EEL 3744—Spring 2018
4 April 2018

Dr. Eric M. Schwartz
11-Jun-18 5:46 PM

Exam 2

Relax! | ***Go Gators!*** | ***Good luck!***

_______________________, _______________
Last Name , First Name

UF's NaviGator AMS and team.

Instructions:

- *Turn off all* ***cell phones*** *and other* ***noise making devices*** *and put away* ***all electronics***.
- *Show* ***all*** *work on the* ***front*** *of the test papers. If you need more room, make a clearly indicated note on the front of the page, "MORE ON BACK", and use the back. The back of the page will* ***not*** *be graded without an indication on the front.*
- *You may use any of your XMEGA documents with* ***limited*** *added material; highlighting and tagging is permissible. You may* ***not*** *use any notes (mine or yours), examples, homework, labs, books, calculators, computer, electronic devices, etc.*
- ***CLEARLY*** *write your name at the top of* ***this*** *test page (and, if you remove the staple, all others). Be sure your exam consists of* ***13*** *distinct pages. Sign your name and add the date below. (If we struggle to read your name, you will lose points.)*
- *The space provided does* ***not*** *necessarily represent the amount of writing necessary.*
- *For anything undefined, if necessary, state a reasonable assumption.*
- *You must pledge and sign this page in order for a grade to be assigned.*
- *In programs, the use of comments results in* ***more*** *partial credit.*
- ***Read*** *each question* ***carefully*** *and* ***follow the instructions***.
- *The point values for problems may be changed at prof's discretion.*
- *Part of your grade on tests, quizzes, labs, etc. is based not only on solving the problem you are presented with, but the manner in which you solve it. For example, there is a difference between two programs that meet the given specifications, but one is an elegant, extensible 20-line solution, while the other is an obfuscated 100-line program that also meets the specifications but would be difficult to extend later. Just as your future employer would value the latter program less than the first, so will I in grading your assignments.*
- *When asked to provide a numerical answer, provide a single number only, e.g., 37.9,* ***NOT*** *expressions like* $3^7\times\sqrt[4]{37}$*, or fractions, like 37/42. Provide the proper number of significant figures.*
- *This exam counts for* ***22****% of your total grade.*
- *Unless otherwise stated assume the following:*
  - *The oscillator frequency is precisely 32 MHz.*
  - *The code should run on an* ***ATxmega128A1U*** *as configured on the* Out of the Box uPAD, uPAD Memory Base, and selected uPAD Backpacks ***without*** *any additional peripherals.*
  - *You can assume the standard bit equates that I have used in class examples (e.g.,* BIT0 = 0b0000 0001, BIT76 = 0b1100 0000b, INV76 = 0b0011 1111*) have already been done for you.*

May the Schwartz be with you!

PLEDGE:

On my honor as a University of Florida student, I certify that I have neither given nor received any aid on this examination, nor I have seen anyone else do so.

PRINT YOUR NAME | SIGN YOUR NAME | DATE (*4 Apr 2018*)

Regrade comments below. Give **page #** & **problem #** and reason for the petition.

Pages	Available	Points
2-4	30	
5	10	
6-9	25	
10-13	35	
TOTAL	100	

Exam 2

_______________________, _______________
Last Name , First Name

[30%] 1. Design an expansion to a XMEGA board (like your µPAD) by adding a **ROM**, an **SRAM**, an **input** port, and three additional components for another problem on this exam, used to control a milk-dispensing robot. The three components are a buzzer, controlled by a single XMEGA output pin, a text message generator driven by a XMEGA asynchronous serial transmission system, and a push-button switch circuit to trigger an external interrupt. Assume that only **a single chip select**, **CS0**, is available for this problem. Complete the figure to the right and the table below for each of parts a-c.

4 min

(1%) a) Add an **8 KB ROM** at the starting address of **0x8000**. Add to the figure on the right and to the table at the bottom of the page.

2 min

Device/Memory Blocks

(1%) b) Add a **single input port**, immediately following the ROM. Add to the figure on the right and to the table at the bottom of the page.

2 min

(1%) c) Add a **4 KB SRAM** immediately following the input port. Add to the figure on the right and to the table at the bottom of the page.

(4%) d) Specify the required **CS0** related-values below for parts a-c. CS0 should minimize the requirements for additional circuitry. The CS0 should **NOT** overlap with any address beyond what is necessary.

5 min

EBI_CTRL =
CS**0**_BASEADDRH =
CS**0**_BASEADDRL =
CS**0**_CTRLA =

8 K (8 k × 8) ROM Address Range:
0x__________ - 0x__________ = 0b______________________________ - 0b______________________________

Input Port Address Range:
0x__________ - 0x__________ = 0b______________________________ - 0b______________________________

4 KB (4 K × 8) SRAM Address Range:
0x__________ - 0x__________ = 0b______________________________ - 0b______________________________

Exam 2

_______________________, _______________
Last Name , First Name

(2%) 3 min

1. e) Derive the **equations for the address part of address decoding**, i.e., X_{Device} = f(Addresses only). These equations should include **ALL** of the addresses (and only addresses). You will design the necessary circuitry in part i.

(1%) 3 min

f) Use USARTF0 on PORTF to connect to the external UART text (SMS) message transceiver (transmitter plus receiver), as shown on the next page. Which PORTF pin will you select to utilize your UART **serial transmitter**? In the context of this problem, your UART **only** needs to transmit a text string to the text (SMS) message transceiver. Be sure to show the **necessary** connections to the text (SMS) message device on the next page.

(1%) 1 min

g) Choose a port on your XMEGA to interface with the **buzzer circuit**. Use PORTF, if possible, for these requirements. Which port pin will you use? Show any necessary connections on the following page.

(3%) 3 min

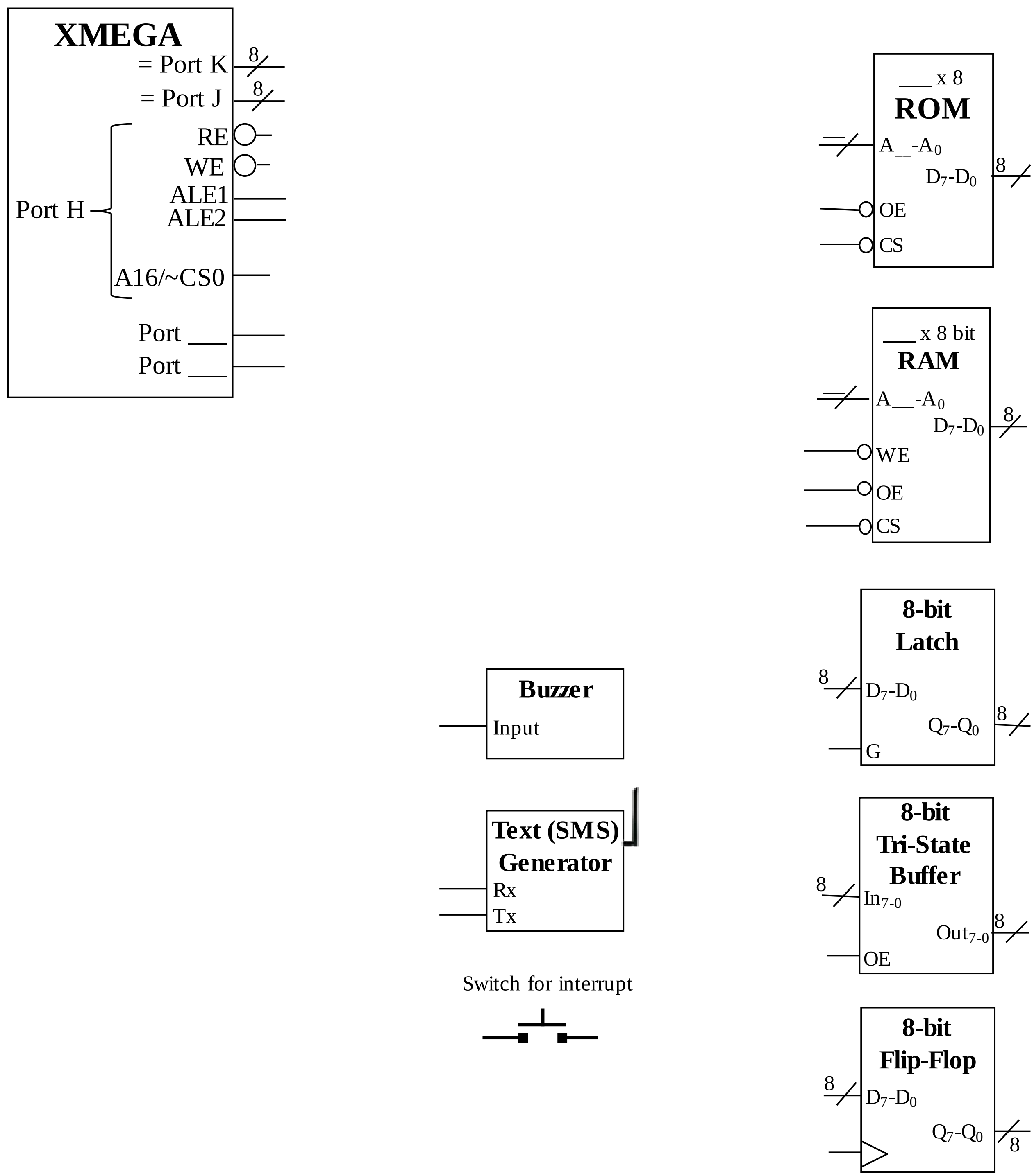
h) Select a port pin on your XMEGA for an external interrupt. This pin will connect to the push button shown on the next page, and a low-to-high logic level transition on the pin will be used to trigger an interrupt. Choose a pin on PORTF, if possible. Which pin will you use? Design the switch circuit on the next page (using the switch shown) so that the output of the switch circuit goes high when pressed. Show any necessary connections on the following page. (On problem 3b, you will write assembly language code to initialize this pin as specified here.)

Exam 2

________________________, ________________
Last Name , First Name

(16%) 1. i) Complete the circuit diagram below, as specified previously. **Please USE LABELS instead of wires! Please USE LABELS instead of wires!** Add **additional SSI** components **only** if necessary, but **ONLY** SSI components. As mentioned previously, if possible, use only PORTF for the Buzzer, the Text (SMS) Generator, and the switch circuit for the external interrupt. **Clearly label** all **relevant** signals. **Please USE LABELS instead of wires!**

10 min



Exam 2

______________________, ______________
Last Name , First Name

[10%] 2. Solve the following short problems related to asynchronous serial communication.

(4%) 4 min

a) Assume that an asynchronous serial communication system is running at **16 MHz** and that the baud rate for the serial system should be set to **32 kHz**. Calculate any specific value(s) that you will need to achieve this rate. You **must** use BSCALE = **–2**.

(3%) 3 min

b) In asynchronous serial communication, data is sampled much faster than data is transmitted. Explain why.

(3%) 3 min

c) Assume that a processor has a UART system **without** a double buffered receiver. Explain the implications of this missing hardware.

Exam 2

________________________, _______________

Last Name , First Name

[25%] 3. As your parents are getting older, their bones are getting weaker. Because they always took great care of you, you have decided to repay them by giving them a glass of milk every day, with hope that it will make their bones strong again. The problem is, you are an engineering student and are therefore extremely busy. Thankfully, you are taking EEL3744 and are now a microcontroller wizard. Since you are unable to visit your parents every day, you are determined to create a milk-dispensing robot, that will automatically pour them a glass of milk **every 24 hours from when they last had milk**. The robot will be controlled by an Atmel XMEGA128A1U microprocessor with the following features and requirements:

10 min

- The technology to automatically pour a glass of milk every **24 hours** and then send you a text (SMS) message **after detecting** that your parents have drank their milk. (You will not handle the detecting technology on this exam.)
- An obnoxious **buzzer** that will alert your parents as soon as fresh milk is poured. The buzzer will play sound when its input pin (as chosen in problem 1) is pulled **high**, and should continue to do so until the robot **automatically detects** that your parents have drank their milk. (Again, this automatic detection is not handled on this exam.) Any time you tell the robot to pour a glass of milk, you **MUST** start the buzzer. Do not worry about turning it off, as this will be done automatically for you.
- A **button**, utilizing the switch circuit configured in problem 1, when pressed, will **potentially** make the robot pour a fresh glass of milk **immediately**. This is to allow your parents more than one glass of milk every 24 hours, if they so desire, although milk should **NOT** be given to your parents more often than every **30 minutes**.

NOTE: All code written for this problem must be written in **assembly language**. **Assume** that the system clock is **32.768 kHz**. (Note that the clock speed is in **k**Hz, not **M**Hz.)

(4%) a) First, you will need a timer to keep track of time. To do so, you must configure TCC0 with an **overflow interrupt** (whose handler you will write in part d), and you must use the **largest possible prescaler** for the timer clock. Write a subroutine, ***INIT_TCC0***, that will initialize the TCC0 module to trigger an overflow interrupt at a rate of your choosing. Remember that you will need to be able to determine intervals of 30 minutes.

5 min

Assembly Code	Comments (or More Assembly Code)

Exam 2

_______________________, _______________
Last Name , First Name

(4%) 3. b) As mentioned before, the robot must be configured to *potentially* pour a glass of milk immediately upon your parents pressing a **button**. To do so, you will configure an external interrupt. Write a subroutine, ***INIT_GPIO***, that will initialize the necessary GPIO pin, as well as the desired external low-to-high interrupt. Use the same pin that you chose in problem 1. You will write the external interrupt handler in part e of this problem.

4 min

Assembly Code	Comments (or More Assembly Code)

(3%) c) Write the interrupt vector initialization **program fragment** for all interrupts used. Assume that this program fragment is appropriately placed in the main **assembly language** routine.

3 min

Assembly Code	Comments (or More Assembly Code)

Exam 2

________________________, ________________
Last Name , First Name

(7%) 3. d) Write the TCC0 overflow interrupt handler. Assume that a memory location, defined with the label **COUNT**, can be used to store the number of timer/counter overflows that have occurred **within the past 24 hours**. If it is determined that 24 hours has elapsed, your routine should store 0x4D (‘M’) into the data memory location 0x3755. (After storing this data, another routine, that you will not write on this exam, will automatically cause the robot to pour your parents a glass of milk.)

8 min

Assembly Code	Comments (or More Assembly Code)

Exam 2

______________________, ______________

Last Name , First Name

(7%) 3. e) Write the external interrupt handler for the button. If more than 30 minutes have elapsed, you must "tell the robot" that it needs to pour milk (by storing 0x4D ['M'] into the data memory location 0x3755, like in the previous part of this problem). Again, assume that memory location referenced by **COUNT** is used to store the number of timer/counter overflows that have occurred **within the past 24 hours**.

8 min

Assembly Code	**Comments (or More Assembly Code)**

Exam 2

______________________, ______________

Last Name , First Name

[35%] 4. Your sketchy hobbyist friend, Roscoe, is having some issues handling serial communication in a small microprocessor application design, and asks for your help in creating a solution.

10 min

Roscoe doesn't tell you much about the project, but apparently the goal is to create a serial communication interface, using the Serial Peripheral Interface (SPI) protocol, between two Atmel XMEGA128A1U microprocessors. The problem is, the data must be **only 6 bits long**, which is not possible with a built-in XMEGA SPI module.

Roscoe further explains that the hardware design for any solution must **only** utilize a single 8-bit GPIO port, PORTA, available on both processors. Though Roscoe is baffled by how to overcome this challenge, **you remember that SPI is simply another procedure for transmitting/receiving serial data through a set of predefined signals**. After asking Roscoe some more questions, you find out only the following *necessary and sufficient* information about his project:

- One of the processors, given the name **SCO0** by Ro**sco**e, is considered the master device in terms of serial communication. Assume that the system clock rate of the master processor is much slower than the slave processor's system clock rate. The other processor, given the name **SCO1**, is considered the slave device. **SCO0 will only ever transmit data to SCO1.**
- The chosen SPI protocol is such that when not transmitting data, the clock polarity must be **low**. Furthermore, when transmitting data, data must be **setup on a rising edge**, and **sampled on a falling edge**. Send the data **as fast as** possible, with no necessary time delay between bits.
- SCO0 will transmit only the **least-significant six bits of data** stored in one of its global variables, to-be-declared as ***uint8_t glo_sco01***, to SCO1. Upon completely receiving the contents of ***glo_sco01,*** SCO1 will call a pre-defined function ***void nothing_suspicious(uint8_t)***, passing in the received contents. This function is available in a pre-built header file, *roscoe.h*, given by Roscoe.
- SCO0 must utilize the TCE1 system available within the XMEGA to determine when to start serial communication between the two processors, i.e., when SCO0 should transmit the contents of ***glo_sco01***. To configure the TCE1 system, a pre-built function ***void tce1_init(void)*** (also available in *roscoe.h*), must be used. After calling ***tce1_init***, every second, an interrupt flag (**OVFIF**) stored within the TCE1 interrupt flag register will be set. This flag should be manually monitored within the serial communication interface for SCO0, to determine when to start serial communication between the two processors. In other words, every second, a 6-bit SPI transmission will begin.
- An interrupt **must** be configured in SCO1, to determine when to read a bit of data transmitted from SCO0. For this problem, this is the only timing that is **extremely critical**. (Remember that the master system clock rate is much slower than the slave processor's system clock rate.)

(3%) a) Draw a timing diagram of a single SPI transmission, modeling the SPI protocol given above, with 0x38 being the data transmitted.

4 min

Exam 2

_______________________, _______________
Last Name , First Name

(3%) 4. b) Design a hardware schematic for the serial communication interface between SCO0 and SCO1. Only include the necessary connections for implementing SPI communication.

4 min

4 min

(4%) c) Write a C function, ***void sco0_gpio_init(void)*** to initialize any GPIO for SCO0.

C Code	Comments (or More C Code)

(4%) d) Write a C function, ***void sco1_gpio_init(void)*** to initialize any GPIO, as well as the **necessary** GPIO interrupt, for SCO1.

4 min

C Code	Comments (or More C Code)

Exam 2

____________________, ______________

Last Name , First Name

9 min

(11%) 4. e) Design a **complete** C program (*sco0.c*) to design the serial communication interface for SCO0. Only include what is necessary for this program, in the context of this problem.

C Code	Comments (or More C Code)

EEL 3744—Spring 2018
4 April 2018

Exam 2

Dr. Eric M. Schwartz

______________________, ______________
Last Name , First Name

(9%) 4. f) Design a **complete** C program (*sco1.c*) to design the serial communication interface for SCO1, **including the necessary interrupt service routine**. Only include what is necessary for this program, in the context of this problem.

9 min

C Code	Comments (or More C Code)