Reprinted with permission from American Laboratory, Vol. 7, No. 7, pp. 69-70.

What's next? Microprocessors

By David G. Larsen, Peter R. Rony, and Jonathan A. Titus

WITH THIS twelfth column in the "Computer Interfacing" series, we would like briefly to list the topics covered in previous columns and to describe the direction that this series will take in subsequent issues.

Beginning with the April 1974 issue of *American Laboratory,* we have discussed:

The advantages of asynchronous serial data transmission

The precision of digital data transmission

Digital codes-binary, *BCD,* and ASCII

Real-time data acquisition, control, and logging

Interfacing with asynchronous serial

How one gets started in digital electronics

The universal asynchronous receiver/transmitter (UART)

First-in first-out (FIFO) memories

Serial data exchange modules

Half and full duplex current loops

An introduction to microprocessors

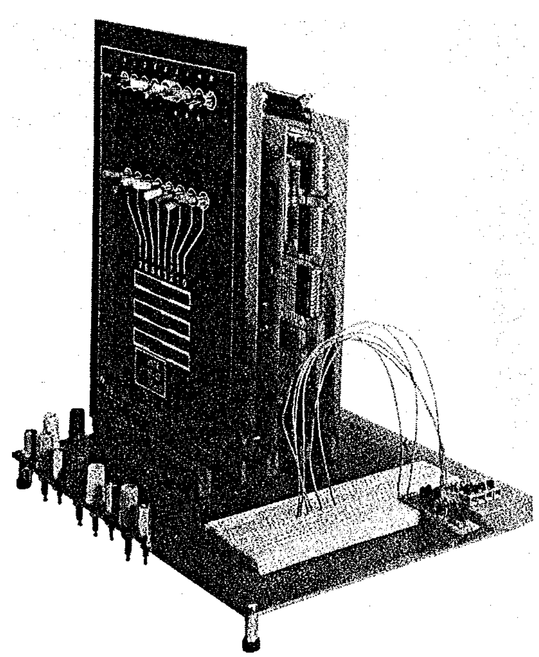
With the exception of the EIA RS-232C interface standard, which we will cover in a future column, the above topics summarize the most important aspects of interfacing via the use of asynchronous serial techniques.

By now, you are aware of the fact that a revolution is occurring in the electronics industry: *microprocessors.* If you had held stock in companies that manufacture microprocessors, this fact would have become quite apparent after RCA's misinterpreted announcement several months ago that microprocessors will soon be incorporated into U.S. automobiles. Rather than rehash an electronics revolution after it is over, we believe that it would be fun to jump into the middle of the one that is occurring at this moment and closely observe events that will have a profound influence on all laboratory instrumentation. Consequently, we would like to devote the next twelve or more columns to the subject of microcomputers: what they are, how they operate, and what they can and cannot do for the laboratory scientist or engineer.

We shall use microprocessor operation and interfacing as a vehicle to probe more deeply into the detailed concepts and techniques of computer interfacing. Please keep in mind that the microprocessor, when complemented by memory, buffers, and I/O devices, is as much a computer as its larger and usually faster rivals, the minicomputers and full size computers. By learning how to interface a microprocessor, you will simultaneously learn the concepts of how to interface a minicomputer or full size computer. The use of interrupts, device selects, software generated strobes, timing loops, and the like are common to all.

To gain full value from some of our forthcoming columns, it would be beneficial to have an understanding of the basic principles of digital electronics. Some very important terms and concepts that you should master include the following: gate, logic element, counter, gated counter, monostable, enable, disable, inhibit, strobe, decoder, multiplexer,

Figure 1 *The Mark 80 microcomputer system.*



Mr. Larsen, Department of Chemistry, and Dr. Rony, Department of Chemical Engineering, are with the Virginia Polytechnic Institute and State University. Dr. Titus is with Tychon, Inc.

demultiplexer, timer, clock pulse, positive edge, negative edge, flip-flop, latch, bus, TRI-STATETM shift register, dynamic RAM, static RAM, ROM, programmable ROM, up/down counter, AND, OR, NAND, NOR, exclusive OR, arithmetic element, and more. We have been hesitant to mention in previous columns our pair of books on digital electronics, *Bugbooks I and II. Logic and Memory Experiments Using TTL Integrated Circuits* (published by E&L Instruments, Inc., Derby, Conn.) but feel that it is not inappropriate to do so at this time. They will bring you to the level of understanding in digital electronics required to interface microcomputers; other digital books, such as the pair marketed by Hewlett-Packard in conjunction with their logic lab, will also help you develop the skills you will need. Digital electronics is a vigorous field, and new texts and reference manuals are appearing at the rate of about one every several weeks.

As we currently envision them, the columns will offer instruction in operating and interfacing a very popular microprocessor, the Intel 8080 8-bit microprocessor, which can perform a simple logic or arithmetic instruction in only 2 *usee* and can directly address 65,536 different memory locations, each containing eight bits of data. Originally priced at $360 each, it is now available for $170 from selected supply sources and will cost no more than $50 in two or three years. The 8080 has some important rivals, e.g., the Motorola 6800 and the Fairchild F8, but it is a worthy selection nevertheless. Each microprocessor has its special features. However, the general concepts developed in this column will be applicable to any microcomputer system.

Standing alone, a microprocessor chip can do nothing. It functions only in the context of a microcomputer system, in which appropriate integrated circuit chips are incorporated to complement the basic function of the microprocessor (pP): to serve as a central processing unit (CPU) in which logic and arithmetic operations and data transfers between registers, memory, and the outside world are performed. In some columns, we will need to focus upon a specific microcomputer system. For this purpose, we have chosen a new system that is specially designed to instruct individuals in all of the details of microprocessor operation and interfacing: the Mark 80 microcomputer *(Figure* 1). This particular system, shown with 4K of solid-state memory and a control panel, is built around the Intel 8080 microprocessor chip. Except for a power supply, it is completely operational. The system is bus structured and has all important inputs and outputs connected to a solderless breadboarding socket, permitting interfacing concepts to be learned, tested, and breadboarded into a digital circuit of one's own design.

We had one basic mission with our previous eleven columns: to encourage laboratory scientists and engineers as well as instrument manufacturers to give serious consideration to the advantages of a simple interfacing technique, INWAS, for data transmission at moderate data rates between laboratory instruments and computers. Our mission with our forthcoming microprocessor columns is to promote an understanding of what microprocessors can do and to encourage the incorporation of microprocessors within instruments in such a way that an individual who is interested and knowledgeable can easily modify stored programs, add interfaces, and in general have complete control over the basic operation of each major instrument in his laboratory.