**Fall 2013: COMP 7300 Advanced Computer Architecture**

**Test 1 (100 pts)**

Grading policy:

¼ Credit for correct answer

¾ Credit for well written and solid justification/facts/arguments.

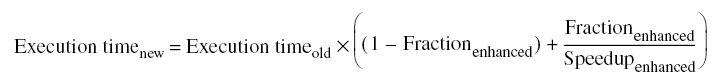
**A) Introduction**

1. "Old" view computer architecture was focused on the instruction set architecture (ISA), le number of registers, le number of addressing modes.... Modern computer architecture focus on meeting specific requirements of the target machine and maximizing performance within key constraints: \_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_, and \_\_\_\_\_\_\_\_\_\_\_\_\_ [**Fill in the blank**]
2. A task **T** requires ***n*** transitions 0->1 or 1->0. This task was executed in time ***t*** consuming energy ***E*** and power ***P***. If this task is clocked to run faster and execute in time ***t***/2, then an energy E’ and power P’ will be consumed. We want to find the new energy E’ and power P’.
   1. What is the value of **P** as a function of ***E***?
   2. What is the new energy ***E’*** consumed as a function of E?
   3. What is the new power ***P’*** consumed as a function of ***E***?
   4. What is the new power ***P’*** consumed as a function of ***P***?

1. A task **T** requires ***n*** transitions 0->1 or 1->0. Of the following parameters- namely clock rate, voltage, and **n**, which ones have an impact on the **energy** consumed? For each parameter, indicate the direction of change (i.e., does an increase of a parameter lead to an increase or a decrease of the energy?).
2. A task **T** requires ***n*** transitions 0->1 or 1->0. Of the following parameters- namely clock rate, voltage, and **n**, which ones have an impact on the **power** consumed? For each parameter, indicate the direction of change (i.e., does an increase of a parameter lead to an increase or a decrease of the power?).

1. Briefly describe techniques to reduce power consumed in a computer system
2. Consider a task that takes time ***t*** to execute. 25% of this task can be optimized to be sped up by a factor 10. The reminaing 75% cannot be sped up. What would be the execution time of this optimized task?

Here is Amdahl's Law:



1. **Exercise**:

We consider a program ***P*** executed on an imaginary processor with a clock rate of 2 GHz. This table provides the number of executions for each type of instruction and the number of cycles. Answer the questions based on this table.

|  |  |  |
| --- | --- | --- |
|  | Number of instructions | (CPI) Clock cycle Per Instruction |
| Load/Store | 500 | 3 |
| Arithmetic | 200 | 2 |
| Logical | 100 | 1 |

1. How many clock cycles in total are needed to execute Program ***P***?
2. What is the average CPI (average number of clock cycles per instruction) for Problem ***P***?
3. What is the latency (in ns) of **a** *Load/Store* instruction?
4. What is the latency (in ns) of **a** *Logical* instruction?
5. How long does Program ***P*** take to execute (in microseconds)?

**B) Memory**

1. Four facts/requirements make an hierarchical memory system necessary and useful. **Cite** these facts/requirements?
2. What kind of memory (DRAM, ROM, EPROM, SRAM, Flash…?) is used for caches? What is the range of access time for such a kind of memory? What is the range of cost per GB?
3. What kind of memory (DRAM, ROM, EPROM, SRAM, Flash…?) is used for the main memory? What is the range of access time for such a memory? What is the range of cost per GB?
4. When a miss occurs, a block must be located and loaded. The time to find the first word in the main memory is called *seek time*. After the first work is located, the block must be loaded from the main memory in the cache taking the *loading time*.
   1. What impact does a **larger** block have on the *seek time*?
   2. What impact does a **larger** block have on the *loading time*?
   3. What impact does a **larger** block have on the *miss penalty (for a read)?*
   4. Explain what *Early Restart* is and how it alleviates the use of a larger block?
   5. Explain what *Critical-Word-First* and how it alleviates the use of a larger block?
5. Explain **when, why,**  and **how** *Write-Back* strategy is used.

**When:** on a write hit

**Why:** avoid to access the main memory (takes time)

**How**: update only cache. Update main memory on replacement. Use a dirty bit to write back only if block was modified.

1. Discuss the impact of **larger** blocks on the ***miss rate*** AND the ***miss penalty***.

**Exercises**

1. **Exercise 1**: Assuming 32-bit addresses (one byte per address), how many total bits are required for a direct-mapped cache with 32 KB of data and 16-word blocks?
2. **Exercise 2:** A processor takes one clock cycle per instruction (CPI = 2) when a reference is a hit on the primary cache. All references are made first to the primary cache at a 2 GHz clock rate. The main memory access time is 100 ns (including the miss handling). The miss rate per instruction at the primary cache is 3%. Suppose we add a secondary cache with a 5 ns access time and is large enough to reduce the miss rate to the main memory to 0.6%.
   1. With a one level cache, what is the miss penalty (in clock cycles) to the main memory?
   2. With one level cache, what is the average number of clock cycle per instruction?
   3. What is the miss penalty for an access to the second-level cache?
   4. With two levels, what is the average number of clock cycle per instruction?
   5. How faster is the processor with multi-level cache?