



Philadelphia University
Faculty of Engineering

Marking Scheme

Examination Paper

Department of CE

Module: Microprocessors (630313)

First Exam

Second Semester

Date: 02/04/2019

Section 1

Weighting 20% of the module total

Lecturer:

Dr. Qadri Hamarsheh

Coordinator:

Dr. Qadri Hamarsheh

Internal Examiner:

Dr. Naser Halasa

Marking Scheme Microprocessors (630313)

The presented exam questions are organized to overcome course material, the exam contains 5 questions; *all questions* are compulsory requested to be answered. Thus, the student is permitted to answer any question out of the existing ones in this section.

Marking Assignments

The following scheme shows the marks assignments for each question. They show also the steps for which a student can get marks along the related procedure he/she achieves.

Question 1 This question is attributed with 5 marks, if answered properly, The answer for this question as the following:

- 1) During which phase of the instruction execution cycle is the program counter (**IP register**) incremented?
 - a) **decode**
 - b) **execute**
 - c) **operand fetch**
 - d) **fetch**
- 2) The size of each segment in **8086** is
 - a) **16kb**
 - b) **24 kb**
 - c) **50 kb**
 - d) **64 kb**
- 3) The **Overflow flag** is based on signed arithmetic.
 - a) **True**
 - b) **False**
- 4) Which of these values is the most acceptable for the data type **REAL4**?
 - a) **1275**
 - b) **'Q'**
 - c) **23.26**
 - d) **"Jennifer",0**
- 5) One of the following **memory models** combines the data and code parts:
 - a) **Small**
 - b) **Tiny**
 - c) **Meduim**
 - d) **Huge**

Question 2 This question is attributed with 3 marks if answered properly, The answer for this question as the following:

Solution

enerate 16-bit MS-DOS Programs

- ✓ **Advantages**
 - **enables calling of MS-DOS and BIOS functions**
 - **no memory access restrictions**
- ✓ **Disadvantages**
 - **must be aware of both segments and offsets**
 - **cannot call Win32 functions (Windows 95 onward)**
 - **limited to 640K program memory**
- ✓ **Requirements**
 - **INCLUDE Irvine16.inc**
 - **Two additional instructions are inserted at the beginning of the startup procedure (main)**

Initialize DS to the data segment using predefined MASM constant @data::

```
mov ax,@data  
mov ds,ax
```

Question 3 This question is attributed with 5 marks, if answered properly. The answer for this question as the following:

a)

Ñ	Instruction	Answer
1)	mov word1, byte2	I
2)	mov word2, 10000h	I
3)	mov si, ds	L
4)	movzx ax, byte1	L
5)	movsx dl, al	I

b)

mov dx, word3	DX = 7FFFh
movsx eax, byte1	EAX = FFFFFFFFh
mov dh, al	DH = FFh
mov ax, [word3+2]	AX = 8000h
mov eax, [word3+4]	EAX = I (illegal)

Question 4 This question is attributed with 3 marks, if answered properly. The answer for this question as the following:

Original Program	Your answer
<pre>INCLUDE irvine32.inc .data arrayA word 4Ah, 3Ch, 2h, 5 DUP (1h) var1 word ffffh .code main PROC mov cx, 4Ch xchg cx, arrayA+4 mov ax, arrayA+5 mov bl, var1 add ax, bx call dumpregs exit main ENDP</pre>	<pre>INCLUDE irvine32.inc .data arrayA word 4Ah, 3Ch, 2h, 5 DUP (1h) var1 word 0ffffh .code main PROC mov cx, 4Ch xchg cx, arrayA+4 mov ax, arrayA+5 mov bx, var1 add ax, bx call dumpregs exit main ENDP end main</pre>

Question 5 This question is attributed with 4 marks, if answered properly. The complete code for this question as the following:

Solution	
<pre>Title Simple Arithmetic .data J sword ? K sword ? L sword ? U1 word ? U2 word ? U3 word ? .code mov J, 3 mov K, -2 mov U1, 254 mov U2, 22 mov ax, J add ax, K mov L, ax mov ax, u1 add ax, u2 mov u3, ax main PROC main ENDP END main</pre>	<p>(1 mark)</p> <p>(1 mark)</p> <p>(2 marks)</p>