CSC 137 Cokgor Homework 5 (5 points)

1) How many hexadecimal digits are needed to represent a binary number with 32 digits? (0.1 point)
   Ans: 32 binary bits => 32 /4 = 8 Hexadecimal digits

2) A microprocessor can access any memory location by outputting the address of that location onto its address bus. You have designed a microprocessor and have made available 12 address lines for the address bus. You are using a memory device that is 8 bits wide (i.e. each memory location holds 1 Byte).
   a. How many Kilobytes is your microprocessor's memory capacity? (State your answer in Kilobytes) (0.1 point)
      2^{10} = 1024 = 1 Kilobyte
      12 address lines => 2^{12} = 4 Kilobytes

   b. You have decided that you will need more memory capacity. How many address lines do you need to make available if you wanted to double the memory capacity? (0.1 point)
      13 address lines => 2^{13} = 8 Kilobytes

3) How many address lines are required for a 2MByte memory? (0.1 point)
   21 address lines.

4) You will be building a memory with 14-bit address and 8-bit data word size. If this memory were constructed from 1K x 1-bit RAM chips, how many chips would be required? (0.2 point)
   14 bits address => 2^{14} = 2^4 x 2^{10} = 16Kbytes. 16 x 8 = 128 memory chips.

5) Can you build a computer with non-volatile memory (e.g. ROM) only? Explain your answer. (0.1 point)
Yes; the instructions can be in the non-volatile memory. The computer will not be able to save data. Many embedded systems can be designed this way.

6) Can you build a computer with volatile memory (e.g. RAM) only? Explain your answer. (0.1 point)
No, we need non-volatile memory to store the startup instructions for the microprocessor. Non-volatile memory will not retain its contents when the power is turned off. The microprocessor will not run when the power is turned on since there will aren't any instructions.

7) What does the Program Counter register (or the Instruction Pointer register) do? (0.1 point)

   Program Counter holds the memory address of the next instruction that processor fetches from the instruction memory.

8) You are designing a simple microprocessor for a specific application. You have determined that you will need 32 unique operations in the instruction set. Each instruction needs two register fields to hold the operands. Each register field can address 16 internal registers.

a) How many bits does your op-code field need to be? (0.1 point)

5 bits. 2^5 = 32 unique operations

b) How many bits does each of your register fields need to be? (0.1 point)

Each 4 bits. $2^4 = 16$

9) We have seen that fields of an instruction encoding can be mixed to achieve different goals depending on the instruction type. Refer to page 20 of the Instruction Set Architecture lecture notes. As examples, the Rm field of an R-format instruction is 5 bits, whereas the address offset field of a D-format instruction is 9-bits. Or, the opcode field of a D-format instruction is 11 bits, whereas the opcode field of an I-format instruction is 10-bits. Answer the following questions with field mixing in mind.
Assume the following instruction format:

opcode	Rm	Rn

a. The overall instruction length is 8 bits, the register fields (R1, R2) are 3 bits each. How many unique opcodes can you have? (0.1 point)
The opcode field is 2 bits; $2^2 = 4$ unique opcodes.

b. If you mix the fields, can you have 3 x two address and 8 one address instructions? Justify your answer. (0.2 point)
Yes, it is possible.

3 x two-address instructions	Opcode field		Rm			Rn		
	0	0	x2	x1	x0	y2	y1	y0
	0	1	x2	x1	x0	y2	y1	y0
	1	0	x2	x1	x0	y2	y1	y0
8 x one-address instructions	Opcode field					Rn		
	1	1	0	0	0	y2	y1	y0
	1	1	0	0	1	y2	y1	y0
	1	1	0	1	0	y2	y1	y0
	1	1	0	1	1	y2	y1	y0
	1	1	1	0	0	y2	y1	y0
	1	1	1	0	1	y2	y1	y0
	1	1	1	1	0	y2	y1	y0
	1	1	1	1	1	y2	y1	y0

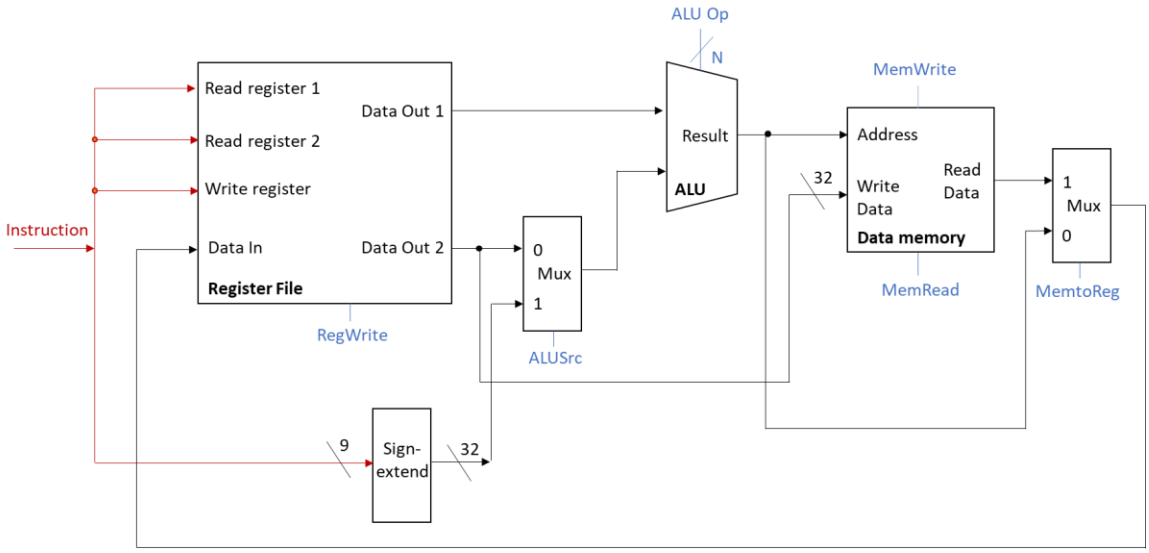
10) If the data storage is arranged according to the little-endian representation, how is the word 0x4CE3F90A stored in memory location 0x080001AC. (0.1 point)

Memory	Address
	0x080001AF
	0x080001AE
	0x080001AD
	0x080001AC
	0x080001AB
	0x080001AA
	0x080001A9

Ans:

Memory	Address
4C	0x080001AF
E3	0x080001AE
F9	0x080001AD
0A	0x080001AC

11) Consider the datapath developed during the class lectures for R-format and D-format instructions (shown below) and answer the following questions.



a. If the RegWrite control signal is stuck at 0 (i.e. not asserted), which instructions shown below would malfunction: (0.4 point)
   i. ADD R0, R1, R2 Malfunction- cannot write back to R0
   ii. SUB R0, R1, R2 Malfunction- cannot write back to R0
   iii. LDR R0, [R1, #4] Malfunction- cannot load to R0
   iv. STR R1, [R0, #4] Works normally. Does not need to write to register file.

b. If the MemWrite control signal is stuck at 0 (i.e. not asserted), which instructions shown below would malfunction: (0.4 point)
   i. ADD R0, R1, R2 Works normally. Does not need data memory access.
   ii. SUB R0, R1, R2 Works normally. Does not need data memory access.
   iii. LDR R0, [R1, #4] Works normally. Does not need data memory access.
   iv. STR R1, [R0, #4] Malfunction. Data will not be written to memory.

c. If the ALUSrc control signal is stuck at 0, which instructions shown below would malfunction: (0.4 point)

i. ADD R0, R1, R2 Works normally. Does not need data memory access.
ii. SUB R0, R1, R2 Works normally. Does not need data memory access.
iii. LDR R0, [R1, #4] Malfunction. Address will not be calculated correctly.
iv. STR R1, [R0, #4] Malfunction. Address will not be calculated correctly.

12) Consider the instruction execution timing table of a single cycle processor given below and answer the following questions:

Instruction	Instruction Fetch	Register Read	ALU Operation	Data Access	Register Write	**Total Time**
LDR (Load Register)	200 ps	100 ps	200 ps	200 ps	100 ps	**800 ps**
STR (Store Register)	200 ps	100 ps	200 ps	200 ps		**700 ps**
ADD	200 ps	100 ps	200 ps		100 ps	**600 ps**
SUB	200 ps	100 ps	200 ps		100 ps	**600 ps**

a) If every instruction in the table takes exactly one clock cycle, what must be the clock cycle time to accommodate all instructions. (0.1 point)
800 ps; the time it takes for the slowest instruction.

b) What would be the clock speed of the processor with the cycle time you calculated above? (0.1 point)
1/800ps = 1.25GHz

c) Assume that both instructions and data are in the same memory space, i.e. there is a single address and data bus pair for memory access. What will be the clock cycle time if you reduce the delay of the memory access by half. (0.1 point)
Slowest instruction delay time: 100ps + 100ps + 200ps + 100ps+100ps = 600ps.

d) What would be the clock speed of the processor after the reduction of memory access time as in (c) above. (0.1 point)
1/600ps = 1.66GHz

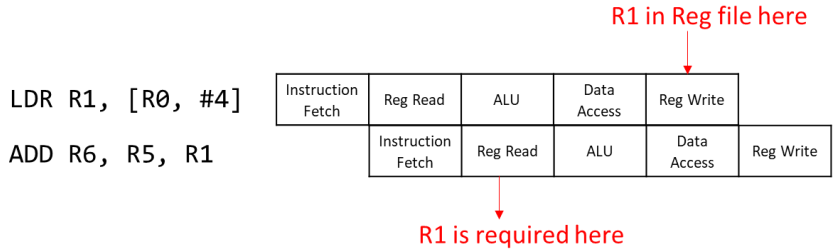
13) Consider the instruction execution for a 5 stage pipelined processor for all instructions as shown below. Each phase takes 200 ps. Instruction Fetch and Data Access use the same memory. Assume that Reg Read and Reg Write phases do not conflict.

Instruction Fetch	Reg Read	ALU	Data Access	Reg Write

Answer the following questions:

a) Can the following program segment be executed without a pipeline hazard? If there is a hazard, then what type? (0.3 point)

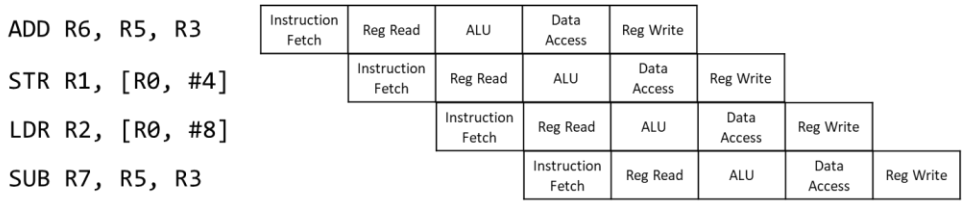
LDR R1, [R0, #4]
ADD R6, R5, R1



The LDR instruction doesn't write R1 until the fifth stage. ADD instruction needs the result in R1. This is a data hazard.

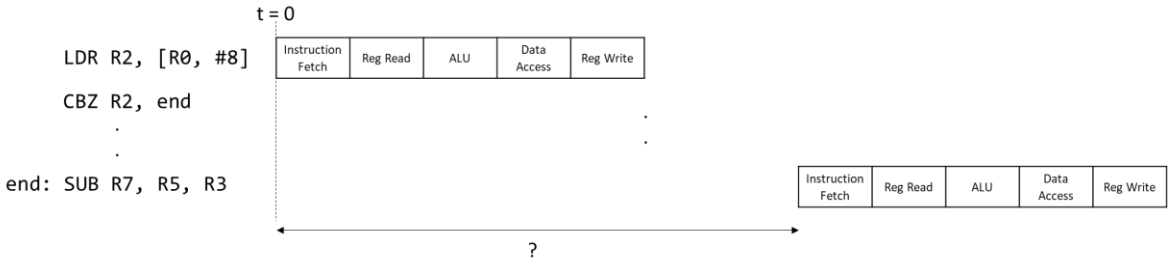
b) Can the following program segment be executed without a pipeline hazard? If there is a hazard, then what type? (0.3 point)

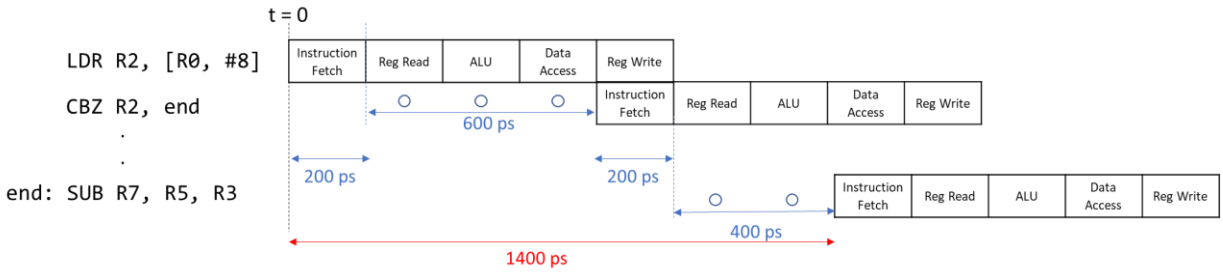
ADD R6, R5, R3
STR R1, [R0, #4]
LDR R2, [R0, #8]
SUB R7, R5, R3



No pipeline hazards. Note that ADD instruction does not use Data Access, therefore there is no conflict with the SUB instruction fetch.

c) The LDR instruction in the program segment shown below is fetched at t=0. If the branch is taken as a result of the CBZ instruction, when will the SUB instruction be fetched? Justify your answer by showing the pipeline. You must avoid any hazards by stalling the pipeline when necessary. (0.4 point)





14) Consider the memory map of a hypothetical processor below. Does this processor implement memory mapped I/O or isolated I/O? Explain your reasoning. (0.1 point)

Flash
DRAM
Peripherals
SRAM
ROM

Memory mapped I/O. Peripherals are in the memory map.

15) Given the Interrupt Vector Table below, answer the following questions: (0.4 point)
Note: Interrupt vector table is the area of the memory where the interrupt vectors are stored. This area of the memory is shown below.

Interrupt Vector Table		
Interrupt Name	*Memory Address*	*Memory Content*
. .	. .	. .
EXTI3	0x00000064	0x0800030C
. . .	. .	. .
Reset	0x00000004	0x2000020C

a) Which address is the Interrupt Vector for Reset located? 0x00000004
b) Which address is the Interrupt Vector for EXTI3 interrupt located? 0x00000064
c) What is the interrupt vector for Reset? 0x2000020C
d) What is the address of the EXTI3 Interrupt Service Routine? 0x0800030C

16) Direct Memory Access (DMA):
A program needs to transfer a block of data from the computer hard drive to the main memory. A DMA request is initiated via the DMA controller. Answer the following questions based on what you learned in the class: (0.4 point)
a) Is this a DMA read or a DMA write transfer? DMA write – from peripheral to memory.
b) Will the data that is being transferred be loaded to the microprocessor register, or will the data be transferred between the peripheral and the memory directly?
Data will be transferred between the peripheral and the memory directly
c) Is it the DMA controller or the processor that generates the corresponding hard drive and memory control signals? DMA controller.
d) Can the processor initiate a separate transfer between an Ethernet port and the main memory during this data transfer? No, during this time the microprocessor has not access to the bus until the transfer is completed.