13.01.20

14.00 - 16.00pm

CMPU 2013 Microprocessors

Basement 1, Kevin Street

Programme Code: DT211C
Module Code: CMPU 2013
CRN: 22510

TECHNOLOGICAL UNIVERSITY DUBLIN

KEVIN STREET CAMPUS

BSc. (Honours) Degree in Computer Science (Infrastructure)

Year 2

SEMESTER 1 EXAMINATIONS 2019/20

Microprocessors

Mr. Frank Duignan
Dr. Deirdre Lillis
Ms. Pauline Martin

Instructions to Candidates
Attempt 3 out of the following 4 questions.
Numbers prefixed by *0x* are in hexadecimal.

Question 1

(a) What will be printed when the following lines of C code are executed on a PC:

(i)

```
uint16_t X = 0xffff;
uint16_t Y = 0x0531;
printf("%x\n",X & ~Y);
```

[3 marks]

(ii)

```
uint16_t X = 0x9084;
uint16_t Y = 0x414a;
printf("%x\n",X | Y);
```

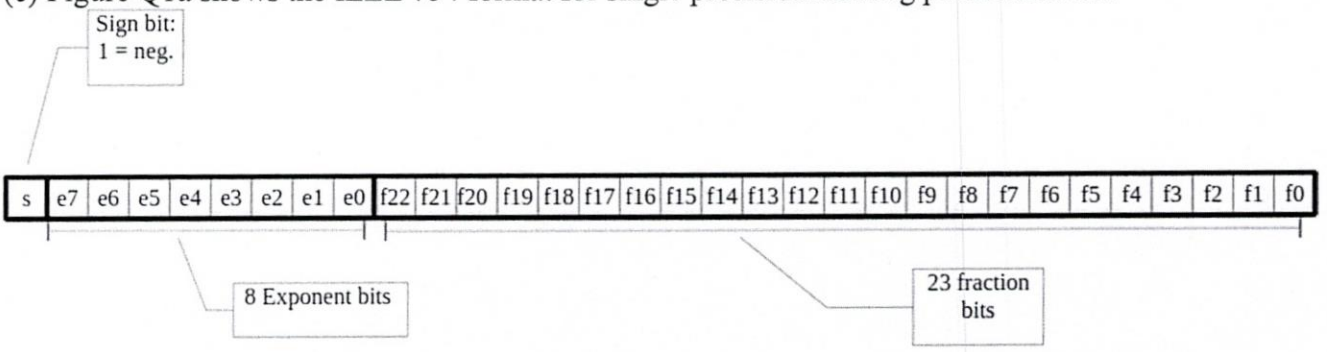
[3 marks]

(b) What will the value of X be after the following C code is executed?

```
int16_t X = -32765;
X = X - 4;
```

[4 marks]

(c) Figure Q1a shows the IEEE 754 format for single precision floating point numbers.



$$Value = (-1)^s \, x \, 2^{(e-127)} \, x \left(1 + \frac{f22}{2} + \frac{f21}{4} + \frac{f20}{8} + ... \frac{f0}{2^{23}}\right)$$

Show how the value +8.75 would be stored in this format.

[6 marks]

(d) A partial ASCII table (in decimal) is provided below (e.g. 'K' = 75):

A : 65	B : 66	C : 67	D : 68
E : 69	F : 70	G : 71	H : 72
I : 73	J : 74	K : 75	L : 76
M : 77	N : 78	O : 79	P : 80
Q : 81	R : 82	S : 83	T : 84
U : 85	V : 86	W : 87	X : 88
Y : 89	Z : 90		

Using this table, show how the C-String "WORLD" is stored in memory.

[4 marks]

(e) A microprocessor's clock speed is 50MHz. How many nanoseconds does each clock cycle last?

[3 marks]

(f) An RS232 serial communications link operates at a baud rate of 9600 bps and uses **even** parity checking.

(i) Referring back to the ASCII table in Q1.d, what will the parity bit be when the character 'R' is transmitted?

[3 marks]

(ii) Assuming 1 start bit, 1 stop bit and 1 parity bit (i.e. a 3 bit overhead per byte) what is shortest possible time it will take to send a data block of 20480 bytes?

[3 marks]

(g)

Arrange the following storage spaces in ascending order of speed:

RAM, Disk, Register, L1-Cache

[4 marks]

Question 2

(a) "General Purpose Input/Output (GPIO) ports on the LPC1114 are like memory locations that are also connected to the real world". Comment on this statement and *outline* the steps you would take to control an LED using a simple C program.

[4 marks]

(b) Figure Q2a shows a pinout of the LPC1114. Write a line of C-code to do each of the following:

(i) Configure pin 28 as an output without altering the configuration of any other pin.

[4 marks]

(ii) Configure pin 14 as an input without altering the configuration of any other pin.

[4 marks]

(iii) Set the state of pin 28 to HIGH

[4 marks]

(iv) Wait for pin 14 to go LOW

[4 marks]

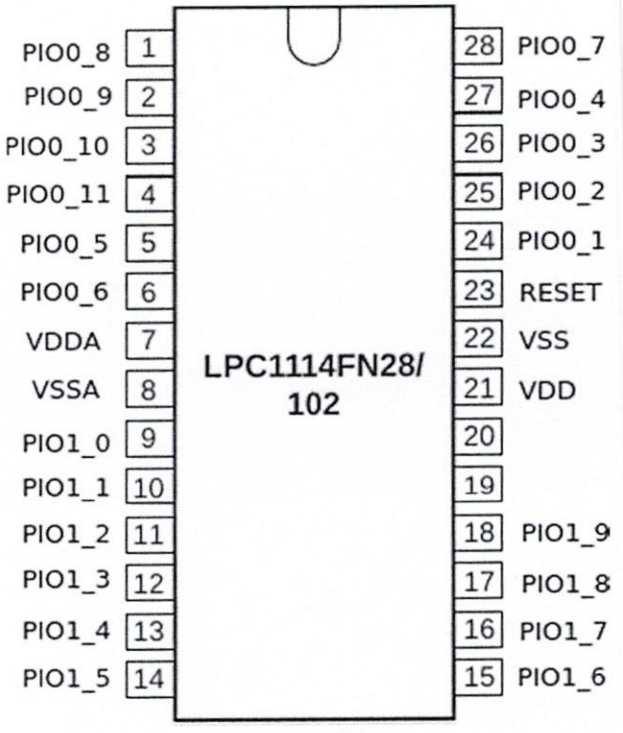


Figure Q2a

(b) Figure Q2b depicts the Systick timer system inside ARM Cortex microcontrollers. The "Down-Counter" counts down from the Reload Register value and emits a pulse when it reaches zero. This pulse triggers a reload of the counter and sends an interrupt request to the CPU.

(i) What number would you place in the Reload Register to generate an interrupt request every millisecond?

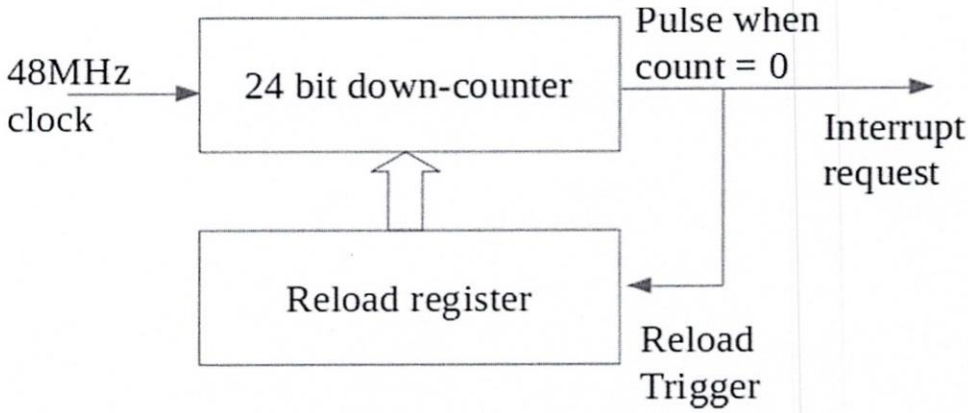
[4 marks]

(ii) How does the CPU find the code that should be executed in response to each interrupt request.

[6 marks]

(iii) Cite two uses for a periodic interrupt such as this.

[3 marks]



Question 3.

(a) The ALU in most microprocessors outputs the following four signals when certain calculation results arise: N, Z, V and C

What are these signals and how are they used for decision making in programs?

[8 marks]

(b) Listing Q3a shows how the function "memset" may be implemented and called.

(i) Identify and state the function of 2 different assembler directives in the listing.

[4 marks]

(ii) Lines A and B contain use different versions of the "Branch" instruction : B and BL. How do these behave differently?

[4 marks]

(iii) The memset function has an error. If the caller passes a value of 0 for the number of bytes to copy it behaves overwrites all of RAM. Why is this and how would you fix this problem?

[10 marks]

(iv) Identify a line of code in the Listing Q3a that uses the following addressing modes:

(a) Immediate addressing

[2 marks]

(b) Register indirect

[2 marks]

(c) PC Relative addressing

[3 marks]

```
    AREA DATA
Buffer SPACE 10
    AREA THUMB,CODE,READONLY
; EXPORTED Symbols can be linked against
    EXPORT Reset_Handler
    EXPORT __Vectors
; Minimal interrupt vector table follows
; First entry is initial stack pointer (end of stack)
; second entry is the address of the reset handler
__Vectors
    DCD 0x10001000
    DCD Reset_Handler

; 'Main' program goes here
Reset_Handler
    LDR R0,=Buffer    ; point at the target memory
    MOVS R1,#0xff     ; fill with 0xff
    MOVS R2,#10       ; buffer is 10 bytes long
    BL memset         ; call on memset    LINE A
stop
    B stop            ; loop here when done    LINEB

memset
; Fills buffer pointed to by R0 with the byte in R1.
; R2 contains the buffer size
    PUSH {LR}
memset_loop
    STRB R1,[R0]
    ADDS R0,R0,#1
    SUBS R2,R2,#1
    BNE memset_loop
    POP {PC}
        end
```

Listing Q3a

Question 4

(a) Listing Q4a shows how a poorly written function can crash a program when is called with the "right" parameters.

(i) Why does the program crash?

[7 marks]

(ii) How can it be fixed?

Your answer should include a detailed explanation of the behaviour of the program stack during execution.

[6 marks]

(b) Many microprocessors make use of instruction pipelining to speed up the execution of programs. The ARM Cortex M0 CPU has a 3-stage pipeline. What happens at each stage?

[6 marks]

```
#include <stdio.h>
#include <string.h>
void Bad_Function(char *data);
int main()
{
    Bad_Function("Please crash now");
}
void Bad_Function(char *data)
{
    char LocalBuffer[5];
    strcpy(LocalBuffer,data);
    printf("%s\n",LocalBuffer);
}
```

Listing Q4a

(c)

(i) What is meant by the term "Calling Convention". Illustrate your answer with a simple example.

[8 marks]

(ii) The ARM Cortex M0 CPU core uses the Link Register (LR) to remember return addresses during function calls. There is only one Link Register in the CPU however. How are nested function calls (more than one level deep) managed?

[6 marks]