

CS224 - Fall 2023- Lab #7

Programming PIC32 Microcontroller (Version 1, December 6, 9:20 pm)

Important Note: This lab can be done in groups of 2 or individually. There are changes in submission of preliminary report and lab work. You must read the entire document carefully.

Dates: (TAs - Tutor)

Section 1: Wed, 13 Dec, 13:30-17:20 in EA-Z04 (Soheil, Mustafa - Begüm)

Section 2: Thu, 14 Dec, 13:30-17:20 in EA-Z04 (Soheil, Kadri - İlhami)

Section 3: Thu, 14 Dec, 8:30-12:20 in EA-Z04 (Onur, Kadri)

Section 4: Fri, 15 Dec, Fri 13:30-17:20 in EA-Z04 (Onur, Pouya - Hakan)

TA Full Name (email address: @bilkent.edu.tr)

M. Kadri Gofralılar (kadri.gofralilar)

Mustafa Yasir Altunhan (yasir.altunhan)

Onur Yıldırım (o.yildirim)

Pouya Ghahramanian (ghahramanian)

Sepehr Bakhshi (sepehr.bakhshi)

Soheil Abadifard (soheil.abadifard)

Tutor Full Name (email address: @ug.bilkent.edu.tr)

Z. Begüm Kara (begum.kara): Sec. 1, 15:30 - 17:20

Hakan Muluk (hakan.muluk)

İlhami Uluğtürkkan (i.ulugturkkan)

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 Training Set card will be used as the hardware environment in this lab.

Summary

Preliminary Work - Developing Codes with SFR's: 30 points

Lab Work: 70 points

1. Compiling, downloading and testing (10 points)
2. DC Motor (25 points)
3. Seven Segment Module (35 points)

Important Notes for All Labs About Attendance, Performing and Presenting the Work

1. You are obliged to read this document word by word and are responsible for the mistakes you make by not following the rules.

2. Not attending to the lab means 0 out of 100 for that lab. If you attend the lab but do not submit the preliminary part you will lose only the points for the preliminary part.
3. Try to complete the lab part at home before coming to the lab. Make sure that you show your work to your TAs and answer their questions to show that you know what you are doing before uploading your lab work and follow the instructions of your TAs.
4. In all labs if you are not told you may assume that inputs are correct.
5. In all labs when needed you have to provide a simple user interface for inputs and outputs.
6. Presentation of your work

You have to provide a neat presentation prepared proper forms as defined below. Your programs must be easy to understand and well structured.

Provide following six lines at the top of your submission for preliminary and lab work (make sure that you include the course no. CS224, important for ABET documentation).

CS224

Lab No.

Section No.

Your Full Name

Bilkent ID

Date

Please also make sure that your work is identifiable: In terms of which program corresponds to which part of the lab.

7. **If we suspect that there is cheating we will send the work with the names of the students to the university disciplinary committee. You can experiment with ChatGPT for learning; however, you cannot use/modify the code generated by it. Such an act is classified as plagiarism. Note that MOSS is capable of detecting ChatGPT code. Furthermore remember that, the code you use from ChatGPT can also be used by another student in the course. Make sure that the code you submit is really yours and has been internalized.**

DUE DATE OF PRELIMINARY WORK: SAME FOR ALL SECTIONS

No late submission will be accepted.

- a. Please upload your preliminary work to Moodle by 1:30 pm on Wednesday Dec 13 for similarity testing by MOSS.
- b. For preliminary work upload a file with the filename **StudentID_FirstName_LastName_SecNo_PRELIM_LabNo.txt** Only a NOTEPAD FILE (txt file) is accepted. Any other form of submission receives 0 (zero). If you are doing the lab in a group of 2, both members must upload a file separately.
- c. Note that the Moodle submission closes sharp at 1:30 pm and no late submissions will be accepted. You can make resubmissions before the system closes, so do not wait for the last moment. Submit

your work earlier and change your submitted work if necessary. Note that only the last submission will be graded.

- d. Do not send your work by email attachment, they will not be processed. They have to be in the Moodle system to be processed.
- e. At the beginning of your submission files include the following **for each group member**. Make sure that each of them is in a separate line: Course No.: CS224, Lab No., Section No., Your Full Name, and your Bilkent ID. No names can be added after the submission of preliminary report. Each student must submit a copy.

DUE DATE OF LAB WORK): (different for each section) YOUR LAB DAY

- a. You have to demonstrate your lab work to your TA for grading. Do this by **12:00** in the morning lab and by **17:00** in the afternoon lab. Your TAs may give further instructions on this and they may make changes. If you wait idly and show your work last minute, your work may not be graded. Make sure that you follow your TA's instructions.
- b. At the conclusion of the demo for getting your grade, you will **upload your Lab Work** to the Moodle Assignment, for similarity testing by MOSS. See lab part submission details below.
- c. Aim to finish all of your lab work before coming to the lab, but make sure that you upload your work after making sure that your work is analyzed by your TA and/or you are given the permission by your TA to upload.
- d. At the beginning of your submission files include the following for each group member. Make sure that each of them is in a separate line: Course No.: CS224, Lab No., Section No., Your Full Name, and your Bilkent ID. No names can be added after the submission of preliminary report.

Part 1. Preliminary Work: Developing Codes with SFR's (30 points)

You must prepare a text file for the preliminary report. At the top of the file provide the following information. If you are doing the lab in a group of two, write both student names and IDs (please make sure that this info is there for proper grading of your work).

CS224 / Lab No.

Sec No

Your Name

Your ID

Date

Your Preliminary Design Report (PDR) should contain the following 4 items (a - d):

- a) Specify the special function registers (SFRs) for the I/O device(s) involved in Part2-b.

- b) C code for Part2-b, with lots of comments, an explanatory header, well-chosen identifiers and good use of spacing and layout to make your program self-documenting.
- c) Specify the special function registers (SFRs) for the I/O device(s) involved in Part2-c.
- d) C code for Part2-c, with lots of 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 Embedded I/O Systems at textbook (particularly 8.6.2 General-Purpose Digital I/O section) and look into ***pic32IOPorts.pdf*** for specifying SFR. You will only use TRISx, PORTx and alternatively LATx registers during this lab.

Part 2. Lab Work (70 points)

a) Compiling, downloading and testing (10 points)

In this part you will learn to use mikroC IDE tool for the C language, targeting the PIC32 microcontroller. You will compile some simple code, use Beti PIC32 Training Set Card as the hardware environment, and download the code to program PIC32 microcontroller. Finally, you will develop a simple application program, and test your code on the board. If this is the first time you are doing this, you may ask your TA for help.

You will find the mikroC IDE development tool already installed on the lab computers. Use mikroC to implement the C program given in the ***ExampleProject.rar*** folder. You should compile and build the code and get its final output. For each build, the final output of mikroC will be a file with '.hex' extension. When you have made the hex file from the C program, you are ready to download it to PIC32. After successfully generating this hex file, use 'mikroBootloader' program which is also provided in ***ProgrammerTools.zip*** to program PIC32 microcontroller with the hex file. Check the manual provided in the zip file for help in mikroBootloader. You can find the 'mikroelektronika USB HID BootLoader v2.1.1.0' already on the lab computers.

About the Beti PIC32 Egitim Seti: Watch the Video

You only need to connect USB cable to small PIC32 daughter board for both power supply and programming. Please check the corresponding schematic file of Beti board module in ***Schematics.zip*** if you need more information. The number of the microcontroller we use is PIC32MX795F512L. You can refer to its datasheet (provided under ***beti.zip***) if you need more information. Additionally, you should watch the video ***beti_intro.mp4*** to learn about wire and jumper connections. Note that you borrow a

Lab-board containing the development board, connectors, etc. in the beginning. The lab supervisor takes your signature. You are responsible for the lab board and you have to return it to the lab supervisor when you are done.

b) DC Motor (25 points)

Use two pushbutton switches to specify the direction of the DC motor. You must choose button 0 and button 1 on the Beti training board. When button 0 is equal to 1, direction of the motor should be clockwise, and when button 1 is equal to 1 its direction should be counterclockwise. When both buttons are clicked, motor should not turn.

Hint: Check push button and DC Motor schematics before starting. In these schematics, J ports are the ones that you connect with a wire. Moreover, understanding the code given in example project will help you a lot in this part, as you will use SFR's in a similar manner.

c) Seven Segment Module (35 points)

Using 4-digit Seven Segment module, show consecutive numbers. If you implement your code correctly, display will be appeared like below box. You can use the code in ***seven_segment.c*** as a starting point.

*Hint: Check seven segment module schematics before starting. Note that you can display only one digit at a certain time. However, if you switch digits fast enough, you can give the illusion that all digits are displayed simultaneously. An example of this is included in ***seven_segment.c****

```
1234
(one second later)
2345
(one second later)
3456
(one second later)
4567
(one second later)
5678
(one second later)
6789
(one second later)
7891
(...)
```

Part 3. Submit Lab Work for MOSS Similarity Testing

1. Submit your Lab Work codes for similarity testing to Moodle.
2. You will upload one file. Use filename **StudentID_FirstName_LastName_SecNo_LAB_LabNo.txt**. If you are doing the project in a group of two, each student must upload a separate file.
3. Only a NOTEPAD FILE (txt file) is accepted. No txt file upload means you get 0 from the lab. Please note that we have several students and efficiency is important.
4. *Even if you didn't finish, or didn't get the codes working, you must submit your code to the Moodle Assignment for similarity checking.*
5. Your codes will be compared against all the other codes in the class, 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!
6. At the beginning of your submission files include the following **for each group member**. Make sure that each of them is in a separate line: CS224, Lab No., Section No., Your Full Name, Bilkent ID.
7. For your preliminary and lab work to be graded you must attend the lab. If you are doing the lab in a group of 2, both members must attend the lab.

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 it another day is not allowed.
2. The questions asked by the TA will have an effect on your 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 the lab, working on the lab, from the time the lab starts until your work is finished and you leave.
5. No cell phone usage during the lab.
6. Internet usage is permitted only to lab-related technical sites.