Xi'an Jiaotong-Liverpool University

西交利物浦大學

PAPER CODE	EXAMINER	DEPARTMENT	TEL
CPT210	Jianjun Chen	Computing	81889137

2nd SEMESTER 2022/23 RESIT EXAMINATION

Undergraduate – Year 3

Microprocessor Systems

TIME ALLOWED: 2 Hours

INSTRUCTIONS TO CANDIDATES

1、 **This is a closed-book examination.**

2、 **Total marks available are 100.**

3、 **Answer all questions.**

4、 **Only English solutions are accepted.**

Condition codes

Code	Suffix	Flags	Meaning
0000	EQ	Z set	equal
0001	NE	Z clear	not equal
0010	CS	C set	unsigned higher or same
0011	CC	C clear	unsigned lower
0100	MI	N set	negative
0101	PL	N clear	positive or zero
0110	VS	V set	overflow
0111	VC	V clear	no overflow
1000	HI	C set and Z clear	unsigned higher
1001	LS	C clear or Z set	unsigned lower or same
1010	GE	N equals V	greater or equal
1011	LT	N not equal to V	less than
1100	GT	Z clear AND (N equals V)	greater than
1101	LE	Z set OR (N not equal to V)	less than or equal
1110	AL	(ignored)	always

Question A (20 Marks)

1. Convert -28.75 into IEEE 754 single-precision format and explain how this number is stored on a little-endian platform. Please write down all intermediate steps clearly for the conversion and write down each byte (in hexadecimal) along with its corresponding memory address. The most significant byte of this number should be stored at 0x8000004D.
(10 marks)

2. Explain what is one's complement encoding and compare it against two's complement encoding, what are its advantages (or disadvantages)?
(5 marks)

3. Convert -72 into sign and magnitude encoding and two's complement encoding. Assume that one byte is used to store this number. Please write down the conversion steps.
(5 marks)

Question B (20 Marks)

1. Assume that the following registers initially hold the values given:

```
r0 : 0xFB32C76D
r1 : 0x0D796B7C
r2 : 0x79C286A4
```

What values are held in registers r4, r5, r6, and r7 after the following ARM instructions are executed?

a. `SUB r4, r0, #0x00000078`

b. `EOR r5, r1, #0x0008C000`

c. `BIC r6, r2, r0`

d. `ADD r7, r1, r2, LSL #4`

(4 * 2 = 8 marks)

2. The instruction STM supports several addressing modes:

STM{*addr_mode*}{*cond*} *Rn*{!}, *reglist*

Answer the following questions:

a. Explain the behaviour of STM under these addressing modes.

(4 marks)

b. What is the result of executing "stmia sp!, {r0, r1, r2}" given the following register status? Please write down the status of the stack as well as the value of SP after execution.

(4 marks)

R0 = 0xAA
R1 = 0xBB
R2 = 0xCC
SP = 0xFF000000

3. A "branch and link" instruction uses a special register called a link register. What is the purpose of the link register? How does a "branch and link" instruction differ from a simple branch instruction?

(4 marks)

```
r0 : 0xFB32C76D
r1 : 0x0D796B7C
r2 : 0x79C286A4
```

What values are held in registers r4, r5 executed?

a. SUB r4, r0, #0x00000078

b. EOR r5, r1, #0x0008C000

c. BIC r6, r2, r0

d. ADD r7, r1, r2, LSL #4

b. r1: 0x0D796B7C

= 0b 0000 1101 0111 1001 0110 1011 0111 1100

EOR 0x 0008C000

= 0b 0000 0000 0000 1000 1100 0000 0000 0000

⇒ 0b 0000 1101 0111 1001 0110 1011 0111 1100

⇒ 0x 0D796B7C

c r2 0x79C286A4

= 0b 0111 1001 1100 0010 1000 0110 1010 0100

r0: 0xFB32C76D

= 0b 1111 1011 0011 0010 1100 0111 0110 1101

⇒ 0b 0000 0100 1100 1101 0111 1000 1001 0010

⇒ 0b 0000 0000 1100 0000 0000 0000 0000 0000

= 0x 00C00000

r2 lsl #4. 0x79C286A40

```
  0x 0D796B7C
+ 0x 79C286A40
-----------
  0x 87918D5BC
```


```
r0 : 0xFB32C76D
r1 : 0x0D796B7C
r2 : 0x79C286A4
```

What values are held in registers r4, r5 executed?

a. SUB r4, r0, #0x00000078

b. EOR r5, r1, #0x0008C000

c. BIC r6, r2, r0

d. ADD r7, r1, r2, LSL #4

Question C (30 Marks)

1. Write an ARM assembly language program to zip two 8-bit unsigned numbers stored in R0 and R1 and store the result into R3. Zipping means:
   i. Place the 0th bit (the most significant bit) of R0 as 0th bit of R3
   ii. Then place 0th bit of R1 as 1st bit of R3
   iii. Then place 1st bit of R0 as 2nd bit of R3
   iv. Then place 1st bit of R1 as 3rd bit of R3
   v. …

R3 16 bits total

$R0_0$	$R1_0$	$R0_1$	$R1_1$	$R0_2$	$R1_2$	$R0_3$	$R1_3$	…	$R0_8$	$R1_8$

Example 1:

R0 = 0b 1 0 1 0 1 1 1 1

R1 = 0b **1 1 1 1 0 0 0 0**

R3 = 0b 1 **1** 0 **1** 1 **1** 0 **1** 1 **0** 1 **0** 1 **0** 1 **0**

Example 2:

R0 = 0b 0 0 0 0 0 0 0 0

R1 = 0b **1 1 1 1 1 1 1 1**

R3 = 0b 0 **1** 0 **1** 0 **1** 0 **1** 0 **1** 0 **1** 0 **1** 0

Question D (30 marks)

Given the following piece of ARM code and also assume a three-stage pipeline:

```
        mov     r1, #0
        mov     r2, #2
L1      add     r1, r1, #1
        add     r3, r3, r1, LSL r2
        cmp     r1, r2
        blt     L1
        mov     r8, r3
```

Note the following instruction cycle times:

Processing Type	Cycles
Any unexecuted (condition code fails)	S
Normal Data Processing	S
Data Processing with register specified shift	I + S
Data Processing with PC written	N + 2S
Data Processing with register specified shift and PC written	I + N + 2S
LDR	N + I + S
LDR into PC	N + I + N + 2S
STR	N + N

Draw the pipeline diagrams of this ARM program. Please clearly indicate the status of each clock cycle and the number of cycles needed for the whole process. (30 marks)

END OF EXAM PAPER