Figure 1: Architecture of the 8080 microprocessor showing the internal registers.

show the output that each of the three
translators produced from the same
original code. In addition, we'll make
some observations about the dif-
fences in adapting the translated
code to CP/M-86 and MS-DOS. Next
month, we'll take a closer (although
still not comprehensive) look at
CP/M-86 and MS-DOS.

Orientation to the 8086

The first thing we have to do is ex-
amine the differences between the
familiar 8080 and Z80 microproces-
sors and the 8086. For reference,
figure 1 shows the registers and ar-
chitecture of the 8080; figure 2 shows
the registers and architecture of the
Z80. We'll make few comments about
these registers because they are
familiar to you if you have 8080 or
Z80 source code that you want to
translate.

Figure 3 shows the registers and ar-
chitecture of the 8086. Since the 8086
is less familiar, we'll take a brief look
at it for orientation. (For further
enlightenment, see The 8086 Book,
Russell Rector and George Alexy,
Osborne/McGraw-Hill, 1980, and
The 8086 Primer, Stephen P. Morse,
Hayden, 1980.) The 8086 is, of
course, a 16-bit microprocessor. The
8086 can access up to 1
megabyte of memory and as many as
65,000 input/output ports. The
megabyte of memory is $2^{20}$ 8-bit
bytes; any two consecutive bytes are
a 16-bit word. Some 8086 instructions
access bytes; others access words.

nally. Externally, however, they ap-
pear different due to the 8-bit bus of
the 8088 and the 16-bit bus of the
8086. This means that programs that
run on the 8088 will also run on the
8086 assuming that the memory
resources and peripheral resources
are the same. In general, statements
in this article that apply to the 8086
apply to the 8088 as well.