CS224-Spring 2018-Lab #7 (Version 2: May 1 2018, 3:43 pm)
Programming PIC32 Microcontroller

Dates: Section 1, Monday, 7 May, 13:40-17:30
Section 3, Tuesday, 8 May, 13:40-17:30
Section 2, Wednesday, 9 May, 13:40-17:30
Section 4, Thursday, 10 May, 13:40-17:30
Section 5, Friday, 11 May, 8:40-12:30
Section 6, Friday, 11 May, 13:40-17:30

[THIS LAB WILL BE DONE INDIVIDUALLY: NO GROUP FORMING IS ALLOWED.]

DURING THE WEEK OF MAY 1 THE LAB ENGINEER BURHAN KAYA WILL BE IN THE LAB TO ANSWER YOUR QUESTIONS.

Purpose: In this lab you will use the C programming language to develop simple applications for the PIC32 microcontroller. The mikroC IDE (Integrated Development Environment) will be used as the software environment, and the Beti PIC32 Trainer Pack will be used as the hardware environment in this lab.

Summary
Part 1 (35 points): Understanding the fundamentals of PIC32 microcontroller.
Part 2 (65 points): Experiments with PIC32 microcontroller.

TRY TO FINISH PART 2 BEFORE COMING TO THE LAB. MAKE SURE THAT YOU DO THE DEMO TO TA.

DUE DATE/TIME OF PART 1 --SAME FOR ALL SECTIONS

a. Please drop your written Preliminary Design Report into the box provided in front of the lab by 13:40 on Monday May 7. No late submissions will be accepted! Your paper submission for preliminary work must match MOSS submission.

   Please **upload your Preliminary work** to the Unilica Assignment by 13:40 on Monday May 7. Use filename **name_surname_SecNo_Lab7_PRELIM.txt** [A NOTEPAD FILE as its extension suggests, which contains only the program part.]

DUE TIME OF PART 2—DIFFERENT FOR EACH SECTION:

a. You have to demonstrate your Part 2 lab work to the TA for grade by **12:15** in the morning lab and by **17:15** in the afternoon lab. Your TAs may give further instructions on this. If you wait idly and show your work last minute your TA may not accept it. Make sure that you follow your TAs' instructions.
b. At the conclusion of the demo for getting your grade, you will **upload your lab work** to the Unilica Assignment, for similarity testing by MOSS. Please see the related section below for further instructions on MOSS submission.

Part 1. Preliminary Design Report: Developing Codes (30 points: 3 X 10 points)

You have to provide a neat presentation prepared by Word or a word processor with similar output quality. Handwritten answers will not be accepted. At the top of the paper on left provide the following information and staple all papers. Please make sure that this info is there for proper grading of your work, otherwise some points will be taken off.

CS224
Section No.: …
Spring 2018
Lab No.:
Your Full Name/Bilkent ID:
.

a) Research and read about SFRs. Explain the differences between TRISx, PORTx, LATx and ODCx ports. Specify the special function registers (SFRs) for the I/O device(s) involved in Part2.a and Part2.b.

b) Give the C code for Part2.a, with comments, an explanatory header, well-chosen identifiers and good use of spacing and layout to make your program self-documenting.

c) Give the C code for Part2.b, with comments, an explanatory header, well-chosen identifiers and good use of spacing and layout to make your program self-documenting.

You can read Chapter 8.6 Embeded I/O Systems in the textbook and learn about SFRs at http://ww1.microchip.com/downloads/en/DeviceDoc/61120D.pdf .

About the Beti PIC32 Trainer Pack

You only need to connect USB cable to the small PIC32 daughter board for both power supply and programming. Please check schematic files of the Beti board posted on Unilica if you need more information. The part number of the microcontroller we use is PIC32MX795F512L. You can refer to its datasheet (posted in Unilica) if you need more information. Additionally there will be small presentation about wire and jumper connection for each lab section. Note that you borrow a Lab-board containing the development board, connectors, etc. in the beginning. You are responsible for the lab board and you have to return all of them to the lab supervisor when you are done, otherwise you will lose some points.

Part 2. Implementation using C and mikroC IDE (70 points: 2 X 35 points)

Part 2.a: Understating and Building a Digital Counter

In this part, you should use 4 push buttons and 8 LED outputs on the Beti board. Two push buttons are used for less significant bit control and the other 2 are used for time control. You display an 8-bit counter on the LEDs. The first **two buttons** control the **position of the least significant bit** (lsb) on the LEDs. According to the button pressed, you should display binary numbers from left to right (00000001,00000010, 00000011, 00000100, …) or from right to left (10000000, 01000000, 11000000, 00100000, …).

Part 2.b: Accelareted speed control

If both the lsb control buttons are pushed simultaneously, or neither, nothing is displayed. The last **two buttons** control the **speed of counting**. Depending on which button is pressed, your counter should change numbers faster or slower than half a second. If both the time control buttons are pushed (or both not pushed), the counter should increment every half second.

Part 3. Submit your code for MOSS similarity testing

After demonstrating your working (or partially working) codes to the grader, you should immediately submit your C codes for similarity testing to the Unilica > Assignment specific for your section. You will upload one file, named **name_surname_SecNo_Lab7_C.txt.** Be sure that the file contains exactly and only the codes which are specifically detailed in Part 2. Check the specifications! *Even if you didn't finish, or didn't get the C codes working correctly, you must submit your code to the Unilica Assignment for similarity checking.* Failure to submit your codes will result in a lab score of 0.   Your codes will be compared against all the other codes in all sections of the course, by the MOSS program, to determine how similar it is, as an indication of plagiarism. **So be sure that the code you submit is code that you actually wrote yourself!** [Warning: DON'T use any code found somewhere on the internet, since others might also find and use it, and MOSS will determine that yours is similar to theirs!]   All students must upload their Part 2 code to Unilica > Assignment while the TA watches, at the end of your demo-for-grading time. Submissions made without the TA observing will be deleted, resulting in a lab score of 0.

Part 4. Cleanup

1. After saving any files that you might want to have in the future to your own storage device, erase all the files you created from the computer in the lab.
2. When applicable put back all the hardware, boards, wires, tools, etc where they came from.
3. Clean up your lab desk, to leave it completely clean and ready for the next group who will come.

LAB POLICIES

1. You can do the lab only in your section. Missing your section time and doing in another day is not allowed.
2. Students will earn their own individual lab grade. The questions asked by the TA will have an effect on your individual lab score.
3. Lab score will be reduced to 0 if the code is not submitted for similarity testing, or if it is plagiarized. MOSS-testing will be done, to determine similarity rates. Trivial changes to code will not hide plagiarism from MOSS—the algorithm is quite sophisticated and powerful. Please also note that obviously you should not use any program available on the web, or in a book, etc. since MOSS will find it. The use of the ideas we discussed in the classroom is not a problem.
4. You must be in lab, working on the lab, from the time lab starts until your work is finished and you leave.
5. No cell phone usage during lab.
6. Internet usage is permitted only to lab-related technical sites.
7. For labs that involve hardware for design you will always use the same board provided to you by the lab engineer.