

Goal: Learning how to write functions in ARM assembly language that can be called from a program written in C.

Background: This lab consists of two programs. Each program calls and tests a function. One program tests a function that computes the absolute value (absval) and the other tests a function that computes the discriminant (disc) of a quadratic, $b^2 - 4ac$. C versions of each function are provided; your assignment is to translate these functions into ARM assembly and run the program to verify that your implementation works.

PART I: Preparation

1. Download the ZIP file called "Lab Assignments.zip" from the course website on Camino.
2. Unzip the file to your desktop. This should create a folder called "Lab Assignments". Open the folder.
3. Find a double click on the file called "COEN 20.eww". This should open the IAR Embedded Workbench program.
4. If step 3 did not open IAR Embedded Workbench, find the program on the Start Menu and open it. Once open, click on "File > Open > Workspace", navigate to the "COEN 20.eww" file inside the "Lab Assignments" folder and open it.

PART II: Absolute value (using the code provided in file absval.c)

1. In the Workspace panel on the left side of the screen, right-click on "Lab07IntegerFunctionsStraightLine" and select "Set As Active".
2. Click on the "+" sign next to "Lab07IntegerFunctionsStraightLine".
3. Click on the "+" sign next to "Source".
4. When compiling, some source code files should be "included" and some should be "excluded". To do this, right-click on the source code filename and select "Options...". This opens a new window with a check box option in the top left labelled "Exclude from build". Set each of the following source code files as follows:

a. absval.c	Include (do NOT check)
b. absval.s	exclude (CHECK)
c. disc.c	exclude (CHECK)
d. disc.s	exclude (CHECK)
e. main4absval.c	Include (do NOT check)
f. main4disc.c	exclude (CHECK)
5. To compile the program, right-click on "Lab07IntegerFunctionsStraightLine" and select "Rebuild All". The build should complete with no errors or warnings.
6. Connect the LM3S811 board to a USB port on your computer. This provides both power (as indicated by the power LED) and a download connection to the device. From the drop-down menus, select "Project > Download > Download active application" to start the download.

7. To run the program, press the reset button (the one closest to the thumbwheel) on the LM3S811 evaluation board. Press the other button to sequence through all the test cases. Verify that it behaves as expected.

PART III: Absolute value (implementing your ARM code in file absval.s)

1. Change each of the following source code files as follows:
 - a. absval.c exclude (CHECK)
 - b. absval.s Include (do NOT check)
2. Edit absval.s to implement the algorithm described in absval.c.
3. Recompile the program, download it to the board, and verify that it works properly.
4. Demonstrate your working program and submit your source code of absval.s to the teaching assistant according to his instructions.

PART IV: Discriminant (using the code provided in file disc.c)

1. Change each of the following source code files as follows:
 - a. absval.s exclude (CHECK)
 - b. disc.c Include (do NOT check)
 - c. main4absval.c exclude (CHECK)
 - d. main4disc.c Include (do NOT check)
2. Recompile the program, download it to the board, and verify that it operates as expected.

PART V: Discriminant (implementing your ARM code in file disc.s)

1. Change each of the following source code files as follows:
 - a. disc.c exclude (CHECK)
 - b. disc.s Include (do NOT check)
2. Edit disc.s to implement the algorithm described in disc.c.
3. Recompile the program, download it to the board, and verify that it works properly.
4. Demonstrate your working program and submit your source code of disc.s to the teaching assistant according to his instructions.