



---

# ELE538 Final Exam Study Guide

Prof. Ken Clowes

Prof. Vadim Geurkov

6 December 2004

Revision History

6 December 2004

Partial draft public distribution.

Revision 0.9

## Table of Contents

What is a Study Guide? .....	1
Questions .....	2
Reference Material .....	5
A/D system .....	5
Timer system .....	6

## What is a Study Guide?

This study guide is made up of questions relating to many of the main topics that will be covered in the Final. The Final includes all the material covered in the lectures and all of the labs.

The study guide is **not exhaustive** (i.e. not all topics you are responsible for are covered.)

You **cannot assume** that Final questions will bear any resemblance to the questions in this Study Guide. Memorizing canned answers is not a good idea; use the questions/answers to help identify areas you need to bone up on.

You should attempt to answer the questions **before** peeking at the answers. If you cannot understand an answer after reviewing the core material, ask you prof for help.

The questions should be considered as “closed book”—i.e. you should be able to answer them without reference to a text book or data sheet. However, some of the questions do require technical details that you are not expected to have memorized. (For example, what is the machine language translation of `ldaa #3` or which bit is used to tell the A/D subsystem to perform continuous conversions?) For these cases, a *Reference Material* section is included.

### Caution

Although the Final covers the entire course, this Study Guide concentrates on the topics not covered in the Quiz and Midterm Study Guides. You should also review those guides in preparation for the Final.

# Questions

## 1. Timer

Q: Timer TOF (v1).

Given the code:

```
FOO equ 128
    ldaa #FOO
    staa TMSK2
```

Which of the following is true?

1. No interrupts are enabled.
2. Only the TOF interrupt is enabled.
3. The TOF interrupt is enabled; others **may** also be enabled.
4. The TOF interrupt is enabled; others **are definitely** also enabled.
5. The TOF interrupt is disabled but one or more other interrupts are enabled.
6. Impossible to determine.

A: The TOF interrupt is enabled.

Q: Timer TOF (v2).

Given the code:

```
FOO equ $128
    ldaa #FOO
    staa TMSK2
```

Which of the following is true?

1. No interrupts are enabled.
2. Only the TOF interrupt is enabled.
3. The TOF interrupt is enabled; others **may** also be enabled.
4. The TOF interrupt is enabled; others **are definitely** also enabled.
5. The TOF interrupt is disabled but one or more other interrupts are enabled.
6. Impossible to determine.

A: The TOF interrupt is disabled but one or more other interrupts are enabled.

Q: Timer TOF (v3).

Given the code:

```
FOO equ $80
    ldaa FOO
    staa TMSK2
```

Which of the following is true?

1. No interrupts are enabled.
2. Only the TOF interrupt is enabled.
3. The TOF interrupt is enabled; others **may** also be enabled.
4. The TOF interrupt is enabled; others **are definitely** also enabled.
5. The TOF interrupt is disabled but one or more other interrupts are enabled.
6. Impossible to determine.

A: Impossible to determine.

Q: Timer TOF (v4).

Given the code:

```
FOO equ $80
    ldaa #FOO
    oraa TMSK2
    staa TMSK2
```

Which of the following is true?

1. No interrupts are enabled.
2. Only the TOF interrupt is enabled.
3. The TOF interrupt is enabled; others **may** be enabled.
4. The TOF interrupt is enabled; others **are definitely** enabled.
5. The TOF interrupt is disabled but one or more other interrupts are enabled.
6. Impossible to determine.

A: The TOF interrupt is enabled; others **may** be enabled.

Q: Using timer Output Compare.

Consider the following code:

```

    org $6000
    sei
    jsr init
    cli
main
    bra main

init:
    ldaa #%01000000
    staa TFLG1
    staa TMSK1
    staa TCTL1
    ldx #ISR
    stx VECTORforThis
    rts

ISR:
    ldaa #%01000000
    staa TFLG1
    ldd $1012
    addd #500
    rti

```

## 2. Bit Operations

Q: Using rotate/and/bmi/beq.

AccA is initialized as %0110x10y (note: the bit values 'x' or 'y' are unspecified.)

```

    rora
    rora
    bmi foo
    ldaa #1
    swi
foo rola
    anda #$41
    beq bar
    ldaa #2
    swi
bar ldaa #3
    swi

```

Which of the following statements about the value in AccA when an swi instruction is executed are true? (Note: there may be more than one true statement.)

1. AccA is 1 if x is 0 and y is 1.
2. AccA is 1 if x is 1 and y is 0.

3. AccA is 2 if x is 0 and y is 1.
4. AccA is 2 if x is 0 and y is 0.
5. AccA is 2 if x is 1 and y is 1.
6. AccA is 3 if x is 0 and y is 1.
7. AccA is 3 if x is 1 and y is 1.

A: The true statements are:

- AccA is 1 if x is 0 and y is 1.
- AccA is 2 if x is 0 and y is 0.

Q: Using `rotate/and/bmi/beq`.

Consider the following code:

```

ldaa #$F1
ldab #$FF
addb #1
rola
inca
clc
rora
swi
    
```

What is AccA when the `swi` is executed?

1. \$72

A: \$72

## Reference Material

This material contains technical details that may be required to answer certain questions.

### A/D system

The bits in the Control/Status register (ADCTL, mapped to address 0x1030) are:

**Figure 1. AD Control/Status Register**

7	6	5	4	3	2	1	0
CCF	-	M*	S	0	n	n	n

The interpretation of the bits is:

CCF 0: conversion NOT complete; 1: conversion complete.

M\* 0: Convert 4 channels; 1: Convert single channel.

S 0: continuous conversion; 1: one-shot conversion.

nnn Channel number (0-7).

## Timer system

### Timer Free-running 16-bit Counter (TCNT@0x100E/0x100F)

This 16-bit free-running counter is incremented by the E clock. When the count overflows from 0xffff to 0x0000, the timer overflow flag bit (TOF) is set in TFLG2. An interrupt can be enabled by setting the interrupt enable bit in TMSK2.

### Timer Input Capture Registers(TICx)

The 3 16-bit input capture registers TIC1, TIC2 and TIC3 (mapped to 0x1010/0x1011, 0x1012/0x1013 and 0x1014/0x1015) latch TCNT's value on a defined transition of an input edge detector.

### Timer Output Compare Registers(TOCx)

The 5 16-bit output compare registers TOC1, TOC2, TOC3, TOC4 and TOC5 (mapped to 0x1016/0x1017, 0x1018/0x1019, 0x101A/0x101B, 0x101C/0x101D and 0x101E/0x101F) set the OCxF (flag) bit when equal to TCNT.

When equality occurs, an interrupt may occur if enabled and/or an output action on a PORTA bit may occur as specified (in TCTL1 for TOC2—TOC5).

### Timer Control Register 1 (TCTL1@0x1020)

The bits are interpreted as follows:

**Figure 2. TCTL1 Register**

7    6    5    4    3    2    1    0  
 OM2 OL2 OM3 OL3 OM4 OL4 OM5 OL5

Each 2-bit OMx/OLx pattern specifies an action:

OMx/OLx	Action
00	Do nothing.
01	Toggle OCx output.
10	Clear OCx output.
11	Set OCx output.

## Timer Control Register 2 (TCTL2@0x1021)

The bits are interpreted as follows:

**Figure 3. TCTL2 Register**

```

7       6       5       4       3       2       1       0
0       0       EDG1B  EDG1A  EDG2B  EDG2A  EDG3B  EDG3A
    
```

Each 2-bit EDGxB/EDGxA pattern specifies an action:

EDGxB/EDGxA	Action
00	Capture disabled.
01	Capture rising edges.
10	Capture falling edges.
11	Capture any edge.

## Timer Interrupt Mask Register 1 (TMSK1@0x1022)

TMSK1 is used to enable interrupts on any of the 5 Output Compare (OC) or 3 Input Capture (IC) registers. The bits are interpreted as follows:

**Figure 4. TMSK1 Register**

```

7       6       5       4       3       2       1       0
OC1I  OC2I  OC3I  OC4I  OC5I  IC1I  IC2I  IC3I
    
```

An OC/ICxI bit enables the corresponding interrupt when set.

## Timer Interrupt Flag Register 1 (TFLG1@0x1023)

TFLG1 is used to indicate events on any of the 5 OC or 3 IC registers. The bits are interpreted as follows:

**Figure 5. TFLG1 Register**

```

7       6       5       4       3       2       1       0
OC1F  OC2F  OC3F  OC4F  OC5F  IC1F  IC2F  IC3F
    
```

An OCxF flag bit is **set** by hardware when TCNT equals the corresponding TOCx register.

An OCxF flag bit is **cleared** by software by writing a ONE to the corresponding bit.

## Timer Interrupt Mask Register 2 (TMSK2@0x1024)

TMSK2 is used to enable interrupts on when a corresponding flag bit is set in TFLG2. The bits are interpreted as follows:

### Figure 6. TMSK2 Register

7	6	5	4	3	2	1	0
TOI	RTII	PAOVI	PAII	0	0	PR1	PR0

The interpretation of the bits is:

- TOI     0/1: Disable/enable Timer Overflow Interrupt.
- RTII    0/1: Disable/enable Real Time Interrupt.
- PAOVI   0/1: Disable/enable Pulse Accumulator Overflow interrupt.
- PAII    0/1: Disable/enable Pulse Accumulator Input interrupt.

### Timer Interrupt Flag Register 2 (TFLG2@0x1025)

TFLG1 is used to indicate timer events. The bits are interpreted as follows:

#### Figure 7. TFLG2 Register

7	6	5	4	3	2	1	0
TOF	RTIF	PAOVF	PAIF	0	0	PR1	PR0

The interpretation of the bits is:

- TOF     **Set** by hardware when TNCT advances from 0xFFFF to 0x0000.  
           **Cleared** by software when a ONE is written to the flag bit.
- RTIF    **Set** by hardware when rising edge on selected tap point.  
           **Cleared** by software when a ONE is written to the flag bit.
- PAOVF   **Set** by hardware when the pulse accumulator rolls over from 0xFF to 0x00.  
           **Cleared** by software when a ONE is written to the flag bit.
- PAIF    **Set** by hardware when an active edge is detected on the PAI input pin.  
           **Cleared** by software when a ONE is written to the flag bit.