DSPCOMMPCIUM/D
Rev. 1.1, 07/1999

Suite56™ PCI Command Converter User's Manual

Motorola, Incorporated
Semiconductor Products Sector
6501 William Cannon Drive West
Austin TX 78735-8598

MOTOROLA

This document contains information on a new product.

Table of Contents

List of Figures

Chapter 1
Suite56™ PCI Command Converter

The Motorola Suite56™ PCI Command Converter is an interface device that, with the assistance of the Motorola Suite56 DSP Software Development Tools Debugger, allows you to send and receive information from your host system to your Motorola DSP based system. The interface is from the PCI bus on the host system, through a target interface module, to the DSP via a JTAG connection.

Note: For more information on the Motorola DSP Suite56 Software Development Tools Debugger, refer to the *Suite56 Software Development Debugger User's Manual* from the Motorola Suite56 Software Development Tools CD or the DSP Tools website on the World Wide Web at the following web address:

http://www.motorola.com/SPS/DSP/tools/documentation

1.1 General Description

The Suite56 PCI Command Converter is used for designing, debugging, and evaluating DSP-based systems. Figure 1-1 shows how the Command Converter can be connected in a target system configuration and used as a hardware evaluation tool or as a software accelerator.

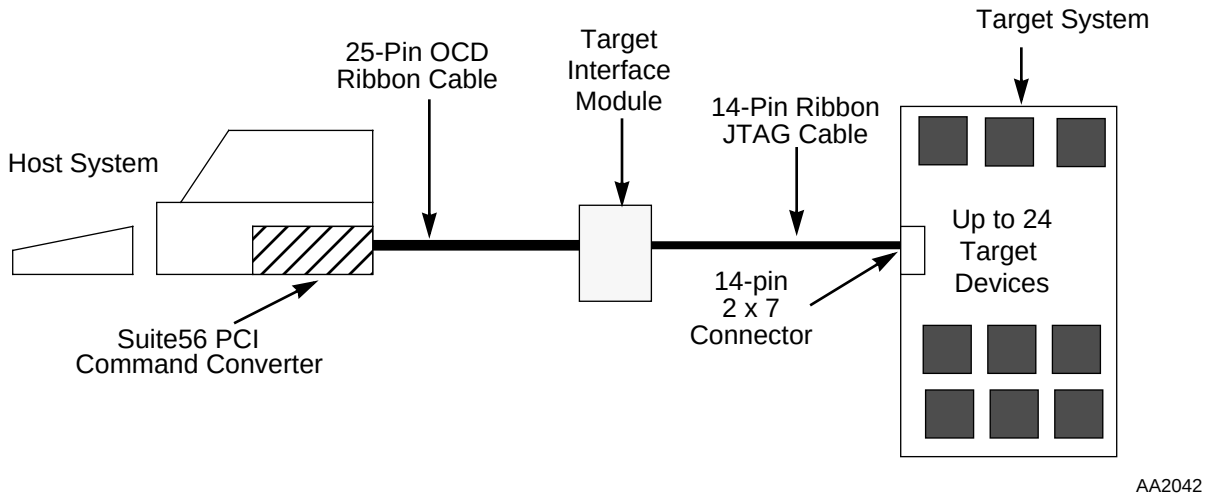


Figure 1-1. Command Converter to Target System Configuration

The target system in Figure 1-1 can be a Motorola DSP application or evaluation board, or any user defined target system. The user defined target hardware must have an access point for the 14-pin JTAG ribbon cable, that may be as simple as a 2-row × 7-set of test points.

The Suite56 PCI Command Converter provides a physical link between the host system and the target via a PCI bus connection on the host card. This link translates the high-level debugger commands into JTAG signals that enable the host system to reset, interrupt, and send commands to the target DSP.

1.2 Operating Environment

The Motorola DSP Suite56 Software Development Tools used for this Suite56 PCI Command Converter supports the following three host systems:

- Windows PC
- Hewlett Packard Workstations
- Sun™ Ultra Workstations

1.2.1 Windows PC Requirements

The minimum hardware requirements for using the Motorola DSP Suite56 Software Development Tools Debugger on a Windows PC include the following:

- Windows PC (Pentium® class processor, or higher)
- Windows® 95 operating system with 16 Mbytes of RAM, Windows® 98 operating system with 32 Mbytes of RAM, or Windows® NT 4.0 operating system with 32 Mbytes of RAM
- CD-ROM drive
- Hard drive with 50 Mbytes of free space
- Mouse and keyboard
- Free PCI bus slot

1.2.2 Hewlett Packard Workstation Requirements

The minimum hardware requirements for using the Motorola DSP Suite56 Software Development Tools Debugger on an HP Workstation include the following:

- HP Workstation running HP-UX version 9.05 or 10.20 (or higher)
- 32 Mbytes of RAM
- CD-ROM drive
- Hard drive with 50 Mbytes of free space
- Mouse and keyboard
- Free PCI bus slot

1.2.3 Sun Ultra Workstation Requirements

The minimum hardware requirements for using the Motorola DSP Suite56 Software Development Tools Debugger on a Sun Ultra Workstation include the following:

- Solaris™ Release 2.5.1 (or higher)
- 32 Mbytes of RAM
- CD-ROM drive
- Hard drive with 50 Mbytes of free space
- Mouse and keyboard
- Free PCI bus slot

Chapter 2
Functional Description

The Suite56 PCI Command Converter interacts with the target DSP with the assistance of the Suite56 Software Development Tools Debugger and the user's host system. The host system interface consists of a program written in the C language that sends commands via a PCI bus to the Suite56 PCI Command Converter. Commands entered from the host system's keyboard are parsed, and a series of low-level command packets are sent to the Suite56 PCI Command Converter. The Suite56 PCI Command Converter translates these low-level command packets into one or more JTAG signals and OnCE commands that are transferred to the target DSP via its JTAG port. The JTAG port provides the necessary control to the target so programs may be loaded or saved, registers read or modified, and hardware breakpoints set or cleared.

2.1 Command Converter Interface Connector

The target application board must have a 14-pin connector to interface to the command converter controller. This interface comprises eight signals and three ground connections on a 7-row × 2-column male pin header, which are spaced on one-tenth inch centers as illustrated in Figure 2-1 on page 2-1.

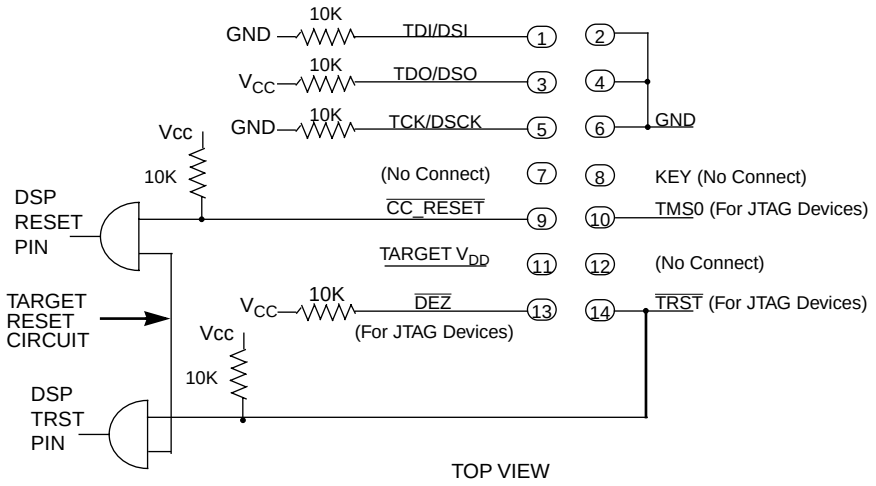


Figure 2-1. Target V_{DD} System JTAG/OnCE™ Interface Connector

Note: Figure 3-1 may not represent the exact JTAG/OnCE configuration needed for every DSP due to possible differences in internal resistors on the pins. Please see the Data Sheet for your DSP for information on internal resistors.

Since the target system will have a resident reset circuit, an AND gate-in-series is recommended with the $\overline{\text{CC\_RESET}}$ signal. This will allow the target DSP to be reset with a valid V_{OL} level from either the target-reset circuit or from the command converter. The pull-down resistors are provided to prevent false signals from being propagated to the JTAG/OnCE circuit when the test data input/debug serial input (TDI/DSI) and test data clock/debug serial clock (TCK/DSCK) lines are active. The test data out/debug serial output (TDO/DSO) pull-up is designed to deassert the Debug Acknowledge signal from the OnCE circuit

Figure 2-2 is the connector coming from the Target Interface Module via a 14-pin ribbon cable. When viewed with pin 8 (keyed) on the right side, this plug connector has all odd numbers on the left side and all even numbers on the right side when viewed from the top. Spacing between pins is one-tenth inch.

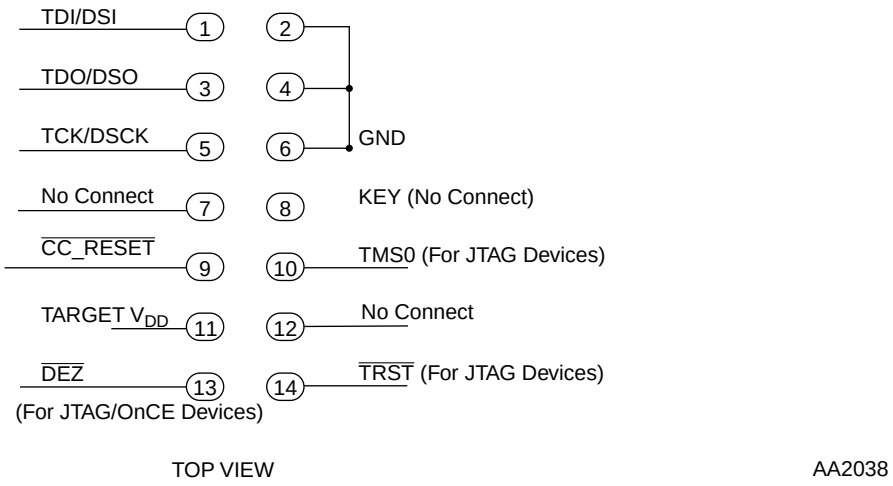


Figure 2-2. Target Interface Module's 14-Pin JTAG/OnCE Connector

2.2 Multiple Target Connections

Multiple target devices may be connected in series, allowing a single command converter, JTAG/OnCE connector to control multiple devices, as in Figure 2-3. Data flows from the JTAG host, into each JTAG implementation through TDI, out through TDO and into TDI in the next chip, eventually returning to the JTAG host.

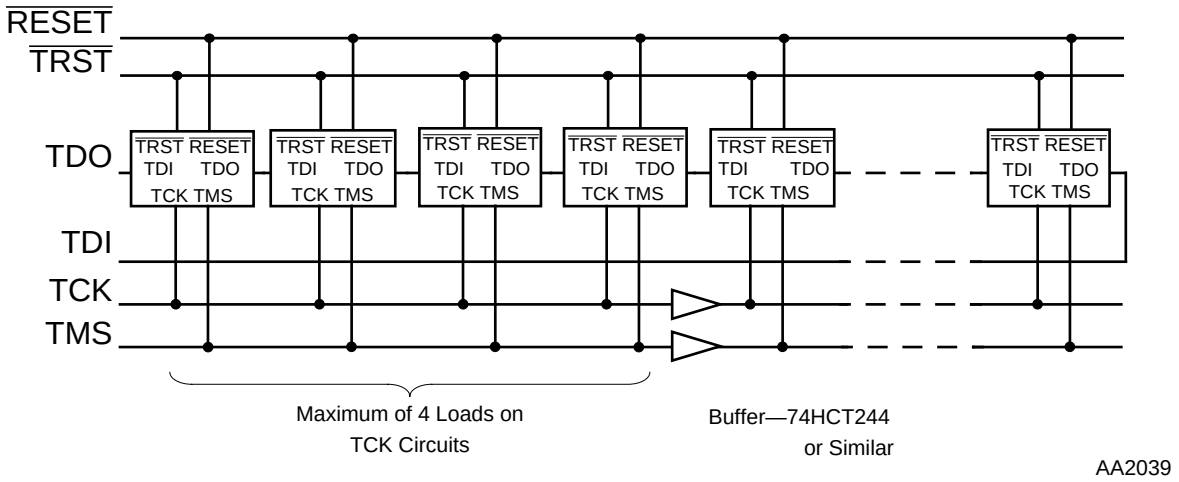


Figure 2-3. Multiple JTAG Target Connections (Serial Method)

2.3 TCK Drive and Timing Considerations

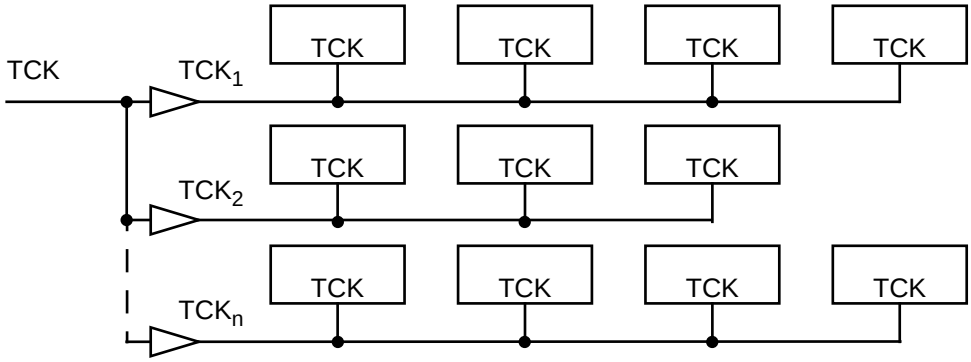
The signals from the command converter are TDO, TCK and TMS, and $\overline{\text{TRST}}$. The TCK signal requires fast rise and fall times dictated by the TCK pin timing specification, and consequently attention must be given to the drive capability of the circuits driving this signal.

When driving the TCK and TMS circuits with a large number of target devices, the user must pay attention to the rise and fall times of TCK and TMS. Excessive capacitance may cause communication problems when driving a single circuit that connects multiple TCK or multiple TMS input pins. Excessive capacitance can also cause communication problems with a single circuit connecting multiple TMS input pins.

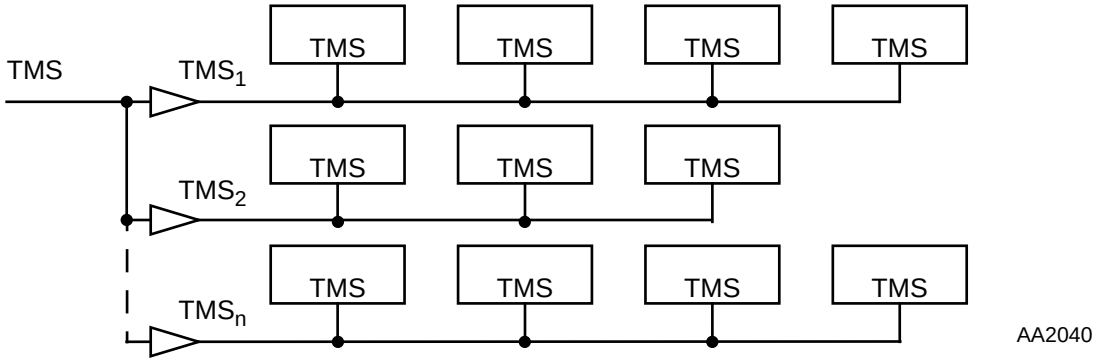
Acceptable transition times may be achieved for TCK and TMS by driving no more than four JTAG inputs from each buffered output. This may be achieved with two configurations.

Figure 2-3 shows one method. Here (in effect) one signal connects each of the TCK inputs, and one signal connects each of the TMS inputs. A buffer is placed in the circuit after each fourth input at most, to restore the signal quality for subsequent inputs. The propagation delay of the buffer is not significant.

Figure 2-4 shows two possible configurations of a second method that also enables signal quality to meet the requirements. In Configuration Number 1, the signal is split and buffered into a number of parallel TCKn signals. Each of these signals may drive up to four TCK inputs.



Configuration Number 1 —Fan Out of TCK at Source



Configuration Number 2 —Fan Out of TMS at Source

Figure 2-4. Multiple JTAG Connectors (Parallel Method)

In Configuration Number 2, the signal is split and buffered into a number of parallel TMSn signals. Each of these signals may drive up to four TMS inputs.

Each method is equally valid. The choice of methods will depend on practical considerations related to each project.

2.4 Resetting Target DSP Devices

The $\overline{\text{RESET}}$ signal and $\overline{\text{TRST}}$ signals from the command converter are typically connected to all target DSP devices on a JTAG chain. All DSP devices on a JTAG chain connected to a specific command converter device are reset when $\overline{\text{RESET}}$ and $\overline{\text{TRST}}$ are asserted by the debugger command FORCE R. Execution control is established immediately after

deassertion of $\overline{\text{TRST}}$ and before any instructions are executed. This sequence of events is illustrated in Figure 2-5.

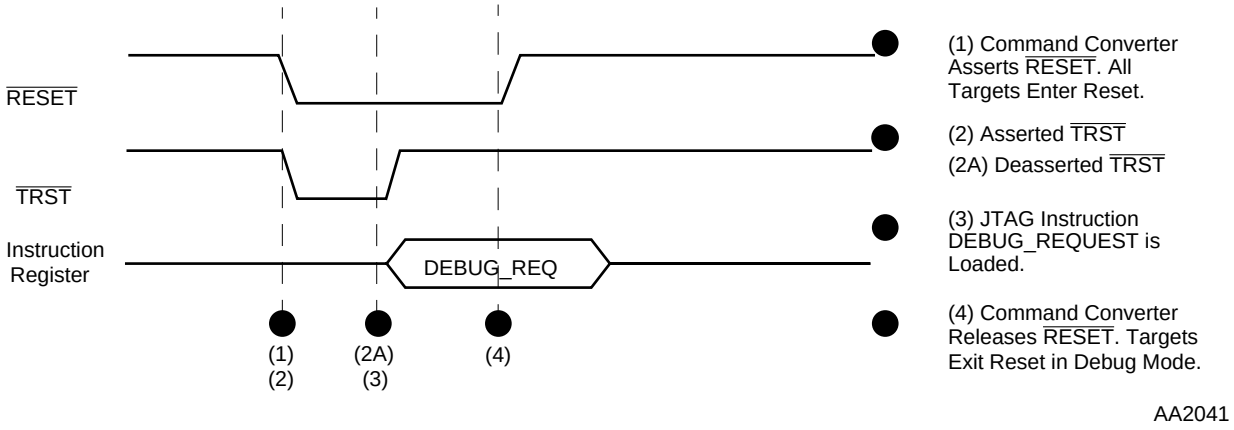


Figure 2-5. Reset JTAG Device with $\overline{\text{RESET}}$ Signal

The JTAG controller is still active after deassertion of $\overline{\text{TRST}}$ during reset and while $\overline{\text{RESET}}$ is held low. The JTAG special instruction DEBUG_REQ is clocked in before $\overline{\text{RESET}}$ is deasserted. When $\overline{\text{RESET}}$ is deasserted, the device is immediately in Debug mode, and no instructions are executed in the DSP.