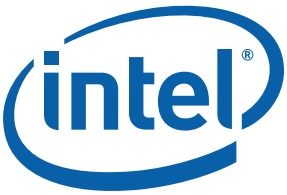


Designing Embedded Systems For Testability

Application Note

October 2007

Order Number: 318518-001US

Contents

Figures

Tables

Revision History

Date	Revision	Description
October 2007	001	Initial release

1.0 Introduction

The purpose of this document is to provide a context of how to utilize the test and debug features to design systems using Intel embedded processors and chipsets.

There are various interfaces of the design which must be implemented. In some cases the interfaces must be shared, depending upon operation at a particular point in time. For example, the eXtended Debug Port (XDP) port utilizing the Test Access Port (TAP) controller during debug and manufacturing/self test equipment utilizing the TAP controller during board verification. Understanding the implementation is key to designing a succesful product in terms of time-to-market.

This Application Note specifically focuses on the TAP controller facilities, XDP facilities, and the XOR facilities provided on the chips.

This document illustrates design considerations for using the Joint Test and Access Group (JTAG)/Tap interface software development, software debug, manufacturing testing, and manufacturing debug. It also explains how to merge an XOR chain with a boundary scan chain.

Each usage model presents unique design challenges which are addressed through an example JTAG/TAP design implementation. The example design is conceptual and has not been fully validated. The design examples intent are to illustrate a circuit implementation merge these concepts.

2.0 References

This Application Note is specific to the Intel® Core™2 Duo processor family and the Intel® 3100 Chipset. Examples of referenced documents for this CPU/Chipset combination are given below. For other CPU/Chipset combinations, consult the appropriate documentation that is specific to that processor or chipset. The design concepts illustrated in the remaining text are the same.

Table 1. Examples of Referenced Documents

Document Type	Example Document Title	Document Number
Processor EMTS	Intel® Core™2 Duo Mobile Processor for Intel® Centrino® Duo Technology Doc Number	639419
Processor Datasheet	Intel® Core™2 Extreme Processor X6800 and Intel® Core™2 Duo Desktop Processor E6000 and E4000Ä Sequences	313278
Chipset Datasheet	Intel® 3100 Chipset Datasheet	313458
Processor Debug Port Design Guide	Debug Port Design Guide for UP/DP Systems	313373
Platform Debug Port Design Guide	Debug Port Design Guide for Intel® 3100 Chipset Systems	315844
Industry Specification	IEEE1149.1-2001 Specification	NA

3.0 Overview

When designing a product it is important to be able to verify the manufacturing and proper operation of the end product. Intel's components provide two key hardware features that aid in the debug and test of such products:

- TAP Controller Interface (CPU and Some Chipsets)
- XOR Chain (On Chipsets Only)

The TAP Controller Interface can be used during the debug phase of the design cycle via an XDP tool provided by third party vendors. The TAP Controller can also be used during the manufacturing phase as a JTAG connection used for implementing boundary scan. As such, the TAP Controller Interface pins serve a dual purpose. It is important to be able to understand the needs of your particular design and how both can be used, early on in the development effort.

This Application Note provides design considerations for using each of the debug and test interfaces mentioned above correctly and for maximum value. Specifically, it will cover the following topics:

- Merging XDP and JTAG implementations together
- Considerations for merging XOR environments with JTAG environments
- Testing Voltage and Grounds

If the reader is not familiar with boundary scan or XDP methodology, the following documents provide the proper background:

- IEEE1149.1-2001 specification
- Debug Port Design Guide for UP/DP Systems, Doc Number 313373
- American Arium* Emulator specification supporting the processor and chipset you are designing with and/or any other supported emulator specification

However, to facilitate the reader a brief description of the XDP debugger, Boundary Scan, and an XOR Chain will be provided.

3.1 XDP Debugger

A debug port is a connection into a target system environment that provides access for performing JTAG operations, run control operations, and system control resources. This tool requires the use of the TAP controller to operate.

Debug ports come in three styles; XDP, XDP-Sinned, and ITP700Flex. The XDP is a 60-pin, small form factor connector, and is the recommended implementation. It provides additional silicon / system debug resources compared to other debug port implementations and provides an expansion path for future capabilities. XDP is not only generally required if debug assistance by Intel is necessary but will save one valuable time during debug.

Most commercially available run control tools, like the EMC-50 from American Arium*, only interface to an XDP port. Additionally XDP extends a serial JTAG implementation by implementing two separate clock domains for the scan chains. Dividing the system scan chains into two domains allows one to increase the operating frequency of the processor scan chain by moving the typically slower chipset JTAG agents to a completely separate clock domain. XDP also provides assistance for managing and debug of power, reset, and clocking implementations.

3.2 Joint Test and Access Group (JTAG) / Boundary Scan

Given today's complex board design/package technology, it becomes very difficult to test the components connectivity, the components themselves, and their desired function within the system. JTAG provides a mechanism which allows this type of testing to occur via the TAP controller.

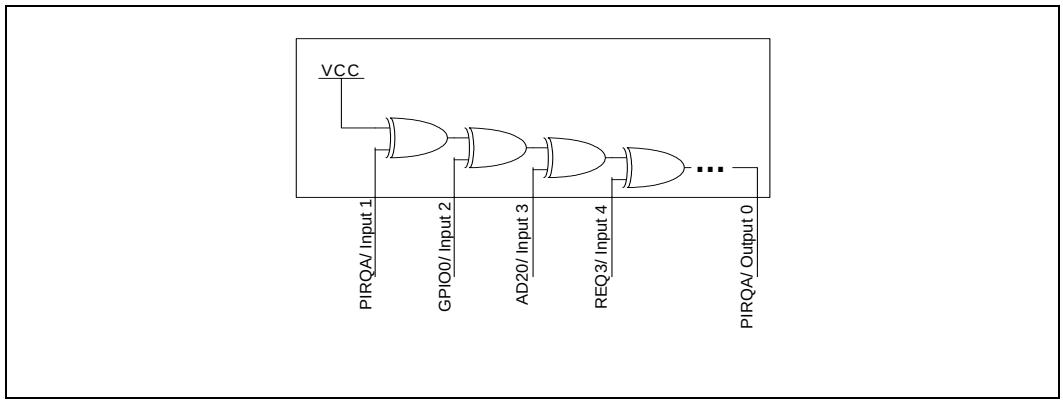
To allow for board level testing Boundary Scan cells containing an input and output flip flop are placed adjacent to every Input/Output (I/O) pin and are then muxed with each I/O pin's output. These cells then are chained together via a serial string which allows one to develop a pattern test to, for example, change the I/O of device A to all outputs and the I/O of device B to all inputs. In this way, the shifted values could be placed on the outputs of device A and sampled on device B. The pattern could then be shifted and or modified to allow a different value at the output and different captured value at the input. If all information matches between the value at the output and corresponding captured inputs and the patterns are sufficient for stressing the interface then the board level connection can be verified.

In addition to allowing these cells to be I/O, the TAP controller can place parts into a High-Z state to isolate components within the system to aid in debug.

3.3 XOR Chains

Even for Chipsets which do not support a TAP controller XOR chains are present. XOR chains allows straps to be set at initialization. The straps place the pins in the chain as inputs with each input attached to one leg of the XOR chain and the other leg attached to the output of the previous XOR device in the chain. The first internal connection in the XOR chain is connected to a logic high and the output of the last XOR device in the chain is connected to a pin which is configured as an output. Figure 1 shows an example of what is described above. Please keep in mind that the figure does not depict the actual subset of any chipset but is used for illustration purposes only.

Figure 1. Subset of XOR Chain



4.0 TAP Controller Considerations

When designing logic to exercise the TAP controller, it is important to find a solution that will allow the design to be easily debugged and tested. The TAP controller is instrumental in achieving these objectives. Although the TAP controller is used for both test modes (XDP and Boundary Scan), the configuration is slightly different so the design has to have the flexibility to be configured for either case to avoid messy and/or expensive rework.

Figure 2 illustrates the conceptual design for XDP configuration while Figure 3 illustrates the conceptual design for the JTAG/Boundary Scan configuration. The major differences between the two modes of operation are:

- XDP requires only the CPU (in most cases) while B-SCAN has a serial chain of all devices
- XDP typically runs at a higher rate of speed and usually requires isolation to reduce bus loading

Figure 2. XDP Conceptual Design

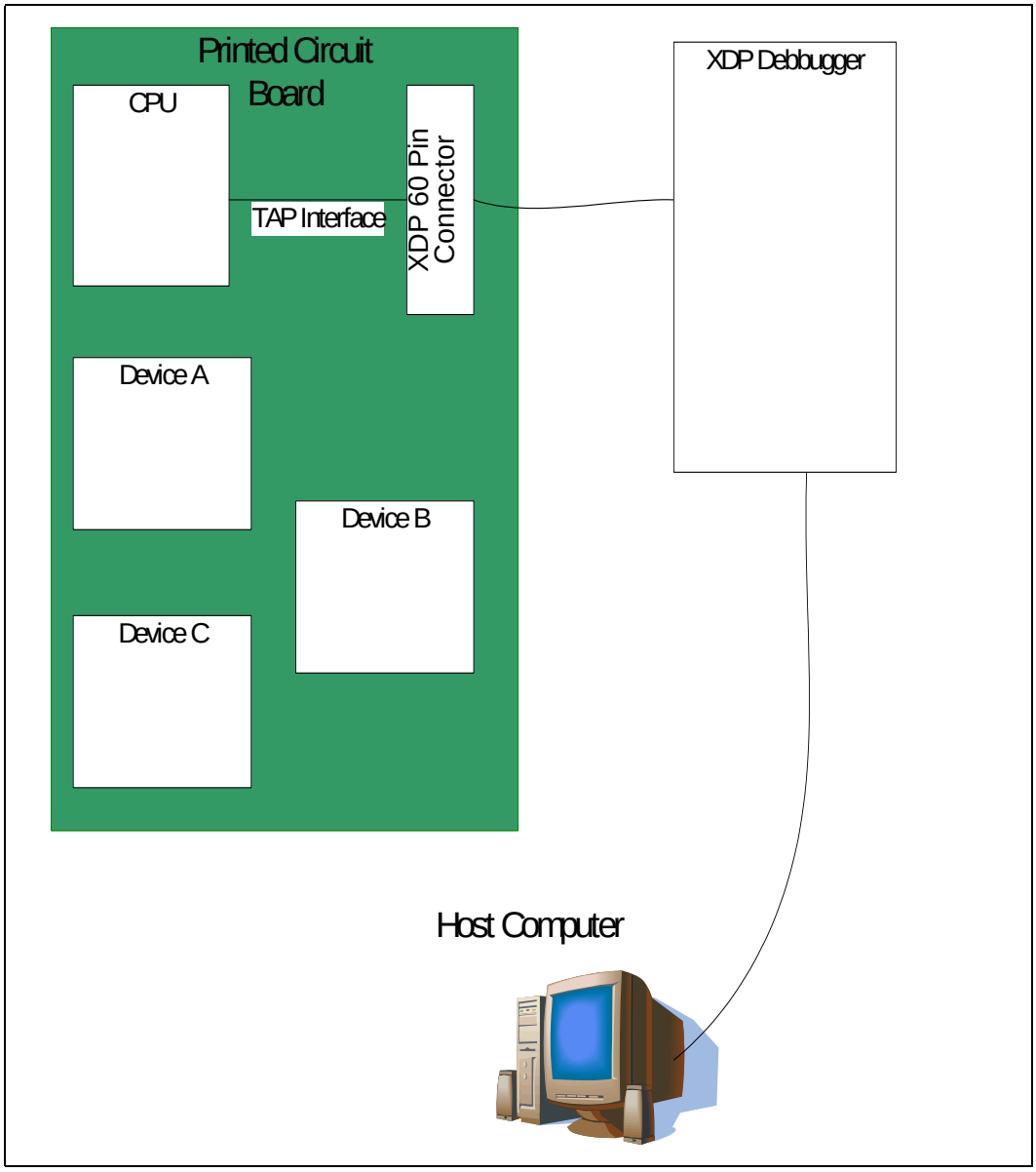
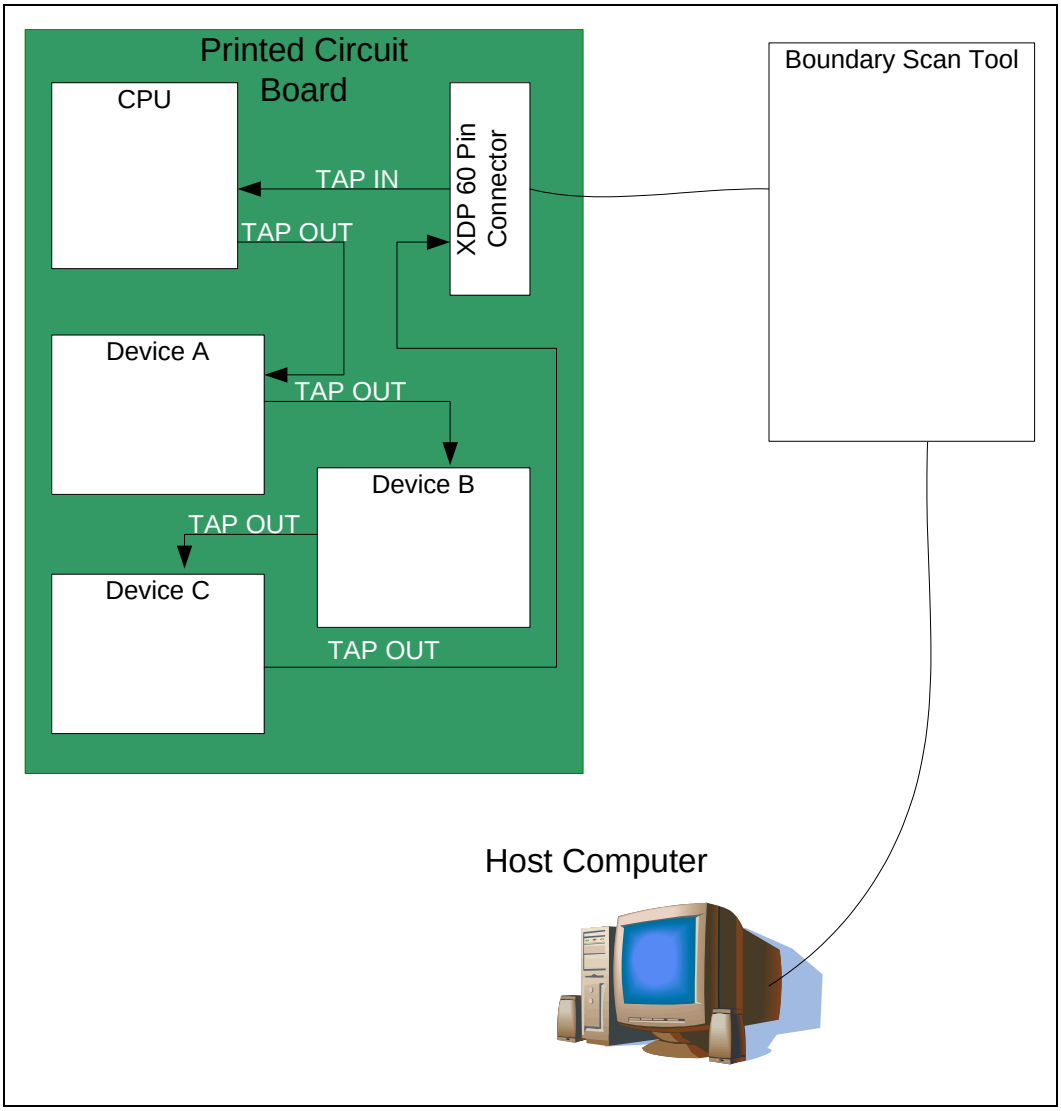


Figure 3. JTAG/Boundary SCAN Conceptual Design



The remainder of this section will outline a methodology by which the two conceptual design implementations described in Figure 2 and Figure 3 can be merged together. Please keep in mind that this description to follow is one of several methodologies.

4.1 Single Physical Processor Package Configuration

Figure 4 shows an implementation which allows both the TAP controller to operate as a boundary scan interface and a debug interface. This implementation allows the ultimate flexibility in the design at the lowest cost via resistor population options. Figure 5 illustrates the same basic design utilizing tri-state buffers and muxes. This configuration will make board level changes quicker and easier but will cost more to implement. This will be a trade-off that each design must make based upon their requirements.

Another option that was considered is jumpers in place of the resistors. However, the intent of this Application Note was to make the design as easy possible to manufacture and produce. With that stated, headers create EMI problems, they are usually through hole devices which create different solder flows, and can be easily bent in a re-work/ repair situation causing additional issues. Therefore, that type of design will not be shown as an implementation and headers will be minimized in the designs shown.

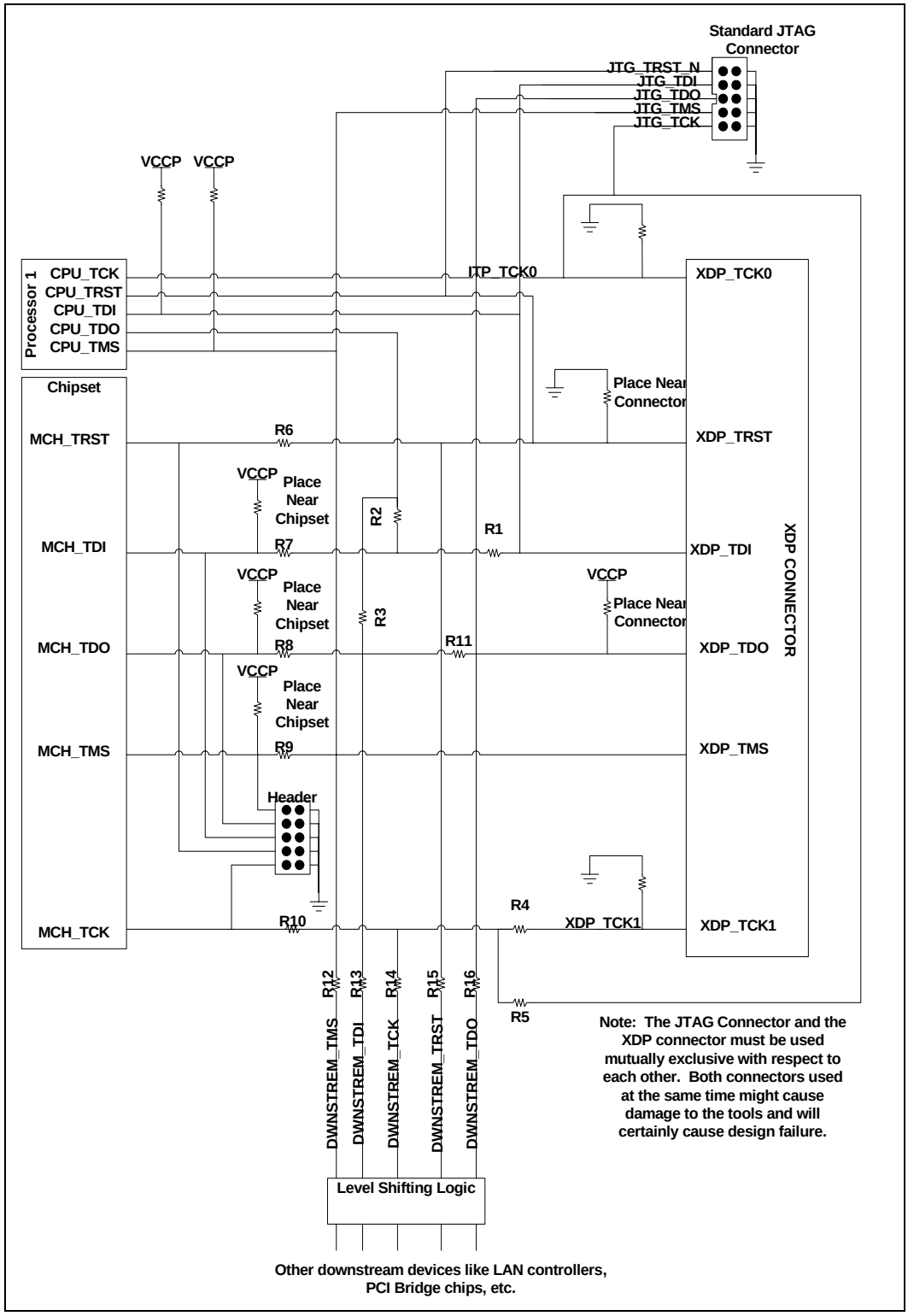
The implementation in Figure 4 shows sixteen 0 ohm resistors, two connectors, and a header all as stuffing options which can be utilized to implement whatever a design might need. The header in the design below is just a path to pull the signals to a known state (VSS in this case) when the chipset is removed from the system.

- **Caution: Care must be taken with the XDP_TDI, XDP_TDO, and XDP_TMS signals as there is a pull-up resistor to VCCP on these signals. If you populate the header for these signals, the pull-up resistors need to be removed.**

The design must still adhere to the rules called out in the debug port design guides as well as the platform design guides for the respective chipset.

Table 2 lists the different stuffing options which can be populated depending upon the desired need. The following sections will break down the exact paths which need to be followed and why they were implemented.

Figure 4. Schematic for uP Resistor Based Tap Controller Design



Population options for selecting which items will be in the chain in a UP configuration in either JTAG or ITP/XDP mode of operations. The CPU must always be in the chain in any configuration.

Table 2. Resistor Stuffing Options for uP TAP Controller Design

Resistors	CPU		CPU & Chipset		CPU & Other Devices		CPU, Chipset, & Other Devices		XDP with CPU		XDP with CPU & Chipset	
	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP
R1		X		X		X		X	X		X	
R2		X	X			X	X			X		X
R3	X			X	X			X	X		X	
R4		X		X		X		X	X		X	
R5	X		X		X		X			X		X
R6		X	X			X	X			X	X	
R7		X	X			X	X			X	X	
R8		X	X			X	X			X	X	
R9		X	X			X	X			X	X	
R10		X	X			X	X			X	X	
R11	X		X			X		X	X		X	
R12		X		X	X		X			X		X
R13		X		X	X		X			X		X
R14		X		X	X		X			X		X
R15		X		X	X		X			X		X
R16		X		X	X		X			X		X

Notes:

1. All resistors listed above are 0 ohm resistors
2. P in the table above denotes the resistor must be populated
3. NP in the table above denotes the resistor must not be populated
4. When resistors R6 through R10 are populated as denoted by an X under the P column, the header must be left open (or unconnected).
5. When resistors R6 through R10 are not populated as denoted by an X under the NP column, the header must be populated (or connected).

Legend: Quick Switch Implementation requires four switches and four jumpers to the enable signals

3 pin jumper controlling switch based on XDP or JTAG usage.

3 pin jumper controlling switch based on Chipset in use or not.

3 pin jumper controlling switch based on Other devices in chain or not.

3 pin jumper controller TDI/TDO routing, please refer to table above to configure

4.1.1 Boundary SCAN Chain Population Option

With the design shown in Figure 4 and Table 2, it is possible to configure the design as a TAP controller which connects to a boundary scan tool which in turn places all JTAG compatible devices in the SCAN chain. However, selection options can also be made to allow isolation of components for debug purposes. Four possible configurations are allowable, each requiring a CPU:

- CPU Only
- CPU with Chipset
- CPU with all logic except Chipset (in that table as “Other Devices”)

- CPU with Chipset and Other Devices

To aid in understanding and allow an easy implementation conversion between resistors to a series of bus switches/tri-state buffers, the resistors have been grouped into four categories.

- R1, R4, and R5 control the clocks and TDI when going between Boundary Scan and XDP mode
- R2 and R3 control the TDI to TDO chaining depending upon mode of operation
- R6, R7, R8, R9, and R10 control routing to the Chipset
- R11, R12, R13, R14, R15, and R16 control routing to Other Devices

The following items must always be mutually exclusive in the circuit:

- R2 and R3
- R4 and R5
- JTAG connector and XDP connector usage
- The HEADER connections and R6, R7, R8, R9, and R10

Figure 4 shows that to remove the Chipset from the scan chain (please keep in mind that not all chipsets support boundary scan - some only support XOR Chaining) R6, R7, R8, R9, and R10 have to be depopulated. At the same time, the HEADER would be populated (XDP_TCK1 and XDP_TRST only) to insure that the signals to the chipset are defined inputs.

If the “Other Devices” are to be taken out of the chain, R11 must be populated and R12, R13, R14, R15, and R16 must be de-populated. Please note that not shown is how to terminate the inputs of the “Other Devices” when it is removed from the chain. Similar type stuffing pull-up/down resistors or a header could be placed below the R12, R13, R14, R15, and R16 to allow the entire set of devices downstream to be terminated to a known value. This termination is outside the scope of this Application Note. Also, in order to meet signal drive capabilities due to multiple devices being on a single chain, some type of buffering must be put in line to allow the output drivers to have enough signal strength. The device listed as level shifting logic will guarantee that buffering.

If downstream devices are operating at a higher voltage level than the processor and chipset, voltage translation circuitry must be put in place. Transistor logic or dual voltage bus switches can be used to achieve this. However, designers must be certain that the voltage levels between the chipset and CPU are compatible. If not, voltage translation would be required after resistors R6, R7, R8, R9, and R10.

R1 and R4 will be depopulated and R5 will be populated when in JTAG/Boundary Scan mode of operation. These resistors will be populated the opposite way when in XDP mode of operation.

R2 and R3 are implementation specific and the table should be referenced to determine how these resistors should be stuffed in a given configuration.

It is important to note that all design rules called out in the Design Guides must be adhered to. Care must be taken to place the resistors/connectors during layout in locations to minimize signal integrity issues due to stub effects. Also, be certain to place proper end line termination. If this is not done, the interface speed may have to be slowed down, which will increase test time or the design may fail.

Figure 5 shows how the resistor implementation in the previous configuration shown in Figure 4 can be modified to use bus switches and tri-state buffers. Four 3-pin headers and/or simple two pole switches could be used to place the parts in any configuration.

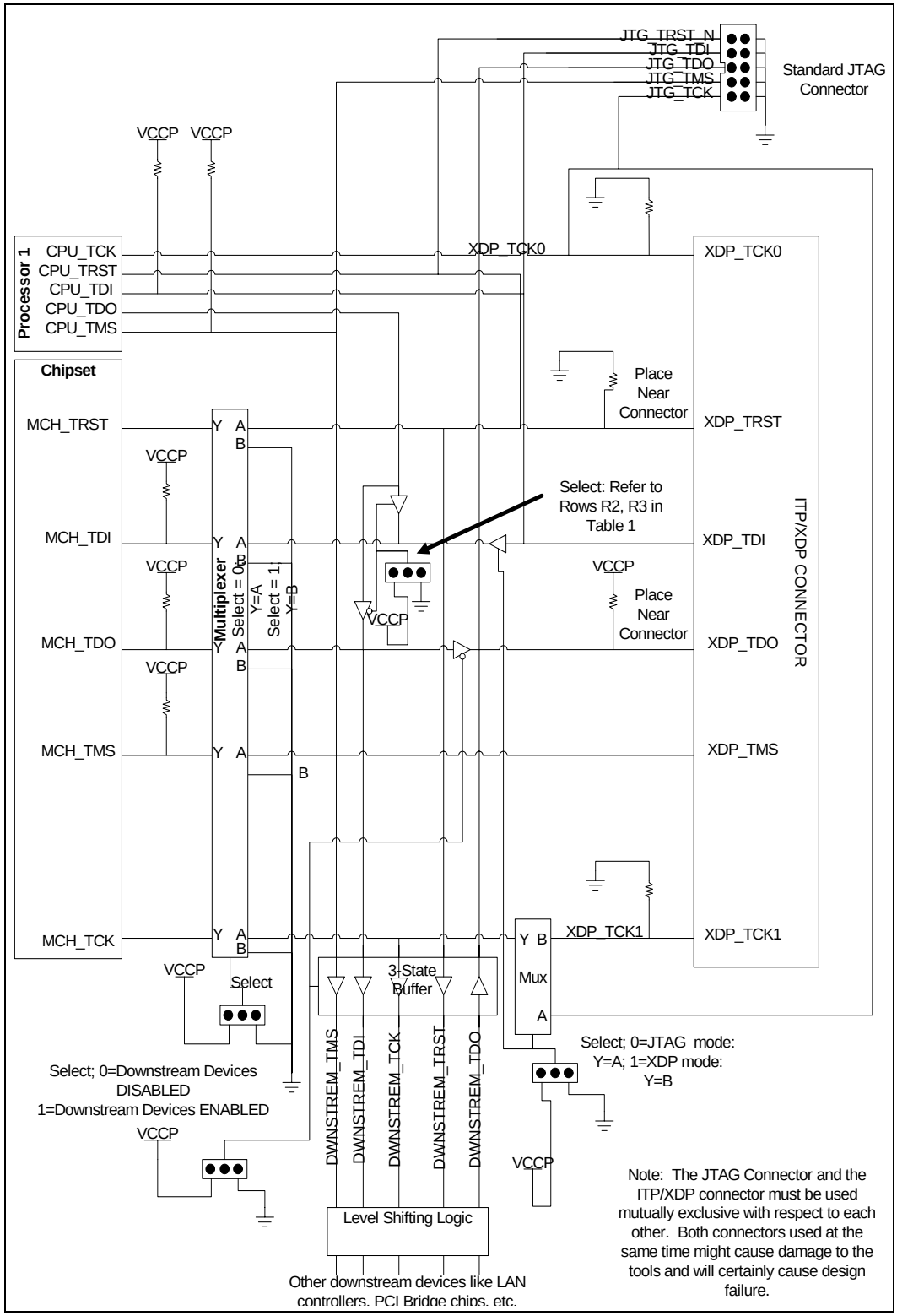
R6, R7, R8, R9, R10, and the HEADER can be replaced with a multiplexer. If voltage translation logic is required between the CPU and Chipset, a dual voltage multiplexer can be used to satisfy that requirement.

R11, R12, R13, R14, R15, and R16 can be replaced with some tri-state buffers. R11 requires an inverter on the select line as it must be stuffed in the opposite orientation as the other resistors.

R1 can be replaced with a buffer while R4 and R5 can be replaced by a multiplexer.

R2 and R3 are mutually exclusive and still need to follow the selection matrix outlined in Table 2. These resistors can be replaced by a buffer and a buffer with an inverter on the select line. R2 has the buffer with no inverter and R3 has the inverted select line driving the buffer with the current design shown.

Figure 5. Schematic for uP Bus Switch / Buffer Based TAP Controller Design



4.1.2 XDP Debug Port Population Option

It is possible to configure the design to connect to an XDP emulator to the CPU and Chipset (if supported). Selection options can be made to allow isolation of components for debug purposes allowing proper operation for the CPU portion of the design. Please keep in mind that when the TAP Controller operates as an emulator, some of the signals to the connector will run at full Front Side Bus speed and some of the TAP controller signals are a divide down of those signals. Therefore, those TAP controller signals may run at fairly fast rates with little ability to adjust that speed (most commonly 33MHz). So isolation of the scan chain reduces the risk associated with design noise, reflections, etc. by making the chain as short as possible and pretty much point to point. Two possible configurations are allowable, each requiring a CPU in the chain:

- CPU Only
- CPU with Chipset

Please keep in mind that not all chipsets support connection to the XDP connector. This will be chipset specific. Also, these TAP controller signals are only a subset of the signals required to make the XDP interface function. Refer to the XDP and Platform design guides to determine the additional signals required.

To connect the XDP connector with the CPU only; R3, R4, and R11 must be populated with 0 ohm resistors and the HEADER must be populated (XDP_TCK1 and XDP_TRST only). R1, R2, R5, R6, R7, R8, R9, R10, R12, R13, R14, R15, R16 must be depopulated.

To connect the XDP connector with the CPU and chipset (if supported) R1, R3, R4, R6, R7, R8, R9, R10, and R11 must be populated with 0 ohm resistors. R2, R5, R12, R13, R14, R15, R16 must be depopulated.

4.2 Dual Physical Processor Package Configuration

To enable a product that supports a dual physical processor package configuration (keep in mind that this may be more than two processors as more than one processor can come in a package) supporting both XDP and JTAG an additional five resistors must be added to the design. These five resistors allow the selection of either processor in the chain or both processors in the chain. Please refer to Figure 7 and Table 3.

Again, the resistors have been grouped to allow easy identification of how bus muxes and tri-state buffers are utilized to work out a board level solution. There are 12 combinations for a TAP controller JTAG interface:

- CPU 0 only
- CPU 1 only
- CPU 0 and CPU 1
- CPU 0 and the Chipset
- CPU 1 and the Chipset
- CPU 0, CPU 1, and the Chipset
- CPU 0 and Other Devices
- CPU 1 and Other Devices
- CPU 0, CPU 1, and Other Devices
- CPU 0, the Chipset, and Other Devices
- CPU 1, the Chipset, and Other Devices
- CPU 0, CPU 1, the Chipset, and Other Devices

The signal groupings are as follows:

- R1, R4, and R5 control the clocks and TDI when going between TAP and XDP mode
- R2 and R3 control the TDI to TDO chaining depending upon mode of operation
- R6,R7, R8, R9, and R10 select CPU 0, CPU 1, or CPU0&CPU1
- R11, R12, R13, R14, R15, and HEADER are for connecting the Chipset
- R16, R17, R18, R19, R20, and R21 are for connecting Other Devices

The following items must always be mutually exclusive in the circuit:

- R2 and R3
- R4 and R5
- R7 and R8
- R9 and R10
- JTAG connector and XDP connector
- The HEADER connections and R6, R7, R8, R9, and R10

In Figure 7, if the Chipset is removed from the scan chain (not all chipsets support boundary scan - some only support XOR Chaining) R11, R12, R13, R14, and R15 must be depopulated. At the same time, the HEADER (XDP_TCK1 and XDP_TRST only) must be populated to insure that signals to the chipset are defined as inputs.

If “Other Devices” are removed from the scan chain, R16 must be populated while R17, R18, R19, R20, and R21 are de-populated. Please note that not shown is the termination to the inputs of the other devices when it is removed from the chain. Similar to the chipset portion of the chain, stuffing option pull-up/down resistors and/or a header could be placed below the R17, R18, R19, R20, and R21 to allow the entire set of devices downstream to be terminated to a known value. This termination was outside the scope of this Application Note and is not further described. Also, in order to meet signal drive capabilities due to multiple devices on a single chain, some buffering must may be required. The device listed as level shifting logic will suffice for that buffering if chosen properly.

For downstream devices operating at a higher voltage level than the processor and chipset, voltage translation circuitry will be required. Transistor logic or dual voltage bus switches can be used to implement this circuit. Care must also be taken to be certain that the voltage levels between the chipset and CPU are compatible. If not, voltage translation would be required after resistors R11, R12, R13, R14, and R15.

R1, R4, and R5 will be populated one way when in TAP Controller mode of operation and populated in the other way when in XDP mode of operation. R1 and R4 will be depopulated when in TAP controller mode of operation and R5 will be populated when in TAP controller mode of operation.

R2 and R3 are implementation specific and the table should be referenced to determine how these resistors should be stuffed in a given configuration.

R6, R7, R8, R9, and R10 are used to select which CPU(s) will be in the chain. There are three possible options:

- CPU 0 only - R7 and R10 populated; R6, R8, and R9 depopulated
- CPU 1 only - R8 and R9 populated; R6, R7, and R10 depopulated
- CPU 0 and CPU 1 - R6, R8, and R10 populated; R7 and R9 depopulated

When using a bus mux/tri-state buffer model were used the circuit shown in Figure 6 or equivalent logic is required to achieve the proper selection points. In Figure 6, the buffers labeled as Rx Replacement are the resistors that the buffers would replace in

the design. For connections between these Rx Replacements please refer to Figure 7. The remainder of the bus mux/tri-state buffer for a dual processor package model looks like the circuit in Figure 5.

Figure 6. Buffer Replacement Circuit

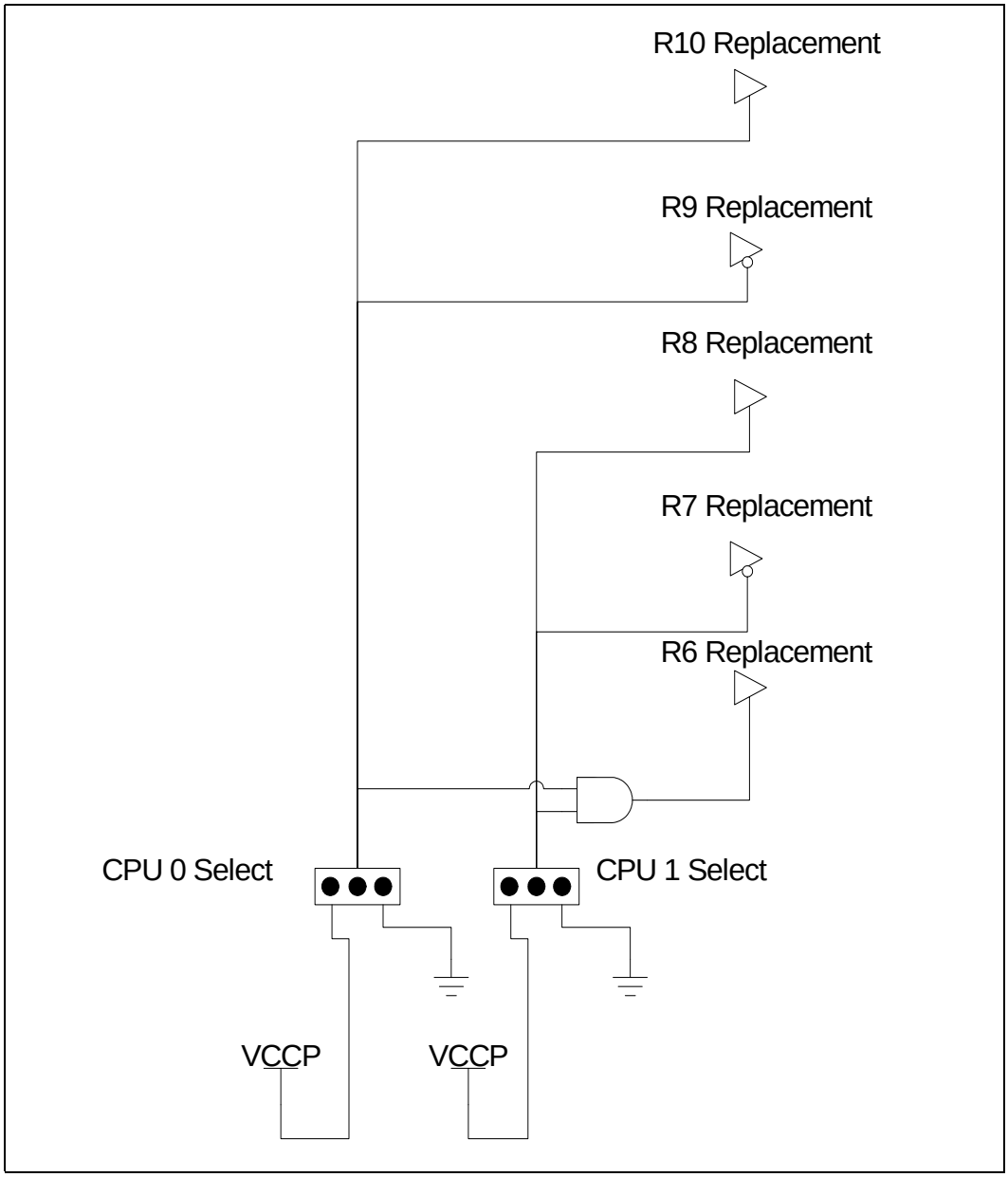


Figure 7. Schematic for DP Bus Switch/Buffer Based Tap Controller Design

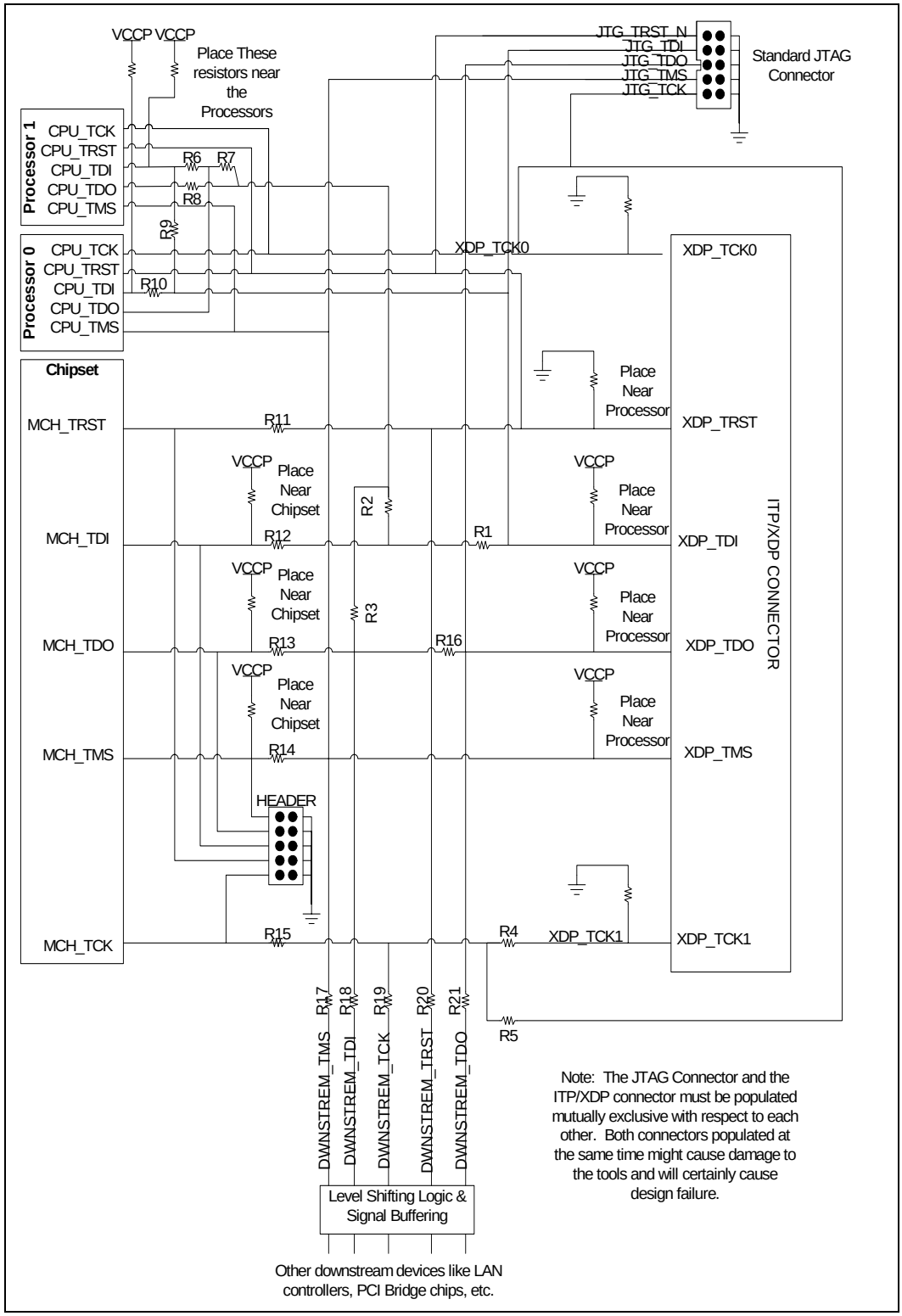


Table 3. Resistor Stuffing Options for DP TAP Controller Design

Resistors	CPU0		CPU1		CPU0&1		CPU0 & Chipset		CPU1 & Chipset		CPU1,2& Chipset		CPU0 & Other Devices		CPU1 & Other Devices		CPU1,2& Other Devices		CPU0, Chipset & Other Devices		CPU1, Chipset & Other Devices		CPU1,2, Chipset& Other Devices		XDP with CPU0		XDP with CPU1		XDP with CPU0 & 1		XDP, CPU0,1, & Chipset	
	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP	P	NP
R1		X		X		X		X		X		X		X		X		X		X		X		X	X		X		X		X	
R2		X		X		X	X		X		X			X		X		X	X		X		X			X		X		X		X
R3	X		X		X			X		X		X	X		X		X			X		X		X	X		X		X		X	
R4		X		X		X		X		X		X		X		X		X		X		X		X	X		X		X		X	
R5	X		X		X		X		X		X		X		X		X		X		X		X			X		X		X		X
R6		X		X	X			X		X	X			X		X	X			X		X	X			X		X	X		X	
R7	X			X		X	X			X		X	X			X		X	X			X		X	X			X		X		X
R8		X	X		X			X	X		X			X	X		X			X	X		X			X	X		X		X	
R9		X	X			X		X	X			X		X	X			X		X	X			X		X	X			X		X
R10	X			X	X		X			X	X		X			X	X		X			X	X		X			X	X		X	
R11		X		X		X	X		X		X			X		X		X	X		X		X			X		X		X	X	
R12		X		X		X	X		X		X			X		X		X	X		X		X			X		X		X	X	
R13		X		X		X	X		X		X			X		X		X	X		X		X			X		X		X	X	
R14		X		X		X	X		X		X			X		X		X	X		X		X			X		X		X	X	
R15		X		X		X	X		X		X			X		X		X	X		X		X			X		X		X	X	
R16	X		X		X		X		X		X			X		X		X		X		X		X	X		X		X		X	
R17		X		X		X		X		X		X	X		X		X		X		X		X			X		X		X		X
R18		X		X		X		X		X		X	X		X		X		X		X		X			X		X		X		X
R19		X		X		X		X		X		X	X		X		X		X		X		X			X		X		X		X
R20		X		X		X		X		X		X	X		X		X		X		X		X			X		X		X		X
R21		X		X		X		X		X		X	X		X		X		X		X		X			X		X		X		X

Notes:

1. All resistors listed above are 0 ohm resistors
2. P in the table above denotes the resistor must be populated
3. NP in the table above denotes the resistor must not be populated
4. When resistors R11 through R15 are populated as denoted by an X under the P column, the header must be left open (or unconnected).
5. When resistors R11 through R15 are not populated as denoted by an X under the NP column, the header must be populated (or connected).

The Dual Package design can replace the resistors with bus muxes and switches but will not be shown in this Application Note. However, the resistor groupings are shown in Table 3 on page 20.

Please remember that all design rules called out in the Design Guides must be adhered to. It is critical with all of these options that proper simulation be done in order to guarantee a working design. Please take care to place the resistors/connectors during layout in locations to minimize signal integrity issues due stub effects. Also be certain to place proper end line termination given a particular configuration. The ramification of not doing this will be a need to slow down the interface which will increase test time or the remote potential does exist that the design will not work.

5.0 XOR Chain Considerations

Each chipset datasheet specifies the XOR chains that exist in the silicon. Be sure to consult the datasheet to determine strapping options that enable each chipset. Also be sure to understand high volume manufacturing test methodology up front and design the system with the XOR chain in mind.

One of the most important thing to understand is that some chips may support Boundary Scan only and some others may support XOR only. In this event, it is important to design in a test methodology to allow sufficient test coverage.

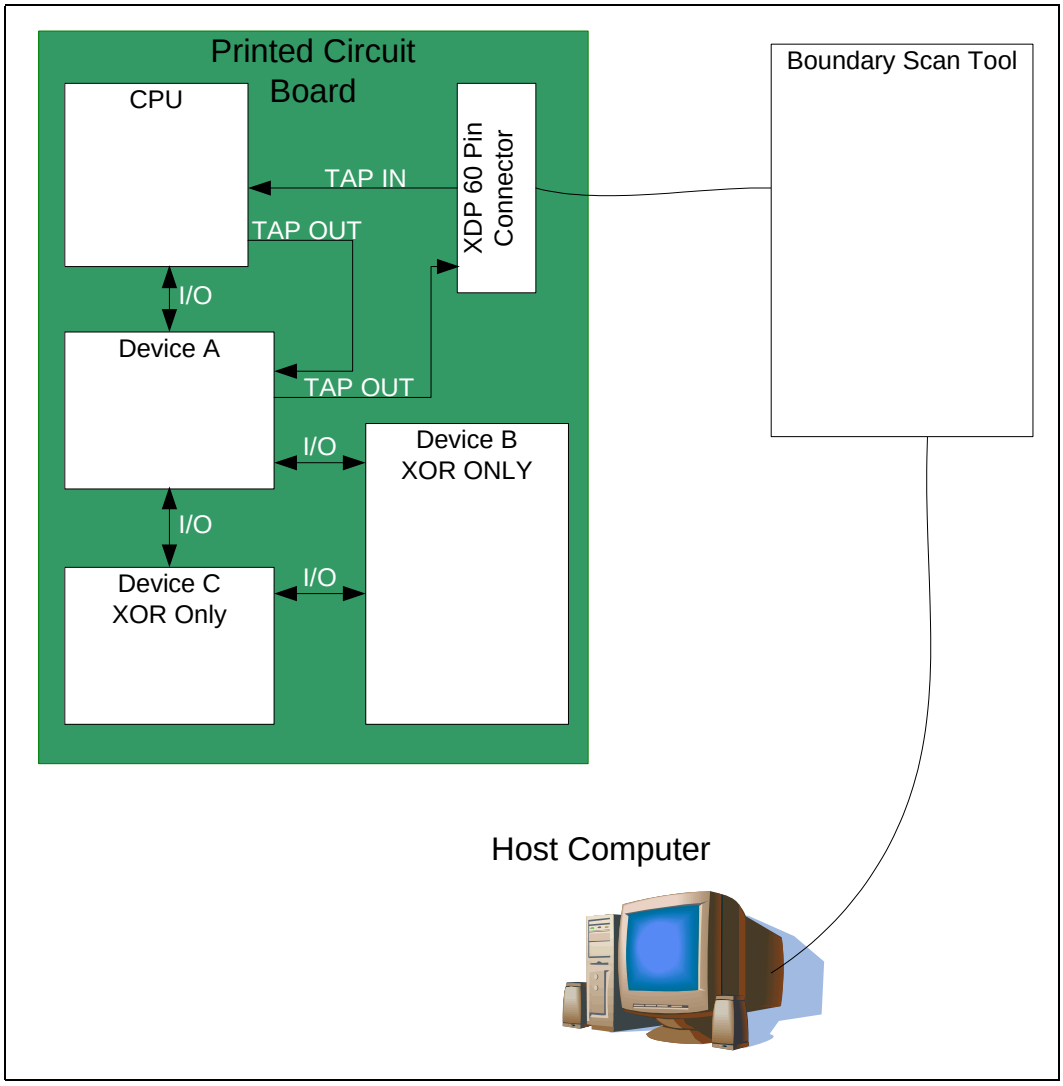
Figure 8 shows a mixed XOR and JTAG/Boundary Scan environment. Notice that the I/O between the CPU and Device A can easily implement boundary scan for board level interconnect testing and device isolation. However, the connections between Device A and Device B and Device A and Device C are not as simple.

Designs should consider how they achieve interconnect testing into this mixed environment. For instance, the design might place Device A in High-Z mode using the TAP controller and use the XOR chains on Device A and Device B to implement the board level interconnect testing. This is recommended when in a mixed environment.

The opposite methodology is also possible. However, care must be taken as one I/O pin will be an output from the XOR chain at all times. I/O not in the chain will be active at a given logic state which can create the potential for I/O driver contention on the board.

In the case of two XOR chains connected together between two devices, such as with Device B and Device C below, components are typically designed to allow XOR testing to be easily and seamlessly implemented. This scenario is the case between a number of our MCH and ICH components.

Figure 8. Mixed XOR and Boundary Scan Environment



6.0 Voltage and Ground Rail Testing

This section is mentioned because it is normally not covered via the afore mentioned methodologies and different considerations must be taken into account.

It is difficult to isolate an individual component to the solder joint failure on a production level board without creating individual power floods (mini-planes) for each component. This methodology is not required nor should it be necessary. The main errors that would need to be caught in a production environment are:

- power/ground shorts before the board components are placed
- power/ground shorts after board components are placed and before the board is powered for the first time

If there are power/ground shorts before the board components are placed, this points to a PCB which was defective.

If there are power/ground shorts after board components are placed but not before the components were placed, a soldering error likely occurred and rework will be required.

Isolation of the faulty area is achieved by removing one component at a time and retesting until the area of failure is found. If the failure is persistent and repeatable then the solder flow process should be tuned.

In order to achieve power and ground testing, probe points must be enabled to allow a tester to check for shorts between the power planes as well as power to ground planes. This can be achieved by making sure test point pads are brought to the outer layers of the board from each plane as well as the ground planes. Each flood must be brought to a test point pad as well.

Designers should contact their ODM or CM to determine their preferred method of testing. If their test methodology is similar, make certain that the test point pads placed at the correct locations and are of the appropriate size for their tester.

7.0 Summary

It is important to consider the test mechanisms for high volume manufacturing for each component in a design. This Application Note describes a method for merging these functions to allow a design to achieve all intended purposes. However, it is recommended that each design consult the ODM or CM for that design to understand their preferred implementations

One major consideration is when the methodologies differ between two chips. For instance, chipsets that typically only support XOR testing methodology and the downstream devices attached to the chipset (such as a GIGE PHY) may only support JTAG boundary scan testing methodology. In the event that this is the case, care must be taken to make certain that one of these methodologies can be used in a high volume manufacturing environment. The designer may decide that it is easier to place the attached part which only supports JTAG into a HIGH-Z state so that it does not interfere with the inputs to the XOR chain. Another approach is to enable the XOR chain and then drive the XOR chain input pins with pattern produced via the JTAG boundary scan logic of the attached devices.

Additional items to consider are:

- Allow flexibility to make certain that an XDP and Boundary Scan enabled design has the ability to isolate the CPU to allowing the XDP port to run at proper speeds.
- Make sure scan chains do not get too long without proper signal buffering when large scan chains are required.
- Introduce multiple scan chains make the design easier and doesn't increase manufacturing time too much.
- Assure appropriate voltage levels are driven to the interfaces via level shifters.
- Provide proper signal termination as defined by the design guides and the IEEE1149 specifications.
- Omit any portion of this design that is not required to save on cost and complexity.