ECSE 324 Winter 2021 Final

Part A: Choose Wisely

2pts each unless otherwise noted.

Input/Output

1. Which of the following can be managed with CPU interrupt hardware? (MS)
    a. Page faults
    b. Divide by zero
    c. Timer events
    d. Breakpoints
2. Tri-state buffers … (MS)
    a. Prevent multiple bus drivers from contending with each other
    b. Have high-impedance output when not enabled
    c. Are only needed for data lines
    d. Are only useful for serial buses
3. Match the following statements with (a) synchronous buses, (b) asynchronous buses, (c) both, and (d) neither: (Matching)
    a. Requires a clock signal
    b. Requires arbitration between multiple bus leaders
    c. Always used for serial communication
    d. Best suited for long buses
4. UART uses a start bit of 0 before sending data bits. Why? (MS)
    a. The falling edge synchronizes transmitter and receiver
    b. The falling edge begins a counter that the receiver uses to determine when to sample data bits
    c. The start bit is used for arbitration
    d. The length of the start bit determines the number of data bits

Memory

5. Why are SRAM cell arrays usually square? (MS)
    a. To reduce word line capacitance
    b. To reduce bit line capacitance
    c. To reduce SRAM access latency
    d. To make the array easier to fit into the larger design

6. *4pts* Assume a 32-bit RISC CPU's main memory is implemented using a byte-addressable 512Mx8 DRAM. The cell array is structured such that each row is 64K bits wide. How many bits wide must the address port be? (MC)
    a. 15 bits
    b. 16 bits
    c. 29 bits
    d. 32 bits
    e. None of the above

7. *4pts* Which of the following statements are true of (a) SRAM, (b) DRAM, (c) both, or (d) neither? (Matching)
    a. Used for smaller memories
    b. Used for larger memories
    c. Is nonvolatile
    d. Is volatile
    e. Addressing takes multiple clock cycles
    f. Cell arrays are organized to be square
    g. Cells must be periodically accessed to maintain state
    h. All accesses take the same amount of time

8. Why do we use DMA? (MS)
    a. To reduce the number of memory accesses
    b. So the CPU can execute other code during long data transfers
    c. To save time by avoiding arbitration for bus accesses
    d. To reduce the number of pipeline stalls in the CPU

9. *4pts* Assume a system with CPU and DMA. In a for loop (1K iterations), the CPU reads a word from one array, and writes a word to another array. This requires m memory accesses per loop iteration. The DMA transfer can perform this transfer with 2 memory accesses per loop iteration, $m > 2$. What is m? (MC)
    a. 2
    b. 4
    c. 6
    d. 8
    e. None of the above

10. Which of the following scenarios take advantage of (a) spatial locality, (b) temporal locality, (c) both? Assume a unified 16KB cache with 32B cache blocks, and word-aligned accesses. (Matching)
    a. A single traversal of an array
    b. A single traversal of a lengthy if-else block of code
    c. Repeated traversal of a loop
    d. Repeated traversal of a linked list

11. How many comparators are required for an 8-way set associative cache? Assume a cache with 4KB capacity, and 32B blocks. (MC)
    a. 1
    b. 2
    c. 4
    d. 8
    e. None of the above

12. Assume a 32-bit address space. Main memory is 512 MB, and a two-way set associative cache holds 64 KB, in 64 B cache blocks. How any blocks in main memory map to any given set in the cache? (MC)
    a. 4096
    b. 8192
    c. 16384
    d. 32768
    e. None of the above

13. Consider the following sequence of instructions. How many memory accesses are required to execute it? (MC)

```
LDR R5, [R3, #16]
ADD R6, R2, R5
MUL R7, R3, R5
```

    a. 0
    b. 1
    c. 3
    d. 4
    e. None of the above

14. Rank order the following in terms of *volatility*, from least (1) to most (4). (Ordering)
    a. DRAM
    b. SRAM
    c. Flash
    d. Magnetic hard-disk drive

15. Consider a 16-bit address space utilizing virtual memory. Assume 8KB pages. The first four entries in the page table follow; select the physical address translation for 0x7AFE. (MC)

Valid	Page Frame
0	1
1	3
1	6
0	0

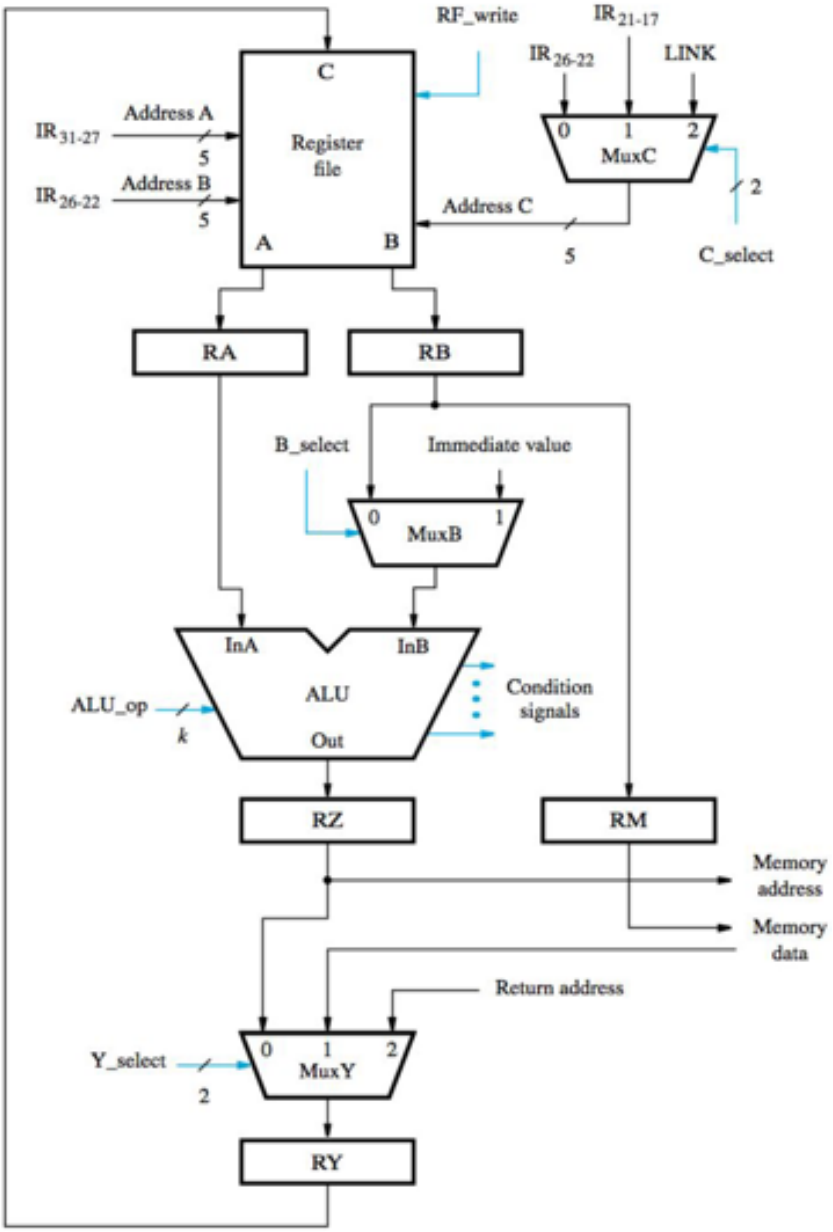
    a. 0x3AFE
    b. 0x7AFE
    c. 0xCAFE
    d. 0x1AFE
    e. None of the above

16. The TLB is … (MS)
    a. A required component for systems with virtual memory
    b. A cache where the tag is a *virtual page number*, and the data is the corresponding *physical page number*
    c. A direct-mapped cache
    d. Part of the processor's memory management unit

Processor Design

17. Which forwarding paths might be needed to forward the value of R2 to the instruction STR R1, [R2, R3]? (MS)
    a. Memory to Compute
    b. Write-back to Compute
    c. Write-back to Memory
    d. Other

18. The datapath below does not support addressing like LDR R0, [R1, R2 LSL#2].



Which of the following changes can be made to support this addressing mode? (MS)

a. Add a shifter circuit between RA and InA
b. Add a shifter circuit between RB and MuxB
c. Add a shifter circuit between MuxB and InB
d. Add a shifter circuit between ALUOut and RZ

19. Which of the following addressing modes are supported by the above datapath? (MS)
    a. LDR R1, [R2]
    b. STR R1, [R2, #4]
    c. LDR R1, [R2], #4
    d. STR R1, [R2], #4

20. Which of the following instructions will fail to operate correctly if B_select is stuck at 0 due to hardware failure? (MS)
    a. LDR R1, [R2, #4]
    b. ADD R1, R2, #10
    c. ADD R1, R2, R3
    d. BEQ R1, R2, LOOP

Part B: The matrices are multiplying! (10 pts)

Complete the following assembly to implement a function `mvm` that multiplies a 2x2 matrix (bytes) with a 2x1 vector (bytes) and saves the result as a 2x1 vector (half words).

Assume that:

- A1 is used to pass the address of the source matrix
- A2 is used to pass the address of the source vector
- A3 is used to pass the address of the result vector

```
mvm:
     PUSH      ________

     MOV       V3, #0
     MOV       V4, #0
     MOV       V5, #0

mvmloop:
     CMP       ________
     BEQ       mvmret

     LDRB      V1, [A1, V4]
     LDRB      V2, [A2, V5]

     MLA       ________

     ADD       V4, V4, #1
     ADD       V5, V5, #1

     CMP       V5, #2
     BNE       mvmloop

     STRH      ________
     MOV       V3, #0
     MOV       V5, #0

     B         mvmloop

mvmret:
     POP       ________
```

Part C: Cache-cache (10 pts)

Assume a 16-bit address space and byte-addressable, little-endian, main memory. The table below illustrates the state of a 16-set direct-mapped cache; each set consists a single block of 16B (with the least significant byte on the right).

Set	Valid	Tag	Data														
			f	e	d	c	b	a	9	8	7	6	5	4	3	2	1	0
0	0	34	83	ab	57	88	31	e7	a0	5e	8f	4c	e4	df	63	77	c6	5e
1	0	1f	d2	ce	aa	7c	82	23	d7	d7	1f	64	76	c7	a3	dc	19	78
2	1	d3	e6	65	8d	3b	e3	53	f8	d7	f8	40	3d	f2	80	42	0a	f0
3	1	2d	0c	b8	00	e4	d7	60	58	b0	71	4e	10	f7	b9	5f	58	c7
4	0	67	5f	91	85	64	de	45	ca	e8	84	b7	c0	41	6e	5c	d4	64
5	1	d3	be	8a	35	b0	9b	75	70	48	08	06	a0	bd	13	c6	69	a0
6	0	22	2a	a4	7b	bd	72	a1	f7	51	19	3b	55	fc	d8	52	30	35
7	1	00	30	4b	12	c0	2f	51	8d	3a	8e	57	c7	8b	be	44	93	a2
8	1	ab	df	cd	5a	f7	7f	4b	cd	b3	58	a5	4d	46	f5	30	91	e6
9	0	12	ee	f9	86	7d	24	dd	98	7c	8c	a7	de	ec	bc	a5	06	cb
a	1	e3	2c	5b	af	c6	92	09	cf	ca	a9	e5	f2	87	d2	91	9b	1d
b	1	df	e8	af	5b	2e	4a	77	e0	0c	1d	d0	ce	0b	b8	ac	a6	9c
c	0	ee	9d	b7	11	e9	68	7c	a3	4f	91	17	2f	b3	3d	85	92	33
d	1	1d	cd	8b	2f	5d	c8	de	16	90	d2	4e	e3	20	12	83	f2	62
e	0	a0	d5	d0	a3	4b	9e	1a	d4	55	a3	9b	1d	98	26	38	a5	e0
f	1	55	a5	65	2b	9f	55	20	72	e6	25	23	07	e1	bf	7f	72	1e

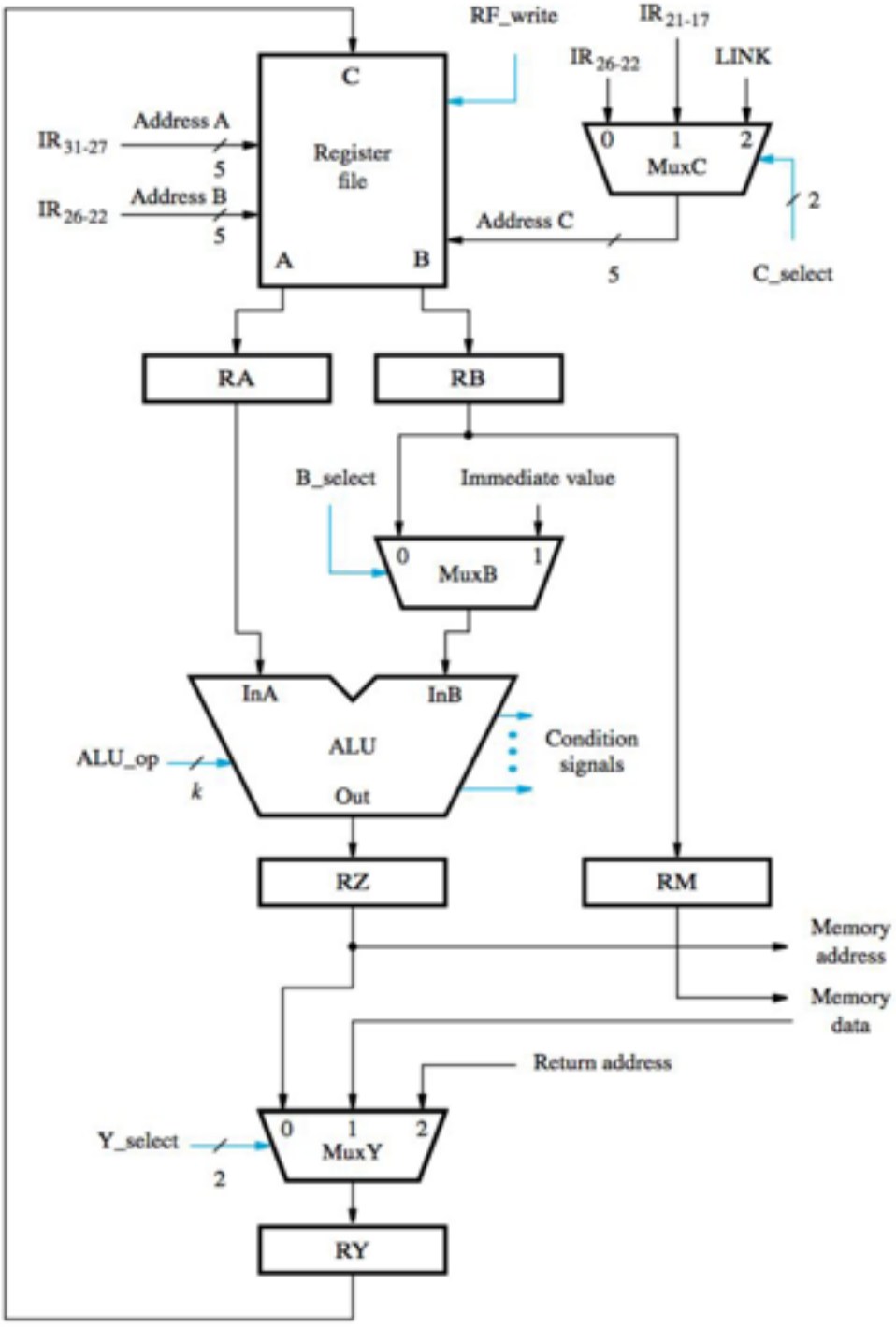
2 pts each unless otherwise noted.

a. Does an access to address 0xd32a hit or miss? (Hit/Miss)
b. Does an access to address 0xab90 hit or miss? (Hit/Miss)
c. What 4B word would be returned by a load from 0x2d34? (SA)
d. *4pts* Calculate the hit rate for the following sequence of accesses. (SA)

0xe3a8, 0xe3ac, 0xe3b0, 0xe3b4, 0xe3b8, 0x55f0, 0xe3bc, 0xeebc

Part D: Processor Design (10 pts)

Consider the following 5-stage RISC CPU datapath.



Use the following sequence of assembly instructions to answer the following questions.

```
LDR R5, [R3, #16]
ADD R6, R2, R5
MUL R7, R3, R5
```

2 pts each unless otherwise noted.

a. How many cycles does this sequence of instructions take to execute **without pipelining**? Assume that every memory access completes in two cycles. (MC)
    a. 7
    b. 11
    c. 15
    d. 19
    e. None of the above
b. Identify all data dependencies in the code sequence. Specify each instruction with a dependency on an earlier instruction, and the register involved. (SA)
c. *3pts* How many cycles does this sequence of instructions require *with pipelining*, but **without forwarding**? Assume that every memory access completes in two cycles. (MC)
    a. 10
    b. 11
    c. 12
    d. 13
    e. None of the above
d. *3pts* How many cycles does this sequence of instructions require *with pipelining and forwarding*? Assume that every memory access completes in two cycles. (MC)
    a. 10
    b. 11
    c. 12
    d. 13
    e. None of the above

Part Z: Bonus!

a. *1pt* Write a haiku or other short poem about ECSE 324, Computer Organization.

b. *1pt* Do you grant Professor Meyer permission to share your poem (at his discretion) on Twitter @bretthmeyer? (Yes/No)