Sample Final Exam
COMP2300/ENGN2219/COMP6300

(Digital Logic Fundamentals)

Q1. Draw the following schematics for an 8-input (8:1) multiplexer.

A. Gate level: as a combination of basic AND, OR, NOT gates. Use as few gates as possible.

B. Module level: as a combination of 2-input (2:1) multiplexers. Use as few 2-input multiplexers as possible.

Q2. In one of the lectures on digital logic fundamentals, we discussed that "general purpose" hardware circuits have some key features that distinguish them from "special purpose" hardware circuits. Discuss briefly with examples.

(Finite State Machines)

Q3. Design an FSM with one input, A, and two outputs, X and Y. X should be one if A has been one for at least three consecutive cycles. Y should be one if A has been 1 for at least two cycles altogether (not necessarily consecutively). Show your state transition diagram, encoded state transition table, next state and output equations, and schematic.

Q4. Explain with examples the significance of sequencing overhead in synchronous digital circuits.

(ARM Assembly)

Q5. Convert the following C code into ARM assembly code. Your code should respect the ARM calling conventions.

C Code

```
int q(int n) {
  if (n == 0)
    return 0;
  else if (n == 1)
    return 1;
  else
    return n + q(n-2);
}
```

Q6. It is well-known that any recursive function has a corresponding iterative solution. Write an iterative solution for the function above C code implements. (No marks will be deducted for syntax errors.) Explain the pros and cons of recursion compared to iteration from a microarchitecture perspective.

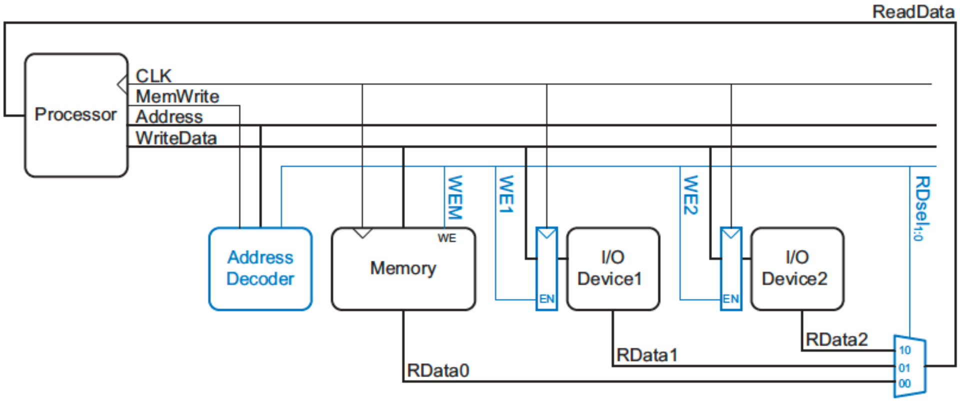
Q7. Consider the following QuAC assembly code. Assume you are considering the Smith$_2$ branch predictor with the initial state of the counter in the branch history table is “Strongly Taken.” What is the prediction accuracy of the Smith$_2$ predictor for the branch instruction (`jpz end`) inside the loop. What is the prediction accuracy if the initial state is “Strongly Not Taken?”

```
@ QuAC Assembly Code
@ Fib number calculator

        movl r1, 0x1
        movl r2, 0x0
        movl r3, 0x19
    loop:
        cmp r3, rz
        jpz end
        add r4, r1, r2
        mov r1, r2
        mov r2, r4
        movl r4, 0x1
        sub r3, r3, r4
        jp loop
    end:
        nop
```

(I/O)

Q8. Suppose that I/O device # 1 is the figure below is assigned the memory address 0x10002000. Show the ARM assembly code for writing the value 20 to I/O device # 1, and for reading the output value from I/O device # 1.



Q9. Briefly explain the characteristics of following I/O systems.

A. Memory-mapped DMA interrupt-driven I/O

B. Port-mapped programmed polling I/O

Q10. A large server is in production at a bank. We can upgrade the CPU to make it 25% faster for $5000; or upgrade its solid-state drive for $3000 to make the drive go 70% faster. It is observed that programs spend 80% of their time computing (CPU is 100% utilized), and 20% of their time waiting for disk to respond. Explain an upgrade of which component would offer a greater benefit for a smaller expense.

(Microarchitecture)

Q11. List all the true and false (anti and output) dependences in the following assembly code. Suppose instructions can be arbitrarily reordered in a hypothetical machine. What kind of hazards are possible when executing the instruction sequence below? Assume the hypothetical machine implements no mitigating strategies we covered in lectures.

```
ADD  R0,  R0,  R4
LDR  R1,  [R0, #16]
ADD  R1,  R1,  #8
STR  R1,  [R0, #16]
SUB  R0,  R1,  #1
```

Q12. The following figure shows the execution an instruction sequence on a CDC 6600 pipelined CPU with a scoreboard. There are several errors in the figure. List all errors you can find.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
i1: LDR r2, [r7,#4]	FE	DE	RR	DI	IS	$EX_{@}$	$EX_{D\$}$	... miss ...				WB					
i2: ADD r4, r2, #1		FE	DE	RR	DI	IS	IS	IS	EX	WB							
i3: ADD r8, r5, #2			FE	DE	RR	DI	IS	EX	WB								
i4: ADD r2, r8, #3				FE	DE	RR	DI	IS	EX	WB							

Q13. Assume the processor in the above question has a register file with eight registers. Now consider the corrected cycle-by-cycle diagram for the execution of the four instructions above. Show the state of the scoreboard when `i3` is in the EX-stage.

Q14. Discuss in which way the following design philosophies are similar or different. Discuss the similarities by explaining their key features. If the design philosophies share no similarity, discuss the differences between them.

A. RISC and VLIW

B. Hardware speculation and statically scheduled in-order superscalars

C. VLIW and hardware speculation

D. Microprogramming vs. hardwired control

Q15. Use a simple multi-cycle pipeline diagram to show the forwarding and stalls needed to execute the following instructions on the pipelined ARM processor. Which registers are being written, and which are being read on the fifth cycle?

```
ADD R0, R4, R9
SUB R0, R0, R2
LDR R1, [R0, #80]
AND R2, R1, R0
```

Q16. How many cycles are required to run the following program on the multicycle ARM processor? What is the CPI of the program? Answer the same questions for a 5-stage ARM pipelined processor with hazard detection.

```
     MOV  R0,  #0
     MOV  R1,  #0
     MOV  R2,  #0
LOOP
     CMP  R2,  R0
     BEQ  L2
     ADD  R1,  R1,  R0
     ADD  R0,  R0,  #1
     B    LOOP
L2
```