REPORT NO. DOT-TSC-FAA-73-4

PRELIMINARY STUDIES OF ACCESS CONTROL AND PROCESSING FOR OCEANIC AERONAUTICAL SATELLITE

David Clapp
Grant Paul
Robert Snow
Gary Wang

JUNE 1973

PRELIMINARY MEMORANDUM

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. DOT-TSC-FAA-73-4	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle PRELIMINARY STUDIES OF ACCESS CONTROL AND PROCESSING FOR OCEANIC AERONAUTICAL SATELLITE		5. Report Date June 1973
		6. Performing Organization Code
7. Author(s) David Clapp, Grant Paul, Robert Snow, Gary Wang		8. Performing Organization Report No. DOT-TSC-FAA-73-4
9. Performing Organization Name and Address Department of Transportation Transportation Systems Center Kendall Square Cambridge, MA 02142		10. Work Unit No. R-3149/FA-311
		11. Contract or Grant No.
12. Sponsoring Agency Name and Address Department of Transportation Federal Aviation Administration Systems Research and Development Service Washington, DC 20591		13. Type of Report and Period Covered Preliminary Memorandum June 1971 - June 1972
		14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

This report concerns problems having a determinative influence on the avionics and ground interfaces for the Aeronautical Oceanic Satellite (AEROSAT) system. There are the general principles and parameters of access control along with a classification of the available options. The access control method which will ultimately be selected has a determinative role in organization of the communications and surveillance functions.

The manner in which surveillance will be performed has important implications on how the entire system functions, and particularly on the air and ground interfaces. There also are (1) a qualitative view of the computer requirements, (2) the technology applicable to the AEROSAT system, and (3) an overview of oceanic air traffic control (ATC) as it exists now and as it is generally planned for the future.

17. Key Words Air Traffic Control, Communication Management, Communication Satellites, Computer Systems Hardware, Surveillance	18. Distribution Statement THIS DOCUMENT CONTAINS PRELIMINARY INFORMATION SUBJECT TO CHANGE. IT IS CONSIDERED AN INTERNAL TSC WORKING PAPER WITH A SELECT DISTRIBUTION. IT IS NOT A FORMAL REFERABLE REPORT. DISTRIBUTION IS EFFECTED BY AND RESPONSIBILITY OF THE AUTHOR.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	54	

Form DOT F 1700.7 (8-69)

FOR OFFICIAL USE ONLY

TABLE OF CONTENTS

1. INTRODUCTION

This report covers a portion of the work performed to date at TSC under the Aeronautical Oceanic Satellite (AEROSAT) program. The reported work is particularly concerned with problems having a determinative influence on the avionics and ground interfaces; namely, access control, surveillance, processing, and systems' interfacing.

Although it is not the intention here to address the complete system design, our topics do touch on fundamental system questions. The access control method which will ultimately be selected has a determinative role in organization of the communications and surveillance functions. The manner in which surveillance will be performed has important implications on how the entire system functions, and particularly on the air and ground interfaces. The organization of processing elements is also a fundamental determinant in overall system design. The system evolution is also influenced strongly by the structure of present oceanic Air Traffic Control (ATC) and practical transitional considerations.

In Section 2, we discuss the options available for access control. The major effort in this area in the coming year will be carried out under contract; our principal activity to date has been in defining the issues to be addressed and preparing a detailed work outline in frequent consultation with FAA, OST, and industry. In this report, we cover the general principles and parameters of access control, the classification of available options, and the scope of work which will continue.

Section 3 looks to computer requirements and technology applicable to the AEROSAT system. In light of the uncertainty of an AEROSAT system concept when this work was undertaken, it was not possible to obtain a quantitative basis for establishing memory size, instruction rates, etc. It was appropriate, however, to address the options for computer system configuration which could handle the functions efficiently and reliably.

Section 4 is an overview of oceanic ATC, as it exists now and as it is generally planned for the future. In the present dynamic environment, details of this system are constantly changing, but the general features are presumably stable.

2. ACCESS CONTROL OPTIONS

2.1 CHANNEL-SHARING

The principal functions involved in access control are: making channel requests, assignments, reassignments, and relinquishments; acknowledging receipt of and compliance with the required orders; and the maintenance of a directory file of current channel assignments for controlling the communications between each aircraft and the ground.

Access control requirements are particularly evident when many aircraft must share a single channel or a small group of channels.* However, it should be pointed out that even in an extreme case in which every aircraft has its own reserved channel, there would still be a requirement for access control to perform the functions listed above. The simplicity of the access control system for this case lies in the small number of channel assignments, etc., which it is called upon to perform rather than in any significantly reduced functionality. Furthermore, that simplicity is gained in trade for an expensive and perhaps unattainable proliferation in the number of independent channels which are constructed. The idea of "one aircraft, one channel" is a useful philosophical endpoint but it is regarded as more of a theoretical than a practical concept.

By a slight increase in functionality of the access control system (and a considerable increase in its complexity), many low-duty-cycle users can efficiently share a common channel or a small group of common channels. Available techniques for accomplishing this include:

a. Full contention among a group of aircraft assigned (perhaps dynamically) to each channel.

---

*We use "channel" in the general sense to include frequency assignment, time-slot assignment, orthogonal code assignment, etc., or combinations thereof.

b. Polling underground control. With "dynamic" polling, no fixed cyclic order need be followed, and the polling rate for each aircraft, or subregion, can be tailored to the operational need of the moment.

c. Orderwire(s) for requests, assignments and acknowledgments. (In simplest form this may be regarded as full contention on much shorter "surrogate" messages.)

d. Hybrids of the above.

Most of the realistic options are probably in the comprehensive "hybrid" category. An example would be a combination of polling for handling large volumes of non-priority messages, with contention and/or orderwire for low-volume priority messages.

2.2 ACCESSING TECHNIQUES

From the viewpoint of the user, four principal concepts for channel management discipline may be recognized. These are referred to as: on demand, as available, as polled, and as scheduled.

ON DEMAND (random access) - In this method, the user determines channel availability. When communication is desired, the user monitors the channel and determines its occupancy. If all channels are occupied, he must wait for one to become available (specific channels would be reserved for emergency use). Once a channel becomes available, the user addresses the desired communicant and upon confirmation delivers his message. Channel discipline depends on observance of procedures by the users. The principal advantage of this accessing technique is the minimum airborne and ground equipment requirements for system implementation (since control and administrative functions are not required). The most serious disadvantage is its susceptibility to channel saturation.

AS AVAILABLE (first come, first served) - In this method, the user requests a channel via a low-rate digital data link. The control center processes the request and connects the user to the desired party (assuming a channel is immediately available). If a channel is not available, the control center acknowledges and stores the request (in order of reception). When the requested link becomes available, the user is alerted and connected to the desired party. The principal advantage of this technique is that the user is freed of the tedious task of searching for an unoccupied channel; also, the user is assured of a communication channel when his turn comes up. A disadvantage is that there is no control of the entire user complement and its communication requirements as a function of time. Consequently, a channel-demand-peaking condition still exists.

AS POLLED (interrogation and response) - In this method, the control center sequentially polls each user within the system via a low-rate digital data link (interrogation), and a coded response from the user indicates when a communication channel is requested. This code is received by the control center, which in turn inserts the accessing information into the forward link when a channel becomes available. Aircraft which have already entered the polling system do not initiate transmissions but wait to be polled; this may have to be modified to cover certain emergency contingencies. The principal advantage of this technique is that it is highly compatible with digitally oriented ATC systems projected for the future, in addition to the capability of smoothing the demand peaks on the satellite. The principal disadvantage of this technique is the increased avionic and ground-equipment complexity and cost.

AS SCHEDULED (assigned time slots) - This method (with the exception of the interrogation cycle being omitted) is similar to the AS-POLLED technique. In the AS-SCHEDULED technique, the users are assigned specific time slots (by the control center) in which to request a channel. The principal advantage of this system is the provision for uniform, efficient use of the satellite capability. The principal disadvantage is ground-equipment and avionic complexity (and cost) since accurate time bases are required by the users and control center.

The accessing technique ultimately selected will most likely be a combination of two or more of the above accessing techniques since it is not clear that any single technique meets all of the requirements. There are several additional considerations which tend to promote hybridization of these pure system types.

a. The technique for managing multiple access may vary as a function of message priority. A reasonable example would be the on-demand or as-available management of high-priority messages combined with polling for lower-priority messages.

b. There is an inherent hybrid character to many access control systems arising from the fact that the ground is a singular type of "user." Ground-originated messages will generally be handled quite differently than aircraft-originated messages.

c. In operational practice, the access control method may be varied somewhat to match the time of day, traffic density, etc.

The full impact of such hybridizing factors cannot be fully grasped in advance; they are best studied in specific contexts at a later stage. The various tradeoffs between avionic cost and complexity, system compatibility, and channel efficiency remain to be investigated.

2.3 OCCUPANCY FACTOR

Some of the most basic variables of the multiple access problem include:

N_A = number of aircraft in the operation (instantaneous airborne count),

R_M = number of messages per unit time per aircraft, on the average (message rate),

L = average message duration, and

C = number of channels available.

Important insight into the problem can be gained by noting that the principal effects of these four variables can be summarized by a dimensionless parameter, the average occupancy factor,

$$F = \frac{N_A R_M L}{C} .$$

In this definition, it is assumed that the denominator, C, is the number of channels actually available for communication; i.e., that any channel relinquished by an exiting aircraft is immediately reassigned to another aircraft.

The parameter, F, is a dimensionless number which depends not only on the design of the communication-management system but also on its operational state. For an overdesigned system, or for any system in slack periods, F may be small enough (<0.1, say) so that free channels are usually available on demand, and queues when they exist are rarely longer than a single message. When conditions change so that F lies near the middle of its range (0 to 1) then busy signals, queues, priority assertions, etc., begin to become important, and become critical as F approaches 1. F cannot be greater than 1 without causing infinite queues or lost messages. In the context of queuing theory, F is the ratio of arrival rate to servicing rate for messages. The importance of the parameter F is indicated by the following table which is based on random message times and a single priority class.

TABLE: DEPENDENCE OF CONFLICTS ON AVERAGE OCCUPANCY FACTOR, F

Average Occupancy Factor (F)	0.1	0.3	0.5	0.7	0.9	0.95
Probability That Random Channel Will Be Busy	0.1	0.3	0.5	0.7	0.9	0.95
Probability That Queue When It Exists Has at Least Two Ahead	0.1	0.3	0.5	0.7	0.9	0.95
Average Length of Queue When One Exists	1.1	1.4	2.0	3.3	10.0	20.0

This discussion indicates that the dependence on F can be severe, and that a rather sophisticated access-control scheme may be required to cope with a variety of operational situations. To illustrate, consider a hypothetical case of 100 airborne aircraft each calling in for 30 seconds, every 20 minutes, using 4 channels. The occupancy factor for this case is

$$F = (100)\ (1/1200)\ (30)/4 = 0.625 .$$

This is a rather busy situation, but one which can be handled without undue strain by several rather straightforward access-control systems. However, if the number of aircraft, or the calling rate, or the message length (any one of them), increases by 50 percent, the new value of F is 0.94. In this new situation, channels are normally busy (94-percent probability); queues are normally multiple (94-percent probability); and the average queue length is 16. Queues of greater than average length are of course possible; the communication-management system would have to be capable of queuing up to 32 requests if it is desired to keep the queue-rejection probability as low as 10^{-4} (queue rejections once every few months).

It should be pointed out that this example is not proposed as describing any particular operational situation but rather to illustrate the importance of the average occupancy factor (F), the insight which it gives into systems options, and the sometimes dramatic effect of changes in the component factors.

2.4 FUTURE WORK

The principal effort in access control in the new fiscal year will focus on the performance of the following tasks.

Task A - Analysis of System Access-Control Options - The objective of this task is to carry out a detailed investigation of multiple access concepts and channel-management discipline applicable to satellite communications links between fixed terminals and mobile platforms. Primary emphasis is to be placed initially on Oceanic Aeronautical Satellite Systems, but the expansion capability to cover maritime applications will also be considered.

Task B - Identification of Candidate Access-Control Approaches - The most promising of the access-control schemes delineated under task A will be identified for further detailed tradeoff analyses and for use in defining the data-terminal transmission requirements.

Task C - Development of Data-Transfer Requirements - The objective of this task is to carry out a detailed investigation of the associated data-terminal transmission requirements for Aeronautical Satellite Systems. Emphasis is to be placed upon the identification, description, and implementation tradeoff analysis of ground terminal tasks and processing elements as required to perform communications management, data-formatting and buffering, and tasks associated with ATC interface functions. The system access-control schemes are to be developed in sufficient detail to define all input-output information-transfer requirements.

Task D - Detailed Studies of Ground and Avionics Equipment - The results of this task represent an essential output of the study program and will be based on the work performed in defining each access-control scheme under tasks A and B. The selected schemes of task B are to be further refined and detailed in description of access-control techniques as well as performance and mechanization details.

Task E - Extension to Other Services - The objective of this task is to consider the possibilities for expanding the capability to applications such as maritime shipping, involving a multiplication of the number of users by a factor of 10 to 100.

3. COMPUTER REQUIREMENT AND TECHNOLOGY ASSESSMENT

3.1 STATEMENT OF WORK

Computers must be supplied at the ground terminal of the AEROSAT system to carry out the access control, surveillance, and communications functions of the system. The nature of the data-processing tasks for the access-control function has been discussed in Section 2 of this report; the data-processing for communications purposes will be estimated at a later date when the interface characteristics with the ATC center(s) are defined. It will be necessary to select and recommend types and sizes of processing equipment, which will do the job and, at the same time, will satisfy the additional requirements of reliability and flexibility.

Reliability Considerations

As in most applications of data-processing equipment to the ATC environment, the oceanic access control-and-surveillance system will be required to provide continuous, uninterrupted service with miniscule probabilities of failure and instant recovery from that unlikely event. The key concept is that of availability: the percentage of time that the system can perform as required.[1] This is a rather imprecise definition which can, however, be made precise with a proper definition of "required system performance," or better, "system performance requirements."

For real-time systems such as the one under discussion, the primary performance requirement is that the system should continue to cycle; i.e., to remain in real-time operation. In other words, there should never be a catastrophic failure. In actual practice, it is most possible to guarantee continuous operation absolutely, but it is practicable to reduce the probability of an interruption to a very low level.

On the other hand, there are in every system some functions which, while useful and desirable, are not essential. If the system is properly designed, partial failure of the system may be tolerated after a reallocation of functions, such that the essential ones are performed at the expense of the non-essential. Further, it may be possible to operate the system for short periods of time in a degraded mode with respect to load if provision can be made to ensure that no high loads are imposed during the period of system incapacity.

We have introduced implicitly two more system-performance requirements: load capacity and functional capability. Load capacity is a measure of the maximum number of entities which can be passed through the system at any one time and is usually expressed in ATC systems in terms of number of aircraft. Clearly, there are frequently subsystems which do not handle aircraft per se but rather deal with flight plans, interfacility messages, display blocks, etc. For these, the load capacity will be defined differently, but for the system as a whole, the major data-processing entity will be the aircraft being tracked.

The functional capability of the system is the set of processes by which the system transforms the various input streams to produce the output streams. Certain of the output streams are essential to the successful operation of the system; those functions which produce the essential output form the set of essential functions which define system functional capability. In a sense, the product of the load-capacity and functional capability is a measure of the "power" of the system, the amount of work done per unit time.

We can now state more precisely the concept of availability. The availability of a real-time system is that percentage of the time when the system is: (a) cycling, and (b) able to perform the set of essential functions on at least the minimum acceptable number of processing units.

Note that there is a transient state to be accounted for in the case of a partial system failure. It may be that the minimum acceptable number of processing units, or aircraft, in the system when operating in a degraded mode is N which is less than the maximum which can be handled by the full system. If, when the failure occurs, the number of aircraft in the system, n(t), is MN, then the "minimum acceptable number" will be $n_a(t)$, a function of time which starts at M and remains equal to the number of aircraft in the system until that number becomes less than or equal to N.

$$n_a t = \begin{cases} M, \text{ at } t = t_0 \\ n(t), \text{ for } N \; n(t) \; M \\ N, \text{ when } n(t) \; N \end{cases}$$

The system must somehow handle that situation.

There are two ways to make a system more reliable, and hence, more available: one way is to increase the reliability of each component, by using high-quality material and by increasingly careful production techniques for instance. The other way is to provide more components arranged such that alternate means are available for performing any task which becomes bogged down through component failure. Clearly, the first method will reach the point of diminishing returns at some time, and if the level of reliability is too low, then the redundancy route will be the only alternative.

Furthermore, there may be parts of the system which cannot be upgraded, or held to high levels of reliability. These parts must be duplicated if they are essential to system operation or they will become the weak links which determine the overall unreliability of the whole system.

Flexibility Considerations

If the requirements placed upon a system are known and reasonably constant in time, then the computer and other subsystem components may be selected with a good deal of confidence that they will be adequate for the job. That is to say, computer-performance evaluation techniques have been developed to the point where they can be used to select equipment to satisfy known constant requirements. On the other hand, where requirements are poorly defined, and/or subject to radical change, no selection method, short of specifying a gross overbuy, can provide equipment guarantees to satisfy the system needs.

A possible out from this situation is to choose modular, expansible equipment that, in effect, allows one to delay making some decisions and/or to change some other decisions as conditions change. The ideal of modularity has been pursued with varying success over the years, of course, and has not proven to be the panacea many have claimed it would be. Nevertheless, these continued efforts have resulted in many types of equipment which can be conveniently reconfigured and expanded through modular organization at many levels.

3.2 DESCRIPTION OF SOME FLEXIBLE MODULAR MACHINES

In recent years, in response to pressures and requirements from a number of directions, the computer industry has developed new packaging techniques as well as new architectural concepts which have led to classes of very flexible, very modular computer configurations. The pressures and requirements are often conflicting, generally resulting in products that reflect compromises in design. Occasionally, however, it is possible to produce a design which is able to meet sets of conflicting requirements by being made flexible and adaptive from the beginning. Two such systems are described here.

The Burroughs Interpreter System

The Burroughs Interpreter, or D-machine, is a modular, microprogrammable computing system derived from an airborne multiprocessor designed for the U.S. Air Force.[2] The original concept called for a very compact, highly modular architecture which could be configured to do many different tasks with widely varying requirements. The Burroughs design was successful enough to warrant the development of commercial products based on its general concepts.

The Interpreter system is described elsewhere,[3,4] so only a summary of the basic design and the features of particular interest to the AEROSAT application will be given here.

Burroughs has developed multiprocessor systems for a number of years based on the concept of interconnecting processors, memories, and peripheral devices as modular units through a central switching network. This concept has been applied to the D825 military system as well as the B5500 and B6700 commercial computers. The Interpreter system is also based on the use of a switch-interlock device, but there are two important differences with prior designs.

The first of these is that the Interpreter switching system is modular and expandable in a number of ways rather than being designed to a particular size and configuration. The parameters which can be varied are the number of processor modules, the number of memory modules, the number of devices, and the width of each data and address path through the switch. The second difference is seen in the fact that there are only three types of units attached to the switch interlock: Interpreters, memories, and devices -- there are no I/O processors or channels in the system. The explanation lies in the special nature of the Interpreter itself.

The Interpreter module is a dynamically microprogrammed processor designed to be implemented modularly in varying word lengths. The basic module has registers eight bits long and these modules may be juxtaposed to obtain word lengths of any size in increments of eight bits.

The fact that the Interpreter is dynamically microprogrammed is important, however, because it is that which allows the system to dispense with specialized I/O processors. In fact, the Interpreter module can become any type of processor desired by merely changing the contents of the control memory. Thus, a system including, say four Interpreters could be configured at one moment as two I/O processors, one handling input and one output; a processor for selection of data from among inputs and formatting of output and a special processor for Fast Fourier Transforms on the selected data, say. During the next interval in time, the entire complex could be changed so as to become a set of identical cooperating processors embodying special capabilities for parallel-processing. Moments later, another change could be effected, and so on. In many applications, this flexibility could result in large performance gains over conventional architectures.

A particular advantage of the modular, multiprocessor system is its inherent redundancy which can be exploited to provide increased reliability. The Interpreter operating system has been designed such that all programs and data are replicated in separate memory modules, and routines are built in which cause redesignation of sources whenever a memory failure is detected. Processors are used independently and interchangeably, and are constantly being monitored for failure by the system. Thus, failures in memories or processors cannot disable the system. A similar redundancy in the switch interlock would provide a system with complete backup capability.

The Navy All-Application Digital Computer

Under the aegis of the Naval Air Systems Command, the Navy is developing a computer system for the 1975 to 1985 period which will -- as its name, the All-Application Digital Computer (AADC), implies -- serve as a standard computer system for all Navy computer installations. These range from airborne weapons directors to ground-based logistics systems. A great deal of modularity is being built into the system starting at the circuit level and including functional elements, such as standard adders, etc., on to complete processing elements, memories, etc. A brief description of the original plan (when it was known as the Advanced Avionic Digital Computer) is given by Entner.[5]

The basic concept is to develop a system architecture that will be capable of doing any kind of computing job by supplying processing modules which have special characteristics but common interfaces. Each module will be built of parts from a standard "library," packaged according to a single LSI philosophy. Basically, each processor will have a "task memory" associated with it to hold the programs it is assigned, while data will be obtained from random-access memory shared by all the processors. This will cut down on memory conflict since at least half of all memory accesses in the usual arrangement are to obtain program words. In the AADC, these words are in the task memory.

The task memories will be small (4096-word) fast memories which are, in turn, loaded from a special Block-Oriented Random Access Memory (BORAM). This new memory will provide rapid (2-μsec) access to the block it contains, and very rapid (150-μsec/word) transfer rate to the task memory. Thus, the change of program can be accomplished in a very few microseconds.

The cost of these systems to the Government is expected to be very low since they will be built in relatively large numbers from a few standard parts. Since programming of the systems will be done largely in high-order languages and since standard program modules are to be produced from which programs may be built, the software development costs are expected to be drastically lowered, also.

3.3 EVALUATION METHODS

There are a number of reasons for wanting to evaluate and/or measure the performance of computer systems, and for each set of conditions, there may be one or more techniques which give acceptable answers. It is the purpose of this section to describe a range of these reasons and techniques, so as to put the current effort in its proper perspective.

The distinction between measurement and evaluation is probably worth stating at this point. As used here, measurement will be the process of taking data during operation of a system, including operation: with simulated inputs and of a simulation model. The definition may at times be loosely extended to include analytical or numerical solutions of equations which serve as models of the system operation. On the other hand, evaluation is the process of using the measurement and other data in a disciplined way to estimate the performance of a system, or to match the performance of a system to a workload.

Reasons for Evaluation/Measurement

With Lucas,[6] we recognize three purposes for attempting to measure and evaluate the performance of a system. The first of these is in the design of new equipment and/or programs, the second is in choosing existing equipment to do a particular job, and the third is in attempting to optimize a given set of equipment and programs for a given job or set of jobs. Johnson, in a very scholarly article[7] despite the mundanity of the source, recognizes four "classes of measures," but his second and third; i.e., purchasing-roles measures and configuration measures, are special cases of our second purpose. The techniques that have developed may be classified according to whether or not they are applicable to these three purposes.

Let us consider each of these purposes in more detail. The design of new equipment and/or programs is a task familiar to most engineers -- the steps involved are the classic ones of establishment of requirements, delineation of alternative designs, analysis of tradeoff studies, and selection of final design. When the equipment to be designed is a general-purpose computer system, then the problems of measurement and evaluation are formidable, because the design goals are apt to be stated in terms of very generalized criteria, ones not tied to any particular application of the system. These criteria generally involve such things as cycle times, memory-access times, types of circuitry, number of registers, I/O rates, and other meaningful but elementary measures. Clearly, this is not the situation dealt with here.

The problem of optimizing a given system for a particular job, set of jobs, or expected mixture of jobs is one which has received a great deal of attention in the past few years. The primary reason is, of course, that it is a problem whose solution may result in a large, immediate payoff. Here, however, the basic form of the solution to the problem is known at the start, and the objective is to discover relatively minor changes which will improve the cost effectiveness of the system. These changes take two forms: removal of unused -- or underused -- components, such as extra memory, I/O channels, or even CPU's, and addition of critical components which can break bottlenecks. Again, this is not a problem which concerns us directly.

Inbetween these two lies the area of our interest. Note that in the first case, neither the hardware/software nor the job to be performed was known; here the search is for measures of performance that can be applied generally, to represent wide ranges of job requirements. In the second case, both the hardware/software and the job characteristics are relatively known; the object is to take meaningful measurements which can be used to make precise statements about the performance of the system.

When the job is fairly defined and the hardware/software are to be chosen, the task is beyond generalized performance measures, such as MIPS (million instructions per second), and yet not amenable to direct measurement, as with hardware monitors. There are techniques which can be used, and have been used with varying success; these will be discussed in the next section.

Methods for Measurement and Evaluation

o Measurement

There are basically two methods to measure system performance: by means of hardware and software. To measure by means of hardware, one attaches to the necessarily pre-existing computer system a special device equipped to monitor, time, and record selection events which occur during system operation. To measure by means of software, one includes within the programs' operating on the system a set of routines, called "artifacts," which cause data to be recorded by the system when selected events occur.

Each of these techniques can provide useful data, but neither is entirely without disadvantage. Hardware measurement has the advantage that it does not interfere with the operation of the system being measured, it itself being essentially outside that system. It has the disadvantages, however, of inflexibility and cost; the hardware must be purchased for a particular use and may or may not be adaptable for other measurements than those for which it was designed, or for other computer systems, for that matter. Software measurement suffers from the reverse situation -- it is relatively inexpensive and flexible in implementation, but its use has effects on the system being measured. This point will be discussed in following paragraphs; suffice it here to say that the assessment of the effect of the measurement on the process is an important part of any analysis of software-measurement data.

o Evaluation

There are two main techniques for evaluating a computer system: first, by modeling its behavior, making some kind of measurements and drawing conclusions from them, and second, by operating a system with a hardware configuration, program set and data set representing the real system in some sense, taking measurements, and drawing conclusions from them. That there is no clear line between two techniques merely reinforces the contention of many that the techniques can be viewed as a kind of continuum, ranging from a completely analytical solution at one end to a complete implementation and measurement at the other.

Consider first the modeling techniques. System modeling is based on the premise that the input data sets, the output data sets, and the processes carried out by the system can be understood and can be described by representations (models) of them that can be more easily manipulated than the sets and the processes themselves. Clearly, for systems of real interest, this premise is only approximately true because either the systems are too complex to be completely understood or models which can adequately represent them are too complex to be usable. Within limits, then, modeling techniques can be used to predict the performance of computer systems; the accuracy of the prediction will depend in large measure on the accuracy with which the input and output data sets and are represented.

An equally important consideration in modeling studies is the selection of measures of performance. The objectives of system operation are supplied from outside of the system itself, by the milieu in which it operates. A management information system requires quick response and the ability to handle complex requests on a complex data base; a multiprogrammed batch operation needs maximum throughput with rapid turnaround times; a real-time control system needs, as discussed in Section 1, guaranteed operation within the constraints of real time, and minimum load and functional capacity requirements.

Whatever the performance measure selected, the system model must provide for its evaluation; however, the system model need usually be only complex enough to account for the first-order effects on that performance measure. Only when a system requires critical fine-tuning are the second-order effects worth pursuing.

Thus, system modeling is the process of selecting a performance measure, choosing a representation of the process, such that it adequately represents the process and still can lead to a solution or set of solutions.

The classical models have been sets of equations which represent the interactions between the variables of the system. Nearly always, these equations have been too difficult to solve unless they are linear or can be replaced by a linear approximation -- "linearized" -- but the use of various numerical techniques in conjunction with increasingly more powerful digital computers has led to more solutions of difficult non-linear systems. The increasing capacity of computers has also allowed the modeling of systems with larger numbers of variables. One of the largest models currently being used is the ocean and atmospheric model at the Geophysical Fluid Dynamics Laboratory, which solves sets of equations linking pressures, temperatures, etc., over the whole earth at points on a 100-mile grid. With the newer computers, this grid size will be reduced to 50 miles on a side.

On the more mundane, and germaine, side, an example of a computer system model is given in the work of Kleindorfer and Kriebel[8], where in a set of 12 equations and inequalities, job-completion rates, turnaround times, job classes, and their delays, and resource utilization rates are related. In addition, they introduce from the world outside the system the condition that the controller variable is the schedule of jobs to be entered into the system or what is the same thing, the priority scheme to be applied to jobs entering the input queue. Finally, they describe a class of performance measures, "...scalar valued function(s), g, of the mean flow times $\overline{F}_j$ of class j jobs, g $(\overline{F}_1, \overline{F}_2, \ldots, \overline{F}_J)$," which can be evaluated on the basis of assumptions about the applied job mix. Thus, the model meets all the criteria described above and is a viable tool for system analysis.

A technique other than actual solution of equations is available for evaluating performance measures, the so-called Monte Carlo method. With this method, neither a closed solution nor a numerical one is sought, but rather a statistical approach is used. The values of the variables are expressed in probabilistic terms, and the equations are written so as to express the relations between them in statistical terms. The success of the technique depends on making so many solutions of the system using sets of random inputs that the results achieve statistical significance. Because large amounts of computation are required, digital computers are used to obtain the solutions; and because computers have increased in power, it has been possible to program more and more complex problems.

An alternative to analytical studies of computer systems is the use of simulation, wherein a model of the system is constructed and "operated," in some fashion, to obtain measurement data. The models of systems used in simulation studies are often not expressed merely as sets of equations, but are frequently composed of collections of computers, computer programs, and selected peripheral equipment. Both Monte Carlo and the usual techniques of measurement while varying parameters in a controlled way are used to obtain the data.

At one extreme, the dividing line between what is simulation and what is an analytic study is very hazy; at the other extreme, the line between simulation and actual operation of the system is also not very clear. One way to evaluate a system would be to install and run it in the actual operational situation -- a technique which is rarely feasible. In lieu of such an extreme and expensive solution, various techniques have been devised which substitute for the actual operational situation one or another simpler tasks which are, in some sense, representative of the load which the system will be required to carry. Kernels, which are, in effect, short, frequently used subroutines characteristic of the expected system usage, and benchmarks, which are complete application programs using standard input data, are two approaches which have been widely used. An approach between these two is the "synthetic job," as devised by Bucholz.[9] This is much more stringent a test than the usual set of kernels because it is a complete process, transforming the input data in a number of ways to produce a prescribed output. At the same time, it is less cumbersome to program than the ordinary benchmark since many extraneous details have been left out.

Thus, we have covered a spectrum of system-evaluation techniques ranging from purely analytical through various other modeling techniques to a complete implementation of the system. In the next section, we will develop a system-evaluation methodology which borrows from many of the techniques mentioned above, and which is here recommended as an analysis-and-evaluation tool in developing the AEROSAT-surveillance computer system.

The "Template" Approach

Each of the evaluation techniques outlined above has merit and can give useful answers under certain circumstances. However, none of them are quite what is needed here. To repeat, we need a method by which computer configurations which already exist in some form or other may be evaluated for use in a system whose general characteristics are known but whose size, complexity, and extent are still in question. The technique chosen should be relatively easy to program, should provide measurement data in standardized, machine-readable form and should be adaptable to all of the configurations under consideration.

The approach finally developed was an electric one which combined features from the kernel, synthetic job, and benchmark techniques. Basically, the idea is this: the operation of the computer subsystem of the system under design is viewed as a set of transformations applied to a given set of input data streams so as to produce a given set of output data streams. If this is the case, then it should be possible to example the transformations involved and select that subset which is representative of the whole; and then, to simplify them by eliminating excessive detail such as seldom-followed conditional paths, elaborate error-recovery routines, paths which duplicate other paths except in relatively minor ways, etc. The new system composed of a possibly simplified input stream, the simplified transformations, and possibly simplified output stream is what we here call a "template" of the original system.

An important and necessary part of the template technique is the set of performance measurements that must be made during the system operation. One is interested in two basic kinds of measures: a "horizontal" or throughput measure, and a "vertical" or resource-allocation measure. In the template method, the first measures are taken by designating "monitor" objects, whose progress through the system is recorded at every stage. These objects might be orders in an inventory-control system, tracks in an ATC system, or interrogations in an information-retrieval system. Each time one of these monitor objects is handled by the system, its identification, the time, and the function performed are recorded for later analysis. Measurements of the second kind are provided for by requiring each functional program and each interrupt-handling routine in the system to record its identifier and the clock reading whenever it is entered and exited.

o Specification of Input Stream

The input stream is developed by hypothesizing the characteristics of a "typical" data-acquisition system -- one which is actually in operation and can easily be simulated, or one which might represent future developments in the field. It is essential to ensure that the input stream has the following two characteristics: (1) it should be accurately and precisely defined in all respects, and (2) it should be defined such that a single parameter can be varied so as to increase the overall data rate.

The best way to implement the input stream for a series of template evaluations would be to produce a set of data on a compatible medium (such as magnetic tape) by means of a scenariogenerator program run on any available computer. Copies of these data for selected values of the loading parameters would then be used at each of the participating installations. In this way, there would be no question about the lack of compatibility because of non-uniform loading, and complete and accurate information would be available about the actual input data.

o Specification of the Output Stream

In a similar way, the output stream is developed by assuming a display or other output subsystem that has characteristics typical of systems which might be used today or in the near future. The output stream must also be precisely defined in extent; that is, the information to be generated, but may be left somewhat vague in form, or table structure. This will ensure that all implementations of the template do the same tasks, and that the system designers are left free to take advantage of any special output features of a particular machine.

One part of the evaluation data to be recorded for later processing is a set of periodic samplings of the output stream. Enough data should be recorded to enable the analyst to verify system performance, through use of a display program or other device, but not so much as to become a burden on the system being evaluated (or so much as to swamp the analyst). The main performance-measurement data are of a different nature, and they are discussed below.

o Specification of Functions to Be Performed

The heart of the template specification will naturally be in the set of functions to be performed. The success and creditability of the method will depend on the choice of functions and on the way in which the functions are specified to the implementor.

There are four principal objectives to be met in the design:

a) The system must be internally consistent, so that when programmed, it can be successfully "operated," or cycled.

b) It must bear a strong functional resemblance to real systems of the type being modeled, so that one may have an intuitive feeling that the resulting measurements had meaning.

c) It must be simple in structure and in detail, so that it may be easily programmed.

d) It must be complex enough to load the computer configurations in the same way and at the same levels as the real systems might.

Note that the last two objectives are somewhat antithetical in nature; this will naturally lead to compromises in the design which must be made carefully so that a fair balance between the two may be obtained. It should be emphasized that the system as specified is not meant actually to perform any real work. The functions implemented are meant to be representative of the processing done by real systems but not to perform useful tasks, necessarily. All this is of no consequence to the purpose of the template implementations which are meant merely to evaluate the performances of the various computer systems considered.

The selection of the functions, as well as the level of detail to be included, is a matter requiring careful attention. The processing selected must be of the same nature as the processing in the real system, yet must somehow be simplified from a programming point of view. One obvious way to do this is to eliminate special cases, error detection and correction, conditional branches where the probability of taking one branch is low, etc. A second way is to eliminate steps in the functions which are similar in processing requirements to steps already included.

A special function, called the "executive job," is added to the list of tasks to be accomplished by those systems which have multiple, independent computing elements. This standard "make-work" job is added, so that a measurement may be made of the computing power potential of the system not used by the template system.

3.4 DESIGN OF REPRESENTATIVE AEROSAT SYSTEM AND ITS TEMPLATE

Although the AEROSAT specifications are far from definitive, it is possible to describe the necessary functions and characteristics of the surveillance and communications subsystems in enough detail to get a good picture of its operation. We will present a description of a relatively unsophisticated system to do the surveillance function, including range measurement, position determination, and related communications tasks. This system will then be abstracted to form a "template" system which could be implemented to evaluate candidate computer subsystems.

System Description

The AEROSAT-surveillance system chosen as a sample system for this discussion is presented with no pretense as to its completeness or engineering, financial, or political suitability. Rather, it is a vehicle about which discussion may be generated.

We start with the assumption that aircraft entering the system do so at the eastern and western extremities of a corridor stretching across the Atlantic. This corridor is of known dimension and contains a known number of discrete tracks to which aircraft may be assigned for passage in either direction. The distribution of the aircraft within the system, the velocities and separations, the algorithm used to assign aircraft to tracks, and variations in loads with season, month, or day of the week are all incidental to the operation of the surveillance system. We will be concerned with such matters as peak instantaneous load, peak daily load, average daily load, and the like.

Within that context, the task of the surveillance system is: (1) to accept messages from traffic-control centers concerning assignment to aircraft to tracks including identification, time of entry, entry point, track number, entry velocity, and entry altitude; (2) to maintain a file entry for each aircraft in the system including identification, track number, velocity, altitude, and position; (3) to interrogate each aircraft in the system periodically according to some specified algorighm; (4) to receive and time tag messages from each aircraft relayed via the two satellites of the system; (5) to convert the transmission times recorded to range measurements and to compute from these measurements the position of each aircraft; (6) to record these positions in the track file; (7) to prepare and transmit to the control center periodic messages giving the identification and position of each aircraft in the system, and (8) to carry out whatever communications functions are implied by the above but not stated explicitly.

We assume that accurate and timely values of satellite parameters are supplied to the surveillance system by an outside source; it is also assumed that calibration (and update) of system parameters is a seldomly performed task relegated to non-busy time, and hence, need not be considered in this analysis.

A Template of the System

The description of the template given below will be sufficient to illustrate its extent and the procedures required to specify it, but will not be a complete set of specifications. A block diagram of the template is given on the following page.

The first step in the specification of the template is to describe precisely the interfaces between the computer system and the (idealized) outside world. In this case, there are two interfaces -- to the satellite subsystem, and to the control center -- with an input to, and an output from, each. At the satellite end, the computer system produces aircraft interrogation messages and receives aircraft position replies in return. At the control-center end, flight plans are received, and aircraft-position messages are generated by the system.

In the real system, the flight-plan messages correspond to actual aircraft entering the system, and the aircraft-position replies correspond to responses of actual aircraft to interrogations by the system -- there is a one-to-one relationship between the interrogations and the replies. In the template, the flight-plan messages are simulated entries of aircraft, and position replies are simulated responses to interrogations but without the

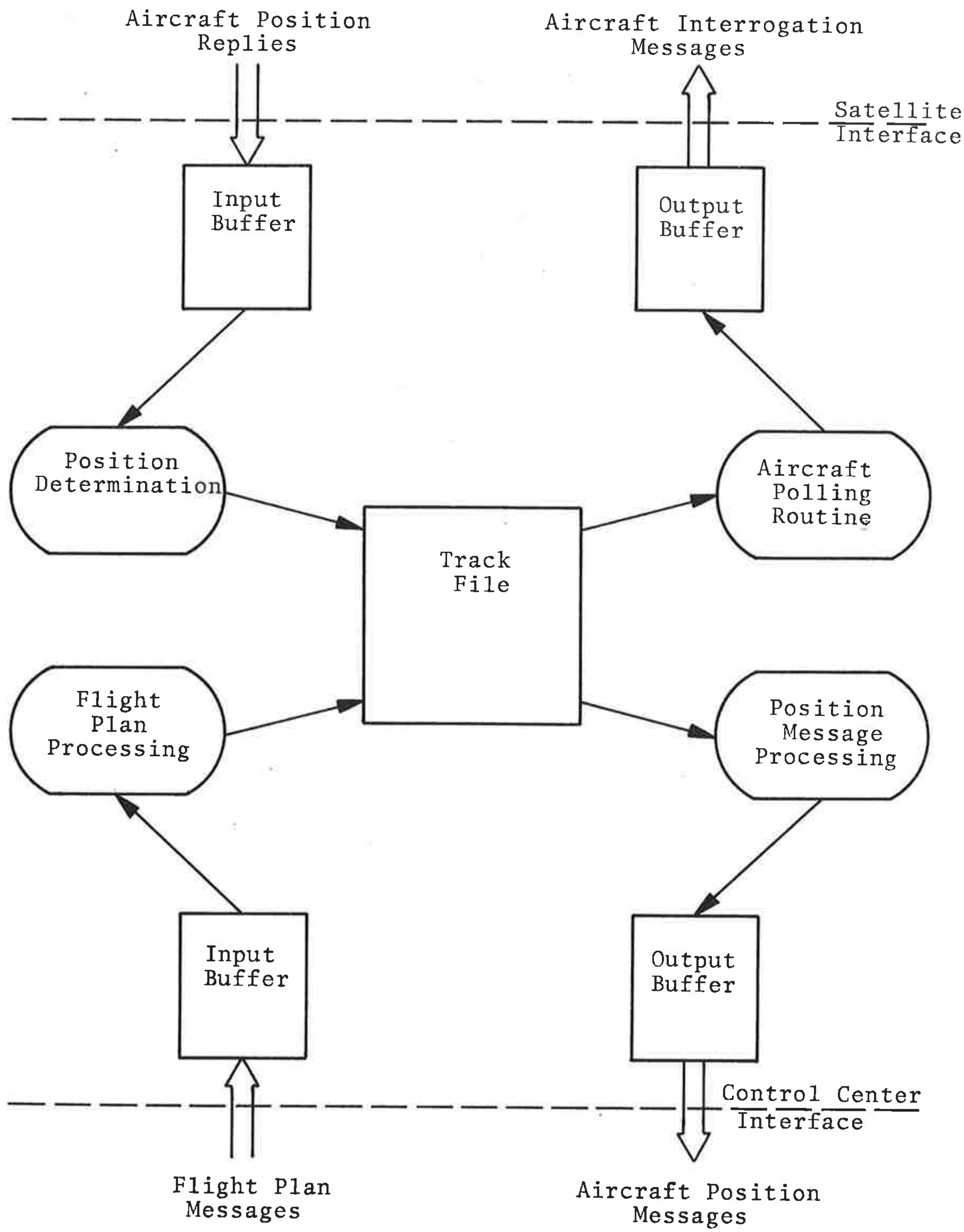


Figure: Block Diagram of Template for AEROSAT Surveillance System one-to-one relation of the real system. In other words, the coupling between interrogations and replies is broken, so that aircraft-position replies may be generated a priori and not depend on the actual running of the template system.

This is not a disadvantage: the system need not actually track aircraft as long as it does the same amount and type of processing that the real system would do. We are measuring processing performance not tracking accuracy.

The input streams, therefore, will be supplied, such that the system can establish track entries in its track file and update them systematically. At the same time, it will be generating aircraft interrogations according to some specified criterion and position messages at some specified period.

Four functions are to be performed by the template system: flight-plan-processing, position determination, aircraft-polling, and position-message-processing. Each of these functions requires the computer system to perform in a different way, spanning the set of processing requirements of the AEROSAT system. The first function, flight-plan-processing, requires string-and character-processing, code conversion-and-formatting, and file maintenance. Position determination involves file-searching and mathematical calculation, while aircraft-polling involves real-time-scheduling, file-searching, and message preparation. Finally, position-message-processing requires code conversion-and-formatting, message preparation, and communications-procedure initiation.

The output streams to be generated by the system are the set of aircraft-interrogation messages and the set of aircraft-position messages. They will be recorded on magnetic tape periodically for later evaluation. These data are used to get a rough verification that the system is actually operating in the correct manner.

The main measurement data to be supplied are of two types: first, every entry to, and exit from, the routines which perform the four processing tasks described above will be recorded on magnetic tape in a specified format; and second, every operation performed on a certain set of specially marked tracks (monitor tracks) will also be recorded on the measurement magnetic tape. These measurement data will be evaluated by means of standard computer programs which can then make direct comparisons among the candidate computer systems.

3.5 PROBLEM-STUDY AREAS

Part of the cost to the user of any computer system is the system-and-application programming which must be available before system operation can begin. For the AEROSAT system, a real-time operation, system, or executive, with error diagnostic and recovery capabilities must be available. Executive programs with these abilities are being, or have been, written for the Enroute (NAS) and Terminal (ARTS) air-traffic-control systems. In addition, the functions to be performed by the AEROSAT system will have to be specified in great detail and coding produced. The ease with which these programs can be supplied will depend in large part on the programming languages and compilers that are implemented on the system chosen.

Operating Systems

Although the science, or art, of operating-system design has matured in the last several years to the point where the construction of workable systems for standard machines is a straightforward process, there remain problems and questions when the computer system is non-standard in one or more of several senses. Not fully explored are systems involving considerable parallelism, those which can be changed dynamically and those which have highly modular construction. It is not to say that operating systems have not been, or could not be, designed for such systems, but the assurance of near optimal performance is not possible at this time.

Research and development are being carried out in these areas (e.g., three, and it may be expected that quite capable and efficient operating systems will be available for the AEROSAT-surveillance system.

Languages and Compilers

At the same time, a great deal of effort has been expended in the development of languages and their compilers, which both cater to the needs of the programmer in making it easier to express his wishes for the program, and also, exploit the special characteristics of the machine being programmed for. The AADC program, in particular, has stressed the necessity to unify the design of the computer and of the compiler, so that they best operate with languages which are oriented toward the mission for which the system is being built. The AEROSAT program will benefit from research into new applications-oriented languages in programs, such as AADC, if a close eye is kept on their progress.

3.6 DESIGN TRADEOFFS

It is usually possible in the design of a system to meet all of the performance criteria with a number of different, alternative plans. It is then frequently possible to find a single criterion, or a small set of criteria, such as cost or overall size or weight, etc., by which to judge the various designs and choose the most nearly optimal one. Sometimes, however, there are conflicting performance requirements or operating characteristics which require a choice to be made in the specifications. The best configuration then is derived as a result of tradeoff studies with respect to the requirements and characteristics.

For the AEROSAT-surveillance system, there will be no intensive requirements as to size, weight, power required, cooling, etc., since this will be a ground-based system. The major considerations will be reliability and the capacity to do the job, as discussed in earlier sections, and cost.

We can illustrate many of the tradeoffs best by describing them in terms of a single machine, say, the Burroughs Interpreter, described above.

In most modern computers, the speed with which a basic operation may be carried out depends most on the access time of the memory used. Various devices, such as look-ahead, cache memories, pipelining, etc., are used to lessen the dependence of instruction timing on the single memory-access time. There is necessarily some relation, however; and in general, faster memories mean faster operation, but also higher costs. Here is a first point of compromise: memory cost versus speed.

In terms of the Interpreter system, there are two memories to consider -- main memory, and control memory. Access to main memory is essentially asynchronous and for a single word transfer. Thus, there is a strong relation between system operation and memory-cycle time. Because of the microprogrammed aspect of the design, however, the relation is not linear with operations since macro-instructions can be coded which drastically reduce the number of memory accesses per operation. The speed of the control memories affects overall machine speed more directly; hence, money invested there in faster components pays off handsomely.

The size of control memory also affects the speed of the machine indirectly since with more memory, more microcode can be stored and fewer changes of microprogram would be required for a given mixture of tasks.

The word length in a computer is also a factor in the performance of the machine, being reflected in the operation-code size, the addressing structure, and the precision with which computations can be carried out. It is clearly a factor in the cost of the machine since it is reflected in the size of registers in the CPU as well as the size of the memory. Thus, a second point of compromise is word length versus performance and cost.

It is interesting to note that a modular system such as the Interpreter can be structured with word lengths varying from one byte of eight bits to any practical length in units of a byte. This means that within a single family of machines, the word length can be tailored specifically to the use of the system.

Finally, the numbers of modules of memory, processor, I/O, or other hardware that are purchased has a direct effect on system performance, reliability, and cost. The correct number must be supplied to provide all the computing power necessary to do the job, and also, enough to ensure reliability through redundancy. Here, there is an interaction with system speed, also, since computation power could be increased by increasing either the speed of a fixed number of CPU's or increasing the number of CPU's at a given speed.

The number of processor modules in an Interpreter system, to continue our example, would have to be large enough to carry out all of the functions desired in the system plus one since the function of the processor can be selected by loading the proper microprogram into its memory.

Many of the same considerations apply to other computer systems such as the AADC, which are built in modular fashion. Clearly, the greater flexibility that is incorporated into a computer system, the easier it becomes to tailor that system to the functional and cost requirements of the overall system design.

4. OCEANIC AIR TRAFFIC CONTROL OVERVIEW

4.1 INTRODUCTION

The use of satellites to provide communications and surveillance in an ATC system has been established as the objective of the AEROSAT project. Particular interest is building in the concept of a satellite-based system for oceanic ATC. The oceanic area is especially attractive for initial implementation for AEROSAT application because of the relative simplicity of the control environment compared to the domestic one. In the North Atlantic principal area for example, air-traffic flow is essentially unidirectional during peak busy hours. The traffic, composed mainly of subsonic air-carrier jets having similar performance characteristics, navigational equipments, and flight-crew proficiency, is routed on parallel tracks with various flight levels. Thus, the oceanic area could provide an operational test bed for future domestic systems. In fact, in oceanic control, large separation minimums severely restrict system capacity because of the fact that present techniques of communications, position determination, and flight-progress-monitoring are necessarily cumbersome and inefficient. The use of relayed voice link is non-real time (approximately 10 to 35 minutes delay) and unreliable. Hence, in oceanic control, the need for increased traffic-handling capability, for simplifying the control process, and for improving safety are most urgent. The application of AEROSAT could provide ideal answers for the upgrading of the present oceanic ATC system.

Once the decision is made to experiment with the AEROSAT system in an oceanic ATC environment, it is mandatory to understand the functional characteristics of both the AEROSAT system and the Oceanic Automation Program to achieve smooth interface at system-integration time. The objective of this section is to report the study results of the present manual oceanic ATC system, the experimental system using data link, and the preliminary Oceanic ATC Automation Plan by the planning group of the FAA. The knowledge gained in this investigation will serve as a technical base for the design of the interface of the AEROSAT ground stations.

4.2 MANUAL OCEANIC AIR-TRAFFIC-CONTROL TECHNIQUES

To have a quick grasp of, and to get acquainted with, the manual oceanic ATC control techniques, we visited the Atlantic Oceanic ATC operation which is a part of the New York ARTCC located at MacArthur Airport in Ronkonkoma, Long Island, New York. Although there are minor variations among oceanic ATC centers, the control process and operation procedure, for the most part, used in the New York Oceanic ATC zone are typical and applicable to all oceanic ATC centers.

The oceanic control sector of NYARTCC has direct interfaces with the Gander, Azores, Bermuda, San Juan, Jacksonville, and Miami control centers. The New York oceanic sector controls all air traffic between the United States and the Azores, the United States and Bermuda, New York and the Bahamas, and between the West Indies and Puerto Rico all in fixed air routes (tracks) at fixed flight levels. For the most part, the United States to Europe air traffic follows the Northern routes, and hence, is controlled by Gander. However, because of weather conditions, some traffic may be re-routed southward, which falls under the jurisdiction of the New York oceanic control sector. Recently, more flights have been scheduled between Europe and South America, and between the United States and Africa, which increases the traffic load in this region considerably.

During 1969, the highest traffic load (100,805) was experienced with the peak (9900) reached in January. Traffic tapered off in 1970 and 1971. The traffic count in early 1972 shows an upswing again; April traffic (10,881) broke the all-time high of aircraft. When the totals are in, the 1972 number of aircraft could reach 120,000; with a daily peak of 581, and a daily average of 360.

At present, the oceanic ATC control follows the identical procedure as the continental ARTCC, using radar/beacon returns displayed at Plan Position Indicator (PPI) scope and manually tracked by "shrimp-boat" techniques. Ocean-bound aircraft are followed within 180 nautical miles from the Kennedy International Airport along fixed air "corridors." The Air Defense Command (ADC) will be alerted if any aircraft is found outside of those corridors. Tracking of aircraft outside of the 180-nm zone is accomplished by enforcing two general reporting rules. Slow aircraft are required to report at every hour, and high-speed jets are required to report every 10-degree longitude flying East-West routes, or every 5-degree latitude flying North-South routes through voice channels. Tracking of aircraft traveling North Atlantic routes follows regular domestic ARTCC operations which interface with the Gander ATC center before it becomes oceanbound.

System Description

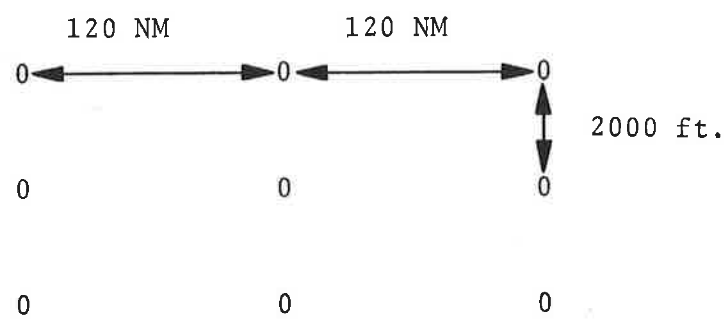
o Standard Separations

Lateral = Basic minimum is 120 nm for the East-West flights and 90 nm for the North-South flights.

Longitudinal = Basic minimum is 30 minutes. Southern routes may have the minimum separation reduced to 15 minutes. Further reduction is possible and is based on mach speed differential between preceding and following aircraft.

Vertical = 2000 feet.

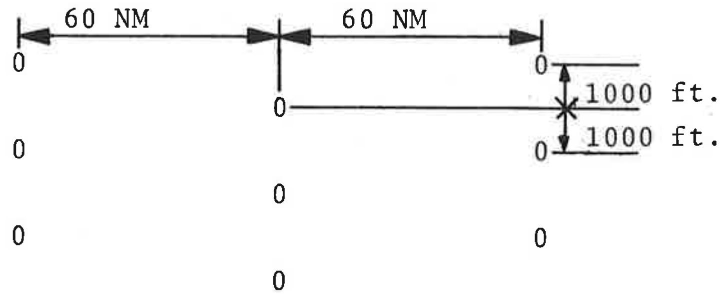
The pattern of the standard separation can be represented as follows:



o Composite Separation

For turbojet aircraft operating at or above flight level 290 within the organized North Atlantic control region, composite separation standards may be applied.

The pattern is shown as follows:



o Establishment of Tracks

To permit the optimal use of the airspace, tracks of minimum time are established manually in the Gander and Shanwick control centers making use of United States and United Kingdom meteorological forecasts, respectively. The two centers, with proper coordination of the New York and Santa Maria oceanic control centers when necessary, negotiate via telephone communication before reaching agreement on an organized track-and flight-level system.

It should be observed that, in practice, the organizers are inclined to give more weight to the desires of the airlines' requesting specific tracks than the meteorological office. Also, it is the oceanic control center with the dominant traffic flow which imposes its choice of track organization.

When the negotiation has been completed, the coordinates of tracks and flight levels are disseminated to other control centers and airlines. Such information is disseminated three hours in advance of each anticipated peak traffic period.

o Flight Plan and Clearance

Based on an established set of tracks and flight levels, the airlines can proceed to establish flight plans to suit optimum flight schedules. The flight plans must be sent, at least one hour before the departure of the aircraft, to all the agencies concerned and to all airports enroute. Upon receipt of the flight plan at oceanic control center (OCC), a flight strip is manually prepared and passed on to the clearance controller who arranges the strips according to the specified tracks and levels, and the estimated hour of arrival. The strip is then in a "non-active" state.

When the aircraft is about to enter the OCC's control zone, the pilot takes initiative in making contact with the center's radio operator, reporting his present position and the estimated hour of arrival in the oceanic zone. The "demand for clearance" is transmitted to the clearance controller, and the strip is then "active." The clearance controller can then see whether the request for clearance is acceptable or not by making a rapid comparison with the strips on his panel. If the request is not acceptable, he decides on one of three recommendations: (1) a different flight level of the same track, (2) a change of track, or (3) a change of time of entry. Whether accepted or not, the radio operator passes the corresponding message to the pilot. If the pilot does not agree, there is an exchange with the clearance controller until an agreement is reached.

When the pilot has accepted his clearance, the controller warns the domestic center presently controlling the aircraft of the clearance issued. It is the responsibility of the domestic control to guide the aircraft to the track, at the flight level, to the speed and time required. The aircraft then begins its oceanic flight and files reports at every fix or reference point, including time of crossing, altitude, mach number, and time estimated to the next reference point.

When the aircraft is about to leave the oceanic control zone, it follows the identical but reversed procedure, so that the control of the aircraft is handed off to a domestic control center for the remainder of its voyage.

4.3 OAKLAND ARTCC--PACIFIC OCEANIC AIR-TRAFFIC-CONTROL ZONE

General

The Pacific Oceanic ATC zone covers about 6.5-million square miles. This center has direct interface with Seattle, Vancouver, and Anchorage ARTCC's in the North, Los Angeles and Tahiti in the South; and Honolulu ARTCC in the West. In general, there are two patterns of air traffic, fixed and random. There are four fixed routes between San Francisco, Los Angeles, and Honolulu (North, Alpha, Bravo, and South routes). Random flight routes appear mostly in the northern regions among Honolulu, Anchorage, and Vancouver/Seattle areas. Southbound traffic to Tahiti is very light. Honolulu traffic to and from San Francisco use the North and Alpha routes, whereas Honolulu traffic to and from Los Angeles use the lower two routes; namely Bravo and South.

The control sectorization is first made in altitude. One sector controller takes care of all the low-altitude traffic defined as 27,000 feet and below. Three more sectors are organized above 27,000 feet. The North sector covers all random traffic routes. South Sector No. 1 takes care of fixed routes, North and Alpha; whereas South Sector No. 2 handles Bravo and South air routes.

The ATC concept is similar to other oceanic ATC centers in which separation is maintained at 15 minutes in trailing, 20 minutes in crossing traffic, and 100 miles between centers of air routes. Upon receipt of traffic from terminal controllers, the oceanic radar controller is responsible for transforming the 5-mile separation minimum into time and space criteria and handing them off to sector controllers. Once traffic is out of radar's reach, it is required to report position updates at every 10 degree longitude via extended VHF links to Aeronautical Radio, Incorporated (ARINC).

The Experimental System

o System Configuration

In 1970, a joint Government-industry-sponsored program was initiated to demonstrate the feasibility of enhancing the safety and efficiency of oceanic ATC by introducing automatic position-reporting through data link. Participants included the FAA, ARINC, and interested international air carriers. Oakland ARTCC was chosen as the experimental site because of the proximity of ARINC ground facility at nearby Foster City, California.

Principal elements of the experimental system include the avionics subsystem, the data-link communication subsystem, and the processing-and-display subsystem. The avionics subsystem consists of an ARINC-qualified inertial navigation system with digital output of latitude and longitude. The system is capable of being coupled into airborne data-link terminals. The data-link subsystem consists of airborne and ground terminals by Conductron and Bendix Corporation through the use of the extended-range VHF at the ARINC ground terminal. Connected to the ARINC communication center by a 2400-baud data-transmission line, the processing-and-display subsystem has the following four major elements:

a. A computer with 32,000 16-bit words for data storage-and-processing control of digital displays.

b. Two digital CRT display consoles--one used as a situation display; the other as a tabular display.

c. A function-keyboard and light-pen combination that allows the operator to select and modify the displays.

d. A teletype station for local data entry and production of hard copy.

o Functional Description

A flight plan for each airplane scheduled to operate in the Oakland center's airspace is filed in the computer memory by Teletype entry at the center. The plan includes airplane identification and type, proposed speed, requested altitude, departure point and proposed time of departure, and route. When the airplane actually enters center airspace, either from the ground or by flying in, an "activate" message will enable the system to show its location, identity, and flight progress on the displays.

Data link-equipped aircraft will transmit continuously updated-position information derived from on-board inertial navigation and air-data systems. Positions will be updated on the displays automatically every 30 seconds. The position of aircraft not equipped with data link will be updated every 120 seconds by extrapolation from the last-known position, as reported by voice message. Flight-progress reports via voice messages will be entered into the computer memory through the Teletype.

The function keyboard will be used to specify major options for the displays, and the light pen to modify a display or select options at a secondary level. The two devices will be used in complementary fashion. For instance, a controller might call for a listing of geographic and background-mapping options for the situation display through keyboard action, then specify a particular combination of display features through light-pen action.

The major options for the situation display, and the variable features that can be specified for each, are as follows:

1) Geographic coverage. This allows the controller to display a "combined" sector of nearly 2-million square miles, or overlapping north or south sectors of approximately one-half of that size. If the controller wishes to narrow his field of vision, or shift it within the basic sectors, he can use zoom or translating window options.

   When a keyboard pushbutton for a situation display is activated, the subsidiary options appear as a "menu" in tabular form at the right-hand side of the display tube. The operator selects from the menu with the light pen--first, in the case of the geographic display, the sector he wishes to view and, subsequently, to change the scale of the display and shift the field of vision. Two tabulations can be displayed simultaneously.

2) Background mapping. Using the light pen, the controller can add or delete features, such as a latitude/longitude grid, sector boundaries, route lines, and fix locations and identifiers, or

brighten or dim individual features at his convenience.

3) <u>Control of track formats</u>. Aircraft symbols and their associated data blocks can be modified and shifted on the display to reduce congestion and visual interference. The symbology can be presented in four different sizes, or deleted entirely. Data blocks can be shifted to any one of eight compass points around the airplane marker.

4) <u>Velocity vector control</u>. A line attached to each airplane track symbol, indicating the direction of flight, can be varied to show the airplane's predicted position in 5, 10, 20, 30, or 40 minutes.

5) <u>Situation display filtering</u>. Permits the controller to eliminate all airplane symbols on the display except those on a route or at a flight level of interest.

Blinking symbology will appear on the display to alert the controller to any of the following situations:

a) An airplane is more than 15 minutes late in reporting at its next intended fix.

b) Cases of potential conflict, defined as aircraft separation of less than 1,000 feet vertically, 100 nm laterally, or 20 minutes longitudinally. The controller can query the computer to determine the expected time of arrival of each aircraft at the intersection of the two tracks.

c) Twenty minutes before an airplane leaves a sector.

d) When a progress report voice message is received from a non-data link-equipped airplane.

When a controller has been alerted to a potential conflict, he can investigate the effects of modifying the flight plan of one or both involved aircraft with an alternate flight-plan-probing routine. If a proposed flight-plan alteration will not resolve the conflict, the proposal is negated in the computer, and a new plan investigated. When the computer determines that a plan will resolve the conflict, the word "clear" is displayed, and the amended flight plan may be adopted.

The principal tabular displays are designed to complement the situation displays and furnish details that cannot conveniently be shown on a situation display. Principal tabular displays are:

1) Flight-plan data. Depressing a function key causes a tabulation of flight plans stored in the computer memory to appear on the cathode-ray tube. The controller can then use a light pen to call for the details of the plan that he wishes to investigate or amend.

2) Flight-progress data. Enables the controller to obtain the last fix of all on selected flights in a given sector, using route or flight level filtering if desired. Data for random-route flight are presented in a separate section of the tabulation, with last reported position substituted for last fix, and time at last reported position for actual time of arrival at last fix.

3) Fix posting data. Provides a summary of traffic at a selected fix, with flight level filtering available if desired. First, a list of 12 fixes on the four West Coast-Hawaii routes is obtained through keyboard action; then, the controller selects the fix of interest by light-pen action.

Alerts and warnings are indicated automatically on the tabular display by blinking messages, similar to those used on the situation display. There are alerts for potential conflicts, overdue aircraft, and handoffs, but not for progress-report voice messages.

The display system is capable of handling approximately 30 aircraft simultaneously, considered enough for one sector.

4.4 OCEANIC AUTOMATION DEVELOPMENT PROGRAM

On the whole, the Oakland experimental system is considered successful. Controllers who have exercised on the system liked it although some of the features provided by the system may be cumbersome or unnecessary. Enough interest in the FAA has been generated to upgrade the oceanic ATC process by bringing an optimal level of automation into the system. Stimulated by the AEROSAT project, an Oceanic ATC Automation Planning group was formed within the FAA whose members include OSEM, SRDS, ATS, NAFEC, and MITRE, to establish a technical development plan for the Oceanic ATC Automation Program.

After a good number of discussions and review meetings, the planning group has completed its first draft of the development plan. Since it is a preliminary draft, only the functional aspects of the automation system will be reported here to reflect a general direction that this program is likely to head toward.

In brief, the program has two development phases: Phase I provides the basic automation level. Phase II adds more automatic checking, probing, and extrapolation functions, and provides the capability of interfacing and use of AEROSAT for carrying out all ATC automation functions in phase I. These functional requirements will now be described briefly.

Phase I - The Basic System

The following functions in Phase I comprise the basic oceanic automation level:

- Flight-Data-processing,
- Flight-Data-recording,
- Tracking-and Control-processing,
- Display-Function-processing,
- Interfacility Communication,
- Supervisory Functions.

Flight-data-processing is defined as all the computer-processing necessary for flight plan and flight-plan-associated inputs and outputs. Flight data may be input from local or remote terminals. Each input message will be checked for format and logic errors, which will result in either message acceptance or message rejection. This function includes the format conversion between ICAO format and NAS Stage-A format.

Flight-data-recording generates hard-copy-updated flight-progress strips to provide a manual backup in the event of a system failure, and a record for each flight. It also provides periodic core dumps to a safe-storage device (disk, tape) for off-line system-analysis and performance-evaluation purposes. Upon request, a hard-copy printout of the meteorological data contained in the system is produced.

The tracking-and-control function calculates the arrival times at fixes, boundary crossings when flight plan or amendment data are entered, and in addition, the position of a flight along its route based on stored flight plan, and meteorological data will be extrapolated. When an aircraft-position report is received either by data link or voice, the tracking function will compare the flight-plan position with the reported position and automatically update the flight-plan position as well as the fix times whenever predefined limits (system parameters) are exceeded. When an aircraft is approaching a sector boundary and the time is less than a predefined limit (system parameter), the control function will trigger an automatic handoff message sent to both the sending-and receiving-sector controllers.

The display-function-processing covers a wide spectrum of activities which include man-machine control functions as exemplified by the Oakland experimental system. Major features include map display with boundary-limit and coverage-area selection, zooming, windowing, flight-route and velocity vector, A/C data block, situation-warning, and tabular displays.

Interfacility communication function provides the capability of exchange of data between Flight Information Regions (FIR) and between FIR's and ARTCC's. It is similar to the exchange of data between ARTCC's and ARTS systems.

Supervisory functions provide the real time control and monitor of system operations, such as task-scheduling, interrupt-handling, system-maintenance and software development, and system recovery after failure.

Phase II - The Enhancement

In phase II automation level, a number of automatic-sensing, probing, and checking functions will be added to those basic functions outlined in phase I. These additional functions include:

- Flight-Plan-checking,
- Alternate Flight-Plan-probing,
- Minimum Time-Track Calculation,
- Flight-Path-Boundary-sensing,
- Simulated Lookahead,
- Conflict Prediction,
- Automatic Handoff,
- Meteorological Data Entry and Display,
- Alert Functions and Display,
- AEROSAT Interface.

5. REFERENCES

1. Dancy, C. A., System Reliability and Recovery, Rept. No. DOT-TSC-FAA-71-16, June 1971.

2. McGonagle, J. D. and R. L. David, Advanced Multiprocessor, AF Avionics Lab., Wright Patterson AFB, OH, Tech. Rept. No. AFAL-TR-69-308, June 1970.

3. Davis, R. L., S. Zucker, and C. M. Campbell, [A] Building block approach to multiprocessing, Proc. SJCC, 1972, p. 685-703.

4. Reigel, E. N., U. Faber, and D. A. Fisher, [The] Interpreter- [A] Microprogrammable building block system, Proc. SJCC, 1972, p. 705-723.

5. Entner, R.S., The Advanced Avionic Digital Computer, Proc. High Level Aerospace Computer Programming Language Conf., NRL, Washington, DC, 29-30 June 1970, p. 15-29.

6. Lucas, H.C., Jr., Performance evaluation and monitoring, Comput. Surv., Vol. 3, No. 3, Sep. 1971, p. 79-91.

7. Johnson, R. R., Needed: [A] Measure for measure, Datamation, 15 Dec. 1970, p. 22-30.

8. Kleindorfer, P. R. and C. H. Kriebel, Analyzing Job Mix in Multiprogrammed Computer Systems, Mgmt. Sci. Res. Rept. No. 166, Carnegie-Mellon Univ., Pgh, PA, Aug. 1969 (AD 695-766).

9. Bucholz, W., [A] Synthetic job for measuring system performance, IBM Syst. J., Vol. 8, No. 4, 1969, p. 309-318.