Apr 1st, 8:00 AM

The Application of Microprocessor Technology in Flight Simulation

Robert L. Schwing

The Application of Microprocessor Technology in Flight Simulation

by

Robert L. Schwing

At periods of time which historically have occurred at ten-year intervals it has been necessary for training simulator manufacturers to break completely with their past practices and evolve new simulator architectures in order to deal with the increasing capabilities required of the simulator. It has happened that in each of the two noteworthy preceding cases, such a decision to make a basic change in simulator architecture has coincided with the availability of new technology with which to implement the change.

Thus, in the early 60's, the concurrent recognition that analog computation was unsuitable for the increasing requirements for simulator fidelity and that the maintenance of an analog simulator in a proper state of adjustment and calibration was becoming almost impossible led to the decision to implement (first by special-purpose designs and then by taking advantage of the capability of general-purpose computers) digital computation as the basis for the simulator performance. Approximately ten years later, in the early 70's, this reconfiguration was, in a sense, completed by the introduction of the Advanced Simulator Technology (AST) generation of simulators by the Link Flight Simulator Division. This change complemented and extended the earlier conversion to digital computation by recognizing that the new capabilities provided by integrated circuit technology at the MSI level made it possible to organize simulator hardware along functional lines for the first time. This ensued from the fact that the compactness offered by MSI technology made it possible to physically group components, chips, converters, etc. in hardware packages which were mappable with the systems of the aircraft being simulated. From this came a simplified structure for all those parts of the simulator outside the main computing complex, which proved to be so effective from the point of view of simplifying design and fabrication of simulator hardware that it was followed in very similar form by other manufacturers.

Beginning in the very late 70's, and continuing into this decade, a similar situation became apparent in which increasingly difficult problems were arising in parallel with the emergence of new technological tools. In the previous cases attention was concentrated on the problems of reduction in cost and complexity of the hardware elements of the simulator. At the present time these are relatively well under control, but have been replaced by a potentially far more costly problem associated with increasing software complexity. To a certain extent, this increasing complexity arises as a corollary to the reduction in hardware characteristic of present-day simulators since many hardware functions have been replaced by software equivalents. In addition, there are demands which have not yet reached their limit for very substantial improvements in the modeling of simulator systems so as to increase their fidelity, and the industry is probably only at the beginning of a cycle in which more and more training capabilities will be provided as part of the simulator package.

The magnitude of the problem which confronts the simulator industry is best pictured by the plot illustrating the growth in software requirements per simulator (expressed in terms of millions of instructions per second) over the past decade (Figure 1). A few key simulators are labeled on this diagram. Other manufacturers' records would undoubtedly show a similar trend.

From the history of increasing complexity shown in Figure 1, and from independent conclusions reached after evaluating the impact of growing requirements for fidelity and increasing complexity of training capabilities, Link postulated that the simulators to be built in the 1980's would reach a complexity requiring the ability to execute at least ten million instructions per second (Mips) before the end of the decade. If the increase in complexity implied only that the purchase of additional CPU's of the same general minicomputer-type which is standard in the industry would be necessary, there would not have been great cause for concern. Hardware costs, including those of the computer, are a relatively small part of total simulator cost. However, it is well known that increases in cost and design lead-time are not linear functions of the magnitude of software imbedded in the simulator. Not only is the cost impact extremely nonlinear, but the risk involved in predicting development schedules and delivery dates has become increasingly high as experience has shown again and again the necessity of making major design changes and reallocations of computer functions during what should be the terminal phases of a development project. The question inescapably arises in the mind of the simulator manufacturer, and presumably in the minds of the customers, whether a point will be reached at which such expedients are no longer effective or entail such gross changes as to imperil the possibility of delivering a simulator that will ever perform to specifications.

The title "MicroSimulation Technology" (MST) was coined to describe the next-generation simulator architecture because of strong expectations as to the role which modern MSI, LSI, and VLSI products would play. However, it was not meant to imply that a solution to the problem which the project confronted depended entirely on whether or not microprocessors were usable within or supplementary to a simulator computing complex. Such usages had already been implemented in delivered simulators, and additional applications are being developed rapidly.

In early 1982 the Link-Miles Operation of the Link Flight Simulation Division introduced the Light Jet Trainer. It dispenses with minicomputers entirely and uses for computing purposes an interconnected network of six INTEL 8086 microprocessors. The light jet trainer represents a technological breakthrough for flight simulators. However, the ability to mechanize all of the computation for a single-engine trainer type aircraft with limited capabilities by an array of microprocessors only provides a glimpse into the provisions and precautions which will have to be taken when such an implementation is expanded by orders of magnitude. Since the essence of the problem which faces simulator manufacturers is the ability to coordinate and execute ever-increasing numbers of software instructions in systems which must be closely coordinated, it is not clear that the situation will not be aggravated if the number of processors is expanded from six up to 30 or 40.

In the consideration of future computational systems, it is interesting to note that the implications of using conventional or super-minicomputers, microcomputers or combinations are beginning to take a back seat to the real decision of whether to adapt a centralized, parallel, and/or distributed architecture. The "parallel" in this terminology refers to the simultaneous carrying out of separate processes. This may or may not coincide with the fact that a "distributed" architecture would physically separate the computing centers in which such parallel processes are taking place. Even though the "centralized" architecture could be an array of minicomputers (or super-mini's) made for simultaneous parallel processing, to the external world (the user) the machine is comparable to current devices but with significantly faster throughput.

It is interesting to note that even though the last decade has seen an order of magnitude decrease in the cost and size of computer components, only an incremental increase in component speed has been realized. With current technology, tens of thousands of gates can be put on a single chip, but no gate is much faster than its TTL counterpart of ten years ago. Since the technological trend clearly indicates a diminishing growth rate for component speed, any major improvement in computation speed must come from the exploitation of concurrent processing. It appears that massive parallelism can be achieved only if the computational algorithm is designed to exploit high degrees of pipelining and multiprocessing in both hardware and software. When a large number of processing elements work simultaneously, coordination and communication usually determine computation speed limits. It has been theorized that a parallel/distributed system architecture for complex flight simulators will require up to 40 microprocessors to achieve a 10 MIPS computing bandwidth. It is presumed that interfacing between processors can be taken care of by a relatively low bandwidth (20,000 w/s) busing system.

Any decision regarding the introduction of a new simulator configuration which includes a drastic change in the composition of the computing complex inherently creates the possibility of incurring an extremely large cost for the reconstitution of a presently adequate software development and support system. There is ample precedent to believe that the redesign of hardware could be a minor part of the total cost involved in such a reconfiguration. It has previously been determined that a reasonably accurate cost estimate for developing a new software system equivalent to the system used to support the B-52 Weapon System Trainer in its development stages is in the order of $3.5 M.

Thus, it is clear, the application of microprocessor technology throughout the flight simulation environment must be based on a simulator architecture which is compatible in both its hardware and software aspects with the projected simulator industry requirements and electronics industry developments of the future. Simulator manufacturers have resisted the urge to fall into the computer business and are therefore dependent on standard computer systems rather than components to support their business. Even though it appears that microprocessors provide the most likely vehicle for technical growth and a favorable cost curve in the next decade, total system capability/availability is the governing factor. Therefore, only total microprocessor families such as the INTEL 8086 family and the MOTOROLA M68000 family can be considered.

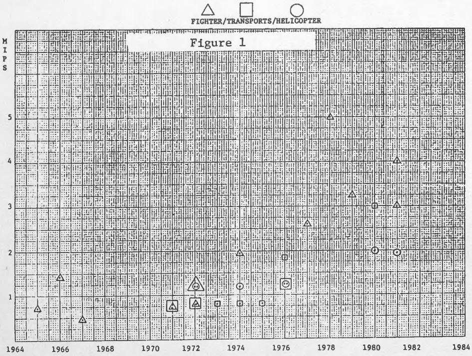
Link realizes the disaster awaiting those who put all their eggs in one microprocessor basket and is keeping an open mind on the evolution of microprocessor families. Both the INTEL 8086 family and the MOTOROLA M68000 family have been applied. However, the ever-increasing software development costs has caused Link to search out an environment for software which is hardware independent. Among the hardware independent environments being researched is Bell Laboratories' UNIX. UNIX is enjoying an almost unbelievable surge in demand even though it has been around since the late 1960's.

For those in the simulator industry, UNIX offers some advantages. The FORTRAN 77 compiler supplied with the UNIX system is standard FORTRAN ANSI 1977, with absolutely no extensions. Since this is one of the standard DOD languages, military standards are met while assuring that code generated will be portable to any system supporting FORTRAN 77. In addition, several vendors are presently working on supplying DOD certified ADA in the UNIX environment. Documentation and configuration management appear to be inherent in the system as well as UNET, an ARPANET style, hardware independent communication package. One vendor is even working at implementing an Ungermann-Bass NET/ONE interface for real-time process control through ETHERNET. However, the most significant fact about UNIX is its popularity in all computer systems from large main frames to microprocessors. UNIX is now available for DEC, GOULD/SEL, Perkin-Elmer, and Harris - the computer mainstays for simulator manufacturers. It is being ported to the INTEL 8086 family and the MOTOROLA M68000 family as well as any other microprocessor which has a sizable user following. It has even shown up as TRS-XENIX!

The flight simulation field is a complex mix of scientific and artistic application encrusted with high-technology. However, many of the most specialized areas of simulation hardware and software which have contributed to erecting a substantial "cost-of-entry" barrier to potential competition in the past are rapidly diminishing. The approximations, subjective judgments, and math model simplifications, previously standard practice because of the high cost of computation, are gradually disappearing, thanks to the microelectronics growth explosion. With the help of microprocessors, the art of simulation is being replaced by the engineering science of simulation.

The future of flight simulation will be based on the microprocessor. Parallel processing in one form or another will be a fact of life. Vendors will provide systems configured from the microprocessor families. The fledgling micro-computer manufacturer, Daniel Data Electronics A/S already markets the UNIMAX -- a system combining up to eight Motorola 68000's on a single 32-bit data bus to achieve performance in the 4.8 MIPS range. Standard, High Order Languages are being implemented on the infant microprocessor systems and it is this fact that will ultimately determine the depth to which microprocessors will become planted in flight simulation. The final test will definitely be supportability and only time will tell on this factor.

About the Author - Robert L. Schwing is the Director of Research and Development, Link Flight Simulation Division, The Singer Company. He came to Link after more than fifteen years of civil service with USAF at Wright-Patterson AFB. He was a key participant in the design and construction of the Flight Control Development Laboratory facilities including the unique engineering flight simulation equipment contained in the buildings. He was Program Manager of the B-52/KC-135 Weapon System Trainer program and Assistant for Software to the Deputy for Simulators, USAF. At Link, Mr. Schwing coordinates all the R&D for Government Simulation Systems and directly manages the R&D efforts of the Government Simulation Development Operations, Binghamton, NY. Mr. Schwing is a registered Professional Engineer and is a member of Tau Beta Pi, Pi Tau Sigma, ASME and AIAA.


FIGHTER/TRANSPORTS/HELICOPTER
Figure 1
MIPS
5
4
3
2
1
1964
1966
1968
1970
1972
1974
1976
1978
1980
1982
1984