Take-home Assignment-6

Advanced Computer Architecture (CS G524)
Semester-II, 2020-21
Department of Computer Science and Information Systems (CSIS)
BITS-Pilani, K K Birla Goa Campus, Goa, India.

Due date: May 24, 2021: 03:00 AM Marks: 45

Instructions: Individual **group** must solve this problem. We can run the tool to measure the similarity of your code with other group. The group's code similarity marks (similarity percentage*Total marks) can be deducted from the group's obtained marks or evaluators feel that the code is copied from somewhere. The penalty can be applied if the group submits the code beyond the due date and time. A penalty can be measured based on the submission's time.

Create a zip file as *<GroupNo>_simulateCache.zip* while submitting it. This file must contain all files, including the observation. Write the proper name of the variables and use the comment for documentation in the CPP.

Assume that you have written a CPP program for calculating the matrix multiplication. Next, the *gcc/cpp* compiler has translated such code for MIPS architecture. We have made the profile on the instruction LW. LW's addresses [word's address] are given in file *LW-sAddrs.txt* as the hexadecimal format. In MIPS, 32-bits comprise the word. The content of the corresponding address is the starting address of the word. The address sequence can be repeat. Let's consider the designed CPP iterated 10,000 times, and the *LW-sAddrs.txt* contains 10-address. In each iteration, the program fetched a block. The block size set a prior. So, at the 11-th iteration (or less than the 11-th iteration, which depends on block size) and onwards LW's address sequence can repeats.

Write a CPP to simulate the generalized cache behavior. The cache has these parameters: block size (b), number of the set (S), number of ways (N) and size of the cache (C). The number of address bits is denoted by A. Arrange the code properly. One can define a header file for the replacement algorithms, etc. Take the size of the cache as inputs. The cache must have these fields: Valid-bit, Tag and Data. Use these field appropriately. In the report, show the cache's valid-bit (binary), tag (hex) and data (hex) information in a table format at the end of 2nd repetition. The user can choose the following cache organization:

1. Fully associative cache (**Marks: 6**)
    a. Block size
        i. 1-word
        ii. 2-word
    b. Replacement algorithms
        i. FIFO (**Marks: 7**)
        ii. LRU (**Marks: 10**)
2. Direct mapped cache (**Marks: 5**)
    a. Block size
        i. 1-word
        ii. 2-word

3. Set associative cache (**Marks: 7**)
   - a. No of ways
     - i. Block size
       1. 1-word
       2. 2-word
   - b. Replacement algorithms
     - i. FIFO
     - ii. LRU

For the LW's address sequence, determine the effective miss rate for the chosen cache size and cache organization. Ignore the startup effect (compulsory misses).

Cache size = 16-words		
Cache organization	**Miss rate**	
	Block=1-word	**Block=2-word**
Direct		

Cache size = 16-words				
Cache organization	**Replacement policy**			
	FIFO		**LRU**	
	Miss rate			
	Block=1-word	**Block=2-word**	**Block=1-word**	**Block=2-word**
Fully Associative				
2-way Associative				
4-way Associative				

Write a report for your observation on cache miss rate, including the cache's status mentioned earlier.
(**Marks: 5+5**)

Note: One can design the same thing using Verilog HDL also.