MIPI Alliance Recommendation for Debug and Trace Connectors

Version 1.10.00 – 16 March 2011

MIPI Board Approved for Public Distribution 19-Apr-2011

Further technical changes to this document are expected as work continues in the Debug Working Group

NOTICE OF DISCLAIMER

This is an informative document, not a MIPI Specification.

Various rights and obligations that apply solely to MIPI Specifications (as defined in the MIPI Membership Agreement and MIPI Bylaws) including, but not limited to, patent license rights and obligations, do not apply to this document.

The material contained herein is not a license, either expressly or impliedly, to any IPR owned or controlled by any of the authors or developers of this material or MIPI®. The material contained herein is provided on an "AS IS" basis and to the maximum extent permitted by applicable law, this material is provided AS IS AND WITH ALL FAULTS, and the authors and developers of this material and MIPI hereby disclaim all other warranties and conditions, either express, implied or statutory, including, but not limited to, any (if any) implied warranties, duties or conditions of merchantability, of fitness for a particular purpose, of accuracy or completeness of responses, of results, of workmanlike effort, of lack of viruses, and of lack of negligence.

All materials contained herein are protected by copyright laws, and may not be reproduced, republished, distributed, transmitted, displayed, broadcast or otherwise exploited in any manner without the express prior written permission of MIPI Alliance. MIPI, MIPI Alliance and the dotted rainbow arch and all related trademarks, tradenames, and other intellectual property are the exclusive property of MIPI Alliance and cannot be used without its express prior written permission.

ALSO, THERE IS NO WARRANTY OF CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, CORRESPONDENCE TO DESCRIPTION OR NON-INFRINGEMENT WITH REGARD TO THIS MATERIAL OR THE CONTENTS OF THIS DOCUMENT. IN NO EVENT WILL ANY AUTHOR OR DEVELOPER OF THIS MATERIAL OR THE CONTENTS OF THIS DOCUMENT OR MIPI BE LIABLE TO ANY OTHER PARTY FOR THE COST OF PROCURING SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA, OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, OR SPECIAL DAMAGES WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THIS OR ANY OTHER AGREEMENT, SPECIFICATION OR DOCUMENT RELATING TO THIS MATERIAL, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Without limiting the generality of this Disclaimer stated above, the user of the contents of this Document is further notified that MIPI: (a) does not evaluate, test or verify the accuracy, soundness or credibility of the contents of this Document; (b) does not monitor or enforce compliance with the contents of this Document; and (c) does not certify, test, or in any manner investigate products or services or any claims of compliance with the contents of this Document. The use or implementation of the contents of this Document may involve or require the use of intellectual property rights ("IPR") including (but not limited to) patents, patent applications, or copyrights owned by one or more parties, whether or not Members of MIPI. MIPI does not make any search or investigation for IPR, nor does MIPI require or request the disclosure of any IPR or claims of IPR as respects the contents of this Document or otherwise.

Questions pertaining to this document, or the terms or conditions of its provision, should be addressed to:

MIPI Alliance, Inc.
c/o IEEE-ISTO
445 Hoes Lane
Piscataway, NJ 08854
Attn: Board Secretary

Contents

Figures

Tables

Release History

Date	Release	Description
2007-06-07	1.00.00	Initial board adopted release.
2011-06-29	1.10.00	Changed name to “MIPI Alliance Recommendation for Debug and Trace Connectors” from “MIPI Alliance Recommendation for Debug: Debug and Trace Connectors”. Added keyed connector part numbers for the basic debug connector. Added recommendation for overlaying multiple interfaces on trace channels.
2011-08-30	1.10.00	Board approved release.

MIPI Alliance Recommendation for TDC

1 Overview

1.1 Scope

This document provides a recommendation for board level mating connectors that supports new debug communication interfaces and protocols (like IEEE 1149.7) while maintaining backward compatibility with existing legacy technology. This document also provides recommendations for board level mating connectors that support trace via a high-speed parallel interface. In addition, the trace connectors also support legacy debug via an IEEE 1149.1 interface and the newer IEEE 1149.7 interface. See [IEEE01] and [IEEE02] for more details.

This document specifies recommendations for different mating connectors including:

- Low cost connector configurations that supports basic debug and minimal trace
- A legacy connector that supports basic debug and a single trace channel
- A new high performance connector that supports basic debug and up to 4 independent trace channels

1.1.1 Document Is

This document recommends a physical mating connector used to connect the MIPI Debug Interface on a target system (TS) via a debug and test controller (DTC). It includes pin mappings (based on that connector) that support the following debug configurations:

- IEEE 1149.7 JTAG in 2-pin mode (Narrow JTAG) debug control
- IEEE 1149.7 JTAG in 4/5-pin mode or legacy JTAG (IEEE Std 1149.1-2001) (Legacy JTAG) debug control
- Narrow JTAG debug control with a single 4-data bit PTI
- Legacy JTAG debug control with a single 4-data bit PTI

This document also recommends two physical mating connectors used to connect a target system (TS) to a debug and test controller (DTC) that supports trace capture via a MIPI Parallel Trace Interface (PTI) [MIPI01]. The connector recommendations fall into two classes:

- A legacy connector (38 signal MICTOR) that provides a migration path from previous trace systems to MIPI. This document recommends pin mappings for the legacy (MICTOR) connector that support the following debug functions:
    - A maximum of 1 channel of high speed parallel trace
        - A maximum of 16 PTI TRC_DATA signals
        - A maximum of 1 PTI TRC_CLK signal
    - Standard debug control
        - via an IEEE 1149.1 interface (with RTCK) or
        - via an IEEE 1149.7 interface

- A new MIPI High Speed Parallel Trace (HSPT) connector (60 signal Samtec QSH/QTH) that provides better signal integrity and more desirable physical performance over the legacy MICTOR. This document recommends pin mappings for the new HSPT connector (QSH/QTH) that support the following debug functions:
    - A maximum of 4 channels of high speed parallel trace
        - A maximum of 40 PTI TRC_DATA signals
        - A maximum of 4 PTI TRC_CLK signals
    - Standard debug control
        - via an IEEE 1149.1 interface (with RTCK) or
        - via an IEEE 1149.7 interface

1.1.2 Document Is Not

This document is not the recommended pin mapping for any debug function or interface not enumerated in Section 1.1.1.

1.1.3 Related Specifications

Specifications for the debug functions supported by this recommendation are enumerated in Section 3.

1.2 Purpose

As the connector was not part of the original IEEE 1149.1 JTAG standard, a large number of different JTAG connectors have emerged. The recommendation of standard connectors is an attempt to avoid this for the future. Different applications and use cases have different requirements for the connector. The definition is flexible enough to handle all these requirements. This avoids the need for target specific adapters – with an impact on communication speed and reliability.

Many debug tools vendors support target systems with high-speed trace interfaces. These tools utilize a number of different mating connectors. This document recommends two connectors for supporting high speed trace and generic debug. The goal is to have this recommendation define a “de facto” industry standard for the trace connection and thus lessen the burden on target system and tools developers that need to support a large number of different mating connections.

The first trace connector is the 38-pin MICTOR connector that has previously been used to support trace connections on many systems. A signal to pin mapping for this connector is provided that supports the run control and trace interfaces specified under MIPI. This recommendation provides a migration path for legacy equipment to support MIPI trace and debug interfaces.

The second trace connector is a 60-pin QSH/QTH connector. This connector provides for more and/or wider trace interfaces, has better electrical characteristics and has a wider range of PCB mounting options than the MICTOR and is the recommended connector to new DTCs and target systems.

2 Terminology

2.1 Definitions

Narrow JTAG: IEEE 1149.7 JTAG in 2-pin mode (T4/T5 class TAP.7 of IEEE 1149.7).

Wide JTAG: IEEE 1149.7 JTAG in 4/5-pin mode (T0-T3 class TAP.7 of IEEE 1149.7) or legacy JTAG (IEEE Std 1149.1-2001).

2.2 Abbreviations

e.g.	For example
p/n	Part Number

2.3 Acronyms

DTC	Debug and Trace Controller
DTS	Debug and Test System
HSPT	High Speed Parallel Trace
ISTO	Industry Standards and Technology Organization
JTAG	Joint Test Access Group
MIPI	Mobile Industry Processor Interface
PCB	Printed Circuit Board
PTI	MIPI Parallel Trace Interface
TS	Target System
T&D	Test and Debug

3 References

[ARM01] ARM IHI 0014N, *Embedded Trace Macrocell™ ETMv1.0 to ETMv3.4: Architecture Specification*, ARM Limited, 8 February 2006.

[IEEE01] IEEE 1149.1™-2001, *IEEE Standard Test Access Port and Boundary-Scan Architecture*, ISBN 0-7381-2944-5, Institute of Electrical and Electronics Engineers, 14 June 2001.

[IEEE02] IEEE 1149.7 ™-2009, *IEEE Standard for Reduced-Pin and Enhanced-Functionality Test Access Port and Boundary-Scan Architecture*, Institute of Electrical and Electronics Engineers, 9 December 2009.

[ISTO01] IEEE-ISTO 5001™-2003, *The Nexus 5001 Forum™ Standard for a Global Embedded Processor Debug Interface*, Version 2.0, IEEE-Industry Standards and Technology Organization, 23 December 2003.

[MIPI01] *MIPI Alliance Standard for Test & Debug – Parallel Trace Interface*, version 1.00, MIPI Alliance, Inc., 19 December 2006.

4 Signal Descriptions

Table 1 contains a summary of all the signal names used to define the pin mappings for all MIPI recommended connectors.

249 **Table 1 Signal Descriptions**

Signal(s)	Type and Direction at Board Header	Reference Voltage	Description
VREF_DEBUG	Analog output	NA	Debug port reference voltage. The pin is used to adjust the DTC input and output buffers to the target voltage. Connected with a resistor to target VCC for I/O buffers driving the debug communications interface and A.C. grounded (Option 1 in Figure 1) with a capacitor or directly connected to target VCC (Option 2 in Figure 1). In the case of Option 1 the resistor shall not have a value greater than 1K Ohms. The capacitor is required for A.C. grounding the cable.
VREF_TRACE	Analog output	NA	Trace port reference voltage. Connected with a resistor to target VCC for I/O buffers driving the trace interface and current limited (Option 1 in Figure 2) with a resistor or directly connected to target VCC (Option 2 in Figure 2). In the case of Option 1 the resistor shall not have a value greater than 1K Ohms.
TCK	In or Out[2]	VREF_DEBUG	IEEE 1149.1 JTAG signals. nTRST is active low signal (sometimes referenced as TRST*).
TMS	Input	VREF_DEBUG	
TDI[1]	Input	VREF_DEBUG	
TDO[1]	Output	VREF_DEBUG	
nTRST	Input	VREF_DEBUG	
RTCK	Output	VREF_DEBUG	Return Test Clock.
TMSC	In/Out	VREF_DEBUG	IEEE 1149.7 bi-directional data signal
TCKC	In or Out	VREF_DEBUG	IEEE 1149.7 clock signal can be driven from the DTC or the TS
TRIGIN	Input	VREF_DEBUG	External debug halt request
TRIGOUT	Output	VREF_DEBUG	External debug halt acknowledge

Signal(s)	Type and Direction at Board Header	Reference Voltage	Description
nRESET	Open Drain	VREF_DEBUG	Target reset signal. Open drain active low signal. May be used bi-directional to drive or sense the target reset signal. Usually driven by DTS to reset target system. The TS is responsible for providing a pull up to VREF_DEBUG on this signal to establish a logic one. This pull-up may be outside chips or within chips. The resistor shall not have a value less than 1K Ohms. The capacitor may be present on the TS to provide the power up reset signal.
nTRST_PD	Input	VREF_DEBUG	Test Reset Pull-Down. Same function as nTRST, but with pull-down resistor on target. This is an active low signal (sometimes referenced as TRST_PD*).
TRC_CLK[N:0]	Output	VREF_TRACE	PTI Trace clock pins. If unused these should be connected to ground or driven to a low logic level.
TRC_DATA[N:0]	Output	VREF_TRACE	PTI Trace data pins. Any unused TRC_DATA pin should be tied to ground.
EXT	In/Out	VREF_DEBUG	Implementation defined debug application signals that can be used for non-specified functions.
TRC_EXT	In/Out	VREF_TRACE	Implementation defined trace sideband signal.
Target Presence Detect	Out	VREF_DEBUG	This signal can be used by the debug tools to indicate that an active target system is connected. See Appendix A for details. If this feature is not required this signal should be tied to ground.

Notes:

1. *When implemented in an IEEE 1149.7 environment, the signals TDI, TDO, and RTCK may be multiplexed with a generic debug application (they function as an EXT).*
2. *The IEEE 1149.7 signal TCK may be sourced by the DTC or the Target System.*

4.1 Reference Voltage Pins

Figure 1 and Figure 2 show the different options for configuring the reference voltages for the debug and trace interfaces on the connector.

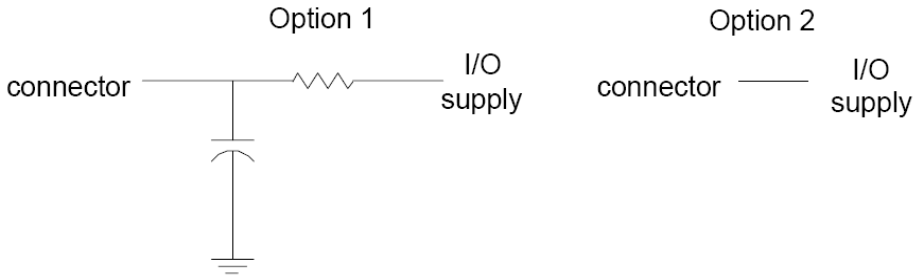


Figure 1 VREF_DEBUG Options

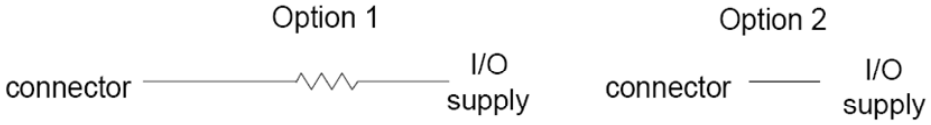


Figure 2 VREF_TRACE Options

4.2 nRESET

Figure 3 shows how the open drain signal nRESET should be handled on the PCB. Note that when nRESET is supported on the connector and target system the DTC should not be connected to the TS when the TS is powered. The “hot connects” may result in unintended resets of the target system.

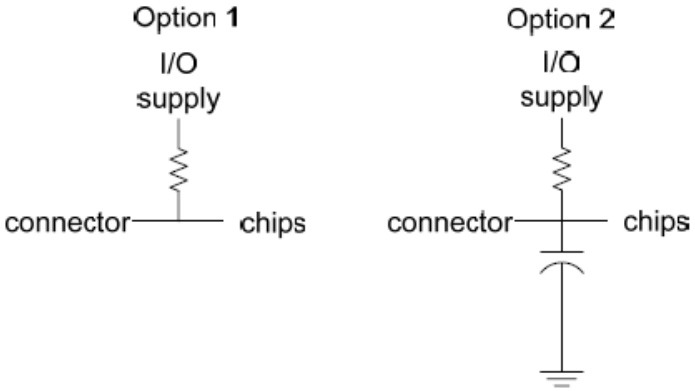


Figure 3 nRESET Signal Handling

4.3 nTRST_PD and nTRST

In target systems with multiple discrete devices, it is possible to have a mixture of devices that support only nTRST_PD or nTRST. The recommendation is to support such systems with a connector configuration and a DTC that have independent nTRST_PD and nTRST signals. The device level nTRST_PD and nTRST signals are then routed to the correct pins on the header. There may be scenarios where independent nTRST_PD and nTRST pins cannot be supported. The following list has some examples:

- 10-pin header required (only one debug reset signal is available)
- DTC does not have dual nTRST_PD/nTRST support

If required, it is possible to combine nTRST_PD and nTRST and present them as a single pin on the connector, but system designers should be aware of the functional limitations imposed by this merger. The IEEE 1149.7 documentation should be consulted, but the following list is a summary of the limitations:

- Combined connector pin functions as nTRST_PD
  - Devices with nTRST_PD input
    - No change in the behavior of the debug and test interface
  - Devices with nTRST input
    - Debug and test interface operates in a mode that is not compliant with IEEE 1149.1
      - The reset is asserted by default.
      - DTC must be able to drive nTRST_PD pin high to enable the debug interface
- Combined connector pin functions as nTRST
  - Devices with nTRST_PD input
    - Debug and test interface operates in a mode that is not compliant with IEEE 1149.7
      - The reset is NOT asserted by default
      - Debug and test logic is not immune to extraneous noise in the system
      - DTC must be able to pulse nTRST low to reset the interface
  - Devices with nTRST
    - No change in the behavior of the debug and test interface

Section 4.3.1 provides a recommendation for merging nTRST_PD and nTRST at the board level.

4.3.1 Merging nTRST_PD and nTRST at the Board Level

Since the nTRST signal and nTRST_PD signal have different default (internal pull-up/pull-down) values, they cannot be tied together. The different logic levels conflict with each other and will drive the resultant signal to an undefined level. Consequently the TRST* and nTRST_PD signals must be kept separate until they reach a driver that overdrives the internal pull-up/pull-down values creating the correct logic level.

Figure 4 and Figure 5 show examples of how board level buffers and pull-ups can be used to merge the nTRST_PD and nTRST functions to a single pin on the connector.

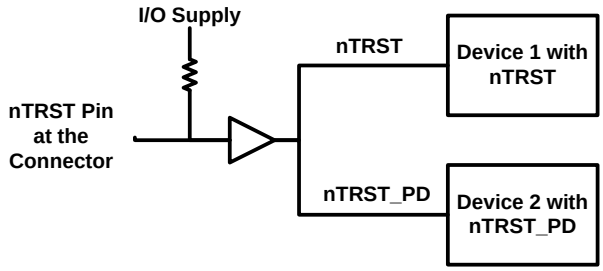


Figure 4 nTRST and nTRST_PD Merged as nTRST

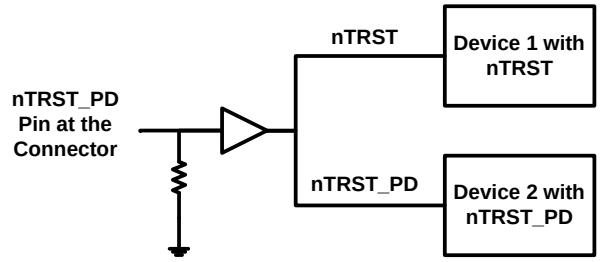


Figure 5 nTRST and nTRST_PD Merged as nTRST_PD

4.4 Exclusive Two-Pin IEEE 1149.7

It is possible for a MIPI debug connector to be connected to a system that initializes (boots) with a two-pin IEEE 1149.7 interface. If this type of target system does not provide an nTRST_PD input, then the nTRST_PD connector pin shall be connected to ground through a resistor.

5 Basic Debug Connector

A scalable 0.05 inch connector provides an inexpensive, small and robust target connection and is available in many variants (including lockable ones) from multiple connector vendors. The pin mapping allows scaling the connection to meet different requirements. This includes very small footprint connections (down to 10 pins), legacy JTAG support (including vendor specific pins) and system level trace support (STM). A key pin (pin removed from target connector) enforces polarization. This allows more flexibility in the connector type and size. It also enables the connection of a larger DTS connector to a smaller TS connection.

5.1 Connector Type

The 0.05 inch connector is available from many sources. It is usually offered as standard dual row header 1.27mm x 1.27mm with 0.4mm square pins. Connectors with special locking or polarization may not be compatible with all debug controllers. Figure 6 contains illustrations of the PCB and cable end connectors.

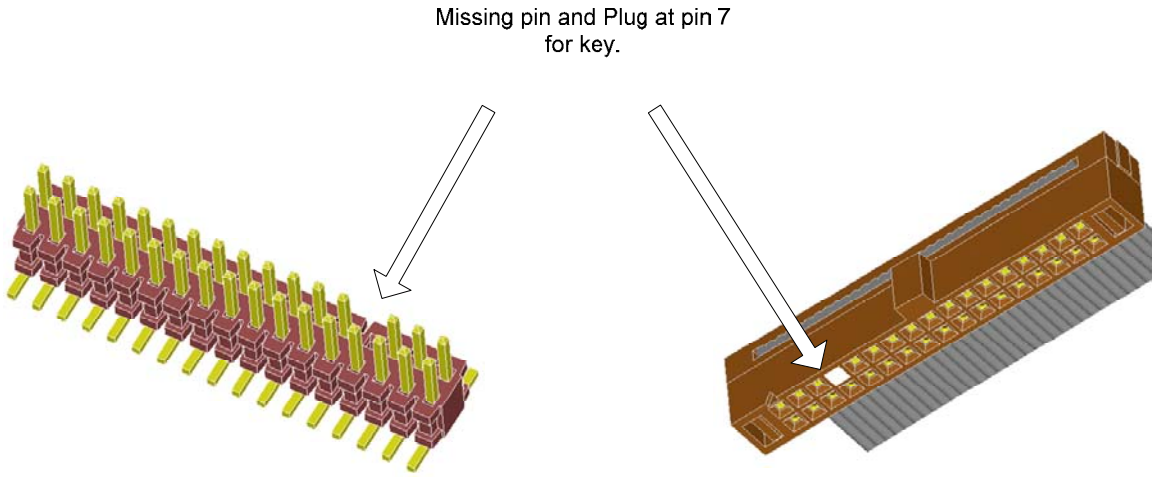


Figure 6 Basic Debug Connector (34-pin)

5.1.1 Connector Sources

The following Samtec identifiers are examples only. Other vendors provide compatible hardware.

- Samtec FTSH (SMT Mount)
  - Horizontal or vertical
  - Without shroud, with shroud or with ejector shroud
  - With or without keying
- Samtec FTSH (Through Hole)
  - Straight or right Angle
  - Without shroud, with shroud or with ejector shroud
  - With or without keying

- Samtec EHF (Ejector Header)
  - Straight or surface mount or right angle

5.1.1.1 PCB Connector

Table 2 shows the Samtec part number templates for the recommended PCB mounted connectors on the target system. These are templates only and do not address the all of the compatible board connectors available.

Table 2 PCB Mounted Connector Part Number Templates

Samtec Part Numbers[1]	Description
FTSH-105-01-<Plating Option>-D<Row Option>-007	10-way with pin 7 omitted
FTSH-110-01-<Plating Option>-D<Row Option>-007	20-way with pin 7 omitted
FTSH-117-01-<Plating Option>-D<Row Option>-007	34-way with pin 7 omitted
FTSH-105-01-<Plating Option>-D<Row Option>-K-007	10-way with keyed shroud, pin 7 omitted
FTSH-110-01-<Plating Option>-D<Row Option>-K-007	20-way with keyed shroud, pin 7 omitted
FTSH-117-01-<Plating Option>-D<Row Option>-K-007	34-way with keyed shroud, pin 7 omitted
EHF-105-01-<Plating Option>-D-<Locking Clip>-007	10-way with ejector head, pin 7 omitted
EHF-110-01-<Plating Option>-D-<Locking Clip>-007	20-way with ejector head, pin 7 omitted
EHF-117-01-<Plating Option>-D-<Locking Clip>-007	34-way with ejector head, pin 7 omitted

Notes:

1. These part numbers are generic templates. Specific part numbers are generated by the vendor when customers order specific variants of the connector. Contact Samtec for more details.

5.1.1.2 Cable Connector

Table 3 shows the Samtec part number templates for the recommended cable end/adaptor connectors on the DTC. These are templates only and do not address the all of the compatible connectors available.

Table 3 Cable Connector Part Number Templates

Samtec Part Numbers[1]	Description
SFMC-105-<Lead Style Option>-S-D-PK-07-N	10-way surface mount with pin 7 filled
SFMC-110-<Lead Style Option>-S-D-PK-07-N	20-way surface mount with pin 7 filled
SFMC-117-<Lead Style Option>-S-D-PK-07-N	34-way surface mount with pin 7 filled
FFSD-05-<Cable Head Option>-<Length>-01-N-PK-07-N	10-way cable with key and pin 7 filled
FFSD-10-<Cable Head Option>-<Length>-01-N-PK-07-N	20-way cable with key and pin 7 filled
FFSD-17-<Cable Head Option>-<Length>-01-N-PK-07-N	34-way cable with key and pin 7 filled

Notes:

1. These part numbers are generic templates. Specific part numbers are generated by the vendor when customers order specific variants of the connector. Contact Samtec for more details.

5.2 Pin Mapping

The pin mapping allows different sizes for extending JTAG functionality or trace features. If legacy JTAG and trace are needed then trace pins are moved from the standard position (pins 12 to 20) to the second variant (pins 22 to 30) with similar pin mapping. This allows minimum size target connectors for all common use cases. Connector sizes of 10, 20 and 34 pins are the recommended connector/cable widths. Unused pins may be used for vendor specific signals.

5.2.1 Rules for Pin Mapping

Designers must review the system requirements in order to determine the connector configuration. The following requirements need to be considered, and then Table 4 should be consulted to determine which header width and pin mapping is appropriate.

- Narrow or Standard JTAG (or both)
- Trace Support
- nTRST_PD, nTRST and nRESET (none, one, two or all three)

5.2.2 Recommended Pin Mappings

Table 4 shows the recommended pin mappings for common use cases.

366 **Table 4 MIPI Recommended Pin Mapping for Basic Debug**

All	Pin#	Pin#	10-pin JTAG + nRESET	10-pin JTAG + nTRST	10-pin JTAG + nTRST_PD	10-pin Narrow JTAG	20-pin JTAG	20-pin Narrow JTAG + Trace	34-pin JTAG + Trace
VREF_DEBUG[1]	1	2	TMS	TMS	TMS	TMSC	TMS	TMSC	TMS
GND	3	4	TCK	TCK	TCK	TCKC	TCK	TCKC	TCK
GND	5	6	TDO	TDO	TDO	EXT	TDO	EXT	TDO
KEY[2]	7	8	TDI	TDI	TDI	nTRST_PD	TDI	nTRST_PD	TDI
Target Presence Detect	9	10	nRESET	nTRST	nTRST_PD	nRESET	nRESET	nRESET	nRESET
GND[3]	11	12					RTCK	TRC_CLK	RTCK
GND[3]	13	14					nTRST_PD	TRC_DATA[0]	nTRST_PD
GND	15	16					nTRST	TRC_DATA[1]	nTRST
GND	17	18					TRIGIN (DBGRQ)	TRC_DATA[2]	TRIGIN (DBGRQ)
GND	19	20					TRIGOUT (DBGACK)	TRC_DATA[3]	TRIGOUT (DBGACK)
GND	21	22							TRC_CLK
GND	23	24							TRC_DATA[0]
GND	25	26							TRC_DATA[1]
GND	27	28							TRC_DATA[2]
GND	29	30							TRC_DATA[3]
GND	31	32							TRC_EXT
GND	33	34							VREF_TRACE

367 ***Notes:***

368 *1. A.C. grounded with capacitor or direct VCC connection.*

369 *2. Pin removed from target connector, grounded in cable.*

3. *ARM Ltd. allows the DTC to supply power to the TS via these pins. This feature is not supported in the MIPI connector.*

5.2.3 Physical Orientation of the Connector on the PCB

The physical orientation of the connector is important for proper board and DTC design. The following recommendations define how to place the Basic Debug Connector on the PCB with the target system.

- Place the even numbered pins of the Basic Debug Connector closest to the Target System(s) that are supported through the connector. See Figure 7 for an illustration. Note that this orientation is not the same as the HSPT Connector as described in Section 6.1.7.
- PCB designers should consult DTC vendor documentation to determine if there are any board clearance issues related to the DTC cable-end connector

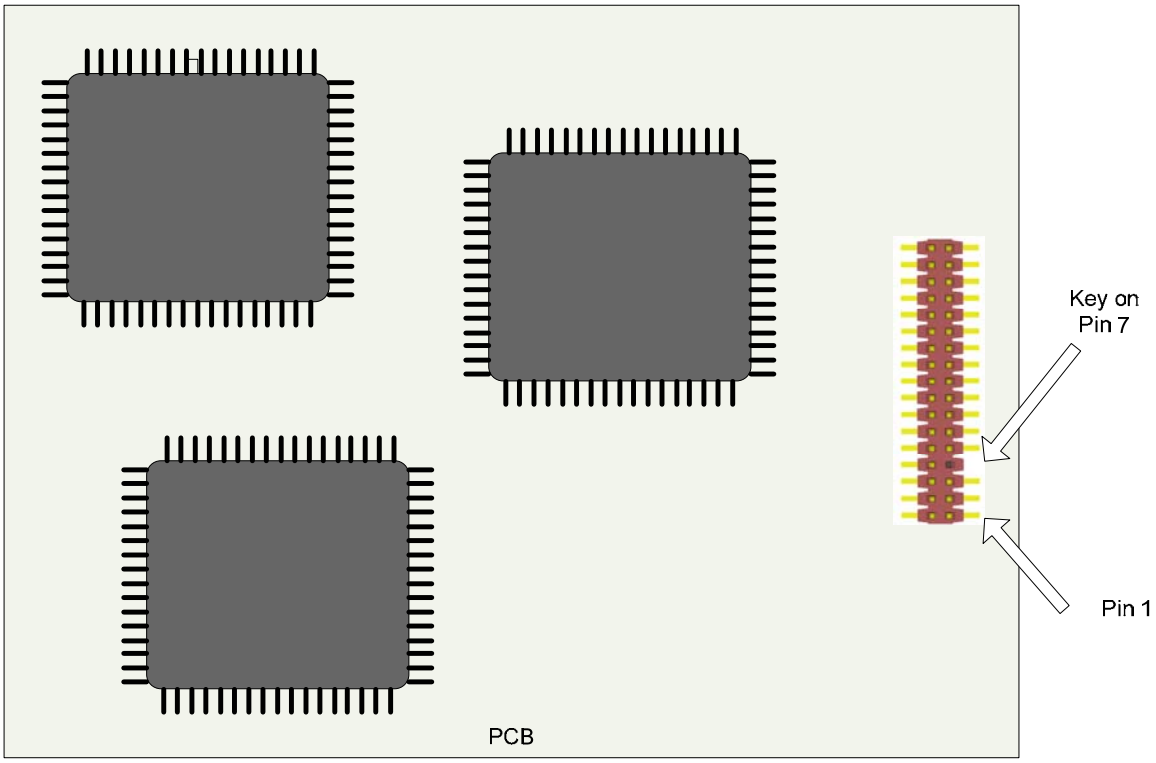


Figure 7 Basic Debug Orientation on the PCB (34-pin Example)

5.2.4 Cable-end Design Recommendations

DTC vendors may implement cable-end connectors with various physical orientations.

- Vertical - Cable to the DTC exits the connector housing perpendicular to PCB surface
- Right-angle – Cable to the DTC exits the connector housing parallel to the PCB surface

For right-angle cable-end connectors, the cable interface to the connector should overhang the odd numbered pins of the connector.

6 High Speed Parallel Trace Connectors

There are two HSPT connectors discussed in this document. This recommendation promotes a high performance connector with 60 data signal. This connector allows for a rich pin mapping that can support the multiple PTI channels that are present in today's complex systems. It also supports a wide variety of mounting and cable-end options.

Many older systems support high speed trace via the 38 pin MICTOR. The PCB mounting limitations ("through hole") and limited number of data signals make this connector less desirable. While a signal mapping is provided for the MICTOR, the goal is to have the QSH/QTH become the standard moving forward.

6.1 Recommended HSPT Connector

The recommended HSPT connector is the Samtec QSH/QTH family with two rows of 30 pins. All 60 pins can be dedicated to trace and debug signals. The 01 lead and 5mm stacking height are the recommended options. Other stacking height options are supported for the PCB and cable-end connections. Figure 8 contains illustrations of the PCB and cable end connectors (QSH and QTH).

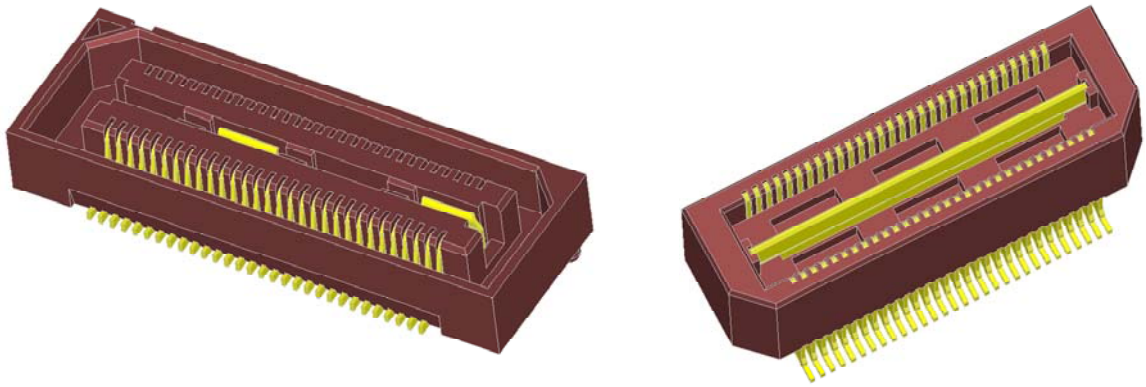


Figure 8 QSH and QTH Connectors

6.1.1 PCB Connector

Table 5 shows the recommended connectors for mounting on the PCB that supports the target system.

Table 5 PCB Mounted Connectors

Samtec Part Numbers[1]	Mount Angle	Mount Type
QSH-030-01-<Plating Option>-D-<Alignment Pin>-<Other Option>	Vertical	Surface
QSH-030-01-<Plating Option>-D-EM<Thickness>-<Other Option>	Edge	Surface

Notes:

1. *Consult Samtec for details on the part numbers.*

6.1.2 Cable Connector

Table 6 shows the recommended connectors for the DTC cable or adaptor boards for DTC connection.

412 **Table 6 Cable or DTC Adaptor Connectors**

Samtec Part Numbers[1]	Mount Angle	Mount Type
QTH-030-01-<Plating Option>-D-<Alignment Pin>-<Other Option>	Vertical	Surface
QTH-030-01-<Plating Option>-D-EM<Thickness>-<Other Option>	Edge	Surface

413 ***Notes:***

414 *1. Consult Samtec for details on the part numbers.*

415 6.1.3 Connector Options

416 Table 7 summarizes the other cable options that are supported under this recommendation.

417 **Table 7 Connector Options**

Option	Summary
Plating Option	-L = Gold -F = Gold Flashing
Alignment Pin	The catalog connectors come with alignment pins (-A) but these connectors can be special ordered without them or with locking clips
Other Option	These specify PCB manufacturing parameters such as Tape & Reel or Pick & Place

418 6.1.4 Connector Pin Mapping

419 Table 8 shows the generic pin mapping for the QSH/QTH connector. This mapping supports a maximum of
420 four PTI channels. Note, the center ground plane must be connected to ground on the PCB.

421 **Table 8 QSH/QTH Pin Mapping with a Maximum of 4 PTIs**

Signal[1]	Pin#	Pin#	Signal[1]
VREF_DEBUG	1	2	TMS/TMSC
TCK	3	4	TDO/EXTA
TDI/EXTB	5	6	nRESET
RTCK/EXTC	7	8	nTRST_PD
nTRST/EXTD	9	10	EXTE/TRIGIN
EXTF/TRIGOUT	11	12	VREF_TRACE[2]
TRC_CLK[0]	13	14	TRC_CLK[1]
Target Presence Detect	15	16	GND
TRC_DATA[0][0]	17	18	TRC_DATA[1][0] or TRC_DATA[0][20]
TRC_DATA[0][1]	19	20	TRC_DATA[1][1] or TRC_DATA[0][21]
TRC_DATA[0][2]	21	22	TRC_DATA[1][2] or TRC_DATA[0][22]
TRC_DATA[0][3]	23	24	TRC_DATA[1][3] or TRC_DATA[0][23]
TRC_DATA[0][4]	25	26	TRC_DATA[1][4] or TRC_DATA[0][24]
TRC_DATA[0][5]	27	28	TRC_DATA[1][5] or TRC_DATA[0][25]
TRC_DATA[0][6]	29	30	TRC_DATA[1][6] or TRC_DATA[0][26]
TRC_DATA[0][7]	31	32	TRC_DATA[1][7] or TRC_DATA[0][27]

Signal[1]	Pin#	Pin#	Signal[1]
TRC_DATA[0][8]	33	34	TRC_DATA[1][8] or TRC_DATA[0][28]
TRC_DATA[0][9][3]	35	36	TRC_DATA[1][9][3] or TRC_DATA[0][29][3]
TRC_DATA[3][0] or TRC_DATA[0][10]	37	38	TRC_DATA[2][0] or TRC_DATA[1][10] or TRC_DATA[0][30]
TRC_DATA[3][1] or TRC_DATA[0][11]	39	40	TRC_DATA[2][1] or TRC_DATA[1][11] or TRC_DATA[0][31]
TRC_DATA[3][2] or TRC_DATA[0][12]	41	42	TRC_DATA[2][2] or TRC_DATA[1][12] or TRC_DATA[0][32]
TRC_DATA[3][3] or TRC_DATA[0][13]	43	44	TRC_DATA[2][3] or TRC_DATA[1][13] or TRC_DATA[0][33]
TRC_DATA[3][4] or TRC_DATA[0][14]	45	46	TRC_DATA[2][4] or TRC_DATA[1][14] or TRC_DATA[0][34]
TRC_DATA[3][5] or TRC_DATA[0][15]	47	48	TRC_DATA[2][5] or TRC_DATA[1][15] or TRC_DATA[0][35]
TRC_DATA[3][6] or TRC_DATA[0][16]	49	50	TRC_DATA[2][6] or TRC_DATA[1][16] or TRC_DATA[0][36]
TRC_DATA[3][7] or TRC_DATA[0][17]	51	52	TRC_DATA[2][7] or TRC_DATA[1][17] or TRC_DATA[0][37]
TRC_DATA[3][8] or TRC_DATA[0][18]	53	54	TRC_DATA[2][8] or TRC_DATA[1][18] or TRC_DATA[0][38]
TRC_DATA[3][9] or TRC_DATA[0][19]	55	56	TRC_DATA[2][9] or TRC_DATA[1][19] or TRC_DATA[0][39]
GND	57	58	GND
TRC_CLK[3]	59	60	TRC_CLK[2]

Notes:

1. *All unconnected signals must be tied to ground.*
2. *This configuration supports four independently clocked PTIs, but they must have the same reference voltage.*
3. *The pin mapping may be configured such that independently clocked channels are present on odd pins 17 to 35 and 37 to 55 or even pins 18 to 36 and 38 to 56. For maximum signal integrity in these scenarios, pins 35 and 36 should not be utilized as trace data signals, but should be connected to ground. This decreases the effective width of one or two of the PTI channels, but it minimizes the risk of cross talk between two adjacent asynchronous signals on the connector or PCB.*

6.1.5 Supported Pin Mapping Configurations

In order to minimize the complexity of DTC design, a finite set of supported channel configuration are defined. These configuration classes are defined in Table 9.

Table 9 Multiple Channel Configurations

Configuration Class	Debug Signals	Max Chan 0[1]	Max Chan 1[1]	Max Chan 2[1]	Max Chan 3[1]	Comment
Class0	Yes	-	-	-	-	This class is a debug only DTC with cable support for the trace header.
Class1	Yes	40	-	-	-	
Class2	Yes	40	20	-	-	
Class3	Yes	40	20	10	-	
Class4	Yes	40	20	10	10	

Notes:

1. *The number of signals in a channel may be less than the maximum designated.*

6.1.6 DTC Support for Multiple PTI Mapping

DTCs may support multiple classes of pin configurations. The DTC vendor should provide this information to the customer in a readily understandable format. For example, a vendor may have a DTC that supports two independent PTI channels and 32 data signals. The supported input configurations may be 1X32-bit PTI or 2X16-bit PTIs. The vendor could report that the DTC supports Class1 [32] and Class2 [16/16]. A similar DTC that supports a third PTI could report Class1 [32], Class2 [16/16] and Class3 [16/8/8].

6.1.7 Physical Orientation of the Connector on the PCB

The physical orientation of the connector is important for proper board and DTC design. The following recommendations define how to place the QSH connector on the PCB with the target system.

- Place the odd numbered pins of the QSH (1 to 59) closest to the Target System(s) supported through the connector. Note that this orientation is not the same as the Basic Debug Connector as described in Section 5.2.3.
- As DTC vendors may implement cable-end connectors with a right-angle connection (see Section 6.1.8) or other connector housings that extend past the physical boundaries of the PCB connector, the PCB must be clear of components that might interfere with the cable connection to the DTC. Figure 9 shows an example of a clearance issue introduced by improper board layout.
    - DTC vendors should attempt to minimize the keep-out or low clearance areas required by their cable-end connections.
    - PCB designers should consult DTC vendor documentation to determine the precise dimensions of the low clearance or keep out areas.
    - A worst-case keep out area is shown in Figure 10. The dimensions for the keep-out area were derived from the dimensions of the Samtec QCD series cable-end connector with screw down option and the QTH-EM with screw-down option. This is a worst-case description and board designers should always consult DTC vendor documentation.

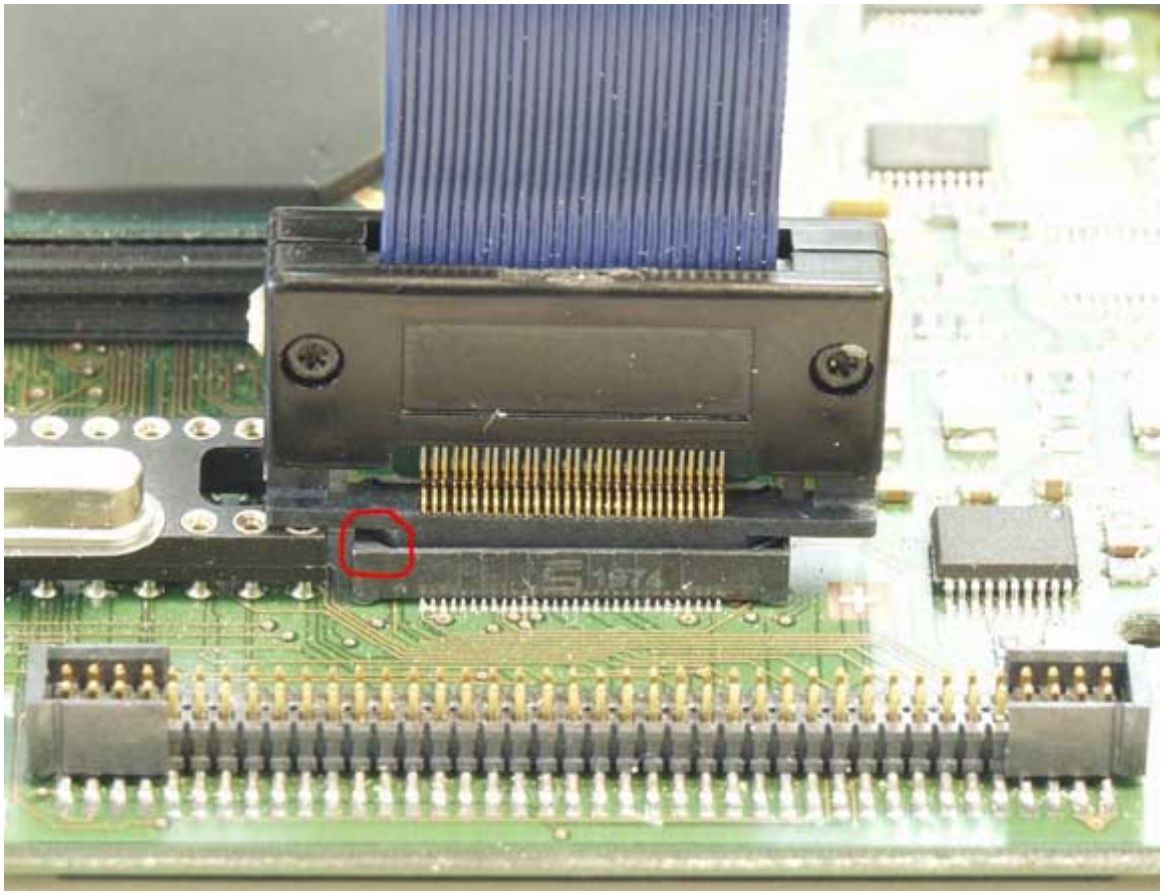


Figure 9 Cable and Board Clearance Example Showing Improper Board Design Relative to the Clearance Required for Cable-end

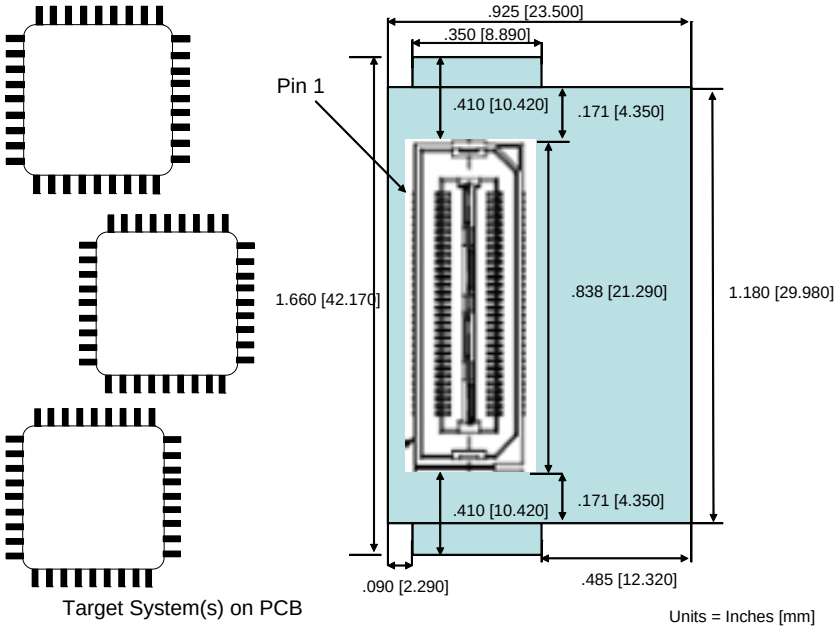


Figure 10 QSH Orientation on the PCB and Low Clearance/Keep-out Area

6.1.8 Cable-end Design Recommendations

As stated in Section 6.1.7, DTC vendors may implement cable-end connectors with various physical orientations.

- Vertical: Cable to the DTC exits the connector housing perpendicular to PCB surface
- Right-angle: Cable to the DTC exits the connector housing parallel to the PCB surface

For right-angle cable-end connectors, the cable interface to the connector should overhang the even numbered pins (2 to 60) of the connector.

6.2 Legacy Connector

The legacy physical mating connector is the AMP 38 signal Centerline Matched Impedance Connector (MICTOR). Figure 11 contains illustrations of the MICTOR PCB and Cable end connectors.

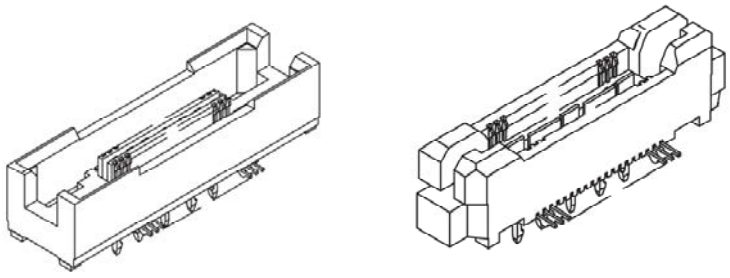


Figure 11 MICTOR PCB (left) and Cable End Connectors

The connectors are listed in Table 10 by AMP (now Tyco Electronics) part number.

Table 10 Legacy (MICTOR) Part Numbers

AMP (Tyco) Part Numbers		Mount Angle	Mount Type	Connector Summary
RoHS and ELV[1]	Legacy			
2-5767004-2	2-767004-2	Vertical	Surface	Mates with p/n 767006 to create right angle board-to-board configuration. Can also be used as board to cable connector.
5767054-1	767054-1	Vertical	Surface	Board to board or board to cable connection.
5767061-1	767061-1	Vertical	Surface	Board to board or board to cable connection.
5767044-1	767044-1	Surface	Straddle	Board to board or board to cable connection.

Notes:

1. *Connectors compliant with Restrictions on Certain Hazardous Substances (RoHS) Directive 2002/95/EC and End of Life Vehicles (ELV) Directive 2000/53/EC.*

6.2.1 Legacy Pin Mapping for Wide JTAG and a Single PTI Channel

Table 11 defines the signal to pin mapping for the legacy MICTOR HSPT connector that supports debug via Wide JTAG and also supports a single PTI with a maximum of 16 TRC_DATA signals. This pin mapping is fully compatible with legacy debug and trace systems from ARM Ltd. that use the MICTOR connector. The pin mapping in Table 11 corresponds to the ARM mapping to support Version 3 of the ETM protocol and provides a way for a PTI interface to share a MICTOR connector with an ETM interface.

Table 11 Legacy MICTOR with JTAG and a 16 Data Signal PTI

Legacy MICTOR with JTAG		16 Data Signal PTI	
Signal	**Pin#**	**Pin#**	**Signal**
TRC_DATA[0]	38	37	TRC_DATA[8][2]
EXT (TRACECTL)[1]	36	35	TRC_DATA[9][2]
Logic '1'	34	33	TRC_DATA[10][2]
Logic '0'	32	31	TRC_DATA[11][2]
Logic '0'	30	29	TRC_DATA[12][2]
TRC_DATA[1][2]	28	27	TRC_DATA[13][2]
TRC_DATA[2][2]	26	25	TRC_DATA[14][2]
TRC_DATA[3][2]	24	23	TRC_DATA[15] [2]
TRC_DATA[4][2]	22	21	nTRST
TRC_DATA[5][2]	20	19	TDI
TRC_DATA[6][2]	18	17	TMS
TRC_DATA[7][2]	16	15	TCK
EXT (VSupply)[1]	14	13	RTCK
VREF_TRACE[3]	12	11	TDO
EXT (EXTTRIG)[1]	10	9	nRESET
TRIGOUT	8	7	TRIGIN
TRC_CLK	6	5	GND
N/C	4	3	N/C
N/C	2	1	N/C

Notes:

1. *TRACECTL, VSupply and EXTTRIG are signals specified in [ARM01] and are not required to support a MIPI PTI.*
2. *Unused TRC_DATA signals should be tied to ground*
3. *VREF_TRACE always provides the reference voltage for the trace interface. If the reference voltage for the debug interface is not the same, a separate debug only connector must be used. See Section 7.2 for details.*

7 Design Considerations

7.1 Electrical Recommendations

The expectations for high speed signaling (> 25 MHz) is shown in Figure 12.

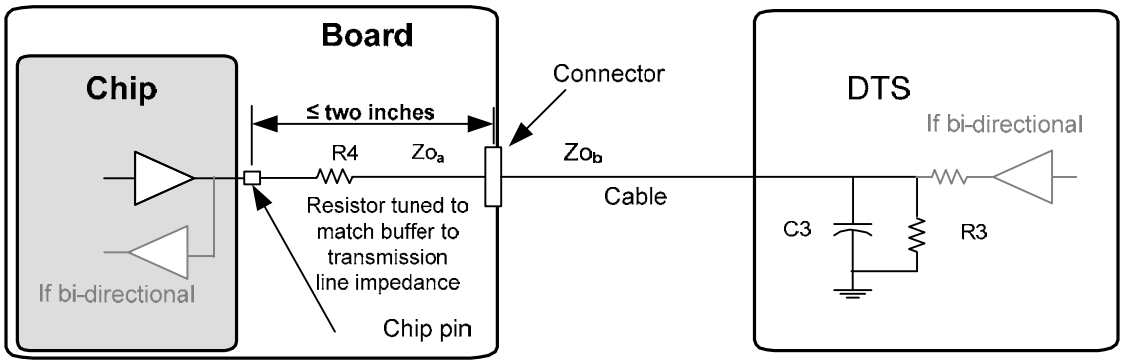


Figure 12 Signaling Model for Trace and Other High-speed Pins

Component values in Figure 12 are as follows:

- R3 shall be greater than or equal to 1K Ohms (parasitic load of DTS)
- C3 shall be smaller than 5pf (parasitic load of DTS)
- R4 shall be tuned to match the impedance of the buffer to the line impedance Zo_a and Zo_b
- Zo_a and Zo_b line impedances shall be 50 Ohms ± 10%

7.2 Signal Buffering

7.2.1 Buffering Basic Debug Signals

On board buffering of basic debug signals is not recommended. This is particularly important since the newer debug technologies (like IEEE 1149.7) may allow pin functions to change the default direction of the I/O signals on the debug interface.

7.2.2 Buffering Trace (PTI) Signals

On board buffering of the trace (PTI) signals is not recommended as it may interfere with the timing characteristics of the interface and may also cause conflicts with the signal termination implemented by the DTC.

7.3 Debug and Trace Connectors on the Same Target Board

It is possible for board designers to provide multiple debug connectors on the same target board. In general, this would be a standard debug connector for use with run control debug and an HSPT connector for trace. Several design considerations should be noted when supporting multiple debug connectors.

7.3.1 Debug Signals to Two Connectors

If the DTS and DTC systems to be used for a target system support independent trace and debug connections, it is recommended that the debug signals be routed to the debug connector only, and not to the trace connector. This configuration will eliminate any performance issues with the IEEE 1149.1 or 1149.7 interfaces that would be introduced by having PCB traces to both connectors.

7.3.2 Legacy Trace Connector and Multiple Reference Voltages

There is only one reference voltage on the HSPT connector pin mapping that is compatible with ARM ETM (Section 6.2.1). In this scenario, only one reference voltage is available on the connector. If the debug interface and the trace interface of your target system have different reference voltages, a separate debug only connector must be place on the PCB to allow the DTS and DTC to differentiate the interface reference voltages.

7.4 TRC_DATA Signal Routing on PCB

Great care must be taken when routing high speed trace signals on the PCB. When using the Samtec QSH/QTH connector, PCB designer should consult the Samtec web site for Final Inch® data regarding board layout and routing.

7.5 Overlaying Multiple PTIs on an ASIC Boundary

Many ASICs designs will have multiple modules that must share the pins of a chip level debug port (see the highlighted portion of Figure 13). These modules implement interfaces that have the characteristics of a PTI, but are not true PTI since they are not directly driving pins on the device boundary. The pin constraints for these devices generally prevent all of the modules from having dedicated PTIs. Physical design constraints (e.g. multiple on-chip die) or protocol constraints prevent the merging of trace sources into a single stream that uses a single PTI. These types of systems generally mange the shared PTI using some relatively simple pin multiplexing logic.

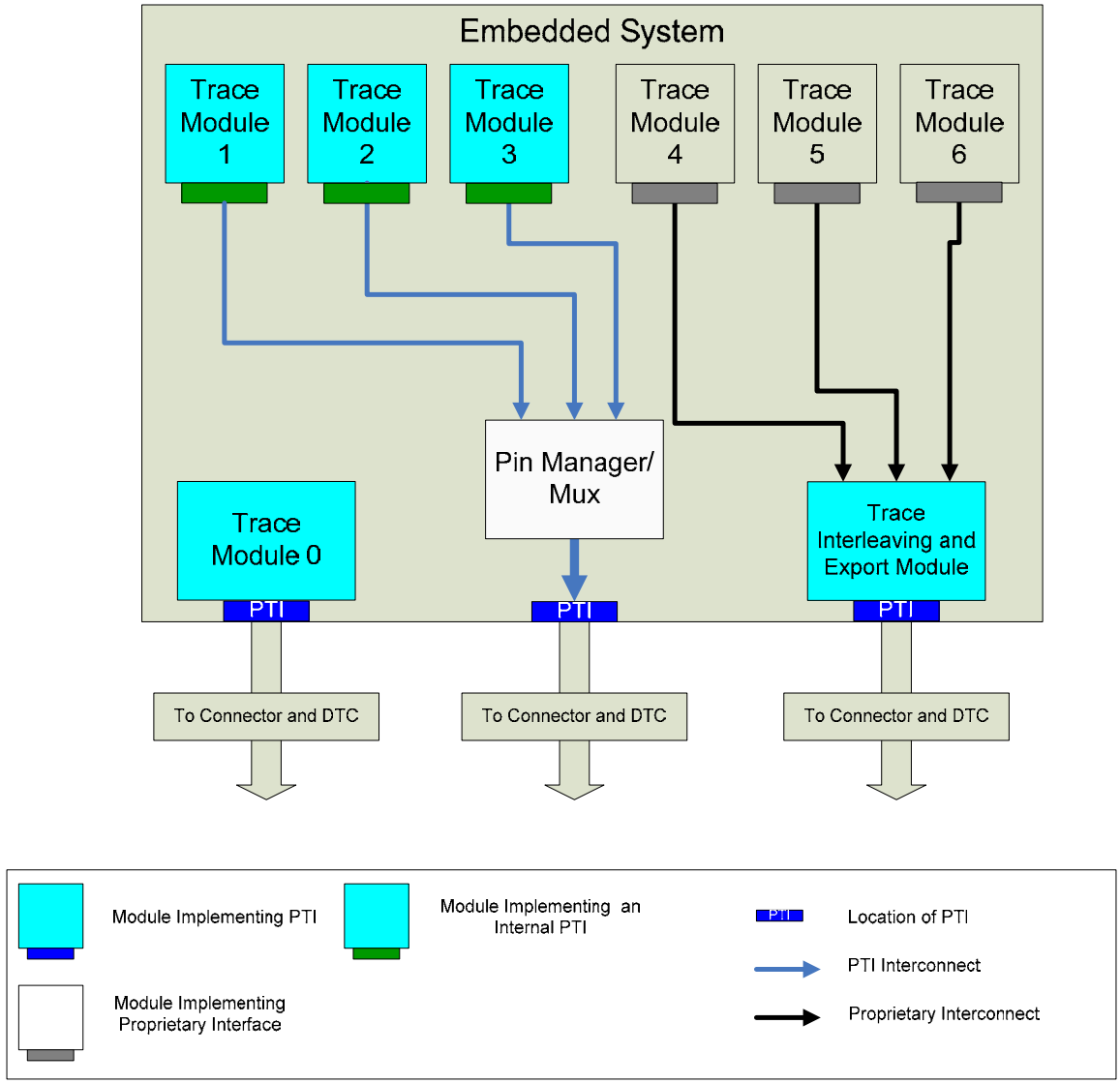


Figure 13 Multiplexed PTIs at the ASIC Level

7.5.1 Simple Shared Interfaces

In general, the simplest approach is to just overlay the PTI interfaces within the SOC with all the clocks sharing the same pin and the corresponding data signals all multiplexed with the same indexes sharing a pin. Table 12 shows a simple overlay of a 10 data single PTI and a 4 data signal PTI that share chip-level debug interface that is 11 pins wide.

Table 12 Simple Multiplexed Pin Mapping

Pin Number	Mapping for Mode A	Mapping for Mode B
0	PTI_A(TRC_CLK)	PTI_B(TRC_CLK)
1	PTI_A(TRC_DATA[0])	PTI_B(TRC_DATA[0])
2	PTI_A(TRC_DATA[1])	PTI_B(TRC_DATA[1])
3	PTI_A(TRC_DATA[2])	PTI_B(TRC_DATA[2])

Pin Number	Mapping for Mode A	Mapping for Mode B
4	PTI_A(TRC_DATA[3])	PTI_B(TRC_DATA[3])
5	PTI_A(TRC_DATA[4])	Not Connected
6	PTI_A(TRC_DATA[5])	Not Connected
7	PTI_A(TRC_DATA[6])	Not Connected
8	PTI_A(TRC_DATA[7])	Not Connected
9	PTI_A(TRC_DATA[8])	Not Connected
10	PTI_A(TRC_DATA[9])	Not Connected

The simple example poses no issues for the connector because only one trace channel can be active at any given time.

7.5.2 Complex Shared Interfaces

The complexity increases when simultaneously sharing the port. If PTI_A from the previous example had a configurable export interface width, then it would be possible to have both a narrow PTIA_A and a PTI_B active on the chip interface simultaneously.

- PTI_A = TRC_CLK[A] + TRC_DATA[A][3:0]
- PTI_B = TRC_CLK[B] + TRC_DATA[A][4:0]

The device is then capable of supporting several PTI modes:

- All debug pins dedicated to PTI_A
- All debug pins dedicated to PTI_B (some are not operational)
- Debug pins allocated to both PTI_A and PTI_B

This flexible debug interface may be very useful for developers, but it poses problems for the signal mapping to the connector. **It should be avoided if at all possible.**

The major issue is that all of the ASIC level debug pins are not assigned a monolithic function. At least one of the pins is a TRC_DATA signal in the exclusive modes and a TRC_CLK signal in the simultaneous modes. Since the connector pins (and possible the DTC input interface) are fixed functions (either clock or data, but not both), it is difficult to define a pin mapping for the ASIC signals that can be supported.

7.5.3 ASIC Pin Mapping for Simultaneous Shared Interfaces

If the ASIC pin mapping is done correctly, it is possible to support the simultaneous trace modes with the MIPI connectors, but under a very severe set of constraints:

- The chip level debug interface can only support a maximum of two simultaneously active PTIs (PTI_A and PTI_B)
- At least one of the PTI modules must have configurable export width
- The DTC must be able to associate TRC_DATA input signals with more than one TRC_CLK input
- The DTC must be able to accept data that does not follow the signal ordering indexes defined in the connector mapping
- The DTC must support the support the connector signal TRC_CLK[1]

- Board designers must take great case to minimize the impact of multiple route for one of the signals
- The only way to utilize the entire width of the interface is to use a DTC that allows data capture over a signal connected to a clock input on the connector. If this is not possible, the width of the widest PTI will be decreased by one TRC_DATA signal.

The generic proposal is to have the ASIC provide a mapping mode where the PTIs share the interface by placing the TRC_CLK signals at opposite ends of the trace interface. The data signals for the first PTI are mapped to the low order pins on the interface with the data indexes incremented. The signals for the second PTI are mapped to the high order bits of the interface with the data indexes decrementing. This provides the greatest flexibility to the system since the widths of the two PTIs can be tuned to provide the desired bandwidth allocation.

The proposal requires a simple set of definitions:

- The pins allocated to for all trace sources are called the Trace Port
- The Trace Port width is defined as $\mathbf{W_t}$ and has pin P[0] to P[$\mathbf{W_t}$ - 1]
- PTI_A has the widest data export width $\mathbf{w_a}$
- PTI_B has the same or narrower data export width $\mathbf{w_b}$

Using the previous definitions, the general rules for ASIC level PTI mapping are defined as follows:

- PTI_A:TRC_CLK is mapped to P[0]
- PTI_A:TRC_DATA[0] is mapped to P[1]
- PTI_A:TRC_DATA[$\mathbf{w_a}$] is mapped to P[$\mathbf{w_a}$ + 1]
- PTI_B:TRC_CLK is mapped to P[$\mathbf{W_t}$ - 1]
- PTI_B:TRC_DATA[0] is mapped to P[$\mathbf{W_t}$ - 2]
- PTI_B:TRC_DATA[$\mathbf{w_b}$ – 1] is mapped to P[$\mathbf{W_t}$ - $\mathbf{w_b}$]

Table 13 expands the example from Table 12 to include a simultaneous shared mode (A+B). Note that the exclusive modes still overlay the TRC_CLK for PTI_A and PTI_B on the same pin. This simple mapping is maintained so that PTI_B can still be received by a DTC that is not capable of supporting a the simultaneous shared export.

609 **Table 13 Simultaneous Export Pin Mapping**

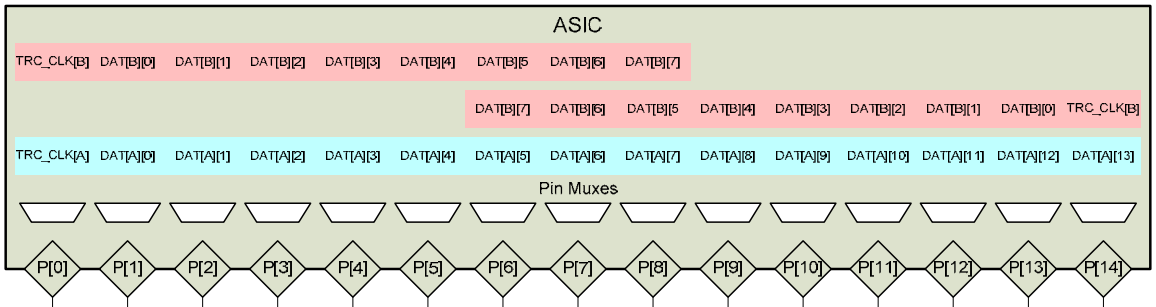
Pin Number	Mapping for Mode A	Mapping for Mode B	Mapping for Mode A+B
0	PTI_A(TRC_CLK)	PTI_B(TRC_CLK)	PTI_A(TRC_CLK)
1	PTI_A(TRC_DATA[0])	PTI_B(TRC_DATA[0])	PTI_A(TRC_DATA[0])
2	PTI_A(TRC_DATA[1])	PTI_B(TRC_DATA[1])	PTI_A(TRC_DATA[1])
3	PTI_A(TRC_DATA[2])	PTI_B(TRC_DATA[2])	PTI_A(TRC_DATA[2])
4	PTI_A(TRC_DATA[3])	PTI_B(TRC_DATA[3])	PTI_A(TRC_DATA[3])
5	PTI_A(TRC_DATA[4])	Not Connected	PTI_A(TRC_DATA[4])
6	PTI_A(TRC_DATA[5])	Not Connected	PTI_B(TRC_DATA[3])
7	PTI_A(TRC_DATA[6])	Not Connected	PTI_B(TRC_DATA[2])
8	PTI_A(TRC_DATA[7])	Not Connected	PTI_B(TRC_DATA[1])
9	PTI_A(TRC_DATA[8])	Not Connected	PTI_B(TRC_DATA[0])
10	PTI_A(TRC_DATA[9])	Not Connected	PTI_B(TRC_CLK)

610 ### 7.5.4 Connector Mapping for Simultaneous Shared Interfaces

611 The rules for mapping the ASIC pins to the connector are outlined as follows:

612 - P[0] is connected to connector pin TRC_CLK[0]
613 - P[1] is connected to connector pin TRC_DATA[0][0]
614 - P[$\mathbf{W_t}$ - 2] is connected to connector pin TRC_DATA[0][$\mathbf{W_t}$ - 2]
615 - P[$\mathbf{W_t}$ - 1] is connected to connector pin TRC_CLK[1]

616 Figure 14 shows a more generic example of the connector signal mapping for a 14 data signal PTI and an 8
617 data signal PTI. The example uses a 15 signal debug port to show how the signals span two 10 signal
618 segments on the connector.



Connector

TRC_CLK[0]	13	14	TRC_CLK[1]
Target Presence Detect	15	16	GND
TRC_DATA[0][0]	17	18	TRC_DATA[1][0] or TRC_DATA[0][20]
TRC_DATA[0][1]	19	20	TRC_DATA[1][1] or TRC_DATA[0][21]
TRC_DATA[0][2]	21	22	TRC_DATA[1][2] or TRC_DATA[0][22]
TRC_DATA[0][3]	23	24	TRC_DATA[1][3] or TRC_DATA[0][23]
TRC_DATA[0][4]	25	26	TRC_DATA[1][4] or TRC_DATA[0][24]
TRC_DATA[0][5]	27	28	TRC_DATA[1][5] or TRC_DATA[0][25]
TRC_DATA[0][6]	29	30	TRC_DATA[1][6] or TRC_DATA[0][26]
TRC_DATA[0][7]	31	32	TRC_DATA[1][7] or TRC_DATA[0][27]
TRC_DATA[0][8]	33	34	TRC_DATA[1][8] or TRC_DATA[0][28]
TRC_DATA[0][9]	35	36	TRC_DATA[1][9] or TRC_DATA[0][29]
TRC_DATA[3][0] or TRC_DATA[0][10]	37	38	TRC_DATA[2][0], TRC_DATA[1][10] or TRC_DATA[0][30]
TRC_DATA[3][1] or TRC_DATA[0][11]	39	40	TRC_DATA[2][1], TRC_DATA[1][11] or TRC_DATA[0][31]
TRC_DATA[3][2] or TRC_DATA[0][12]	41	42	TRC_DATA[2][2], TRC_DATA[1][12] or TRC_DATA[0][32]
TRC_DATA[3][3] or TRC_DATA[0][13]	43	44	TRC_DATA[2][3], TRC_DATA[1][13] or TRC_DATA[0][33]
TRC_DATA[3][4] or TRC_DATA[0][14]	45	46	TRC_DATA[2][4], TRC_DATA[1][14] or TRC_DATA[0][34]
TRC_DATA[3][5] or TRC_DATA[0][15]	47	48	TRC_DATA[2][5], TRC_DATA[1][15] or TRC_DATA[0][35]
TRC_DATA[3][6] or TRC_DATA[0][16]	49	50	TRC_DATA[2][6], TRC_DATA[1][16] or TRC_DATA[0][36]
TRC_DATA[3][7] or TRC_DATA[0][17]	51	52	TRC_DATA[2][7], TRC_DATA[1][17] or TRC_DATA[0][37]
TRC_DATA[3][8] or TRC_DATA[0][18]	53	54	TRC_DATA[2][8], TRC_DATA[1][18] or TRC_DATA[0][38]
TRC_DATA[3][9] or TRC_DATA[0][19]	55	56	TRC_DATA[2][9], TRC_DATA[1][19] or TRC_DATA[0][39]
GND	57	58	GND
TRC_CLK[3]	59	60	TRC_CLK[2]

Figure 14 Example of PTI Signal Overlays at the ASIC Boundary

Annex A Debugger and Target Presence Detection

The marked GND-detect pin may be used to detect the presence of the DTS (debug controller) by the TS (target) or the presence of the TS by the DTS. The pin is directly connected to GND if any side does not require detection. The following example circuit allows the detection in both directions with one GND pin.

This circuit uses the following premises:

- If target power is detected, the target is both powered and connected
- If target power is not detected then the DTS does not drive the connected signal to ground
- If target power is not detected and the DTS is not driving the connected signal the DTS determines if the DTS is connected by detecting a voltage below the threshold on this signal
- Once target power is detected the DTS may drive the connected signal to a zero
- When target power is off, the connected path to the TS power supply appears as a near ground potential

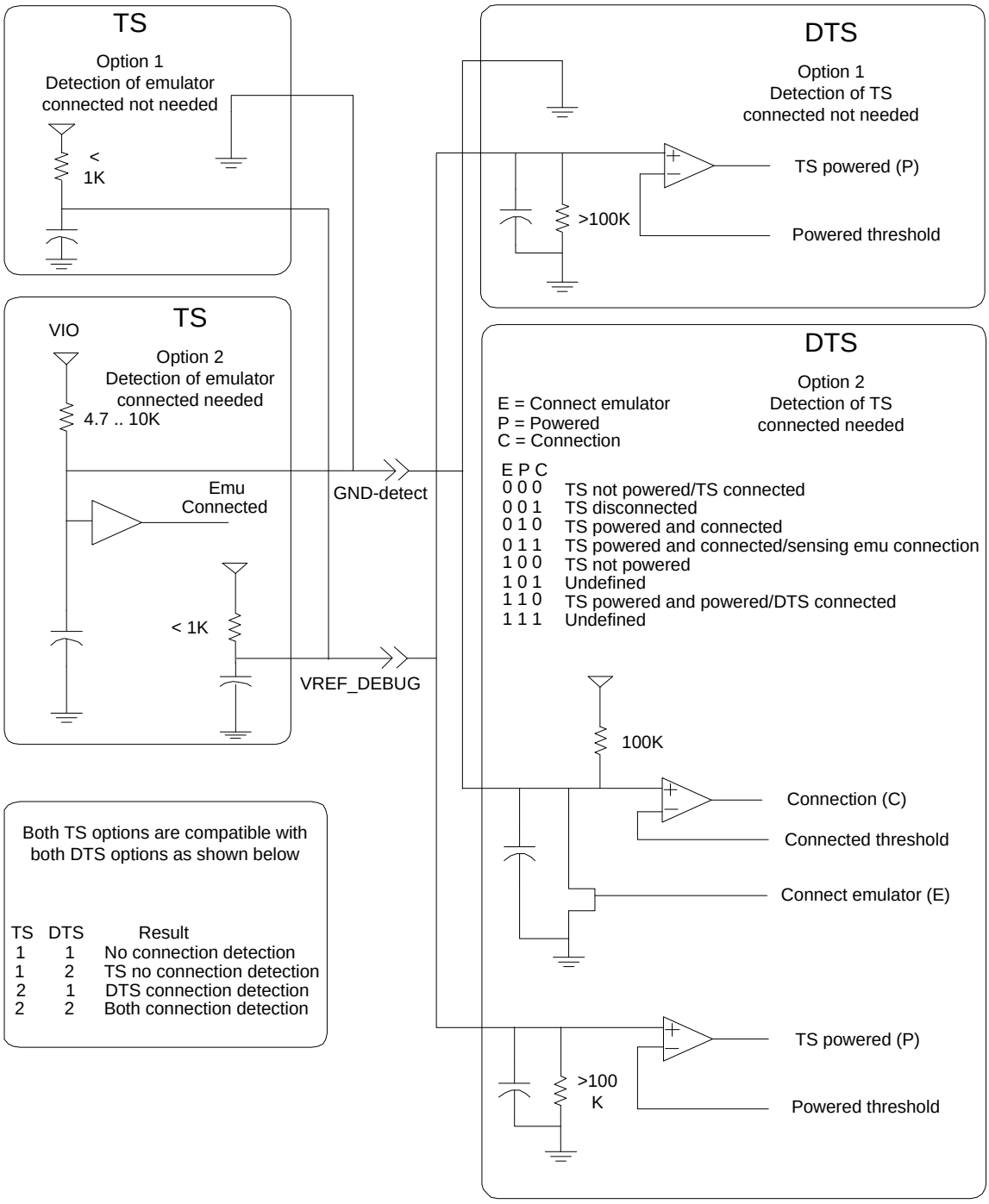


Figure 15 Bi-directional Presence Detect

Annex B ARM ETM Pin Mapping

The ARM ETM is a very popular on-chip debug component. It is anticipated that many systems may require mapping a MIPI PTI and an ARM ETM interface on the same connector (and device pins). This appendix to the connector recommendation defines the mapping the ETM output to one or more PTI channels on the new MIPI HSPT connector. The Table 14 through Table 20 shows the mappings for ETMv1, ETMv2 and ETMv3 signal protocols.

B.1 Standard ETM Pin Mapping

Table 14 ETMv1/2 to PTI Mapping

PTI Signal (C^1 = 0-3)	ETMv1 (W^2 = 4, 8, 16)	ETMv2 (W^2 = 4, 8, 16)
TRC_CLK[C]	TRACE_CLK	TRACE_CLK
TRC_DATA[C][0]	PIPESTAT[0]	PIPESTAT[0]
TRC_DATA[C][1]	PIPESTAT[1]	PIPESTAT[1]
TRC_DATA[C][2]	PIPESTAT[2]	PIPESTAT[2]
TRC_DATA[C][3]	TRACESYNC	PIPESTAT[3]
TRC_DATA[C][W+3:4]	TRACEDATA[W-1:0]	TRACEDATA[W-1:0]

Notes:

1. *C = PTI Channel Index on the connector*
2. *W = ETM Port Width*

The ETMv3 signal protocol has an optional TRACECTL signal. The use of this signal is being deprecated, but some legacy tools still require it for proper trace capture. When the tools requirements for a system are known, one of the two pin mappings in Table 15 should be used.

Table 15 ETMv3 to PTI Mapping

PTI Signal (C^1 = 0-3)	ETMv3 (W^2 = 1-32)	ETMv3 (W^2 = 1-32)
	Recommended	**Legacy Only**
TRC_CLK[C]	TRACE_CLK	TRACE_CLK
TRC_DATA[[C][0]	GND	TRACECTL
TRC_DATA[[C][1]	TRACEDATA[0]	TRACEDATA[0]
	. . .	
TRC_DATA[C][W-2]	TRACEDATA[W-2]	TRACEDATA[W-2]
TRC_DATA[C][W-1]	TRACEDATA[W-1]	TRACEDATA[W-1]

Notes:

1. *C = PTI Channel Index on the connector*
2. *W = ETM Port Width*

There may be scenarios where ASICs and PCBs are required to support both ETMv3 signal formats. Since ASIC pins are a finite resource, care must be taken at the ASIC level to insure that the overlay of both signal formats maximizes the utility of the pins assigned to the interface. This can be achieved by assigning

the TRACECTL signal to the lowest order data pin when it is required. The trace data signals are then mapped to the remaining signals on the interface. When TRACECTL is not required, the trace data signals are mapped beginning at the lowest order data signal on the interface. Figure 16 illustrates this overlay for a ASIC level trace interface with 10 signals.

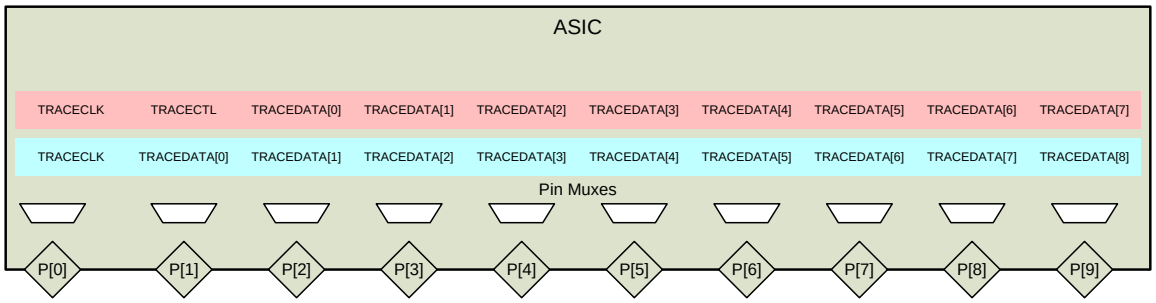


Figure 16 ASIC Level Multiplexing with, and without, TRACECTL

The overlay method previously described results in the connector signal mappings shown in Table 16. This type of pin mapping may not be supported natively by many debug tools because the data signal overlay is not a one-to-one mapping. Adapters may be required to get the proper signal alignment for the tool**.**

Table 16 ETMv3 to PTI Mapping for Overlay Scenario

PTI Signal (C^1 = 0-3)	ASIC Pin Index	ETMv3 (W^2 = 1-32)	ETMv3 (W2 = 1-32)
		Legacy Mode	No TRACECTL Mode
TRC_CLK[C]	0	TRACE_CLK	TRACE_CLK
TRC_DATA[[C][0]	1	TRACECTL	TRACEDATA[0]
TRC_DATA[[C][1]	2	TRACEDATA[0]	TRACEDATA[1]
		. . .	
TRC_DATA[C][W-2]	W-1	TRACEDATA[W^3-2]	TRACEDATA[W-2]
TRC_DATA[C][W-1]	W	TRACEDATA[W^3-1]	TRACEDATA[W-1]

Notes:

1. *C = PTI Channel Index on the connector*
2. *W = ETM Port Width*
3. *For a given number of pins used for the PTI, the Legacy Mode ETM Port Width W has one less data signal than the No TRACECTL Mode.*

B.2 Demultiplexed ETM Pin Mapping

The demultiplexed pin mapping requires two (2) PTI channels with a common clock.

Table 17 Demultiplexed ETMv1 to PTI Mapping

PTI Signal (Ca^1= 0/2)	ETMv1 (W^2 = 4, 8, 16)	PTI Signal (Cb1 = 1/3)	ETMv1 (W^2= 4, 8, 16)
TRC_CLK[Ca]	TRACE_CLK	TRC_CLK[Cb]	TRACE_CLK

PTI Signal (Ca[1]= 0/2)	ETMv1 (W[2] = 4, 8, 16)	PTI Signal (Cb1 = 1/3)	ETMv1 (W[2]= 4, 8, 16)
TRC_DATA[Ca][0]	PIPESTAT_A[0]	TRC_DATA[Cb][0]	PIPESTAT_B[0]
TRC_DATA[Ca][1]	PIPESTAT_A[1]	TRC_DATA[Cb][1]	PIPESTAT_B[1]
TRC_DATA[Ca][2]	PIPESTAT_A[2]	TRC_DATA[Cb][2]	PIPESTAT_B[2]
TRC_DATA[Ca][3]	TRACESYNC_A	TRC_DATA[Cb][3]	TRACESYNC_B
TRC_DATA[Ca][W+3:4]	TRACEDATA_A[W-1:0]	TRC_DATA[Cb][W+3:4]	TRACEDATA_B[W-1:0]

Notes:

1. *Ca and Cb = PTI Channel Index on the connector. The demultiplexed mapping is supported using symmetric channels on the connector (a/b = 0/1 or 2/3).*
2. *W = ETM Port Width.*

Table 18 Demultiplexed ETMv2 to PTI Mapping

PTI Signal (Ca[1]= 0/2)	ETMv1 (W[2] = 4, 8, 16)	PTI Signal (Cb[1] = 1/3)	ETMv1 (W[2]= 4, 8, 16)
TRC_CLK[Ca]	TRACE_CLK	TRC_CLK[Cb]	TRACE_CLK
TRC_DATA[Ca][0]	PIPESTAT_A[0]	TRC_DATA[Cb][0]	PIPESTAT_B[0]
TRC_DATA[Ca][1]	PIPESTAT_A[1]	TRC_DATA[Cb][1]	PIPESTAT_B[1]
TRC_DATA[Ca][2]	PIPESTAT_A[2]	TRC_DATA[Cb][2]	PIPESTAT_B[2]
TRC_DATA[Ca][3]	PIPESTAT_A[3]	TRC_DATA[Cb][3]	PIPESTAT_B[3]
TRC_DATA[Ca][W+3:4]	TRACEDATA_A[W-1:0]	TRC_DATA[Cb][W+3:4]	TRACEDATA_B[W-1:0]

Notes:

1. *Ca and Cb = PTI Channel Index on the connector. The demultiplexed mapping is supported using symmetric channels on the connector (Ca/Cb = 0/1 or 2/3).*
2. *W = ETM Port Width.*

B.3 Multiplexed ETM Pin Mapping

With the multiplexed pin mapping, each device pin is multiplexed between two ETM output signals. These are sampled on opposite edges of the trace clock.

Table 19 Multiplexed ETMv1 to PTI Mapping

PTI Signal (C[1] = 0-3)	ETMv1 (W[2] = 4, 8, 16)
TRC_CLK[C]	TRACE_CLK
TRC_DATA[C][0]	PIPESTAT[0] + TRACESYNC
TRC_DATA[C][1]	PIPESTAT[1] + TRACEDATA[1]
TRC_DATA[C][2]	PIPESTAT[2] + TRACEDATA[2]
TRC_DATA[C][3]	TRACEDATA[0,3]
TRC_DATA[C][4]	TRACEDATA[4,5]
TRC_DATA[C][5]	TRACEDATA[6,7]
TRC_DATA[C][6]	TRACEDATA[8,9]
TRC_DATA[C][7]	TRACEDATA[10,11]

PTI Signal (C^1 = 0-3)	ETMv1 (W^2 = 4, 8, 16)
TRC_DATA[C][8]	TRACEDATA[12,13]
TRC_DATA[C][9]	TRACEDATA[14,15]

Notes:

1. *C = PTI Channel Index on the connector.*
2. *W = ETM Port Width.*

Table 20 Multiplexed ETMv2 to PTI Mapping

PTI Signal (C^1 = 0-3)	ETMv2 (W^2 = 4, 8, 16)
TRC_CLK[C]	TRACE_CLK
TRC_DATA[C][0]	PIPESTAT[0] + PIPESTAT[3]
TRC_DATA[C][1]	PIPESTAT[1] + TRACEDATA[1]
TRC_DATA[C][2]	PIPESTAT[2] + TRACEDATA[2]
TRC_DATA[C][3]	TRACEDATA[0,3]
TRC_DATA[C][4]	TRACEDATA[4,5]
TRC_DATA[C][5]	TRACEDATA[6,7]
TRC_DATA[C][6]	TRACEDATA[8,9]
TRC_DATA[C][7]	TRACEDATA[10,11]
TRC_DATA[C][8]	TRACEDATA[12,13]
TRC_DATA[C][9]	TRACEDATA[14,15]

Notes:

1. *C = PTI Channel Index on the connector.*
2. *W = ETM Port Width.*

Annex C Nexus Pin Mapping

The Nexus 5001 Forum™ has defined a standard for a debug interface that is very popular in the embedded automotive sector. It is anticipated that many systems may require mapping a MIPI PTI and a Nexus interface on the same connector (and device pins). This annex to the connector recommendation defines the mapping the Nexus output interface to one or more PTI channels on the new MIPI HSPT connector. Table 21 and Table 22 show the pin mappings.

Table 21 Nexus Message Interface to PTI Mapping

PTI Signal (C^1 = 0-3)	Nexus Aux Output (W^2 = 1-32)
TRC_CLK[C]	MCKO
TRC_DATA[C][0]	MSEO[0]
TRC_DATA[C][1]	MSEO[1]
TRC_DATA[C][W+1:2]	MDO[W-1:0]

Notes:

1. *C = PTI Channel Index on the connector.*
2. *W = Nexus Port Width.*

The Nexus standard also defines extended signals used for basic debug functions. These signals can be mapped to the MIPI connectors using Table 22. The overlay scheme applies to any connector that supports the designated MIPI signal.

Table 22 Nexus Extended Pin Mapping

MIPI Signal Name	Nexus Signal
TRIGIN	EVTI
TRIGOUT	EVTO
nTRST_PD	RSTI
RTCK	RDY