UNIVERSITY OF MORATUWA

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.Sc. Engineering
2011 Intake Semester 7 Examination

CS4342 ADVANCED COMPUTER ARCHITECTURE

Time allowed: 2 Hours September / October 2015

ADDITIONAL MATERIAL: *None*

INSTRUCTIONS TO CANDIDATES:

1. This paper consists of **5** questions in **6** pages.
2. Answer any **4** questions.
3. Start answering each of the main questions on a new page.
4. The maximum attainable mark for each question is given in brackets.
5. This examination accounts for 50% of the module assessment.
6. This is a closed book examination.

   NB: It is an offence to be in possession of unauthorised material during the examination.
7. Only calculators approved by the Faculty of Engineering are permitted.
8. Assume reasonable values for any data not given in or with the examination paper. Clearly state such assumptions made on the script.
9. In case of any doubt as to the interpretation of the wording of a question, make suitable assumptions and clearly state them on the script.
10. This paper should be answered only in English.

Question 1 (25 marks)

Suppose, just after graduation you were hired by Intel Inc. Your team is responsible for designing the 6th generation of Intel Core processors. Intel is hoping that the chips will kill off the tablet trend and persuade consumers to invest in hybrid or all-in-one Windows-based devices instead.

6th generation of Intel Core processors are expected to be capable of starting up in 0.5 seconds and have 2.5 times the performance of previous generations. It is also possible that devices running on the chip will offer users up to 3x more battery life than what they are currently used to.

– Part of the write up is extracted from www.wired.com

(i) Briefly explain the type of design you would recommend for each of the following items. Provide at least 2 reasons for each of your selection.

a) Number of processing elements (e.g., single core, multi core, or many core). [4]

b) Static vs. dynamic scheduling of instructions. [4]

c) Cache design (e.g., number of levels, cache size, cache associatively, etc.). [4]

(ii) a) What is the expected performance-to-power gain? [2]

b) What design options would you recommend to achieve the expected performance-to-power gain calculated in question ii(b)? Briefly discuss. [4]

(iii) Based on performance tests it has been identified that the memory sub-system on the 5th generation Intel Core processors is a major performance bottleneck. This is due to memory system not being able to provide the required bandwidth when each CPU core is running at its full speed. Waiting time for memory was 42% of the total time. Therefore, one of your team members suggested that by further increasing the memory bandwidth, it would be possible to gain the desired overall speedup of 2.5.

a) How much speed up in the memory sub-system will be required to achieve the desired overall speed up?

Hint: You may use the Amdahl's law given below for the calculation:

$$Speedup_{Overall} = \frac{1}{(1 - Fraction_{Enhanced}) + \frac{Fraction_{Enhanced}}{Speedup_{Enhanced}}}$$ [3]

b) Do you agree with your team member's idea of gaining such a speedup by improving only the bandwidth of the memory sub-system? Briefly explain. [4]

Question 2 (25 marks)

Consider the following Assembly code. The values of `F10` and `R5` are predefined.

```
Loop:
      LD     F1,0(R1)
      LD     F2,0(R2)
      DADD   F3,F1,F2
      DADD   F4,F3,F10
      SD     F4,0(R1)
      DADDUI R1,R1,#-4
      DADDUI R2,R2,#-4
      BNZ    R1,R5,Loop
```

(i) Identify all data dependencies in the above code. [4]

(ii) Calculate the number of clock cycles required for a single iteration of the above loop. Use the following table for latencies of Floating Point (FP) operations in MIPS. [3]

Instruction Producing Result	Instruction Using Result	Latency in Clock Cycles
FP ALU operation	Another FP ALU operation	3
FP ALU operation	Store double	2
Load double	FP ALU operation	1
Load double	Load or Store double	0

(iii) Improve the above code to reduce the number of clock cycles. [3]

(iv) Recalculate the clock cycles required by an iteration of the new loop. [2]

(v) Unroll the loop by a factor of 2 to further reduce the number of clock cycles. [10]

(vi) How much speedup can we gain per iteration compared to the original code? [3]

Question 3 (25 marks)

Following write up is extracted from the paper titled "Niagara: A 32-way Multithreaded SPARC Processor" by P. Kongetira, K. Aingaran, and K. Olukotun.

> "Niagara supports 32 threads of execution in hardware. The architecture organizes four threads into a thread group; the group shares a processing pipeline, referred to as the *Sparc pipe*. Niagara uses eight such thread groups, resulting in 32 threads on the CPU. Each SPARC pipe contains level-1 caches for instructions and data. The hardware hides memory and pipeline stalls on a given thread by scheduling the other threads in the group onto the Sparc pipe with a zero cycle switch penalty.
>
> The 32 threads share a 3-Mbyte level-2 cache. This cache is 4-way banked and pipelined for bandwidth; it is 12-way set associative to minimize conflict misses from the many threads. Commercial server code has data sharing, which can lead to high coherence miss rates. In conventional SMP systems using discrete processors with coherent system interconnects, coherence misses go out over low-frequency off-chip buses or links, and can have high latencies. The Niagara design with its shared on-chip cache eliminates these misses and replaces them with low latency shared-cache communication.
>
> Each thread has a unique set of registers and instruction and store buffers. The thread group shares the L1 caches, translation look-aside buffers (TLBs), execution units, and most pipeline registers. We implemented a single-issue pipeline with six stages (fetch, thread select, decode, execute, memory, and write back).
>
> The memory interface is four channels of dual-data rate 2 (DDR2) DRAM, supporting a maximum bandwidth in excess of 20 Gbytes/s, and a capacity of up to 128 Gbytes."

(i) Based on the above description answer the following questions.

a) Briefly explain what is meant by "single-issue pipeline with six stages". [3]

b) Briefly explain what is meant by "12-way set associative". [3]

c) Briefly explain what is meant by "cache is 4-way banked and pipelined". [3]

d) What is the functionality of the "translation look-aside buffer (TLB)"? [3]

e) Briefly explain what is meant by the following statement.
"The hardware hides memory and pipeline stalls on a given thread by scheduling the other threads in the group onto the Sparc pipe with a zero cycle switch penalty." [3]

f) Draw the cache/memory hierarchy while labelling necessary components and capacities. No need to draw all the execution units. [4]

(ii) Compare and contrast (i.e., similarities and dissimilarities) architectures of Niagara and Graphical Processing Units (GPUs). [6]

Question 4 (25 marks)

(i) How are superscalar processors different from multi-core processors? Briefly describe. [4]

(ii) "GPUs are essentially vector processors. For example, a Streaming Multiprocessor (SM) in a GPU is essentially a 32-wide vector unit".

Do you agree or disagree with this statement? Justify. [6]

(iii) Following figure shows the status of a system that supports Tomasulo algorithm.

Tomasulo Example Cycle 5

Instruction status:

Instruction		j	k	Issue	Exec Comp	Write Result
LD	F6	34+	R2	1	3	4
LD	F2	45+	R3	2	4	5
MULTD	F0	F2	F4	3		
SUBD	F8	F6	F2	4		
DIVD	F10	F0	F6	5		
ADDD	F6	F8	F2			

	Busy	Address
Load1	No	
Load2	No	
Load3	No	

Reservation Stations:

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
2	Add1	Yes	SUBD	M(A1)	M(A2)		
	Add2	No					
	Add3	No					
10	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

Register result status:

Clock		F0	F2	F4	F6	F8	F10	F12	...	F30
5	FU	Mult1	M(A2)		M(A1)	Add1	Mult2			

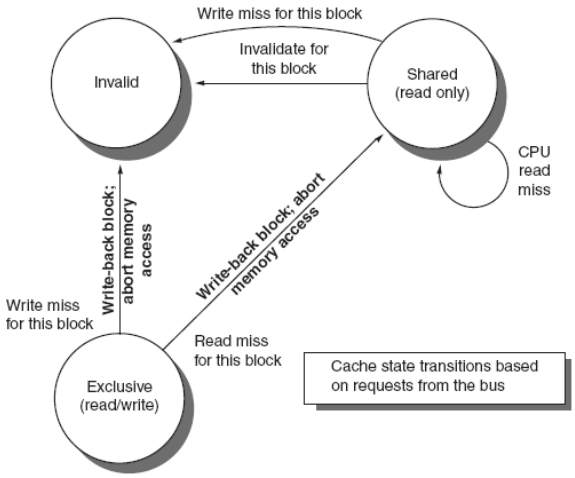
Discuss how the state will change during the 6th clock cycle. [10]

(iv) In an era where we have already hit the *Instruction Level Parallelism (ILP) Wall*, speculative forking proposes a new alternative to exploit further parallelism. Speculative forking allows a program to be dynamically multi-threaded by forking a new thread at every function and loop entry.

Briefly describe how such a design can lead to better performance. [5]

Question 5 (25 marks)

(i) Following figure is extracted from a state transition diagram of a Snoopy Cache Coherence protocol. Briefly describe what causes each of the 5 state transitions. [5]



(ii) On a given system 50% of the instructions are load and store. Miss penalty is 25 clock cycles. Instruction miss rate is 2% and data miss rate is 4%. If *n* instructions are to be loaded and executed, how many clock cycles will be wasted due to memory stalls? Suppose Clocks Per Instruction (CPI) is 1. [4]

(iii) Traditional cache architectures use SRAM-based cache hierarchies. However, advancements in technology now enable caches to be built from other technologies, such as Embedded DRAM (EDRAM), Magnetic RAM (MRAM), and Phase-change RAM (PRAM), in both 2D chips or 3D stacked chips.

Briefly discuss how such a hybrid cache design can lead to low cache access latency, lower power, and high density. [5]

(iv) One of the architectural barriers we have reached is the *utilization wall*, where the percentage of transistors that a chip design can switch at full frequency drops exponentially with each processor generation because of the power constraints.

Briefly explain how does the *c-cores* in the Greendroid mobile application processor overcome the utilization wall. [5]

(v) Consider a warehouse-scale datacentre with 100,000 nodes. Do you recommend having hard disks or solid state drives within the datacentre? Discuss while considering the following specifications.

	Hard Disk	**Solid State Drive**
No of disks that can be attached to a node	2	4
Capacity	1 TB	500 GB
Access Time	6 ms	0.1 ms
IO Performance	400 IOPS	6,000 IOPS
Mean Time To Failure	3 years	10 years
Energy	12 W	3 W
Cost	Rs. 10,000	Rs. 25,000

[6]

---------- END OF THE PAPER ----------