Alan Palayil

ECE 242 Midterm 2

Q1: 15/20
Q2: 13/15
Q3: 15/15
Q4: 19/20
64/70

1) a) CMP.W -(A2), D1

Compares the contents of -(A2) with D1

A2 → A2 - 2 = 8008 , D1 - (A2) → 0006 - 0002 = 0004

N=0 Z=0 V=0, C=0

b) BSET.B D2, (A5)+

A5 → $800E , 3 bit is compared of $1529 and value is stored

c) ASL.W D3, D6

Destination operand is shifted 5 times left arithmetically

A124 → 1010 0001 0010 0100

Shift 1 → 0100 0010 0100 1000

Shift 2 → 1000 0100 1001 0000

Shift 3 → 0000 1001 0010 0000 X=0, N=0, Z=0, V=0, C=0

Shift 4 → 0001 0010 0100 0000

Shift 5 → 0010 0100 1000 0000 → 2480

d) ROXL.W #3, D5 (X = 1)

D5 = 8422 → 1000 0100 0010 0010 (X=1)

Rotate 1 → 0000 1000 0100 0101 (C=1)

Rotate 2 → 0001 0000 1000 1010 (X=0, C=0)

Rotate 3 → 0010 0001 0001 0100 (X=0, C=0)

X=0, C=0, N=0, V=0, Z=0

Register	Current	New
D0	$A452	
D1	$0066	
D2	$0003	
D3	$0005	
D4	$A6AA	
D5	$8422	$2114
D6	$A124	$2480
D7	$BED2	

-1

Register	Current	New
A0	$8002	
A1	$800F	
A2	$800A	$8008
A3	$8004	
A4	$8010	
A5	$800E	$8011
A6	$8002	
A7	$8000	

-2

Memory	Current	New
$8000	$AB32	
$8002	$B302	
$8004	$C020	
$8006	$800A	
$8008	$0002	
$800A	$BCDE	
$800C	$FFFE	
$800E	$1529	$1529

Status Register					
	X	N	Z	V	C
a.	X	0	0	0	0
b	X	X	X	X	X
c	0	0	0	1	0
d	0	0	0	0	X

2) a)

MC68K	MIPS
CISC architecture	RISC architecture
It is a big-endian only architecture	It is a bi-endian architecture (big & little)
This has 8 general purpose registers (32-bit) & 7 address registers with (24 bits)	This has a 32 general purpose registers (32-bit)

b) Stage 1) Instruction fetch
Stage 2) Instruction Decode
Stage 3) Execution
Stage 4) Data Memory Access
Stage 5) Write back to register

With the five stages, the five processes can run in parallel after the instruction is decode stage begins. The instruction register will be populated with data.

c) We can use the difference between the stack pointer and the frame pointer. The stack pointer (SP) points at the top of the stack, while the frame pointer (FP) points at the top of the stack frame i.e. a constant for a particular function. As space gets allocated for more variables, the SP moves to point to the remaining memory's beginning location. Hence, the current stack size can be calculated by taking the difference of the SP & FP values.

not answering the question

3) a) and $t1, $t0, $v0

Format: and rd rs rt

Opcode	Rs	Rt	Rd	Function code
000000	01000	00010	01001	00000 100100

The machine code is 0x 010 24 824

b) or $s5, $a0, $t7

Format: or Rd, rs, rt

Opcode	Rs	Rt	Rd	Function code
000000	00100	01111	10101	00000 100101

The machine code is 0X 008FA825

c) srav $a0, $a1, $t8

Format: srav rd, rt, rs

Opcode	Rs	Rt	Rd	Function code
000000	11000	00101	00100	00000 000111

The machine code is 0x03052007

1110 0 111
A S
101 0 0 101

4) ORG $1000
START:

MOVE.B #$00, D1 // Clear D1 to 00.

LABEL:
~~MOVE.B #$00, (A1)~~ E1 //
For test and set flag

LoopA TAS (A1)

BNE LoopA

no need...

ANDI.B #$A5, D1 // Clears K, M, N P

ORI.B #$A5, D1 // Set J, L, O and Q.

CMPI.B #$A5, D1 // Test to compare

BEQ LABEL // If the set isn't correct

END START