

NASA CONTRACTOR REPORT 177458

AN EVALUATION PLAN OF BUS ARCHITECTURES AND PROTOCOLS USING THE NASA/AMES
INTELLIGENT REDUNDANT ACTUATION SYSTEM

P. De Feo
and
M. Chen
SPARTA Inc.
Laguna Hills, CA

(NASA-CR-177458) AN EVALUATION PLAN OF BUS
ARCHITECTURES AND PROTOCOLS USING THE NASA
AMES INTELLIGENT REDUNDANT ACTUATION SYSTEM
(Sparta) 34 p CSCI 01C

N88-12482

g3/05 Unclass
0110670

CONTRACT NAS2-12081

August 1987

The NASA logo, consisting of the word "NASA" in a bold, sans-serif font with a stylized "N" and "A".

AN EVALUATION PLAN OF BUS ARCHITECTURES AND PROTOCOLS USING THE NASA/AMES
INTELLIGENT REDUNDANT ACTUATION SYSTEM

P. De Feo
and
M. Chen
SPARTA Inc.
Laguna Hills, CA

Prepared for
Ames Research Center
under Contract NAS2-12081



National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

TABLE OF CONTENTS

- 1.0 INTRODUCTION
- 2.0 BUS ARCHITECTURES
 - 2.1 BUS TOPOLOGIES
 - 2.1.1 STAR
 - 2.1.2 RING
 - 2.1.3 FULLY CONNECTED
 - 2.1.4 LINEAR
 - 2.2 BUS CONTROL
 - 2.2.1 TOKEN PASSING
 - 2.2.2 CSMA/CD
 - 2.2.3 CSMA/CA
 - 2.2.4 CSMA/PA
 - 2.2.5 COMMAND/RESPONSE
 - 2.3 MESSAGE TRANSFER SCHEMES
 - 2.4 MESSAGE CHARACTERISTICS
 - 2.5 INITIALIZATION
 - 2.6 FLOW CONTROL
 - 2.7 TRANSMISSION RATES
 - 2.8 FAULT TOLERANCE
 - 2.9 TIME SYNCHRONIZATION
- 3.0 BUS EVALUATION CRITERIA
- 4.0 ANALYTICAL COMPARISON OF PERFORMANCE
- 5.0 IRAS BASED EVALUATION TECHNIQUES
 - 5.1 ARCHITECTURE CONSIDERATIONS
 - 5.2 BUS PROTOCOLS
 - 5.2.1 MIL-STD-1553B
 - 5.2.2 ETHERNET
 - 5.2.3 DATAC
 - 5.2.4 HSIS

5.3 SOFTWARE SUPPORTING CAPABILITIES

- 5.3.1 USER INTERFACE
- 5.3.2 BUS HANDLING
- 5.3.3 UTILITIES

6.0 CONCLUSIONS

7.0 REFERENCES

1.0 INTRODUCTION

The next generation of military and commercial aircraft will require increasingly high integrated control systems to achieve the desired level of operational performance and fault tolerance. This is particularly true in the case of tactical fighters which have stringent requirements of maneuvering performance and weapon delivery capabilities. Configurations with thrust vectoring, for example, will incorporate highly integrated flight and propulsion systems control. Fire control systems will also be integrated with Flight Control Systems (FCS). Technologists are now exploring the concept of an integrated Vehicle Management System (VMS) which would integrate all the flight critical subsystems: flight control, propulsion control, power management and control, and thermal management. At the same time, there will continue to be a certain degree of integration and interface with the avionics system, such as integrated sensors for navigation and flight control. Traditionally, each subsystem and component operate mostly independently from each other. However advanced design of a flight system requires significant interactions and data exchanges among them. Because most subsystems and components are implemented digitally, the required communication media will be bus oriented.

Modern, advanced technology has made it possible to produce powerful microprocessors at low cost featuring low volume, weight, and power requirements which can effectively be used for performing local, computational intensive tasks. The objective is to provide computational power tailored to the needs of each specific application, and to correspondingly decrease the computational load of the Flight Control Computer (FCC). One of these promising applications includes the failure detection and isolation, the reconfiguration management, and the control of redundant, fault tolerant, actuation systems configurations. This new actuation system concept was demonstrated with the Intelligent Redundant Actuation System (IRAS), a flexible, experimental system, with ample reconfiguration capabilities, which was designed, developed and demonstrated for NASA/Ames by SPARTA, Inc. (Ref. 1). The actuation system is controlled by dedicated, microprocessor based, Digital Control Processing Units (DCPU). DCPUs perform the local functions of: a) position control, b) failure detection and isolation, and c) reconfiguration management. DCPUs, in simplex or redundant configurations, can be dedicated to the actuation system of a single control axis, or can be shared among several actuation systems, if arranged in multiplexed configurations. An example of an integrated Flight System configuration, including an advanced actuation system like IRAS, is shown in Fig. 1.

Systems like IRAS have specific communication requirements with the FCCs which can be satisfied either by dedicated links, or by bus structures which, in turn, can be dedicated or shared with

other resources. The objective of this preliminary study is to analyze: a) promising bus configurations and protocols, and b) methods of experimentally evaluating the merits and drawback of competing configurations.

This study was performed for NASA/Ames Research Center under contract No. NAS2-12081. The authors wish to acknowledge the support, technical advices and comments of Messrs K.C. Shih, technical monitor, and N. Rediess of NASA/Ames throughout the duration of this program.

2.0 BUS ARCHITECTURES

Distributed architectures consist of several interconnected processors executing independently and asynchronously with respect to each others. The processors communicate by exchanging messages via data busses. The interconnection mechanisms and communication protocols are designed to meet the communication bandwidth, response time, data throughput and fault tolerance requirements. Candidate architectures differ from each other relative to the overall bus topology, the bus access control logic, the message transfer scheme and characteristics, the bus initialization procedures, the control of the bus traffic, the data bus transmission rate, the provisions for failure detection and reconfiguration, and the time synchronization mechanisms. These issues are further analyzed to determine the relative merits and drawback of competing architectures for supporting FCS applications in general and the communication between FCCs and DCPUs in particular.

2.1 Bus topologies

The most commonly used network topologies are: Star, Ring, Fully Connected, and Linear Bus. Large systems, which interconnect many terminals, often use a combination of those basic configurations, which are briefly described.

2.1.1 Star topology.

This configuration uses a central hub and a number of satellite terminals (Fig. 2). All communication functions, including handshaking, message conversion, and transmission error checking are performed by the hub. This configuration is well suited for those applications where a large amount of hardware and software resources can be physically concentrated in a single location (the hub) and then shared among many peripherals (satellite terminals) which can be dispersed in a wide geographical area. Telephone switching systems often use this configuration. In case of FCS applications, the FCCs could be configured as hub, and other processors, including the local DCPUs, as satellite terminals. This configuration has, however, the important drawback, that all communication must flow through, and then require the intervention of, the FCCs. This is highly undesirable in FCSs which require high frequency communications among many processing nodes, for exchanging data which might be of local relevance only. An example is the communication among DCPUs, for signal voting. The FCC intervention in this type of communication, creates an unnecessary increase of the FCCs workload, and it increases the communication overhead time. Another limitation of the star configuration is that the failure of the FCCs communication control system could bring the entire system down. For these reasons, this configuration is not appropriate for FCS applications.

2.1.2 Ring topology.

In a ring structure (Fig. 3) messages are transmitted from each terminal to the next, in fixed clockwise or anti-clockwise directions, until the final destination is reached. Several messages can travel within the ring at any given time, between different pairs of adjacent terminals. Each terminal, however, can only process one message at a time; then a transmission which requires access to a busy terminal, is momentarily held and stored until that terminal becomes available. The design of the ring structure and of the bus protocol must take into account the fault tolerance requirements of FCS. Specifically the loss of all communications paths must be prevented as a result of a single failure of the ring structure. This can be accomplished by allowing messages to flow in either directions (clockwise and counter-clockwise), or by building redundant rings, or by a combination of the above methods.

2.1.3 Fully Connected topology.

This topology provides direct connection between every pair of terminals (Fig. 4). It can be very effective in case of a small number of terminals, but as the number of terminals increases, the complexity of the hardware grows rapidly. Adding or deleting terminals requires complex modifications of the current configuration. This topology can provide fast data transmission rates and small latency times; it is highly fault tolerant due to the many available alternate communication paths between each pair of terminals. It is especially well suited for local, critical computer networks with very high throughput requirements. The hardware complexity of the terminals and the large number of interconnecting cables, however, makes it very difficult to install, modify and maintain such systems in an aircraft environment. For this reason this configuration is not suited for FCS applications.

2.1.4 Linear topology.

Linear topologies provide the simplest interconnections for multiple processors and have characteristics which make them well suited for FCS applications (Fig. 5). Two examples of linear architectures for advanced integrated avionics systems of military aircraft are shown in Fig. 6. Significant advantages of these topologies are: a) they can easily be modified by adding or deleting terminals, b) they can be structured in fault tolerant configurations, and c) the supporting software and hardware technologies are well developed. Linear topologies are very well suited for carrying the communications between FCCs and DCPUs by using either dedicated Actuation Control Bus (ACB) architectures or connecting the DCPUs to existing FCS Bus systems. The advantage of using a dedicated ACB is that the resultant hierarchical architecture (Fig. 6) provides for functional independence, fault confinement, high throughput and low latency. The disadvantage is the additional hardware which is required to

implement this configuration. If an existing FCS bus is utilized, the resultant architecture is single level, instead of hierarchical, and the data throughput and latency time are affected by the demand of all the other processors sharing the same bus.

2.2 Bus control

Control of the access to a bus and of the information flow through it, is performed by bus controllers. Bus control configurations can be centralized or autonomous.

In centralized configurations a single bus controller is active at any given time. An independent bus monitor continuously tests the state of the bus controller and transfers control to other terminals in case of detected failures. All communication, between the bus controller and other terminals, or among terminals, can only be initiated by the bus controllers. For this reason communication among terminals require more time than communication between bus controller and terminal. The MIL-STD-1553B is an example of a highly structured, centralized bus controller. In FCS applications bus controllers typically reside in the FCCs.

In autonomous configurations each terminal is also a bus controller, and therefore direct communications can be made among all terminals which reduces the required transmission time. The challenge, in this case, is to avoid more than one terminal at a time to gain control of the bus, and the possible collision of different messages. Several mechanisms have been developed to eliminate this problem; the most commonly used are:

2.2.1 Token passing.

This is a control method of Collision Avoidance (CA) in which a "free token", authorizing use of the bus is passed in a logical ring from one user to another. A terminal can gain access of the bus only when in possession of the token. Token passing protocols are very attractive because of the versatile structure of terminal priority schemes and of token possession time which can be built to satisfy a broad area of applications. Access to the token can be provided according to deterministic or probabilistic logic. Terminals can be assigned different levels of priority, statically or dynamically, which affects the relative frequency and duration of the periods of Token control. Token passing protocols are also attractive because new terminals can easily be added, with no modification to the existing structure. This bus control method, supported by a deterministic control logic for token passing, is well suited for FCS applications because of the structured and repetitive nature of the interprocessor communications which consist, primarily, of messages of predefined length to be transmitted at predefined intervals. Conversely, a probabilistic logic for token passing is not

appropriate because it can not guarantee the completion of time critical message exchanges within precisely allocated time slots.

2.2.2 Carrier Sense Multiple Access with Collision Detection (CSMA/CD).

This protocol provides random bus access to all terminals. Prior to gaining control of the bus, a terminal must sense the carrier and determine that the bus is available. The terminal then transmits the message (if the bus is found busy, the terminal makes other attempts, after random periods of times). A very small probability exists that two terminals simultaneously sense the bus, both determine that the bus is available, and both initiate message transmission. The two messages then collide and garble each other. To detect collision, transmitting terminals monitor the state of the message, a few clock cycles after transmission. If the message is found to be garbled, collision is detected and the transmission is attempted again after a random period of time. A popular bus which uses this concept is Ethernet, a very fast bus (10. MBPS) most often used in Local Area Network (LAN) systems, but also suited for FCS applications, as discussed later.

2.2.3 Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA).

The procedures for bus access are the same as those described for the CSMA/CD protocol. The difference is that the CSMA/CA protocol uses techniques for avoiding collision of messages, rather than for detecting collision. A commonly used avoidance technique requires each terminal to transmit a short access signal prior to transmitting the actual message. In the rare event that two terminals transmit simultaneous access signals, a collision occurs which results in both access signals being garbled. If this condition is detected, both transmitting terminals interrupt the transmission process and, after a random period of time, attempt again to gain control of the bus.

2.2.4 Carrier Sense Multiple Access with Positive Acknowledgment (CSMA/PA).

This is a mechanism to ensure that a message has been correctly received. In case that the transmitter does not receive a positive acknowledgment message from the receiver within a preestablished period of time, then transmission is tried again. If positive acknowledgment is not received within a maximum number of tries, then a failure condition is flagged.

2.2.5 Command/Response.

This is a centralized control method in which the bus controller authorizes the use of the bus to other terminals which, in turn, acknowledges the authorization back to the bus controller. The MIL-STD-1553 uses this protocol.

Typically data busses use a combination of techniques to transfer control, and to avoid and detect collision of messages. For example the Boeing sponsored Digital Autonomous Terminal Access Communication (DATAC) data bus, uses a combination of token passing and collision avoidance concepts. Each terminal has a preprogrammed schedule of activities which establishes the time window of bus control, the information to be transmitted and the destination, and the information to be received. DATAC protocol also provides a degree of synchronization among terminals. This is accomplished by three timers, all programmable, which are implemented in each terminal, and which control the transmission process (transmit gap, synchronization gap, and terminal gap).

2.3 Message transfer scheme.

Two message transfer schemes are possible. They are: a) terminal to terminal and b) broadcast. Both transmission schemes can be implemented with or without positive identification. Terminal to terminal transmissions are from one source (the transmitter) to a single destination (the receiver). Broadcast transmissions are, instead, intended to be received by all terminals. In this case each terminal must first identify each message and then must determine if it is of relevance or not. Clearly broadcast is a more effective way of transmission than terminal to terminal. Broadcasting, however, lacks the extensive handshaking capabilities which are feasible in case of terminal to terminal transmissions which, for these reasons, are a preferred method of transmission in case of safety critical applications, like FCS.

2.4 Message characteristics

The bus protocol establishes the allowable formats and information content of each transmission packet. Typical information to be included are:

- 1) Transmission type; broadcast or terminal to terminal;
- 2) Destination address, in case of terminal to terminal transmission;
- 3) Message length, number of words, number of bits per word;
- 4) Message priority;
- 5) Message type: data message, control transfer, failure status, acknowledgement;
- 6) Timing information: message start/end time, variable time tags.

The control, timing and handshaking information which can be attached to each message, can provide and support: a) powerful failure detection mechanisms, b) reconfiguration management strategies, and c) synchronization schemes among all intercommunicating processors. They do, however, increase the latency and transmission time. FCS applications are flight safety critical, as well as time critical. They require rapid and complete identification of all failures and very fast and predictable transmission times. For these applications it might be effective to provide messages, terminals or both, with a priority control scheme, so that those messages for which fast transmission is crucial, can be processed without undue delay; lower priority messages would require longer transmission time. For instance high priority can be assigned to the position control commands from the FCCs to the DCPUs, and low priority to the failure status of the actuation control systems from the DCPUs to the FCCs.

Examples are shown of two message formats. One refers to a 1553B, controller to remote terminal, transmission (Fig. 7); the other refers to a DATAC transmission (Fig. 8). The complex structure of the 1553B protocol, is reflected in messages which require numerous control bits which: a) define source and destination address, and message length, b) monitor and control the transmission, and c) flag any transmission failures. The DATAC message format is much simpler than the 1553, it includes less overhead bits, and provides less transmission monitoring and controlling capabilities.

2.5 Bus initialization.

Bus initialization occurs at power-on time. Functions performed during initialization include: set-up of the bus controllers; initial handshaking and synchronization of all terminals; and self diagnostics. Extensive diagnostics procedures are highly desirable in flight critical applications because: a) the operational reliability and fault tolerance requirements assume a system which is initially fully operational and b) the system is not designed to withstand two simultaneous failures, and therefore it is assumed that that condition has only an extremely low, acceptable probability to occur (simultaneous failures is a condition in which a new failure occurs prior to the system correctly reconfiguring to mask previous failures). If a system goes in line with an undetected failure, then any subsequent failure would be simultaneous to the preexisting one, and this condition could have catastrophic consequences, in flight critical applications.

Initialization is an ideal time to perform extensive failure detection procedures because of the lack of the time constraints

existing during real time operation. A subset of those procedures are also used during real time operation, every time a bus reconfiguration is needed as a result of detected failures or other reasons.

2.6 Flow control.

Not all terminals connected to a bus system operate at the same rate. If a fast terminal attempts to communicate with a slower terminal, under utilization of the bus and of the fast terminal might occur. Flow control includes a set of procedures attempting to optimize bus utilization by keeping the data rate close to the nominal values. Data buffering and stop-and-go handshaking techniques have been proposed for this purpose which, however, further increases the existing bus transmission overhead. These techniques do not appear to be effectively applicable to FCS applications which use a controlled and rather uniform set of terminals.

2.7 Data bus transmission rate.

The actual speed at which data can travel through the bus is measured in Bits-Per-Second (BPS), or MegaBits-Per-Second (MBPS). Transmission speed is a function of: a) the hardware media (electrical wire or optical fiber cable), b) the number of terminals connected to the bus, and c) the distance of separation among terminals.

A clear trend is established for increasing use of optical cables in FCS applications primarily because they are insensitive to Electro-Magnetic Interferences (EMI), and have high transmission bandwidth. The technology of electrical wires, however, is well developed and it is still most commonly used. With current technology, transmission rates of 10 MBPS or higher are achievable with both media.

The distance between terminals affects the transmission rate and the transmission error rate. In the case of FCS applications the distances involved are typically short; the resultant transmission rates are relatively high and error free, compared to applications which require long distances.

2.8 Failure detection and fault tolerance.

Flight safety applications like FCS require rapid detection and identification of failures and bus reconfiguration. Failure detection algorithms which can be exercised off-line, during bus initialization for example, include: program memory check-sum, parity checks, watch-dog timers (driven by internal control

timers, like those provided by DATAC), and others. Several techniques also exist to test each terminal during the real-time operation of the bus. The most commonly used are based on low frequency (1 Hz or less), periodic polling of each terminal by the bus controller. Failure of a terminal to correctly respond to two consecutive pollings, indicates a failed terminal. The F-16 uses this technique to identify failed terminals. Message integrity can also be tested using a combination of software techniques (parity, check-sum and others).

Very fast reconfiguration times are required for FCS applications. In fact the entire system is in a very vulnerable state until reconfiguration is completed, as previously discussed. Reconfiguration is achieved by transferring bus control from a failed bus controller to another terminal, by rerouting messages to by-pass failed portion of the bus, and by using available redundant components.

2.9 Time synchronization techniques.

Information among processors can be exchanged using dedicated links or bus structures. In case of dedicated links, for all practical purposes, no significant delays are typically associated with the data exchange. Furthermore these delays are constants, from transmission to transmission, and therefore they can be compensated, if necessary. In case of bus structures, instead, delays due to latency and finite transmission rates can be significant, and vary from transmission to transmission depending on the availability of the bus, the message size and type, and the state of the receiver. Techniques for synchronizing terminals and for time referencing each message must then be used, if the messages contain time sensitive variables, as typically in the case of FCSs. Two techniques often used are: global time reference and time tagging.

Global time reference is a technique in which a terminal, typically the bus controller, is designated as the time master, and it transmits its own internal time to all other terminals which then synchronize with that time. The internal time of the bus controller becomes effectively the global time of the system. The bus controller periodically updates the global time so that local time shifts can be compensated.

In case of time critical applications, it might not be sufficient that all terminals use the same time reference, and it might be necessary to actually time tag every variable exchanged, to establish the exact time of reference for each variable. This process is often and effectively applied in case of time critical variables, like inertial platform data. It does, however, introduce additional transmission overhead time and therefore must be judiciously applied only to those transmissions which require it.

3.0 BUS EVALUATION CRITERIA

The overall evaluation for a data bus system for critical FCS applications, must include the following quality parameters:

- 1) Flexibility. This involves Bus control, message transfer schemes and message characteristics, bus initialization, flow control, and ease of modifying;
- 2) Bus speed. This includes Bus transmission rates and Bus latency;
- 3) Fault tolerance. This includes failure detection and isolation mechanisms, redundancy and reconfiguration strategies and time;
- 4) Time synchronization. This includes global time reference and time tagging capabilities.

Flexibility is important because a bus architecture must be able to support transmissions of a large variety of messages among many terminals. Flexible architectures support several transmission schemes; the user can then select the optimum scheme to satisfy the specific requirements of each transmission. For example, broadcast methods can be used to transmit messages to more than one terminal. Pilot selected aircraft control and operational modes, and aircraft state information, can be broadcasted, from the FCCs to the DCPUs of each effector, so that the actuation control gains can be appropriately adjusted in all control axes, at the same time. Time critical position commands, which vary from effector to effector, can be effectively transmitted using terminal to terminal transmission schemes. Time critical variables can be time tagged; or high priority can be assigned to those messages which include time critical variables.

Another important measure of flexibility is the ease of modifying configurations. In fact, during their life cycle (ten years or longer) FCS are continuously upgraded for improved performance. This might require the addition of new terminals to an existing bus, or the deletion of others. It is important that enhancements of this kind can be performed with minimum modifications to the existing equipment. Typically, when a new terminal is added, it is necessary to modify the bus controllers so that they can identify the new terminal. In the case of the DATAC bus the three timers embedded in each terminal must be preprogrammed to accommodate transmission from/to the new terminal.

Bus speed, fault tolerance and time synchronization are clearly important variables for critical FCS applications. The evaluation of these parameters is further discussed later.

Each candidate bus configuration and protocol must be evaluated relative to the four quality parameters. A preliminary evaluation can be conducted simply using analytical methods. Dynamic parameters, however, like transfer rates, latency times, and time for failure detection and reconfiguration, can not be estimated with sufficient accuracy using analytical methods only; they must be evaluated in real time environments which simulate actual operational situations. The IRAS can provide such environment, in a cost effective manner, as discussed in the following section of this document.

Relative to transmission rates and latency times, the bottom line parameter is the the total elapsed time between the time a variable is computed in one processor, and the time it is available for use in another processor. The elapsed time is affected by hardware and software implementation characteristics, as well as by the operational scenarios. Clearly the selection of a bus protocol and bus configuration has the biggest impact on the resultant elapsed times. It is important to notice, however, that even if protocol, configuration, message format and operational scenario are kept constant, the elapsed time varies from message to message as a result of the many, possible different states of all relevant hardware components, at the time each transmission is initiated. In fact some of those components might not even be available immediately for a pending transmission if they are still busy performing previously scheduled transmissions. The software structure of the receiving and transmitting processors, and their relative execution cycle skewness are other parameters which can also randomly effect the elapsed time of each transmission. This is particularly true in the case of unsynchronized configurations. Finally, operational scenarios can add additional uncertainties relative to total elapsed time, primarily in configurations where hardware resources, like the actual bus lines, are shared among many processors.

It is now apparent that the overall elapsed time of different protocols, architectures, and operational scenarios must be defined in terms of average values and distribution spreads. To evaluate these parameters, it is required: a) to simulate and to exercise each configuration and scenario of interest in a real time environment, and b) to collect a large number of data, so that statistical distributions of the elapsed time can be determined.

During the real time operation of the bus, dynamic reconfigurations are required after a failure is detected. A failed terminal, for example, might need to be logically removed, so that the data throughput, timing sequences and bus control are not affected. This is especially important for autonomous configurations, to prevent a failed terminal, for example, from taking control of the bus and never releasing it. The time needed to identify a failure and to appropriately reconfigure is crucial, because: a) until the reconfiguration process is

completed, the system is in an extremely vulnerable state, as previously discussed, and b) the reconfiguration process inevitably results in transmission delays which, if large enough, could effect the aircraft dynamics in an unacceptable manner. Like transmission elapsed times, failure detection and reconfiguration times vary from case to case, and therefore they must also be evaluated in real time environments, and expressed in terms of average values and statistical distributions.

4.0 ANALYTICAL COMPARISON OF BUS PERFORMANCE

Four busses have been identified which can effectively support FCS applications. They are: MIL-STD-1553B, DATAC, Ethernet, and HSIS. The 1553B is an established bus commonly used in military avionics systems. DATAC is a bus, still in the experimental stage, which is sponsored by Boeing and is intended to support avionics and FCS applications (DATAC is the acronym for Digital Autonomous Terminal Access Communication). Ethernet, also known as IEEE-802.3, is the most commonly used in Local Area Network (LAN) applications. The High Speed Interconnect System (HSIS) is a bus still in the development stage, which incorporates the SAE AE-9B linear token passing protocol. HSIS is being developed by Sperry Defense Products.

An analytical, preliminary comparison of the specifications of the four busses is shown in Table 1. The purpose of this preliminary evaluation is to describe the relevance of some major characteristics to the specific application. A dynamic analysis must be performed to accurately evaluate the real time performance of each bus, as previously discussed.

- 1) The 1553B is the only centralized bus controller; the other three busses all use autonomous controllers, which provide increased flexibility of transmission and shorter transmission overhead, as previously discussed.
- 2) Bus access control is deterministic in all cases except Ethernet which has random access control. Deterministic control is preferable for FCS applications which are based on the repetitive, deterministic execution of preestablished sequences of algorithms and I/O processes. Probabilistic access typically requires less overhead. Unpredictable message transmission latencies, however, can occur which might be unacceptable in case of time critical messages; the probability of this happening is inversely proportional to the bus transmission rate and directly proportional to the maximum length of the messages to be transmitted.
- 3) The preferred message transfer scheme is terminal to terminal with acknowledgment, relative to transmission integrity and rapid identification and confinement of faults; this scheme however, requires a high transmission overhead. The fastest transmission scheme is broadcast without acknowledgment, which does provide limited capabilities of rapid failure detection and identification. Speed of transmission and rapid failure detection and identification are both very important in FCS applications. An optimum balance of these two parameters must be made on a case to case basis, depending on the nature of each message.
- 4) The maximum message length (data words only, not including protocol) is 32, 256, 750 and 4096 words for the 1553B, DATAC, Ethernet, and HSIS respectively. In case of the 1553, some long transmissions can require multiple messages. This is most likely

to occur in highly integrated FCS architectures, which require exchanges of large amount of data. The minimum message length is 1 word for all busses except Ethernet which has a minimum of 23 words (the original application of Ethernet, Local Area Network, requires transmission of very large messages, typically much larger than 23 words). Because the communications between FCCs and DCPUs, consist of short messages of a few words each, this characteristic of Ethernet unduly increases the overhead of many transmission.

5) The flexibility of declaring the priority of each message is very useful in FCS applications which typically require a combination of time critical and non time critical messages. The transmission latency of time critical messages can be reduced if a high priority level is assigned to them. Only HSIS has encoded priority.

6) The availability of bus control bits provides the capability of controlling and interrogating terminals, and initiate diagnostics procedures. 1553 and HSIS have that flexibility.

7) Error flags, embedded in each message, provide rapid identification of garbled messages and components malfunctions. They are available in 1553 and HSIS.

8) In the transmission protocol of each bus, a fixed area is reserved to identify the source/destination terminal addresses and/or to provide an identifier to the message itself. The 1553 has 5 bits to identify each terminal, so that the maximum number of terminals is 32. DATAC uses a broadcast transmission mode which does not requires the identification of source or destination addresses. In this case a field of 12 bits is provided which univocally defines each message, from each terminal. From that identifier each terminal can independently determine if a message is of relevance and, if this is the case, the appropriate storage locations. Ethernet reserves 48 bits for terminal identification. Hardware considerations, however, limit the maximum number of terminals to 256. The total address field of HSIS is 16 bits long. Of these, 7 are used to identify the terminals, and the other 9 define hardware subaddresses within each terminal.

9) All busses provide self test capabilities as part of the initialization procedures, following power-up. During real time operation the 1553 and HSIS provide failure reports, upon request or automatically. DATAC and Ethernet do not provide similar reports; failed terminals, however, automatically put themselves off line.

10) The 1553, as previously discussed, can interconnect up to a maximum of 32 terminals, which is sufficient for most applications. Highly integrated FCS configurations could easily require more than than 32 terminals to be interconnected. In this case, multiple bus structures would be necessary. The other three busses provide adequate interconnection capabilities for all FCS applications.

11) The bandwidth of the 1553 is the smallest (1. MBPS) of the four busses and marginal for most state-of-the-art FCS applications, which require increasingly higher transfer rates. Bandwidth of 10. MBPS are currently achievable and better reflect current needs.

12) A certain level of built-in redundancy is provided for all busses except Ethernet. All four busses can, of course, be arranged in redundant configurations, independently from the available built-in level of redundancy.

13) Global time of reference is only available to HSIS. Global time can be used to implement computational frame synchronization among many terminals, a technique commonly used in FCS applications. Time tagging is still necessary in case of time critical variables.

The overhead associated with the transmission of selected messages can be estimated based on each bus characteristics. The overhead is determined by dividing the total number of overhead bits by the total number of message bits (overhead bits + data bits) in each transmission. The assumption is made that the time critical communications between FCCs and DCPUs, for each control axis, include: a) one 16 bits word from the DCPU to the FCC every 50. msec. (the position feedback information); b) one 32 bits word, every 5. msec., from each DCPU to all others within the same control channel (cross-channel voting); and c) two 32 bits words from the FCC to the DCPU every 10. msec. (one word defines the position command, the other word defines the current aircraft control and operational mode). Transmission overhead, shown in Table 2, are very high, primarily because the messages are very short. In all cases, the overhead decreases with the length of the message; it is consistently the lowest in the case of DATAC and the highest in the case of Ethernet. The reason why Ethernet overhead is so high is that the minimum number of data words, per transmission, is 23. If a transmission of less than 23 words must be executed, the unused portion of the data field is still transmitted, which of course increases the overhead. Based on the estimated values of overhead times, and on the bus bandwidth, approximate estimates can be made of total transmission times. It must be noted that high overhead does not necessarily imply long transmission times; it only implies longer times than in the case of low overhead.

As previously discussed, accurate evaluations of dynamic parameters can only be made by exercising each bus in real time environments, which simulate the actual operational environments. The IRAS can effectively be used for this purpose. All four busses can be accurately evaluated by using a combination of actual hardware, simulation and emulation techniques. The capability also exists to provide some general simulation tools to evaluate bus concepts not yet implemented in hardware.

5.0 IRAS BUS EVALUATION TECHNIQUES

The IRAS can provide a flexible environment for analyzing several bus architectures and bus protocols using a combination of simulation and emulation techniques and, if available and practical, actual hardware.

5.1 Architecture considerations.

It is not too restrictive to limit the dynamic evaluation of bus architectures to linear bus topologies, because they are almost exclusively used in FCS applications. The capabilities must be provided, however, for simulating and evaluating linear bus topologies which: a) exclusively support the FCCs/DCPUs communications, and b) support a variety of FCC communication requirements, including those between the FCCs and the DCPUs. In case of a shared bus, it is essential to simulate the load on the bus due to transmissions other than those between FCCs and DCPUs.

Relative to DCPUs configurations, the capabilities must be provided for simulating: a) simplex or redundant DCPU configurations each dedicated to a single aircraft effector (like the current IRAS configuration), and b) DCPU configurations which are multiplexed among many effectors. The distinction between dedicated and multiplexed configurations is that, in multiplexed configurations, the position commands from the FCCs to all effectors served by the same DCPU can be combined in a single message.

A schematic of a simulation environment, implemented within the IRAS, is shown in Fig. 9. The schematic shows three separate terminals, which are directly connected to the bus. They are:

- 1) The DCPUs enclosure, modified by the addition of a Data Bus Interface Card (DBIC), and the removal of the Emulated Flight Control Computer (EFCC), which is implemented as a Single Board Computer (SBC). The objective of this component is to simulate several DCPU configurations, dedicated or multiplexed, single string or redundant.
- 2) An enclosure containing the EFCC card and a DBIC card. This component provides the capabilities of simulating FCCs architectures.
- 3) An enclosure containing either an SBC and a DBIC, or an IBM-PC and a DBIC. This component, called Environment and Bus Controller (EBC), can be configured to perform several tasks, depending on

the objectives of the analysis. They are: a) the generation of messages to simulate the bus loading due to messages not related to DCPUs; b) the simulation of a bus monitor; c) real time recording of bus traffic and bus performance evaluator.

The EBC can be implemented as a SBC identical to those already included in the IRAS. The advantages of this approach are: a) it is the lowest cost approach; b) the 68000 based SBC is very powerful, and it comes equipped with 500k on board memory; c) all terminals interfacing the bus have identical hardware, which decreases the software development cost, and it provides additional configuration flexibility (for example, a redundant FCC structure can be easily implemented).

The alternate way to implement the EBC is by using an IBM-PC. The advantages are: a) the capability of storing very large amount of bus traffic data, using the available disk storage space; b) the availability of many statistical analysis tools; c) ample availability of utilities and support functions like graphics displays and data file handlers. The SBC and the IBM-PC based configuration are both attractive for different reasons. The IBM-PC is the preferred choice, however, because many tools are readily available for that machine, which are necessary to collect, file, analyze and display the extremely large set of data that are required for evaluating bus dynamic performance.

5.2 Bus protocols.

The four promising bus architectures which have been identified as having the promise for effectively supporting FCS application in general, and FCCs/DCPUs communication in particular, are: MIL-STD-1553B, DATAC, Ethernet, and HSIS. Different approaches for analyzing the dynamic capabilities and performance of each of those busses, are described.

5.2.1 MIL-STD-1553B

The 1553B is an operational bus, commonly used in avionics applications. Bus controllers, compatible with 68000 microprocessors and IBM-PCs are available in the market. The most effective way to analyze that protocol is then by using actual hardware. The IRAS can be configured so that the EFCC operates as bus controller, and the Arbitrator as remote terminal. Depending on the objective of the analysis, the EBC can be configured as bus monitor, for analyzing the time of failure detection, isolation and reconfiguration; or as the environment processor, for simulating the demand on the bus by other terminals.

Software modules, executing in the three terminals, control and monitor the experiment, including: the initiation of all data transfers (terminal to terminal, terminal to controller, and controller to terminal); monitoring and timing all discrepancies between data sent and data received; and time tagging all

processes to determine time latency and transmission time of each individual message.

5.2.2 Ethernet

Ethernet is an operational data bus. Hardware compatible with the IRAS architecture (68000 and IBM-PC) is available in the market which can effectively be used for analyzing dynamic performance, using methods much similar to those used for the 1553. The Ethernet terminals, contrary to the 1553, are all identical and act autonomously from each other. In this case two terminals can be used for transmitting messages representative of those required for FCCs/DCPUs communication; the third terminal acts as the environment terminal as well as the bus monitor and time tags and records all transmissions. The objective again is to analyze overall transmission delays and the effects of message collision and transmission retrieval sequences.

5.2.3 DATAC

A different approach must be taken to evaluate this bus because it is still in a development stage. At the current time, only custom experimental hardware implementations exist of that bus. The DATAC protocol can be simulated, however, so that an evaluation of the bus performance can be made that, although not as accurate as in the previous two cases, is still significantly better than in the case of just performing an analytical evaluation. The DATAC protocol can be simulated using the 1553 or Ethernet terminals.

The 1553 and DATAC have similar message bandwidth and word length. They also use much similar terminal to host processor interface procedures for controlling the process of receiving and transmitting messages; they are based on CPU interrupt and Direct Memory Access (DMA). The 1553 broadcast transmission mode closely resembles the DATAC transmission mode. All DATAC terminals have identical priority for bus access; this can be simulated by evenly and dynamically assigning bus controlling status to all 1553 terminals (the time required by the 1553 for the dynamic allocation must be biased out from the result of the simulation). Both busses have a built-in level of internal redundancy, which can be used for analyzing the distribution of reconfiguration times, following induced failures. Important differences also exist between the two protocols which must be taken into account by properly interpreting the results. In some cases, the differences might limit the scope of the analysis to some extent. Major differences exist in the area of bus access scheme (CSMA/CA for DATAC, and Command/Response for 1553); and failure detection and isolation (local to each terminal for DATAC, centrally controlled by the Bus controller and monitor for the 1553). Other minor differences exist relative to message format and length which can easily be compensated for.

DATAC and Ethernet also have significant similarities and differences. Both use a broadcast transmission mode (Ethernet

also supports terminal to terminal transmissions); they use the same scheme of carrier sensing and the same terminal to host interface procedures. Some major difference are: Ethernet does not have internal redundancy; Ethernet bus access include collision detection techniques (DATAAC only utilizes collision avoidance techniques). Other differences in transmission and word formats between the two busses can easily be compensated by taking advantage of the very high transmission speed of Ethernet compared to DATAAC.

For best results both methods of simulation can be used. In fact the dynamic reconfiguration capabilities of DATAAC can be better estimated by using the 1553 based simulation, while DATAAC transmission delays can be better estimated using Ethernet based simulation.

5.2.4 HSIS

HSIS is a recently developed bus for which hardware is not as yet available. Like in the case of DATAAC a simulation environment can be implemented in the IRAS, which uses components with similar characteristics. HSIS is a high speed bus which uses token passing, a powerful, flexible way of controlling bus access, which is quite different from the access scheme of the 1553 and Ethernet. Therefore 1553 or Ethernet hardware can not be used for this purpose. In fact a realistic simulation environment for the HSIS requires a hardware implemented token passing protocol. A promising simulation environment of HSIS can be developed using ARCnet, a LAN protocol, which is implemented in hardware compatible with the IRAS hardware (IBM-PC and Multibus). Significant similarities exist between HSIS and ARCnet, including the message acknowledgement schemes. The implementation approach of ARCnet in the IRAS would be identical to that of Ethernet.

5.3 Software supporting capabilities.

The analysis of the dynamic performance of bus architectures require the development of several software implemented capabilities in the area of user interface, bus access handling, and general supporting capabilities or utilities, which are briefly discussed.

5.3.1 User interface.

The user interface provides the experimenter with the capabilities of defining the experimental environment relative to the hardware configuration (definition of the protocol, and role assignment to each terminal); the communication among terminals (message type and frequency, and time tagging); and the duration of the experiment (data transmission time, data buffer size).

5.3.2 Bus handling.

These software modules provide the low level interfaces to the hardware (interrupt handlers, I/O and interfaces), implement the user directives (transmission schedulers, timing tagging and monitoring) and provide some failure insertion mechanisms.

5.3.3 Utilities.

They are general functions for data collection, analysis and displays.

6.0 CONCLUSIONS

A preliminary analysis has been made of the characteristics of bus architectures and protocols which can be effectively applied to FCS applications. Of particular interests were the communication requirements between the FCCs and the microprocessor based, dedicated controllers for redundant, reconfigurable actuation control systems. Four different busses were found promising for the intended application: MIL-STD-1553B, DATAC, Ethernet, and HSIS.

An analysis was also made of the feasibility of using the NASA/Ames IRAS laboratory for analyzing the dynamic performance of those busses in an experimental environment representative of the actual operational environment. The analysis shows that the IRAS can be used effectively for this purpose, and that this approach would require only minor modifications to the existing systems, and it would provide very high quality data in most cases.

7.0 REFERENCES

- Ref.1 "IRAS requirements and preliminary system design"
May 1985. Contract NAS2-12081

ORIGINAL PAGE IS
OF POOR QUALITY

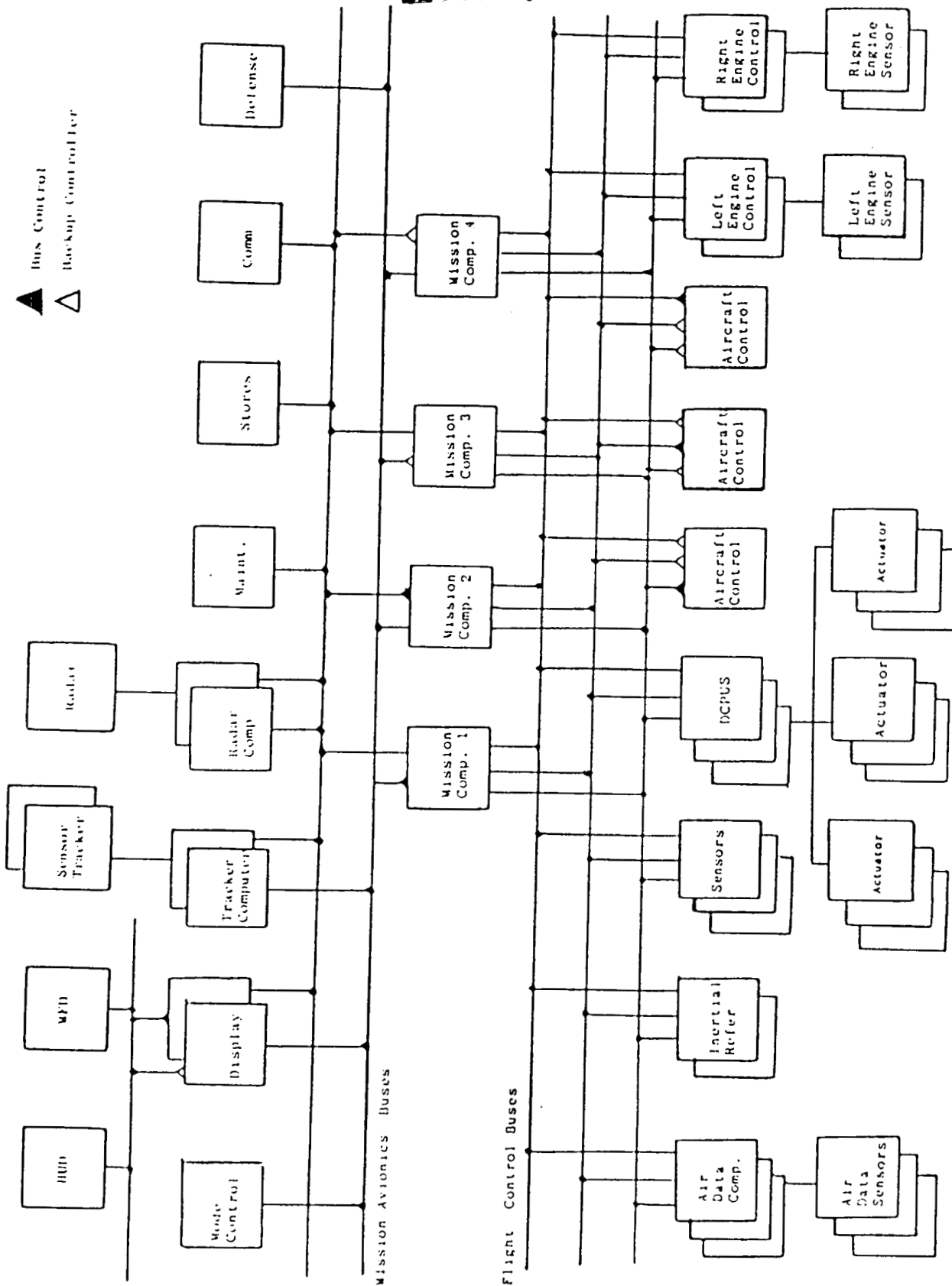


FIG. 1 Integrated Flight System for Advanced Fighter

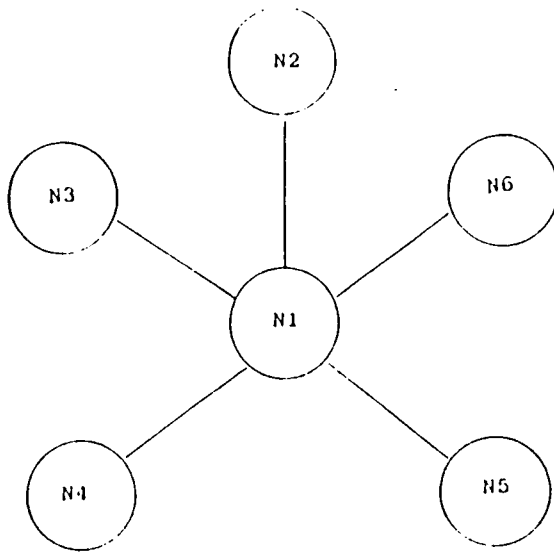


FIG. 2 STAR TOPOLOGY

N_i = BUS CONTROLLER/ REMOTE TERMINAL/ TRANSMITTER/ RECEIVER

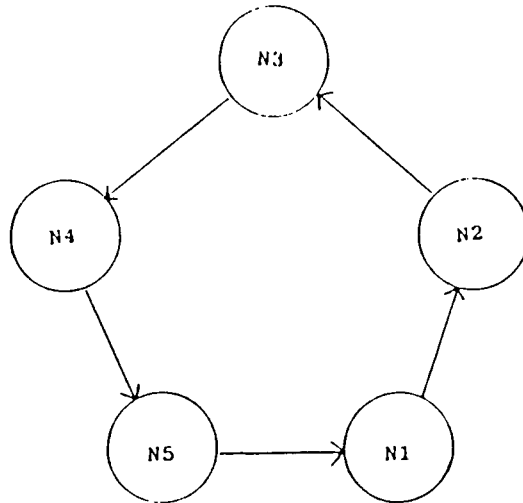


FIG. 3 RING TOPOLOGY

ORIGINAL PAGE IS
OF POOR QUALITY

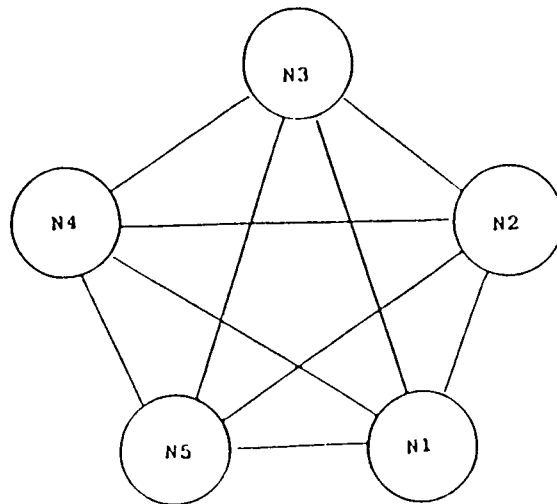
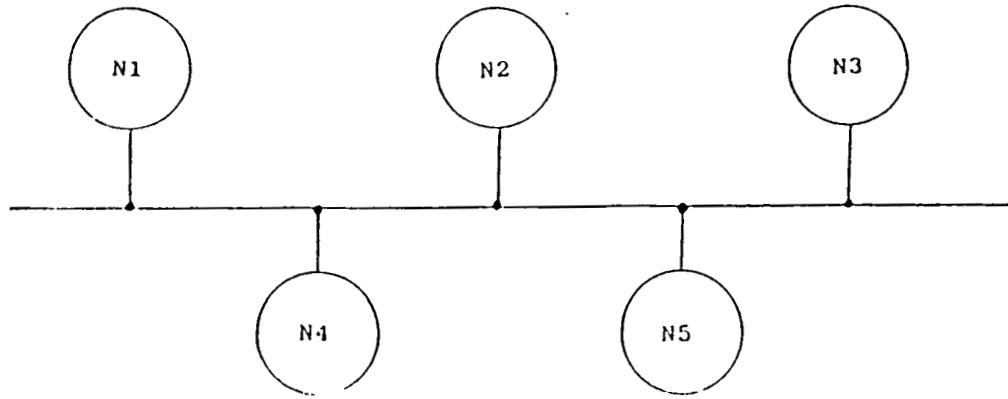
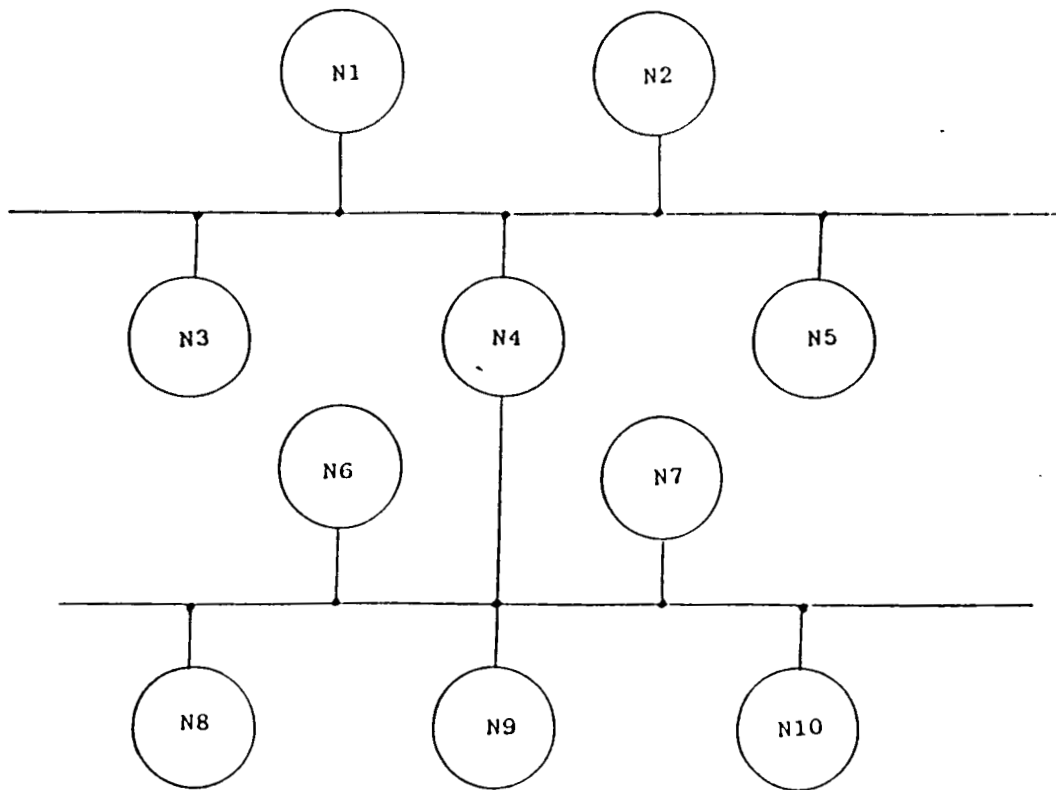


FIG. 4 FULLY CONNECTED TOPOLOGY

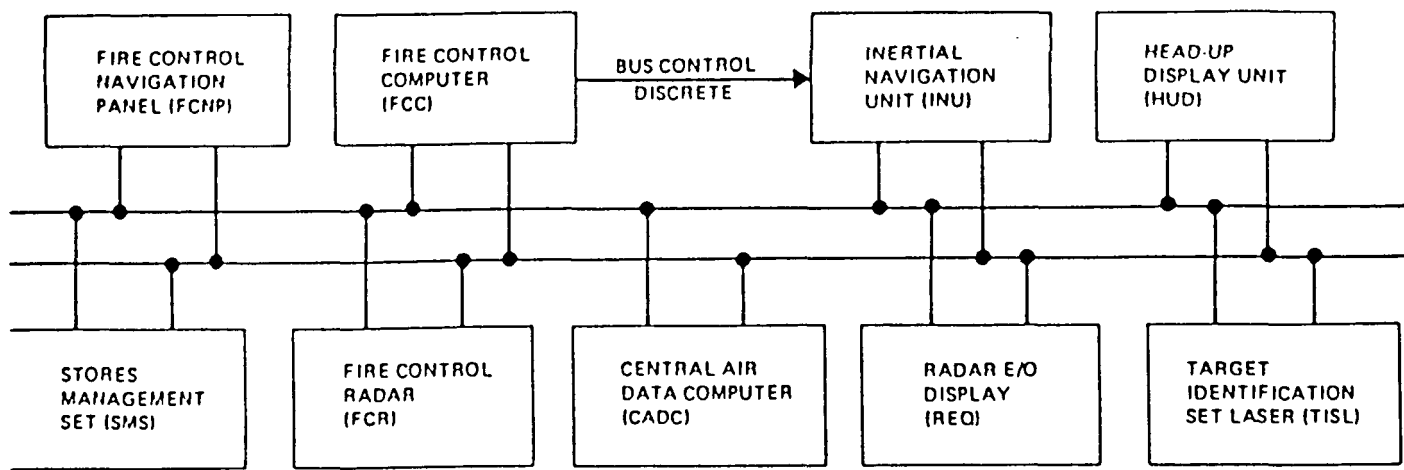


5A. SINGLE LEVEL

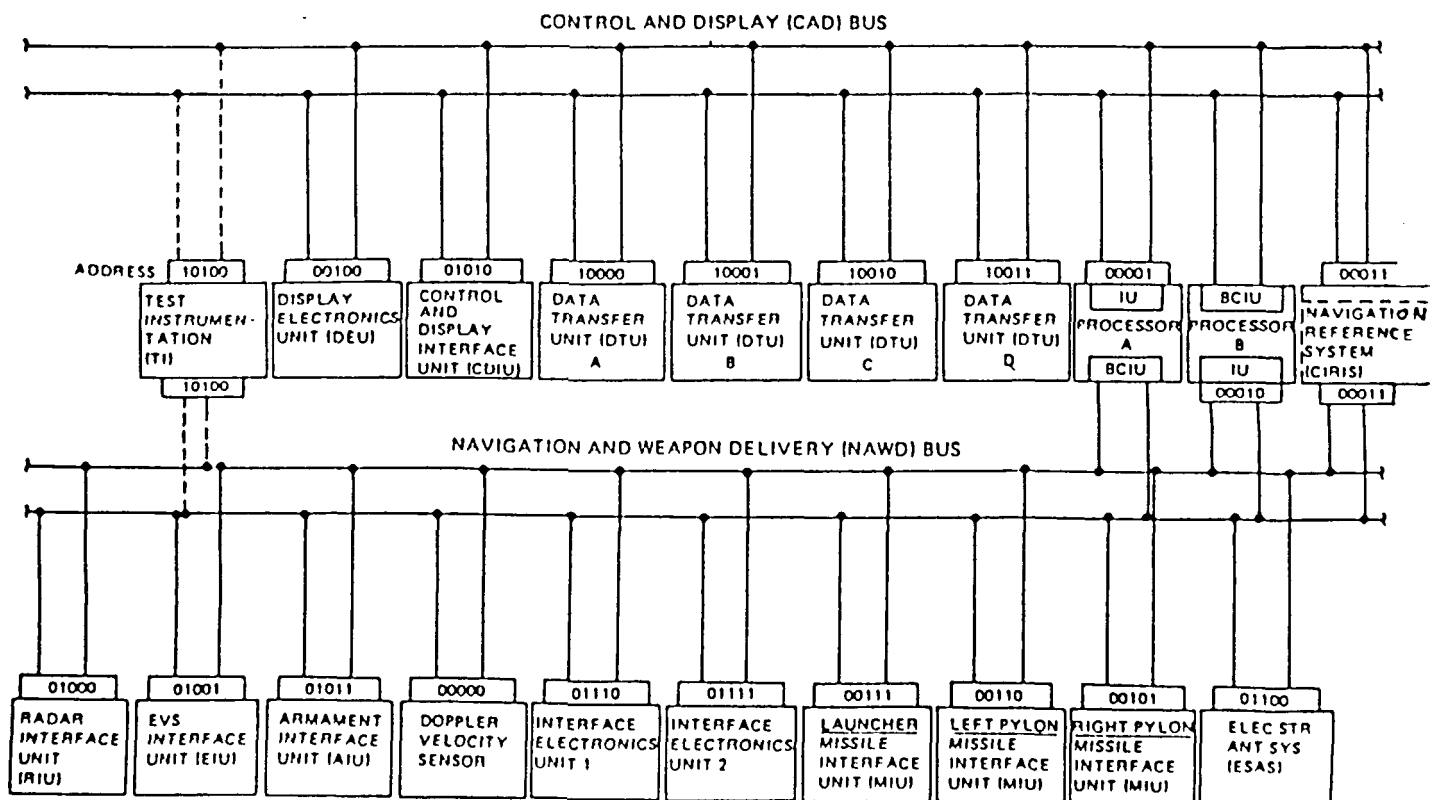


5B. MULTIPLE LEVELS

FIG. 5 LINEAR TOPOLOGIES



6A. SINGLE LEVEL - F-16 AVIONICS



6B. MULTIPLE LEVELS - B-52 AVIONICS

FIG. 6 LINEAR TOPOLOGIES OF AVIONICS SYSTEMS

ORIGINAL PAGE IS
OF POOR QUALITY

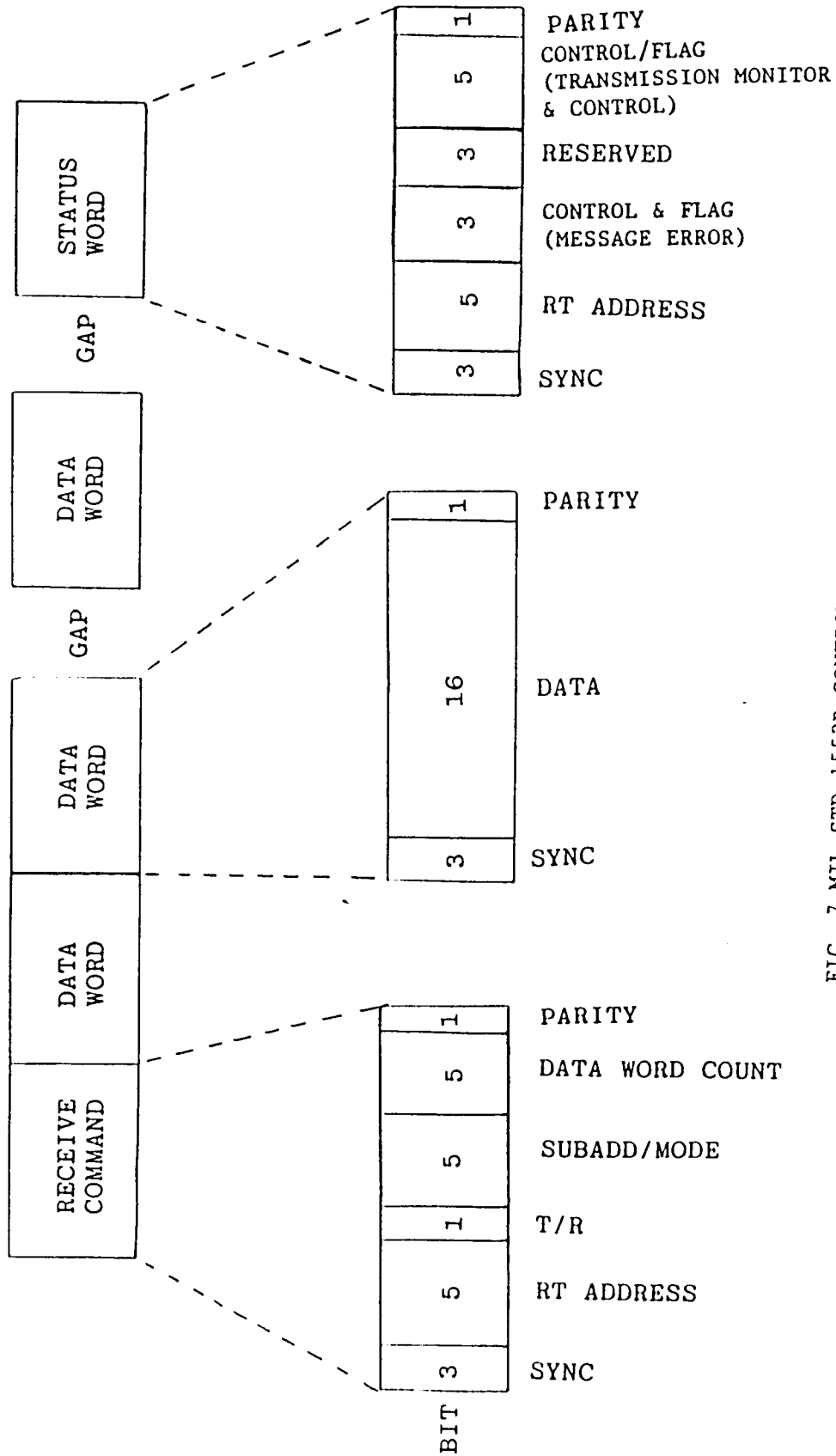


FIG. 7 MIL-STD-1553B CONTROLLER TO RT TRANSFER FORMAT

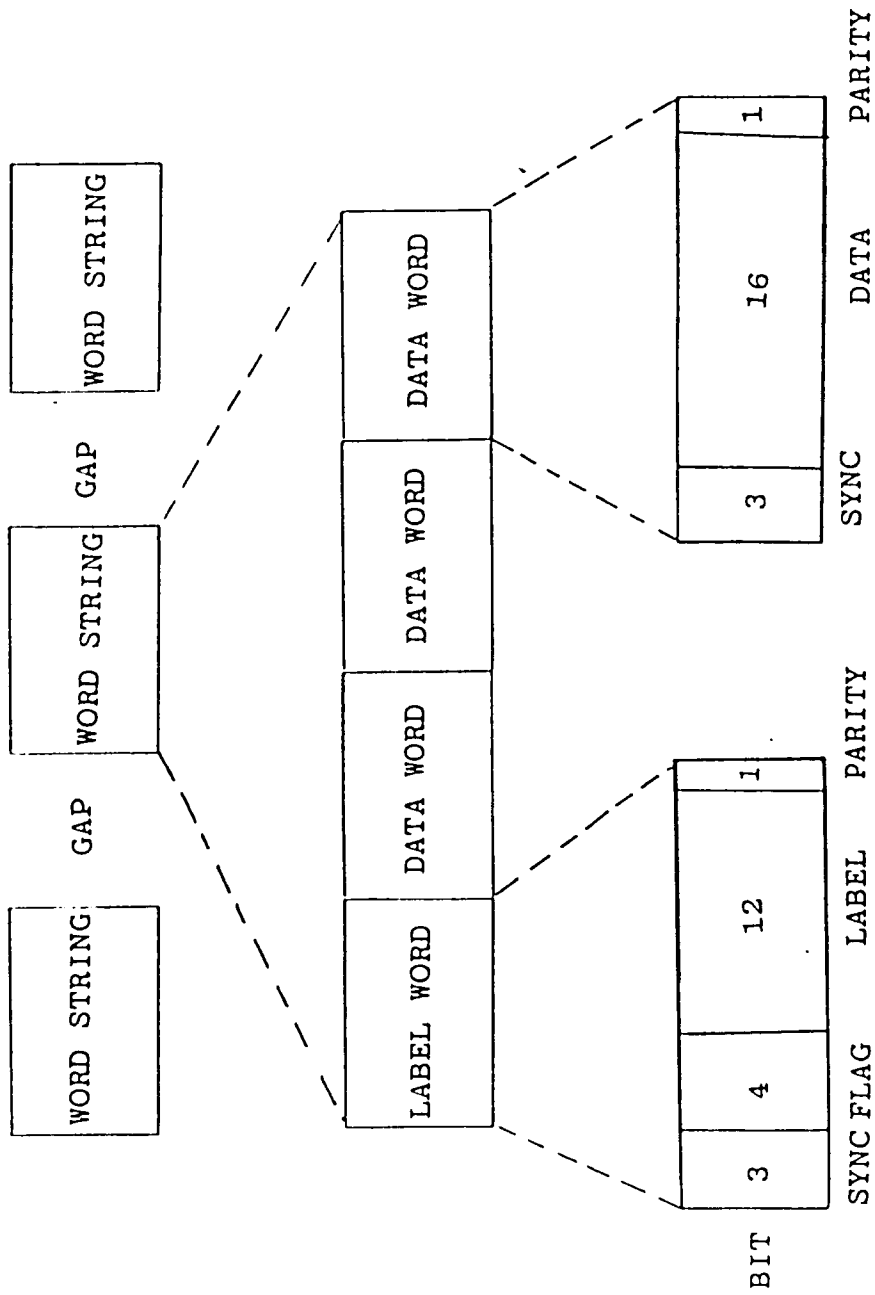


FIG. 8 DATAC MESSAGE FORMAT

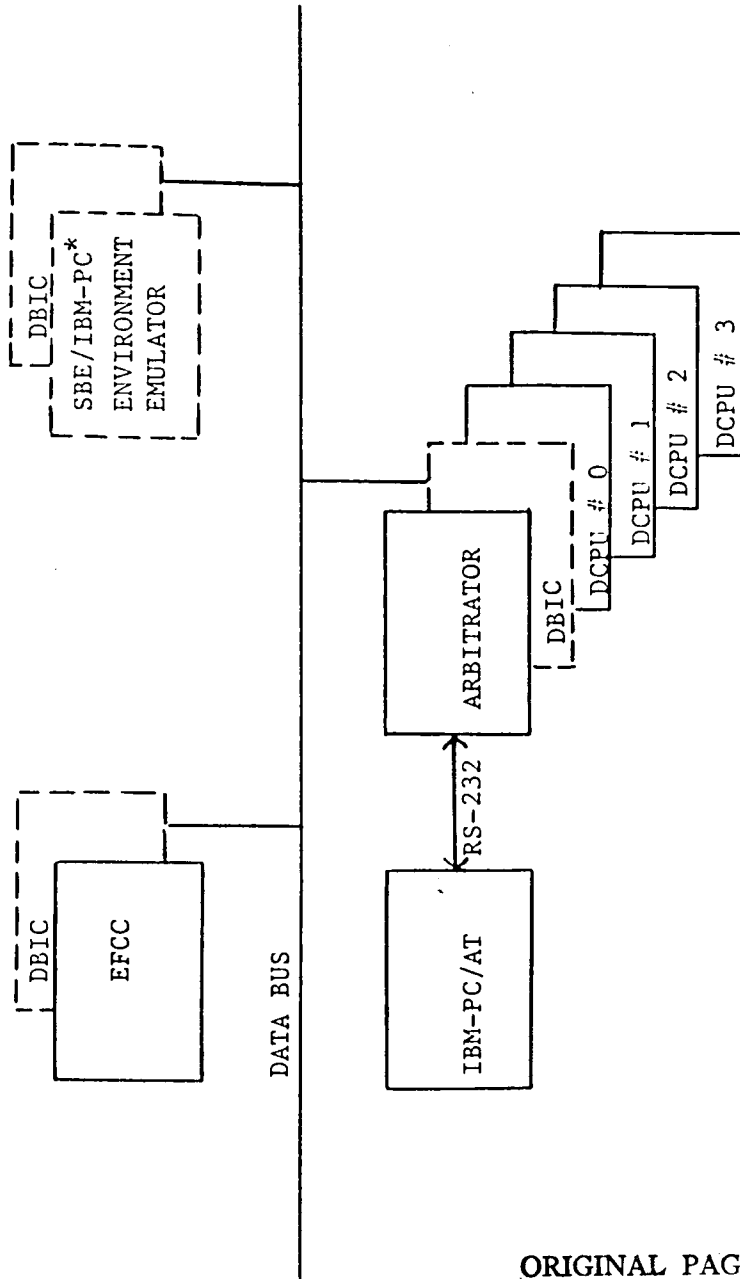


FIG. 9 IRAS ENVIRONMENT FOR BUS EVALUATION

□ : CURRENTLY INCLUDED IN THE IRAS

□ : ADDITIONAL REQUIRED

EFCC : EMULATED FLIGHT CONTROL COMPUTER

DBIC : DATA BUS INTERFACE CARD

* : IBM-PC OR SBE SINGLE BOARD COMPUTER

ORIGINAL PAGE IS
OF POOR QUALITY

FUNCTIONS		DATA BUS	MIL-STD-1553B	BOEING DATAC	IEEE ETHERNET	SPERRY HSIS
BUS CONTROL	CENTRALIZED BUS CONTROLLER		YES (BC/RT)	NO	NO	NO
	AUTONOMOUS BUS CONTROLLER		NO	CSMA/CA	CSMA/CD	TOKEN PASSING
	ACCESS CONTROL		DETERMINISTIC	DETERMINISTIC	PROBABILISTIC	DETERMINISTIC
MESSAGE TRANSFER	BROADCAST WITH ACKNOWLEDGEMENT		NO	NO	NO	YES
	BROADCAST WITHOUT ACKNOWLEDGEMENT		YES	YES	YES	YES
	TERMINAL TO TERMINAL WITH ACKNOWLEDGEMENT		NO	NO	NO	YES
	TERMINAL TO TERMINAL WITHOUT ACKNOWLEDGEMENT		YES	NO	YES	YES
MESSAGE CHARACTERISTICS	# BITS PER DATA WORD (OVERHEAD/DATA)		4/16	4/16	0/16	0/16
	# DATA WORDS PER MESSAGE (MIN/MAX)		1/32	1/256	23/750	0/4096
	ENCODED PRIORITY		NO	NO	NO	YES
	BUS CONTROL BITS		YES	NO	NO	YES
	ERROR FLAGS		YES	NO	NO	YES
	SOURCE/DESTINATION ADDRESS (# OF BITS)		5/5	NONE/12	48/48	16/16
FAILURE DET./REP.	SELF TEST		YES	YES	YES	YES
	ERROR STATUS REPORT		UPON REQUEST	NO	NO	AUTOMATIC
MOD. FOR ADD NEW TERMINALS			BUS CONTROLLER	COMMUNICATING TERMINALS	COMMUNICATING TERMINALS	COMMUNICATING TERMINALS
HARDWARE SPEC.	MEDIA		WIRE/FIBER OPTIC (W/FO)	W/FO	W/FO	FO
	MAX # OF TERMINALS		32	128	256	128
	MAX BUS BANDWIDTH (MBPS)		1	2	10	100
	MAX REDUNDANCE LEVEL		DUAL	DUAL	SIMPLEX	QUAD
	GLOBAL TIME REFERENCE		NO	NO	NO	YES

TABLE 1 - BUS COMPARISON CHART

DATA BUS	1553B	DATAC	ETHERNET	HSIS
1 WORD * 16 BITS (16 BITS)	72%	54%	97%	84%
1 WORD * 32 BITS (32 BITS)	64%	38%	94%	72%
2 WORDS * 32 BITS (64 BITS)	39%	23%	89%	57%

TABLE 2 - TRANSMISSION OVERHEAD

ORIGINAL PAGE IS OF POOR QUALITY

1. Report No. NASA CR 177458		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle An Evaluation Plan of Bus Architectures and Protocols using the NASA Ames Intelligent Redundant Actuation System				5. Report Date August 1987	
				6. Performing Organization Code	
7. Author(s) P. De Feo and M. Chen				8. Performing Organization Report No.	
9. Performing Organization Name and Address SPARTA Inc. 23041 de la Carlota, Suite 400 Laguna Hills, CA 92653				10. Work Unit No.	
				11. Contract or Grant No. NAS2-12081	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546				13. Type of Report and Period Covered Contractor Report	
				14. Sponsoring Agency Code 307-04-02	
15. Supplementary Notes Point of Contact: Technical Monitor, K. C. Shih, MS 210-5 NASA Ames Research Center (415) 694-6687 Moffett Field, CA 94035 FTS 464-6687					
16. Abstract Means for evaluating data bus architectures and protocols for highly integrated flight control system applications are needed. This report describes the criteria and plans to do this by using the NASA/Ames Intelligent Redundant Actuation System (IRAS) experimental set-up. Candidate bus architectures differ from one another in terms of: Topology, access control, message transfer schemes, message characteristics, initialization, data flow control, transmission rates, fault tolerance, and time synchronization. The evaluation criteria are developed relative to these features. A preliminary, analytical evaluation of four candidate data busses (MIL-STD-1553B, DATAC, Ethernet, and HSIS) is described. A bus must be exercised in a real-time environment to evaluate the dynamic characteristics. A plan for real-time evaluation of these four busses by using a combination of hardware and simulation techniques is presented.					
17. Key Words (Suggested by Author(s)) Data bus, flight control system, redundant actuation system, fault tolerant systems			18. Distribution Statement Unclassified - Unlimited STAR Category 05		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 32	22. Price*