offset within a word, the controller computes the remainder of B divided by N. That is, the word address is given by:

$$W = \left\lfloor \frac{B}{N} \right\rfloor$$

and the offset is given by:

$$O = B \bmod N$$

As an example, consider the values in Figure 10.7, where $N = 4$. A byte address of 11 translates to a word address of 2 and an offset of 3, which means that byte 11 is found in word 2 at byte offset 3†.

10.19 Using Powers Of Two

Performing a division or computing a remainder is time consuming and requires extra hardware (e.g., an Arithmetic Logic Unit). To avoid computation, architects organize memory using powers of two. Doing so means that hardware can perform the two computations above simply by extracting bits. In Figure 10.7, for example, $N = 2^2$, which means that the offset can be computed by extracting the two low-order bits, and the word address can be computed by extracting everything except the two low-order bits. Figure 10.8 illustrates the idea:

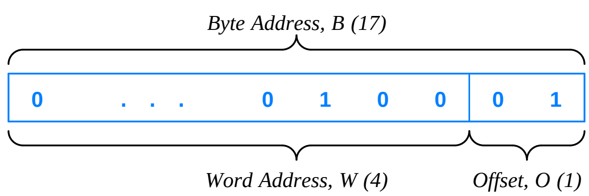

Figure 10.8 An example of a mapping from byte address 17 to word address 4 and offset 1. Using a power of two for the number of bytes per word avoids arithmetic calculations.

We can summarize:

> *To avoid arithmetic calculations, such as division or remainder, physical memory is organized such that the number of bytes per word is a power of two, which means the translation from a byte address to a word address and offset can be performed by extracting bits.*

†The offset is measured from zero.