|  |  |
| --- | --- |
| **Unit 1: Chapter 1:Numericals on Performance** | |
|  | Suppose that we are considering an enhancement that runs 10 times faster than the original machine, but is usable only 40% of the time. What is the overall speedup gained by incorporating the enhancement?   1. What is the overall speedup if you make 10% of a program 90 times faster? 2. What is the overall speedup if you make 90% of a program 10 times faster? |
|  | The owner of a shop observes that on average 18 customers per hour arrive and there are typically 8 customers in the shop. What is the average length of time each customer spends in the shop? |
|  | A bench mark program is run on a 40 MHz processor. The executed program consists of 100,000 instruction executions, with the following instruction mix and clock cycle count:   |  |  |  | | --- | --- | --- | | **Instruction type** | **Instruction count** | **Cycles per instruction** | | Integer arithmetic | 45,000 | 1 | | Data transfer | 32,000 | 2 | | Floating point | 15,000 | 2 | | Control transfer | 8000 | 2 | |  |  |  |   Determine the effective CPI, MIPS rate and execution time for this program. |
|  | Consider two different machines, with two different instruction sets, both of which have a clock rate of 200 MHz. The following measurements are recorded on the two machines running a given set of benchmark programs:     1. Determine the effective CPI, MIPS rate and execution time for each machine. 2. Comment on the results |
|  | Early examples of CISC and RISC design are the VAX 11/780 and the IBM RS/6000, respectively. Using a typical benchmark program, the following machine characteristics result:    The final column shows that the VAX required 12 times longer than the IBM measured in CPU time.   1. What is the relative size of the instruction count of the machine code for thisbenchmark program running on the two machines? 2. What are the CPI values for the two machines? |
|  | Four benchmark programs are executed on three computers with the following results:    The table shows the execution time in seconds, with 100,000,000 instructions executed in each of the four programs. Calculate the MIPS values for each computer for each program. Then calculate the arithmetic and harmonic means assuming equal weights for the four programs, and rank the computers based on arithmetic mean and harmonic mean. |
|  | Consider the execution of a program that results in the execution of 2 million instructions on a 400-MHz processor. The program consists of four major types of instructions. The instruction mix and the *CPI* for each instruction type are given below, based on the result of a program trace experiment:  The instruction mix and the CPI for each instruction type are given below based on the result of a program trace experiment:    Calculate the average CPI and the corresponding MIPS rate when the above program is executed on a uniprocessor with the above trace results. |
|  | In the above example 7, assume that the program can be executed in eight parallel tasks or threads with roughly equal number of instructions executed in each task. Execution is on an 8-core system with each core(processor) having the same performance as the single processor originally used. Coordination and synchronization between the parts adds an extra 25,000 instruction executions to each task.  Assume the same instruction mix as in the example for each task, but increase the CPI for memory reference with cache miss to 12 cycles due to contention for memory.   1. Determine the average CPI. 2. Determine the corresponding MIPS rate. 3. Calculate the speedup factor. 4. Compare the actual speedup factor with the theoretical speedup factor determined by Amdhal’s law. |
|  | Consider two processors P1 and P2 executing the same instruction set. Assume that under identical conditions, for the same input, a program running on P2 takes 25% less time but incurs 20% more CPI as compared to the program running on P1. If the clock frequency of P1 is 1GHz, then calculate the clock frequency of P2. |
| **Unit 1: Chapter 2:Numericals on Memory** | |
|  | Design a 16-bit memory of total capacity 8192 bits using SRAM chips of size 64 \* 1 bit. Give the array configuration of the chips on the memory board showing all required input and output signals for assigning this memory to the lowest address space. The design should allow for both byte and 16-bit word accesses |
|  | A processor accesses main memory with an average access time of T2. A smaller cache memory is interposed between the processor and main memory. The cache has a significantly faster access time of T1 << T2. The cache holds, at any time, copies of some main memory words and is designed so that the words more likely to be accessed in the near future are in the cache. Assume that the probability that the next word accessed by the processor is in the cache is H, known as the hit ratio.   1. For any single memory access, what is the theoretical speedup of accessing the word in the cache rather than in main memory? 2. Let T be the average access time. Express T as a function of T1, T2, and H. What is the overall speedup as a function of H? 3. In practice, a system may be designed so that the processor must first access the cache to determine if the word is in the cache and, if it is not, then access main memory, so that on a miss (opposite of a hit), memory access time is T1 T2. Express T as a function of T1, T2, and H. Now calculate the speedup and compare to the result produced in part (b). |
|  | Design and show the organization of memory chip of size 16M x 16 using 2048K x 8 memory chips. Find the total number of external connections |
|  | Design a 8M x 32 size memory chip using memory chips of   1. 512K x 16 2. 256K x 8   Draw the diagrams showing proper connections for each of the above cases |
|  | Design a 32M x 32 size memory chip using memory chips of 2048K x 16 |
|  | Design a 16M x 16 size memory chip using memory chips of i) 1024K x 8 ii) 16K x 8 |
|  | Design a 8M x 4 size memory chip using memory chips of 1024K x 4 |
|  | Design a 4M x 64 size memory chip using memory chips of 256K x 16 |
|  | Design a 64M x 64 size memory chip using memory chips of 1024K x 32 |
|  | A Computer system has main memory consisting of 1M words. It also has a 4K word cache organised in the block set associative manner with 4-blocks per set and 64 words per block. Calculate the number of bits in each of the tag, set and word fields of the main memory address format. |
|  | A block set associative cache consists of a total of 1K blocks divided into 8 blocks per set. The main memory contains 1M memory locations each consisting of 64 words.   1. How many bits are there in main memory address 2. How many bits are there in each of the tag, set & word fields? |
|  | A main memory contains 16K blocks of 512 words each. A cache memory consists of 128 lines. Calculate the address field bits & draw the lines showing the mappings for each of the following   1. Direct mapping 2. 4-way set associative mapping |
|  | A set- associative cache consists of 512 lines, divided into 8-line sets. Main memory contains 8K blocks of 64 words each. Show the format of main memory addresses and diagrammatically show the mapping of the main memory blocks to cache lines |
| **Unit 1: Chapter 3:Numericals on CPU** | |
| 1. | Assume an instruction set that uses a fixed 16-bit instruction length. Operand specifiers are 6-bits in length. There are K two-operand instructions and L zero-operand instructions. What is the maximum number of one-operand instructions that can be supported? |
| 2. | Let the address stored in the program counter be designated by the symbol X1. The instruction stored in X1 has an address part (operand reference) X2. The operand needed to execute the instruction is stored in the memory word with address X3. An index register contains the value X4. What is the relationship between these various quantities if the addressing mode of the instruction is (a) direct; (b) indirect; (c) PC relative; (d) indexed? |
| 3. | An address field in an instruction contains decimal value 14. Where is the corresponding operand located for:  a) immediate addressing?  b) direct addressing?  c) indirect addressing?  d) register addressing?  e) register indirect addressing? |
|  |  |