School of
Computing Science

CSC1104 – Computer Organisation and Architecture

Tutorial 2:
Computer Functions and Cache Memory

Assoc. Prof. Cao Qi
Qi.Cao@Glasgow.ac.uk

Objectives

- Practice computer functions on instruction cycles and interrupts.
- Understand memory systems and cache memory.

Questions:

1. Consider a hypothetical 32-bit processor having 32-bit instructions composed of two fields: the first byte (8-bit) contains the opcode; and the remainder 24-bit as the address to load the operands (or load the immediate operand).
    a) What is the maximum directly addressable memory capacity (4-byte data length per unique address)?
    b) How many bits are needed for the program counter (PC) register and instruction register (IR)?

2. A computer has a cache, main memory, and a disk used for virtual memory. If a referenced word by the CPU is in the cache, 20 ns (nanoseconds) are required to access it.

   If it is in main memory but not in the cache, 60 ns are needed to load it from the memory into the cache, and then the word is referenced by the CPU from the cache.

   If the word is not in main memory, 12 ms (milliseconds) are required to fetch the word from disk to the memory, followed by 60 ns to copy it from the memory to the cache, and then the word is referenced by the CPU from the cache.

   The cache hit ratio is 0.9. The main memory hit ratio is 0.6. What is the average time in nanoseconds (ns) required to access a referenced word by the CPU on this computer?

3. There is a computer system with the following specifications:
    - Installed main memory: 4 GiB (GibiBytes), with each byte being accessible by a unique memory address.
    - Size of a cache line: 64 B (bytes).
    - Total cache capacity: 2 MiB (MebiBytes).
    - Direct mapped cache method.

    Answer the following questions about this computer system.

    I. How many address bits are required to access each byte location in the main memory?
    II. How many cache lines are there?
    III. How many blocks in the main memory?
    IV. What is the address breakdown of the cache: _____ bits tag + _____ bits line + _____ bits word?