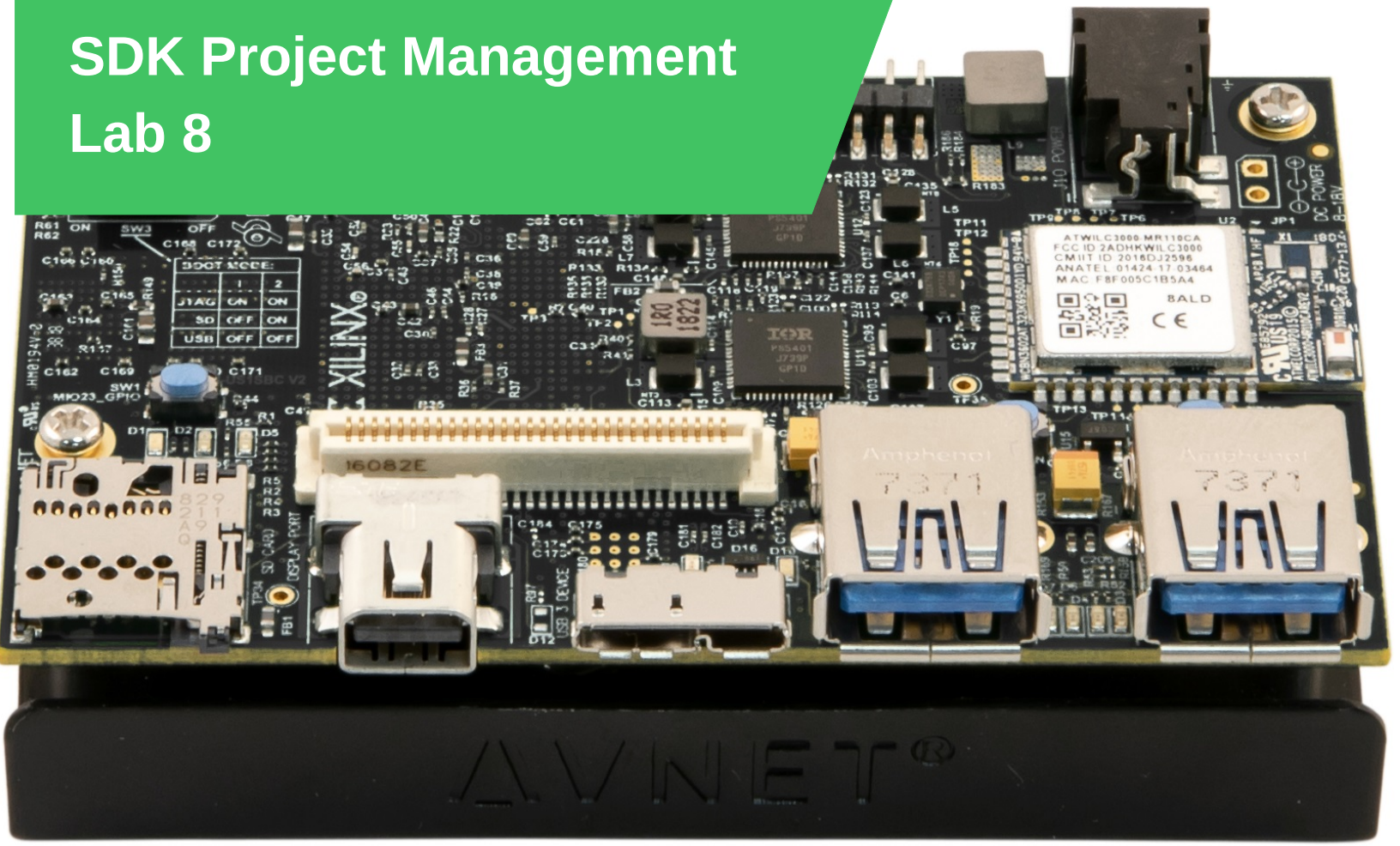
Avnet Technical Training Course

SDK Project Management
Lab 8



Tools:	2019.1
Training Version:	v11
Date:	July 2019

Lab 8 Overview

You should not share or archive your workspace simply by zipping it up and sending it off. Workspaces are just a container of software projects, and your preferences are user- and location-specific. If you simply copy the workspace to a different location in the system, it is not guaranteed to work.

Software projects, including board support packages, and software applications that you create in your workspace, **can** be shared with other team members or archived into a source control system. To do this, share or archive a collection of source files and SDK metadata files in the project directory. This lab will show you the steps how to do this.

Lab 8 Objectives

When you have completed Lab 8, you will know how to:

- Create a complete project archive
- Create a new, duplicate project by importing your archive
- Create a new application, and import the sources for that application

Experiment 1: Create a Complete SDK Project Archive

This experiment will show you the proper method for archiving a project for sharing. It is important to note that this is a multi-step process.

I. Archive the project sources in an archive file

II. Archive the Run/Debug configurations in an archive directory, if desired

III. Archive debug breakpoints, if desired

IV. Preserve SDK (including repository settings) preference files, if desired

Experiment 1 General Instruction:

Export the SDK projects to a single-file archive, including the hardware platform, BSP, and all applications.

Experiment 1 Step-by-Step Instructions:

1. In SDK, select **File → Export**.
2. Select **General → Archive File**. Click **Next >**.

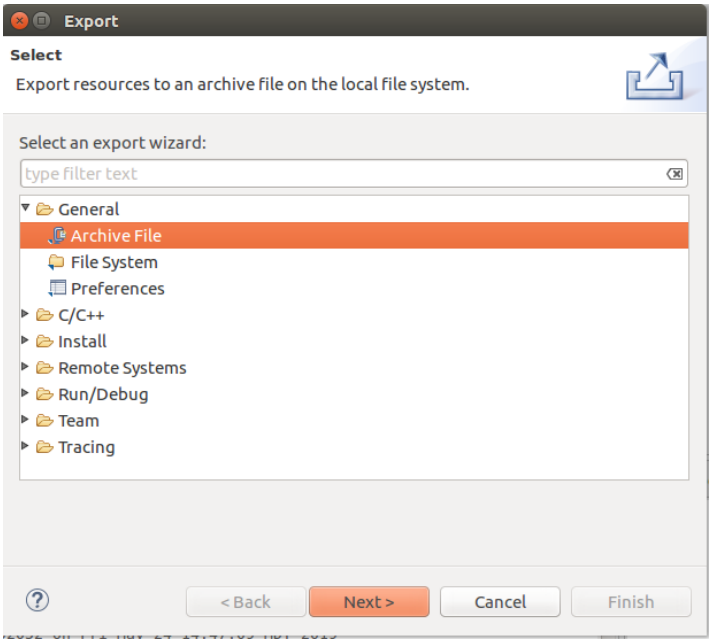


Figure 1 – Export Archive File

3. Select all the checkboxes or click the **Select All** button. Click **Browse** and name the archive 'Lab08_project_export.zip' in the `ZynqSW\2019_1` Speedway directory. Click **Save**. Click **Finish**.

*****If this is unsuccessful go back to the Project Explorer right click on ZynqHW and click Refresh. Then return to Step 1.*****

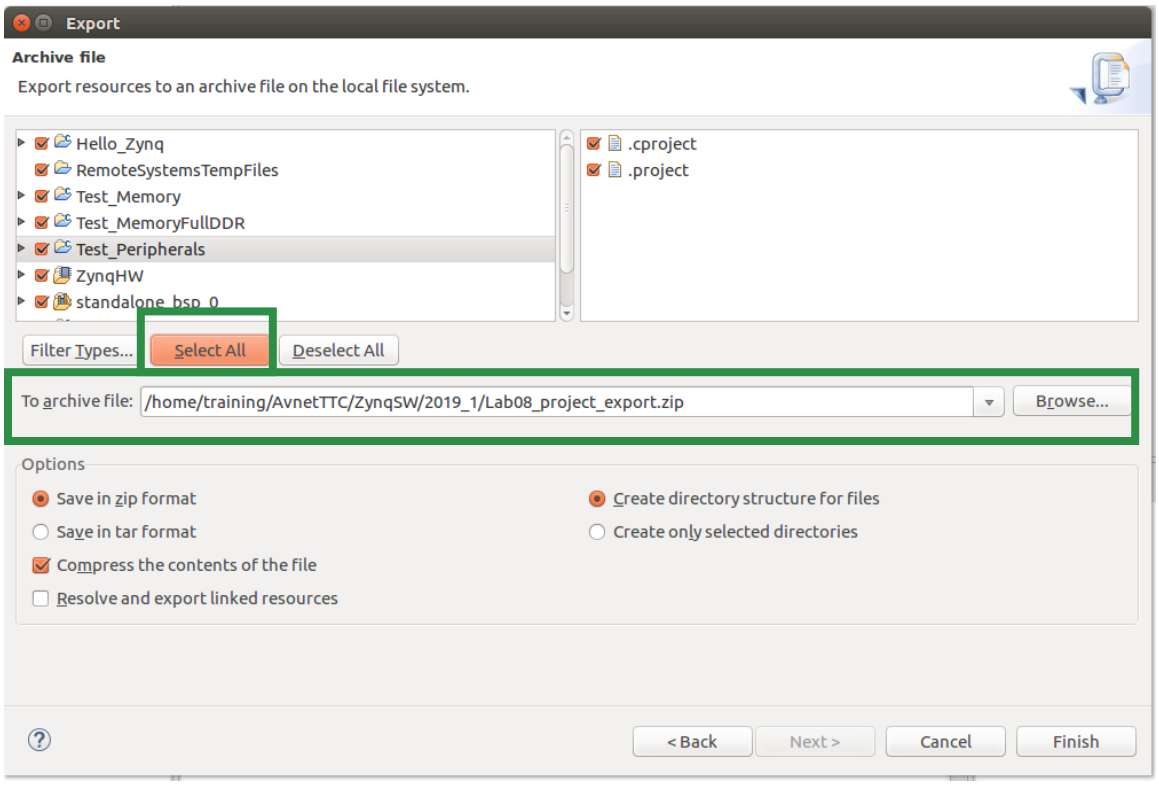


Figure 2 – Export All

4. Select **File → Export**. Select **Run/Debug → Launch Configurations**, then click **Next >**.

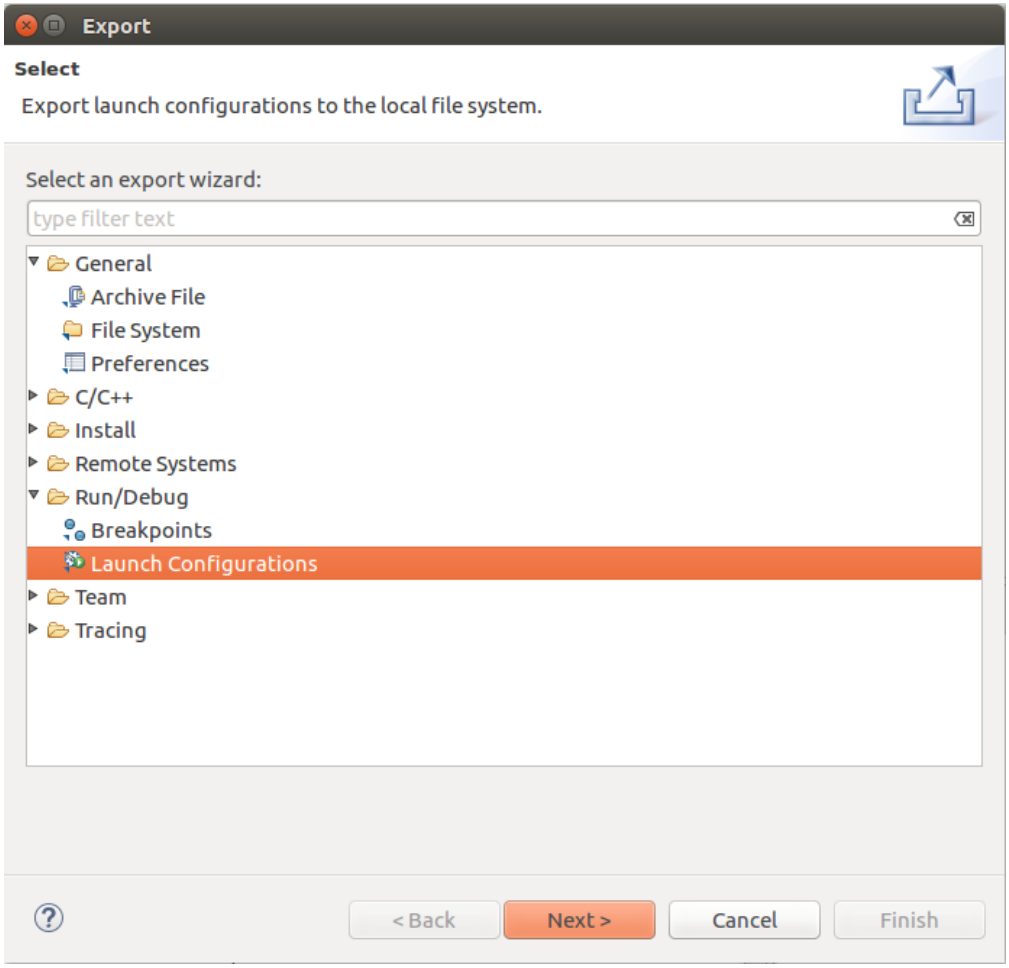


Figure 3 – Export Run/Debug Launch Configurations

5. Click the **Select All** button. Browse to `/home/training/AvnetTTC/ZynqSW/2017_4/`. Click **Make New Folder**, and name it `Lab08_config_export`. Click **OK** to accept the entered name and then click **Finish** on the Export Launch Configurations window.

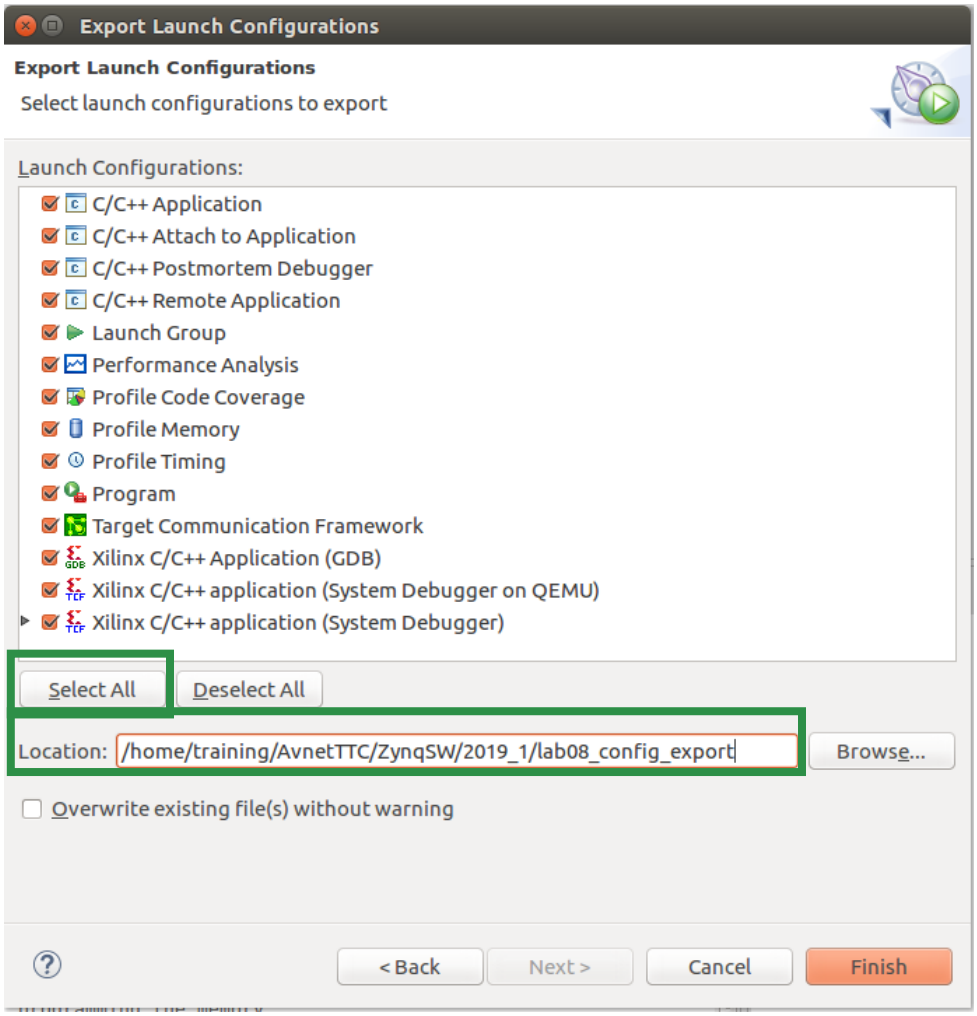


Figure 4 – Export Launch Configurations

Although our Debug exercise did not include a lot of breakpoint setting, you can imagine that some engineers invest a lot of time in setting up a debug environment. If they wanted to transfer that environment, including breakpoints, those must be exported explicitly.

6. Select **File → Export**. Select **Run/Debug → Breakpoints**, then click **Next >**.

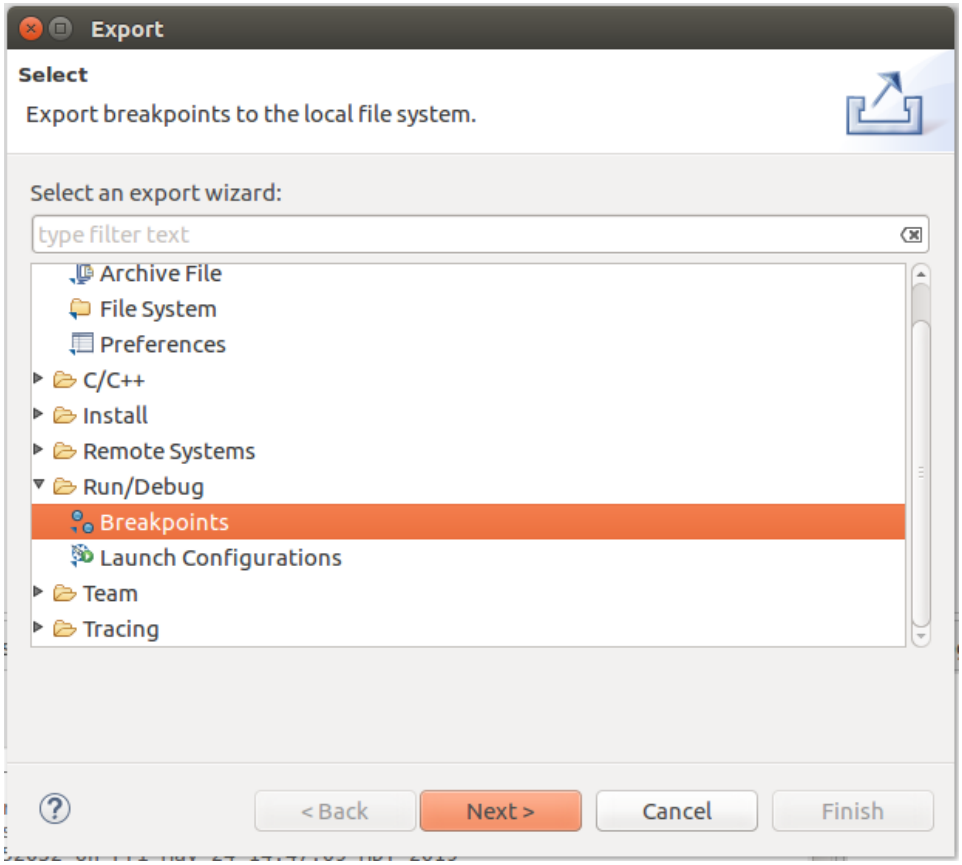


Figure 5 – Export Breakpoints

7. Click the **Select All** button. Browse to `/home/training/AvnetTTC/ZynqSW/2019_1/`. Name the file `Lab08_breakpoint_export.bkpt`. Click **Save** to accept the entered name and then click **Finish** on the Export window.

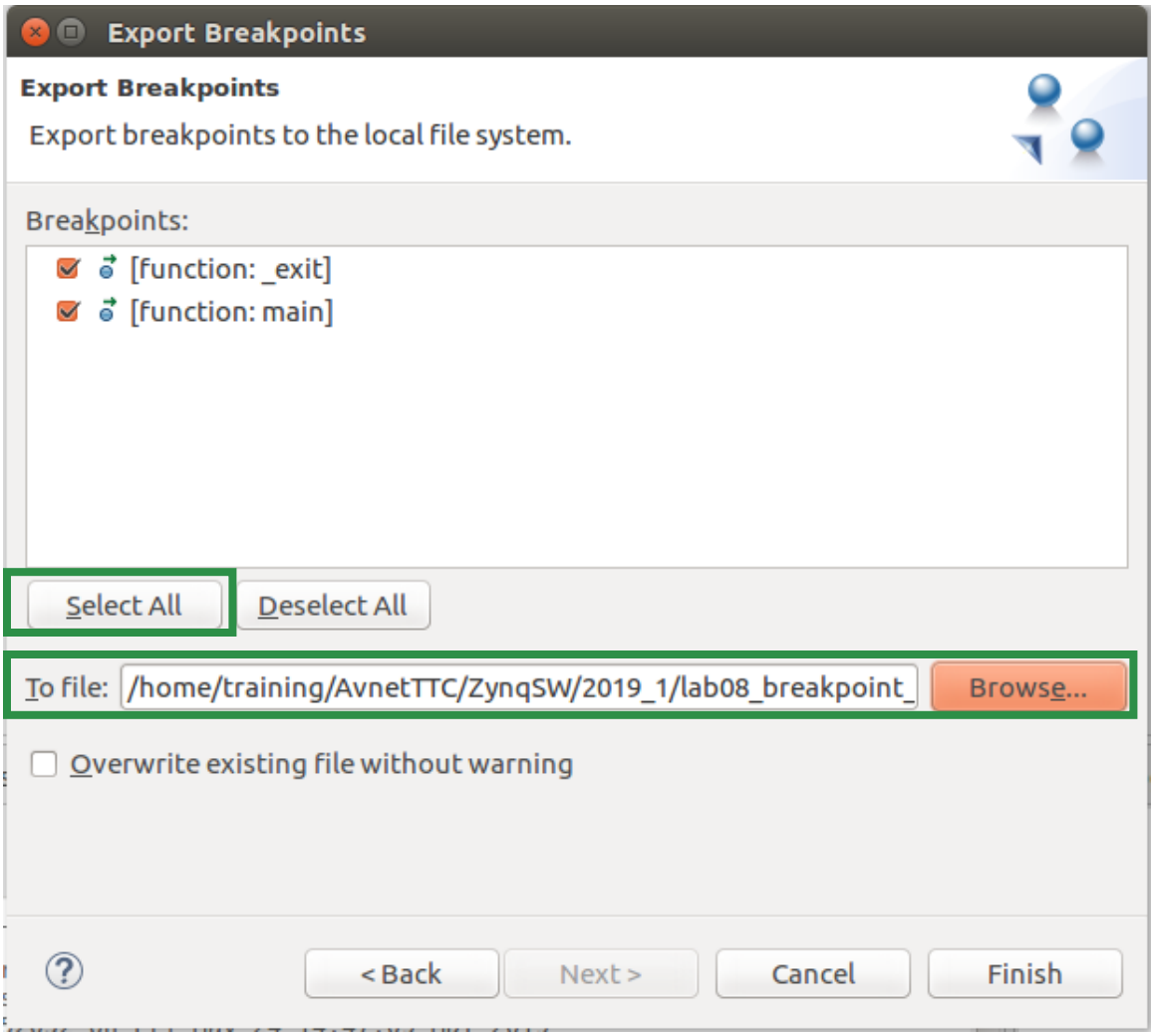


Figure 6 – Export Breakpoints

8. Create a folder labeled **Lab08_repositories_export** in your /home/training/AvnetTTC/ZynqSW/2019_1 directory.

9. A manual step is required if you want a complete backup of workspace preferences. Your project currently makes use of local repositories, so it is good practice to perform this activity since repository settings are not captured in any of the Exports that you have already created. A specific **File → Export** menu option does not exist to capture the repositories. However, this information is contained in underlying settings files. You should save these file as well as any repositories referenced. Browse in a File Explorer from your WorkSpace to:

   ```
   .metadata\.plugins\org.eclipse.core.runtime\.settings
   ```

10. Make a copy of the files in this folder and copy them into the Lab08_repositores_export folder you just created. Incidentally, these files also contain the customizations that you made when creating the new linker scripts for your applications. Restoring these files will also allow the tools to remember your Linker Script Generator settings.

11. Browse to `/home/training/AvnetTTC/ZynqSW/2019_1/` in Windows Explorer to view what was created. Note the two files and two directories that were created. You can archive all four of these items together, along with any repositories you were using, to share with a colleague to recreate your SDK workspace.

Figure 7 – Archive Files/Directory Created

Question:

Answer the following question:

- *What is the advantage of exporting your Workspace items as opposed to simply zipping the workspace?*

Experiment 2: Import a Shared Project Archive

Now that you've created an archive, we'll next learn how to import an archive. We will assume that you now are the recipient of the archive files and directories created in Experiment 1. What steps are necessary to duplicate the workspace in a different location?

Experiment 2 General Instruction:

Create a new workspace and import all the previous projects from the lab8_project_export.zip archive.

Experiment 2 Step-by-Step Instructions:

1. Select **File → Switch Workspace → Other**.
2. Call it 'New_Workspace' then click **OK**.

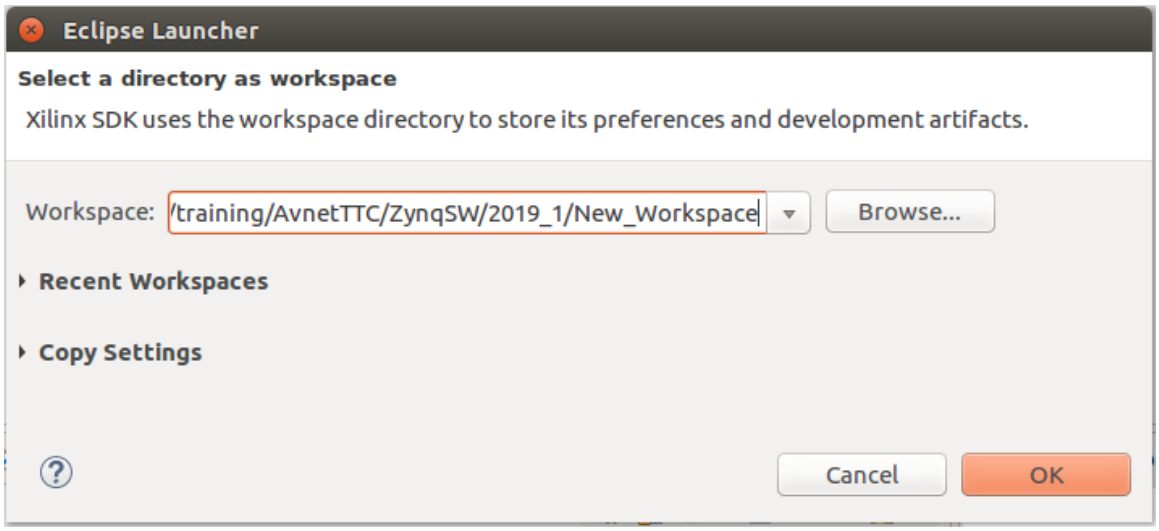


Figure 8 – New Workspace Created

3. Close the *Welcome* screen, if necessary. You should observe that you have a new workspace with no projects – no hardware platform, no BSP, and no applications.

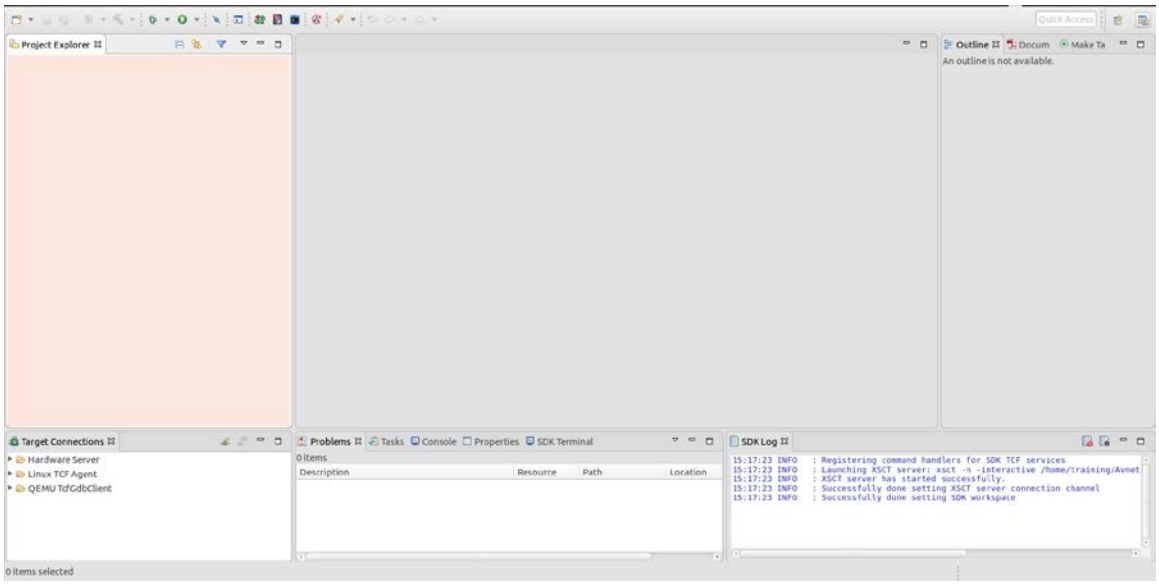


Figure 9 – New Workspace is Empty

4. Select **File → Import**. Select **General → Existing Projects into Workspace**, then click **Next >**.

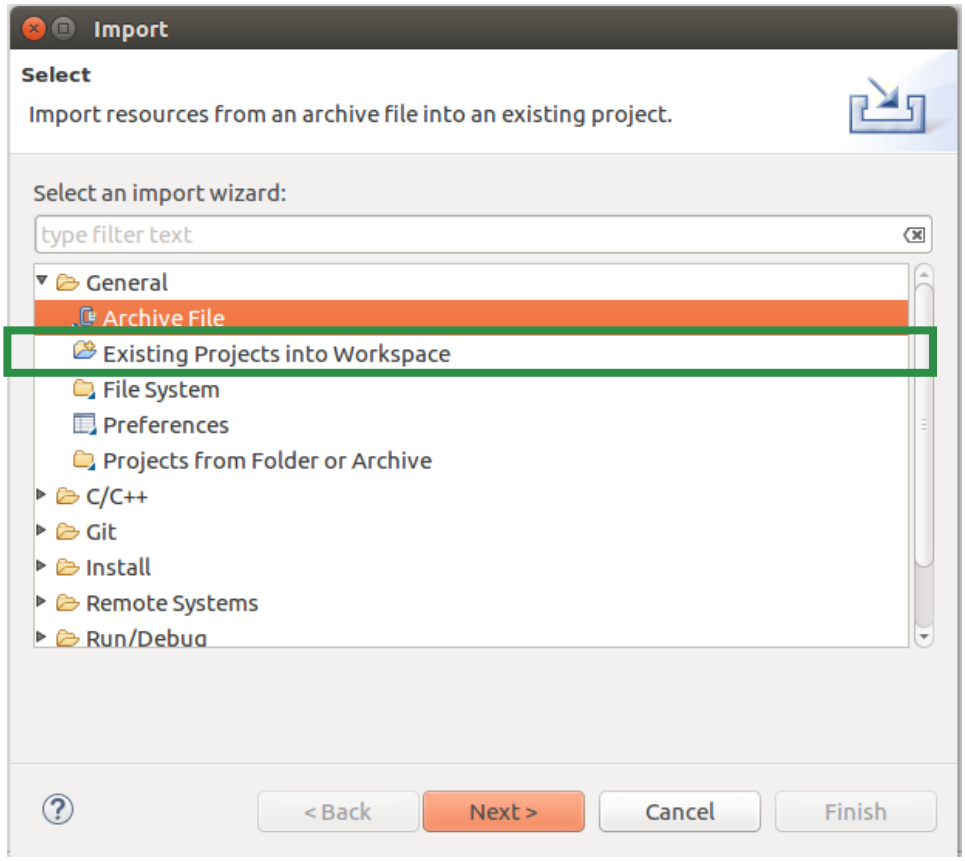


Figure 10 – Import from Archive File

5. Select the radio button for **Select archive file**. Browse, select, and open `Lab08_project_export.zip`. Click the **Select All** button. Click **Finish**.

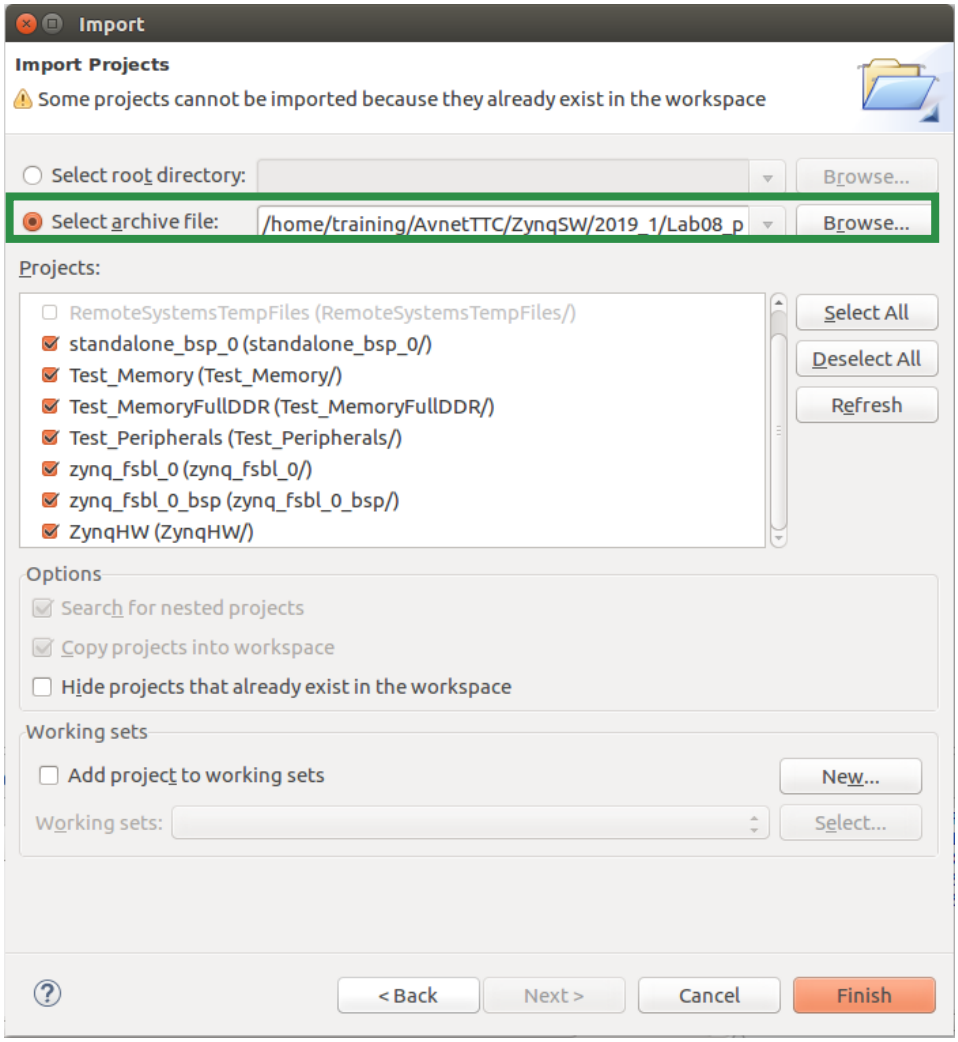


Figure 11 – Import from Archive File

6. You should now see your Projects restored in *Project Explorer*, as shown below.

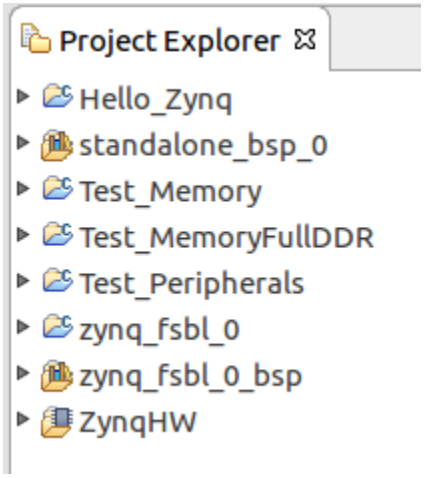


Figure 12 – Projects Restored

7. All of the projects should import and then automatically build.

8. Select **File → Import**. Select **Run/Debug → Launch Configurations**, then click **Next >**.

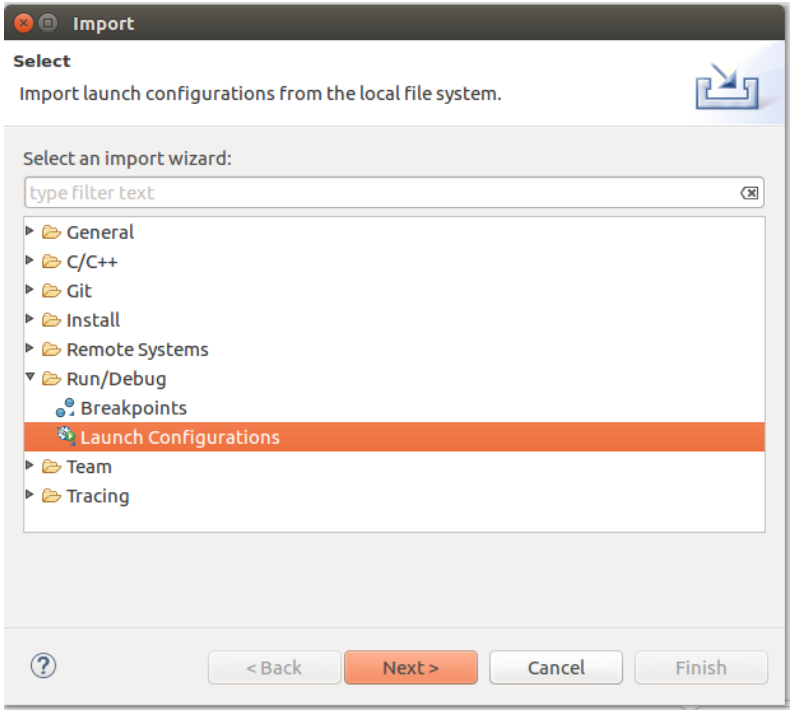


Figure 13 – Import Launch Configurations

9. Browse to `/home/training/AvnetTTC/ZynqSW/2019_1/Lab08_config_export` and click **OK**. Select the **Lab08_config_export** checkbox. Verify that the configurations on the right also have their checkboxes checked. Click **Finish**.

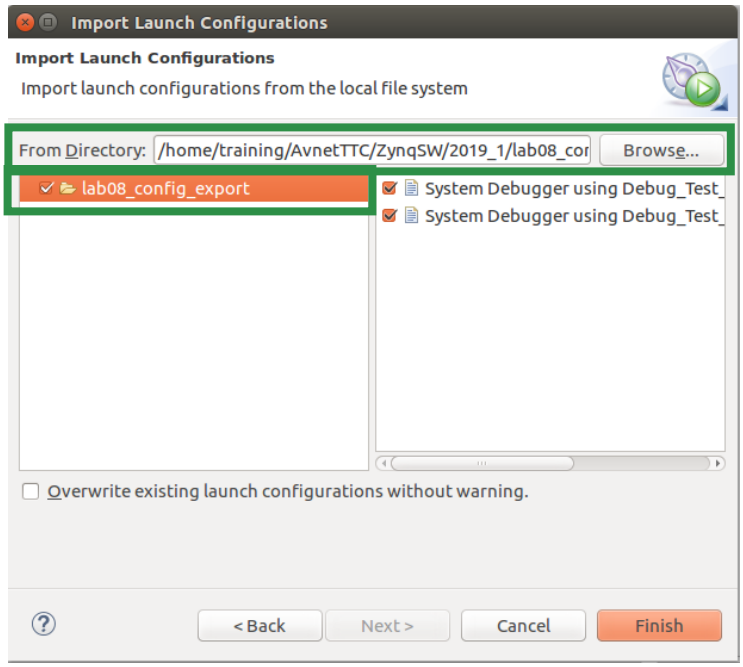


Figure 14 – Import Launch Configurations

10. Verify the Configurations were imported by selecting **Run → Debug Configurations**. Then expand both **Xilinx C/C++ application** items (GDB and System Debugger). You should see four Debug configurations (Under System Debugger). Click **Close**.

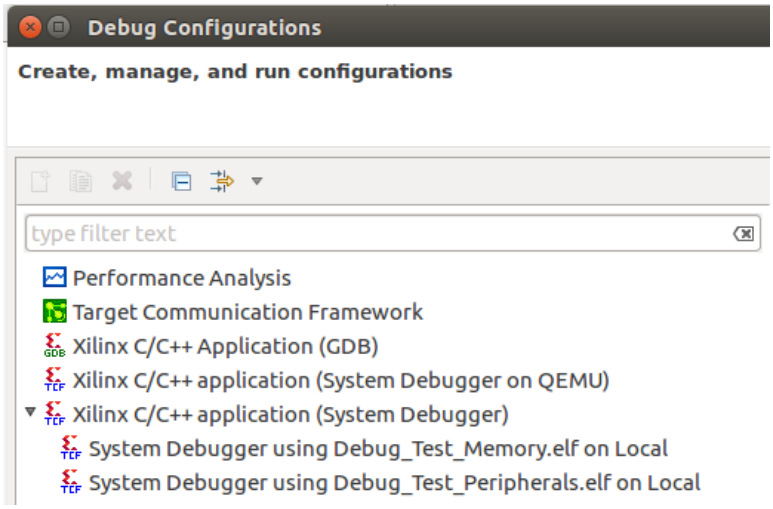


Figure 15 – Run/Debug Configurations Restored

Next, we'll restore the breakpoints.

11. Select **File → Import**. Select **Run/Debug → Breakpoints**, then click **Next >**.

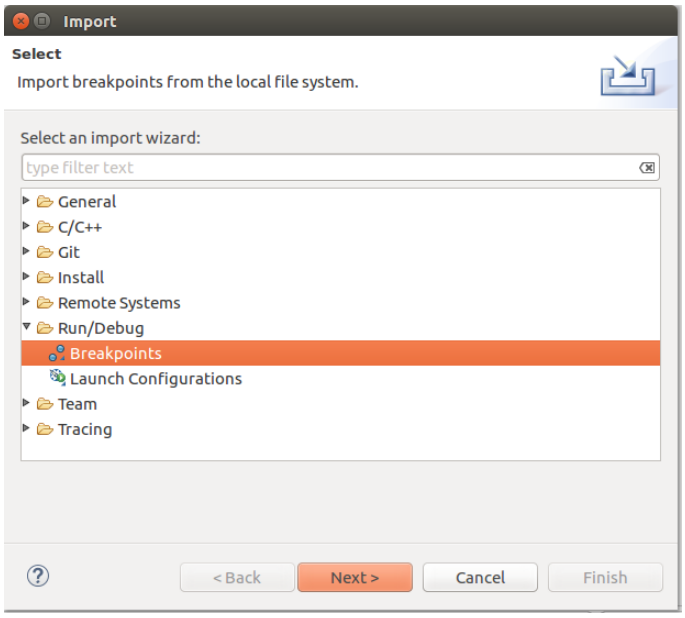


Figure 16 – Import Breakpoints

12. Browse to the `/home/training/AvnetTTC/ZynqSW/2017_4/` folder, select `Lab08_breakpoint_export.bkpt`, and click **Open**. Click **Finish**.

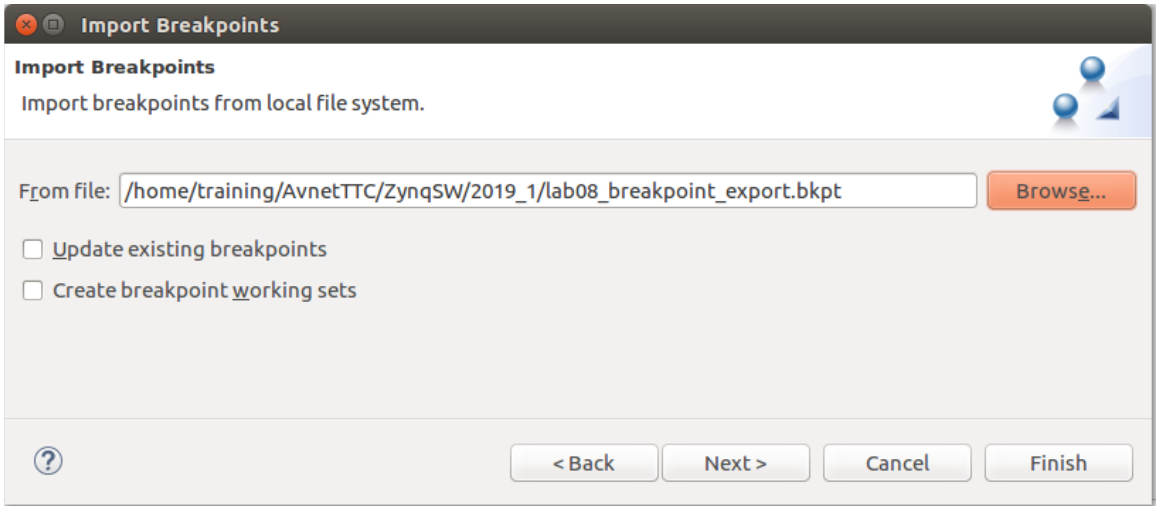


Figure 17 – Import Breakpoints

13. Verify the Breakpoints were imported by selecting **Window → Perspective → Open Perspective → Debug**. Click on the *Breakpoints* tab. You should see three breakpoints.

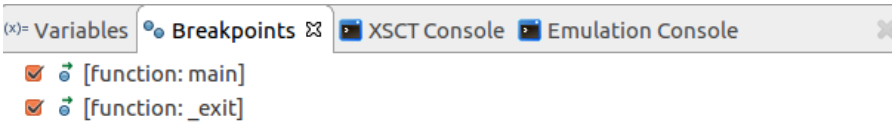


Figure 18 – Run/Debug Configurations Restored

14. Close the Debug perspective by right-clicking on it (in the upper right-hand corner) and selecting **Close**.

Next, we'll restore the project preferences, which would include the repositories (if we had them) as well as our previous Linker Script Generator settings.

15. Since we're restoring a settings file for SDK that gets read at launch, SDK cannot be open during this restore. Close SDK.

16. Browse in Windows Explorer to the `Lab08_repositories_export` folder. Copy and paste the files to the following folder in the **New_Workspace**:

    ```
    .metadata\.plugins\org.eclipse.core.runtime\.settings
    ```

    Note: If you had received this archive from a colleague and if the archive included repositories, it would also include file **`com.xilinx.sdk.sw.prefs`**. You would need to copy it as well and then edit the **`com.xilinx.sdk.sw.prefs`** file. You would need to find all repository entries in that file, make sure you have copies or access to those repositories, and then update the repository entries to point to your own copies of the repositories.

17. Launch SDK and open the **New_Workspace**.

18. You can verify that these new settings have taken affect by reviewing the **Generate Linker Script** settings for Hello_Zynq. If the sections all point to ps7_ram_0, then you have successfully restored those preferences. If it points to DDR (the default with no preference file), then it was not successful.

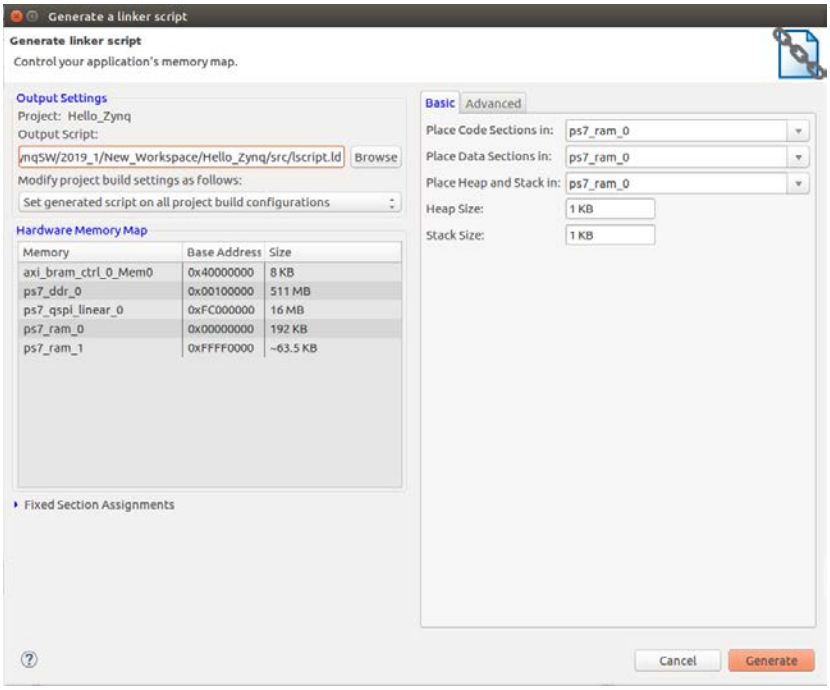
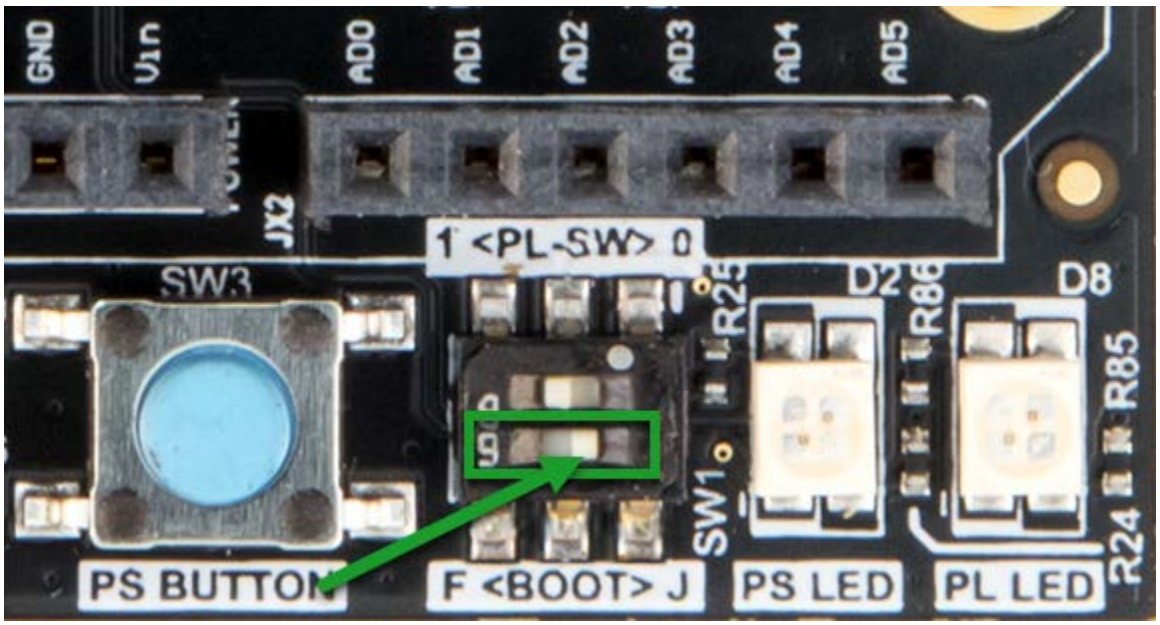


Figure 19 – Preferences Restored as seen in Generate Linker Script

19. You now have a duplicate of your previous workspace. Set your board to Cascaded JTAG MODE.

a. MiniZed – Set outside dip switch towards J (JTag Boot)



MiniZed
Figure 20 –MiniZed Boot Selection

20. Connect the USB-UART-JTAG (J2)
21. Download the PL Bitstream .
22. Download Hello_Zynq to test.
23. If you have more time, go to the Exploring Further section.
24. To avoid confusion, select **File → Switch Workspace** and go back to **SDK_Workspace**.

Exploring Further

If you have more time and would like to investigate more…

- Run the test applications in your regenerated workspace.

This concludes Lab 8.

Revision History

Date	Version	Revision
12 Nov 13	01	Initial release
23 Nov 13	02	Revisions after pilot
01 May 14	03	MicroZed.org Training Course Release
10 Dec 14	04	Update to Vivado 2014.3
07 Jan 15	05	Update to Vivado 2014.4
18 Mar 15	06	Finalize SDK 2014.4
Oct 15	07	Update to SDK 2015.2
Aug 16	08	Updated to SDK 2016.2
Jun 17	09	Updated to 2017.1 for MiniZed + Rebranding
Feb 18	10	Updated to Vivado/SDK 2017.4
July 19	11	Updated to Vivado/SDK 2019.1

Resources

www.minized.org

www.microzed.org

www.picozed.org

www.zedboard.org

www.xilinx.com/zynq

www.xilinx.com/sdk

www.xilinx.com/vivado

www.xilinx.com/support/documentation/sw_manuals/ug949-vivado-design-methodology.pdf

www.xilinx.com/support/documentation/sw_manuals/ug1046-ultrafast-design-methodology-guide.pdf

Answers

Experiment 1

- *What is the advantage of exporting your Workspace items as opposed to simply zipping the workspace?*

If you zip and share your SDK workspace, there is a very good chance that it will not work when opened again. The SDK workspace is full of absolute paths, so unless the recipient unzips the SDK workspace to the exact same location, it won't fully work. It might appear to work initially, but it is likely not going to build properly.

If you export the workspace, it is fully transportable.