**Cache-Mapping Function :**

• The transformation of data from main memory to cache memory is referred to as memory **mapping** process.

• This is one of the functions performed by the memory management unit (**MMU**).

• Because there are fewer cache lines than main memory blocks, an algorithm is needed for mapping main memory blocks into cache lines.

• There are three different types of mapping functions in common use and are **direct**, **associative** and **set associative**

The three techniques are discussed below:

**1- Direct Mapping** :This is the simplest among the three techniques. Its simplicity stems from the fact that it places an incoming main memory block into a specific fixed cache block location. The placement is done based on a fixed relation between the incoming block number, i, the cache block number, j, and the number of cache blocks, N: j = i mod N

The main **advantage** of the direct-mapping technique is its simplicity in determining where to place an incoming main memory block in the cache.

Its main **disadvantage** is the inefficient use of the cache. This is because a number of main memory blocks may compete for a given cache block even if there exist other empty cache blocks. This disadvantage should lead to achieving a low cache hit ratio.

According to the direct-mapping technique the MMU interprets the address issued by the processor by dividing the address into three fields:

1*.* ***Word*** field = log2 B, where B is the size of the block in words.

2. ***Block***field = log2 N, where N is the size of the cache in blocks.

3. ***Tag***field = log2 (M/N), where M is the size of the main memory in blocks.

4. The number of **bits in the main memory address** = log2 (B x M)

**Example 1:** Consider, for example, the case of a main memory consisting of **4K** blocks, a cache memory consisting of **128** blocks, and a block size of **16** words.

