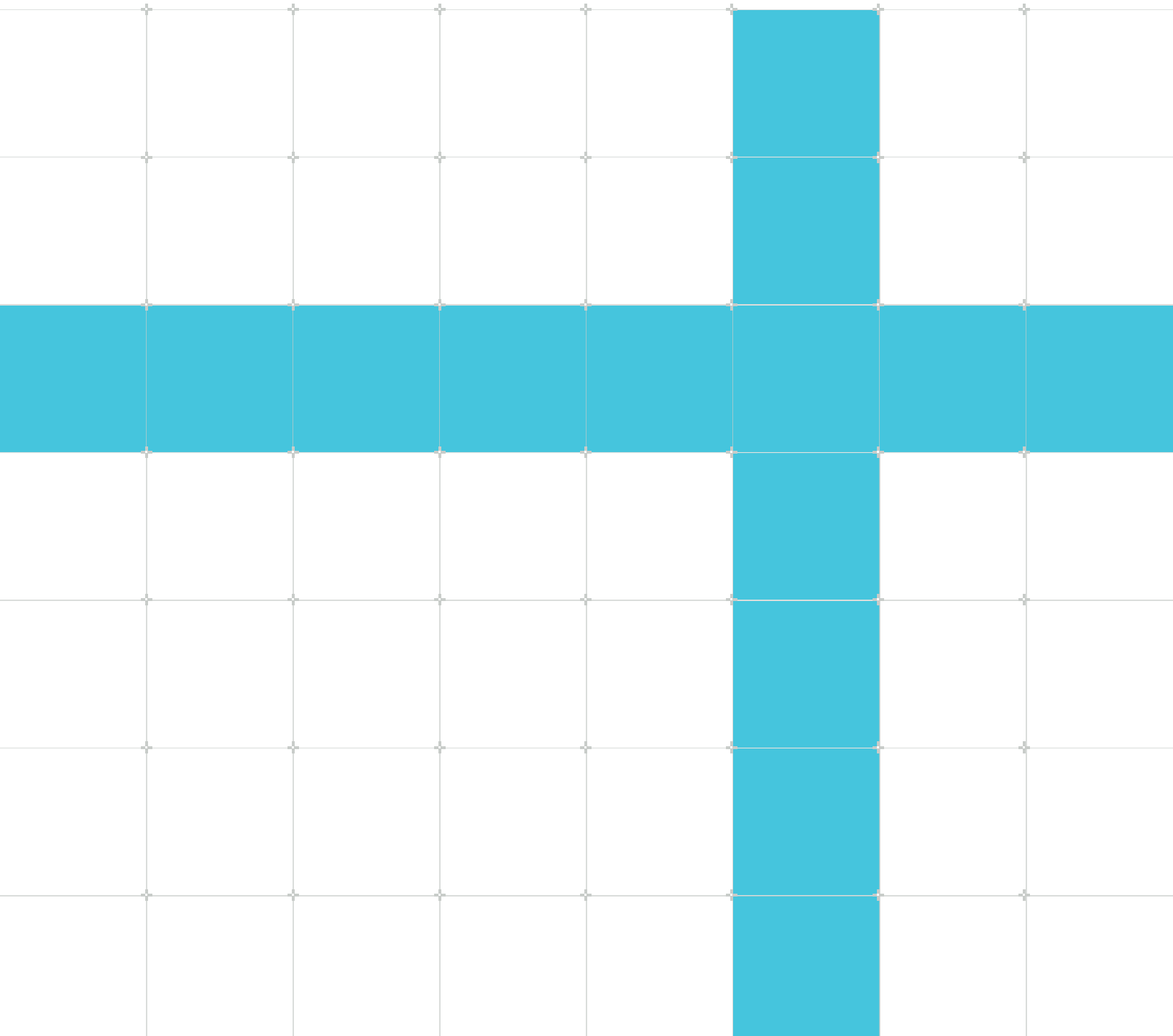
SoC Bring-Up Using the Platform Configuration Editor PCE

Version 1.0

Issue 02
102737_0100_02_en



SoC Bring-Up Using the Platform Configuration Editor PCE

Release information

Document history

Issue	Date	Confidentiality	Change
0100-02	19 November 2019	Non-Confidential	First release

Proprietary Notice

or indirectly, in violation of such export laws. Use of the word "partner" in reference to Arm's customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349|version 21.0)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is Final, that is for a developed product.

Feedback

Arm® welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on https://support.developer.arm.com

To provide feedback on the document, fill the following survey: https://developer.arm.com/documentation-feedback-survey.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

We believe that this document contains no offensive language. To report offensive language in this document, email terms@arm.com.

Contents

1. Overview

This tutorial shows how to use the Platform Configuration Editor (PCE) in Arm DS-5 Development Studio. By the end of this guide, you should be able to connect to an Arm-based System-on-Chip (SoC), use the PCE to detect its underlying system architecture and configure any missing elements of the system. By using the PCE, you will save time and avoid the pitfalls of manually creating debug configuration scripts.

2. Arm SoC Debug and Trace

In order to provide effective debug and trace support for a SoC, a debugger needs a certain amount of information. The debugger needs to know:

- Which devices are present in the SoC (including the type and configuration details of each device, and connection details such as CoreSight base address)
- How these devices relate or connect to each other (their topology)

DS-5 debugger can automatically detect most of this information. However, debugger sometimes fails to detect certain features. The information presented by the SoC may be incomplete or corrupted, or information may be missing because parts of the SoC are powered down, or devices inside the SoC may interfere with topology detection.

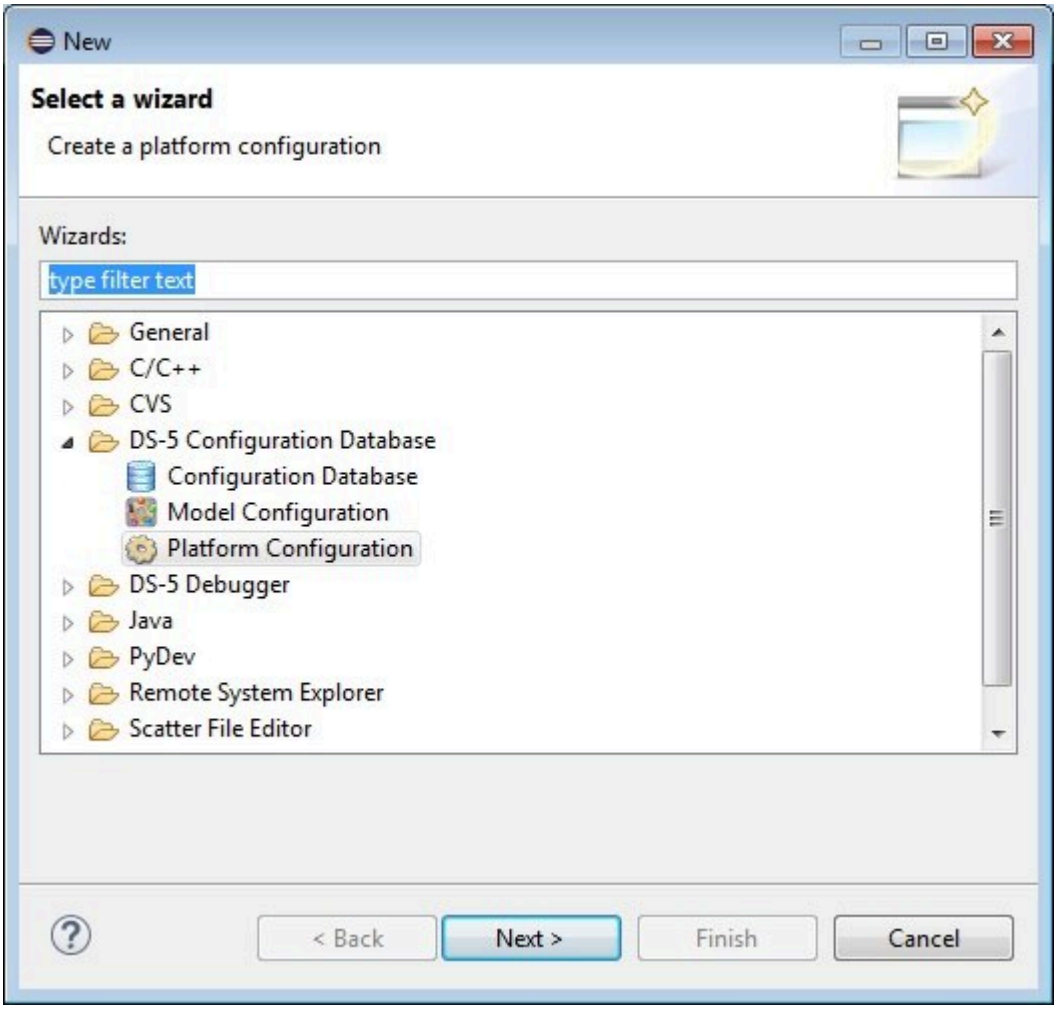
This means that a debugger often has to work with incomplete or incorrect information about a SoC. This can greatly affect the level of debug and trace functionality that the debugger can offer. The debugger could try to make assumptions based on its knowledge of other SoCs, but the flexibility of Arm-based designs means that these assumptions are often limited in their success. Often a CoreSight topology diagram is available, but we have to find some way to get the information in that topology diagram into the debugger.

3. How the PCE is Useful

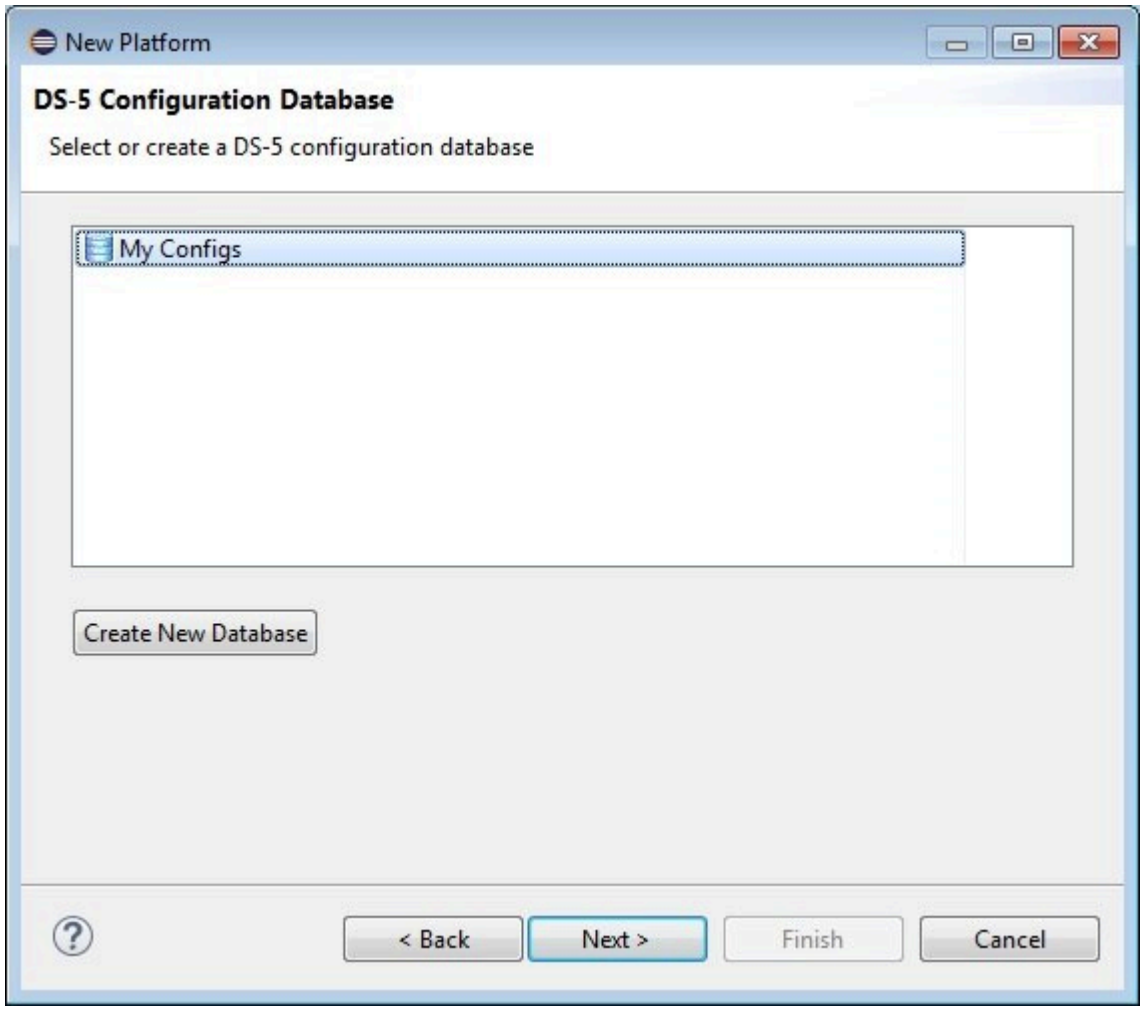
DS-5 contains Platform Configuration Editor (PCE), which provides a simple and flexible way to add platform configuration to DS-5 debugger. In this example, the PCE is used to bring-up a SoC that in the past has caused problems. The SoC contains a mix of Cortex-A and Cortex-M cores, and the device and topology information that it presents is incomplete.

4. SoC Bring-Up in DS-5

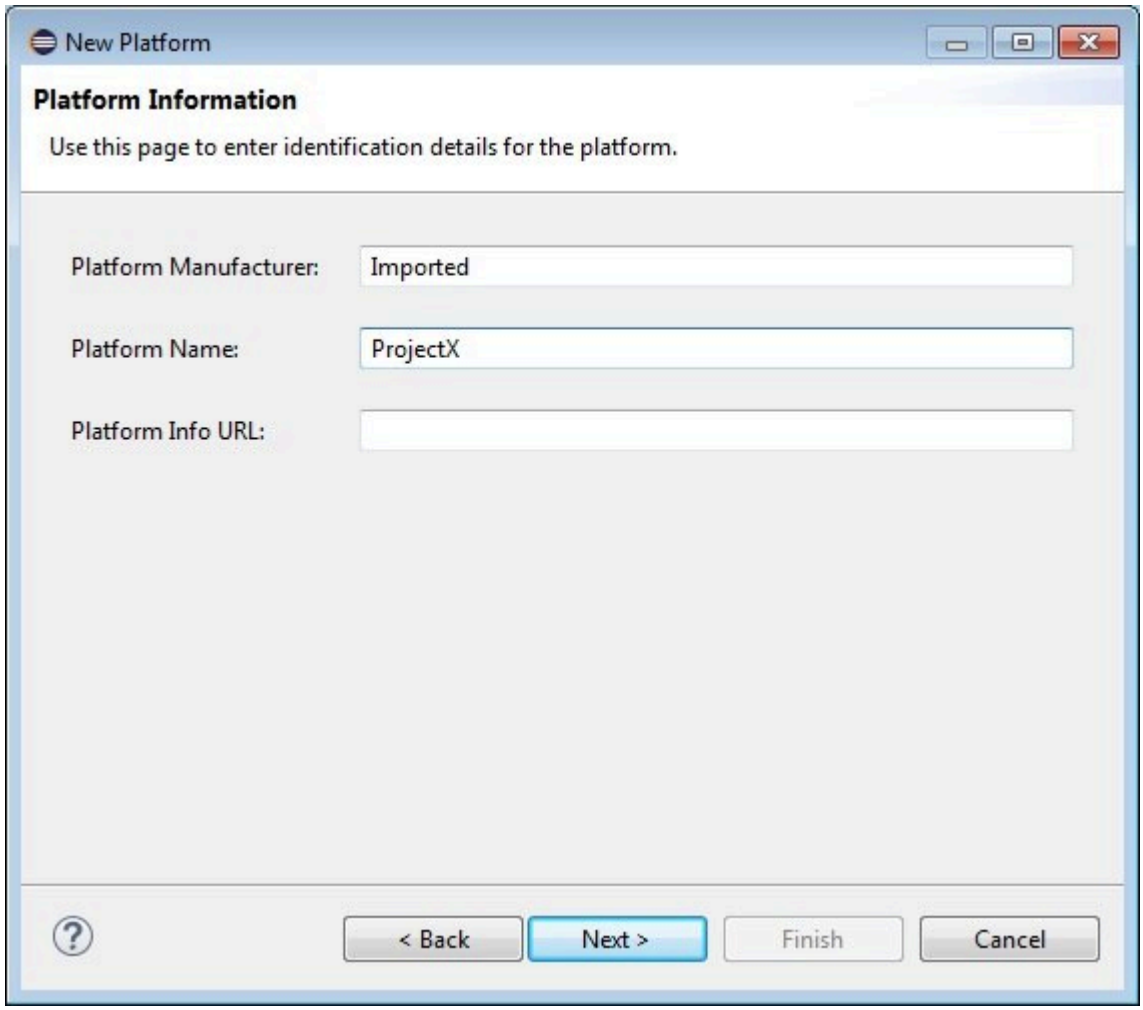
The PCE is launched by selecting File > New > Other... from the DS-5 menu and then selecting DS-5 Configuration Database > Platform Configuration:



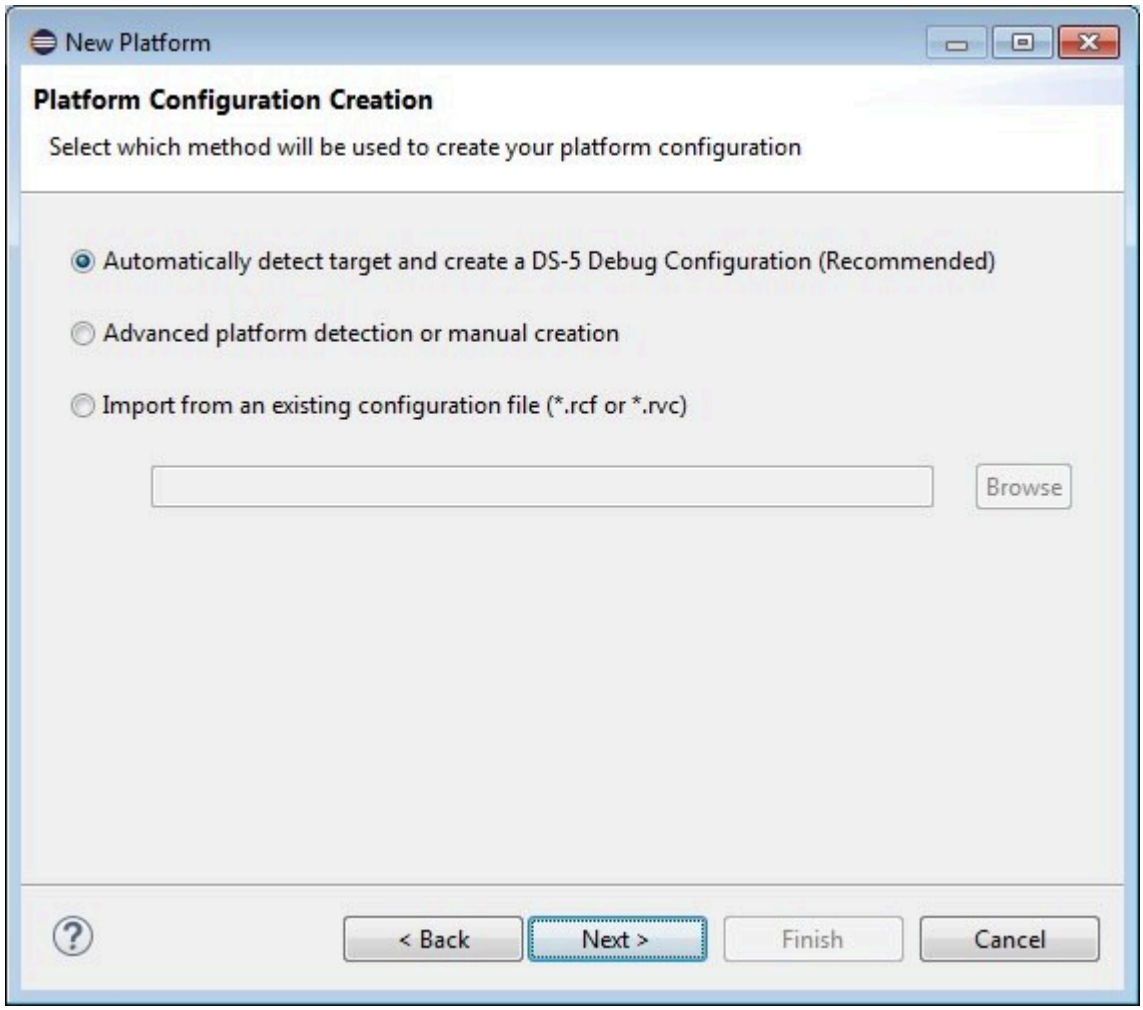
All DS-5 platform configurations have to be stored in a configuration database. Select a configuration database, or create a new one:



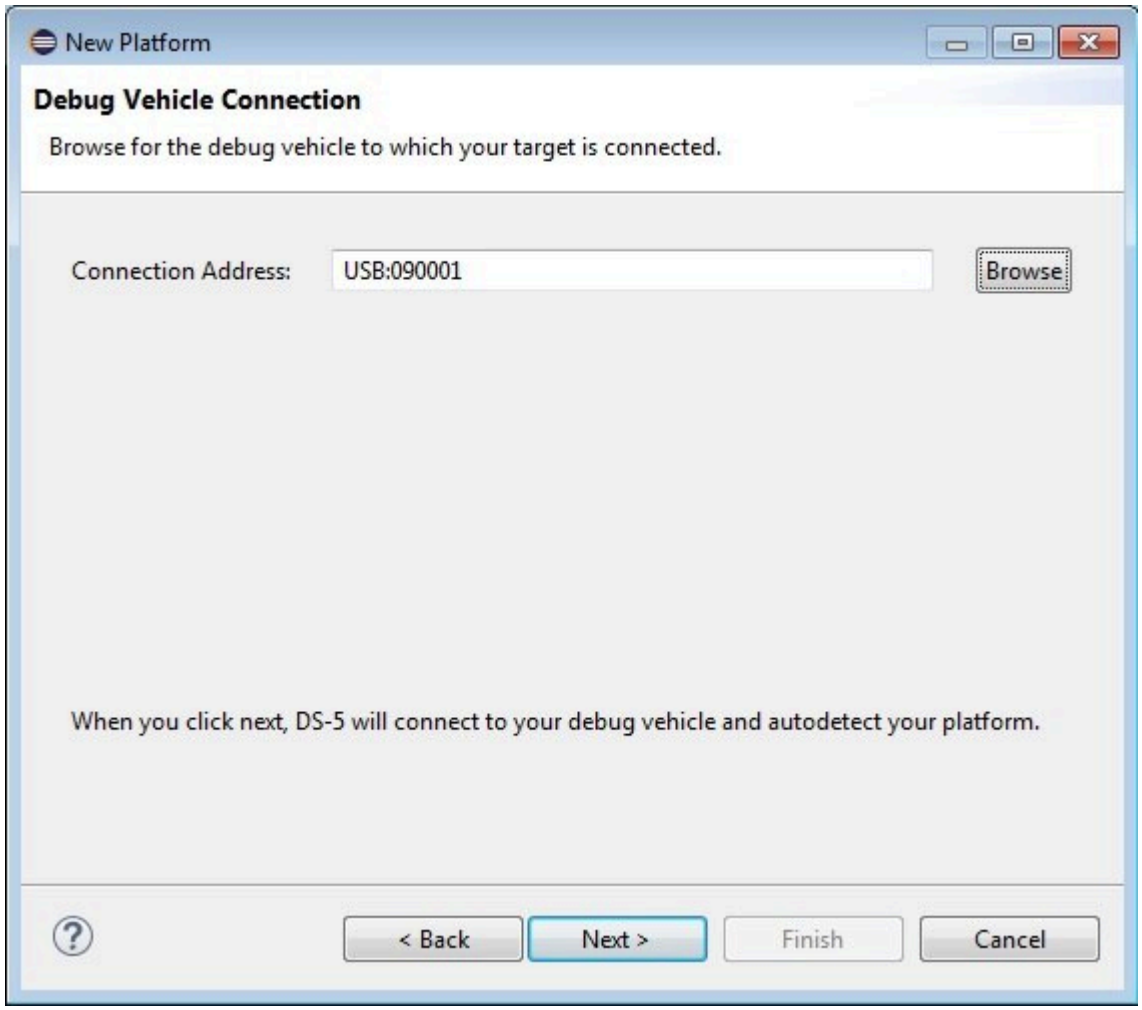
I need to give a manufacturer and platform name:



In the next step, there is a choice of workflows. I would like DS-5 to do as much as possible, I dont want to get involved in manual device addition or advanced configuration, and I dont have an existing file to import. So I can just accept the recommended workflow:

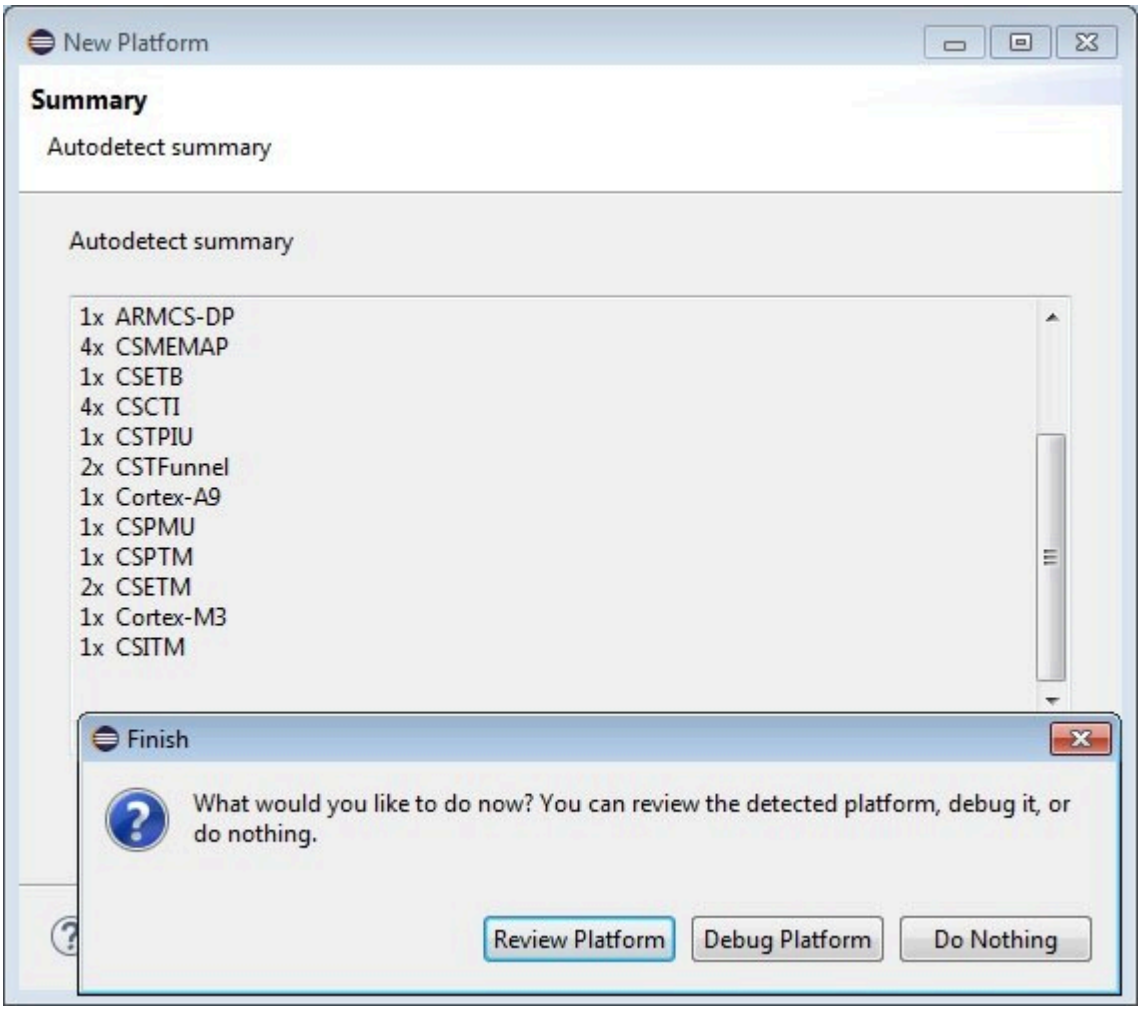


Once Ive selected my DSTREAM unit, DS-5 will connect to the SoC and try to read all the information it needs for debug and trace:

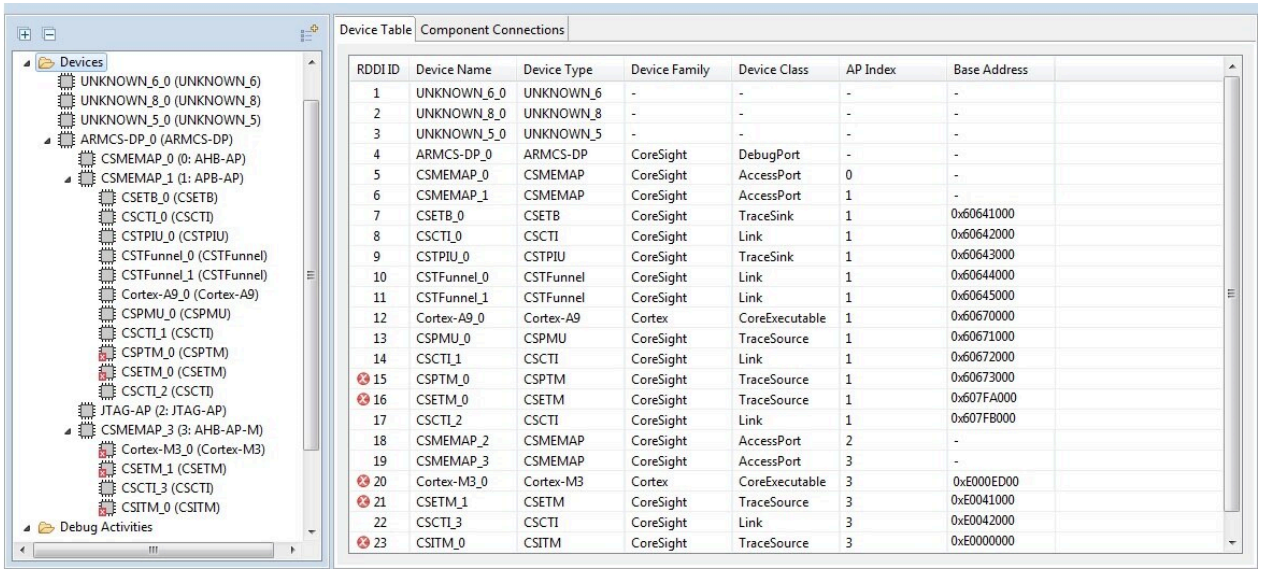


After DS-5 has gathered all the information it can from the SoC, Im presented with a simple summary of the devices that were found in the SoC. I can choose to do nothing now (DS-5 will save the platform configuration its just generated, and Ill be able connect and debug later) or I can debug straight away.

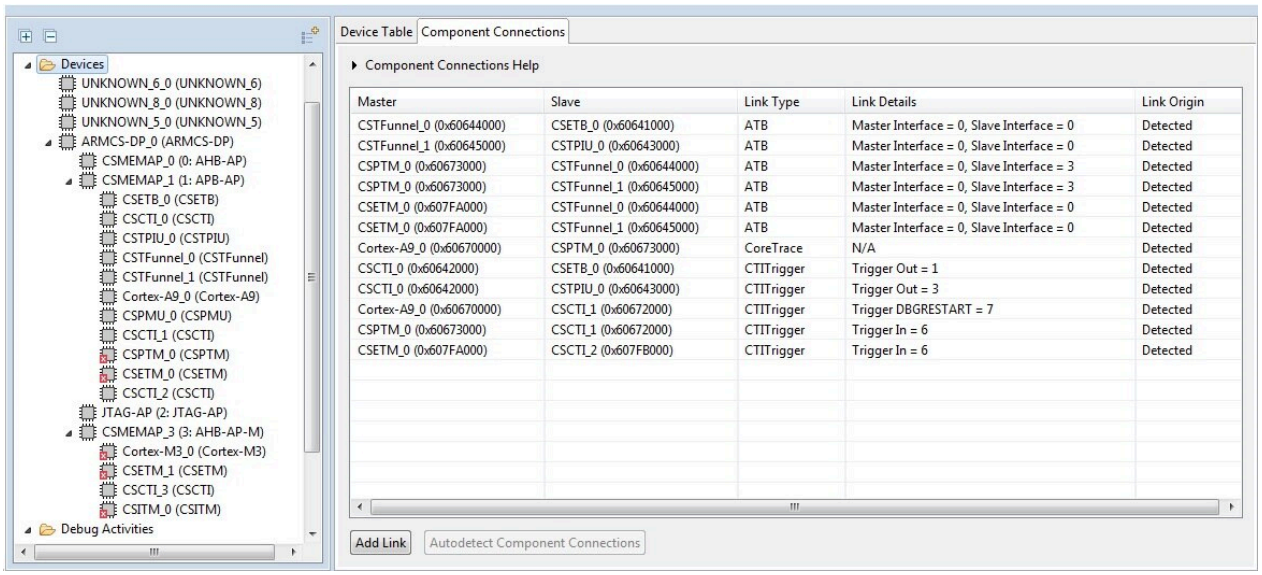
There is no need to rebuild the configuration database. That has already been done and represents a simplification over earlier DS-5 releases. The generated platform configuration will be the best that DS-5 can give from the information it could acquire from the SoC, so there may be missing functionality (particularly trace).



From this device summary, I can see a Cortex-A9 and a Cortex-M3. When I try to debug the target, I find that debug works well for both cores, but I only have trace options for the Cortex-A9. I dont have the ability to configure and collect trace for the Cortex-M3, so it looks like DS-5 could not acquire all the information it needed from the SoC. When I review the platform I can see this device summary:



The left pane shows my device hierarchy: I have a DAP, which provides access to a number of Access Ports (APs) of various types, which in turn provide access to the devices. Summary device details are shown in the right pane. If I select Component Connections I can view the topology information acquired from the SoC:

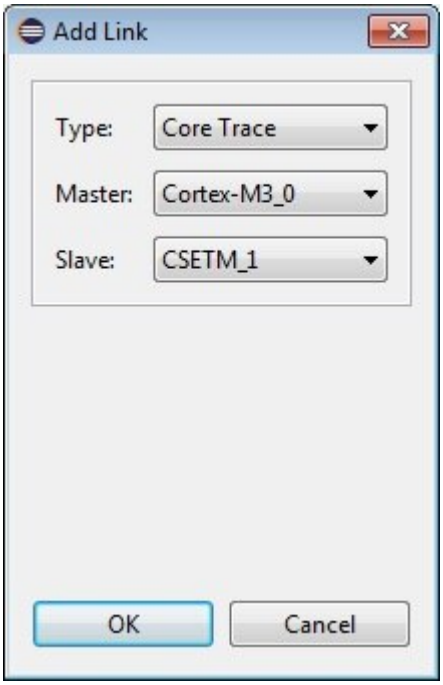


Here I can see that the topology information acquired from the SoC is incomplete, and may also be corrupted. I can see two cores (Cortex-A9 and Cortex-M3) but three core trace macrocells (a PTM and two ETMs). One of the ETMs seems to be spare. It is connected to the trace funnels (at port 0) but doesnt seem to be connected to either of the cores.

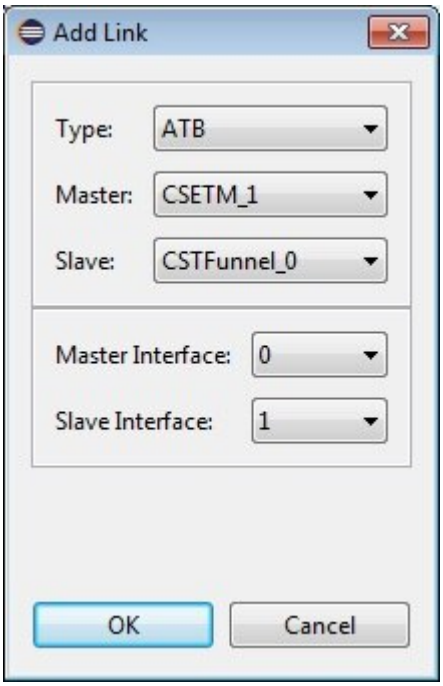
Topology for the Cortex-A9 looks pretty good. It is associated with both a PTM and a CTI, and the PTM is connected to the trace funnels (at port 3), then onwards to the ETB and the TPIU. This topology information was used by DS-5 when it generated the platform configuration, and the trace support that my DS-5 now offers for the Cortex-A9 works just fine. However theres

no topology information for the Cortex-M3, and this is reflected as a lack of functionality in my generated platform configuration.

Adding the link between the Cortex-M3 and its ETM is simple; we should assume that the core and its ETM are on the same AP:



However connecting the ETM to the funnels is more difficult because we have to choose a funnel port. We know that ports 0 and 3 are taken, but that leaves us with a choice of 6 ports. I could use information from a topology diagram if I had one but since I dont have the necessary information my only option is to try each port in turn:



When I save my changes, the platform configuration is automatically rebuilt and I can connect for debug and trace. In this case Im lucky: port 1 is the first port I tried and its the correct port, so now I have trace working for the Cortex-M3 as well as for the Cortex-A9. Although I had to enter some information manually I did not have to open the generated platform configuration and do any manual scripting, and this represents a significant improvement.

5. Summary

SoC bring-up in a debugger can be a tricky process that in the past might demand a level of debug expertise and some manual scripting. The PCE bring-up tool in DS-5 contains a number of important features that bring significant benefits to the SoC bring-up process:

- The PCE, with support for automatic detection of platform configuration information from SoC, reduces time and effort needed to bring up a new SoC.
- The PCE does not make assumptions about topology information that cannot be read from the SoC. This makes it easier for the user to spot the missing information, which can then be added.
- Manually adding missing topology information can be done through the main user interface, using simple dialogs. Theres no need to hand-edit complex topology files.

6. Troubleshooting

As mentioned earlier, configuring an Arm-based SoC for bring-up can be tricky. Arm provides detailed technical support for our customers, so if you are finding it difficult to bring-up a particular device, don't hesitate to get in touch with us. You can raise a support case here.