EC605: Computer Engineering Fundamentals

DS-5 Design Environment Tutorial

Goals

- Introduction to DS-5 design environment.

Overview

This tutorial will give instructions showing how to start a personal virtual machine and load the DS-5 environment.

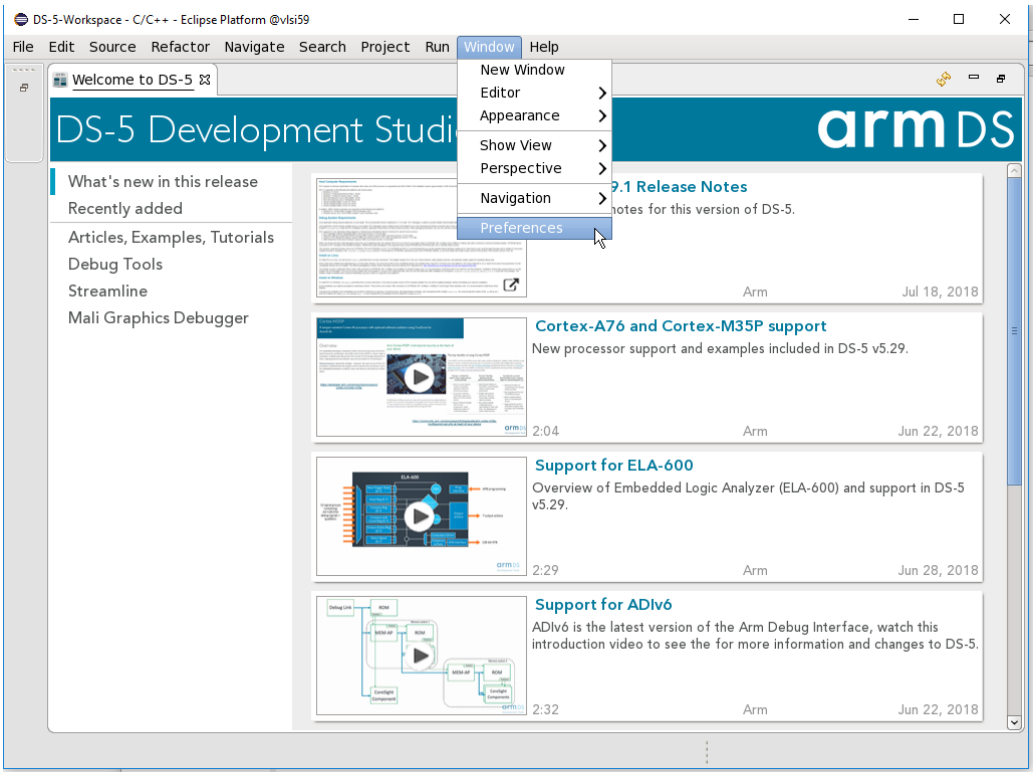
Follow "Task 1" and "Task 2" to set up the DS-5 environment.

You may then follow Tasks 3 and 4 to run your first hello-world application on the DS-5 software.

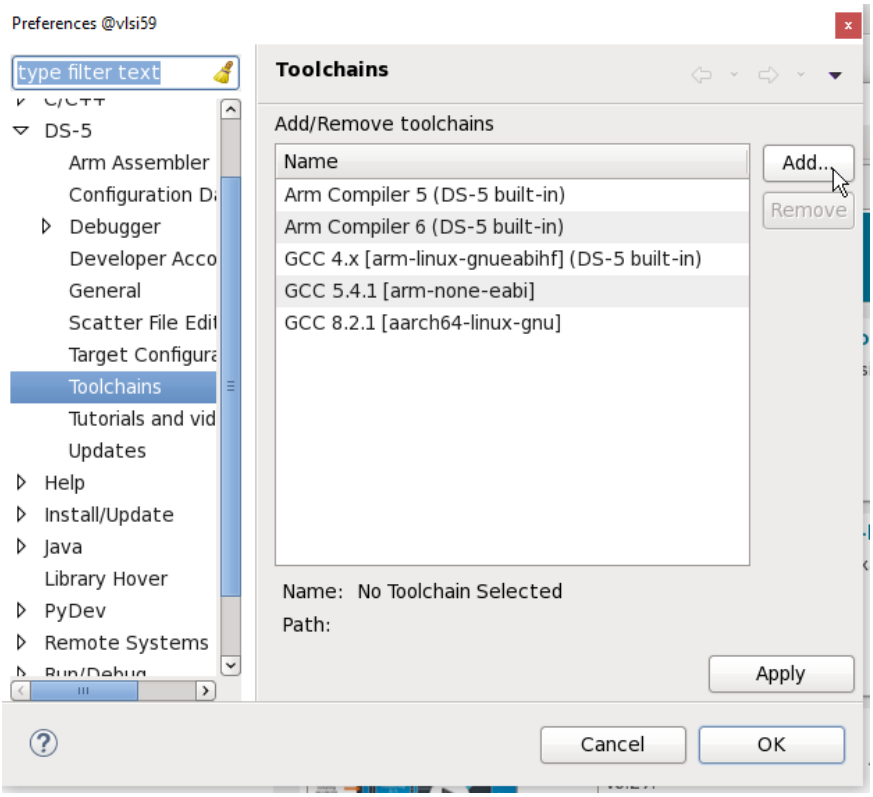
Task 1: Start and setup the DS-5 Design Environment on PC

1. Open the terminal and type in the following command:
   > source DS-5.CE.sh
2. When asked to select a workspace choose somewhere in your home directory. The default location should be fine *(/home/student/DS-5-Workspace*).
3. Implement the following steps on the Local DS-5 to add the GCC 8.2.1 [aarch64-elf] toolchain to the local version DS-5:

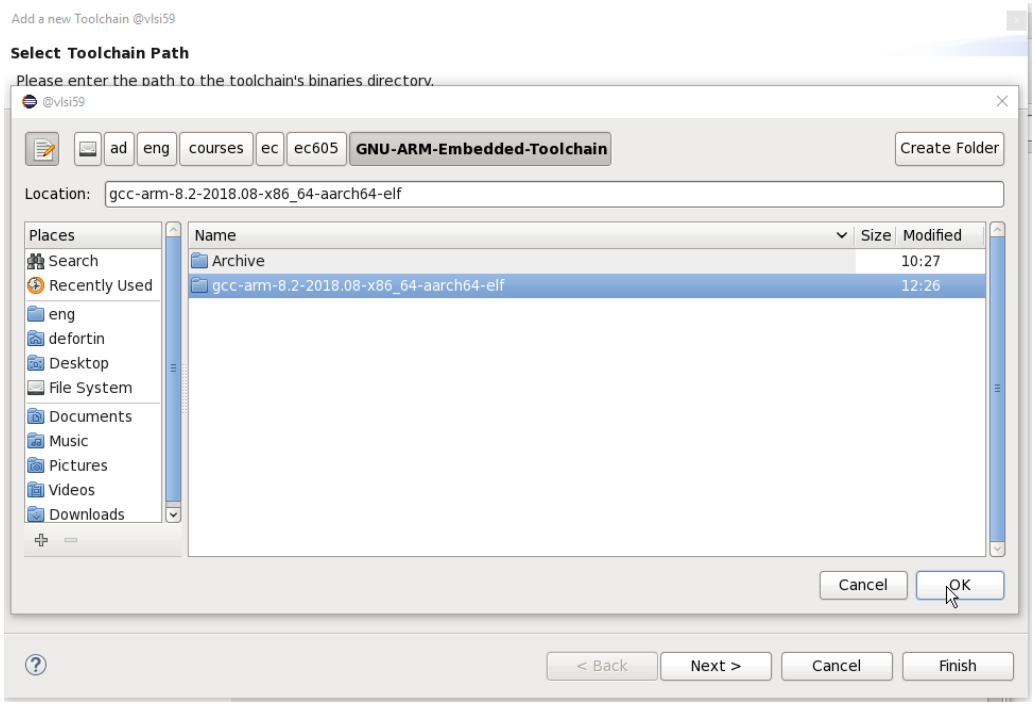
- Go to "Window" > "Preferences"



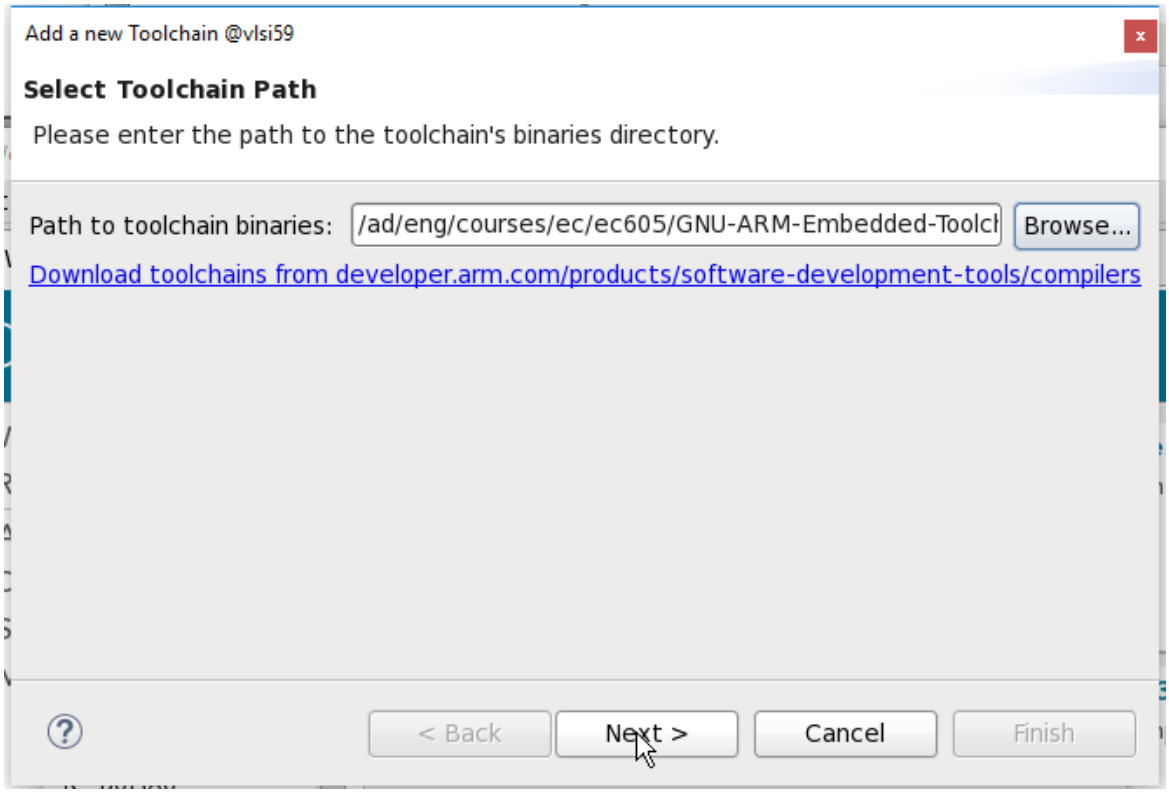
- On the left side of the Preference windows expand the “DS-5” tab and select “Toolchains”
- On the right side of the Preference windows click on the “Add…” button



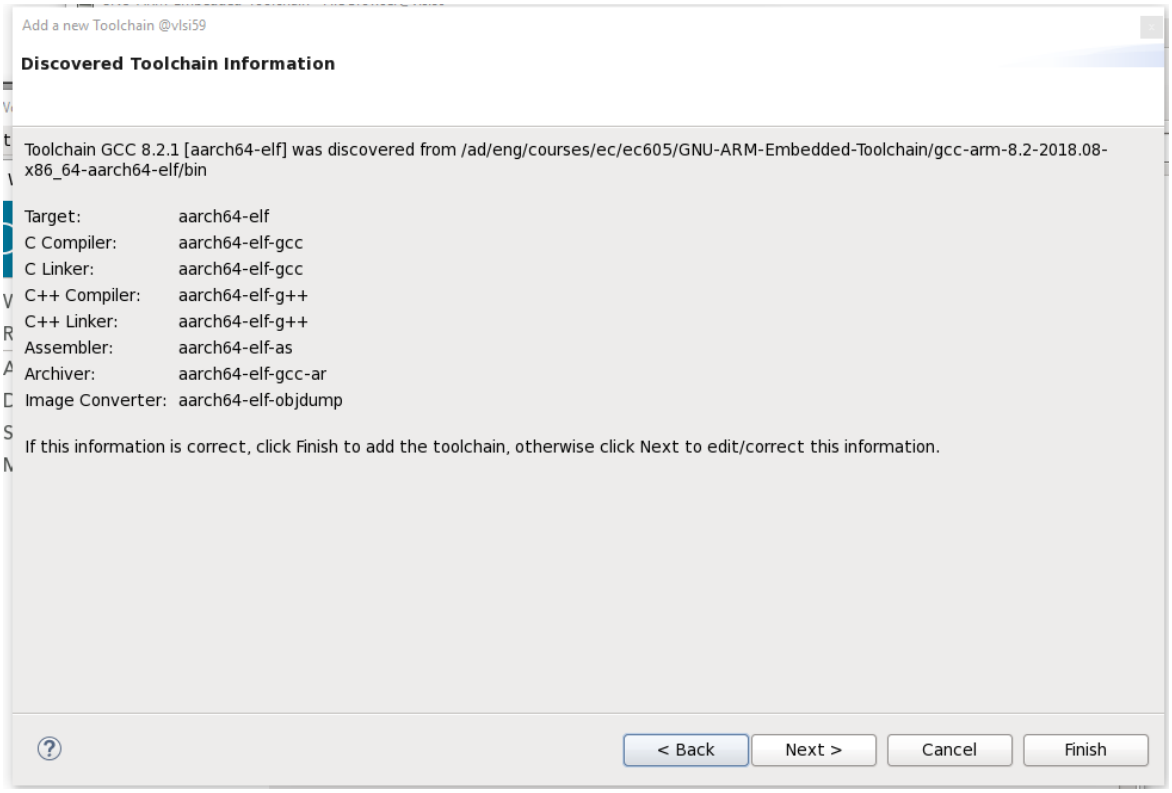
- Navigate to “/ad/eng/courses/ec/ec605/GNU-ARM-Embedded-Toolchain/”
- Select “gcc-arm-8.2-2018.8-x86_64-arrch64-elf” and hit “OK”



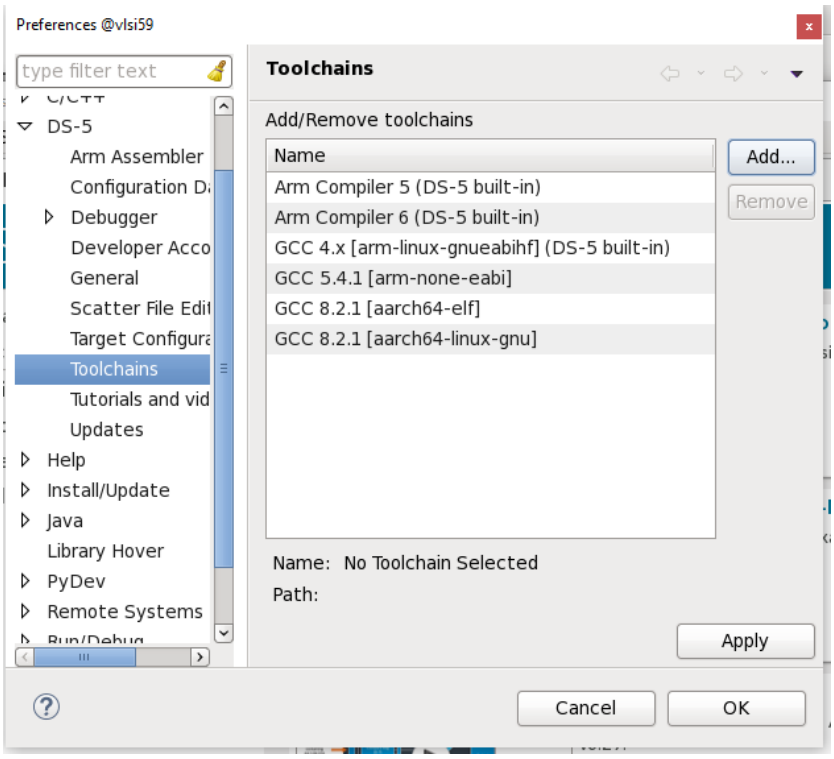
- Press “Next”



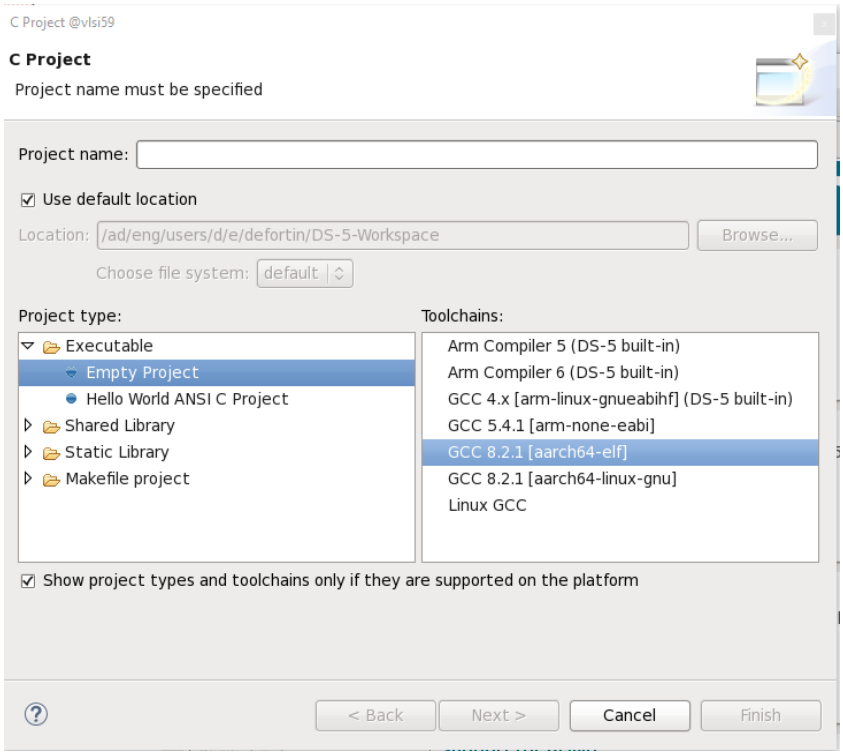
- Hit “Finish”



- Hit “OK” to add the toolchain



- Once the application is back up you should see the GCC 8.2.1 [aarch64-elf] toolchain on the new project:



- You only need to run this once, once the toolchain is added it will be remembered.

Task 2: Open a new project

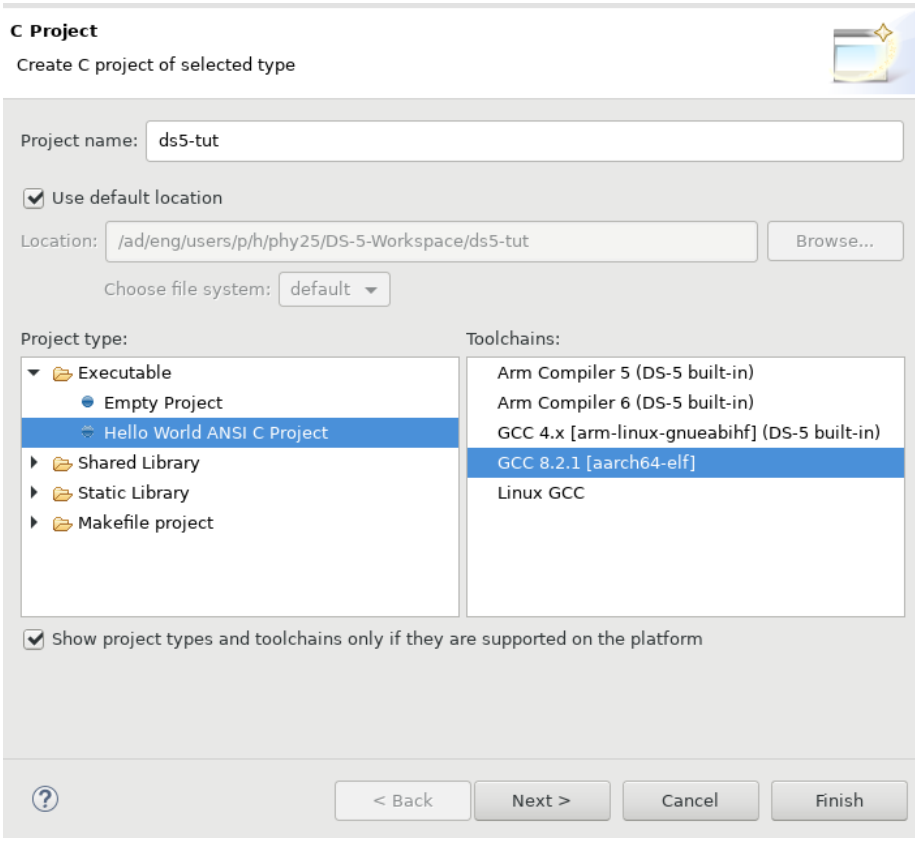
- Start up the DS-5 environment.
- When asked to select a workspace choose somewhere in your home directory. The default location should be fine *(/home/student/DS-5-Workspace*).

Task 3: Create a new hello world application

1. Select *File → New → C Project*.

2. Give your project a name in *Project name*.

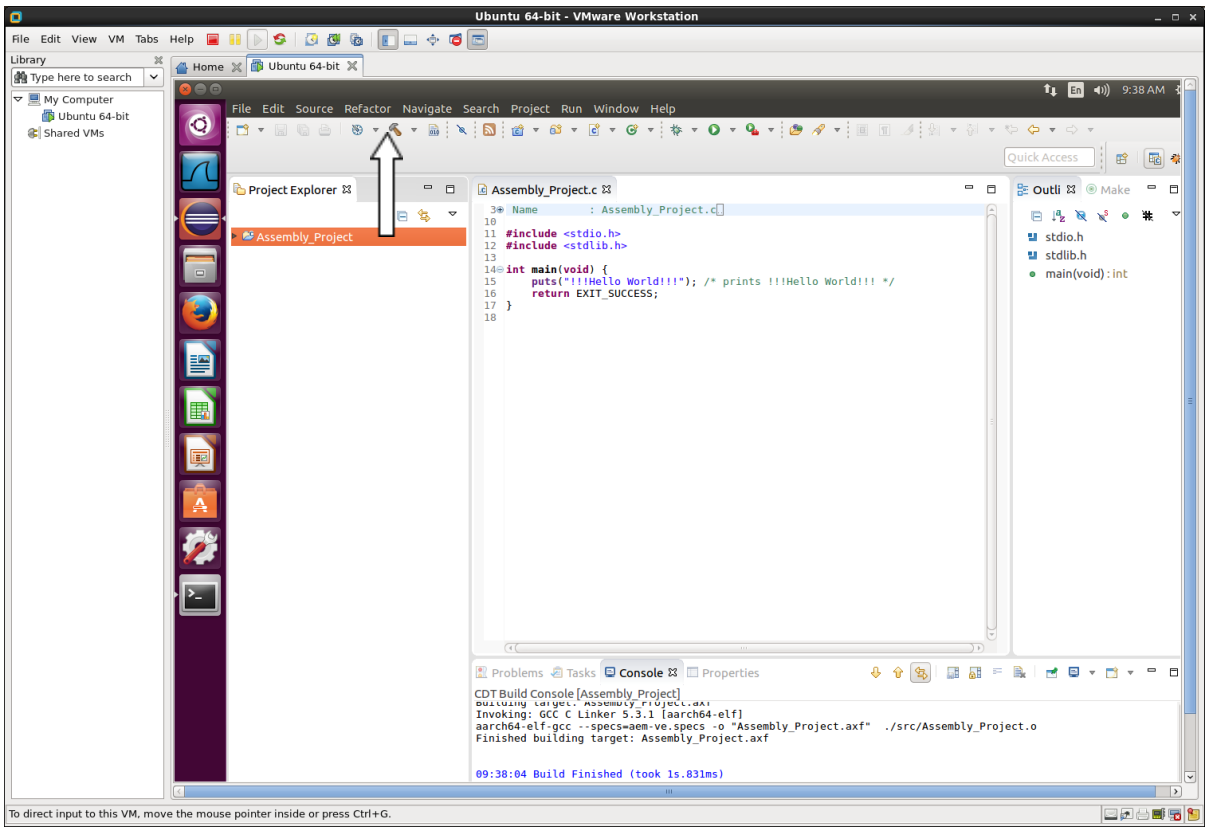
3. Choose the *GCC 8.2.1 [aarch64-elf]* toolchain on the right side and select the *Hello World ANSI C project* on the left.



4. Click Finish and wait for the project to be created.

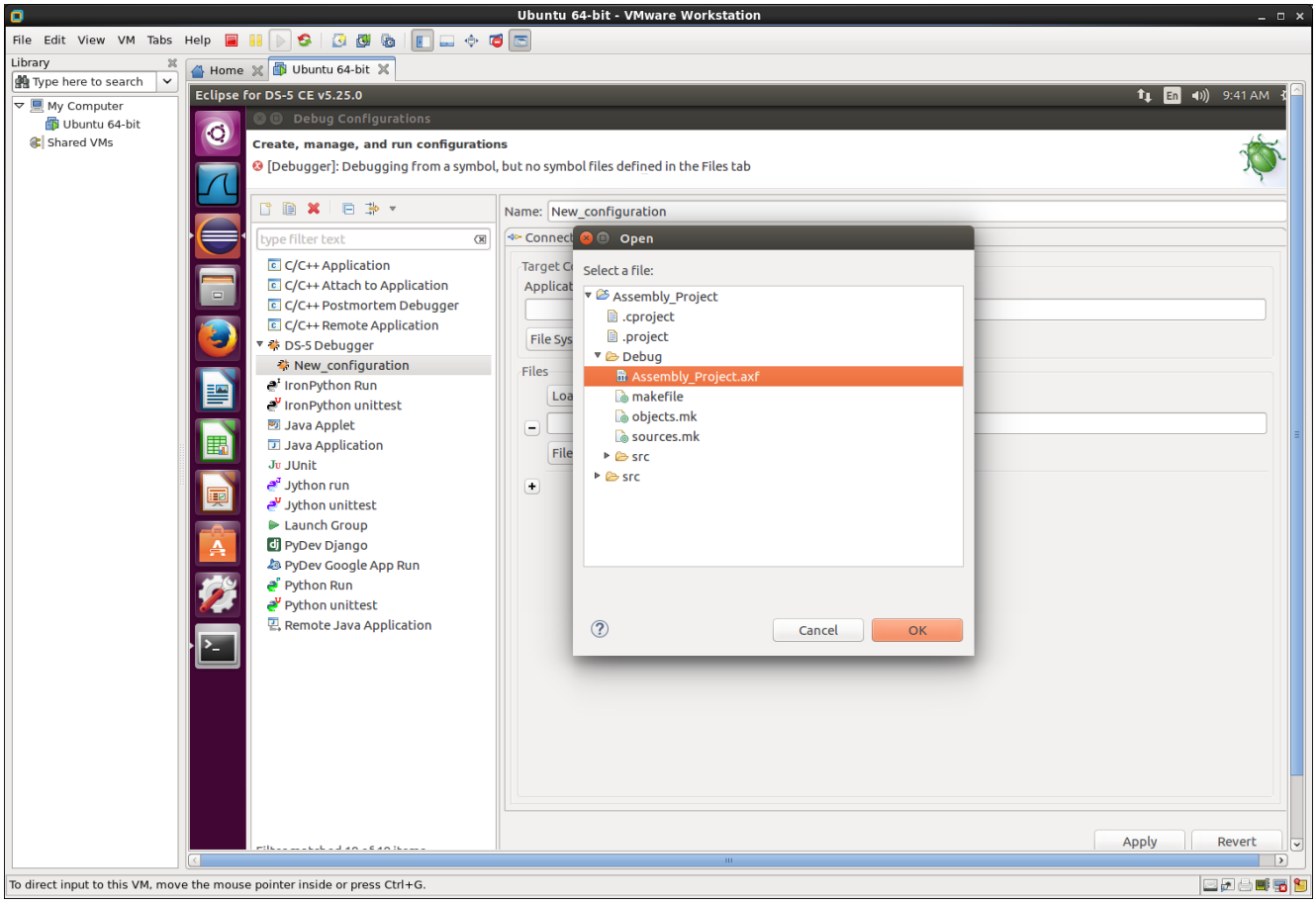
5. Next, modify the build parameters to successfully build the project for the ARM simulator.
   a. Right click on the project and select *Properties*
   b. Navigate to *C/C++ Build → Settings*
   c. Select *Miscellaneous* under *GCC C Linker*
   d. In the Other flags box type in *--specs=aem-ve.specs*
   e. Click OK

6. Make sure that compilation works by selecting the project and clicking the *build* icon or right click on the project and select *Build Project*.



Task 4: Simulating the project using DS-5

1. First create a new debug configuration:
   a. Select *Run → Debug configurations…*
   b. Double click on *DS-5 Debugger*
   c. Select the connection platform as *Debug ARMv8*, under *ARMv8-Ax1 Foundation Platform Bare Metal Debug*
   d. In the files tab, click on *workspace* under *Target Configuration* and browse into *Project Name → Debug → Project_Name.axf*



e. Click *Apply* and *Debug*
f. If a window asking to *perspective switch* appears, select *Yes*

2. The debugger will load a simulated ARM processor and breakpoint at the start of your main function.

3. From here you can *step* through the functions of the application and view the output on the *Target Console* (you may need to move the black *"Fast Models"* window).

   Use the controls provided in the Debug Control view to step through the application:

   - Click to continue processing code.

   - Click to interrupt or pause processing code.

   - Click to step through the code.

   - Click to step over source line.

   - Click to step out.

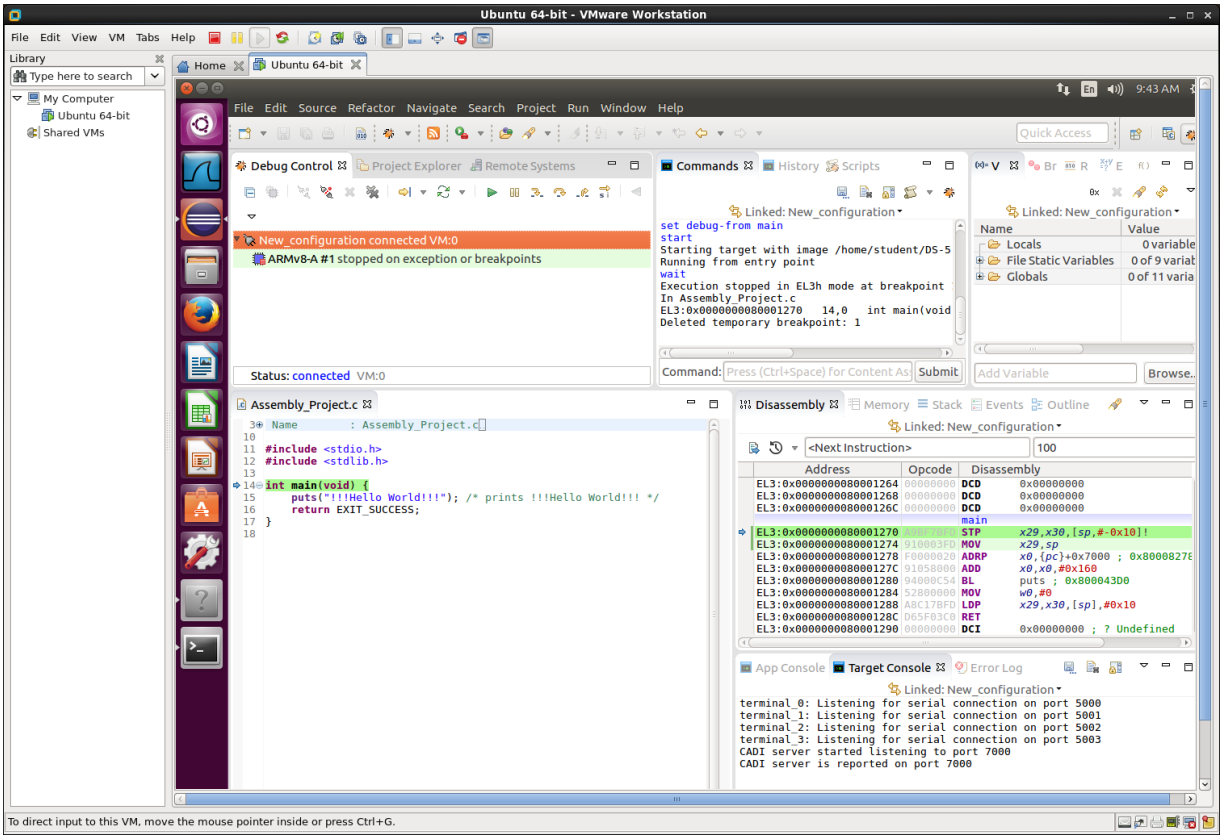
   - This is a toggle. Select this if you want the above controls to step through

instructions.

 - Debug from main()

4. To disconnect from the simulator and load a new revision of the build project, click the *disconnect* button found in the *Debug Control* window.

   a. Use the *perspective switch buttons* to return to the main workspace environment.



Working with Assembly language on DS-5

1. Add an empty C project by right clicking in the “Project Explorer” space -> selecting “new” -> “New C Project”
2. Choose “Empty Project” once the window pops up.

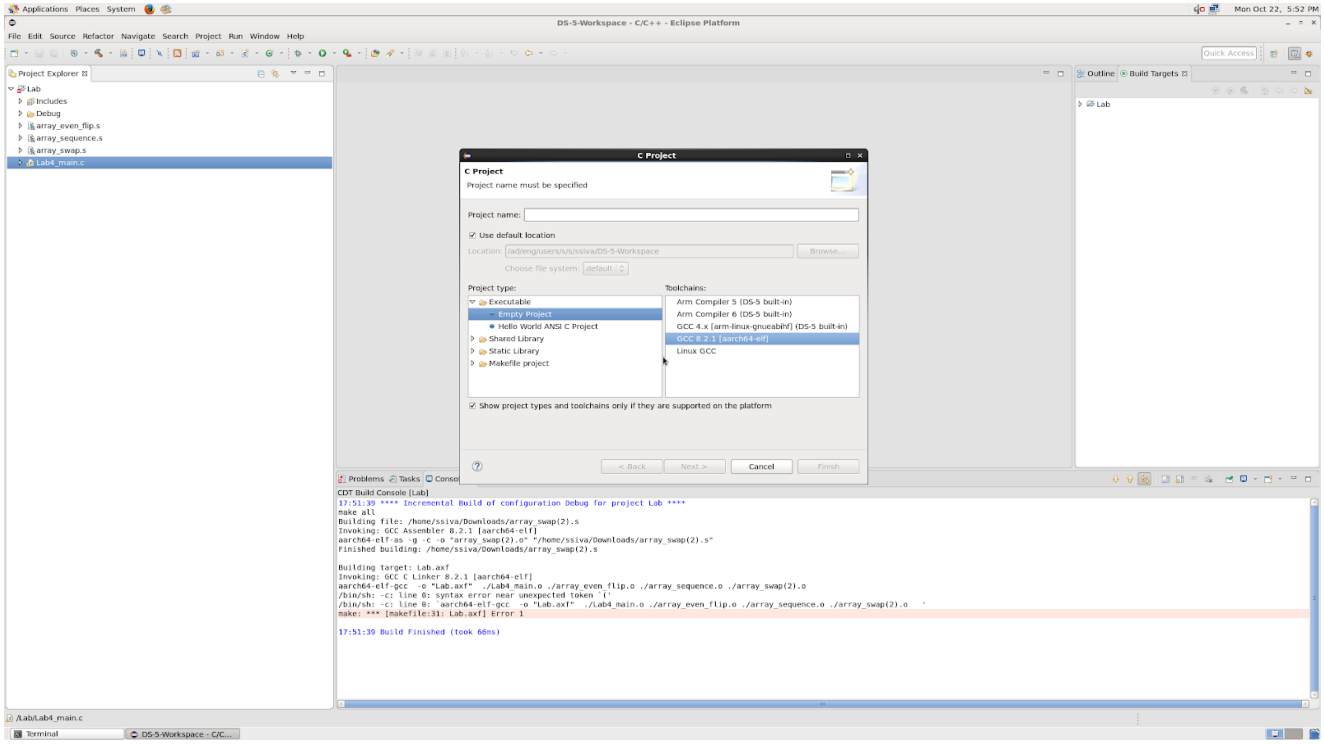


Fig 1

3. Make sure to select the appropriate toolchain as was mentioned in the initial DS-5-Hello_World tutorial.

4.To add a new file, click on the icon (indicated by the mouse pointer in Fig 2). Pick your source folder (generally appears by default), and appropriately name the file (including the extension).

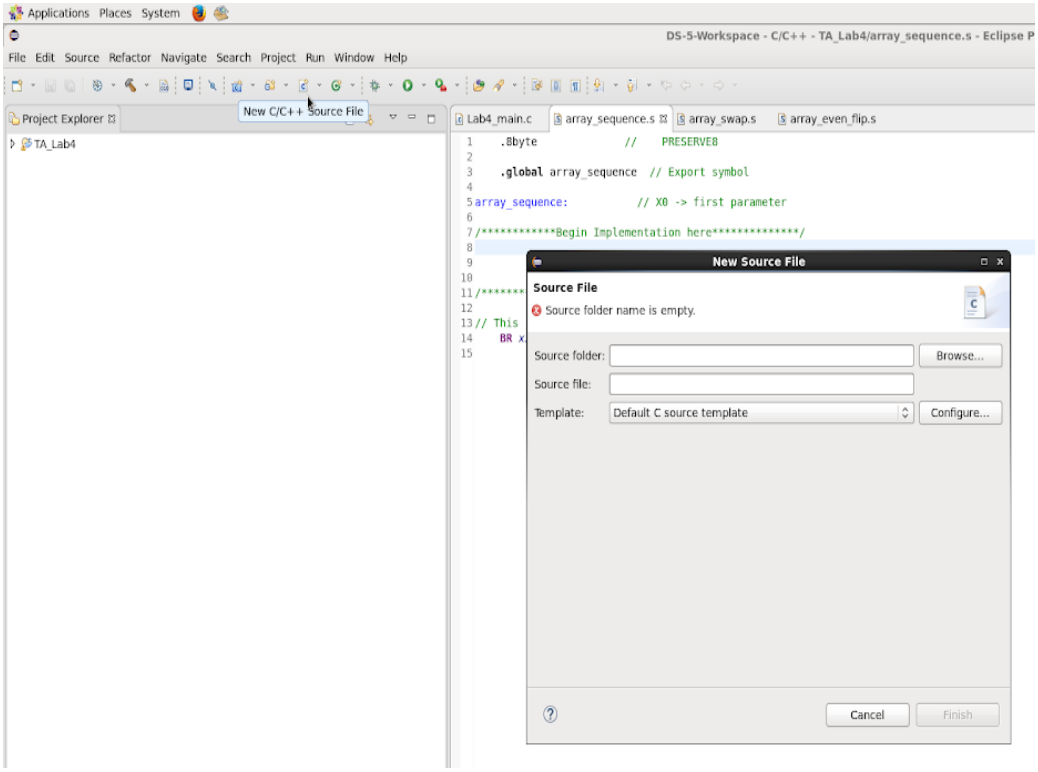


Fig 2

5. In order to avoid errors, check that you are working with the files from inside the project directory, and not the “src” directory. Working from inside the src directory is known to cause unnecessary errors.

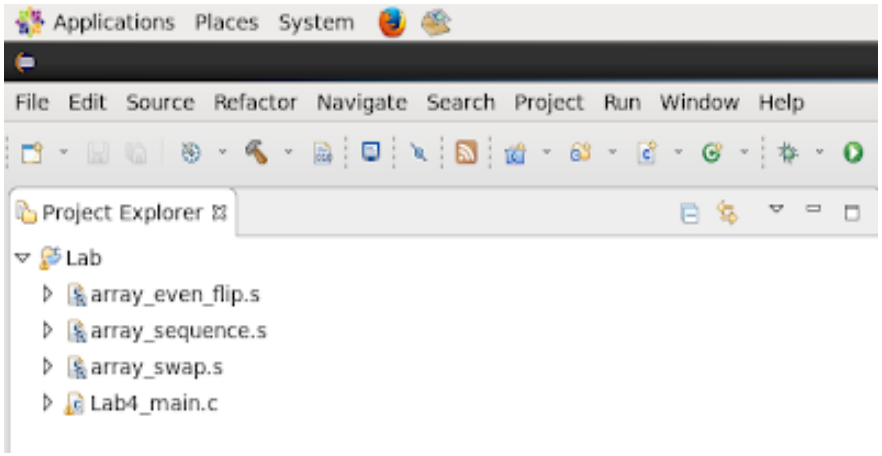


Fig3

6. Another potential problem that you might run into is build error caused by working with linked files. A linked file could be created by importing a file to your project and renaming it. In such cases, remove any linked files, and paste your back into a newly created file. Also don’t forget to put the GCC linker flags on any new projects.

7. Once you are done coding, click on the build icon (the tiny hammer icon on the toolbar), or alternatively “right click on your project -> build project”.

8. You should see a success message in your console if your build was successful. A red cross (Fig 5) on your project folder or error messages in your console (Fig 6) are indicative of an unsuccessful build.

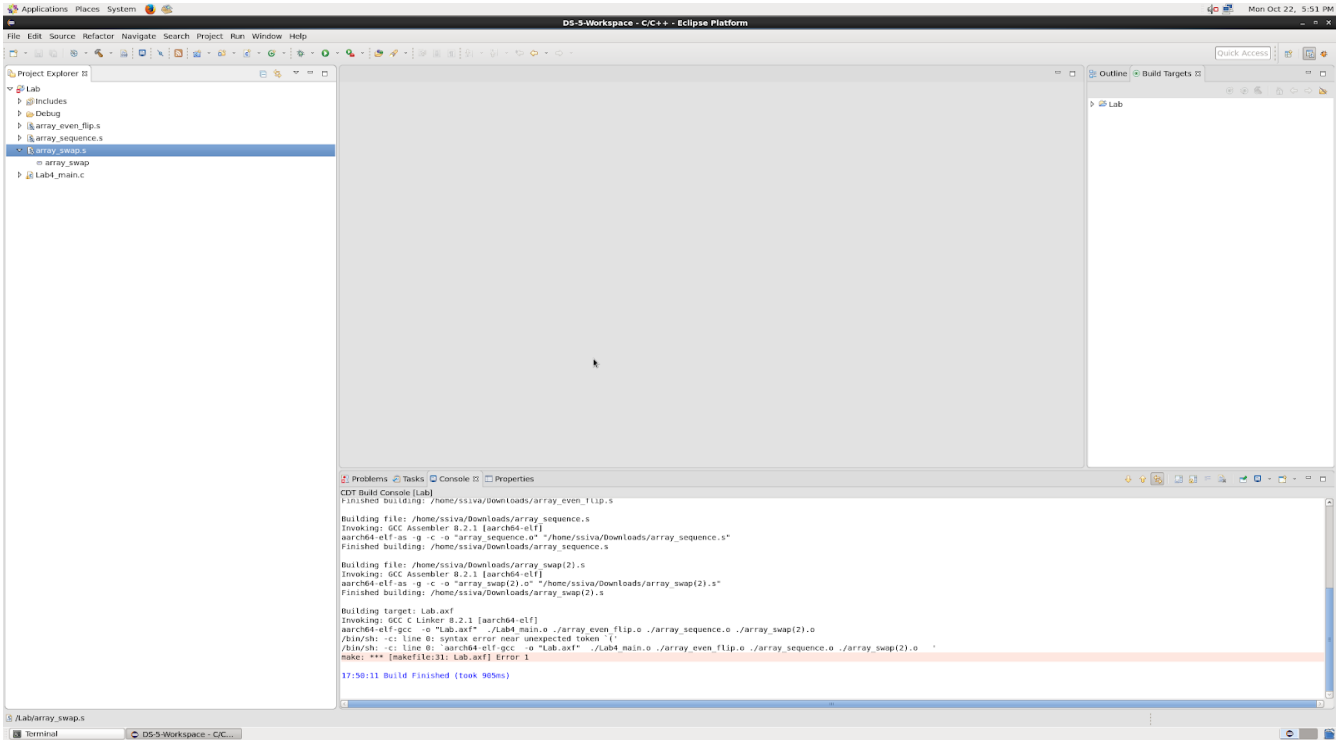


Fig4

Fig 5

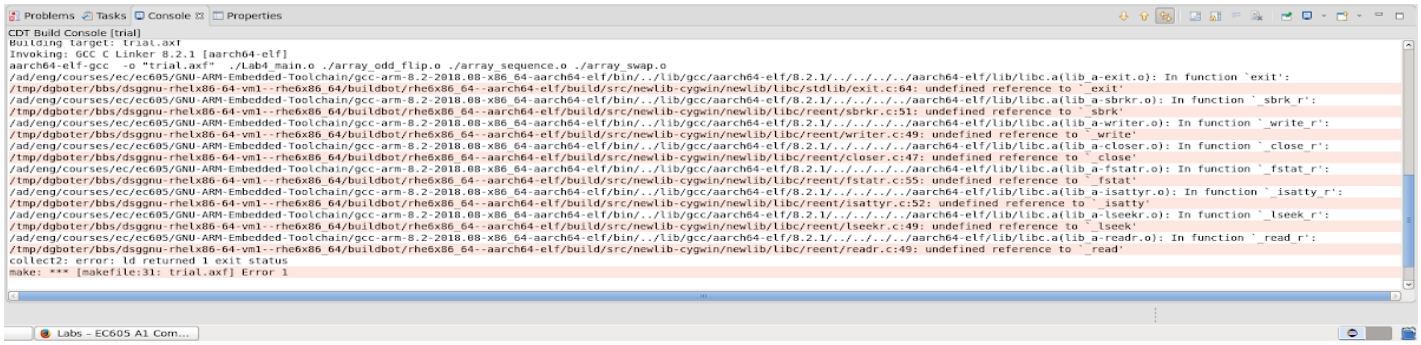


Fig 6

9. A successful build, on the other hand, should display a “Build Finished” message (Fig 7).
Note: In case you encounter a “.axf” error, refer back to points 5. and 6, to see if that fixes the problem.

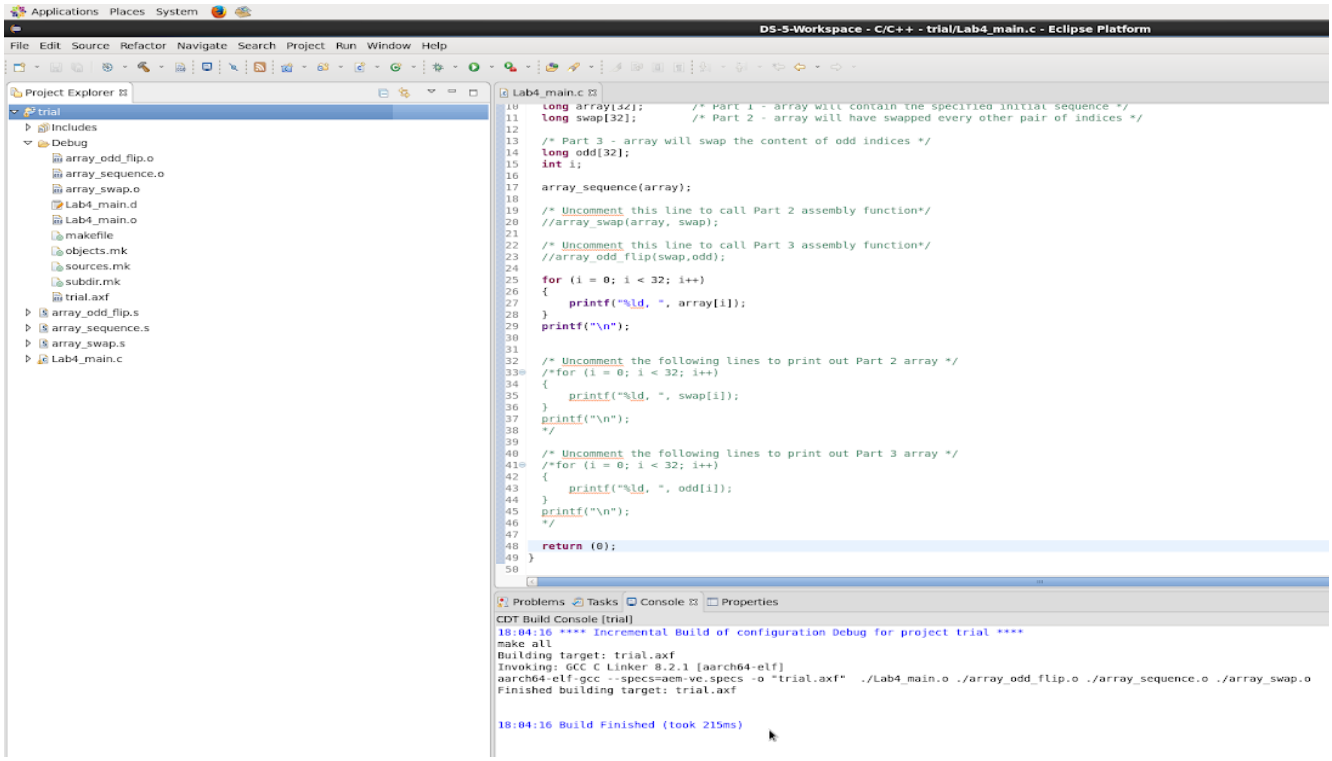


Fig 7

10. Once you’ve navigated the build process, it’s time to “debug” your code. Detailed instructions on debugging the code have already been given in the basic hello-world tutorial. In short, you would have to Click on Run -> Debug Configurations -> Select Bare Metal Debug -> Select the right .axf file -> click on “OK”.

11. You should now see the Debug screen. Once the process stabilizes, click on the green arrow (Continue(F8)) to execute the code, as indicated by the cursor in Fig 8.

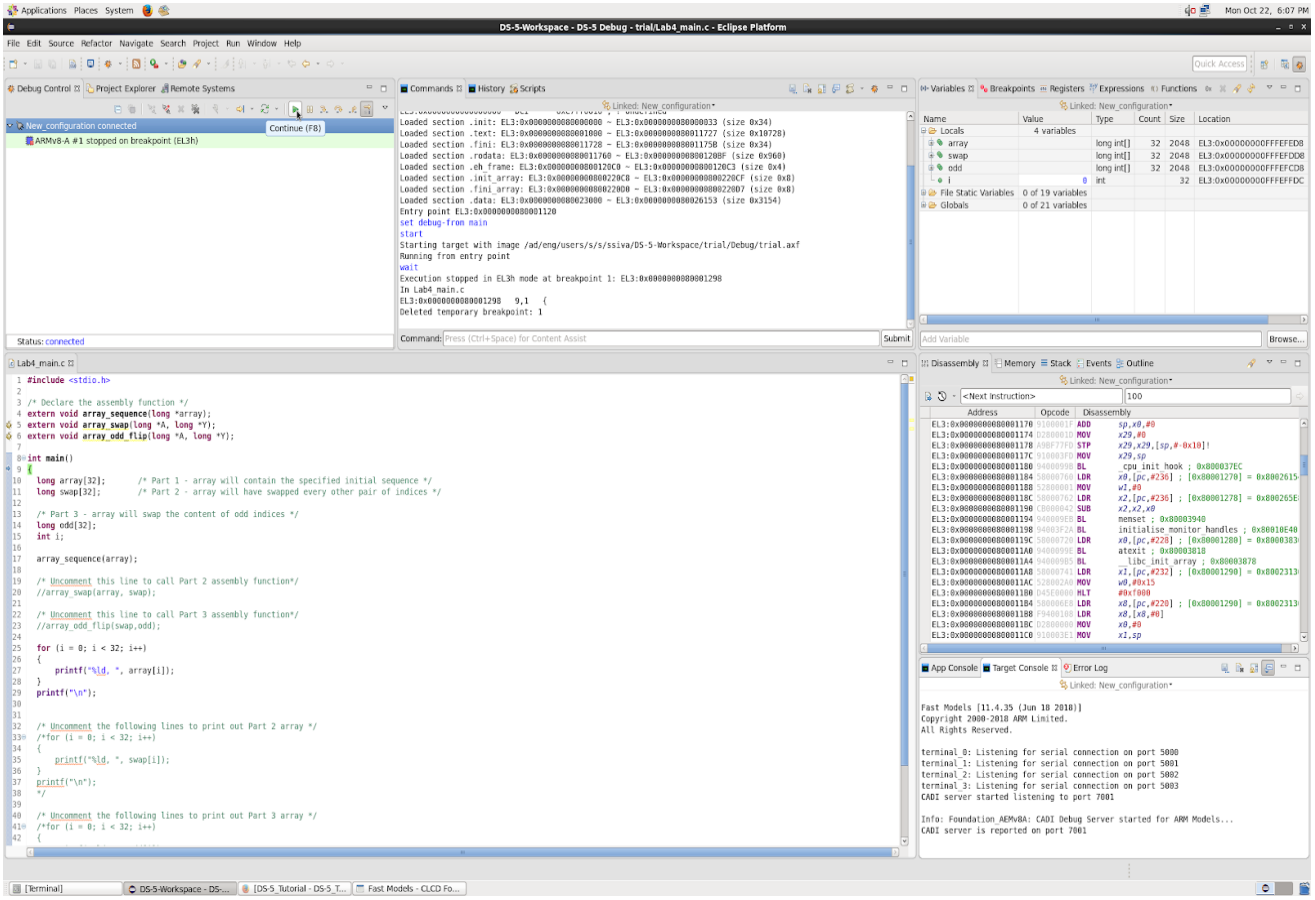


Fig 8

12. If all works fine, you should hopefully see no errors in the “Target Console” (bottom right, Fig 9). It is also common to see a “source not found” warning (Fig 10) upon executing your code. You can disregard this message and continue to work on your code.

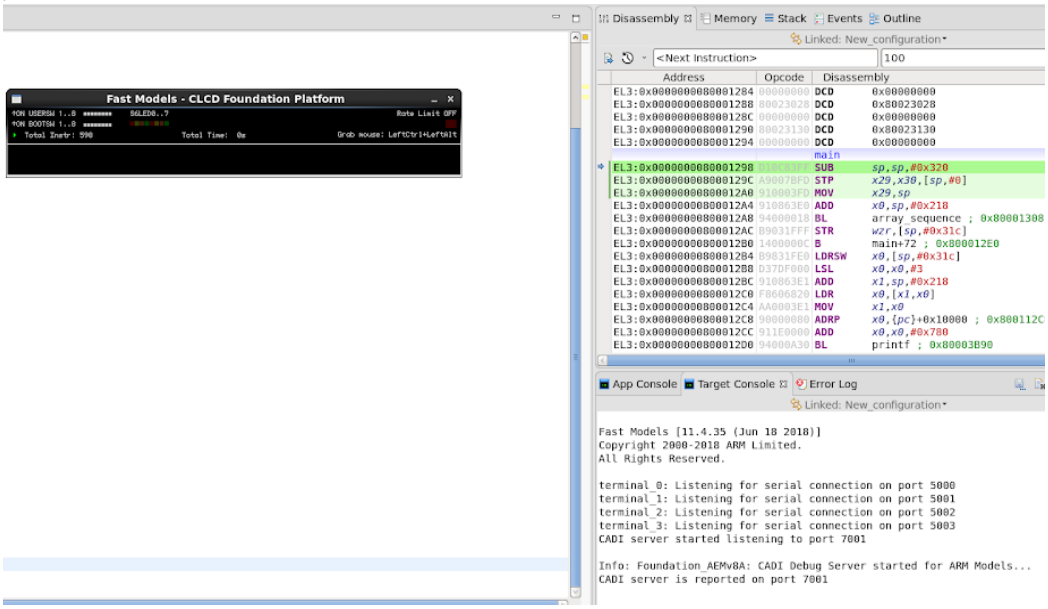


Fig 9

Fig 10

13. If you have instructed the .c file to output results, you should hopefully see them on your “target console” (Fig 11).

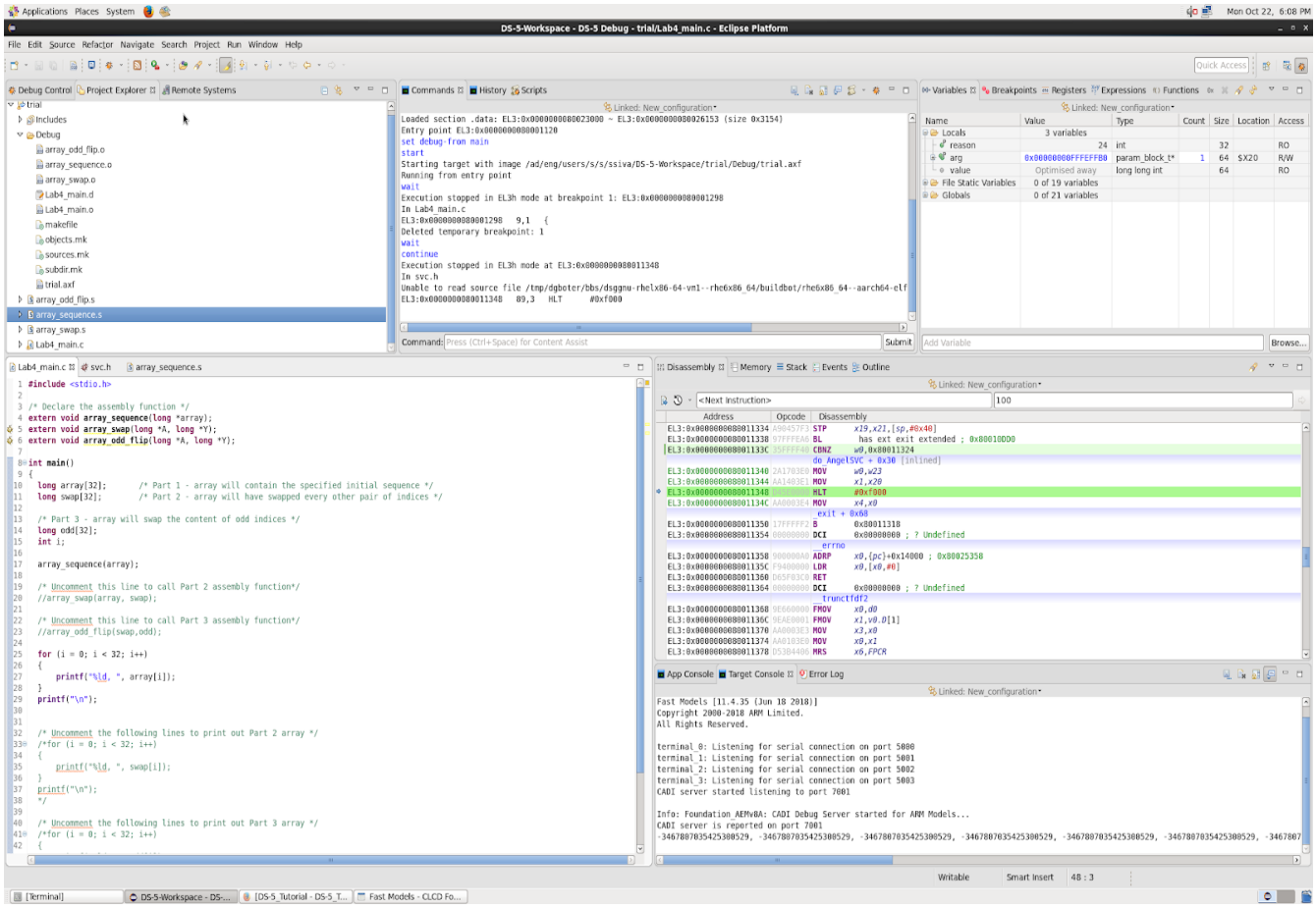


Fig 11