IMPLEMENTATION OF A *DISTRIBUTED BUS ARCHITECTURE* ON A TRAINER SYSTEM

BRIAN D. CROUCH
ROBERT A. SEVERINO
HUGHES AIRCRAFT COMPANY
GROUND SYSTEMS GROUP
FULLERTON, CALIFORNIA

ABSTRACT

Trainers with a centralized architecture have inherent limitations on expandability, reconfigurability and ease of integration. The motivation for this project stems from the inability of the centralized architecture to address the complexity of individual, subteam, and team communication requirements for future trainers. With recent advances in commercial implementations of local area network technologies, trainer systems can be designed to overcome these limitations and offer other advantages. This paper discloses the research, selection, integration and demonstration of a distributed bus architecture for trainer systems. First in the paper, an analysis of top-level trainer system requirements is given from which specific bus performance characteristics are derived. An in-depth discussion of the network selection process includes an explanation of the network selection criteria along with the specific characteristics of several comparable local area networks. Next, a description is given of the implementation of a commercial fiber optic ETHERNET network. The upper layers of network protocol software are presented. This bus system was demonstrated as a validation of research findings. Data throughput rates, derived from both theoretical predictions and laboratory measurements, are disclosed for typical trainer configurations. A performance model written in BASIC, which allows predictions of the bus performance based upon user input of trainer configuration, is described and will be distributed to interested parties.

COMPARISONS OF CENTRALIZED AND DISTRIBUTED ARCHITECTURES

Overview of Existing Architectures

Trainer system architectures have been experiencing an evolutionary change from centralized processing designs to distributed processing designs. Past and current trainer systems have had a rigid architecture sized to the particular application. Trainers which use the distributed bus architecture are the next step in trainer evolution and will complete modularization of the individual functional areas, such as student station and instructor station processing, within the system. This approach will give military customers a cost effective trainer requiring minimum interfacing between functions. The improved trainer is flexible enough to incorporate the communication for additional functions after the system is initially integrated.

Figure 1 illustrates a centralized trainer architecture. With this approach, expandability is limited because of the system's complicated and rigid interface design. Figure 2 represents a trainer design using the distributed bus architecture. Options, including instructor stations, additional student stations, and peripherals, can be easily added to the trainer after system integration and installation.

Expandability

One advantage of the distributed approach is expandability of architecture. In the traditional schemes, the number of student stations which could exist in a training system was determined by the upper limits of the processing speed, memory capacity, and I/O capability of the central processor. Since this rigid configuration of student stations had to be established early in the system design phase, expansion of the trainer to include even one more student station was impractical after delivery of the system to the customer. The new approach partitions each student station into a subsystem containing both hardware and software elements. Expansion after the original system integration is easily accomplished by adding student station subsystems onto the local communications network. The communications network should be a high speed interface which supports a large number of nodes, on the order of one hundred. Training peripherals such as voice synthesizers and video discs can also be added easily after system integration using the modular approach, since each peripheral would have its own internal processor and network interface.

Reconfigurability

In the centralized design, functions for each node must be established in advance. Using the distributed approach, since there are no central nodes, functions may be changed on line. For example, should an instructor station fail under the distributed approach, another node may be dynamically allocated to perform that function. In the centralized approach, communications from one student station to another must go through the central processor adding overhead which slows down the whole system. When a bus is used, each node is fully interconnected

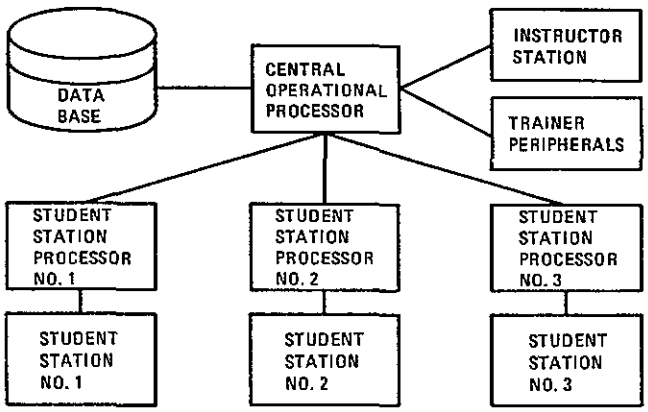


Figure 1. Centralized Trainer Architecture. This design utilizes a central operational processor to direct all activities within the system.

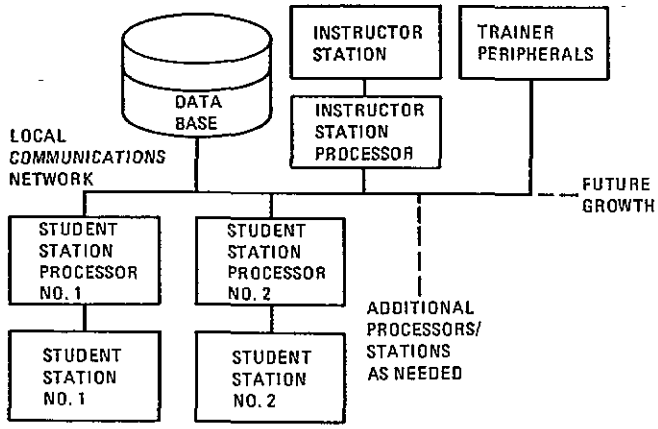


Figure 2. Distributed Bus Architecture. Decentralized processing nodes are utilized to achieve a flexible system architecture.

to every other node eliminating overhead and speeding up transmission times. An additional advantage of the distributed approach is that peripheral nodes, such as speech synthesizers, can be shared by all other nodes without incurring the overhead of going through a central processor.

Reliability

Another problem of the centralized architecture is its central point of failure. In this design, the system reliability greatly depends on the Mean-Time-Between-Failure (MTBF) of the central processor (typically one thousand hours). The only common element in the distributed approach is a passive bus which has a MTBF which is several orders of magnitude higher than that of the central processor. System testing with a failed central processor is impossible in the centralized approach, whereas student-to-student interfaces may still be tested in the distributed approach.

Ease of Integration

The system architecture using the centralized approach contains complicated hierarchical connections in the central node. For example, there may be priority problems depending on the relative positions or jumper configurations of the interface cards connecting the student stations. The distributed approach is simple to integrate because no node has more than one interface to the rest of the system. In addition, there are no hierarchies of communications paths, since each node is directly connected to a shared communications medium.

Vendor Interfaces

It is difficult to connect devices from different vendors in the centralized design because of their often differing interface protocols. Protocols which have been standardized, such as RS232, are too slow for most trainer interfaces. High speed Direct Memory Access (DMA) interfaces between different vendor's equipment are limited in number. By comparison, under the distributed approach, hundreds of vendors support local area network standards and offer interfaces to main frames, minicomputers, microcomputers, graphics systems, terminals, and other peripherals.

Team Interconnection

In the centralized approach, it is difficult to mix simulation of different prime systems or several teams within the same trainer because the hierarchical structure does not allow efficient communication among all elements. By comparison, the distributed bus architecture offers the unique capability of interfacing different groups of nodes through intelligent addressing, direct communication, and the lack of any central node overhead.

Cost

The central processor requires much greater processing capability because of the overhead incurred with the communications monitoring and control function. Using the distributed approach, however, no processing is required within the computer for communication due to the intelligence contained within the interface to each node. This distributed approach interface, which is similar in cost to the centralized approach interface, eliminates the need for extra processing capability in the central processor. A comparison of the number of interfaces required for each design shows that for one instructor station and N student stations, there is a requirement for 2N interfaces using the centralized approach. Only N + 1 interfaces are required for the distributed approach.

In the centralized design, the central operational processor is typically different from the other processors in the system. This requires logistics for at least two sets of technical manuals, additional training, maintenance support, and spares. There is a greater chance that all distributed processors will be of the same type, thereby saving half of the logistics costs. A final point is that the vendor-specific point-to-point interfaces of the centralized architecture have gone up in price over the years. Meanwhile, industry standardization and heightened competition have driven

down the prices of distributed bus products, and have contributed towards providing higher quality and greater capability.

REQUIREMENTS ANALYSIS

Architectural Considerations

Local communications networks can be divided into five classes: (a) point-to-point, (b) star, (c) shared memory, (d) ring, and (e) global bus. Their configuration and characteristics are illustrated in Figure 3 and described below.

a. The point-to-point architecture provides for complete interconnection of all processors. Although this architecture provides high data throughput, it is very expensive to implement due to the very large number of hardware elements; it is difficult and expensive to expand; and its complexity makes it more prone to failure and difficult to repair.

b. The star architecture is more practical to implement. This is a good architecture for systems with a central processing unit with several other processors slaved to it. But for training systems we want the capability for each node to communicate directly with any other node. The data throughput between nodes not directly connected is low, and the processor in the center of the star is a potential single point of failure. This approach is also difficult to expand because each added node requires additional hardware in the central node.

c. Shared memory is good for a small number of processors with closely coupled applications, but this configuration is impractical and expensive for typically more than four processors. Also, the shared memory is a single point of failure.

d. Ring structures are easy to expand. But, this architecture requires that data be passed through each node between the source and destination node, resulting in low throughput. Also, since each node must handle all the data, each node is a potential single point of failure.

e. The global bus provides efficient interprocessor communication since any node can communicate directly with any other node. Expansion is accomplished by placing another node on the interface medium, which is typically passive cabling. This architecture is less prone to failure than the other four described above.

The global bus architecture was chosen because of its expandability (new nodes simply tap into the communications cable), reliability (simple interface and message flow), and throughput (messages are sent from the source node directly to the destination node).

Detailed Requirements

The next step was to define the desirable characteristics of the global bus. The most obvious characteristic is whether data is transferred by a serial or parallel bus. Typically, serial buses require that extra control bits be placed at the beginning and end of a message, whereas parallel buses have control lines in parallel with data lines. However, serial data buses transmit data in a stream of self clocking bits while parallel buses require a handshake with each message that is transferred. The result is that in some cases a parallel bus will have a higher throughput over relatively short distances. But, as the physical transmission distance increases, the data rate decreases. This is because the transfer of a message on a parallel bus requires that the handshake signals make a couple of round trips on the bus.

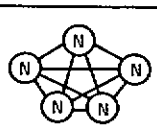
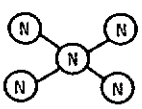
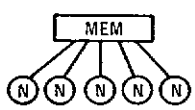
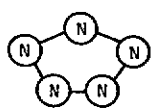
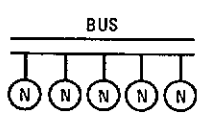
	ARCHITECTURE				
	(a) POINT-TO-POINT	(b) STAR	(c) SHARED MEMORY	(d) RING	(e) GLOBAL BUS
					
CRITERIA					
THROUGHPUT	EXCELLENT	FAIR	EXCELLENT	FAIR	GOOD
RELIABILITY	POOR	FAIR	FAIR	FAIR	GOOD
EXPANDABILITY	POOR	GOOD	POOR	GOOD	GOOD

Figure 3. Comparison of Five Basic Local Communication Network Architectures. Although the global bus does not have the highest throughput, it has the best overall characteristics and is the architecture selected for the distributed bus architecture.

Serial buses provide a higher degree of reliability than parallel buses since, effectively, only one signal line is required to support the communications network. This fact and the degraded performance characteristics and increased cost of parallel buses with increasing bus length are the reasons why it is preferable to use a serial global bus.

For the trainer applications studied, the physical bandwidth of the bus should be on the order of 10 Mbps. However, the physical bandwidth of a bus should not be confused with its data throughput. Generally, it takes more time to gain control of a bus and connect it with the destination than it does to send the data. The bus access scheme and the message addressing scheme have as great an impact on data throughput as the physical bandwidth of the bus. These two features are described below.

Bus access schemes fall into basically two categories: demand access and polling. Demand access is a scheme in which any node can transmit data on the bus when it detects that the bus is not busy. If two or more nodes attempt to transmit data simultaneously, interference results (referred to as a collision), the nodes detect the collision, then terminate their transmissions. Each node sustains a time delay (randomly determined or predefined) before retransmitting.

Under demand access, as the offered load from each node approaches the physical bandwidth of the bus the number of pending transfers increases, increasing the number of collisions and decreasing the throughput of the bus. The loads offered in typical trainer systems, however, do not even approach this overload point. Another problem with this scheme is that there is no guarantee of access to the bus (within a very short time span). However, based on the performance of the test bed bus and as predicted through simulation, all nodes in a typical trainer receive sufficient access with over 80% of the bus capacity available as spare. Finally, a transmission using a demand access scheme has to be long enough to fill the entire bus so that a node can be certain its transmission was not interfered with. This means that the demand access scheme is inefficient at transferring small messages.

Polling involves checking with each node on the bus in some predefined order. When a node is found that has data to transmit, it is granted access to the bus. On parallel buses, polling is implemented by special control lines. On serial buses, polling is accomplished by passing a special control word (called a token) between nodes. The node possessing the token is allowed to transmit data.

The disadvantages of polling are that it requires more circuitry to implement and, in some cases, its control is centralized in one node. It is preferable to have the control of bus access distributed among the nodes, avoiding a potential single point of failure, but introducing complex recovery procedures should the controlling node fail. Based upon the described advantages and disadvantages of the polling and demand access schemes above, it was decided that either a demand access or a polling bus access scheme with distributed control was the best approach.

The message addressing scheme (destination of a message) is usually defined by an address preceding the data . Some buses have addresses physically assigned to nodes, and only the addressed node will receive the message. Other buses have multicast addressing in which an address can specify any combination of nodes as the destination. Broadcast addressing is a special case of multicast addressing in which all nodes are selected as the destination. Multicast addressing is a desirable and important feature. If one node wanted to send the same data to ten destinations, a bus without multicast addressing would require ten times longer to transfer the data than would a similar bus with multicast addressing.

Most buses perform some type of data validation that involves parity checking or use of a check character. Either method is acceptable. Check characters require less bus bandwidth on large blocks of data. Parity checking is more efficient on small amounts of data.

In summary it was decided that the global bus should have serial data transfer (better performance over longer distances than a parallel bus), high speed (physical bandwidth on the order of 10 Mbps), low cost, distributed control, data validation (parity or check characters), and broadcast/multicast addressing (messages may be simultaneously sent to multiple destinations). The bus access scheme should be either demand access or polling.

SELECTION OF A DISTRIBUTED BUS ARCHITECTURE

The industry was surveyed and eight global buses were found which could potentially be used for trainers. We limited the comparison to buses which could support a minimum of 16 nodes over a total distance of 100 meters, had some form of data checking, were currently available or available in the near future, and had a bandwidth of at least 1 Mbps. Figure 4 shows how the eight buses studied compared to the criteria specified.

The ETHERNET local area network meets all of the requirements and was chosen to be the communications network for the distributed bus architecture. ETHERNET is a 10 Mbps, demand access, broadcast network allowing up to 1000 nodes to communicate over 2-1/2 kilometers of coaxial or fiber optics cable. ETHERNET is an industry leader with low cost interfaces available for many computers. The IEEE 802.3 Committee and the European Computer Manufacturers Association have developed local area network standards that incorporate ETHERNET.

Six companies were found within the industry which were producing ETHERNET interfaces: Interlan, 3Com, DEC, Associated Computer Consultants, Ungermann-Bass, and APH Consulting. The Interlan products were found to have the best combination of high performance, ready availability, low cost, and a direct computer interface. They possess a high degree of onboard intelligence and hardware resources which minimize the overhead imposed on the host computer.

FEATURES	GLOBAL BUS TYPE							
	ETHERNET	HSD MULTIDROP	HUGHES DATA BUS	HYPERCHANNEL	PCL11-B	SHINPADS	MIL-STD 1553	IEEE 488
INTERFACES VARIOUS BRANDS OF COMPUTERS	X			X			X	X
SERIAL	X		X	X		X	X	
POLLING		X	X		X	X	X	X
DISTRIBUTED CONTROL	X	X	X	X				
MULTICAST ADDRESSING	X		X			X		
EFFICIENT TRANSFER OF SMALL MESSAGES			X		X	X	X	X
EFFICIENT TRANSFER OF LARGE MESSAGES	X	X	X			X		
LOW COST	X				X			X

Figure 4. Comparison of Global Bus Characteristics

Five semiconductor companies are integrating many of the ETHERNET interface board functions onto VLSI controllers which is resulting in even lower cost/higher performance products.

Since the initial survey, the ETHERNET/IEEE 802.3 network has become the dominant force in the local area network market place. There are today over 25 companies building ETHERNET interfaces to over 300 computer systems. Even if a dedicated interface does not exist for a given device, the device may be integrated into an ETHERNET network using one of the ETHERNET-to-RS232 or ETHERNET-to-8-bit-parallel type devices. Network protocol and applications software packages are available for many computers and are also becoming standardized. Refer to Appendix 1 for a partial listing of the ETHERNET interfaces available today.

IMPLEMENTATION OF A TEST BED PROTOTYPE

Hardware Configuration

For the test bed prototype two in-house computers, a DEC MINC23 and a DEC PDP 11/24, were integrated into an ETHERNET network, as shown in Figure 5. To connect a computer to the network, an interface board, as shown in Figure 6, is plugged directly into the computer backplane bus. The interface board, which is controlled by a software driver, takes a message in the host computer memory and transmits it onto the network. Multiple receiver buffers and address recognition capability ensure proper message reception. A transceiver unit electrically connects the interface board to the ETHERNET cable. The ETHERNET cable may consist of coaxial cable and/or fiber optics cable. A fiber optics ETHERNET requires the addition of an optical star coupler and the use of a different transceiver. The advantages of a fiber optic cable include greater security, longer cable segments without repeaters, and EMI/RFI immunity for about the same cost as coaxial cable. One additional benefit resulting from the use of ETHERNET is peripheral sharing - a single input/output or mass storage device may be shared by many computers over the network. The MINC23 was able to access the high capacity disk drive, high speed printer and mag tape unit on the PDP 11/24.

Network Interface Software

Next, the software necessary to interface a trainer system to ETHERNET was generated. This software consisted of two parts. The first part is a driver, provided by the ETHERNET controller vendor, which allows application programs running on the host computer to make standard operating system calls to control the operation of the network interface board. The second part of the network interface software allows trainer system users to access high level network control functions directly from the trainer system software. The majority of the network control functions are transparent to the user. Services provided to the trainer system user include: address management, message prioritization, file transfer, message formatting, flow control, end-to-end data assurance, diagnostic functions and device sharing.

A structured, hierarchical model for data communications protocols has been developed by the International Standards Organization (ISO). However, commercial implementations of the entire model are oriented to networks which have many diverse devices spread out over a wide geographic area on many individual local area networks. The special requirements of trainer systems caused our network interface software design to be based

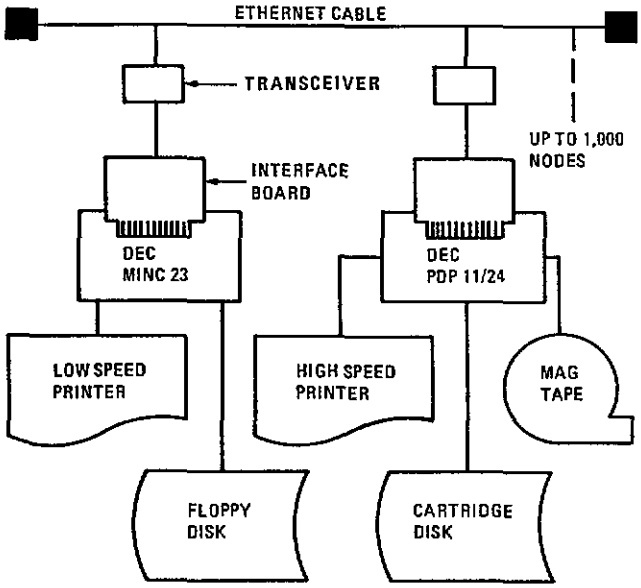


Figure 5. ETHERNET Test Bed Block Diagram. Use of an ETHERNET local communications network allows high speed, low cost and low overhead data communications. The network can span 500 meters of cable without repeaters and up to 2500 meters with repeaters, while still maintaining a 10 Mbps data rate.

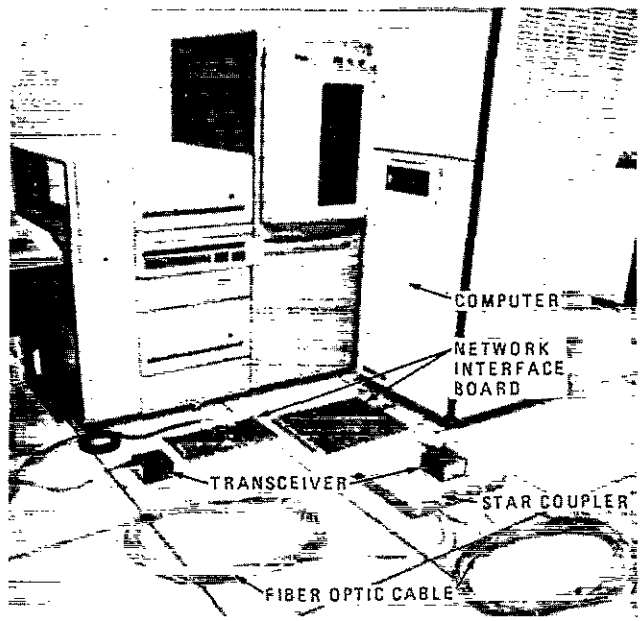


Figure 6. Typical Fiber Optic ETHERNET Components. Fiber optic and/or coaxial ETHERNET cables may be used for about the same cost. Advantages of fiber optics include EMI/RFI immunity, longer cable segments, and greater security.

upon a subset of the ISO model. Advantages gained by this approach include higher performance (less overhead on the host computer) and the ability to implement special features like team exercise setup, exercise download and message prioritization.

In order to support a high effective data throughput and impose little overhead on the exercise designer, instructor, or the trainer system software, a carefully designed network interface software package is being developed. Trainer system users may access high level network control functions through Fortran or operating system calls. The services provided to trainer system users are highlighted below.

- Address Management. Logical names (Gunner #6, Instructor #2, Team #1, All Maintenance Students, All Students, etc.) provided by the user are mapped onto physical and multicast network addresses.
- Message Transmission/Reception. Single commands are used to request transmission of messages assembled in memory and to provide buffers for message reception. Messages that are larger than the maximum size permitted on the network are automatically divided into smaller pieces for transmission and are reassembled at the receiving end.
- Message Prioritization. Users wishing to transmit a high priority message may request all other users to cease transmissions for a specified amount of time.
- File Transfer. Trainer data base, exercises, and system software files may be reliably transferred throughout the network. Common data and software may be maintained at a single location if desired.
- Flow Control. Should a node ever experience a critical traffic load, it will allocate additional buffers and/or abort low priority messages, as necessary. Similarly, should the entire network ever experience critical traffic loads, low priority messages will be queued for later transmission.
- End-to-End Data Assurance. Users are guaranteed reliable transmission of messages by a user-transparent system of acknowledgements and automatic retransmission requests for messages received with errors.
- Diagnostics. The network interface boards collect extensive network statistics. Users may obtain the operational status of the entire network or a particular node at any time by the use of status requests, loopback tests, and other maintenance aids.
- Resource Sharing. Disk drives, printers and other peripherals may be accessed by any computer on the network, thus reducing trainer system hardware costs.

Testbed Network Performance Measurements

In order to determine the overhead and throughput capabilities of our test bed network, a timing study was performed. Using our PDP 11/24 computer, a low-end minicomputer, messages were transmitted onto the network continuously -- as fast as the operating system (RSX-11M V4.0) and ETHERNET controller (Interlan NI1010) would allow. The results are shown below:

Data Bytes/Message	Number of Messages Transmitted/Second	Effective Data Throughput
46	174	64,032 bits/Sec
200	159	254,400 bits/Sec
1500	86	1,032,000 bits/Sec

Note that these results are for a single node transmitting onto the network. Full utilization of the network occurs when many nodes interleave their transmissions. The operating system, CPU execution speed, and ETHERNET controller processing speed limit the number of messages that can be transmitted, as shown by the 46 data byte case (much less data yet only twice as many messages as the 1500 byte case). Typical trainer computers are medium-power minicomputers, and thus are able to drive the network interface at a much greater speed. In addition, most effective use of the bus bandwidth is achieved by combining smaller messages into larger messages.

The results of this study were fed into a computer simulation of the ETHERNET Network, as described in the following section.

PREDICTED PERFORMANCE OF A TYPICAL TRAINER SYSTEM

In order to predict the performance of future trainer system configurations and traffic loads, a computer simulation program of ETHERNET was designed and implemented. This simulation program was written in BASIC and will be given to interested parties. Program users enter traffic load and response time characteristics for any number of nodes, allowing trainer system designers to model the communications performance of any trainer configuration. Timing studies performed on the ETHERNET testbed configuration allowed validation and fine tuning of the model.

This simulation model was used to predict the performance of a trainer consisting of 42 student stations and 10 instructor stations. The following results point out the high performance obtained from using an ETHERNET-based trainer system architecture:

- Exercise download of the training programs to each student station was accomplished in well under one minute. This process takes into account the assignment of exercises by ten different instructor stations to their respective students.
- Even during the busiest second in the exercise, an effective data throughput of over 1 Mbps was maintained.
- The maximum bus utilization was only 9.8% allowing over 90% of the bandwidth available for future growth.
- The network traffic included one built-in wraparound test from each instructor station to each of the respective student stations once per minute.
- The data throughput also included integrated team data being exchanged once per second among each of the team members.

SUMMARY

State-of-the-art approaches to implementing local communications networks were investigated, and the global bus architecture was selected as the best approach. The various characteristics of global buses were studied, and the features which would produce a high degree of reliability and a high effective data throughput were determined. The characteristics desired in the bus were compared with implementations currently available or under development.

ETHERNET was selected as the best network for the distributed bus architecture. An ETHERNET network was implemented, allowing network interface software development and performance studies to be done. The network interface software developed is a generic, highly intelligent software package allowing the exercise designer and instructor to fully utilize the network capabilities with minimal overhead and high performance. A model of ETHERNET was developed allowing trainer system designers to predict the communications performance of future trainer systems. Based on theoretical predictions and laboratory measurements, an ETHERNET-based trainer system architecture allows reliable, low-cost communications between all the elements of a typical large-scale trainer system.

In conclusion, the expandability, reconfigurability, reliability, ease of integration, number of vendor interfaces, team interconnection, and low costs of the distributed bus architecture make this a desirable design for both the developer and the user.

ACKNOWLEDGEMENTS

The authors wish to thank Bradley M. Seltzer, Systems Engineering Group Head at Hughes Aircraft Company, for his invaluable technical guidance, assistance in the preparation of initial drafts of this paper, and for his ETHERNET simulation program.

Appendix 1. Commercially-Available Ethernet Interfaces
(Partial Listing)

Computer/Backplane Bus	Vendor																			
	Interlan	3-Com	Ungermann-Bass	Associated Computer Consultants	Communications Machinery Corp.	Excelan	Bridge Communications	DEC	Able Computers	Texas Network Systems	Perex	Tecmar	Intel	EnLink	HP	Xerox	Gould/SEL	AT&T	Codenoll	Honeywell
UNIBUS	X	X		X	X	X		X	X											
QBUS	X	X			X	X		X	X											
MULTIBUS	X	X			X	X	X						X							
VMEBUS					X	X														
VERSABUS				X	X															
Gould/SEL																	X			
IBM Channel				X																
IBM Bisync Devices							X													
IBM PC, compatibles	X	X	X									X								
Data General Nova, Eclipse, MV	X																			
DEC Personal Computers	X																			
HP 9000, 150, 3000															X					
S100 Bus											X									
Apple II	X													X						
NCR Tower						X														
Xerox microcomputers, word processors, work stations, file servers, print servers																X				
Fiber Optic Ethernet			X																X	
Bridge to Sytek Broadband Net							X													
AT&T 3B Series																		X		
Honeywell DPS 6, 8, 88																				X
RS232 port	X		X				X			X										
RS449, IEEE 488, 8-16-32 bit parallel, V.35 interfaces			X																	
X.25				X			X													
8-bit Parallel interface	X	X																		

ABOUT THE AUTHORS

Mr. Brian D. Crouch is a Systems Engineer with the System Engineering and Support Department, Engineering Services and Support Division of Hughes Aircraft Company. He is currently responsible for systems engineering and software development for trainer systems research and development projects on local area networks and computer graphics. He is also responsible for systems engineering and system integration for the Position Location Reporting System Trainer. Previously he was responsible for systems engineering for the Remotely Piloted Vehicle Institutional Trainer. Earlier he was the Project Engineer for the Distributed Modular Trainer IR&D project. Mr. Crouch holds a B.S. from the College of William and Mary and is pursuing a Master's degree in Computer Engineering at the University of Southern California.

Mr. Robert A. Severino is Group Head, System Engineering and Support Department, Engineering Services and Support Division, with 7 years experience in training technology: currently coordinator for research and development on trainer systems and director of IR&D efforts in this area. *Performed the proposal management* function for the Remotely Piloted Vehicle Institutional Trainer and for sonar operator, maintenance and tactical trainers. Previously, performed system design for projects monitored by the Program Management Office and was system engineer on the Roland Institutional Trainer Program and technical director on a trainer design study. Earlier, performed exercise *creation, software testing, and system* integration for the Firefinder Trainer, and conducted NTDS maintenance training for a foreign military customer. BSE, 1977, Harvey Mudd College.