COMP4300
Computer Architecture, Spring 2022
Final Examination

(If you are unsure of the answer to any question, tell all your reasoning for giving the answer you did, for partial credit)

1. (15 points) Suppose that a redesign of a given machine speeds up execution time for ALU operations by a factor of 10. When the optimization is implemented, the overall average speedup of execution time for the machine is 1.3. What percentage of the original execution time was spent doing ALU operations?

2. (6 points) What is the theoretical maximum speedup you could get in the previous problem by speeding up ALU operations infinitely?

3. (5 points) Suppose on a non-pipelined single-processor machine, you have the following breakdown: floating instructions make up 3% of the dynamic instruction count, and take 10 cycles to execute. Load/store instructions take 5 cycles to execute and make up 30% of the mix. Jumps take 5 cycles and make up 20%. All other instructions average 1 cycle. What is the average CPI?

4. (6 points) What are the three types of hazards that lead to pipeline stalls?

5. (7 points) On a PDP-8, an instruction to deposit and clear the accumulator resides at address 0x005. The address to which the accumulator's value is to be deposited is 0x010. Give the binary for the 12 bits of this instruction. See last page of exam for PDP-8 programming card.

___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___ ___

6. (10 points) Describe how superscalar processors exploit parallelism.

7. (5 points) On the AUBIE processor, how many 32-bit words does an instruction require if it does not use an address or immediate operand? How many 32-bit words are required if an address is part of the instruction? If an immediate is part of the instruction?

8. (6 points) In the pseudo-assembly code below, identify the data hazards by the two instructions involved in each dependence . Identify the hazard as an RAW (true), WAR (anti) or WAW (output) hazard. For example, if there is a write-after-write dependence between operation i and operation j, you would write "i -> j (waw)".

```
LD  R1, R2(100)      ; instruction 1

LD R4,  #3000        ; instruction 2

ADD R2, R1, R4       ; instruction 3

SUB R5, R2, R1       ; instruction 4

MUL R4,  R1, R2      ; instruction 5
```

9. (5 points) True or false: PDP-8 instructions are of variable length.

10. (5 points) In a machine with 36-bit physical addresses, a cache block is 2K bytes in size. The cache can hold 1MB. It is direct-mapped. How many bits is the index value used with this cache?

11. (5 points) In a machine with 36-bit physical addresses, a cache block is 2K bytes in size. The cache can hold 1MB. It is direct-maped. How many bits make up the offset portion of the address?

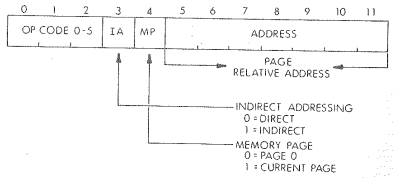
12. (5 points) In a machine with 36-bit physical addresses, a cache block is 2K bytes in size. The cache can hold 1MB. It is direct-mapped, How many bits make up the tag portion of the address?

13. (10 points) In a computer with load/store architecture and a cache memory system, 30% of the instructions access memory (i.e. loads and stores). A cache hit takes 1 cycle and a miss take 10 cycles, and the miss rate for instructions is 1% and for data 5%. What is the average memory access time per instruction?

14. (10 points) For a machine most of whose code involves nested FOR-loops, which option would likely yield better performance, and why: a 1-bit or a 2-bit branch predictor ? Why?

BASIC INSTRUCTIONS

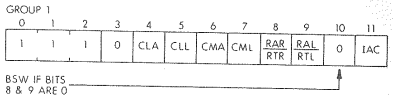
AND	0000	logical AND	2.6
TAD	1000	2's complement add	2.6
ISZ	2000	increment, and skip if zero	2.6
DCA	3000	deposit and clear AC	2.6
JMS	4000	jump to subroutine	2.6
JMP	5000	jump	1.2
IOT	6000	in/out transfer	—
OPR	7000	operate	1.2



Memory Reference Instruction Bit Assignments

GROUP 1 OPERATE MICROINSTRUCTIONS (1.2μsec)

			Sequence
NOP	7000	no operation	—
CLA	7200	clear AC	1
CLL	7100	clear link	1
CMA	7040	complement AC	2
CML	7020	complement link	2
RAR	7010	rotate AC and link right one	4
RAL	7004	rotate AC and link left one	4
RTR	7012	rotate AC and link right two	4
RTL	7006	rotate AC and link left two	4
IAC	7001	increment AC	3
BSW	7002	swap bytes in AC	4



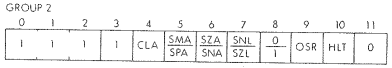
Logical Sequences:

1—CLA, CLL
2—CMA, CML
3—IAC
4—RAR, RAL, RTR, RTL, BSW

Group 1 Operate Instruction Bit Assignments

GROUP 2 OPERATE MICROINSTRUCTIONS (1.2μsec)

			Sequence
SMA	7500	skip on minus AC	1
SZA	7440	skip on zero AC	1
SPA	7510	skip on plus AC	1
SNA	7450	skip on non-zero AC	1
SNL	7420	skip on non-zero link	1
SZL	7430	skip on zero link	1
SKP	7410	skip unconditionally	1
OSR	7404	inclusive OR, switch register with AC	3
HLT	7402	halts the program	3
CLA	7600	clear AC	2



Logical Sequences:

1 (Bit 8 is Zero)—Either SMA or SZA or SNL
1 (Bit 8 is One)—Both SPA and SNA and SZL
2 —CLA
3 —OSR, HLT

Group 2 Operate Instruction Bit Assignments

COMBINED OPERATE MICROINSTRUCTIONS (1.2μsec.)

			Sequence
CIA	7041	complement and increment AC	2, 3
LAS	7604	load AC with switch register	2, 3
STL	7120	set link (to 1)	1, 2
GLK	7204	get link (put link in AC bit 11)	1, 4
CLA CLL	7300	clear AC and link	1
CLL RAR	7110	shift positive number one right	1, 4
CLL RAL	7104	shift positive number one left	1, 4
CLL RTL	7106	clear link, rotate 2 left	1, 4
CLL RTR	7112	clear link, rotate 2 right	1, 4
SZA CLA	7640	skip if AC=0, then clear AC	1, 2
SZA SNL	7460	skip if AC=0 or link is 1, or both	1
SNA CLA	7650	skip if AC=0, then clear AC	1, 2
SMA CLA	7700	skip if AC<0, then clear AC	1, 2
SMA SZA	7540	skip if AC<0	1
SMA SNL	7520	skip if AC<0 or link is 1, or both	1
SPA SNA	7550	skip if AC>0	1
SPA SZL	7530	skip if AC>0, and if the link is 0	1
SPA CLA	7710	skip if AC>0, then clear AC	1,2
SNA SZL	7470	skip if AC=0 and link=0	1