

# THE UNIVERSITY OF NEW SOUTH WALES

## COMP2121 MICROPROCESSORS AND INTERFACING Sample Final Examination

- This sample paper is an indicative of what you can expect in the final exam. However, it is by no means comprehensive. The structure of the final exam is also different to this sample exam.
- Time Allowed: 3 Hours
- AVR Instruction Set Sheet and ASCII Table are provided

### 1. Basics Concepts ( $12 \times 3 = 36$ marks)

- PART 1** Describe the difference(s) between microprocessors and microcontrollers. What is ATmega64?
- PART 2** Explain the concept of endianness. Which endianness does AVR use? Please give an example.
- PART 3** Function can be recursive. Can macro be recursive? Why?
- PART 4** Describe differences between memory-mapped I/O and separate I/O.
- PART 5** In ADC, what is resolution? And what is accuracy? What are the differences between these two terms? Please give an example.
- PART 6** What is stack frame? Draw a memory map to show the basic structure of a stack frame. Please list instructions that can access stack in AVR.
- PART 7** What is watchdog timer? What should be done to set up watchdog timer before to use it?

**PART 8** What is aliasing in ADC? How to avoid aliasing?

**PART 9** What does USART stand for? In asynchronous serial communication, how is receiver clock is synchronized to a transmission operation of the transmitter?

**PART 10** What is interrupt vector table? How many interrupts are available in Atmega64? The following is the part of interrupt vector table in an AVR program. Is this part of vector table correctly set? Why?

```
; interrupt vector table
.org 0x0000
rjmp RESET
rjmp INT0
rjmp INT1
```

**PART 11** There is no software interrupt available in AVR. How can you implement a software interrupt in AVR?

**PART 12** The keypad is a typical input device in microcontroller application. Write a high level description (algorithm) that specifies how the input data from keypad is obtained by the microcontroller.

## 2. Miscellaneous Questions (31 marks)

### PART 1

NOT APPLICABLE

**PART 2** Consider the content of the AVR program memory in hex format as shown below.

- (a) What sequence of 4 ASCII characters does the content correspond to?
  - (b) What 2-byte signed integer values in base 10 does the content correspond to?
  - (c) What 2-byte unsigned integer values in base 10 does the content correspond to?
- (6 marks)

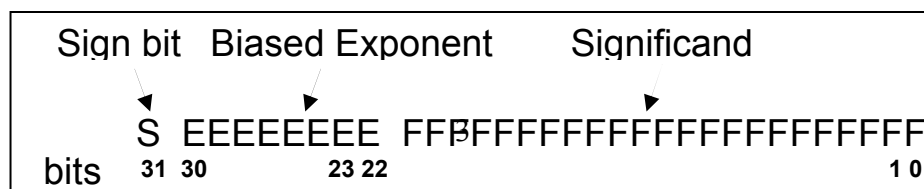
3C	4E	53	5A
----	----	----	----

**PART 3** Consider the following AVR assembly code:

```
.MACRO delay
loop: subi @0, 1
      sbci @1, 0
      nop
      nop
      nop
      nop
      brne loop      ; taken branch takes two cycles.
.ENDMACRO
```

All instructions in the program are 2 bytes long. What is the size of the code in bytes? How many parameters does the macro have? What is the range of values of each parameter? The code can generate a delay. What is the range of the delay? Assume the processor frequency is 8 Mhz. (7 marks)

**PART 4** Consider two single precision floating point numbers x and y in IEEE 754 format, where x= 0x50240000 and y=0x40080000. The IEEE 754 format is given as follows:



What is the decimal value of  $x+y$ ? Please show your work. **(4 marks)**

**PART 5** How many bits do you need to represent a~z 26 letters and 0~9 ten digits? Can you encode them with the 4x4 keypad ? If no, why? If yes, how?  
**(5 marks).**

### **3. AVR Assembly Programming and Design (33 marks)**

**All programs must be well commented.**

**PART 1** Write an AVR assembly program to find the max value of an integer array, A. Your program must satisfy the following requirements.

- 1) Each element is a 2-byte signed integer.
- 2) Array A is stored contiguously in the FLASH.
- 3) Your program must define and use at least one MACRO.
- 4) The array length is 10.

**(7 marks)**

**PART 2** An array of ten 2-byte integers are stored in the AVR program memory. Write a program to convert the array to different endianness format and store it back to the data memory.

Your program must satisfy the following requirements:

1. Your program must use at least one function.
2. All local variables and parameters must be stored in the stack space.
3. You must describe the stack frame structures for the function used in your program. **(8 marks)**

### **PART 3**

Consider to design an embedded system to control the speed of a DC motor.

The operating specification of the system is given below:

1. The speed of the motor is input from the keypad
  - i. Assume that the motor is driven by a PWM signal. And there is a formula to determine the duty cycle for different motor speed when the motor spins without any extra load.
2. The motor is started by an external push button operation
3. The speed deviation must be controlled within a specified limit.
4. If the speed deviation exceeds the limit, the LED bar is set to ON within 1 second to alarm the user and at the same time the motor is stopped.

Your design should include:

1. definition of all tasks in your system.
2. task scheduling diagram that specifies the execution timing frame for each task
3. interrupt design that includes what kind of interrupts you are going to use and the purpose for using each of the interrupts
4. a code template that includes interrupt handling subroutines

Note, the code template has not to be complete. You can use comment lines for sections of code that involve detail setting information and can be inserted later. Examples are given below.

```
; insert code here to set up timer0 for 1 second timer out  
; insert code here to enable timer0 overflow interrupt
```

**(18 marks)**

