

and output. You can use XYBASIC to look at an individual bit on an input port with SENSE, or to test a particular bit of a variable with TEST. You can wait for a particular event to occur with WAIT. And XYBASIC even lets you link programs with assembly language routines, using CALL and SCALL; if you already have assembly programs, you can use them from within a XYBASIC program and pass information to them in a natural way.

XYBASIC also lets you perform bit manipulation functions like ROTATE and SHIFT which were previously possible only in assembly language. And you can concatenate variables, split variables into 8-bit values, convert between binary and BCD representations, and perform logical operations such as AND, OR, XOR and NOT.

Software interrupt = much more power

XYBASIC offers a software interrupt feature which allows concurrent processing, effectively multiplying the power of your computer to a substantial degree. Suppose for example that you are conducting a chemical experiment: start with 250 cc of solution A, apply heat, and heat to 75 degrees C. With XYBASIC you can ENABLE an interrupt which will continuously monitor a digital thermometer hooked up to the experimental apparatus. You can let XYBASIC continue processing data from the experiment, and automatically shut down the heat when the solution reaches the desired temperature!

You can use the DELAY command to incorporate real time delays into a program -- it's like having a real time clock in your computer.

Software test instrument

If you are not certain whether your computer is operating correctly, or if you add new hardware to your system, you would normally write an assembly language test program. But a much better way to test is to use XYBASIC in conjunction with your usual test instruments, like an oscilloscope or voltmeter. Write a simple XYBASIC program, and use the interactive features of XYBASIC to help pinpoint problems. Comparing assembly language programming with XYBASIC is like comparing a manual typewriter with a word processor -- both can produce a beautiful manuscript, but there's a world of difference in the amount of effort needed.

Powerful direct mode

Here's another difference between assembly language and XYBASIC. Without the involved process of editing, assembling, loading and running an assembly language program, you can use the interactive capabilities of XYBASIC's direct mode to print the information you want instantly. For example, you can find the value on an input port or put a value on an output port directly.

Unique one-step debugging

Since XYBASIC includes an editor, you do not need to load it separately; therefore you can change your program instantly. The additional steps