## Lab 4: SRIO Single Unfair

### Purpose

The purpose of this lab is to demonstrate a “real” application using Type 11 SRIO.

### Project Files

The following files are used in this lab:

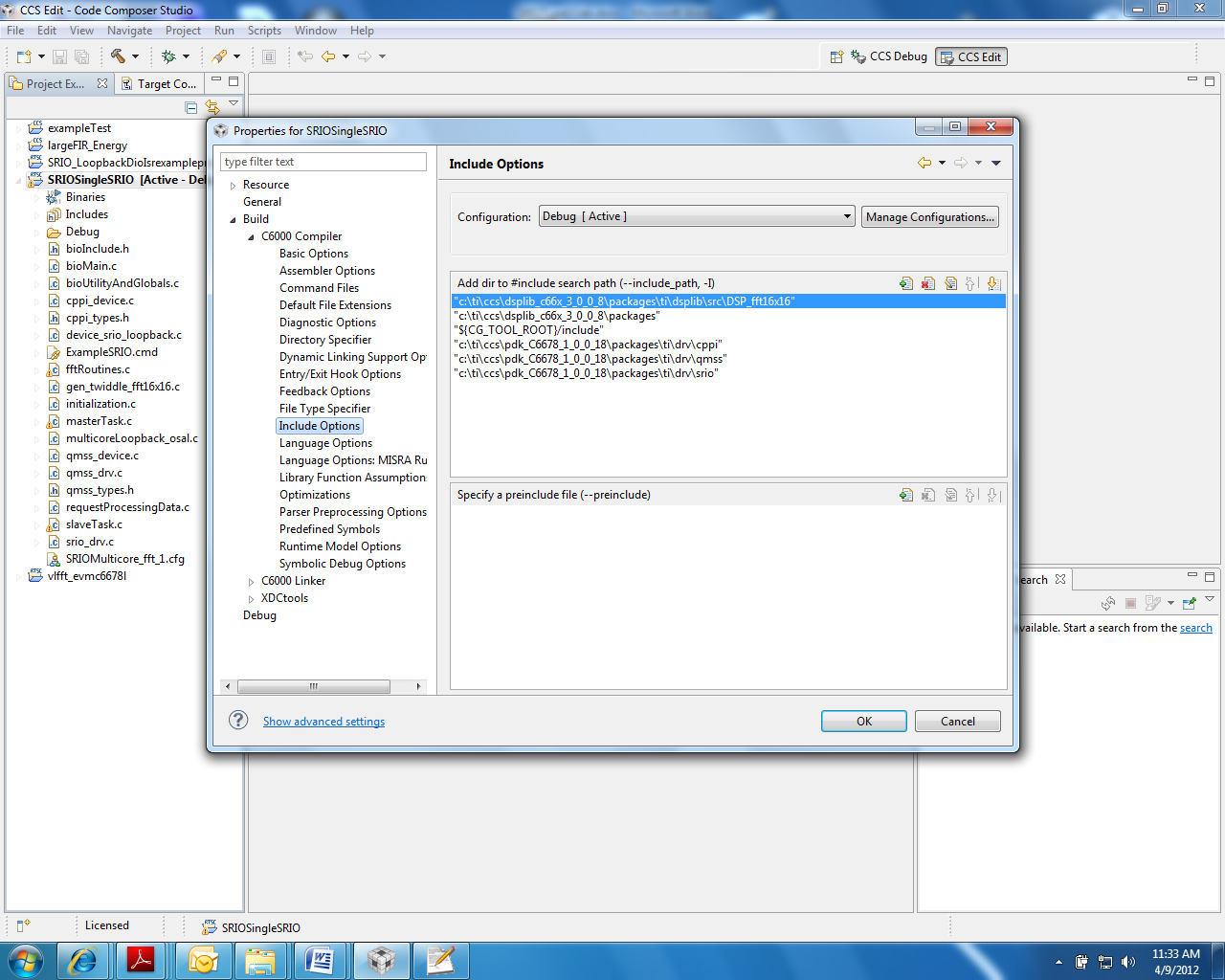
* bioInclude.h
* bioMain.c
* bioUtilityAndGlobals.c
* cppi\_device.c
* cppi\_types.h
* device\_srio\_loopback.c
* ExampleSRIO.cmd
* fftRoutines.c
* gen\_twiddle\_fft16x16.c
* initialization.c
* masterTask.c
* multicoreLoopback\_osal.c
* qmss\_device.c
* qmss\_drv.c
* qmss\_types.h
* requestProcessingData.c
* slaveTask.c
* srio\_drv.c
* SRIOMulticore\_fft\_1.cfg

### TASK 1: Load the Project

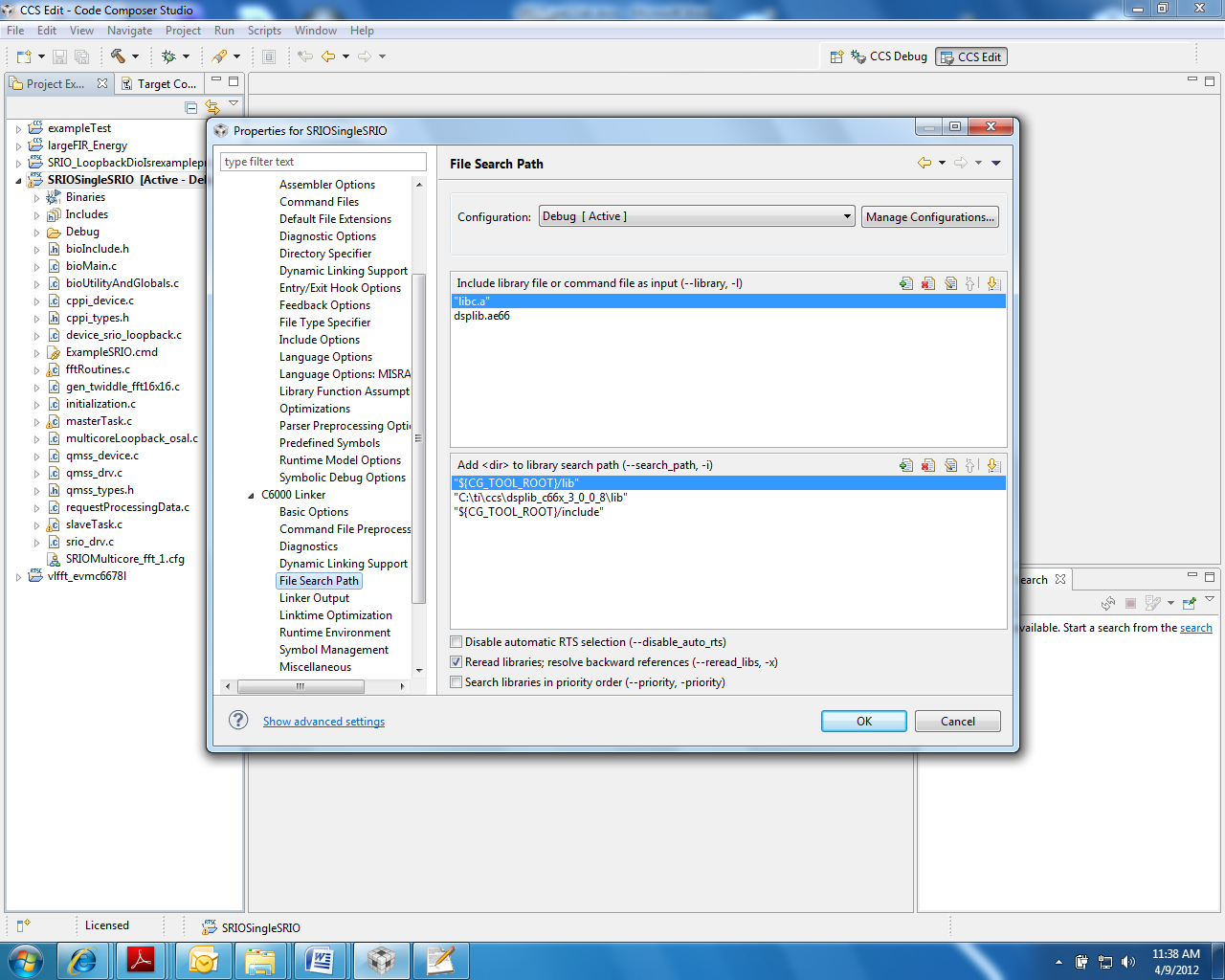
1. Copy the project folder to your local development environment:

\\GTSNOWBALL\gguser\training\Internal TTT 04-10-12\LABS\SRIO\_fft\_singleSRIO\_working

1. Start CCS and import the project.
2. Update the include path in the Project Properties. Refer to the path as defined below. You must modify all paths to either an absolute address where you put your tools, or refer to a relative address. You can use {PDK\_INSTALL\_PATH} or similar if it is defined.

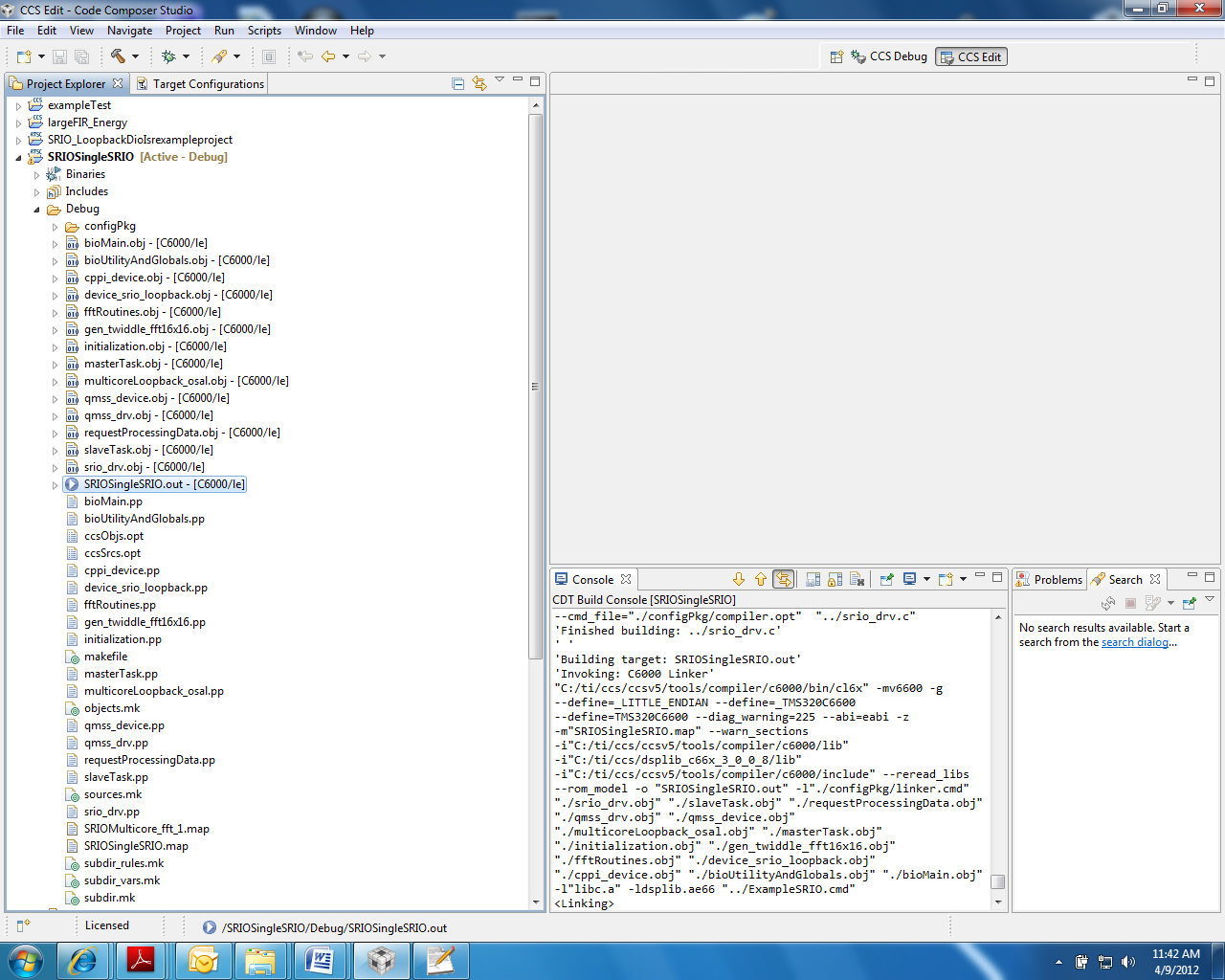


1. Do the same for the Linker File Search Path, as shown below.



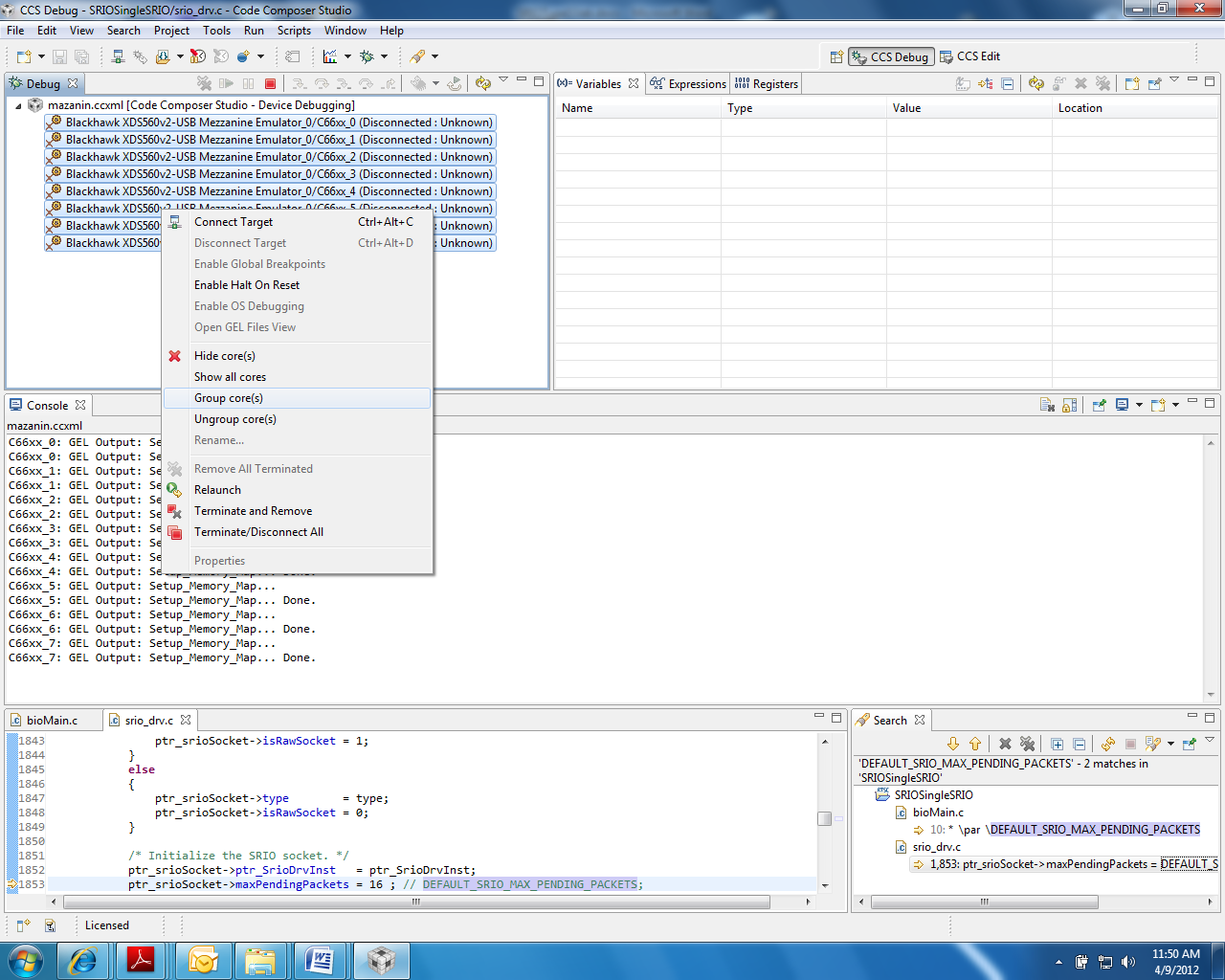
### TASK 2: Build the Application

1. Clean the build.
2. Build the project.
3. Verify that the executable (.out) was built by looking at the debug directory (assuming the build configuration is debug configuration).

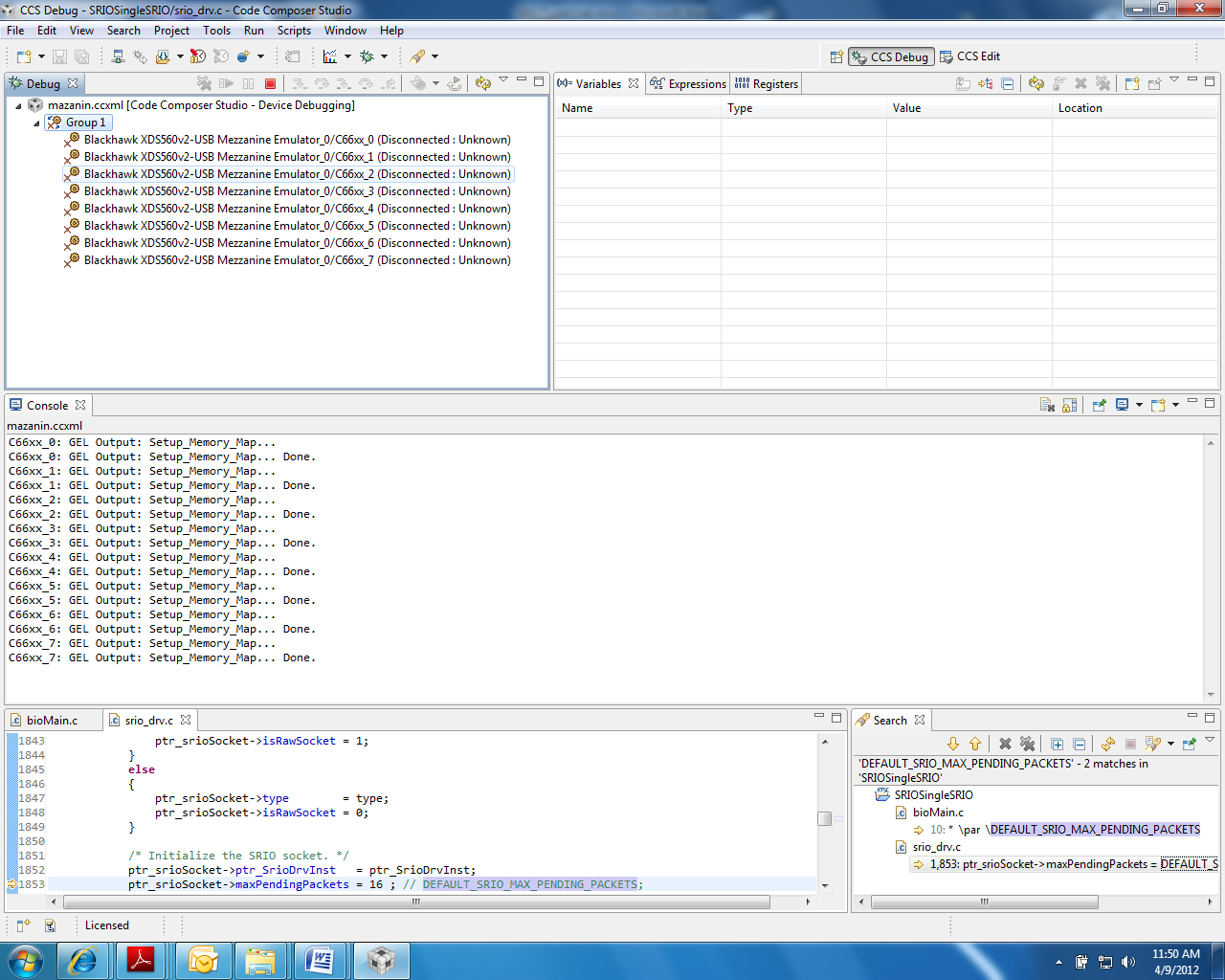


### TASK 2: Launch the Target

1. Power on the EVM, connect the USB cable to the emulator, wait for the EVM to finish boot (the red light is ON).
2. Launch your debugger.
3. Group all cores into one group as follows:



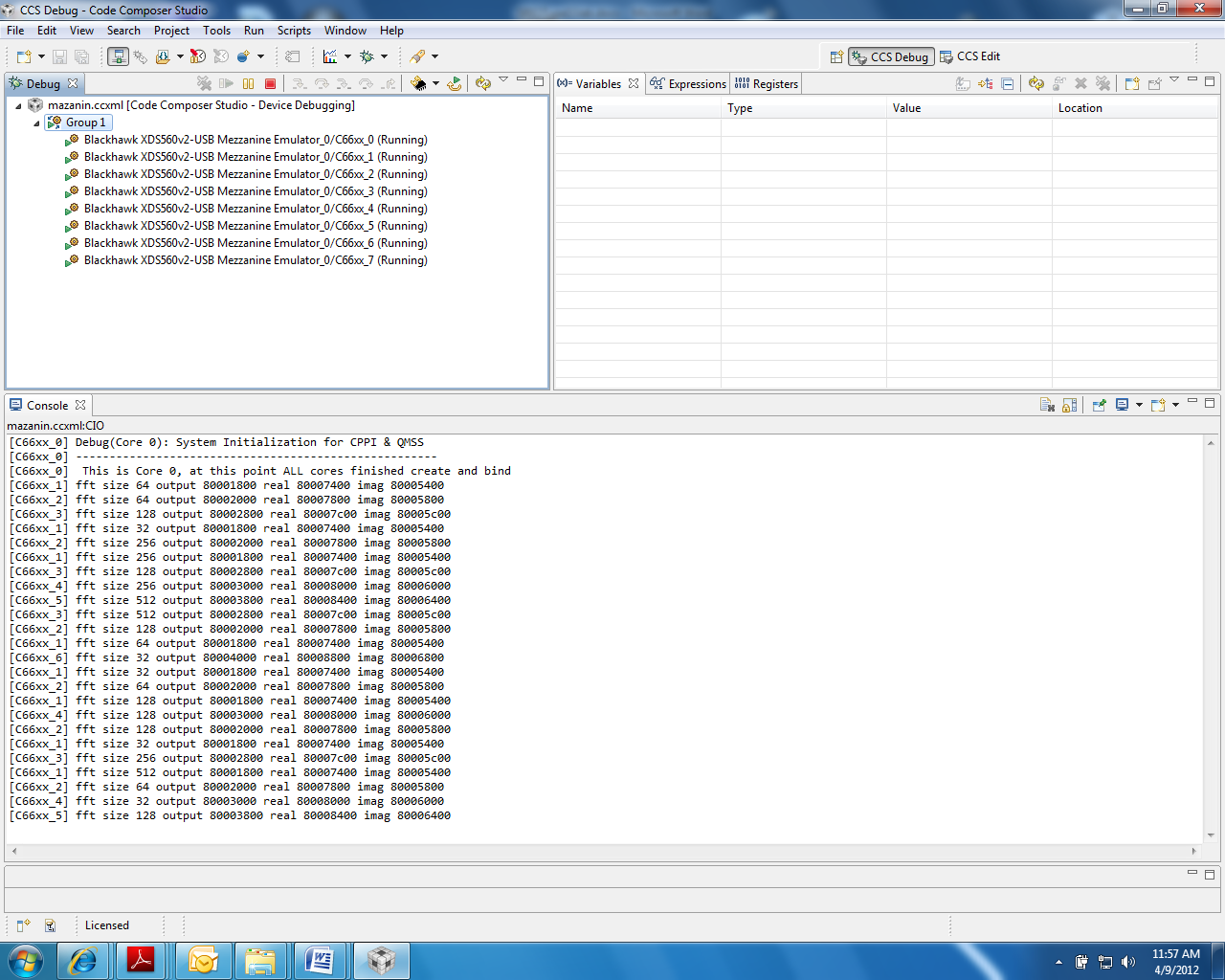
After grouping, Group 1 is defined and displayed as shown below:



### TASK 3: Load and Run

1. Select Group1, and connect all cores in the group by one of the three ways:
   1. From the RUN menu, select Connect Target.
   2. Right click on the group name and choose Connect Target.
   3. Click the Connect Target icon.
2. Load the code to all cores in the group:
   1. From the RUN menu, select Load.  
      OR
   2. Click the Load icon.
3. Run the code in one of the following ways:
   1. Press F8.
   2. From the RUN menu, select Resume.
   3. Click on the Resume icon (green arrow).

The output results appear as follows:



1. Observe the results, then suspend the run:
   1. From the RUN menu, choose Suspend  
      OR
   2. Click on the Suspend icon (the yellow “pause” lines)

### TASK 4 (optional): Debug the Project

When you have finished the previous task, you can optionally download the same project with some embedded bugs and then debug the code.

1. Load the project to your development environment:

\\GTSNOWBALL\gguser\training\Internal TTT 04-10-12\LABS\SRIO\_fft\_singleSRIO\_WithBugs

1. Start CCS and import the project.

Challenge Question: Can you debug the code and make it build and run properly?

1. There are two bugs. One bug is in the build process. Something is missing from the .cfg file.  
   The second bug is in the run. The code was originally developed for 6670 and then converted to 6678.

*Maintain Blank Page Here*