National Exams December 2018

17-Comp-A3, Computer Architecture

3 hours duration

NOTES:

1. If doubt exists as to the interpretation of any question, the candidate is urged to submit with the answer paper, a clear statement of any assumptions made.
2. This is an OPEN BOOK EXAM.
   An approved Casio or Sharp calculator is permitted.
3. FIVE (5) questions constitute a complete exam paper.
   The first five questions as they appear in the answer book will be marked.
4. Each question is of equal value.
5. Most questions require an answer in essay format. Clarity and organization of the answer are important.

Marking Scheme

1. (a) 5 marks (b) 5 marks (c) 10 marks
2. (a) 5 marks (b) 10 marks (c) 5 marks
3. (a) 5 marks (b) 5 marks (c) 10 marks
4. (a) 5 marks (b) 10 marks (c) 5 marks
5. 20 marks
6. (a) 5 marks (b) 5 marks (c) 10 marks

1. **(a)** A simple processor has a memory address space of 512K words. Its memory is word addressable. In the first implementation of the processor all instructions took the form:

```
OP Destination, Source, nextPC
```

where Destination, Source, and nextPC were word addresses in memory and OP was some operation to be performed. For example, the following instruction would read the values stored at addresses 0x2000 (hexadecimal) and 0x1000, add them, write the result to address 0x1000, and then update the program counter (PC) to 0x3000 (the address where the next instruction to execute is at in memory):

ADD 0x1000,0x2000,0x3000

The instruction encoding used 64 bits for all instructions and was as follows:

7	19	19	19
OP	Destination	Source	nextPC

The top row shows the width in bits of each field.

The first implementation had 127 instructions in total. Assume that the second implementation of the architecture has to: (1) Remain backward compatible with the first implementation (i.e., should be able to correctly execute all the instructions of the first implementation), and (2) use 64-bits to encode each instruction. Can the new implementation introduce more instructions? Can it introduce one more instruction? How about 10 more instructions? Can it introduce instructions that use registers?

(b) A program comprises 5 instructions, two of which are memory reads, reading 4 bytes from memory each. Each instruction is encoded using 32-bits. What is the minimum number of bytes that a processor will have to read from memory to execute this program assuming that each instruction is executed exactly once?

(c) Discuss the pros and cons of using polling to communicate with peripheral devices.

2. **(a)** Two 16-bit registers A and B contain the values 0x FFFE (shown in hexadecimal) and 0x0012, respectively. What is the value of A + B in decimal, if (i) A and B are to be interpreted as unsigned integers, and (ii) A and B are to be interpreted as signed integers in 2's complement. Explain your answer.

(b) The IEEE standard for single precision floating point representation of real numbers uses 32-bits in total comprising an 8-bit exponent E, a sign bit S, and a 23 bit mantissa M. The number encoded is:

(-1)^S x 2^(E – 128) x 1.M

Given any two floating point numbers A and B, can A * B (product) be represented precisely in this format? If not, how many more bits would be needed for the mantissa, exponent and sign?

(c) Explain how a two-dimensional array A of 16-bit values having 64 rows and 32 columns will be stored in memory. Where in memory would be A[R][C] where R and C integers if the array is stored starting from memory address 0x1000?

3. **(a)** A processor has a 4GB byte-addressable address space. How will a 32KB, 4-way set-associative cache with 64-byte blocks be indexed. Explain your answer.

   (b) A block cached in set 0x2 (hexadecimal) of the cache described in part (a) is tagged with 0x11 (hexadecimal). What is the range of addresses it contains in hexadecimal?

   (c) Why do modern processors use caches? What would be the pros and cons of having more registers instead?

4. **(a)** Why are branch, jump and other control flow changing instructions a challenge for pipelined processor implementations?

   (b) A graphics processor (GPU) such as the GTX1080 is about 100 times slower compared to a general processor (CPU) such as the i7-8900k when executing a single multiplication. However, there are programs for which the GPU can be orders of magnitude faster the CPU. How is this possible? What are the properties of such programs?

   (c) Why would we want to build a dedicated, specialized video encoder/decoder? What are the pros and cons compared to a software implementation using a general-purpose processor?

5. A multi-cycle implementation of a processor requires 4 cycles for the LI instruction (load immediate), 3 cycles for all ARITHMETIC and BRANCH instructions, 5 cycles for the MEMORY READ instruction, and 4 cycles for the MEMORY STORE instruction. No other instructions exist. The

implementation can be modified so that the clock frequency is increased by 3% but at the expense of needing 5 cycles for the LI instruction.

On the average, programs execute instructions with the following frequency:

LI	10%
ARITHMETIC	50%
MEMORY READ	20%
MEMORY STORE	10%
BRANCH	10%

Which implementation (original or modified) will execute programs faster and by how much assuming the aforementioned mix of instructions? Explain your answer.

6. **(a)** A memory chip has the following interface: A0-A21 are 22 single bit input address lines specifying which row is accessed, a single bit input signal R/W! specifies whether the access is a read (1) or a write (0), E is a single bit input signal that must be 1 to access the chip. The data values that are read or written appear on the four D1-D3 single bit output/input pins. When E is 0 the D1-D0 pins are in high-Z. What is the total capacity of this memory chip in bytes?

(b) Using as many as necessary of the chips described in part (a), synthesize the equivalent of an 8-bit wide memory that has a 8MB total capacity. You can use a few additional logic gates as needed.

(c) A processor's memory *interface* has the following signals: L0-L31 are 32 single-bit output address lines specifying which memory address is accessed, a single bit output ME signal is 1 when an access is taking place, a single bit R/W! output signal is 1 or 0 for reads and writes respectively, D0-D7 are eight single-bit bi-directional data signals that either provide the value to be written or expect the value to be read. Connect one 8MB chips with the interface described in part (b) to this memory interface. The chip should be activated for accesses in the 8MB range starting at address 0x1000 0000.