TMS320C6000 DSP Designing for JTAG Emulation Reference Guide

Literature Number: SPRU641
July 2003

Texas Instruments

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Preface

Read This First

About This Manual

This document assists you in meeting the design requirements of the XDS510™ emulator with respect to JTAG designs and discusses the XDS510 cable (manufacturing part number 2617698-0001). This cable is identified by a label on the cable pod marked **JTAG 3/5 V** and supports both standard 3-volt and 5-volt target system power inputs.

The term *JTAG* as used in this document refers to Texas Instruments scan-based emulation, which is based on the IEEE 1149.1 standard.

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.

Related Documentation From Texas Instruments

The following documents describe the C6000™ devices and related support tools. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

TMS320C6000 CPU and Instruction Set Reference Guide (literature number SPRU189) describes the TMS320C6000™ CPU architecture, instruction set, pipeline, and interrupts for these digital signal processors.

TMS320C6000 Peripherals Reference Guide (literature number SPRU190) describes the peripherals available on the TMS320C6000™ DSPs.

TMS320C6000 Technical Brief (literature number SPRU197) gives an introduction to the TMS320C62x™ and TMS320C67x™ DSPs, development tools, and third-party support.

TMS320C64x Technical Overview (SPRU395) gives an introduction to the TMS320C64x™ DSP and discusses the application areas that are enhanced by the TMS320C64x VelociTI™.

TMS320C6000 Programmer's Guide (literature number SPRU198) describes ways to optimize C and assembly code for the TMS320C6000™ DSPs and includes application program examples.

TMS320C6000 Code Composer Studio Tutorial (literature number SPRU301) introduces the Code Composer Studio™ integrated development environment and software tools.

Code Composer Studio Application Programming Interface Reference Guide (literature number SPRU321) describes the Code Composer Studio™ application programming interface (API), which allows you to program custom plug-ins for Code Composer.

TMS320C6x Peripheral Support Library Programmer's Reference (literature number SPRU273) describes the contents of the TMS320C6000™ peripheral support library of functions and macros. It lists functions and macros both by header file and alphabetically, provides a complete description of each, and gives code examples to show how they are used.

TMS320C6000 Chip Support Library API Reference Guide (literature number SPRU401) describes a set of application programming interfaces (APIs) used to configure and control the on-chip peripherals.

Trademarks

Contents

1 Designing Your Target System's Emulator Connector (14-Pin Header) 7

2 Bus Protocol 9

3 IEEE 1149.1 Standard 9

4 JTAG Emulator Cable Pod Logic 10

5 JTAG Emulator Cable Pod Signal Timing 11

6 Emulation Timing Calculations 12

7 Connections Between the Emulator and the Target System 14
7.1 Buffering Signals 14
7.2 Using a Target-System Clock 16
7.3 Configuring Multiple Processors 17

8 Mechanical Dimensions for the 14-Pin Emulator Connector 18

9 Emulation Design Considerations 20
9.1 Using Scan Path Linkers 20
9.2 Emulation Timing Calculations for SPL 22
9.3 Using Emulation Pins 24
9.4 Performing Diagnostic Applications 28

Figures

1 14-Pin Header Signals and Header Dimensions 7
2 JTAG Emulator Cable Pod Interface 10
3 JTAG Emulator Cable Pod Timing Diagram 11
4 Target-System-Generated Test Clock 16
5 Multiprocessor Connections 17
6 Pod/Connector Dimensions 18
7 14-Pin Connector Dimensions 19
8 Connecting a Secondary JTAG Scan Path to an SPL 21
9 EMU0/1 Configuration 25
10 EMU0/1 Configuration With Additional AND Gate to Meet Timing Requirements 27
11 Suggested Timings for the EMU0 and EMU1 Signals 27
12 EMU0/1 Configuration Without Global Stop 28
13 TBC Emulation Connections for n JTAG Scan Paths 29

Tables

1 14-Pin Header Signal Descriptions 8
2 Emulator Cable Pod Timing Parameters 11

Designing for JTAG Emulation

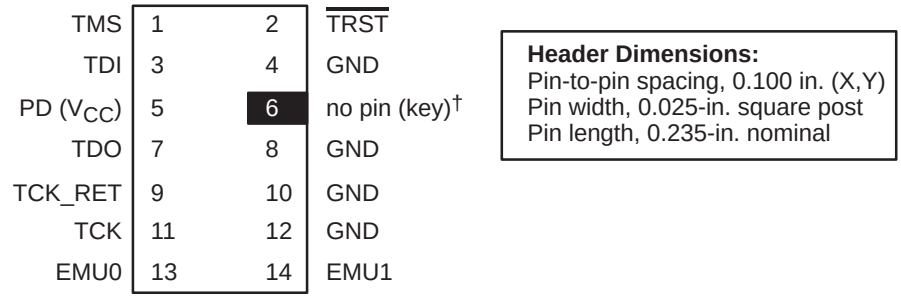
This document assists you in meeting the design requirements of the XDS510™ emulator with respect to JTAG designs and discusses the XDS510 cable (manufacturing part number 2617698-0001). This cable is identified by a label on the cable pod marked **JTAG 3/5 V** and supports both standard 3-volt and 5-volt target system power inputs.

The term *JTAG* as used in this document refers to Texas Instruments scan-based emulation, which is based on the IEEE 1149.1 standard.

1 Designing Your Target System's Emulator Connector (14-Pin Header)

JTAG target devices support emulation through a dedicated emulation port. This port is a superset of the IEEE 1149.1 standard and is accessed by the emulator. To communicate with the emulator, **your target system must have a 14-pin header** (two rows of seven pins) with the connections that are shown in Figure 1. Table 1 describes the emulation signals.

Figure 1. 14-Pin Header Signals and Header Dimensions



† While the corresponding female position on the cable connector is plugged to prevent improper connection, the cable lead for pin 6 is present in the cable and is grounded, as shown in the schematics and wiring diagrams in this document.

Table 1. 14-Pin Header Signal Descriptions

Signal	Description	Emulator† State	Target† State
TMS	Test mode select	O	I
TDI	Test data input	O	I
TDO	Test data output	I	O
TCK	Test clock. TCK is a 10.368-MHz clock source from the emulation cable pod. This signal can be used to drive the system test clock	O	I
$\overline{\text{TRST}}$‡	Test reset	O	I
EMU0	Emulation pin 0	I	I/O
EMU1	Emulation pin 1	I	I/O
PD(V_{CC})	Presence detect. Indicates that the emulation cable is connected and that the target is powered up. PD should be tied to V_{CC} in the target system.	I	O
TCK_RET	Test clock return. Test clock input to the emulator. May be a buffered or unbuffered version of TCK.	I	O
GND	Ground		

† I = input; O = output

‡ Do not use pullup resistors on $\overline{\text{TRST}}$: it has an internal pulldown device. In a low-noise environment, $\overline{\text{TRST}}$ can be left floating. In a high-noise environment, an additional pulldown resistor may be needed. (The size of this resistor should be based on electrical current considerations.)

Although you can use other headers, recommended parts include:

straight header, unshrouded — DuPont Connector Systems part numbers: 65610−114, 65611−114, 67996−114, 67997−114

2 Bus Protocol

The IEEE 1149.1 specification covers the requirements for the test access port (TAP) bus slave devices and provides certain rules, summarized as follows:

- The TMS/TDI inputs are sampled on the rising edge of the TCK signal of the device.
- The TDO output is clocked from the falling edge of the TCK signal of the device.

When these devices are daisy-chained together, the TDO of one device has approximately a half TCK cycle setup to the next device's TDI signal. This type of timing scheme minimizes race conditions that would occur if both TDO and TDI were timed from the same TCK edge. The penalty for this timing scheme is a reduced TCK frequency.

The IEEE 1149.1 specification does not provide rules for bus master (emulator) devices. Instead, it states that it expects a bus master to provide bus slave compatible timings. The XDS510 provides timings that meet the bus slave rules.

3 IEEE 1149.1 Standard

For more information concerning the IEEE 1149.1 standard, contact IEEE Customer Service:

Address:
IEEE Customer Service
445 Hoes Lane, PO Box 1331
Piscataway, NJ 08855-1331

Phone:
in the US and Canada: (800) 678−IEEE
outside the US and Canada: (908) 981−1393

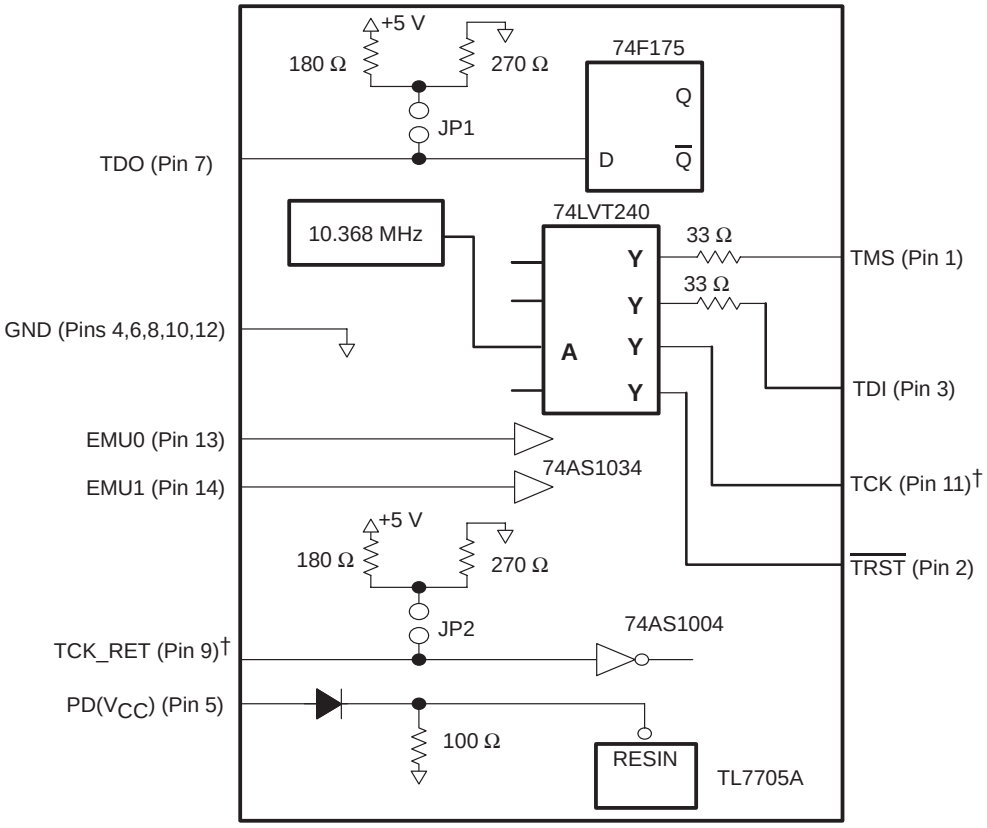
FAX: (908) 981−9667
Telex: 833233

4 JTAG Emulator Cable Pod Logic

Figure 2 shows a portion of the emulator cable pod. These are the functional features of the pod:

- Signals TDO and TCK_RET can be parallel-terminated inside the pod if required by the application. By default, these signals are not terminated.
- Signal TCK is driven with a 74LVT240 device. Because of the high-current drive (32 mA I_{OL}/I_{OH}), this signal can be parallel-terminated. If TCK is tied to TCK_RET, then you can use the parallel terminator in the pod.
- Signals TMS and TDI can be generated from the falling edge of TCK_RET, according to the IEEE 1149.1 bus slave device timing rules.
- Signals TMS and TDI are series-terminated to reduce signal reflections.
- A 10.368-MHz test clock source is provided. You may also provide your own test clock for greater flexibility.

Figure 2. JTAG Emulator Cable Pod Interface



† The emulator pod uses TCK_RET as its clock source for internal synchronization. TCK is provided as an optional target system test clock source.

5 JTAG Emulator Cable Pod Signal Timing

Figure 3 shows the signal timings for the emulator cable pod. Table 2 defines the timing parameters. These timing parameters are calculated from values specified in the standard data sheets for the emulator and cable pod and are for reference only. Texas Instruments does not test or guarantee these timings.

The emulator pod uses TCK_RET as its clock source for internal synchronization. TCK is provided as an optional target system test clock source.

Figure 3. JTAG Emulator Cable Pod Timing Diagram

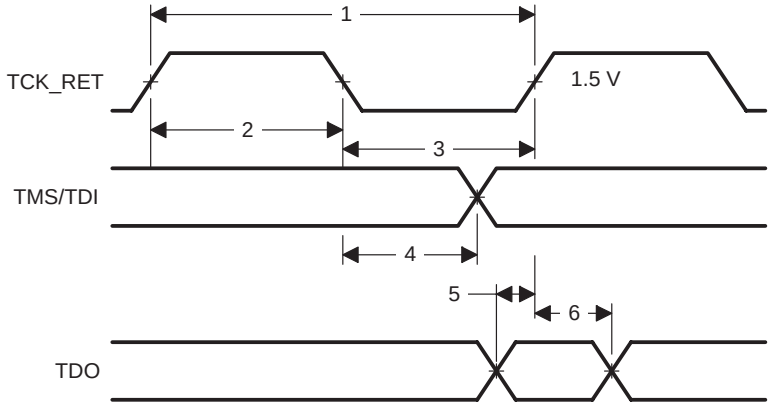


Table 2. Emulator Cable Pod Timing Parameters

No.	Reference	Description	Min	Max	Units
1	$t_{c(TCK)}$	TCK_RET period	35	200	ns
2	$t_{w(TCKH)}$	TCK_RET high-pulse duration	15		ns
3	$t_{w(TCKL)}$	TCK_RET low-pulse duration	15		ns
4	$t_{d(TMS)}$	Delay time, TMS/TDI valid from TCK_RET low	6	20	ns
5	$t_{su(TDO)}$	TDO setup time to TCK_RET high	3		ns
6	$t_{h(TDO)}$	TDO hold time from TCK_RET high	12		ns

6 Emulation Timing Calculations

The following examples help you calculate emulation timings in your system. For actual target timing parameters, see the appropriate device data sheets.

Assumptions:

$t_{su(TTMS)}$	Target TMS/TDI setup to TCK high	10 ns
$t_{d(TTDO)}$	Target TDO delay from TCK low	15 ns
$t_{d(bufmax)}$	Target buffer delay, maximum	10 ns
$t_{d(bufmin)}$	Target buffer delay, minimum	1 ns
$t_{(bufskew)}$	Target buffer skew between two devices in the same package: $[t_{d(bufmax)} - t_{d(bufmin)}] \times 0.15$	1.35 ns
$t_{(TCKfactor)}$	Assume a 40/60 duty cycle clock	0.4 (40%)

Given in Table 2 (on page 7):

$t_{d(TMSmax)}$	Emulator TMS/TDI delay from TCK_RET low, maximum	20 ns
$t_{su(TDOmin)}$	TDO setup time to emulator TCK_RET high, minimum	3 ns

There are two key timing paths to consider in the emulation design:

- The TCK_RET-to-TMS/TDI path, called $t_{pd(TCK\_RET-TMS/TDI)}$, and
- The TCK_RET-to-TDO path, called $t_{pd(TCK\_RET-TDO)}$.

Of the following two cases, the worst-case path delay is calculated to determine the maximum system test clock frequency.

Case 1: Single processor, direct connection, TMS/TDI timed from TCK_RET low.

$$t_{pd(TCK\_RET-TMS/TDI)} = \frac{\left[t_{d(TMSmax)} + t_{su(TTMS)}\right]}{t_{(TCKfactor)}} = \frac{[20ns + 10ns]}{0.4} = 75ns\ (13.3\ MHz)$$

$$t_{pd(TCK\_RET-TDO)} = \frac{\left[t_{d(TTDO)} + t_{su(TDOmin)}\right]}{t_{(TCKfactor)}} = \frac{[15ns + 3ns]}{0.4} = 45ns\ (22.2\ MHz)$$

In this case, the TCK_RET-to-TMS/TDI path is the limiting factor.

Case 2: Single/multiprocessor, TMS/TDI/TCK buffered input, TDO buffered output, TMS/TDI timed from TCK_RET low.

$$t_{pd\,(TCK\_RET-TMS/TDI)} = \frac{\left[t_{d\,(TMSmax)} + t_{su\,(TTMS)} + t_{(bufskew)}\right]}{t_{(TCKfactor)}}$$

$$= \frac{\left[20ns + 10ns + 1.35ns\right]}{0.4}$$

$$= 78.4ns\ (12.7\ MHz)$$

$$t_{pd\,(TCK\_RET-TDO)} = \frac{\left[t_{d\,(TTDO)} + t_{su\,(TDOmin)} + t_{d\,(bufmax)}\right]}{t_{(TCKfactor)}}$$

$$= \frac{[15ns + 3ns + 10ns]}{0.4}$$

$$= 70ns\ (14.3\ MHz)$$

In this case, the TCK_RET-to-TMS/TDI path is the limiting factor.

In a multiprocessor application, it is necessary to ensure that the EMU0−1 lines can go from a logic low level to a logic high level in less than 10 µs. This can be calculated as follows:

$$t_r = 5(R_{pullup} \times N_{devices} \times C_{load\_per\_device})$$
$$= 5(4.7\ k\Omega \times 16 \times 15\ pF)$$
$$= 5.64\ \mu s$$

Refer to the device datasheet for the actual R_{pullup} value.

7 Connections Between the Emulator and the Target System

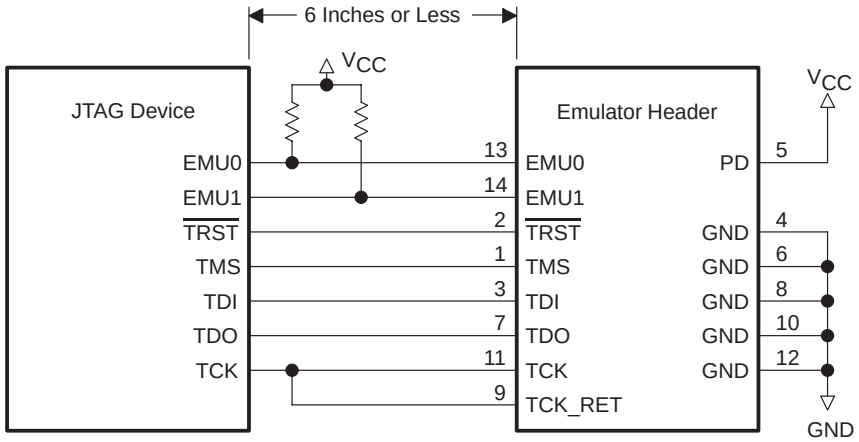
It is extremely important to provide high-quality signals between the emulator and the JTAG target system. Depending upon the situation, you must supply the correct signal buffering, test clock inputs, and multiple processor interconnections to ensure proper emulator and target system operation.

Signals applied to the EMU0 and EMU1 pins on the JTAG target device can be either input or output (I/O). In general, these two pins are used as both input and output in multiprocessor systems to handle global run/stop operations. EMU0 and EMU1 signals are applied only as inputs to the XDS510 emulator header.

7.1 Buffering Signals

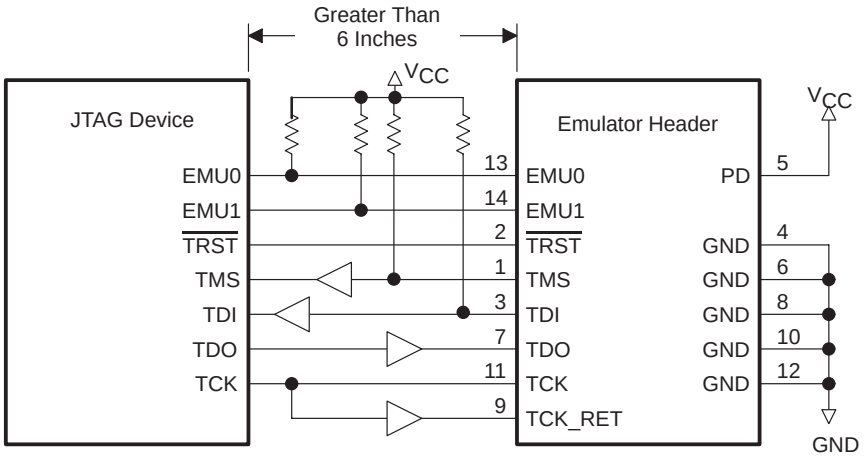
If the distance between the emulation header and the JTAG target device is greater than 6 inches, the emulation signals must be buffered. If the distance is less than 6 inches, no buffering is necessary. The following illustrations depict these two situations.

- **No signal buffering.** In this situation, the distance between the header and the JTAG target device should be no more than 6 inches.



The EMU0 and EMU1 signals must have pullup resistors connected to V_{CC} to provide a signal rise time of less than 10 µs. Refer to the device datasheet for the recommended resistor value.

❑ **Buffered transmission signals.** In this situation, the distance between the emulation header and the processor is greater than 6 inches. Emulation signals TMS, TDI, TDO, and TCK_RET are buffered through the same package.

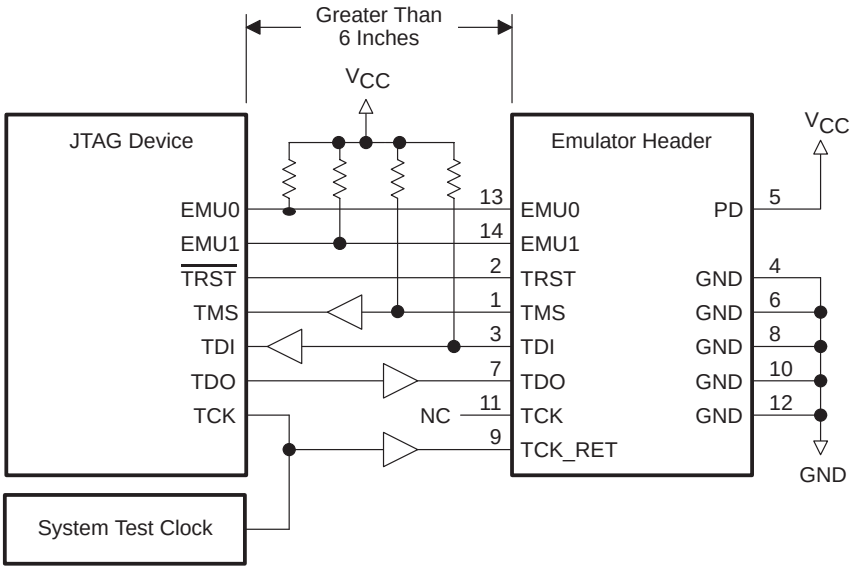


- The EMU0 and EMU1 signals must have pullup resistors connected to V_{CC} to provide a signal rise time of less than 10 µs. Refer to the device datasheet for the recommended resistor value.
- The input buffers for TMS and TDI should have pullup resistors connected to V_{CC} to hold these signals at a known value when the emulator is not connected. Refer to the device datasheet for the recommended resistor value.
- To have high-quality signals (especially the processor TCK and the emulator TCK_RET signals), you may have to employ special care when routing the PWB trace. You also may have to use termination resistors to match the trace impedance. The emulator pod provides optional internal parallel terminators on the TCK_RET and TDO. TMS and TDI provide fixed series termination.
- Since $\overline{\text{TRST}}$ is an asynchronous signal, it should be buffered as needed to insure sufficient current to all target devices.

7.2 Using a Target-System Clock

Figure 4 shows an application with the system test clock generated in the target system. In this application, the TCK signal is left unconnected.

Figure 4. Target-System-Generated Test Clock



Note: When the TMS/TDI lines are buffered, pullup resistors should be used to hold the buffer inputs at a known level when the emulator cable is not connected.

There are two benefits to having the target system generate the test clock:

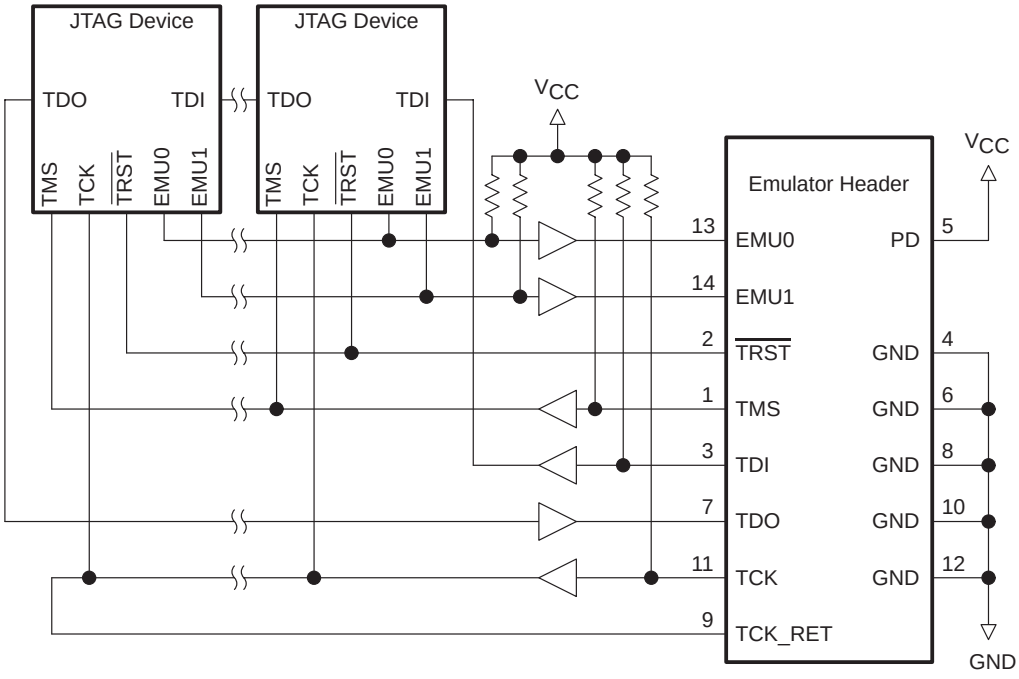
- The emulator provides only a single 10.368-MHz test clock. If you allow the target system to generate your test clock, you can set the frequency to match your system requirements.
- In some cases, you may have other devices in your system that require a test clock when the emulator is not connected. The system test clock also serves this purpose.

7.3 Configuring Multiple Processors

Figure 5 shows a typical daisy-chained multiprocessor configuration, which meets the minimum requirements of the IEEE 1149.1 specification. The emulation signals in this example are buffered to isolate the processors from the emulator and provide adequate signal drive for the target system. One of the benefits of this type of interface is that you can generally slow down the test clock to eliminate timing problems. You should follow these guidelines for multiprocessor support:

- The processor TMS, TDI, TDO, and TCK signals should be buffered through the same physical package for better control of timing skew.
- The input buffers for TMS, TDI, and TCK should have pullup resistors connected to V_{CC} to hold these signals at a known value when the emulator is not connected. Refer to the device datasheet for the recommended resistor value.
- Buffering EMU0 and EMU1 is optional but highly recommended to provide isolation. These are not critical signals and do not have to be buffered through the same physical package as TMS, TCK, TDI, and TDO. Unbuffered and buffered signals are shown in section 7.1.

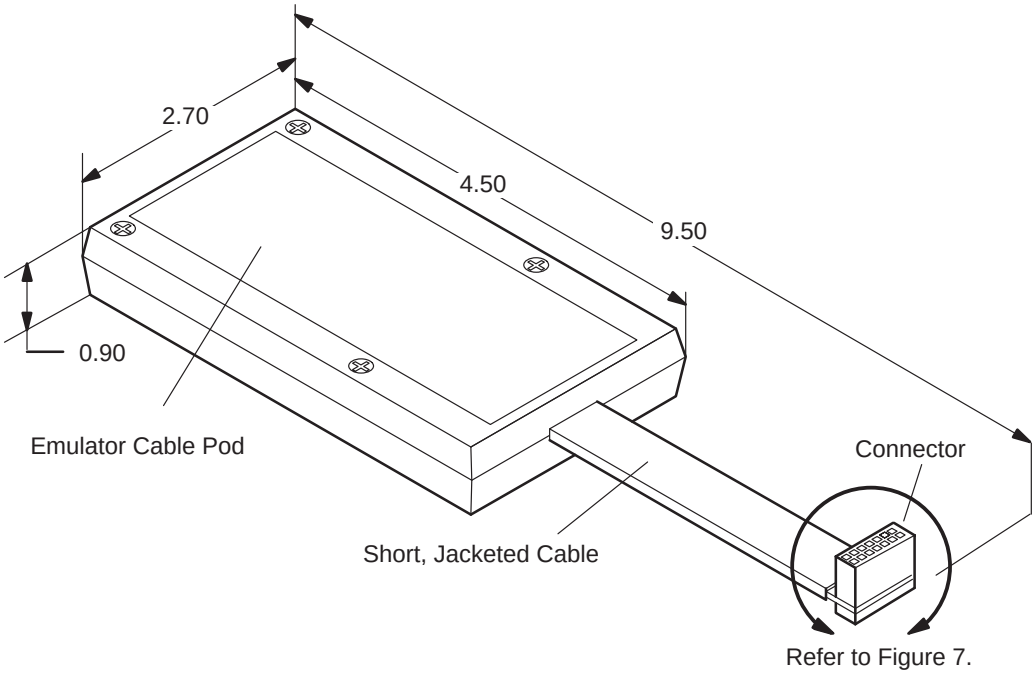
Figure 5. Multiprocessor Connections



8 Mechanical Dimensions for the 14-Pin Emulator Connector

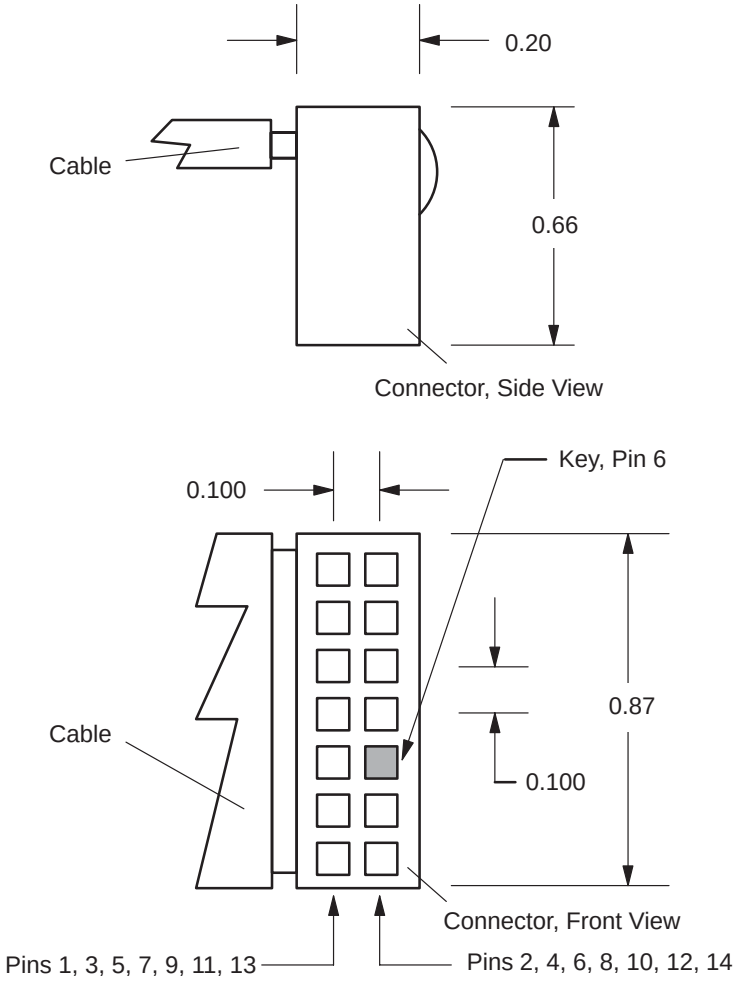
The JTAG emulator target cable consists of a 3-foot section of jacketed cable, an active cable pod, and a short section of jacketed cable that connects to the target system. The overall cable length is approximately 3 feet 10 inches. Figure 6 and Figure 7 show the mechanical dimensions for the target cable pod and short cable. Note that the pin-to-pin spacing on the connector is 0.100 inches in both the X and Y planes. The cable pod box is nonconductive plastic with four recessed metal screws.

Figure 6. Pod/Connector Dimensions



Note: All dimensions are in inches and are nominal dimensions, unless otherwise specified.

Figure 7. 14-Pin Connector Dimensions



Note: All dimensions are in inches and are nominal dimensions, unless otherwise specified.

9 Emulation Design Considerations

This section describes the use and application of the scan path linker (SPL), which can simultaneously add all four secondary JTAG scan paths to the main scan path. It also describes the use of the emulation pins and the configuration of multiple processors.

9.1 Using Scan Path Linkers

You can use the TI ACT8997 scan path linker (SPL) to divide the JTAG emulation scan path into smaller, logically connected groups of 4 to 16 devices. As described in the *Advanced Logic and Bus Interface Logic Data Book* (literature number SCYD001), the SPL is compatible with the JTAG emulation scanning. The SPL is capable of adding any combination of its four secondary scan paths into the main scan path.

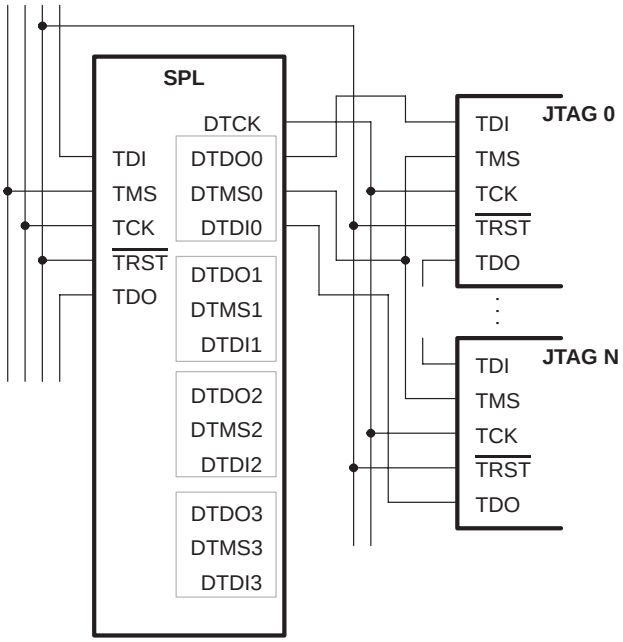
A system of multiple, secondary JTAG scan paths has better fault tolerance and isolation than a single scan path. Since an SPL has the capability of adding all secondary scan paths to the main scan path simultaneously, it can support global emulation operations, such as starting or stopping a selected group of processors.

TI emulators do not support the nesting of SPLs (for example, an SPL connected to the secondary scan path of another SPL). However, you can have multiple SPLs on the main scan path.

Although the ACT8999 scan path selector is similar to the SPL, it can add only one of its secondary scan paths at a time to the main JTAG scan path. Thus, global emulation operations are not assured with the scan path selector. For this reason, scan path selectors are not supported.

You can insert an SPL on a backplane so that you can add up to four device boards to the system without the jumper wiring required with nonbackplane devices. You connect an SPL to the main JTAG scan path in the same way you connect any other device. Figure 8 shows you how to connect a secondary scan path to an SPL.

Figure 8. Connecting a Secondary JTAG Scan Path to an SPL†



† Voltage translators should be used between the SPL (5V) and the C6000 (3V).

The $\overline{\text{TRST}}$ signal from the main scan path drives all devices, even those on the secondary scan paths of the SPL. The TCK signal on each target device on the secondary scan path of an SPL is driven by the SPL's DTCK signal. The TMS signal on each device on the secondary scan path is driven by the respective DTMS signals on the SPL.

DTDO on the SPL is connected to the TDI signal of the first device on the secondary scan path. DTDI on the SPL is connected to the TDO signal of the last device in the secondary scan path. Within each secondary scan path, the TDI signal of a device is connected to the TDO signal of the device before it. If the SPL is on a backplane, its secondary JTAG scan paths are on add-on boards; if signal degradation is a problem, you may need to buffer both the $\overline{\text{TRST}}$ and DTCK signals. Although less likely, you may also need to buffer the DTMS*n* signals for the same reasons.

9.2 Emulation Timing Calculations for SPL

The following examples help you to calculate the emulation timings in the SPL secondary scan path of your system. For actual target timing parameters, see the appropriate device data sheets.

Assumptions:

$t_{su(TTMS)}$	Target TMS/TDI setup to TCK high	10 ns
$t_{d(TTDO)}$	Target TDO delay from TCK low	15 ns
$t_{d(bufmax)}$	Target buffer delay, maximum	10 ns
$t_{d(bufmin)}$	Target buffer delay, minimum	1 ns
$t_{(bufskew)}$	Target buffer skew between two devices in the same package: $[t_{d(bufmax)} - t_{d(bufmin)}] \times 0.15$	1.35 ns
$t_{(TCKfactor)}$	Assume a 40/60 duty cycle clock	0.4 (40%)

Given in the SPL data sheet:

$t_{d(DTMSmax)}$	SPL DTMS/DTDO delay from TCK low, maximum	31 ns
$t_{su(DTDLmin)}$	DTDI setup time to SPL TCK high, minimum	7 ns
$t_{d(DTCKHmin)}$	SPL DTCK delay from TCK high, minimum	2 ns
$t_{d(DTCKLmax)}$	SPL DTCK delay from TCK low, maximum	16 ns

There are two key timing paths to consider in the emulation design:

- ❑ The TCK-to-DTMS/DTDO path, called $t_{pd(TCK-DTMS)}$, and
- ❑ The TCK-to-DTDI path, called $t_{pd(TCK-DTDI)}$.

Of the following two cases, the worst-case path delay is calculated to determine the maximum system test clock frequency.

Case 1: Single processor, direct connection, DTMS/DTDO timed from TCK low.

$$t_{pd\,(TCK–DTMS)} = \frac{\left[t_{d\,(DTMSmax)} + t_{d\,(DTCKHmin)} + t_{su\,(TTMS)}\right]}{t_{(TCKfactor)}}$$

$$= \frac{[31ns + 2ns + 10ns]}{0.4}$$

$$= 107.5ns\ (9.3\ MHz)$$

$$t_{pd\,(TCK–DTDI)} = \frac{\left[t_{d\,(TTDO)} + t_{d\,(DTCKLmax)} + t_{su\,(DTDLmin)}\right]}{t_{(TCKfactor)}}$$

$$= \frac{[15ns + 16ns + 7ns]}{0.4}$$

$$= 9.5ns\ (10.5\ MHz)$$

In this case, the TCK-to-DTMS/DTDL path is the limiting factor.

Case 2: Single/multiprocessor, DTMS/DTDO/TCK buffered input, DTDI buffered output, DTMS/DTDO timed from TCK low.

$$t_{pd\,(TCK–TDMS)} = \frac{\left[t_{d\,(DTMSmax)} + t_{(DTCKHmin)} + t_{su\,(TTMS)} + t_{(bufskew)}\right]}{t_{(TCKfactor)}}$$

$$= \frac{[31ns + 2ns + 10ns + 1.35ns]}{0.4}$$

$$= 110.9ns\ (9.0\ MHz)$$

$$t_{pd\,(TCK–DTDI)} = \frac{\left[t_{d\,(TTDO)} + t_{d\,(DTCKLmax)} + t_{su\,(DTDLmin)} + t_{d\,(bufskew)}\right.}{t_{(TCKfactor)}}$$

$$= \frac{[15ns + 15ns + 7ns + 10ns]}{0.4}$$

$$= 120ns\ (8.3\ MHz)$$

In this case, the TCK-to-DTDI path is the limiting factor.

9.3 Using Emulation Pins

The EMU0/1 pins of TI devices are bidirectional, three-state output pins. When in an inactive state, these pins are at high impedance. When the pins are active, they function in one of the two following output modes:

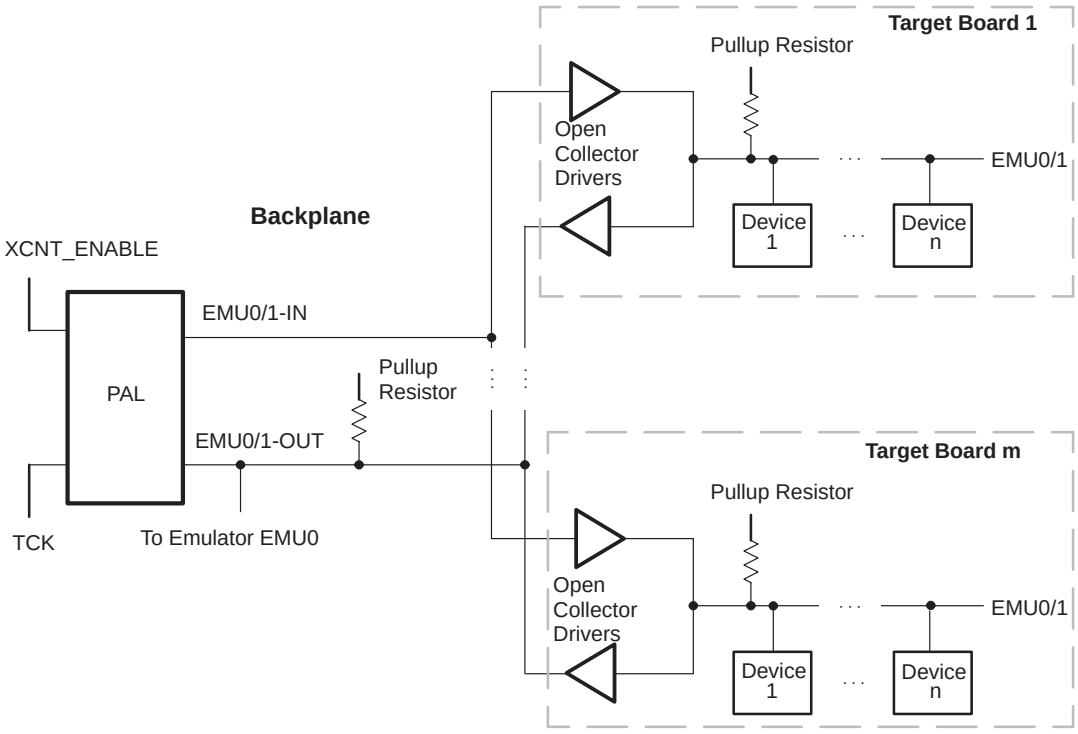
- **Signal Event**
  The EMU0/1 pins can be configured via software to signal internal events. In this mode, driving one of these pins low can cause devices to signal such events. To enable this operation, the EMU0/1 pins function as open-collector sources. External devices such as logic analyzers can also be connected to the EMU0/1 signals in this manner. If such an external source is used, it must also be connected via an open-collector source.

- **External Count**
  The EMU0/1 pins can be configured via software as totem-pole outputs for driving an external counter. If the output of more than one device is configured for totem-pole operation, then these devices can be damaged. The emulation software detects and prevents this condition. However, the emulation software has no control over external sources on the EMU0/1 signal. Therefore, all external sources must be inactive when any device is in the external count mode.

TI devices can be configured by software to halt processing if their EMU0/1 pins are driven low. This feature, in combination with the use of the signal event output mode, allows one TI device to halt all other TI devices on a given event for system-level debugging.

If you route the EMU0/1 signals between boards, they require special handling because these signals are more complex than normal emulation signals. Figure 9 shows an example configuration that allows any processor in the system to stop any other processor in the system. Do not tie the EMU0/1 pins of more than 16 processors together in a single group without using buffers. Buffers provide the crisp signals that are required during a RUNB (run benchmark) debugger command or when the external analysis counter feature is used.

Figure 9. EMU0/1 Configuration



Notes: 1) The low time on EMUx-IN should be at least one TCK cycle and less than 10 μs. Software will set the EMUx-OUT pin to a high state.

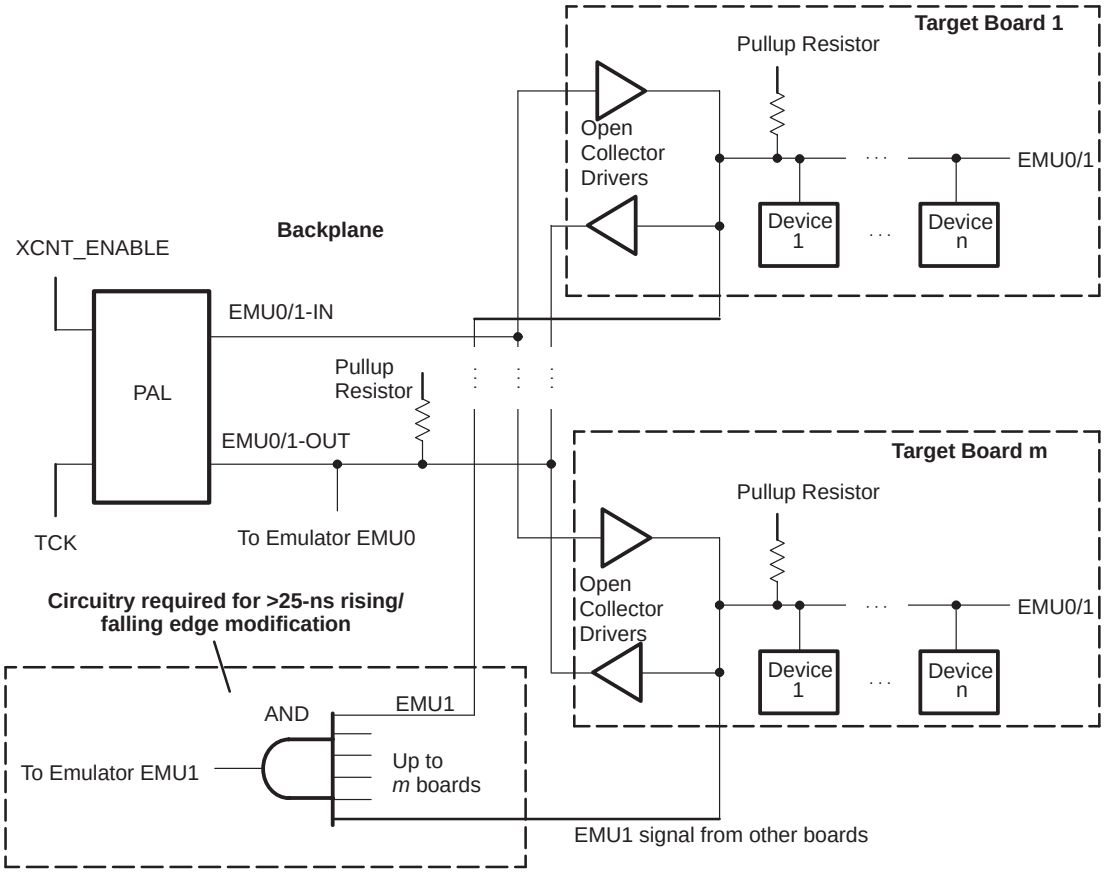
2) To enable the open-collector driver and pullup resistor on EMU1 to provide rising/falling edges of less than 25 ns, the modification shown in this figure is suggested. Rising edges slower than 25 ns can cause the emulator to detect false edges during the RUNB command or when the external counter selected from the debugger analysis menu is used.

These seven important points apply to the circuitry shown in Figure 9 and Figure 10 , and the timing shown in Figure 11:

- ❏ Open-collector drivers isolate each board. The EMU0/1 pins are tied together on each board.
- ❏ At the board edge, the EMU0/1 signals are split to provide IN/OUT. This is required to prevent the open-collector drivers from acting as a latch that can be set only once.
- ❏ The EMU0/1 signals are bused down the backplane. Pullup resistors are installed as required.

- The bused EMU0/1 signals go into a PAL® device, whose function is to generate a low pulse on the EMU0/1-IN signal when a low level is detected on the EMU0/1-OUT signal. This pulse must be longer than one TCK period to affect the devices, but less than 10 μs to avoid possible conflicts or retriggering, once the emulation software clears the device's pins.
- During a RUNB debugger command or other external analysis count, the EMU0/1 pins on the target device become totem-pole outputs. The EMU1 pin is a ripple carry-out of the internal counter. EMU0 becomes a *processor-halted* signal. During a RUNB or other external analysis count, the EMU0/1-IN signal to all boards must remain in the high (disabled) state. You must provide some type of external input (XCNT_ENABLE) to the PAL to disable the PAL from driving EMU0/1-IN to a low state.
- If sources other than TI processors (such as logic analyzers) are used to drive EMU0/1, their signal lines must be isolated by open-collector drivers and be inactive during RUNB and other external analysis counts.
- You must connect the EMU0/1-OUT signals to the emulation header or directly to a test bus controller.

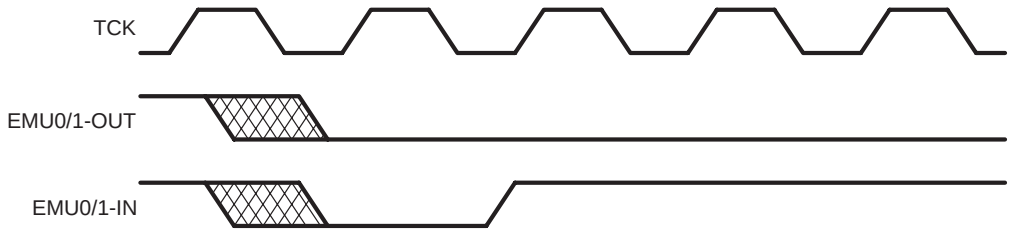
Figure 10. EMU0/1 Configuration With Additional AND Gate to Meet Timing Requirements



Notes: 1) The low time on EMUx−IN should be at least one TCK cycle and less than 10 μs. Software will set the EMUx−OUT pin to a high state.

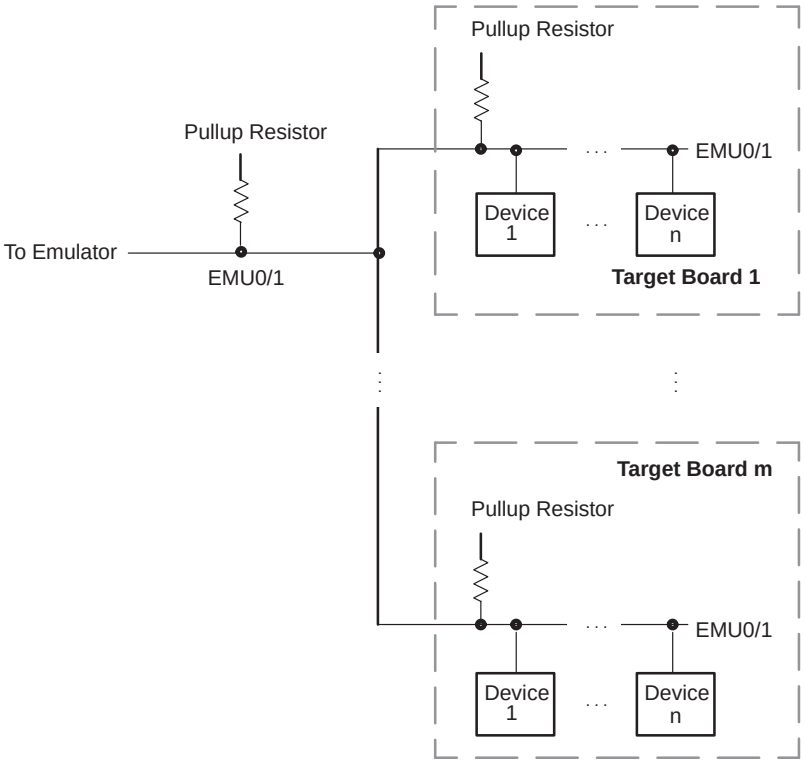
2) To enable the open-collector driver and pullup resistor on EMU1 to provide rising/falling edges of less than 25 ns, the modification shown in this figure is suggested. Rising edges slower than 25 ns can cause the emulator to detect false edges during the RUNB command or when the external counter selected from the debugger analysis menu is used.

Figure 11. Suggested Timings for the EMU0 and EMU1 Signals



If having devices on one target board stopped by devices on another target board via the EMU0/1 signals is not important, then the circuit in Figure 12 can be used. In this configuration, the global-stop capability is lost. It is important not to overload EMU0/1 with more than 16 devices.

Figure 12. EMU0/1 Configuration Without Global Stop

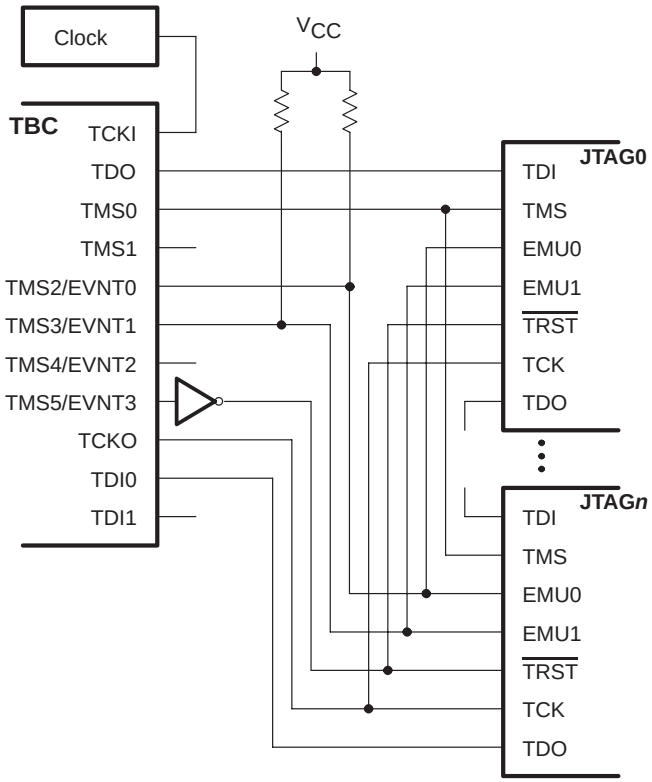


Note: The open-collector driver and pullup resistor on EMU1 must be able to provide rising/falling edges of less than 25 ns. Rising edges slower than 25 ns can cause the emulator to detect false edges during the RUNB command or when the external counter selected from the debugger analysis menu is used. If this condition cannot be met, then the EMU0/1 signals from the individual boards should be ANDed together (as shown in Figure 10) to produce an EMU0/1 signal for the emulator.

9.4 Performing Diagnostic Applications

For systems that require built-in diagnostics, it is possible to connect the emulation scan path directly to a TI ACT8990 test bus controller (TBC) instead of the emulation header. The TBC is described in the Texas Instruments *Advanced Logic and Bus Interface Logic Data Book* (literature number SCYD001). Figure 13 shows the scan path connections of *n* devices to the TBC.

Figure 13. TBC Emulation Connections for n *JTAG Scan Paths*†



† Voltage translators should be used between the TBC (5V) and the C6000 DSP (3V).

In the system design shown in Figure 13, the TBC emulation signals TCKI, TDO, TMS0, TMS2/EVNT0, TMS3/EVNT1, TMS5/EVNT3, TCKO, and TDI0 are used, and TMS1, TMS4/EVNT2, and TDI1 are not connected. The target devices' EMU0 and EMU1 signals are connected to V_{CC} through pullup resistors and tied to the TBC's TMS2/EVNT0 and TMS3/EVNT1 pins, respectively. The TBC's TCKI pin is connected to a clock generator. The TCK signal for the main JTAG scan path is driven by the TBC's TCKO pin.

On the TBC, the TMS0 pin drives the TMS pins on each device on the main JTAG scan path. TDO on the TBC connects to TDI on the first device on the main JTAG scan path. TDI0 on the TBC is connected to the TDO signal of the last device on the main JTAG scan path. Within the main JTAG scan path, the TDI signal of a device is connected to the TDO signal of the device before it. $\overline{\text{TRST}}$ for the devices can be generated either by inverting the TBC's TMS5/EVNT3 signal for software control or by logic on the board itself.

Index

B

block diagram
  connecting a secondary JTAG scan path to an SPL 21
  EMU0/1 configuration 25
  EMU0/1 configuration with additional AND gate to meet timing requirements 27
  EMU0/1 configuration without global stop 28
  JTAG emulator cable pod interface 10
  multiprocessor connections 17
  TBC emulation connections for n JTAG scan paths 29
buffering 14
bus protocol 9

C

cable pod 10
  dimensions 18
configuring multiple processors 17
connector
  14-pin header 7
  14-pin, dimensions 19
  dimensions, mechanical 18

D

diagnostic applications 28

E

EMU0/1
  configuration 25
    with additional AND gate to meet timing requirements 27
    without global stop 28
  emulation pins 24
  rising edge modification 27
emulation
  JTAG cable 7
  timing calculations 12, 22
emulation design considerations 20
  timing calculations for scan path linkers 22
  using scan path linkers 20
emulator
  connection to target system, JTAG mechanical dimensions 18
  emulation pins 24
  signal buffering 14
  target cable, header design 7

H

header, 14-pin 7
  dimensions 7
  header signals 7
  JTAG 7

I

IEEE 1149.1 specification, bus slave device rules 9

J

JTAG emulator
buffered signals 15
no signal buffering 14
pod interface 10
signal timings 11
timing parameters 11

N

notational conventions 3

P

protocol, bus 9

R

related documentation from Texas Instruments 3
run/stop operation 14

S

scan path linkers
secondary JTAG scan chain to an SPL 21
suggested timings 27
scan paths, TBC emulation connections for JTAG scan paths 29
signal descriptions, 14-pin header 8
signals
buffering for emulator connections 14
description, 14-pin header 8
timing 11

T

TBC emulation connections for n JTAG scan paths 29
test clock 16
timing calculations 12, 22
trademarks 4