

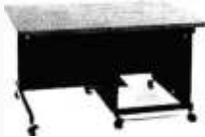
CHOOSE...

Choose an Apple Desk



A compact Bi-Level desk ideal for the Apple computer system. This 42" x 29½" desk comes with a shelf to hold two Apple disk drives. The top shelf for your TV or monitor and manuals can also have an optional paper slot to accommodate a printer. It is shown here with the optional Corvis shelf which will hold one Corvis disk drive. The Corvis shelf is available on the 52" x 29½" version of the Apple desk.

Choose a Micro Desk



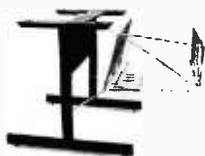
The Universal Micro desk accommodates the S-100 type microcomputers. The desk is available in four sizes: 17.75 inch, 19.06 inch, and 20.75 inch wide openings with 24 inch front-to-rear mounting space. The fourth size is a 20.75 inch wide opening with a 26.50 inch front-to-rear mounting space.

Choose a Mini Rack



Mini racks and mini micro racks have standard venting, cable cut outs and adjustable RETMA rails. Choose a stand alone bay or a 48", 60", or 72" desk model in a variety of colors and wood tones. A custom rack is available for the Cromemco.

Choose a Printer Stand



The Universal printer stand fits the:

Centronics 700's	Diablo 1600's & 2300's
Dec LA 34	T.I. 810 & 820
NEC Spinwriter	Okidata Slimline
Lear Siegler 300's	Anadex 9500's

Delivery in days on over 200 styles and colors in stock. Dealer inquiries invited.

ELECTRONIC SYSTEMS FURNITURE COMPANY

17129 S. Kingsview Avenue
Carson, California 90746
Telephone: (213)538-9601

Once the program is working correctly, the programmer can reedit the 8086 source to trim out the unnecessary register saves and flag manipulations to speed up the code and make it smaller. If the source code is in Zilog mnemonics, or if a filter program is available to convert to Zilog mnemonics, we recommend the pair of programs from Seattle Computer Products. The Seattle translator and its ASM86 assembler should be used together for optimum benefit.

XLT86 from Digital Research is a program that takes 8080 source code and converts it into 8086 source code in an intelligent manner using data-flow-analysis techniques. It will produce better code than either of the other two translators if two conditions are met: (1) no Z80 instructions are used, and (2) the source code is not bigger than 6K bytes (assuming the CP/M system is a 64K-byte system). XLT86 should be used with ASM86 under either CP/M-80 or CP/M-86.

MS-DOS versus CP/M-86

In the trade press and various advertisements, we see claims that conversion of CP/M-80 programs to MS-DOS (IBM PC-DOS, Lifeboat Associates' SB-86, Seattle Computer Products' 86-DOS) is as easy, or even easier, than from CP/M-80 to CP/M-86. In fact, some differences are seen in interfacing to the two 16-bit operating systems. With the assumption that the program had to operate in a 64K-byte region under CP/M-80, we will bypass memory-allocation questions. The remaining issues that have to be addressed are:

- How does a program gain access to operating system resources?
- How are file control blocks used to manipulate files?

Gaining access to the operating system under CP/M-80 requires placing the function code in the C register, placing the information address in the DE registers, and calling location 05 hexadecimal, the CP/M-80 entry point. If the system

call returns a value, it returns the value to the A register. A so-called warm boot under CP/M-80 is accessed by jumping to location 00 hexadecimal, which reads in the operating system and resets the disk system.

Now let's look at similar functions under CP/M-86 and MS-DOS.

Gaining Access to CP/M-86

Gaining access to CP/M-86 requires placing the function code in the CL register, placing the byte parameter in the DL register or placing the word parameter in the DX register, placing the data segment in the DS register (the data segment is usually not changed for a converted program), and executing a software interrupt, INT #224. The result is returned in the AL register if it is a byte value; if the result is a word value, it is returned in both the AX and BX registers. Double-word values are returned with the offset in the BX registers and the segment in the ES register. Conversion of programs from CP/M-80 to CP/M-86, then, requires replacing the call to location 5 with the software interrupt INT #224.

Another necessary change involves the warm boot. Under CP/M-80, the warm boot may be accessed by a system call with a function code of 0 for a jump to location 0. CP/M-86, however, does not support the jump to location 0. As a result, you must change this program exit in the translated program if the program is to run correctly.

Provided that the call to location 5 is replaced with INT #224, that the warm boot change is made, and that the registers are mapped correctly, there should be little problem in getting the translated program to access the CP/M-86 system functions.

Gaining Access to MS-DOS

Although MS-DOS has a "preferred" mechanism through a software interrupt, INT #33, for accessing the system, an additional mechanism is provided for "pre-existing" programs that is compatible with CP/M-80 calling conventions, at least for functions in the range of