

# READ ME FIRST

## Freescale/ARM<sup>®</sup> SC140V3, SDMA, ARM11<sup>™</sup> Tool Initialization Guide

Revision 1.04

This document explains how to configure Freescale's CodeWarrior<sup>™</sup> tools and ARM's RealView<sup>®</sup> Development tools to establish a multi-core debug environment on the MXC 05, i.300-30, MXC275-30, and MXC300-30 development systems from Freescale.

Freescale<sup>™</sup> and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

## *Table of Contents*

<i>1</i>	About This Document.....	3
1.1	Introduction.....	3
1.2	Related Documents .....	3
<i>2</i>	System Overview .....	4
<i>3</i>	Setting Up the ARM RealView Development Tools.....	6
3.1	RVDS and CodeWarrior Compatibility.....	6
3.2	Installation and Set-up Process .....	6
<i>4</i>	Setting Up the Freescale CodeWarrior Tools .....	8
<i>5</i>	Single Core Debug.....	9
5.1	ARM9/ARM11 .....	9
5.2	SC140V3.....	10
5.3	SDMA .....	12
<i>6</i>	Multi-Core Debug.....	13
6.1	SC140V3 and ARM9/ARM11.....	13
6.2	SDMA and ARM9/ARM11 .....	15
6.3	SC140V3 and SDMA .....	18
6.4	SC140V3, SDMA, and ARM9/ARM11 .....	19
	APPENDIX.....	23
	Appendix A Definitions.....	23
	Appendix B Software & Trace Debug Block Diagram .....	24

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

# 1 About This Document

## 1.1 Introduction

This paper explains how to set up and implement a multi-core debug environment for the MXC91221, MXC 05, i.300-30, MXC275-30, and MXC300-30 development systems from Freescale. To complete the setup, you configure Freescale's CodeWarrior™ Development Studio for StarCore® and SDMA Architectures, and ARM's RealView Development Tools.

## 1.2 Related Documents

Freescale's StarCore Users' Manual  
Freescale's Targeting StarCore and SDMA Manual  
Freescale's StarCore and SDMA Quick Start for Nexus Trace  
ARM RealView ICE User Guide  
ARM RealView ICE Installation Guide  
ARM RealView Development Suite Installation Guide  
ARM RealView Debugger Target Configuration Manual  
ARM RealView Debugger User Guide

## 2 System Overview

The software development system for the Freescale MXC91221, MXC 05, i.300-30, MXC275-30, and MXC300-30 platforms consists of an evaluation board and software development kits from Freescale Semiconductor, Inc. and ARM Limited.

The MXC91221 platform consists of three cores: ARM9, SC140V3, and SDMA.

The MXC 05, i.300-30, MXC275-30, and MXC300-30 platforms consist of three cores: ARM11, SC140V3 and SDMA.

Appendix B has a block diagram of the communication channels for Freescale and ARM development tools with the silicon.

Any of these systems rely on the ARM RealView Development Suite (RVDS) tools for debugging the ARM core and the Freescale CodeWarrior tools for debugging the StarCore and SDMA cores. The two toolsets use the RealView ICE with MXC Support unit and an ARM debug protocol to debug all three cores. Similarly, CodeWarrior tools and RVDS use ARM's RealView Trace with MXC Support unit and the debug protocol to capture trace data.

### *Single Core Debug*

If your goal is to perform single-core debugging without having to stop and start multiple cores, you have two options. You can use just the ARM tools to perform single-core debugging. Alternately, you can use just the CodeWarrior tools to perform single-core debugging of the SDMA core or the StarCore.

***Multi-Core Debug***

If your goal is to perform multi-core debugging (where you debug multiple cores simultaneously), you can use a combination of the CodeWarrior and ARM tools.

<b>To Debug these processors/boards</b>	<b>Use these tools</b>
ARM	RVDS
ARM and SDMA	RVDS and CodeWarrior
StarCore and ARM	RVDS and CodeWarrior
SDMA and StarCore	CodeWarrior
ARM, SDMA, and StarCore	RVDS and CodeWarrior

**Table 1 – Multi-Core Debug Options**

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

## 3 Setting Up the ARM RealView Development Tools

This section explains how to set up RealView Development Suite (RVDS), RealView ICE with MXC Support (RVI MXC), and RealView Trace with MXC Support (RVT MXC) for use with the Freescale MXC91221, MXC 05, i.300-30, MXC275-30, and MXC300-30 platforms. Read this section in conjunction with the RVDS Installation Guide, the RVI Installation Guide, and the RVI User Guide. This section highlights differences between the standard documentation and this custom installation.

### 3.1 *RVDS and CodeWarrior Compatibility*

RVI MXC v1.6 is compatible with RVDS 2.1.1, RVDS 2.2 SP1, RVDS 3.0, and RVDS 3.0 SP1. Please contact your supplier to obtain the latest RVDS distributions.

RVI MXC v1.6 is compatible with CodeWarrior for StarCore and SDMA v1.0 and v1.1. Please note that to fully enable multi-core MXC support, you must install a firmware update available from Freescale. For users of CodeWarrior for StarCore and SDMA v1.2, or later, the firmware update can be found within the RVI\_MXC\_Firmware\_Templates directory within the CodeWarrior product installation. If you do not have CodeWarrior v1.2 or later, and require the firmware update, please contact your regional Freescale Sales & Support office. Sales and Support contact details can be found at:

<http://www.freescale.com/support>

### 3.2 *Installation and Set-up Process*

First follow the instructions in the RealView Development Suite Install Guide to install RVDS and obtain a license to use the product.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

If you are going to use RealView Trace, follow the instructions in Chapter 6.4 of the RealView ICE User Guide to connect the RealView Trace unit to the RealView ICE unit.

Next, install the RVI MXC software and firmware. The RVI MXC unit firmware is already configured for ARM and StarCore debug. To debug the SDMA, you must install the firmware update referenced in Section 3.1 of this document.

**To set up RVI MXC and (optionally) RVT MXC, follow these steps:**

1. Install the RVI MXC software.  
**Software:** RVI with MXC Support CDROM.  
**Book Reference:** RealView ICE Installation Guide.
2. Connect RVI and RVT to the network and configure the network properties.  
**Book Reference:** Chapter 3 of the RealView ICE User Guide.
3. Use the RVI Update utility to install the Freescale firmware update  
NOTE: The Freescale firmware update adds SDMA debug capability to RVI; without the update's patch, you cannot perform full multi-core debugging.  
**Software:** Freescale RVI MXC firmware update.  
**Book Reference:** Chapter 7 of the RealView ICE User Guide.
4. Set the RVT MXC trace probe mode-selector switch to 'ETM' to trace ARM9/ARM11 or to 'NEX' (LED illuminates) to trace StarCore.
5. Connect the RVI and RVT to the target.  
**Book Reference:** RealView ICE User Guide

NOTE: Any change to the RVT MXC trace probe mode requires a power cycle of the RVI / RVT units to take effect.

The ARM RealView Development tools are now configured for use with the Freescale MXC91221, MXC 05, i.300-30, MXC275-30, and MXC300-30 platforms.

## 4 Setting Up the Freescale CodeWarrior Tools

Consult the *CodeWarrior Development Studio for StarCore and SDMA Architectures Quick Start* for information on:

- installing the CodeWarrior software
- creating, building, and running a sample project

After following the instructions in the Quick Start to install the CodeWarrior software, you can return to this initialization guide. If you want to better familiarize yourself with the CodeWarrior software, complete the entire Quick Start.

To set up the CodeWarrior tools to collect Nexus trace, refer to the Quick Start for Nexus trace. Additional information about Nexus setup is in application note AN3290, which you can download from the CodeWarrior for StarCore and SDMA product page:

<http://www.freescale.com/codewarrior>



## 5 Single Core Debug

### 5.1 ARM9/ARM11

To debug the ARM9 core on an MXC91221 platform, or the ARM11 core on an MXC05, i.300-30, MXC275-30, or MXC300-30 platform, follow these steps:

1. Start RealView RVD Debugger.
2. Click the ‘Click to Connect to a Target’ link from the main code window. The ‘Connection Control’ window appears.
3. Right-click the ‘RealView-ICE’ entry.
4. Select ‘Configure...’ from the options presented (or ‘Configure Device Info’ in RVDS 2.2’).
5. Click ‘Browse’ – the ‘RVConfig’ configuration application starts and a list of connectable RVIs appears. Double click an RVI to select it.
6. To configure the scan chain, click the ‘Auto Configure Scan Chain’ button. A successful auto configuration shows five devices on the scan chain. If this scan fails, perform manual configuration:
  - a. Click ‘Add Device’
  - b. Add devices in this order:

MXC91221	MXC05, i.300-30, MXC275-30, and MXC300-30 Platforms
ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC	ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

With this ordering, ETMBUF has Tap ID '0' and SJC has Tap ID '4'.

7. Click the ARM1136JF-S or ARM926EJ-S entry.
8. Click 'Device Properties' in the right hand window. To select trace capture from the RVT unit, select just 'ETM.' To select trace capture from the on-chip ETB, select both 'ETM' and 'ETB.'
9. Select File → Save.
10. Close the application.
11. In the 'Connection Control' window, under RealView-ICE, double click to connect to the target (or click the 'ARM11' or 'ARM9' box in RVDS 2.2).
12. Select File → Load Image... to download an image to the target.
13. For trace, select View → Analysis Window to display the 'Analysis' window, then select Edit → Connect Analyzer... (you must set the mode switch to ETM on the trace probe).  
NOTE: The target board trace port mux must select the relevant ARM9 or ARM11 ETM trace also.

For in-depth information about configuration options, consult Section 4 of the RealView ICE User Guide.

## 5.2 SC140V3

To debug the SC140V3 core on an MXC 05, i.300-30, MXC275-30, or MXC300-30 platform, follow these steps:

**NOTE:** These i.300-30 instructions also apply to MXC 05, MXC275-30, and MXC300-30 hardware.

1. Select File → New
2. In the Project page select 'StarCore and SDMA Stationery'.
3. Give your project a name and specify a location in which to save it.
4. In the New Project window, expand 'i.300-30,' then select 'StarCore.'
5. Click the **OK** button.

6. A project window appears. Select Project → Make to compile the project.
7. Select Edit → ‘StarCore Stationery Settings’. The Target Settings window appears.
8. Under ‘Remote Debugging’ options, set the ‘Connection’ to ‘StarCore RVI’.
9. Click the ‘Edit Connection’ button. A warning message appears.
10. Click the **OK** button. Specify the appropriate ‘Server IP Address’ for your board. Leave the port number at the default setting 3010. Also, specify the appropriate JTAG Clock Speed for your board.
11. Select Project → Debug to debug the project.

You may now use the Debugger commands under the “Debug” drop down menu to control the target.

## 5.3 SDMA

To debug the SDMA core on an MXC 05, i.300-30, MXC275-30, or MXC300-30 platform, follow these steps:

**NOTE1:** These i.300-30 instructions also apply to MXC 05, MXC275-30, and MXC300-30 hardware.

**NOTE2:** RVI MXC Firmware Templates are required to debug the SDMA core.

1. Select File → New
2. In the Project page select ‘StarCore and SDMA Stationery’.
3. Give your project a name and specify a location in which to save it.
4. In the New Project window, expand ‘i.300-30’ and select SDMA.
5. Click the **OK** button.
6. A project window appears. Select Project → Make to compile the project.
7. Select Edit → ‘StarCore Stationery Settings’. The Target Settings window appears.
8. Under ‘Remote Debugging’ options, set the ‘Connection’ to ‘StarCore RVI’.
9. Click the ‘Edit Connection’ button. A warning message appears.
10. Click the **OK** button. Specify the appropriate ‘Server IP Address’ for your board. Leave the port number at the default setting 3010. Also, specify the appropriate JTAG Clock Speed for your board.
11. Select Project → Debug to debug the project.

You may now use the debugger commands under the “Debug” drop menu to control the target.

## 6 Multi-Core Debug

Multi-core debugging is debugging multiple cores connected to a JTAG chain. You create a separate project to control and debug each core. To debug an ARM9 or SC140V3 core on an MXC91221 or ARM11 platform, or to debug an SC140V3 core on an MXC 05, i.300-30, MXC275-30, or MXC300-30 platform, follow the instructions below.

**NOTE:** RVI MXC Firmware Templates are required to perform multi-core operations.

### 6.1 SC140V3 and ARM9/ARM11

To implement multi-core debugging on the SC140V3 and ARM9 or ARM11, you configure RVDS and CodeWarrior tools separately.

**Note:** If you did not configure the ARM tools previously (as explained in steps 2 to 5), you must configure them *before* using the CodeWarrior tools. Otherwise, the autodetection adversely affects the CodeWarrior connection to the StarCore and SDMA devices.

**To configure ARM tools, follow these steps:**

1. Start RealView RVD Debugger.
2. Click the 'Click to Connect to a Target' link from the main code window. The 'Connection Control' window appears.
3. Right-click the 'RealView-ICE' entry
4. Select 'Configure...' from the options presented (or 'Configure Device Info' in RVDS 2.2').
5. Click 'Browse' – the 'RVConfig' configuration application starts and a list of connectable RVIs appears. Double click an RVI to select it.
6. To configure the scan chain, click the 'Auto Configure Scan Chain' button. A successful auto configuration shows five devices on the scan chain. If this scan fails, perform manual configuration:

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

- a. Click ‘Add Device’
- b. Add devices in this order:

MXC91221	MXC05, i.300-30, MXC275-30, and MXC300-30 Platforms
ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC	ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC

With this ordering, ETMBUF has Tap ID ‘0’ and SJC has Tap ID ‘4’.

7. Click the ARM1136JF-S or ARM926EJ-S entry.
  8. Click ‘Device Properties’ in the right hand window. To select trace capture from the RVT unit, select just ‘ETM’. To select trace capture from the on-chip ETB, select both ‘ETM’ and ‘ETB.’
  9. Select File → Save
  10. Close the application.
  11. In the ‘Connection Control’ window, under RealView-ICE, double click to connect to the target (or click on the ‘ARM11’ or ‘ARM9’ box in RVDS 2.2).
  12. Select File → Load Image... to download an image to the target.
  13. For trace, select View → Analysis Window to show the ‘Analysis’ window, and then select Edit → Connect Analyzer... (you must set the mode switch to ETM on the trace probe).
- NOTE: The target board trace port mux must select the relevant ARM 9 or ARM11 ETM trace also.

For in-depth information about configuration options, consult Section 4 of the RealView ICE User Guide

Inside CodeWarrior Development Studio for StarCore and SDMA, you create a Single Core project.

**To configure CodeWarrior tools, follow these steps:**

**NOTE:** These i.300-30 instructions also apply to MXC 05, MXC275-30, and MXC300-30 hardware.

1. Select File → New
2. In the Project Tab select ‘StarCore SDMA Stationery’.
3. Give your project a name and specify a location in which to save it.
4. In the New Project window, expand ‘i.300-30’ and select ‘StarCore.’
5. Click the **OK** button.
6. A project window appears. Select Project → Make to compile the project.
7. Select Project → Debug to debug the project.

You cannot use RVD to perform synchronization operations (start/stop/step) because the devices are not connected to RVD.

## **6.2 SDMA and ARM9/ARM11**

To implement multi-core debugging on the SDMA and ARM9 or ARM11, you configure RVDS and CodeWarrior tools separately.

**Note:** If you did not configure the ARM tools previously (as explained in steps 2 to 5), you must configure them *before* using the CodeWarrior tools. Otherwise, the autodetection adversely affects the CodeWarrior connection to the StarCore and SDMA devices.

**To configure ARM tools, follow these steps:**

1. Start RealView RVD Debugger

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

2. Click the ‘Click to Connect to a Target’ link from the main code window. The ‘Connection Control’ window appears.
3. Right-click the ‘RealView-ICE’ entry
4. Select ‘Configure...’ from the options presented (or ‘Configure Device Info’ in RVDS 2.2’).
5. Click ‘Browse’ – the ‘RVConfig’ configuration application starts and a list of connectable RVIs appears. Double click an RVI to select it.
6. To configure the scan chain, click the ‘Auto Configure Scan Chain’ button. A successful auto configuration shows five devices on the scan chain. If this scan fails, perform manual configuration:
  - a. Click ‘Add Device’
  - b. Add devices in this order:

MXC91221	MXC05, i.300-30, MXC275-30, and MXC300-30 Platforms
ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC	ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC

With this ordering, ETMBUF has Tap ID ‘0’ and SJC has Tap ID ‘4’.

7. Click the ARM1136JF-S or ARM926EJ-S entry.
8. Click ‘Device Properties’ in the right hand window. To select trace capture from the RVT unit, select just ‘ETM’. To select trace capture from the on-chip ETB, select both ‘ETM’ and ‘ETB’.
9. Select File → Save.
10. Close the application.



11. In the 'Connection Control' window, under RealView-ICE, double click to connect to the target (or click on the 'ARM11' or 'ARM9' box in RVDS 2.2).
12. Select File → Load Image... to download an image to the target.
13. For trace, select View → Analysis Window to show the 'Analysis' window, and then select Edit → Connect Analyzer... (you must set the mode switch to ETM on the trace probe).

NOTE: The target board trace port mux must select the relevant ARM9 or ARM11 ETM trace also.

For in-depth information about configuration options, consult Section 4 of the RealView ICE User Guide

Inside CodeWarrior Development Studio for StarCore and SDMA, you create a Single Core project.

**To configure CodeWarrior tools, follow these steps:**

**NOTE:** These i.300-30 instructions also apply to MXC 05, MXC275-30, and MXC300-30 hardware.

1. Select File → New.
2. In the Project Tab select 'StarCore SDMA Stationery'.
3. Give your project a name and specify a location in which to save it.
4. In the New Project window, expand 'i.300-30' and select 'SDMA.'
5. Click the **OK** button.
6. A project window appears. Select Project → Make to compile the project.
7. Select Project → Debug to debug the project.

You cannot use RVD to perform synchronization operations (start/stop/step) because the devices are not connected to RVD.

## 6.3 SC140V3 and SDMA

To implement multi-core debugging on the SDMA and SC140V3, you configure just the CodeWarrior tools.

Inside CodeWarrior Development Studio for StarCore and SDMA, you create a Single Core project.

**To configure CodeWarrior tools, follow these steps:**

**NOTE:** These i.300-30 instructions also apply to MXC 05, MXC275-30, and MXC300-30 hardware.

1. Select File → New
2. In the Project Tab select 'StarCore SDMA Stationery'.
3. Give your project a name and specify a location in which to save it.
4. In the New Project window, expand 'i.300-30' and select 'SDMA.'
5. Click the **OK** button.
6. A project window appears. Select Project → Make to compile the project.
7. Select Project → Debug to debug the project.

When you start debugging a multi-core project, the IDE downloads the related targets to the appropriate cores, with correct settings for multi-core debugging. The IDE opens a separate debugging window for each project in the multi-core project. If you prefer, you can kill (remove) all debugging windows while you work with the project windows. To do so, select Multi-Core Debug > Kill All.

When debugging a multi-core project, you can use multi-core debugging commands. (You can use the standard single-core debugging commands to debug each individual core project). You use the **Multi-Core** debug menu to access the multi-core debugging commands. The menu has these commands:

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

**Multi-Core Debug > Run All**—starts execution of all cores as close to the same time as possible. Another name for this action is *synchronous run*.

**Multi-Core Debug > Stop All**—stops execution of all cores as close to the same time as possible. Another name for this action is *synchronous stop*. Synchronous stop means that when any of the executing cores stops (as for a software breakpoint or explicit stop command), all the other cores stop as well.

**Multi-Core Debug > Kill All**—kills all multi-core debugging sessions as close to the same time as possible.

**Multi-Core Debug > Restart**—stops, then restarts, execution of all cores as close to the same time as possible.

## 6.4 SC140V3, SDMA, and ARM9/ARM11

To implement multi-core debugging on the SC140V3, SDMA, and ARM9 or ARM11, you configure both the CodeWarrior tools and the ARM tools.

**Note:** If you did not configure the ARM previously (as explained in steps 2 to 5), you must configure them *before* the using the CodeWarrior tools. Otherwise, the autodetection adversely affects the CodeWarrior connections to the StarCore and SDMA devices.

**To configure ARM tools, follow these steps:**

1. Start RealView RVD Debugger
2. Click the ‘Click to Connect to a Target’ link from the main code window. The ‘Connection Control’ window appears.
3. Right-click the ‘RealView-ICE’ entry.
4. Select ‘Configure...’ from the options presented (or ‘Configure Device Info’ in RVDS 2.2’).
5. Click ‘Browse’ – the ‘RVConfig’ configuration application starts and a list of connectable RVIs appears. Double click an RVI to select it.
6. To configure the scan chain, click the ‘Auto Configure Scan Chain’ button. A successful auto configuration shows five devices on the scan chain. If this scan fails, perform manual configuration:
  - a. Click ‘Add Device’
  - b. Add devices in this order:

MXC91221	MXC05, i.300-30, MXC275-30, and MXC300-30 Platforms
ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC	ARM → ETMBUF ARM9 → ARM926EJ-S ARM11 → ARM1136JF-S Motorola → StarCORE140 Motorola → SDMA Motorola → SJC

With this ordering, ETMBUF has Tap ID ‘0’ and SJC has Tap ID ‘4’.

7. Click the ARM1136JF-S or ARM926EJ-S entry.
8. Click ‘Device Properties’ in the right hand window. To select trace capture from the RVT unit, select just ‘ETM’. To select trace capture from the on-chip ETB, select both ‘ETM’ and ‘ETB’.
9. Select File → Save.
10. Close the application.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

11. In the 'Connection Control' window, under RealView-ICE, double click to connect to the target (or click on the 'ARM11' or 'ARM9' box in RVDS 2.2).
12. Select File → Load Image... to download an image to the target.
13. For trace, select View → Analysis Window to show the 'Analysis' window, and then select Edit → Connect Analyzer... (you must set the mode switch to ETM on the trace probe).  
NOTE: The target board trace port mux must select the relevant ARM 9 or ARM11 ETM trace also.

For in-depth information about configuration options, consult Section 4 of the RealView ICE User Guide

Inside CodeWarrior Development Studio for StarCore and SDMA, you create a Single Core project.

**To configure CodeWarrior tools, follow these steps:**

**NOTE:** These i.300-30 instructions also apply to MXC 05, MXC275-30, and MXC300-30 hardware.

1. Select File → New
2. In the Project Tab select 'StarCore SDMA Stationery'.
3. Give your project a name and specify a location in which to save it.
4. In the New Project window, expand 'i.300-30' and select 'SDMA.'
5. Click the **OK** button.
6. A project window appears. Select Project → Make to compile the project.
7. Select Project → Debug to debug the project.

When you start debugging a multi-core project, the IDE downloads the binary files to the appropriate cores, with correct settings for multi-core debugging. The IDE opens a separate debugging window for each project in the multi-core project. If you prefer, you can kill (remove) all

debugging windows while you work with the project windows. To do so, select **Multi-Core Debug > Kill All**.

When debugging a multi-core project, you can use multi-core debugging commands. (You can use the standard single-core debugging commands to debug each individual core project). You use the **Multi-Core** debug menu to access the multi-core debugging commands. The menu has these commands:

**Multi-Core Debug > Run All**—starts execution of all cores as close to the same time as possible. Another name for this action is *synchronous run*.

**Multi-Core Debug > Stop All**—stops execution of all cores as close to the same time as possible. Another name for this action is *synchronous stop*. Synchronous stop means that when any of the executing cores stops (as for a software breakpoint or explicit stop command), all the other cores stop as well.

**Multi-Core Debug > Kill All**—kills all multi-core debugging sessions as close to the same time as possible.

**Multi-Core Debug > Restart**—stops, then restarts, execution of all cores as close to the same time as possible.

## APPENDIX

### Appendix A Definitions

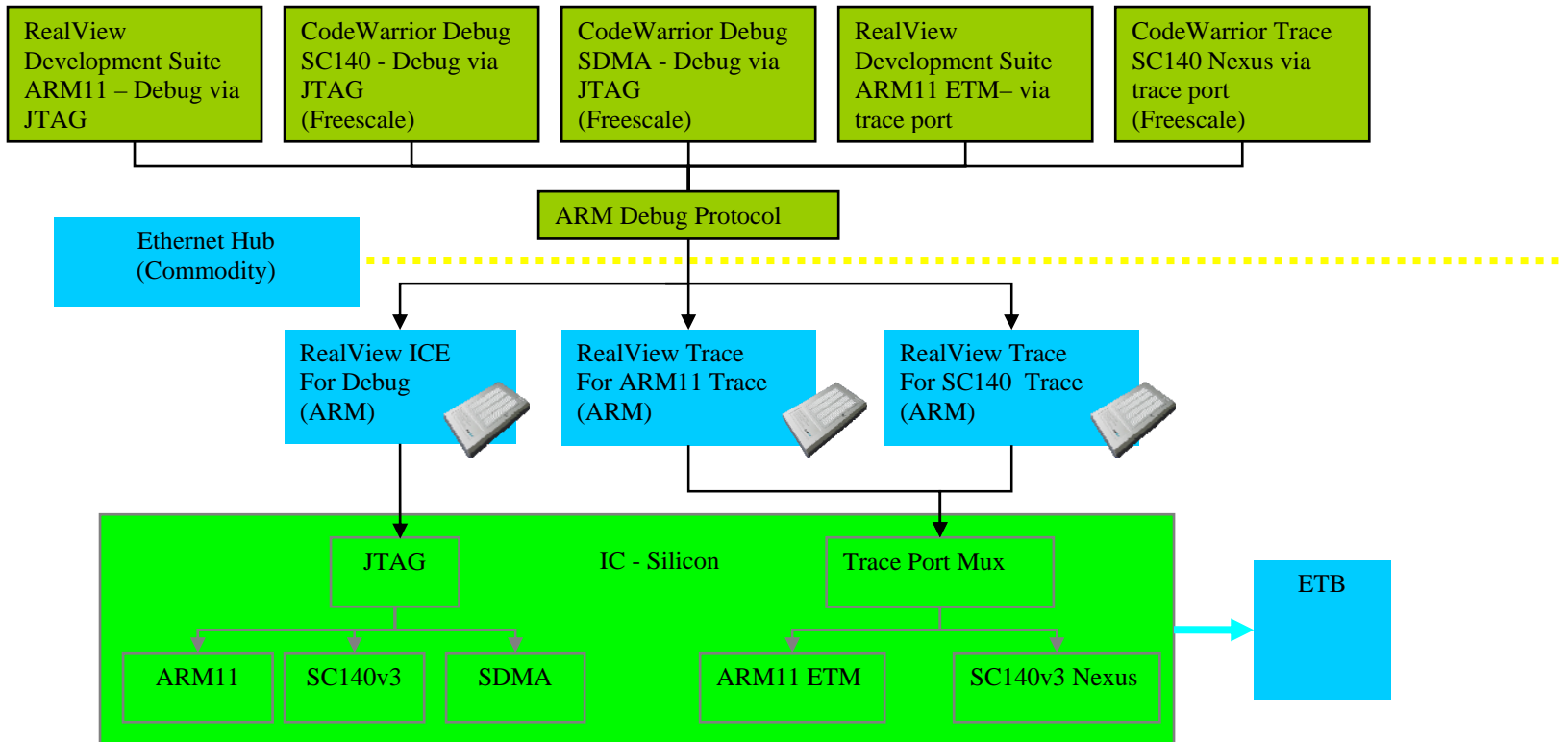
ETB	Embedded Trace Buffer™ component
ETM	Embedded Trace Macrocell™ component
JTAG	Joint Test Action Group. Refers to IEEE Standard 1149.1 – defined as a four-pin interface, plus power and ground, used for processor testing.
Nexus	An on-board module that you use to collect trace
RVDS	RealView Development Suite
RVD	RealView Debugger
RVI	RealView ICE Run Control Unit
RVI MXC	RealView ICE with MXC Support product, containing custom software and templates for the StarCore DSP
RVT	RealView Trace Capture Unit
RVT StarCore	RealView Trace with MXC Support product, containing a custom ETM/Nexus switchable trace probe
Trace	Collection of data about program execution

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004

## Appendix B Software & Trace Debug Block Diagram



Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. CodeWarrior is a trademark or registered trademark of Freescale Semiconductor, Inc. in the U.S. and/or other countries. StarCore is a registered trademark of Freescale Semiconductor, Inc.

The ARM logo and RealView logo are registered trademarks of ARM Ltd.

All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004