Developments in Business Simulation & Experiential Exercises, Volume 8, 1981

SIMULATIONS AND MICROPROCESSORS

Ronald L. Jensen, Emory University

ABSTRACT

Microprocessors do not necessarily lead to micro simulations, or do they? This paper takes a quick look at some of the advantages and limitations of using current microprocessors for educational simulations.

CAPACITY

Microprocessors are subject to nearly the range of capacities as were found on large systems five or six years ago, but most are similar to two popular systems which will be used as the basis for this paper. These are the Apple][and the TRS-80 micro systems.[1] Fundamental to any computer's capacity for simulation is the memory, and it is almost impossible to have too much. Memory is so inexpensive that it does not pay to go with less than the maximum (48k on an Apple or TRS-80). After allowing for a built in BASIC language program, the net maximum available user memory is about the same for most systems. Minimal systems may rely on cassette tape to Store programs, This is too slow for programs of any size, so the replacement is a disk system. A 60k byte mini-floppy disk could store 5 programs of 12k Size, or 2 of 30k size (quite large) or many very small programs. It also comes in models up to 370k bytes (characters) which store proportionately more data and programs. Larger (8") floppy disks gain up to 4 time the storage of the mini-floppy disks and 'hard' disks, similar to those used on very large systems, are available up to 80,000k capacity, at prices from 4 to 25 times that of the floppy disks. Hard copy options are available for well under a thousand dollars. A full system with maximum memory, two mini-floppy disks, and hard copy output could cost the user about $3,200 for one of the popular systems.

PROGRAM EXECUTION

The micro processor is smaller than main-frame computers and most mini-computers, but it is also slower. Real execution speeds can vary by an order of magnitude by both selection of the hardware and the choice of programming language. The language implementation seems to be more important than the processor design details. Many microprocessor programs are intended for interactions with the user, and these may be very effective. Speed may not be too important if the program is designed to minimize apparent user delays, but this is equally true for users on time shared systems. Many micro systems have smaller screen displays than most time shared systems, so this may restrict some uses, or at least require special design considerations.

PROGRAM DEVELOPMENT

Simulations may be developed on a microprocessor as opposed to moving existing programs from a larger system. Some operating systems provide very good development aids, and some are very poor. You need to have good text and program editing features, but some "popular" systems provide almost none, and then you must find one from a third party, build your own, or put forth much more effort and time in the development process, The language chosen may also influence how easily a simulation may be developed. BASIC is the most common language, but the implementations vary widely, CONDUIT publishes a comparison of many of the BASIC dialects [33, which vary from very rich to very sparse versions. A routine which is easy to implement on one system may be very difficult on another. Three of the most popular versions [2, 5, 6] are very similar, but are not fully compatible. The Microsoft BASIC [5] is available as both an interpreter and compiler. The compiled version of a program may occupy 30% less memory and execute 2-3 times faster than the interpreter version of the same program. Under the CP/M operating system [4], which is available on a wide variety of microprocessors, the user may have a choice of BASIC, FORTRAN, COBOL, PASCAL, PL/1 and other programming languages. These are respectable implementations, but you can not expect them to have all the features or performance of the same languages on a large system.

TRANSPORTING PROGRAMS

While use of a system such as CPIM insures certain operational standards, there Is no standard for the physical media for microprocessors. This makes it difficult and expensive to distribute physical copies of a simulation except for use on similar systems. A "data link' connection may be made between two dissimilar microprocessors to transfer programs, but this requires special programs and hardware. You may provide source listings and let the recipients enter the program on their own system. This is tedious for a program of any length, and requires a compatible language implementation on each system. Many microprocessors have only BASIC available. As shown in the CONDUIT manual they are not really compatible beyond a very limited level. This means that the simulation must be re-developed for each different microprocessor. Until true standard media and languages are regularly available, this will limit transporting simulations.

REFERENCES

[1] Apple Computer Inc. • The DOS Manual, Cupertino, CA, 1980.

[2] Apple Computer Inc., APPLESOFT BASIC PROGRAMMING REFERENCE MANUAL, Cupertino, CA, 1978.

[3] CONDUIT, BASIC GUIDE, The University of Iowa, Iowa City, IA, 1979.

[4] Digital Research Corp., CP/M Reference Manual, Pacific Grove, CA, 1980.

[5] Microsoft Consumer Products, Microsoft BASIC Reference Manual, Bellevue, WA, 1980.

[6] Tandy Corporation, TRSDOS & DISK BASIC Reference Manual, Fort Worth, TX, 1979.

[1] Apple][is a trademark of Apple Computer Inc. TRS-80 is a trademark of Tandy Corporation. CP/M is a trademark of Digital Research Corp.