Range			
Doppler (KHz)	4/100	4/150	4/190 @ X 6/295 @ K_u
Angle (deg)	±60	±60	±60
Acquisition Window			
Range (μs)	NA	3	1
Doppler (KHz)	20	20	20/30
Time (sec)			
Initial	0.5	1.0	1.0
Reacq.	0.2	0.1	0.1
Angle (fraction of beamwidth)			
Class I	0.3/0.47	NA	NA
Class II	0.43/0.67	0.43/0.67	0.43/0.67
Class III	0.57/0.67	0.57/0.67	0.57/0.67@X 0.8/1.0 @K_u

To prevent erroneous acquisition of clutter and its designation as a target, the clutter mainlobe is acquired and tracked prior to attempting target acquisition. For approaching low altitude targets, the doppler region searched to acquire the target is positioned above the mainlobe clutter doppler and for receding targets is positioned below the mainlobe clutter doppler.

Range and Doppler Track - The range and doppler track system can be mechanized in several ways depending on the final application. For a missile seeker that makes maximum use of digital signal processing, the spectrum analyzer approach appears to be desirable. The generation of the range and doppler tracking errors using an FFT digital signal processor is illustrated in Figure 19.

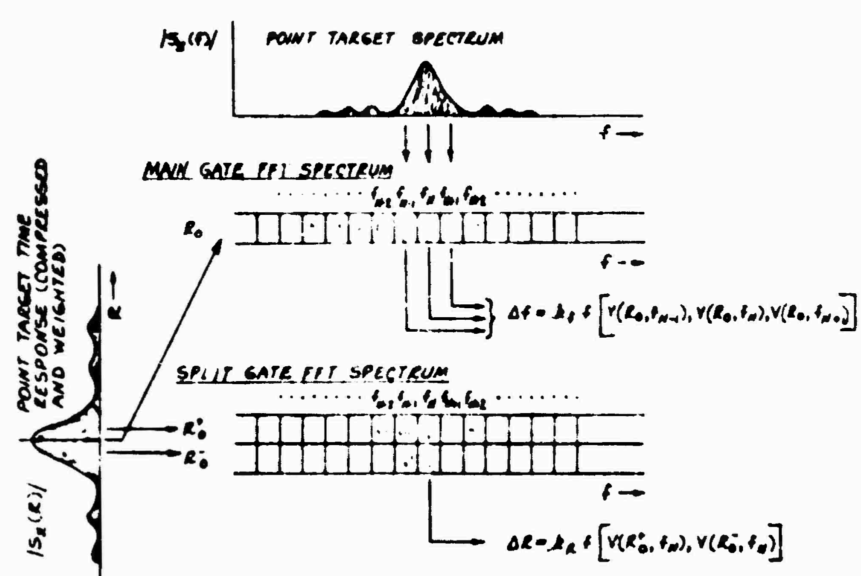


Figure 19 Range & Doppler Tracking Error Generation - Spectrum Analyzer Implementation

The doppler tracking error is generated from the spectrum of the main range gate return by combining the outputs of the center three doppler cells (see subsection 4.2.6). The main range gate is kept centered on the peak of the target return by the range tracking loop which uses the range error obtained from combining the outputs of the split gate FFT spectrums (see subsection 4.2.6). The range and doppler errors are filtered in the guidance data processor to develop the estimated target doppler, $\hat{f}_{tgt}$, and the estimated target range, $\hat{R}_{tgt}$. The estimated target doppler and ranges are then fed via D-A convertors to the front and rear IF receivers to complete the tracking loops.

Angle Track - The multi-channel digital spectrum analyzer approach to the determination of monopulse pitch and yaw errors also appears to be consistent with the optimum use of digital signal processing in the various radar sensors. This type of angle error processing is illustrated in Figure 20. The amplitude and phase of the difference channel signal relative to the sum channel is directly proportional to the angle error for the target being tracked. One important feature of this type of processing is that the target being tracked does not have to be in any particular doppler cell to allow the angle errors to be determined. For example, in the case of a blinking jammer engaged by a SA-CW seeker, the Doppler tracking error can grow large during the time the jammer is on due to the target maneuvers. However, when the jammer turns off it is not necessary to center the target in the doppler spectrum before determining the angle errors as long as the target is still in one of the FFT doppler cells.

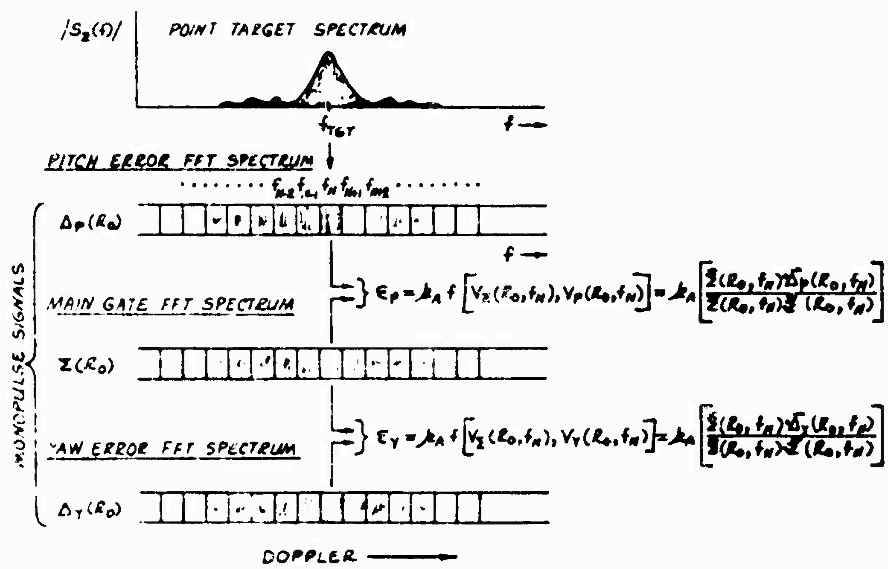


Figure 20 Angle Tracking Error Generation Spectrum Analyzer Implementation

Electronic Counter-Measures - The ECCM techniques that are considered in each radar sensor address ECM threats, such as, barrage, spot, and deceptive jammers. The spot and barrage jammers are best handled by a home-on-jam type receiver. The digital signal processor is capable of handling this type of signal as a normal signal and is able to identify the jammer signal from a skin track signal. The use of a narrow band front end filter at RF provides the following additional advantages.

1. Only energy within the pass band of these filters can degrade missile performance. As an example, out-of-band AM or FM jammer energy will not result in cross modulation products being developed in the front or rear receiver mixers.

2. The front and rear receivers image response has been suppressed 36 db by the RF pre-selection filters. In fact, the lack of image suppression results in the missile seeker being equally responsive to jammers at the image frequency. A swept noise jammer, as an example, could be within the seeker's pass band twice every sweep effectively doubling the duty cycle of receiver jamming.

The deceptive type of ECM threat can best be handled by a combination of a monopulse antenna and digital signal processing logic.

The use of a monopulse antenna and the simultaneous signal processing of all monopulse antenna channels, greatly reduces susceptibility to the amplitude modulation type jammer.

4.2.2 Digital vs Analog Signal Processing

From the foregoing analysis of radar sensor types applicable to air-to-air missiles, the following significant points emerge concerning digital versus analog signal processing.

1) The digital signal processor has an analog-digital interface with the front receiver only. The receiver furnishes monopulse data and AGC levels.

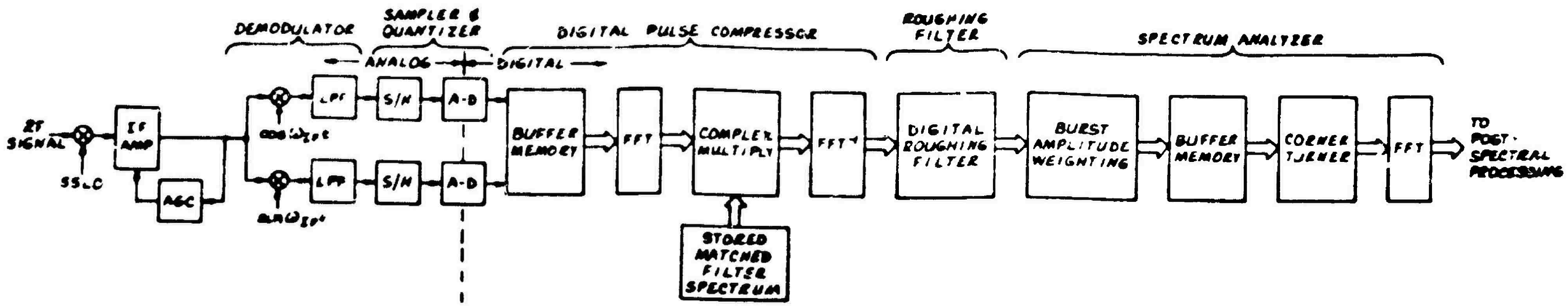
2) The range and doppler tracking loops can be closed through a digital processor as in the case of the seeker head and autopilot loops.

3) The digital signal processing function consists of two sub-functions; spectrum analysis and signal processing logic as detailed in Section 4.2.4.

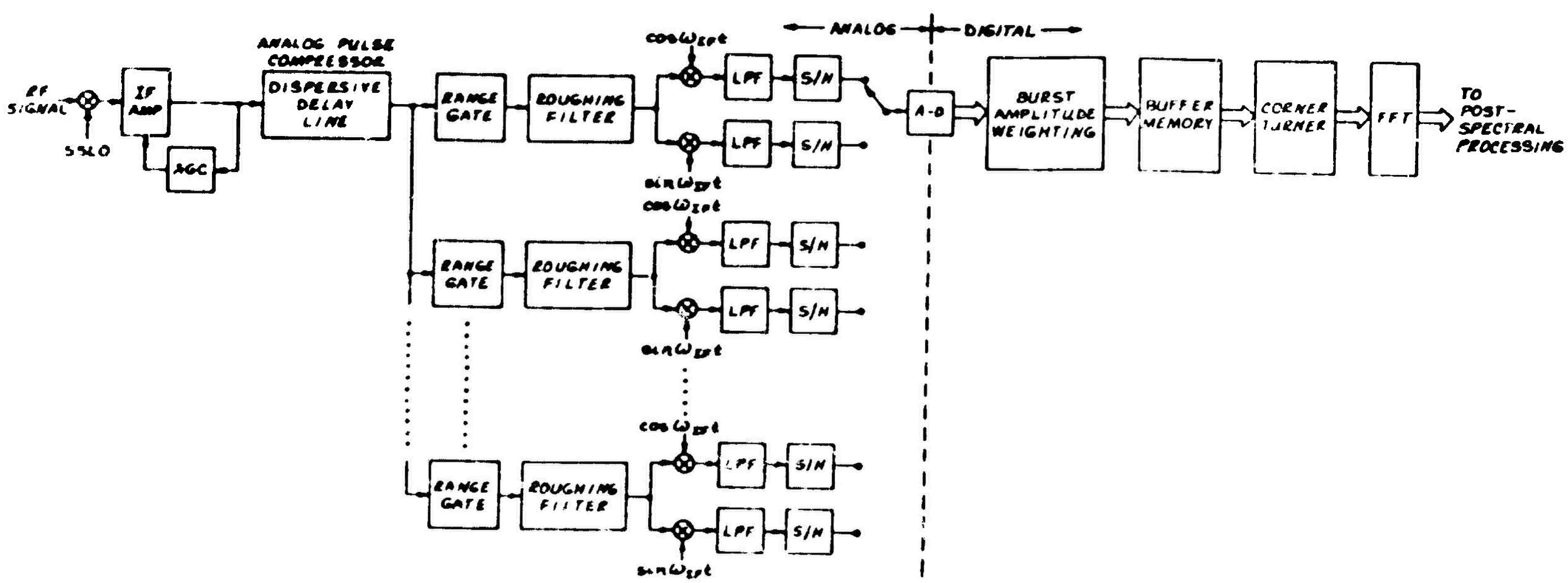
As can be seen from the basic functional block diagrams for the three radar sensors of interest (Figures 15,16 and 17), the interface between analog and digital signal processing has been defined to be at that point in the receiver where the signal spectrum has been narrow-banded (e.g., from 1.0 KHz to 20.9 KHz IF bandwidth). However, for the pulse doppler radar sensors, one could consider moving the analog-digital interface "up-stream" to the pulse compression function. There are both advantages and disadvantages to doing this and these are discussed in the following sections.

4.2.2.1 Digital Pulse Compression

A functional signal processing block diagram for the monopulse sum channel is shown in Figure 21(A) where the analog digital interface has been moved forward to include the pulse-compression function. Shown in Figure 21(B) for comparison purposes, is the equivalent signal processing functional block diagram where analog pulse compression is used. It should be



(A) Digital Pulse Compression



(B) Analog Pulse Compression

Figure 21 Digital vs Analog Pulse Compression - Functional Block Diagrams noted that there are other possible configurations utilizing digital pulse compression such as converting back to analog after pulse compression.

Signal processing utilizing digital pulse compression can be divided into five major subsystems: Demodulator, sampler and quantizer, pulse compressor, roughing filter, and spectrum analyzer. The demodulator converts the chirped signal spectrum centered in the IF passband into baseband in-phase and quadrature channels. The demodulator consists of an IF frequency local oscillator with quadrature outputs feeding a balanced mixer pair followed by two low pass filters to separate the baseband I and Q terms from the mixer 2nd harmonic terms. The sample/hold circuits continuously sample each channel followed by two A-D converters encoding the sampled time-varying channels into a corresponding digital data stream. The signal processing from this point onward is entirely digital.

Pulse compression is performed on the chirped signal by Fourier transforming the signal channel, using the Cooley-Tukey FFT algorithm, and then multiplying it by its matched filter spectrum, and transforming the resulting dechirped signal back to the time domain with another FFT operation. At the same time, a function analogous to range gating has taken place in the A-D conversion process, as each time sample corresponds to a sample in range. The dechirped pulses from the pulse compressor are integrated in a digital bandpass filter which is essentially N filters, filtering N sets of range samples. This is analogous to the N roughing filters used in the N range channels in the analog pulse compression system shown in Figure 21(B). It is in these filters that both systems obtain the major portion of their clutter rejection. At this point in the digital process, the bandwidth of each of the N multiplexed channels has been sufficiently narrowed to allow the sample set from the roughing filter to be sampled or "thinned". The thinned data stream is then processed by a Doppler spectrum analyzer routine, data being first digitally time weighted. A spectrum analysis is then performed on each Dwell/Burst for each of N range channels by performing an FFT on the digital sample set.

The process of pulse compression can be best illustrated by observing Figure 22. A chirped pulse anywhere in the sampling interval has the same amplitude spectrum and quadratic phase term but has a different linear phase term depending upon its position in the sampling interval. If the time reference is taken to be the center of that interval, then a pulse f(t) centered in that interval has a transform $F(\omega)$, a pulse shifted to the right, $f(t - \tau)$, a transform $F(\omega)e^{-j\omega\tau}$ and a pulse shifted to the left, $f(t + \tau)$, a transform $F(\omega)e^{-j\omega\tau}$. Matched filtering by multiplying by the complex conjugate spectrum, $F^*(\omega)$, simply removes the quadratic phase term, compressing the pulse in the time domain, but leaves the linear phase term intact and thus maintaining the pulse's position in the sampling interval. For an ideal pulse compression the transformation would be

$$F(\omega)F^*(\omega)e^{-j\omega\tau} \quad \longleftrightarrow \quad \frac{\sin(t - \tau)}{(t - \tau)}$$

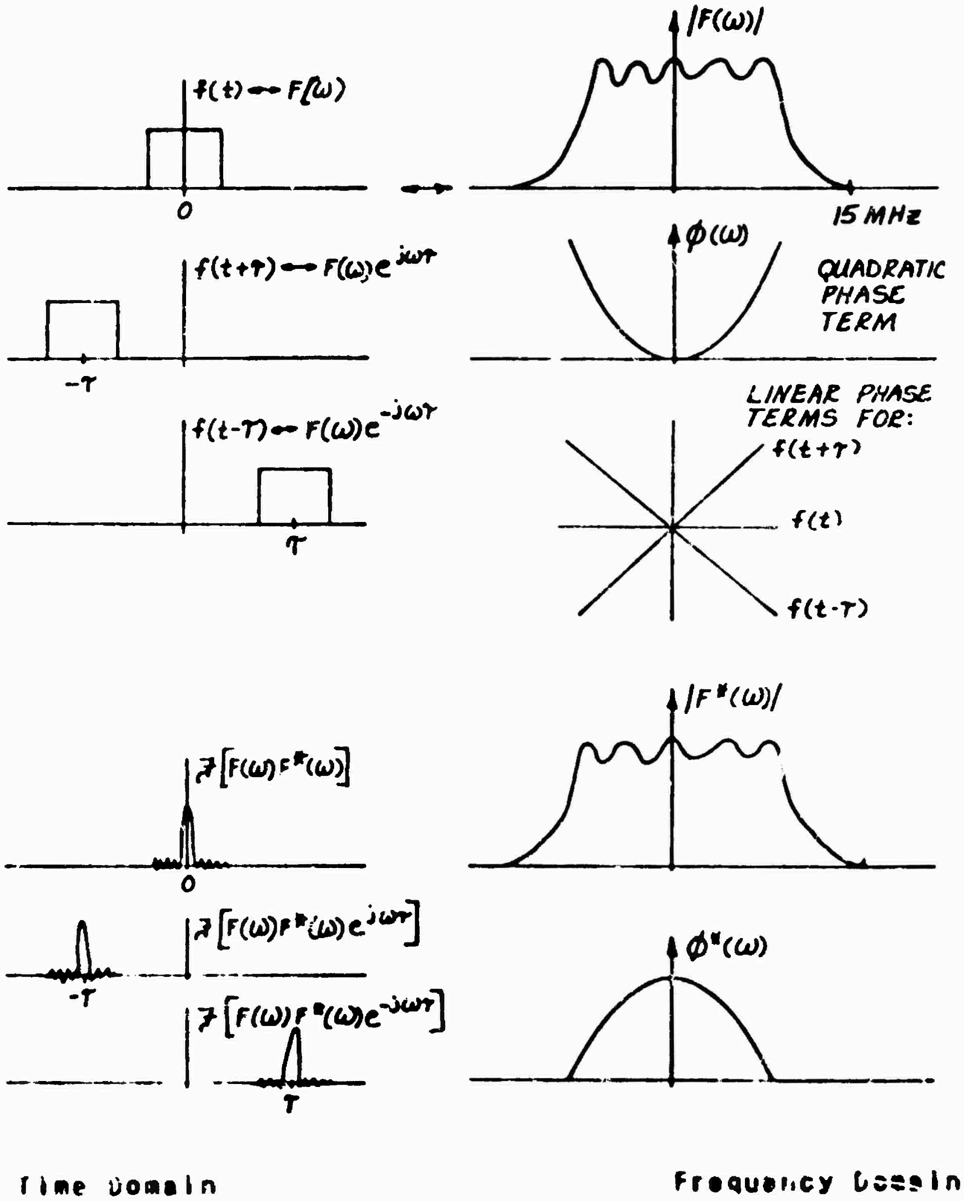


Figure 22 Pulse Compression Waveforms and Spectra.

Weighting of the time sidelobes can be implemented by multiplying the matched filter function $F^*(\omega)$ by a frequency weighting function, $W(\omega)$, to mismatch the filter slightly but reduce the size of the time sidelobes.

4.2.2.2 Advantages/Disadvantages of Digital Pulse Compression

It has been determined for the purposes of this study that digital pulse compression will not be considered for sizing the computer loads presented in Section 6.0. Present technology favors analog pulse compression using acoustic dispersive delay lines, especially for airborne applications where a premium is placed on small size, weight, and power consumption. The design and development of the acoustic dispersive delay lines has made significant advances over the past several years negating some of the motivation for the development of digial pulse compression systems. However, digital pulse compression does offer some significant advantages for many applications where a premium is not placed on size weight, and power. The advantages /disadvantages of digital pulse compression relative to analog pulse compression are summarized in Table 17.

TABLE 17

ADVANTAGES/DIS-ADVANTAGES OF DIGITAL PULSE COMPRESSION RELATIVE TO ANALOG PULSE COMPRESSION

ADVANTAGES	DIS-ADVANTAGES
o Performance - Can be close to theoretical, better side-lobe levels can be achelved.	o A-D Converter - Must be wideband and have a wide dynamic range (e.g., 10 MHz chirp requires 10 MHz Data rate; approx. 70 db dynamic range required in clutter environment implies 11 to 12 bits).
o Stability - Compression performance not affected by temperature or age.	o Physical Characteristics - size, weight, power consumption greater than acoustic dispersive delay lines.
o Flexibitliy - Stored matched filter function can be readily modified to handle other waveforms.	
o Time Sidelobe Reduction - the stored matched filter function can include spectral weighting to reduce sidelobes (e.g., Cos freq weighting).	

4.2.3 Spectrum Analyzer Type Digital Signal Processors

A functional block diagram outlining the basic structure of the spectrum analyzer type of digital signal processor is shown in Figure 23. This type of digital signal processing is used for all three types of radar sensor selected for air-to-air missile applications. The only variation in this configuration for the different sensor types/missile classes is the number of channels multiplexed for A-D conversion and processing. This is illustrated in Figures 24 and 25 which show the functional signal processing block diagrams of the CW and PD radar seekers respectively.

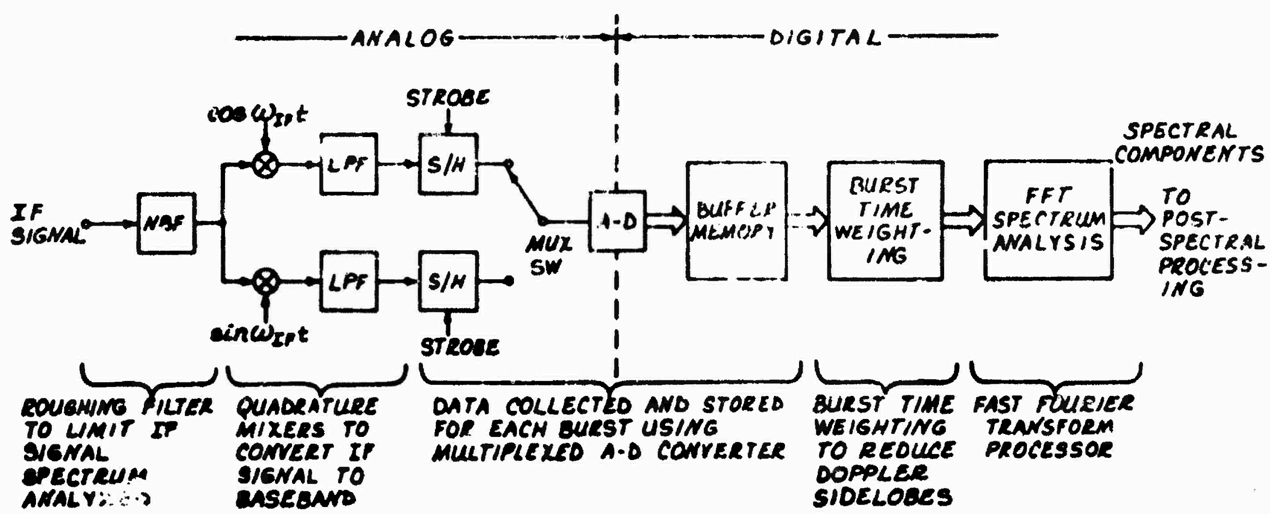


Figure 23 Radar Signal Processing Functional Flow, Hybrid Analog with Digital Spectrum Analyzer

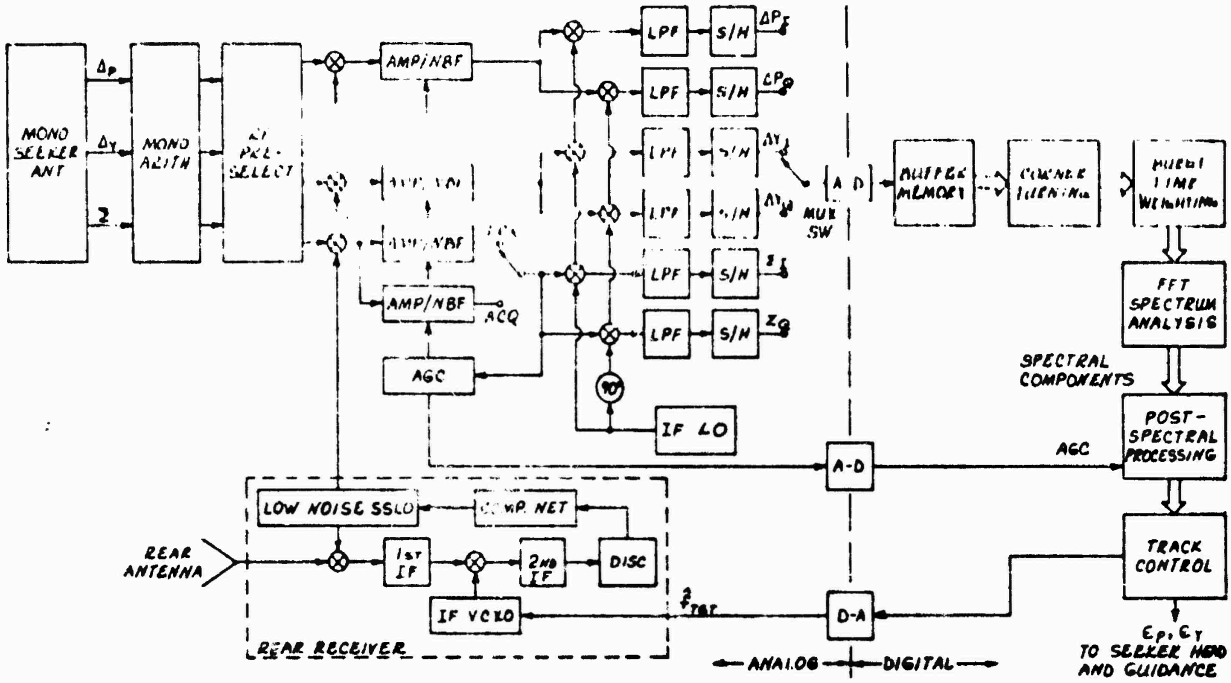


Figure 24 CW Radar Sensor with Digital Signal Processor

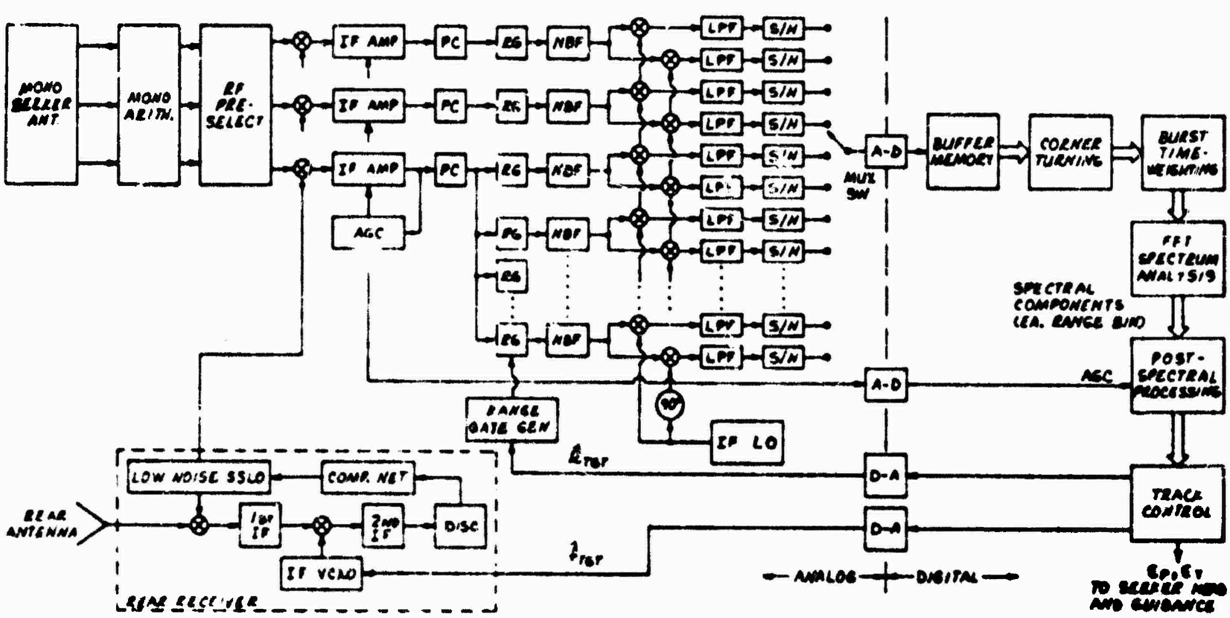


Figure 25 PD Radar Sensor with Digital Signal Processor

The CW radar seeker utilizes three identical spectrum analyzer channels to process the monopulse signals in the target track mode and a single channel to process only the sum signal in the acquisition mode. The only difference between the track mode spectrum analyzer and the acquisition mode spectrum analyzer being the width of the spectrum analyzed (i.e., the intermediate frequency bandwidth of the roughing filter).

The PD radar seeker, which employs range tracking, has as a minimum five spectrum analyzer channels in the track mode (i.e., Σ, Δ_p, Δ_y, Σ+, Σ- - splitgate). The number of spectrum analyzer channels in the acquisition mode depends on the specific missile mechanization (see candidate sensor configurations in subsection 4.2.1).

It should be noted that the FFT type of spectrum analyzer is by no means the only type of spectrum analyzer that could be used in conjunction with digital signal processing. Recent advances in charge-coupled devices (CCD) and acoustic delay lines have made analog multi-channel spectrum analysis a feasible alternative to digital FFT spectrum analysis. However, for the purposes of this study, which is to develop computer requirements, the FFT type of spectrum analyzer will be assumed.

4.2.3.1 Spectrum Analyzer Operation

Roughing Filter - The purpose of the roughing filter is to limit the extent of the IF signal spectrum being analyzed. The roughing filter is centered on the part of the spectrum required by controlling the frequency of the radar frequency (RF) local oscillator (LO). In the track mode, the estimated doppler frequency of the target being tracked is used to control the local oscillator frequency. In the acquisition mode, a search generator controls the LO frequency by moving it in discrete steps to cover the doppler ambiguity region.

Conversion to Baseband - The band limited IF signal is converted to its in-phase and quadrature baseband components by mixing the IF signal with an in-phase and quadrature IF LO reference followed by low-pass filtering to eliminate the high frequency second harmonic terms.

Analog-to-Digital Conversion - The in-phase and quadrature baseband signals are digitized using a single time-multiplexed A-D convertor. The use of a single A-D convertor to digitize both channels minimizes hardware and eliminates channel mismatch errors due to differences in A-D convertor characteristics (e.g., conversion accuracy). In multi-channel spectrum analyzers, the use of a single multiplexed A-D achieves considerably more hardware savings compared to dedicated A-Ds. The A-D sampling rate per in-phase and quadrature baseband channel must be greater than or equal to the IF roughing filter bandwidth to prevent spectral foldover (aliasing) of the spectrum being analyzed. For example, if the roughing filter bandwidth is 1.0 KHz, the A-D sampling rate per in-phase and quadrature channel must be at least 1.0 KHz requiring a composite A-D conversion rate of 2.0 KHz for a single complex channel spectrum analyzer and $\geq$ 6.0 KHz for three complex channel system, etc.

Buffer Memory - A buffer memory is also required since the FFT processing cannot commence until at least one-half of the number of complex data samples has been collected. Also, when multiple complex channels are sampled (e.g., signals from multiple range gates or signals from the three monopulse channels) using a single time-multiplexed A-D, the complex samples must be "rearranged/sorted" (referred to as corner-turning) prior to FFT processing. In this case FFT processing is delayed until all of the complex samples for a complete dwell/burst have been input to the buffer. Such a buffer could be the main data base memory of a processor as opposed to a separate unit.

Corner-Turning - As indicated above, a corner-turning operation on the buffered complex samples is required prior to FFT processing, when multiple-channel information has been collected using a single multiplexed A-D. The data sequenced into the buffer for K channels may be represented as:

$$V_{1I}(1), V_{1Q}(1), V_{2I}(1), V_{2Q}(1), -- V_{KI}(1), V_{KQ}(1), V_{1I}(2), V_{1Q}(2)$$
$$---- V_{1I}(n), V_{1Q}(n), ---- V_{KQ}(n)$$

For FFT processing the data is rearranged or sorted into the following sequence:

$$V_{1I}(1), V_{1Q}(1), V_{1I}(2), V_{1Q}(2), -- V_{1Q}(n), V_{2I}(1), ---- V_{2Q}(n),$$
$$---- V_{KI}(1), V_{KQ}(1), ---- V_{KQ}(n)$$

This sequence of complex samples allows sequential processing using conventional general purpose computer indexing techniques and a single 2-point transform subroutine/program module as described in Section 6.

Burst Amplitude Weighting - As indicated in Figure 23, Burst amplitude weighting (sometimes called Burst Time Weighting) is used to suppress/reduce spectrum sidelobes at the expense of broadening the spectrum mainlobe. The theory of burst amplitude weighting is explained in Appendix B. It should be noted, that the burst amplitude weighting may be executed on the complex samples in real-time as they are input to the buffer instead of after the corner-turning operation.

Fast Fourier Transform Processor - The spectrum of the weighted data is generated by using a discrete Fourier transform. This operation can be represented mathematically as:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}$$

where,

$x(n)$ = nth complex sample of the data sequence being transformed

$W_N = e^{\frac{-j2\pi}{N}}$

N = Number of complex samples

The most direct computation of the FFT requires an amount of computation proportional to N^2. However, by exploiting the symmetry and periodicity properties of the complex exponential sequences, a fast Fourier transform procedure has been developed which dramatically reduces the amount of computation required (i.e. $N \log_2 N$ instead of N^2 computations). FFT processing is explained in more detail in subsection 4.2.4.

4.2.3.2 Basic Spectrum Analyzer Relationships

The spectrum of interest is defined by passing the signal to be analyzed through a roughing filter. The roughing filtering can be performed on the in-phase and quadrature baseband signals or at IF The bandwidth of the baseband roughing filters is one-half the bandwidth of the IF roughing filter to preserve the same signal spectrum.

$$\beta_{IF} = 2\beta_{BB}$$

The A-D convertor rate, f_{A-D}, must be sufficiently high to satisfy the Nyquist criteria for the roughing filter output spectrum. i.e.

$$f_{A-D} \geq \beta_{IF}$$

or

$$f_{A-D} \geq 2\,\beta_{BB}$$

The data collection interval, T_B, (also called burst time or dwell) is determined by first specifying the desired spectrum granularity or (FFT doppler cell width, β_{CELL}) or visa versa. i.e.

$$T_B = \frac{1}{\beta_{CELL}}$$

The number of points in the FFT spectrum is the same as the number of original data points taken, N_{FFT}. The total bandwidth covered by the FFT spectrum is the product of the number of data points and the FFT doppler cell width. i.e.

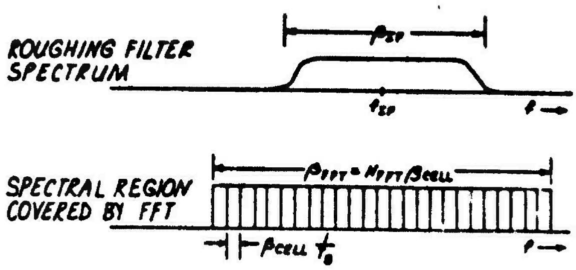
$$N_{FFT} = f_{A-D} \cdot T_B$$
$$= N_{FFT} \cdot \beta_{CELL}$$

The effect of burst time weighting is to reduce the amplitude of the spectral sidelobes at the expense of broadening the spectrum mainlobe. For cosine-squared amplitude weighting, the 3db width, $\beta_{\omega\tau}$ of the weighted spectrum is:

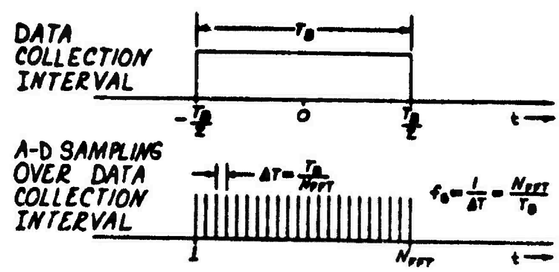
$$\beta_{\omega\tau} = 1.6\ \beta_{CELL}$$

Burst amplitude weighting theory is discussed in Appendix A.

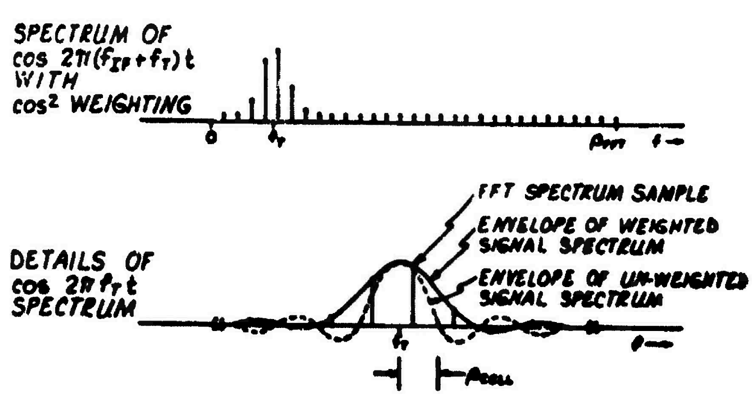
The above spectrum analyzer relationships are illustrated in Figure 26. Also shown in this figure is an example of the spectrum of a sinusoidal signal. Note, that the position of the FFT spectral samples relative to the peak of the envelope of the signal spectrum depends on the exact signal doppler, f_t.



(A) Roughing Filter/FFT Filtering -Frequency Domain



(B) FFT Data Collection Time Domain



(C) Sinusoidal Signal Spectrum - Time Weighting

Figure 26 Basic Spectrum Analyzer Relationships

4.2.4 Mode Control

The function of a radar sensor and associated signal processor at any given time in air-to-air target engagements is determined by the progress through a specific chain of interrelated operational modes culminating in the acquisition and tracking of the designated target. For the radar sensors described in this report the following seven operational modes have been defined:

1) Pre-launch
2) Launch
3) Clutter Acquisition
4) Target Acquisition
5) Track Initiation
6) Target Track
7) Mainlobe Clutter Track

The designation of any one of the above modes for sensor operation requires a control hierarchy and modular structure similar to that described in Subsection 4.6 for missile mode control in single computer systems.

Figure 27 shows a compatible control structure autonomous to radar subsystems, with a real-time executive performing the functions of: mode designation, conflict resolution and input-output interface with associated avionics and missile subsystems and real-time inputs. Individual mode supervisors are called by the executive in accordance with mode

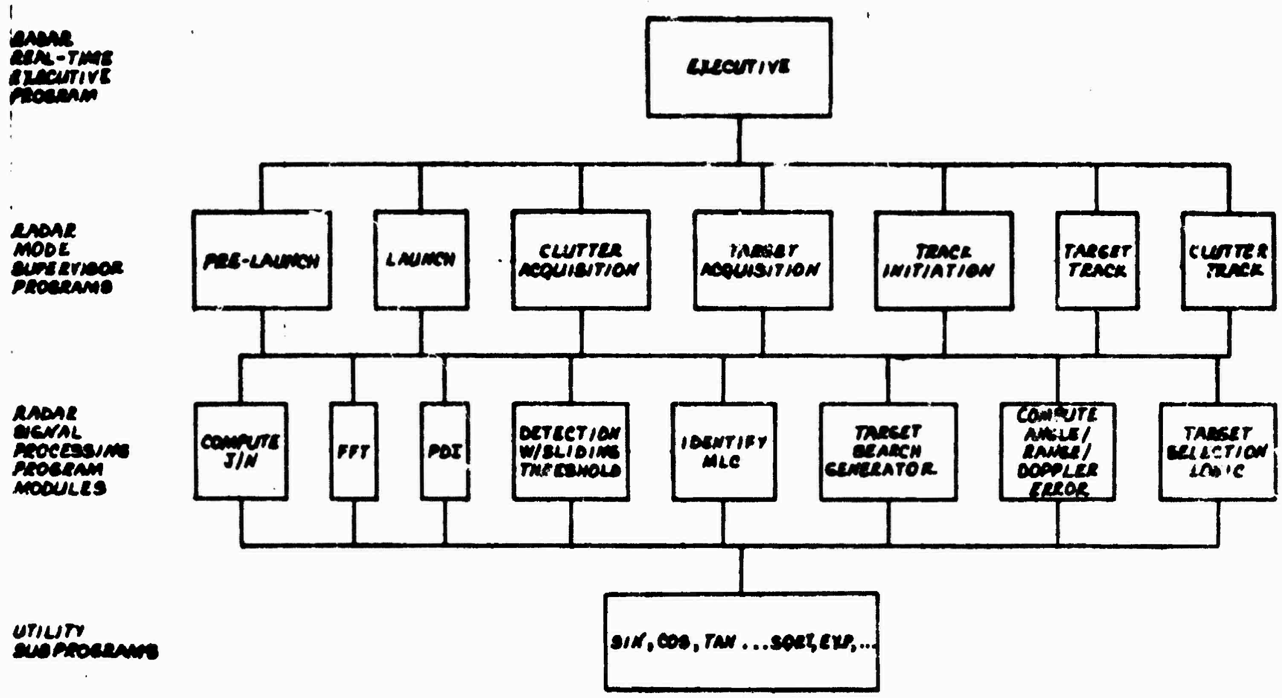


Figure 27 Radar Sensor Mode Control, Modular Hierarchical Program Structure

selection logic routines responsive to status inputs from the mode supervisor programs and external inputs to the sensor subsystem. Mode supervisors in turn call the required radar signal processing program modules (e.g. post-detection integration, beta blanking logic etc.) described in subsection 4.2.5 which again, in turn, call for supporting utility routines. Such a program control structure provides the necessary modularity and flexibility to enable different/improved radar sensor types to be accomodated at a later date without major redesign of the software.

Both the executive program and the individual mode supervisors are described in greater detail in the following paragraphs. Supporting signal processing program modules are described in the following subsection.

4.2.4.1 Executive Program

Figure 28 is a first-level flow diagram of the radar mode control executive program which is applicable to all radar sensor types selected in this study. Decisions to select or change modes are made by the executive program, based on flags raised by the individual mode supervisors indicating mode complete/incomplete or results demanding alternative mode selection and/or input-output actions.

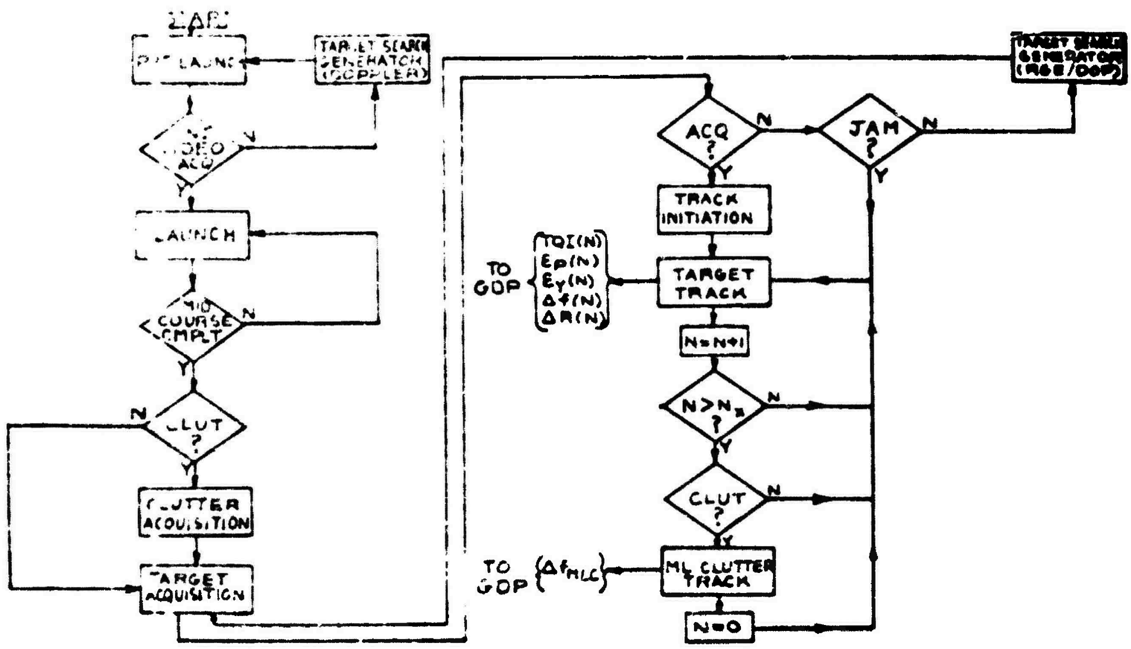


Figure 28 Executive Control Flow Diagram

Commencing with the Pre-Launch Mode, the launch aricraft designates target velocity (and range for PD sensors), to the missile radar sensor. In the Launch Mode, initiated by umbilical separation from the launching aircraft, predicted target and mainlobe clutter doppler frequencies are computed. Based on the predicted dopplers, a decision is made by the executive as to whether or not mainlobe clutter must be acquired. If no clutter flag is set, the Target Acquisition Mode is selected and conversely, in a clutter situation the Clutter Acquisition Mode is initiated, mainlobe clutter is acquired, and its doppler noted. In the Target Acquisition Mode, signal processing algorithms restrict the search region to avoid the main-lobe clutter signal.

Note, that for missiles employing mid-course guidance, the radar Launch Mode is effectively prolonged until the acquisition phase is called, (i.e. the predicted mainlobe clutter and target doppler are updated on a regular basis). If target acquisition is attempted and no targets are detected, a check is made to determine if the missile is being jammed. If no jamming is detected, a Target Search Generator program is called which moves the target acquisition roughing filter in discrete steps over the computed range of possible target dopplers until a detection is obtained. After the target has been acquired, control is transferred to the Track Initiation Mode supervisor. This supervisor provides the verification of target acquisition by re-acquiring the target in a narrower bandwidth (smaller doppler cell). Also, in this mode, the range and doppler tracking loops are initialized.

In the Target Track Mode, boresight errors are generated for skin track or jamming targets, and range and doppler tracking errors for skin track targets. These data are used in the final guidance data processing along with a track quality indicator, (TQI), which is effectively a measure of signal-to-noise ratio for the skin track mode or jammer-to-noise ratio for the home-on-jam mode. On a regular basis during the time the target is being skin-tracked, the executive calls the Clutter Track Mode supervisor which provides an update of the main-lobe clutter doppler. If the target doppler approaches "too close" to the mainlobe clutter doppler, the TQI signal is modified to indicate that a "clutter coast" is desired until the target doppler either increases or drops below the main-lobe clutter doppler. Control continues under the Target Track Mode supervisor until target intercept.

4.2.4.2 Mode Supervisors

The supporting radar mode supervisor programs are described in greater detail in the following paragraphs.

Prelaunch Mode - The Pre-Launch Mode Supervisor flow diagram is shown in Figure 29. In the Pre-Launch Mode, the first task is to lock up the missile's rear receiver to the RF frequency that the AI radar is using to illuminate the target. This is an all analog step requiring no digital signal processing. After rear lock is accomplished, the target doppler frequency is designated to the missile by injecting an RF signal into the missiles front receiver that has a frequency offset proportional to the target's doppler. In this process, the missiles acquisition roughing filter is moved in discrete steps over the complete target doppler ambiguity region until a detection is obtained. After the injected video has been acquired a "Pre-Launcr Mode Complete" flag is set.

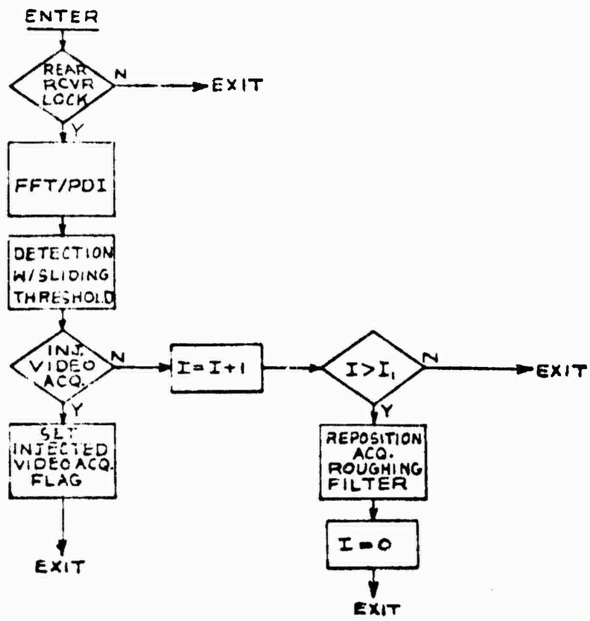


Figure 29 Radar Prelaunch Mode Supervisor, Flow Diagram

Launch Mode - The Launch Mode Supervisor flow diagram is shown in Figure 30. The Launch Mode is called when the Pre-Launch mode is complete and a Missile-Away indication is received by the executive. The purpose of the computations made in Launch Mode is to predict the mainlobe clutter (MLC) doppler and target doppler used for the purpose of initializing the Clutter and Target Acquisition Modes. Also, based on these predicted dopplers, a decision is made by the executive to bypass the clutter acquisition phase if the doppler separation between the MLC and the target is large enough.

As indicated in Section 4.2.4.1, if a midcourse mode is employed, the executive will continue to designate the Launch Mode Supervisor to recycle/refresh the doppler predictions until completion of midcourse is indicated by the guidance data processing.

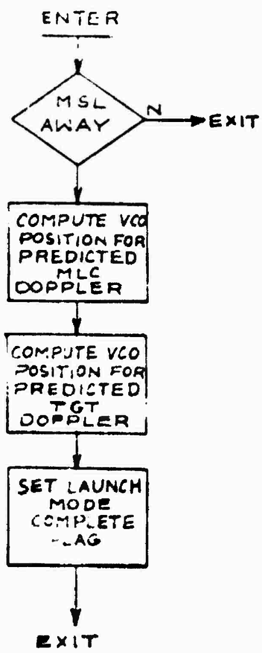


Figure 30 Radar Launch Mode Supervisor, Flow Diagram

Clutter Acquisition Mode - The logic flow diagram for the Clutter Acquisition Mode Supervisor is shown in Figure 31. The Clutter Acquisition Mode is called prior to the Target Acquisition Mode when the possibility of the target doppler being confused with the clutter doppler exists. The first step in this procedure is to center the acquisition roughing filter (filters for the range-gated PD systems) on the predicted clutter doppler.

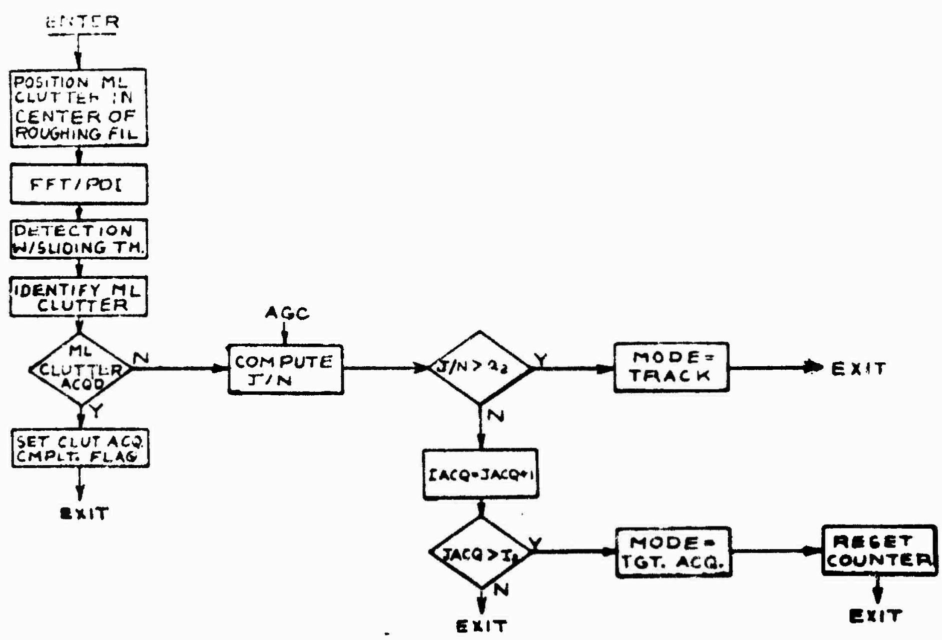


Figure 31 Radar Clutter Acquisition Mode Supervisor, Flow Diagram

An acquisition sequence is then performed which consists of ten consecutive 5 msec dwells. On each dwell the FFT is computed for each range channel resulting in a matrix consisting of M range cells and N doppler cells. The magnitude of the complex signal in each range-doppler cell is calculated and placed in an average value MXN array. The outputs for 10 dwells are summed in this array (this process is called post-detection integration) to give an improvement in signal-to-noise ratio. When the 10 dwells have been completed, a detection procedure is performed on each range channel using a sliding window type of threshold. (See Section 4.2.5 for a description of the FFT/PDI logic and detection with sliding threshold logic). The resulting signals that pass the detection threshold are then summed in the range dimension resulting in a doppler array containing the sum of the signals that have passed the threshold. This array is then searched to locate the greatest return which is designated the mainlobe clutter and its doppler, f_{MLC}, saved. Note, this doppler is used to initialize the clutter track filter (see Section 4.2.5).

If the mainlobe clutter is successfully identified, a "Clutter Acquisition Complete" flag is set and the executive responds by calling the Target Acquisition Mode supervisor. If mainlobe clutter is not identified, the Jammer-to-Noise ratio is computed to determine if the missile is being jammed thus obscuring main-lobe clutter. If the missile is being jammed a flag is set, the executive then transfers control to the Track Mode Supervisor, and target acquisition is bypassed. If the missile is not being jammed, clutter acquisition is again attempted and if after a specified number of attempts, mainlobe clutter has not been acquired (no flag set), the executive calls the Target Acquisition Mode Supervisor.

Target Acquisition Mode - The flow diagram for the Target Acquisition Mode Supervisor is shown in Figure 32.

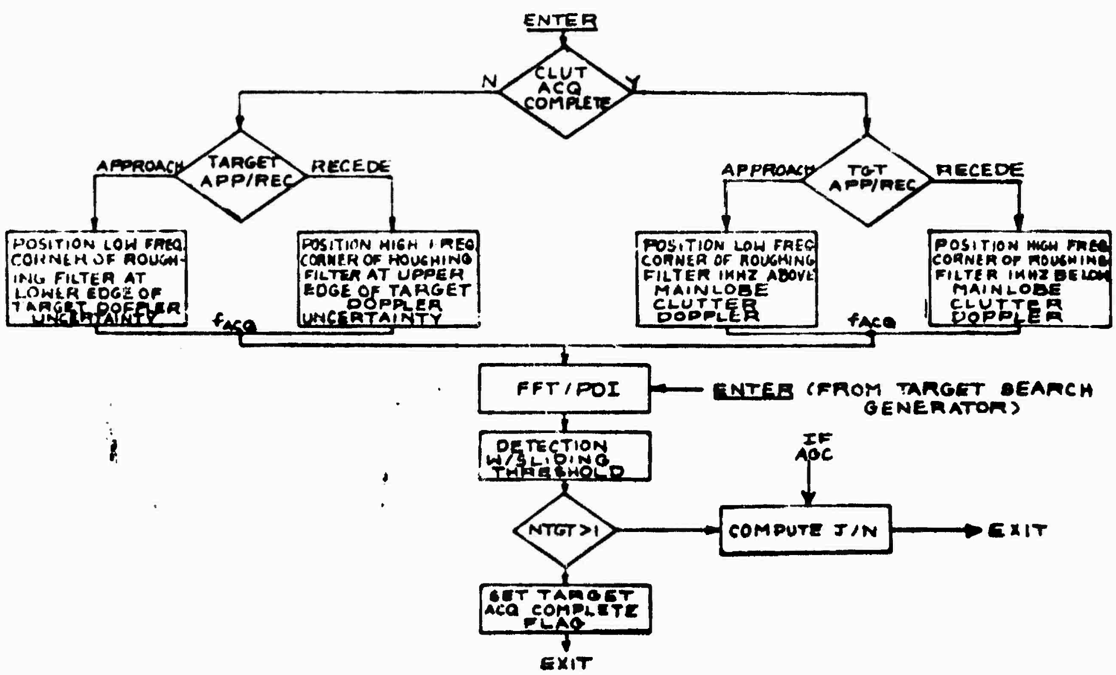


Figure 32 Radar Target Acquisition Mode Supervisor, Flow Diagram

The first part of the control sequence is concerned with determining the initial position of the acquisition roughing filter(s). Whether or not target acquisition must take place in a clutter environment is indicated by the clutter acquisition complete flag set by the Clutter Acquisition Mode Supervisor. If acquisition must take place in a clutter environment, the acquisition roughing filter is positioned such that its low frequency corner is approximately 1 KHz above the mainlobe clutter doppler for approaching targets and its high corner 1 KHz below the mainlobe clutter doppler for receding targets. For approaching targets in a non-clutter environment, the high frequency corner of the roughing filter is positioned at the upper edge of the target doppler uncertainty region.

Once the initial roughing filter position is determined, subsequent roughing filter positions and range-gate positions are determined by the target range/doppler search generator program module which is discussed in detail in subsection 4.2.6. The target Range/Doppler search generator generates a sequence of range gate and roughing filter positions such that the target range and doppler ambiguity region (designated as one search "Frame") is covered in cyclic manner. For example, if three range and doppler positions are required to cover the amibiguity region:

$$---, R_3F_3, R_1F_1, R_2F_1, R_3F_1, R_1F_2, R_2F_2, R_3F_3, R_1F_1, R_2F_1 ---, R_3F_3, ---$$

$$\longleftarrow \text{Frame \#N} \longrightarrow \longleftarrow \text{Frame \#N+1} \longrightarrow$$

For a specified range and doppler, a target acquisition sequence is performed that is identical to that performed to acquire mainlobe clutter. The only difference is that after detection with the sliding window threshold, the detected signals are not summed in range. If there has been one or more targets detected for this range-doppler position, the "target acquisition complete" flag is set and the executive proceeds to call up the Track Initiation Mode Supervisor. If there were no target detections, the Jammer-to-Noise ratio is computed. If it is determined that the missile is being jammed, the executive immediately designates the Track Mode Supervisor where Home-On-Jam track will be initiated. If the missile is not being jammed, the supervisor calls up the target Range/Doppler search generator which generates the range and doppler positions for the next part of the Range/Doppler ambiguity region. This process continues until target acquisition is accomplished.

Track Initiation Mode - The flow diagram for the Track Initation Radar Mode Supervisor logic is shown in Figure 33. The Track Initiation Mode performs the function of acquisition verification by reacquiring the target in a narrower-band roughing filter and a narrower doppler cell (i.e., the track "dwell" of 20 to 40 msec as opposed to 5 msec). The Track Initiation Mode also furnishes estimates of range error and doppler error for guidance data processing to initialize the range and doppler tracking filters.

The first step in the Track Initiation Mode is to center the narrow-band track mode roughing filter and the track mode range gate on the range and doppler "coordinates" found in target acquisition. The spectrum of the roughing filter is then examined using an FFT or a single dwell. Target detection for each range channel is again accomplished with a sliding threshold.

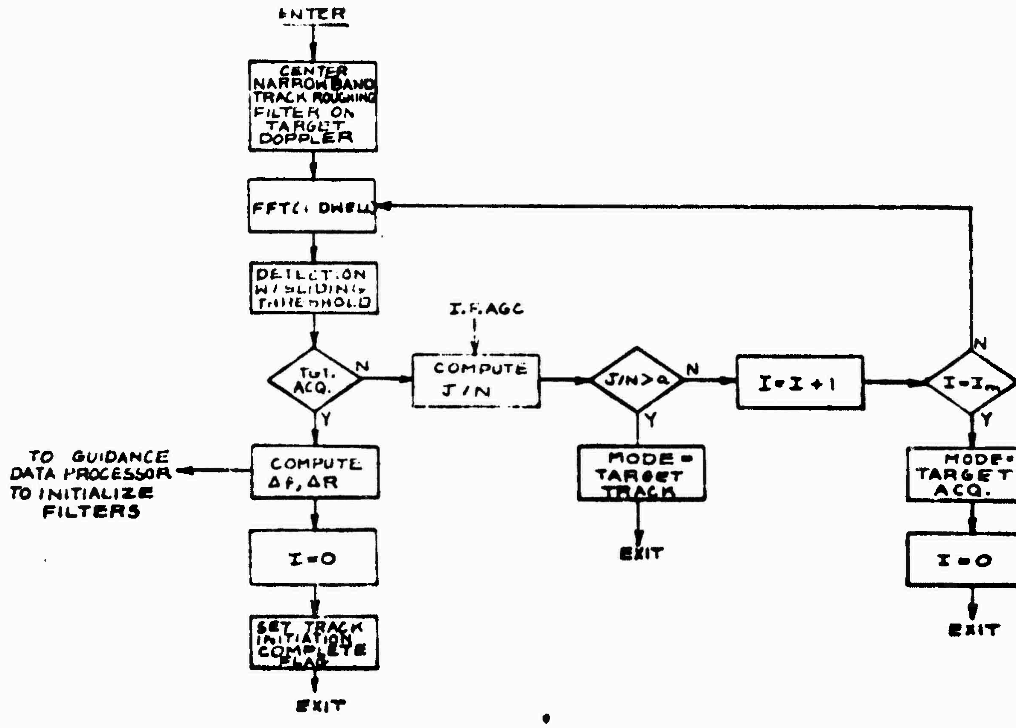


Figure 33 Radar Track Initiation Mode Supervisor, Flow Diagram

Note that a minimum of three range channels are involved in this process (one channel for an SA-CW missile), namely: the sum channel main tracking gate and the leading and lagging split gates which are used to determine range error. If target acquisition is accomplished, the range and doppler tracking errors are computed and a "track initiation complete" flag is set. If target acquisition is not accomplished, the Jammer-to-Noise ratio is computed. If the missile is found to be jammed, the executive designates the Target Track Mode Supervisor (for HOJ Tracking). If the missile was not being jammed, reacquisition is attempted for a specified number of times and then control reverts back to the Target Acquisition Mode Supervisor.

Target Track Mode - Flow diagrams illustrating the Target (Skin) Track Mode Supervisor logic are shown in Figure 34. The objective of this logic is to continuously provide target track information for guidance data processing on the target being engaged whether or not that target is being "skin-tracked" or is jamming. An explanation of the track mode of operation is given below.

The first step in the target track mode control sequence is to center the track roughing filters on the predicted target doppler and the track range gates on the predicted target range. Both these estimates are generated by the guidance data processing modules. Data is then collected for a single dwell (typically 20 to 40 msec as opposed to 5 msec in the acquisition mode). For an SA-CW radar sensor, data is collected for 3 channels (Σ, Δ_p, Δ_y). For the PD radar sensor, data is collected on two additional channels to allow a determination of range error using a split range gate on the monopulse sum channel. After computation of the channel spectrums using an FFT, a detection process is performed on the sum channel main tracking gate to determine if any skin track targets are present. If at least one target is present track processing continues in the skin-track mode and the next step is to determine if any of the detected targets lie outside the seeker antenna mainlobe.

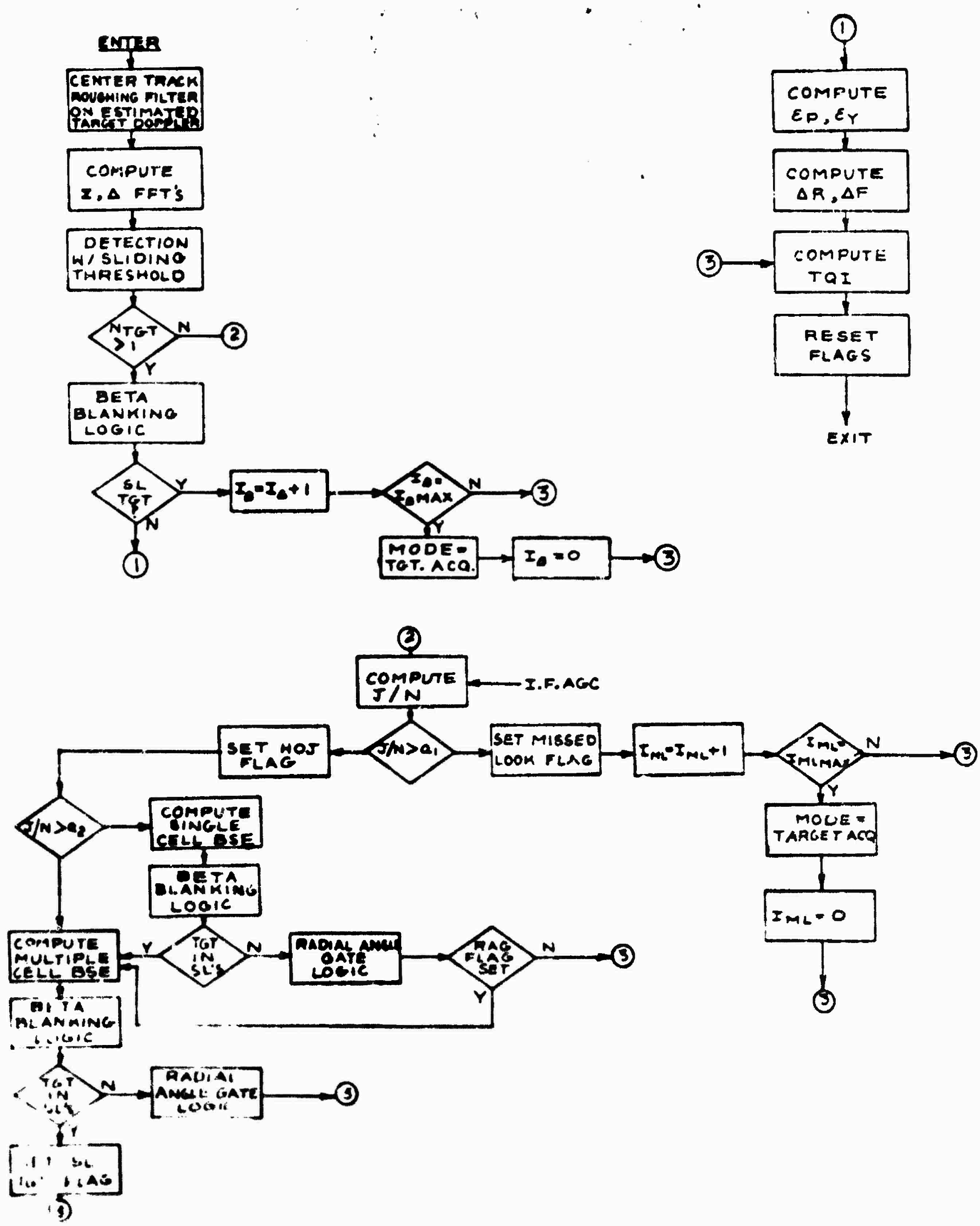


Figure 34 Radar Target Track Mode Supervisor, Flow Diagram

This is done by the Beta Blanking logic module which is described in Section 4.2.6. If no targets are found in the seeker mainlobe, the track quality indicator (TQI) program module is executed which in turn provides an output to the guidance data processing group indicating that no targets have been found. After a specified number of observations with no target showing up in the seeker mainlobe, the executive is flagged and mode control is switched back to the Target Acquisition Supervisor. For all targets passing the Beta blanking check, the pitch and yaw boresight errors and the range and doppler tracking errors are computed. These algorithms are discussed in detail in Section 4.2.6. The next step is to compute the TQI for the angle, range, and doppler errors. The TQI function indicates the signal-to-noise ratio for each error which is inversely proportional to the variance on the measured errors. i.e.

$$\sigma^2_{\epsilon p} = \frac{1}{S\epsilon/N_p} \qquad \sigma^2_{\Delta k} = \frac{1}{S_\epsilon/N_\epsilon}$$

The detailed TQI logic is explained in section 4.2.6.

Target selection logic is employed immediately after the TQI function to select a single target when multiple targets are present plus the logic necessary to handle range and/or doppler gate stealer types of deceptive ECM. The details of the target selection logic are described in Section 4.2.6.

If no targets were detected in the output of the Σ channel main tracking gate, the Σ channel Jammer-to-Noise (J/N) ratio is computed to determine if the seeker is being jammed. If the seeker is not being jammed, a missed look flag is set to tell the TQI function that no data is available on this dwell. After a specified number of missed looks in a row, the executive redesignates the Target Acquisition Supervisor. If jamming is detected, the Home-On-Jam (HOJ) flag is set to inform the TQI program module. If the jammer-to-noise ratio is large enough, the HOJ boresight errors are computed from a single FFT cell. This allows the radar sensor to take advantage of the difference in jammer spectrums to possibly obtain guidance data on a single jammer in a multiple jammer formation. If the J/N ratio is not adequate for this purpose, the boresight errors are computed for all FFT cells and averaged to obtain a single pitch and yaw error estimate. (The HOJ angle error algorithm is the same as that shown for skin track which are discussed in subsection 4.2.6). Beta blanking is used to eliminate jammers in the sidelobes. If the jamming target passes the beta blanking check, it is further examined by the radar angle gate logic (subsection 4.2.6). The objective of the radar angle gate logic is to force early resolution of a single blinking jammer in a multiple blinking jammer environment. After passing these checks the TQI is computed and the data used by the guidance data processing modules. Note, in the case where the HOJ boresight error (BSE) is computed for only a single FFT cell, if either beta blanking or radial angle gate logic resets it, the multiple cell BSE is computed in an endeavor to obtain useable guidance information.

Main-Lobe Clutter Track Mode - The objective of the Clutter Track Mode Supervisor is to keep track of the mainlobe clutter doppler when the target engagement is taking place under ground clutter conditions. The functional logic flow diagram for the Clutter Track Mode Supervisor is shown in Figure 35. This operation is the same as the Clutter Acquisition Mode. If mainlobe clutter is acquired, the mainlobe clutter tracking error, Δf_{MLC}, and a track quality indicator are computed and used by the guidance processing clutter doppler estimator module.

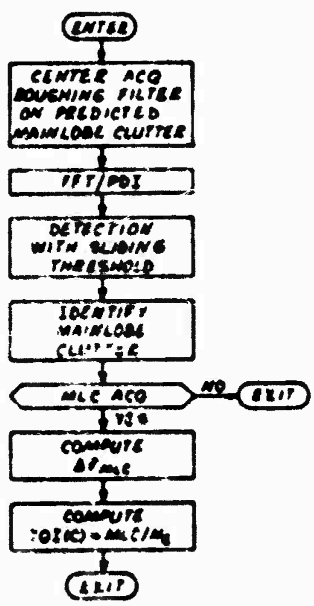


Figure 35 Radar Clutter Track Mode Supervisor, Flow Diagram

4.2.5 Signal Processing Design Requirements

The critical signal processing parameters for several candidate radar sensor systems are summarized in Table 18. These systems and their associated signal processing parameters were chosen as being practical and typical for the respective missile classes.

It must be pointed out that there are many other possible radar sensor system configurations and waveforms in addition to those chosen for the candidate systems in this study. Also, there is considerable latitude in the selection of the signal processing parameters for the candidate systems e.g. number of range gates, roughing filter size, etc. To determine what an optimum system configuration would be to meet a more specific/peculiar set of mission requirements would involve a detailed cost-performance tradeoff which is beyond the scope of this study. However, the results of this study do indicate the "cost" in terms of computer sizing/loading to achieve a system with a specific guidance capability. Since a modular design approach has been adhered to, the flexibility exits to configure and assess alternative systems.

All of the candidate systems defined employ digital signal processing for both the acquisition and track modes. Also note that all of the pulse doppler systems employ high PRFs for the dual purpose of eliminating doppler ambiguities and for maximizing the waveform duty cycle. This means that these systems are range ambiguous and cannot provide range accuracy better than the range designation accuracy of the AI radar. There is a technique to measure range in a range ambiguous system by transmitting multiple PRFs and then processing the results by a mathematical formulation known as the "Chinese Remainder Theorem". However, this technique has not been assumed in any of the candidate systems. A brief description of each candidate radar sensor system is given below.

TABLE 18

CANDIDATE RADAR SENSOR SYSTEMS

SIGNAL PROCESSING DESIGN REQUIREMENTS

	MISSILE CLASS				
	I	II	III	III	
SIGNAL PROCESSING PARAMETER	SA-CW	SA-PD (DEDICATED ILLUMINATOR)	(MIN) A-PD	(MAX) SA-PD (SHARED ILLUMINATOR)	COMMENTS
ACQUISITION					
Doppler Ambiguity (KHz)	20.0	20.0	20.0	20.0	
Range Ambiguity (μsec)	--	3.0	1.0	3.0	
PRF (KHz)	CW	270.0	1000.0	270.0	
No. of Range Gates	--	5	10	15	
Pulse Width (μsec)	--	0.2	0.1	0.2	
Roughing Filter Width (KHz)	10.0	10.0	10.0	20.0	
Observation Interval/Dwell(msec)	5.0	5.0	5.0	8.0	
Clutter Transient Time (msec)	--	--	--	1.6	
Effective Data Interval(msec)	5.0	5.0	5.0	6.4	
FFT Doppler Cell Width (Hz)	200.0	200.0	200.0	156.0	
Doppler Resolution	320.0(3)	320.0(3)	320.0(3)	250.0(3)	
FFT Samples/Dwell	64	64	64	128	
FFT Sample Rate (KHz)	12.8	12.8	12.8	20.0	
A-D Rate(Multiplexed) (KHz)	25.6	128.0	256.0	600.0	

TABLE 18 (Continued)

SIGNAL PROCESSING PARAMETER	MISSILE CLASS I SA-CW	II SA-PD (DEDICATED ILLUMINATOR)	III (MIN) A-PD	III (MAX) SA-PD (SHARED ILLUMINATOR)	COMMENTS
No. of Dwells/Sequence	10	10	10	1	
No. of Sequences/Frame	2	6	2	1	
Frame Time (msec)	100.0	300.0	100.0	100.0	
No. of Frames/sec	10.0	3.3	10.0	10.0	
No. of Range-Doppler Cells/Frame	100	1500	1000	1500	
No. of Range-Doppler Cells/Second	1000	5000	10000	15000	
P_{FA} (Single Cell)	0.69×10^{-3}	0.385×10^{-3}	0.69×10^{-4}	0.46×10^{-4}	
P_{DET} (Single Frame)	0.26	0.59	0.26	0.26	
Req'd SNR (Ref to 100Hz-BW) (db)	5.6	14.1	6.8	14.2	Based on: P_{CWM}=0.95 in 1 sec; W/T =1 sec; Includes 3db misc loss and Swerling I target
TRACK					
No. of Monopulse Channels	3	5 (2)	5 (2)	5 (2)	
Roughing Filter Width (KHz)	1.0	1.0	1.0	20.0	
Effective Data Interval (msec)	20.0	20.0	20.0	6.4	
FFT Doppler Cell Width (Hz)	50.0	50.0	50.0	156.0	

TABLE 18 (Continued)

SIGNAL PROCESSING PARAMETER	MISSILE CLASS I SA-CW	II SA-PD (DEDICATED ILLUMINATOR)	III (MIN) A-PD	III (MAX) SA-PD (SHARED ILLUMINATOR)	COMMENTS
Doppler Resolution $\sqrt{\sigma^2_w}$ (Hz)	80.0	80.0	80.0	250.0	
FFT Samples/Dwell	64.0	64.0	64.0	128.0	
FFT Sample Rate (KHz)	3.2	3.2	3.2	20.0	
A-D Rate (Multiplexed) (KHz)	19.2	32.0	32.0	200.0	
Frame Time (msec)	20.0	20.0	20.0	100.0	
No. of Frames/sec (Track Data Rate)(Hz)	44 (1)	44 (1)	44 (1)	10.0	
No. of Range-Doppler Cells/Frame	150	250	250	500	
No. of Range-Doppler Cells/sec	2200	11000	11000	5000	

Note (1) Results from two 50 msec track monitor dwells per second

(2) Includes two split gate channels for range error

(3) Includes $\cos^2$ burst time weighting

4.2.5.1 Class I Missiles (SA-CW Sensor)

The significant digital signal processing design parameters for the Class I, SA-CW, Radar Sensor shown in Table 18. The target ambiguity region for the SA-CW sensor is completely searched with two roughing filter positions. For each roughing filter position, ten 5 msec dwells are post-detection integrated to improve signal-to-noise ratio. This results in a frame time of 100 msec (50 msec/sequence x 2 sequences/frame with one sequence = 10 dwells x 5 msec/dwell = 50 msec).

In the track mode the roughing filter bandwidth is reduced to 1.0 KHz and data for three monopulse channels is collected over a 20.0 msec dwell. A 40 msec dwell which results in 25 Hz doppler cells also appears to be reasonable based on the results shown in section 5.3 for miss distance as a function of time delay. The effective time delay would be approximately 30 msec for the 40 msec dwell (20 msec from center of dwell to completion plus 10 msec for signal processing).

4.2.5.2 Class II Missiles (SA-PD Sensor)

There are actually two types of SA-PD radar sensor systems shown in Table 18. The Class II missile system has a dedicated illuminator, whereas the Class III (max) system has a shared illuminator. Note that the Class II system employs both a range and doppler search to cover the acquisition ambiguity. Three range gate and two roughing filter positions are used to cover the ambiguity region. The data collected at each range-doppler position in the ambiguity region consists of ten 5 msec dwells (one data collection sequence) which are post-detection-integrated (PDI) to improve the signal-to-noise ratio. One frame consists of six data collection sequences and takes 300 msec (50 msec per sequence = 5 msec/dwell * 10 dwells).

In the track mode, the roughing filter bandwidth is reduced to 1.0 KHz and the same 64-point FFT is used for a 20 msec dwell which yields 20 50 Hz wide doppler cells covering the 1.0 KHz bandwidth.

4.2.5.3 Class III Missiles (A-PD Sensor)

This system differs from the Class II SA-PD system not only in that it is an active system, (on-board PD transmitter), but also in that the complete range ambiguity (1.0 μsec) is searched on each dwell. The complete range doppler ambiguity region is searched with only two roughing filter positions. As was the case with the SA-CW and SA-PD systems, the data collected for each range-doppler position is ten 5 msec dwells which are post-detection-integrated to improve signal-to-noise ratio. One frame for this system consists of two data collection sequences and takes 100 msec.

The track mode parameters are identical to the Class II SA-PD system.

4.2.5.4 Class III Max. Missiles (SA-PD Sensors)

This system is included as representative of a more advanced Class III system. The primary difference between this system and the other three candidates is that it is assumed that the AI radar must support many simultaneous engagements resulting in short dwell times at a relatively low data rate.

Because of the assumed limited AI radar dwell time, this system searches the complete range doppler ambiguity space on each dwell. It is not possible to employ post-detection integration because of the limited dwell time. Therefore, detection decisions are made each dwell (a single dwell constitutes a frame in this system). Note also that 1.6 msec out of the total dwell time is used to allow filter transients due to large amplitude out of band clutter to die out. This yields the effective dwell time of 6.4 msec.

In the track mode, the AI radar dwell time does not change. The track signal processing parameters are unchanged from acquisition with the exception that only five channels of data are processed instead of fifteen. Note that the capability of processing 15 channels exists since it is required in the acquisition mode. Therefore, a more sophisticated ECCM logic than that described in subsection 4.2.6 could be implemented at little additional computer cost.

4.2.5 Signal Processing Program Modules

In this subsection, radar signal processing algorithms and their corresponding program modules are defined for selection according to the type of radar sensor used in a given missile. A total of twenty one unique program modules have been defined for radar sensor control and signal processing. Seven of these modules (SP-1 through SP-7) are mode supervisor programs described in subsection 4.2.4. The remaining fourteen modules (SP-8 through SP-21) are described in the following paragraphs. A complete listing is given in the computer requirements summary at the end of this subsection.

Compute Jammer to Noise Ratio (J/N) -

A functional flow diagram of the jammer-to-noise ratio computation logic, Module SP-8 is shown in Figure 36. The logic is general enough to handle single or multiple range gates. This computation depends on knowing the gain of the receiver so that the noise level due to thermal noise can be computed. The sum channel IF AGC level is encoded on each dwell for this purpose. Note, that the magnitude of the return in each range-doppler cell is computed in the FFT/PDI program module.

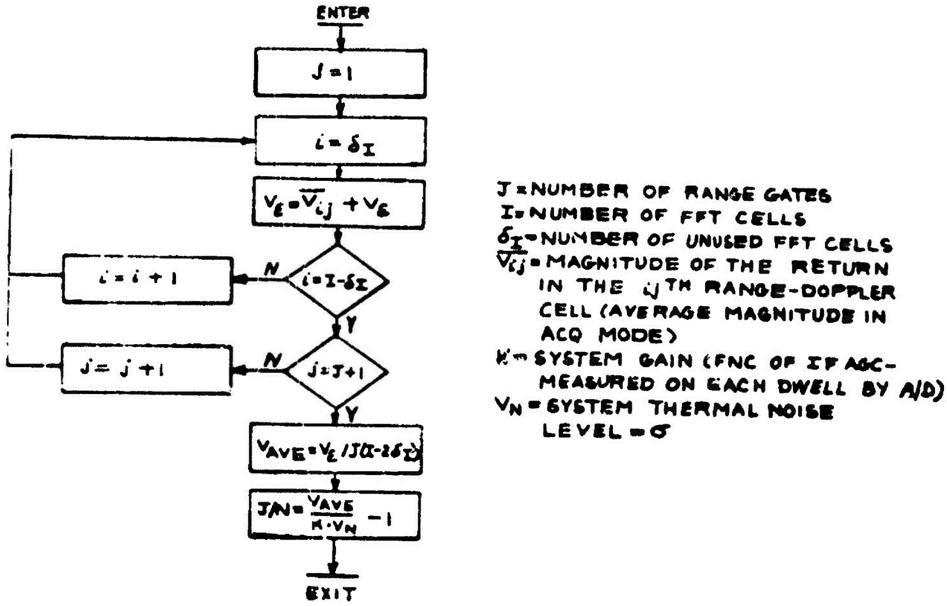
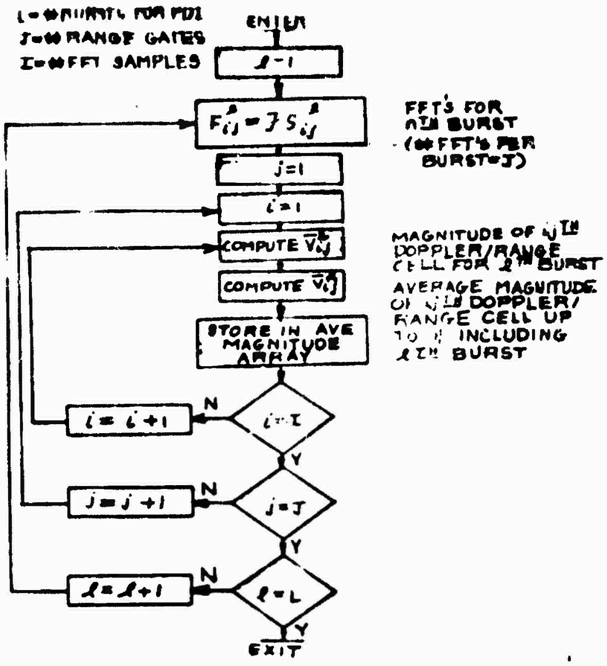


Figure 36. Computer Jammer to Noise Ratio (J/N) Program Module SP-8, Flow Diagram

Post-Detection Integration -

A functional flow diagram of the post-detection integration logic, Module SP-9, is shown in Figure 37. Also shown in this figure are the constants for the different signal processor configurations which are discussed in subsection 4.2.3. For each of the "L" bursts in an acquisition sequence, the magnitude of each complex doppler cell is computed range gate by range gate and stored in a "average value" array of dimensions IxJ. The time available for one pass (L=1) through the PDI logic is approximately equal to the data collection interval (e.g. 5 msec) at which time the next FFT output is available.



CONSTANTS FOR THE DIFFERENT SIGNAL PROCESSOR CONFIGURATIONS

	L	J	I
SA-CW	10	1	64
S-PD/DI	10	5	64
A-PD	10	10	64
SA-PD	1	15	128

EQUATIONS FOR MAGNITUDE/AVE MAGNITUDE

$$V_{ij}^{\ell} = MAX(|F_{ij_I}^{\ell}|, |F_{ij_Q}^{\ell}|) + 1/4\ MIN(|F_{ij_I}^{\ell}|, |F_{ij_Q}^{\ell}|)$$

$$\bar{V}_{ij}^{\ell} = \bar{V}_{ij}^{\ell-1} + (V_{ij}^{\ell} - \bar{V}_{ij}^{\ell-1})/\ell$$

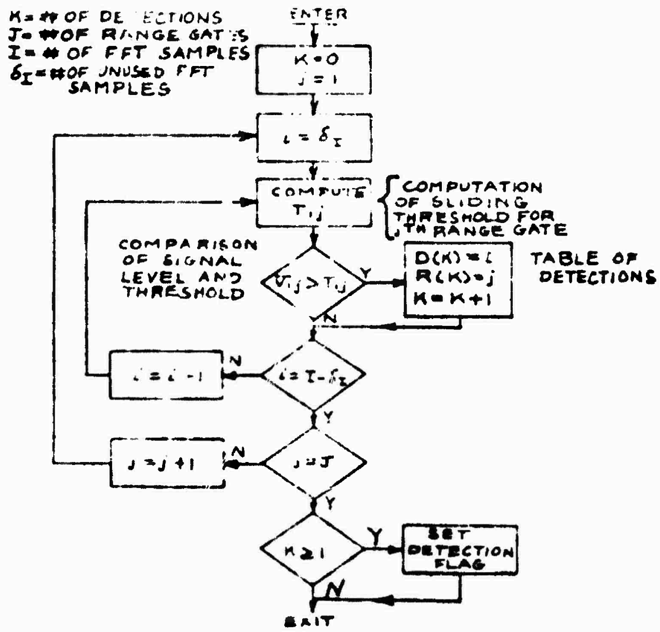
FFT OPERATION INCLUDES:

- STORAGE OF DATA TO ALLOW SEQUENTIAL OR PARALLEL PROCESSING
- CORNER TURNING LOGIC
- COS^2 BURST TIME WEIGHTING

Figure 37. FFT Post Detection Integration Program Module SP-9, Flow Diagram

Detection with Sliding Threshold -

The logic flow diagram for detection with a multi-channel sliding threshold, Module SP-10, is shown in Figure 38. The output from this routine is a table of detections of targets that crossed the threshold. The detection with the sliding threshold is done range gate by range gate. More complicated systems could involve sliding a range-doppler window across the range doppler matrix. However, for the purpose of this study the simpler approach was taken. Note, that this same routine is also used to find detections in the track mode where only a single Σ channel is examined. (module SP-11).



CONSTANTS FOR THE DIFFERENT SIGNAL PROCESSOR CONFIGURATIONS

	J	I	δ_I	N_M
SA-CW	1	64	7	9
SA-PD	15	128	14	9
SA-PD/D.I.	5	64	7	9
A-PD	10	64	7	9

NOTE IN TRACK MODE $J = 1$, $I = 64$, $\delta_I = 7$, $N_M = 9$ FOR ALL SEEKER TYPES

COMPUTATION OF THRESHOLD T_{ij}

$$T_{ij} = \frac{1}{N_M} \cdot A_{TH} \cdot \sum_{\alpha = i-\left(\frac{N_M-1}{2}\right)}^{i+\left(\frac{N_M-1}{2}\right)} V_{\alpha j}$$

WHERE N_M = # FFT SAMPLES IN SLIDING WINDOW
A_{TH} = THRESHOLD SCALING FACTOR

Figure 38. Detection with Sliding Threshold Program Modules SP-10 and SP-11, Flow Diagram

Target Search Generator (Wideband Doppler) -

The functional flow diagram for the target doppler search generator logic, Module SP-12, is shown in Figure 39. This search generator covers the complete range of possible target dopplers ($f_{D_{MIN}}$ to $f_{D_{MAX}}$) looking for the injected video. The number of dwells for any one roughing filter position depends on whether or not this is the first try at acquisition. This search generator can also be used as a last resort to sweep out the entire doppler spectrum if attempts at acquisition in a limited region fail.

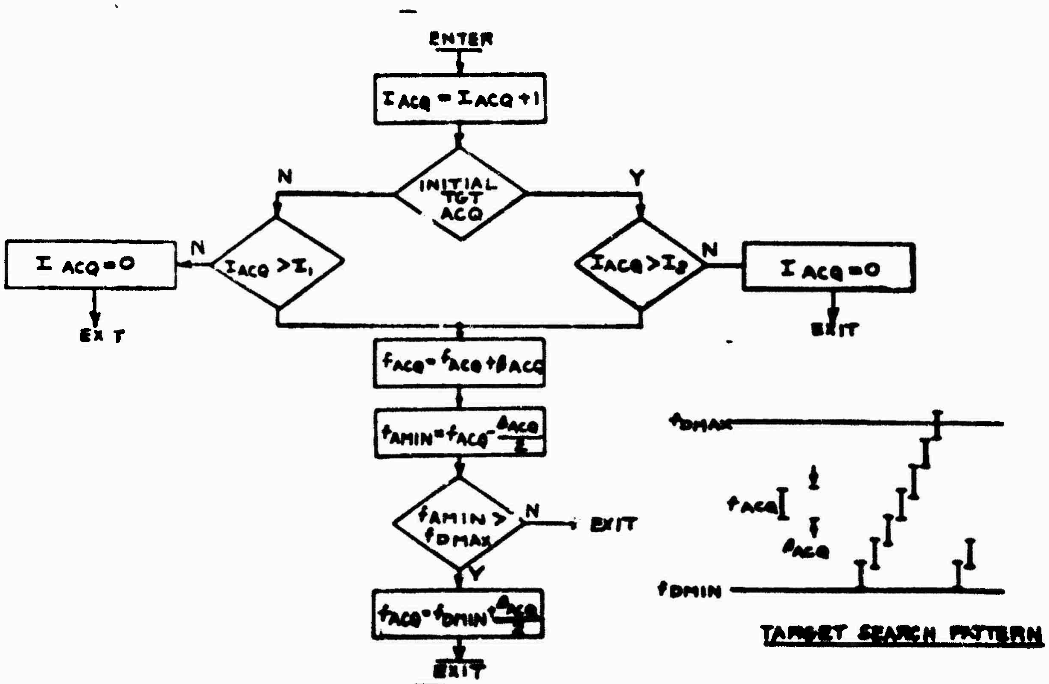


Figure 39. Target Search Generator, (Wideband Doppler) Program Module SP-12, Flow Diagram

Target Search Generator (Range and Doppler) -

The functional flow diagram for the target range and doppler search generator logic, Module SP-13, is shown in Figure 40. Also shown in this figure is a possible range doppler search pattern where 3 range gate bank and 3 doppler roughing filter positions are searched. This logic is designed to search all roughing filter positions for a given range gate bank setting. If the target is approaching, the roughing filter center frequency is incremented in steps to search the doppler region above the mainlobe clutter. If the target is receding, the roughing filter center frequency is decremented in steps to search the doppler region below mainlobe clutter.

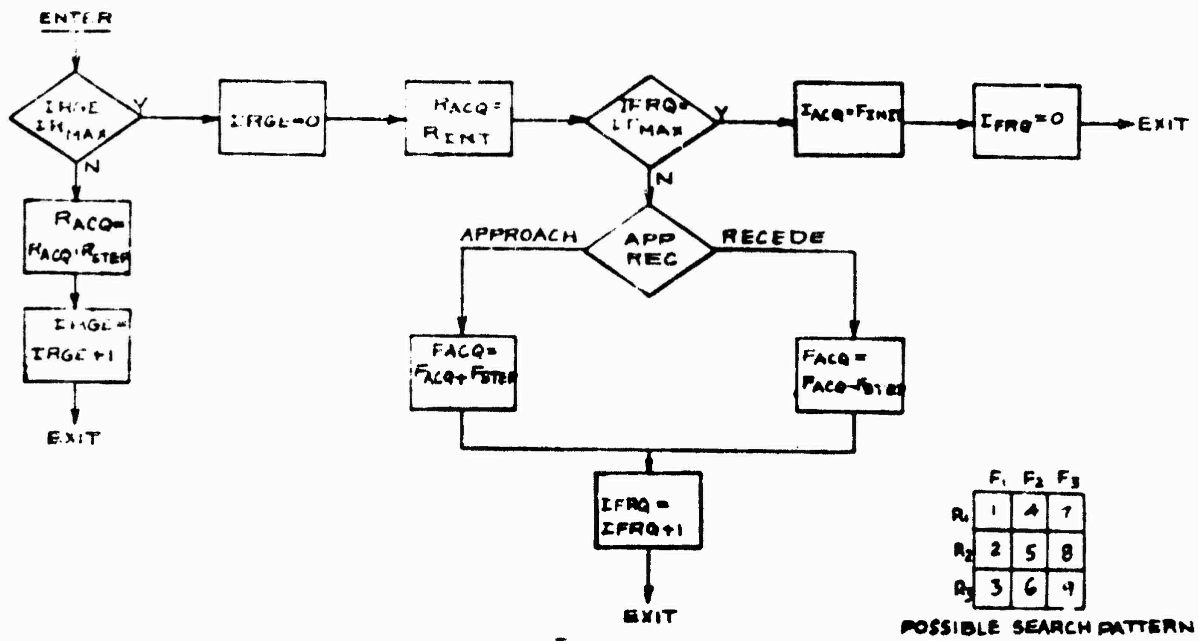


Figure 40. Target Search Generator (Range and Doppler) Program Module SP-13, Flow Diagram

Compute Angle, Range, and Doppler Errors -

The angle, range, and doppler error program modules SP-14, SP-15, and SP-16, are defined in Figure 41. In the case of a jamming target, only the angle error is computed. For low jammer-to-noise ratios, the pitch and yaw angle errors are computed for each doppler cell in the spectrum analyzer output array and averaged to reduce thermal noise errors. For strong jamming targets, the pitch and yaw angle errors are computed for a single doppler cell to take advantage of the inherent non-uniformity of the jammer noise spectrums.

ANGLE $\quad \epsilon_P = k_{BSE} \left[\frac{S_{\Sigma I}(i_T) S_{PI}(i_T) S_{\Sigma Q}(i_T) S_{PQ}(i_T) \cos\theta_{CP} + S_{\Sigma I}(i_T) S_{PQ}(i_T) - S_{\Sigma Q}(i_T) S_{PI}(i_T) \sin\theta_{CP}}{S_{\Sigma I}^2(i_T) + S_{\Sigma Q}^2(i_T)} \right]$

RANGE $\quad \Delta R = k_R \left[\frac{\sqcup - \boxminus}{\sqcup + \boxminus} \right]$

$$\sqcup = \mathrm{MAX}\left[|S_{\Sigma^+ I}(i_T)|, |S_{\Sigma^+ Q}(i_T)|\right] + 1/4\ \mathrm{MIN}\left[|S_{\Sigma^+ I}(i_T)|, |S_{\Sigma^+ Q}(i_T)|\right]$$

$$\boxminus = \mathrm{MAX}\left[|S_{\Sigma^- I}(i_T)|, |S_{\Sigma^- Q}(i_T)|\right] + 1/4\ \mathrm{MIN}\left[|S_{\Sigma^- I}(i_T)|, |S_{\Sigma^- Q}(i_T)|\right]$$

DOPPLER $\quad \Delta F = k_F \left[\frac{AC + BD}{C^2 + D^2} \right]$

$$A = S_{\Sigma I}(i_0 - 1) - S_{\Sigma I}(i_0 + 1)$$
$$B = S_{\Sigma Q}(i_0 - 1) - S_{\Sigma Q}(i_0 + 1)$$
$$C = S_{\Sigma I}(i_0 - 1) - 2S_{\Sigma I}(i_0) + S_{\Sigma I}(i_0 + 1)$$
$$D = S_{\Sigma Q}(i_0 - 1) - 2S_{\Sigma Q}(i_0) + S_{\Sigma Q}(i_0 + 1)$$

NOTE: IF $|i_T - i_0| \geq 2$

$\Delta F = (i_T - i_0) \cdot \beta_{CELL}$

WHERE

θ_{CP} = PHASE DIFFERENCE BETWEEN SUM & PITCH CHANNELS - PREDETERMINED

i_T = NUMBER OF DOPPLER CELL CONTAINING THE TARGET

i_0 = REFERENCE DOPPLER CELL - DESIGNATED DOPPLER TRACKING POINT

k_{BSE}, k_R, k_F = ERROR SLOPE CONSTANTS

$S_{\Sigma I}(i)$, $S_{\Sigma Q}(i)$ = COMPLEX COMPONENTS OF SUM CHANNEL FFT's i^{TH} DOPPLER CELL

$S_{\Sigma^+ I}(i)$, $S_{\Sigma^+ Q}(i)$ = COMPLEX COMPONENTS OF SUM CHANNEL "EARLY GATE" FFT's i^{TH} DOPPLER CELL

$S_{\Sigma^- I}(i)$, $S_{\Sigma^- Q}(i)$ = COMPLEX COMPONENTS OF SUM CHANNEL "LATE GATE" FFT's i^{TH} DOPPLER CELL

$S_{PI}(i)$, $S_{PQ}(i)$ = COMPLEX COMPONENTS OF PITCH CHANNEL FFT's i^{TH} DOPPLER CELL (SIMILAR FOR YAW)

Figure 41 Angle, Range and Doppler Track Error Program Modules SP-14, SP-15 and SP-16

With skin track targets, the angle, range and doppler errors are computed for each detected target in the track narrow-band roughing filter. It is not necessary for a target to be centered in the doppler array to compute its errors. The angle errors are computed using the spectrum analyzer complex outputs of the sum, pitch, and yaw channels. The range error is computed using similar outputs of the split range gates (early and late sum channels). Doppler error is computed relative to the center cell by using the center three cells in the array. If the target is not in the center cell, doppler error is measured as the number of cells different from the center cell times the single cell doppler width.

Beta Blanking Logic -

The functional flow diagram for the beta blanking logic, Module SP-17, is shown in Figure 42. The purpose of the beta blanking logic is to determine if any of the detected targets are outside the mainlobe of the missile seeker antenna. This is accomplished by comparing the magnitude of the targets (polar) difference signal to the targets sum signal. Whenever the magnitude of the targets difference signal is greater than the magnitude of the targets sum signal, the target is declared to be in the sidelobes.

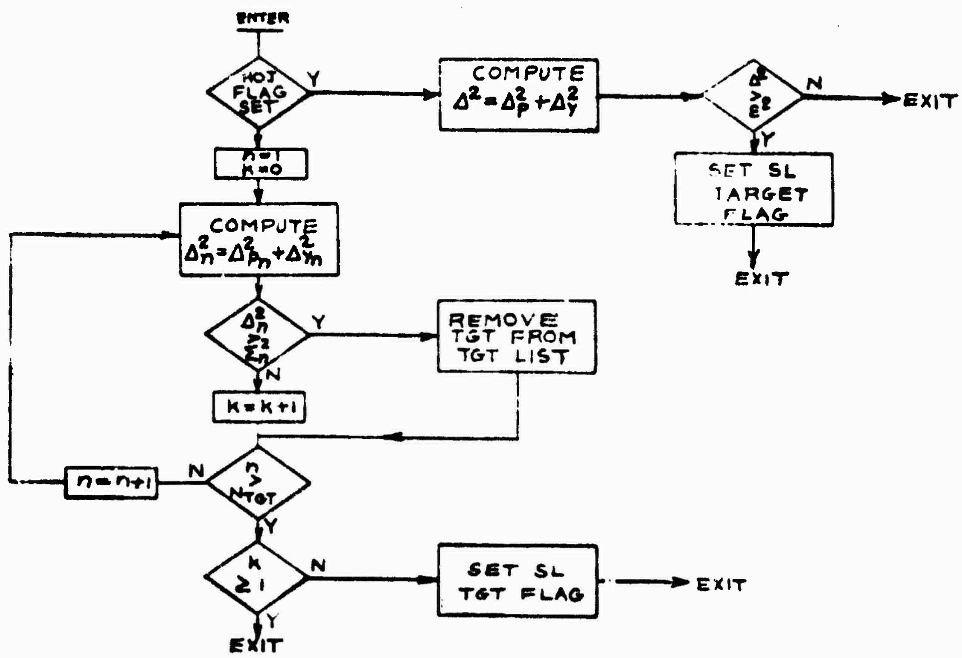


Figure 42. Beta Blanking Logic Program Module SP-17, Flow Diagram

This is illustrated in Figure 43 for a typical monopulse antenna pattern. As can be seen the method is not foolproof, however, the probability of rejecting sidelobe targets by this method can be quite high (measurements have indicated up to 90% rejection or larger can be achieved using beta blanking). Targets that do not pass the beta blanking check are removed from the "list" of detected targets and are not processed further. If no targets pass the beta blanking test, a "SL Target" flag is raised.

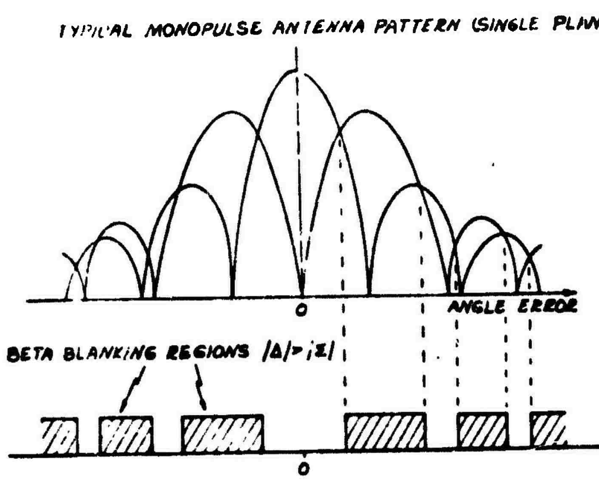


Figure 43. Typical Monopulse Antenna Patterns Showing "Beta Blanking" Regions

Radial Angle Gate Logic -

The functional logic flow diagram, for the radial angle gate logic, Module SP-19 is shown in Figure 44. The objective of this module is to obtain discrimination of a single blinking jammer in a multiple blinking jammer formation. This is accomplished by comparing the polar boresight error ($\varepsilon_P^2 + \varepsilon_Y^2$) to a computed threshold level. The computed threshold level increases with time to a specified maximum if no observations pass the threshold check and decreases with time to a specified minimum if all observations pass the threshold check. A "Rag Blank" flag is raised for use in the TQI function when the observation exceeds the threshold level.

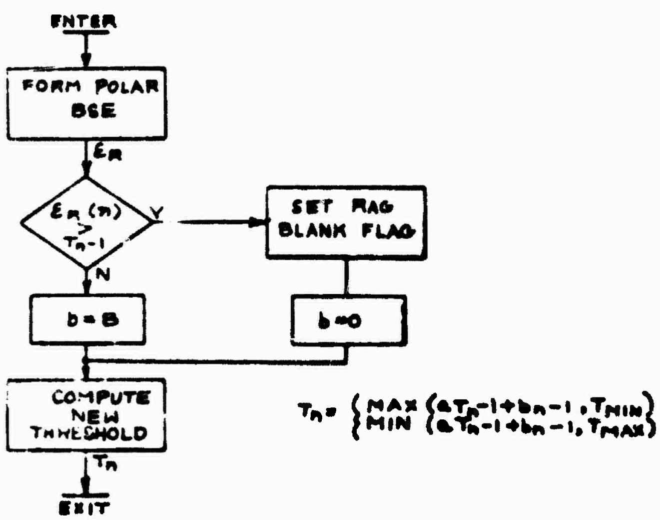


Figure 44. Radial Angle Gate Logic Program Module SP-19, Flow Diagram

Identify Main Lobe Clutter (MLC) -

A functional flow diagram for the identify mainlobe clutter logic, Module, SP-20, is shown in Figure 45. The signals that have passed the sliding threshold (i.e. detected signals) are first averaged in range for each doppler cell. Next, the doppler cell average magnitude is scanned to find the largest signal which is designed mainlobe clutter.

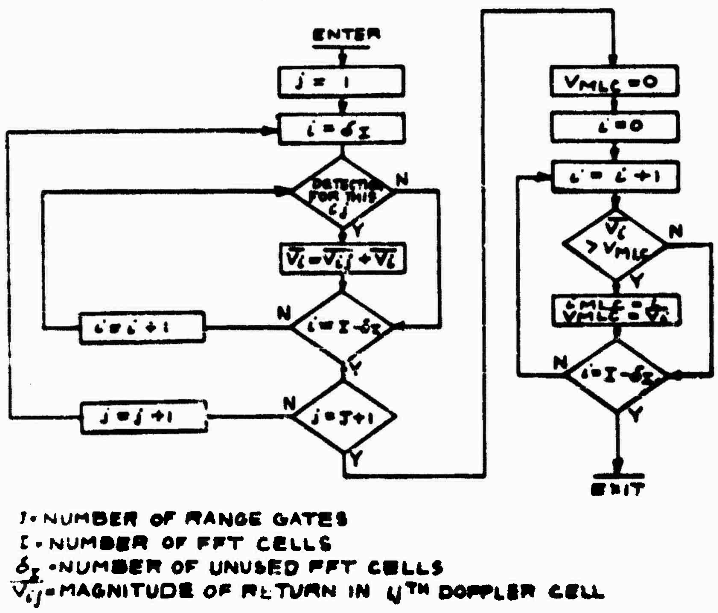


Figure 45. Identify Mainlobe Clutter (MLC) Program Module SP-20, Flow Diagram

Track Quality Indicator (TQI) Logic -

A functional flow diagram for the TQI logic program, Module SP-18, is shown in Figure 46. The objective of the TQI (Track Quality Indicator) function is to indicate to the guidance data processor the quality of the information passed along for each of the detected targets. For jamming targets, the TQI is computed as the pitch and yaw jammer-to-(thermal) noise ratio. If the jammer is in the antenna sidelobes or does not pass the radial angle gate the TQI is set equal to zero. For skin track targets the TQI is set equal to zero if there is a missed look or the missile is found to be in a clutter situation (target doppler too close to the mainlobe clutter doppler, hence the clutter warning flag is raised by the guidance data processor based on its estimates of the target and mainlobe clutter dopplers). Otherwise, the TQI's are computed for angle, range and doppler errors.

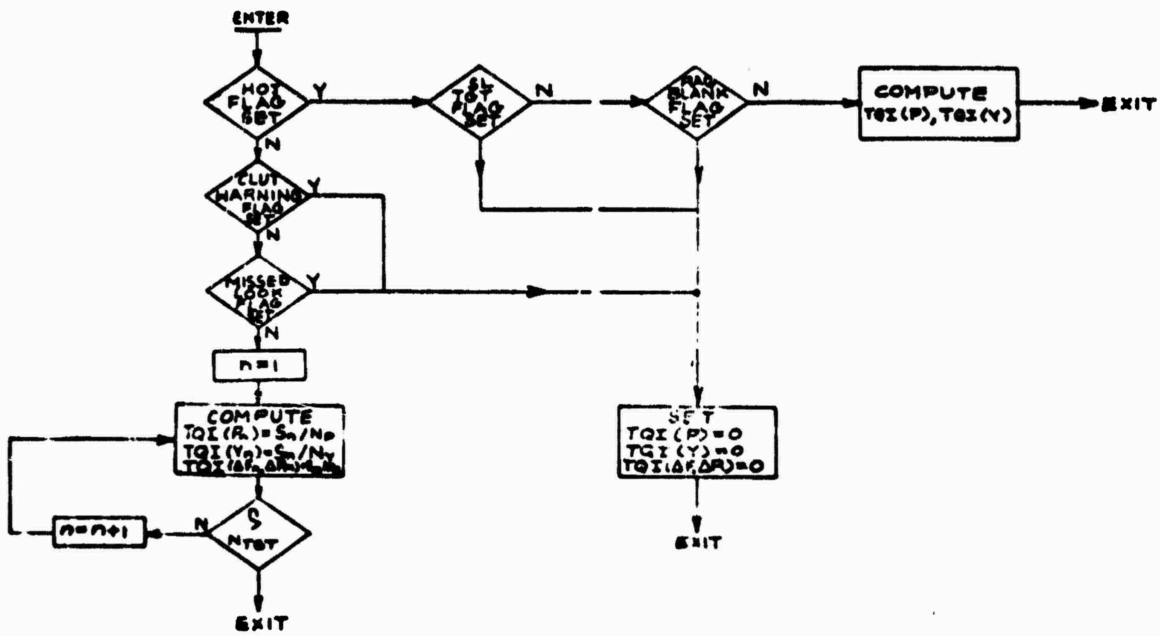


Figure 46. Track Quality Indicator (TQI) Logic Program Module SP-16, Flow Diagram

Target Selection Logic -

The functional flow diagram for the target selection logic, Module SP-21 is shown in Figure 47.

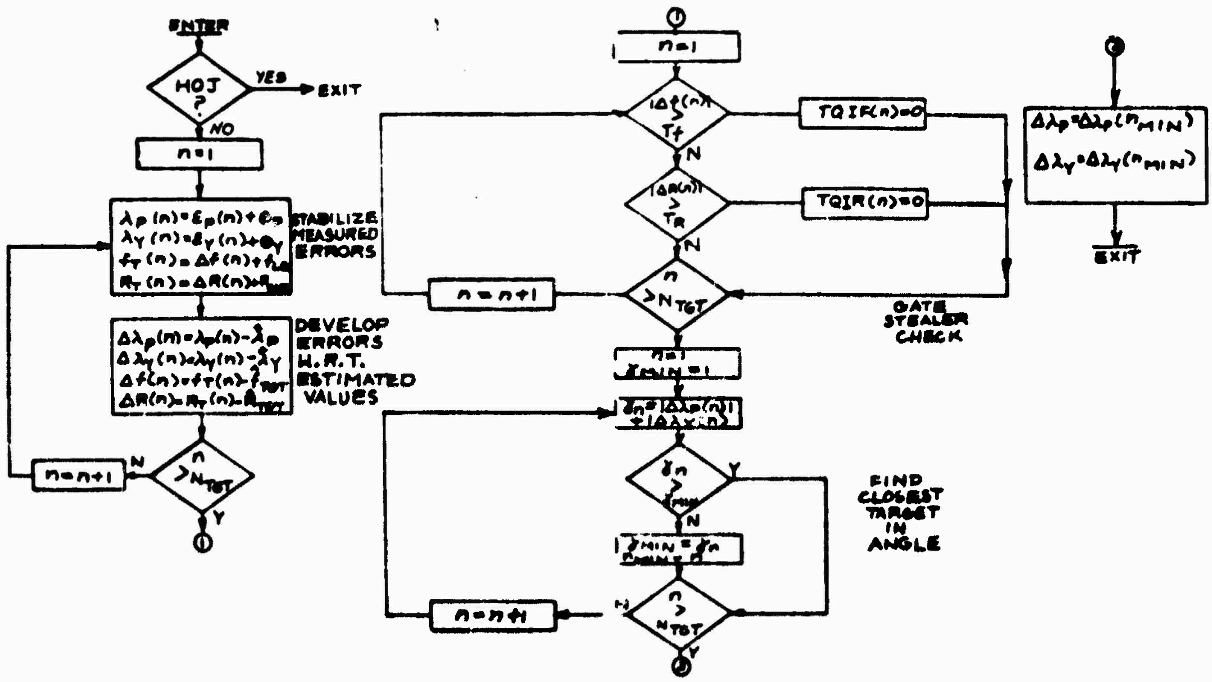


Figure 47 Target Selection Logic Program Module SP-21, Flow Diagram

The input to the target selection logic is a list of all targets detected by the digital signal processor in the target track mode. This list is structured as follows:

$\Sigma p(1), TQIP(1)$

$\Sigma y(1), TQIY(1)$

$\Delta F(1), TQIF(1)$

$\Delta R(1), TQIR(1)$

. .

. .

. .

. .

$\Sigma p(N_{TGT}), TQIP(N_{TGT})$

$\Sigma y(N_{TGT}), TQIY(N_{TGT})$

$\Delta F(N_{TGT}), TQIF(N_{TGT})$

$\Delta R(N_{TGT}), TQIR(N_{TGT})$

The first action taken by the target selection logic is to stabilize the measured errors. The pitch and yaw boresight errors are converted to line-of-sight angles and the range and doppler errors to target range and target doppler. The next step, which can be combined with the first step, is to compare the stabilized measurements with the current Kalman filter estimates to develop a set of "stabilized" errors. A check is then made to locate any range and/or doppler gate stealers by comparing the stabilized range and doppler errors to a threshold value. Exceeding the threshold implies a larger range or doppler motion than could be expected by target-missile physical motion. The action taken is to set the track quality indicator for the excessive range or doppler error to zero. This means that if this target is selected for angle track on the basis of the next check for minimum angle error, the range or doppler measuremnet will not be used and the estimate coasted until the next legitimate measurement.

The target that is passed along to the Kalman filter is that target closest in LOS angle to the present LOS estimate. This target is found by computing the sum of the absolute values of the pitch and yaw LOS errors and finding the minimum value in the target list. Since only one target is identified by the target track logic in the HOJ mode, no target selection logic is required.

Computer Requirements Summary -

Digital signal processing computer requirements for the candidate Class I, II, and III missile radar sensors identified in subsection 4.2.1 are summarized in Table 19, 20 and 21 respectively.

These tables list the digital signal processing programs modules previously defined, and give the corresponding add/subtract, multiply/divide and load/store operations for the worst-case/critical path through each module. Similarly, program modules used in the worst-case/critical path of each radar mode are checked-off, and the corresponding instruction counts for both functional program modules and supporting utility routines are given as totals together with instruction mixes, for determining worst-case throughput in Section 6. (See individual mode supervisor flow charts for full complement of program modules per model. Total operation counts are also given without the FFT, and without both FFT and PDI functions, for system design flexibility.

Program memory sizes are given to cover the total instructions required for each program module with an additional 30% included to account for subroutine linkages and other miscellaneous overhead operations. Operations counts are increased by 30% when converting these to Kops in Section 6.

Data memory requirements are driven by the Clutter Acquisition Mode, due to the relatively rapid sequence of short radar dwell periods, multiple range gates (for SA-PD and A-PD sensors), and the need to store arrays of data acquired/processed during one dwell, for further processing in a subsequent dwell interval. Data memory totals given in each of the following tables make provision for 3 complex and 2 real data arrays, with additional space for a table of 12 target detections and scratch-pad storage. Array sizes are determined by the number and type of data points/doppler cells (N), (2 memory locations for each complex data point/doppler cell), and the number of range gates (R), i.e.:

	ARRAY STORAGE REQTS.		
ARRAY TYPE	CW	SA-PD	A-PD
Complex (Nx2xR)	128	640	1280
Real (N x R)	64	320	640

TABLE 19 RADAR SIGNAL PROCESSING COMPUTER REQUIREMENTS

CW RADAR SENSOR FOR CLASS I MISSILE

MODULE	NAME	ADD/ SUB	MPY/ DIVIDE	LOAD/ STORE	UTILITIES	RADAR MODE: PRE-LAUNCH	RADAR MODE: LAUNCH	RADAR MODE: CLUT. ACQ.	RADAR MODE: TGT. ACQ.	RADAR MODE: TRACK INIT.	RADAR MODE: TGT. TRACK	RADAR MODE: CLUT. TRACK	MEMORY: *PROGRAM	MEMORY: DATA RAM	MEMORY: ROM
SP-1	Pre-Launch Mode Supervisor	2	0	15	(10-U21,5)	X							33		
SP-2	Launch Mode Supervisor	8	5	31			X						57		
SP-3	Clutter Acq. Mode Supervisor	4	0	23	(10-U21,5)			X					46		
SP-4	Track Acq. Mode Supervisor	2	0	16	(10-U21,5)				X				34		
SP-5	Track Init. Mode Supervisor	4	0	26	(1-U21,5)					X			50		
SP-6	Target Track Mode Supervisor	7	0	40	(3-U21,3,5)						X		77		
SP-7	Clutter Track Mode Supervisor	4	1	19	(10-U21,5)							X	42		
SP-8	Compute J/N	51	3	204				X	X	X			32		
SP-9	Post-Detection Integration	2560	640	8320		X		X	X			X	65		
SP-10	DET2W/Sliding Threshold (Mult.Ch)														
SP-11	Det.W/Sliding Threshold (1 Ch)	450	50	400		X		X	X	X	X		55		

TABLE 15 (Continued)

MODULE	NAME	ADD/ SUB	MPY/ DIVIDE	LOAD/ STORE	UTILITIES	PRE-LAUNCH	LAUNCH	RADAR MODE CLUT. ACQ.	TGT. ACQ.	TRACK INIT.	TGT. TRACK	CLUT. TRACK	MEMORY *PROGRAM	DATA RAM	ROM
SP-12	Target Search Gen-[illegible] Doppler	[illegible]	0	17									30		
SP-13	Target Search Gen-[illegible] Doppler	6	3	10					X				[illegible]		
SP-14	Compute Angle Errors	8	20	24							X		68		
SP-15	Compute Doppler Error	16	6	25						X	X		53		
SP-16	Compute Range Error	6	2	20									47		
SP-17	Data Blanking Logic	13	8	63	(4-SORT)						X		42		
SP-18	Compute TGI	2	12	30							X		26		
SP-19	Radial Angle Gate Logic	6	4	18	(1-SORT)								42		
SP-20	Identify MLC	24	0	100				X				X	40		
SP-21	Target Selection Logic	40	0	181							X		110		
			TOTAL	OPERATIONS		69937	64	70337	70213	6869	19573	70076			
				MIX		21:14:65	10:11:71	21:14:65	21:14:65	24:14:62	21:14:65	21:14:65			
	OPERATION TOTALS (WORST-CASE)		W/O FFT MIX	OPERATIONS		12337	64	12737	13610	1129	1333	12476			
				MIX		24:3:73	18:11:71	24:6:73	24:6:73	46:5:49	41:9:53	24:6:70			
			W/O FFT CPDI	OPERATIONS		817	64	1217	1090	1129	1333	956			
				MIX		55:6:39	18:11:71	44:4:52	47:5:48	46:5:49	41:9:53	50:5:45			
										MEMORY TOTALS:			975	586	150

LEGEND:

X Denotes worst-case/critical path utilization for worst-case throughput requirement
* Totals include 33% additional short operations for subroutine linkages and other miscellaneous overhead instructions.

TABLE 29 RADAR SIGNAL PROCESSING COMPUTER REQUIREMENTS

SA-PD/DI RADAR SENSOR FOR CLASS II MISSILE

MODULE	NAME	ADD/SUB	MPY/DIV	LD/ST	UTILITY	PRE-LAUNCH	LAUNCH	RADAR CLUT. ACQ.	RADAR TGT. ACQ.	RADAR TRACK INIT.	RADAR TGT. TRACK	RADAR CLUT. TRACK	MEMORY *PROGRAM	MEMORY DATA RAM	MEMORY ROM
SP-1	Pre-Launch Mode Supervisor	2	0	15	(50-U21,3,5)	X							33		
SP-2	Launch Mode Supervisor	8	5	31			X						57		
SP-3	Clutter Acq. Mode Supervisor	4	0	23	(50-U21,3,5)			X					46		
SP-4	Track Acq. Mode Supervisor	2	0	16	(50-U21,3,5)				X				34		
SP-5	Track Init. Mode Supervisor	4	0	26	(1-U21,5)					X			50		
SP-6	Target Track Mode Supervisor	7	0	40	(5-U21,3,5)						X		77		
SP-7	Clutter Track Mode Supervisor	4	1	19	(50-U21,3,5)							X	42		
SP-8	Compute J/N	251	3	1004				X	X	X			32		
SP-9	Post-Detection Integration	12800	3200	41600		X		X	X			X	65		
SP-10	Det.W/Sliding Threshold (Mult.Ch)	2250	250	1500		X		X	X			X	55		
SP-11	Det.W/Sliding Threshold (1 Ch)	450	50	300						X	X		-		

TABLE 21 RADAR SIGNAL PROCESSING COMPUTER REQUIREMENTS

A-PD RADAR SENSOR FOR CLASS III MISSILE

MODULE	NAME	ADD/SUB	MPY/DIV	LD/ST	UTILITY	PRE-LAUNCH	LAUNCH	RADAR MODE CLUT. ACQ.	TGT. ACQ.	TRACK INIT.	TGT. TRACK	CLUT. TRACK	MEMORY *PROGRAM	DATA RAM ROM
SP-1	Pre-Launch Mode Supervisor	2	0	15	(100-U21)	X							33	
SP-2	Launch Mode Supervisor	8	5	31			X						57	
SP-3	Clutter Acq. Mode Supervisor Mode Super-	4	0	23	(100-U21) 3,5 3,5			X					46	
SP-4	Track Acq. Mode Supervisor	2	.	16	(100-U21) 3,5				X				34	
SP-5	Track Init. Mode Supervisor	4	.	26	(1-U21,5)					X			50	
SP-6	Target Track Mode Supervisor	7	3	40	(5-U21,) 3,5						X		77	
SP-7	Clutter Track Mode Supervisor	4	1	19	(100-U21) 3,5							X	42	
SP-8	Compute S/N	501	3	2004				X	X	X			32	
SP-9	Post-Detection Integration	25600	6400	83200		X		X	X			X	65	
SP-10	Det.w/Sliding Threshold (Mult Ch)	4500	500	3000		X		X	X			X	55	
SP-11	Det.w/Sliding Threshold (1 Ch)	450	53	300						X	X		-	
SP-12	Target Search Gen-U.B. Doppler	6	0	17									38	

PRECEDING PAGE BLANK-NOT FILMED

TABLE 21 (Continued)

MODULE	NAME	ADD/SUB	MPY/DIV	LD/ST	UTILITIES	RADAR MODE: PRE-LAUNCH	LAUNCH	CLUT. ACQ.	TGT. ACQ.	TRACK INIT.	TGT. TRACK	CLUT. TRACK	MEMORY *PROGRAM	DATA RAM	ROM
SP-13	Target Search Gen-Age/ Doppler	4	0	10					X				18		
SP-14	Compute Angle Errors	8	20	24							X		68		
SP-15	Compute Dop-pler Error	10	6	25						X	X		53		
SP-16	Compute Range Error	6	2	28						X	X		47		
SP-17	Beta Flanking Log	13	8	63	(4-SQRT)						X		42		
SP-18	Compu - TOI	2	12	30							X		36		
SP-19	Radial Angle Gate Logic	6	4	18	(1-SQRT)								42		
SP-20	Identify MLC	1320	0	4320				X				X	40		
SP-21	Target Selec-tion Logic	48	0	181							X		118		
OPERATION TOTALS (WORST-CASE)		TOTAL		OPERATIONS		731227	44	739375	733740	9175	31733	736844			
				MIX		21:13: 66	18:11: 71	21:13: 66	21:13: 66	23:10: 66	21:14: 55	21:13: 66			
		W10 FFT		OPERATIONS		123227	44	131375	125740	3415	1333	128844			
				MIX		24:6: 70	18:11: 71	24:6: 70	24:6: 73	28:2: 70	45:10: 45	24:6: 70			
		W10 FFT CPDI		OPERATIONS		8017	44	16175	10540	3415	1333	13644			
				MIX		56:6: 38	18:11: 71	39:3: 58	47:5: 48	28:2: 70	45:10: 45	43:4: 53			
										MEMORY TOTALS:			975	5510	150

LEGEND:

* Totals include 30% additional short operations for subroutine linkages and other miscellaneous overhead instructions.

4.3 Anti-Radiation Missile (ARM) Sensors

The ARM sensor relies upon the radio-frequency radiation from the target aircraft radar, and this in turn demands a wider range of performance compared to the semi-active and active radar sensors described in the previous subsection. Presently there are no airborne ARM sensors in service, but the Navy BRAZO air to air missile is currently in the development phase. ARM sensors are applicable to class II and III missiles.

4.3.1 Data Acquisition

Data acquisition in ARM sensors is achieved through the "target-identification-acquisition-system" TIAS on-board the launch aircraft. The TIAS is essentially an electronic intelligence (ELINT) receiver which receives the radiated energy and identifies a friend or foe. TIAS designation accuracies for airborne ARM targets are listed in Table 22.

TABLE 22

TARGET RADIATION CHARACTERISTICS & TIAS DESIGNATION ACCURACIES

PARAMETER	TARGET-RADIATION	TIAS DESIGNATION ACCURACY
Frequency	5.0 - 18.0 GHz	±5.0 MHz
Pulse Width	0.1 - 10.0 μsec	±0.2 μsec
PRF	0.2 - 350.0 KHz	±500.0 Hz
Angle	360 deg	±5.0 deg

4.3.2 Search and Detection

The target search and detection function in an ARM sensor is relatively simple due to initializing by the TIAS in terms of the desired threat target.

4.3.3 Acquisition

Acquisition of the desired target relies upon a number of discriminants which are provided in the ARM signal processor and described below. Figure 48 is a functional block diagram of an ARM processor and Table 23 lists typical design requirements.

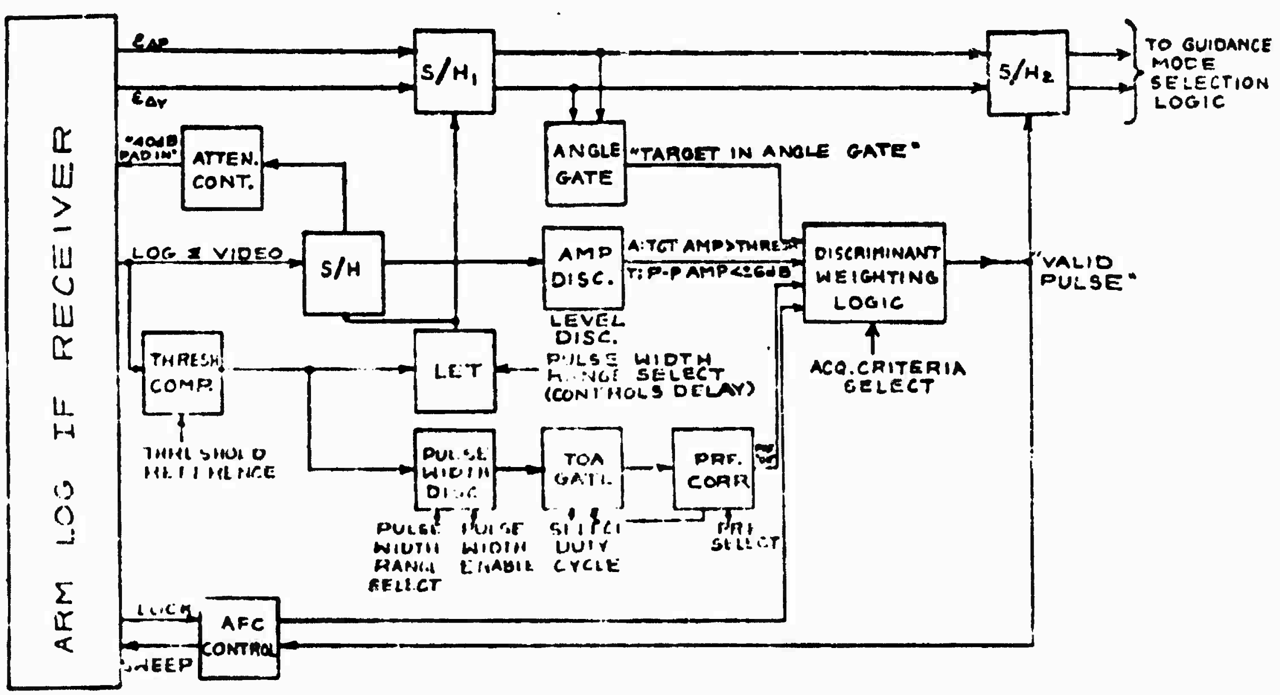


Figure 4. ARM Sensor - Functional Block Diagram

TABLE 23

TYPICAL ARM SENSOR DESIGN REQUIREMENTS

PARAMETER	DESIGN REQUIREMENTS
Dynamic Range	70 db
Beam Forming (4 channels)	$\Sigma \pm \Delta P$, $\Sigma \pm \Delta Y$
Video Bandwidth	10 MHz
Equivalent Receiver Bandwidth	100 MHz
Log Video Processing	
Gain Tracking	±1/2 db
Phase Tracking	$\pm 5^{\circ}$
Narrowband Frequency	±15MHz
Leading Edge Tracking	0.2 µsec.
Pulsewidth (3 ranges)	0.1 to 10 µsec.
PRF Correlation - range (f_{τ})	200 Hz to 350 KHz
PRF Correlation - bandwidth	$f_{\tau} \pm 10\%$
TOA Gate	
Power Level	CFAR Threshold
Pulse-to-Pulse Amplitude Window	±6 db
Angle Gate	
Acquisition	$\pm 5^{\circ}$ both Axes
Track	$\pm 1^{\circ}$ both Axes

The three receiver channels feed a beam forming network which forms the $\Sigma \pm \Delta_p$ and $\Sigma \pm \Delta_y$ beams. The latter outputs and the channel are video detected to feed four gain-matched logarithmic video amplifiers. These logarithmic-gain amplifiers allow operation to be maintained over a wide variation in received signal power. The respective angle channels and Σ channel are summed and simultaneously sampled by separate sample and hold amplifiers (S/H) for subsequent angle and pulse to pulse amplitude discrimination.

The ARM processor provides a narrowband (20 MHz) and a wideband (300 MHz) filter network. The narrow band filter is used against non-frequency agile targets.

The ARM video signal processor section provides the additional discriminants in order to cope with other radiating targets and/or ground clutter as follows.

LET - Leading Edge Tracking is used to time gate the multipath signal from the direct signal path.

Power Level - The signal level must exceed a variable threshold level before being gated into the receiver.

Pulse-to-Pulse Amplitude - After target acquisition, a level window is put around the target level and returns outside this window are rejected.

Angle Gate - Target signals outside an angular window set around the antenna boresight, are rejected. This window can be narrowed to provide further discrimination after the target has been acquired.

Pulse Width - Received target pulses are divided into three broad ranges from 0.1 to 10 μsec. This discriminant is used to pregate the PRF and TOA discriminants.

TOA - The time-of-arrival discriminant is obtained by phase-locking the PRF oscillator to a real-time analog pulse train from the avionics. The result is to put a coarse time gate around the selected target pulse.

PRF Correlator - The PRF correlator performs a frequency measurement of its input signals and provides an output indication when the PRF of the input pulse train is equal to a preselected value. The preselected PRF comparison value is supplied by the avionics control computer in the form of a 7-bit binary word corresponding to PRF values between 100 Hz and 350 KHz. The allowable deviation of the input PRF to still pass the PRF correlation test is ±10 percent of the nominal PRF reference value.

Acquisition Criterion - The target acquisition criterion is variable and selectable depending upon the target's capabilities and available designation information from the avionics. Each discriminant can be selected or deleted upon command.

Discriminants Weighting Logic - The discriminants weighting logic receives inputs from all the discriminants previously described and determines if a given pulse is valid. If valid, the pitch and yaw angle signals which were sampled by sample-and-hold amplifiers are converted to digital values for translation into seeker head and missile guidance commands. The valid target pulse is also used to update the AFC loop.

4.3.4 Angle Extraction

Pitch and yaw angle error signals are obtained from a conventional monopulse antenna system requiring full aperture gain over the complete frequency band to provide an ideal antenna for dual mode RF sensors.

4.3.5 Electronic Counter-Counter Measures (ECCM)

Inherent in the ARM processor are the discriminants which overcome the countermeasures used against ARM sensors. Frequency agility is accomodated by increasing the bandwidth of the antenna and receiver. Pulse-width variation, and PRF agility are handled by increasing the discriminants to the wider pulse and the longer duty cycle respectively.

The shut-down of the radar is best handled by an active terminal sensor. The dual mode sensor is discussed in a subsequent section.

4.3.6 Digital Signal Processing

At the present time, digital signal processing for ARM sensors appears to be limited to certain video processing functions, e.g., discriminant weighting logic, and overall ARM sensor control due to the wide bandwidths and narrow pulse widths being processed.

4.4 Infra-Red Sensors

4.4.1 Type 1 Reticle Sensors

The Type 1 reticle sensor is the most simple of the three types identified earlier and described in 3.1.2.2. Guidance information is derived from a single channel electrical signal which consists of an amplitude modulated carrier frequency. The amplitude modulated carrier frequency is generated by a rotating mechanical chopper (reticle) located in the image plane of a focusing optical system. A simplified version of such a reticle is shown in Figure 49.

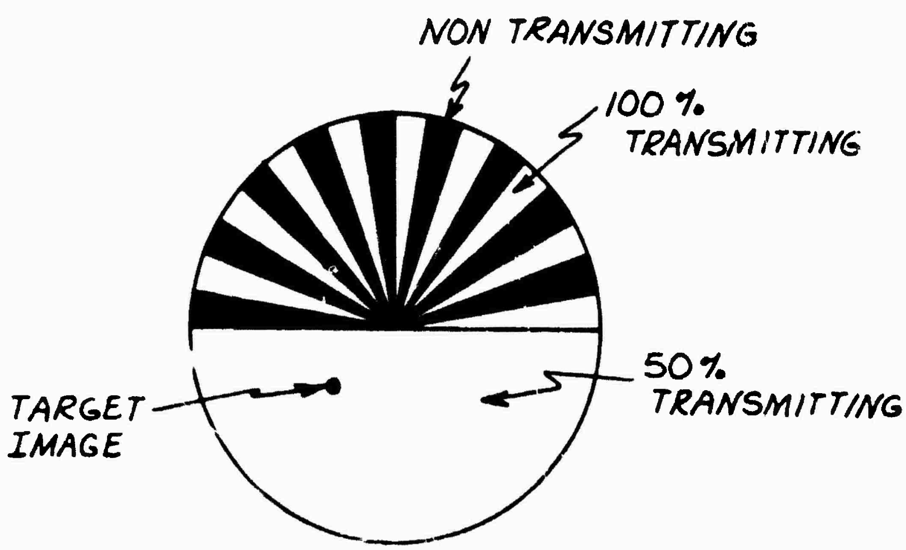


Figure 49. Simplified Amplitude-Modulation Reticle for Type 1 Sensors.

Relative motion between the reticle and the target image generates an interrupted carrier wave (ICW) of the frequency f_C where $f_C = 2N\dot{\alpha}_M$ where the carrier is off for a time $1/2\dot{\alpha}_M$ and is on for a time $1/2\dot{\alpha}_M$. If such an ICW signal is passed through a filter whose center is at f_C and bandwidth $2\dot{\alpha}_M$, the output is a signal at the carrier frequency which is amplitude modulated at the rate $\dot{\alpha}_M$. The IF signal to noise ratio for a square IF filter is

$$(S/N)_{IF} = \frac{H}{NEPD}$$

where:

$$NEPD = \left(\frac{f_o}{A_o \tau_o}\right) \frac{\sqrt{N\,\Delta\Omega\dot{\alpha}_M}}{0.433D^*}$$

where:

H - Received radiant intensity at front of dome.

f_o - Focal length of optics

A_o - Optics collecting area

τ_o - Optics transmission

$\Delta\Omega$ - Solid angle field of view of the total reticle

$\dot{\alpha}_M$ - Reticle rotation rate

D* - Detector sensitivity in $Hz^{\frac{1}{2}}$ Watt

NEPD - Noise Equivalent Power Density of sensor

If this signal is now half-wave rectified at the carrier frequency and the output filtered about $\dot{\alpha}_M$, the resulting signal is a sine wave of frequency $\dot{\alpha}_M$. This signal is then phase detected and used to drive the missile servo and seeker tracking loops. For large $(S/N)_{IF}$ the post detection signal to noise ratio is given by

$$(S/N)_{POD} = (S/N)_{IF}\ (1/\pi)\ (\Delta f_{IF} / \Delta f_M)^2$$

In this case the only effective digital signal processing would be target track initiation, updating and ECCM/flare discrimination utilizing the outputs of the analog signal processing section.

4.4.2 Type 2 Image Plane Scanner

The Type 2 sensor is a scanning array which for convenience will be taken as a linear array scanning the optical image plane in a direction perpendicular to its length. The scanning method is shown in Figure 50. This method of scene scanning provides a great deal more information than that of the Type 1 Sensor. In the latter, the single reticle and detector combination gave a single sine or square wave whose phase relationship was indicative of the direction in which the average target was off boresight.

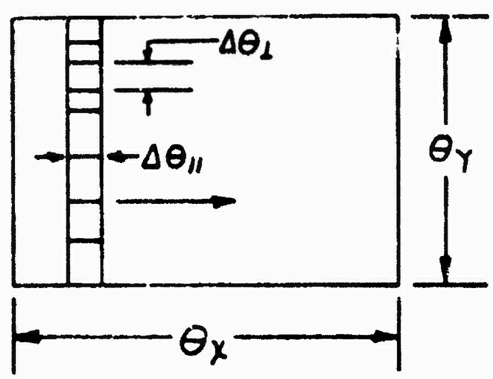


Figure 50. Linear Array Scanner - (Type II Sensor)

In the Type 2 sensor there are N channels of spatial information in the vertical (y) direction and M intervals of temporal information in the horizontal (x) direction. Each of the M intervals will have a temporal length equal to the detector dwell time on a point source target and by appropriate clocking relative to the beginning of a scan, each interval can be directly related to x-position in the field-of-view.

If the amplifier/filter bandwidth is properly matched to a point source pulse response ($\Delta f_n \Delta_\tau = 1/4$), then the signal-to-noise ratio $(S/N)_F$ for a uniform background is given by:

$$(S/N)_F = \frac{H}{NEPD}$$

where:

$$NEPD = \left(\frac{f_o}{A_o \tau_o}\right) \frac{\sqrt{\theta_x \theta_y \dot{\alpha}/N}}{1.06\ D^*}$$

where:

f_o - Telescope focal length

A_o - Objective collecting area

τ_o - Optics transmission

D^* - Detector sensitivity

θ_x, θ_y - Angular extent of optical field-of-view in the x and y direction

$\dot{\alpha}$ - Scanning frame rate

N = Number of detectors in linear array, and

NEPD = Noise Equivalent Power Density of System in watts/cm^2 at objective of system.

In this case there are N channels of information. If the background is uniform and there is a single point source in the field-of-view then vertical channel response and time of detection in the scan constitutes target position data. However, even in this simple case it is necessary to measure the mean noise power in each channel and set a threshold above which the signal plus noise must rise to assure a low false alarm rate. In more complicated background situations it is necessary to resort to more sophisticated signal processing to separate the "true" target from the background clutter.

The functional block diagram for a linear array IR sensor employing both analog and digital signal processing is shown in Figure 51. The analog signal processor consists of a cell selector/multiplexer, preamp, and double threshold logic circuitry. For any one position of the linear array in the scanned field-of-view, the 3-cell sliding window detector operates to identify point targets and reject edges as caused by large extended targets e.g. the horizon or clouds. The operation of the 3-cell detector is basically to compare the output of the center of the three cells with the sum of all three. This is accomplished by using effectively a tapped delay line as shown in Figure 52. When the array scans across an "edge", the sum of the three cells will always be larger than the scaled center cell resulting in no detection. A second thresholding operation is required to reject detections from the fine structure of IR clutter such as variations in cloud backgrounds. This is accomplished by comparing the signal passing the first threshold to a threshold which is the average of many cells. This is effectively a CFAR type of operation as implementated in radar signal processors.

With this type of thresholding operation, only legitimate IR point targets will be encoded and stored in the digital signal processors buffer memory, with a matrix position derived from the "scan decoder". After acquisition, the target is "tracked" into the center of the matrix array field-of-view by the sensor guidance data processing/seeker head loop.

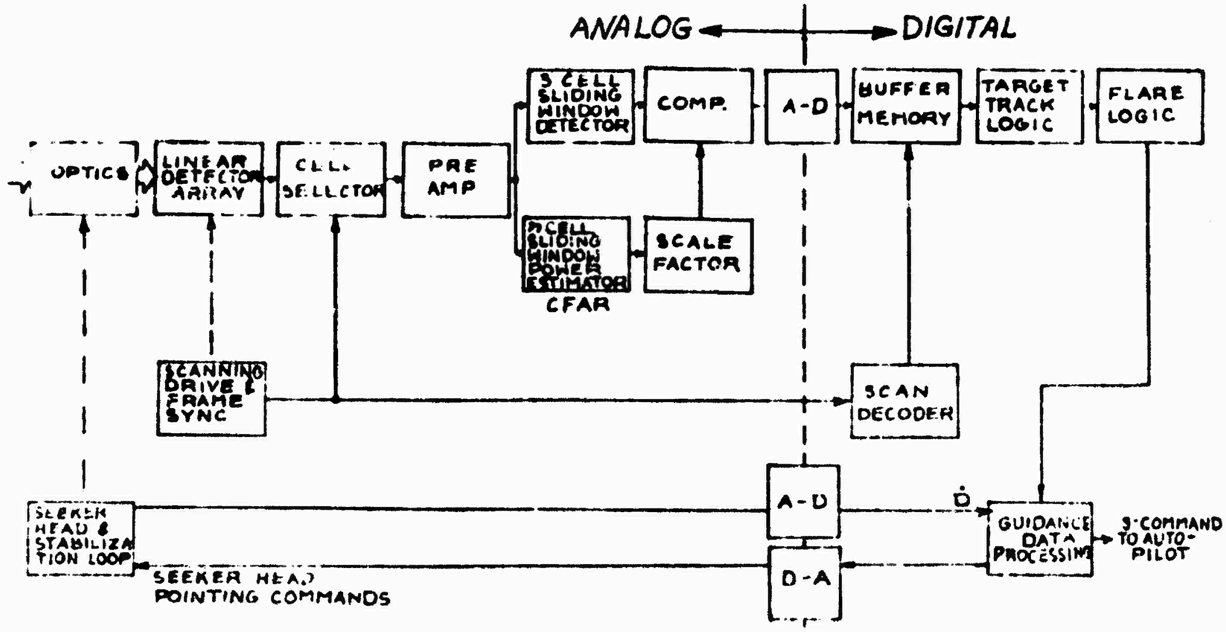


Figure 51. Type 2 Image Plane IR Sensor-

Functional Block Diagram

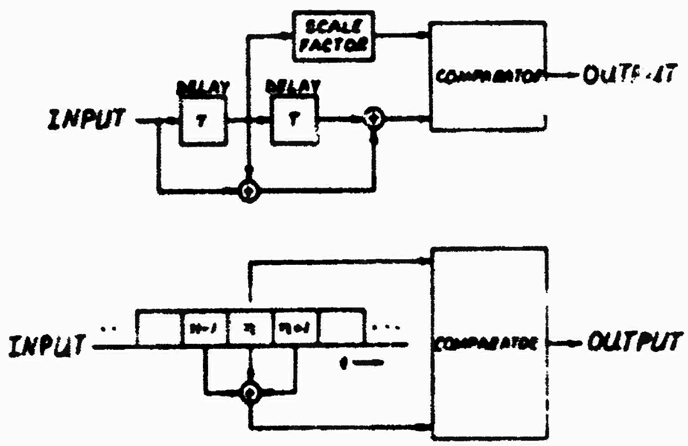


Figure 52. 3-Cell Sliding Window Detector

Once the target is in the center of the array only a small number of cells about the matrix center are examined to develop the target track information. The flare logic consists of entering a coast mode for a specified period of time when two targets are suddenly detected in the tracking window. Flares having high aerodynamic drag, will very quickly move outside of this narrow tracking FOV and tracking is then re-established.

4.4.3 <u>Type 3 MXN Matrix Array</u>

The functional block diagram for a Type 3 IR Sensor is shown in Figure 53. Type 3 sensors perform the same functions as Type 2 except that the scanning linear array is replaced by a non-scanning square array of detectors each of which integrates over a frame time. The frame time is determined by the rate at which the information is read out.

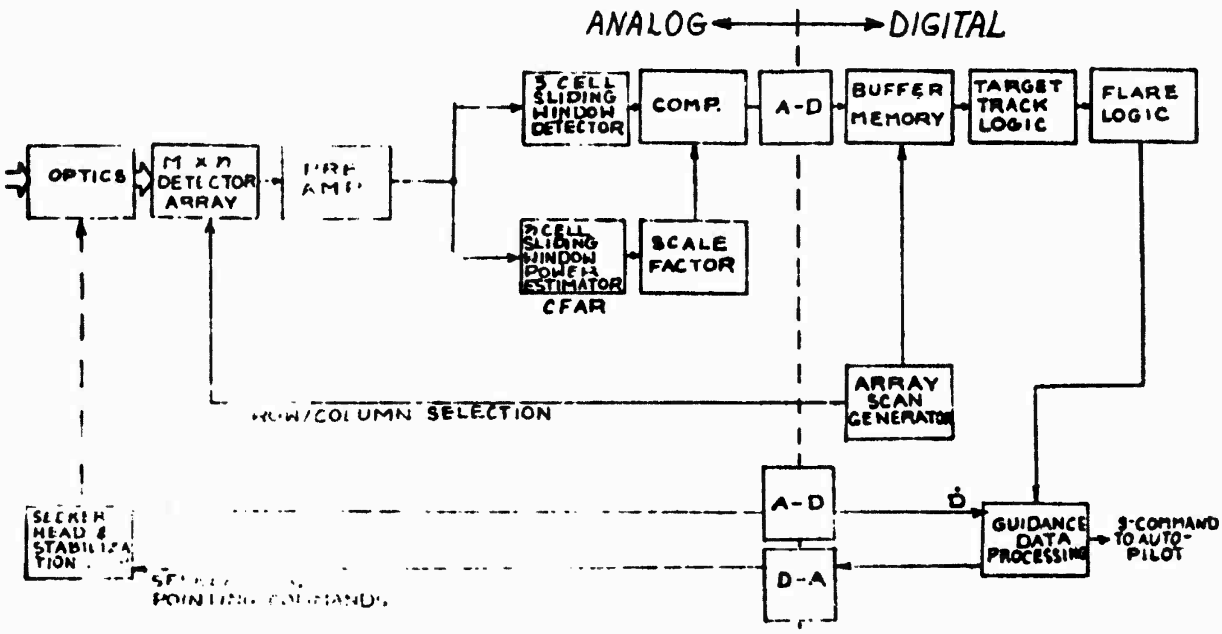


Figure 53. Type 3 MXN Matrix Array IR Sensor Functional Block Diagram

Suppose that the matrix is square with N X N = N^2 elements as shown in Figure 54. For the same elemental detector size and for the same frame rate this situation gives a voltage signal-to-noise ratio which is $\sqrt{N}$ times that of the scanning linear array because each detector integrates the received signal during a complete frame time. Otherwise exactly the same signal processing applies as described in section 4.4.2.

Δa_{11}	. .	. .	. .	Δa_{1N}
. .	. .	. .	. .	. .
. .	. .	. .	. .	. .
. .	. .	. .	. .	. .
Δa_{N1}	. .	. .	. .	Δa_{NN}

Figure 54. Matrix Array of Detectors

4.5 Multimode Sensor Systems

Multimode sensor systems incorporate a mix of the single mode sensors previously described, to provide near optimum target homing data for every missile flight phase, by recognizing the performance limitations of each sensor type for a given intercept scenario and target environment. Multimode sensor systems are considered for Class II and III missiles only due to their greater sophistication and longer range, compared to more simple Class I, short-range weapons. Two types of multimode sensor systems have been identified: dual mode and triple mode, the former for Class II missiles only, by virtue of size, weight and power considerations, whereas both dual and triple mode sensor systems are evaluated for Class III missiles.

4.5.1 Dual Mode Sensor Systems

Table 24 is a listing of five possible dual mode sensor combinations. The candidate sensors for the Class II missile are IR/SAR, AR/SAR, and ARM/SAR. Note that combinations 2, 4 and 5 incorporate a semiactive radar (SAR) and in fact the SAR mode can be used effectively without the other modes. However, the IR sensor does provide the capability for low altitude tail-chase more effectively than a SA-CW radar sensor. Figure 18 shows that SA-PD can also provide good tail-chase capability and therefore the only real advantage in the IR/SAR combination is some additional effectiveness against ECM. The active radar (AR) sensor provides little additional capability over SAR for Class II missile intercept ranges. The SAR/ARM constitutes the best

TABLE 24

DUAL MODE SENSOR COMBINATIONS & APPLICATIONS BY MISSILE CLASS

	MISSILE CLASS			
	II		III	
SENSOR COMBINATION	MIDCOURSE	TERMINAL	MIDCOURSE	TERMINAL
1. IR/ARM				
IR	X	#	X	#
ARM	#	X	#	X
2. IR/SAR		#		
IR	X	#	X	#
SAR	#	#	X	#
3. IR/AR				
IR	X	#	X	#
AR	X	#	X	#
4. SAR/AR		#		
SAR	#	#	X	#
A	X	#	X	#
5. SAR/ARM		#		#
SAR	#	#	X	#
ARM	#	X	#	X

LEGEND:

(X) - Not Effective

(#) - Effective

candidate for dual-mode for either the Class II or Class III missile. The ARM sensor can provide both midcourse and terminal (degraded) guidance. This is a distinct asset to SAR illuminator power requirements since ARM can be employed at launch.

In summary, an attractive dual-mode candidate which exhibits advantages over a single mode sensor system is SAR/ARM.

4.5.2 Triple Mode Sensor Systems

Three triple mode sensor systems are evaluated in Table 25 for Class III missiles. Again the SAR/ARM combination is effective for the reasons described above, and hence the two most practical candidates are SAR/ARM/IR or SAR/ARM/AR. The limitations of the IR sensor and the degree of improved capability it adds to the multimode sensor system would not justify the IR sensor. In the light of the foregoing assessment, the best-choice triple-mode sensor system which provides distinct advantages is the SAR/ARM/AR. Figure 55 is a first level functional block diagram of an SAR/ARM/AR sensor.

In terms of hardware savings through common/shared equipment, the antenna is shared by all three sensor types and one receiver and signal processor is common to the SAR and AR sensors since the additional AR transmitter is the only distinction between the two radars. However, due to the wider bandwidth and other unique performance characteristics of the ARM sensor, a separate/additional receiver and analog signal processor is required to support this target sensor.

TABLE 25

TRIPLE MODE SENSOR COMBINATIONS FOR CLASS III MISSILES

SENSOR COMBINATION	MISSILE MODE MIDCOURSE	MISSILE MODE TERMINAL
1. IR/ARM/SAR		
IR	X	X
ARM	#	X
SAR	X	#
2. IR/SAR/AR		
IR	X	#
SAR	X	#
AR	X	#
3. SAR/ARM/AR		
SAR	X	#
ARM	#	X
AR	X	#

LEGEND:

(X) - Not Effective

(#) - Effective

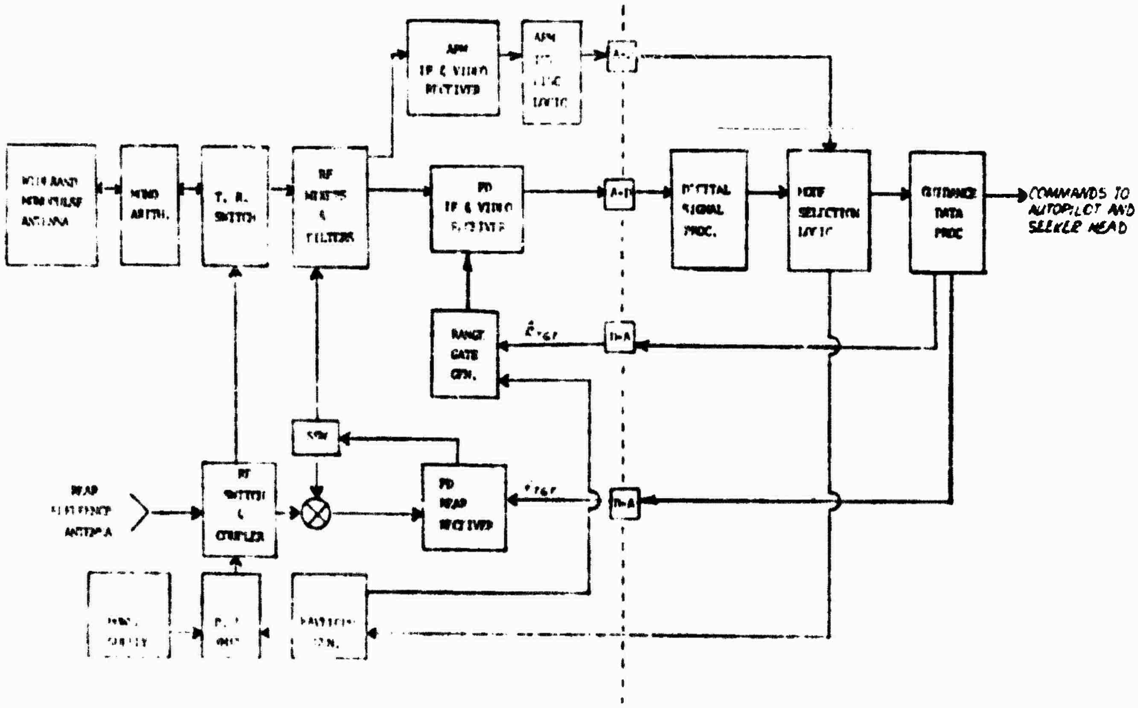


Figure 55 Triple Mode SAR/ARM/AR Sensor System Block Diagram

4.5.3   Multimode Sensor Operating Modes

A description of multi-sensor systems by operating mode is given in the following paragraphs.

Integrated Data Acquisition - Data acquisition for SAR and ARM sensors is the same as for their individual modes. The SAR sensor requires the AI radar to acquire the target and initialize the SAR sensor. ARM requires a TIAS to detect radar radiation, verify the threat radar sensor, and initialize the ARM mode. The requirements for each sensor have been stated previously for SAR and for the ARM mode.

Acquisition Processing - The acquisition of a target during launch can employ either the SAR or ARM sensor. For large targets (>$10m^2$) the SAR is the logical choice while a small radiating target can best be handled by the ARM sensor. Acquisition ranges beyond 80 nmi can be obtained by an ARM sensor against any airborne radar and 50 nmi against small (<$0.5m^2$) radar targets with the SAR sensor. The active sensor (AR) is best employed during the terminal phase. The range limitation for an AR sensor is weight and cost. Acquisition will be accomplished by the SAR or ARM sensor with terminal guidance depending upon the AR sensor.

Track Processing - Track processing for the multimode sensor is identical to the track processing required for individual sensors. The hand-over logic from the midcourse sensors (SAR or ARM) to the terminal sensor (AR) will be controlled by the AR sensor, and depends on the signal-to-noise ratio of the AR sensor. The advantage of the ARM/AR modes is the capability of multiple firings and a "launch and leave" capability. It should be made clear, that with a certain unique antenna design, full aperture gain over the complete ARM band can be obtained and this eliminates the need for two separate antennas, one for ARM and another for SAR.

4.6 Fuzing

In this section the capabilities and complexities of typical air-to-air missile fuzing systems and their amenity to modular digital tecnniques are discussed. All systems under consideration are integrated with the missile guidance function and hence the application of a fuze system to a specific class of missile is contingent upon the guidance program module(s) used and their corresponding data outputs. The term target detection device (TDD) refers to the target sensor section of the fuzing system (Figure 56) as distinct from the safing & arming and warhead sections. Digital techniques become practical and most effective in sensor signal processing and optimally timing the generation of the firing command, using available guidance information and computing the time of warhead detonation following target detection.

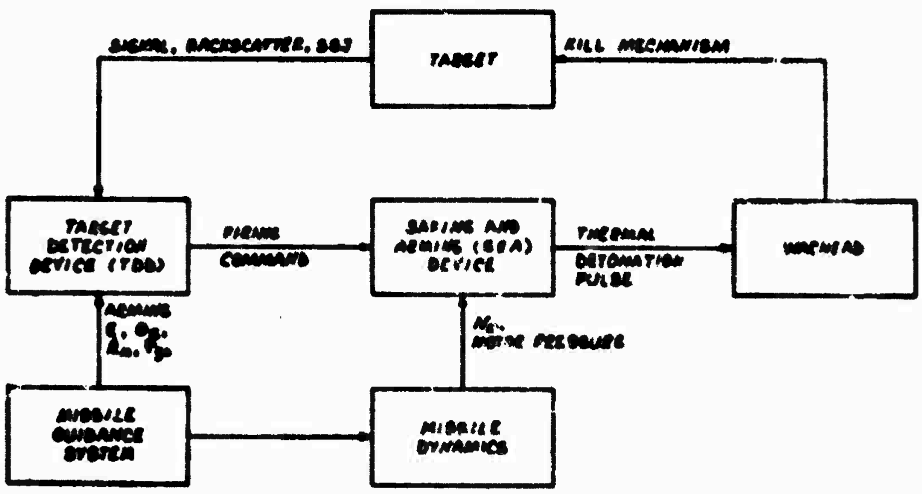


Figure 56. Fuzing System - General Functional Block Diagram

4.6.1 TDD Types vs Missile Applications

Various TDDs are described in the following paragraphs which, although not all inclusive, nevertheless cover a broad spectrum of fuze types. Table 26 lists the various types of TDDs and their application by missile class. The selection of a TDD for a specific missile class is not inflexible however, since there is no technical reason why active radar fuzes are not applicable to Class I missiles. The decision not to include them in this class is based on high cost and complexity. Semi-active radar fuzes were not chosen for sophisticated Class III missiles due to their greater susceptibility to clutter and chaff. Optical fuzes were not considered for Class II and III missiles because of prefunction problems in an aerosol environment, and capacitor fuzes were similarly rejected since they are extremely range limited, i.e. target detection range of 2x missile length.

TABLE 26

TARGET DETECTION DEVICE (TDD) TYPES VS MISSILE APPLICATIONS

TTD TYPE	MISSILE CLASS I	II	III
Electrostatic/Capacitor	X		
Semi-Active CW Radar			
CW-Doppler	X	X	X
Active Radar			
Delayed Local Oscillator (DLO)		X	X
Injection Locked Pulse Doppler (ILPD)		X	X
Pseudo-Random Coded (PRC) CW Doppler		X	X
Active Optical	X		

Electrostatic/Capacitance TDD - Capacitance TDDs, as shown in Figure 57, are based upon the principle that the capacitance of a charged body changes when a second object enters the electrostatic field.

In practice an electrostatic field is set up by a number of charged electrodes, and, to simplify the design of the target sensor, the charge and associated field is sinusoidally pulsated. The pulsation frequency is not critical to the mechanization. Two modes of target detection are provided: one based upon the rate of change of field strength and the other upon the amplitude of the detected field.

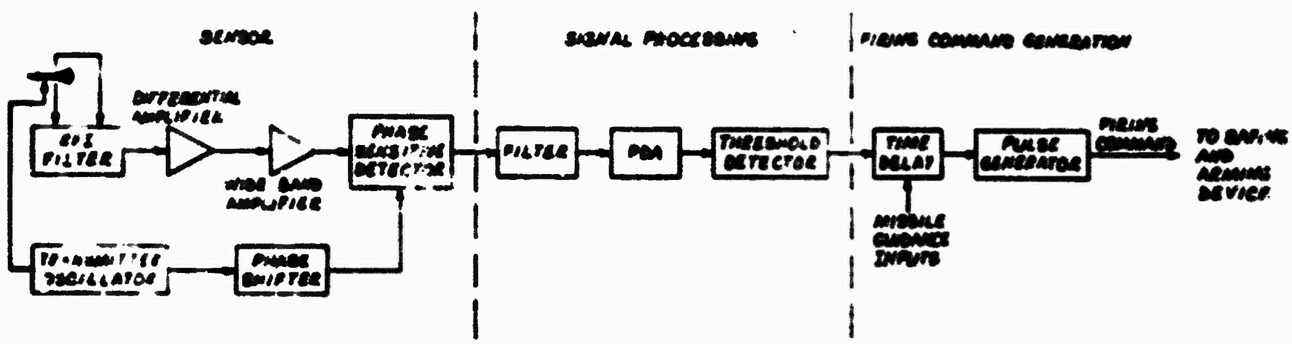


Figure 57. Capacitance TDD Functional Block Diagram

Proximity Mode - In the proximity mode of operation the sensor responds to a sign change in the rate of change of the electrostatic field caused by the intrusion of a moving object. Geometric considerations have shown that this change of sense can be arranged to occur on a surface of a cone whose base faces forward where the axis is coincident with the longitudinal axis of the missile. In non-parallel path engagements the axis of the detection cone rotates in a sense that increases the probability of a warhead fragment striking the target. Figure 58(A) illustrates the proximity zone of a capacitance TDD.

Grazing Mode - This mode corresponds to the detection of an object a few inches from the missile (see Figure 58(B), and is the only mode of detection for an object on the forward longitudinal axis of the missile since the detection characteristics of the proximity mode exhibit a null on the nose of the missile. The graze mode can be employed as a back-up to the proximity mode.

(A) - Proximity Zone

(B) - Graze Zone

Figure 58 Capacitance TDD Proximity and Graze Influence Zones

Semi-Active CW Doppler Radar TDD - Semi-active CW radar TDDs sense the target doppler shift close to intercept using an intercept arm gate, enabled by a doppler signal from the seeker signal processor, and generating a firing command for the safing and arming device. The firing command is generated when the target enters the narrow beam formed by fixed antennas mounted on the side of the missile seeker.

The SA-CW doppler TDD detects aircraft illuminator-derived signals reflected from the target or energy received directly from jammers aboard the target. Since the signal level may vary over wide limits, two direct coupled broad beam antennas, and associated circuitry, detect the signal, adjust the system sensitivity, and provide a reference signal for a differential detector used to trigger the fuze.

A block diagram of a basic fixed-angle, SA-CW TDD is shown in Figure 59. The doppler return from the target is received by the broad and narrow beam antennas. The broad beam signal is larger than the narrow beam signal in all directions except in the preferred direction of the narrow beam antennas. Diode attenuators are activated in the home-on-jam mode to increase the sensor dynamic range and reduce the probability of a premature firing command due to clutter. In each channel the doppler return is mixed with the oscillator output to produce an output at IF. These signals are amplified in the fuze receiver the gain of which is controlled by the detected broad channel output. The doppler is recovered by mixing the IF signal with the rear signal in a balanced mixer. The detected doppler signals are amplified in a video amplifier and applied to the fuze logic. This logic compares the signals in the narrow and broad channels, and generates a delayed fuzing command when this difference exceeds a prescribed level.

Significant/distinguishing characteristics of a semi-active CW radar fuzing system are:

1) Less complex and less expensive than an active system

2) Less burn through capability against noise jammers than the active fuze.

3) Susceptibility to clutter and chaff (a sharp cutoff range is not feasible because of the CW mode)

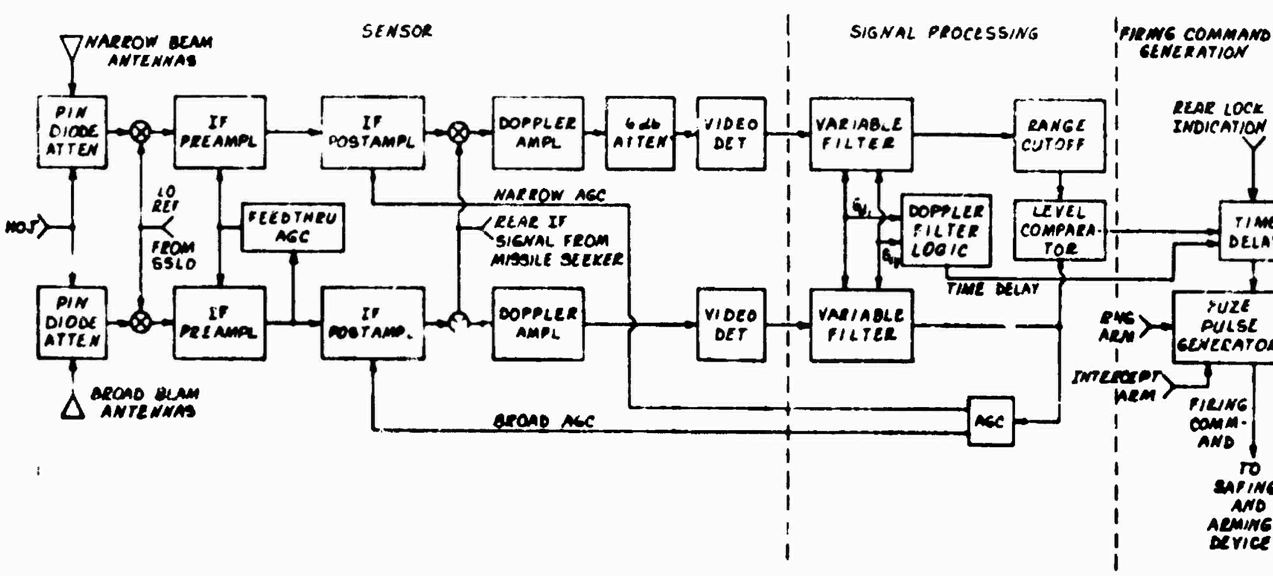


Figure 59. Semi-Active CW Doppler Radar Fuzing System Functional Block Diagram

Active Pulse Doppler Delayed Local Oscillator (DLO) TDD:

The DLO TDD employs an active pulse-doppler target sensor which is enabled by an intercept arm command from the missile seeker signal processor. A series of pulses are transmitted to the target and the reflected energy received by the sensor exhibits a doppler shift proportional to the closing velocity. A functional block diagram of a DLO TDD is shown in Figure 60. The received energy is processed through a balanced mixer which is activated by delayed energy from the transmitter such that pulse to pulse coincidence occurs at a specified range from fuze to target. The envelope is detected and integrated in the boxcar generator and passed through an automatic gain controlled (AGC) amplifier to a set of parallel filters. One of the filters passes the expected range of dopplers, while the other is set to pass another portion of the spectrum using the same bandwidth. The output of the noise filter is used to normalize the signal level out of the doppler filter. The normalized signal is then compared with a threshold to initiate a firing command.

Some characteristics of this TDD are:

1) Range Cutoff
2) Electronics are cheaper than PRC active fuze
3) No inherent FOJ capability, although some could be added
4) Separate transmit and receive antennas are necessary.

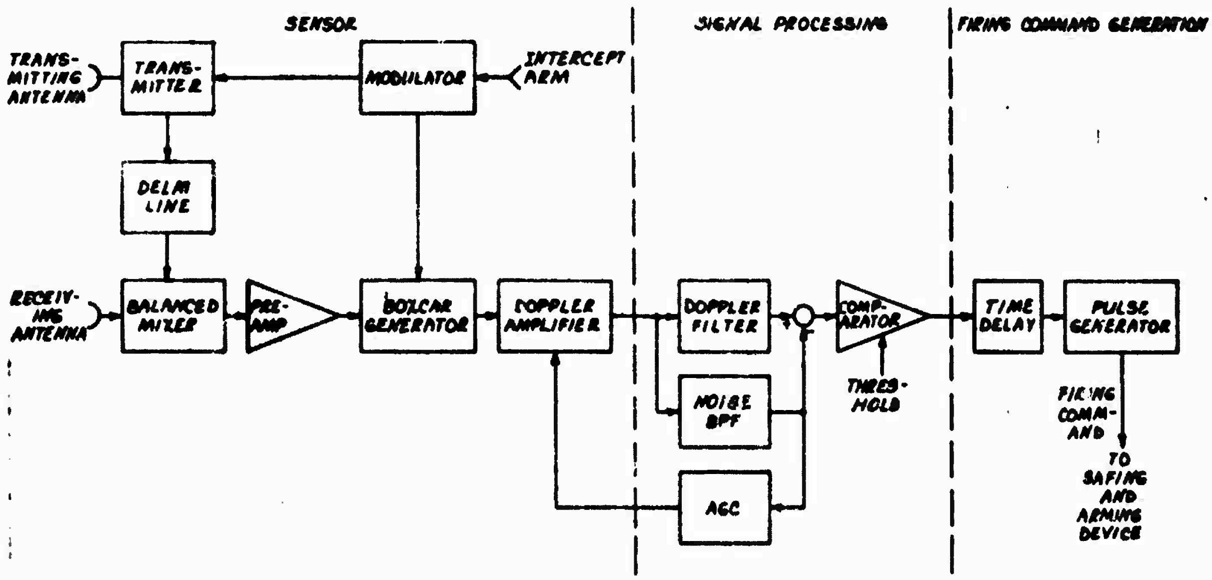


Figure 60. Active Pulse-Doppler, Delayed Local Oscillator (DLO) Target Detection Device.

Injection Locked Pulsed Doppler (ILPD) TDD - In this TDD (Figure 61) an injection locked transmitter and homodyne receiver are employed to provide coherent detection. With this approach, the transmitter serves as a power amplifier for the local oscillator frequency. The pulsed transmitter is locked to a CW Gunn oscillator which provides the local oscillator drive to the mixer.

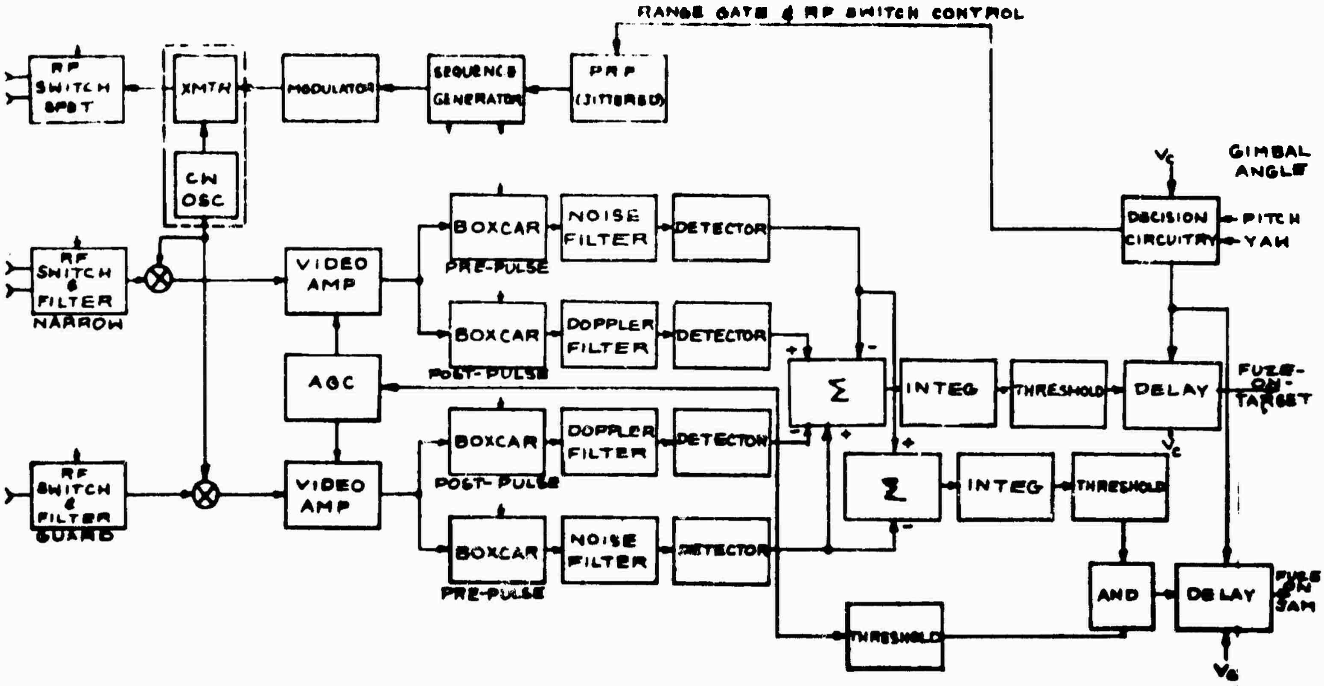


Figure 61. Injection Locked Pulsed Doppler (ILPD) TDD, Functional Block Diagram

Signal processing includes provisions to compensate for broadband noise, and for sources observed in the sidelobes of the narrow antenna pattern. Separate and concurrent fuze-on-jam capability is provided.

Two receivers are provided, one for the narrow beam antennas, and one for the guard antennas. Identical processing is employed so that the resulting outputs contain only the differences due to the antenna patterns. Both receivers are gated on twice during each transmit period, once just before radiation and again just after radiation. The pre-transmit signals are used as a reference measuring the noise level and the level of extraneous signals such as RFI or jamming. These signals provide an independent passive fuze-on-jam, (FOJ), capability to discriminate target returns from extraneous signals. The post-transmit signals are in-range target returns plus the signals in the earlier gate. A fast acting AGC system is used to adjust the gain at both narrow and guard channel video amplifiers over a wide range to accomodate large variations in signal level. The information for the AGC loop is derived from the guard channel and adjusts the gain of the narrow channel to the proper amount such that the fuzing threshold will be exceeded by a target return in the narrow channel.

Notable advantages of the ILPC TDD are:

1) Coherent detection for the elimination of feedthrough and the identification of moving targets.

2) Narrow beam sidelobes are protected by means of a guard beam.

3) Separate fuze-on-jam channel is provided for those cases where noise precludes the normal fuze on skin mode.

Active CW Doppler, Pseudo-Random Phase Code TDD - The pseudo-random code (PRC) TDD uses a pseudo-random code modulation technique in which the phase relationship of an RF carrier is changed by a binary code sequence. Specifically, the carrier phase is changed by 180 degrees each time the modulating code sequence changes from one to zero or from zero to one. Thus, a transmitted waveform is generated having the form:

$$E_t = A\,[\cos(\omega t + u(t)\pi)]$$

where: $u(t) = 0$ or 1

A block diagram of a low-power CW microwave PRC TDD is shown in Figure 62.

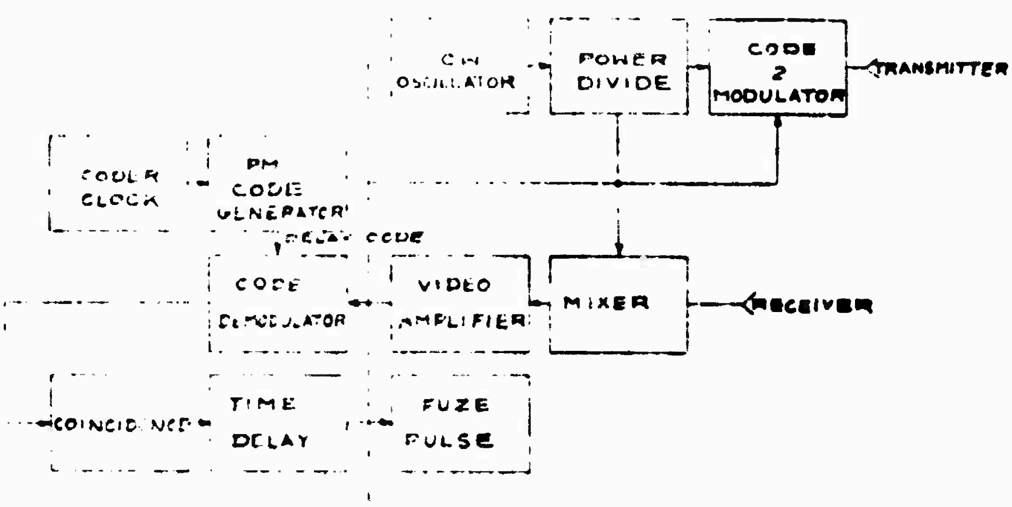


Figure 62. Active CW Doppler PSC TDD Functional Block Diagram

A CW microwave signal is generated by the transmitter and routed through a coupler for local oscillator drive to the balanced mixer. The remainder of the output signal is then modulated in a diode phase modulator, such that the phase of the output has either a 0° or 180° phase relationship to the input. The modulated signal is then divided and radiated by the two antennas. The 0° or 180° phase modulation is selected by the two logic states of a digital feed-back shift register (FSR), or pseudo-random code generator, which is in turn driven by the clock oscillator. Signals returned to the two antennas from a target, or other reflective source, are recombined and routed to the mixer.

Homodyne action between the local oscillator signal and the modulated return signal produces a bi-phase coded doppler output. The coded doppler signal is then passed to a video correlator where it is mixed with a delayed form of the code used to drive the modulator. If the target return signal originates from a range corresponding to the amount of delay between the modulating code and the correlating code, the signal will be demodulated and the doppler can then be filtered, amplified, detected, and used to initiate a firing signal. When the target return delay differs slightly from the correlation delay, the correlator output will contain coded and decoded elements which can be filtered to remove the still coded portion. The amplitude of the decoded, or correlated doppler, is reduced however, and the amplitude of the correlated portion continues to decrease as the difference in delay increases until a delay equal to one bit of the code is reached. At this time, only uncorrelated frequency components are obtained. These are eliminated by the doppler filter. This auto correlator action establishes the range response of the fuze.

The principal advantages of the PRC are:

1) High average-to-peak power ratio (the system is CW so that average power equals peak power)

2) Unambiguous range measurement to large ranges (a long code gives the unambiguous range)

3) Good range resolution (the bit width determines the range solution)

4) Adaptable parameters (code clock frequencies, codes, delay ranges, etc. can be varied by logic commands)

Active Optical TDD -

A typical active optical TDD is shown in Figure 63. This system is of the pulse-amplitude-modulated active, optical class (optical monopulse) and utilizes threshold detection with range gating to establish short and long range cut offs. These fuzes can be optically configured in several ways one of which is shown in Figure 63 for analysis purposes. This uses a single package G_aA_s laser array and simple optics, in this case reflecting cones and reflecting wedges to obtain a full 360° field around the missile. The incorporation of a position sensitive Schottky photodiode as the detecting element permits

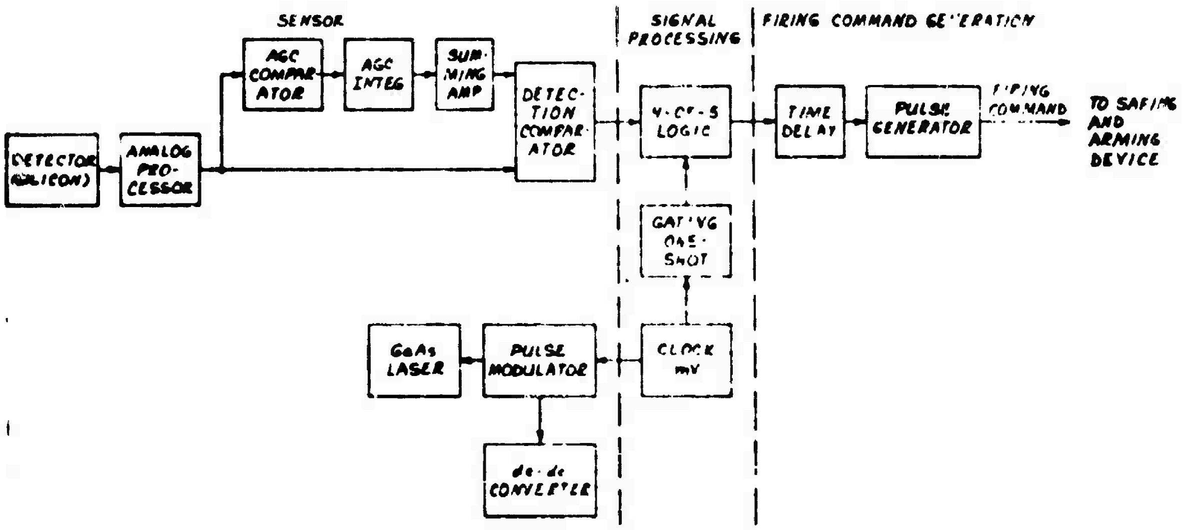


Figure 63. Active Optical TDD, Functional Block Diagram

bearing or polar angle, determination as well as range and range rate.

Another form of the optical TDD is the pulse amplitude system (proximity TDD) utilizing amplitude only without range gating. The pulse width for such a system is wide, i.e. up to several microseconds, with gating of the receiver only to prevent noise false alarm triggering during the interpulse interval. This is a "look-while-transmit" system and although low-cost is subject to aerosol backscatter triggering. This fuze has no range cut off and is dependent on target radar cross section at the optical frequency.

Significant characteristics of active optical TDDs are:

1) Excellent ECM operation

2) Severe prefunction problems in an aerosol environment

3) Range limited to about 50 ft/15.2m

4.6.2 Digital vs Analog Functions

A parallel can be drawn with the missile seeker sensors when considering digital implementations of fuze target detection devices, due to similarities in the types of target sensors employed.

The greatest impact of digital processing, however, can be made in the timing of the firing command to the safing and arming device since this has a direct bearing on the effectiveness of any given warhead. Further, the use of more sophisticated timing algorithms using data from the missile guidance system has been inhibited to date by the limitations of analog circuit technology and design techniques. With the foregoing observations in mind, the time delay function has been singled-out for further analysis and digital implementation as described in the following paragraphs.

4.6.3 Time Delay Algorithms

The time delay algorithms selected for this study are but two of many possibilities. The overriding objective is to time the warhead detonation to insure intercept at the target by the warhead byproducts. This is accomplished by a knowledge of intercept kinematics at the time of target detection by the TDD.

The selection of a proper algorithm is determined by the information available from the missile guidance system and the accuracy of this data. This is apparent from the algorithms presented in conjunction with the input data accuracies shown in Table 27 Part of the information needed is not available in Class I missiles and hence the inputs of relative velocity and miss distance become constants derived from analysis. Conversely all the required real-time data is available in Class II and III missiles. In addition the information available is more accurate for Class III missiles compared to Class II systems, resulting in improved performance for the Class III case.

TABLE 27

TIME DELAY ALGORITHM

AVAILABLE REAL-TIME DATA & ASSOCIATED RMS ACCURACIES

(UNITS: meters, sec., rad)

MISSILE CLASS

PARAMETER	I	II	III
V_R	N/A	$0.005\ V_R$	$0.005\ V_R$
α	N/A	$\frac{V_R t_{go}}{2000}$	$\frac{V_R t_{go}}{2000}$
β	N/A	$\frac{\pm 20}{V_R}$	$\frac{\pm 15}{V_R}$
D	N/A	3.6	3.0
β_1, β_2	0.01	0.01	0.01
$\varepsilon_P, \varepsilon_Y$	0.005	$\frac{\pm 12}{V_R t_{GO}}$	$\frac{\pm 10}{V_R t_{GO}}$
ϕ'	$\frac{0.005 V_R t_{go}}{D}$	$\frac{\pm 12}{D}$	$\frac{\pm 10}{D}$

LEGEND:

V_R - Relative velocity

α - Angle between miss distance vector and missile centerline

β - Angle between V_R and missile centerline

D - Miss distance

β_1, β_2 - Yaw and pitch components of the long range line of sight gimbal angles

$\varepsilon_P, \varepsilon_Y$ - Pitch and yaw components of the boresight error

ϕ' - Polar angle describing intersection of relative velocity vector with plane normal to missile centerline

The best algorithm for a system is therefore determined by the type of information available and its accuracy. The two algorithms presented for computer sizing purposes were selected to provide two viable approaches.

Table 28 shows the improvement in lethality achieved in progressing from a system without a TDD i.e., guidance only, to a TDD incorporating a simple time delay as a function of closing velocity and finally a system using the second time delay algorithm described in the following paragraphs.

TABLE 28

WARHEAD LETHALITY VS FUZE COMPLEXITY

Fuze/Miss Distance (feet)	50	100	150
(meters)	15.2	30.5	45.7
Guidance Only	0.42	0.11	0.05
Guidance, TDD & Simple Time Delay	0.58	0.23	0.09
Guidance, TDD & Complex Time Delay	0.73	0.32	0.16

Program Module E-1

For missile guidance systems which provide the parameters V_K, α, β and D, (listed in Table 27), complemented by the constants given in Table 29, the algorithm shown in Figure 64 optimizes the timing of the firing command.

TABLE 29

TIME DELAY PROGRAM MODULE F-1 CONSTANTS

ω	-	Fuze detection angle
V_P	-	Fragment velocity
K	-	Target size figure
θ	-	Fragment throw angle

Figure 65 illustrates the intercept geometry associated with Module F-1.

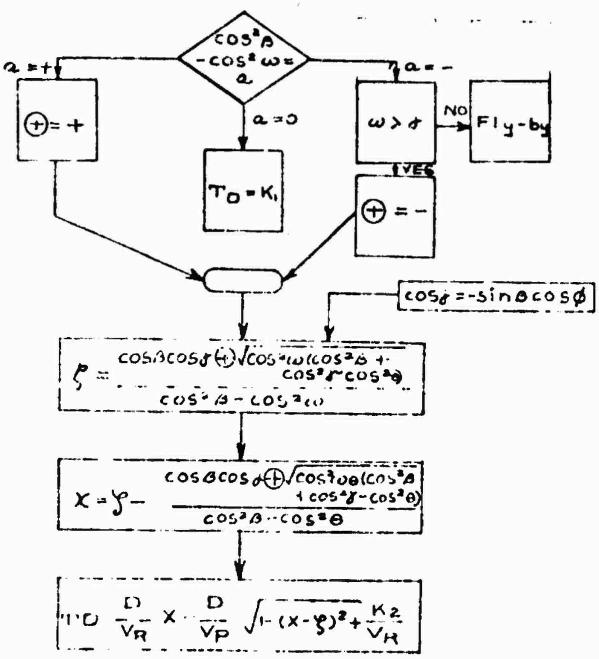


Figure 64 Time Delay Program Module F-1, Flow Diagram

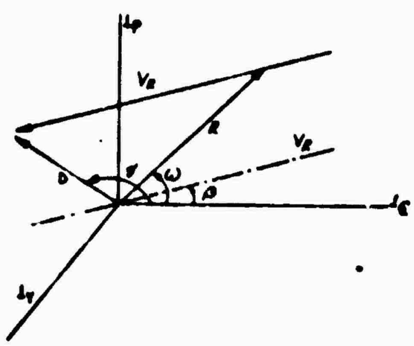


Figure 65 Module F-1 Intercept Geometry

Program Module F-2 -

For guidance systems providing V_R, D, β_1, β_2, ε_P and ε_Y (Table 27) as real-time inputs together with the constants V_P and K (Table 29), the algorithm shown in Figure 66 would provide improved lethality for a Class II or III missile. Figure 67 illustrates the intercept geometry associated with Module F-2.

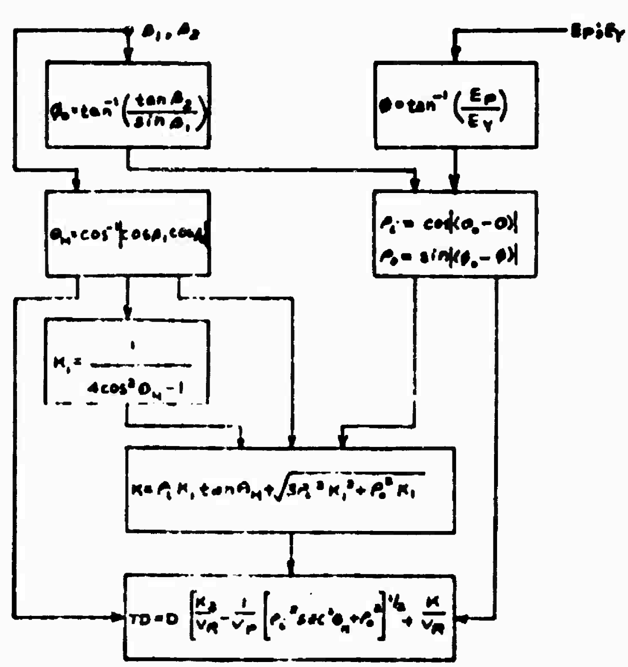


Figure 66 Time Delay Program Module F-2, Flow Diagram

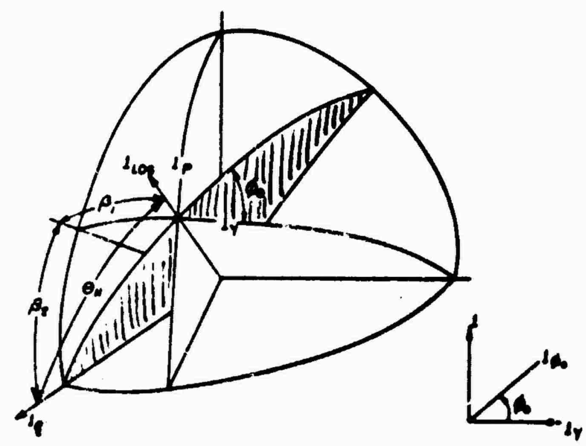


Figure 67 Module F-2 Intercept Geometry

4.6.4 Computer Requirements Summary

Table 30 lists the computer requirements for each time delay module described in the preceding subsection, together with coordinate transformation algorithms for interfacing the fuze with the missile guidance system. Both time delay modules require only a small program (< 128 words), while data base requirements do not exceed 12 words. The need for the guidance interface modules (FX1, and FX2) would be subject to further optimization and integration of the two coordinate systems.

TABLE 30

FUZING

COMPUTER REQUIREMENTS SUMMARY

MODULE NAME		ADD/ SUB	MULT/ DIVIDE	LOAD/ STORE	UTILITIES	*EQUIV ADDS	MEMORY *PROGRAM	DATA RAM	ROM
F1	Time Delay 1	17	15	56	3-COS 3-SQRT.	579	114	5	7
FX1	Co-Ordinate Transformation for F1	6	24	58	3-SQRT 2-ATAN 2-VEC.ROT.	948	114	0	5
F2	Time Delay 2	9	38	89	4-COS 2-TAN 1-SIN 2-ATAN 1 A COS 2 SQRT	1302	140	3	3
FX2	Co-Ordinate Transformation for F2	3	10	21	2-SQRT	231	44	0	4

LEGEND:

*Totals include additional 30% short operations for subroutine linkages and other miscellaneous overhead instructions.

4.7 Mode Control

For the purpose of this study, mode control refers to the selection and execution of a specific set of missile control functions (e.g., seeker head control, estimation, guidance, etc.) to meet the performance requirements of a specific/distinct operational phase. As such, mode control is more applicable to the single computer missile system where all missile functions must be executed sequentially yet still in accordance with the system sampling and computational time delay constraints.

In contrast, totally distributed computer systems (Section 7) are characterized by the assignment of essentially autonomous functions to separate dedicated processors, eliminating the need for function selection control, (Ref. 1).

For the single computer case, Table 31 illustrates the various function mixes required for each operational mode of a Class II missile with the initiating conditions for each mode shown in Table 32. Hence, for the single computer system, conventional real-time computer programming practice applies, where a real-time executive program is used to monitor real-time events/program interrupts, resolve priority conflicts and maintain smooth system operation to meet the mission operational requirements

TABLE 31

MISSILE CONTROL MODE FUNCTION MIXES

CLASS II MISSILE

	CONTROL MODE					
FUNCTION	TEST	INITIALIZE	TGT ACQN.	LAUNCH	MIDCOURSE	TERMINAL
Head Aim		X				
Track & Stabn.			X	X	X	X
Radome Compensation					X	X
Estimation				X	X	X
Guidance					X	X
Attitude Reference		X	X	X	X	X
Autopilot				X	X	X
Sensor Sig. Proc.			X	X	X	X
Fuzing						X
Telemetry (Flight Test)	X	X	X	X	X	X
Test	X					

TABLE 32

MISSILE CONTROL MODE INITIATING PARAMETERS/EVENTS

CLASS II MISSILE

CONTROL MODE	PARAMETER/EVENT	SOURCE
Test	Power-on	Launch aircraft
	Command	Missile Control set/unit
Initialize	Tests complete and satisfactory	Missile computer test program
Target Acquisition	Head-aim satisfactory	Launch aircraft missile control set/unit
Launch	Umbilical separation (Interlock broken)	Umbilical (Pilot controlled)
Mid-Course	Elapsed time from launch	Missile interval timer
Terminal (including fuzing)	Elapsed time R_{MT} or t_{go}	Missile interval timer and guidance program

Figure 68 illustrates a modular hierarchical programming structure for the single computer missile system where calls are made downward to subordinate program modules to select and execute the functions pertinent to the active missile mode. The net result of this mode control function is the calling and configuration of specific function timing templates as shown in Figure 69 in response to mode initiating real-time events. These templates are structured into groups of minor intervals (see time line analysis of 6.2.2) corresponding to the shortest data sampling/update interval e.g., stability loops. A complete template being determined by the longest data interval required in the function mix e.g., guidance.

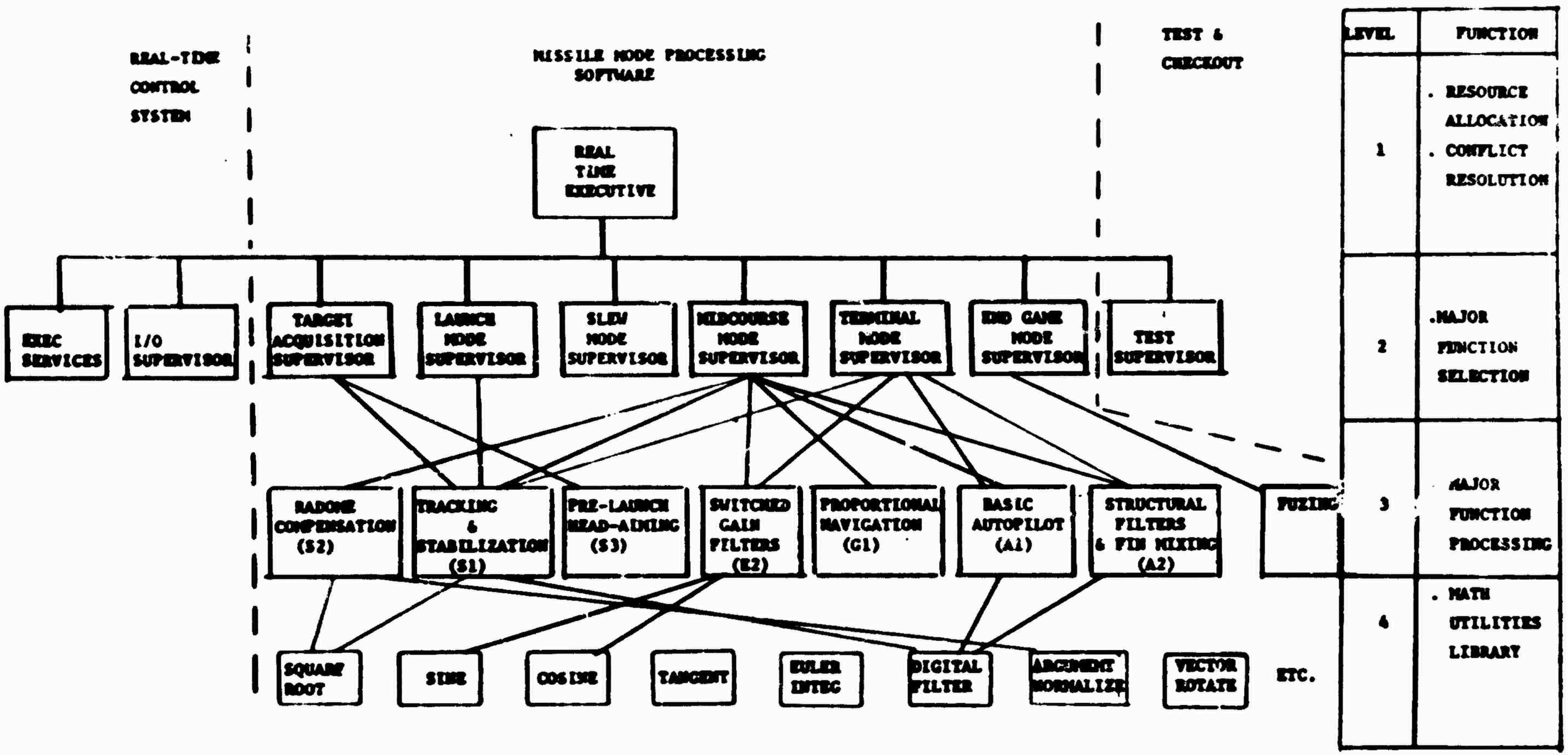


Figure 68 Missile Mode Control, Single Computer System, Modular Hierarchical Control Stucture

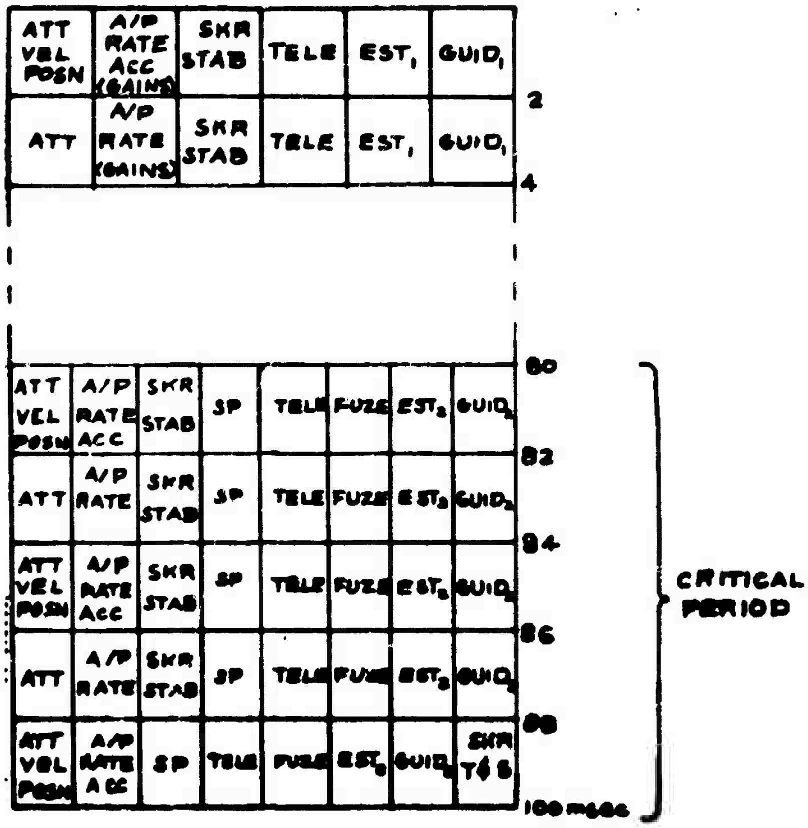
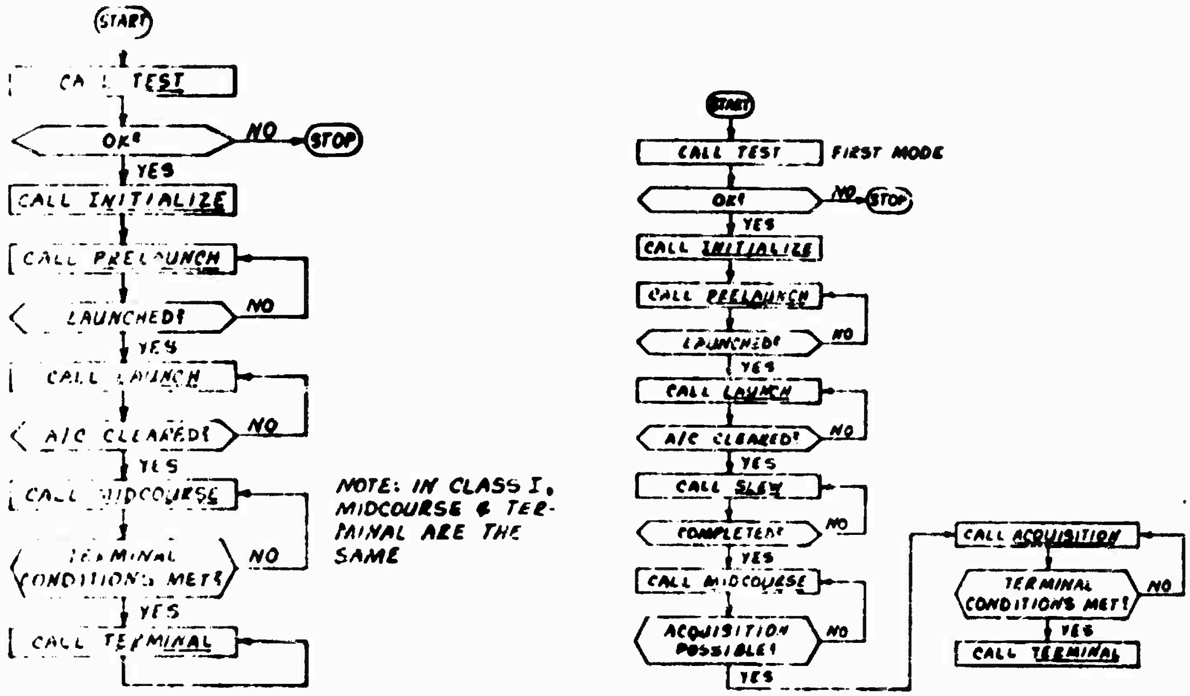


Figure 69 Typical Function Mix Template, Midcourse Mode

4.7.1 Executive Programs

Figures 70(A) and (B) are first level flow diagrams illustrating the executive control function for Class I, II and III missiles. Calls to mode supervisor programs result in these subordinate modules calling for missile guidance modules (e.g. S1, S2, A1, G1 etc.) as defined in the Phase I Final Report. The executive and supervisory programs are purely logical in nature and hence require only "short" computer instructions. Both programs are fixed and stable for a given missile class and would therefore reside in program/read-only memory.



(A) Class I and II Missiles (B) Class III Missile

Figure 73 Executive Control Flow Diagrams

Memory and throughput requirements are determined in the following subsections and summarized at the end of this section of the report.

4.7.2 Mode Supervisor Programs

Mode supervisory programs responsible for templating the missile guidance functions are described in more detail in the following paragraphs.

__Test Mode__ - Figure 71 is a flow diagram of the Test Mode Supervisor for either a Class I, II, or III missile. Each test subroutine (See Section 4.9) is itself a block of code executed sequentially, and the call to the routine is merely a jump to the first location in the sequence.

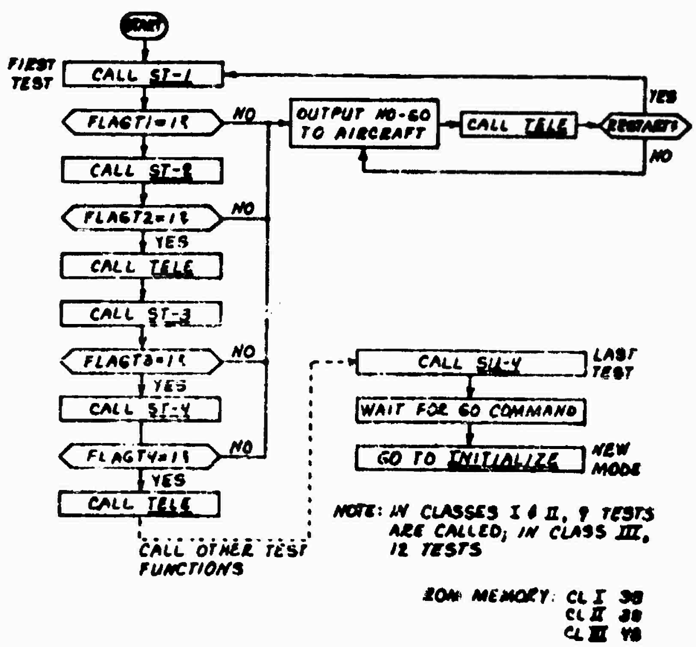


Figure 71 Test Mode Supervisor, Flow Diagram

At the time of the jump, the supervisor stores its current instruction address to return to the program when the test subroutine has been executed. This causes the supervisor to check a flag generated by the test routine, and if the test has been successful the next test is called. Failure to pass a test causes a specific external status line to be set which alerts the launch aircraft missile control set to the no-go situation and sets the computer into a telemetry only mode until it is either shut down or a restart from the aircraft is received.

After every two tests are passed, the telemetry program module (Section 4.8) is called so the state of the system can be assessed on the launch aircraft. Since the last test in the sequence is a telemetry test, the computer idles until an interrupt, generated by either the aircraft computer or by test personnel, is received, indicating that these external monitors are satisfied with the system operation up to this point. Control is then passed to the executive which calls for the Initialize Mode Supervisor.

Thirty eight words of ROM program memory are needed for the Test Mode Supervisor of a Class I or II missile, while forty eight are needed for Class III. Note that there is no inherent throughput requirement on this mode, as each test is called only once; the requirement is dictated by the amount of time allotted to the overall test function.

Initialize Mode - Once the Test mode is completed and the computer and subsystem are known to be working properly, the executive calls the Initialize Mode Supervisor to ready the system for launch. Several control flags are set to zero, and the computer inputs the required data from the aircraft (see Figure 72). When this is completed, the missile fins (wings, tails) are commanded to zero degrees so unwanted aerodynamic moments will not be induced on the missile during launch.

The only expected differences between missile classes in the initialize mode are the number of inputs to be read over the missile/aircraft umbilical. These inputs are listed in Figure 72 and result in the memory requirements shown for control of this mode. Again, computer throughput requirements are not affected by the initializaiton procedure, other than that all pre-launch functions must be accomplished within some predefined time interval. This interval should always be long enough that its impact on computer speed is at most minimal, i.e. it should not drive the computer performance and hence the size, weight and power consumption of the machine.

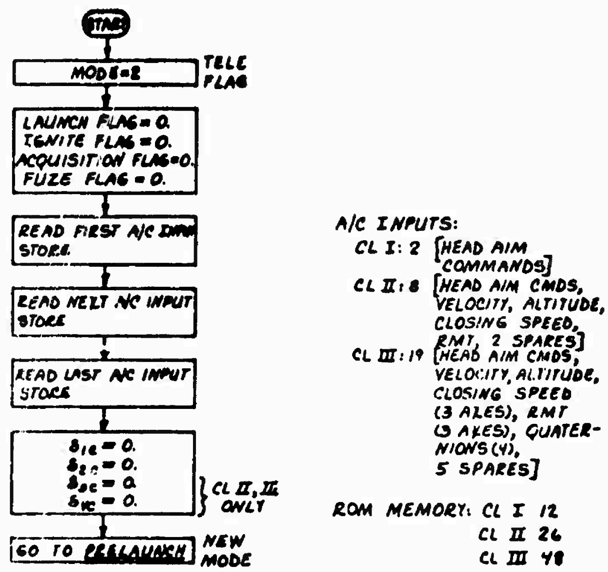


Figure 72 Initialize Mode Supervisor Flow Diagram

Prelaunch Mode - This mode takes the missile from initialization to launch, as shown in Figures 73 through 75. In Classes I and II, three distinct sub-modes can be identified in prelaunch: pre-acquisition, acquisition, and post-acquisition. The first of these in a Class I missile (Figure 73) calls for head-aim and telemetry functions, with gimbal angle commands being received over the umbilical and compared with gimbal angle readouts. Note also that fin commands, zeroed during initialization mode, are repeatedly sent out in order to overcome any possible D-A drift.

Head-aim continues until the seeker is pointed to within some predetermined error from the commanded angle. Until this occurs, the mode supervisor keeps calling the head aim sequence, as shown in the figure.

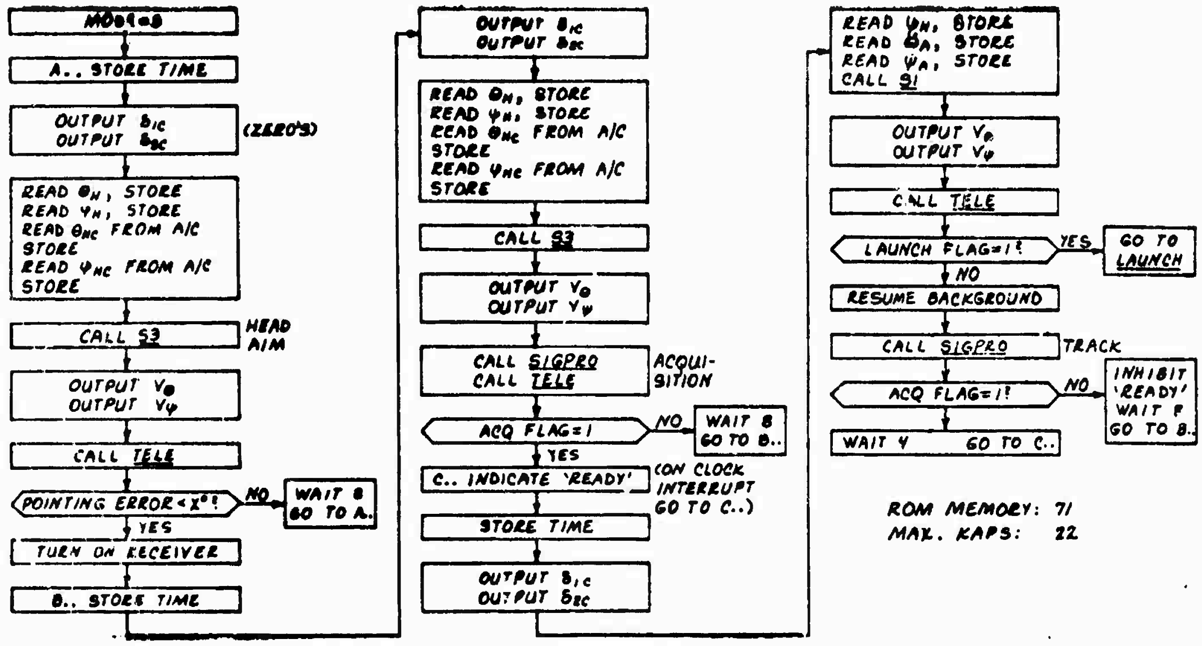


Figure 73 Pre-Launch Mode Supervisor, Flow Diagram,

Class I Missile

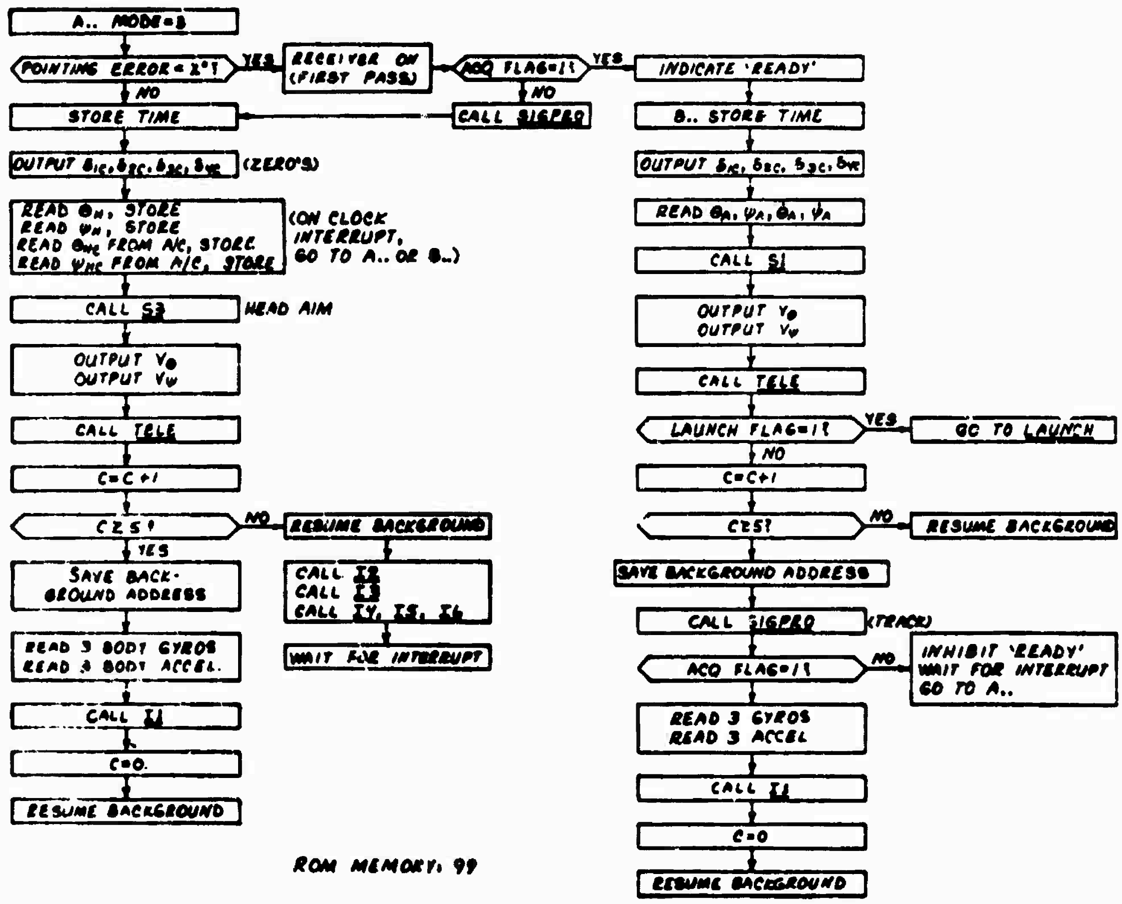


Figure 74 Pre-Launch Mode Supervisor Flow Diagram,

Class II Missile

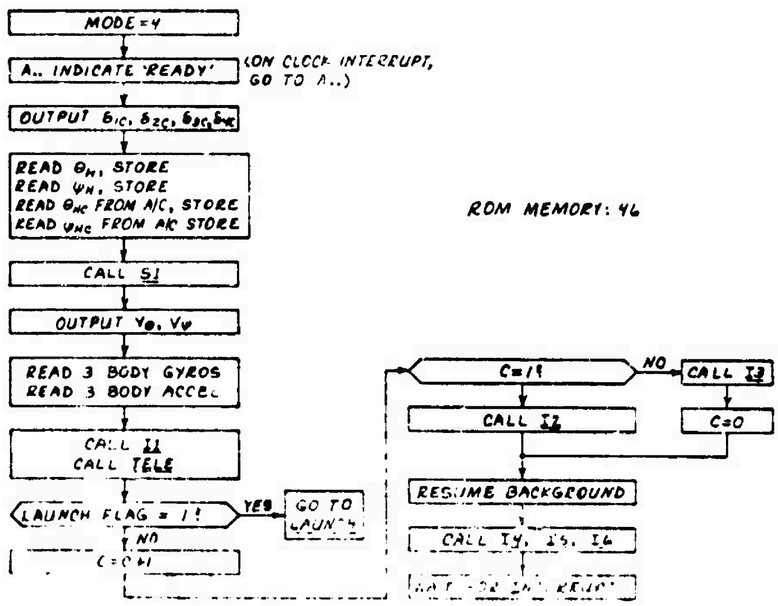
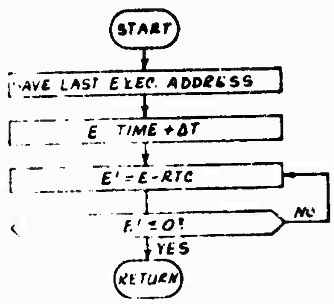


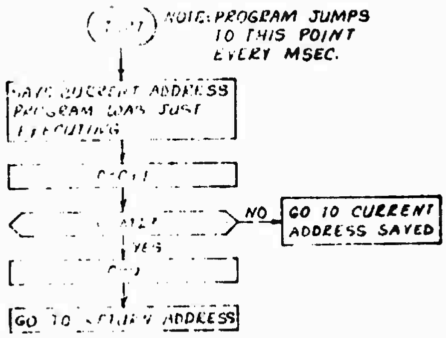
Figure 75 Pre-Launch Mode Supervisor, Flow Diagram, Class III Missile

Cycling is accomplished by means of a WAIT routine (see Figure 76) which allows the computer to idle until the required number of milliseconds (8, in this first case) have elapsed since the sequence was last started.

Once the pointing error is within limits, the acquisition submode is entered, similar to head-aim except that acquisition signal processing is called after the Head-Aim program module. (For clarity in these mode diagrams, all signal processing is identified by the call SIGPRO. The specific function being



(A) Wait Routine



(B) Real-Time Clock Interrupt Routine

Figure 76 Mode Supervisor Utilities, Flow Diagrams.

accomplished is evident from its location in the control stream). As Figure 73 indicates, it is assumed that all of the acquisition phase functions can be accomplished in an 8 msec interval, after which the phase recycles. If the signal processing operations overrun the interval, they can be handled as background/interleaved functions, as explained below.

When the target is acquired, the acquisition flag is set, after which the Head-Aim module (S3) is no longer needed. The seeker is then stablized along the line of sight by the Track and Stabilization routine, Module S1, called every 4 msec. Other functions performed during this phase include sending a ready indication to the aircraft, issuing zero fin commands, and telemetry. Also on every pass a check is made on whether the umbilical has separated (which sets the launch flag), in which case control immediately transfers to the Launch Mode supervisor. If this has not occurred, the track signal processor is called.

As the Class I controller is configured, track signal processing is treated as a background function. That is, it is not an operation that must be completed every 4 msec, as the SD and TELE functions must. Rather, if the calculations involved are going to overrun the basic minor interval (4 msec for Class I), the signal processing is interrupted and its current instruction address stored. The 4 msec routines are then re-executed, after which the tracking function resumes from where it had left off. That the function will be completed within its allotted update time is assured by designing sufficient throughput capability into the computer.

In general, all routines that run at a slower rate than the basic minor interval will be considered background functions, to be called as time permits. The interrupt routine, triggered by pulses from the real-time clock, is shown in Figure 4.7-9, and is used to control the interruption of the background calculations.

Note that, if the target has been lost, the supervisor reverts back to the acquisition, sub-mode.

The Pre-launch supervisory program for a Class II (max.) missile is shown in Figure 74, and is more complex than the Class I supervisor because of the attitude reference updating that begins immediately after initialization (see Figure 5.2-5 of the Phase I report).

While the basic minor interval is now 2 msec, missile attitude is updated every 10 msec; and velocity, position, and aerodynamics estimates are updated every 50 msec. These last functions are considered background operations, while the 10 msec routine is triggered by a counter, C. Note that either the background or the attitude determination can be interrupted by the 2 msec clock.

Except for the added functions, the Class II mode control is essentially the same as Class I. In Class III, however (Figure 75), acquisition does not occur until after launch, so this phase is omitted from the Pre-Launch mode, accounting for the lower memory requirement in the Class III supervisor compared with Class II. Since the attitude reference functions operate at a higher data rate in Class III, they have been removed from the backgorund block and placed in the main control stream.

Launch Mode - The Launch mode supervisor for a Class II missile is shown in Figure 77. In this mode, the missile flies without guidance commands, the fins commanded to zero, until the launch aircraft is cleared, a duration of less than a half second. In each class, therefore, the supervisor is a repeat of the last phase of Pre-launch mode, with some additional functions added to prepare the missile for other flight phases. Specifically, in each background block, the guidance filters and autopilot gain selection routines are now included, so that their transient responses may die out before their outputs are needed for guidance.

Although only the Class II flow is shown on the figure, memory requirements for all three classes are given.

Slew Mode - Only Class III missiles execute an air slew, the control for which is shown in Figure 78. Missile motion is controlled by the slew algorithm (Module A3) and slew autopilot (Module A4); and the seeker continues to be aimed by the S3 module, driven by outputs from the attitude reference system. All other functions are the same as during Launch mode.

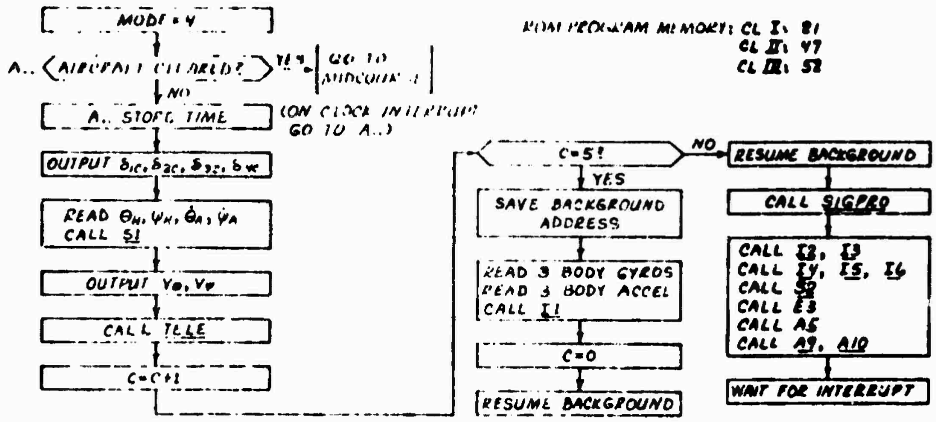


Figure 77 Launch Mode Supervisor

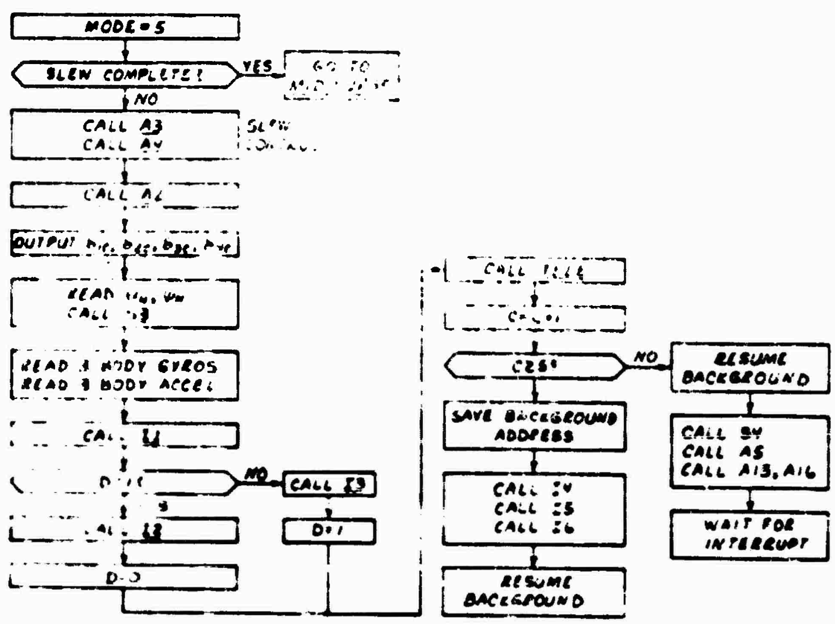


Figure 78 Slew Mode Supervisor, Flow Diagram

(Class III Missile only)

Midcourse Mode - In a Class I missile, Midcourse and Terminal modes are the same, so a separate Midcourse is not discussed. In Class II (Figure 79), Midcourse continues until either range-to-go or time-to-go drop below some specified value; proportional navigation guidance is used, the seeker is stabilized along the target line of sight, and autopilot and attitude reference functions are executed at their required rates. In short, all of the functions needed for guided missile flight are utilized in Midcourse.

The basic minor interval is 2 msec in Class II, and since the basic autopilot calculations need only be executed at a 250 Hz rate, these operations can be skipped on alternate passes through the supervisor routine. The attitude determination module, I1, which runs at a 100 Hz rate, is called by the counter, C, while all of the slower 50 msec routines are handled as background functions. As shown, 69 words of program memory are needed for control.

Note that, should the target be lost during flight, the executive does not respond by calling for the acquisition supervisor as in Class III missiles, since the desired line of sight angle is not known, but instead sets the warhead detonation flag to initiate missile self-destruct.

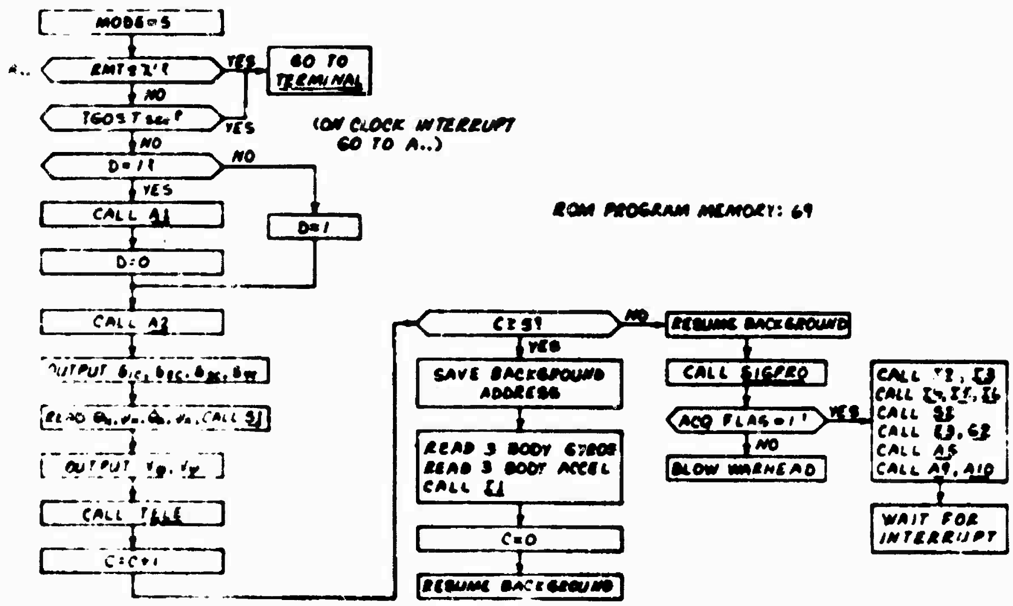


Figure 79 Midcourse Mode Supervisor, Flow Diagram (Class II Missile)

Class III Midcourse (Figure 80) is essentially the same as Class II, except that some of the update rates are changed. Also, guidance commands are not computed on board, but are received via an uplink. Seeker pointing commands are also received via the uplink (the target having not been acquired at this point) and the system uses the linear 9-state feedback program for stabilization. 69 words of memory are required for the supervisor.

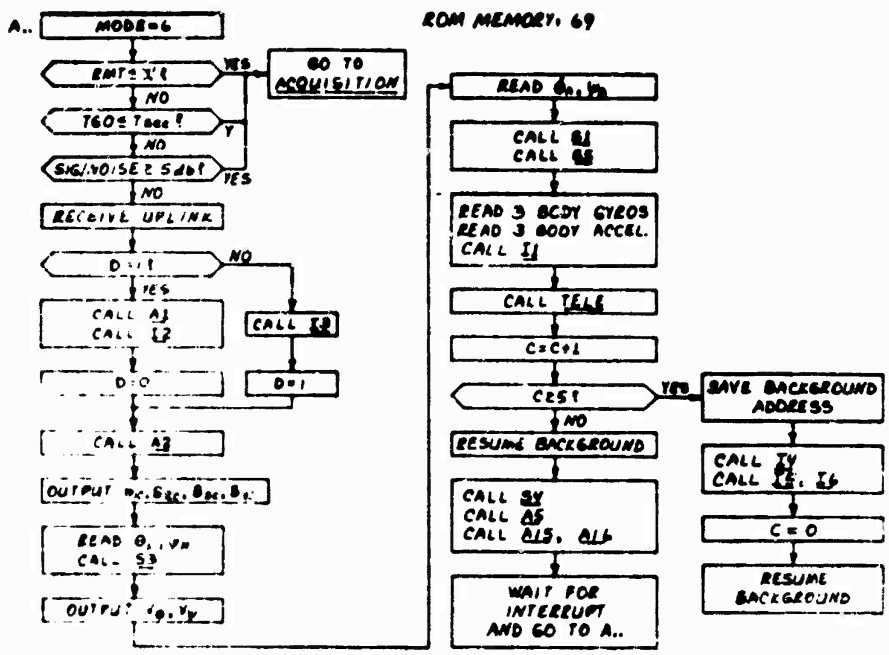


Figure 80 Mid-Course Supervisor Flow Diagram

(Class III Missile)

<u>Acquisition Mode</u> - This mode applies only to Class III missiles that acquire the target during Midcourse. It is in fact identical to the Midcourse mode except that in the background block the acquisition signal processor is called in addition to the other low update rate routines. Once the target is acquired, the executive transfers control to the Terminal Mode Supervisor.

Only 63 words of program memory are needed in this mode, a reduction from the Midcourse requirement because the tests at the beginning of Midcourse are no longer needed.

Terminal Mode - Figure 81 shows the Class I Terminal Mode Supervisor, including the call to the fuzing algorithm when estimated time-to-go drops below a specified value. The minor interval in this phase of flight is 4 msec, and seeker stabilization, autopilot, fuzing and telemetry functions are called at this rate. Note that, should the target be lost at any time, the warhead is detonated, self-destructing the missile.

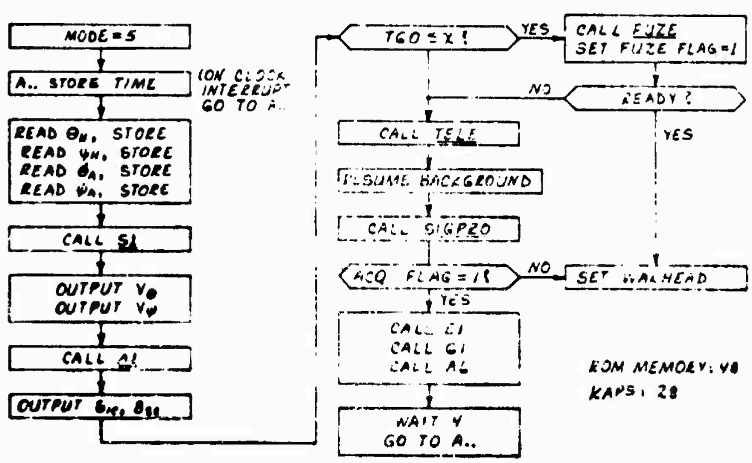


Figure 81 Terminal Mode Supervisor, Flow Diagram (Class I Missile)

The maximum throughput requirement for mode control is determined by the Terminal mode, since all of the missile functions are being performed at this time and the number of supervisor operations per minor interval are also a maximum. For Class I, 28 thousand equivalent adds per second are needed, along with 48 word of ROM memory.

The Class II and III missile Terminal Mode Supervisors are shown in Figures 82 and 83, respectively. They both have 2 msec minor intervals, and to save time the pitch and yaw basic autopilot functions (outer acceleration loop closures), which each run at 4 msec intervals, are split up and executed in alternate minor intervals. The Class II missile self-destructs when the target is lost, but the Class III executive transfers control back to the acquisition mode supervisor. Memory and throughput increase in the higher classes, keeping pace with the increasing number of subroutines that must be called.

4.7.3 Computer Requirements Summary

Table 33 summarizes the memory requirements for the mode control function as it applies to the single computer system, for each missile class. Including the two utility routines shown and a 30% increase for uncertainty, from 280 to 629 ROM words are needed.

Throughput in Kaps, including the 30% uncertainty factor ranges from 36 to 117. These must be included with all other missile functions when determining the maximum computer loads for a single central computer system.

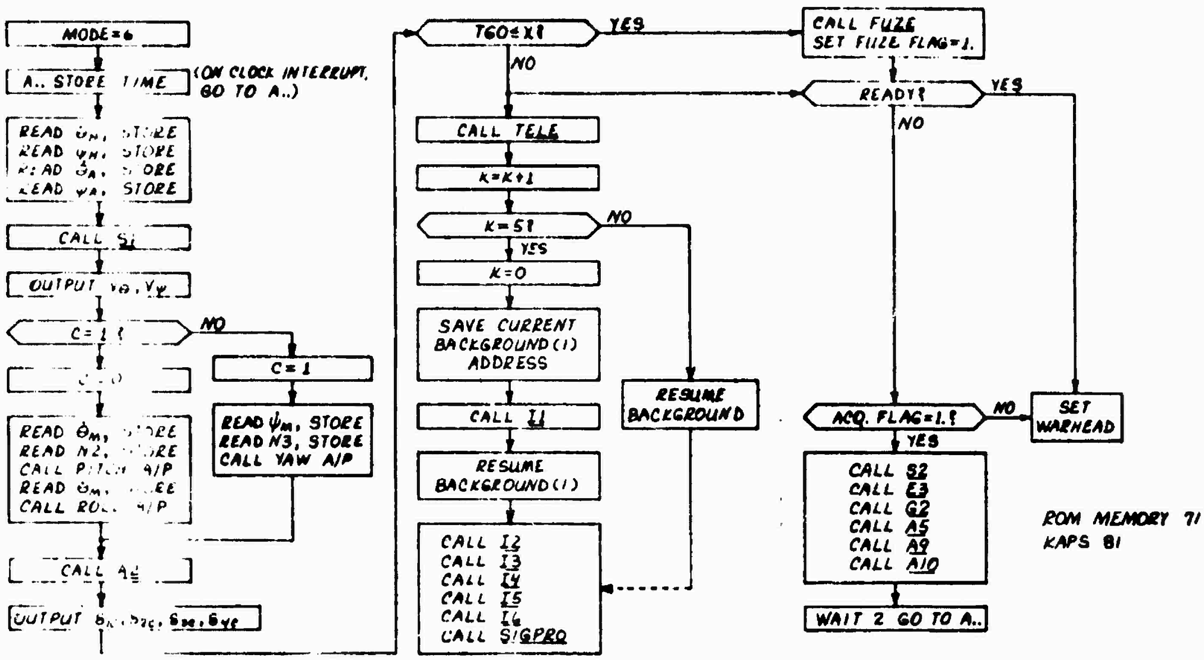


Figure 62 Terminal Mode Supervisor, Flow Diagram,

(Class II Missile)

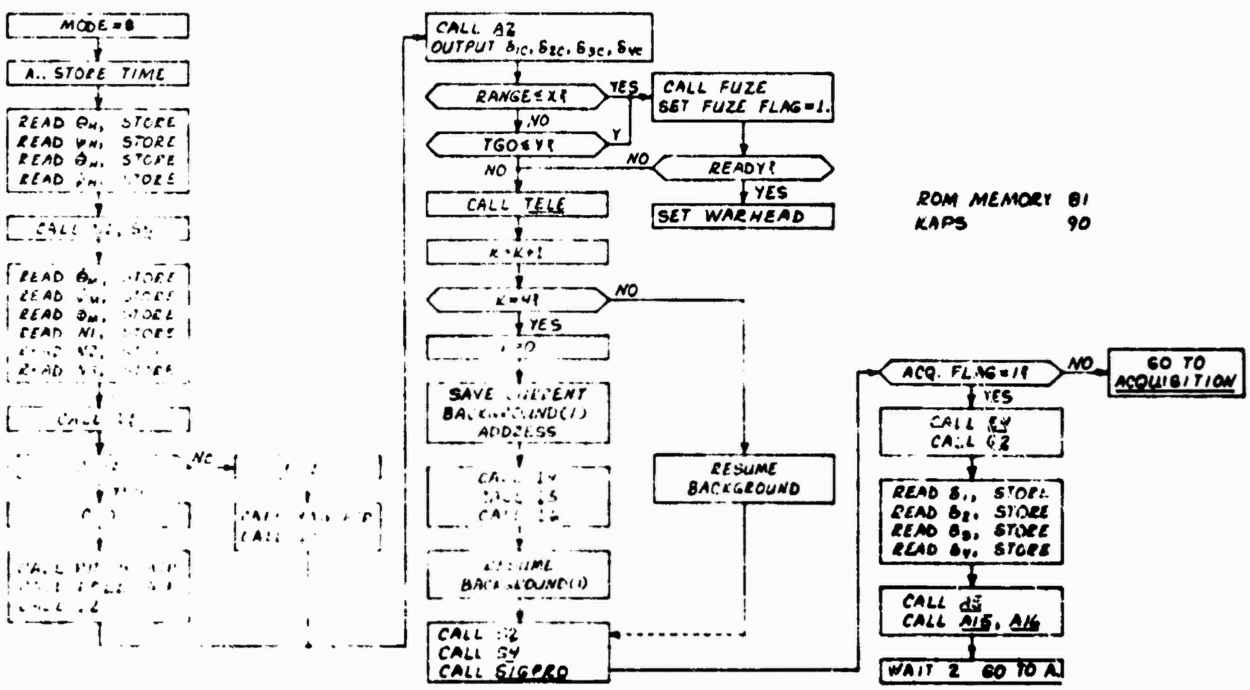


Figure 63 Terminal Mode Supervisor, Flow Diagram,

(Class III Missile)

TABLE 33

MODE CONTROL FOR SINGLE COMPUTER SYSTEMS

COMPUTER REQUIREMENTS SUMMARY

(PROGRAM MEMORY & THROUGHPUT)

	MISSILE CLASS		
OPERATING MODE	I	II	III
Test	38	38	48
Initialize	12	26	48
*Prelaunch	71	99	46
Launch	21	47	52
Slew	-	-	52
Midcourse	-	69	69
Acquisition	-	-	63
Terminal/End Game	48	71	81
Utility Routines (Interrupt, Wait)	25	25	25
Contingency (30%)	65	113	145
TOTALS Program Memory (Wds):	280	488	629
TOTALS **Throughput (Kops):	36	105	117

LEGEND:

* Classes I & II acquire in Prelaunch Mode. Class III acquires during Acquisition (Post-Launch) Mode.

** Terminal Mods, including 30% uncertainty factor.

4.8 Telemetry

The telemetering of missile performance data to ground-based receiving equipment becomes most effective when the missile is flown in an all-up tactical configuration and in a realistic intercept environment. Consequently, in a digital missile the on-board computer system must be capable of supporting telemetry throughout the test flights and without interfering with the normal guidance and control functions.

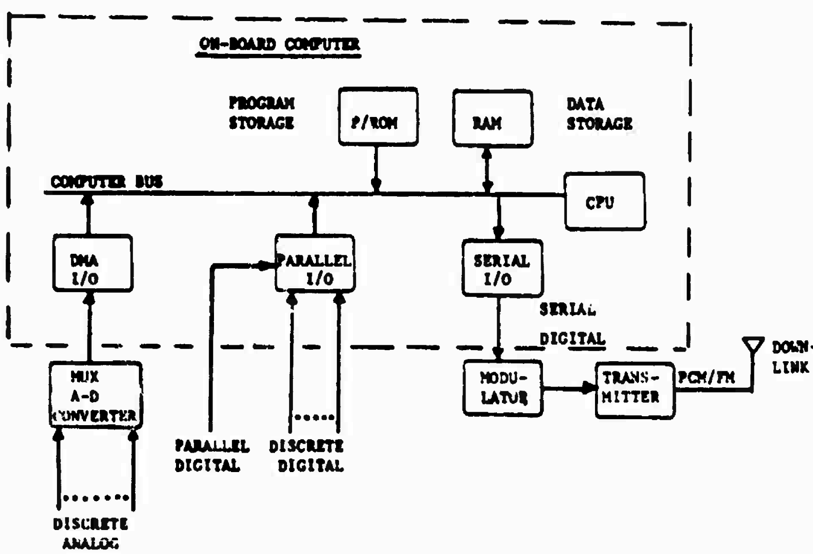


Figure 84 Computer Controlled Telemetry Data Acquisition/ Transmission System Block Diagram

Throughput and memory requirements for telemetry must therefore be added to the computer loads for missile guidance when specifying the computer requirements for a single computer system configuration, or, alternatively the telemetry load can be assigned to a separate microcomputer, dedicated to the telemetry function which in turn could be removed entirely from final production models of the missile.

Since the data to be downlinked is stored in computer memory as digital words, the simplest and most accurate means of transmittal to the ground equipment is serial pulse-code-modulation (PCM) using frequency modulation of a radio frequency carrier i.e. PCM/FM.

Figure 84 shows a modular on-board missile computer with data acquisition and transmission modules added for the telemetry function. Analog test data (e.g. battery voltages, analog pick-offs) are time multiplexed for sampling, A-D conversion and direct entry into assigned locations in computer memory (RAM). Parallel digital and discrete data sources (e.g. shaft-encoders and relays) are input to adjacent RAM locations via programmed input-output channels. The data to be telemetered should be ordered into contiguous locations for sequential access and output word-serial, bit-parallel to a buffered serial I/O channel which temporarily stores and converts each parallel computer word into a serial bit stream, adding message synchronizing, control and word parity bits. Completely formatted messages can then be output to a modulation module (I/2 modem) which converts the serial bit stream into a frequency modulated signal for transmission via the transmitter and antenna to the PCM ground telemetry facility. Figure 85 shows typical PCM telemetry formats for missile flight testing.

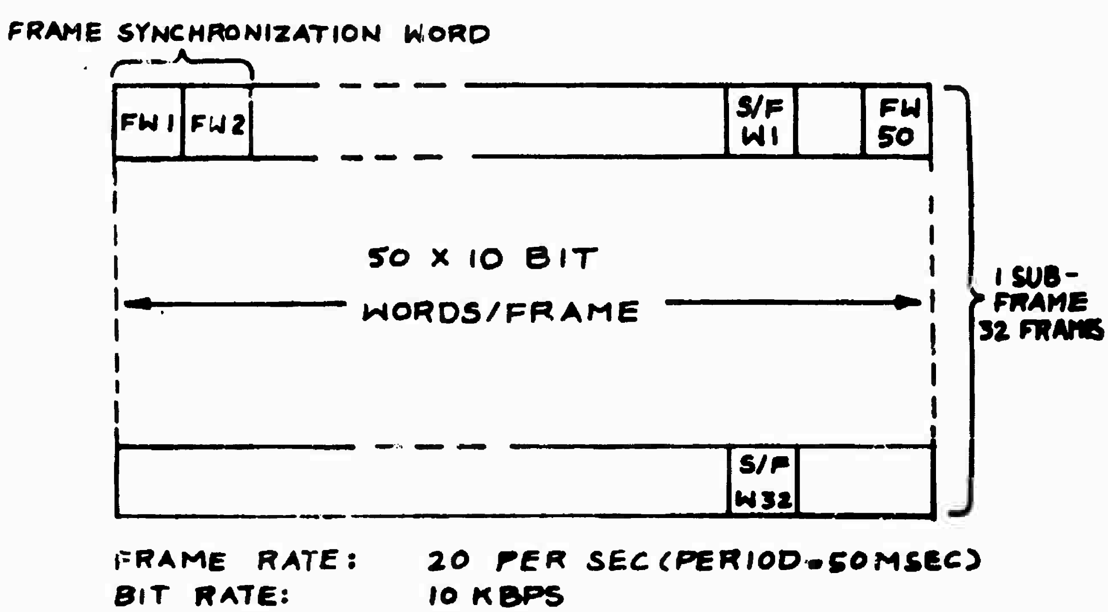


Figure 85 PCM Telemetry Formats

4.8.1 Telemetry Algorithms

In single computer system configurations telemetry is just one of many functions the computer must execute, and hence data transmission can not be performed continuously as in analog or federated/distributed computer systems. Telemetry data must therefore be arranged into blocks of words which can be sent out at a rate of one block every minor timing interval (see Subsection 6.2.2), and formatted with parity, identification (ID) and synchronization information to permit digital decommutation/demultiplexing by the ground equipment.

In a federated/distributed computer system a processor would be dedicated to the telemetry task and consequently the timing constraints are relaxed.

Since the single computer system presents the most critical case for computer sizing purposes, the following discussion will be confined to this method of implementing the telemetry function. Computer control of the telemetry function in a single computer system is performed as follows. During each minor timing interval, the active mode supervisor calls the telemetry subroutine which is indexed to transfer sequential blocks of data to the serial I/O channel. A beginning of block (BOB) address pointer (I) is used such that prior to exiting the telemetry subroutine, I is incremented, so that on the subsequent call a new block of data can be accessed.

The telemetry subroutine TELE is shown in Figure 86. Depending on the current mode of the missile (Prelaunch,

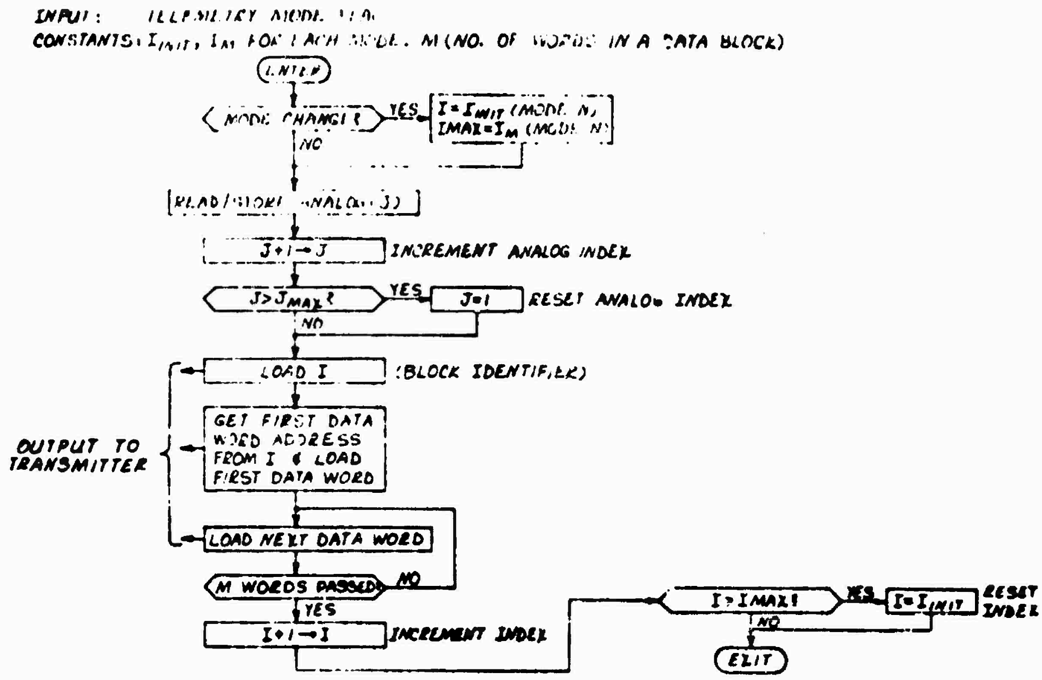


Figure 86 TELE Subroutine, Flow Diagram

Midcourse, etc.), a different initial value of the BOB address pointer I is chosen, as well as the final value the pointer will have within the particular mode, hence, the data block to be transferred can be changed from mode to mode, if desired.

A separate parameter, J, is used to keep track of analog input parameters not used in missile guidance computations but required for telemetering. The computer reads one of these voltages on each pass through the subroutine and stores the value in RAM memory from which it is called later for transmittal. Once read, therefore, an analog quantity can be treated as any other variable in a particular data block. As the TELE subroutine is configured, the same analog signals are read regardless of missile mode. This arrangement could be modified by a software change, if desired.

The first word output during each minor interval is the data block identifier, which is the current contents of the BDB address I pointer. The word contained in memory location I is the address of the first data point in the particular data block being called. The subroutine then automatically indexes through the string of variables until a programmed number, M, has been output, i.e. end of block. Following this, I is incremented so that in the next minor interval the next data block will be sent out. If all of the data blocks for a particular mode have been sent, the I is reset to its intial value for that mode, and the entire process recycles.

By organizing the required data into blocks, it can be readily changed from flight to flight and also from mode to mode during a particular flight, since the pointer register I points to the address of the first word in the data block, which in turn is used for "relatively addressing" the telemetry data.

4.8.2 Telemetry Data Lists

Tables 34, 35, and 36 list representative blocks of data by missile class. For Class I and minimum Class II missiles each data block is limited to 2 words, to minimize the computer load. For maximum Class II and for Class III missiles, telemetry data blocks contain 4 words each. The data shown in the tables is based on the algorithms used in each missile class, and these are listed in Chapter 5 of the Phase I Final Study Report. Two analog voltages are converted and telemetered for each actuator and gimbal and for the receiver electronics. Test flags included in the lists are internally generated indications of the status of test routines that the computer runs through prior to launch. These would only be telemetered prior to launch.

Although a certain number of data blocks are projected for each missile class, any particular block can be sent out at any desired rate. Assume, for example, that in a minimum Class I missile all 18 data blocks are to be sent out each major telemetry cycle, and that it is desired to transmit the antenna rates, block 15, twice as often as the other variables. If the 8JB addresses occupy location 100 to 117 in memory (corresponding to I = 100 to I = 117), then all that is necessary to send block 15 twice as often is to put the 8UB address for this block (e.g., the location of $\dot{\theta}_A$) in both locations 105 and 114, and to add location 118 to the memory requirements for the telemetry function.

TABLE 34

TELEMETRY DATA LIST

CLASS I MISSILE

BLOCK NO.	PARAMETERS	BLOCK NO.	PARAMETERS	BLOCK NO.	PARAMETERS
1.	Battery voltage Motor Fire volt.	9.	Receiver voltage (2)	17.	$\dot{\lambda}_p$ $\dot{\lambda}_y$
2.	Test Flag 1 Test Flag 2	10.	Fuze Parameter (2)	18.	N_{pc} (P.Acc.Command) N_{yc} (Y.Acc.Command)
3.	Test Flag 3 Test Flag 4	11.	A/P Band Mode	19	N_{LP} (P.Accelerometer) N_{LY} (Y.Accelerometer)
4.	Scculation Flag Umbilical Away	12.	θ_G (P. Gimbal ∠) ψ_G (Y. Gimbal ∠)	20.	$\dot{\theta}_m$ (P. Body Rate) * $\dot{\psi}_m$ (Y. Body Rate)
5.	Actuator #1 Volt (2)	13.	θ_c Command ψ_c Command	21.	δ_1 * δ_2
6.	Actuator #2 Volt (2)	14.	δ_1 Command * δ_2 Command		
7.	Pitch Gimbal Volt (2)	15.	$\dot{\theta}_A$ (P. Gimbal Rate) * $\dot{\psi}_A$ (Y. Gimbal Rate)		
8.	Yaw Gimbal Volt. (2)	16.	ϵ_P (P. Boresight Error) ϵ_Y (Y. Boresight Error)		

* Indicates Data Block is sent out at a higher rate (nominal x2)

NOTE: Blocks 19,20 & 21 are for maximum Class I only.

TABLE 35

TELEMETRY DATA LIST

CLASS II MISSILE

Blocks 1 through 21 of Class I Missile Data plus the following:

MINIMUM

BLOCK NO.	PARAMETER
1.	Actuator #3 volt. (2)
2.	$\delta 3$ Command *
	$\dot{\delta} 3$
3.	N_A (Axial Acc.) *
	$\dot{\phi}_m$ (Roll Body Rate)
4.	$\bar{R}$ (Measured RMT)
	$\hat{R}$ (Estimated RMT)
5.	$\hat{\dot{R}}$ (Estimated V_C)
	δR (Roll Fin Command)
6.	δp (Pitch Fin Command)
	δy (Yaw Fin Command)
7.	C_1 (A/P Gain)
	C_2 (A/P Gain)
8.	C_3 (A/P Gain)
	K (Filter Gain)
9.	G (Filter Gain)
	H (Filter Gain)
10.	ϵ_{rp} (Radome Comp.)
	ϵ_{ry} (Radome Comp.)
11.	a_{t_i} (Tgt. Acc.)
	a_{t_r} (Tgt. Acc.)

MAXIMUM

CLASS II (MIN.) DATA PLUS THE FOLLOWING:

BLOCK NO.	PARAMETER
12.	Actuator #4 Volt * (2)
	$\delta 4$ Command
	$\dot{\delta} 4$
13.	$\dot{R}$ (Measured V_C)
	M_{11} (Covariance Elem)
	M_{22} (Covariance Elem)
	M_{33} (Covariance Elem)
14.	ϕc (Roll Command)
	αp (P. Angle of Attack)
	αy (Y. Angle of Attack)
	q (Dyn. Pressure)
15.	4 (SA) Elements
16.	Missile Mass
	3 (MA) Elements

Notes: All Class II (Max.) Data Blocks contain 4 variables.

* Indicates data block is sent out at a higher rate (nominal x2)

TABLE 36

TELEMETRY DATA LIST

CLASS III MISSILE

Class I: Blocks 1 through 21 — Arranged in blocks of

Class II: Blocks 1 through 16 — 4 variables each

Plus the following:

MINIMUM

BLOCK NO.	PARAMETERS
1.	GLY (Seeker Comp.)
	GLP (Seeker Comp.)
	T_C (Seeker Comp.)
	TQI (Track Quality Indicator)
2.	4 calculated aero variables
3.	V_C (Guidance Gain)
	V_C' (Guidance Gain)
	P_1 (Guidance Gain)
	P_2 (Guidance Gain)

MAXIMUM

BLOCK NO.	PARAMETERS
4.	4 [P] elements of seeker quadratic control *
5.	K_1 (Slew A/P Gain)
	K_2 (Slew A/P Gain)
	U_2 (Slew A/P Gain)
	U_3 (Slew A/P Gain)

* Indicates data block is sent out at a higher rate (nominal x2).

4.8.3 Computer Requirements Summary

Memory requirements for the telemetry function are listed in Table 37. Data memory locations are included for each value of I, (assuming a software pointer) and locations must be added for those blocks that are sent out at a higher rate i.e. repeated. For the purposes of this study, those blocks listed with an asterisk in Tables 34 through 36 are assumed to be sent out twice as often as the other variables, resulting in the additional locations shown in Table 37.

In each telemetry mode, an initial and a final value of the BUB address pointer (I) must be provided, and in Missile Class I and II 3 telemetry modes are assumed: Pre-Launch, Pre-Acquisition; Pre-Launch, Acquisition; and Post-Launch. The first of these covers testing and missile checkout prior to target acquisition and the second covers all activity from acquisition to launch. A Class III missile breaks the Post-Launch mode into Launch/Slew, Midcourse, and Terminal modes, since in this Class each of these modes may have widely differing characteristics.

Lastly, space must be provided for each of the analog variables that are read into the computer prior to transmittal. These were listed in previous tables, and their number is shown in Table 37. The subtotal of RAM data memory locations is shown and 20% of this subtotal is added for contingency purposes. The TELE subroutine of Figure 84 requires approximately 50 words of program memory (ROM) resulting in the total memory

TABLE 37

TELEMETRY COMPUTER REQUIREMENTS (MEMORY)

MEMORY ASSIGNMENTS	MISSILE CLASS I MIN	I MAX	II MIN	II MAX	III MIN	III MAX
Data Blocks	18	21	32	21	24	26
Repeated Data Blocks	2	4	6	4	5	6
Telemetry Modes	3	3	3	3	5	5
(2 Locations/Mode)	(6)	(6)	(6)	(6)	(10)	(10)
Analog Sources	13	13	15	17	17	17
SUBTOTAL(RAM)	39	44	59	48	56	59
20% Contingency(RAM)	8	9	12	10	12	12
Programs (ROM)	50	50	50	50	50	50
TOTALS RAM:	47	53	71	58	68	71
ROM:	50	50	50	50	50	50

requirements shown in the table.

Using the smallest intervals shown in Chapter 5 of the Phase I Final Report and assuming 16-bit words are downlinked, bit rates from 12 Kilobits to 40 Kilobits per second are required depending on missile class, (Table 37).

The telemetry subroutine, except for the read function, is composed of short operations (load, jump, etc.). Analog data inputs are assumed to take 10 add times each.

TABLE 38

TELEMETRY COMPUTER REQUIREMENTS (THROUGHPUT & DATA RATE)

	MISSILE CLASS					
	I		II		III	
PARAMETER	MIN	MAX	MIN	MAX	MIN	MAX
Minor Interval (msecs)	4.0	4.0	2.0	2.0	2.0	2.0
Computer Wds/Minor Interval	3	3	3	5	5	5
Serial Bit Rate (Kilobits/Sec)	12.0	12.0	24.0	40.0	40.0	40.0
Data Repetition Interval (msecs)	72.0	92.0	72.0	46.0	54.0	60.0
COMPUTER THROUGHPUT (Kaps)	16.0	16.0	32.0	36.0	36.0	36.0

4.9 Test

For the purposes of this study missile readiness tests encompass the on-board computer(s), guidance and control programs, input-output interfaces, telemetry, seeker and missile control servos. Such tests are performed by the execution of test program modules in the on-board computer(s) prior to missile launch and in response to a command from the aircraft avionics/central integrated test subsystem (CITS) computer via the umbilical interface. Test programs are therefore assumed to be executed off-line without severe timing constraints with memory requirements becoming the chief consideration.

In the case of a single computer system the execution of all test programs depends upon the serviceability of the missile computer, specifically: the I/O channel, CPU, program memory, data memory and power supply. A single failure in any one of the latter computer components would therefore inhibit missile subsystem tests.

For federated/distributed computer systems the avionics-missile test command would be distributed to each subsystem computer, such that, a computer failure would be synchnomous with a specific subsystem failure, (e.g. radar sensor, IR sensor, guidance, autopilot, fuze, telemetry), thereby isolating a fault to a line replaceable unit (LRU).

In all cases, test result reporting is in the form of a "go/no-go" indication to the launching aircraft avionics/CITS computer.

4.9.1 Computer Self-Tests

Instruction Execution Test- The first computer self-test is an operational check of the instruction set, using the central processing unit (CPU) and main memory. Operands with predefined bit patterns, such as all 1's or alternating 1's are used to ensure that subtle failure modes are not present in either the CPU or in memory transfers. Figure 87 is a flow diagram of the instruction test module, ST-1.

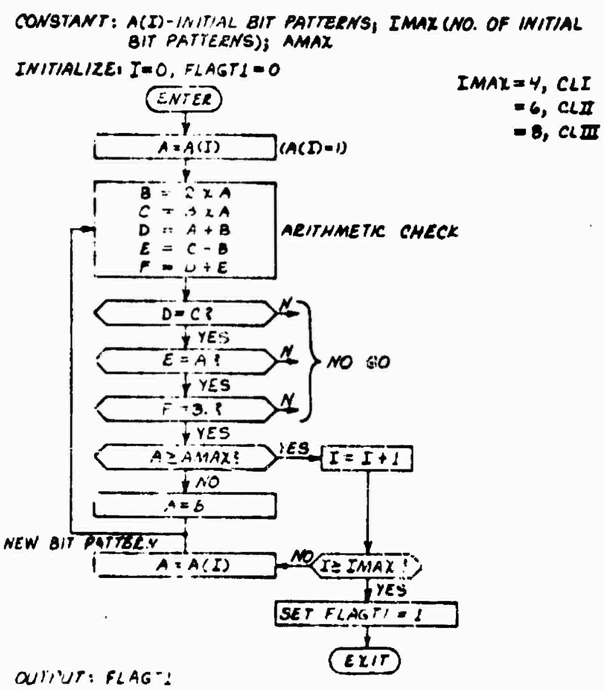


Figure 87. Computer Instruction Test Program Module ST-1, Flow Diagram

Four basic arithmetic operations are performed on the operand A, which could be unity on the first pass through the routine. Pre-computed results are stored in main memory to verify each test result. On subsequent passes, the single binary "1" bit is placed in increasingly significant bit positions of the A word, and the arithmetic operations are rechecked. When each bit location has been exercised, two "1" bits (e.g., binary 101) are placed in A, and the entire test recycles. In addition to arithmetic and logical operations jump instructions, indirect addressing, and shift operations are checked in the process, prior to any other tests being conducted. The output of this test is a go/no-go flag indicating whether or not the test was successfully completed.

Module ST-1 occupies approximately 50 words of read-only program memory with read/write memory being used for intermediate data storage. For general register machines the entire test would be repeated using different operational registers and associated instructions.

Data memory requirements for module ST-1 are 10 plus 1 for every initial value of A. Requirements for this and other modules are summarized in subsection 4.9.4.

Program Memory Test - In read-only or programmable read-only memory, (P)ROM, tests, programs table-look-up and discrete constants are checked by adding each word successively in the CPU and comparing the total sum with a "check-sum" word which is the last word in memory. Figure 88 is a flow diagram of the program memory test module ST-2. This routine is common to all missile classes since it is a repeated do-loop.

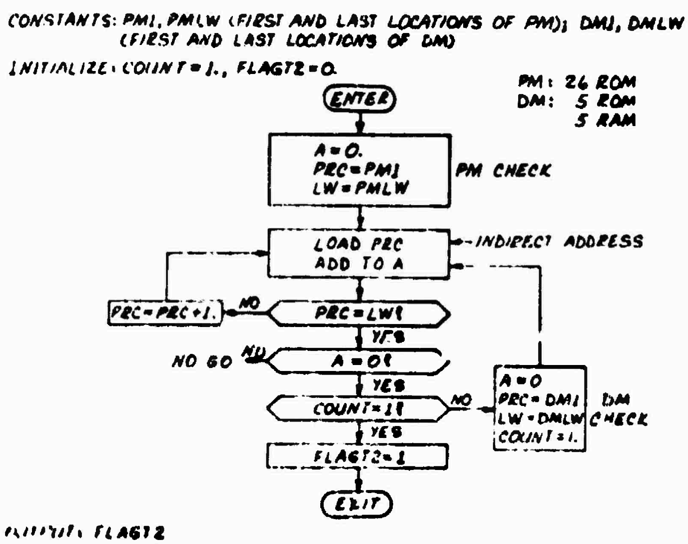


Figure 88. Computer Program Memory Test Module ST-2, Flow Diagram

Data Memory Test - Read/Write memory (RAM) is checked by first writing a specific data value in successive memory locations and then reading these out and checking against the fixed reference value for correctness. Figure 89 is a flow diagram of the data memory test module ST-3 which checks every memory location in this way, using data word A, which can be set at different values on successive passes through the routine for added confidence in memory operation. Approximately 30 words of program memory are needed for this routine.

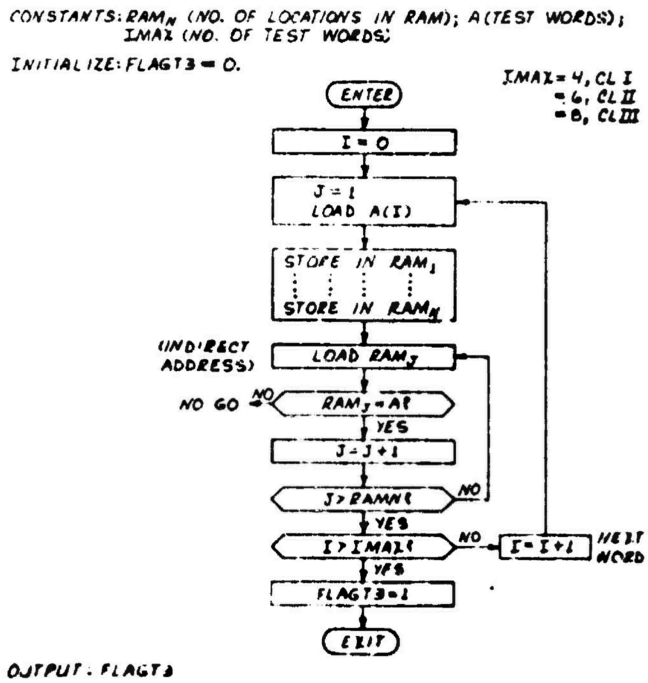


Figure 89. Computer Data Memory Test Module ST-3, Flow Diagram

Operational/Tactical Software Tests - Tactical software tests complement the previous test in that instructions and memory are checked simultaneously using actual operating subroutines. As an example the flow diagram for module ST-4 in Figure 90 defines a set of inputs for the basic track and stabilization program module S1 of a Class I missile (see Figure 4.1.3, page 4-12 of the Phase I Final Study Report) executing the module and comparing the outputs with the expected values which have been precomputed and stored in program memory. Obviously, this test can be run on one or all of the tactical subroutines, but if all of the previous tests have been successful it is only necessary to execute those modules which make use of a wide range of instruction words and use widely separated memory locations. For the purposes of this study, it is assumed that two subroutines are called for Class I missiles, three for Class II, and four for Class III. Further, an average of 6 set inputs and 4 expected outputs are assumed for each routine tested, leading to the requirements listed in subsection 4.9.4 for each missile class.

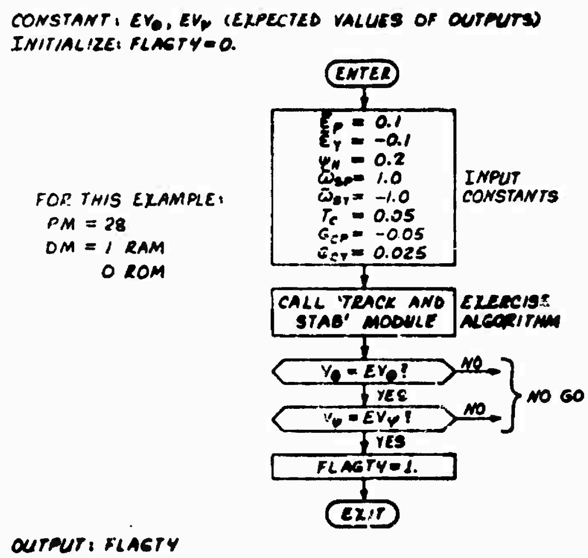


Figure 90. Computer Operational/Tactical Software Test Module ST-4, Flow Diagram

4.9.2 Computer Interface Tests

I/O Instruction Tests - The execution of computer input-output instructions and associated data transfers over the I/O channels are checked by test program Module IT-1, (Figure 91), using either an external addressable buffer or spare D-A and A-D channels connected back to back. Special bit test patterns are output as data words to the designated external device (buffer or D-A convertor) and transferred back from the buffer or via the A-D convertor for verification against the original test word. Assuming 5 different bit patterns are sent out and compared, memory requirements for the routine are approximately 20 for program and 6 for data.

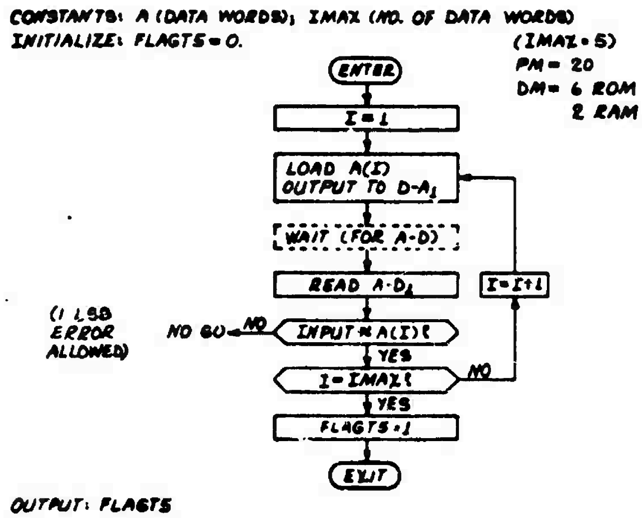


Figure 91. Computer I/O Instruction Test Module IT-I, Flow Diagram

Analog Multiplexer A-D Convertor Tests - This test is similar to the I/O instruction test, except that its purpose is to check-out analog multiplexer channels and associated A-D convertor. Module IT-2, Figure 92 is an expansion of IT-1 with more extensive bit pattern checks to ensure subtle failure modes are not present in the interface hardware. Single or multiple bits are placed in the test words A in the same manner as Module ST-1. Spare channels are assumed in each mulitplexer together with two dedicated D-A convertors. The data memory requirement is 9 locations (including 5 initial bit patterns of A), while the required program memory space is 30 words.

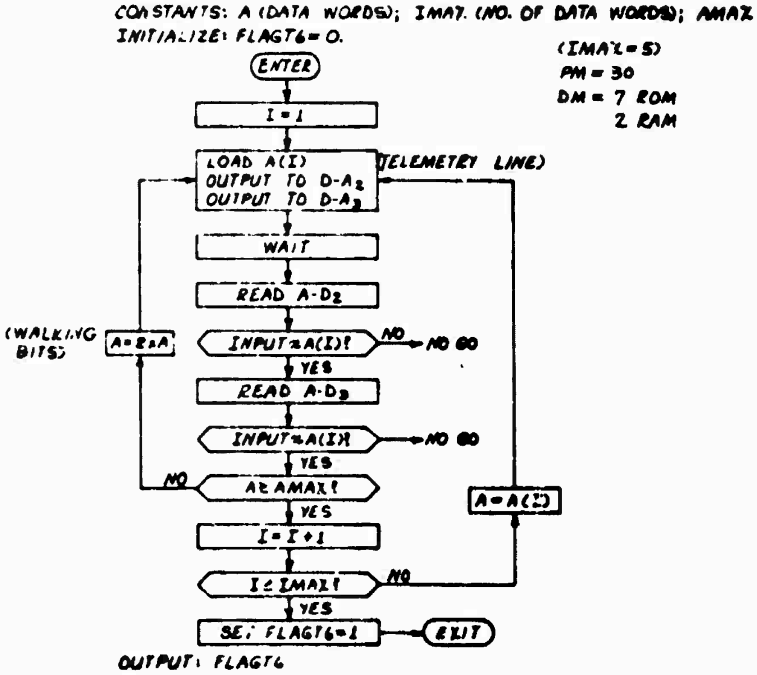


Figure 92. Analog Mulitplexer A-D Convertor Test Module IT-2, Flow Diagram

Analog Multiplexer A-D/D-A Convertor Zero Tests - An extension of the IT-2 module is required to verify that for zero digital commands to the seeker gimbals and control actuators the platform and fin positions are correspondingly at zero/undeflected. Module IT-3, shown in Figure 93, accomplishes the latter task. The amount of program memory needed depends on missile class, as shown. More extensive I/O testing of this nature can be accomplished with the subsystem tests discussed in subsection 4.8.3.

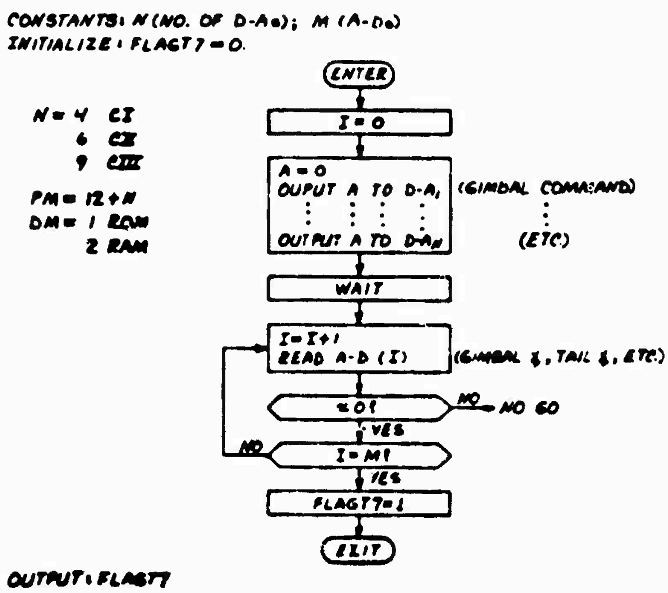


Figure 93. Analog Multiplexer A-D/D-A Convertor Zero Test Module IT-3, Flow Diagram

Umbilical Test - This test requires interaction with the aircraft avionics computer. The missile computer sends specific bit patterns to the avionics computer for retransmission back to the missile computer for verification. Module IT-4 performs this test, and is identical to IT-2. Program memory requirements total 25 locations.

4.9.3 Missile Subsystem Tests

These tests exercise the seeker gimbals control actuators, inertial sensors, and telemetry subsystems, and also serve as an additional check of D-A and A-D operation. In tactical situations requiring fast reaction times, the seeker and actuator tests would not be practical due to the response time of the servos.

Target Seeker Gimbal Tests - Module SU-1, Figure 94, outputs a step command sequence to each of the seeker gimbals in turn. As shown, the sequence, in degrees of gimbal angle, is 1, 0, -2, 0, 4, 0, -8, 0...etc., up to a maximum value IMAX. Note that the HEAD AIM subroutine (Module S3, Phase I Final Study Report) is needed for this test in order to close the gimbal servo loops, and that this subroutine must be called at 8-16 msec intervals as determined by the real-time clock. The acceptance or rejection of the gimbal response, however, is not made until a pre-determined time (DTIME) after the step command has been sent out. This delay time is a function of the step size, as indicated in the algorithm, and allows the gimbal sufficient time to reach the desired angle.

Program memory requirements total 40 locations for this module, and data memory requirements total 12.

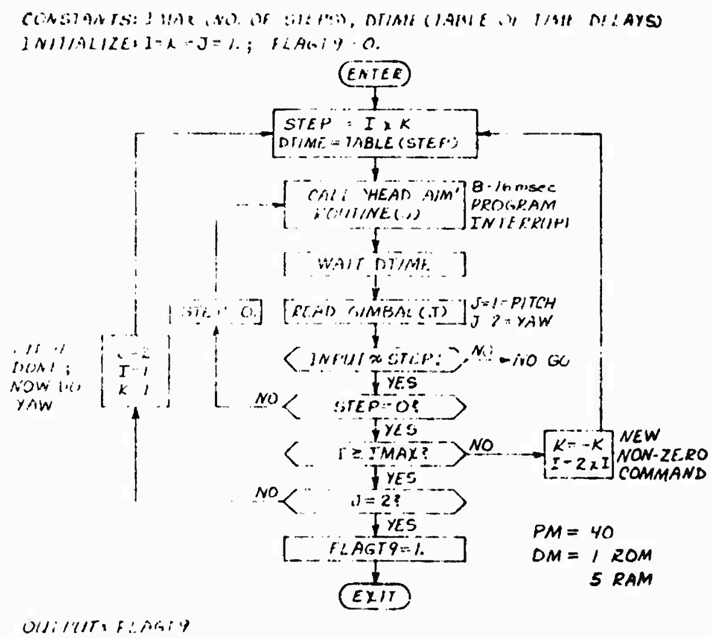


Figure 94. Target Seeker Gimbal Test Module SU-1, Flow Diagram

<u>Missile Fin Actuator Tests</u> - Control actuator tests are not practical for wing-mounted missiles due to the resulting moments applied to the missile and pylon. Class III missiles installed in a bomb bay could be sequenced through step response tests of various amplitudes (positive and negative) to insure all four fin actuators were functioning properly. Module SU-2 performs this test, and is similar to SU-1, except that four actuators are tested instead of two gimbals, and an actuator position loop closure routine is executed at a higher rate instead of the head-aim routine. Memory requirements are unchanged from Module SU-1, since the routine cycles through the same locations for each actuator tested.

Inertial Sensor Tests - Inertial sensor tests require interaction between missile and launch aircraft. The aircraft computer is required to transfer via the umbilical instantaneous values of aircraft acceleration and rotation rate. The missile computer program (Module SU-3, Figure 95), compares these with values input from the missile instruments using predetermined error margins.

To minimize these errors the time interval between aircraft instrument readings and missile readings should be small. It is assumed that any orientation difference between the aircraft and missile inertial systems are compensated for in the values transferred from the aircraft.

For the three accelerometers and gyros tested, the program memory requirements total 31 locations for Module SU-3, and data storage totals 5 locations.

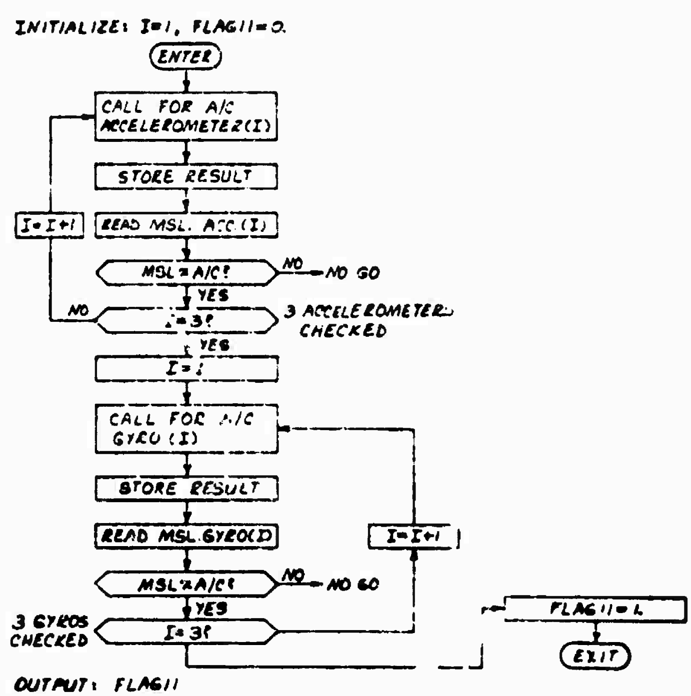


Figure 95. Inertial Sensor Test Module SU-3, Flow Diagram

Telemetry Tests - The telemetry function is checked by outputting a multi-word test message for verification by the aircraft computer and ground test personnel prior to missile launch. The telemetry test program Module SU-4 is shown in Figure 96, with the missile computer outputting the required data in the first part of the routine. The computer waits for an interrupt from the aircraft computer which confirms an error-free message and initiates the next missile mode. If a discrepancy in the received data is detected this interrupt would not be given.

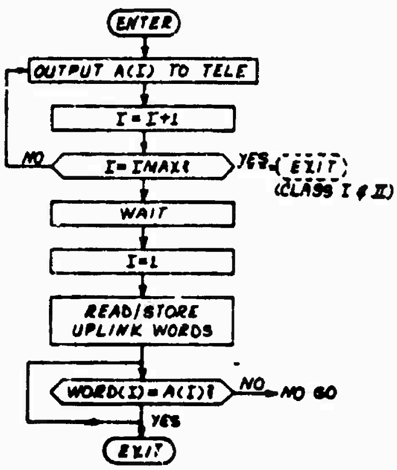


Figure 96. Telemetry Test Module SU-4, Flow Diagram

Tactically, the telemetry test would not be performed in a Class I or II missile. In a Class III missile, where a command link exists between aircraft and missile, the aircraft would transmit the test message back to the missile for verification and the data link established. This is shown in the second part of Module SU-4.

Assuming a 10 word test pattern, memory requirements for the first part of Module SU-4 are 7 locations for the program and 12 for data with the complete module requiring 24 programs and 22 data locations.

4.9.4 Computer Requirements Summary

A complete listing with summary descriptions of the missile test modules is given in Table 39. Memory requirements for the test algorithms are summarized in Tables 40, 41, and 42 for each missile class. Total memory requirements (programs and data) for test purposes range from 360 to 698 locations.

TABLE 39

MISSILE TEST PROGRAM MODULE LISTING

(A) SELF TEST

MODULE	NAME	FUNCTION	METHOD	MEMORY WDS
ST-1	Instruction Test	Check ALU, Addressing Jumps, etc.	Call and operate on various constants. Compare results with expected answers.	60-200
ST-2	Memory Test	Check ROM Memory	Add all PMEOM words together, check sum	40
ST-3	Memory Test	Check RAM Memory	Write words into RAM, read back out and check	50
ST-4	Software Test	Check operational algorithms	Call individual subroutines, using stored inputs: compare results with expected answers	60-125
			TOTAL:	210-415

TABLE 39 (CONTINUED)

(B) INTERFACE TESTS

MODULE	NAME	FUNCTION	METHOD	MEMORY WDS.
IT-1	Instruction Test	Check In-Out Instructions	D-A/A-D wraparound. Output special words, read them back in.	25
IT-2	Multiplexer Test	Check MPX and Operational A/D	Output special word patterns, read back in through spare MPX channel	40
IT-3	D-A/A-D Test	Check operational D-As and MPX channels	Output zero command to subsystems, read in responses	25
IT-4	Umbilical Test	Check data interface with A/C	Send test words to A/C, receive them back and check	40
			TOTAL:	130

TABLE 39 (CONTINUED)

(C) SUBSYSTEM TESTS

MODULE	TYPE	FUNCTION	METHOD	MEMORY WDS.
SU-1	Seeker Test	Check operation	Output step commands, read responses	50
SU-2	Actuator Test	Check operation	Same as above	50
SU-3	Inertial Test	Check operation	Get g and turn rates from A/C, compare with on-board instruments	50
SU-4	Telemetry	Check operation	Send test pattern out, wait for confirmation	20-50
			TOTAL:	170-200

TABLE 40

MISSILE TEST COMPUTER REQUIREMENTS

CLASS I MISSILE

MODULE	PROGRAM MEMORY (ROM)	DATA (ROM)	MEMORY (RAM)	EXECUTION TIME (msec)
ST-1	50	6	8	0.2
ST-2	26	5	5	9.0
ST-3	30	7	3	5.7
ST-4	62	0	1	1.0
IT-1	20	6	2	0.2
IT-2	30	7	2	1.9
IT-3	16	1	2	0.1
SU-1	40	7	5	*
SU-4	7	11	1	0.2
TOTALS:	281	50	29	18.3 (Plus Gimbal excursion time)

Notes (Tables 40, 41, & 42)

+ Assuming 1 μsec add time

10 μsec input/output time

* Several seconds needed for gimbal travel

TABLE 41

MISSILE TEST COMPUTER REQUIREMENTS

CLASS II MISSILE

MODULE	PROGRAM MEMORY (ROM)	DATA MEMORY (ROM)	(RAM)	EXECUTION TIME (msec)
ST-1	100	8	8	0.3
ST-2	26	5	5	23.1
ST-3	30	9	3	21.9
ST-4	93	0	1	1.5
IT-1	20	6	2	0.2
IT-2	30	7	2	1.9
IT-3	16	1	2	0.2
SU-1	40	7	5	*
SU-4	7	11	1	0.2
	-----	---	---	------
TOTALS:	364	54	29	49.3 (plus gimbal excursion time)

TABLE 42

MISSILE TEST COMPUTER REQUIREMENTS

CLASS III MISSILE

MODULE	PROGRAM MEMORY (ROM)	DATA (ROM)	MEMORY (RAM)	EXECUTION TIME (msec)
ST-1	150	18	14	0.4
ST-2	26	5	5	51.2
ST-3	30	11	3	36.5
ST-4	124	0	1	2.0
IT-1	20	6	2	0.2
IT-2	30	7	2	1.9
IT-3	21	1	2	0.2
IT-4	25	7	2	1.2
SU-1	40	7	5	*
SU-2	40	7	5	*
SU-3	31	0	5	0.2
SU-4	24	11	11	0.4
TOTALS:	561	80	57	94.2 (plus gimbal and tail excursion time)

Also shown in the tables is an estimated execution time for each of the modules, assuming 1 sec add, 8 sec multiply and data input or output 10 sec, and A-D conversion time 10 sec. With these nominal values, the total time needed to run through the complete test routine varies between 18 and 95 msec, depending on missile class. As mentioned previously, throughput is not important since tests are executed before launch i.e. off-line. In a tactical situation a launch delay of less than 100 msec is assumed to be acceptable.

Execution times for the control subsystem test modules SU-1 and SU-2 are not given in the figures, since the time needed for gimbals and/or tail surfaces to move through about 100 degrees of travel each would be measured in seconds instead of milliseconds and would render the other module execution times insignificant. The number and the size of the steps could be reduced, of course, but the total test time would still be dominated by these servo control checks.

Apart from the foregoing servo tests, memory checks require the most time to execute, since every word in ROM memory is readout and added and data is written into every RAM location, readout, and compared to the reference word. The number of memory locations used in the sizing was determined from the maximum values listed in the Phase I Final Study Report, plus the number of locations needed for test, telemetry, and mode control.

When computing execution time for Module ST-4, an average of 500 equivalent adds was assumed for each subroutine tested.

5. DIGITAL MISSILE PERFORMANCE

Simulations and analyses were performed for the functions of estimation, guidance autopilot/control and signal processing to confirm the algorithm formulations, digital implementation requirements and to develop function performance versus algorithm complexity data.

The performance of the systems under consideration can be expected to vary as the operational algorithms change in complexity and in their update rate.

A program module which uses several input parameters and processes these data using sophisticated mathematical techniques should provide improved performance, but at the expense of a greater number of memory locations, both for the initial data and the storage of partial results. The data sampling rate and hence the algorithm execution rate, required to achieve superior performance similarly drives the computer throughput and hence the machines' architecture, circuit technology and packing density/degree of large-scale-integration.

Consequently as each function is evaluated, its performance will be compared with the complexity of the algorithm used in generating the function. For example, the miss distance achieved with the four-state guidance law should be less than that with a proportional navigation law, but the throughput and memory requirements of the four state law will be considerably higher. Plots of miss distance for the various laws versus their memory and throughput requirements show the cost increment for corresponding increments in achieved performance.

Performance of a digital autopilot is best evaluated in terms of its response to a step g-command. Any analog autopilot, properly designed, will have stability margins and bandwidths suitable for the intended mission, and the design may range from a simple fixed gain system to one involving some form of real-time adaptivity. In a digital missile, the adaptive capability can be extended to ever increasing levels of sophistication, as described in the Phase I Final Study Report, with the result that bandwidth and stability margins can be maintained nearly constant over a wide range of operating conditions, leading to improved intercept performance.

Regardless of how bandwidth and stability are obtained (i.e., how the autopilot gains are chosen), the effects of digitization will be essentially the same, that is, for a particular set of flight conditions it can be assumed that the autopilot gains were optimally chosen, and the important digital parameters (data rate, quantization, and computation time) can be investigated to determine their effects on autopilot performance. The primary effects of coarse digitization are: a decrease in gain and phase margins which evidences itself in a growing tendency for the autopilot to limit cycle. This tendency can be readily seen in a forward time step response simulation and can be quantitatively measured by observing the fin (tail, wing) duty cycle that results from the autopilot command.

Duty cycle is defined as the total fin travel, in degrees, over a specified length of time:

$$\underset{(4\text{-fin})}{\text{Duty Cycle}} \quad \sum_{i=1}^{4} \int_{0}^{T} |\dot{\delta}_i(\tau)|\, d\tau$$

where $\dot{\delta}$ is fin rate. If the missile limit cycles in lateral accelerations, the fins must oscillate to support it, and the duty cycle will continually increase. The magnitude of the duty cycle bears directly on the amount of oil that must be carried in a hydraulic blow-down actuator system, and on the battery size and capacity needed by an electric system, hence duty cycle is an important consideration in missile design, and, together with the peak-to-peak amplitude of limit cycle oscillations, becomes an important measure of autopilot performance.

In a similar manner, the effectiveness of a given sensor signal processor can be determined by the effective improvement in signal-to-noise ratio compared to simpler configurations demanding less throughput and memory.

The results of the above performance vs computer characteristics analyses are reported in the following subsections.

5.3 Estimation and Guidance Performance

Estimation and guidance performance were investigated by simulation of a single missile (representative of Class II) utilizing, in turn, each of the guidance-estimation combinations to be tested. A common intercept scenario was used for all cases, and two types of performance data were generated:

(1) Performance in the presence of target maneuvers initiated at various times, but without heading error and measurement noise.

(2) Performance in the presence of heading error and measurement noise, with and without target maneuver (presented in the form of rms miss for 10-flight Monte Carlo runs).

Although the extent of these tests was necessarily limited, the results allow an approximate evaluation of performance in comparison with computational cost, including the effects of data rate and computational time delay.

Complete descriptions of the tested algorithms are presented in Reference R.1, but a brief summary will be given here. Guidance laws investigated were, for the most part, restricted to proportional navigation (PN, algorithm G-1) and the four-state law (4SL, algorithm G-2). Estimation algorithms included:

E2. Switched-gain filter. (This was simulated by determining approximate-optimum steady-state GH filter gains versus range for the radar and target models assumed, and tabulating these as functions of range in the on-board computer).

E3A. (A variant of the decoupled estimator, E3, in which the coupled nonlinear prediction equations are replaced by simple uncoupled predictors by removing Coriolis and centrifugal terms).

E3. Decoupled Kalman filter. Determines gains by propogating three 3x3 covariance matrices, but utilizes coupled nonlinear differential equations for state prediction).

E4. Coupled Kalman filter. (Propagates a coupled 9x9 covariance matrix for gain determination).

5.1.1 Test Scenario

The test scenario utilizied is a rather severe one designed to accentuate performance differences of the various algorithms. It is a "tail chase" engagement involving large target accelerations and a low average closing velocity. The target achieves its maximum acceleration of 6.5 g via a 2-second ramp initiated at a variable time, and the maximum is then maintained until intercept. The initial conditions are depicted in Figure 97 and additional parameters are given in Table 43.

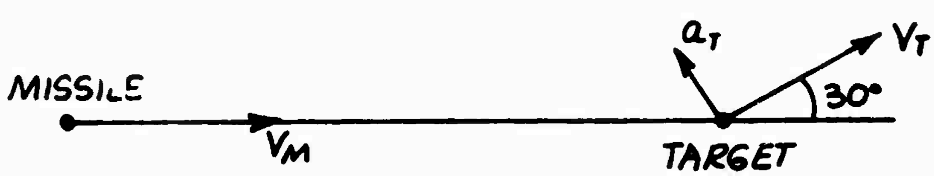


Figure 97 Test Scenario-Initial Conditions

The initial heading error of the missile is approximately 28°, and its thrust history is such that it attains a maximum velocity of about 2400 fps/731 mps about 5 seconds after launch.

The seeker is assumed to be a radar with a reference range (unity SNR) R_o of 51.6 kft/15.7 Km. It measures range and angle at a data rate of 10 times per second, and the associated error parameters are displayed in Table 44.

TABLE 43 TEST SCENARIO INTERCEPT PARAMETERS

Altitude (ft/m)	5,000/1524
Initial Range (ft/m)	6,000/1829
Initial missile velocity (fps/mps)	1,000/305
Target velocity (fps/mps)	950/290
Max. target acceleration (g)	6.5
Time of flight (sec)	6 approx.

TABLE 44 TEST SCENARIO RADAR MEASUREMENT ERRORS (RMS)

	ANGLE ERROR	RANGE ERROR
Range-independent	0.6 mr	0
Range-dependent	$9.0 \left(\frac{R}{R_o}\right)^2$ mr	$25.2 \left(\frac{R}{R_o}\right)^2$ ft
Glint (ft/m)	13/4 (3 Hz bandwidth)	5/1.5 (3 Hz bandwidth)

The missile and target dynamics were simulated in planar (three degree of freedom) fashion on a CDC 6700 computer. A block diagram of the guidance and estimation models is presented in Figure 98. The complete estimation and guidance algorithms were programmed as described in Reference R.1, while the autopilot dynamics were represented by a cubic transfer function from commanded acceleration to achieved lift:

$$\frac{N_L}{N_C} = \frac{1}{\left(\frac{s}{7} + 1\right)^3}$$

In general the radome characteristics and antenna stabilization were assumed to be perfect, so that the seeker measurement errors arose solely from the sources described above.

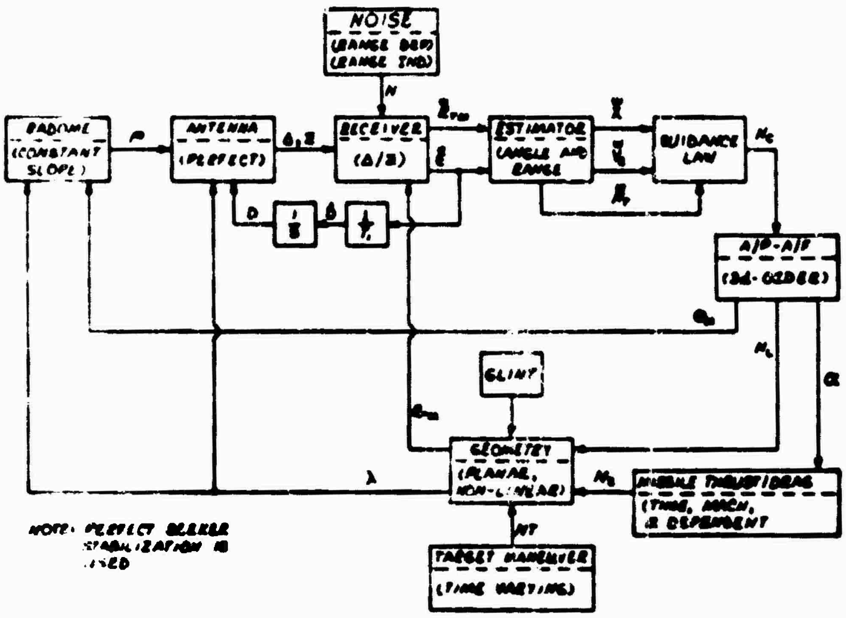


Figure 98 Guidance and Estimation Simulation Model

5.1.2 Estimator Performance

In general, performance data was generated for particular combinations of estimation and guidance algorithms. However, certain observations may be made about the performance of the estimators themselves:

Although the Kalman filters (E3 and E4) provide considerably better estimation performance then the switched-gain E2 algorithm, these investigations have thus far revealed no appreciable advantage of the E4 algorithm over the E3, as far as performance is concerned. Thus the E4 filter, with its far greater computation cost, does not appear to be cost-effective in the cases considered. (This conclusion could be modified, however, upon more complete investigation of cases where range information is severely degraded).

The E3A algorithm is not cost effective, because it saves very little in computation and noticeably degrades the state estimates, in comparison with the E3 algorithm. In particular, transients in target acceleration are responded to more slowly and tracked less accurately.

Since these filters operate in range and angle coordinates, the states experience severe transients just before intercept, which are not well tracked, in general, by the filter estimates. This has little impact on miss distance, but if the estimator outputs are to be used for fuzing purposes these final estimates may be important. If the filter is to be modified to improve these final estimates, it is probable that the state and covariance prediction equations would have to be solved at very high integration rates near intercept, which is clearly an undesirable solution. Probably a preferable alternative is to replace the state prediction equations by relations based on solution of the intercept triangle, as follows. Given estimates of range r, range rate $\dot{r}$ and LOS rate $\dot{\lambda}$ (the resultant of the pitch and yaw values $\dot{\lambda}_1$ and $\dot{\lambda}_2$), we may compute (see Figure 99).

$$V = \text{relative velocity} = \sqrt{\dot{r}^2 + r^2\dot{\lambda}^2}$$

$$x = \text{distance to intercept point} = -r\dot{r}/V$$

$$t_{go} = \text{time to go to intercept} = x/V$$

$$m = \text{miss distance} = r^2\dot{\lambda}/V$$

These transformations having been performed, we may now predict the values of the states at any future time t (assuming constant V) as follows:

$$x(t) = x-Vt$$

$$r(t) = \sqrt{m^2 + x^2(t)}$$

$$\dot{r}(t) = -Vx(t)/r(t)$$

$$\dot{\lambda}(t) = Vm/r^2(t)$$

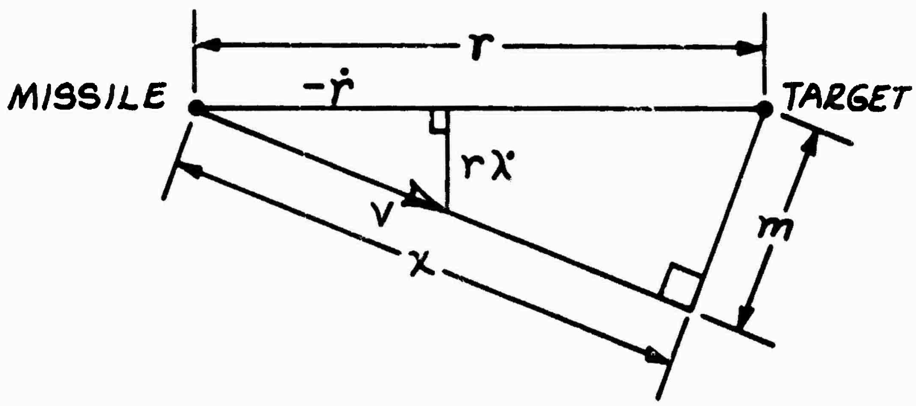


Figure 99 Estimation Performance Evaluation - Intercept Geometry

With these modifications in the state prediction operation, improved estimation can be achieved near intercept. The computation of the filter gains is less critical, and can be accomplished in an approximate manner.

5.1.3 Combined Estimation - Guidance Performance

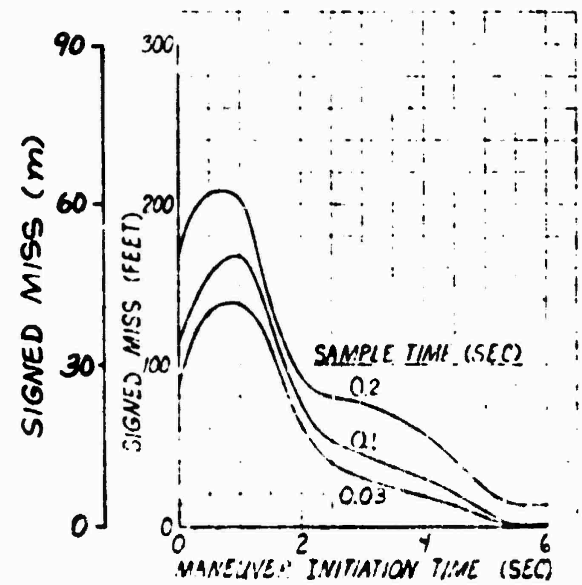
The algorithms chosen as typical of the various missile classes, for purposes of performance comparisons, are:

Class I: Estimator E2 (switched gains)
Guidance Law G1 (proportional navigation)

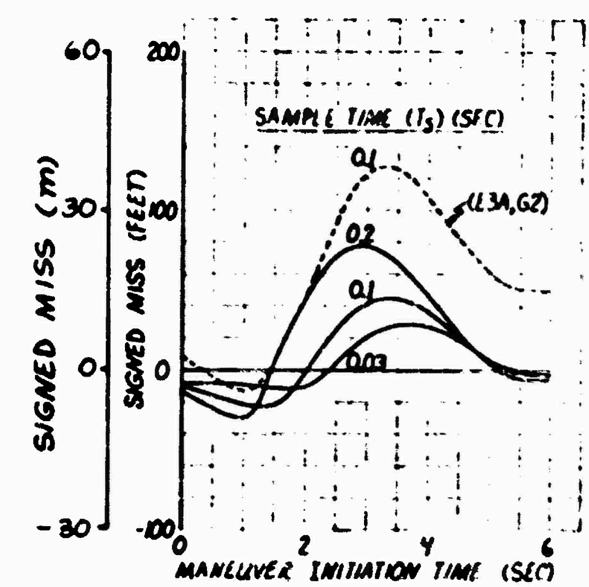
Classes II&III: Estimator E3 (decoupled Kalman)
Guidance Law G2 (four-state law)
(performance unchanged with coupled estimator E4).

In Figure 100 these combinations are compared in the chosen intercept scenario and at various data rates, on the basis of the miss distance they produce due to target maneuver alone, versus time of initiation of the target maneuver. This data was generated by removing all measurement noises, but leaving the estimator in the system in order to properly include its dynamic response. (For tests of this type, target acceleration was a step rather than a ramp.) The miss values are ascribed a sign (direction) as well as a magnitude, because a planar simulation was used to generate them. Also included is a single curve exemplifying the performance of the modified E3 (E3A) filter. Several observations are readily apparent from the figure:

(1) The obvious superiority of the (E3, G2) combination over the (E2, G1) combination (principally in the early-maneuver cases).



(A) Class I Missile



(B) Class II and III Missiles

Figure 100 Combined Estimation and Guidance Performance-Miss Distance versus Target Maneuver-Various Data Rates

(2) The considerable degradation in performance when the E3 estimator is simplified to the E3A variant. The difference is most pronounced for late maneuvers, because it is mostly late in the flight that it becomes important to properly model the dynamics (especially the Coriolis accelerations) in the prediction equations.

(3) The degradation which may be experienced when the data rate is allowed to become too slow.

A more complete performance comparison is presented in Figure 101, in the form of rms miss from Monte Carlo simulations with all measurement noises included, and with the maneuver initiation time uniformly distributed over the duration of the flight. Three estimation-guidance combinations are compared (including the fully coupled E4 estimator, whose performance is substantially the same as the E3), at three different data rates for each. Also included are comparisons against a non-maneuvering target, in which case the miss is due solely to measurement noise and initial heading error.

The rms miss is plotted against computational cost to facilitate cost-effectiveness trade-offs. Although the data is of limited accuracy (due to the necessarily small number of Monte Carlo runs used, namely ten), the principal trends are readily apparent. Without target maneuver the performance differences are not great, but with target maneuver considerable improvements in miss can be realized (at the expense of additional computation) by increased data rates or by the use of more sophisticated algorithms.

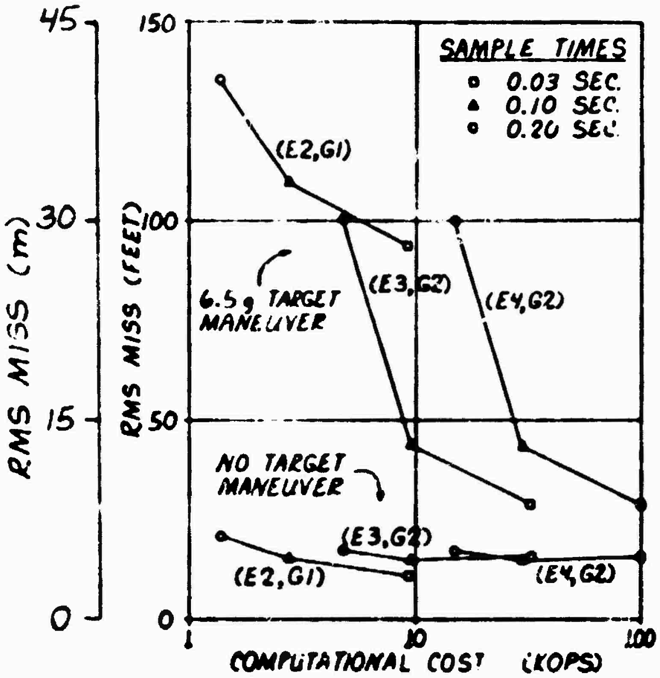


Figure 101 Combined Estimation and Guidance Performance-
RMS Miss Distance Versus Computational Burden

The generally high level of the misses is largely attributable to the severity of the intercept scenario utilized for the simulations.

5.1.4 Effects of Time Delay

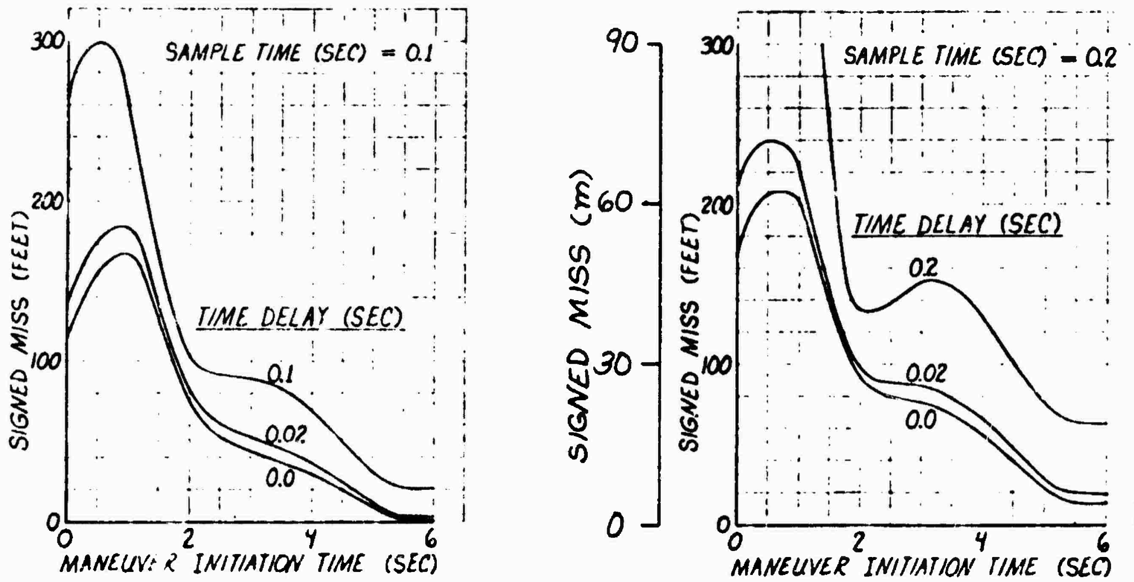
A major factor impacting the requirements for computer capacity, especially when signal processing is entirely digital, is the allowable computational time delay between the reception of a radar return (or other measurement) and the generation of guidance commands. Accordingly, Figures 102 and 103 show how this parameter affects the performance data previously presented.

The delay may vary from a minimum near zero (as was assumed in the previous data) to a maximum equal to the data sampling interval.

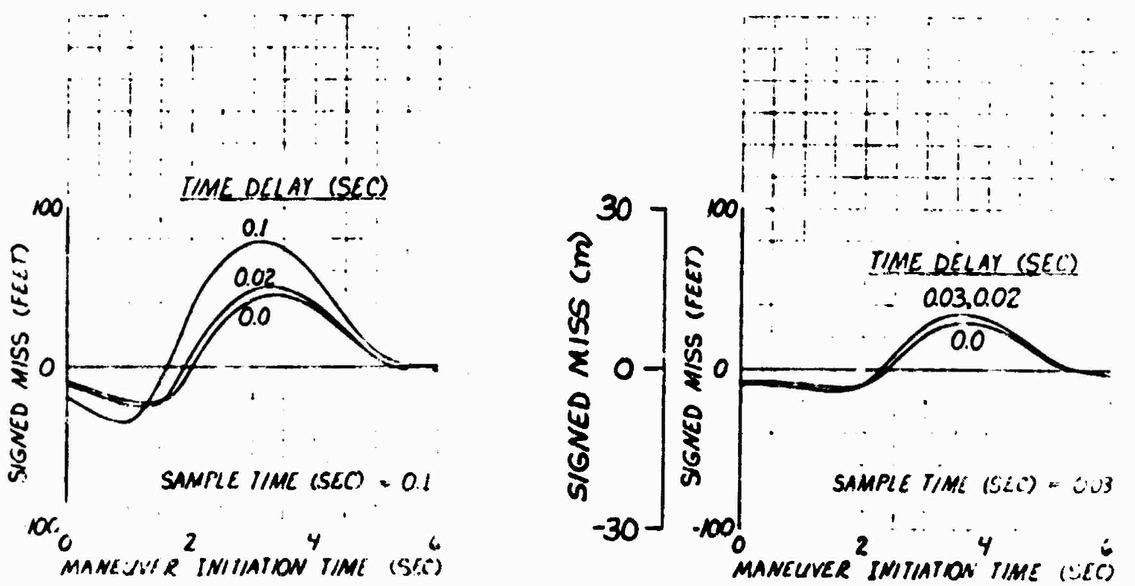
In many cases it may be advantageous to partially compensate for the effects of time delay by the use of an additional pass through the filter's prediction equations (or perhaps a simplified version of them), to bring the state estimates and guidance commands up to date at the time when they finally become available. The cost of these extra operations is relatively minor; however, such a compensation operation was not employed in these simulations, so that the degradations shown here are probably somewhat pessimistic.

In general, it can be seen that an increase in computational time delay degrades both the miss due to target maneuver (Figure 102), and the rms misses with and without maneuver (Figure 103). A delay of 20 msec appears to be of little consequence in the cases studied, but delays approaching the sample time T_S can cause important increases in miss distance, especially in the presence of target maneuvers.

An apparent discrepancy appears in Figure 103(B), where the rms miss without target maneuver appears to become worse when the data rate is increased. There are two possible explanations for this:

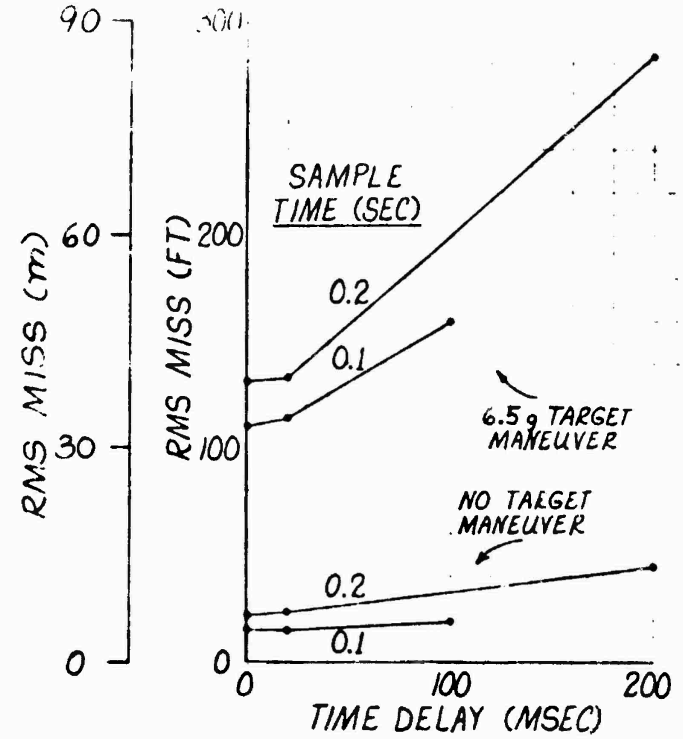


(A) Class I Missile

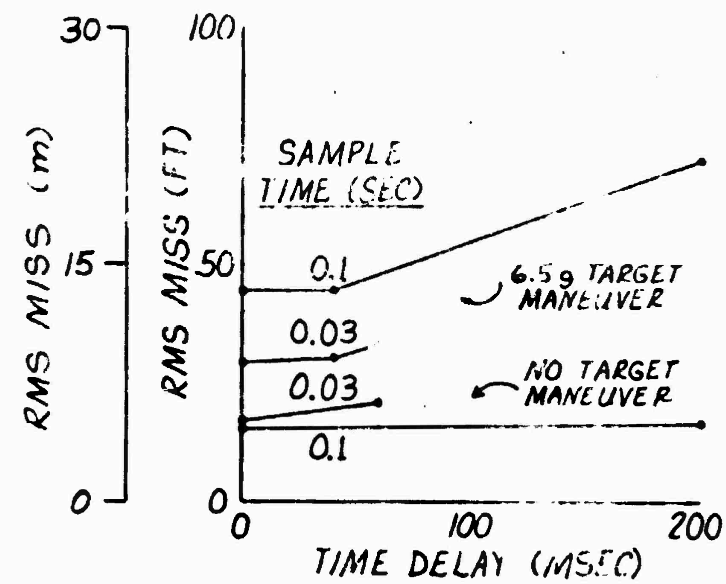


(B) Class II and III Missiles

Figure 102 Miss Distance Versus Computational Time Delay - Maneuvering Target



(A) Class I Missile



(B) Class I and II Missiles

Figure 103 RMS Miss Distance Versus Computational Time Delay

(1) The rms values are only approximate, because only ten Monte Carlo runs were processed.

(2) The filter assumes that the measurement errors are uncorrelated, whereas the glint errors (which are the dominant errors in this scenario) have a finite bandwidth of 3 Hz. The degradation due to the white-noise assumption can be expected to become more severe as the data rate increases and successive measurements become more highly correlated.

5.1.5 Effects of Computing Precision

One of the critical decisions to be made in the digital implementation of missile guidance and control functions is the degree of computing precision required. In this study, the effects of limited precision were investigated in two separate ways:

(1) Since the most sensitive portion of the guidance and estimation algorithms was expected to be the propagation of the covariance matrix in the Kalman filters, a simplified (non-Monte Carlo) covariance propagation program was used to investigate precision effects on the calculated covariance values and on the resulting filter performance.

(2) In the primary simulation program, variable computing precision was simulated by truncation of all simulated filter computation results at a specified number of bits, and the effects were investigated in Monte Carlo fashion.

Covariance Propagation Studies - Utilizing techniques which are well known in the field of estimation theory (see, for example, Reference R.2), a covariance propagation program was developed, which utilizes the equations of the estimation algorithms to simultaneously propagate three separate covariance matrices:

(1) The "optimum" covariance matrix, which would be generated by a filter using very high precision and which would, under proper conditions, closely approximate the true performance achieved by that filter.

(2) The "calculated" covariance matrix, which is generated by the limited-precision computer, and which results in erroneous filter gains and degraded estimation performance.

(3) The "true" covariance matrix, which indicates the performance actually realized by the degraded filter.

<u>OPTIMUM</u>

$$M = \phi P \phi^T + Q$$
$$K = M H^T [H M H^T + R]^{-1}$$
$$P = (I - K H) M$$

<u>CALCULATED</u>

$$\hat{M} = \hat{\phi} \hat{P} \hat{\phi}^T + Q$$
$$\hat{K} = \hat{M} H^T [H \hat{M} H^T + \hat{R}]^{-1}$$
$$\hat{P} = (I - \hat{K} H) \hat{M} \quad \text{(STANDARD FORM) } \underline{\text{OR}}$$
$$\hat{P} = (I - \hat{K} H) \hat{M} (I - \hat{K} H)^T + \hat{K} \hat{R} \hat{K}^T \quad \text{("JOSEPH" FORM)}$$

<u>TRUE</u>

$$\tilde{M} = \phi \tilde{P} \phi^T + Q$$
$$\tilde{P} = (I - \hat{K} H) \tilde{M} (I - \hat{K} H)^T + \hat{K} R \hat{K}^T$$

Figure 104. Covariance-Matrix Propagation Equations

The covariance simulation represents a six-state (planar) system, and assumes floating-point capability using a mantissa whose length can be varied by program input. The equations used, as presented in Figure 104 are appropriate to the case where the limited mantissa length affects only the covariance matrix and gain calculations, but not the accuracy of propagation of the state estimates. This implies that state propagation is done in double precision, because such an approach significantly increases estimator performance while adding little to the computational cost (since state prediction represents a small fraction of the cost of a Kalman filter). In these simulations, "double precision" was simulated by using the full 48-bit mantissa of the CDC 6700 computer.

One of the places where truncation error is liable to be serious is in the computation of the matrix $(I - \hat{K}H)$ utilized in Figure 104. When a measurement of the first state is made, for example, the (1,1) element of $(I-\hat{K}H)$ is $(1-k_1)$, where k_1 is the first element of the gain vector $\hat{K}$ and is given by

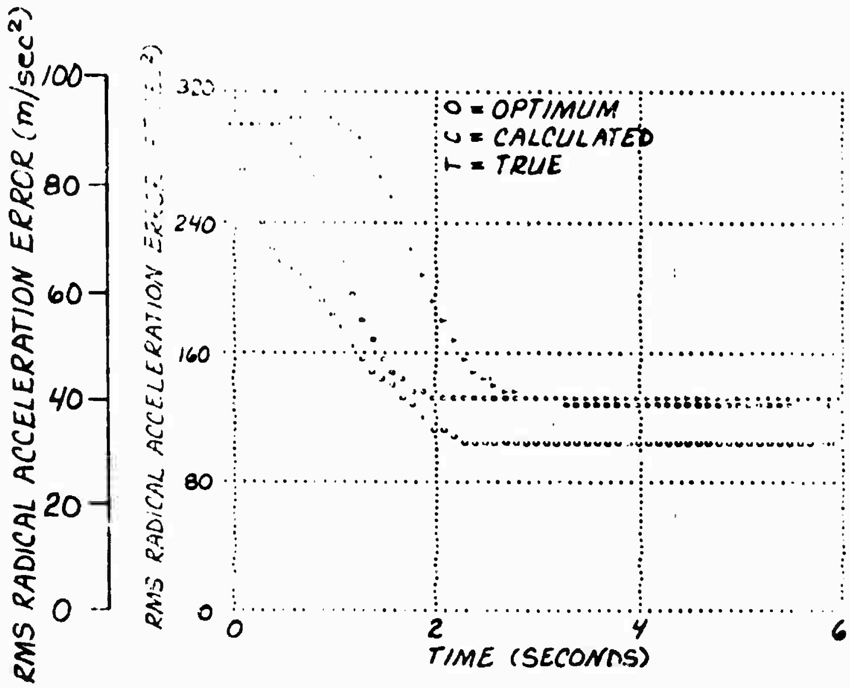
$$k_1 = \frac{M_{11}}{M_{11} + \sigma_M^2}$$

where M_{11} is the (1,1) element of the predicted covariance matrix M and σ_M^2 is the measurement-error variance. When σ_M^2 is much smaller than M_{11}, which is often the case (especially at the beginning of the filter operation), k_1 will be near unity and mantissa truncation can cause serious errors in the difference $(1-k_1)$. This difficulty can be avoided by the simple artifice of calculating the difference directly, as

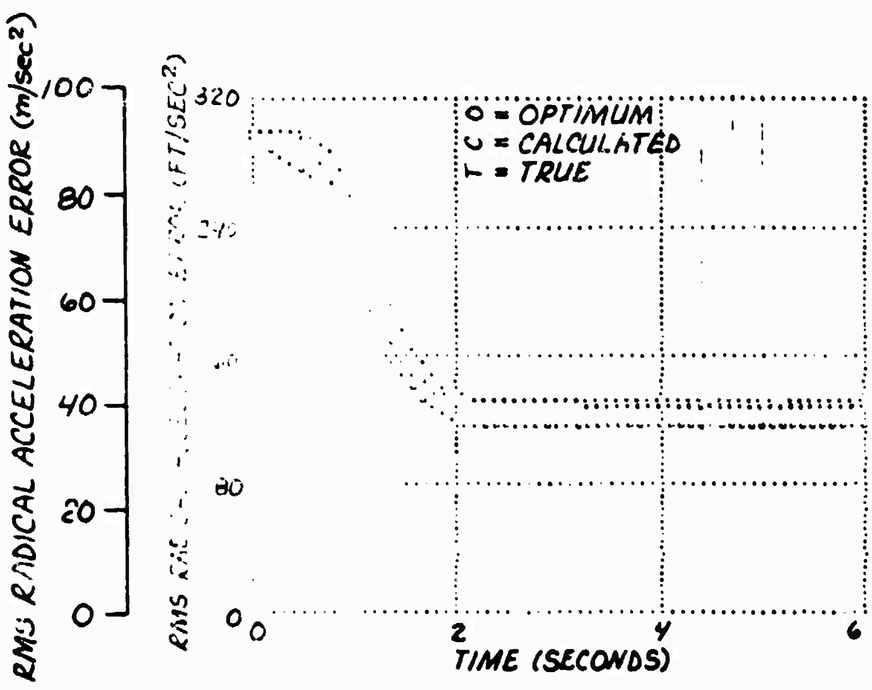
$$(I - \hat{K}H)_{11} = (1-k_1) = \frac{\sigma_M^2}{M_{11} + \sigma_M^2}$$

This modification was employed in both the covariance program and the Monte Carlo simulation, and was seen to yield some improvement at shorter mantissa lengths.

Some typical results are shown in Figure 105, in the form of time histories of the rms radial acceleration errors as indicated by the three covariance matrices, for mantissas of 6 and 8 bits (including sign). The scenario described previously was used to generate appropriate time histories of the state variables for determining the elements of the state transition matrix and the measurement-error covariance matrix R. It may be seen from the results that, as has often been observed, the actual performance of the filter (as indicated by the true covariance matrix) is generally degraded less than the elements of the calculated covariance matrix. For a 6-bit mantissa, performance degradation is quite noticeable but might still be acceptable for some applications, especially since it is very near the optimum towards the end of the flight.



(A) 6-Bit Mantissa



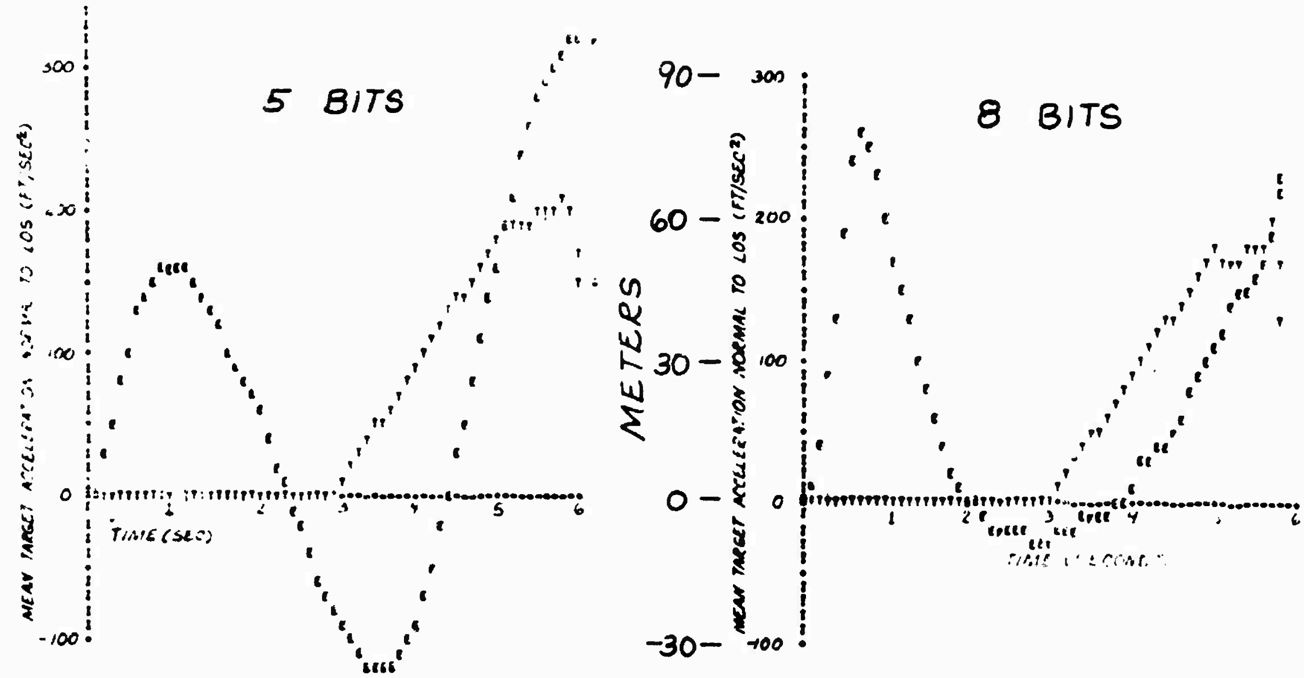
(B) 8-Bit Mantissa

Figure 105. Comparison of Optimum, Calculated, and True Covariance Matrix Elements (Floating-Point with Double Precision State Propagation).

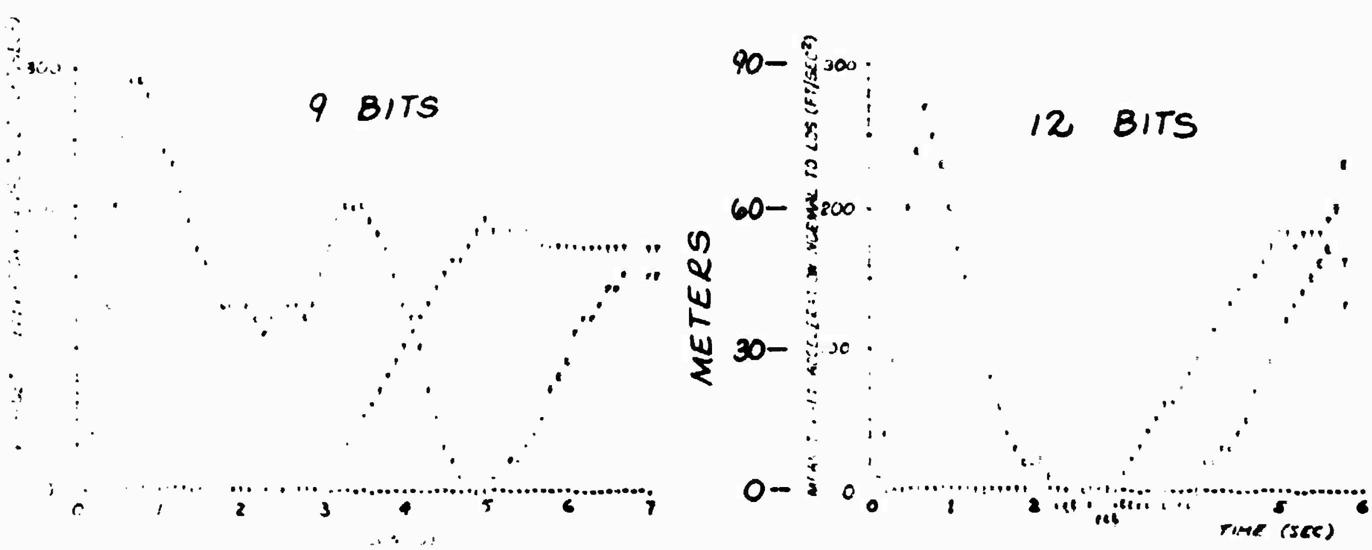
Monte Carlo Studies - For these investigations, the complete (planar) system simulation, modified for limited filter precision, was rerun using the standard engagement scenario. Statistics were gathered from sets of ten Monte Carlo runs with all standard noise sources present, except that the target maneuver initiation time was kept fixed at three seconds.

The ultimate performance criterion can be considered to be rms miss, but it is instructive also to examine estimation performance. For selected mantissa lengths, Figure 106 shows averages (over the ten Monte Carlo runs) of the true and estimated target accelerations normal to the line of sight, with state prediction performed in double and single precision. It is evident that the shorter mantissa lengths tend to cause oscillatory behavior, slow recovery from initial transients, and/or sluggishness in responding to target maneuvers. Also evident is the fact that considerable improvement is realized in the estimator and the mantissa length requirements relaxed, when state propagation is performed in double precision. (In all cases, guidance-law computations were performed with double-precision.)

In terms of rms error in estimation of range rate, a similar comparison is made in Figure 107. Finally, Figure 108 shows how precision affects the rms miss distance.

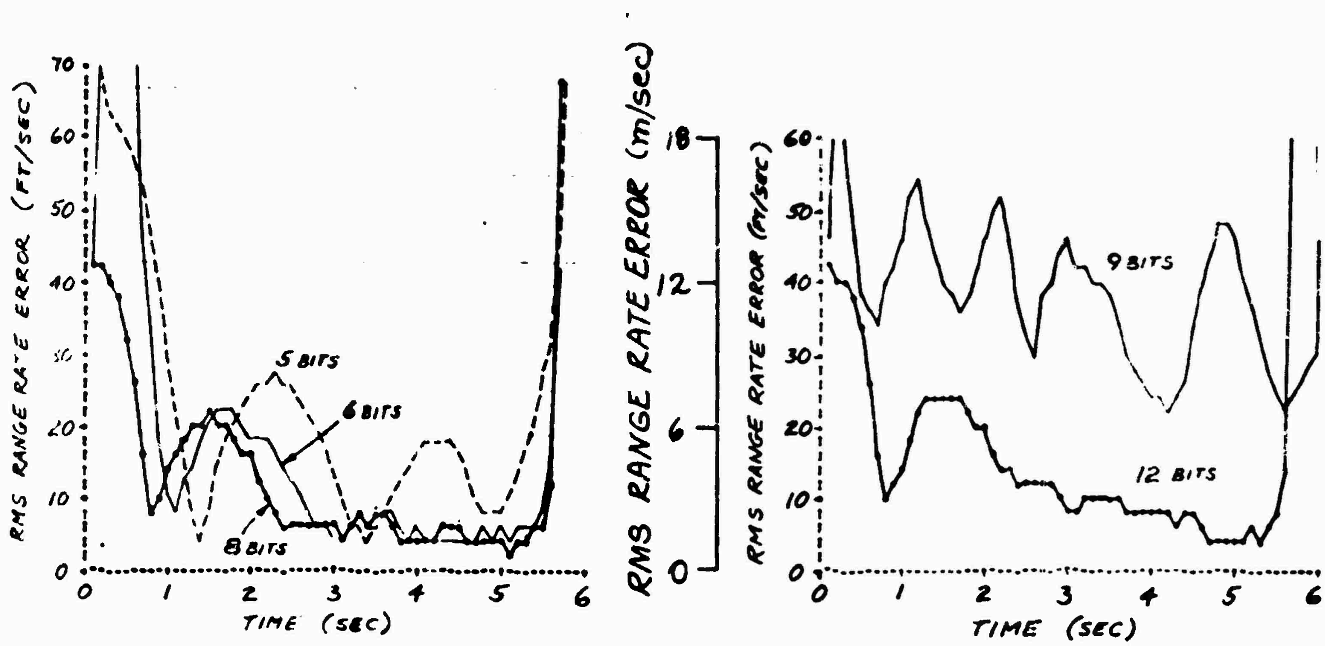


(A) With Double-Precision State Propagation



(B) With Single-Precision State Propagation

Figure 106. Effects of Mantissa Length on Target Acceleration Estimation (E = Estimated, T = True).



(A) With Double-Precision State Propagation

(B) With Single-Precision State Propagation

Figure 107. Effects of Mantissa Length on Range Rate Estimation.

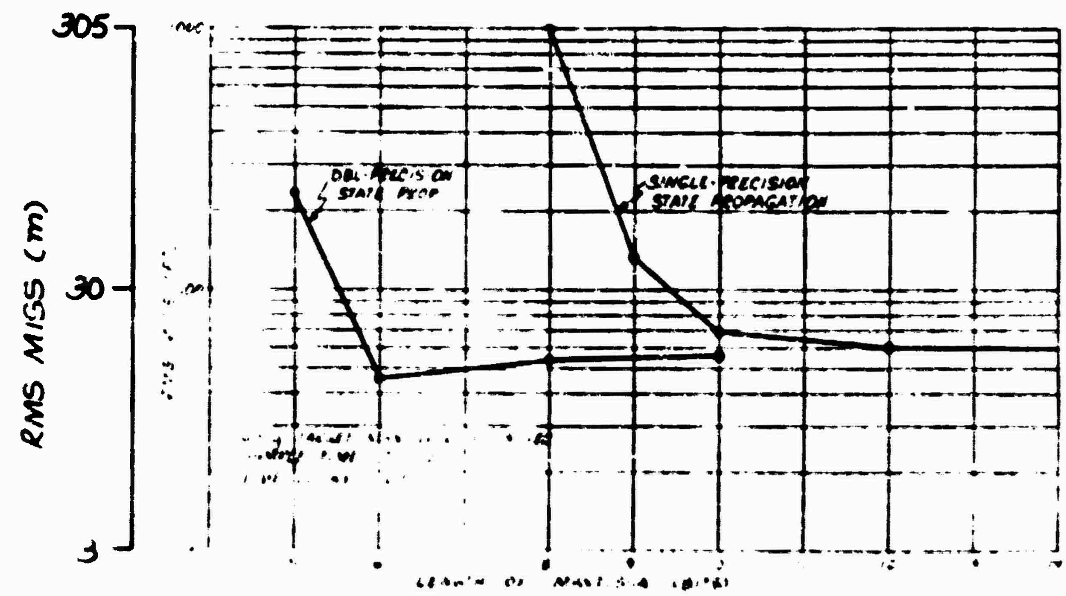


Figure 108. Effects of Mantissa Length on RMS Miss.

From this point of view, it would appear that significant miss degradation does not set in (assuming double-precision state propagation) until the mantissa length is reduced to 5 bits. However, it is dangerous to choose the mantissa length based on this criterion alone, as can be seen by the degradation of estimation performance at short word lengths, which probably indicates the possibility of numerical difficulties for other values of the system parameters. Some indication of incipient danger can be seen in the fact that, early in the flight, the covariance matrix of the 6-bit filter loses positive definiteness for a time although it later recovers from this condition. The same problem was observed, although to a lesser degree, at 7 bits. In Table 45, is presented a comparison of the range position gain (the filter gain by which the range estimate is adjusted due to a range measurement) during the first second of flight, for various mantissa lengths. Although this gain should never exceed unity, signs of inaccuracy are already apparent at 8 bits, and the gains are considerably in error at 6 bits.

TABLE 45

COMPARISON OF FILTER RANGE GAINS DURING FIRST 1-SECOND INTERVAL OF FLIGHT

TIME (SECS)	MANTISSA LENGTH 48 BITS	12 BITS	8 BITS	6 BITS	5 BITS
0	1.00	1.00	1.00	1.00	1.00
.1	1.00	1.00	1.00	1.00	1.00
.2	.83	.67	1.08	1.00	1.00
.3	.70	.50	.84	6.00	.69
.4	.60	.41	.69	1.00	.50
.5	.53	.38	.59	$\approx$ 0	.38
.6	.48	.37	.52	1.12	.34
.7	.45	.39	.46	.75	.28
.8	.44	.42	.44	.59	.25
.9	.43	.43	.42	.50	.25
1.0	.43	.43	.41	.47	.25

From the investigations reported on here, it is reasonable to conclude that

(1) Double precision should generally be used for state propagation, because it results in performance improvements which are very great in comparison with the small added cost.

(2) Covariance propagation may be performed with a minimum of eight bits (including sign) for the mantissa.

5.1.5 Effects of Degraded Range Information

The range data available to the tracking filter may become degraded or may be completely denied. Among the possible causes of this are:

(1) ECM action by the target.

(2) Range designation accuracy insufficient to resolve range ambiguities (results in a range bias).

When such a condition arises, it may become advisable to modify the guidance and estimation algorithms to minimize the inevitable performance degradation. Such situations make performance prediction quite difficult because the results may depend very strongly on such things as:

(1) How the algorithms (especially range prediction or coasting) are modified,

(2) The quality of the information available before range data is denied,

(3) The engagement geometry, and

(4) The target behavior subsequent to commencement of ECM.

It is thus quite difficult to conduct tests from which useful conclusions can be drawn. Nevertheless, a limited number of Monte Carlo runs were conducted in the standard scenario, with range measurements denied and with particular magnitudes of error in the initial estimate of the radial component of target velocity. The results are depicted in Figure I09.

Although performance degrades as expected for positive errors, negative errors appear to decrease the miss in the (E2,G1) case, a result which can probably be ascribed to the particular scenario used. In any case, the (E3,G2) combination maintains its superiority over the (E2,G1) system even in degraded cases, which should be expected since it is always possible to modify such a system so that it degrades to the (E2,G1) system in the degraded-range case.

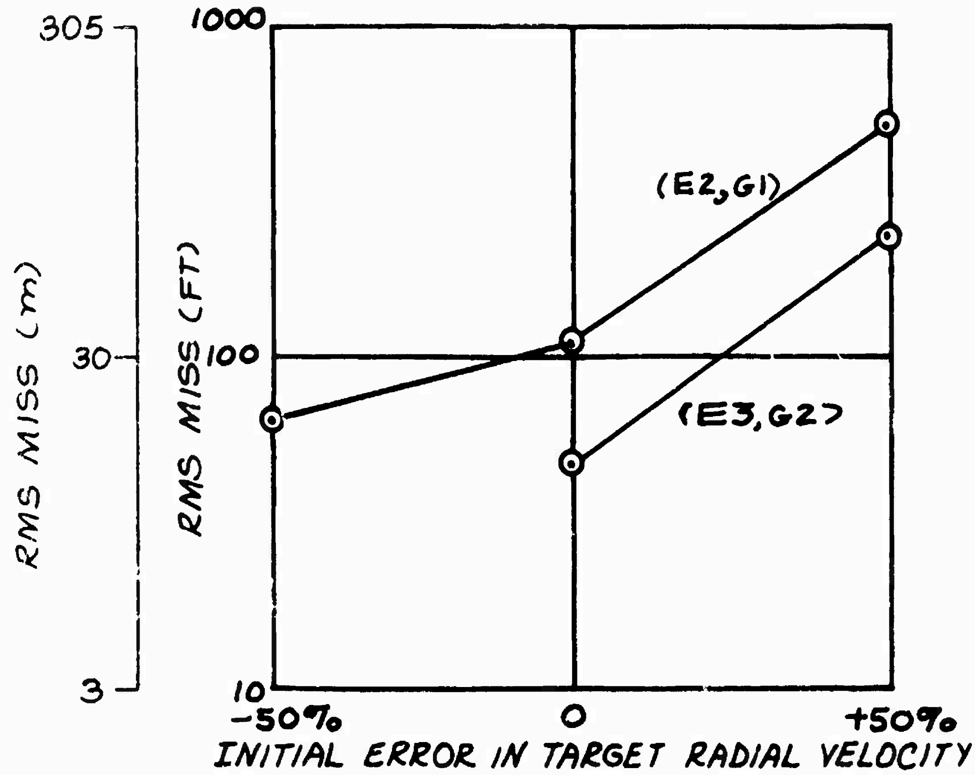


Figure 109. Performance in Range-Denied Environment

5.1.7 Conclusions

From the performance studies described here, several conclusions can be drawn regarding the guidance and estimation algorithms and the preferred design features of the missile computer:

(1) At least for the scenarios studied here, there appears to be no advantage in performance which would justify the much greater computational cost of a fully coupled Kalman filter for estimation.

(2) When a decoupled Kalman filter is utilized, the complete nonlinear, coupled differential equations should be used for state prediction.

(3) To a certain extent, cost-effectiveness (in terms of miss versus computation time) improves as data rate is increased. The optimum data rate will depend on seeker characteristics and on the other loads on the computer.

(4) Moderate computational time delays (on the order of 20 msec) cause little performance degradation, but delays approaching the sample time in duration can have serious consequences, especially at the lower data rates.

(5) In short word-length computers, accuracy of Kalman filter state propagation should be preserved by the

use of double/triple - precision for the state computations. In general, an equivalent mantissa length of 12 to 16 bits appears adequate for this purpose.

(6) An eight-bit mantissa (including sign) appears adequate, in most cases, for Kalman filter covariance-matrix propagation.

5.2 Autopilot Control

This subsection describes the tests and results obtained through the digital simulation of a three-degree of freedom (3DOF) autopilot. The simulation model is first outlined with the response of an "unrestricted" (32-bit) digital mechanization compared to the unquantized analog counterpart. Peculiarities of the digital mechanization due to non-linear effects are then described together with the corrective measures taken.

The effects of computing precision are explored for the cases of: 8, 12, 16 and 32-bits fixed-point, (sign plus magnitude), respectively. A distinction is made between precision versus word length since the former is achievable on various word length machines subject to a throughput penalty for byte manipulation. Performance sensitivity to data sampling rates over the range of 125 Hz to 1000 Hz are reported followed by computational delay and A-D/D-A quantization effects. Lastly the computer requirements are summarized for each of the three generic classes of air-to-air missile.

5.2.1 Simulation Model

A typical pitch (or yaw) planar autopilot, shown in analog form in Figure 110, was converted to digital operation for simulation on a CDC 6700 general purpose digital computer to determine performance sensitivity to computing precision, A-D/D-A quantization level and sampling rate. All of the resulting digital arithmetic shown in Figure 112 was digitized step by

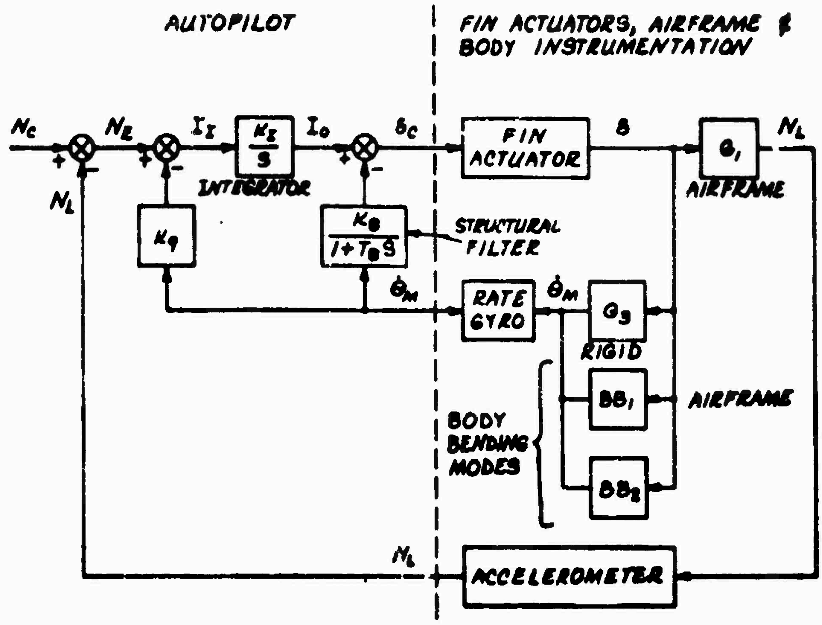


Figure 110 Planar Pitch (or Yaw) Analog Autopilot/Airframe Model

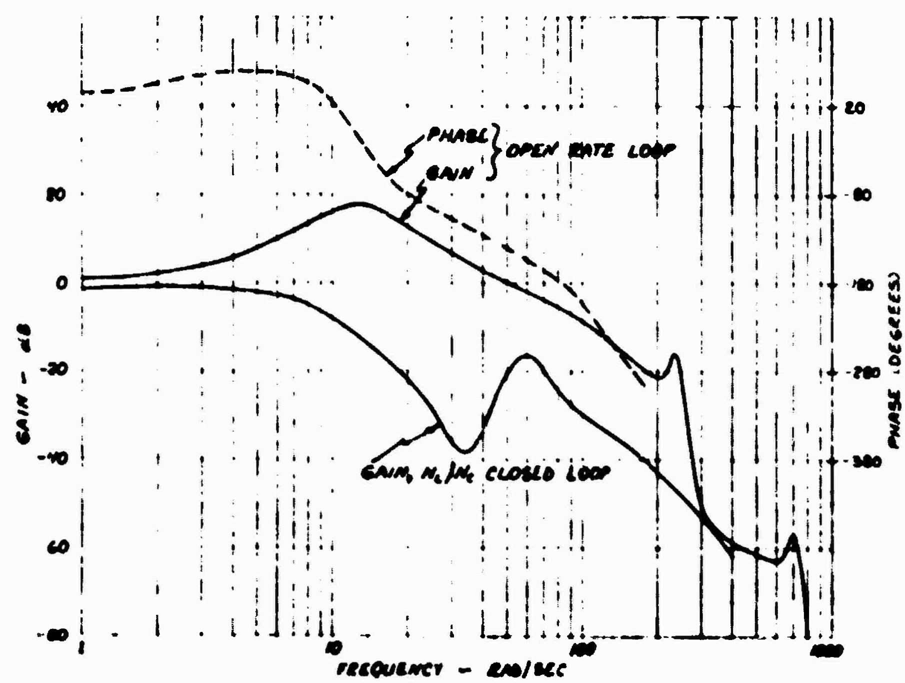


Figure 111 Planar Pitch (or Yaw) Analog Autopilot Bode Plots

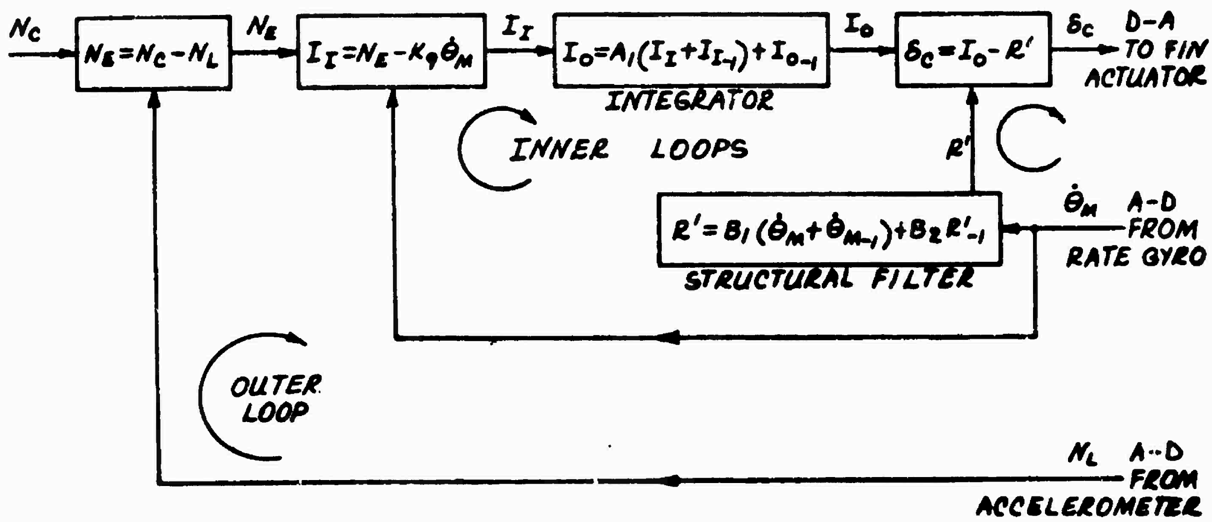


Figure 112 Planar Pitch (or Yaw) Digital Autopilot

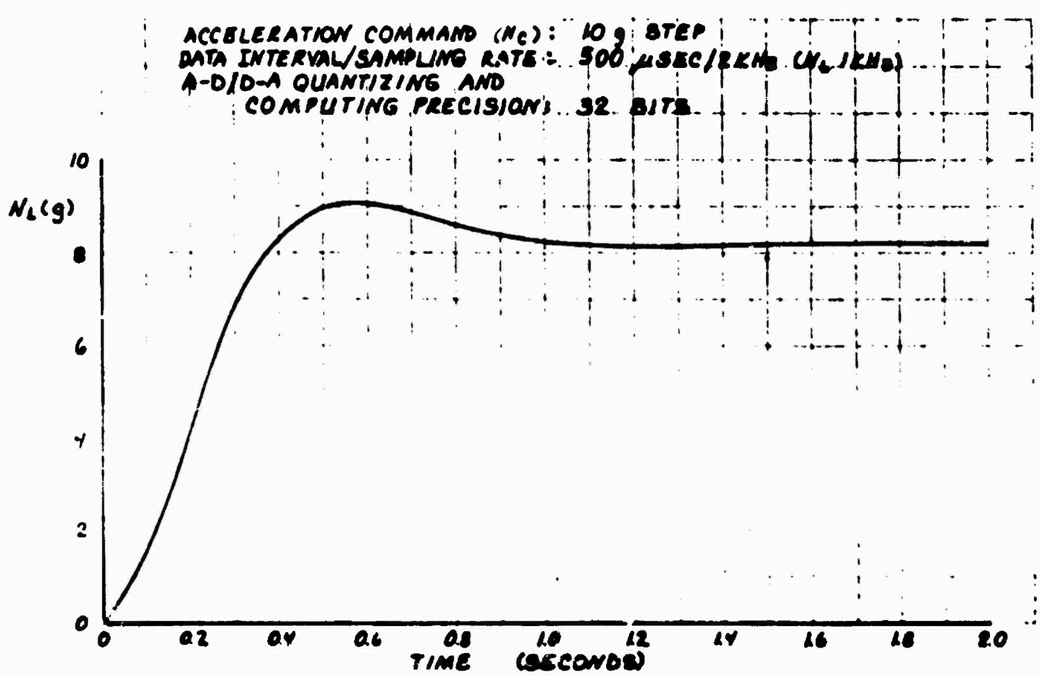


Figure 113 Digital Autopilot Step Response

"Unrestricted" Data Interval and Computing Precision step in fixed-point format with the choice of 8,12,16 or 32-bit precision, while the A-D and D-A interfaces to the remaining analog section were quantized independent of the computing precision selected.

Sampling rates for the rate and acceleration feedback paths were also independently variable, and provision was made for delaying the output of the calculated fin command for a fixed interval.

Two body bending modes were included in the analog portion of the simulation to observe when foldover of these frequencies became a problem. The parameters associated with these modes and with the missile rigid body motion are listed in Table 46 together with the autopilot constants used in the study. These values were chosen to give a relatively low stability system, i.e. one that would quickly point out the destabilizing effects of digitization.

The open-loop frequency response of the continuous rate loop is shown in Figure I12, as is the gain plot of the closed-loop N_L/N_C autopilot. Inner loop gain margin is seen to be only 3.2 db, while the closed loop bandwidth is approximately 7.5 rad/sec. Body bending peaks were not destabilizing.

The bandwidth of the closed-loop response was corroborated by running the forward-time digital simulation in an essentially "unrestricted" mode i.e., 32 bits precision and 2 KHz data rate. The corresponding step response of Figure I13.

TABLE 46

AUTOPILOT/AIRFRAME PARAMETERS

AUTOPILOT

K_I	2.82
K_g	0.15
K_8	0.115
T_8	0.019 secs.

RATE GYRO ω_N = 377 rad/sec, ζ = 0.7

FIN ACTUATOR ω_N = 150 rad/sec, ζ = 0.5

AIRFRAME

$$G_1 = 6.26 * \frac{1 + 0.00456S + 0.000878S}{1 + 0.0457S + 0.00616S} \quad (\omega_N = 12.7 \text{ rad/sec})$$

$$G_3 = 8.62 * \frac{1 + 3.91S}{1 + 0.0457S + 0.00616S}$$

$$BB_1 = \frac{-0.00368S}{1 + 0.00021S + 0.000017S} \quad (\omega_N = 242 \text{ rad/sec}, \ \zeta = 0.025)$$

$$BB_2 = \frac{-0.000938S}{1 + 0.00007S + 0.000002S} \quad (\omega_N = 697 \text{ rad/sec}, \ \zeta = 0.025)$$

compares rather closely to a second order system with damping ratio equal to 0.6 and bandwidth equal to 7.4 rad/sec, and the steady-state gain of 0.83 similarly matches that of the frequency response of Figure 111. Figure 113 therefore becomes the baseline step response to which the performance of each digital autopilot test case is compared.

5.2.2 <u>Digital Peculiarities</u>

As stated above, each of the digital arithmetic operations in the autopilot was quantized in fixed-point format consequently requiring that the maximum value of each externally and internally generated variable be determined. For convenience, powers of two were chosen for each of these maximum values, and a partial listing is given below (reference Figure 112):

<u>Variable</u>	<u>Maximum Value</u>
N_C	± 32 g
N_L	± 32 g
$\dot{\theta}_M$	± 256 deg/sec
N_E	± 32 g
I_I	± 32
I	± 16
δ^o_C	± 32 deg

The values shown for the input and output of the forward path integrator, are somewhat deceptive. This function has the potential of giving a very large output for a very small input and internal quantization of the integrator equation of Figure 112 can cause the entire autopilot to become effectively turned off. As an example of this consider that the constant A_I in Figure 112 is

$$A_I = K_I T_S / 2$$

where T_S is the sampling interval. For K_I = 2.82 and T_S = 1.0 millisecond, A_I becomes 0.00141. A 10 g step command into the autopilot will then produce a product $A_I I_I$ of 0.0141. However, if the computer only has 8 bit precision, the least significant bit (LSB) of the output variable I_O is equal to 0.126 and consequently, the output of the integrator remains at zero for the first pass through the equations. It is a simple matter to show that the output will stay at zero indefinitely, effectively shutting off the autopilot.

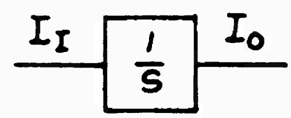
In this example, the data interval is too short to allow the input value to be integrated up to the least significant bit (LSB) level of the output. The problem can be solved by using greater precision in the computer or extending the data interval (the former reducing the output LSB, the latter integrating over a long period), but whatever the combination used, there will be some value of input for which the system will not work. 12-bit precision and 8 msec data interval, for instance, will not pass a step g command smaller than 0.7 g.

A method of avoiding this difficulty is to perform the integration in two stages, separating the input from the output. This is illustrated in Figure 114 (A) and (B), where the one - and two - stage methods are compared respectively. In the latter scheme, the input is integrated up to a variable X_4, whose maximum value is only slightly larger than the LSB level of the output, I_O. The LSB value of X_4, therefore, is very small and the integration is able to proceed with small inputs. When X_4 reaches an output LSB (plus or minus), the output is incremented (or decremented) by X_4, and X_4 is reset to zero. A comparison of the capabilities of these two integration schemes is shown below:

Precision/Data Interval	Integrator	Smallest Autopilot Step Input Allowed
8 bits/1 msec	1-stage	89 g
	2-stage	0.7 g
12 bits/8 msec	1-stage	0.7 g
	2-stage	0.0003 g

Somewhat the same consideration must be given to the digital structural filter, since the bi-linear Z-transformation which produces the equation shown in Figure 112 also results in the constants B_1 and B_2 being functions of the data interval. However, for maximum body rates of 150-200 deg/sec and a steady state gain of 0.125 (K_θ), the maximum value of $B_2 R_{-1}$ product can be kept consistently at 32, and the $B_1 \dot{\theta}_M$ product somewhat less consistently (but sufficient for this study) at 2.0. At a data interval of 4 msec, this corresponds to a LSB of 1.4 deg/sec for $\dot{\theta}_M$ with 8 bit precision, and 0.09 deg/sec for 12 bits.

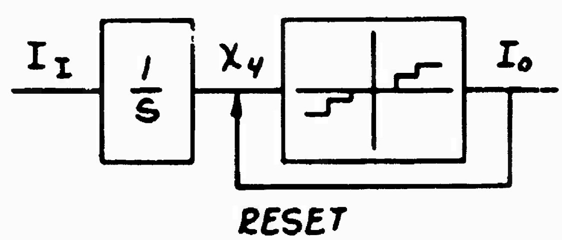
1. $\chi_1 = Q_1(A_1 * I_I)$
2. $\chi_2 = Q_1(A_1 * I_{I_{-1}})$
3. $\chi_3 = Q_2(\chi_1 + \chi_2)$
4. $QI = Q_3(\chi_3 + I_{I_{-1}})$



$Q_1 \triangleq$ QUANTIZER

NOTE: IF $\chi_3 < Q_3$, OUTPUT I_O WILL NOT CHANGE

(A) Single-Stage Digital Integrator



1. $\chi_1 = Q_1(A_1 * I_I)$
2. $\chi_2 = Q_1(A_1 * I_{I_{-1}})$
3. $\chi_3 = Q_2(\chi_1 + \chi_2)$
4. $\chi_4 = Q_2(\chi_3 + \chi_{4_{-1}})$
5. $\chi_4 \geq Q_3$? $\xrightarrow{\text{NO}}$ $I_O = I_{O_{-1}}$

   ↓ YES
6. $I_O = (\chi_4 + I_{O_{-1}})$
7. RESET χ_4 TO ZERO

(B) Two-Stage Digital Integrator

FIGURE 114 - INTEGRATION SCHEMES

Throughout this study, "data interval" refers to the sampling interval of the inner two autopilot loops (Figure 112) the outer (accelerometer) loop is always sampled at half the inner loop rate. The D-A is updated at the inner loop rate.

5.2.3 Computing Precision

Figure 115(A) shows the response of the autopilot to a 10g step input using a 1 KHz data rate and computing precisions of: 16, 12, and 8 bits (magnitude + sign), respectively. Two-state integration was used, with the A-D and D-A convertors quantizing at the same level as the computer precision selected. The 16 bit response is identical to that of Figure 113, while with 12 bits the system enters a small oscillation mode soon after the initial period of the response. The peak-to-peak amplitude of this oscillation, measured after 2 seconds, is only 0.2 g; and the frequency of oscillation is about 3.5 Hz.

The oscillatory mode becomes dominant when the computing precision is reduced to 8 bits, with the peak-to-peak amplitude becoming 1.9 g at a frequency of 3.0 Hz. The oscillation slowly decays but whether it disappears eventually was not investigated, since the very lightly damped response is in itself unacceptable.

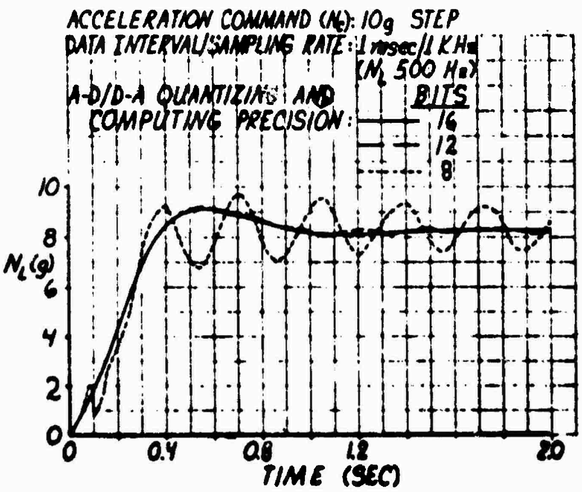
The amplitude of the oscillation is of interest because it was predictable from the discussion in the previous section of this report where an 8 bit/1 msec system was shown to be unable to pass a step command below 0.7 g. Twice this value, or 1.4 g would be the minimum peak-to-peak amplitude square wave that could be passed, leading to the suspicion that internal quantizing in the integrator is the cause of the oscillatory response of Figure 115. However, when quantizing was virtually removed from the integrator (32 bits) but left everywhere else, the oscillation was not completely eliminated, but the amplitude was reduced to 0.9 g and frequency remained unchanged.

To determine whether truncation of the fin actuator command was at fault, the above case (un-quantized integrator) was re-run with the rate loop closure and the D-A quantized to 16 bits. This left only the structural filter, accelerometer loop closure, and middle loop closure quantized at 8 bits. Results were only slightly affected, the oscillation amplitude remaining at 0.9 g.

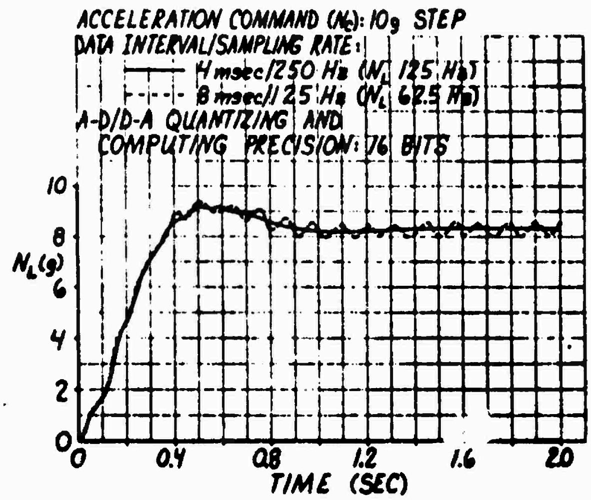
The investigation was not continued further since quantization of the A-Ds (0.25 g, 2 deg/sec quantum levels) and internal quantizing of the structural filter undoubtedly cause the 0.9 g oscillation, and it is obvious that it would be difficult to make an 8 bit/1 msec system work satisfactorily.

5.2.4 <u>Data Sampling/Update Rate</u>

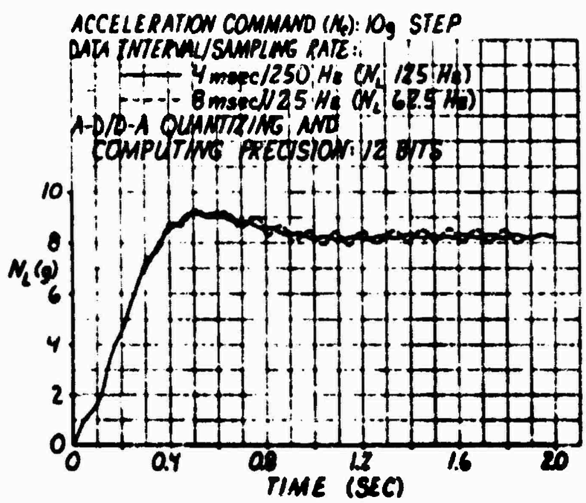
Figures 115(B), (C) and (D) show the autopilot step responses corresponding to computing precisions of: 16, 12, and 8 bits respectively, at data intervals of 4 and 8 msec. Both the 16 and 12 bit systems have negligible oscillations at the shorter interval, but both have approximately 0.5 g peak-to-peak oscillations at 10 HZ when the data interval is lengthened to



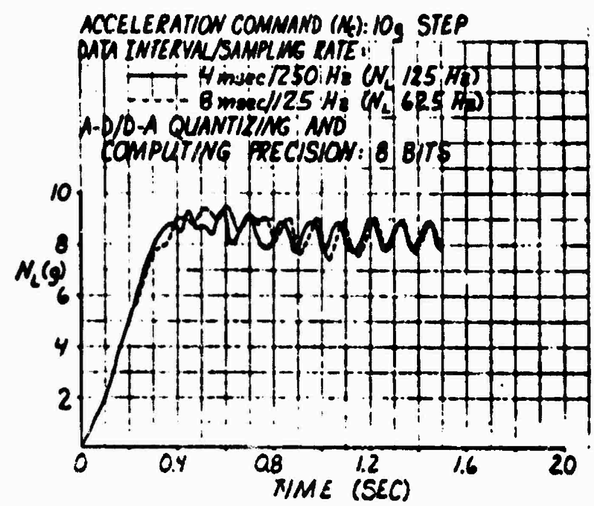
(A) High Data Rate (1KHz) vs Precision/Quantizing



(B) High Precision/Quantizing (16 Bits) vs Data Interval/Rate



(C) Medium Precision/Quantizing (12 Bits) vs Data Interval/Rate



(D) Low Precision/Quantizing (8 Bits) vs Data Interval/Rate

Figure 115 Digital Autopilot Step Responses

8 msec. The 8 bit system, however, exhibits a 1 g oscillation at either of the two data intervals, and in both cases this is less than the amplitude for a 1 msec interval. This is consistent with the previous discussions, in that, allowing the integrator more time to integrate helps to alleviate some of the adverse effects of quantization. Figure 116 summarizes the above effects with a plot of oscillation amplitude versus data interval for 8, 12, and 16 bit precision.

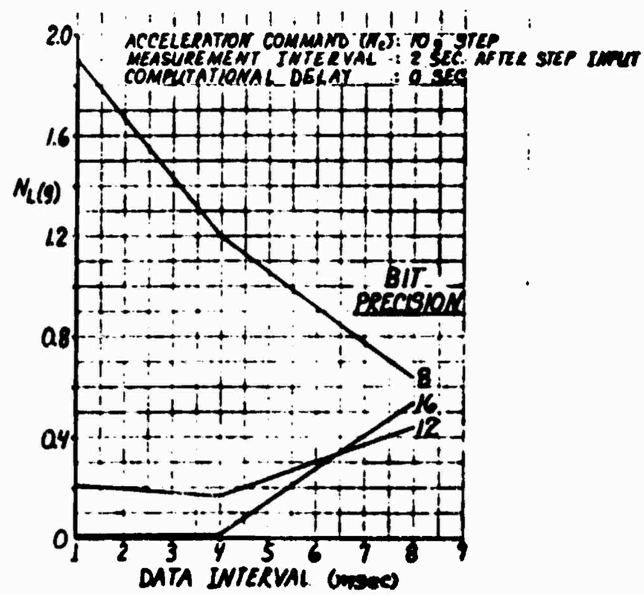


Figure 116 Digital Autopilot Step Response Oscillation Amplitude vs Data Interval.

The above results can be presented in a more concise way by considering the duty cycle of the fin actuator over the two seconds of the step response. Duty cycle is the total travel of the fin, obtained by continuously integrating the absolute value of the fin rate. Since everything in the simulation other than the digital computer model is linear, an oscillation in the missile g response means that fin must also be oscillating. The larger and faster the oscillation, the greater will be the fin duty cycle and the total volume of hydraulic oil used/passed in the actuator.

Figure 117 shows the duty cycle resulting (over 2 seconds) from step responses of the different systems considered above. The 32-bit curve is essentially an un-quantized system, and it can be seen that a 16 bit system follows it closely. The 32-bit/1 msec point (4.5 degrees duty cycle) can be taken as the baseline value, and going to an 8 msec interval obviously increases the duty cycle by almost an order of magnitude, corresponding to the observed fact that the 16-bit system oscillates for data intervals of 8 msec.

The 8-bit system is too lightly damped to be acceptable, but the 12-bit system actually outperforms the 16-bit at larger data intervals, an anomalous result pointing out the inherent non-linearities in a digital system. At smaller intervals, the 16-bit system is clearly superior, as expected. The fluctuations in the 12- and 8-bit curves of Figure 117 (all of the indicated data intervals were run on the simulation) are unexplained, but assumed to be nonlinear phenomena.

From these results it can be generally concluded that an acceptable digital autopilot should implemented with 16-bit precision and with an inner loop data interval of 4 msec maximum.

5.2.5 Computational Delay and Interface A-D/D-A Quantization

The effects of computer computation time and of D-A and A-D quantization were determined for the 16-bit/4 msec system. In Figure I18, fin duty cycle is presented as a function of these parameters. Obviously, the 8-bit interface gives considerably inferior results compared to the other quantization levels, but there is little basis for choosing a 16-bit A-D and D-A interface and if the computation time delay does not exceed 600 μsec, a 10-bit interface will suffice.

5.2.6 Computer Requirements Summary

Based on the simulation analyses described above, Table 47 summarizes the recommended computer parameter values for each generic missile class. The highest data rate (500 Hz) and shortest computational delay (600 μ secs) is required for the Class III system with some relaxing of these parameters to 250 Hz and 800 μsecs for Class I missiles.

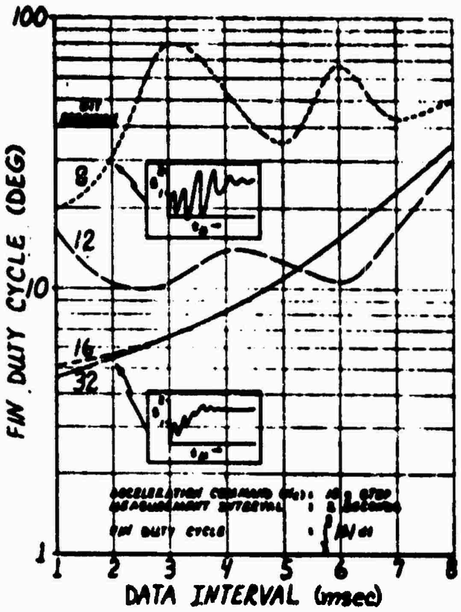


Figure 117 Digital Autopilot Step Response Fin Duty Cycle vs Data Interval

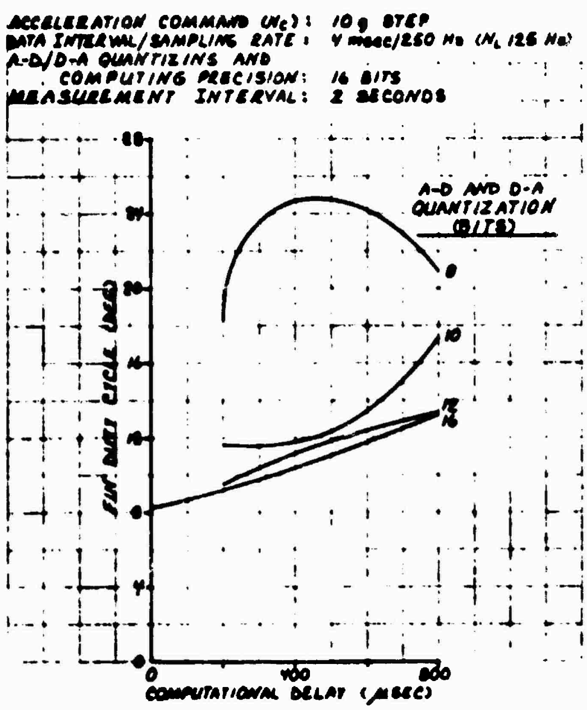


Figure 118 Digital Autopilot Step Response Fin Duty Cycle vs Computational Delay

TABLE 47

AUTOPILOT

COMPUTER REQUIREMENTS

Parameter	Missile Class I	II	III
Computing Precision (Bits)	16	16	16
A-D/D-A Quantization (Bits)	10	12	12
Computational Delay (μ secs) (each of 3 channels)	800	660	600
Rate Loop Data Interval (msecs)	4	2	2
Equivalent Adds (autopilot and structural filters)	183	615	615
Throughput (Kaps) (autopilot and structural filters)	76	315	342

It should be noted that the Class I data interval has been reduced to 4 msec from the 8 msec shown in the Phase I final study report.

The greatest change in the data reported in Phase I study report is in the throughput (Kaps) required for the autopilot and structural filter functions. Previously, the equivalent adds needed by each Class were averaged over the data interval, but it now becomes clear from the performance simulations that the full interval cannot be used to perform the autopilot calculations. Instead, these calculations must be executed within the allowable computational time delay, resulting in the increased throughput values shown in Table 47.

It should also be noted that these values apply only to the "basic autopilot", (Module A1), and the "structural filter and fin mix", (Module A2), which are the high data rate modules discussed in the Phase I report, compared to the 20 Hz induced roll reduction, gain determination and angle of attack variation modules.

5.3 Signal Processing Performance

Of the seven radar operational modes defined and discussed in subsection 4.2.4 the target acquisition mode has been selected to evaluate the effectiveness versus complexity of digital signal processing techniques and algorithms, since this mode is the most crucial in the target engagement process.

5.3.1 Signal Processor Tradeoff Parameters

The primary requirements that affect target acquisition, given in Table 18 subsection 4.2.5, are:

Doppler Ambiguity: 20 KHz

Range Ambiguity: 1/PRF

Probability of Detection: 0.95 in 1.0 sec

False Alarm Time: 1.0 sec

Range ambiguity is specified as 1/PRF since for all of the range gated systems of interest in this study, the range ambiguity due to the interpulse spacing of the pulse-doppler waveforms is less than the AI radar range designation accuracy. The primary signal processor parameters that affect acquisition performance are:

1. Number of ranges gates implemented
2. Range gate width
3. Number of points in FFT
4. FFT doppler cell width
5. Number of dwells post-detection integrated

The manner in which these parameters are interrelated may be seen by considering the data presented in Figure 119 which shows the single look probability of detecton vs the required cumulative probability of detection and the number of observations (looks).

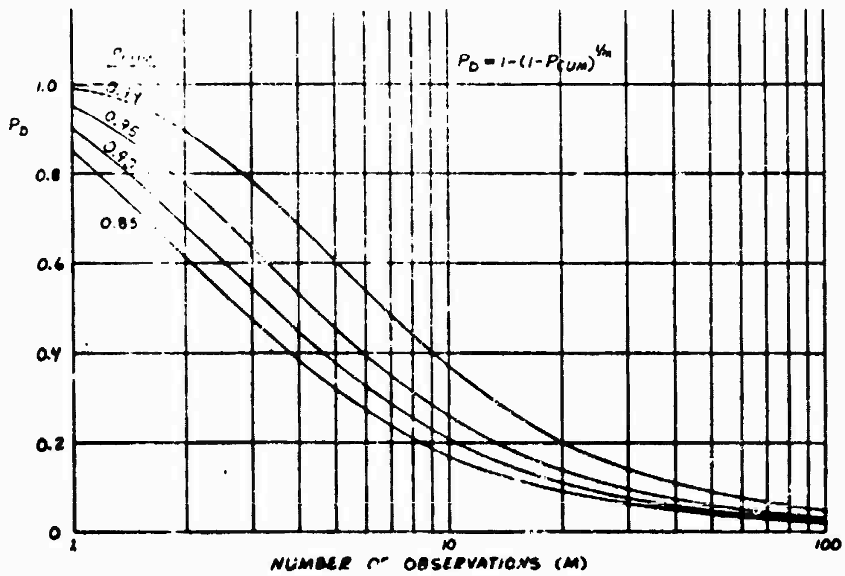


Figure 119. Single Look Probability of Detection (P_D) vs Required Cumulative Probability of Detection (P_{CUM}) and Number of Observations (M)

The time it takes to make one observation of the total range-doppler ambiguity is called the frame time. If the complete doppler ambiguity is spanned by a single FFT spectrum and the complete range ambiguity is spanned by a bank of range gates, one frame consists of a single range-doppler "look". On the other hand, if the FFT spectrum covers 1/3 of the doppler ambiguity and the number of range gates x pulse width covers 1/3 of the range ambiguity, it takes 3x3=9 looks to make up 1 frame.

The single look probability of detection depends on the signal-to-noise ratio available per range-doppler cell and upon the allowable false alarm probability per range-doppler cell. The signal-to-noise ratio per range doppler cell depends on the doppler cell width for thermal noise (SNR ~ $1/\beta_{CELL}$) and upon both the doppler cell width and range gate width for clutter environments (SNR ~ $1/\beta_{CELL} \times \tau_{GATE}$). The required single cell false alarm probability is (Reference 5-3).

$$P_{FA} \approx \frac{0.69}{n'}$$

where n' is the number of range-doppler cells examined in the false alarm time, T_{FA}

n' = (No. of range-doppler cells per frame) x (T_{FA}/frame time)

The relationship between P_{FA} and P_{DET} is illustrated in Figure 120. It is apparent that to achieve a specified P_{FA}, in a given noise environment, that the threshold bias, ν_B, must be adjusted accordingly. The threshold bias level directly affects the signal-to-noise level necessary to achieve a specified detection probability. Therefore, having decided on the number of range-doppler cells to be examined in the false-alarm time, the signal-to-noise ratio to achieve the required single look detection probability follows directly. Section 5.3.2 presents tradeoffs for the four candidate radar sensor signal processing systems that were specified in subsection 4.2.3. The common measure of performance for the various systems and their parameter variations is the signal-to-noise ratio in a 100 Hz bandwidth to achieve the specified cumulative detection probability and false alarm time.

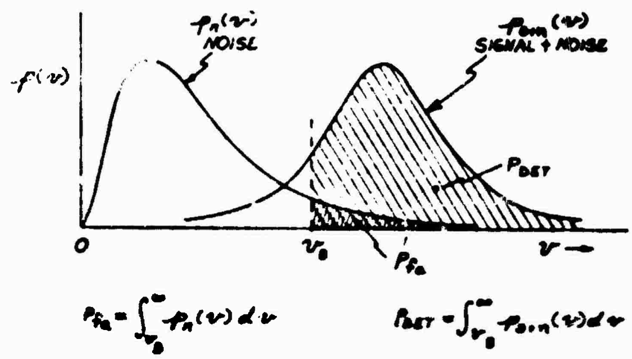


Figure 120. Probability of Detection and False Alarm Relationships.

The maximum allowable post-detection-integration time is dictated primarily by target acceleration/deceleration along the missile-target line-of-sight. This is seen by considering the following equations:

Doppler Frequency - $f_D \approx -\frac{1}{\lambda}\dot{R}_{MT}$

Doppler Rate-of-Change - $\dot{f}_D \approx -\frac{1}{\lambda}\ddot{R}_{MT}$

Where: R_{MT} - Missile-to-target range

$\ddot{R}_{MT}$ - $2(a_{MSL}/_{LOS} + a_{TGT}/_{LOS})$, the acceleration along LOS.

λ - Wavelength

The doppler frequency shift due to missile acceleration/deceleration is compensated for in the guidance data processors target tracking filter which utilizes integrated missile longitudinal acceleration. The doppler frequency rate-of-change due to target acceleration only (At X-Band) is then:

$$\dot{f}_D \Big|_{TGT} = 20 * a_{TGT} \Big|_{LOS}$$

The maximum expected target LOS acceleration is approximately 6 g's (194 fps²/59 mps²) which gives

$$\dot{f}_D \Big|_{6gTGT} = -3860 \text{ HZ/SEC}$$

The change in doppler for this acceleration as a function of observation time is:

T(sec)	Δf(Hz)
0.01	39
0.05	193
0.10	386
0.20	772

These results indicate that for a FFT doppler cell width of 200 Hz, the target doppler will shift by one whole doppler cell in 50 msec which is ten 5 msec dwells. The net effect is to reduce the effective post-detection-integration SNR gain during the times of peak target acceleration. This effect is somewhat mitigated by the spectral spreading caused by burst amplitude weighting.

5.3.2 Signal Processing Performance Tradeoffs

CW Radar Sensors - Performance tradeoff data for the CW radar sensor are shown in Figures 121 and 122.

The data presented in Figure 121 shows the SNR required in a 100 HZ bandwidth to achieve the specified cumulative probability of detection and false alarm time as a function of roughing filter bandwidth and FFT size. In deriving this data it was assumed that the effective number of FFT cells was as follows:

FFT Size	Effective Number of FFT Cells
32	25
64	50
128	100

It was also assumed that the roughing filter bandwidth is divided by the effective number of FFT cells to obtain the FFT cell width and required dwell time (e.g. for 32-pt FFT and 5 KHz roughing filter bandwidth, B_{CELL} = 5000/25 = 200 Hz and T_{DWELL} = 1/200Hz = 5 msec).

From Figure 121 it is seen that the 32-pt FFT gives the best performance with a 5 KHz roughing filter, the 64 or 128-pt best performance with a 10 KHz roughing filter and the 128-pt FFT with a 20 KHz roughing filter. The common factor for the three "best" choices is the FFT cell width of 200 Hz corresponding to a 5 msec dwell. The nominal SA-CW radar sensor specified in Table 18 (subsection 4.2.5) has a roughing filter bandwidth of 10 KHZ, a 64-point FFT, and a 5 msec dwell which appears to be a reasonable choice based on the data of Figure 121.

Figure 122 shows the effect of the number of dwells post-detection integrated for a 64-point FFT as a function of roughing filter bandwidth. Again ten dwells seems to be a reasonable choice, and the limitations on post-detection-integration time are not violated.

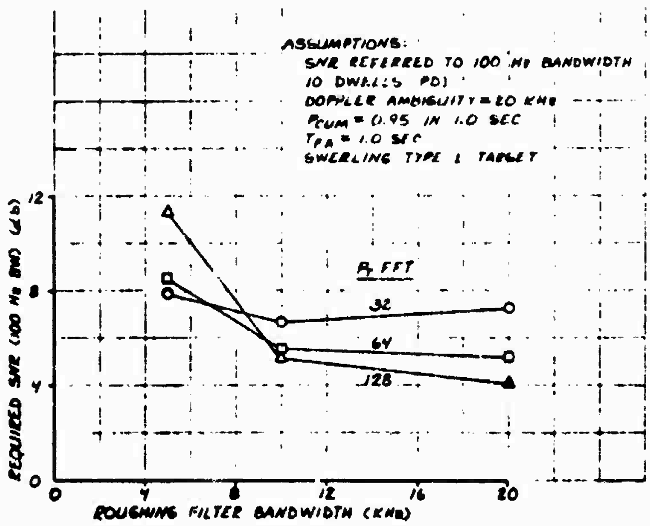


Figure 121. SNR vs Roughing Filter Bandwidth and FFT size for SA-CW Radar Sensor

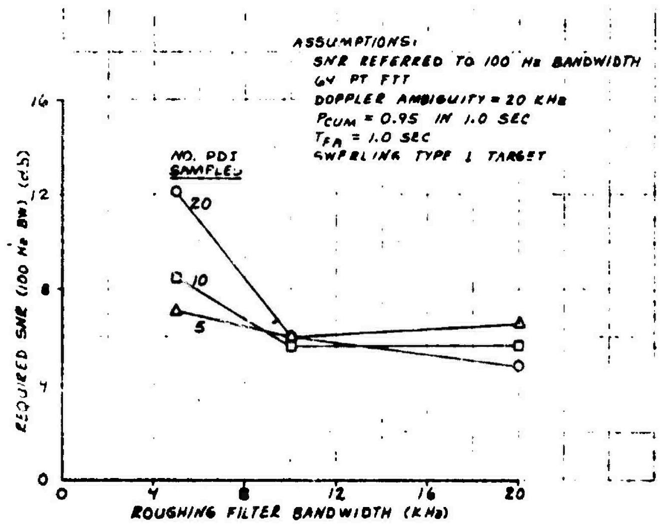


Figure 122. SNR vs Roughing Filter Bandwidth and Number of Samples PDI for SA-CW Radar Sensor

PD Radar Sensors - Performance tradeoff data for PD radar sensors is shown in Figures 123, 124 and 125. The data in Figures 123 and 124 show the signal-to-noise ratio required in a 100 HZ bandwidth to achieve the specified cumulative detection probability and false alarm time for a 64 or 128-pt FFT as a function of roughing filter bandwidth and the fraction of the range ambiguity covered by the range gate bank.

A ratio of R_{GATE}/R_{AMB} = 1.0 implies the total range ambiguity is covered by the sensors range gate bank. Note, for this example it is assumed that 16 gates are required to cover the range ambiguity. (A maximum of 10 gates corresponding to the candidate A-PD sensor, would not make a significant difference). This data shows the expected result that covering a larger percentage of the range ambiguity covered, the more frames per second can be achieved, thus lowering the per frame SNR requirement. The candidate SA-PD sensor for the Class II missile has 5 gates covering 1/3 of the range ambiguity, a 64-point FFT a 10 KHz roughing filter, 5 msec dwell, with 10 dwells post-detection-integrated. The candidate A-PD Class III sensor has the same parameters except that is has 10 gates covering the complete range ambiguity. The A-PD parameters appear to be "near optimum" while the SA-PD acquisition performance could be improved with increased mechanization (more "cost"). The performance of the 128-point FFT is reduced over that of the 64-point FFT primarily because smaller doppler cells are used to cover the roughing filter bandwidth. This results in more range-doppler cells for the specified false alarm time thus lowering the false alarm probability per cell which is achieved by raising the threshold. The increased threshold requires a larger per cell SNR to achieve the specified per frame SNR.

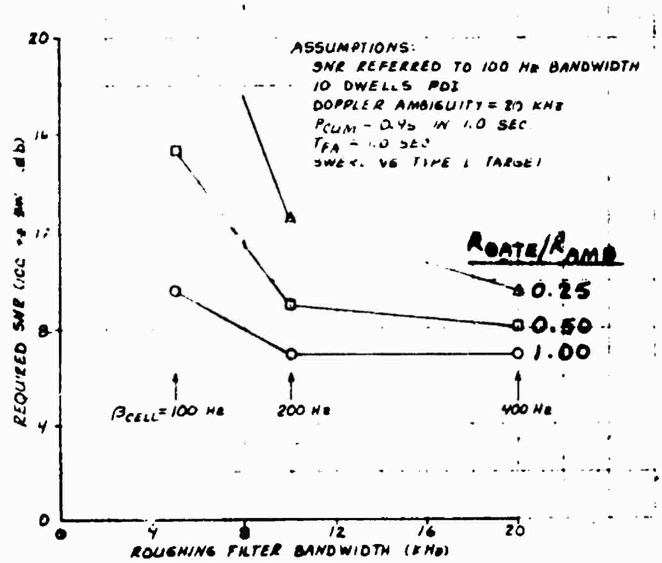


Figure 123. SNR vs Roughing Filter Bandwidth and Range Gate Coverage for 64-point FFT

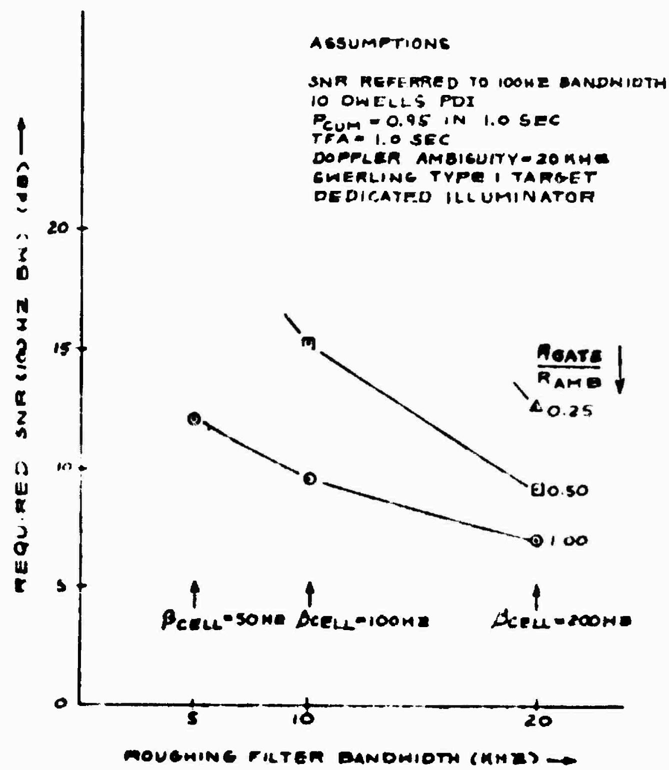


Figure 124. SNR vs Roughing Filter Bandwidth and Range Gate Coverage for 128-point FFT

The effect of the number of dwells post-detection-integrated for a 64-point FFT and 1/3 range ambiguity coverage is shown in Figure 125 as a function of roughing filter bandwidth. It is seen that 10 dwells again appears to be a reasonable choice.

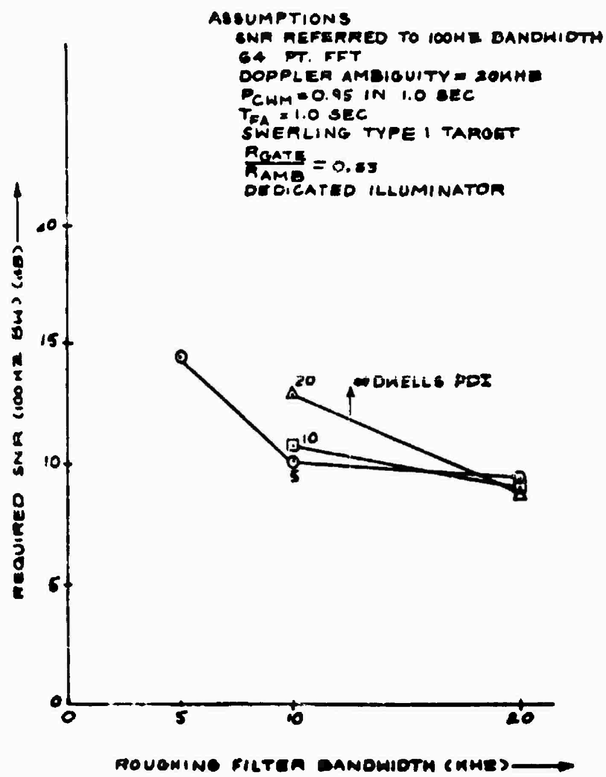


Figure 125. SNR vs Roughing Filter Bandwidth and Number of Dwells PDI for PD Radar Sensor

5.3.3 Required SNR vs Computer Throughput

Based on the previous performance data, it should be possible to buy improved acquisition performance at the expense of increased computing load. That this is the case is shown in Figures 126 and 127. Figure 126 shows the required acquisition signal-to-noise ratio (in a 100 Hz reference bandwidth) vs signal processing computer throughput rate (measured in kops) and for the SA-CW missile. The parameters used in computing this curve are shown on the figure. The primary variable parameter is the roughing filter bandwidth which varied from 5 to 20 kHz with the FFT cell width held constant at 200 Hz and the number of dwells post-detection-integrated held constant at 10. The Class I radar sensor parameters correspond to the middle data point on the curve.

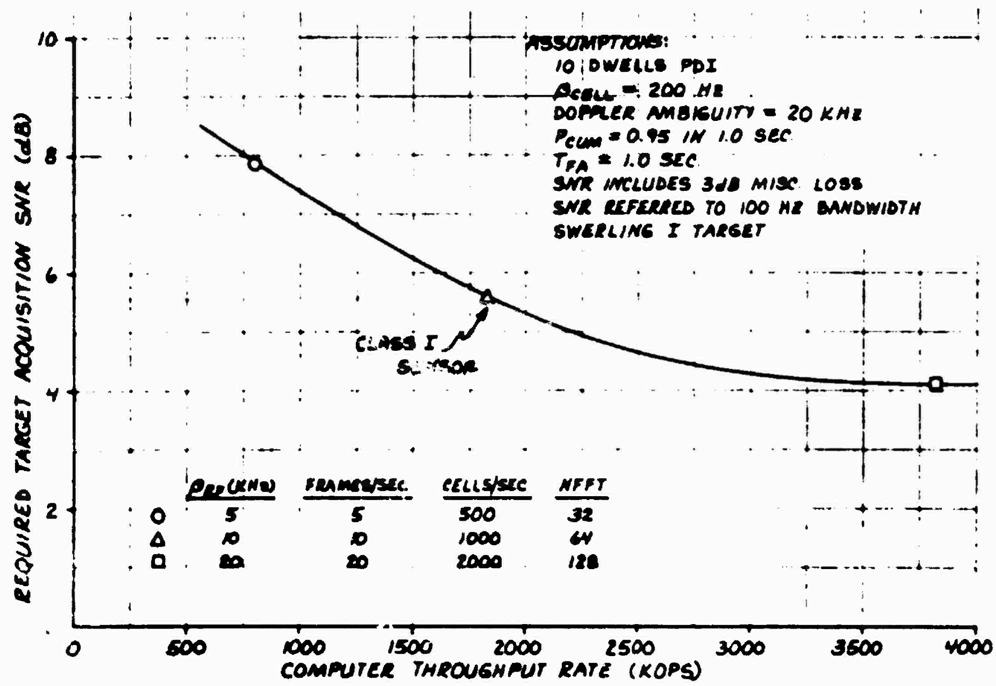


Figure 126 Required Target Acquisition SNR vs Worst-Case Signal Processing Computer Throughput Rate for SA-CW Radar Sensor

Figure 127 shows the required target acquisition SNR vs the worst case signal processing computer throughput rate for the pulse-doppler radar sensor. The parameters varied to generate this curve were the number of points in the FFT and the number of range gates mechanized relative to the number of range gates required to span the complete range ambiguity. The parameters held constant were the range-doppler cell size (0.2 μsec x 200 Hz), the range doppler target ambiguity region (3.2 μsec x 20 kHz), and the number of dwells post-detection integrated (10). As can be seen from the resulting curve(s), there is a very definite tradeoff between computer throughput rate and the required target acquisition SNR. For reference purposes, the data from Figure 126 has been included showing how the SA-CW requirements compare to PD. It is quite obvious the SA-CW system is superior in terms of computer throughput to achieve a specified target acquisition SNR. However, it must be pointed out that this data applies to target acquisition in a thermal noise limited environment. The PD sensor will have a clear performance advantage in a clutter environment (e.g. against low altitude receding targets). Also, the PD sensor has the ability to resolve targets in range which is important in engaging multiple target formations.

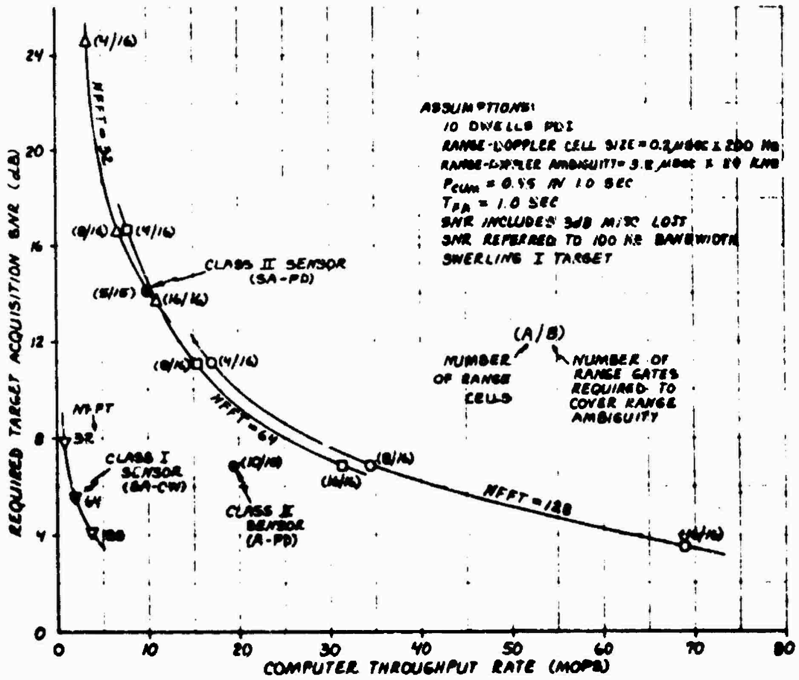


Figure 127 Required Target Acquisition SNR vs Worst-Case Signal Processing Computer Throughput Rate for Pulse-Doppler Radar Sensor

The "operating points" for the Class I, II, and III radar sensors are also shown in Figure 127. Note, that the Class III sensor (A-PD), uses less range gates to cover its range ambiguity along with a 64-point FFT. The reduction in computer throughput is directly a function of the number of gates for a given FFT size so that this point corresponds to the (16/16) point shown on the 64-point FFT curve.

6. MISSILE COMPUTER PERFORMANCE REQUIREMENTS

Based on the functional analyses and simulations described in the previous two sections of this report and the work contained in the Phase I Final Report (Ref. R.1), worst case computer requirements are presented in this section for each missile class in terms of throughput, as thousands of operations per second (Kops), and associated instruction mix, (i.e. percentage breakdown of add/subtract, multiply/divide, load/store/logical/branch), and number of data versus program memory locations. The expression of throughput in terms of thousands of equivalent adds per second (Kaps), as in the Phase I Final Report, has been discontinued in favor of Kops and instruction mix, since the latter is more useful in assessing the performance capabilities of different candidate computers. Kaps assume a fixed ratio of multiply and divide to add execution times, and hence apply to a specific computer. Computer requirements are given for each major missile function, (e.g. signal processing, guidance, autopilot, fuzing etc.), and in totum, as a composite load, to provide the flexibility to configure either a federated/distributed computer system or a single central computer mechanization.

6.1 System Timing Constraints

Worst case computer throughput requirements are driven by the allowable computational delay following data sampling i.e., the tolerable time lag between the instantaneous sensing of the real-time environment and the application of compensating missile control surface actions. The number of missile functions to be executed by a computer within the computational delay period is determined by the computer system configuration selected. Single computer systems would be required to time-multiplex all missile guidance and control functions, while maintaining the data sampling rate and computational delay criteria on an individual function basis, thereby creating a high throughput requirement.

In federated computer systems, whole or semi-autonomous functions would be assigned to separate, dedicated machines thereby minimizing the throughput requirement for each computer.

Two distinct computational delays can be identified in missile guidance and control systems determined by the type of control loop i.e:

1) Body motion

2) Guidance/Steering

Figure 128 illustrates the above loops in a typical missile guidance and control system incorporating a gimballed-platform type target seeker. Switches are shown with data sampling rates covering the respective ranges of all three classes of missile.

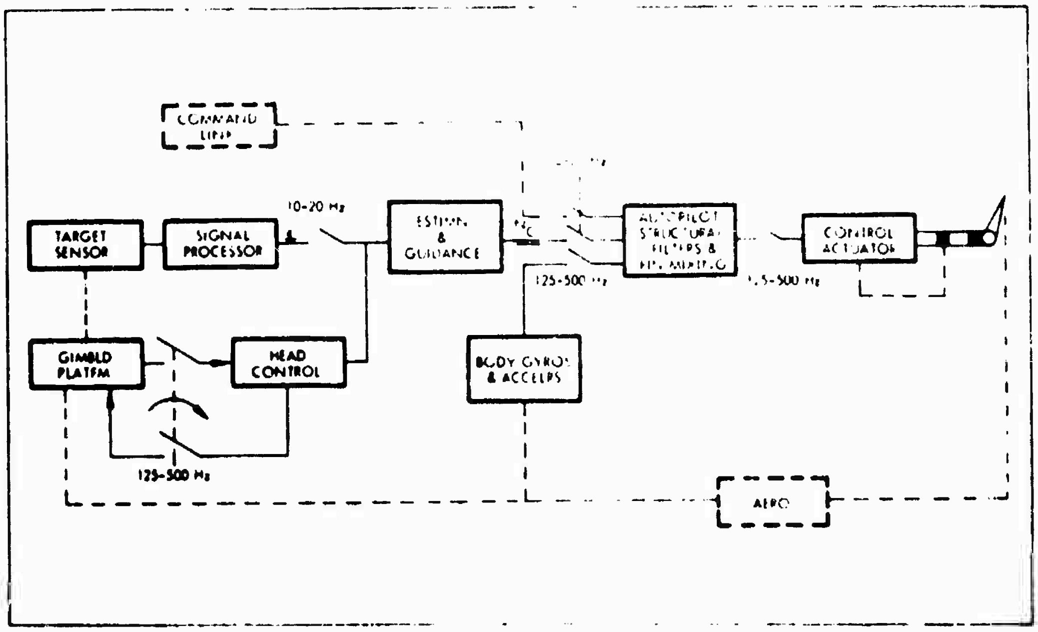
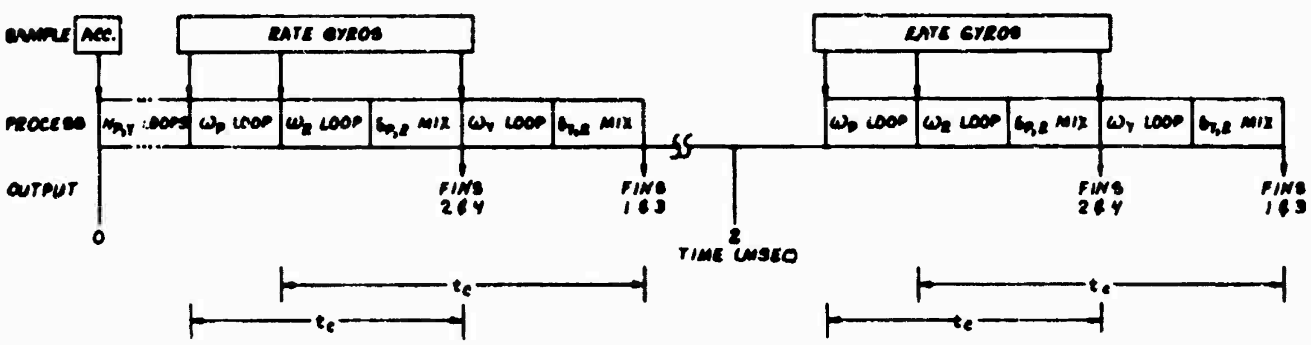


Figure 128 Sampled Data Missile Guidance and Control System Block Diagram

Body motion loops encompass the missile planar stability and control loops (i.e. autopilot) and target seeker gimballed-platform stabilization loops. The maximum allowable computational delay for such loops has been determined through simulations (subsection 5.4) to be 0.8 msec for Class I missiles and 0.6 msec for Class II and III weapons.

However, functions can be computed on an individual control channel basis (i.e., pitch, roll, yaw) with a certain degree of time-skew between channels, such that pitch channel functions can be computed and then the roll channel, and fin commands output to the corresponding fin control actuators (fin nos. 2 and 4), followed by the yaw channel mixed with the roll fin command for fin nos. 1 and 3, as illustrated in Figure 129.



NOTES:
1. DATA SAMPLING/UPDATE RATES
   STEERING COMMAND (N_c): 10 Hz
   BODY ACCELERATION ($\ddot{y}_m$): 250 Hz
   BODY RATE (ω_m): 500 Hz
2. t_c - COMPUTATIONAL TIME DELAY (>0.8/0.6 MSEC. FOR CLASS I/II, III MISSILES RESP)

Figure 129 Control Channel Computational Time Skewing.

No degradation of missile performance occurs for time-skewing within the limits shown in the figure. This characteristic can therefore be used to advantage in single computer systems where channel functions must be computed sequentially with resulting real-time skew between fin commands.

Guidance or steering loops involving target homing command generation, using on-board target sensors or, alternatively, a command data link, require considerably lower data sampling rates, compared to the body motion loops, and the allowable computational delay is correspondingly greater i.e. 20 msec approximately.

6.2 Worst-Case Throughput

Taking into consideration the above system time delay constraints and the operation counts given in Section 4 of this report and the Phase I Final Report, worst-case throughputs and associated instruction mixes have been determined for each major function and as a composite load using the following relationships:

$$\text{Throughput (Kops)} = \frac{(N + 0.3N)}{T_C}$$

$$\text{Mix (\%)} = \frac{100.N_{A/S}}{N + 0.3N} \qquad \frac{100.N_{M/D}}{N + 0.3N} \qquad \frac{100\,(N_{L/S} + 0.3N)}{N + 0.3N}$$

Where, N - Total number of critical-path computer operations, regardless of type, as given in Section 4.

T_C - Allowable computational delay in milliseconds.

$N_{A/S}$ - Number of add and subtract operations

$N_{M/D}$ - Number of multiply and divide operatons.

$N_{L/S}$ - Number of memory to register load and store operations.

As stated earlier, the program module instruction counts, given in the computer requirements tables of Section 4, are increased by 30% when converting to Kops and associated instruction mixes, to allow for the additional short operations necessary to achieve a fully operational program.

6.2.1 Major Functions

Tables 48 through 50 list all major missile functions and the corresponding throughput and instruction mix requirements for each missile class. Functions have been segregated into body motion, steering and satellite/support categories in accordance with the basic system timing constraints. Sensor signal processing loads are given for the radar clutter acquisition mode and the target track mode since, although the former is not used in the missile steering function, it nevertheless entails the greatest number of computer operations and highest throughput compared to all other sensors and modes.

TABLE 48

WORST-CASE THROUGHPUT REQUIREMENTS

(BY MAJOR FUNCTION)

CLASS I (MAX.) MISSILE

MAJOR FUNCTION	NO. OF OPERATIONS	COMPUTATIONAL DELAY (msec)	THROUGHPUT[1] (Kops)	INSTRUCTION MIX[1]
<u>BODY MOTION/STABILITY LOOPS:</u>				
Head Control (S1)	101	0.8[2]	82.0[2]	13/15/72
Autopilot (A1)	78	0.8[3]	84.5[3]	17/9/74
<u>STEERING LOOP</u>				
Signal Processing[4] (SP-3, SP-8, SP-9, SP-10, SP-20)	70,337 (12,737)	50.0	1,829	16/11/73 (18/5/77)
Signal Processing[5] (SP-6, SP-11, SP-14, SP-15, SP-17, SP-18, SP-21)	19,573 (1,333)	20.0	1,285	16/11/73 (32/6/62)
Estimation ($E1_2$)	159			19/21/60
Guidance (G1)	40			13/25/62
<u>SATELLITE/SUPPORT</u>				
Telemetry	50	4.0	16.3	0/0/100
Fuzing	14	20.0	0.9	16/5/79

<u>NOTES:</u>

(1) Includes 30% additional short operations for subroutine linkages and other miscellaneous overhead operations.
(2) One Channel
(3) Two Channels
(4) Radar Clutter Acquisition Mode (Missile Target Acquisition Submode), ten 5 msec dwells.
(5) Radar Target Track Mode (Missile Terminal Mode)

( ) Without FFT Spectrum Analysis

TABLE 49

WORST-CASE THROUGHPUT REQUIREMENTS

(BY MAJOR FUNCTION)

CLASS II (MAX.) MISSILE

MAJOR FUNCTION	NO. OF OPERATIONS	COMPUTATIONAL DELAY (msec)	THROUGHPUT[1] (Kops)	INSTRUCTION[1] MIX
BODY MOTION/STABILITY LOOPS:				
Head Control (S1)	101	0.6[2]	109.4[2]	13/15/72
Autopilot (A1, A2)	182	0.6[3]	262.8[3]	15/13/72
STEERING LOOP				
Signal Processing[4] (SP-7, SP-8, SP-9, SP-10, SP-20)	368,505 (164,505)	50.0	9581	16/10/74 (19/4/77)
Signal Processing[5] (SP-6, SP-11, SP-14, SP-15, SP-16, SP-17, SP-18, SP-20)	31,733 (1,333)	20.0	2094	16/11/73 (32/6/62)
Estimation (E2)	245			18/21/61
Guidance (G2)	239			16/21/63
SATELLITE/SUPPORT				
Attitude Reference (I1, I2, I3, I4, I5, I6)	3427	100.0	44.5	19/15/66
Telemetry	50	2.0	32.5	0/0/100
Fuzing	357	20.0	23.2	15/21/64

NOTES:

(1) Includes 30% additional short operations for subroutine linkages and other miscellaneous overhead operations.
(2) One Channel
(3) Two Channels
(4) Radar Clutter Acquisition Mode (Missile Target Acquisition Submode), ten 5 msec dwells.
(5) Radar Target Track Mode (Missile Terminal Mode)

( ) without FFT Spectrum Analysis

TABLE 50

WORST CASE THROUGHPUT REQUIREMENTS

(BY MAJOR FUNCTION)

CLASS III (MAX.) MISSILE

MAJOR FUNCTION	NO. OF OPERATIONS	COMPUTATIONAL DELAY (msec)	THROUGHPUT (Kops) (1)	INSTRUCTION MIX (1)
<u>BODY MOTION/STABILITY LOOPS:</u>				
Head Control (S1, S5)	295	0.6 (2)	319.6 (2)	17/12/71
Autopilot (A1, A2)	182	0.6 (3)	262.8 (3)	15/13/72
<u>STEERING LOOP</u>				
Signal Processing (4) (SP-3, SP-8, SP-9, SP-10, SP-20)	739,375 (131,375)	50.0	19,224	16/10/74 (19/4/77)
Signal Processing (5) (SP-6, SP-11, SP-14, SP-15, SP-16, SP-17, SP-18, SP-20)	31,733 (1,333)			15/11/73 (32/6/62)
		20.0	2103	
Estimation ($E4_2$)	375			16/23/61
Guidance ($G2_2$)	239			16/21/63
<u>SATELLITE/SUPPORT</u>				
Attitude Reference (I1, I2, I3, I4, I5, I6)	2226	10.0	289.4	19/14/67
Autopilot Gains (A5, A15, A16)	3271	100.0	42.5	23/10/70
Telemetry	50	2.0	32.5	0/0/100
Fuzing	318	2.0	20.7	16/24/60

<u>NOTES:</u>

(1) Includes 30% additional short operations for subroutine linkages and other miscellaneous overhead operations.
(2) One Channel
(3) Two Channels
(4) Radar Clutter Acquisition Mode (Missile Target Acquisition Submode), ten 5 msec dwells.
(5) Radar Target Track Mode (Missile Terminal Mode)

( ) Without FFT Spectrum Analysis

6.2.2 Composite/Single Computer

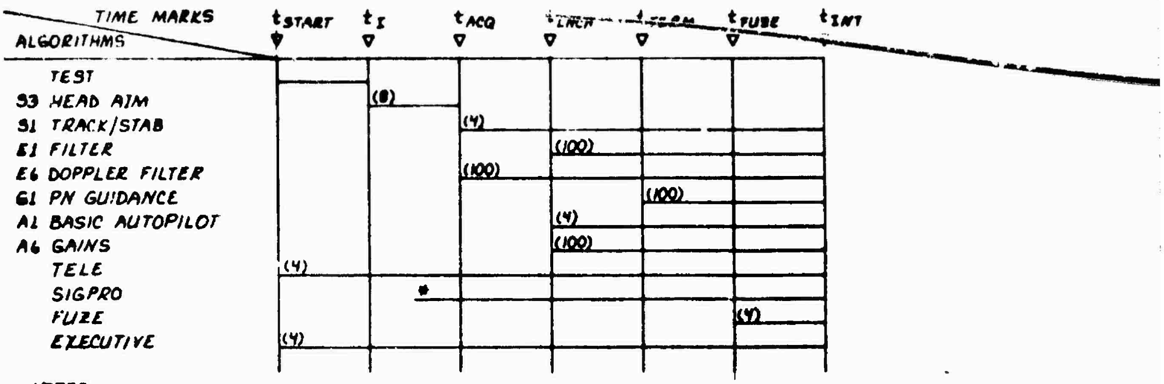
Due to the time multiplexing of all missile functions in the single computer case, the determination of worst-case throughput entails the following preliminary design steps.

1) Mission time line analysis

2) Critical path determination

Mission Time Line Analysis - Time line analyses have been performed to determine the worst-case function mix during the flight path/mission of each missile class. The results of these analyses were used to define the missile mode supervisor programs described in Section 4.6. Figures 130, 131 and 132 are function time-line diagrams for the three missile classes respectively.

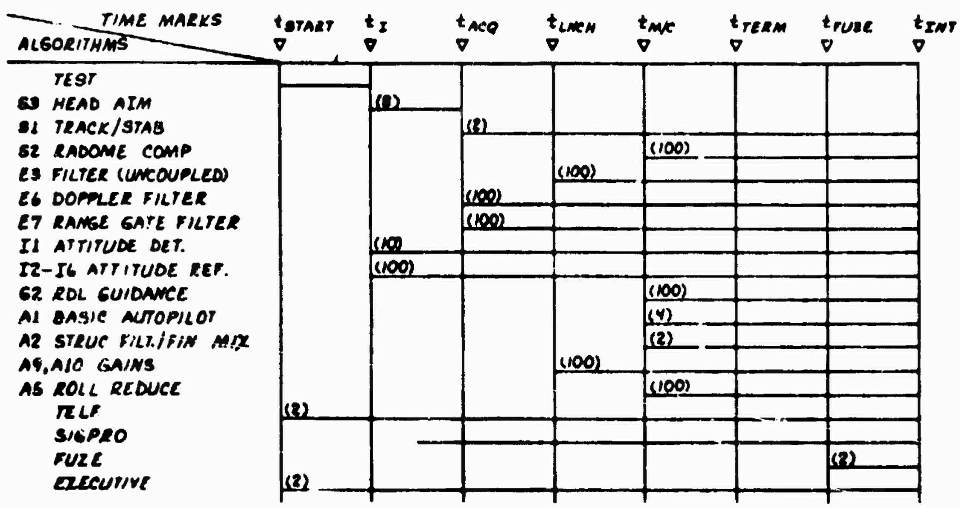
Such diagrams show function activity on a coarse timing basis. In terms of throughput, reference to Tables 48, 49 and 50 shows that, despite the greater number of functions being active in the missile Terminal Mode, the radar sensor signal processing throughput requirement for clutter acquisition far exceeds the aggregate throughput of all other functions. However, without radar signal processing, the missile Terminal Mode represents the most complex mode for single computer guidance and control systems.



NOTES:
( ) ALGORITHM UPDATE INTERVAL IN MILLISECONDS
t_I - SYSTEM IS INITIALIZED
t_{ACQ} - TARGET IS ACQUIRED
t_{LNCH} - MISSILE IS LAUNCHED
t_{TERM} - MISSILE ENTERS TERMINAL MODE
t_{FUZE} - FUZING CALCULATIONS BEGIN
t_{INT} - MISSILE INTERCEPTS TARGET

* SEE SECTION V FOR SIGNAL PROCESSING MODES AND CONTROL

Figure 130 Function Time Line Diagram - Class I (Max.) Missile



NOTES:
( ) ALGORITHM UPDATE INTERVAL IN MSEC.
t_I - SYSTEM IS INITIALIZED
t_{ACQ} - TARGET IS ACQUIRED
t_{LNCH} - MISSILE IS LAUNCHED
$t_{M/C}$ - MISSILE ENTERS MIDCOURSE MODE
t_{TERM} - MISSILE ENTERS TERMINAL MODE
t_{FUZE} - FUZING CALCULATIONS BEGIN
t_{INT} - MISSILE INTERCEPTS TARGET

Figure 131 Function Time Line Diagram - Class II (Max.) Missile

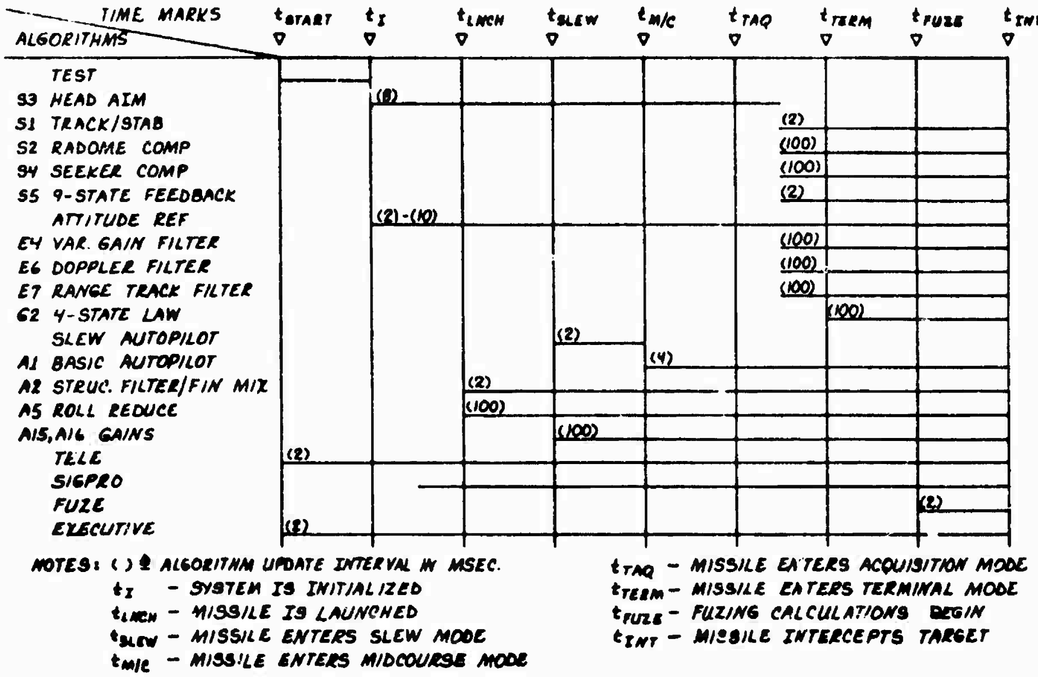


Figure 132 Function Time Line Diagram - Class III (Max.) Missile

Critical Path Determination - To determine the peak throughput for a single computer system, a fine timing analysis was performed using the 100 msec steering loop sampling interval and the Terminal Mode function mix for each missile class. Figures 133, 134 and 135 illustrate the distribution of processing functions over the 100 msec interval which is divided into 2 or 4 msec minor intervals depending on the highest data sampling rate used for the missile body motion/stability loops. Functions pertaining to the latter are completely executed during each minor interval subject to the additional computational delay constraint, and the remaining functions are assumed to be partially executed within each minor interval until completion within a specific multiple of such intervals according to the system timing constraints.

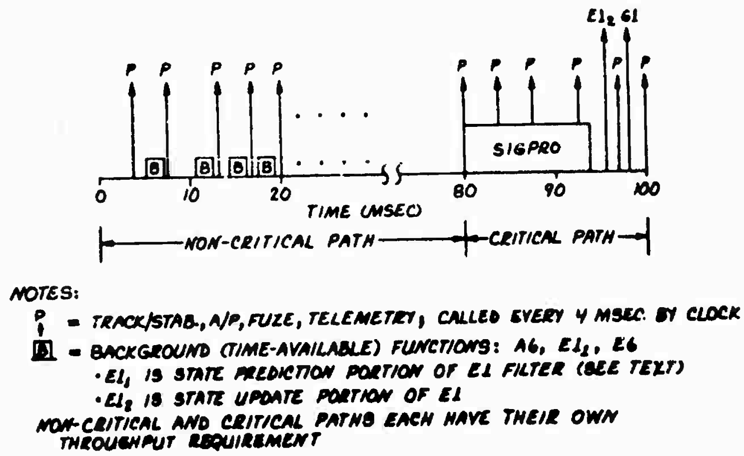


Figure 133 Function Multiplexing for Single Computer Systems, Terminal Mode, Class I (Max.) Missiles

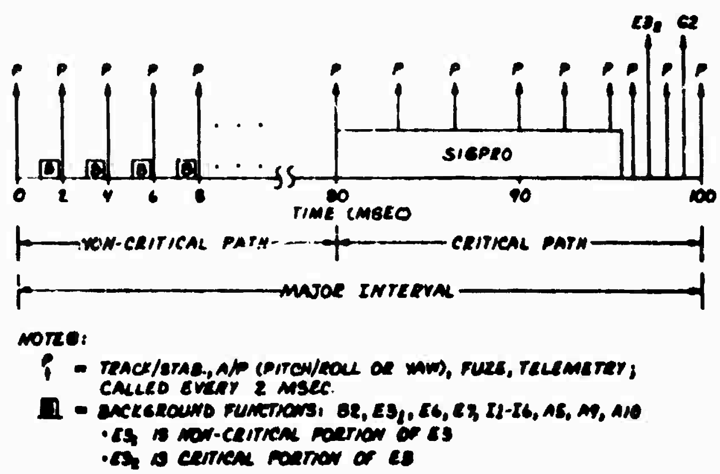


Figure 134 Function Multiplexing for Single Computer Systems, Terminal Mode, Class II (Max.) Missiles

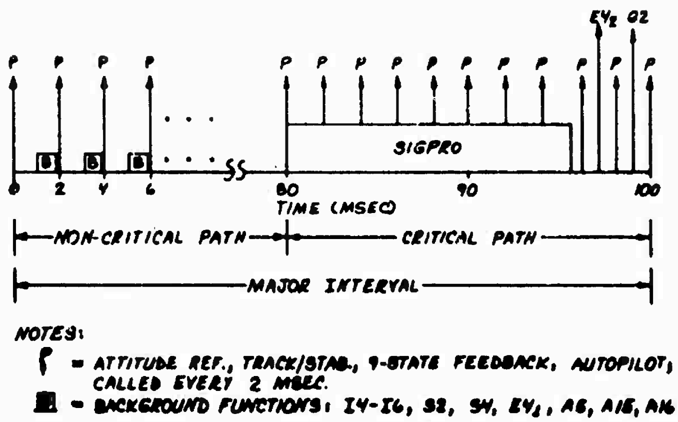


Figure 135 Function Multiplexing for Single Computer Systems, Class III (Max.) Missiles

The critical path or timing interval occurs when the signal processing functions are included in each 100 msec major interval. Since the allowable computational time delay for signal processing and associated estimation and guidance algorithms is limited to 20 msec it is important to minimize the throughput burden due to other functions during this interval. Hence, in Class I missiles, the state-prediction portion of the fixed-gain guidance filters is executed before the critical path period and only the filtering of boresight error signals is performed within the 20 msec period. Similarly, for Class II and III missiles, the Kalman filter algorithm (E3) is divided into two parts, the first part containing preparatory functions, using data computed during the previous major interval, such that the critical interval is devoted to executing algorithms which depend entirely upon current target tracking data.

Tables 51 through 53 summarize the worst-case throughput requirements, (including 30% contingency), for single computer systems by missile class for missile Acquisition and Terminal Modes respectively.

TABLE 51

WORST CASE THROUGHPUT REQUIREMENTS
FOR SINGLE COMPUTER SYSTEMS
CLASS I (MAX.) MISSILE

MISSILE MODE	FUNCTIONS	THROUGHPUT[1] (Kops)	INSTRUCTION MIX[1]
	All	1854	16/10/74
Acquisition[2]	W/O FFT	357	17/4/79
	W/O FFT&PDI	58	19/2/79
Terminal[3]	All	1376	16/11/73
	W/O FFT	191	20/8/72

NOTES

(1) Includes 30% additional short operations for subroutine linkages and miscellaneous overhead operations to achieve operational program.

(2) Average throughput over a 50 msec interval (i.e., ten 5 msec dwells)

(3) Average throughput over a 20 msec interval.

TABLE 52

WORST CASE THROUGHPUT REQUIREMENTS

FOR SINGLE COMPUTER SYSTEMS

CLASS II (MAX.) MISSILE

MISSILE MODE	FUNCTIONS	THROUGHPUT[1] (Kops)	INSTRUCTION[1] MIX
	All	9691	16/10/74
Acquisition[2]	W/O FFT	1787	18/4/78
	W/O FFT&PDI	290	23/4/73
Terminal[3]	All	2420	15/11/74
	W/O FFT	443	16/11/73

NOTES

(1) Includes 30% additional short operations for subroutine linkages and miscellaneous overhead operations to achieve operational program.

(2) Average throughput over a 50 msec interval (i.e. ten 5 msec dwells)

(3) Average throughput over a 20 msec interval.

TABLE 53

WORST CASE THROUGHPUT REQUIREMENTS

FOR SINGLE COMPUTER SYSTEMS

CLASS III (MAX.) MISSILE

MISSILE MODE	FUNCTIONS	THROUGHPUT [(1)] (Kops)	INSTRUCTION [(1)] MIX
	All	19,748	16/10/74
Acquisition [(2)]	w/O FFT	3,941	18/5/77
	W/O FFT&PDI	946	22/7/71
Terminal [(3)]	All	2782	16/11/73
	W/O FFT	805	17/12/71

NOTES

(1) Includes 30% additional short operations for subroutine linkages and miscellaneous overhead operations to achieve operational program

(2) Average throughput over a 50 msec interval (i.e., ten 5 msec dwells)

(3) Average throughput over a 20 msec interval

Figures 136 and 137 are plots of worst case throughput versus missile class with and without the radar signal processing loads respectively.

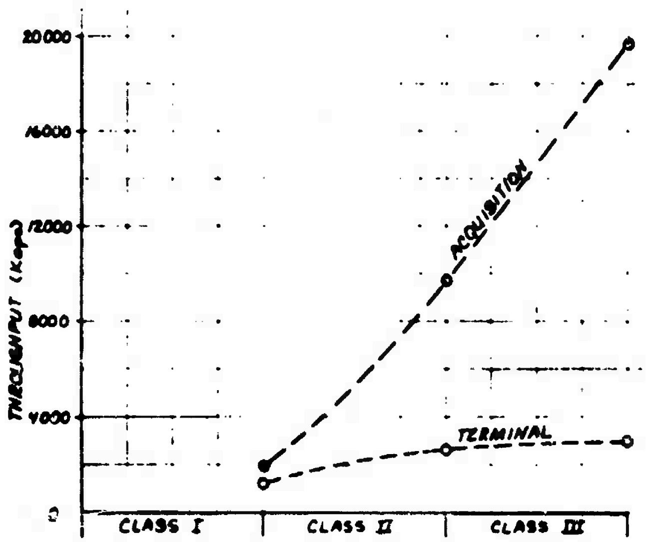


Figure 136 Worst-Case Throughput vs Missile Class for Single Computer Systems, Including Radar Signal Processing

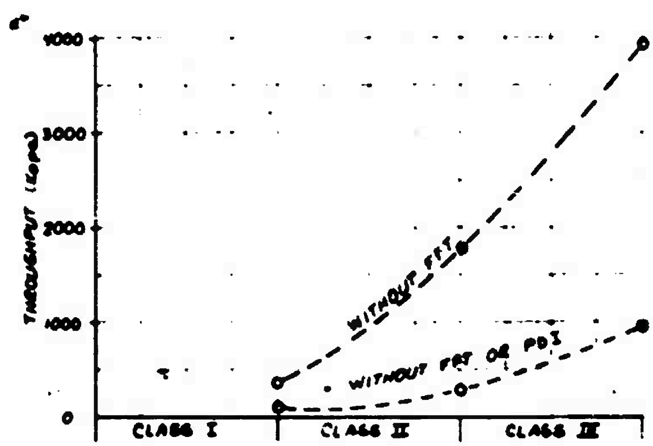


Figure 137 Worst-Case Throughput vs Missile Class for Single Computer Systems, Acquisition Mode, Without Radar Signal Processing

Add_Time - Since the add time is a fundamental performance indicator for a general-purpose computer, coupled with the ratio of multiply to add time, the composite operations counts and associated instruction mixes for the single computer case have been transcribed to provide a choice of multiply/add ratio within the bounds of the required worst-case throughputs.

Figure 136 through 140 provide multiply/add vs add time plots for each missile class and for composite throughput rates with and without radar signal processing. Add or multiply time refers to an instruction fetch, operand fetch, (i.e., addend or multiplier, both from main memory) and instruction execution with respect to the existing contents of the accumulator, (i.e., augend or multiplicand).

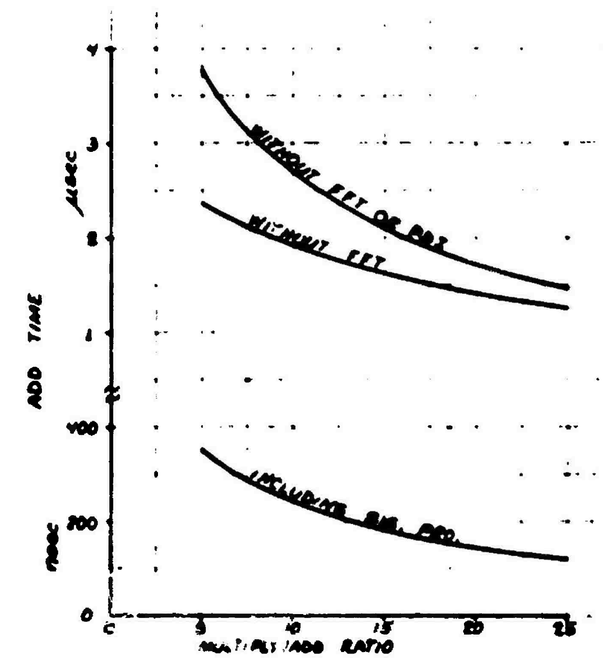


Figure 136 Computer Multiply/Add Time Ratio vs Add Times - Class : (Max.) Missile

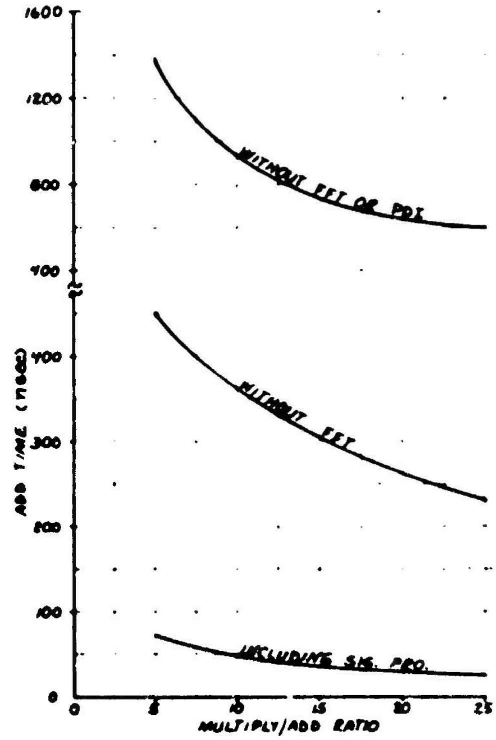


Figure 139 Computer Multiply/Add Time Ratio vs Add Time - Class II (Max.) Missile

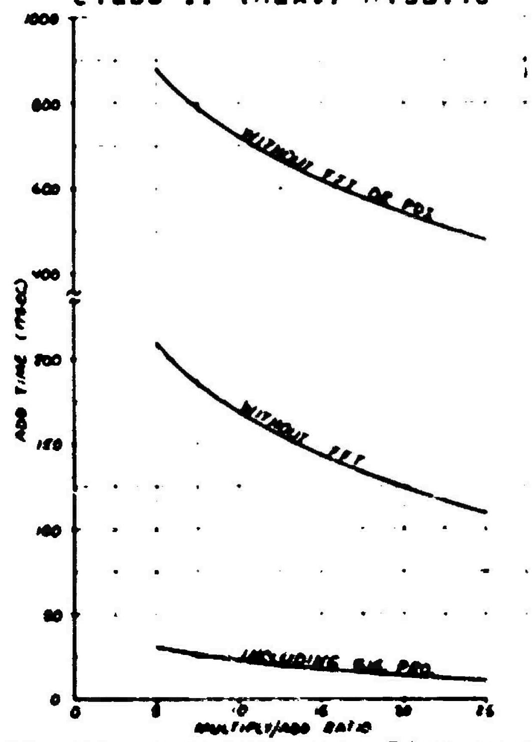


Figure 140 Computer Multiply/Add Time Ratio vs Add Time - Class III (Max.) Missile

6.3 Memory Requirements

Tables 54 through 56 present the worst-case memory requirements for each of the three classes of missile, by major function, and in terms of program (ROM), real-time data (RAM), and constants/table-look up data (ROM), to provide flexibility in computer and memory design techniques.

For single computer systems, composite, worst-case memory requirements are given in Table 57 for each missile class.

As stated in Section 4, all program memory requirements given in the tables include 30% additional instructions for subroutine linkages and other miscellaneous overhead operations necessary to achieve a completely operational program.

TABLE 54

WORST CASE MEMORY REQUIREMENTS

(BY MAJOR FUNCTION)

CLASS I (MAX.) MISSILE

FUNCTION	PROGRAM MEMORY (ROM)	DATA MEMORY (RAM)	DATA MEMORY (ROM)
Head Control	84	26	16
Autopilot	116	10	8
Signal Processing	975	506	150
Estimation	143	12	36
Guidance	52	7	4
Fuzing	46	6	4
Telemetry	50	53	
Test	281	29	50
Utilities	312	88	-
*Mode Control	280	-	-
TOTALS	2339	817	268

NOTES:

* For single computer systems

TABLE 55

WORST CASE MEMORY REQUIREMENTS

(BY MAJOR FUNCTION)

CLASS II (MAX.) MISSILE

FUNCTION	PROGRAM MEMORY (ROM)	DATA MEMORY (RAM)	DATA MEMORY (ROM)
Head Control	272	50	359
Autopilot	336	62	121
Signal Processing	975	2780	150
Estimation	606	38	-
Guidance	354	40	-
Attitude Reference	494	111	58
Telemetry	50	58	-
Fuzing	228	5	12
Test	364	29	54
Utilities	369	105	-
*Mode Control	488		
TOTALS	4536	3278	754

TABLE 56

WORST-CASE MEMORY REQUIREMENTS

(BY MAJOR FUNCTION)

CLASS III (MAX.) MISSILE

FUNCTION	PROGRAM MEMORY (ROM)	DATA MEMORY (RAM)	DATA MEMORY (ROM)
Head Control	449	62	359
Autopilot	1222	96	1222
Signal Processing	975	5510	150
Estimation	2610	71	-
Guidance	354	40	-
Attitude Reference	494	111	58
Telemetry	50	71	-
Fuzing	184	3	7
Test	561	57	80
Utilities	426	127	-
*Mode Control	629	-	-
TOTALS	7954	6148	1726

TABLE 57

TOTAL MEMORY REQUIREMENTS FOR SINGLE COMPUTER SYSTEMS

MISSILE CLASS	PROGRAM MEMORY (ROM)	DATA MEMORY ROM	DATA MEMORY RAM	TOTAL DATA MEMORY	TOTAL MEMORY
I	2339	268	817	1085	3424
II	4536	754	3278	4032	8568
III	7954	1726	6148	7874	15,828

7. MODULAR COMPUTER DEFINITION

In order to define the modular computer architecture and preferred software characteristics capable of supporting the entire range of missile functions and configurations analyzed in the previous sections of this report and the Phase 1 Final Report, while incorporating growth features to accomodate performance and technological improvements throughout the life cycle of a missile system, computer design requirements must be broadened to include the latter and other important system design considerations in addition to the throughput and memory requirements given in the previous section.

Although programmable digital techniques have been shown to offer improved performance and greater flexibility than traditional hardwired analog implementations, the direct substitution of a single, real-time, general-purpose, digital computer to perform the on-board guidance and control task does not provide an optimum solution in many cases. While throughput could be satisfied for all missile types with a single, high-performance, mini-class computer and a dedicated, special-purpose, sensor signal processor, an excessive performance margin results in Class I and II missiles. Also the centralization of a single standard computing unit presents form-factor incompatibilities across the range of missiles, together with a poor electrical interface. In addition to the latter deficiencies, peculiar to the missile application, design, assembly and checkout of major missile sections/functions (e.g., seeker, guidance, autopilot, attitude reference, umbilical/command-link interface, warhead fuzing) as completely operational modules is not possible with a central computer design approach.

In the light of the above preliminary observations and to achieve a more balanced design, the following all-inclusive criteria were established for this study task:

1. Missile form factors, design, construction and test
2. System growth to accommodate performance and technological improvements without major redesign
3. Missile subsystem and avionics interfaces
4. Computer loads, function antonomy and input-output traffic
5. Available computers, components and mature/proven architectural features
6. Avionics software experience.

Such a comprehensive top-down approach to the problem ensures a more practical modular design specifically for missile applications and provides a greater degree of flexibility for the missile system designer.

7.1 Missile Form Factors, Design, Construction and Test

7.1.1 Form Factors

Since the missile presents a unique form factor situation for the packaging of guidance and control components, profiles and dimensions of missiles representative of the three generic classes discussed in this report were reviewed (Figure 141). Body diameters typically range from 5 in/12.7 for Class I missiles, to 8 in/20.3 and 11 in/28cm for Class II & III missiles respectively. The functional partitioning and arrangement of major component parts in tandem along the longitudinal axis becomes a design prerequisite in each case due to the rigid constraints of the fundamental air vehicle design.

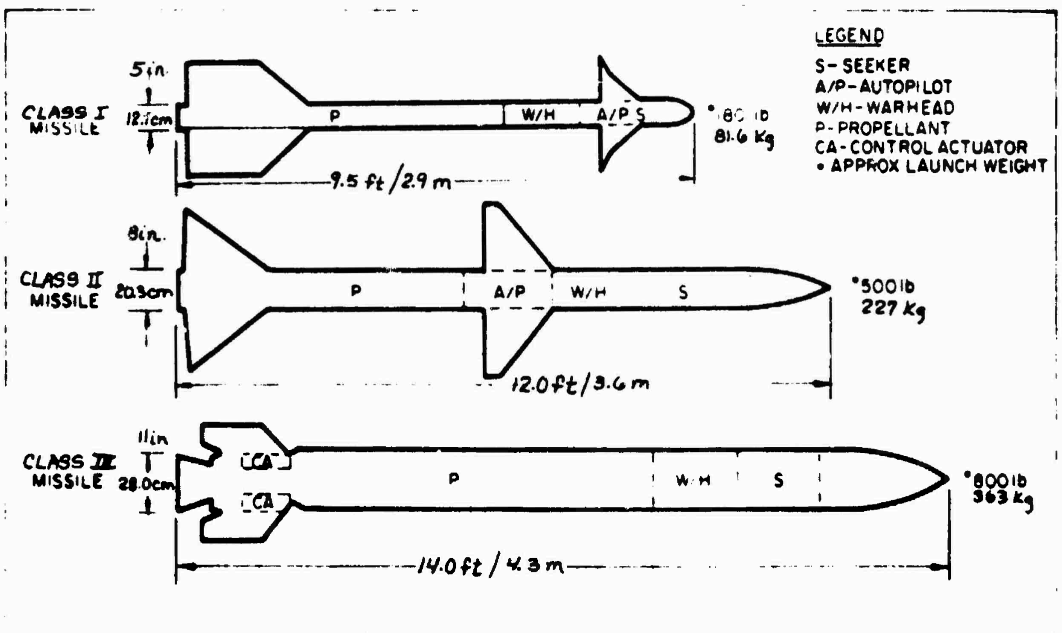


Figure 141 Air to Air to Air Missile Form Factors
Modularity and Packaging Constraints

7.1.2 Missile Design and Manufacture

Missile sections i.e., radome, seeker, warhead, autopilot/fins, propulsion unit and tail are designed, manufactured and tested as individual assemblies before final assembly into a complete missile.

7.1.3 Missile Maintenance Philosophy

In a similar manner to the design and manufacture of missiles, within Navy maintenance levels, guidance and control sections are given a go no-go check prior to assembly to the warhead and motor sections in the carrier electronic workshop (Intermediate Level). Faulty sections are replaced with a substitute section. Defective sections are returned to Overhaul and Repair shops at shore depots (Depot-Level) for corrective action.

7.1.4 Single vs Federated Computer Systems

The choice of a single central computer system versus a federated/distributed microcomputer system has important system modularity and interface implications in the context of the above missile form factor, design, manufacture, test and maintenance considerations, as illustrated in Figure 142.

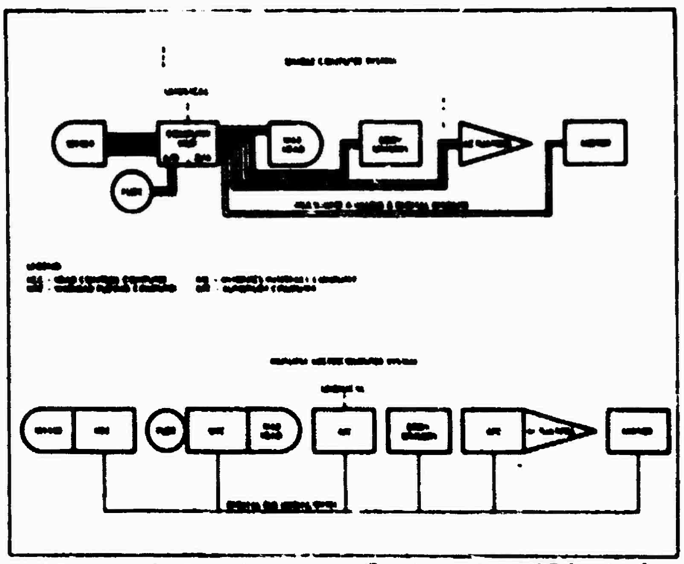


Figure 142 Single Computer vs Federated/Distributed Micro-Computer Within Missile Form Factor and Modular Assembly Constraints

The federated/distributed computer system supports the traditional subsystem autonomy of seeker, fuze and autopilot and, in addition, allows telemetry to be included/deleted without involvement in the operational software as in the case of the single computer system.

Similarly, system growth would be more straightforward if a complete seeker, guidance, autopilot or fuze assembly could be replaced with an improved version without upset to the remainder of the system. The following subsection explores this aspect of system design more fully.

7.2 System Growth Without Major Redesign

Table 58 shows the application of the various guidance and control function program modules defined in the Phase I Final Report, by missile class.

The implication of changes in module complements for each missile class from a growth aspect is shown in Table 59, where, performance improvements are correlated with actual module additions and/or deletions for each major function, including those addressed in the Phase II Study.

7.2.1 Single Computer Systems

For single computer systems, growth entails additions and/or deletions to the computer software, and this in turn demands a high degree of software modularity and discipline in the software generation and documentation process, since estimated program sizes range from 2K to 8K words over the three missile classes.

TABLE 58

GUIDANCE & CONTROL PROGRAM MODULE APPLICATIONS

Major Function	No.	Program Module	Missile Class I Min	I Max	II Min	II Max	III Min	III Max
	S1	Basic TCS	X	X	X	X	X	X
		Basic TCS with Estimation			X	X	X	X
Track & Stabilization	S2	Radome Compensation			X	X	X	X
	S3	Head Aiming (Initializing)	X	X	X	X	X	X
	S4	Gyro & Torque Disturbance Compensation					X	X
	S5	Linear 9-State Feedback Control						X
	E5	Lead-Lag Noise Filter	X					
State Estimation	E1	Fixed Gain Filters		X				
	E2	Switched Gain Filters			X			
	E3	Variable Gain Filters (Decoupled Kalman)				X	X	
	E4	Variable Gain Filters (Coupled Kalman)						X
Guidance	G1	Proportional Navigation	X	X	X			
	G2	4-State Law, Range Desensitized				X	X	X
	A1	Basic A/P	X	X	X	X	X	X
	A2	Structural Filters & Fin Mixing			X	X	X	X
	A3	Induced Roll Reduction				X	X	X
	A6	Band-Switched Gains	X	X	X			
Autopilot	A9	Gain Determination				X		
	A10	with Aero Estimates (α)						
	A13	Gain Determination					X	
	A14	with Aero Estimates (Fin Cross Coupling)						
	A15	Gain Determination						X
	A16	with Aero Estimates (Single Panel Model)						
	I1	Attitude Determination				X	X	X
	I2	Velocity Determination				X	X	X
Attitude Reference	I3	Position Determination				X	X	X
	I4	Angle of Attack Determination				X	X	X
	I5	Aero Parameter Estimation				X	X	X
	I6	Mass & Balance Estimation				X	X	X

LEGEND: X - Denotes use of program module

TABLE 59

MODULAR GROWTH REQUIREMENTS

PROGRESSING FROM CLASS I (MIN) TO CLASS III (MAX)

MISSILE CLASS

MAJOR FUNCTION	I MIN	I MAX	II MIN	II MAX	III MIN	III MAX
Mode Control	S1, S3	No Change	Add:Est.ES2	No Change	Add S3	Add S5
Sensor & Sensor *Signal Processing	I6	Change I4 to Co Radar Add: SP1-11, SP-13-15, SP-17, 18, 20, & 21	Change to: SA-PD Radar Add: SP16		Change to: A-PD Radar	
State Estimation	E5	Change to E1	Change to E2	Change to E3	No Change	Change to E4
Guidance	G1	No Change	No Change	Change to G2	No Change	No Change
Autopilot	A1, A6	No change	Add A2	Delete:A6 Add: A5, A9 A10.	Delete:A9,A13 Add: A13, A14	Delete A13, A16 Add: A15, A16
Attitude Reference	None	None	None	Add: I1-6	No Change	No Change
Telemetry	TELE	"TELE" with increasing no. of data points and data rate.				
**Fuzing (TDD)	None	Add: Simple time delay (TD)	Delete: TD Add: F1	No Change	Delete: F1 Add: F2	No Change
Test	ST1-4 IT1-3 SU1&4	No Change	Delete: ST1,4 Add: ST1/1 ST4/1, IT3/1	No Change	Delete: ST1/1 ST4/1, IT3/1 SU4, Add: ST1/2, ST4/2, IT3/2, SU2, 3, 4/1	No Change
***Mode Control	M1, 3, 5, 8, 11, 17	No Change	Delete: M5,8 11, 17. Add: M6, 9, 12, 15, 18	No Change	Delete: M1, 3, 6, 9, 12, 18. Add: M2, 4, 7, 10, 13, 14, 16, 19.	No Change

NOTES:

- * Basic module set with alternate utilities
- ** See Section 4.6 for choice of fuze TDD.
- *** Single computer systems only.

In addition, provisions should be made to install a higher-performance computer to accommodate the increase in throughput while at the same time stipulating that software written for the lower performance missile computer be portable and run on the higher performance machine without major software redesign.

7.2.2 Federated/Distributed Computer Systems

Growth in federated/distributed computer systems could be supported by the assignment of a medium-performance microcomputer to each major missile section, (e.g., seeker, fuze, autopilot), subject to further computer system partitioning considerations (Section 7.3), such that, growth would be confined to physically modular missile sections.

Program sizes for major missile functions are modest, e.g., for Class I/II/III: Autopilot - 116/336/1222 words, Head Control - 84/272/449 words resp., which, coupled with the physical separation from other functions, aids software modularity and documentation. Further, with the exception of radar signal processing, throughput remains below 320 Kops for each major function from Class I (min.) through to Class III (max.) missile systems.

7.3 Avionics and Missile Subsystem Interfaces

While an on-board missile system is virtually self-contained, its connection to the carrier aircraft weapons control system (AWCS) via the umbilical cable before launch and via a radio frequency command link during flight, in the case of Class III missiles, impacts both on the electrical interface and the design of the missile guidance and control system as an extension of the avionics.

Avionics system integration, from both hardware and software aspects, has undergone critical evaluation and development during the past 2 years, through such programs as the Digital Avionics Information System (DAIS), Refs. R.4, R.5, with the resulting specification of a standard, digital, time-division, command/response, multiplex data bus and associated standard terminal modules, (MIL-STD-1553A, 30 April 1975), together with the on-going definition of a standard higher-order programming language (DOD-I, Ref. R.6).

Whereas these developments would be of little significance to existing analog missile guidance and control systems employing a multi-wire discrete interface with the AWCS, (Figure 143), their importance to the design of future digital missiles is particularly noteworthy.

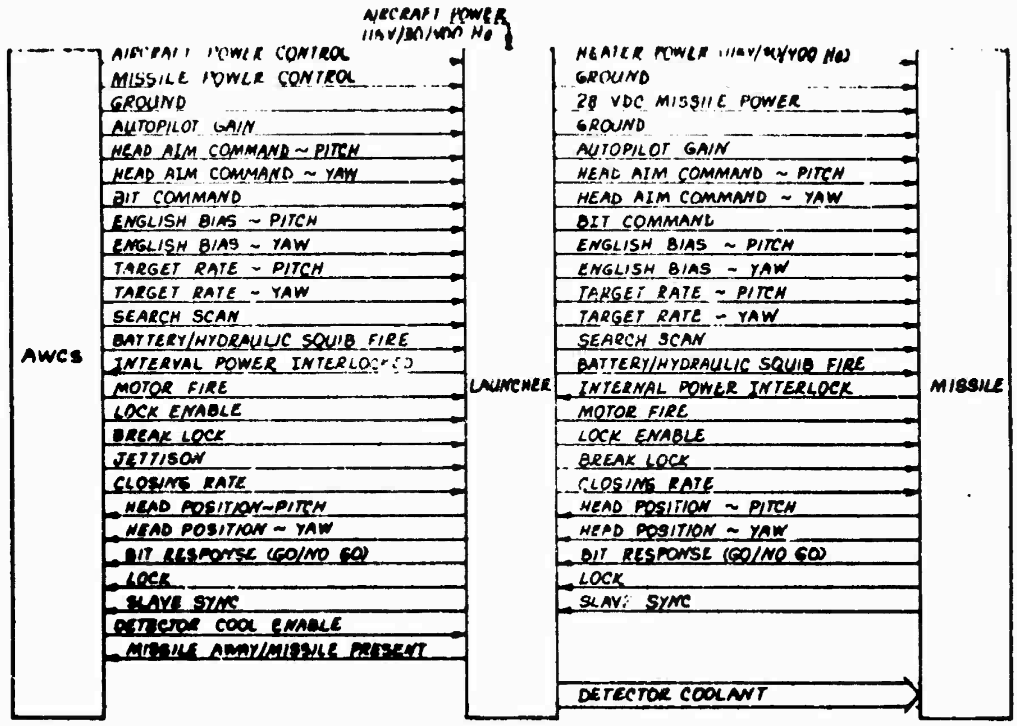


Figure 143 Avionics-Missile Umbilical Interface - Analog Missile System

7.3.1 Avionics Interface

The use of a single-wire, serial-digital (or two-wire redundant) umbilical connection to the AWCS for the transfer of all missile initializing and test commands and data can achieve up to two orders of magnitude reduction in number of umbilical wires. Further the adoption of the MIL-STD-1553 interface would provide an additional cost reduction through the use of a standard terminal unit(s) in the missile. Figure 144 illustrates the compatibility of a digital missile with an avionics system using the 1553 data bus concept. Interface could be provided either to each major function computer, in the case of federated/ distributed computer configurations, or alternatively, to a single central computer. The federated system would afford direct access to each major section of the missile for test and maintenance purposes.

7.3.2 Missile Subsystem Interface

Since digital missile guidance and control systems are, in-effect, pilot-less avionics systems without displays, and are notably more similar in the case of Class III missiles, the adoption of the 1553 interface between missile subsystems together with digital avionics system design practices becomes a serious consideration in missile computer system design. Testing major missile subsystems/sections would be via a common, simple, digital interface resulting in greater standardization of test equipment and a reduction in system life cycle cost.

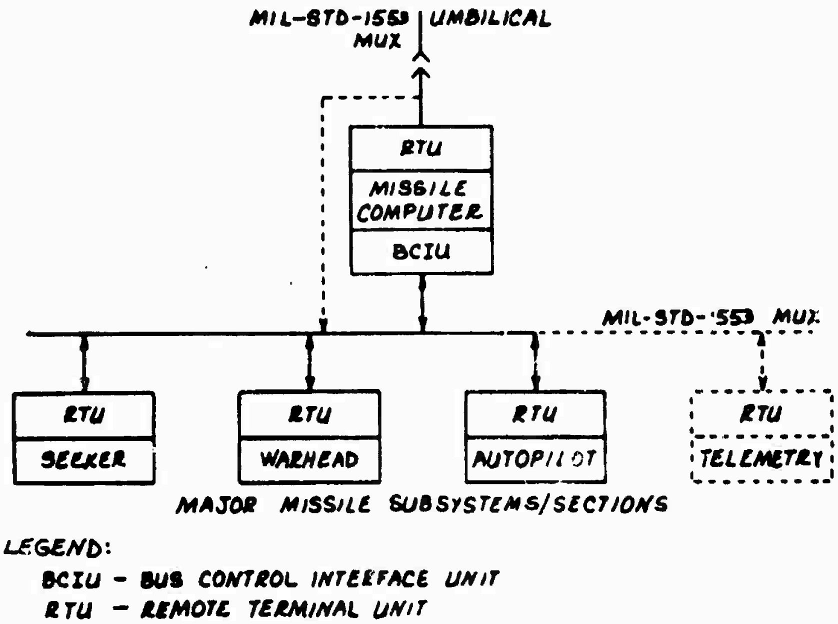


Figure 144 Avionics - Missile MIL-STD-1553 Umbilical Interface - Digital Missile

7.4 Computer Loads, Function Autonomy and Input-Output Traffic

Based on the system timing constraints discussed in subsection 6.1 and the computer loads given by major function and in totum in 6.2 and 6.3, the compatible implementation of federated and single central computer systems is subject to the following throughput and input-output interface considerations.

7.4.1 Federated/Distributed Computer Systems

For federated/distributed computer systems, the most logical separation of primary missile guidance and control functions is into the three semi-autonomous, functional groups listed below:

Group I Steering Command Generation

Group II Seeker Head Stabilization and Control

Group III Missile Stabilization and Control (i.e. Autopilot)

Figure 145 illustrates the above partitioning, observing the system sampling rates and allowable computational delays.

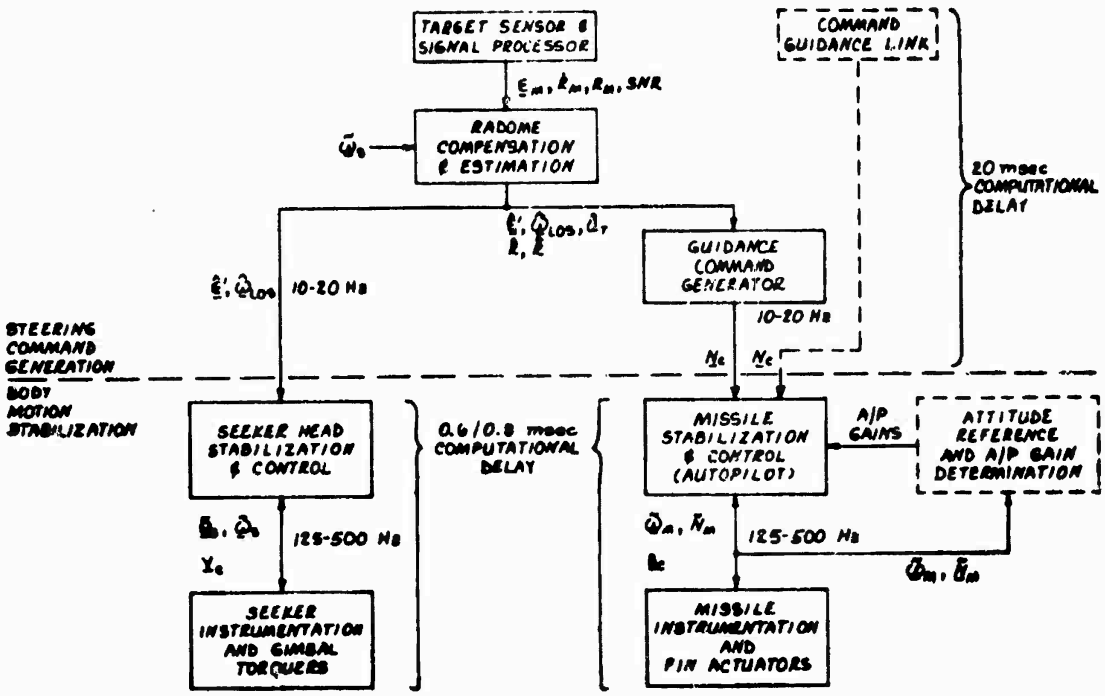


Figure 145 Missile System Function Autonomy

Additional supporting functions with the same degree of autonomy and simple, low-speed, input-output interface are:

1) Attitude Reference
2) Fuzing
3) Telemetry

7.4.1.1 Steering Command Generation

The results of sensor signal processing, radome compensation and estimation, as tandem functions, are used for seeker head stabilization and control, (viz: $\hat{\varepsilon}$ and $\hat{\omega}_{LOS}$), and missile stabilization and control, ($\hat{\varepsilon}, \hat{\omega}_{LOS}$, $\hat{R}$, $\hat{\dot{R}}$ and $\hat{a}_T$). This feature, together with the low update rate of 10-20Hz and small number of parameters involved, provides a practical interface for the partitioning of guidance and control functions in federated computer systems.

Table 60 lists the number of computer operations required for each major function in the steering command loop (SCL) and for clutter acquisition. The execution of all functions in the steering command loop is subject to the 20 msec overall computational time delay constraint as determined in the simulation analyses. However, due to the time required for data collection from the radar receiver, the time available for sensor signal processing, radome compensation, estimation and guidance is reduced. The allotment of time to individual functions is subject to the performance of candidate computer configurations as discussed in the following section.

TABLE 60

COMPUTER OPERATIONS & ALLOWABLE COMPUTE TIME

FOR

STEERING COMMAND LOOP (SCL) & CLUTTER ACQUSITION

FUNCTION	MISSILE CLASS I (1) ACQ.	I SCL TRACK	II (1) ACQ	II SCL TRACK	III (1) ACQ	III SCL TRACK
FFT(2)	57,600(10)	18,240(3)	334,000(50)	30,400(5)	608,000(100)	30,400(5)
PDI	11,520	N/A	57,600	N/A	115,200	N/A
Other Sig. Pro.	1,217	1,333	6,905	1,333	16,175	1,333
Radome Compensation(3)	N/A	N/A	N/A	341	N/A	341
Estimation	N/A	159	N/A	245	N/A	375
Guidance	N/A	[illegible]	N/A	239	N/A	239
Allowable Compute Time (msec)	50.0	20.0	50.0	20.0	50.0	20.0

NOTES:

( ) No. of 64-pt complex FFTs.
(1) 10 dwells.
(2) Includes time-weighting (all missile classes), and corner-turning (multiple range-gating Class II & III only).
(3) In the steering command loop, but not in the time-critical path.

7.4.1.2 Seeker and Missile Stabilization and Control

Since both seeker and missile require stabilization against body motion using dedicated gimbal and body instrumentation respectively, sampled at the higher sampling rates (125-500Hz), and, furthermore, since the stabilization process is highly repetitive, the dedication of a separate computer to each task would be both convenient and practical. Each of these stabilization and control computers would then receive steering commands on a more quasi - static basis, (10-20Hz), asynchronous to the repetitive stabilization task.

Throughput requirements for each of the latter computers are summarized in Table 61.

TABLE 61

WORST-CASE COMPUTER THROUGHPUT REQUIREMENTS FOR SEEKER AND MISSILE STABILIZATION & CONTROL LOOPS

STABILIZATION &	THROUGHPUT (KOPS) & INSTRUCTION MIX MISSILE CLASS I (MAX)	II (MAX)	III (MAX)
Seeker Head	82.0 (13/15/72)	109.4 (13/15/72)	319.6 (17/12/71)
Autopilot	84.5 (17/9/74)	262.8 (15/13/72)	262.8 (15/13/72)

7.4.1.3 Supporting Functions

Of the remaining supporting functions, viz: attitude reference, autopilot gain determination, fuzing and telemetry, only the latter two are realistic candidates for federation, since attitude reference and autopilot gain determination are closely associated with the autopilot and its data inputs.

Attitude reference utilizes many of the autopilot data inputs (i.e. ω_M, N_M, h), and in turn, autopilot gains are determined by table-look-up as a function of Mach number and angle of attack computed by the attitude reference algorithms. Hence these two supporting functions would be best co-located with the autopilot.

Warhead fuzing time delay algorithms require few data transfers from the missile estimation algorithms ($\hat{\epsilon}_M$, θ_G, $\dot{R}_M$, t_{go}) and at a low update rate (10-20 Hz). Since the fuzing system is virtually self-contained, complete with its own target sensor, a separate processor for the TDD is practical and the low throughput required (23.2 Kops max) makes this function an ideal candidate for a simple N-MOS microcomputer.

Telemetry is essentially a data gathering and formatting operation using both guidance and control data and general missile data. The former could be obtained by direct-memory-access (DMA) to data stored in the steering and body motion stabilization and control processor memories, while the latter would be acquired direct from the respective sensors, with A-D conversion as required. Table 62 lists the throughput requirements for fuzing and telemetry by missile class.

TABLE 62

WORST-CASE COMPUTER THROUGHPUT

REQUIREMENTS FOR FUZING & TELEMETRY

SUPPORTING/ SATELLITE FUNCTION	THROUGHPUT (Kops) & INSTRUCTION MIX MISSILE CLASS I	II	III
Fuzing	0.9 (16/5/79)	23.2 (15/21/64)	20.7 (16/24/60)
Telemetry	16.3 (0/0/100)	32.5 (0/0/100)	32.5 (0/0/100)

7.4.2 Single Computer Systems

The execution of all missile guidance and control functions in a single, conventional, general-purpose computer is currently impractical due to the high throughput rates required for radar digital signal processing. Hence a "minimum federated computer system" is necessary where some, if not all, signal processing functions are executed in a high-speed pre-processor(s). Figure 146 illustrates the necessary evacuation of individual signal processing tasks from the single GP computer when progressing from a Class I through to a Class III missile system. In effect, more and more of the radar signal processing functions and, indirectly, the steering command generation loop, are pushed out of the GP computer, in order that the machine can handle the remaining body motion and other supporting functions, (e.g. fuzing, telemetry etc.).

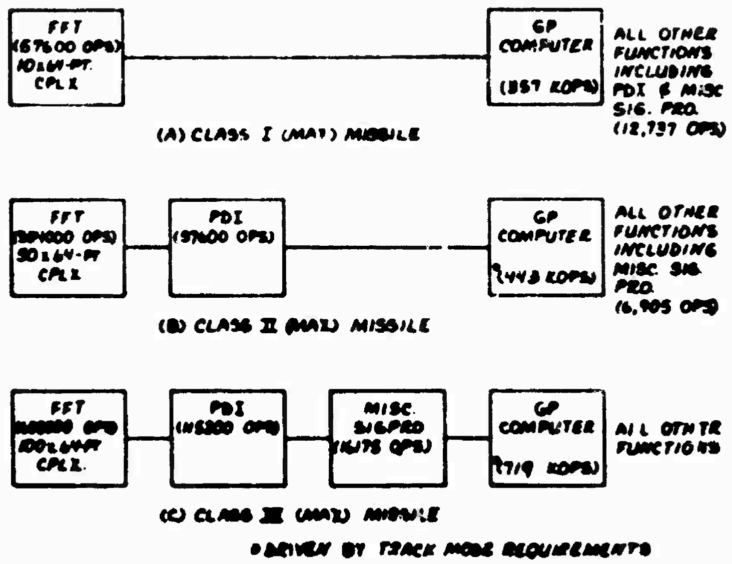


Figure 146 Minimum Federated Computer System Configurations (Single GP Computer).

The following sections address processors and machine architectures capable of supporting the computer loads, and hence the practical division of the total computing time budget for each missile class as shown in Table 60.

7.5 Computer Loads vs Available/Proven Computers

As indicated in the previous subsection, the throughput capabilities of existing and proven computer architectures have a direct bearing on the choice of computer system configuration to support any given worst-case computer load.

In this subsection, the performance characteristics of existing MIL Spec. mini and micro general-purpose computers are used to determine their effectiveness in executing the functions defined for fully federated computer systems and "minimum - federated systems" using one GP computer.

Since, as stated earlier, a single central computer system is not practical due to the high throughput requirements imposed by radar signal processing, the resulting "minimum - federated computer system" will be considered first followed by a totally federated system.

7.5.1 Single GP Computer Systems, (Minimum Federated System)

Tables 63 through 65 show the performance capabilities of three, currently available, MIL-Spec., missile minicomputers, pitted against the worst-case computer throughput loads for radar Clutter Acquisition and missile Terminal Modes respectively, for each of the three generic missile classes, excluding FFT processing in all cases. It can be seen from the results and conclusions indicated that a relatively high-speed GP computer (VELARC) could accommodate the Class I and II throughput requirements, excluding FFT and PDI signal processing, while providing radar mode control. However, in the case of the Class III system, all radar functions must be removed from the single GP computer to avoid saturation. Figure 146 illustrated this throughput migration in block diagram form.

The accommodation of the excess radar signal processing functions, in conjunction with the remaining steering loop algorithms, using compatible processor architectures, is therefore discussed in the next subsection which addresses federated computer systems design.

TABLE 63

SINGLE COMPUTER SYSTEMS

REQUIRED THROUGHPUT VS AVAILABLE MINICOMPUTER PERFORMANCE

CLASS I (MAX) MISSILE

		COMPUTER					
		VELARC (Raytheon DPC)		HARPOON (IBM 4PiSP-0B)		BRAZO (CDC 469)	
OPERATION TYPE	MIX	Execution Time (µsec)	Total (µsec)	Execution Time (µsec)	Total (µsec)	Execution Time (µsec)	Total (µsec)
Short	96/92	0.75	72/69	3.5	336/322	2.4	230/221
Long	4/8	*6.0	24/48	*12.0	48/96	*12.0	48/96
Totals:	100/100	-	96/117	-	384/418	-	278/317
Computer Kops:		1041/855		260/239		360/315	
Required Kops:		357/191		357/191		357/191	
Conclusion:		Overkill		**Marginal		Marginal	

NOTES:

(/) (Acquisition/Terminal) Modes respectively without FFT, but including PDI for Acquisition Mode (see Table 51).

*Multiply time weighted to accommodate small proportion of divide operations.

** That is without FFT & PDI

TABLE 64

SINGLE COMPUTER SYSTEMS

REQUIRED THROUGHPUT VS AVAILABLE MINICOMPUTER PERFORMANCE

CLASS II (MAX) MISSILE

		COMPUTER					
		VELARC (Raytheon DPC)		HARPOON (IBM 4PiSP-0B)		BRAZO (CDC 469)	
OPERATION TYPE	MIX	Execution Time (μsec)	Total (μsec)	Execution Time (μsec)	Total (μsec)	Execution Time (μsec)	Total (μsec)
Short	96/89	0.75	72/67	3.5	336/311	2.4	230/213
Long	4/11	6.0	24/66	12.0	48/132	12.0	48/132
Totals:	100/100	-	96/133	-	384/443	-	276/345
**Computer Kops:		1041/752		260/225		359/290	
**Required Kops:		290/443		290/443		290/443	
Conclusion:		Overkill		Not Acceptable		Not acceptable	

NOTES: (/) (Acquisition/Terminal) Modes respectively
* Multiply time weighted to accommodate small proportion of divide operations
** Without FFT & PDI. (See Tables 52 & 53).

TABLE 65

SINGLE COMPUTER SYSTEMS

REQUIRED THROUGHPUT VS AVAILABLE MINICOMPUTER PERFORMANCE

CLASS III (MAX) MISSILE

		COMPUTER					
		VELARC (Raytheon DPC)		HARPOON (IBM 4PISP-OB)		BRAZO (CDC 469)	
TYPE	OPERATION MIX	Execution Time (μsec)	Total (μsec)	Execution Time (μsec)	Total (μsec)	Execution Time (μsec)	Total (μsec)
Short	93/88	0.75	69/66	3.5	325/308	2.4	223/211
Long	7/12	*6.0	42/72	*12.0	84/144	*12.0	84/144
Totals:	100/100	-	111/138	-	409/452	-	307/355
**Computer Kops:		901/724		244/221		325/281	
**Required Kops:		946/805		946/805		946/805	
Conclusion:		Not acceptable		Not acceptable		Not acceptable	

NOTES

(/) (Acquisition/Terminal) Mode respectively
* Multiply time weighted to accomodate small proporation of divide operations
** Without FFT & PDI. (See Tables 52 and 53).

7.5.2 Federated/Distributed Computer Systems

Based on the partitioning rationale discussed in subsection 7.4.1, a feasible federated missile computer system comprises the following dedicated processors:

1. Steering Command Generation
2. Seeker Head Stabilization and Control
3. Missile Stabilization and Control (Autopilot)
4. Fuzing Time Delay
5. Telemetry

Steering command generation entails the execution of radar signal processing, radome compensation, estimation and guidance functions within a maximum allowable computational delay consistent with acceptable miss distance, (refer to subsection 5.3.4). Since it has already been shown that a conventional GP missile computer lacks the throughput capability to execute all radar signal processing functions within an acceptable time delay, an analysis of the FFT operation is given in the following paragraphs followed by an overview of compatible machine architectures and execution speeds.

7.5.2.1 Fast Fourier Transform (FFT) Processing

The Cooley-Tukey FFT algorithm (Ref. R.7) is executed by repeating a complex 2-point transform iteratively $N/2 \operatorname{Log}_2 N$ times per range bin, thereby constituting a major processing load for digital radar signal processing. A description of the FFT algorithm together with programming examples and hardware implementations is given in the following paragraphs.

Basic Cooley-Tukey FFT Operation - The basic operation (B) and common denominator of the Cooley-Tukey algorithm consists of one addition, one subtraction and one multiplication involving two data points and a stored multiplier - all of which can be either real or complex quantities. Figure 147 is a flow diagram illustrating the basic operation with complex data points (a+jb and c+jd) and multiplier (x+jy). The operation yields two new complex values:

$$(a+c) + j(b+d) \quad (1)$$

and

$$[X(a-c) - Y(b-d)] + j\,[X(b-d) + Y(a-c)] \quad (2)$$

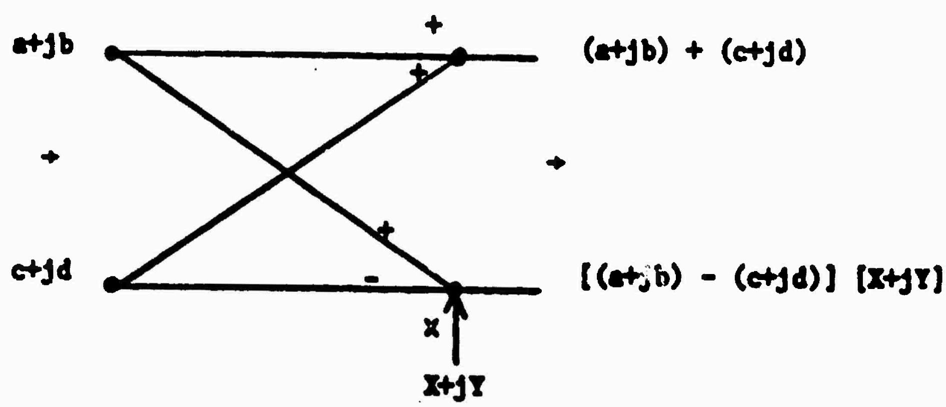


Figure 147 Basic FFT Operation - Flow Diagram

Thus, with complex data points, the basic operation requires the following computer arithmetic operations:

Adds:	3
Subtracts:	3
Multiplies:	4

For N data points (N being any power of 2) the basic operation is repeated $N/2 \log_2 N$ times total using original data pairs and resulting pairs and following an iterative process in accordance wtih the complete FFT algorithm (Figure 148).

Computer Programs - Programs to implement the basic FFT operation with complex data points on both single and three-address machines are given in Table 66.

It is assumed that all input data points are fully buffered requiring 2N memory locations, although the basic operation involves only 4 of these at a time plus an additional 4 for intermediate results (scratch-pad memory). With this arrangement, memory requirements for the basic operation, excluding input-output programs and indexing or data sorting programs are as follows:

Use	Locations	Type
Data Points	2N+4	RAM
Multiplier constants	N	ROM
Basic operation sub-program	K (10 or 30)	ROM

Complete FFT Algorithm - Figure 148 is a flow diagram for a 16-point FFT showing $\log_2 N$ iterations (4) of N/2 basic operations (8), i.e. 32 basic operations total.

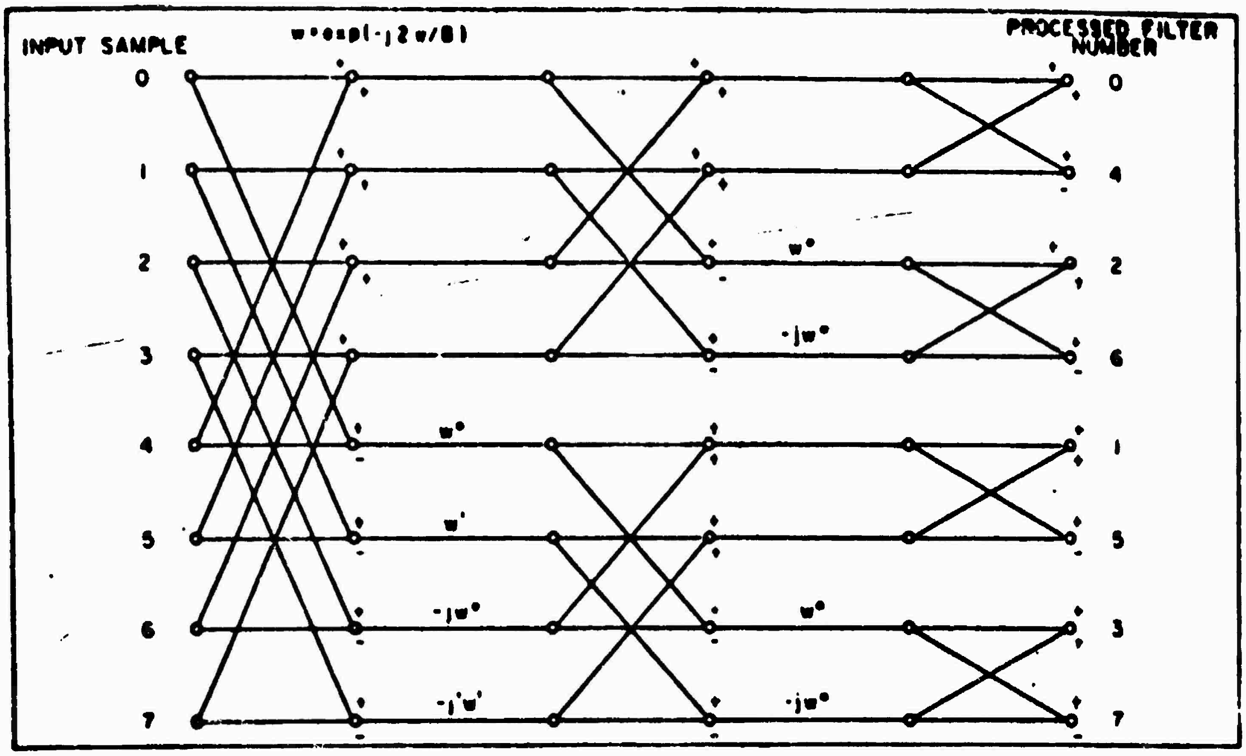


Figure 148 FFT Algorithm - Flow Diagram, N = 8

To use the same sub-program (Table 66) for $N/2 \log_2 N$ basic operations requires the adoption of one of the following three data addressing schemes:

1. Direct Addressing (straight-line programming - no indexing)
2. Data sorting/re-ordering
3. Address modification, through index registers

Direct Addressing/Straight-Line Programming - This would require the same program to be re-written $N/2 \log_2 N$ times with new addresses each time to directly address the required data points and multiplier constants. Table 67 shows the resulting program sizes for values of N between 16 and 1024, using direct addressing compared to indexing with a common subroutine. The inefficiency of the former scheme is self-evident but could be tolerable for small values of N with currently available, masked-programmed, semiconductor ROMs at 4, 8 and 16 Kbits per LSI chip.

Data sorting/Re-ordering - An alternative to the direct addressing of data points in each instruction is to re-order the data such that the sub-program for the basic operation (Table 66) addresses the same locations in a 10-word, scratch-pad memory, but obtains the required pair of data points and multiplier constant as a result of an earlier sorting process.

A significant speed advantage could be obtained using a second (micro) computer as a pre-processor to handle the data sorting/re-ordering task and overlap the 2-point FFT process performed in a follow-on computer.

TABLE 66

BASIC FFT OPERATION, SINGLE & 3-ADDRESS

COMPUTER PROGRAMS

SINGLE-ADDRESS PROGRAM		3-ADDRESS PROGRAM	
Instruction	Operation	Instruction	Operation
LDA	a → A	SUB	a−c → (2N+1)
SUB	a − c → A	SUB	b−d → (2N+2)
STA	A → (2N+1)	ADD	a+c(1)
LDA	b → A	ADD	b+c →(N+1)
SUB	b−d → A	MPY	X(a−c) → (2N+3)
STA	A →(2N+2)	MPY	Y(a−c) → (2N+1)
LDA	a → A	MPY	X(b−d) → (2N+4)
ADD	a+c → A	MPY	Y(b−d) → (2N+2)
STA	A →(1)	SUB	X(a−c)−Y(b−d) → (N/2+1)
LDA	b → A	ADD	X(b−d)+Y(a−c) → (3N/2+1)
ADD	b+c→A	Total	
STA	A → (N+1)	Instructions: 10	
LDA	a−c→ A		
MPY	X(a−c) → A,B		
STA	A→ (2N+3)		
LDA	(a−c) → A		
MPY	Y(a−c) → A,B		
STA	A→ (2N+1)		
LDA	b−d→ A		
MPY	X(b−d) → A,B		
STA	A→ (2N+4)		
LDA	(b−d) → A		
MPY	Y(b−d) → A,B		
STA	A → (2N+2)		
LDA	X(a−c) → A		
SUB	X(a−c)−Y(b−d) → A		
STA	A → (N/2+1)		
LDA	X(b−d) → A		
ADD	X(b−d)+Y(a−c) → A		
STA	A → (3N/2+1)		

Total
Instructions: 3[illegible]

TABLE 67

FFT ALGORITHM, PROGRAM SIZE, DIRECT ADDRESSING/

STRAIGHT-LINE PROGRAMMING VS INDEXING/COMMON SUBROUTINE

DATA POINTS (N)	PROGRAM SIZE Direct Addressing Single	3-Address	Indexing Single	3-Address
16	960	320	88	48
32	2,400	800	120	80
64	5,760	1,920	184	144
128	13,440	4,480	312	272
256	30,720	10,240	568	528
512	69,120	23,040	1,080	1,040
1,024	153,600	51,200	2,104	2,064

Address Modification (through index registers) - To utilize a single, common 2-point transform sub-program for the entire FFT algorithm requires modification of subroutine operand addresses and the usual method of achieving this in a GP computer is to utilize hardware index registers whose contents can be added to the displacement addresses contained in the instructions of the common sub-program.

Input-Output - For high-speed and a minimum of buffering and programmed operations the direct memory access (DMA) I/O is the most effective method for block transfers.

Outputting of transformed data in re-ordered form could be achieved in a similar manner to the inputting operation by bit-reversing the counters for memory block addressing thereby automatically re-ordering the data to the original sequence.

Figure 149 is a first level flow diagram of the FFT program for a single computer.

Computer Configurations

Table 68 lists the four major computer configurations for FFT processing and their relative execution speeds as a function of the number of 2-point transform arithmetic units employed (Ref. R.8). Of these four machine architectures, the single sequential configuration, as a minimum hardware version, is more practical for on-board missile applications. Further, the 2-point transform arithmetic "unit" could be considered as either a full hardware implementation or a combination of more simple hardware and supporting software/firmware routines. With this latter concept in mind, both conventional GP computers and their FFT optimised counterparts were reviewed. Figure 150 illustrates the candidate computer configurations considered with their respective performance capabilities summarized in Table 69, using the basic 2-point and 64-point complex transforms as common benchmarks.

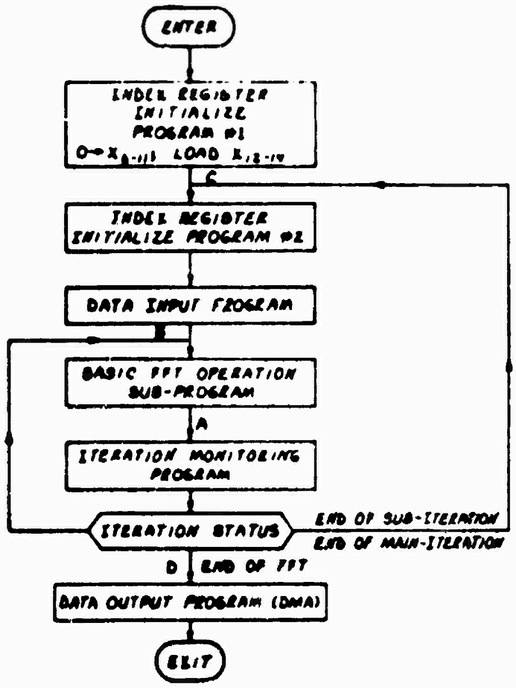


Figure 149 FFT Execution, Main Flow

TABLE 68

FFT PROCESSOR CONFIGURATIONS VS EXECUTION TIME

Configuration	No. of 2-pt Transform Units	FFT Execution Time
Single sequential	1	$N/2 \log_2 N \times b$
Pipeline	$\text{Log}_2 N$	$N/2 \times B$
Parallel iterative	$N/2$	$\text{Log}_2 N \times B$
Array	$N/2 \log_2 N$	b

Figure 153 FFT Processor Configurations

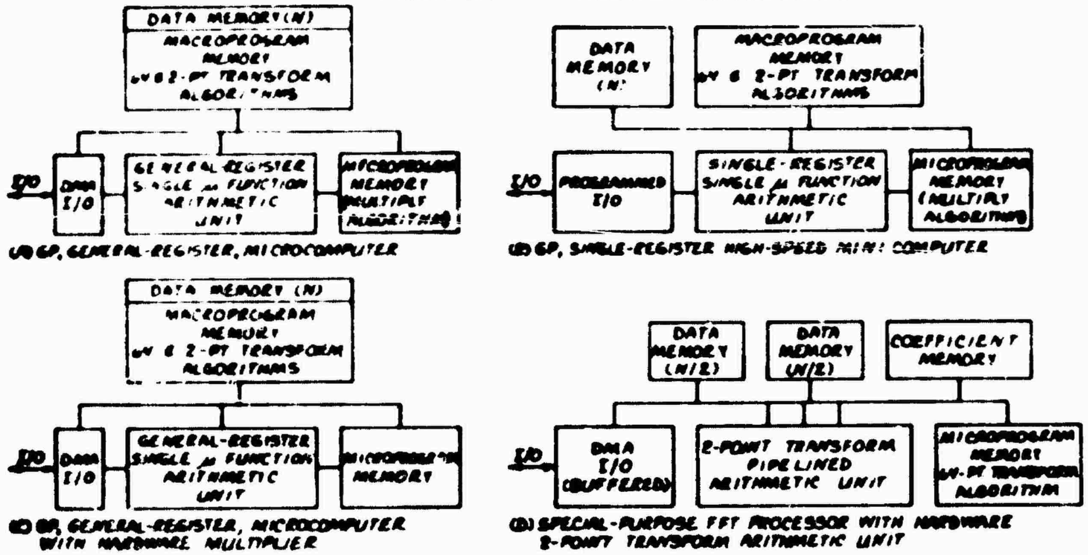


TABLE 64

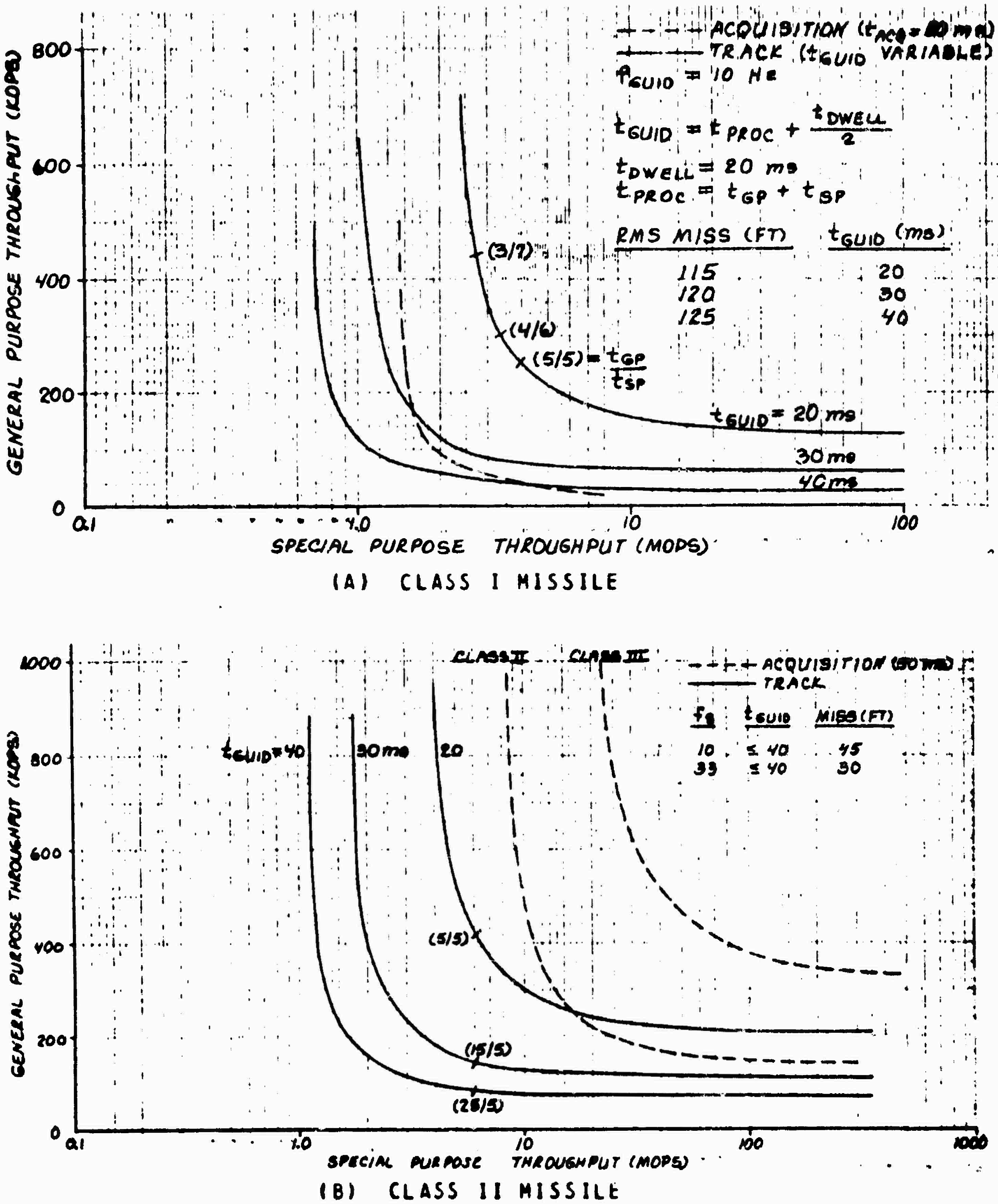
FFT PROCESSOR CONFIGURATIONS VS PERFORMANCE

Computer Type	Configuration	Execution Times: Basic 2-Pt. Complex Transform	Execution Times: 64-Pt. Complex Transform	Comments
Special-Purpose Processors	Parallel FFT Processor (Raytheon GPSP)	133ns	19.2 μs	Two, 2-point cplx. transform arith. units All bipolar s/c circuitry incl. memories.
	Single FFT Processor (Raytheon GPSP)	200ns	38.4 μs	One 2-point cplx. transform arith. unit All bipolar s/c circuitry inc. memories.
Standard General Purpose Computers	Single Microcomputer (AMD 2900-series)	14 μs	2.7ms	General-register with hardware multiplier. Straight-line programmed. All bipolar s/c circuitry incl memories.
	Single Minicomputer (Eclipse)	42 μs	6 ms	Single-register firmware multiplier. Straight-line programmed. All bipolar s/c circuitry incl. memories.
	Single Microcomputer (Intel 3000-series)	87.2 μs	16.7 ms	General-register, firmware multiplier. Straight-line programmed. All bipolar s/c circuitry incl. memories

7.5.2.2 Steering Command Generation

With a knowledge of what is practical in terms of computer types and throughputs versus the required processing load, a timing analysis was performed to determine the most effective load distribution between conventional GP and FFT/PDI-optimised computer architectures.

Figures 151 (A) and (B) are curves showing GP computer throughput versus FFT and PDI - optimised processor throughputs for the radar Clutter Acquisition and missile Track Modes, for Class I and II & III missiles respectively. The optimum design points chosen for each class are given in Table 70.



(A) CLASS I MISSILE

(B) CLASS II MISSILE

Figure 151 FFT/PDI-Optimized Processor vs Conventional GP Processor Throughput Tradeoff

TABLE 70

GP COMPUTER THROUGHPUTS & COMPUTE TIMES

FOR

STEERING COMMAND LOOP (SCL) & CLUTTER ACQUISITION TIME CRITICAL

PATHS

	MISSILE CLASS					
	I		II		III	
FUNCTION	*ACQ	(SCL) TRACK	*ACQ	(SCL) TRACK	*ACQ	(SCL) TRACK
FFT		8.1		1.5		0.192
PDI	3.2		1.78		0.456	
Other Sig Pro	1.8 (87.9)		3.22 (278.7)		4.544 (462.7)	
Radome Compensation	N/A	N/A	N/A	N/C	N/A	N/C
Estimation	N/A	11.9	N/A	18.5	N/A	19.808
Guidance	N/A	(167.3)	N/A	(127.7)	N/A	(127.8)
Total Compute Time (msec)	5.0	20.0	5.0	20.0	5.0	20.0

NOTES:

* One 5 msec dwell
N/A Not applicable
N/C In the steering command loop, but not in the time-critical path.
( ) GP Computer throughput requirement (Kops), including additional 30% overhead.

Hence, of the several different types of computer architecture and associated FFT processing performance reviewed in Table 69, two types will provide the processing speeds required to meet the Class I, II and III radar signal processing loads, namely:

1) Single microcomputer (bipolar) with hardware multiply
2) Single microcomputer (bipolar) with 2-point complex transform arithmetic module

Due to the high processing speed of the above computers and the short processing time allowed, the remaining processing tasks associated with signal processing, estimation and guidance can be accommodated by a conventional gp computer with 100, 300 and 500 Kops throughput capability respectively.

7.5.2.3 Body Motion Stability and Control

Table 71 applies the seeker head and missile body motion stability and control throughput requirements to an available MIL-Spec., bipolar, microcomputer set, (Intel 3000-series), configured as a 16-bit processor. While there is overkill in the Class I case, Class II and Class III are well matched. This shows that an N-MOS processor would be a better fit for Class I applications.

TABLE 71

FEDERATED/DISTRIBUTED COMPUTER SYSTEMS

REQUIRED THROUGHPUT VS AVAILABLE BIPOLAR MICROCOMPUTER PERFORMANCE

FOR BODY MOTION STABLIZATION & CONTROL LOOPS

BODY MOTION LOOP	OPERATIONS		MICROCOMPUTER EXECUTION Time (µsec)	MISSILE CLASS I Opn. Mix	I Execution Time (µsec)	II Opn Mix	II Execution Time (µsec)	III Opn Mix	III Execution Time (µsec)
Seeker Head Stabilization and Control	Short		1.2	85	102.0	85	102.0	88	105.6
	Long		14.0	15	210.0	15	210.0	12	168.0
		Compute Time (µsec)			312.0		312.0		273.6
	Totals	Kops:			320.5		320.5		365.5
		Reqd. Kops:			82.0		109.4		319.6
Missile Stabilization and Control (Autopilot)	Short		1.2	91	109.2	87	104.4	87	104.4
	Long		[illegible]	[illegible]	126.0	13	182.0	13	182.0
		Compute Time (µsec)			235.2		286.4		286.4
	Totals	Kops:			425.2		349.1		349.1
		Reqd., Kops:			84.5		262.8		262.8

7.5.2.4 Supporting/Satellite Functions (Fuzing and Telemetry

The remaining missile functions of fuzing and telemetry are similarly sized against an available, MIL Spec., microcomputer set (AMD 9080A-2DM) in Table 72. From the results obtained, based on estimated computer loads, a N-MOS microcomputer with a software multiply routine matches the fuzing throughput requirements for Class II and III missiles and has overkill for Class I and all telemetry requirements.

TABLE 72

FEDERATED/DISTRIBUTED COMPUTER SYSTEMS

REQUIRED THROUGHPUT VS AVAILABLE MICROCOMPUTER PERFORMANCE

FOR FUZING & TELEMETRY

FUNCTION	Type	OPERATIONS Execution Time (μ sec)	MISSILE CLASS I Opn. Mix	I Execution Time (μ sec)	II Opn Mix	II Execution Time (μ sec)	III Opn Mix	III Execution Time (μ sec)
	Short	2.275	95	216.1	79	179.7	76	172.9
Fuzing	Long	172.5	5	862.5	21	3622.5	24	4140
		Compute Time (μ sec)		1078.6		3802.2		4312.9
	Totals	Kops	92.7		26.3		23.18	
		Reqd. Kops	0.9		23.2		20.7	
	Short	2.275	100	227.5	100	227.5	100	227.5
	Long	172.5	0		0		0	
Telemetry		Compute Time (μ sec)		227.5		227.5		227.5
	Totals	Kops		439.5		439.5		439.5
		Reqd Kops		16.3		32.5		32.5

7.5.2.5 Recommended Federated/Distributed Computer System

As a result of the previous computer system analyses, the recommended federated/distributed computer system for missile guidance and control is of the form shown in Figure 152.

Six microcomputers (μC_1-μC_6), of varying performance capabilities are matched to the respective, semi-autonomous, missile functions which in themselves follow the physical partitioning of major missile functions for design, manufacture, test and maintenance.

The body motion stabilization and control processors (μC_3 and μC_5) are colocated with their respective sensors and actuators and execute the control loop functions in an uninterrupted cyclic manner.

A 2-wire, serial digital multiplex bus, as defined in MIL-STD-1553A, forms the interface between all microcomputers and the carrier aircraft AWCS computer.

Before launch the AWCS computer controls the missile microcomputers via a BCIU to subordinate microcomputer RTUs. (The BCIU of microcomputer no.1 (μC_1) functions as an RTU during the prelaunch mode, Ref. R.9 para 3.1).

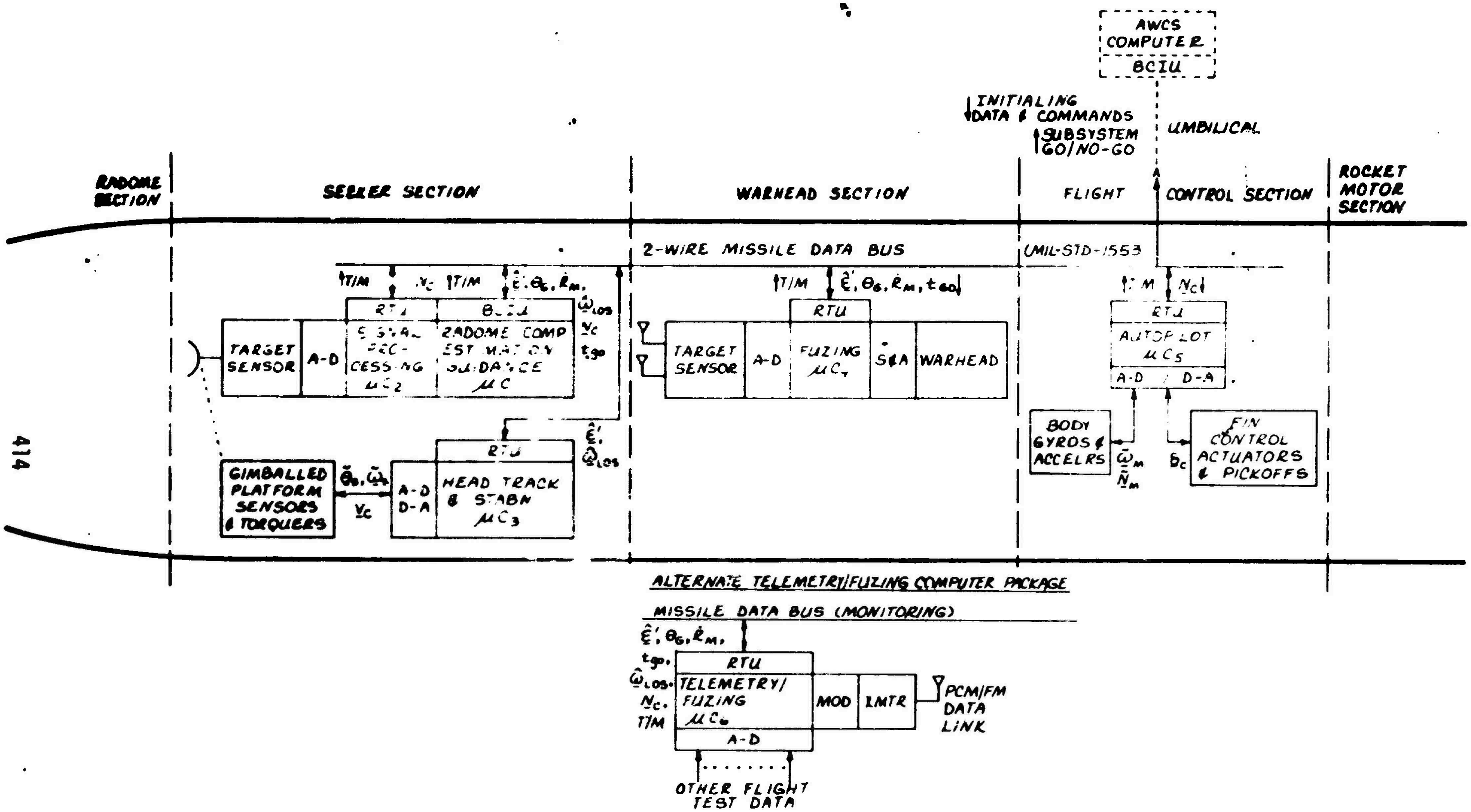


Figure 152 Recommended Federated/Distributed Microcomputer System

After launch, the body motion stabilization and control computers (μC_3 and μC_5) receive appropriate steering and g commands from microcomputer no. 1 (μC_1) asynchronously at the low-frequency, 10 to 40 Hz, update- rate. Input of these parameters is by direct-memory-access to μC_3 and μC_5 memories after serial to parallel conversion by their respective SDIO modules. The fuzing computer (μC_4) receives its input data in a similar manner.

Steering command generation is accomplished with a high-speed microcomputer (μC_2) performing the FFT/PDI functions under the control of a medium-speed microcomputer (μC_1) which also executes radar mode control, post-processing, radome compensation, estimation and guidance functions, all within the maximum allowable computational delay for steering command generation.

Pre-processed radar data is transferred from μC_2 to μC_1 across a parallel DMA I/O interface at the end of each dwell or series of dwells, as in the case of the clutter acquisition mode.

For missile flight tests, the warhead section and associated microcomputer (μC_4), is replaced by a telemetry package and dedicated microcomputer (μC_6). The telemetry computer and its associated RTU operate as a bus monitor for digital data gathering, with the additional analog test data being input via an analog multiplexer/A-D converter (ADAC) I/O module.

Fuzing and telemetry are handled by separate CPU-on-a-chip type medium/low-speed microcomputers.

Both the launch aircraft and test equipment have direct access to each microcomputer via the common bus, (e.g. MIL-STD-1553), enabling fault isolation to the major subassembly level.

7.6 Modular Computer Definition

The available mini and microcomputers evaluated in the context of missile guidance and control system requirements, while supporting the respective computer loads, lack common modularity features, and, even more important they lack a common programming language. These two deficiencies constitute major drawbacks for low-cost, modular growth, design flexibility and simple logistics in missile systems.

The solution to the first problem identified above, lies in the definition of standard, major/macro-function computer modules each with a standard interface to a common interconnecting bus. Such minimal standardization would provide the means of changing the performance of a given computer configuration by interchanging memory modules with different cycle times, central processing units (CPUs) with different computing features, and input-output channel types to suit the specific I/O situation, in order to achieve the desired performance and programming features for a specific missile computing task. In other words, to achieve a best-fit of computer hardware configuration at lowest-cost and without restrictions on future growth to accommodate changing technology both in performance and circuit packing densities.

Missile guidance and control computing requirements have been shown to demand a wide range of computing throughput rates together with memory capacities, therefore a corresponding range of related computer configurations is needed with the modularity and common interface characteristics previously identified.

A review of state-of-the-art, special-purpose, digital signal processors and conventional general-purpose computer architectures (see Figure 150 and Table 69) has revealed significant commonalities in both major functional components (macro-function units) and organization. The only major difference evident in processors performing high-speed fast Fourier transforms (FFTs) versus regular Von Neumann type GP computer designs is the use of a special arithmetic module, optimized for the rapid execution of the basic 2-point complex transform, in place of the general-purpose arithmetic and logic unit. In addition to signal processing commonalities, general-purpose computers have reached a level of maturity over the past 25 years resulting in certain established design features being commonly employed to improve performance and aid programming. The recent advent of large-scale-integrated (LSI) circuit microcomputers as high-volume computer component sets has further complemented the standardization trend.

In the light of the above preliminary studies and observations, a family of macro-modular microcomputers as shown in Figure 153, offers an effective means of meeting the processing requirements of all three classes of A-A missile, with the option of configuring either single or federated computer systems according to the specific constraints of a given missile.

Table 73 lists ten major types of macro-function modules required to support the range of microcomputer configurations shown in Figure 153.

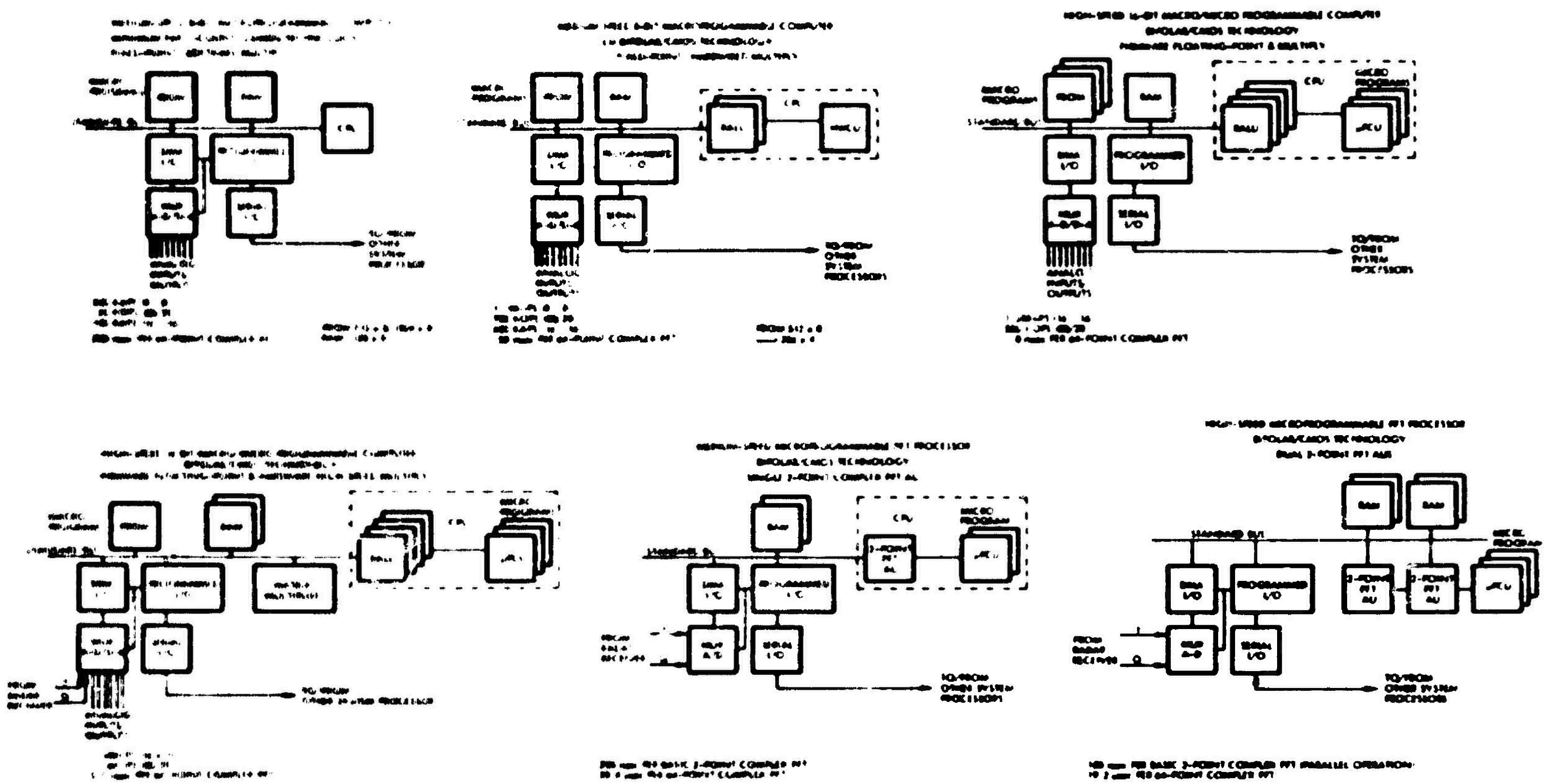


Figure 153 Macro-Modular Microcomputer Family for On-Board Missile Signal Processing and Control

TABLE 13

MACRO-MODULAR MICROCOMPUTER

LSI MACRO-FUNCTION MODULES

MODULE GROUP	LSIC MODULE	NAME	CHARACTERISTICS	MODULAR MICROCOMPUTER APPLICATION
1	RAM-1	Random Access Memory	High-speed @ 100 nsec access.	High-speed data storage, single computer systems. Signal processing.
	RAM-2	Random Access Memory	Medium-speed, @ 500 nsec access	Medium-speed data storage federated computer systems.
2	PROM-1	Electrically Programmable Read-Only Memory	High-speed, field programmable. 100 nsec access.	High-speed macro/micro program storage, single computer systems.
	PROM-2	Electrically Programmable Read-Only Memory	Medium-Speed, field programmable/re-programmable 500 nsec access/	Medium-speed macro program storage federated computer systems.
3	CPU-1	Central Processor Unit	Medium-speed, 8-bit byte, 2 µsec register-register add, fixed instruction set.	Medium-speed federated computer systems
4	RALU-1	Register Arithmetic and Logic Unit Slice	High-speed, 4-bit slice, general-register, 130 nsec cycle	High-speed, single computer systems, word lengths from [illegible]
5	FFT-1	2-Point Complex FFT Arithmetic Unit	High-speed, 8x8, pipeline, 250 nsec/operation.	High-speed, dedicated FFT processors.
6	HWCU-1	Hardwired Control Unit	High-speed, fixed instruction set.	High-speed, low-parts count, central, federated, or FFT computer systems
7	µPCU	Microprogrammed Control Unit	Medium-speed, 512 microinstruction addressing.	Medium-high speed single computer and FFT processor control
8	HWMP-1	Hardware Multiplier	High-speed, parallel multiplier. 200 nsec 8x8 bit.	High-speed GP computers performing FFT functions. Multiply speed improvement for other configs.
9	ADAC-1	Analog-Digital, Digital-Analog Converter	Medium-speed, 12-bit A-D and D-A converter with multiplexed inputs and outputs, 0-100 KHz	Multiplexed, low frequency, guidance and control analog input-output interface
	PDIO-1	Parallel Digital (Programmed) Input-Output Channel	Low-Speed data/command transfers under program control	Initializing DMAIC and interfacing with SDIO
	SDIO-1	Serial Digital Input-Output Channel	Medium-speed (up to 1Mbps) word serial, bit-serial data/command transfers.	Inter-subsystem/computer, modem and avionics interface
	DMAIO	Direct Memory Access Input-Output Channel	High-speed data transfers to/from main memory at memory cycle rate.	High-speed signal processor s/c and multiplexed stability loop I/O transfers.
10	PLA	Programmable Logic Array	Combinational Logic networks	Dedicated controllers and decoders e.g. missile up-link

7.7 Software

In the past, the accent has been on the conservation of memory space in avionics applications due to size, weight and power limitations and the use of magnetic-core or plated-wire memory systems. This in turn, emphasized the need for "tight" code or the highly efficient use of program memory in avionics computers. The compatible programming language for this design goal is symbolic assembly language since it achieves one-to-one correspondence with the final machine/object code.

Assembly languages, while efficient in the use of memory space, have the following drawbacks:

1. Peculiar to one computer.
2. Highly flexible in terms of programmer-peculiar routines
3. Difficult to read and spot errors.
4. Difficult to re-understand and modify - even by the original programmer.
5. Difficult to impose and sustain modular and structured programming techniques.
6. All software generated peculiar to a specific computer - non portable.
7. As a result of the above deficiencies - costly to verify and maintain.

Due to increasing labor costs and diminishing computer memory costs, the latter as a result of large-scale-integrated (LSI) semiconductor memory modules, the accent has shifted from tight-code for avionics computers to the following criteria:

1. Common high-order programming language

2. Structured design

3. Modularity

7.7.1 Software Cost

Total software cost for a fully commissioned computer system is measured in terms of average cost per instruction which includes: initial design; coding; verification of coded programs and the subsequent updating/maintenance necessary to meet the system performance specification.

A parallel with software development can be found in hardware except that the former has not reached the same level of maturity and formalization in design practices and control procedures.

Software cost can therefore be defined and summarized as follows:

$$\text{Cost per Instruction} = \frac{\text{Design cost + Coding cost + Verification cost + Maintenace cost}}{\text{No. Lines of Code}}$$

Costs identifed are predominatly labor costs and these in turn depend upon:

1. Firmness of requirements

2. Proportion of new vs proven algorithms

3. Size of program

4. Complexity of program

The resulting number of lines of code are attributable to:

1. No. functions assigned to software

2. Level of programming language

A survey of software costs experienced in the development of several recent tactical computer systems shows that real-time, operational software costs typically range from \$40 - \$60 per instruction, with off-line, non-operational programs costing \$8 - \$30 per instruction, depending on the proportion of new vs existing routines.

7.7.2 Cost Reduction Measures

From the foregoing observations, software cost can be reduced/minimized by:

- o Well Defined Requirements
- o Re-cyclable Program Modules
- o Small-Medium Size Programs (500 - 2K words)
- o Use Modular, Structured Design
- o Perform Balanced Hardware/Software Design Trade-off
- o Keep No. Lines of Code at Minimum i.e.
  - Use Higher-Order Language
  - Use machine architecture which minimizes overhead code (load and store)

7.7.3 Computer Architecture vs Software Cost

In contrast to rising software costs, which are predominantly man-hour dependent, computer semi-conductor hardware costs are rapidly diminishing.

While earlier mini-computer architectures were simple, to minimize hardware cost, such machines required a large number of overhead instructions to implement a given system function, typically 50-70% of the toal operational program, which in turn resulted in a high software cost compared to more efficient machine architectures.

To evaluate the benefits of lower-cost LSI computer hardware features with respect to software cost, established architectural features have been identified and their impact on hardware, throughput and software (Ref.R.10) is shown in Table 74. A simple 8-bit, single-accumulator computer was used as a reference with incremental improvements added and related to hardware, software and missile function execution. From this assessment it can be seen that computer instructions and hence software cost are reduced in each case, assuming assembly language coding.

TABLE 74

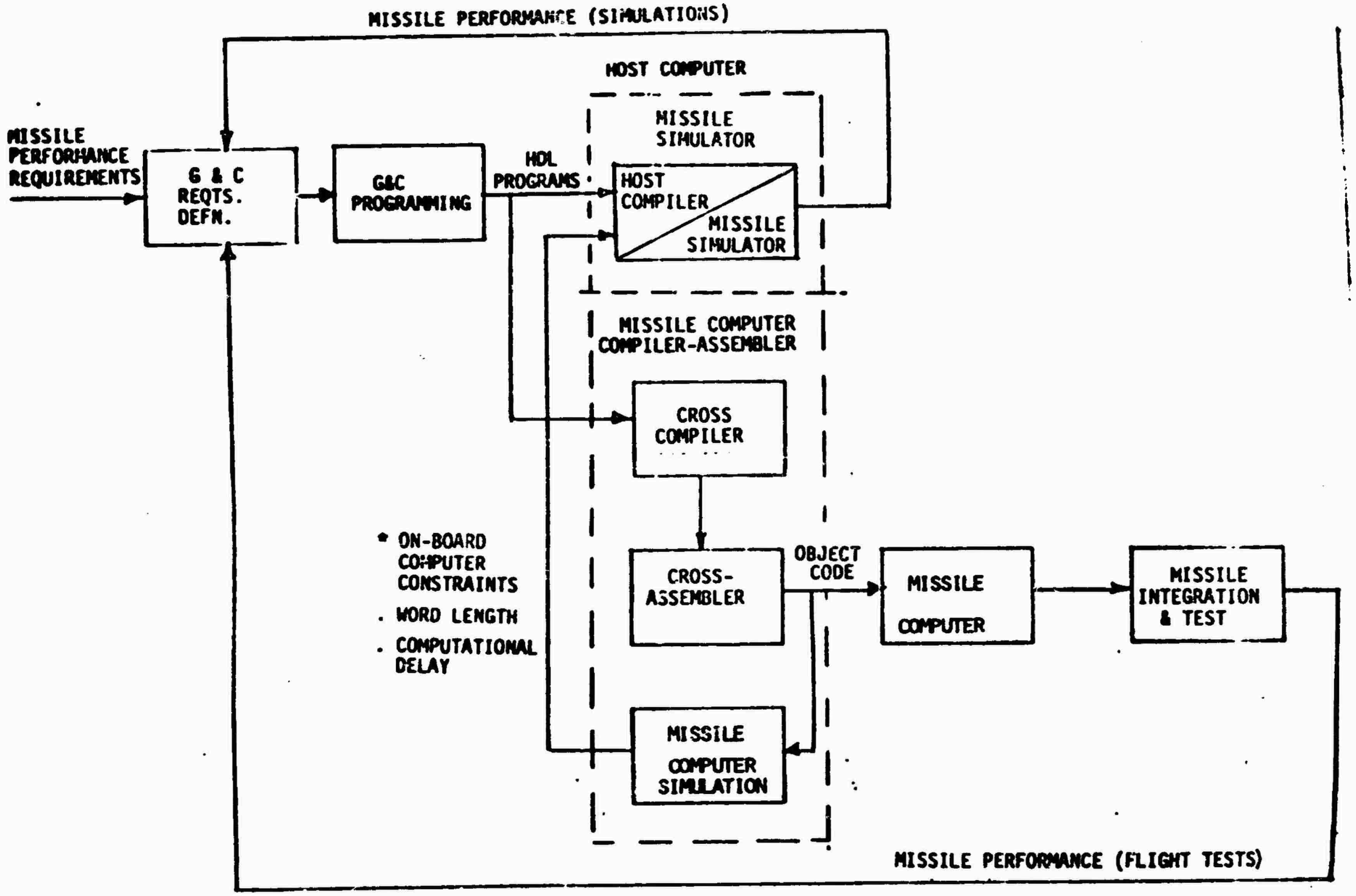
COMPUTER ARCHITECTURE VS SOFTWARE COST

(COMPARED TO SIMPLE 8-BIT SINGLE-ACCUMALATOR MACHINE)

	CENTRAL PROCESSING UNIT											
FEATURE	LOGIC	REGS.	FILE RAM	CONTROL (P)ROM	$	MAIN MEMORY RAM	MAIN MEMORY (P)ROM	MAIN MEMORY $	THROUGHPUT Kops	SOFTWARE INSTRUCTIONS	SOFTWARE $	APPLICATION
16-BIT WD. LENGTH	MORE	MORE			MORE	MORE	MORE	MORE	MORE	LESS	LESS	G&C
GP REGISTERS			MORE		MORE	LESS	LESS	LESS	MORE	LESS	LESS	T & S, G & C SIG. PROC.
PUSH-DOWN/POP-UP STACK		MORE	MORE		MORE		LESS	LESS	MORE	LESS	LESS	T & S, G & C SIG. PROC.
FLOATING-POINT	MORE	MORE	MORE	MORE	MORE	LESS	LESS	LESS	MORE	LESS	LESS	G & C
DMA I/O	MORE	MORE				LESS	LESS	LESS	MORE	LESS	LESS	T & S, G & C SIG. PROC.
2-ADDRESS INSTRUCTIONS				MORE	MORE	LESS	LESS	LESS	MORE	LESS	LESS	T & S, G & C SIG. PROC.
HARDWARE MULTIPLY	MORE	MORE			MORE				MORE			SIG. PROC.

7.7.4 Higher Order Programming Languages

In addition to the standardization, control and visibility afforded by higher order languages (HOLs) for software development albeit at the expense of less efficient code, (typically 10-20% reduction in throughput and a similar increase in memory capacity), the use of a common higher order language (e.g. DOD-1) for all military software can improve cost efficiency in digital missile development. Both the initial simulation of missile performance and the generation of object code for the on-board missile computer(s) can be accomplished interactively using the same HOL program modules. Figure I54 illustrates a unified approach to missile software development using a host computer and associated cross-compilers and assemblers. The customery approach is to code in one language for missile performance simulations e.g. FORTRAN, and re-code in a different language (i.e. assembly) for the on-board missile computer.



NOTES: * USED FOR 2ND PASS THROUGH MISSILE SIMULATIONS.

Figure 154 Unified Guidance and Control System Software Development Process for Digital Missiles.

References

R.1 "Modular Digital Missile Guidance System Study" Phase I Final Study Report, AD-784 969/8GA.

R.2 Jazwinski, A.H., "Stochastic Processes and Filtering Theory", Academic Press, New York, 1970.

R.3 Meyer and Meyer, "Radar Target Detection - Handbook of Theory and Practice", Academic Press, 1973.

R.4 Scarpino, F.A., Goodstein, R., "Digital Avionics Information System (DAIS) Integrated Test Bed Development", AIAA Digital Avionics Systems Conf., April 1975.

R.5 McCoy, B., "DAIS Avionic Software Development Techniques", AIAA Digital Avionics Systems Conf., April 1975.

R.6 Fisher, D.A., "Woodenman" Set of Criteria and needed characteristics for a Common DOD High Order Programming Language, Institute for Defense Analyses, Working Paper, August 13, 1975.

R.7 J.W. Cooley & J.W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", Math. Comp., V. 19, 1965, pp297-301. MR 31#2843.

R.8 Berglund, G.D., "Fast Fourier Transform Hardware Implementations - An Overview", IEEE Trans. on Audio & Electroacoustics Vol. Au-17, No. 2, pp104-108, June 1969.

R.9 "Military Standard Aircraft Internal Time Division Command/Response Multiplex Data Bus", MIL-STD-1553A, 30 April, 1975.

R.10 Flores, I., "Hardware Features That Aid Programming", Computer Design, pp48-54, January 1967.

APPENDIX A

LIST OF ABBREVIATIONS

Acc	Accelerometer
ACOS	Arc Cosine
A-D	Analog to Digital
ADAC	Analog to Digital, Digital to Analog
Converters	
AFC	Automatic Frequency Control
AGC	Automatic Gain Control
AI	Airborne Interceptor
AM	Amplitude Modulation
A-PD	Active Pulse Doppler
APN	Augmented Proportional Navigation
AR	Active Radar
ARM	Anti Radiation Missile
ASIN	Arc Sine
ATAN	Arc Tangent
AWCS	Aircraft Weapon Control System
BOB	Beginning of Block
BSE	Boresight Error
BW	Bandwidth
C	Number of Magnitude Bits
CCD	Charge Coupled Device
CFAR	Constant False Alarm Rate
CITS	Central Integrated Test System

cm	Centimeter (Meter x 10^{-2})
CPU	Central Processor Unit
CW	Continuous Wave
D	Miss Distance
D-A	Digital to Analog
DADD	Double Precision Add
DAIS	Digital Avionics Information System
db	Decibel
deg	Degree
DLO	Delayed Local Oscillator
DMA	Direct Memory Access
DMAIO	Direct Memory Access Input/Output
() DOF	() Degrees of Freedom
DTIME	Pre-Determined Time
ECCM	Electronic Counter-Countermeasures
ECM	Electronic Countermeasures
F	Signal to Quantization Noise Ratio
f	Frequency
FFT	Fast Fourier Transform
FM	Frequency Modulation
fps	Feet per Second
ft	Feet
FOJ	Fuse on Jam
FOV	Field of View
FSR	Feedback Shift Register

G	Gain
g	Acceleration of Gravity
GHz	Gigahertz (Hertz x 10^9)
GP	General Purpose
HAW	Homing All The Way
HMPY	Hardware Multiplier
HOJ	Home on Jam
HOL	Higher Order Language
hr	Hour
HWCU	Hardwired Control Unit
Hz	Hertz
I	Address Pointer
ICW	Interrupted Carrier Wave
ID	Identification
IF	Intermediate Frequency
ILPD	Injection Locked Pulse Doppler
in	Inch
INT	Integrator
I/O	Input/Output
I & Q	Inphase and Quadrature
IR	Infrared
IRCM	Infrared Countermeasures
J	Analog Channel Number
J/N	Jammer to Noise Ratio
kbits	Kilobits (bits x 10^3)

kft	Kilofeet (Feet x 10^3)
kg	Kilogram (Gram x 10^3)
KHz	Kilohertz (Hertz x 10^3)
km	Kilometer (Meter x 10^3)
Kops	Thousand Operations Per Second
lb	Pound
LET	Leading Edge Track
LIM	Limiter
LO	Local Oscillator
LOS	Line of Sight
LRU	Line Replaceable Unit
LSB	Least Significant Bit
m	Meter
MACH	Mach Number
MC	Mid-Course
MCI	Major Computing Interval
MHz	Megahertz (Hertz x 10^6)
MLC	Mainlobe Clutter
mm	Millimeter (Meter x 10^{-3})
Mops	Million Operations Per Second
mps	Meters Per Second
MR	Milliradian (Radian x 10^{-3})
msec	Millisecond (Second x 10^{-3})
n	Acceleration
NA	Not Applicable
nmi	Nautical Mile

N-MOS	N-Channel Metal Oxide Semiconductor
P	Probability
PCM	Pulse Code Modulation
PCU	Programmed Control Unit
PD	Pulse Doppler
PDI	Post Detection Integration
PDIO	Parallel Digital Input/Output
PLA	Programmable Logic Array
PN	Proportional Navigation
PRC	Pseudo-Random Coded
PRF	Pulse Repetition Frequency
PROM	Programmable Read Only Memory
P/Y	Pitch/Yaw
Q	Quadrature
R	Range
RALU	Register Arithmetic & Logic Unit
RAM	Random Access Memory
RDL	Range Desensitized Law
RF	Radio Frequency
RFI	Radio Frequency Interference
RIG	Rate Integrating Gyro
RMS	Root Mean Square
ROM	Read Only Memory
rps	Radians Per Second
S & A	Safing and Arming

SA-CW	Semi-Active CW
SA-PD	Semi-Active Pulse Doppler
SAR	Semi-Active Radar
SCL	Steering Command Loop
SCR	Signal to Clutter Ratio
SDIO	Serial Digital Input/Output
sec	Second
S/H	Sample and Hold
()SL	()State Law
SNR	Signal to Noise Ratio
SQRT	Square Root
TDD	Target Detection Device
TERM	Terminal
TIAS	Target Identification Acquisition System
TOA	Time of Arrival
TQI	Track Quality Indicator
TV	Television
TVC	Thrust Vector Control
V	Velocity

APPENDIX B

BURST AMPLITUDE WEIGHTING - THEORY OF OPERATION

B1 INTRODUCTION

A functional flow diagram illustrating burst amplitude weighting is shown in Figure B1 below. Burst amplitude weighting is shown applied to a signal which has been observed over a limited time duration (i.e. T_B) in order to reduce the resulting frequency spectrum sidelobes.

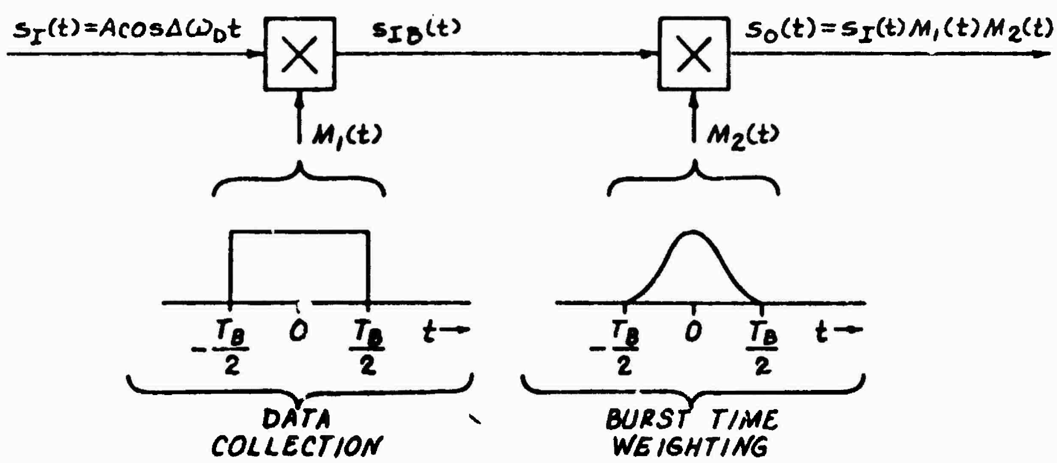


Figure B-1 Burst Amplitude Weighting - Functional Flow Diagram

B2 UNWEIGHTED SIGNAL SPECTRUM

Consider the signal of interest, $S_{I\beta}(t)$, to be a sinusoid observed for a time duration, T_β. This is represented as the product of the input signal, $S_I(t)$, and a rectangular pulse, $M_1(t)$. The spectrum of $S_{I\beta}(t)$ is determined as follows:

$$S_{I\beta}(t) = S_I(t) \cdot M_1(t)$$

taking the Fourier transform of both sides

$$S_{I\beta}(\omega) = S_I(\omega) * M_1(\omega)$$

where (*) denotes convolution

for $S_I(t) = A \cos \Delta\omega_D\, t$

$$S_I(\omega) = \frac{A}{2}\, \delta(\omega - \Delta\omega_D) + \frac{A}{2}\, \delta(\omega + \Delta\omega_D)$$

$$\text{where } \delta(\omega) = \begin{cases} 1 & \omega = 0 \\ 0 & \omega \neq 0 \end{cases}$$

for $M_1(t) = \mathrm{RECT}\left(\frac{t}{T_\beta}\right)$

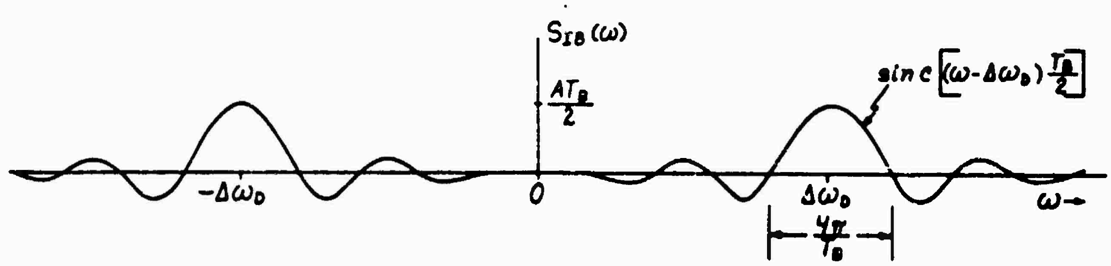
$$M_1(\omega) = T_\beta \ \mathrm{Sinc}\ \frac{\omega T_\beta}{2}$$

$$\text{where } \mathrm{Sinc}(x) = \frac{\mathrm{Sin}\,X}{X}$$

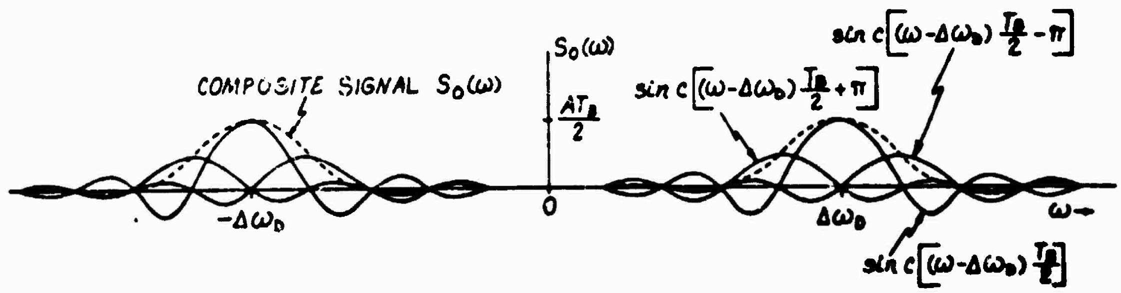
performing the convolution we obtain,

$$S_{I\beta}(\omega) = \frac{AT_\beta}{2}\ \mathrm{Sinc}\left[(\omega - \Delta\omega_D)\frac{T_\beta}{2}\right] + \frac{AT_\beta}{2}\ \mathrm{Sinc}\left[(\omega + \Delta\omega_D)\frac{T_\beta}{2}\right]$$

The spectrum of the unweighted/time-limited signal, $S_{I\beta}(\omega)$, is shown in Figure B2A. Note that the width of the spectrum mainlobe varies inversely with the burst duration (observation time/dwell time), T_β. Only in the case of an infinitely long observation time does the spectrum approach the two impulse situation corresponding to the Fourier transform of $\cos(\Delta\omega_D t)$.



(A) Spectrum of Unweighted/Time-Limited Signal



(B) Spectrum of Weighted/Time-Limited Signal

Figure B2 Burst Amplitude Spectral Relationships

B3 WEIGHTED SIGNAL SPECTRUM

Burst amplitude weighting is applied symmetrically over the observation interval of the signal of interest. The resultant weighted signal spectrum, $S_0(\omega)$, is determined as follows:

$$S_0(t) = S_{I\beta}(t) \cdot M_2(t)$$

$$\text{Let } M_2(t) = a + (1-a)\cos\left(2\frac{\pi\tau}{T_\beta}\right) \quad |t| \le \frac{T_\beta}{2}$$

$$\text{Where, } a = 0.5 \text{ for } \cos^2 \text{ weighting}$$

The restriction on the absolute value of t for $M_2(t)$ can be removed by rewriting $M_2(t)$ as:

$$M_2(t) = \text{RECT}\left(\frac{t}{T_\beta}\right)\left[a + (1-a)\cos\left(\frac{2\pi\tau}{T_\beta}\right)\right]$$

rewriting $S_0(t)$ in terms of the input signal:

$$S_0(t) = S_I(t) \cdot M_1(t) \cdot M_2(t)$$

$$= S_I(t) \cdot \text{RECT}\left(\frac{\tau}{T_\beta}\right) \cdot \text{RECT}\left(\frac{\tau}{T_\beta}\right) \cdot \left[a + (1-a)\cos\left(\frac{2\pi\tau}{T\beta}\right)\right]$$

Since the two RECT functions are defined over the same interval, only one is necessary, i.e.:

$$S_0(t) = S_I(t) \cdot \text{RECT}\left(\frac{\tau}{T_\beta}\right) \cdot \left[a + (1-a)\cos\left(\frac{2\pi\tau}{T_\beta}\right)\right]$$

taking the Fourier transform of both sides

$$S_0(\omega) = S_{I\beta}(\omega) * \left[a\,\delta(\omega) + \left(\frac{1-a}{2}\right)\delta\left(\omega - \frac{2\pi}{T_\beta}\right) + \left(\frac{1-a}{2}\right)\delta\left(\omega + \frac{2\pi}{T_\beta}\right)\right]$$

We see that the effect of the weighting is to produce a convolution of the burst time limited signal spectrum, $S_{I\beta}(\omega)$, with three impulses that are separated by $\omega = \frac{2\pi}{T_\beta}$ which corresponds to the spacing of the first null of the sinc function. Performing the indicated convolution we obtain:

$$
\begin{aligned}
S_e(\omega) = {} & \frac{\alpha A T_\beta}{2} \operatorname{Sinc}\left[(\omega - \Delta\omega_D)\frac{T_\beta}{2}\right] + \frac{(1-a)}{2} A T_\beta \operatorname{Sinc}\left[(\omega - \Delta\omega_D)\frac{T_\beta}{2} - \pi\right] \\
& + \frac{(1-a)}{2} A T_\beta \operatorname{Sinc}\left[(\omega - \Delta\omega_D)\frac{T_\beta}{2} + \pi\right] + \frac{\alpha A T_\beta}{2} \operatorname{Sinc}\left[(\omega + \Delta\omega_D)\frac{T_\beta}{2}\right] \\
& + \frac{(1-a)}{2} A T_\beta \operatorname{Sinc}\left[(\omega + \Delta\omega_D)\frac{T_\beta}{2} - \pi\right] + \frac{(1-a)}{2} A T_\beta \operatorname{Sinc}\left[(\omega + \Delta\omega_D) + \frac{T_\beta}{2} + \pi\right]
\end{aligned}
$$

The resultant weighted spectrum and the convolution to accomplish it is shown in Figure B-28. Note, that the situation corresponds to the case for $a = 0.5$ which is the so called cosine-squared weighting. Note also, that for this weighting that the spectrum mainlobe is broadened by a factor of 1.6 (3 db bandwidth).

APPENDIX C

UTILITY SUBROUTINE REQUIREMENTS AND DEFINITION

Utility subroutines in the computer software are defined as those routines which provide basic mathematical functions. They are used to support a variety of missile functional algorithms and are called as frequently as necessary to satisfy functional computations. Those necessary for an on board missile computer are listed in Table C-1 with associated instruction type and memory requirements. For this tabulation, the equivalent adds are based on a ratio of eight multiplies to one add and a 30% overhead burden was assumed. As shown, the subroutines have modest requirements and do not by themselves dictate computer sizing.

Some of the routines such as the digital filters provide a level of flexibility which is attractive in a small computer. the algorithms used to provide a four pole filter, for example, can be used to implement a transfer function which is a fourth order lag or a lead/lag filter with any order numerator up to four. The only difference from one transfer function to another are the values of the constant coefficients input to the software routine.

The algorithms used to define utility requirements are as shown in Table C-2. All the trignometric functions are based on series expansions and a four term expansion is considered adequate for most applications. The table look-up routines only become significant when functions of three variables are required and this has been assumed necessary for a Class III missile only.

Table C-3 defines the mix of standard routines assumed for computer sizing for each missile class. A minimum of 375 words of program memory is required with a corresponding minimum of 141 words of data memory. The maximum requirement results in 489 words and 177 words respectively.

TABLE C-1

UTILITY SUBROUTINES - REQUIREMENTS SUMMARY

Module	Name	Add/ Sub.	Mult./ Divide	Load/ Store	Other Utilities	*Equiv. Adds	Memory Program ROM	Memory Data RAM
U1.	SIN	4	6	3		72.	23	8
U2.	COS	4	5	3		62.	21	8
U3.	TAN	4	6	3		72	23	8
U4.	ATAN	4	6	3		72	23	8
U5.	ASIN	5	4	4	1-SQRT	102	26	8
U6.	ACOS	4	4	4	1-SQRT	101	24	7
U7.	SQRT	3	4	2		49	37	25
U8.	EXP	4	4	2		48	20	7
U9.	EULER INTEG.	4	1	11		30	34	2
U10.	LIMIT	2	0	9		14	14	3
U11.	VECTOR ROTATION	8	9	46		164	57	1
U12.	DOUBLE ADD	3	0	11		18	18	1
	DIGITAL FILTER							
U13.	1 Pole	2	3	7		43	17	7
U14.	2 Poles	4	5	13		74	31	9
U15.	3 Poles	6	7	19		105	45	11
U16.	4 Poles	8	9	25		137	59	13
	TABLE LOOK UP							
U17.	1 way	4	1	8		26	15	5
U18.	2 way	12	4	24		88	35	8
U19.	3 way	28	9	56		203	45	12
U20.	AVG. NORMAL	9	1	20		48	22	5
	FFT							
U21.	64 pt. FFT	1152	768	3456		13978	33	
U22.	128 pt. FFT	2688	1792	8064		32614	33	
	CORNER TURNING							
U23.	64 pt.	64	0	256		416	15	
U24.	128 pt.	128	0	512		832	15	
	BURST WEIGHTING							
U25.	64 pt.	0	128	256		1664	15	64
U26.	128 pt.	0	256	512		3328	15	128

NOTES:

* Includes additional 30% overhead burden.

TABLE C-2

UTILITY SUBROUTINE ALGORITHMS

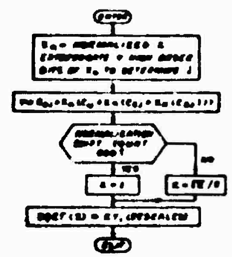
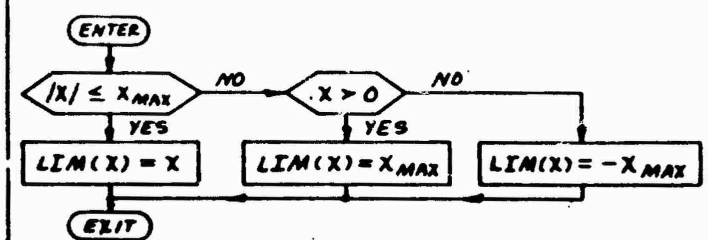
MODULE CODE	MODULE NAME	INPUTS	CONSTANTS	ALGORITHM	OUTPUT
U1	SIN	X	A_1, A_3, A_5, A_7, A_9	$SIN(X) = X(A_1 - X^2(A_3 - X^2(A_5 - X^2(A_7 - X^2(A_9)))))$	SIN(X)
U2	COSINE	X	A_0, A_2, A_4, A_6, A_8	$COS(X) = A_0 - X^2(A_2 - X^2(A_4 - X^2(A_6 - X^2(A_8))))$	COS(X)
U3	TANGENT	X, $(X^2 < \pi^2/4)$	A_1, A_3, A_5, A_7, A_9	$TAN(X) = X(A_1 + X^2(A_3 + X^2(A_5 + X^2(A_7 + X^2(A_9)))))$	TAN(X)
U4	INVERSE TANGENT	X, $(X^2 \le 1)$	A_1, A_3, A_5, A_7, A_9	$ATAN(X) = X(A_1 + X^2(A_3 + X^2(A_5 + X^2(A_7 + X^2(A_9)))))$	ATAN(X)
U5	INVERSE SINE	X, $(0 \le X \le 1)$	A_0, A_1, A_2, A_3	$ASIN(X) = \pi/2 - (A_0 + X(A_1 + X(A_2 + X(A_3)))) \sqrt{1-X}$	ASIN(X)
U6	INVERSE COSINE	X, $(0 \le X \le 1)$	A_0, A_1, A_2, A_3	$ACOS(X) = (A_0 + X(A_1 + X(A_2 + X(A_3)))) \sqrt{1-X}$	ACOS(X)
U7	SQUARE ROOT	X	C_{ij}		SQRT(X)
U8	EXPONENTIATION	X	A_0, A_1, A_2, A_3, A_4	$EXP(X) = A_0 + X(A_1 + X(A_2 + X(A_3 + X(A_4))))$	EXP(X)
U9	EULER INTEG.	$\Delta T, \dot{X}, X, S_X, S_{\dot{X}}$		$\Delta X = \Delta T\ \dot{X}$ (SCALED) , $X = X + \Delta X$	X
U10	LIMIT	X, X_{MAX}			LIM(X)

TABLE C-2 (CONT)

UTILITY SUBROUTINE ALGORITHMS

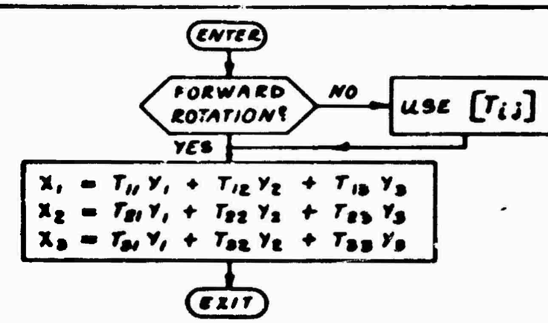
MODULE CODE	MODULE NAME	INPUTS	CONSTANTS	ALGORITHM	OUTPUT
U11	VECTOR ROTATION	$[T_{ij}]$, Y, F			X
U12	DOUBLE ADD	(X_1, X_2), (Y_1, Y_2)		$(Z_1, Z_2) = (X_1, X_2) + (Y_1, Y_2)$	(Z_1, Z_2)
U13	1 POLE DIGITAL FILTER	$X_1, X_{-1}, Y_{-1}, A, B, C$		$Y = AX + BX_{-1} - CY_{-1}$ (FILL PAST VALUES FOR NEXT ITERATION)	Y
U14	2 POLE DIGITAL FILTER	$X, X_{-1}, X_{-2}, Y_{-1}, Y_{-2}, A, B, \ldots E$		$Y = AX + BX_{-1} + CX_{-2} - DY_{-1} - EY_{-2}$ (FILL PAST VALUES FOR NEXT ITERATION)	Y
U15	3 POLE DIGITAL FILTER	$X, X_{-1}, X_{-2}, X_{-3}, Y, Y_{-2}, Y_{-3}, A, B, \ldots G$		$Y = AX + BX_{-1} + CX_{-2} + DX_{-3} - EY_{-1} - FY_{-2} - GY_{-3}$ (FILL PAST VALUES FOR NEXT ITERATION)	Y
U16	4 POLE DIGITAL FILTER	$X, X_{-1}, X_{-2}, X_{-3}, X_{-4}, Y_{-1}, Y_{-2}, Y_{-3}, Y_{-4}, A, B, \ldots I$		$Y = AX + BX_{-1} + CX_{-2} + DX_{-3} + EX_{-4} - FY_{-1} - GY_{-2} - HY_{-3} - IY_{-4}$ (FILL PAST VALUES FOR NEXT INTERATION)	Y
U17	1-WAY TABLE LOOK-UP	α_n, L_F		$F(\alpha) = F(\alpha_n) + (F(\alpha_{n+1}) - F(\alpha_n)) * \Delta_\alpha$	$F(\alpha)$
U18	2-WAY TABLE LOOK-UP	α_n, β_n, L_F		$F_1 = F(\alpha_n,\beta_n) + (F(\alpha_n,\beta_{n+1}) - F(\alpha_n,\beta_n)) * \Delta_\beta$ $F_2 = F(\alpha_{n+1},\beta_n) + (F(\alpha_{n+1},\beta_{n+1}) - F(\alpha_{n+1},\beta_n)) * \Delta_\beta$ $F(\alpha,\beta) = F_1 + (F_2 - F_1) * \Delta_\alpha$	$F(\alpha,\beta)$

TABLE C-2 (CONT)

UTILITY SUBROUTINE ALGORITHMS

MODULE CODE	MODULE NAME	INPUTS	CONSTANTS	ALGORITHM	OUTPUT
U19	3-WAY TABLE LOOK-UP	$\alpha_n, \beta_n, \gamma_n, L_F$		$F_1 = F(\alpha_n, \beta_n, \gamma_n) + (F(\alpha_n, \beta_n, \gamma_{n+1}) - F(\alpha_n, \beta_n, \gamma_n)) * \Delta_\gamma$ $F_2 = F(\alpha_n, \beta_{n+1}, \gamma_n) + (F(\alpha_n, \beta_{n+1}, \gamma_{n+1}) - F(\alpha_n, \beta_{n+1}, \gamma_n)) * \Delta_\gamma$ $F_3 = F(\alpha_{n+1}, \beta_n, \gamma_n) + (F(\alpha_{n+1}, \beta_n, \gamma_{n+1}) - F(\alpha_{n+1}, \beta_n, \gamma_n)) * \Delta_\gamma$ $F_4 = F(\alpha_{n+1}, \beta_{n+1}, \gamma_n) + (F(\alpha_{n+1}, \beta_{n+1}, \gamma_{n+1}) - F(\alpha_{n+1}, \beta_{n+1}, \gamma_n)) * \Delta_\gamma$ $F_5 = F_1 + (F_2 - F_1) * \Delta_\beta$ $F_6 = F_3 + (F_4 - F_3) * \Delta_\beta$ $F(\alpha, \beta, \gamma) = F_5 + (F_6 - F_5) * \Delta_\alpha$	$F(\alpha, \beta, \gamma)$
U20	ARGUMENT NORMALIZATION	α, N_{BT}, L_{BT}		FIND INTEGER BREAKPOINT, n, SUCH THAT $\alpha_n \leq \alpha \leq \alpha_{n+1}$ WHERE $0 \leq n \leq N-2$. $BIN = \alpha_{n+1} - \alpha_n$ $\Delta = (\alpha - \alpha_n) / BIN$ NOTE: $0 \leq \Delta \leq 1$ (INTERPOLATION) $n = 0, \Delta < 0$ (EXTRAPOLATION BELOW α_0) $n = N-2, \Delta > 1$ (EXTRAPOLATION BEYOND α_{N-1})	α_n
U21	64-POINT FFT	64 COMPLEX DATA POINTS/SAMPLES	32 COMPLEX MULTIPLIERS ($\omega = \exp(-j2\pi/N)$)	COOLEY-TUKEY, ITERATIVE 2-POINT COMPLEX TRANSFORM (BUTTERFLY)	64 COMPLEX SPECTRAL COMPONENTS
U22	128-POINT FFT	128 COMPLEX DATA POINTS SAMPLES	64 COMPLEX MULTIPLIERS ($\omega = \exp(-j2\pi/N)$)	COOLEY-TUKEY, ITERATIVE 2-POINT COMPLEX TRANSFORM (BUTTERFLY)	128 COMPLEX SPECTRAL COMPONENTS
U23	64-POINT CORNER-TURNING	64 COMPLEX DATA POINTS x NO. OF RANGE GATES		SEE SUBSECTION 4.2.3.1	64 COMPLEX DATA POINTS x NO. OF RANGE GATES
U24	128-POINT CORNER-TURNING	128 COMPLEX DATA POINTS x NO. OF RANGE GATES		SEE SUBSECTION 4.2.3.1	128 COMPLEX DATA POINTS x NO. OF RANGE GATES
U25	64-POINT BURST WEIGHTING	64 COMPLEX DATA POINTS	64 COS^2 MULTIPLIERS	SEE APPENDIX B	64 COMPLEX DATA POINTS
U26	128-POINT BURST WEIGHTING	128 COMPLEX DATA POINTS	128 COS^2 MULTIPLIERS	SEE APPENDIX B	128 COMPLEX DATA POINTS

TABLE C-3

UTILITY SUBROUTINES VERSUS MISSILE CLASS

UTILITY ROUTINE	MISSILE CLASS I	II		III	
		Min	Max	Min	Max
SIN	X	X	X	X	X
COS	X	X	X	X	X
TAN				X	X
ATAN		X	X	X	X
ASIN	X	X	X	X	X
ACOS				X	X
SQRT	X	X	X	X	X
EXP.		X	X	X	X
EULER INTEG.	X	X	X	X	X
LIMIT	X	X	X	X	X
VECTOR ROTATION	X	X	X	X	X
DOUBLE ADD	X	X	X	X	X
DIGITAL FILTER					
1 Pole					
2					
3	X	X			
4 Pole			X	X	X

TABLE C-3

UTILITY SUBROUTINES VERSUS MISSILE CLASS

UTILITY ROUTINE	MISSILE CLASS I	II Min	II Max	III Min	III Max
TABLE LOOK UP					
1 way	X	X	X		
2					
3 way				X	X
ARG. Normal	X	X	X	X	X
FFT (64 pt.)	X	X	X	X	X
CORNER TURNING (64 pt.)	X	X	X	X	X
BURST WEIGHTING (64 pt.)	X	X	X	X	X
TOTAL Program Memory (words)	375	418	432	489	489
Data Memory (words)	141	156	158	177	177

LEGEND: X Denotes use of utility routine

DISTRIBUTION LIST

Chief of Naval Research
800 N. Quincy Street
Arlington, VA 22217
ATTN: Code ONR-211 (5)
Code ONR-437 (1)

Director
ONR Branch Office
495 Summer Street
Boston, MA 02210 (1)

DCASR
666 Summer Street
Boston, MA 02210 (1)

Director
U. S. Naval Research Laboratory
Washington, DC 20375
ATTN: Code 2629 (6)
Code 2627 (6)
Code 5493 (1)

Defense Documentation Center
Bldg 5
Cameron Station
Alexandria, VA 22314 (12)

Chief of Naval Material
Deputy Chief of Naval Material
for Development
Washington, DC 20362
ATTN: Code NMAT 032 (1)
Code NMAT 0322 (1)

Chief of Naval Operations
Deputy Chief of Naval Operations
for Development
Washington, DC 20350
ATTN: Code NOP 982E2L (1)
Code NOP 506F (1)
Code NOP 983F (1)

Commander
Naval Sea Systems Command
Washington, DC 20362
ATTN: Code SEA 034 (1)

Commandant of the Marine Corps
Deputy Chief for Research Development
and Studies
Washington, DC 20380
ATTN: Code MC-RD-1 (1)

Headquarters Air Force System
Command
Andrews Air Force Base, MD 20334
ATTN: Code AFSC/DLCA (1)
Code AFSC/SDZ (1)
Code AFSC/XRL (1)

Armament Development and Test Center
Eglin Air Force Base, FL 32542
ATTN: Code ADTC/DLM (1)
Code ADTC/XR (1)

Air Force Avionics Laboratory
Wright Patterson Air Force Base, OH 45433
ATTN: Code AFAL/AA (1)
Code AFAL/RWT (1)

U. S. Army Missile Command
Redston Arsenal, AL 35809
ATTN: DRSMI/RGG (1)

Commanding Officer
Naval Weapons Center
China Lake, CA 93555
ATTN: Code 40 (1)
Code 403 (1)
Code 406 (1)
Code 4032 (1)
Code 3924 (1)
Code 3944 (1)
Code 39 (1)

Commanding Officer
Naval Air Development Center
Warminster, PA 18974
ATTN: Code 2072 (1)

Commanding Officer
Naval Avionics Facility
Indianapolis, IN 46218
ATTN: Code D-823 (1)

Code SEA 654311 (1)

Commander
Naval Air Systems Command
Washington, DC 20361
ATTN: Code NAIR 360B (1)
Code NAIR 360E (1)
Code NAIR 03P21 (1)
Code NAIR 52022 (1)
Code NAIR 5333 (1)
Code NAIR 503 (1)
Code NAIR 5105 (1)

The Analytic Sciences Corporation
6 Jacob Way
Reading, MA 01867
ATTN: C. F. Price (1)

Applied Physics Laboratory
Johns Hopkins University
Johns Hopkins Rd
Laurel, MD 20810
ATTN: Dr. R. E. Gordozdos (1)