TECHNOLOGICAL UNIVERSITY DUBLIN

Grangegorman

_____

TU856 – BSc. (Honours) in Computer Science

Year 1

_____

SEMESTER 2 EXAMINATIONS 2023/24

_____

CMPU1019 Microprocessor Systems

Internal Examiner(s): Frank Duignan

External Examiner(s):

Instructions To Candidates: Answer 3 of the following 4 questions

Exam Duration: 2 hours

Special Instructions : Numbers with the prefix **0x** are in hexadecimal

Question 1:
(a) Using your calculator or otherwise, determine the 16 bit results of the following C-language calculations. Express your answer in hexadecimal.

(i) 0xef00 & (1 << 15)

[3 marks]

(ii) 0x28a8 | 0xd645

[3 marks]

(b) What value will X have after the following C-code fragment?
int8_t X; // X is a signed 8 bit number
X = -125;
X = X – 5;

[5 marks]

(c) A computer system is required to store unsigned numbers in the range 0 to 8191. What is the minimum number of bits required to store one of these numbers?

[4 marks]

(d) Comment on the roles played by each of the following in computer systems:
(i) The address bus

[2 marks]

(ii) The data bus

[2 marks]

(iii) The instruction pipeline

[2 marks]

(e) A partial ASCII character set is shown in Table Q1e. Numeric values shown are in decimal.

(i) A serial communications link which operates with even parity checking sends the character 'A'. Will the parity bit associated with this transmission be 1 or 0?

[3 marks]

(ii) What is the difference between synchronous and asynchronous serial communications?

[3 marks]

(iii) A serial link sends data a rate of 9600 bits per second. Assuming each character transmitted requires a total of 11 bits, how long will it take to transmit the message "HELLO WORLD"? Note: there is no need to transmit the null character at the end of the string. Express your answer in microseconds.

[3 marks]

0	48	:	58	A	65	K	75	U	85
1	49	;	59	B	66	L	76	V	86
2	50	<	60	C	67	M	77	W	87
3	51	=	61	D	68	N	78	X	88
4	52	>	62	E	69	O	79	Y	89
5	53	?	63	F	70	P	80	Z	90
6	54	@	64	G	71	Q	81		
7	55			H	72	R	82		
8	56			I	73	S	83		
9	57			J	74	T	84		

Table Q1e

(f) How does the storage of value 0x12345678 differ between little and big endian computer systems?

[3 marks]

Question 2.

(a) Listing Q2a contains the code to configure an Input/Output pin on the STM32L031 microcontroller.

    (i)   When would you use this function?

          [2 marks]

    (ii)   Port->MODER points at memory address 0x50000000, BitNumber = 4 and Mode = 1

         i.   What is the value of **Mode** after line A is executed?
         ii.   What is the value of **Port->MODER** after line B is executed?
         iii. What is the value of **Port->MODER** after line C is executed?

          [9 marks]

    (iii)   Explain how "Pull-up" resistors are commonly used to convert mechanical switch or button movements into electrical signals suitable for a microcontroller's digital inputs.

          [4 marks]

(b) Figure Q2a shows an STM32L031 connected to two LED's (Red and Blue) and a momentary push-button. The following pin configuration functions are available to you:

void pinMode(GPIO_TypeDef *Port, uint32_t BitNumber, uint32_t Mode);
// Possible values for Port are : GPIOA, GPIOB
// Possible values for Mode are 0: The bit in question will be an input
//   1: The bit in question will be an output

void enablePullUp(GPIO_TypeDef *Port, uint32_t BitNumber);

    (i)   How would you use these configuration functions to configure the input/output pins for use as shown in Figure Q2a

          [6 marks]

    (ii)   Write a function that will turn on the red LED without affecting other input/output pins.

          [4 marks]

    (iii)   Write a function that will turn off the blue LED without affecting other input/output pins.

          [4 marks]

    (iv)   Write a function that will return a 1 if the button is pressed, 0 otherwise.
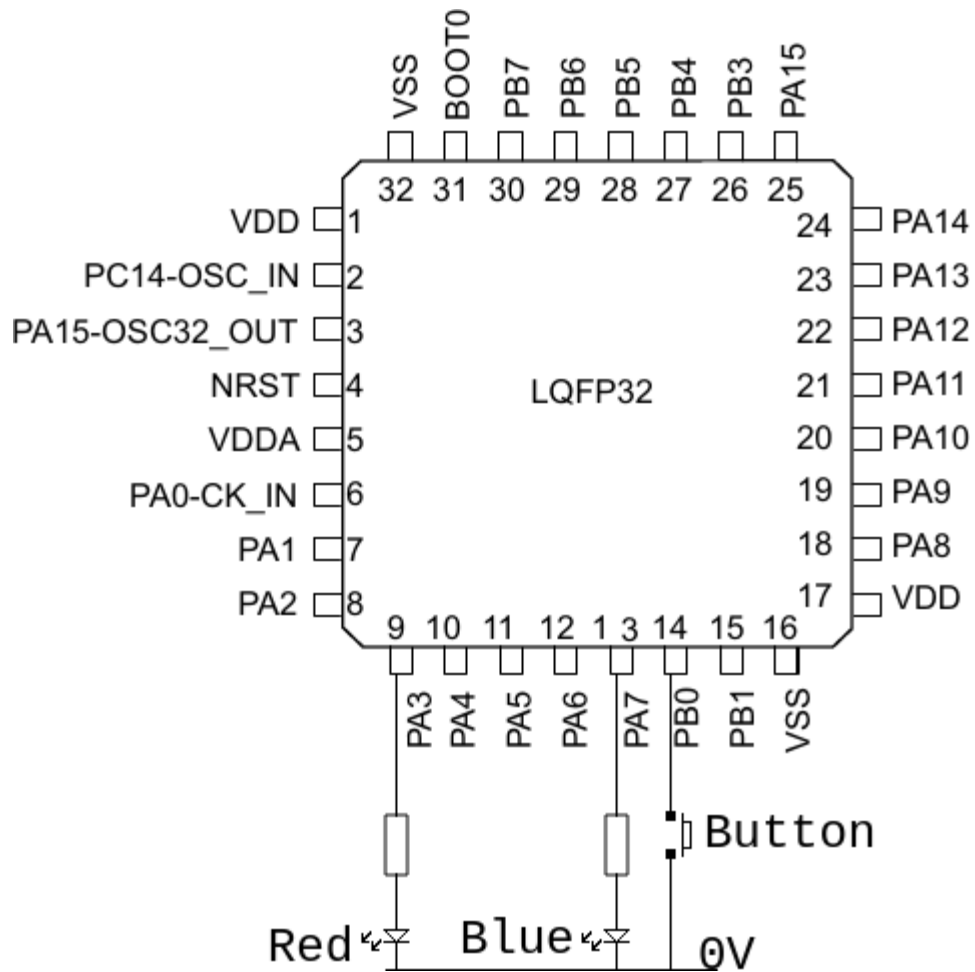
          [4 marks]

Figure Q2a

```
void pinMode(GPIO_TypeDef *Port, uint32_t BitNumber, uint32_t Mode)
{
uint32_t mode_value = Port->MODER;
Mode = Mode << (2 * BitNumber);  // Line A
mode_value = mode_value & ~(3u << (BitNumber * 2)); // Line B
mode_value = mode_value | Mode; // Line C
Port->MODER = mode_value;
}
```

Listing Q2a

Question 3:
(a) The contents of some of the STM32L031 core registers are as shown below. Also shown are the contents of some memory locations. What number goes where when each of the following instructions is executed one after another in a program?

(i) PUSH R1

[3 marks]

(ii) PUSH R0

[3 marks]

(iii) POP R2

[3 marks]

Contents of Registers

Register	Contents
R0	0x00000006
R1	0x00000007
R2	0x00000008
SP	0x20000ffc

Contents of RAM

Address	Contents
0x20000fec	0x00000001
0x20000ff0	0x00000002
0x20000ff4	0x00000003
0x20000ff8	0x00000004
0x20000ffc	0x00000005

(b) What happens when the following assembly language instructions are executed?
```
MOVS R1,#0
LDR R0,=0x40000000
STR R1,[R0, #0x10]
```

[5 marks]

(c) Listing Q3a shows an ARM Thumb assembler listing for a function that copies one block of memory to another

(i) What happens when Lines A, B and C are executed?

[9 marks]

(ii) Show how you would call this function in assembly language. The function should copy the 0x20 bytes starting at memory location 0x1000 to memory address 0x2000.

[5 marks]

(iii) What is meant by the Arm Architecture Procedure Call Standard?

[5 marks]

```
AREA THUMB, CODE, READONLY
EXPORT my_memcpy
; void my_memcpy(uint8_t *dest, uint8_t * src, uint32_t n);
; on entry, R0 = dest, R1 = src, R2 = n
my_memcpy
my_memcpy_loop
CMP R2,#0
BEQ my_memcpy_exit ;----    LINE A
LDRB R3,[R1]
STRB R3,[R0] ;----------    LINE B
SUBS R2,R2,#1
ADDS R0,R0,#1
ADDS R1,R1,#1
B my_memcpy_loop
my_memcpy_exit
BX LR ;-----------------    LINE C

end
```

Listing Q3a

Question 4.
(a) What ARM Cortex M0 Arithmetic flags are set by the following instructions sequences
   (i)   EORS R0,R0

                                                                    [3 marks]
   (ii) LDR R0,=0xfffffffe
         ADDS R0,R0,#2

                                                                    [3 marks]
   (iii) MOVS R0,#4
         SUBS R0,#8

                                                                    [3 marks]
   (iv) LDR R0,=0x80000000
         SUBS R0,#2

                                                                    [3 marks]
(b) What is meant by each of the following terms:
   (i)   Hardware Interrupt Request

                                                                    [2 marks]
   (ii) Interrupt Vector Table

                                                                    [2 marks]
   (iii) Interrupt Handler

                                                                    [2 marks]
(c) Outline the sequence of events that occurs when an STM32L031 receives and processes (handles) an interrupt request

                                                                    [4 marks]

(d) Listing Q4a shows a partial listing of an STM32L031 assembly language program.
   (i)   What happens during the execution of this program?

                                                                    [3 marks]
   (ii) Complete the function zero_fill such that it fills the area of memory pointer to by the first parameter with zeros. The size of the buffer is specified in the second parameter.

                                                                    [8 marks]

```
    .equ   SIZE,200
    .data
// put variables here
buffer: .word 0

    .text
    .syntax unified
    .cpu cortex-m0plus
    .global   main
main:
    ldr R0,=SIZE
    bl malloc
    ldr R1,=buffer
    str R0,[R1]

    ldr R0,=buffer
    ldr R1,=SIZE
    bl zero_fill

    ldr R0,=buffer
    ldr R0,[R0]
    bl free
exit:
    B .
// C-signature: void zero_fill(uint8_t *buffer,uint32_t len);
zero_fill:
    push {LR}
    // insert code here
    pop {PC}
```

Listing Q4a