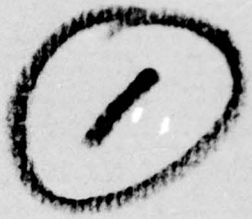
AD-A040 956 SCIENCE APPLICATIONS INC MCLEAN VA F/G 9/2
ADVANCED SIGNAL PROCESSING SOFTWARE VERIFICATION AND VALIDATION--ETC(U)
1976 N00014-75-C-0656
UNCLASSIFIED NL

1 OF 1
ADA040956

END

DATE FILMED
7-77



B.S.

526,466
(1)

SCIENCE APPLICATIONS, INC.

ADVANCED SIGNAL PROCESSING SOFTWARE VERIFICATION AND VALIDATION - FINAL REPORT

1976

Under Contract No.
N00014-75-C-0656

42p.

Prepared for:

Department of the Navy
Naval Research Laboratory
Washington, D.C.

SCIENCE APPLICATIONS, INC.
8400 Westpark Drive, McLean, Virginia 22101
(703) 821-4300

708407

TABLE OF CONTENTS

ADVANCED SIGNAL PROCESSING SOFTWARE V&V FINAL REPORT

I. PURPOSE OF THIS REPORT

The purpose of this report is to summarize the technical progress made by Science Applications, Inc. (SAI) on Contract No. N00014-75-C-0656 over the period June 1975 through August 1976. During these months, SAI personnel at the Naval Research Laboratory (NRL) were primary participants in the design, development, verification, and validation of the Architecture Research Facility (ARF) and the Computer Family Architecture Benchmark Test Programs. These two areas comprised the primary technical efforts in the Military Computer Family Architecture Project (CFA).

II. APPROACH OF THIS REPORT

This report will outline the objectives of the Computer Family Architecture Project and will summarize the tasks completed by SAI toward attaining those objectives. Included as appendices are copies of the monthly progress reports covering the course of the contract.

III. OBJECTIVES OF THE PROJECT

The Computer Family Architecture Project has as its stated mission the selection of a commercially available computer architecture that would provide the basis for a family of military computers in the 1980's. In order to make a valid selection, the parent project gave birth to sibling efforts involving the design, development, verification, and validation of (1) an Architecture Research Facility (ARF) and (2) a set of candidate architecture benchmark test programs. A computer architecture can be described on the ARF using the Instruction Set Processor Language (ISPL) for which the ARF has a compiler. Using the simulation capabilities of the ARF, the benchmark test programs were to indicate the performance characteristics of the candidate computer

to page 2

architectures in a quantifiable manner. The CFA selection committee could then select the "best" architecture based on quantitatively justifiable figures of merit derived from evaluating benchmark performance on the ARF simulator corresponding to each of the particular architecture descriptions. The ARF approach provided a controlled environment in which to effectively execute a comparative study of computer architectures at the instruction set level.

A. ARF Design and Development

Mr. Jeff Entwistle of SAI concentrated his efforts on the design, development, and verification of the Architecture Research Facility at NRL. Mr. Entwistle was instrumental in completing the following tasks on the ARF:

- Primary participant in the preliminary design of the software configuration of the ARF at NRL;
- Studied and analyzed the tradeoffs between the DEC-10 and PDP-11 machines and their associated operating system environments for satisfying ARF requirements;
- Having done a great deal of preliminary work in the BLISS systems programming language on both the DEC-10 and PDP-11 at NRL, he had a primary part in selecting the implementation languages used in the ARF development phase;
- Re-wrote many parts of the ISPL compiler from Carnegie Mellon University (CMU);
- Coordinated changes required for CMU's simulator to run at NRL;
- Designed and built a post-processor to the ISPL compiler to reformat the compiler output;
- Defined interface protocols for various modules of the ARF;
- Specified and coded many of the ARF functional modules.

B. CFA Benchmark Design and Development

Beginning in October 1975, Mr. J. Carter Crafford, Jr. joined the project and assumed responsibility for the benchmarking tasks in evaluating the candidate computer family architectures. To that end, Mr. Crafford performed the following tasks at NRL:

- Evaluated potential candidate CFA benchmark test programs;
- Designed, built, and maintained a program to quantitatively evaluate candidate computer architectures based upon selection criteria and normalization procedures defined by the CFA Selection Committee. This program would store and retrieve architecture data and selectively compute composite figures of merit for ranking the candidate computer architectures under evaluation. Results produced by this program provided the quantitative justification for the selection of the "finalist" architectures chosen at the February meeting of the CFA Selection Committee;
- Defined a benchmark program interface to the ARF. Designed and developed FORTRAN programs to generate ARF simulator command files from hexadecimal core image dumps of coded benchmarks;
- Defined and wrote the coding specification for the Fast Fourier Transform (FFT) benchmark test program;
- Coded the FFT, Character search, Bit test, set, or reset, and Runge Kutta Integration test programs for the three "finalist" architectures;
- Defined and coded test data sets and drivers for validating coded benchmark test programs;
- Assisted CMU researchers in computing S, M, and R measures for benchmark test programs.

The appendix includes a copy of Mr. Crafford's CFA benchmark development phase diary and comments requested of each of the programmers involved in the benchmark work.

APPENDIX A

CFA PROJECT PROGRESS REPORTS

17 May 1976

Naval Research Lab
4555 Overlook Ave., S.W.
Washington, D.C. 20375
Attn: William Smith

Subject: Computer Family Architecture Project Progress Report

Dear Sir:

In June 1975 Mr. J. Carter Crafford, Jr. and Jeff Entwisle of Science Applications, Inc. (SAI) began work on Contract No. N00014-75-C-0656 at NRL. This letter reports technical progress during the period June 1975 - May 1976. Technical progress consisted mostly of specification and coding of the Architecture Research Facility.

Early in the reporting period, much discussion was dedicated to selection of implementation language and operating system. SAI undertook analysis of many aspects of both PDP-10 and PDP-11 machines and various operating systems. FORTRAN was chosen as the implementation language for its purported transportability while the PDP-10 TOPS was chosen for the initial system.

During the middle of the reporting period, SAI re-wrote many parts of the CMU's ISPL compiler and coordinated changes to CMU's simulator. SAI also performed the following tasks at NRL:

- developed benchmark test program coding specifications;
- designed, coded, and maintained a quantitative selection criteria evaluation program used by the CFA selection committee to evaluate candidate CFA architectures;
- defined benchmark-ARF interface. Designed and developed programs to generate hexadecimal core image dumps used in building ARF simulator command files.
- coded benchmark programs for the PDP-11, IBM 360/370, and Interdata 8/32 architectures;
- designed and coded test data set drivers for validating coded benchmark test programs.

During the latter part of the reporting period, much design was undertaken by SAI especially with regard to the REFORMAT program and associated interfaces to the rest of the ARF. During the current month of May, SAI supervised

Science Applications, Inc. 1651 Old Meadow Road, Suite 620, McLean, Virginia 22101, (703) 790-5900

Other SAI Offices: Albuquerque, Ann Arbor, Atlanta, Boston, Chicago, Huntsville, La Jolla, Los Angeles, McLean, Palo Alto, Santa Barbara, Sunnyvale, and Tucson

the integration of the system and will validate and verify the coded benchmark test programs using the test data drivers previously developed, complete benchmark program development and debugging, and generate ARF simulator command files corresponding to the validated benchmark test programs for use in evaluating the candidate CFA architectures on the ARF.

Enclosed as Appendix A are internal NRL detailed status reports that incorporate the activities for the months of October 1975 through May 1976.

Sincerely yours,

SCIENCE APPLICATIONS, INC.

Lawrence P. Wiesen

Lawrence P. Wiesen
Director of Tactical Systems

Encl.: a/s

/ntp

STATUS REPORT FOR MONTH OF OCTOBER

TASK TITLE: Architecture Research Facility
LABORATORY TASK NUMBER: 54B02-31

REF : (a) NRL memo 5403-347A:JS:la of 12 Sep 1975

(1) Agenda for Design Meeting
(2) NRL memo 5403-422:HE:gls of 3 Nov 1975
(3) Summary of Design Meeting
(4) Revised ARF Schedule
(5) Proposed Milestones and Deliverables for ARF

1. Progress

a. The PDP-11 has been moved to Building 16, Room 238. In addition, two offices, Rooms 229 and 240, are now occupied by ARF personnel.

b. An ARF design meeting was held 8-10, 14-15 October. Mario Barbacci and Gary Barnes attended the 8-10 October sessions of the meeting. Enclosure (1) is the agenda followed at the design meeting. Enclosure (3) is a brief day-to-day summary of the meeting. This summary details decisions that were made during the course of the meeting. Included as enclosures to the summary are the documents written to support a decision, advise on a decision or to explain the decision made.

c. On 25 October, Dave Parnas held a meeting to discuss both the ARF and the selection criteria for CFA. Some of the decisions agreed upon at 25 October meeting reversed decisions made at the week-long design meeting. Enclosure (2) contains the results of the 25 October meeting.

d. The revised schedule (enclosure (4)) is based on the incremental design and implementation approach that is now being used by ARF. It is intended that the incremental approach to ARF will permit system integration and testing to proceed in parallel with system development. Consequently, less time at the end of the schedule (enclosure (4)) is devoted to system integration than was assigned in the previous schedule included in reference (a).

e. Proposed milestones and deliverables (enclosure (5)) are based on the schedule.

2. Plans for November 1975

a. Three programmers are being hired for work on ARF. All three are being hired as intermittent employees. One person will work full-time. The other two are students and are expected to work from 10 to 30 hours per week. It is hoped that the two part-time intermittents will start work the week of 17 November. The other will hopefully start 1 December.

b. Obtain a format for coding specifications.

c. Complete design and coding specifications of the RTM Sequencer.

d. Complete an interface design of the table access routines used by the RTM Sequencer, the CLI and the Reformat program.

e. Complete a coding specification for the table access routines for the PDP-10.

f. Have a minimal communications system for the PDP-10 and PDP-11 running.

g. Complete a document that describes the ARF run-time environment on the PDP-11 running under DOS.

h. Complete a description of the functions to be performed by the reformat program.

i. Investigate the utility of the CMU simulator for implementation of ARF state 2. Make minor modifications as necessary to the simulator in order to make it presentable to CFA at the 3-4 December meeting.

j. Plan to present the ISP compiler with possibly some post-processing at the 3-4 December CFA meeting.

k. Plan to present the stage 2 simulator to CFA at that meeting.

l. Complete a design and specification of a minimal CLI for stage 3 of ARF.

3. Problems

a. After the initial problems of deciding on the incremental approach, none.

4. Major Procurements

a. None.

5. Planned Procurements

a. License for Digital Equipment Corporation's disk operating system (DOS) for the PDP-11.

6. Milestones

a. See enclosure (5).

7. Trips and Visitors

a. M. Barbarcci and G. Barnes of CMU attended 8-10 October design meeting.

b. D. Parnas, S. Fuller, and D. Siewiorek attended 25 October meeting.

STATUS REPORT FOR MONTH OF NOVEMBER

TASK TITLE: Architecture Research Facility
LABORATORY TASK NUMBER: 54B02-31

(1) J. McHugh, et.al., "Table Interface Functions for the Architecture Research Facility", NRL Technical Memorandum 5403-431:JMcH:gls 19 Nov.

(2) J. McHugh, "PDP-10 Storage Layouts for ARF Stage 3", NRL Technical Memorandum, 5403-464:JMcH:gls of 26 Nov.

(3) G. Lloyd, "Running ARF stage 2 Software", Preliminary NRL Technical Memorandum of 3 Dec.

(4) J. Entwisle, "The NRL ISP Compiler", NRL Technical Memorandum 5403-467:JE:gls of 3 Dec.

(5) J. McHugh, "Minimal Stage 3 ARF CLI Design", NRL Technical Memorandum 5403-484:JMcH:dmf of 8 Dec.

(6) H. Elovitz, "The Architecture Research Facility: An Introduction", NRL Technical Memorandum 5403-472:HE:gls of 2 Dec.

1. Progress

a. Three programmers reported for work on ARF during the month of November. Their names are Michael Koster, Steven Mann and Alan Parker.

b. Enclosure (1) documents the interface design of the table access routines.

c. Enclosure (2) documents the PDP-10 table formats for the necessary ARF tables.

d. A minimal communications system for the PDP-10 and PDP-11 is running and is used to obtain listings from the PDP-10.

e. The CMU simulator is being used for ARF stage 2. Modifications are proceeding to make it a more viable simulator. Enclosure (3) is preliminary documentation of Stage 2 ARF.

f. Stage 1 ARF has been completed and is running. Enclosure (4) contains preliminary documentation on the post-prosessor and the NRL ISP compiler.

g. A design for the minimal CLI for ARF stage 3 is complete. Enclosure (5) is documentation of this design.

h. An overview document of ARF has been prepared. A draft of this document is enclosed (enclosure (6)).

2. Plans for December 1975

a. Present overview of the Architecture Research Facility at CFA Selection Committee meeting Dec.3-4 at Fort Monmouth.

b. Complete Stage 2 ARF.

c. Commence table access routine Coding.

d. Complete a reformat design.

e. Complete coding specification of the RTM sequencer.

3. Problems

a. None.

4. Major Procurements

a. None.

5. Planned Procurements

a. License for Digital Equipment Corporations disk operating system (DOS) for the PDP-11.

b. Modifications to above procurement to permit efficient operation in paged environment.

6. Milestones met

a. None proposed for this month, but Stage 1 ARF is running (Milestone for Dec 20)

7. Trips and Visitors

a. Greg Lloyd to CMU for purpose of learning about SIMU10, the simulator being used in ARF stage 2.

STATUS REPORT FOR MONTH OF DECEMBER

TASK TITLE: Architecture Research Facility
LABORATORY TASK NUMBER: 54B02-31

(1) H. Elovitz, "The Architecture Research Facility: An Introduction", NRL tech memo 5403-472:HE:gls of 2 Dec 1975
(2) M. Barbacci, CMU Inter-Office Correspondence of 16 Dec 1975
(3) M. Barbacci, CMU Inter-Office Correspondence of 29 Dec 1975
(4) R. A. Parker, "Error Message Routines for Table Interface Functions", NRL tech memo 5403-470:RAP:dmf of 17 Dec 1975
(5) J. McHugh, "Byte Handling Routines for the PDP-10", NRL tech memo 5403-478:JMcH:kt of 12 Dec 1975
(6) J. McHugh, "Half-Word Routines for the PDP-10", NRL tech memo 5403-515:JMcH:hg of 23 Dec 1975
(7) J. McHugh, "More Byte Handling Routines for the PDP-10", NRL tech memo 5403-497:JMcH:dmf of 15 Dec 1975
(8) M. Koster, "Control -C Intercept Routine", NRL tech memo 5403-3:MK:dmf of 6 Jan 1976
(9) J. Entwisle, "Preliminary Design Criteria for the REFORMATTER", NRL tech memo 5403-5:JE:dmf of 2 Jan 1976
(10) J. Entwisle, "Coding Specifications for the Reformat Program", NRL tech memo 5403-518:JE:dmf of 23 Dec 1975
(11) J. Shore, "Interface and Coding Specifications for RTM and ISP Breakpoint Flag Functions", NRL tech memo 5403-516:JS:btm of 22 Dec 1975
(12) J. McHugh, "FORTRAN Coding Standards for the ARF", NRL tech memo 5403-457:JMcH:dmf of 9 Dec 1975
(13) J. McHugh, "PDP-10 Storage Layouts for ARF Stage 3", NRL tech memo 5403-464:JMcH:gls of 23 Dec 1975
(14) J. McHugh et al., "Table Interface Functions for the Architecture Research Facility (ARF)", NRL tech memo 5403-431:JMcH et al.:gls of 22 Dec 1975

1. Progress

a) An overview of the Architecture Research Facility was presented at the CFA selection committee 4 December at Fort Monmouth (Revision enclosed (enclosure (1)).

b) An ISP seminar was schedule to be held at CMU on 14, 15 and 16 January with Mario Barbacci lecturing. Enclosures (2) and (3) indicate attempts to inform potential participants about the seminar.

c) Stage 2 ARF completed.

d) Error handler for table accessing routines completely specified (enclosure (4)).

e) Error handler coded and debugged.

f) Some utility programs have been specified. Code is proceeding. (Enclosures (5), (6) and (7)).

g) A control C Intercept routine has been specified (enclosure (8)).

h) Reformat design is complete (enclosure (9)).

i) First draft of the reformat coding specifications has been written (enclosure (10)).

j) The breakpoint interface for the RTM sequencer has been defined; coding specifications have been written (enclosure (11)).

k) FORTRAN coding standards have been proposed (enclosure (12)).

l) The PDP-10 storage layouts have been revised (enclosure (13)).

m) The table interface functions have been revised (enclosure (14)).

2. Plans for January 1976

a) Produce a formal RTM sequencer design document.

b) Complete the coding specification of the RTM sequencer.

c) Start and complete the coding of the RTM breakpoint interface.

d) Start coding the RTM sequencer.

e) Complete the SNOBOL preprocessor of FORTRAN code. This will substitute parameter values for parameter names in FORTRAN text, permitting more readable code.

f) Complete the coding of control -C intercept routine.

g) Complete coding specification of table access routine.

h) Start coding the table access routines.

i) Complete the coding of several specified utility programs (enclosures (5), (6) and (7)).

j) Complete the coding specification of reformatter.

k) Start the reformatting coding.

l) Complete the coding specification of minimal CLI.

m) Complete the PDP-11 table layouts.

n) Hire one or two intermittents to replace S. Mann and M. Koster.

3. Problems

a) Two programmers have resigned — Steve Mann as of 15 December; M. Koster will be transferring in February.

4. Major Procurements

a) None

5. Planned Procurements

a) License for Digital Equipment Corporations disk operating system (DOS) for the PDP-11.

b) Modifications to above procurement to permit efficient operation in paged environment.

6. Milestones Met

a) ARF presented at CFA (3-4 December)

b) Stage 1 ARF running (20 December)

c) Stage 2 ARF running (20 December)

7. Trips and Visitors

a) Honey Elovitz to Fort Monmouth to present ARF to CFA meeting.

f) The coding specification for the RTM sequencer is complete, (enclosure (4)).

g) The RTM breakpoint interface has been coded. Debugging is proceeding.

h) The interfaces for some Architecture Research Facility register functions have been specified, (enclosure (5)).

i) The code for the table access routines have been specified, (enclosure (6)).

j) Some selected table access routines have been released for coding.

k) The reformat program has been specified, (enclosure (7)).

l) One new intermittent has been hired, Paul Strauss.

m) The table access routines document has been revised to reflect a change in error handling. Enclosure (8) is version 5 of the table access document. A pre-processor for assigning function-identifiers and creating error messages has been written, debugged and documented, (enclosure (9)).

n) A testing package has been written, debugged and documented, (enclosure (10)).

o) A tty-spooling utility program has been written and is very useful.

2. Plans for February 1976

a) Start coding the RTM sequencer.

b) Complete the SNOBOL preprocessor of FORTRAN code. This will substitute parameter values for parameter names in FORTRAN text, permitting more readable code.

c) Start the reformat program coding.

d) Complete the coding specification of the minimal CLI.

e) Complete the PDP-11 table layouts.

f) Continue coding the table access routines as specifications are finalized.

g) Complete coding of the breakpoint interface for the RTM sequencer.

h) Complete the coding of PDP10-PDP11 communications interface.

3. Problems

a) None

4. Major Procurements

a) A GE-Terminet 1200 has been delivered on a leased basis. We are using this as a line printer for the PDP-10.

b) A procurement request has been submitted to purchase DOS and consulting services. Enclosure (11) is a copy of the request.

5. Planned Procurements

a) As indicated above

6. Milestones Met

a) None (no milestones were scheduled for January).

b) "How goes it"

Although no ARF milestones were scheduled for January, I consider paragraph 1 to list intermediate accomplished milestones.

Currently, we are behind in one milestone, the PDP-10/11 communications package. I do not consider this a major aspect of ARF and expect that it will take only several days to complete.

ARF Stage 3 is due to be operational 2 February and is an ARF milestone. This will not be met. ARF Stage 2 was operational much earlier than expected and turned out to contain capabilities much closer to the planned Stage 3 capabilities. Consequently, ARF Stage 3 has evolved into ARF Stage 4, which is scheduled as a 5 April milestone.

Currently, we are behind in our planned schedule. I believe this is because the design and coding specification documentation has taken much longer than we had planned for in our initial schedule. But, I feel that this schedule slippage is not indicative of a slippage in the final schedule. After producing several design document and coding specifications, we feel that coding and debugging will proceed more smoothly than otherwise.

7. Trips and Visitors

a) Honey Elovitz, Paul Strauss, Jeff Entwisle, and R. Alan Parker to CMU for ISP seminar.

STATUS REPORT FOR MONTH OF FEBRUARY

TASK TITLE: Architecture Research Facility
LABORATORY TASK NUMBER: 54B02-31

REF: (a) NRL memorandum 5403-56:HE:dmf of 2 Feb 1976

(1) NRL tech memo 5403-6:HE:dmf of 6 Jan 1976, RTM Sequencer Design for the Architecture Research Facility, Version 2
(2) NRL tech memo 5403-4:JS:HT:gls of 5 Jan 1976, Coding Specifications for the RTM Sequencer, Version 2
(3) NRL tech memo 5403-516:JS:btm of 22 Dec 1975, Interface and Coding Specifications for RTM and ISP Breakpoint Flag Functions, Version 4
(4) NRL tech memo 5403-19:GL:aka of 8 Jan 1976, Interface Specifications for ARF Register Functions, Version 2
(5) NRL tech memo 5403-99:GL:aka of 25 Feb 1976, Coding Specifications for ARF Register Functions
(6) NRL tech memo 5403-518:JE:JE of 22 Jan 1976, Coding Specifications for the Reformat Program, Version 2.3
(7) NRL tech memo 5403-77:JMcH:jmch of 19 Feb 1976, The ARF FORTRAN Pre-Processor
(8) ltr from M. Barbaci, New Command for NRLSIM of 22 Jan 1976
(9) NRL tech memo 5403-54:HE:aka of 30 Jan 1976, OPAQUE-DOPAQUE Feature Addition to ARF
(10) NRL tech memo 5403-62:HE:aka of 3 Feb 1976, OPAQUE-DOPAQUE and ENABLE-DISABLE
(11) NRL tech memo 5403-464:JMcH:gls of 23 Dec 1975, PDP-10 Storage Layouts for ARF Stage 3, Version 6
(12) NRL tech memo 5403-431:MCcH et al.:gls of 22 Dec 1975, Table Interface Functions for the Architecture Research Facility (ARF), Version 6
(13) NRL tech memo 5403-522:JMcH:dmf of 31 Dec 1975, Coding Specifications for Table Interface Functions, Version 2.0
(14) NRL tech memo 5403-457:JMcH:dmf of 9 Dec 1975, FORTRAN Coding Standards for the ARF, Version 3
(15) NRL tech memo 5403-479:JMcH:gls of 17 Dec 1975, String Handling Routines for the PDP-10 and PDP-11 FORTRAN, Version 3
(16) NRL tech memo 5403-57:RAP:aka of 2 Feb 1976, Test Routine for String Routines: STRTST
(17) NRL tech memo 5403-73:JE:JE of 11 Feb 1976, PDP-10 Storage Layout for Opcodes
(18) NRL tech memo 5403-75:JS:gls of 17 Feb 1976, Variation, Class, and Element Assignments for RTM Operation Types, Version 1
(19) NRL tech memo 5403-74:HE:JE of 11 Feb 1976, Interface and Coding Specifications for Variation, Class, and Element Functions

1. Progress

a) The RTM sequencer is being coded. New versions of the RTM design and coding specifications have been released (enclosures (1) and (2)).

b) A new version of the RTM and ISP breakpoint flag interface and coding specification has been released (enclosure (3)). The coding of this module is complete.

c) A new version of the interface specification for the ARF register functions is complete (enclosure (4)). Additionally, the coding specification for these functions is complete (enclosure (5)).

d) A new version of the Reformat coding specification is complete (enclosure (6)).

e) The SNOBOL preprocessor coding is completed. Documentation is enclosed (enclosure (7)).

f) An added capability has been added to the ARF at the suggestion of Mario Barbaci (enclosure (8)). After some discussion about the best way to implement the OPAQUE feature, (enclosure (9)) enclosure (10) explains the new capability. The PDP-10 storage layouts document (enclosure (11)) and the table interface functions document (enclosure (12)) had to be modified to provide support for the OPAQUEing feature. Additionally, the coding specifications for the table interface functions had to be modified accordingly (enclosure (13)).

g) A new version of the table interface routines coding specifications has been released (enclosure (13)). The binding functions and many of the get and set functions have been coded.

h) Several miscellaneous documents have been revised (enclosures (14), (15) and (16)).

i) Enclosures (17), (18) and (19) reflect a design decision to provide more needed information in the RTM operations opcode assignments.

2. Plans for March 1976

a) Start the reformat program coding.

b) Complete the coding specification of the Stage 4 CLI.

c) Complete the PDP-11 table layouts.

d) Complete the coding specification for selected paging functions needed for the PDP-10 version of ARF.

e) Complete the coding of the table interface routines.

f) Complete the coding of the ARF register functions.

g) Complete the coding of the PDP-10/11 communications module.

h) Determine the necessary modifications for PDP-11 DOS and proceed with obtaining these modifications.

i) Update and distribute the ISP primer.

3. Problems

a) None.

4. Major Procurements

a) None.

5. Planned Procurements

a) None.

6. Milestones Met

a) None (stage 3 was due to be operational 2 Febraury but as explained in last month's report (reference (a)) it is not).

b) "How Goes It".

During the course of the design and implementation of the Architecture Research Facility, we have learned that the semantics of the ISP language are very ill-defined. Consequently, we have been making many decisions that define the semantics of ISP. In some cases, our decisions may be unpopular with some of the ISP writers but, we feel that very few people know what they want or need. We have tried in all cases to logically decide upon the best solution. Some specific examples follow:

1) Logical operators

ISP permits the writer to define the length of ISP registers There is no restriction that the lengths of the two operands in a logical operation (AND, OR, etc.) be the same. This creates an ambiguity. Our choices were:

a) Truncate the longer operand on the right.

b) Truncate the longer operand on the left.

c) Extend the smaller operand on the left and zero-fill.

d) Extend the smaller operand on the right and zero-fill.

e) Extend the smaller operand on the left and 1-fill.

f) Extend the smaller operand on the right and 1-fill.

We decided that c) was the correct choice. (This seemed intuitively to be what is meant. Choice c) is also consistent with what occurs when registers of differing lengths are operands of other operations (i.e., add, subtract)).

2) Shift and rotates

The ARF performs most of the ISP operations on 64-bit temporary registers, into which the ISP register contents are placed prior to operation execution (right justified, zero-filled). In most situations the ISP writer wants the shift or rotate to be performed on the specified ISP register and not a 64-bit temporary register that contains the smaller ISP register's contents. But, an ambiguity arises. The destination register specified may be longer than the operand. Our choice was

a) Shift/rotate based on the operand's length.

b) Shift/rotate based on the destination register's length.

Our decision was a) (again because we felt that this was intuitively what was wanted). Had we chosen b), the possibility of the destination being smaller would add the problem of shifting (rotating) first and truncating or truncating first. Luckily, we can avoid this problem.

3) Carry-out

Operations are performed on 64-bit temporary registers. Carry-out bits remain until truncation occurs on storing into a destination register.

4) Masks

The ISP language provides the facility for a writer to explicitly extract/insert a sequence of bits from/into a register. This is specified by a mask that has the format ⟨N1:N2⟩ where N1 and N2 specify bit names in the register. The colon is a short hand method for denoting the range of bits "between" N1 and N2, inclusively.

An ISP a writer may define a register either as REG1⟨0:32⟩ or REG2⟨32:0⟩. Consequently, the meaning of REG1⟨5:2⟩ is unclear (as is REG2⟨5:7⟩). This is complicated by the possibility (and syntactic legality) of the following statement

REG1⟨2:5⟩ = REG1⟨5:2⟩.

The compiler apparently generates the identical mask for ⟨2:5⟩ and ⟨5:2⟩ above; the effect of the mask is identical. But, an ISP writer could expect bit reversal in this situation.

The ARF will clearly define the ISP mask semantics in the ISP PRIMER to keep the ISP writer informed of the semantics of the language.

5) Redefinitions

ISP permits the redefinition of an array or register over another array or register. This permits the redefined array or register to occupy the same or a subset of the storage area as the base array or register.

Unfortunately, the syntax of ISP permits the following redefinition: M[0:3]⟨2:0⟩:= MW[0:3]⟨2:0⟩

when MW has been declared

MW[0:50]⟨15:0⟩;

the above redefinition permits M to be specified over non-contiguous bit fields of MW. The ARF is not going to support non-contiguous redefinition. We feel that the cost of such support would be too high for a facility that doesn't seem necessary.

All of the cases mentioned above are examples of the necessity for us to define the semantics of ISP. We have documented all such decisions in the ISP Primer so that any ISP writer intending to use the ARF can determine the restrictions and limitation with which he will have to live.

7. Trips and Visitors

a) None.

1. Progress

a) The code for the RTM sequencer is complete. New versions of the RTM design and coding specifications have been released (enclosures (1) and (2)).

b) A new version of the RTM and ISP breakpoint flag interface and coding specification has been released (enclosure (3)). This was a result of additional error checking in the run-time ARF.

c) A new version of the interface specification for the ARF register functions has been produced (enclosure (4)). Additionally, the coding specification for these functions has been modified (enclosure (5)). Coding of these functions are complete.

d) A new version of the Reformat coding specification has been released (enclosure (6)). Coding of the Reformat program is complete.

e) Documentation for the preprocessor has been updated (enclosure (7)).

f) A new version of the table interface routines interface specification (enclosure (8)) and coding specifications have been released (enclosure (9)). These changes reflect the addition of the conditional counting capability.

g) The string handling utilities have been expanded (enclosure (10)).

h) A tech memo has been written to clarify the representation of ARF masks. The necessary functions to access the mask have been specified (enclosure (11)).

i) The necessary modifications to the PDP-11 DOS has been determined. Unfortunately, DEC bids approximately $6,000.00 to do the modifications required. We have decided to proceed in house with the modifications when possible.

2. Plans for April 1976

a) Debug and test the reformat code.

b) Test the table functions.

c) Complete the coding of the paging functions needed for the PDP-10 version of ARF.

NRL ISP compiler.

d) Complete the PDP-10-11 communications module.

e) Complete the coding of the minimal CLI.

f) Complete testing of the RTM sequencer.

g) Integrate the RTM sequencer table functions and CLI.

h) Integrate the Reformatter and NRL ISP compiler.

i) Get ARF 4 running on the PDP-10.

3. Problems

a) The modifications needed to the PDP-11 disk operating system (DOS) have turned out to be extremely expensive to have DEC do. Consequently, ARF personnel must do the necessary modifications.

4. Major Procurements

a) We are renting two TI portable terminals for inputting text to the PDP-10.

5. Planned Procurements

a) None.

6. Milestones Met

a) None (none were scheduled for this month).

b) "How Goes It".

Most of the code for the system to be brought up on the PDP-10 has been written. It needs to be debugged, tested, and integrated into one system. I hope to accomplish most of this within the next month. All of the code has been read by at least one person other than the coder (usually the designer of the module). It is interesting to note that we have found many bugs that would not have been caught until run-time by this review process. These errors have not been just in the code; several have been in the coding specifications where omissions or errors have been found while reading the code. This is interesting since all coding specifications were read several times prior to being released for coding in order to find such errors. An aside to the progress of the ARF is that Steve Crocker of ISI is interested in "consolidating the various ISP dialects" that appear to be evolving from several research projects.

He plans to have a meeting at ISI in May to discuss how to do the consolidation and what the dialects are. Attendees would be from CMU, NRL, DEC, University of Illinois, ISI and TRW.

7. Trips and Visitors

a) Mario Barbacci to NRL to discuss compiler modifications.

STATUS REPORT FOR MONTH OF APRIL

TAST TITLE: Architecture Research Facility
LABORATORY TASK NUMBER: 54B02-31

REF:
(a) Status and Capabilities of ARF II of April 26, 1976

(1) NRL tech memo 5403-516:JS:btm of 22 Dec 1975, Interface and Coding Specifications for RTM and ISP Breakpoint Flag Functions, Version 6
(2) NRL tech memo 5403-518:JE:JE of 22 Jan 1976, Coding Specifications for the Reformat Program, Version 4
(3) NRL tech memo 5403-522:JMcH:dmf of 31 Dec 1975, Coding Specifications for Table Interface Functions, Version 3.1
(4) NRL tech memo 5403-112:JMcH of 1 Mar 1976, Minimal Paging System Functions for The Unpaged ARF, Version 1.1
(5) NRL tech memo 5403-154:JMcH:rmg of 22 Mar 1976, A Do it Yourself CLI Kit for the Stage 3 ARF, Version 1.0
(6) NRL tech memo 5403-464:JMcH:gls of 23 Dec 1975, PDP-10 Storage Layouts for ARF Stage 3, Version 7
(7) NRL tech memo 5403-176:JE:je of 1 Apr 1976, Interface Specifications for System Functions
(8) NRL tech memo 5403-208:HE:rmg of 26 Apr 1976, The CLI-RTMSEQ Interface
(9) NRL tech memo 5403-145:HE:rmg of 15 Mar 1976, A Dump Package for the ARF, Version 3
(10) NRL tech memo 5403-110:HE:aka of 27 Feb 1976, ISP Primer for the Architecture Research Facility, Version 2
(11) Task Assignments for month of May

1. Progress

a) A new version of the RTM and ISP breakpoint flag interface and coding specification has been released (enclosure (1)). This was a result of a minor design change to add a function to clear all RTM breakpoints.

b) A new version of the Reformat coding specification has been released (enclosure (2)). Coding of the Reformat program is complete. Testing is planned at the completion of the testing of the table functions.

c) A new version of the table interface routines coding specifications have been released (enclosure (3)). These changes reflect the addition of the conditional counting capability.

d) The table functions have been tested.

e) The paging subsystem for the PDP10 has been designed and specified. (enclosure (4)).

f) The minimal CLI has been designed. (enclosure (5)).

g) The PDP10 storage layouts document was modified to reflect the addition of conditional count fields. (enclosure (6)).

h) Several system functions needed by the reformatter have been specified and coded. (enclosure (7)).

i) The CLI-RTMSEQ interface has been documented. (enclosure (8)).

j) A dump package has been designed and specified and coding has started. (enclosure (9)).

k) A technique for testing the RTM sequencer has been planned. It consists of writing ISP's that will exercise the paths in the sequencer. These ISP's are being prepared.

2. Plans for May 1976

a) Complete the coding and testing of the paging subsystem for the PDP-10.

b) Test the reformat code.

c) Complete the PDP-10-11 communications module.

d) Complete the specifications and coding of the minimal CLI.

e) Complete testing of the RTM sequencer.

f) Integrate the RTM sequencer, table functions and CLI.

g) Get ARF 4 running on the PDP-10.

h) Release a version of the ISP Primer.(enclosure (10)).

3. Problems

a) None

4. Major Procurements

a) None

5. Planned Procurements

a) None

6. Milestones Met

a) None(stage 4 and 5 ARF were to be operational on PDP10)

b) "How Goes It".

We have failed to meet the 5 April milestones. ARF 4 and 5 are not operational on the PDP10. This delay is not unexpected. As explained in a previous report, initial documentation and revisions required more time than estimated.

We hope to have ARF operational on the PDP10 some time in June. Enclosure (11) details the task assignments for ARF personnel for May. From this schedule it can be seen that all testing and coding of the ARF should be completed by May 28. In order to test the Reformatter, the table functions and paging system must be integrated into ARF. Additionally, to test the RTM sequencer, the table functions must be integrated. Consequently, the completing of the ARF testing should indicate that integration is complete. What remains is a thorough end-to-end test of ARF by several people outside of the ARF project or at least on the periphery.

Furthermore, several restrictions on the initial version of ARF will be removed during June. We intend to relax the redefinition restriction mentioned in enclosure (10) and provide the simulation of a BAIL-OUT RTM operation newly added to the ISP compiler. (Note: the compiler still is a moving target.)

The milestone for May 14 will not be met. This milestone required ARF to be operational on the PDP11. We have been unable to transfer any of the code to the PDP11 (the PDP10-11 communications package is not finished) and consequently do not know how transferable the code will be. Additionally, the tables and access functions must be redesigned for the PDP11 and recoded. We do not have the manpower to proceed with this effort in parallel with getting ARF operational on the PDP10. In fact, it is questionable whether a PDP11 version of ARF is really needed at this time. A fair amount of redesign and thinking must be devoted to the ARF design prior to a workable PDP11 ARF.

The PDP11 transfer should be ignored for the time being - at least until we get more feedback as to the utility of the PDP10 ARF design and capability. An operational PDP10 ARF will provide this information.

Stage II ARF has been operational for several months. Several modifications have resulted in a much more useful and reliable simulation than expected. Consequently, this stage of ARF will be able to serve any immediate demands of the CFA committee. In fact, reference (a) indicates that the PDP11 ISP is completed and has been debugged using ARF II. Additionally, D. Siewiorek and M. Barbacci are debugging PDP11 programs using the stage II simulator.

7. Trips and Visitors

a) None

15 June 1976

Naval Research Lab
4555 Overlook Avenue, S.W.
Washington, D.C. 20375

Attn: William Smith

Subj: Computer Family Architecture Project Progress Report

Dear Sir:

The purpose of this letter is to report technical progress for this reporting period, May 1 - May 31, 1976.

During the month, SAI personnel completed the following tasks at the Naval Research Laboratory:

- Completed Phase 1 Benchmark Test Programs for the Computer Family Architecture Project Candidate Architecture Evaluation;
- Completed development and documentation of Benchmark Test Program Drivers for Phase 1 and 2 code debugging preliminary to establishment of final test data sets;
- Entered test run evaluation phase of Phase 1 Test Programs;
- Continued development of NRL's Architecture Research facility.

In the forthcoming month of June, SAI will complete the following objectives:

- Complete Phase 1 and 2 Benchmark Test Program development, debugging, and test evaluation;
- Continue development of Architecture Research facility.

Sincerely yours,

SCIENCE APPLICATIONS, INC.

J. Carter Crafford, Jr.

J. Carter Crafford, Jr.
Scientist 2

JCC:jf

Science Applications, Inc. 1651 Old Meadow Road, Suite 620, McLean, Virginia 22101, (703) 790-5900

Other SAI Offices: Albuquerque, Ann Arbor, Atlanta, Boston, Chicago, Huntsville, La Jolla, Los Angeles, McLean, Palo Alto, Santa Barbara, Sunnyvale, and Tucson

12 July 1976

Naval Research Laboratory
Code 5493
Washington, D.C.

Attn: William Smith

Subj: Advanced Signal Processing Software V&V Progress Report under
Contract No.: N00014-75-C-0656

Dear Sir:

During the month of June 1976, SAI personnel at the Naval Research Laboratory continued their support function to the Computer Family Architecture Project. Using test data sets and test data drivers previously defined and built by these same personnel, they have been the key point of contact for the verification of the CFA Benchmark Test Programs developed at the Naval Research Laboratory and Brown University.

In the month of July 1976, SAI will complete benchmark test program verification in preparation for their use in the final evaluation of the candidate architectures using the Architecture Research Facility (ARF) at Carnegie-Mellon Institute. SAI personnel will be primary participants in these final evaluations.

Sincerely yours,

SCIENCE APPLICATIONS, INC.

J. Carter Crafford, Jr. (ntp)

J. Carter Crafford, Jr.
Scientist

/ntp

Science Applications, Inc. 1651 Old Meadow Road, Suite 620, McLean, Virginia 22101, (703) 790-5900

Other SAI Offices: Albuquerque, Ann Arbor, Atlanta, Boston, Chicago, Huntsville, La Jolla, Los Angeles, McLean, Palo Alto, Santa Barbara, Sunnyvale, and Tucson

APPENDIX B

BENCHMARK DEVELOPMENT PROGRESS REPORT

11 August 1976

Naval Research Laboratory
4555 Overlook Ave., S.W.
Washington, D.C. 20375

Attn: William Smith

Subj: Computer Family Architecture Project Progress Report

Dear Sir:

This letter reports the technical progress made during the period of July 1976 on Contract No. N00014-75-C-0656 at NRL. During this time, SAI personnel participated in the final phases of development and evaluation of benchmark test programs that will determine the quantitative justification for selecting a military computer family architecture.

Early in the reporting period, SAI concentrated its efforts on verifying the Runge Kutta Integration and Fast Fourier Transform benchmark test programs. In mid-July, SAI personnel travelled to Carnegie-Mellon University to assist in computing S, M, and R measures for the statistical analysis of the architectures based on a defined set of benchmark programs. Since that trip, SAI has concentrated on validating the FFT benchmarks. Inaccessibility of the required hardware and software features hampered both the development and verification phases of this test program. This situation will have similar effects on the validation phase.

Throughout the month of August, SAI will complete the validation of these benchmarks. Included in this letter as an appendix is a copy of the progress report diary and comments covering the CFA benchmark development cycle that was sent on request to the architecture evaluation team at Carnegie-Mellon University.

Sincerely,

SCIENCE APPLICATIONS, INC.

J. Carter Crafford, Jr

J. Carter Crafford, Jr.

Encl.: a/s

/ntp

Science Applications, Inc. 8400 Westpark Drive, McLean, Virginia 22101, (703) 821-4300

Other SAI Offices: Albuquerque, Ann Arbor, Arlington, Atlanta, Boston, Chicago, Huntsville, La Jolla, Los Angeles, Palo Alto, Santa Barbara, Sunnyvale, and Tucson.

CFA PROGRESS REPORT DIARY AND COMMENTS

TO: David Lamb, Carnegie Mellon University

FROM: J. Carter Crafford, Jr., NRL Code 5493

DATE: 26 July 1976

INTRODUCTION

Under a contract with the Naval Research Laboratory from my company, Science Applications, Inc., I began working on the Army-Navy Computer Family Architecture Project in October of 1975. Beginning in February 1976, I was assigned the task of (1) understanding and (2) writing the coding specification for the COOLEY-TUKEY FAST FOURIER TRANSFORM Algorithm Benchmark Test Program. This endeavor eventually produced the coding spec used in the evaluation. What follows is a general weekly diary of my efforts during the benchmark program development cycle.

BENCHMARK PROGRAMMER DIARY

March

Spent greater part of March learning the PDP-11 memory management scheme and programming the FFT benchmark (D) for the 11.

April

5-9

Wrote a FORTRAN driver for the FFT and a FORTRAN FFT subroutine to validate the driver.

12-23

Learned IBM and INTERDATA assembly languages from principles-of-operations manuals and gained general familiarity with machine operations.

26-30

Wrote a FORTRAN driver for bit test, set, and reset benchmark (F). Generated test data set to debug benchmark programs.

NOTE

Since no decision had been made to date as to how we were to initially debug test programs we wrote, we took the initiative to develop FORTRAN drivers and define test data sets to validate our code. Since the decision was made not to pass all values by reference, interface routines also had to be built. These routines turned out to be troublesome, time consuming, and in the final analysis, inconsistent with the calling conventions assumed at CMU.

May

3-7

Coded PDP-11 and IBM character search benchmarks (E). Defined test data sets for character search and linked list (H).

10-14

Wrote character search and linked list drivers. Tested linked list on PDP-11 and character search on INTERDATA 7/32.

17-21

Refined IBM character search. Wrote Runge Kutta integration (G) driver and Runge Kutta FORTRAN subroutine for testing.

24-28

Wrote both the IBM and INTERDATA FFT Benchmark (D) programs. Re-iterated on IBM and 8/32 character search code. Instructed Dahlgren DoD programmer in use of NRL DEC-system 10 for program development.

June

1-4

Wrote INTERDATA bit test, set, and reset benchmark (F) program from Phase II assignments. Clean up and keypunch drivers and benchmark code in preparation for benchmark testing at INTERDATA marketing office on 7/32.

NOTE

At NRL the only hardware we had immediately available was the DECsystem 10 with a MACY-11 assembler and a PDP-11/40 with no mulitply, divide, extended instruction set, floating point, or memory management. We ran our INTERDATA programs on a 7/32 at their local marketing office in which there was but one fellow who knew how to operate the machine. On the IBM 370 at the Army Computer Systems Command, we discovered how to get assemblies but never how to link edit a FORTRAN driver with an assembly subroutine; hence, no debugging except what we did on one pass at CMU.

7-11

Further developed the FFT driver. Documented and ran test cases of all drivers (for test programs D, E, F, G, and H) with FORTRAN subroutines. Sent documentation on drivers for facilitating initial debugging to Marty Michel, CMU, and other DoD programmers.

14-18

Assembled and cleaned up IBM character search and FFT benchmarks. Tested and debugged PDP-11 linked list and character search test programs. Received tape from Marty Michel with PDP-11 FFT, character search, linked list, and Runge Kutta as well as card decks for INTERDATA FFT, Runge Kutta, I/O interrupt kernel A, and device handler. Began assembly and debug phase of Michel's programs.

Spent week in Pittsburgh at CMU calculating S, M, and R measures for the FFT benchmarks.

21-25

Compiled FORTRAN drivers on IBM. Tested linked list and character search on INTERDATA. Tested and debugged all but floating point requisite programs on PDP-11. Assembled and debugged on INTERDATA. Wrote Runge Kutta (G) benchmark for INTERDATA from Phase II assignments.

June 28-2 July

Debugged and tested INTERDATA benchmarks for sending to CMU. Attempted to link and run benchmarks and drivers on IBM without success by virtue of incorrect JCL. Tested and debugged all but floating point requisite benchmarks for PDP-11.

July

6-9

Concentrated on test/debug of FFT and Runge Kutta benchmarks. We successfully tested the Runge Kutta programs Marty Michel and I had written. Lacking an operational on-line debugger for the INTERDATA, we never successfully tested the FFT benchmarks. Having no floating point processing capability available on our PDP-11, I implemented, debugged, and tested a fixed-point version of the FFT benchmark. Since the coding specs did not allow such a version, this effort simply demonstrated the feasibility of achieving the algorithm. Attempted to have made by an INTERDATA analyst, a tape of assembler listing files. Mario informs me that they were not entirely readable.

12-16

Spent this week prior to going to CMU trying to learn how to run the ARF simulator and to run the INTERDATA FFT benchmark. Prepared for Pittsburgh by attempting to learn how to calculate S, M, and R measures. Compiled assembly listings of all FFT benchmark test programs in preparation for S, M, and R measure calculations.

19-23

Spent week in Pittsburgh at CMU calculating S, M, and R measures for the FFT benchmarks.

PERSONAL COMMENTS ON THE CANDIDATE ARCHITECTURES

DIGITAL EQUIPMENT CORPORATION PDP-11

Of the three finalist architectures, the PDP-11 was the only one with which I had had previous experience. Although I like the consistency of the left-to-right operand positioning some would argue, and have, that it is "the traditional convention of the world" to have source and destination operands switch positions dependent upon the particular instruction to which they belong. Of course, one great advantage and joy in programming the PDP-11 is the fantastic power of its addressing modes. This power, however, somewhat loses it's charm for the poor soul charged with calculating S, M, and R measures, as I am painfully aware. Had I my way in light of my experience with the FFT benchmark on this architecture, I would desire load and store multiple instructions as well as an increase in the number of available registers both for the floating point and general register sets. The saving grace here is the wonderful system stack with which any PDP-11 programmer soon becomes intimately familiar.

Bring on a 32-bit version of the 11 so we can address directly what we will and let memory management take a back seat. Make the 32-bit model upward compatible with the 16-bit software as has INTERDATA and we shall have no equal.

INTERDATA 8/32

I have yet to lay eyes on a true blue 8/32. My experience of the last four months with INTERDATA and it's machine is somewhat tainted I must say by the local Washington, D.C. marketing office. Although the people there were kind enough to allow us to use the facilities with which they had been "blessed", which I trust did not consist in any way of the top of their line in hardware or software, we were dependent upon one man in the entire office who knew the operation of the machine and it's operating system. Over the course of visits, we were able to learn enough of the job control language to edit, assemble, compile, link, and execute our test programs and their drivers. In attempting to debug the FFT benchmarks, we discovered no operational debugging aid existed. Mario can relate the results of the INTERDATA analyst's attempt to write a tape with the assembly listings of our files.

On the brighter side of INTERDATA, I must confess I generally like the INTERDATA architecture. The 32-bit word structure provides the desirable addressability. The vector-like interrupt pointer table with the automatic register set switching seems to me to have the advantage of the PDP-11's vectored interrupt structure. The bit test, set, and reset instructions made that benchmark a snap. I am told that tactical communications systems do a great deal of bit manipulation and these instructions might present some advantages there. The large number of floating point registers was advantageous for the FFT.

In short, I see the INTERDATA 8/32 as the happy medium between the advantages and disadvantages of the IBM 370 and the PDP-11. Too bad INTERDATA hasn't been around long enough to develop some decent software.

INTERNATIONAL BUSINESS MACHINE'S 370

Here again, I must confess to a tainted experience with the IBM 370 since we were able to get no further than assembling our code on the Army Computer System Command's facility running in a batch mode with one day turnaround. This initial experience of observing Mr. Burr rastle with the glorious job control language leads me to conclude that one has here been given the capabilities to do so much that he oft finds himself able to accomplish very little. In our experience, more effort went into attempting to debug the JCL than the benchmark programs themselves.

Many I hear bemoan and detest the name of IBM and in my recent experience as a virgin to its charms, I am tempted to join their ranks. IBM never seems to stop short on anything and their instruction set would fill even the most ravenous epicurean. Some items on their menu, the TRANSLATE and EXECUTE instructions for example, the novice IBM programmer, myself for example, might well gloss over until one more experienced in their use condescends to enlighten him.

From what I've seen and heard of IBM's interrupt structure, I would prefer to avoid the same. Having grown up with the PDP-11 and other minicomputers, I prefer to get my hands on the machine.

I suppose then my present position on the IBM 370 is that I should reserve judgment until such time I have opportunity to gain more experience in its virtues. It seems to me one cannot intelligently critique anything or anyone until they know well their subject. Nevertheless, I think the IBM will prove itself not the best choice here regardless of my opinion.

SOME PERSONAL COMMENTS ON THE PROJECT AS A WHOLE

I believe in theory the CFA project is laudable in its efforts to make a quantitatively justifiable and thereby, it is hoped, intelligent selection of a commercially available computer architecture for military tactical applications in the 1980's. In hindsight, it would perhaps have been more cost effective to ship the entire project to Carnegie-Mellon University thereby capitalizing on not only the technical expertise and computer hardware facilities available there, but also facilitating centralization of both the managerial and developmental functions. Geographic separation of the management, the work force, and the equipment necessary to the success of the project has caused undue problems in communication of managerial decisions affecting the entire work force, the completion of work assignments to specification and according to the deliverable time schedule, and to the overall cost effectiveness of the project.

A case in point is the definition of benchmark test data sets and driver interfaces. We here at NRL took the initiative to define and build a set of test data drivers and their associated interfaces for the purpose of debugging our coded benchmarks. To this end and lacking further information, we made a set of assumptions regarding the interface addressing passed parameters that proved incompatible with those allowed. The result was duplication of effort and the necessity for reprogramming at CMU of all the benchmarks developed at NRL.

Since an ARF existed at CMU, the funds expended in the specification of the same at NRL could have perhaps been more wisely spent albeit such a conclusion is most probably validly drawn only in hindsight.

It is deplorable that the Navy's funding situation of late has somewhat drooped the sails. Nevertheless, I trust what is learned from this project does not on deaf ears fall.