MICROPROCESSORS IN AIRCREW TRAINING DEVICES

Richard J. Sylvester, President
Systems Productivity & Management Corporation
Dayton, Ohio

ABSTRACT

Microprocessors are already embedded in Aircrew Training Devices (ATDs) for display generation, input/output control, and other special purpose applications. This paper deals with the key factors relating to the use of microprocessor systems (rather than minicomputers) in performing the central computational function for ATDs. Such factors include key performance parameters to be considered when replacing a 32-bit minicomputer by a system of microprocessors; requirements for a recent Air Force ATD system; benchmarks for computational performance; and specific changes to ATD prime item development specifications.

INTRODUCTION

To limit the scope of technical activity to reasonable bounds, the C5A/C141 Aerial Refueling Part Task Trainer (ARPTT) was chosen as an example for which micros might be used. A recent draft Request for Proposal for this system was used as guidance in this study. Data from comparable systems were used to estimate the C5A/C141 ARPTT computaton requirements. A "sizing" exercise based on the B-52 ARPTT using micros to satisfy computational requirements was performed. This exercise plus considerable information from microprocessor vendors provides the bulk of the data presented here. Such technical and performance data are summarized.

KEY PERFORMANCE PARAMETERS

The key parameters for distributed microprocessor systems are similar to those for minicomputer systems. These are discussed below.

1. Floating Point. Real-time simulator performance requires mathematical computing in floating point to solve the large, highly accurate flight and aerodynamic equations. Because of the precision, accuracy, and numerical range of these equations, the numbers involved require 32 bits or more of representation. Thus, the arithmetic computing elements must have adequate execution times. Software floating point implementations require a fairly large number of instructions which result in long execution times. Hardware floating point implementation requires only a few instructions with the actual floating point calculation being done in hardware. The hardware performance floating point is approximately 100 times faster than a software implementation of the same functions.

2. Data Bus Considerations. In single bus systems all commmunication activities must use the system bus resource. As the microprocessor system size and performance levels increase (i.e. as more microcomputers are added to the bus) to meet larger application requirements, the reserve bus bandwidth becomes depleted. Eventually, as the system computational capabilities and resources are increased, the total system performance will decrease due to inadequate bus capacity.

In systems where bus control exchanges are synchronized to a common bus clock, no data will be lost or destroyed when system bus reserve bandwidth is exceeded. What will happen is the system functions demanding system bus resources will have to wait longer periods of time to be serviced. To alleviate this situation specially dedicated buses may be added to relieve the single bus congestion.

Memory buses independent of, but coupled to, the main system bus by dual ported memories play a large role in increasing the main system bus effective capacity. This is extremely important where single task partitioning among multiple microprocessors is required to obtain specified functional performance levels.

Multichannel, I/O buses independent of, but coupled to, CPU local buses, local memories, I/O peripherials, and the main system bus by dual ported memories and special I/O controllers are again useful in reducing the main system bus traffic.

Bus bandwidth is the computer system transfer rate capability to distribute data among various sources and destinations across the system bus or buses. It is essentially the lifeline of the computing system. When exceeded by various demands, limited bandwidth degrades the system performance. The capability to determine the reserve bus bandwidth is an essential requirement in the evaluation and design phases for specific distributed microprocessor system applications.

3. Execution Times. How useful are microprocessor execution times in evaluating total system performance? System performance estimates depend very strongly upon what tools and methods are used to measure and analyze the execution times.

One common measure of microprocessor execution performance is MIPS (Millions of Instructions Per Second). The MIPS value states only the number of instructions that can be executed in a unit time. "MIPS" ignores the type of instruction set, internal and external architectures, and the language implementation efficiencies to be employed for a specific system application. For example, a low MIPS microprocessor with a powerful instruction set and architecture may outperform a high MIPS one with many instructions and limited architecture. In this sense MIPS is not always a very precise indicator of a microprocessor's total system performance capabilities.

Benchmarks are special programs written to determine the performance of microprocessors within a given product line or between different manufacturers. Benchmarks can be a better technique than "MIPS" for evaluating microprocessor performances. The utility of a benchmark, however, depends upon the source of the benchmark and the purpose it is intended to convey. The Whetstone Benchmark was developed at the National Physical Laboratory, Teddington, England, to provide a meaningful way to evaluate scientific computational performance of machines with different architectures and language implementations. The Whetstone Benchmark has a good mix of both logical and mathematical instructions to provide an indication of different microprocessor performances in scientific applications.

Adequate benchmarks for all classes of ATDs do not presently exist due to the great design variability possible for the software for a given ATD implementation.

4. Memory Considerations. The direct memory addressing capability of a microprocessor has a considerable impact upon the total system capability and performance. The more extensive the addressing (coupled with proven virtual memory techniques and memory management functions), the more flexible and adaptable the microprocessor becomes, permitting a wider range of different types of applications. Adequate memory addressing reduces the burden on software engineers to design around memory limitations.

Memory usage can be categorized into two types (local or private and global). Local memory capabilities should be used as much as practical to enhance the performance and expansion by reducing memory dependence and traffic on the main system bus. Local memory usage generally enhances memory access time by 300 to 400% over that of global memory requiring use of the main system bus. Microprocessors are now available that have local or private memory capabilities of up to 16 megabytes. These systems are dual ported which also permits partitioning of a given memory for local and global applications.

5. Operating Systems. Microprocessor operating systems have evolved from little more than hardware monitors to the current multitasking and multiprocessing types. The latest operating systems are as sophisticated and powerful as those for some large main frames. Furthermore, microprocessor operating systems are fully configurable to meet the system and application requirements. The newer microprocessors have many of the operating system elements in hardware to improve the performance. These hardware elements include task switching, memory management, and interrupt handlers.

Operating systems for micros must be configured at a much lower level of detail than those for minis. This activity requires a more detailed knowledge of operating systems on the part of the ATD contractor.

MEETING C5A/C141 ARPTT REQUIREMENTS

The commercial microprocessor products which have the potential to satisfy the specified C5A/141 ARPTT requirements are evaluated below. The requirements which must be fulfilled include the following: (1) all system components and packages are currently available in off-the-shelf form and on a plug-together basis, (2) multitasking and multiprogramming operating systems are complete and validated, (3) the multiprocessing operating system is complete and validated, (4) FORTRAN 77 and Assembly languages are available, (5) data paths are a minimum of 16 bits, and (6) there exist system bus hardware arbitration, upward compatibility and floating point capability. (See Table 1 below).

HARDWARE BOARD LEVEL VENDORS	COPROCESSOR FLOATING POINT AVAILABLE	SYSTEM BUS ARBITRATION	MEETS SPECS CITED	MICRO PROCESSOR MULTI-TASKING	MULTI-PROCESSING	LANGUAGES
INTEL	8087 287 NOW	YES PRIORITY MASTER/ MASTER	YES	RMX-86 RMX-88 RMX-80	MMX-800 -MMX-80 -MMX-86 -MMX-88	FORTRAN 77 COBOL PASCAL BASIC PL/M ASSEMBLY
MOTOROLA	MC68881 4Q 83	YES PRIORITY LEVEL	NO	RMS-86K	MSP/ 68000 4Q83	FORTRAN 77 COBOL ASSEMBLY
ZILOG	4Q83	NO	NO	ZEUS	NONE AS OF 1 DEC 82	FORTRAN ASSEMBLY
TEXAS INSTRUMENTS	1421 FIRMWARE (BIT SLICE) NOW	NOT COMPLETE	NO	RX	NONE AS OF 1 DEC 82	FORTRAN PASCAL BASIC ASSEMBLY
DIGITAL EQUIPMENT	FPF-11 NOW	YES MASTER/ SLAVE	YES	RSX-11S RSX-11M	NONE AS OF 1 DEC 83	FORTRAN ASSEMBLY BASIC COBOL

Evaluation of Vendor Capabilities
Programming Language and Operating System
Table 1-1

OPERATING SYSTEMS VENDORS	MULTI PROCESSORS (16 Bit)	LOCAL AREA NETWORK (Optional)	MEETS ALL SPECIFICATIONS CITED
INTEL	8080A (8 Bit) 8085A (8 Bit) 8088A 8086A 186 286 386 (32 Bit)	ETHERNET (10 Megabit) (100 Stations) (500 Meters)	YES
MOTOROLA	68000 Series (16/32)	NO INFO	YES
ZILOG	Z 8000 Series	Z Net	NO
TEXAS INSTRUMENTS	TMS-9900 1481 BIT SLICE SERIES	IBM SYSTEM	NO
DIGITAL EQUIPMENT	LSI-11 PDP-11 SERIES	ETHERNET DEC NET	NO

Evaluation of Vendor Capabilities
Programming Language and Operating System
Table 1-2

HARDWARE BOARD LEVEL ----- VENDORS	MASS STORAGE					
	HARD DISK TO 330 M BYTES	WINCHESTER TO 84 M BYTES	MAGNETIC TAPE 9-TRACK	FLOPPY DISK SYSTEMS	MAGNETIC TAPE CARTRIDGE TO 60 M BYTES	BUBBLE MEMORY TO 512 K BYTES
INTEL	SBC-220 YES	SBC-215 YES	Independent Vendors YES	SBC-204 SBC-208 YES	SBC-217 YES	SBC-251 SBC-254-4 YES
MOTOROLA	No Info As of 1 Dec 82	No Info As of 1 Dec 82	None Indicated	YES	NO	NO
ZILOG	None Indicated	None Indicated	Independ- Vendors	YES	NO	NO
TEXAS INST.	NO	Independent Vendors	NO	YES	NO	NO
DIGITAL EQUIP.	YES	NO	RTO Series	YES	YES	YES

Evaluation of Vendors Capabilities
Hardware & Peripherals
Table 1-3

1. Evaluation of Individual Manufacturers

1.1 Texas Instruments. As of February, 1983, T.I. did not have a multiprocessor operating system nor microprocessor floating point coprocessor available. There were no immediate development plans.

1.2 Zilog. As of February, 1983, Zilog had no multiprocessing operating system nor hardware floating point processor available. There were no immediate development plans.

1.3 Digital Equipment Corporation. As of February, 1983, DEC had a variety of hardware available but did not have a multiprocessing operating system for the LSI-11.

1.4 Motorola. Motorola has in development a multiprocessing operating system and hardware mathematics coprocessor with a release date of late in 1983.

1.5 Intel Corporation. As of early 1983, Intel was the only company investigated that had everything to implement a complete distributed microprocessor system meeting the ATD requirements. Also to aid in the development of a distributed microprocessor system, Intel has a wide variety of ICE (In Circuit Emulation) systems. Intel advertises total upward compatibility. All software developed for the 8086 microprocessor is upward compatible with the 8088, 80186, 80286 and 80386.

BENCHMARKS

This section highlights key technical evaluations; namely, hardware floating point and operating systems performance benchmarks.

1. Intel Benchmark. The data supplied for this benchmark was obtained from Intel Corporation to show the Whetstone floating point performance of the 8086/8087 and 80286/80287 microprocessor systems. The following table gives this data.

MICRO	MICRO CLOCK	COPROCESSOR	COPROCESSOR CLOCK	WHETSTONE
APX-286/10	8 mhz	80287	5mhz	150 KOPS
86/30 SBC	5 mhz	8087	5mhz	100 KOPS

Table ?

The Whetstone benchmark unit (KOPS) implies thousands of floating point operations per second. The APX-286/10 and the 86/30 are single board computers that may be purchased off the shelf.

The architecture of the above microsystems is such that the 8086/80286 micros perform logic and data access functions, while the 8087/80287 perform floating point calculations. The micros and their coprocessors operate in mixed concurrent and parallel modes.

2. Author's Floating Point Software cBenchmark (P.E. 8/32 Mini vs. Intel 286/287 Micro. A comparison is made between the floating point performance of a Perkin Elmer 8/32 minicomputer used in the B52 Aerial Refueling Simulator (1977 vintage minicomputer) to that of an Intel SBC 286-10 single board computer with a 287 mathematics coprocessor.

Two assembly language programs were coded implementing the same algorithm for both processors. The algorithm implemented is a randomly chosen flight equation used in the B-52 ARPTT. The form of the equation is:

$$Z = MgCos\ Theta\ Cos\ Phi + Zs\text{-}Cos\ X + Xs\ sinX + Zt$$

Neither of the two assembly language programs was run on its respective target machine; however, it is felt that these programs are coded sufficiently well for the purpose of calculating the required execution times.

The implementation of the equation used "table look up" techniques to assess the trigonometric functions where the address computations are not done in floating point. Consequently, the timing is represented in terms of floating point and table search as shown below.

	PE 8/32D	Intel 286/287
Floating Point	20 uSec	161 uSec
Table Search	107 uSec	311 uSec
Total Time	127 uSec	472 uSec

Table 3

The overall performance factor is 3.7 to 1 for this example. Note that the floating point time ratio is 8 to 1, while the table search is 3 to 1. For the PE 8/32D the Whetstone rating is 900 KOPS, whereas the Intel board is 150 KOPS giving a Whetstone performance ratio of 6 to 1. Consequently, the real performance ratio between the PE 8/32D used on the B-52 ARPTT and the Intel single board computer ranges between 3 to 1 and 8 to 1 depending on how much floating point computation is designed into the software. The key point here is that the details of the software design itself can influence substantially the benchmark comparisons. There is no way to accurately benchmark ATD computer performance without detailed knowledge of the software design approach.

3. RMX-86 Operating System Benchmark. This section contains a benchmark which relates the performance of a RMX-86 base operating system on two different Intel singe board computers. One implementation is using Intel's SBC-86/30 single board computer having a 8086 microprocessor. The other implementation is using an Intel SBC-286/10 board containing a 80286 microprocessor. The fundamental difference is that the 80286 microprocessor implements in hardware the following functions: memory management, task switching, and an interrupt handler. The 8086 microprocessor implements the above functions by means of RMS-86 operating system software.

The specifications for the benchmark are as follows: (1) 8086 microprocessor operating at 5 mhz, (2) program and data in local memory, and (3) software functions execution times are linear with respect to clock speed. As indicated in the performance ratio column of Table 4, it is important to implement as much as possible all operating system functions in hardware if speed is important.

OPERATING SYSTEM	FUNCTION ** IMPLEMENTATION	FUNCTION	MICRO-PROCESSING	CLOCK SPEED	EXECUTION SPEED	PERFORMANCE RATIO *
RMX-86	SOFTWARE	TASK SWITCHING	8086	5mhz	1.2mSec	1.0
RMX-86	SOFTWARE	TASK SWITCHING	8086	8mhz	750uSec	1.6
RMX-86	SOFTWARE	INTERRUPT LATENCY	8086	5mhz	240uSec	1.0
RMX-86	SOFTWARE	INTERRUPT LATENCY	8086	8mhz	150uSec	1.6
RMX-86	HARDWARE	TASK SWITCHING	80286	10mhz	17uSec	70
RMX-86	HARDWARE	INTERRUPT	80286	10 mhz	3uSec	80

* Based on 8086 Operating at 5mh Clock
x** Primary Implementation

Operating System Benchmark
Table 4

BUS BANDWIDTH

Distributed microprocessor bus performance is a complex subject. The author presents below a concise outline of the major considerations involved in bus performance.

1. Overhead Introduced by System Functions. The multitasking and multiprocessing operating systems are the major contributors to overhead times. The multitasking operating system affects the total system overhead at task execution level; that is for task switching interrupt handling, exception handling, error detection and correction, and data validation functions. The component overhead is due to the main system bus priority and bus arbitration control hardware.

The multiprocessing operating system and its priority level and bus arbitration hardware logic affect the main system bus overhead. The effects of the hardware overhead only become a serious problem when the software and system configuration cause an excessive number of main system bus accesses. This occurs when the design forces a large number of small data transfers between many single board computers resident on the main system bus, causing the overhead factor to become critical.

The multiprocessor system overhead is caused by main system bus interprocessor message transfers and system calls. If the software and system configuration design create the requirement for a large number of these functions, then the multiprocessing operation system overhead also can become a critical factor.

2. Multibus Interprocessor Protocol (MIP). The MIP specifies the implementation, environment and form of the multiprocessor operating system software and main system bus hardware. The MIP specifications permit intertask synchronization, message, and data transfers residing on different single board computers as easily as task switching on single board microcomputers. Communication between tasks can be either tightly or loosely coupled, depending upon the application requirements.

Loosely coupled tasks require only single messages to be sent to the receiving computer. Tightly coupled tasks provide synchronization by means of handshaking.

3. Optimal Considerations for Single System Bus. The following rules are configuration guidelines to aid in determining the performance level of distributed microprocessor single bus systems.

Each SBC microcomputer operates as independently as possible with the following local resources: local data memory, local ROM memory for program storage, local I/O system, and local specialized funcions such as floating point.

Main system bus bandwidth resources should be limited when possible to interprocessor and system data transfers and message communications.

When a single task is partitioned among multiple SBC microcomputers, application system design should be directed towards total system performance causing system bus bandwidth resources minimization and microprocessor efficiency maximization.

System bus parallel priority scheme and system application design can optimize system performance.

Data transfers when possible should be medium to large blocks.

ATD COMPUTATIONAL PERFORMANCE EXAMPLE

This section of the paper uses the computational characteristics of the B52 Aerial Refueling Part Task Trainer (ARPTT) as an estimate of the C5A/C141 ARPTT computational requirements.

1. B52 ARPTT Data. The B52 ARPTT was implemented using a single Perkin-Elmer 8/32D minicomputer in each trainer. From informal data used to track the development of the project software, Table 5 was derived. This table presents an estimate of computational performance at the time of completion of system development. The table lists the real time functions implemented in the B52 ARPTT, the memory requirements for data used by each function, the memory requirements for the executable instructions (code) and the execution times on the PE 8/32D.

It is clear by examining the data that the flight/motion and the visual display functions dominate memory and timing utilization for this ATD. These functions represent over 50% of the utilization of memory and execution time.

2. Sample Micro Layout for B52 ARPTT. Table 6 shows the B52 ARPTT real time functions distributed over a total of eight single board computers (SBC) each residing on the same single main system bus. In this example, the following assumptions are made.

A. The ratio of the PE 8/32D minicomputer to the Intel 286/287 microprocessor system performance is 6 to 1 as indicated by the Whetstone performance comparison.

B. Single board computer utilization at 50% or less was a design goal based on the uncertainty in the time execution and bus loads and based on the inexpensiveness

REAL TIME FUNCTIONS	MINICOMPUTER INTERDATA 8/32D DATA *	CODE *	TIME **
Supervisor	780	20	6.2
Instruction Aids	132	1,872	7.5
Host VDU.	13,760	9,960	142
Boom Control	1,408	3,656	17.6
Visual	984	4,680	80.4
Hydraulics	360	1,920	5.0
Fuel	3,320	5,380	49.3
AFCS	780	3,000	22.5
Engines	1,184	1,460	7.2
Flight/Motion	11,300	9,600	247.5
Control Loading	1,140	3,564	23.7
Sound	688	1,268	6.7
Test/Preflight	424	536	1.0
Cycle Time Check	168	478	1.6
Digital Read Out	948	4,288	5.6
Subroutine/ Libraries IOS/IOC	2,024	4,168	23.0
Global Data Library			
VDU Page Generator	12,960	21,136	7.7
TOTALS	52.4K	77.1K	655.0

* Bytes
** Milliseconds/Second

B-52 ARPTT
Computational Performance
Table 5

of the processor hardware.

C. The flight/motion task is partitioned equally among single board microcomputers (SBC) A, B and C, assuming no transport delays.

D. The two tasks "Host VDV" and "VDU page generation" between SBC D and E.

E. Task partitioning of the B52 ARPTT tasks is based upon execution times.

Fewer SBC could be used in the design (perhaps five) if one wished to increase the risk by decreasing the execution margin; however, unless the number of ATDs being purchased is large, the additional hardware expense is easily offset by the risk reduction in the software design.

To estimate the bus bandwidth used in this example, it is further assumed that all data for each function (shown on Table 5) is transferred in block form over the bus during each cycle of execution of that function. This provides the bus utilization for application data. In addition fifteen system calls and messages per application cycle were included to represent operating system bus overhead. This estimate is based on one call per function per cycle shown in Table 6. Under these assumptions only 40% of the single bus capacity was used.

Again, it must be emphasized that the software implementation approach affects these margins considerably.

SPECIFICATION CHANGES

The prime item development specification (PIDS) for the C5A/C141B ARRPTT was reviewed for changes based on the concepts of distributed microprocessor systems. Four categories of additions/changes are suggested. These deal with board level products, margin reassessment, operating system configuration, and the support concept; each category is discussed below.

1. Board Level Commercial Products. To avoid the contractor temptation to construct unnecessary special purpose microprocessor configurations, the distributed micro based central computational system should be constructed of commercially available (and second sourced) board level products. Where the cost is not prohibitive, as few different single board computer (SBC) types should be used (as practical), leading to much board level commonality even at the expense of some performance inefficiency. Commercially availalble and compatible system buses should also be used. This will lead to a lower parts cost and tend to curb contractor development at the piece part level.

2. Margin Reassessment. Timing, memory, and I/O margins were cited at 25% in the PLDS. As the cost of computer hardware drops, it makes sense to require additional margin (perhaps as much as 50%) to ease the development and support burden. The uniform application of margin over each SBC may no longer be reasonable since certain processors may exercise functions known not to expand while other processor functions may see all the growth. For a distributed system of this nature it may be wiser to indicate the margin in terms of functional growth if that is possible.

If, above and beyond margin, hardware add-on capability is specified, then the acceptance tests should include tests in the expanded hardware mode.

As bus layout and bandwidth are important system throughput parameters, a "bus bandwidth burner" analogous to the "time burner" is recommended for the system. This would permit the more precise estimation through measurement of the actual bandwidth margins.

3. Operating System. The selection of commercially available operating systems should be specified, but the operating system will require configuration for not only the nominal hardware topology but also for various growth

B52 ARPTT (8/32D) / SINGLE BOARD MICROCOMPUTER ASSIGNMENT					
REAL TIME FUNCTIONS	EXECUTION TIMES M SEC/SEC	SBC DESTINATION ASSIGNMENT (INTEL 286-10)	REAL TIME FUNCTIONS	EXECUTION TIMES PER FUNCTION M SEC/SEC	TOTAL EXECUTION TIME M SEC/SEC
Flight/Motion	247.5	A	Flight/Motion	495.0	500.0
			RMX-86 O.S.	5.0	
		B	Flight/Motion	495.0	500.0
			RMX-86 O.S.	5.0	
		C	Flight/Motion	495.0	500.0
			RMX-86 O.S.	5.0	
Host VDU	142.0	D	Host VDU	426.0	431.0
			RMX-86 O.S.	5.0	
VDU Page Generation	7.7	E	VDU Page Generation	426.0	431.0
			RMX-86 O.S.	5.0	
Visual	80.4	F	Visual	482.0	530.6
Engines	7.2		Engines	43.2	
			RMX-86 O.S.	5.0	
Fuel	49.3	G	Fuel	295.8	536.0
AFCS	22.5		AFCS	135.0	
Boom Control	17.6		Boom Control	105.0	
			RMX-86 O.S.	5.0	
Supervisor	6.2		RMX-86 O.S.	5.0	
Instruction Aids	7.5	H	Instruction Aids	37.2	310.2
Sound	6.7		Sound	45.6	
Test Preflight	1.0		Test Preflight	40.2	
Digital Readout	5.6		Digital Readout	33.6	
IOS/IOC	2.3		IOS/IOC	138.0	
Cycle Time Check	1.6		Cycle Time Check	9.6	

Microprocessor Assignment Example
Table 6

modes. Documentation on how to reconfigure the operating system for these growth modes should be procured.

4. Support Concept. The support concept implicit in the PIDS is one of procurung the latest computer and operating system versions and continually accommodating all vendor changes. This may prove to be inappropriate for distributed micro systems. Spare parts may be delivered with the system. Freezing the hardware configuration and the operating system for a period of time (i.e. seven to ten years) followed by a complete replacement of the computational system may be more cost effective. This will require further study.

ABOUT THE AUTHOR

Richard J. Sylvester has over twenty-eight years of engineering, technical and management experience encompassing system and software requirements definition and validation, computer resource support planning, software development methodology and management, configuration control, computer program design and testing, software verification, validation and certification, software related research and development, mathematical modeling and simulation. He obtained a B. E. in civil engineering and a M.E. in structural engineering from Yale University. His Ph.D. in civil engineering and applied mathematics was earned at the University of Colorado. He has worked in various capacities for Martin-Marietta, Aerospace Corporation, General Research Corporation and the Aeronautical Systems Division, USAF, Wright-Patterson Air Force Base, Ohio, among others. He is currently president of The Systems Productivity & Management Corporation of Dayton, Ohio.