

AD-A049 466

ARMY ELECTRONICS COMMAND FORT MONMOUTH N J
COMPUTER FAMILY ARCHITECTURE SELECTION COMMITTEE. (U)
SEP 77 W E BURR, A H COLEMAN, W R SMITH
ECOM-4525

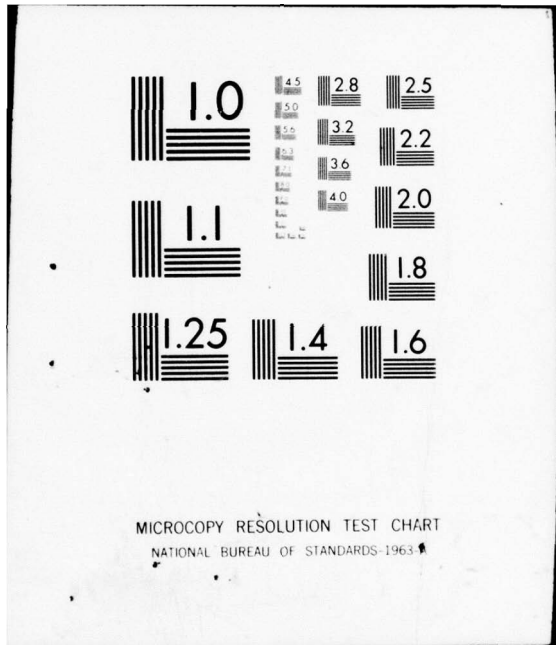
F/G 9/2

UNCLASSIFIED

NL

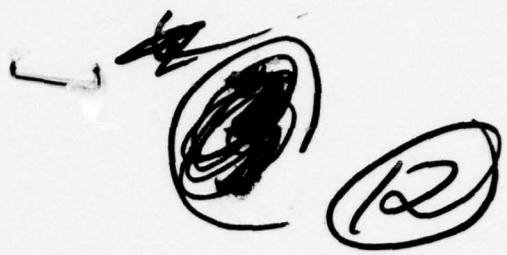
1 of 1
AD
A049466





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A 049466



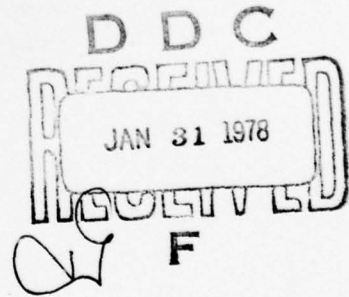
Research and Development Technical Report
ECOM - 4525

**SUMMARY OF THE FINAL REPORT OF THE ARMY/NAVY COMPUTER
FAMILY ARCHITECTURE SELECTION COMMITTEE**

AU NO. _____
JDC FILE COPY

William E. Burr
Aaron H. Coleman
Center for Tactical Computer Sciences
Communications/Automatic Data Processing Laboratory

William R. Smith
Naval Research Laboratory



September 1977

DISTRIBUTION STATEMENT
Approved for public release;
distribution unlimited.

ECOM

US ARMY ELECTRONICS COMMAND FORT MONMOUTH, NEW JERSEY 07703

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The citation of trade names and names of manufacturers in this report is not to be construed as official Government indorsement or approval of commercial products or services referenced herein.

Disposition

Destroy this report when it is no longer needed. Do not return it to the originator.

14 REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER ECOM-4525	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9	
4. TITLE (and Subtitle) Summary of the Final Report of the Army/Navy Computer Family Architecture Selection Committee.		5. TYPE OF REPORT & PERIOD COVERED Summary Rept.	
6. AUTHOR(s) William E. Burr, Aaron H. Coleman William R. Smith (NRL)		6. PERFORMING ORG. REPORT NUMBER	
7. PERFORMING ORGANIZATION NAME AND ADDRESS Center For Tactical Computer Sciences (CENTACS) DRSEL-NL-BC Fort Monmouth, N.J. 07703	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 16 LL7 62701 AH92B109	8. CONTRACT OR GRANT NUMBER(s)	17 B1
11. CONTROLLING OFFICE NAME AND ADDRESS 12 31 p.	13. REPORT DATE 11 Sep 1977	9. PERFORMING ORGANIZATION NAME AND ADDRESS Center For Tactical Computer Sciences (CENTACS) DRSEL-NL-BC Fort Monmouth, N.J. 07703	14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) CENTACS DRSEL-NL-BC Fort Monmouth, N.J. 07703
15. SECURITY CLASS. (of this report) Unclassified	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.	
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)			
18. SUPPLEMENTARY NOTES This report consists of a summary and nine (9) volumes. It is the result of joint Army/Navy work. The complete report maybe separately published by the Naval Research Laboratories.			
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Computer Family Architecture, Military Computer Family, Software Compatability, Support Software, Life Cycle Cost.			
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) An Army/Navy Computer Family Architecture (CFA) Selection Committee, comprising ten Army and seventeen Navy Organizations was organized by the Naval Research Laboratory and the Army Electronics Command in 1975 to select a proven, well-known computer architecture to be the basis of a Military Computer Family (MCF). The Selection Committee met five times in the period between October, 1975, and August, 1976, and evaluated nine computer architecture candidates in accordance with criteria established by the Committee. The Committee applied a preliminary screening process to select three candidates (IBM S/370, DEC PDP-11, and Inter-			

DDC
JAN 31 1978
F

037 620

AB

20. data 8/32) for more intensive evaluation. This final evaluation process considered experimentally determined architectural efficiency, support software availability, life cycle cost, and architecture licensing. As a result of this process, the Committee ranked the three architecture finalists in the following order:

1. FDP-11
2. S/370
3. 8/32



The technology

TABLE OF CONTENTS

	<u>PAGE</u>
1. INTRODUCTION	1
2. BACKGROUND	1
3. THE CFA/MCF PROJECT	2
4. THE CFA SELECTION COMMITTEE	3
5. ARCHITECTURE CANDIDATES	7
6. SELECTION PROCEDURE	7
a. Preliminary Screening Procedure	9
b. Final Candidates Evaluation	14
(1) Architecture Efficiency Evaluation	14
(2) Support Software Evaluation	17
(3) Life Cycle Cost Evaluations	18
(4) Licensing	22
c. Final Selection/Recommendations	22
7. CONCLUSIONS	24

APPENDIX - Listing of Volumes of "The Final Report of the CFA
Selection Committee

A-1

NTIS	<input checked="" type="checkbox"/>
DDC	<input type="checkbox"/>
UNANNOUNCED	<input type="checkbox"/>
JUSTIFICATION	
BY	
DISTRIBUTION/AVALIABILITY CODES	
DATE	
A	

1. INTRODUCTION

This report describes the work performed by an Army/ Navy Committee, representing 10 Army and 17 Navy organizations, to select a Computer Family Architecture (CFA) for use with a proposed software compatible family of military computers and associated systems/support software. This family is known as the Military Computer Family (MCF).

This report summarizes the contents of a full report on the work of the CFA Selection Committee (i.e. "Final Report of the CFA Selection Committee"). References to this full report will be made herein, in accordance with the table of contents shown in the Appendix.

2. BACKGROUND

The Department of Defense is spending over six billion dollars yearly for ADP systems. A large portion of this goes for acquisition of militarized computers and associated software that are used in tactical and strategic areas. Traditionally, these computers have been specified by the individual organizations (military project offices or commercial contractors) responsible for the development of each system. More often than not, computer selections are based upon local schedule, funding, or profit considerations, rather than the impact that the selection would have on long range hardware/software logistics costs. The result has been that the large number of types of computers used in Army and Navy systems are causing serious problems in the development and maintenance of software for those systems.

Military computers are usually procured as integral components of larger systems (e.g. radars, missile systems); the hardware issues have historically been given more attention than the logistics of the software, and in consequence, military computers normally have only the most primitive sort of support software. The development cycles for weapons systems are generally long enough (5 to 10 years) that the military computers in these systems are often obsolete before they are ever delivered to the Field Army or the Fleet. Past computer standardization efforts in the military have concentrated on hardware packaging or obscure architectures of such small market that there has been no incentive for the computer industry at large to invest in developing software and hardware compatible with these computers. The end result of these conditions is that the military pays over and over for development of computer systems that frequently fall far short of performance expectations.

This can be contrasted with the situation in the commercial OEM (original equipment manufacturer) marketplace. Here computers are produced for the much larger commercial market by the thousands or even the tens of thousands. A number of manufacturers such as DEC, Data General, and Interdata have software compatible product lines, covering a wide range of processors of varying

capabilities. Due to fierce competitive market pressures, system deficiencies are corrected, or the systems disappear. New products are developed much more quickly, and full advantage is taken of the advances in semiconductor device technology. Finally, due to the much larger user bases of commercial computers, and the competitive pressures of the marketplace, the support software bases of successful commercial computers are usually far superior to their military equivalents and are frequently improved or augmented by organizations seeking a share of this market.

A solution to many of the software problems with contemporary military computers would be to produce a family of software-compatible militarized computers. Moreover, if such a family were based upon a proven, commercial instruction-set architecture, then it would be possible to capture a good mature support software base, and to be certain that any architectural shortcomings were known and recognized. As the commercial system evolved, and the architecture was extended to meet the competition, the military computer family could also take advantage of these same extensions. Adhering to an established family in this way would avoid the architectural mavericks that limited-production military computers are prone to be.

3. THE CFA/MCF PROJECT

Since early 1975, the Center for Tactical Computer Sciences (CENTACS) of the U. S. Army Electronics Command and the Naval Air Systems Command (NAVAIR) have been supporting a cooperative Army/Navy effort to develop such a family of military computers, based upon a common instruction-set architecture.

The fundamental premise of the MCF project is that software compatibility should be achieved by the adoption of an existing, proven computer architecture for the MCF, thereby minimizing the risks inherent in the design of a new computer architecture and permitting the "capture" of an existing and evolving software base. In this context, computer architecture is distinguished from implementation considerations, and is defined as the structure of the computer which a machine level programmer needs to know in order to write all programs which will run correctly on the computer. For example, the architecture of the IBM S/370 is defined in the IBM System/370 Principles of Operations Manual. There are many implementations of the architecture (370-158, 370-168, etc.), but only one architecture, and every implementation will execute the same software. Another premise upon which the Army/Navy cooperative effort is based is the goal of software transportability from prior generation military computers to the MCF, most probably via emulation. In other words, the Army and Navy cannot abandon its investment in existing software. There is a strong analogy here with IBM's continued support of such machines as the 1401 and the 7090 via emulation, when the 360 family was introduced.

The first task of the MCF project was the selection of the CFA. CENTACS and the Naval Research Laboratory cooperated to lead that effort, and the following sections of this report describe how that selection was made.

The second task of the project is to develop a System Implementation Plan, which in a commercial organization would probably be called a product plan, to define the form, fit, and function characteristics of the MCF and the individual family members. The instruction-set architecture of the processors, not the detailed logic design, will be specified, so that various military equipment manufacturers (in general, not the manufacturer of the commercial version of the CFA) will be able to independently develop MCF members to meet the form, fit, and function requirements of the MCF, and to run the CFA instruction set. This approach will permit multiple sources for the various family members, and will allow manufacturers to take maximum advantage of rapidly developing semiconductor technology. The goal is a line of plug-compatible modules that can be interconnected as computer systems in a variety of configurations, to meet a wide range of performance/application requirements.

A similar Support Software Implementation Plan contract is planned for FY 1978. This plan will attempt to take maximum advantage of the existing support software base for the selected CFA.

4. THE CFA SELECTION COMMITTEE

The mechanism for selecting the CFA was a joint Army/Navy Selection Committee. In order to achieve a wide representation of military computer requirements in this effort, letters were sent to Army and Navy Laboratories, System Centers, and Project Managers, inviting them to nominate "candidate" architectures, and to participate in the CFA selection process as members of the CFA Selection Committee. Ten Army and seventeen Navy organizations assigned representatives to the Selection Committee, which was active between October 1975 and August 1976. The members of the Committee are listed in Table 1. Subcommittees were established to handle specific tasks. The Committee officers were:

Chairman - Aaron H. Coleman, ECOM
Vice Chairman - William R. Smith, NRL
Secretary - William E. Burr, ECOM

The Subcommittees and their chairmen are listed in Table 2.

Of the several procedural rules adopted by the Committee, the most important was the requirement for a 2/3 vote of the members present to carry a committee motion.

TABLE 1
 Army/Navy CFA Committee Membership
 Army Members

<u>Organization</u>	<u>Primary</u>	<u>Alternate</u>
U. S. Army Electronics Command	D. Hadden/E Lieblein	J. Mercurio
Project Manager, Army Tactical Data Systems	A. R. DeNezzo	1LT R. Atkinson
Project Manager, Navigation/Control Systems	2LT N. Herndon	
Satellite Communication Agency	R. Perle	J. Perrone
Project Manager, PATRIOT Missile System	CPT R. Sabin	R. Flights
U. S. Army Computer Systems Command	MAJ B. Blood	
U. S. Army Aviation Systems Command	W. Klees	
U. S. Army Missile Command	Douglas A. Wise	
ECOM Avionics Laboratory	Henry R. Chambers	
PMO, Multi Service Communications Systems	A. Buray	

TABLE 1 (cont.)
 Army/Navy CFA Committee Membership
 Navy Members

<u>Organization</u>	<u>Primary</u>	<u>Alternate</u>
Naval Underwater Systems Center Newport	T. Conrad	D. Watson
Fleet Combat Direction Systems Support Activity	R. G. Estell	
Naval Post Graduate School	LT B. E. Allen G. L. Barksdale	
Naval Avionics Facility, Indianapolis	Dr. Jack Chaney	C. Eckert
Naval Air Development Center	C. Mattes	C. Joeckel
Fleet Combat Direction Systems Support Activity, Dam Neck	J. D. Warner	C. D. Upshur
Naval Surface Weapons Center, Dahlgren	W. L. McCoy	E. W. Nichols
Naval Air Test Center	J. P. Sharatz	G. S. Ryan
Pacific Missile Test Center, Pt. Mugu	M. Stevens Paul L. Miller	R. Lindsey
Naval Undersea Center	J. K. Fogerty	T. L. Cloer
Naval Electronics Laboratory Center	N. L. Tinkelpaugh	
Naval Ship Research & Development Center	L. M. Culpepper	C. M. Chernick
Naval Underwater Systems Center, New London	A. Clearwaters	H. Watt
Naval Surface Weapons Center, White Oak	Dr. L. Haynes	
Naval Training Equipment Center	C. F. Summer	L. Healy
Naval Sea Systems Command	W. H. Hill	
Naval Research Laboratory	S. Fuller	

TABLE 2
 Army/Navy Selection Committee Subcommittee
 Chairmen

<u>Subcommittee</u>	<u>Chairman</u>	<u>Organization</u>
Interdata	William Burr	ECOM
IBM S/370	Al Clearwaters	NUSC, New London
AN/GYK-12	Norman Taupeka	PM, ARTADS
AN/UYK-7	Henry Hill	NAVSEA
AN/UYK-20	John Sharatz	NATC
Burroughs B-6700	Dave Hadden	ECOM
PDP-11	Dan Siewiorek	NRL/CMU
ROLM	Len Haynes	NSWC, White Oak
SEL 32	W. L. McCoy	NSWC, Dahlgren
Test Program	William Burr	ECOM
Selection Criteria	Sam Fuller	NRL/CMU
Architecture Test	Sam Fuller	NRL/CMU
Support Software Evaluation	Ed Lieblein	ECOM
Final Selection Methodology	William R. Smith	NRL
Licensing	Sam Levine	ECOM
Auditor for Initial Screening and Support Software Evaluation	Harold Stone	Univ. Mass.

5. CANDIDATE ARCHITECTURES

The basic mechanism for deciding which architectures should be considered by the committee was to ask Army and Navy organizations to nominate candidate architectures. These nominations were augmented by the Committee in its early meetings. The architectures which were considered by the Committee are:

Burroughs B6700
DEC PDP-11/70
IBM S/370
Interdata 8/32
Litton AN/GYK-12
ROLM 1664
Systems Engineering Laboratories SEL 32
Univac AN/UYK-7
Univac AN/UYK-20

On the list of candidates the S/370 and the B6700 are large scale commercial data processing type architectures. The PDP/70, SEL-32, Interdata 8/32 and the ROLM are classical OEM type minicomputers, and the AN/GYK-12, AN/UYK-7, and the AN/UYK-20 are three of the most widely used military computers.

Although the above list of architectures is not all inclusive, most of the Army and Navy organizations who nominated candidates went through their own internal screening process, considering a much wider selection of architectures prior to making their nominations. As a result, the nine architectures considered by the Committee represent the best candidates for a family of computers for military applications, according to the consensus of over two dozen Army and Navy organizations. Each architecture needed a defender.

6. SELECTION PROCEDURE

It was apparent to the Committee after much discussion, that there were certain key, critical characteristics that should be well satisfied by the selected CFA. Further, it became apparent that it made sense to perform an initial screening and ranking of the candidates, based on these characteristics, so that the obviously least acceptable candidates could be discarded and those with the most potential could be retained and investigated much more thoroughly. An initial screening process was therefore devised to select several "best final candidates" for more detailed evaluation.

After the initial screening process was completed, the three final candidates were subjected to a test program experiment to measure the efficiency of the architectures. The support software bases of the three architectures were studied, and life-cycle cost models were constructed to determine if one of the three architectures had a decisive economic advantage. Finally, the manufacturers were contacted to determine the conditions under which they would be willing to license their architectures for production by military vendors. This process is illustrated in Figure 1, and is described in more detail below.

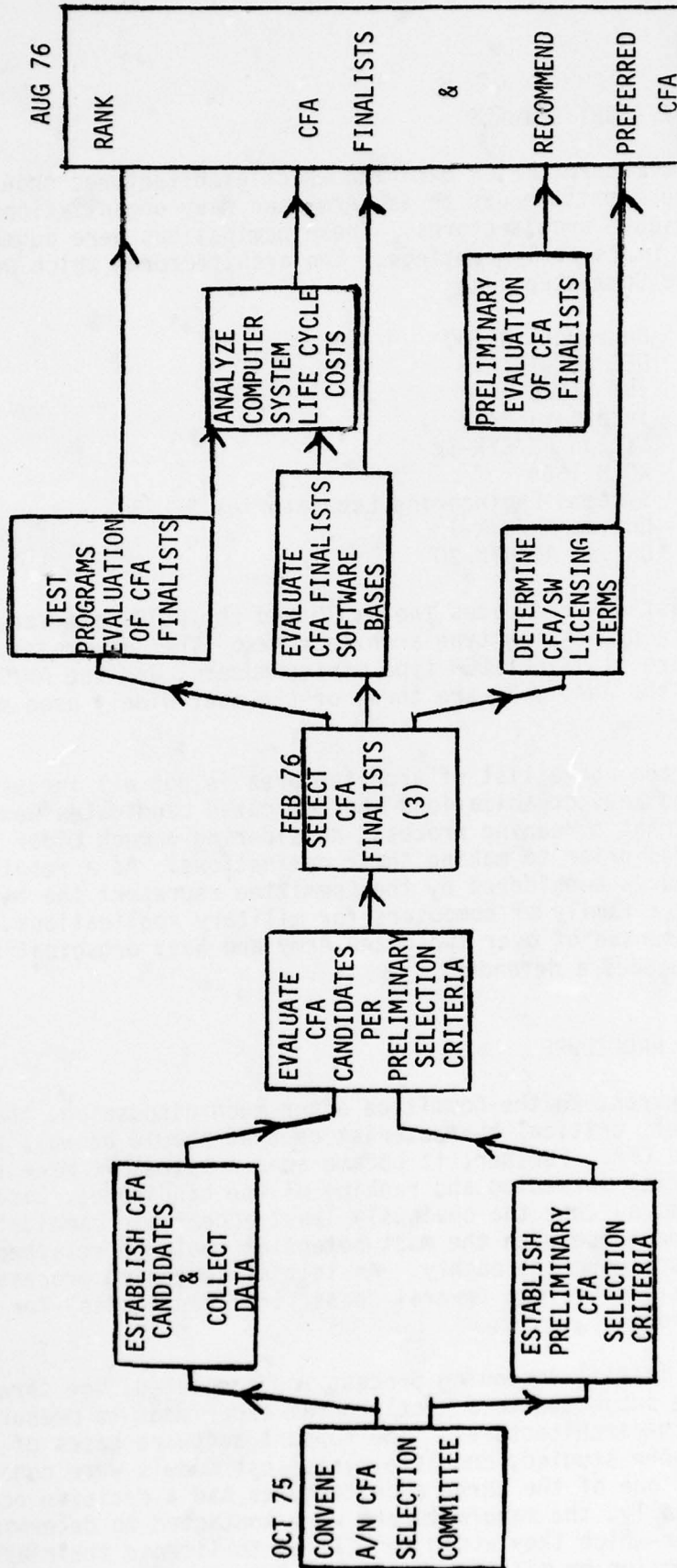


Figure 1 - CFA Selection Process

a. Initial Screening

The Selection Committee decided to select the final candidate architectures from the initial list by means of two kinds of criteria. The first kind of criteria, which served as pass/fail tests of architectural adequacy, were called "absolute criteria". The committee planned to eliminate all architectures which did not completely satisfy these criteria. Absolute criteria included such requirements as a satisfactory protection mechanism, and a virtual to physical address translation mechanism. The second kind of criteria were called "quantitative criteria". The quantitative criteria were intended to provide a relative ranking of the architectures in terms of characteristics which the committee believed were important measures of a computer architecture. Quantitative criteria included such characteristics as the size of the physical address space, the size of the virtual address space, the number of bits which had to be moved to save that state of the machine under various circumstances, and the size of the installed user base. A listing and very brief description of the absolute and quantitative criteria are shown in Table 3. The reader should see Volume II of the CFA Committee Final Report for a detailed discussion of these criteria. Each quantitative criterion was assigned a weighting factor by each committee member organization. An average weighting factor was computed for the entire committee for each criterion. The quantitative criteria scores for each candidate were normalized, weighted, and summed to give a composite figure of merit for each architecture.

Subcommittees were created to evaluate each architecture, in terms of the absolute and quantitative criteria. A meeting of the full committee was then devoted principally to verifying the consistency and correctness of the evaluations of the candidate architectures. In addition, the results of this evaluation were audited by a consultant to ensure the consistency and correctness of the evaluation.

A principal difficulty in making the evaluations was the imprecision of most of the reference manuals of the candidate architectures, requiring frequent communication with the manufacturers in some cases. Certain of the manuals, as typified by the IBM S/370 Principals of Operation Manual, appeared to be complete and precise definitions of an architecture. Others left essential architectural details ambiguously defined or not defined at all.

The results of the absolute and quantitative criteria evaluations are summarized in Table 4. The PDP-11 and the IBM S/370 were the only two architectures which clearly passed all the absolute criteria, and they also were among the top three in the quantitative criteria evaluation. The Interdata 8/32 was also selected as a finalist on the basis of its very strong showing on the quantitative criteria, despite a nagging technical uncertainty concerning the state of the machine after interrupts, which the committee was never able to resolve to its own satisfaction.

TABLE 3

ABSOLUTE CRITERIA FOR CFA EVALUATION

- (1) VIRTUAL MEMORY SUPPORT. The architecture must support a virtual to physical translation mechanism.
- (2) PROTECTION. The architecture must have the capability to add new, experimental (i.e., not fully debugged) programs that may include I/O without endangering reliable operation of existing programs.
- (3) FLOATING POINT SUPPORT. The architecture must explicitly support one or more floating point data types with at least one of the formats yielding more than 10 decimal digits of significance in the mantissa.
- (4) INTERRUPTS AND TRAPS. It must be possible to write a trap handler that is capable of executing a procedure to respond to any trap condition and then resume operation of the program. The architecture must be defined such that it is capable of resuming execution following any interrupt.
- (5) SUBSETABILITY. At least the following components of an architecture must be able to be factored out of the full architecture:
 - Virtual-to-Physical Address Translation Mechanism
 - Floating Point Instructions and Registers (if separate from general purpose registers)
 - Decimal Instructions Set (if present in full architecture)
 - Protection Mechanism
- (6) MULTIPROCESSOR SUPPORT. The architecture must allow for multiprocessor configurations. Specifically, it must support some form of "test-and-set" instruction to allow the implementation of synchronization functions such as P and V.
- (7) CONTROLLABILITY OF I/O. A processor must be able to exercise control over any I/O Processor and/or I/O Controller.
- (8) EXTENDIBILITY. The architecture must have some method for adding instructions to the architecture consistent with existing formats. There must be at least one undefined code point in the existing opcode space of the instruction formats.
- (9) READ ONLY CODE. The architecture must allow programs to be kept in a read-only section of primary memory.

TABLE 3 (cont.)

QUANTITATIVE CRITERIA FOR CFA EVALUATION

(1) VIRTUAL ADDRESS SPACE

- (a) V_1 : The size of the virtual address space in bits.
- (b) V_2 : Number of addressable units in the virtual address space.

(2) PHYSICAL ADDRESS SPACE

- (a) P_1 : The size of the physical address space in bits.
- (b) P_2 : The number of addressable units in the physical address space.

(3) FRACTION OF INSTRUCTION SPACE UNASSIGNED

(4) SIZE OF CENTRAL PROCESSOR STATE

- (a) C_{s1} : The number of bits in the processor state of the full architecture.
- (b) C_{s2} : The number of bits in the processor state of the minimum subset of the architecture (i.e., without Floating Point, Decimal, Protection, or Address Translation Registers).
- (c) C_{m1} : The number of bits that must be transferred between the processor and primary memory to first save the processor state of the full architecture upon interruption and then restore the processor state prior to resumption.
- (d) C_{m2} : The measure analogous to C_{m1} for the minimum subset of the architecture.

(5) VIRTUALIZABILITY

K : is unity if the architecture is virtualizable as defined in [Popek and Goldberg, 1974] otherwise K is zero.

(6) USAGE BASE

- (a) B_1 : Number of computers delivered as of the latest date for which data exists prior to 1 June 1976.
- (b) B_2 : Total dollar value of the installed computer base as of the latest date for which data exists prior to 1 June 1976.

(7) I/O INITIATION

I : The minimum number of bits which must be transferred between main memory and any processor (central, or I/O) in order to output one 8-bit to a standard peripheral device.

(8) DIRECT INSTRUCTION ADDRESSABILITY

D : The maximum number of bits of primary memory which one instruction can directly address given a single base register which may be used but not modified.

TABLE 3 (cont.)

(9) MAXIMUM INTERRUPT LATENCY

Let L be the maximum number of bits which may need to be transferred between memory and any processor (CP, IOC, etc.) between the time an interrupt is requested and the time that the computer starts processing that interrupt (given that interrupts are enabled).

The reader is cautioned that the application of these criteria requires a great deal of interpretation. The Selection Committee went to some considerable effort to arrive at comparable interpretations for each architecture. It may not be at all obvious from the simple definitions presented here, how the actual values used by the committee were calculated. This is documented in detail in Volume II of the CFA Committee Final Report, and the interested reader should refer to Volume II.

TABLE 4

CANDIDATE SCORES ON ABSOLUTE AND QUANTITATIVE CRITERIA

<u>Architecture</u>	<u>Quantitative Criteria Score</u>	<u>Absolute Criteria Score</u>
8/32	1.68 (Best)	Problem with interrupts and traps
PDP-11 /70	1.43	Passed all
S/370	1.36	Passed all
AN/GYK-12	.94	Failed floating-point
ROLM 1664	.92	Failed virtual memory mapping
B6700	.91	Failed protection
SEL-32	.86	Failed virtual memory mapping
AN/UYK-7	.46	Failed floating point
AN/UYK-20	.44 (worst)	Failed protection

b · Final Candidates Evaluation

(1) Architecture Efficiency Evaluation

A Test Program Subcommittee was appointed at the first Selection Committee meeting. This subcommittee proposed a set of 23 potential test programs, which were believed to be representative of the operations performed in military data processing applications. The Committee ranked these programs by their relative importance, and the top twelve programs were selected as the basis of the Test Program Experiment. These twelve programs are listed and briefly described in Table 5.

Each of the twelve test Programs was a relatively small kernel-type program, most were subroutines, and most were defined as "structured" programs in a Program Definition Language (PDL). Programmers were then asked to "hand compile" the programs into the assembly languages of the respective machines from their PDL descriptions. This procedure was followed to minimize the effects of programmer variations. No large scale programs from "real" military systems were coded, because of the excessive expense involved in coding and testing a statistically significant set of such programs. High level language programs were not tested, because there is no practical way to separate the effects of compiler efficiency from the effects of architecture efficiency which the experiment was intended to measure.

Slightly over one hundred test program samples were coded by sixteen programmers at participating organizations. The experiment was designed using analysis of variance techniques to give the best possible estimates of the relative efficiency of the three architectures.

Three measures were defined to gauge the efficiency of the architectures, independently of hardware implementation features such as cycle time. These measures were:

- S - The static storage requirement for the program in bits.
- M - The number of bits of program and data which were transferred between the processor and main memory during execution of a program. The M Measure is intended to be an index of the memory bandwidth requirements of an architecture.
- R - The number of bits of program and data which were transferred among the internal processor registers during execution of a program. The R Measure is intended to be an index of the processor bandwidth requirements of an architecture.

TABLE 5
TEST PROGRAMS

1. I/O kernel, four priority levels, requires the processor to field interrupts from four devices, each of which has its own priority level. While one device is being processed, interrupts from higher priority devices are allowed.
2. I/O kernel, FIFO processing, also fields interrupts from four devices, but without consideration of priority level. Instead, each interrupt causes a request for processing to be queued; requests are processed in FIFO order. While a request is being processed, interrupts from other devices are allowed.
3. I/O device handler, processes application programs' requests for I/O block transfers on a typical tape drive, and returns the status of the transfer upon completion.
4. Large FFT, computes the fast Fourier transform of a large vector of 32bit floating point numbers. This benchmark exercises the machine's floating point instructions, but principally tests its ability to manage a large address space.
5. Character search, searches a potentially large character string for the first occurrence of a potentially large argument string. It exercises the ability to move through character strings sequentially.
6. Bit test, set, or reset tests the initial value of a bit within a bit string, then optionally sets or resets the bit. It tests one kind of bit manipulation.
7. Runge-Kutta integration numerically integrates a simple differential equation using third-order Runge-Kutta integration. It tests floating-point arithmetic.
8. Linked list insertion inserts a new entry in a doubly-linked list. It tests pointer manipulation.
9. Quicksort sorts a potentially large vector of fixed-length strings using the Quicksort algorithm. Like FFT, it tests the ability to manipulate a large address space, but it also test the ability of the machine to support recursive routines.
10. ASCII to floating point converts an ASCII string to a floating point number. It exercises character-to-numeric conversion.
11. Boolean matrix transpose transposes a square, tightly-packed bit matrix. It tests the ability to sequence through bit vectors by arbitrary increments.
12. Virtual memory space exchange changes the virtual memory mapping context of the processor.

The S, M and R measures are indicators of the relative amounts of hardware capability that are necessary when implementing an architecture to do a certain job. That is, larger S measure means that correspondingly more memory will be required to handle a given set of applications programs. Clearly, the architecture that can execute the programs with the smallest S is desirable. Similarly, M and R are indicators of the relative hardware speed/bandwidth requirements for memory and processor implementations.

The S, M and R raw data were gathered with the help of a special ISP language compiler and simulator system. The three architectures were described in ISP (Instruction Set Processor), a formal language for describing computers at the instruction/register level. These ISP descriptions were then compiled and run on the ISP simulator which was designed to automatically gather statistics of register and memory activity during execution of the test programs on the simulated candidate architectures. See Volume IV of the Committee Report for a detailed treatment of the ISP System and its use in the CFA effort.

The final results reflect the performance of each candidate architecture for each measure. Those results are shown in Table 6. This experiment is described more fully in Volume III of the final Committee Report.

TABLE 6
TEST PROGRAM EXPERIMENT RESULTS

<u>Architecture</u>	<u>S</u>	<u>M</u>	<u>R</u>
Interdata 8/32	.83	.85	.83
PDP-11	1.00	.93	.94
IBM S/370	1.21	1.27	1.29

The results are normalized so that unity indicated average performance; the lower the score on any of the measures, the better the architecture handled the set of test programs. In other words, the results indicate that the S/370 needs 21 percent more memory than the average to store the test programs, the 8/32 needs only 83 percent as much memory as average, and the PDP-11 is nearly average in its use of memory. The differences between the S/370 results and the average of the results of the other two architectures were statistically significant at the 95 percent confidence level, but the differences between the 8/32 and the PDP-11 results were not statistically significant at this confidence level. The differences between the 8/32 and the S/370 results were also statistically significant for the S and M measures at the 95 percent confidence level.

(2) Support Software Evaluation

A support Software Evaluation Subcommittee was appointed to study the support software bases of the three final candidate architectures. This subcommittee began by defining an extensive menu of support software tools, which might be useful in military systems development. Committee member organizations were then asked to rate each tool by its utility in developing software for military weapon systems. The 28 most important support software tools were selected from this rating. The CFA candidate manufacturers and other commercial sources were investigated as to the availability of these 28 software tools for each architecture. Table 7 lists the basic tool types on the required support software menu.

TABLE 7

MENU OF REQUIRED SOFTWARE TOOL TYPES

- Compilers
- Macro Assemblers
- Interactive Source Language Editors
- Interactive Symbolic Debuggers
- Extended Overlay Linker
- Test Case Design Advisors
- Integrated Library
- Text Processing System
- Data Base Management System
- GP System Simulator
- Time Sharing Operating System (TSOS) + VMM
- Language Independent Monitors
- Test Data Generator
- Non-Interactive Symbolic Debugger
- Computer System Simulator
- Batch Source Language Editors
- Language Dependent Monitors
- TSOS + MPOS + VMM
- Basic Assembler
- RTOS + TSOS
- Test Instrumenters & Analyzers
- Automatic SW Production & Test
- Basic Linker
- Standards Enforcers
- Reformatters
- Test Data Auditor
- Simple Overlay Linker
- Data Base Design Aid

The cost to develop each item of support software was estimated. The total cost to develop the selected support software items was estimated to be approximately 41 million dollars. The estimated value of the support software bases for each of the final candidate architectures is summarized in Table 8 below; also shown is the estimated cost to eliminate deficiencies as compared to the desired support software base:

TABLE 8. TACTICAL SUPPORT SOFTWARE BASE EVALUATION

<u>Architecture</u>	<u>Estimated Value of Current SSW Base</u>	<u>Estimated Cost To Eliminate Deficiency</u>
8/32	\$15.3 M	\$25.9 M
PDP-11	\$22.2 M	\$19.1 M
S/370	\$32.3 M	\$ 9.6 M

See Volume V of the Committee Report for a detailed treatment of the support software evaluation.

(3) Life Cycle Cost Evaluations

A Final Selection Methodology Subcommittee was formed at the third Selection Committee meeting to investigate and pursue a methodology for combining the results of the committee's evaluations into a single evaluation criterion which would be realistic and meaningful to DOD management. This subcommittee proposed a method of converting the architecture and software evaluation results to life cycle costs so that a final selection could be aided by data based on the comparative economics of using each of the candidate architectures in military computer systems.

Two separate computer life cycle requirements models were used for the cost analyses. Both used the data gathered in the Architecture Efficiency Evaluation and the Support Software Base Evaluation described previously to convert the modeled requirements into dollar costs.

The first model is a "top-down" model which represents total life cycle requirements for DOD computers in the 1978-1990 time period, using each of the three final candidate architectures for the MCF. It was based upon extrapolating trends in DOD wide expenditures and requirements for military computer hardware and software.

Figure 2. Summarizes results of computing CFA life cycle costs summed over the years 1978 to 1990 for the three candidates, for certain conditions. To simplify comparisons, the total assumed costs (approximately \$1 billion) are normalized with respect to the IBM S/370.

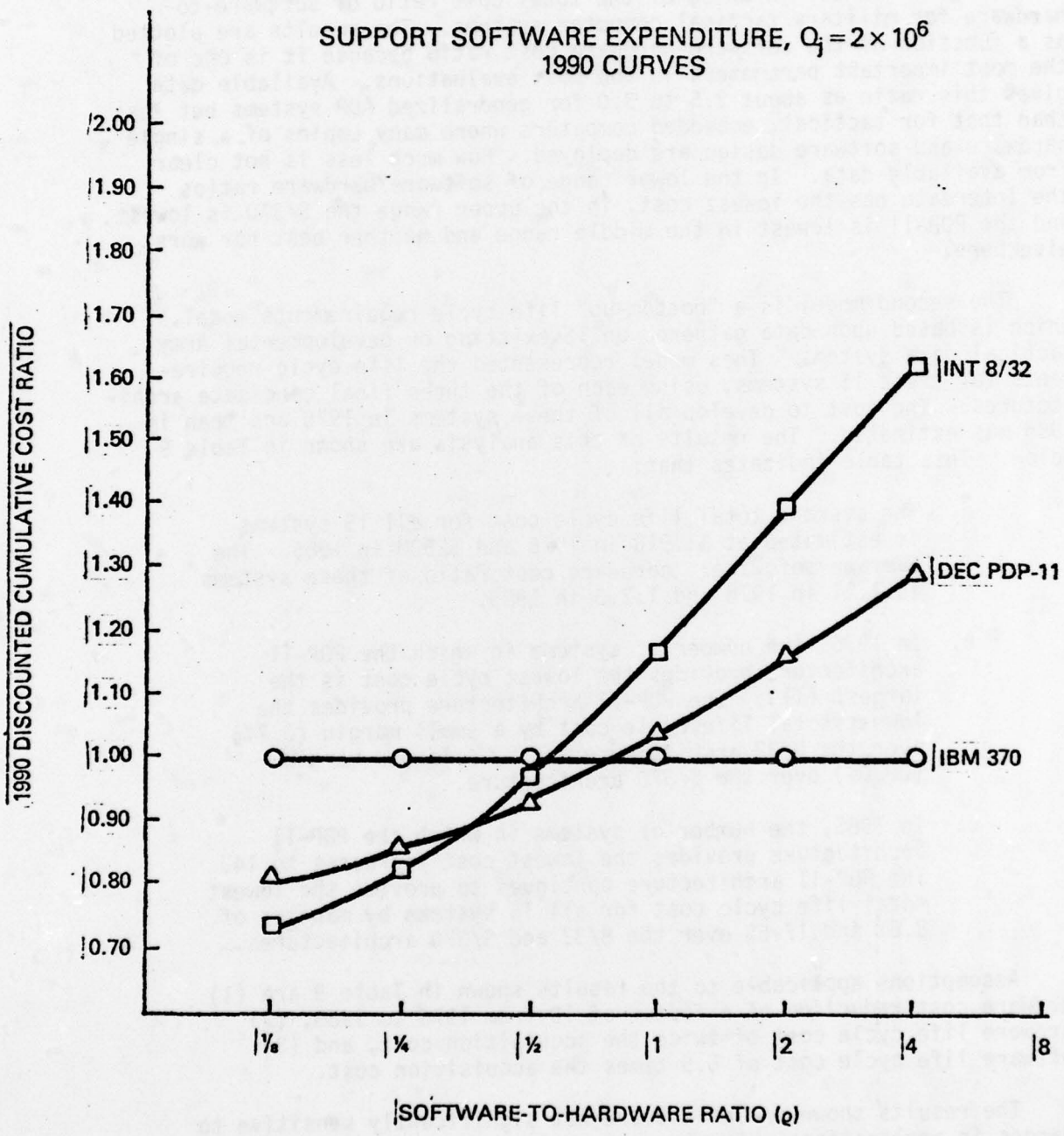


FIGURE 2. Top Down Life Cycle Cost Curves

The results are shown for specific values of two of the model input parameters. The first is an expenditure rate (\$2M/year) for completing development of the support software base of each candidate. The second is a range of values (x-axis) of the total cost ratio of software-to-hardware for military tactical computer systems. The results are plotted as a function of the software-hardware cost ratio because it is one of the most important parameters in the cost evaluations. Available data gives this ratio as about 2.5 to 3.0 for generalized ADP systems but less than that for tactical, embedded computers where many copies of a single hardware and software design are deployed. How much less is not clear from available data. In the lower range of software/hardware ratios the Interdata has the lowest cost, in the upper range the S/370 is lowest, and the PDP-11 is lowest in the middle range and neither best nor worst elsewhere.

The second model is a "bottom-up" life cycle requirements model, which is based upon data gathered on 15 existing or developmental Army tactical data systems. This model represented the life cycle requirements for these 15 systems, using each of the three final candidate architectures. The cost to develop all of these systems in 1976 and then in 1985 was estimated. The results of this analysis are shown in Table 9 below. This table indicates that:

- a. The average total life cycle cost for all 15 systems is estimated at \$1.91B in 1976 and \$250M in 1985. The average software: hardware cost ratio of these systems is 1:11 in 1976 and 1:2.3 in 1985.
- b. In 1976, the number of systems in which the PDP-11 architecture provides the lowest cycle cost is the largest (11). The PDP-11 architecture provides the lowest total life cycle cost by a small margin (3.7%) over the 8/32 architecture and by a larger margin (20.0%) over the S/370 architecture.
- c. In 1985, the number of systems in which the PDP-11 architecture provides the lowest cost increases to 14. The PDP-11 architecture continues to provide the lowest total life cycle cost for all 15 systems by margins of 8.8% and 17.6% over the 8/32 and S/370 architectures.

Assumptions applicable to the results shown in Table 9 are (1) hardware cost reduction of a factor of 10 from 1976 to 1985, (2) hardware life cycle cost of twice the acquisition cost, and (3) software life cycle cost of 5.5 times the acquisition cost.

The results shown in Table 9 are not significantly sensitive to changes in applications software cost or in the annual support software investment for the selected CFA.

A limited sensitivity analysis was performed with both models. If lower estimates are made for software development costs (relative to hardware costs), and/or if faster development of the support software base is projected (so that all three architectures rapidly acquire a complete

A. AVERAGE TOTAL LIFE CYCLE COSTS (\$000,000)

Type Cost	1976	1985
Hardware	\$1750	\$175
Software	<u>162</u>	<u>75</u>
TOTAL	\$1912	\$250

B. 1976 ARCHITECTURE COMPARISON

Architecture	# System Preferences	Relative Total Cost*		
		HDW	SW	Total
8/32	1	.92	1.33	.96
PDP-11	11	.91	1.00	.96
S/370	3	1.16	.67	1.12

C. 1985 ARCHITECTURE COMPARISON

Architecture	# System Preferences	Relative Total Cost*		
		HDW	SW	Total
8/32	-	.92	1.20	1.00
PDP-11	14.5	.91	.91	.91
S/370	0.5	1.16	1.09	1.09

* with respect to average cost; 1.00 equals average cost

TABLE 9. Summary: Bottom Up Life Cycle Cost Analysis

support software base), then the Interdata 8/32 eventually becomes the least expensive architecture, because of its efficient architecture as indicated by the test program results. If very high software development cost estimates are made, and/or very slow support software development is projected, then the S/370 becomes the least expensive architecture because of its advantage in support software. Figure 2 illustrates this behavior. In the intermediate ranges of software cost estimates, where top-down and bottom-up results were in the best agreement, the PDP-11 appears to have a slight cost advantage. However, compared to the expected errors in the results due to the uncertainties in the input data and assumptions, the life-cycle cost differences between the two models and among the three candidate architectures are small. The software/hardware ratio which is one of the most important factors in both models is one of the hardest to pin down with supporting data, and the results of both models can be made to change by using values from different sources for the same input parameters. The strongest conclusion to be derived is that the results agree and that, in terms of life cycle cost, all three candidates would provide comparable choices for the CFA. See Volume VI of the Final Report for details of the life-cycle cost evaluations.

(4) Licensing

Meetings were held with IBM, DEC, and Interdata to discuss the terms and conditions under which they would grant a non-exclusive license to the Government to use their architecture for militarized processors. All three manufacturers were cooperative and proposed terms for such an agreement. Although the proposed licensing agreements were a significant factor in the final selection process, the details cannot be given here, due to the confidential nature of the discussions. Volume VII of the Final Report, which is restricted to internal Government use, contains the details of the licensing proposals.

c. Final Selection/Recommendations

The selection Committee held its fifth and final meeting on 24 to 26 August 1976 at the Naval Underwater System Center, Newport, R. I., for the purpose of selecting the recommended architecture for the MCF. At this meeting, the results of the evaluations discussed in the preceding sections of this article were considered by the committee and discussed at length. Based upon that data, and upon other concerns specifically considered by the committee during its discussion of the final selection, the respective strengths and weaknesses of each architecture can be summarized as follows:

- A. INTERDATA 8/32, The 8/32 was the highest rated architecture on the Quantitative Criteria, and the Test Program Results. The 8/32 has a good interrupt structure for real-time processing. On the other hand, the software base is relatively weak, which consequently compromised its performance in the life cycle cost evaluations. There was a nagging question about how well the state of the machine was preserved after interrupts.

B. IBM S/370. The strongest virtue of the S/370 is its large support software base. The S/370 performed well on the life-cycle cost analyses under assumptions of maximum relative cost of software development. The S/370 is the only architecture demonstrated as an easily virtualized computer in a standard product line. On the other hand, its interrupt structure was considered cumbersome for real time control applications. The test program results indicate that the architecture is significantly less efficient than the 8/32 and the PDP-11. There was also concern that small subset versions might not prove cost-effective for low-end applications, and that there was insufficient experience with the S/370 in OEM type applications.

C. PDP-11. The PDP-11 enjoys a good support software base, performed relatively well on the Test Programs, and has a good interrupt structure for real-time control applications. It enjoys a slight advantage on the cost models for a range of reasonable assumptions. Small scale (micro-processor) implementations are practical and have been built. On the negative side, the 16 bit virtual address space is a limitation and it may be expensive to add a virtual machine capability to the architecture.

The committee made four final recommendations:

A. The DEC PDP-11 was determined by a vote of 14 to 4 to be the most advantageous architecture for the MCF, the IBM S/370 was ranked second, and the Interdata 8/32 was ranked third.

B. The committee unanimously agreed that a single instruction-set architecture should be selected for the MCF, that the selection of only one architecture is more important than which one of the candidates is selected, and that any one of the three final candidate architectures could provide a satisfactory basis for the MCF.

C. The committee agreed that an effort should be made to relieve the limitations of the selected architecture. In the case of the PDP-11 the major limitation is the small (16 bit) virtual address space.

D. A single organizational structure must be established to control the architecture, or major incompatibilities between different implementations will surely result.

See Volume VIII of the Final Report for details of the CFA final selection/recommendation process.

7. CONCLUSIONS

It is sometimes asserted that military systems have unique requirements which preclude the use of a general purpose commercial instruction set. Developers of computer based weapons systems often assert that they alone have such severe "real-time" constraints that they compel the use of a particular processor. It is worth noting that the Selection Committee compared three of the most widely used military architectures with six of the most widely used commercial architectures and found that the military architectures were deficient compared to the commercial architectures in terms of those architectural characteristics believed to be most important in tactical military applications. It is worth noting also that none of the military architectures had any unique features which proved advantageous, while all three were found to have architectural shortcomings. Moreover, the support software available for the three military architectures is relatively weak. Considering how easily modern microprogrammable processor hardware may be adapted to a given instruction-set architecture, there appears to be little reason to continue to use little-known or immature developments in future military computer systems.

The PDP-11 is one of the most successful architectures, in terms of user acceptance, in the history of the computer industry. It has been manufactured in the tens of thousands, and is widely used in almost every sort of OEM application. An extensive support software base exists for it, and DEC will continue to develop and support the architecture for the foreseeable future. It is clearly a satisfactory choice for the Military Computer Family. With the MCF intelligently defined and implemented, it will make available a family of militarized processors with excellent software development tools, and the capability to develop and maintain software on less expensive commercial equipment. This in turn will result in substantial cost and quality benefits in the application of computers to military systems.

APPENDIX
Listing of Volumes of
"The Final Report of the CFA Selection Committee"

A-1 Volume I - Introduction

Volume I explains the background, rationale and organization of the Computer Family Architecture effort and the Selection Committee.

A-2 Volume II - Selection of Candidate Architecture and Initial Screening

Volume II describes the initial candidate selection, and discusses architectural issues pertinent to CFA evaluation. The evaluation criteria applied to the architectural candidates for preliminary screening are described in detail, and the results of that evaluation are discussed.

A-3 Volume III - Evaluation of Computer Architectures via Test Programs

Volume III discusses the development of the measures used to gauge architectural efficiency and describes the test programs selected for the evaluation. The method of specifying the test programs and the structure of the programming experiment to minimize programmer effects are also discussed.

A-4 Volume IV - Architecture Research Facility: ISP Description, Simulation, Data Collection

Volume IV discusses the use of the ISP machine architecture description language in describing the candidate architectures. It describes the ISP interpreter facility and its application to simulation of the candidates and in gathering the measurements discussed in Volume III.

A-5 Volume V - Procedure for and Results of the Evaluation of the Software Bases of the Candidate Architectures for the Military Computer Family

Volume V describes a menu of support software tools determined to be important to the development of military software. It discusses how a subset of those tools were selected as the necessary software base for the Military Computer Family and the results of a study to determine the availability and value of these tools.

A-6 Volume VI - Life Cycle Cost Analyses of the Computer Family Architecture Candidates

Volume VI describes the methodology used to compute and compare the life cycle costs of the CFA finalists and describes two life cycle models

(top-down and bottom-up) and the results of applying the methodology to those two models.

A-7 Volume VII - CFA/Software Licensing Discussions with the Three CFA Finalists (For Official Use Only)

Volume VII addresses the technical, financial, and legal issues arising out of discussions with the owner/manufacture of each candidate computer architecture and describes the outcome of these discussions.

A-8 Volume VIII - CFA Final Selection

Volume VIII discusses the consideration by the Selection Committee of the results of the architecture evaluations described in Volumes II through VII of this report. The influences that the various results had on the final selection are described.

A-9 Volume IX - A Consideration of Issues in the Selection of a Computer Family Architecture

Volume IX addresses questions and controversial issues regarding the CFA Selection process that arose from both within and without the Selection Committee during the course of the CFA effort.