MICROCOMPUTER STANDARDIZATION CONCEPTS

W. J. Dejka
U.S. Naval Electronics Laboratory Center

Introduction

As has often been observed, the microcomputer is causing a major revolution in system design and implementation. The advent of microcomputers has made it possible to achieve many designs that are unique and innovative. Many other designs have resulted in significant cost savings which were not considered possible until microelectronics technology produced the small, programmable computer-on-a-chip.

But in addition to the potential advantages of using microcomputers, there exists the possibility of high costs (particularly in software) through (1) proliferation of hardware and software and (2) the serious and complicated test, repair, and support problem. System implementors, program managers in military electronics, and other agencies involved in research and development are increasingly concerned with achieving standardization, so as to further expedite the acceptance of microcomputer technology and reduce excess costs of development and support.

The Computer Society Workshop on Concepts for Microcomputer Standardization held at the Naval Air Development Center in San Diego last June addressed various approaches to—and possible cost benefits of—microcomputer standardization. Included was a review of the impact of microcomputer standards on industry, as well as a review of those factors which might evolve naturally into industry standards. An important standardization factor discussed was the current state of the art, together with its direction in the near-term future and projections for future research and application. The workshop attempted to correlate the following factors to achieve logical conclusions and recommendations for action, but the complexity of the tradeoffs limited the extent of the results:

Several approaches to standardization were considered as alternatives:
- architecture, functional description, and instruction set definition;
- microcomputer interfaces and bus definition;
- microcomputer family (e.g., UYK-30, AYK-14, etc.);
- family of microprogramable microcomputer building blocks;
- specifications of classes of microcomputer application and support systems specs;
- high-order languages.

This is not an exhaustive list of alternatives. Other options can be identified as modifications of the above. For example, rather than a single high-order language, one can identify and standardize on a family of application languages. Or one can develop a family of application languages within a high-order parent language. Nevertheless, the options must be reviewed and the potential cost savings identified.

Among the areas receiving the greatest attention were (1) the need for a common lexicon (i.e., definitions, terminology, etc.); (2) the use of standard electronic modules and microcomputers (a standard building block at the printed circuit board level); (3) high-order language with a subset of application languages; and most important (4) the need for future research into the problems of testing, diagnosis, and repair of microcomputer systems. The area of microcomputer testing was identified and touched upon by all participants at the workshop.

Microcomputer design applications vary from the replacement of random logic to the use of many thousands of microcomputers in a single application. This diversity of applications presents the problem of a diversity of standards that may be applicable. In fact, several different standardization approaches may be desirable. So, rather than attempting to concentrate on a specific approach to standardization, the following discussion will present each alternative together with its supporting background material. All possible approaches to standardization have been considered and evaluated against generally accepted cost criteria. Recommendations and guidelines are based on the review and analysis of the data.

Standardization concepts

Standard lexicon. The need exists in any technical field for an established set of definitions and terminology which will promote understanding and foster communication in the research and development community. The basic aims of these standards are

1. Easy procurement through agreed upon definitions of terms such as bit-slice, direct memory access, machine cycle, random access time, etc.
2. Provide authoritative documents such as specifications which will properly describe customer needs.

3. Technical documents should use standard definitions to avoid or reduce to a minimum the possibility of misunderstanding.

4. Other benefits such as reduced investment in inventory, simplified servicing and facility in training are the result of common terminology.

Through its Standards Committee, the IEEE Computer Society pursues standards to assist in the development of new technology. This includes the pursuit of microcomputer-related definitions and technology. The committee welcomes all inputs.

This area can also be assisted by the various industrial organizations such as the Electronic Instrumentation Association or Aerospace Instrumentation Association, whose industrial members have a common objective to standardize. The need is not pressing now, but with the anticipated proliferation of hardware and software, these industrial organizations will certainly play a key role in standards for industry.

The military must also produce a set of standard terminology if the IEEE Computer Society standards are inadequate for its needs. (This problem was not addressed in the workshop.)

Standard microcomputer specifications. In many applications the basic requirements for a microcomputer are the same, and it is possible to provide a single or set of specifications which result in significant cost savings through

1. clarifying the definitions of objectives;
2. determining the classes of requirements to be fulfilled;
3. making feasible the identification and assessment of resources through standard specs;
4. providing surveys, execution of design, and evaluation of results through standard specs.

The frequently identified problem with specifications that represent broad categories of microcomputers is the difficulty in agreeing with a standard specification and then administering such a spec. In the past, specifications in the military have been invoked successfully. However, the general problem of specifying microcomputers is subject to many variables—the change of technology, the nature of application, the design objective such as MTBF of 0.99 for 10 years (space applications) or MTBF of 0.9999 for 5 hours (aircraft application). In general, the specification should be application driven, not computer technology driven. From a cost perspective, standard microcomputer hardware in effect achieves standard specification. This aspect will be discussed in later paragraphs.

Standard architecture, functional description, and instruction set. A finer level of detail in specification of microcomputers is represented by standard architecture, functions, and instruction sets. Though at first glance it appears that the developer will have freedom to use his products, it soon becomes clear that architecture and instruction sets are limitations on the designer and represent more development cost. Further, such an approach does nothing to reduce support costs (cost of ownership). The purchaser and/or user will have to bear the burden of unique hardware and software support costs. The need to specify microcomputers can be greatly assisted by a common lexicon which is agreed to by all concerned rather than standard specifications.

One last comment concerning standard specification: it is necessary to specify how to test a microcomputer. The methodology to achieve testing of microcomputers is an area requiring significant research. To specify performance without specifying how to test and evaluate results is not sufficient. Testing is a critical aspect of all military systems and is becoming more important in commercial systems.

Family of microcomputers. A basic trend suitable to a given company or perhaps the military is the use of a family of microcomputers which consists of a set of CPU's complete with control and I/O and a set of memory units that can be selected to meet a given processing requirement. Once a set of hardware is specified, the software support and high-order language will tend towards a standard. Though this approach is not widely accepted except within large organizations, the advance of microelectronics technology makes it attractive.

The family of printed circuit cards that is selected today can evolve into a family of devices. Some examples of families of microcomputers now exist. One is the UYK-30, a microcomputer based on the Intel 3000 series and the first microcomputer in the Navy's standard electronic module program (SEM). An AMD-2900 is currently designed to emulate the UYK-20 (PDP-11) and the Nova. Such a family is highly attractive since it promises to capture existing software. The principal criticism of the family of microcomputers is that it might not be optimized in size, weight, or capability for a given application. However, this is frequently true of any standard hardware.

The payoff is a direct result of common hardware in different applications. This approach controls the hardware and to some extent the software. It can also be directed towards the development of optimized microcomputers (for example, each member can be designed with built-in testing circuitry to a common maintenance philosophy). This approach should be vigorously pursued by the military to reduce support costs and hence life-cycle costs.

Family of microprogrammable building blocks: LSI devices/PC boards. The wide variety of microprogrammable LSI devices offers the opportunity to tailor hardware and software to the specific application. The family of bit-slice devices such as the AMD 2900 and the MM5701 devices is typical. It includes the arithmetic logic units, the controller devices, timing and clock generator, memory, and I/O devices such as the UART. Therefore the family can be used in many ways to meet the design requirements. However, this option is fine if performance optimization is the only design consideration. From a support aspect, this approach leads to unique (new) assemblies and modules. It also requires software support and hence additional system support and development cost.

This approach is ideal for applications where weight, size, and performance are critical and a tailored design is required. It should not be ignored as a necessary option for selected designs, the support parameters being secondary.

Emulation: standard instruction sets. The feasibility of emulating an existing computer with currently available microcomputer building blocks has been demonstrated repeatedly. Some examples include the NELC project to emulate a PDP-11 and Nova with an AMD 2900; the emulation of a UYK-20 computer on an AMD 2900, the emulation of a NOVA/ROML with a MM 5701. The attractiveness of emulation is based on its potential to save on software development and support costs.

However, there is no control of the hardware which leads to costly and complicated logistics, difficulties in
test and repair, and little control over configuration. This approach is attractive when applied in conjunction with a standard microcomputer family. Minicomputer manufacturers have accepted this concept and are pursuing it as the answer to the microcomputer revolution. Digital Equipment Corporation has introduced the LSI-11, a microcomputer which has the same instruction set as the PDP-11 and which uses much of the same software as the PDP-11.

The Micro-Nova which emulates the Nova is Data General’s hope to maintain a competitive position in the microcomputer industry. A similar approach, now pursued by National Semiconductor Corporation, is a family of microcomputers each with different speeds (different technologies) but all with a common instruction set. In summary, this approach is attractive for the military or other large corporations, but the question of software capture through emulation must still be answered in the courts of law.

High-order language with special application languages. Perhaps the most interesting and most desirable approach is the development of special-purpose application languages within a parent high-order language. A powerful high-order language such as the one proposed by the Department of Defense (DoD-1) has the constructs, syntax, etc., to provide the structure for a family of special-purpose, highly efficient languages and will further provide a framework for control of high-order languages. This approach was highly recommended by the software community in attendance at the workshop.

Standards mnemonics (assembly language). A selection of a set of standard mnemonics for assembly language description is a highly complicated process, and agreement cannot be reached easily. There is little agreement on the criteria for selection, and it isn’t clear where the cost saving will be and to what order of magnitude these savings will be. The workshop was not interested in standard assembly language mnemonics.

Standard high-order language. It is an accepted proposition that programming a microcomputer for many applications requires a high-order language. Early in the application of microcomputers it was thought that, because of the need to optimize the hardware and minimize memory, microcomputers would be programmed in source code or an assembly language. However, recent developments with PLM—a language for Intel microcomputers—have demonstrated significant efficiencies in software optimization. As new efforts in developing high-order languages are started, greater efficiencies will be achieved.

However, a general-purpose high-order language is applicable on other machines (e.g., PDP-11, Nova, etc.) and translatable to the microcomputer. Obtaining agreement on a high-order language is extremely difficult. This area is still subject to evolutionary change and would best not be standardized.

One reasonable approach to achieving a standard high-order language might be through the Department of Defense efforts on a high-order language, commonly referred to as DoD-1. This language can be developed to be appropriate for microcomputers, and this possibility should be strongly recommended to the military.

Standard special-purpose application languages. Microcomputers that are applied in special areas can be programmed with special languages optimized for the specific technical area. One such area is manufacturing/industrial process control. The language will consist of control instructions which set up feedback control. It will configure a proportional-integral-derivative controller as a single instruction. This language development is currently underway at UC Berkeley. It is typical of one of many specialized areas where a special language is desirable. These languages are going to be developed, and ultimately standards will evolve. These will be determined by user acceptance.

Conclusions and recommendations

The conclusions resulted in certain recommendations for different interest groups.

IEEE Computer Society. The Computer Society should continue to pursue standard definitions and terminology to the greatest extent possible. Additional workshops are needed in the areas of microcomputer hardware development and design, high-order language/special purpose languages, and the testing and diagnostics of microcomputers.

Industry-system design. Within any organization, a set or standard family of microcomputers should be pursued. This effort will pay off in reduced design cost, improved departmental communication, and better product development. The establishment of industry standard definitions, interfaces, and software should be encouraged.

Navy Department. The Navy should encourage and support the development of common definitions and terminology. A definitions standard can be developed and enforced. Standard microcomputer families can be developed through the Standard Electronic Module (SEM) Program. These standards should be developed to emulate existing tactical computers, e.g., UYK-20. The Navy should pursue the development of special-purpose application languages within the parent language DoD-1. Finally, the Navy can foster research in the area of microcomputer applications and test and diagnostics.