McGill

DECEMBER 2021
FINAL EXAMINATION

Computer Organization

ECSE 324

14-Dec-2021 6.30pm

EXAMINER: Prof. Christophe Dubach ASSOC. EXAMINER: Prof. Brett Meyer

STUDENT NAME:		**McGILL ID:**									

EXAM:	CLOSED BOOK x ~~OPEN BOOK ☐~~
	SINGLE-SIDED x ~~PRINTED ON BOTH SIDES OF THE PAGE ☐~~
	MULTIPLE CHOICE ANSWER SHEETS: ~~YES ☐~~ NO x NOTE: The Examination Security Monitor Program detects pairs of students with unusually similar answer patterns on multiple-choice exams. Data generated by this program can be used as admissible evidence, either to initiate or corroborate an investigation or a charge of cheating under Section 17 of the Code of Student Conduct and Disciplinary Procedures. ANSWER BOOKLET REQUIRED: ~~YES ☐~~ NO x EXTRA BOOKLETS PERMITTED: ~~YES ☐~~ NO x ANSWER ON EXAM: YES x ~~NO ☐~~
	SHOULD THE EXAM BE: RETURNED x ~~KEPT BY STUDENT ☐~~
CRIB SHEETS:	~~PERMITTED ☐~~ Specifications: (ex: one 8 1/2X11 handwritten double-sided sheet) NOT PERMITTED **x**
DICTIONARIES:	~~TRANSLATION ONLY ☐ REGULAR ☐~~ NOT PERMITTED x
CALCULATORS:	NOT PERMITTED x ~~PERMITTED (Non Programmable) ☐ PERMITTED (Programmable) ☐~~
ANY SPECIAL INSTRUCTIONS: e.g. molecular models	

Part A: General knowledge **[30 points]**

A.1: multiple choice questions [1 point] each.

Only select a single answer. If more than one answer is selected, you will get zero for that question.

1. The first transistor was invented:

   a. at Bell Labs in 1948
   b. at IBM in 1972
   c. at Intel in 1930
   d. at Sun Microsystems in 1962
   e. at Dell in 1990

2. What defines a Load/Store architecture?

   a. arithmetic instructions operates on registers only
   b. load and store are the only instructions that access memory
   c. branch instructions use relative addressing
   d. registers are 32 bit wide
   e. memory addresses are 32 bit wide

3. An unconditional branch instruction:

   a. only changes the PC if a condition is true
   b. always compare two values held in registers
   c. never modifies the value of the PC
   d. update the content of the link register
   e. none of the above

4. Loops in ARM assembly are implemented using:

   a. branch instructions to labels
   b. brand-and-link instructions
   c. push and pop instructions
   d. interrupts
   e. load instructions

5. Which instructions is not available in ARM assembly?

   a. logical shift right
   b. logical shift left
   c. arithmetic shift right
   d. arithmetic shift left
   e. branch to an address contained in a registers

6. Assuming the following C variable declarations:

```
int arr[6];
int i;
int v;
```

Which ARM instruction implements the following C statement, assuming *arr* is in R0, *i* in R1 and *v* is in R2?

```
v = arr[i];
```

a. LDR R1, [R2, R1, LSL#2]
b. LDR R2, [R0, R1, LSL#2]
c. LDR R1, [R2, R1, LSL#4]
d. LDR R2, [R0, R1, LSL#4]
e. None of these

7. Which of these ARM instructions uses PC-relative addressing?

a. LDR R0, [PC, #4]
b. SUB R1, R2, PC
c. ADD, R0, PC, R1
d. All of them
e. None of them

8. Consider an ARM branch instruction where the offset field contains the binary value "00000000 00000000 00000010". Suppose that this branch instruction is at located address 0x10000000. What is the target address?

a. 0x10000000
b. 0x10000002
c. 0x10000004
d. 0x10000008
e. 0x10000010

9. The ARM CPSR contains:

a. the PC
b. a zero condition flag
c. a S bit
d. a branch target address
e. an opcode

10. After assembling, the content of the symbol table contains:

a. an address for each symbol
b. the object memory map
c. all the instructions
d. Greek letters
e. the PC

11. The linker:

    a. needs a C file to produce a binary
    b. concatenates object files into a single binary
    c. loads the program into memory
    d. execute the program by jumping to the start address
    e. none of these

12. When polling an I/O device, the program must:
    a. wait for an interrupt to be raised
    b. wait for the device to be ready by reading the status register
    c. write into the status register
    d. use the control register
    e. change the interrupt vector table

13. The interrupt vector table contains:

    a. the addresses of the interrupt service routines
    b. the addresses of all branch instructions
    c. the instructions corresponding to each interrupt service routine
    d. the content of the CPSR
    e. the return address

14. When the CPU is in the IRQ mode, it:

    a. cannot uses branch instructions
    b. uses a set of shadow registers different from the ones in normal mode
    c. never uses the CPSR
    d. none of these
    e. all of these

15. An asynchronous bus synchronizes devices using:

    a. only a master-ready signal
    b. only a slave-ready signal
    c. both a master-ready and slave-ready signal
    d. a clock signal
    e. a clock signal and a master-ready signal

16. Bus arbitration:

    a. is only used for asynchronous buses
    b. is only used for synchronous buses
    c. uses a master-ready signal
    d. requires a DMA module
    e. ensures only one master uses the bus at any given time

17. How does a device know that it should respond to a specific bus request?

a. by using the address lines and an address decoder
b. by using the data lines and a register
c. by looking at the content of the status register
d. by using a tri-state buffer
e. none of these

18. What is the purpose of the start and stop bit in a UART?

a. they indicate the number of data bits to transfer
b. they identify null terminated strings
c. they change the clock frequency of the receiver
d. they synchronize the receiver and transmitter
e. none of these

19. DRAMs:

a. use two coupled inverters to store a bit in each cell
b. must be refreshed periodically
c. are non-volatile
d. all of the above
e. none of the above

20. How many address bits are there in a 16x8 RAM:

a. 16
b. 3
c. 8
d. 4
e. None of these

21. If you are tasked to design a 1M x 32 multi-chip memory using 512K x 8 RAM chips. How many chips do you need?

a. 16
b. 8
c. 4
d. 1
e. None of these

22. A DMA controller:

a. is an I/O device connected to the bus
b. can trigger an interrupt once a transfer is done
c. can access the main memory
d. all of these
e. none of these

23. Assume a fully associative cache which stores 128 cachelines/blocks of 64 bytes each. How many comparators are found in the hardware implementation of this cache?

a. 1
b. 64
c. 128
d 512
e. 2048

24. The page table:

a. contains the virtual address corresponding to each page
b. contains the physical location of each page in memory if present
c. is stored in the CPU
d. has a fixed size of 4KB
e. is always stored in SRAM

25. The TLB is:

a. a CPU register
b. a buffer for a transmitter device on a bus
c. a cache for the page table
d. required to implement virtual memory
e. at most 4KB in size

26. In a non-pipelined 5-stage RISC CPU, what is the expression that controls when the PC register is update?

* is the logical AND operator and + is the logical OR operator
Tx means current stage is x
MFC = Memory Function Completed
BR means the CPU is processing a branch instruction and the branch is taken

a. T1 * MFC
b. T3 * BR
c. T1 * MFC + T3 * BR
d. T2 * MFC + T3 * BR
e. T3 * (MFC + BR)

27. In the typical 5-stage pipelined RISC CPU seen in the lecture, what is the name of the third stage?

a. Memory
b. WriteBack
c. Decode
d. Execute/Compute
e. Fetch

28. What is the key characteristic of a “carry-lookahead” adder?

a. three inputs can be added together
b. the adder can be used to subtract
c. the carry dependency between between the 1-bit full adders is removed
d. the addition can be pipelined
e. the adder can handle signed integer

29. Which decimal number can never be precisely encoded with a fixed-point representation?

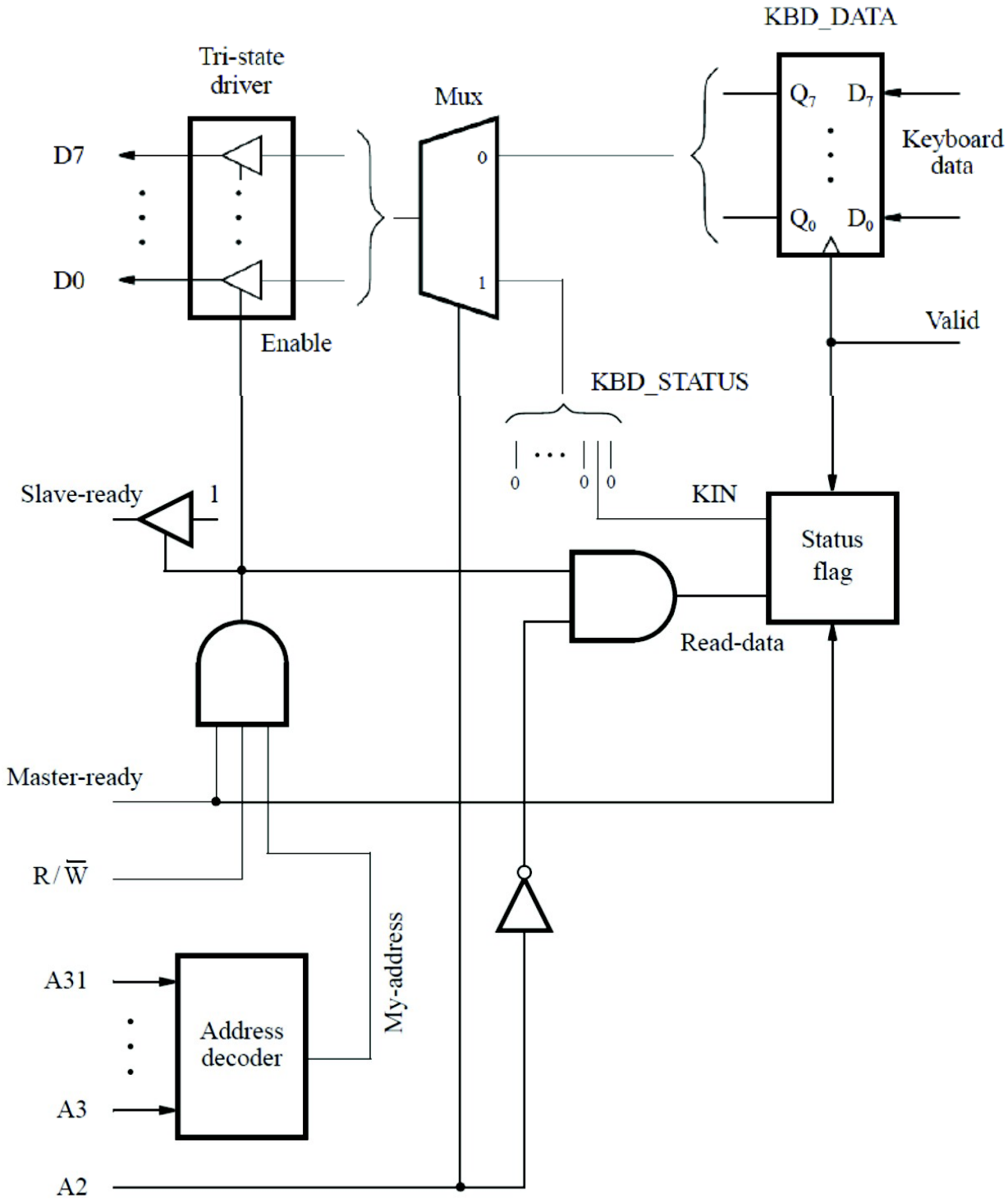
a. 72.25
b. -1.0
c. 22.2
d. 9.75
e. 1.5

30. A value in the IEEE 754 floating-point representation is represented as $(-1)^S * (1.m) * 2^{e-127}$ where *s* is the a sign bit, *e* is an 8-bit exponent and *m* is a 23-bit mantissa. How is the decimal value 2.25 encoded?

a. s=0, $e=128_{10}$, m=001 0000 0000 0000 0000 0000 0000 0000
b. s=0, $e=128_{10}$, m=100 1000 0000 0000 0000 0000 0000 0000
c. s=0, $e=127_{10}$, m=001 0000 0000 0000 0000 0000 0000 0000
d. s=0, $e=127_{10}$, m=000 1000 0000 0000 0000 0000 0000 0000
e. s=1, $e=128_{10}$, m=000 1000 0000 0000 0000 0000 0000 0000

Part B: I/O **[10 points]**

In the next questions, we will consider the circuit below which corresponds to the input interface of a keyboard.



For multiple choice questions, select a single answer. If more than one answer is selected, you will get zero for that question.

1. What the signals D0 .. D7 connected to? **[1 point]**

   a. The control lines of the bus
   b. The main memory
   c. The device status register
   d. The device data register ← okay as well (appear in register on the figure)
   e. The data lines of the bus

2. What is the purpose of the multiplexer? **[1 point]**

   a. To deal with any interrupts raised by the device
   b. To ensure only one device is connected to the bus as any given time
   c. To select between the value contained in the data or status register
   d. To produce data only when master-ready is asserted
   e. To decode the address on the bus address lines

3. Assuming the KBD_DATA register is memory mapped at address 0x4000 and KBD_STATUS is at address 0x4004. What should be the **binary** value of each of the following signals for Slave-ready to be 1? **[6 points]**

If the value of the signal does not matter, write an X.

Master-ready:	1
R/W :	1
A2:	X
A31,A30,...,A3:	0000 0000 0000 0000 0100 0000 0000 0
KIN:	X
Valid:	X

4. What happens with this I/O interface when R/W is 0, Master-ready is 1 and My-address is 1? **[1 point]**

   a. Slave-Ready becomes 1
   b. The Tri-state buffers are enabled for the bus data lines
   c. Nothing happens
   d. Some data is written into the KBD_DATA register
   e. the KBD_STATUS register is udpatded

5. This circuit is designed to interface with a: **[1 point]**

   a. Synchronous bus
   b. Asynchronous bus
   c. Serial Bus
   d. All of these
   e. None of these

Part C: Caches **[10 points]**

For this problem, assume a byte addressable memory with a 32-bit address space, and a cache with the following configuration:

- 2 ways set-associative cache
- cache line (a.k.a. block) size of 32B
- total number of sets is 64
- LRU cache replacement policy

1. What is the total size of this cache? **[1 point]**

4 KB (or 128 blocks / cache lines)

2. How many bits of the address determines the set? **[1 point]**

6 bits

3. How many bits of the address are used to select a byte in a cache line? **[1 point]**

5 bits

4. How many bits are required for the tag? **[1 point]**

21 bits

5. Assume the following memory load access happens in sequence. Determines for each access whether it is a hit or a miss in the cache: **[4 points]**
(*hint: 0x800 == 2048_{10} and be aware of the cache line size!)*

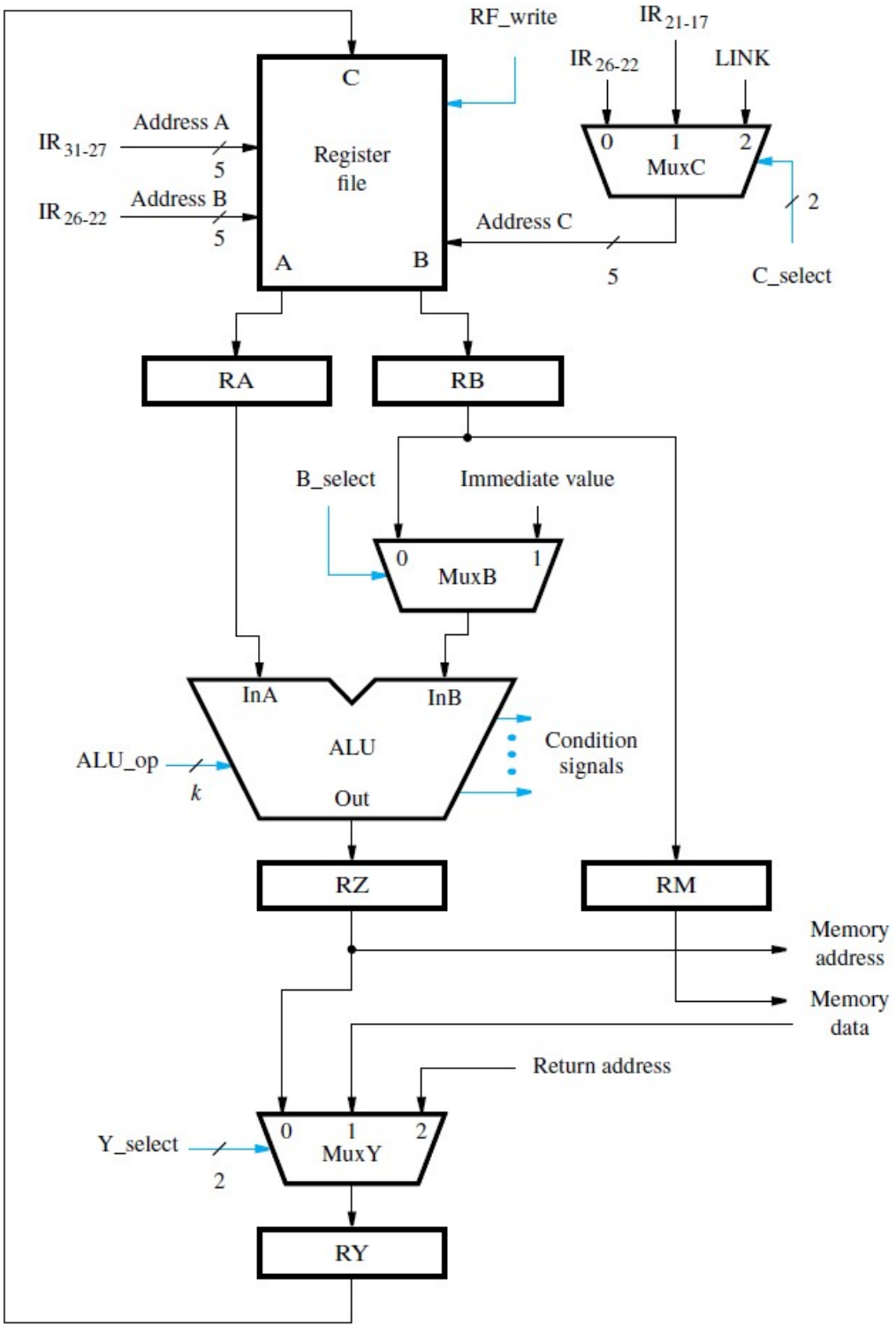
0x00000000	a. miss	b. hit
0x00000001	a. miss	b. hit
0x00000001	a. miss	b. hit
0x00000801	a. miss	b. hit
0x00000802	a. miss	b. hit
0x00001001	a. miss	b. hit
0x00000001	a. miss	b. hit
0x00000801	a. miss	b. hit

6. Given the same trace of memory accesses as above, how many hits would this produce in the case of a 4-ways set-associative cache with 32 sets, and with the same replacement policy and cache line size? **[2 points]**

5 hits

Part D : Processor Design **[10 points]**

Consider the following 5-stage RISC CPU datapath:



Given the following sequence of assembly instruction:

ADD R1, R2, R3
SUB R5, R1, R2
LDR R1, [R1,#4]

For multiple choice questions, select a single answer. If more than one answer is selected, you will get zero for that question.

1. How many cycles are required to execute this sequence of instruction assuming all memory accesses complete in 1 clock cycle and the processor is **not pipelined**? (note that there are still registers between each stage) **[1 point]**

a. 3
b. 5
c. 7
d. 15
e. 16

2. When the LDR instruction executes through the stages, what are the value of the following control signals?

2.1 Address C (5 bits) **[1 point]**

00001

2.2 B_select (1 bit) **[1 point]**

1

2.3 Y_select (2 bits) **[1 point]**

01

3. Identify all the data dependencies in the instruction sequence. For each dependency specify the two instructions involved and the register. **[2 points]**

ADD -> SUB with R1
ADD -> LDR with R1

4. How many cycles are required to execute this sequence of instruction assuming all memory accesses complete in 1 clock cycle and the processor is **pipelined**? **[2 points]**

a. 3
b. 7
c. 9
d. 10
e. 15

1pt for answer c
2pt for answer d

5. Assuming there is forwarding path connecting Rz to InA. How many cycles are required to execute this sequence of instruction assuming all memory accesses complete in 1 clock cycle and the processor is **pipelined**? **[2 points]**

a. 3
b. 7
c. 9
d. 10
e. 15

1pt for answer b
2pt for answer c