

ARM

series

SIGNUM SYSTEMS

JTAGjet-ARM Driver for IAR Embedded Workbench

Installation Instructions

JTAGJET-ARM DRIVER FOR IAR EMBEDDED WORKBENCH

Installation Instructions

SWII-RDI1511- Driver 2.26.03.12.32



© 2003 Signum Systems
11992 Challenger Court • Moorpark, CA 93021
Phone 805.523.9774 • Fax 805.523.9776
support@signum.com
signum.com

Purpose *This document describes the installation process for the RDI 1.5.1 (Remote Debug Interface) driver used with the Signum's JTAGjet-ARM emulator and the software configuration processes for IAR Embedded Workbench (EWARM) so it can be used with JTAGjet-ARN as one integrated development suite.*

Installing the Driver

1. Insert the *Development Tools for Microsoft Windows* CD into the CD-ROM drive. In the Master Setup dialog box, select RDI 1.5.1 Driver for Signum Emulators.

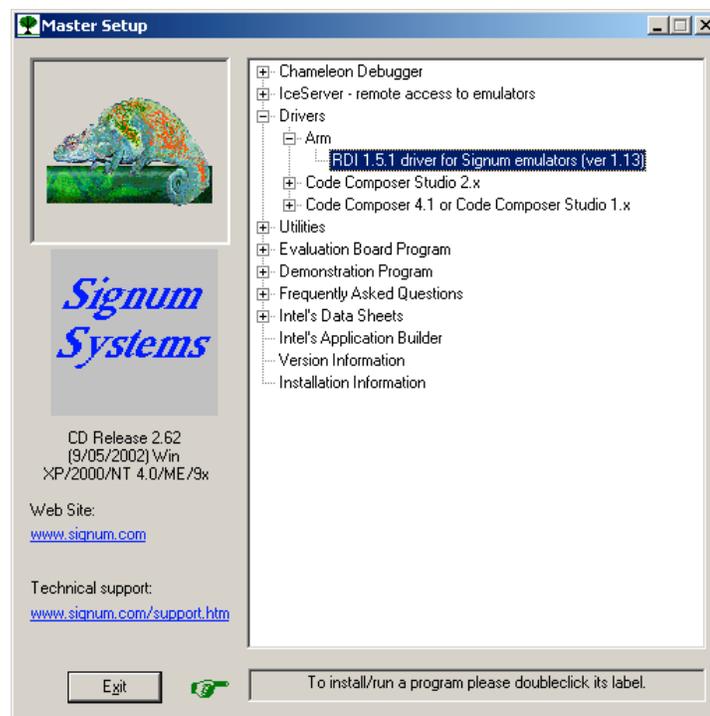


FIGURE 1 Master Setup dialog box.

2. When the Setup program starts, select the program folder for the program icons and the destination folder for the driver to be installed in.

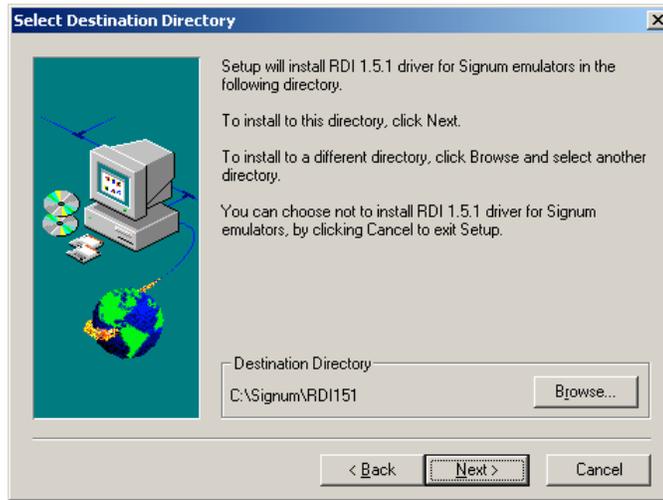


FIGURE 2 Selecting the folder for the driver.

In the confirmation dialog box, press the Next button to copy the driver files to your hard drive:

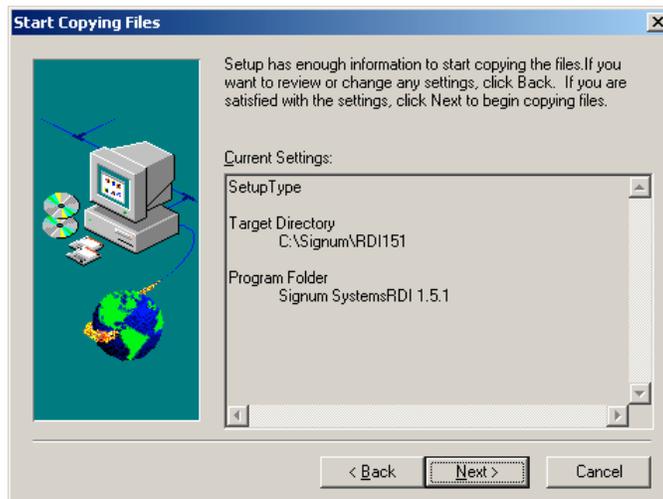


FIGURE 3 Confirming the driver location.

3. If you are running Windows XP, 2000 or NT 4.0, the parallel port driver (ParalUSP.sys), necessary to communicate with the emulator is installed. Restart your system if the Setup program asks you to do so.
4. Complete the installation process including the system re-start on Windows XP, 2000 and NT systems. Connect the emulator to a parallel port of your computer. For best performance, make sure that the port is configured for the EPP mode. Currently, the emulator supports only the EPP, Standard and Bi-Directional modes (no ECP).

5. Connect the JTAG cable to the JTAG connector on your ARM system or evaluation board. Use either the 20-pin or the 14-pin JTAG connector, depending on your target board. For examples of connecting the emulator with selected target boards, see *Appendix I*. Turn the emulator on first, and then turn on your target board. Finally, configure your debugger, as described in the *Configuring the EWARM* section.

Configuring the EWARM

With the ARM RDI 1.5.1 Driver for Signum Emulators installed, you are ready to configure your Embedded Workbench

IAR Embedded Workbench

To configure the IAR Embedded Workbench:

1. Choose Options from the Project menu. The Options for Target dialog box appears. Select the C-SPY category. In the Setup tab, choose RDI as the driver, as in Figure 4.

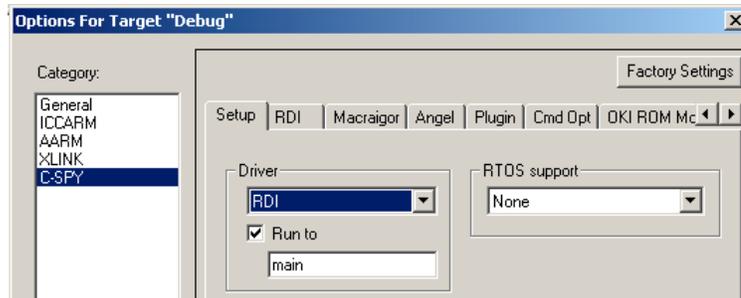


FIGURE 4 Selecting the driver type for C-Spy.

2. Select the RDI tab. In the Manufacturer RDI Driver edit box, navigate to the SigJdsRDI.dll file in the C:\Signum\RDI151 directory, as in Figure 5.

JTAGJET-ARM DRIVER FOR IAR EMBEDDED WORKBENCH

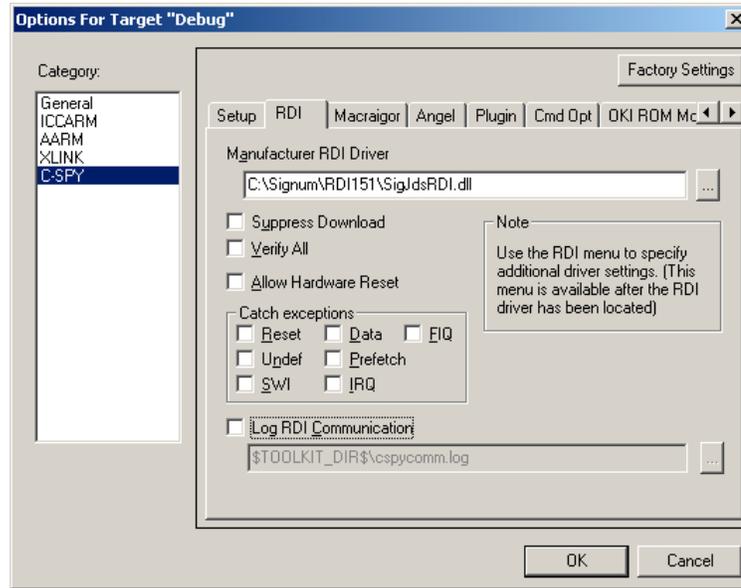


FIGURE 5 Selecting the RDI driver DLL file for C-Spy.

Press OK to close the dialog box.

3. In the RDI menu, select Configure to open the JDS Configuration dialog.

Note!

To modify the JTAGjet emulator configuration at a later time, make sure that the debugger is not running. Press the system Start button and select Signum Systems RD11.5.1 from the Programs menu. This will execute the RDIconfig configuration program.

Configuring the Connection

1. In the JDS Configuration dialog that appears, make sure that the Connection tab is selected.

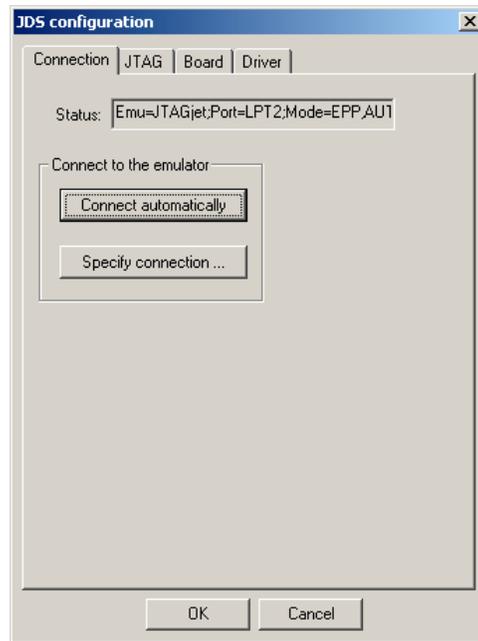


FIGURE 6 Configuring the emulator: the Connection tab.

- To establish a debugger-emulator connection with minimum effort, press the Connect Automatically button. The connection parameters appear in the Status box.
 - For added control over the connection, press the Specify Connection button to set up the communication port and its parameters manually.
2. Once the connection has been established, select the JTAG tab in the JTAG configuration dialog. Choose your target device from the CPU drop-down list.

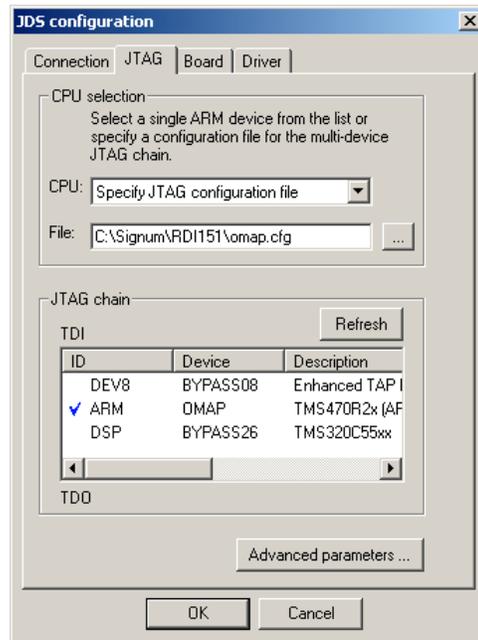


FIGURE 7 Configuring the emulator: the JTAG tab.

- If your JTAG chain contains a single ARM device, select the processor from the CPU list. The name (ID) and device type, along with a short description, are displayed in the JTAG Chain group box.

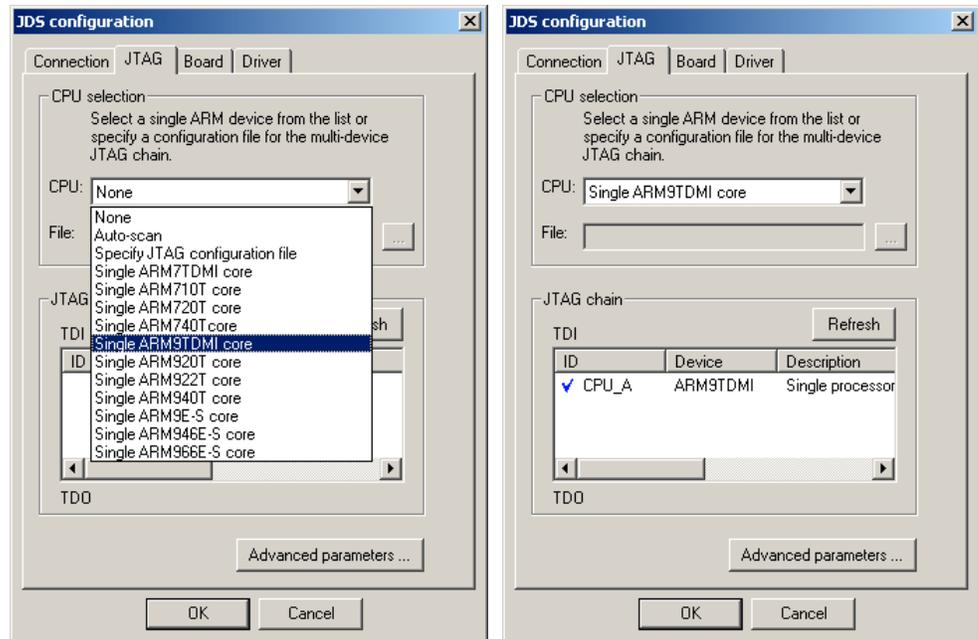


FIGURE 8 Selecting a single target CPU.

- If your JTAG chain contains multiple devices, load the chain definition from your JTAG configuration file. (For a description of the file's format, see section JTAG Chain Configuration File below). From the CPU drop-

JTAGJET-ARM DRIVER FOR IAR EMBEDDED WORKBENCH

down list in the JDS Configuration dialog, select Specify JTAG Configuration File. Enter the file path and name in the File text box. Alternatively, use the file browser button located next to the File text box to navigate to your file. After the file has been loaded, the chain description is displayed in the JTAG Chain group box. In the ID column, click on the device you want to debug. Note that you cannot select a bypass device.

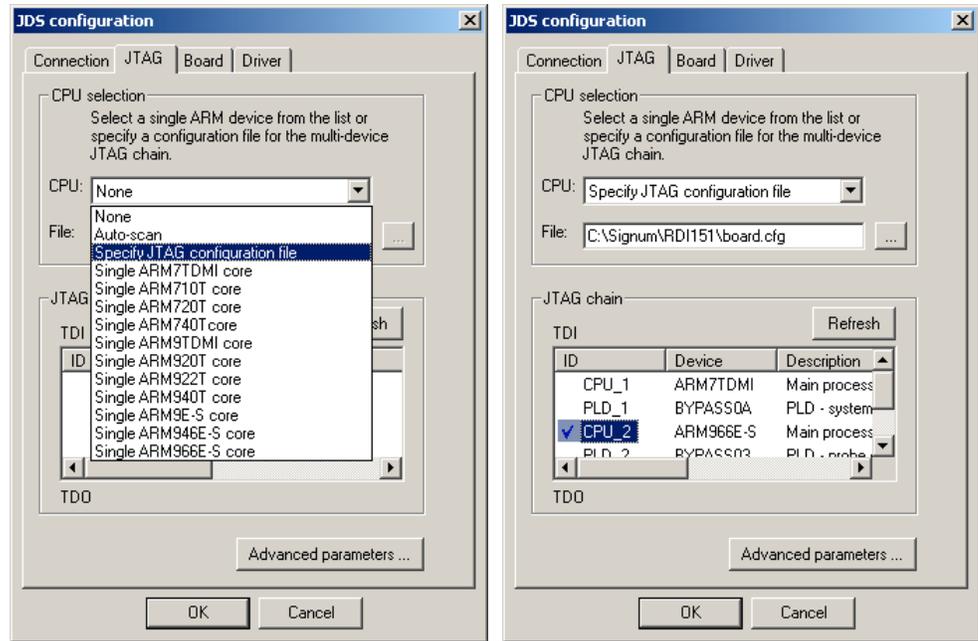


FIGURE 9 Selecting a multi-device JTAG chain.

3. Click the Board tab and browse for the startup macro file.

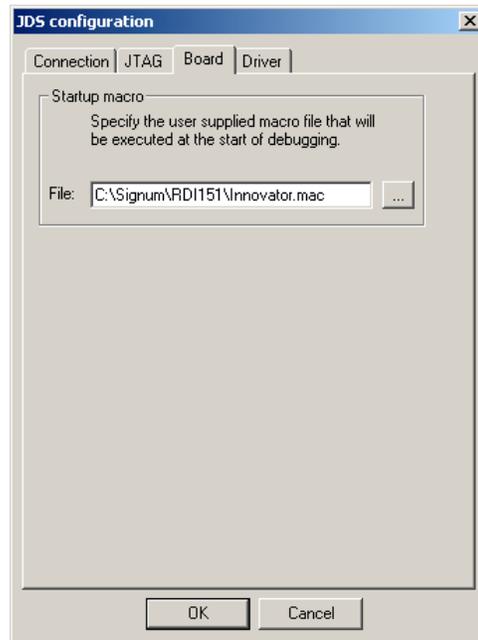


FIGURE 10 Configuring the emulator: The Board tab.

4. Click the Driver tab to configure driver protocol logs and error reports.
 - To display the driver protocol log in the Log window, select Log Enable. If you also need to store the log in a file, enter the file name in the File text box.
 - To enable the driver to generate descriptive error messages, select Show Error Messages. This option does not affect the way your debugger displays its own error messages; it is designed simply to augment and clarify those debugger messages that tend to be cryptic or are limited to error codes only.

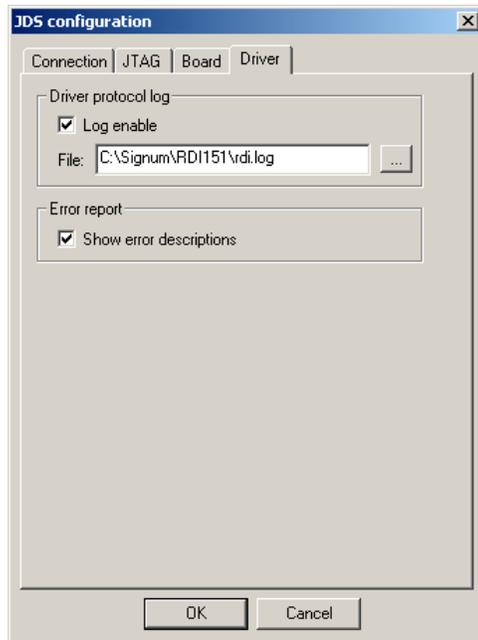


FIGURE 11 Configuring the emulator: the Driver tab.

5. Click OK. In the Choose Target dialog box, click OK again to accept your settings and connect to the target processor.

JTAG Chain Configuration File

A multi-device JTAG daisy chain is configured using a ASCII text JTAG chain configuration file with extension .cfg. With the exception of comment lines, each line in a .cfg file refers to a separate device. Thus in general, the file format is as follows:

```

ID1   Device1   Description1
ID2   Device2   Description2
.
.
.
etc.,
    
```

where

<i>ID</i>	Is a unique name identifying the device, e.g., "CPU_1", including the double quotes.
<i>Device</i>	Is the type of the device, such as ARM7TDMI. Non-ARM devices should be bypassed, and specified as BYPASSxx, where xx denotes the length of the instruction register

Description in a two-digit format. For instance, BYPASS0A denotes a bypassed device with a 10-bit (0a hex) instruction register. Is a coment text.

In general, enclose in double quotes all names, words or phrases that contain non-alphanumeric characters. Lines that begin with the semicolon (;) are treated as comments and are ignored. The order in which the JTAG devices are specified in the configuration file is significant: the first line corresponds to the device closest to the TDI, the second one to the next device in the chain, and so on. Finally, the last line describes the device on the TDO side of the chain.

Board Startup Macro File

Some application boards may require to be properly set up before a debug session can begin. Use a startup macro to configure on-board memory before the debugger attempts to access that memory. The macro is also a good place for disabling the watchdog that otherwise may reset the CPU soon after the debugger starts running.

A board startup configuration file is an ASCII text file with the .mac extension. The following commands can be placed in that file:

- | | |
|---------------------------|---|
| sd <address> = <value> | Write a <value> of DWORD (32bit) type to memory at address <address>. |
| sw <address> = <value> | Write a <value> of WORD (16bi) type to memory at address <address>. |
| sb <address> = <value> | Write a <value> of BYTE (8bit) type to memory at address <address>. |
| reset | Reset the CPU. |
| emu <parameter> = <value> | Set one of the following emulation parameters: <ul style="list-style-type: none"> • cmdline (argument string for the debugee) • semihosting_enabled (0 – disabled, 1 – enabled) |

Empty lines, spaces and comments starting with a semicolon are ignored.

The following is an example of a typical startup macro file.

JTAGJET-ARM DRIVER FOR IAR EMBEDDED WORKBENCH

```
; MyTarget.mac - RDI Startup macro for MyTarget board.
; Version 1.00 7/10/02 - Initial version

; Initialize memory
sd 0xFFFECC10 = 0x00203339 ; CS0 configuration
sd 0xFFFECC14 = 0x00001139 ; CS1 configuration
sd 0xFFFECC18 = 0x00001139 ; CS2 configuration
sd 0xFFFECC1C = 0x00001139 ; CS3 configuration

; Disable ARM watchdog
sw 0xFFFECC808 = 0x00F5
sw 0xFFFECC808 = 0x00A0

; Configure the emulator
emu semihosting_enabled=0 ; disable semihosting

; End of file 'MyTarget.mac'
```

Appendix I

CONNECTING THE JTAG PROBE TO SELECTED ARM TARGET BOARDS

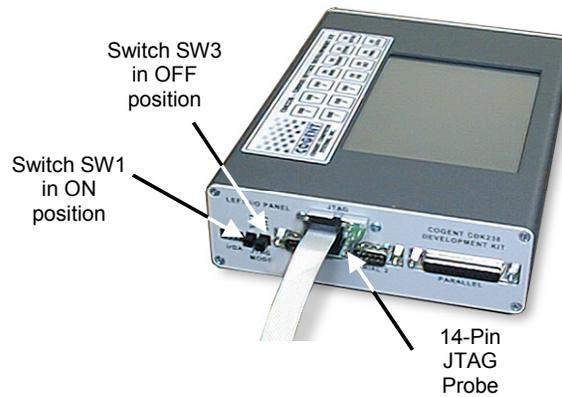


FIGURE 12 COGENT CDK238–Cirrus EP73xx Development Kit. Note the correct positions of SW1 and SW3 switches.

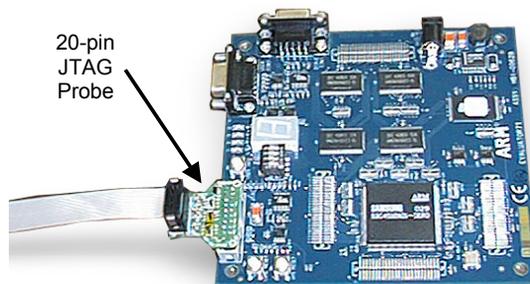


FIGURE 13 ARM Evaluator-7T Board with the Samsung KS32C50100 microcontroller.

JTAGJET-ARM DRIVER FOR IAR EMBEDDED WORKBENCH

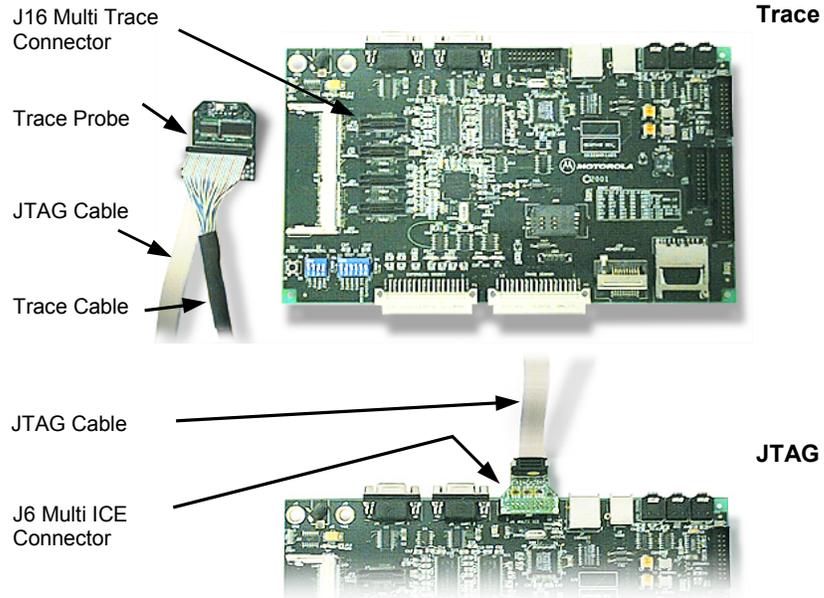


FIGURE 14 Motorola X9328MX1ADS Target Board. Trace Option (top): insert the trace probe with the JTAG cable attached to it into the J16 Multi Trace connector. JTAG-Only Option (bottom): insert the JTAG probe into the J6 Multi ICE connector.

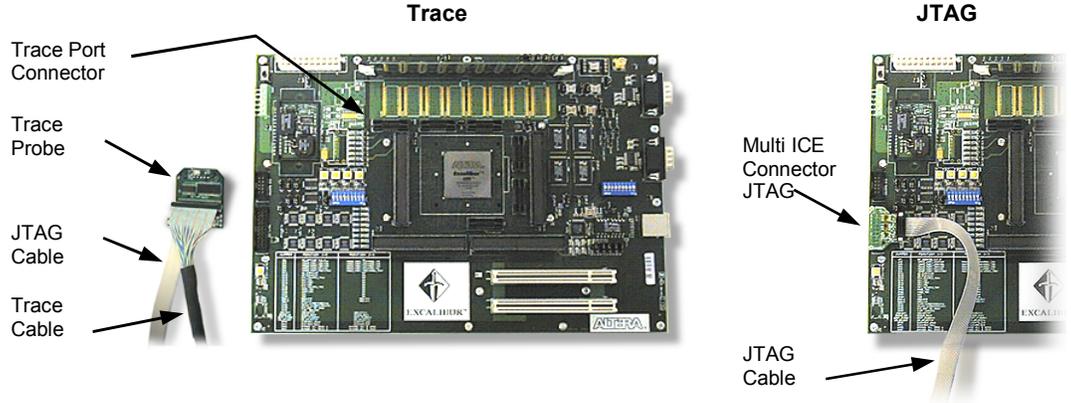


FIGURE 15 Altera Excalibur Target Board. Trace Option (left): insert the trace probe with the JTAG cable attached to it into Trace Port connector. JTAG-Only Option (right): insert the JTAG probe into the Multi ICE connector.

□