NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 2 EXAMINATION 2023-2024

CE3001/CZ3001/SC3050 – ADVANCED COMPUTER ARCHITECTURE

Apr/May 2024 Time Allowed: 2 hours

INSTRUCTIONS

1. This paper contains 4 questions and comprises 6 pages.

2. Answer **ALL** questions.

3. This is a closed-book examination.

4. All questions carry equal marks.

5. The appendix provides the LEGv8 instruction formats.

---

1. (a) In the context of optimizing machine performance, an engineer at Company C introduces enhancement E1, which accelerates 45% of the original instructions by a factor of 3. Concerned about the complexity and cost effectiveness of E1, the management proposes an alternative enhancement E2. If E2 is applied to an as-yet-undetermined fraction of the original instructions, and speeds them up by a factor of 2, what percentage of all instructions should be optimized using enhancement E2 to achieve an equivalent overall speedup as obtained with enhancement E1.

(8 marks)

(b) The hexadecimal value of the current content of the program counter (PC) in a LEGv8 processor is 0x100100A8 as shown in Table Q1b. The code intends to change the PC value to 0x100000FC. State the addressing mode used and calculate the required offset (loop value) in hexadecimal for the control instruction.

Find the maximum address of the instruction memory to which the control of execution of a LEGv8 code could be moved backward by the unconditional control instruction.

Note: Question No. 1 continues on Page 2

Table Q1b

Program counter value in hexadecimal	Instruction
0x100000FC	Loop: LDUR X5 [X2, #0]
----	-----
0x100100A8	CBNZ X1, Loop

(9 marks)

(c) Use a neat diagram to show the datapath of a single-cycle architecture that supports the execution of both "XORI X5, X6, #5" and "STUR X3, [X4, #16]". Note that the datapath needs to have only minimal number of multiplexers (control signals can be simplified).

(8 marks)

2. Listing Q2 shows a code segment that is intended to be executed in a 5-stage pipelined LEGv8 processor. The program counter is updated with the branch target address at the Execute stage. Let the initial values in hexadecimal be `X7=0x0000000000010000` and `X8=0x0000000000001100` (`CBNZ`: *branch if not equal to 0*).

Listing Q2

```
I1    loop:  LDUR  X0, [X7,#0]
I2           LDUR  X1, [X8,#0]
I3           ADD   X2, X1, X0
I4           XORI  X3, X2,0x00F
I4           STUR  X3, [X7,#0]
I5           ADDI  X7, X7, #8
I6           SUBI  X8, X8, #16
I7           CBZ   X8, Finish
I8           B     loop

     Finish
```

(a) Calculate the steady-state Cycles Per Instruction (CPI) of the code segment in Listing Q2 with the help of a reservation table for the execution of the code if full data forwarding is allowed. Show the forwarded paths and the dependencies. Find the total number of loop iterations.

(8 marks)

Note: Question No. 2 continues on Page 3

(b) The code segment shown in Listing Q2 is now intended to be executed in a two-way superscalar processor. In the superscalar processor, both ways can be used for all the instructions. Find the CPI achieved by the superscalar architecture. Note that full data forwarding is allowed.

(7 marks)

(c) Perform unrolling by a factor of 2 and do necessary reordering. Use the unrolled and reordered code segment to be executed in a two-way superscalar machine. In the superscalar processor, both ways can be used for all the instructions. Find the CPI achieved by the superscalar architecture. Note that full data forwarding is allowed. Comment on the change in CPI when compared to Q2(b).

(10 marks)

3 (a) Name two cache organization schemes.

(4 marks)

(b) Consider a memory system composed of a 2 GB Byte-addressable main memory, and a 64 MB four-way set-associative cache with a block size of 512 Bytes. Determine the width of the address (in number of bits), and the widths of the tag, index, and offset fields in the address, respectively.

(8 marks)

(c) Following Q3(b), what is the miss rate of the cache when a program accesses the main memory in a totally random fashion, that is, random memory addresses are uniformly accessed for a sufficiently long time?

(6 marks)

(d) Tables Q3a and Q3b show the results of cache miss rate and average memory access time when a system is configured with different cache block sizes. Assume a cache hit takes 2 clock cycles. The cache miss penalties with different block sizes are provided in Table Q3b.

(i) Compute the missing entries X, Y and Z in Tables Q3a and Q3b.

(5 marks)

Note: Question No. 3 continues on Page 4

Table Q3a: Cache Miss Rate vs. Block Size

Block Size (Bytes)	Cache Size			
	4KB	16KB	64KB	256KB
16	10.95%	**X%**	1.95%	0.79%
64	6.82%	2.39%	1.02%	0.47%
256	8.36%	2.93%	0.86%	0.32%

Table Q3b: Average Memory Access Time (AMAT) vs. Block Size

Block Size (Bytes)	Miss Penalty	Cache Size			
		4KB	16KB	64KB	256KB
16	62	8.79	3.96	3.21	2.49
64	68	6.64	3.63	2.69	**Z**
256	92	**Y**	4.70	2.79	2.29

(ii) Between cache miss rate and average memory access time, which one is the more important performance metric to consider when designing a cache?

(2 marks)

4 (a) List the four types of processor architectures in the processor taxonomy according to Flynn's classification. Which of them are designed to take instruction-level parallelism into account?

(6 marks)

(b) Briefly explain the usage of the specifiers `__global__` and `__shared__` in CUDA C programming (in no more than 4 sentences).

(4 marks)

(c) Assume that there is only one branch instruction in a program. Consider the following sequence of actual outcomes of the branch instruction (N, N, N, N, N, N, N, N, N, N, T, T, N, N, T, N), where T means that the branch is taken, and N means that the branch is not taken. What is the prediction accuracy for the last 6 occurrences of this branch, i.e., (T, T, N, N, T, N), if the 2-bit branch predictor as shown in Figure Q4a is applied where the initial state of the predictor is unknown?

(6 marks)

Note: Question No. 4 continues on Page 5

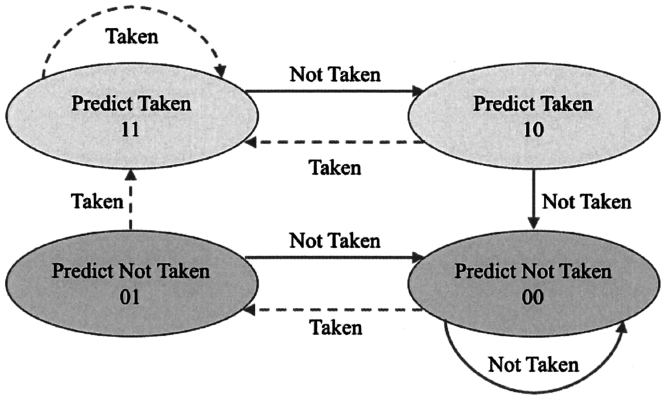


Figure Q4a

(d) The code snippet in Figure Q4b shows a C program that uses a CUDA kernel `saxpy()` to compute the SAXPY (Single precision A.X plus Y) operation:

$$\mathbf{Y} = A\mathbf{X}+\mathbf{Y}$$

where `A` is a scalar, while **`X`** and **`Y`** are vectors each consisting of `N` floating-point numbers.

```
Line
1   __global__
2   void saxpy(int n, float a, float *x, float *y){
3     int i = blockIdx.x * blockDim.x + threadIdx.x;
4     if (i < n)
5        y[i] = a*x[i] + y[i];
6     }
7
8   int main(void){
9     int N = 1024;     // size of vectors X and Y
10    float A = 7.0;
11    X = (float *)malloc(N*sizeof(float));
12    Y = (float *)malloc(N*sizeof(float));
13    // get values of vector X and Y
:          :
:          :
n     saxpy<<<.....>>>(N, A, d_X, d_Y);
:          :
:     return 0;
n+k   }
```

Figure Q4b

(i) Complete the code shown in Line ***n*** if the number of threads per block is set as 128. (Hint: indicate the necessary parameters.)

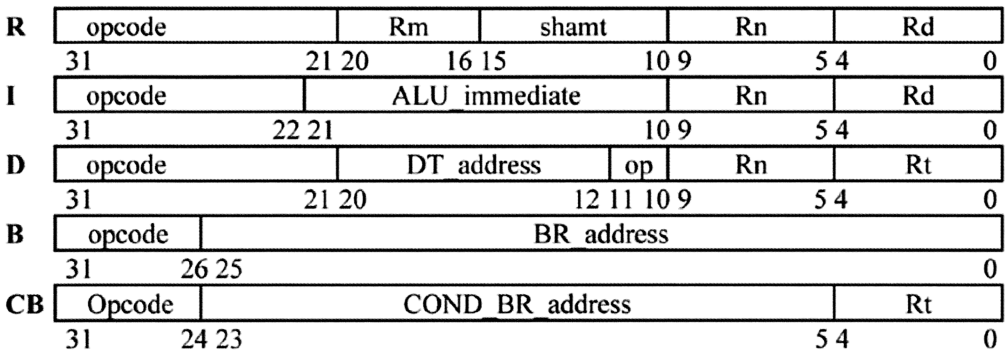
(4 marks)

Note: Question No. 4 continues on Page 6

(ii) Following Q4(d)(i), assume a Stream Multiprocessor (SM) in a GPU has sufficient register and shared memory resources to reside all the blocks. What is the total number of warps that will be created by launching the kernel?

(5 marks)

Appendix - Instruction Formats



END OF PAPER

CE3001 ADVANCED COMPUTER ARCHITECTURE
CZ3001 ADVANCED COMPUTER ARCHITECTURE
SC3050 ADVANCED COMPUTER ARCHITECTURE

Please read the following instructions carefully:

1. **Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**

2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.

3. Please write your Matriculation Number on the front of the answer book.

4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.