Overview of the military computer family architecture selection

by WILLIAM E. BURR and AARON H. COLEMAN
US Army Electronics Command
Ft. Monmouth, New Jersey
and
WILLIAM R. SMITH
Naval Research Laboratory
Washington, DC

ABSTRACT
This paper presents an overview of the selection process employed to choose a single Computer Family Architecture (CFA) to be used in a new Military Computer Family (MCF) intended for use in Army and Navy Systems. A joint Army/Navy Selection Committee studied the suitability of a number of architectures, and intensively evaluated three "final candidate" architectures, before selecting one, the PDP-II, for use with the MCF.

INTRODUCTION
Since early 1975 the Center for Tactical Communications Sciences (CENTACS) of the Army Electronics Command, and the Naval Research Laboratory have been engaged in a cooperative project to develop a software compatible family of military computers, based upon a common architecture, and suitable for a wide range of military land, sea, and air applications. That project is known as the Military Computer Family (MCF) Project, and the computer architecture to be used by the MCF is known as the Computer Family Architecture (CFA).

The MCF is based upon a strategy that included the following:
• Selection of architectural design or designers unbundled from implementation or implementers.
• Standardization on architecture design as the foundation on which software investment is made.
• Consideration of commercially successful architectures for which software already exists as candidates for DOD adoption.
• Technology independence, that is the anticipation of multiple implementations of the same architecture, implementations which might differ in technology (e.g., semiconductor vs magnetic memory), environmental specifications (e.g., volume or power constraints), or reliability assurance (e.g., MIL-qualification vs warranties or incentives).
• Multiple sources of supply for the various processors and other modules of the family.
• Support (probably via emulation) of existing software for the principal existing Army and Navy military computers.

The first step in the development of the MCF was the selection of the architecture to be used. The selection was made by an Army/Navy CFA Selection Committee during the period between October 1975 and August 1976 as a result of evaluating and comparing candidate architectures. The CFA committee began with the initial selection of nine candidate architectures, narrowed the initial set to three finalists: the IBM S/370, the DEC PDP-II, and the Interdata 8/32, and finally chose the PDP-II. This paper discusses the basic premises of the CFA selection process, and summarizes the actions of the Selection Committee.

WHAT IS A MILITARY COMPUTER?
Computers are used in the DOD for a wide range of applications. Many administrative, research and laboratory applications are run on the same commercial computers which are used in industrial and business applications. Many military computer applications, however, require "militarized" computers, which can operate in battlefield, shipborne, and airborne environments, and survive exposure to severe shock, vibration, radiation, and thermal stress. The applications for these computers are usually similar to the applications of commercial OEM computers, that is they are usually embedded in some larger system, such as a missile system, or a radar, and the computer itself is just one component, and not necessarily the most important component, of a larger system.
For the purposes of this article and the five articles which follow it, it is the hostile physical environment—not the types of computations, response times, data rates, or throughput requirements—which fundamentally distinguish the commercial OEM application/computer from the military application/computer.

RATIONALE FOR A COMPUTER FAMILY
ARCHITECTURE (CFA)

The Army and Navy currently use and maintain an inventory of over one hundred different computer types. Practically all of these machines have a design personality which must be catered to through specialized software and maintenance support. This inventory is regularly justified as the only means of applying processing capability where it is needed with a minimal cost investment. That is, off-the-shelf procurement or specialized designs aimed at specific operational applications is deemed to be the only satisfactory way of meeting the wide range of speed, power, weight, size, etc., requirements imposed by these applications.

There is little or no argument against satisfying a multitude of environmental and processing speed requirements through a combination of machines of varying hardware technological characteristics. Physical constraints ranging over orders of magnitude leave little choice but to meet these head-on with suitable device technologies, if cost-effective weapons systems are to be put into operation. The current proliferation of computer types is more a de facto result of platform and project managers trying honestly to choose the most appropriate machines out of a sea of unrelated available computer types, than a result of unformed procurement practices. Moreover, platform and project managers face heavy pressures to locally optimize the costs and schedules of their own projects and relatively little direct pressure to reduce the long term life cycle costs of both hardware and, particularly, software.

The greatest penalties arising out of such proliferation are in the efficiency, timeliness and both non-recurring and recurring costs of system software. However, it is not necessary for differences in computer technology requirements to mandate differences in software characteristics. A number of examples of commercial capitalization on this principle are well known—the IBM 360/370 and the Digital Equipment Corporation PDP-11 product lines. What has been gained from this approach is a line of computers of varying processing capabilities but which are software compatible and enjoy the support of a common set of system and applications programs. The main goal, then, of the Computer Family Architecture is to provide the Army and the Navy with a design for a series of computers (a family) with the variety of members necessary to satisfy the requirements of various platform and battlefield applications, while at the same time providing a single software system capability which will serve each and every member of that computer family.

Central to the success of the Computer Family is the selection and precise specification of the family architecture. The term “computer architecture” means quite different things to different people, so a definition is necessary. Here we follow the example of S/360. In an introductory paper on the IBM S/360, Amdahl, et al., defined computer architecture as: “The term architecture is used here to describe the attributes of a system as seen by the programmer, i.e., the conceptual structure and functional behavior, as distinct from the organization of the data flow and controls, the logical design, and the physical implementation.”

This definition of architecture specifically excludes details of hardware implementation. The instructions and registers which programmers “see” are part of the architecture, but the data buses are not. For example, the IBM 360/30 used 8 bit data paths, the 360/40 used 16 bit paths, and the 360/50 used 32 bit paths, but all three are the same architecture, and can execute the same programs. As another example, the PDP-11 Unibus is not an integral part of the architecture (indeed PDP-11’s have been built with at least three different bus structures), but the use of dedicated memory locations for communication with I/O devices is an architectural feature, because the programmer does not see the unibus, but he does see the dedicated I/O registers.

Selection of the CFA was guided by the principle that the bulk of computer processing improvements over the last two decades have risen out of technology advances rather than out of architectural changes, and that this principle is likely to remain in effect for at least another generation or more of computer systems. It is more promising, then, for the Army and Navy to adopt an already successfully demonstrated extant computer architecture, commercial or military, and to use that architecture to reap the benefits of technology advances while enjoying the benefits of software stability. The selection of an existing architecture carries with it an understanding of the strengths and weaknesses of that architecture and also a useful inventory of support and applications software already developed.

OTHER LEVELS OF STANDARDIZATION

The MCF has chosen to standardize at the instruction-set level. Many other levels of standardization are conceptually possible. One attractive alternative might be to standardize on a single Higher Order Language. There are efforts under way to fix a single HOL for DOD, and these efforts do not conflict with the MCF approach. However, there are a number of problems with this approach:

- HOL’s are much more complex than instructions sets. Consequently no two different compilers for the same language have ever been truly compatible, as anyone who ever tried to convert any large set of programs from one “standard” FORTRAN or COBOL compiler to another will attest. Differences between supposedly compatible compilers becomes very difficult to resolve.

From the collection of the Computer History Museum (www.computerhistory.org)
Another alternative might be to standardize on the assembly language, rather than the instruction set. An example of such standardization is found in the Interdata Family of 16 and 32 bit computers (5/6, 6/6, 7/32 and 8/32) which have similar, but not identical 16 and 32 bit architectures, and which rely on a Common Assembly Language and a "smart" assembler program to resolve the differences. This approach is attractive, but requires that the underlying architectures be quite similar. In particular, programs that are to be transferable from one architecture to another have to be written to avoid incompatibilities and thus it may not be possible to take full advantage of either.

A third alternative would have been to settle upon a standard "micro architecture." This approach assumes that all future military computers will be microcoded (not an unlikely assumption), and asserts that it is the internal register-level architecture of the processor, and the micro-code which should be standardized. This approach, and the use of read/write microstore, would permit application-tailored macro instruction-sets. This approach, however, has a number of disadvantages, including the following:

- The micro-architecture of a computer is much more directly related to the performance capabilities of the computer, than is the instruction set. It is not clear that a single micro-architecture can effectively satisfy a wide range of performance requirements.
- Micro-architectures are closely related to component technology and hardware design. Since rapidly improving device technology is the driving force in the computer industry, it would be a mistake to select a micro-architecture which is more or less closely tied to contemporary technology.
- The configuration control of the firmware needed to implement a number of tailored instruction sets, or user developed microcode, and of compilers and operating systems for different tailored macroinstruction sets, promises nightmare-like problems in the diverse environment of military laboratories, system centers, project and platform managers, and system developers (contractors).

Standardization at the computer architecture level is the safe, proven and accepted approach. It is the only answer to complete software transportability across a wide range of computer implementations, which has stood the test of time in industry-wide applications. The success of the IBM S/360 and S/370 families, the PDP-11 family, and several other instruction set compatible families, have demonstrated the practicality of this approach. Amdahl and National Semiconductor have proven that independent manufacturers can build compatible versions of a well-defined architecture (the S/370). The CFA approach is based upon the premise that the Army and Navy should try to take maximum advantage of existing commercial technology, rather than try to push it in new directions.

THE SELECTION COMMITTEE

The first task of the Army/Navy cooperative effort was the selection of the computer family architecture to be used. The Naval Research Laboratory (NRL) led this effort for the Navy under the sponsorship of the Naval Air Systems Command. NRL and CENTACS agreed to perform this task.

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 selection process as members of the CFA Selection Committee. Ten Army and seventeen Navy organizations assigned representatives to participate in the Selection Committee.

The Army and Navy cooperative effort has been entirely voluntary, and was not imposed upon the Army and Navy by DOD. It resulted from the discovery that both the Army and Navy independently had similar efforts under way, from the belief that military data processing requirements in the three services are similar, and from the realization that the combined funding and application bases of the Army and Navy would enhance the success of such a program. Air Force observers have attended the Selection Committee Meetings, and are participating in the next phase of the MCF project, involving systems implementation, or product planning.

SUMMARY OF THE SELECTION PROCESS

The CFA Selection Committee held five meetings between 1 October 1975 and 26 August 1976. The procedure developed by the committee for selecting the architecture is depicted in Figure 1 and may be summarized as follows:

a. Select Initial Candidates—The Committee approved a list of nine candidate architectures for examination. Table I shows nine candidate architectures.

b. Establish Initial Ranking Procedure—The Committee developed a set of "absolute" and "quantitative" criteria as measures of computer architecture effectiveness for a wide range of military computer-based systems applications.

c. Evaluate the Candidate Architectures—Subcommittees were established to evaluate each candidate architecture in accordance with the established absolute
BURROUGHS B6700
IBM 370
INTERDATA 8/32
GYK-12
PDP-11
ROLM 1664
SEL 32
UYK-7
UYK-20

Absolute Criteria
IBM 370
PDP-11

Quantitative Criteria
INTERDATA 8/32
PDP-11
IBM 370
GYK-12
ROLM 1664
BURROUGHS B6700
SEL 32
UYK-7
UYK-20

Test Program Analysis: S, M, R Measures
INTERDATA 8/32
PDP-11
IBM 370

Support Software Analysis
IBM 370
PDP-11
INTERDATA 8/32

Data Rights Licensing Analysis
IBM 370
PDP-11
INTERDATA 8/32

Top-down Life Cycle Cost Analysis
PDP-11
IBM 370
INTERDATA 8/32

Bottom-up Life Cycle Cost Analysis
PDP-11
IBM 370
INTERDATA 8/32

Final Decision
PDP-11
IBM 370
INTERDATA 8/32

INITIAL SCREENING

DETAILED ANALYSIS

DECISION

Figure 1—CFA selection process
### TABLE I—CFA Candidate Scores on Absolute and Quantitative Criteria

<table>
<thead>
<tr>
<th>ARCHITECTURE</th>
<th>QUANTITATIVE CRITERIA</th>
<th>ABSOLUTE CRITERIA</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTERDATA 8/32</td>
<td>1.68 (BEST)</td>
<td>Problem with interrupts and traps</td>
</tr>
<tr>
<td>PDP-11</td>
<td>1.43</td>
<td>Passed all</td>
</tr>
<tr>
<td>IBM S/370</td>
<td>1.36</td>
<td>Passed all</td>
</tr>
<tr>
<td>AN/GYK-I2</td>
<td>.94</td>
<td>Failed floating point</td>
</tr>
<tr>
<td>ROLM/NOVA</td>
<td>.92</td>
<td>Failed virtual memory mapping and interrupts/traps</td>
</tr>
<tr>
<td>B6700</td>
<td>.91</td>
<td>Failed protection</td>
</tr>
<tr>
<td>SEL-32</td>
<td>.86</td>
<td>Failed virtual memory mapping</td>
</tr>
<tr>
<td>AN/UYK-7</td>
<td>.46</td>
<td>Failed floating point</td>
</tr>
<tr>
<td>AN/UYK-20</td>
<td>.44 (WORST)</td>
<td>Failed protection</td>
</tr>
</tbody>
</table>

and quantitative criteria. Table I shows the list of nine candidate architectures and their relative performance in this evaluation. Fuller, Stone and Burr describe the criteria and the evaluation process in detail in their paper.

d. **Selection CFA Finalists**—The Selection Committee reviewed the architecture evaluations in detail, and selected three candidate architectures: the IBM S/370, the DEC PDP-11, and the Interdata 8/32 as CFA finalists for further examination.

e. **Describe the Finalists in ISP**—The three final candidate architectures were described in a formal register transfer language, ISP. These ISP descriptions were used to simulate the candidate architectures, and collect the data required for the test program evaluation. Barbacci, Siewiorek, Gordon, Howbrigg, and Zuckerma describe the use of ISP.

f. **Test Program Evaluation**—Just over 100 test program "kernels" were coded by 16 programmers to evaluate the relative efficiency of the three final candidates. The results of this evaluation are summarized in Table II. Fuller, Burr, Shaman, and Lamb describe the test program evaluation in their paper.

g. **Support Software Base Evaluation**—A subcommittee was formed to evaluate the support software bases of the three final candidates. The results of this evaluation are summarized in Table II, and Lieblein, Wagner and Stone describe this evaluation.

h. **Life Cycle Cost Analysis**—A subcommittee was formed to evaluate comparative life cycle costs of the MCF for each of the three final candidates. Two different analyses were performed, one using a "top-down" model and the other using a "bottom-up" model. These evaluations are described by Cornyn, Coleman, Smith and Svirsky.

i. **Licensing**—A series of meetings were held with the manufacturers of the final candidates to establish proposed licensing arrangements for the CFA finalists. Due to the confidentiality of the licensing discussions, they are not reported on here, but they were a significant factor in the final selection.

j. **Final CFA Selection/Recommendation**—All the data acquired in the preceding steps was reviewed and the Committee voted the relative ranking of the CFA finalists.

### RESULTS

The Selection Committee held its fifth and final meeting on 24-26 August 1976 at the Naval Underwater System Center, Newport, R. I., for the purpose of selecting the recommended architecture for the MCF. At that meeting the data discussed in the preceding sections of this report was considered at length. The data considered in that discussion is summarized in Table II.

Based upon the data presented in Table II, 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.
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.

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; this compromised the S/370’s performance in the life cycle cost evaluations. 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 program evaluations, 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 (microprocessor) 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.
REFERENCES
