

# General Cache Equations

Most sane computer architects would design their caches so that the number of sets in the cache and the number of bytes per cache line are powers of 2. If this is the case, then we can partition the bits of the address to identify the tag, set number, and offset for that address in the cache. However, if those cache parameters are not powers of 2, then we cannot simply use address bits and must do some calculations for finding the tag, set number, and offset for a given address.

## Offset

The offset is an index into a cache line to find the data of a given address. For example, if we have cache lines that contain four bytes each, then we can bring in blocks of memory of size four and the possible offsets for addresses are 0, 1, 2, and 3. Mapping addresses, we would find that in the first possible block, address 0 has an offset of 0, address 1 has an offset of 1, address 2 has an offset of 2, and address 3 has an offset of 3. Then address 4 would be in a new block and have an offset of 0. We can see that the offset is based on taking the remainder of the address divided by the number of bytes in a cache line.

Therefore, the offset is calculated by the following equation:

$$\text{offset} = \text{address} \% \text{number of bytes per cache line}$$

## Tag

Given that we're placing multiple bytes of memory into one cache line, we need to separate them into blocks so that we bring in the same pieces of memory together into the cache. As before, if the cache lines can hold four bytes, then addresses 0 through 3 are in one block, addresses 4 through 7 are in another block, and so on. If we number the blocks starting at 0, we can see that an address maps to a block based on integer division by the number of bytes in each line. We can call the block number a "tag" we can use to determine if a given address is in a cache line.

Therefore, the tag is calculated by the following equation:

$$\text{tag} = \left\lfloor \frac{\text{address}}{\text{number of bytes per cache line}} \right\rfloor$$

## Set

A cache can have any number of sets to hold cache lines. Each block of memory must map to the same set so that it can be easily found if it is in the cache. Mapping blocks to sets is similar to

mapping addresses to offsets.

Therefore, the set number is calculated by the following equation:

$$\text{set number} = \text{tag} \% \text{ number of sets in the cache}$$

Or expanding:

$$\text{set number} = \left\lfloor \frac{\text{address}}{\text{number of bytes per cache line}} \right\rfloor \% \text{ number of sets in the cache}$$

## When the Cache Parameters are Powers of 2

As mentioned before, powers of 2 make finding the tag, set, and offset for a given address much easier. If the number of bytes per cache line is a power of 2, then we can see that the offset calculated will be exactly the same as the  $\log_2(\text{bytes per line})$  least significant bits of the address.

If both the number of bytes per cache line and the number of sets in the cache are powers of 2, then the set number can be indicated by the  $\log_2(\text{number of sets})$  least significant bits to the left of the bits that represent the offset. The tag can be reduced to be the remaining most significant bits of the address since those bits will now uniquely identify blocks of memory that map to a given set.