



# **PCI Express™ Technical Training Day**

**Milpitas, CA**

**3/22/2004**



Copyright© 2004 by PCI-SIG  
All Rights Reserved  
PCI-SIG Confidential

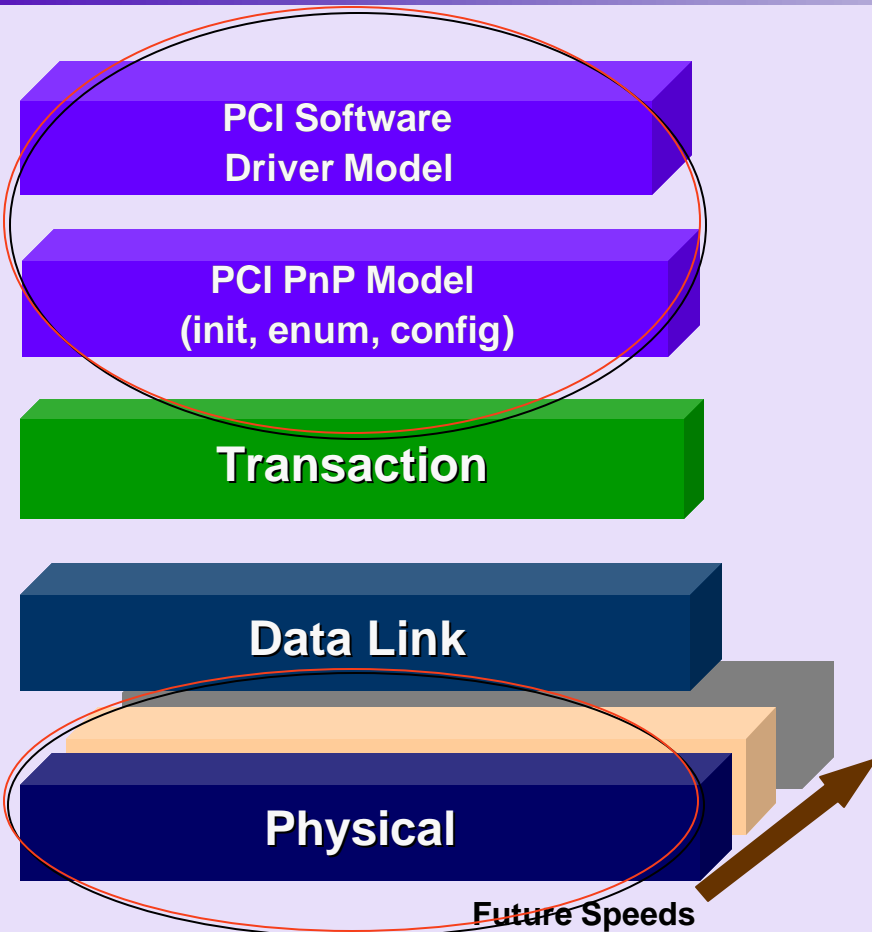
# PCIe™ Training Day Agenda

9:00	PCIe Overview	<i>Neshati</i>
9:45	PCIe Physical Layer	<i>Schoenborn</i>
12:00	LUNCH	
1:00	PCIe Configuration & Software	<i>Cowan</i>
2:45	BREAK	
3:00	PCIe Compliance Tools	<i>Choate/Froelich</i>
5:00	Q&A	
5:15	BREAK	
5:30	Compliance Workshop Orientation	<i>Neal/Kelley, et al</i>

# Agenda

9:00	<b>PCIe Overview</b>	<i>Neshati</i>
9:45	PCIe Physical Layer	<i>Schoenborn</i>
12:00	LUNCH	
1:00	PCIe Configuration & Software	<i>Cowan</i>
2:45	BREAK	
3:00	PCIe Compliance Tools	<i>Choate/Froelich</i>
5:00	Q&A	
5:15	BREAK	
5:30	Compliance Workshop Orientation	<i>Neal/Kelley, et al</i>

# PCIe Layered Architecture



- **Advanced Capabilities:**
  - ✓ RAS: CRC data integrity, hot-plug, error logging/reporting
  - ✓ Power management
  - ✓ QoS and isochronous support
  - ✓ Extended configuration
  - ✓ New form factors
- **PCI Compatibility:**
  - ✓ Configuration and PCI driver model
- **Protocol:**
  - ✓ Load/Store architecture
  - ✓ Fully packetized split-transactions
- **Physical Interface:**
  - ✓ Point-to-point, dual-simplex (full-duplex)
  - ✓ Low-voltage differential signaling
  - ✓ Embedded clocking
  - ✓ Scaleable frequency
  - ✓ Scalable width
  - ✓ Supports connectors and cables

**PCIe Layered Architecture Enables Modularity, Reuse & Scalability**



# PCIe Technology Features

- |   |   |
|---|---|
| <ul style="list-style-type: none"> <li>• PCI Compatibility</li> <li>• VC Mechanism</li> <li>• High Bandwidth</li> </ul>     | <ul style="list-style-type: none"> <li>• Smooth migration, SW re-use, simple validation</li> <li>• QoS &amp; isochrony</li> <li>• Peak traffic loads, overprovisioning for QoS</li> </ul>   |
| <ul style="list-style-type: none"> <li>• Flow Control</li> <li>• Reliable Link Layer</li> <li>• E-CRC</li> </ul>            | <ul style="list-style-type: none"> <li>• Buffer size flexibility, cost flexibility</li> <li>• No dropped packets, simplified SW, high availability</li> <li>• Maintain communication for HA or diagnostics</li> <li>• End-to-end reliable transport for service availability</li> </ul>                   |
| <ul style="list-style-type: none"> <li>• Error Reporting</li> <li>• Power Management</li> <li>• Single PHY Layer</li> </ul> | <ul style="list-style-type: none"> <li>• System management, serviceability, availability</li> <li>• Optimize density, support cold spares</li> <li>• Reduced emissions in EMI-sensitive environments</li> <li>• Interoperability, no multi-mode PHY, no SKU proliferation, bounded investments</li> </ul> |
| <ul style="list-style-type: none"> <li>• High Speed Serial</li> </ul>   | <ul style="list-style-type: none"> <li>• Reduced cost, pin count, PCB layers &amp; area</li> </ul>  |

# PCIe Technology Benefits

## **Better Visualization**

- Beyond AGP8x
- Multimedia/Isochrony

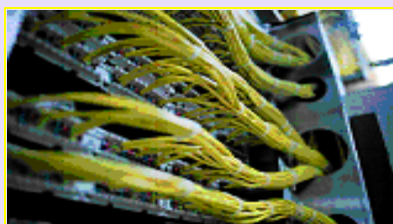


## **Innovative Form Factors**

- Client Modules (e.g. ExpressCard\*)
- Server Modules
- ATCA

## **Improved Reliability**

- Hot Plug/Hot Swap
- Error Logging/Reporting
- E-CRC



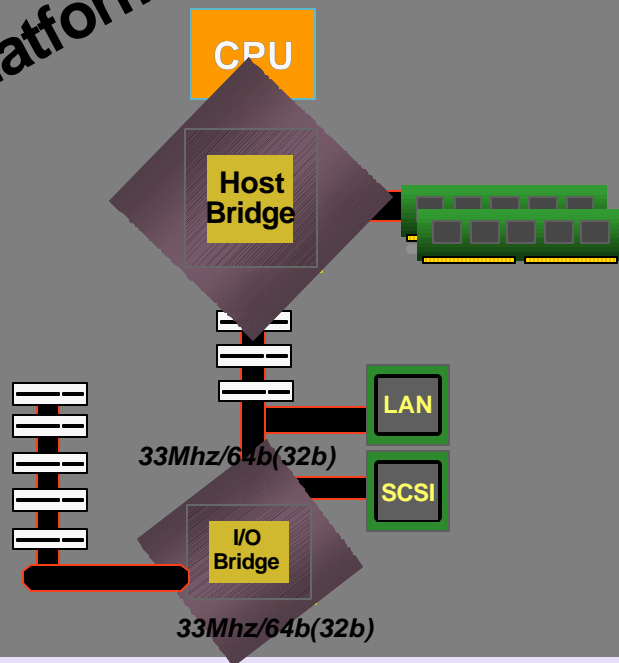
## **Enhanced Connectivity**

- GbE/10 GbE
- IBA
- FC

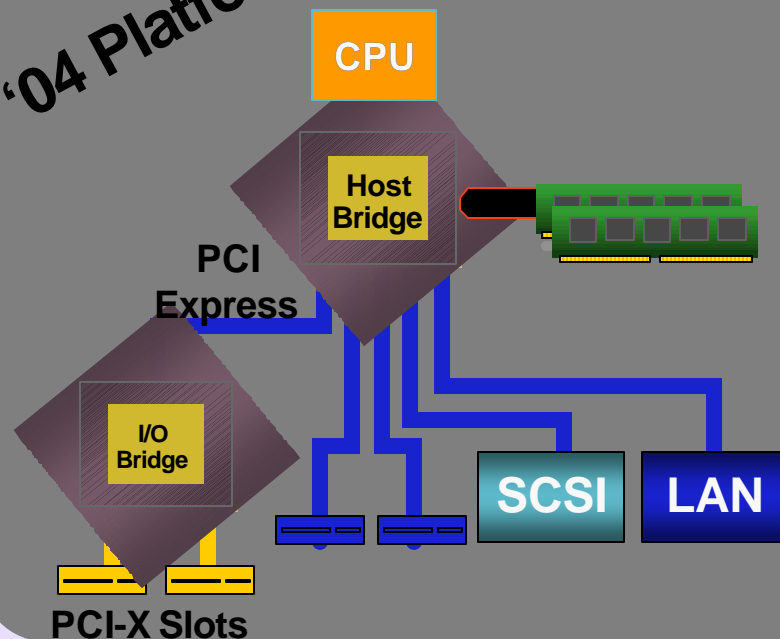
**PCIe Provides Performance Scalability Into Next Decade**

# PCIe In Servers

'03 Platform

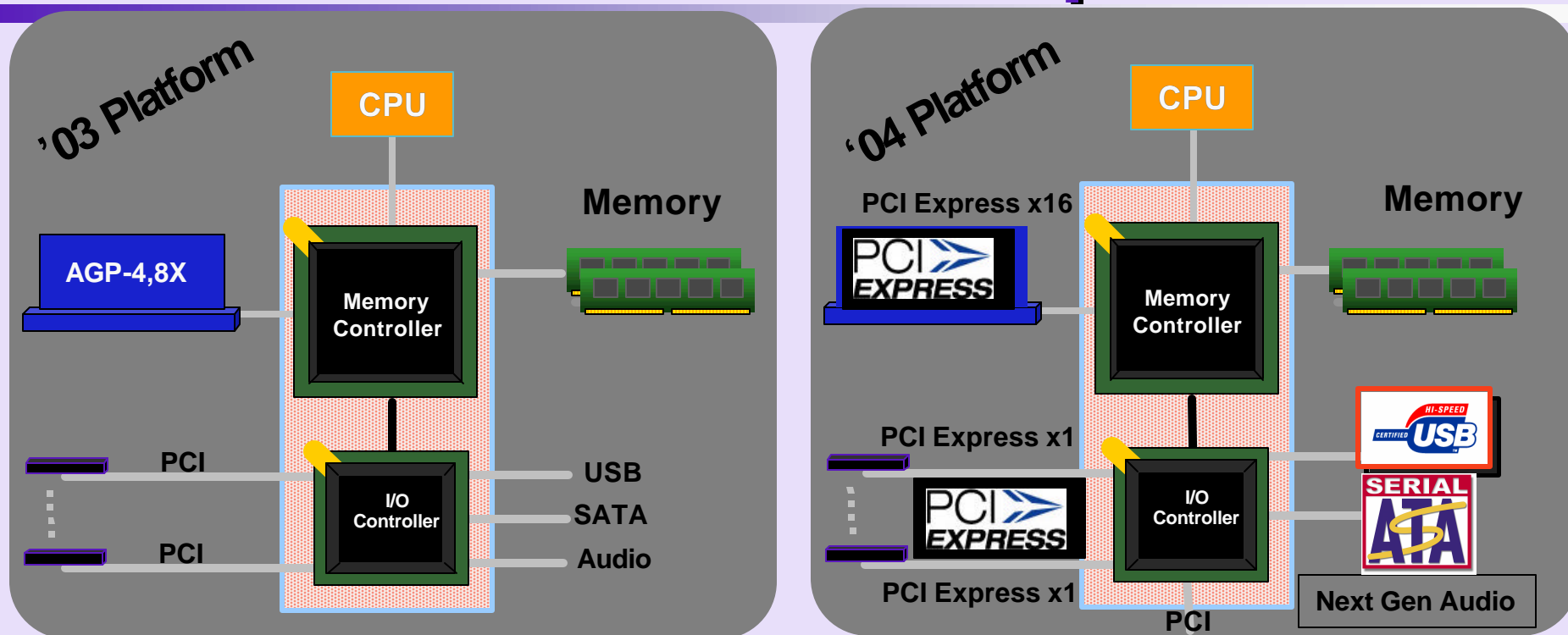


'04 Platform



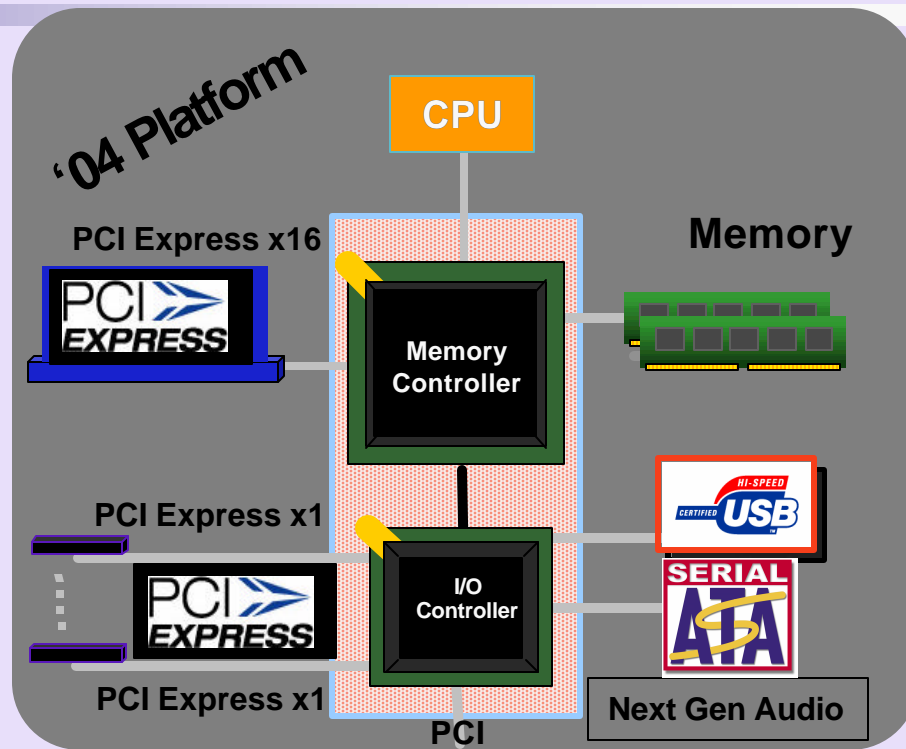
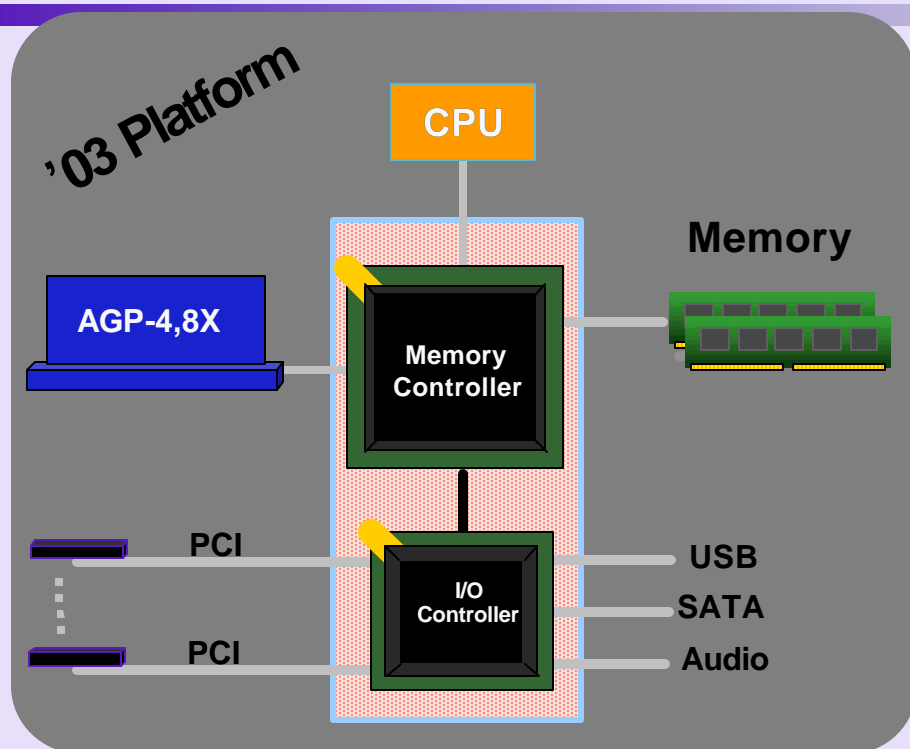
- Adapters can connect directly to the memory controller (removes extra bridge)
  - ✓ Lower latency and lower cost
  - ✓ Fewer pins and less board space
- Bandwidth to support 10Gb technologies
  - ✓ “Future proofs” 2004 servers
- Leverages the volume economics of desktop, mobile, and communications

# PCIe In Desktop



- Greater bandwidth and performance
  - ✓ PCI Express x16 next generation graphics for 8GB/s
  - ✓ PCI Express x1 for greater I/O performance for GbE, 1394, etc for 500MB/s
- New usage models
  - ✓ Isochronous support for streaming media for TV tuners, graphics, camera
  - ✓ Native hot-plug support for new form factors and modules (e.g. ExpressCard\*)

# PCIe In Notebooks



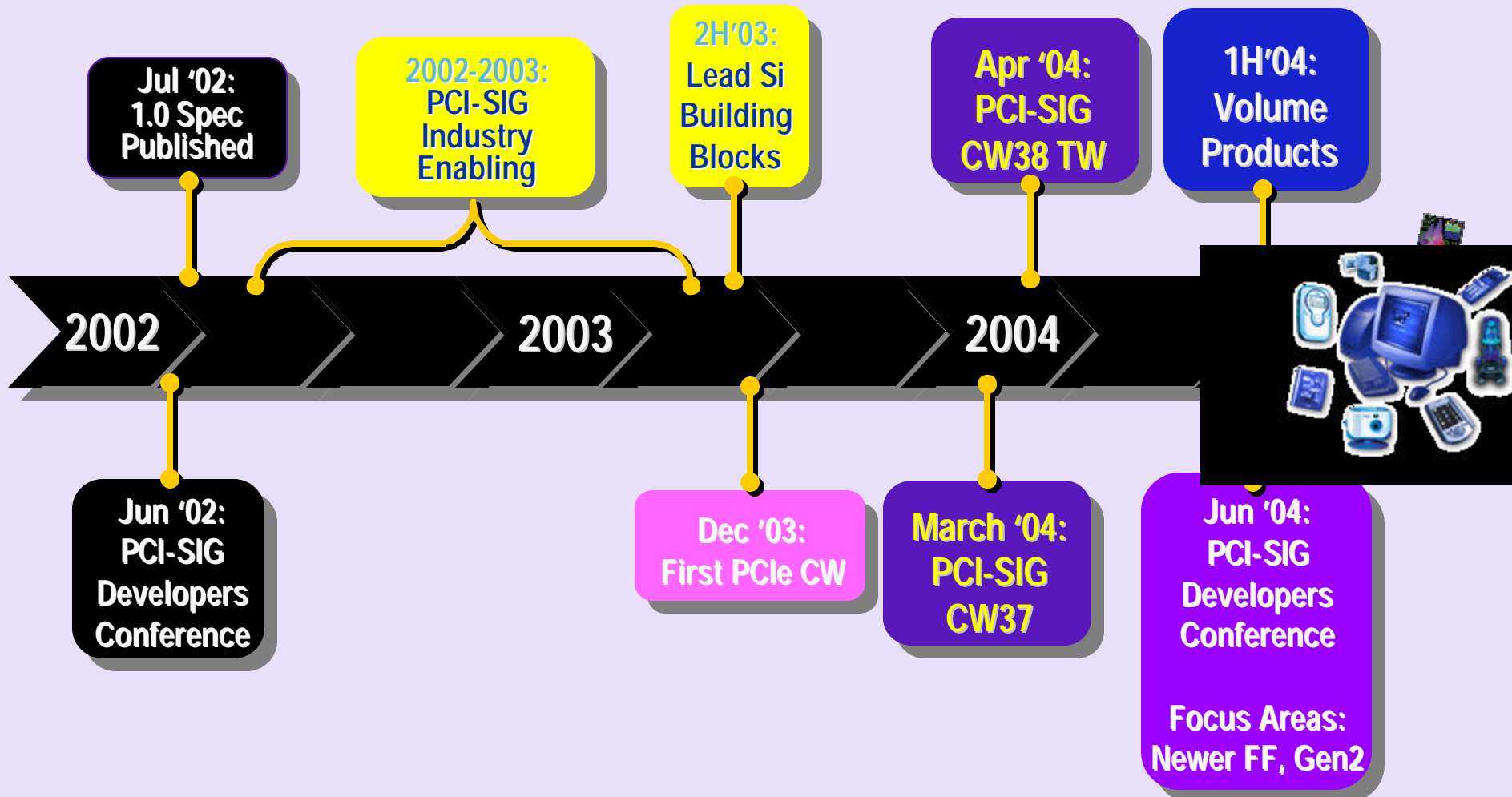
- ExpressCard\* provides value for OEM's, IT, and end users
  - ✓ User upgradability and flexibility with PC Card replacement
  - ✓ Expandability protects investment in notebooks
- Next generation docking solution
- Wireless FF

# PCIe Architecture Adoption

Application	Prior Technology	PCIe Configuration	Market Segment
Chipsets	Heterogeneous I/O	Unified I/O	Desktop, Mobile, Enterprise
Graphics	AGP 8X	PCI Express x16	Desktop, Mobile
General Purpose I/O	PCI	PCI Express x1	Desktop, Mobile
Gigabit LAN	PCI or LOM	PCI Express x1 or LOM	Desktop, Mobile
Client Plug-n-Play	PC Card	ExpressCard*	Desktop, Mobile
Internal Mobile	Mini PCI	PCI Express Mini Card	Mobile
Graphics	AGP 8X Pro	PCI Express x16	Workstations
Bridge	PCI-to-PCI-X	PCI Express-to-PCI-X	Enterprise
General Purpose I/O	PCI/PCI-X	PCI Express x4, x8	Enterprise
Server Module	n/a	PCI Express Server I/O Module	Enterprise
Communication Fabrics	Proprietary or Ethernet	Advance Switching or Ethernet	Communications
Communication Control & Host Based Backplane	PCI	PCI Express	Communications
Communication Chip-to-Chip data	PCI/SPI/CSIX/Other	PCI Express or Advanced Switching	Communications

**Applications Transitioning to PCIe Across the Industry**

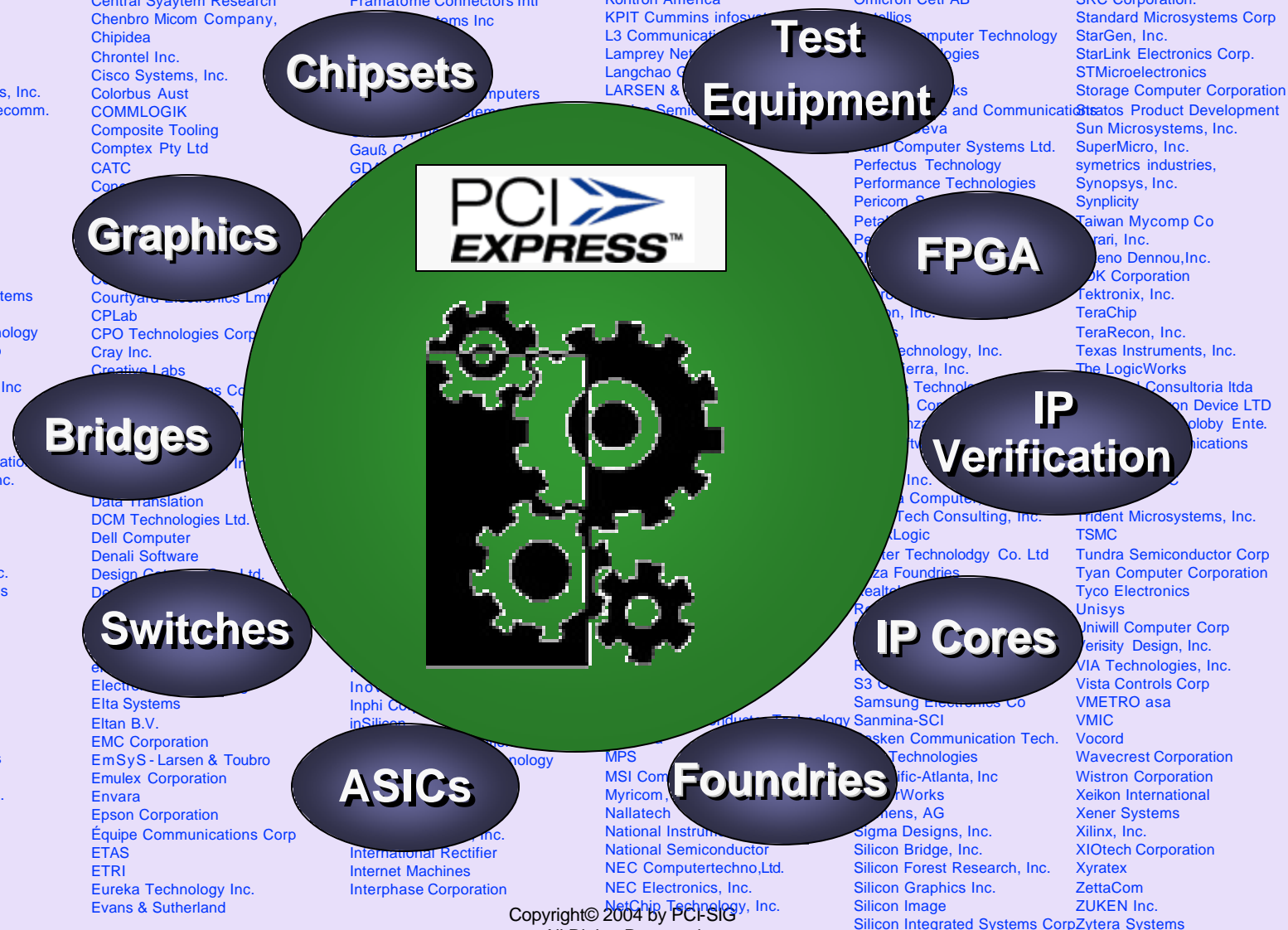
# PCIe Technology Progress



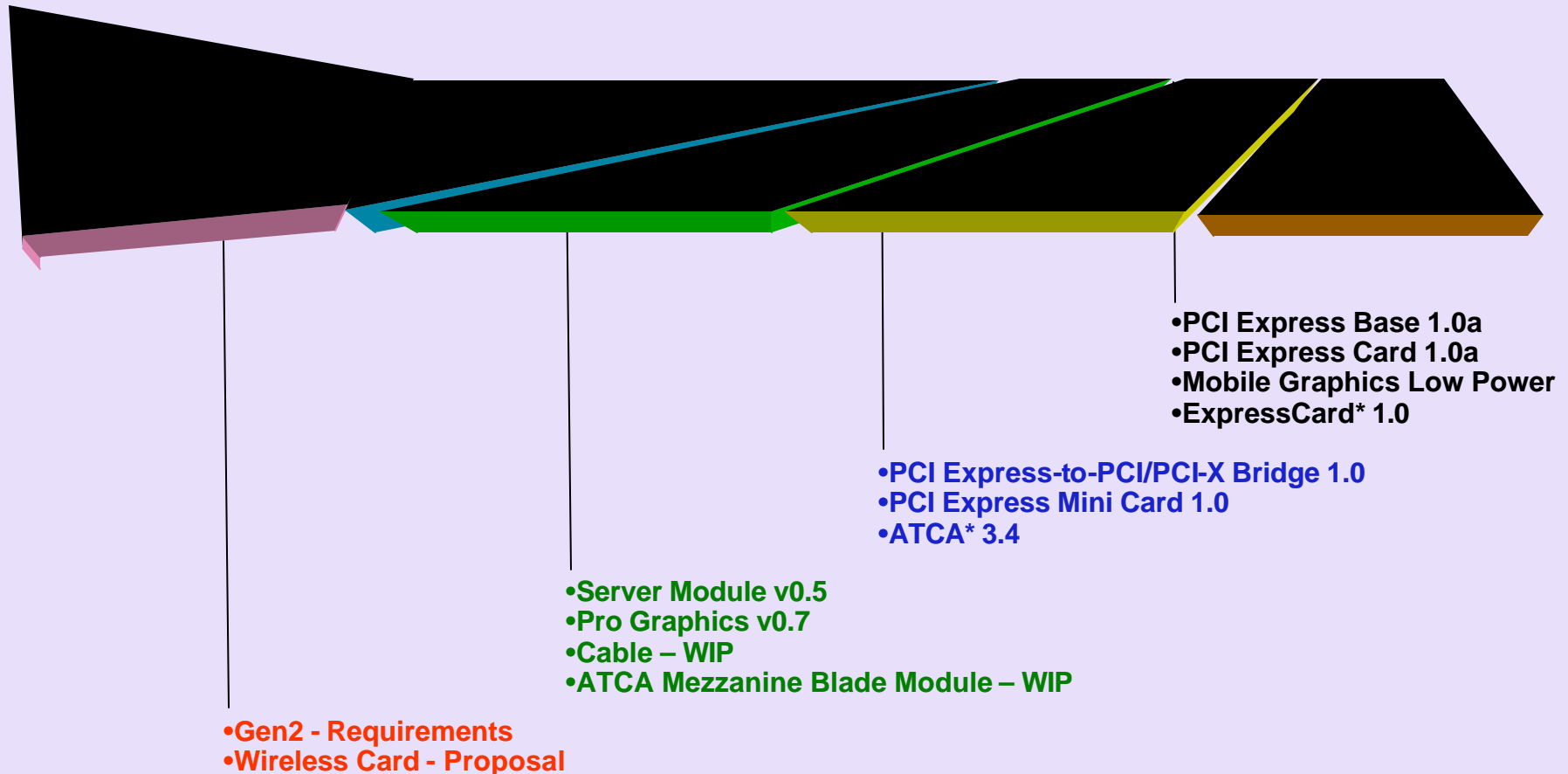
*PCI-SIG World-Class Industry Enabling Delivers TTM Value To Members*



# PCIe Ecosystem



# PCIe Specification Roadmap



# Summary

- PCI Express Architecture
  - ✓ Base & Card 1.0a compliance checklists, test considerations and test software available from PCI-SIG
  - ✓ Compliance test boards available from PCI-SIG
  - ✓ PCIe 1.1 specification (consolidated errata and ECNs) on the way
  - ✓ ExpressCard\* 1.0 specification and design checklist available from PCMCIA
  - ✓ Other FF enhancements/extensions in progress
    - Server I/O module, Pro Graphics, Cable, etc
- Taiwan Compliance Workshop
  - ✓ Planned for April 04 in Taipei
  - ✓ More workshops being planned in US & Taiwan
- PCI-SIG Developers Conference: June 04, SJCC

# Agenda

9:00	PCle Overview	<i>Neshati</i>
9:45	<b>PCle Physical Layer</b>	<i>Schoenborn</i>
12:00	LUNCH	
1:00	PCle Configuration & Software	<i>Cowan</i>
2:45	BREAK	
3:00	PCle Compliance Tools	<i>Choate/Froelich</i>
5:00	Q&A	
5:15	BREAK	
5:30	Compliance Workshop Orientation	<i>Neal/Kelley, et al</i>

# Objectives

- Common goals
  - ✓ Build on PCI's success
  - ✓ Interoperable PCI Express products
  - ✓ High value, low cost
  
- Spec balance
  - ✓ Under-specifying leads to products not interoperating
  - ✓ Over-specifying leads to higher product costs without additional value

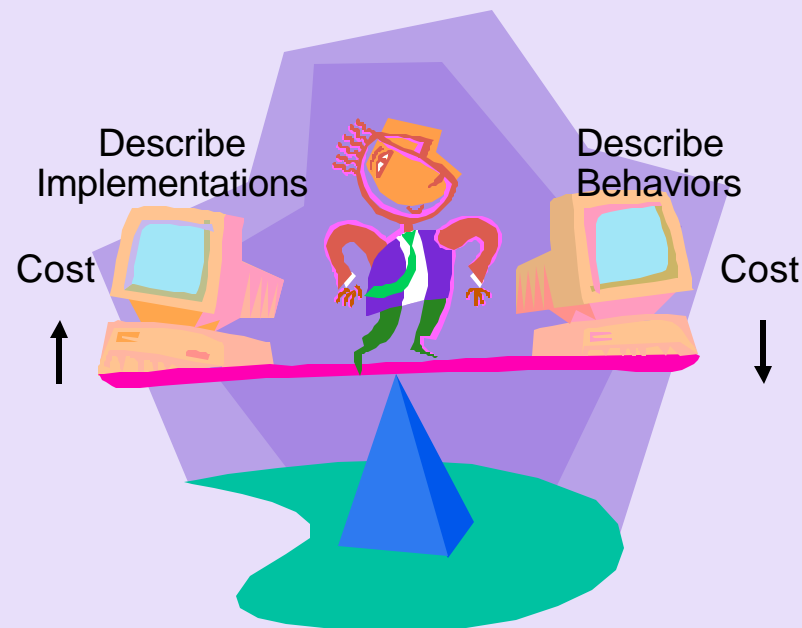
**Work with the PCI-SIG to meet this balance**

# Topics

- PCI Express Phy specification
- Phy design basics
- Reference Clock design
- Board Design
- Jitter
- Phy validation

# PCI Express Specifications

- Unambiguously describe how PCI Express interfaces behave
  - ✓ Interoperable products
  - ✓ Vibrant competitive market
    - Encourage multiple implementations
  
- State what must be done without stating how to do it
  - ✓ Specify the interface behavior
  - ✗ Do not specify how to implement
  - ✗ Do not specify the system it operates in

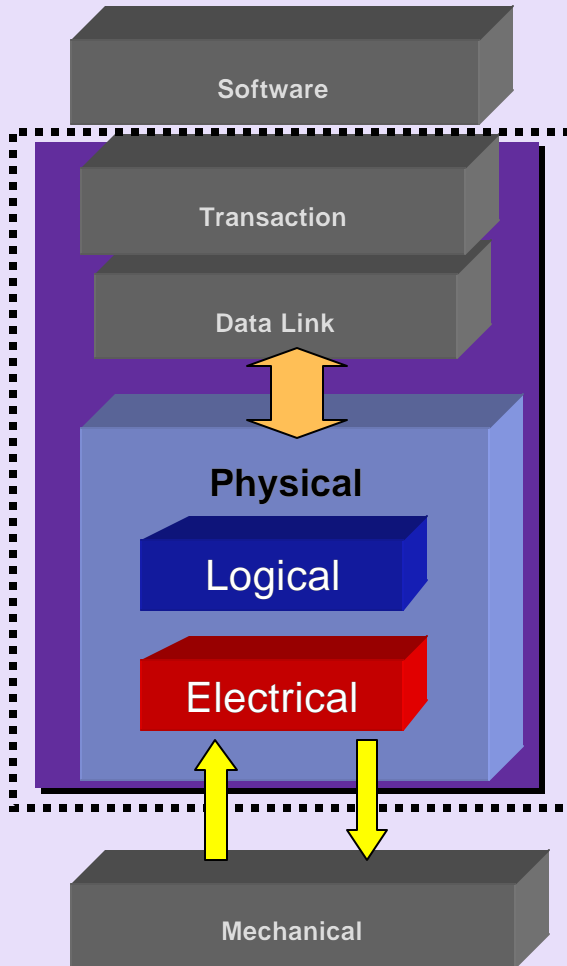




# PCI Express Physical Layer

- PCI – PHY is digital in nature
  - ✓ Parallel multi-drop bus
  - ✓ Bits, a clock, setup and hold times...
    - We essentially ignored jitter
- PCI Express – PHY is analog in nature
  - ✓ Based on serial technology
  - ✓ Techniques developed by the communication industry
- A transition for microprocessor based systems
  - ✓ Microwave theory/physics challenges dominate
  - ✓ Two PLLs communicating directly

# PHY Layer Design Basics



## ■ Logical Functions

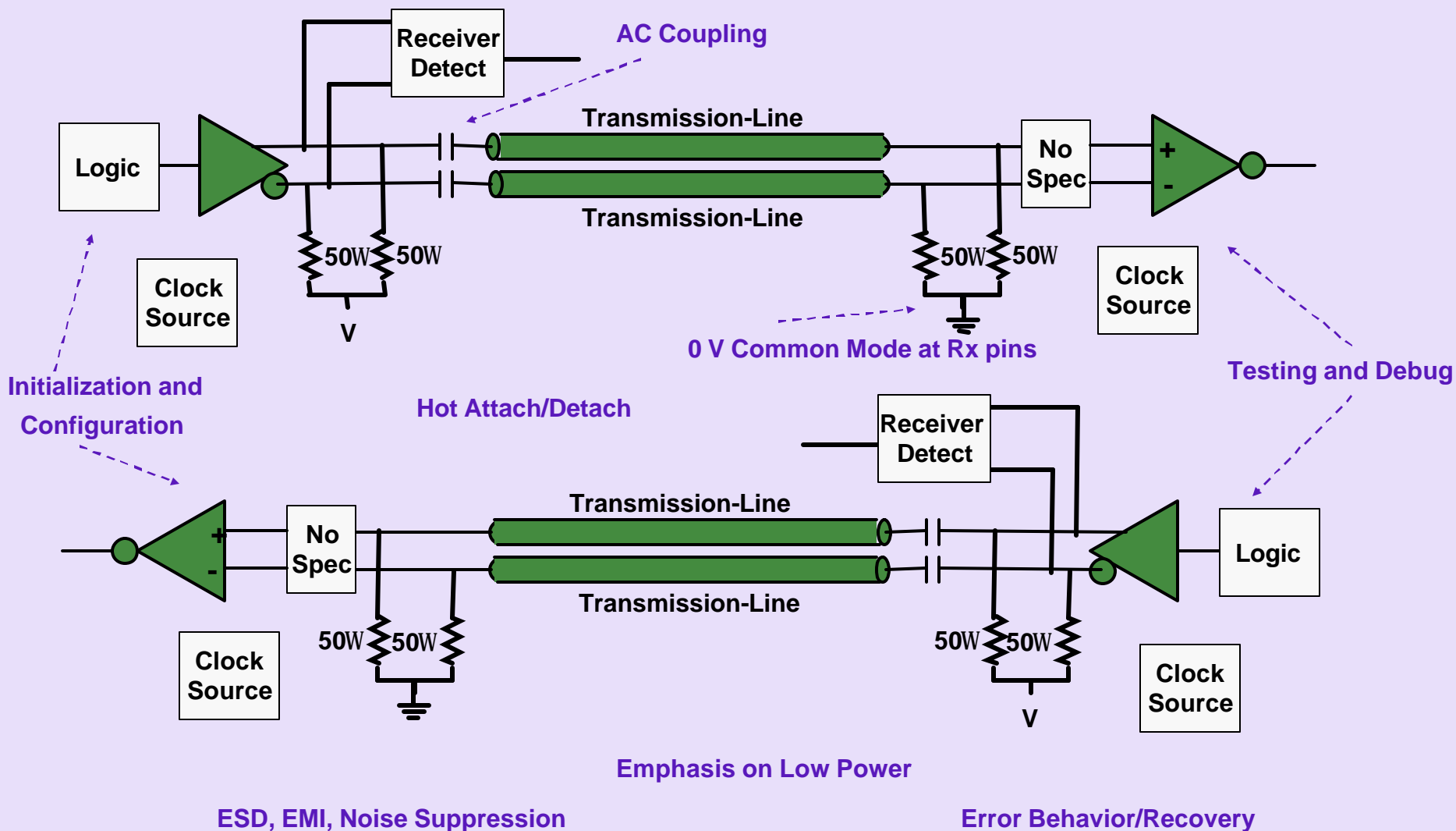
- ✓ Encoding/decoding/scrambling
- ✓ Reset, initialization, De-skew
- ✓ Built in Test Modes
- ✓ Configuration:
  - Speed, Link width, Lane mapping, Polarity
- ✓ Link Power management

## ■ Electrical Functions

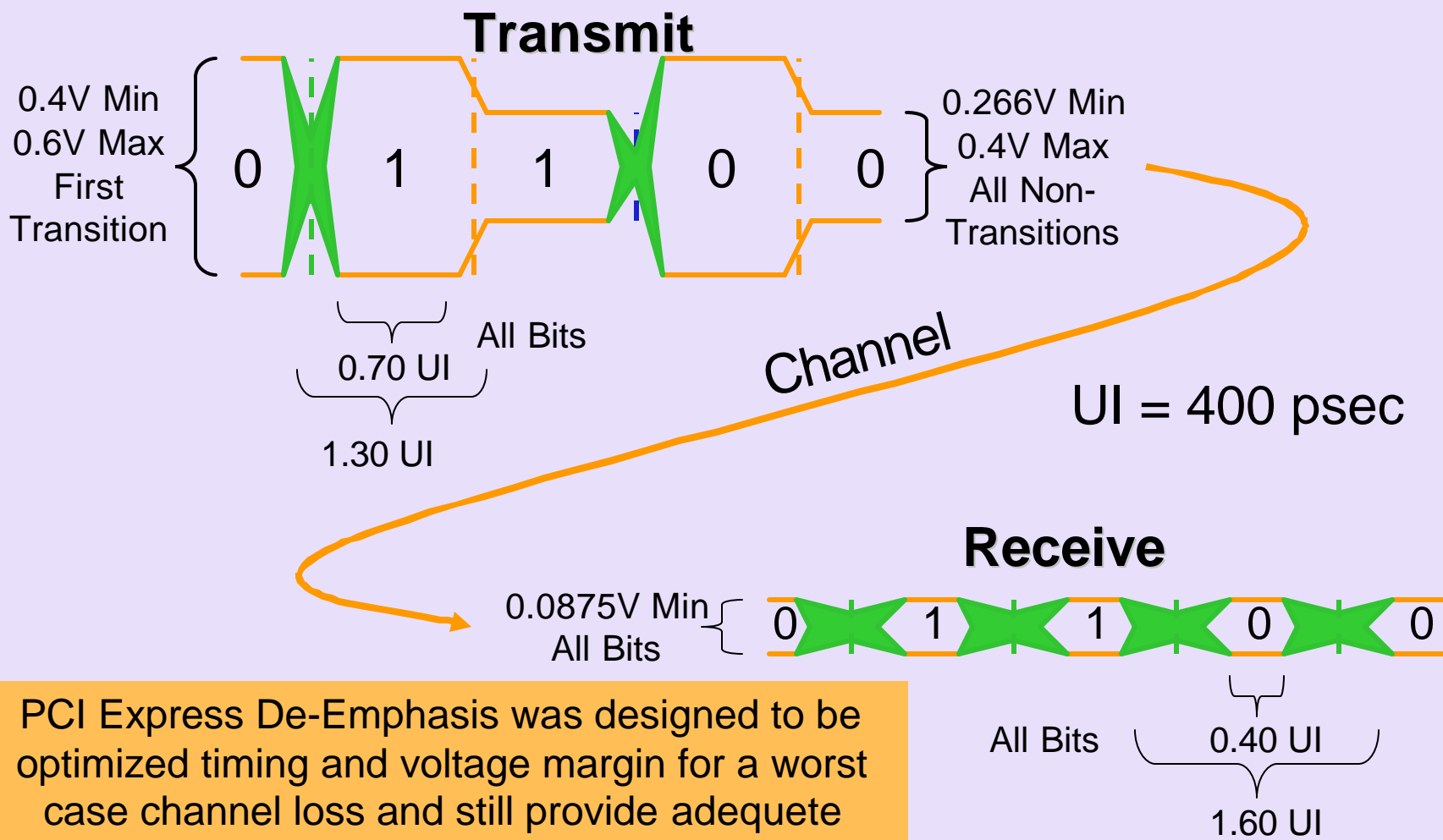
- ✓ Transmitter/Receiver
- ✓ Clocks/PLLs, Clock/Data Recovery

**PHY layer upgrades do not affect upper layers**

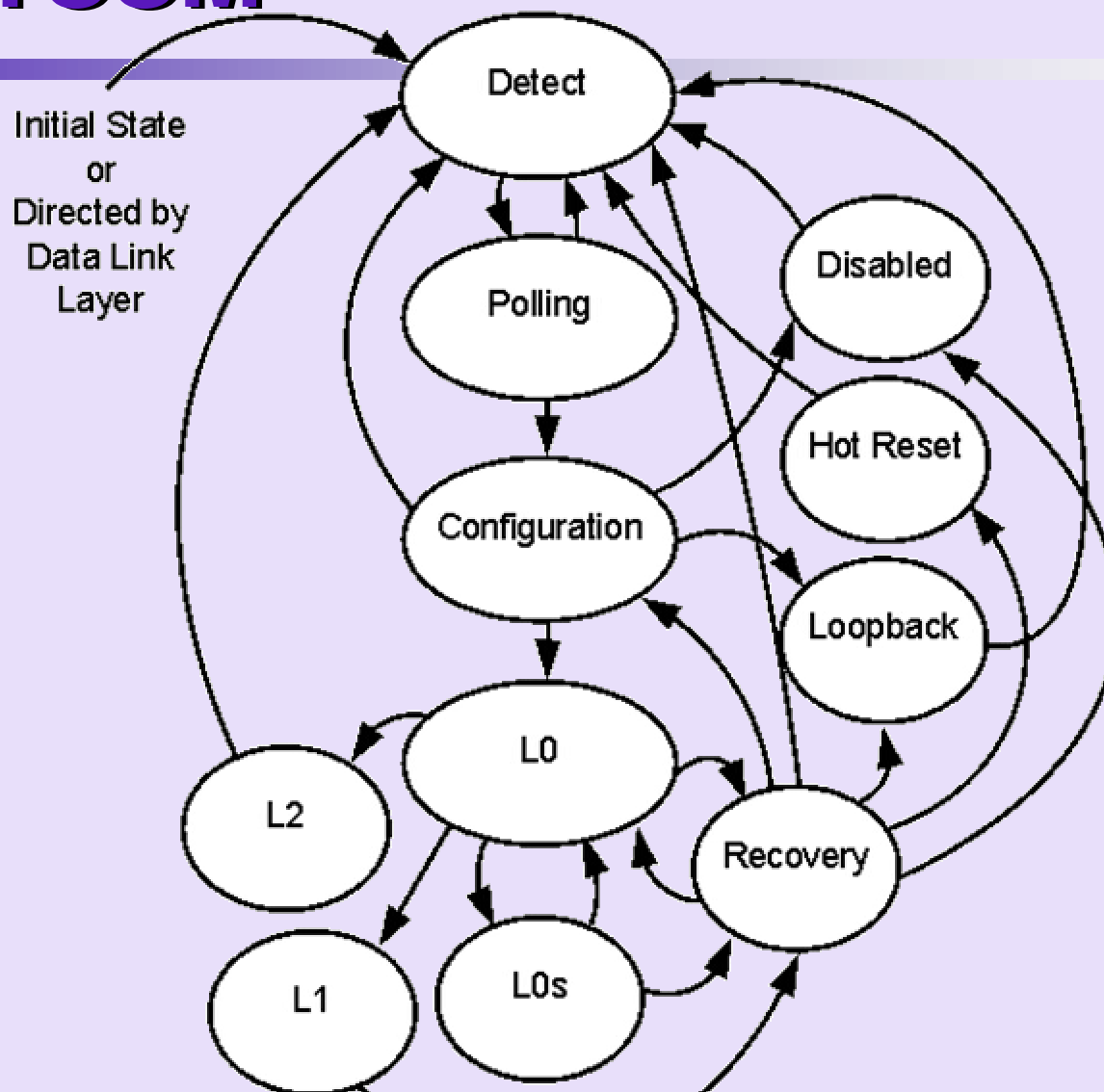
# Pictorial Summary



# Electrical Specs



PCI Express De-Emphasis was designed to be optimized timing and voltage margin for a worst case channel loss and still provide adequate margin for the best case channel loss



\* Link Training and  
Status State Machine

# Clocking Options

- All lanes within a port must transmit data using one frequency
- The ports at each end of a link may transmit data at slightly different frequencies
  - ✓ Tolerance =  $\pm 300$  ppm each

If a spread spectrum clock (SSC) is used to modulate the data rate then both ports in a link must use the same modulated clock source

Reference Clock

If SSC  
then one  
reference  
clock

General  
Purpose  
Reference  
Clock

Electrical

TX/RX

TX/RX

RX/TX

RX/TX

Electrical

# Physical Layer Status

- Logical and electrical specifications are mature
  - ✓ We are continuing to answer questions but no comments

- Compared to where we were 1 year ago

We should feel proud of our success

- Are we done?

Not yet, but we are making progress

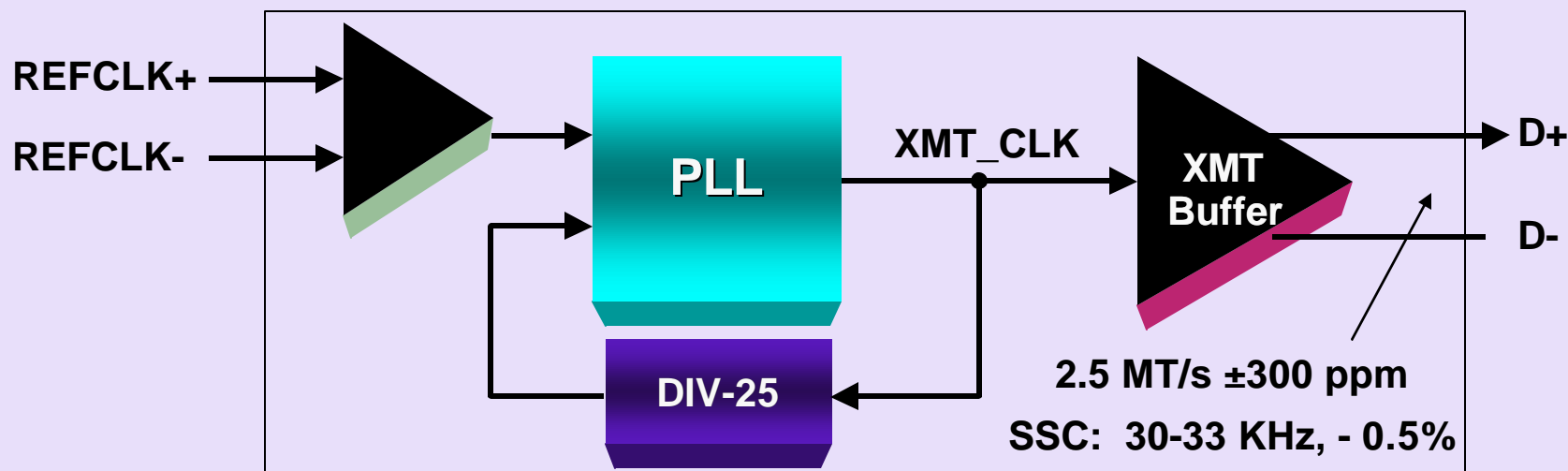


# Our Physical Layer Challenges

- Increase our industry knowledge base
  - ✓ What tradeoffs should we make?
    - Tx, Rx implementations
    - PLLs; construction and number
    - Clock distribution (both on-chip and off-chip)
    - Channel construction
    - Clock and data recovery algorithms
    - Voltage and jitter budget allocation
    - Test, debug, validation

# Reference Clock Design

- Not a part of the PCI Express base specification
  - ✓ The clocks implied by the base spec are generated from Reference Clock



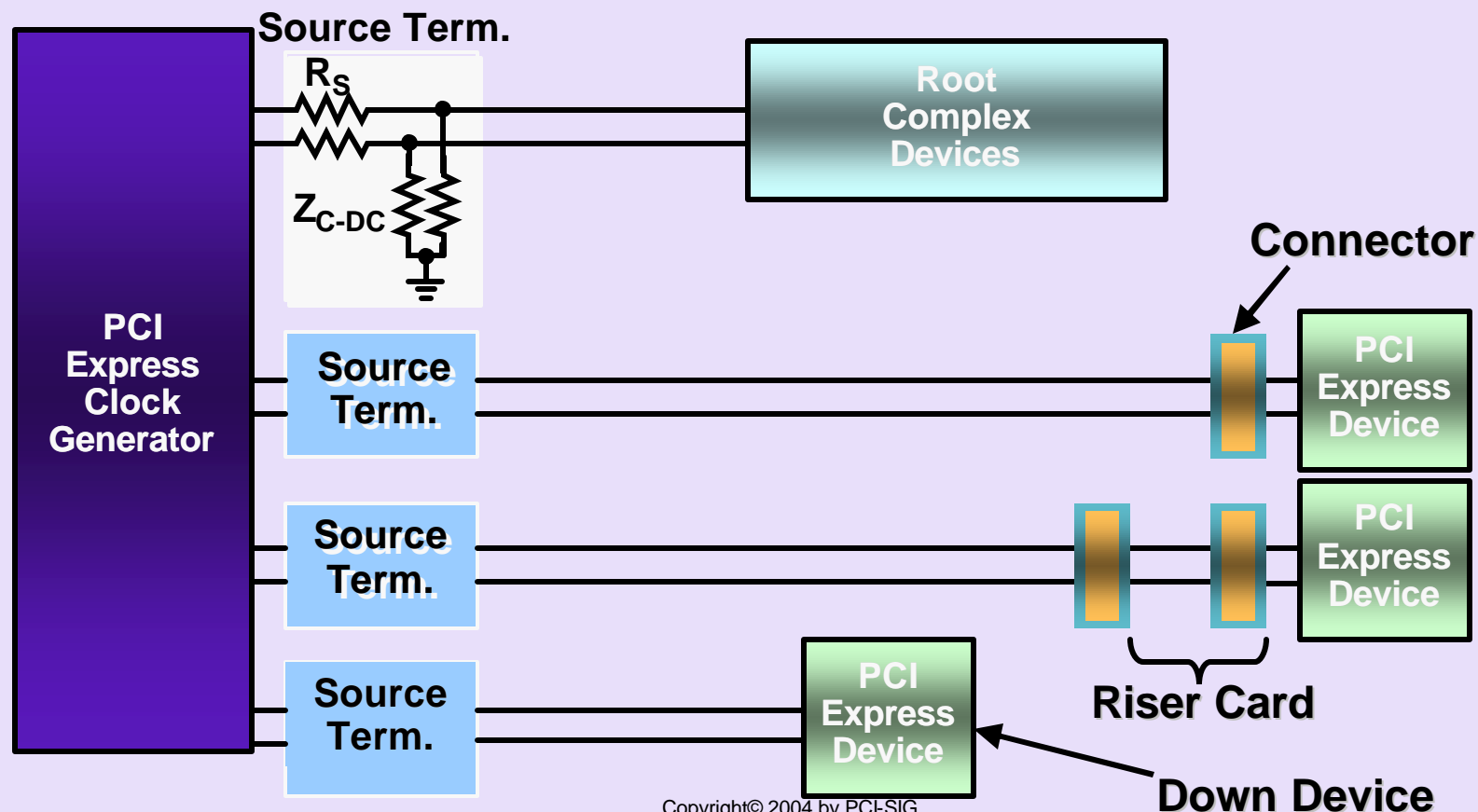
- ✓ REFCLK specified in Form Factor specifications
- ✓ This talk targets the Card Electro-Mechanical Spec.

# Reference Clock Background

- Common platform clock at 100 MHz  $\pm$ 300 ppm
  - ✓ Future process friendly
  - ✓ Differential clock
    - Typically CML
    - Existing low-cost technology
  - ✓ Low swing (0.7 volt nom., single-ended)
  - ✓ Ground referenced

# Reference Clock Distribution

- Distributed to Root Complex and all connectors
  - ✓ Point to point routing
  - ✓ No need of matched routes (matched clock phase)



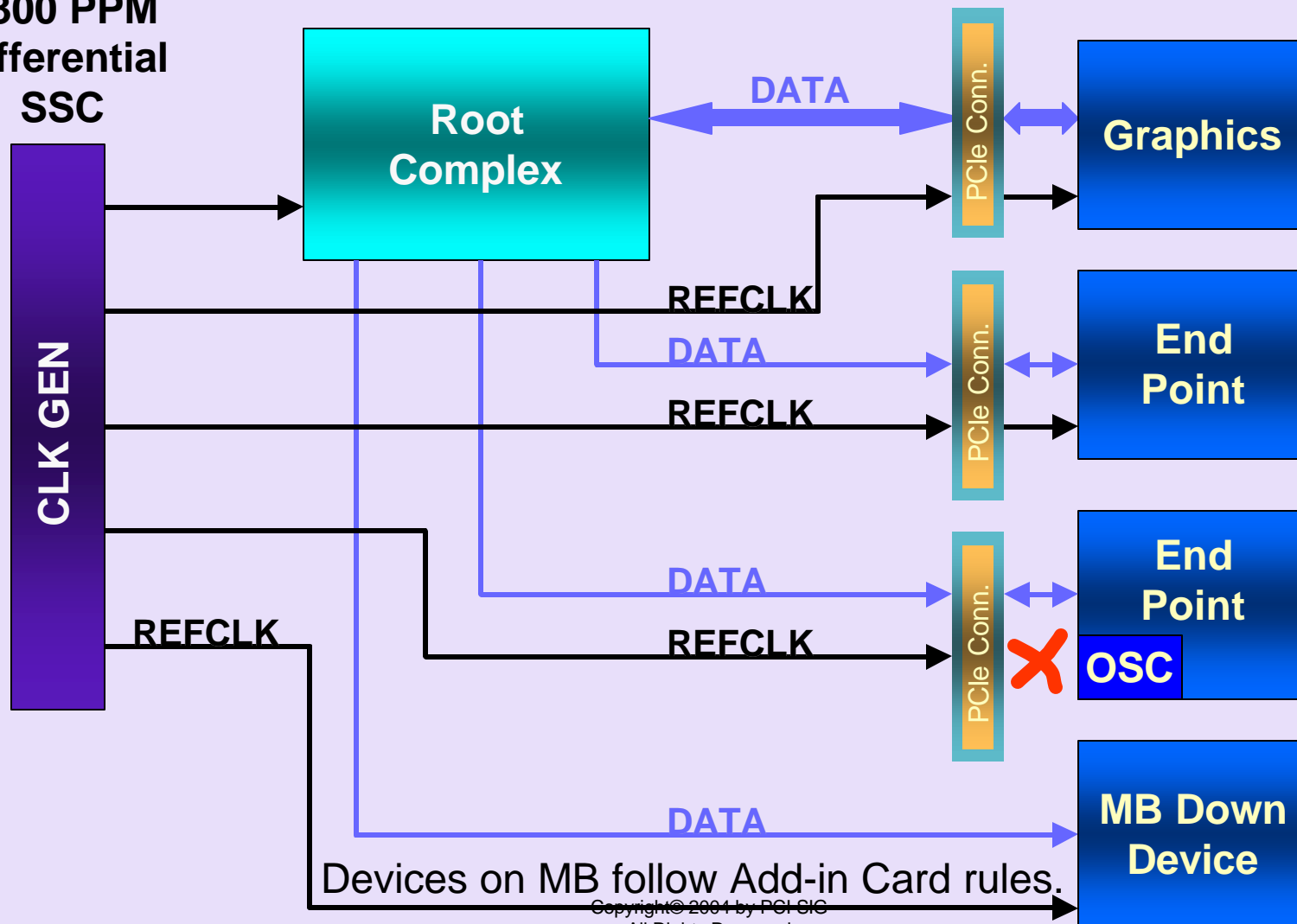
# Reference Clock (cont.)

- Base spec requires tracking of modulated Data. Spread Spectrum Clocking (SSC) .
  - ✓ Data can be modulated:
    - 0.5% downspread at 30-33 KHz
    - Triangular or “Hershey-Kiss” profile
- Add-in cards are not required to use the REFCLK
  - ✓ **BUT**, must receive and transmit with same SSC as PCI Express connector REFCLK
  - ✓ Can't use **ONLY** a local oscillator on the card.
- Devices on the motherboard follow Add-in Card rules.

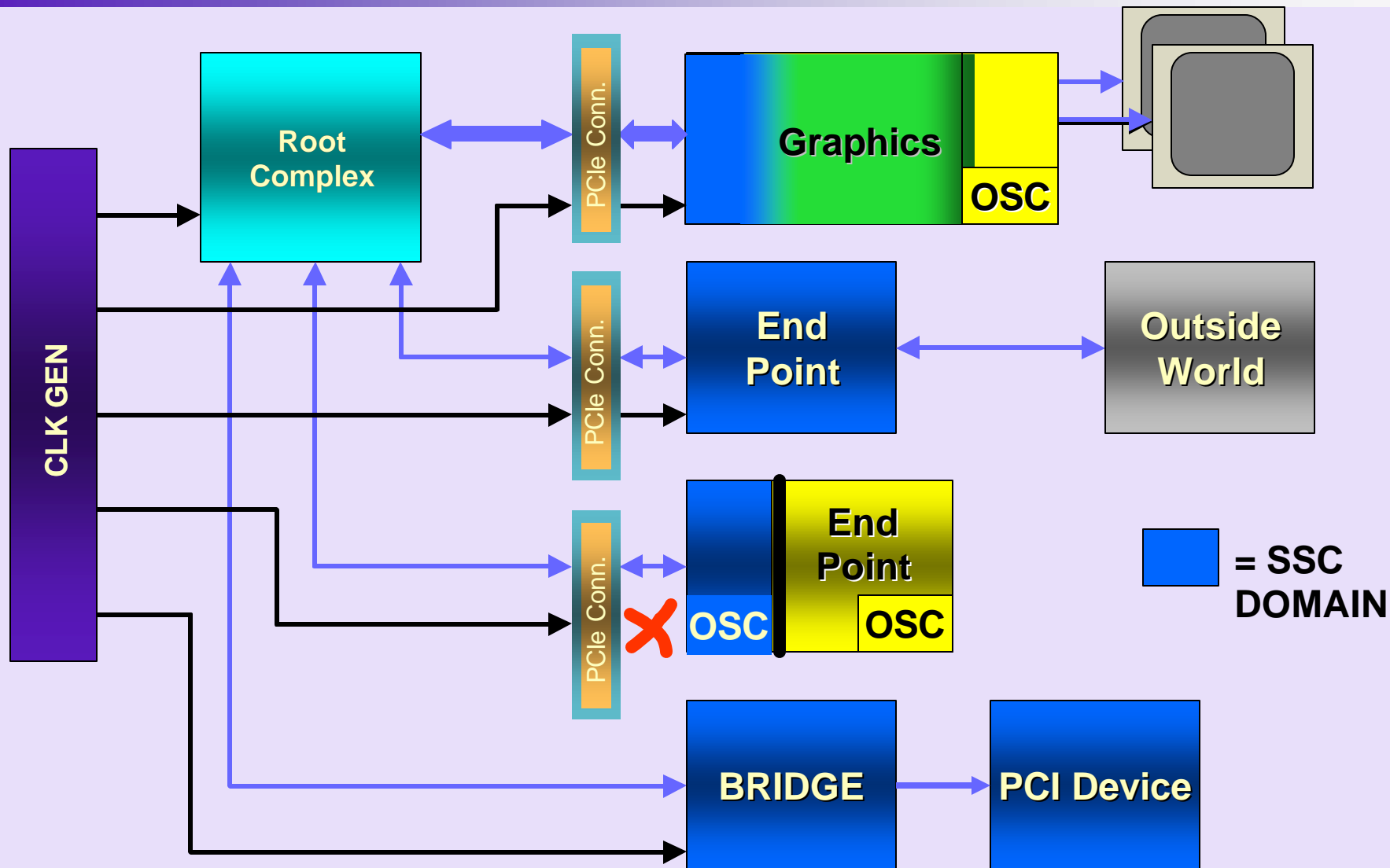
**Slide Updated**

# Reference Clock Usage

100 MHz  
±300 PPM  
Differential  
SSC



# SSC Across the Platform





# Topologies Simulated

- GOAL: Assure existence of a robust solution space
  
- Topologies from CEM that were studied:
  - ✓ Device down on the Motherboard
  - ✓ CEM with one connector
  - ✓ CEM with a riser card and 2 connectors
  - ✓ CEM with one connector and add-in card termination

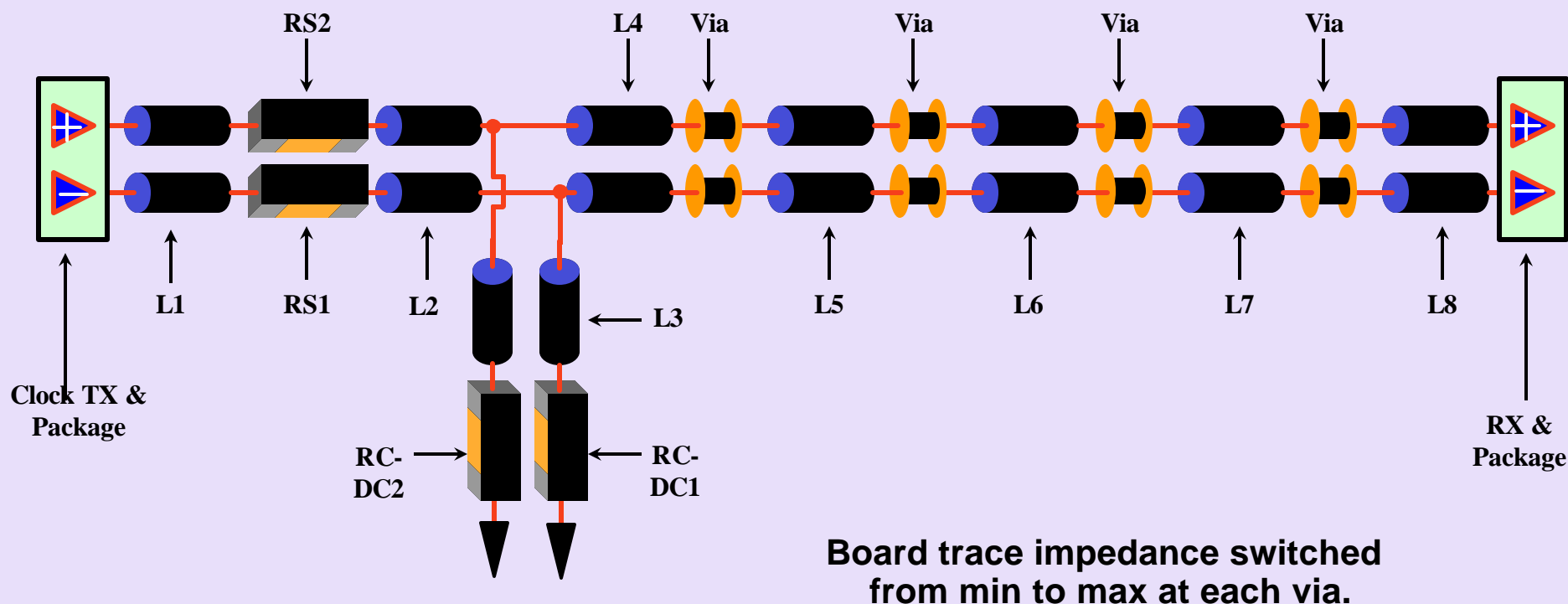
# Models

- Nom. 100  $\Omega$  diff. trace models with  $\pm 20\%$  Max and Min corners
- Tuned clock generator driver model to cover CEM Clock
- Foxconn PCI Express connector
- Example solution space sweep matrix:

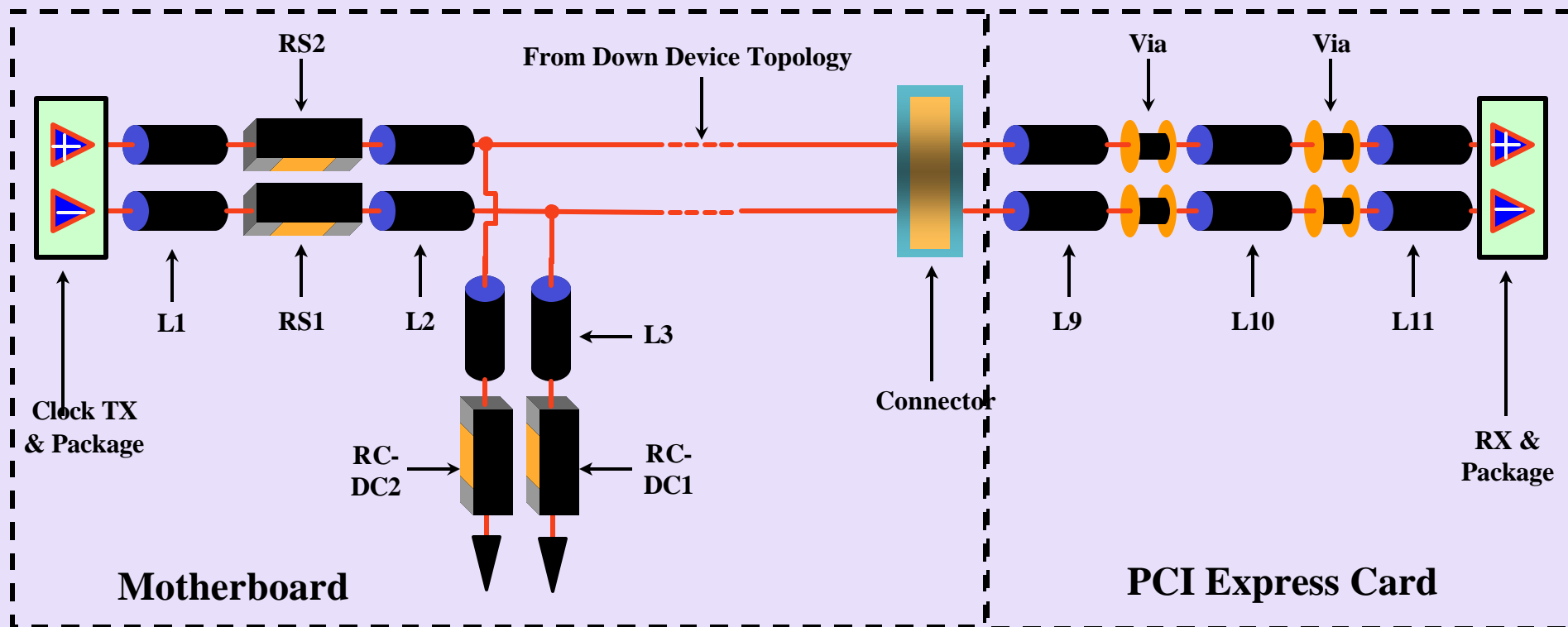
Element	Max	Min	Notes
L1	0.5"		Coupled Golden Channel transmission line models. Target 100 $\Omega \pm 20\%$ differential
L2, L3	0.2"		
L4, L5, L6, L7, L8	3.0"	0.2"	Swept in 0.2" increments (1" - 15" total length)
RX & Package	X	X	1.5 pf – 3 pf load, diff variance 0.25 pf. Wirebond & Flipchip package
RS1, RS2	34.65 $\Omega$	31.35 $\Omega$	33 $\Omega \pm 5\%$
RC-DC1, RC-DC2	50.5 $\Omega$	49.5 $\Omega$	50 $\Omega \pm 1\%$

# Down Device Topology

- For Root Complex and other motherboard devices without connectors

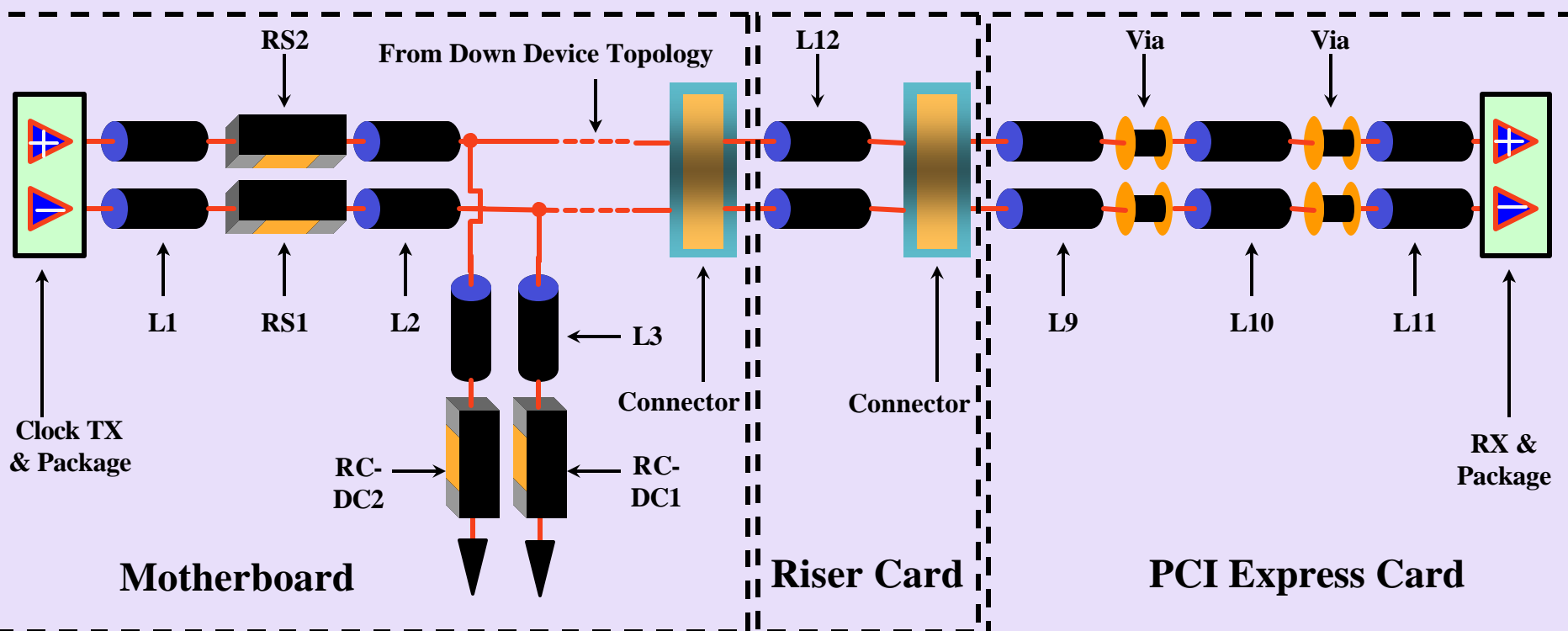


# CEM Connector Topology



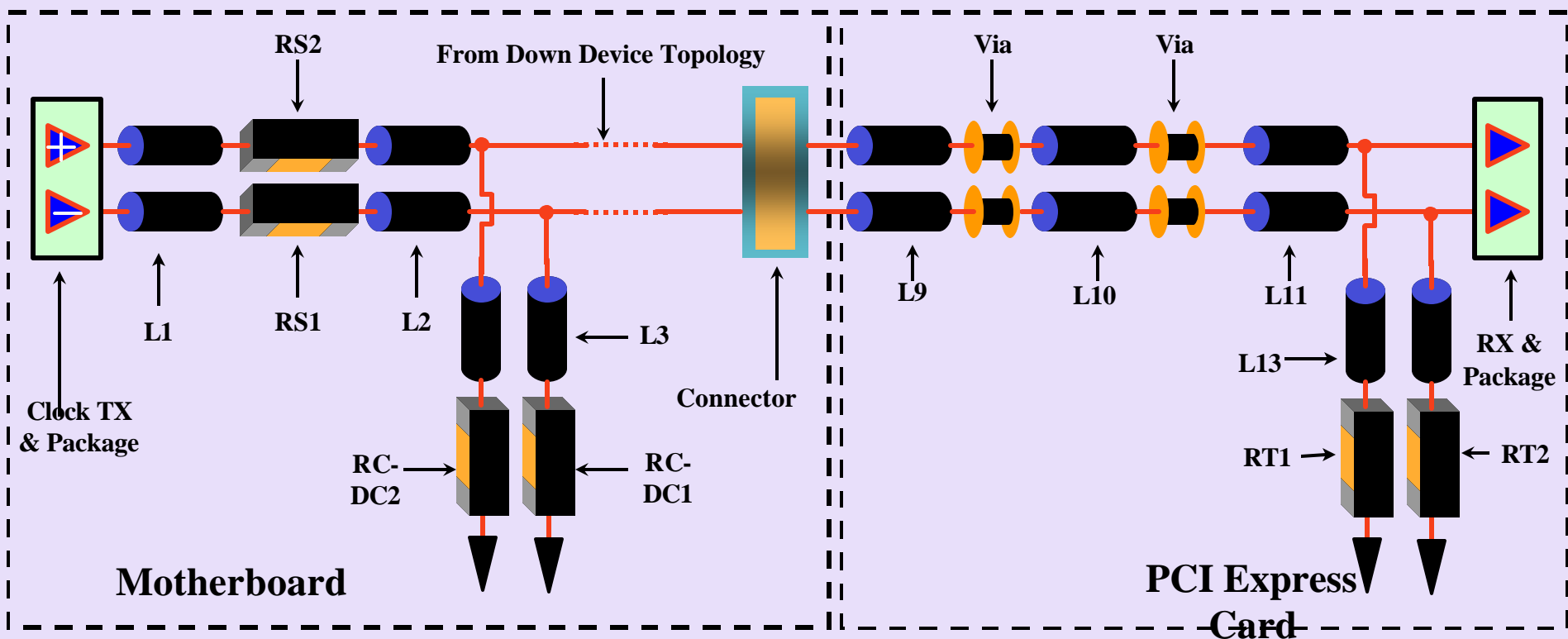
# CEM Riser Card Solution Space

- Riser Card adds second connector
  - ✓ Riser belongs to motherboard budgets
  - ✓ REFCLK specs checked at connector on riser into standard load



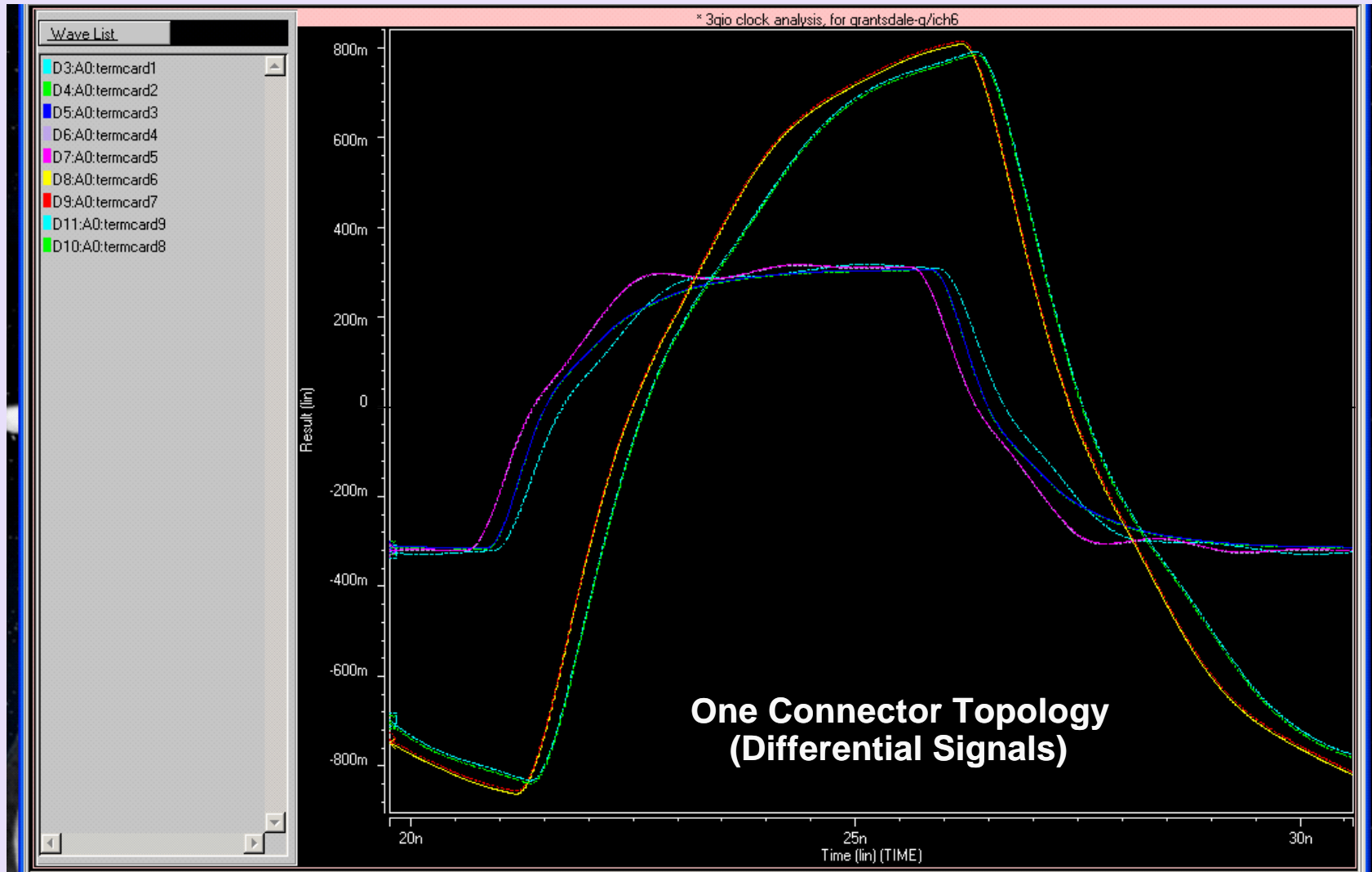
# Receiver REFCLK Termination

- Receiver Terminations ( $50\ \Omega$ ) added

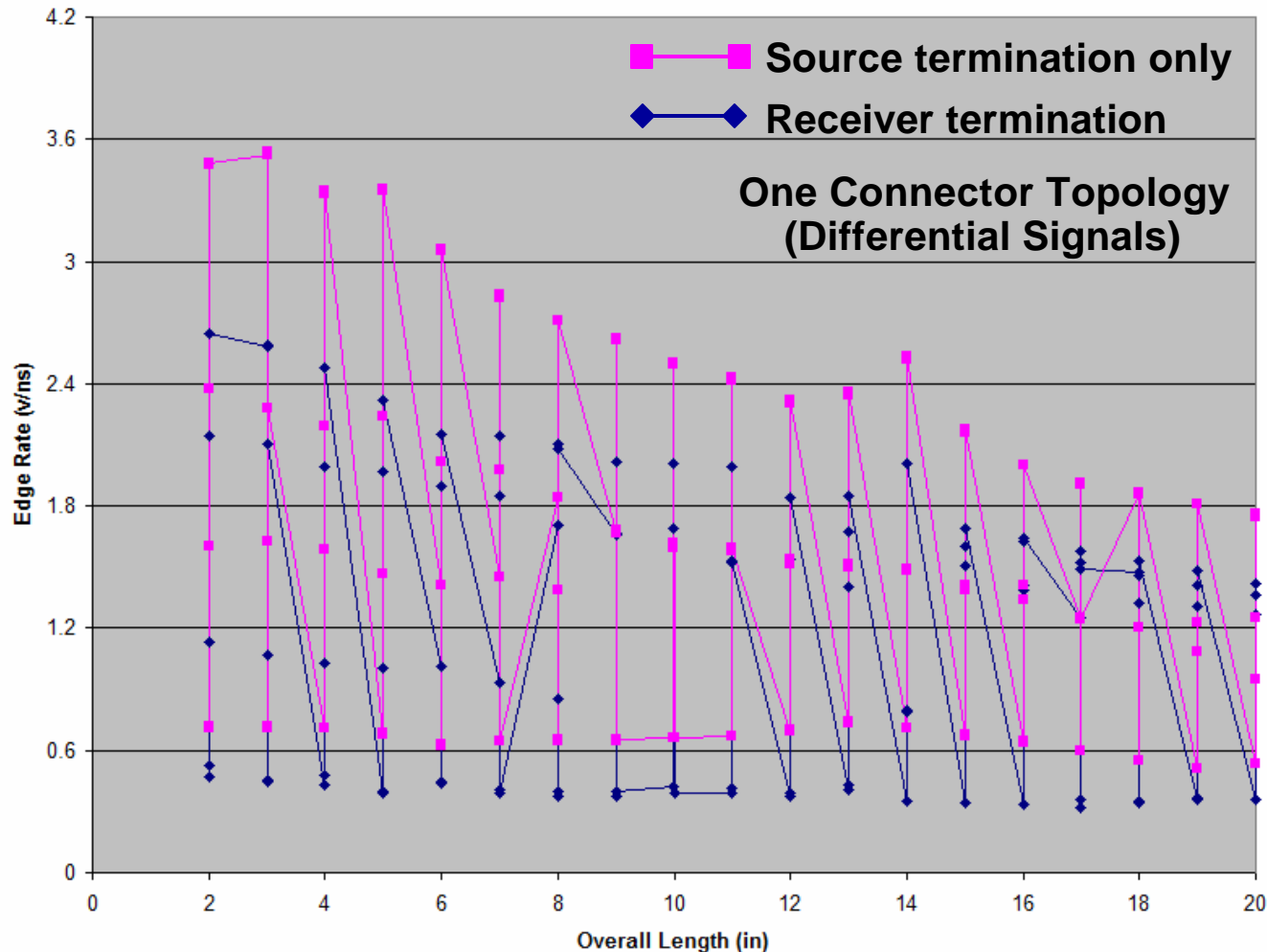


# Min. Edge Rate Example

## - Source and Receiver Terminated



# Edge Rate Vs. Length





# CEM REFCLK Topology Summary

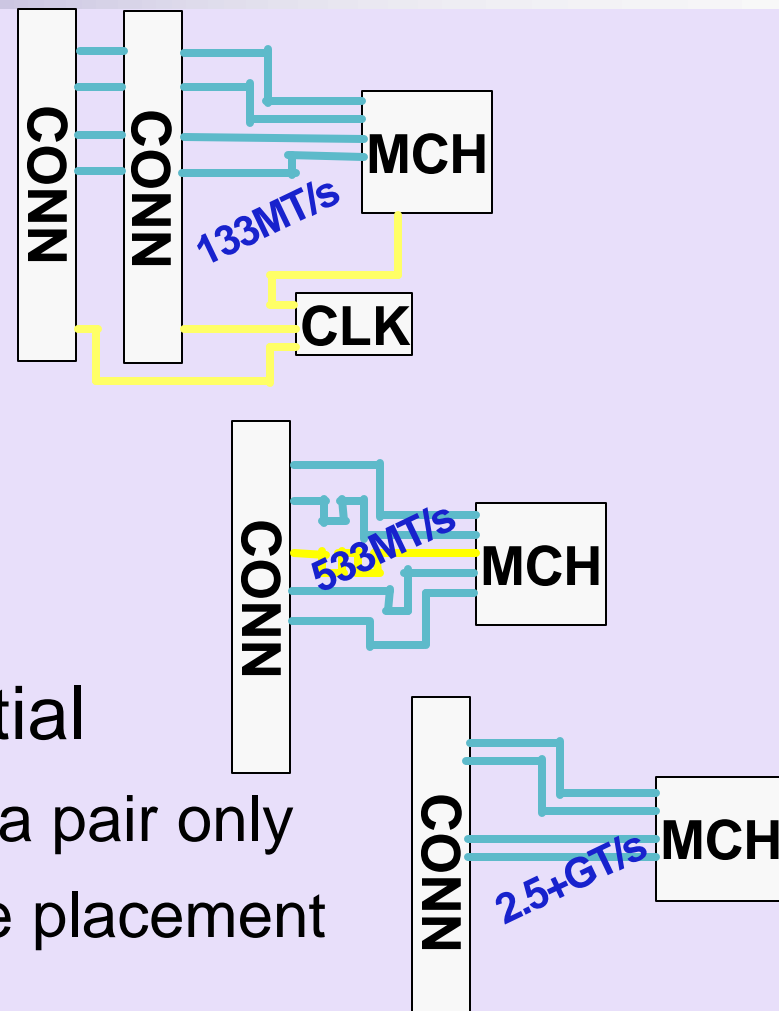
- Simulations show solution space to 16" total interconnect length.
  - ✓ There are a few short riser card cases that fail for low slew rate
  - ✓ Verification of clock topology is required
- Recommending 4" max trace length on add-in card
  - ✓ Consistent with Data trace recommendations in CEM Spec
- Receiver Termination allowed
  - ✓ Slew rate and clock swing cut in half
  - ✓ Designer should check for impact on input jitter

# Board Design

- Background
- Layout considerations
- Simulations
- Summary

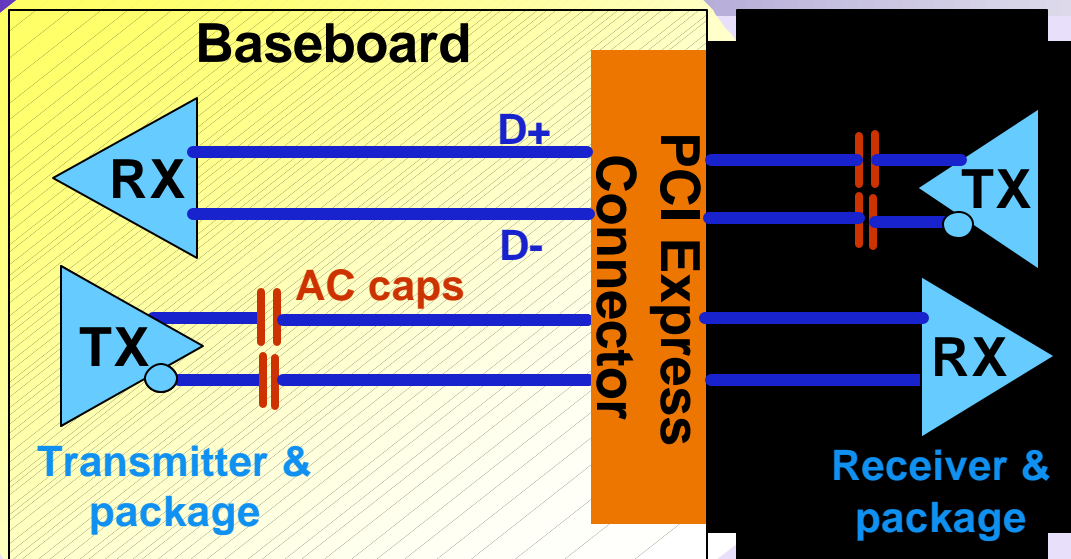
# Bus Topologies

- PCI common clock
  - ✓ Meet setup/hold timing
  - ✓ Multi-drop parallel I/O
- AGP source synchronous
  - ✓ Match all data to strobe
  - ✓ Single strobe, multiple data
- PCI Express serial differential
  - ✓ Point-to-point, match per data pair only
  - ✓ Longer route, creative device placement

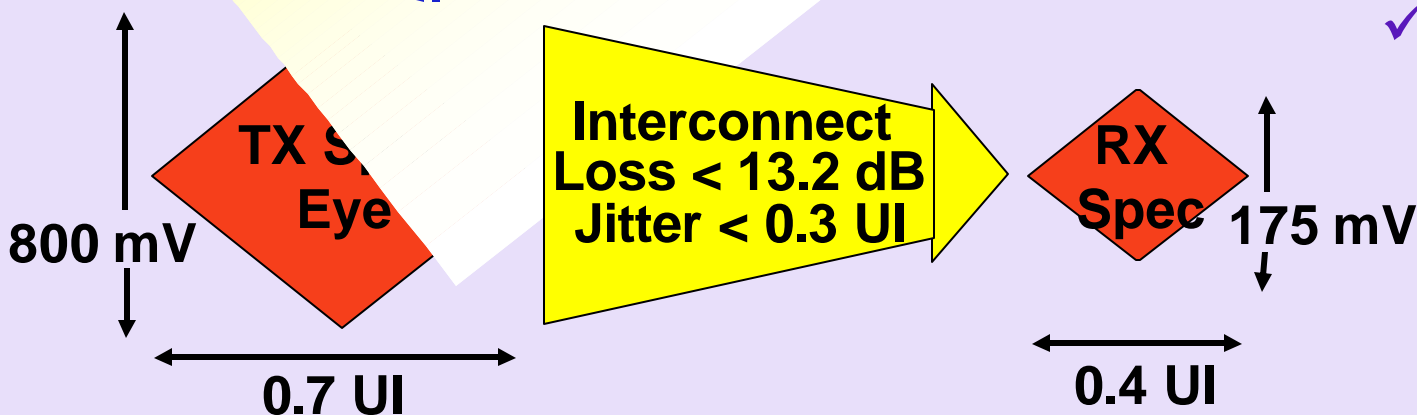


**Point-to-point routing is straightforward**

# Differential



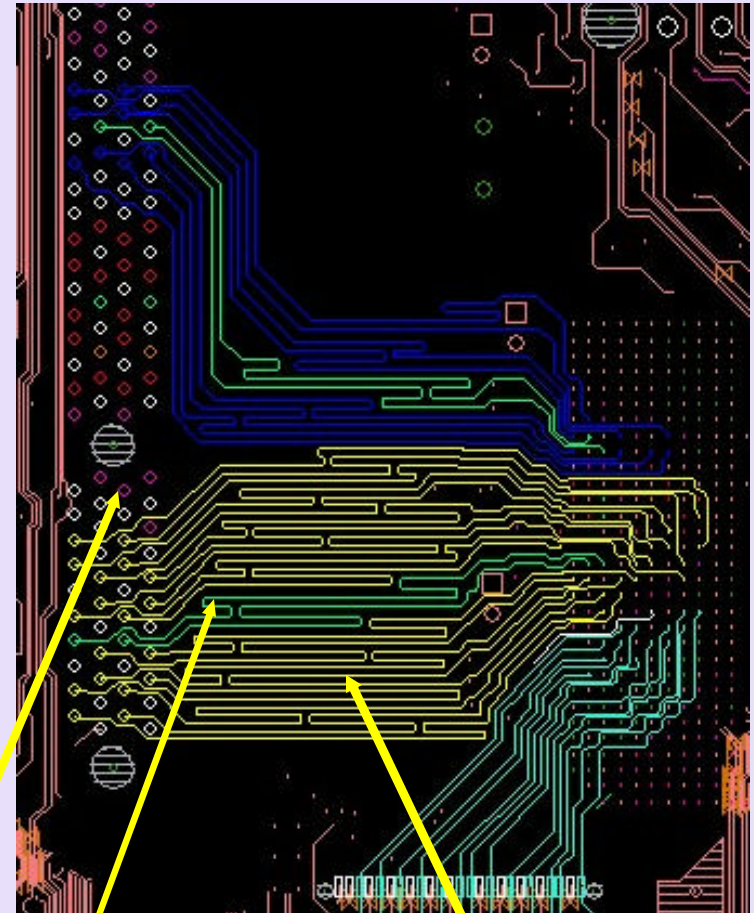
- ✓ AC coupled
- ✓ Lane-to-lane de-skew
- ✓ Polarity inversion
- ✓ On-chip equalization
- ✓ On-chip terminations



UI = Unit Interval 400ps

# AGP8X Layout Challenges

- Data and Strobe must be length matched
  - ✓ Serpentine routing is needed for length matching
- Short Motherboard Trace Lengths
  - ✓ 2"–6" max MCH to connector
- Tight Timing Budget
  - ✓ Data-to-strobe timing skew



AGP Connector

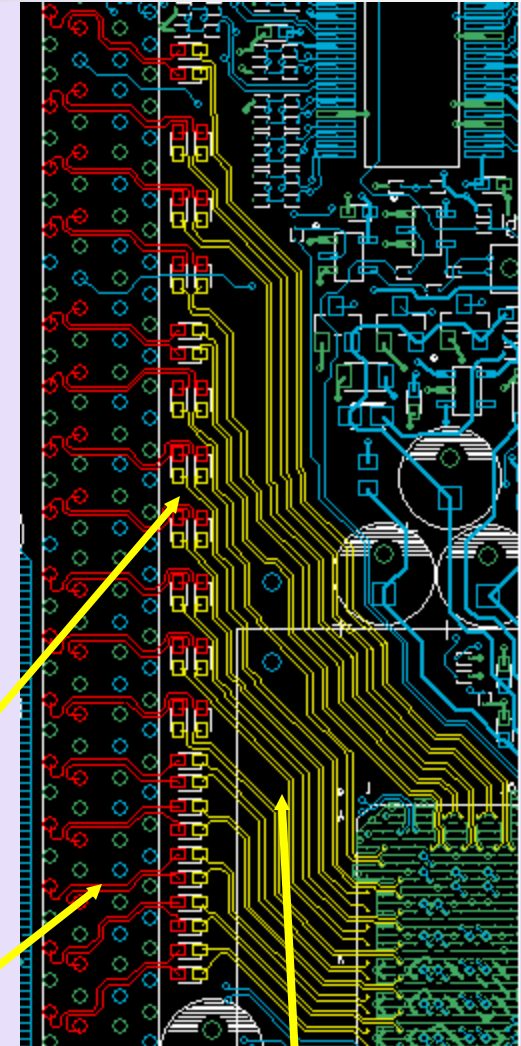
Strobe

Data

# PCI Express makes layout easy

- Trace length matching between pairs is not required
  - ✓ Embedded clock simplifies routing rules
- GND reference preferred
  - ✓ Avoid splits and voids
- Use GND stitching vias when changing layers
- Longer motherboard trace
  - ✓ 12"+ possible

AC Coupling Caps  
x16 PCI Express Connector



No trace serpentine 46

# Interconnect budget

Interconnect	Loss	Jitter
Baseboard (connector and/or riser card)	6.6 dB	0.19 UI
Add-in card	1.4 dB	0.035 UI
Near-end crosstalk	2.7 dB	0.075 UI
Impedance mismatch	2.5 dB	
<b>Total</b>	<b>13.2 dB</b>	<b>0.3 UI</b>

- Loss and Jitter are key parameters
- Impedance is not as critical
- Maintain differential pair symmetry
- Design tradeoffs for PCB: component loss vs. trace length

**Manage loss and symmetry to meet budget**

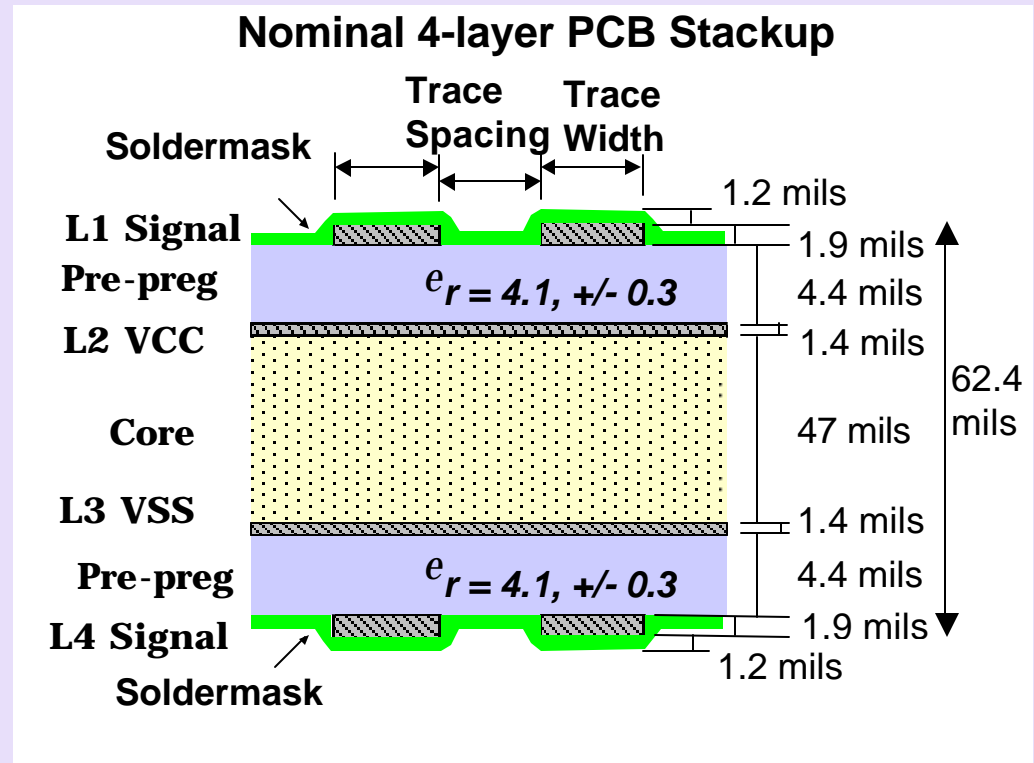
# Board Design

- ✓ Background
- Layout considerations
- Simulations
- Summary



# Stackup design

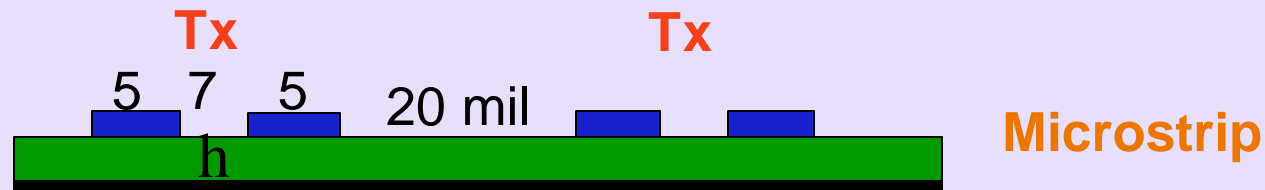
- No new PCB technology required
- Standard 4-layer stackup 0.062" thick PCB
- Microstrip ½ oz Cu plated **Ok**
- Stripline 1 oz Cu (6+ layers) **Better**



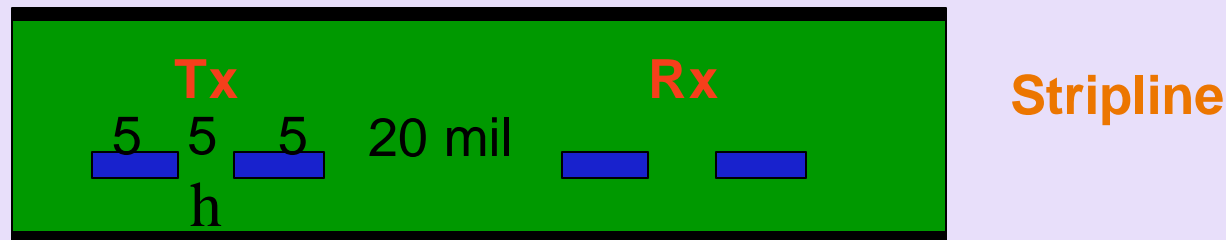
**Follow simple layout rules & design tradeoffs**

# Trace Geometry & Impedance

- Use wider trace width  $\Rightarrow$  Minimize loss
- Use wider traces for long routes
- More pair-to-pair spacing  $\Rightarrow$  Minimize crosstalk
- Target differential  $Z_0$  of  $100\ \Omega \pm 20\%$



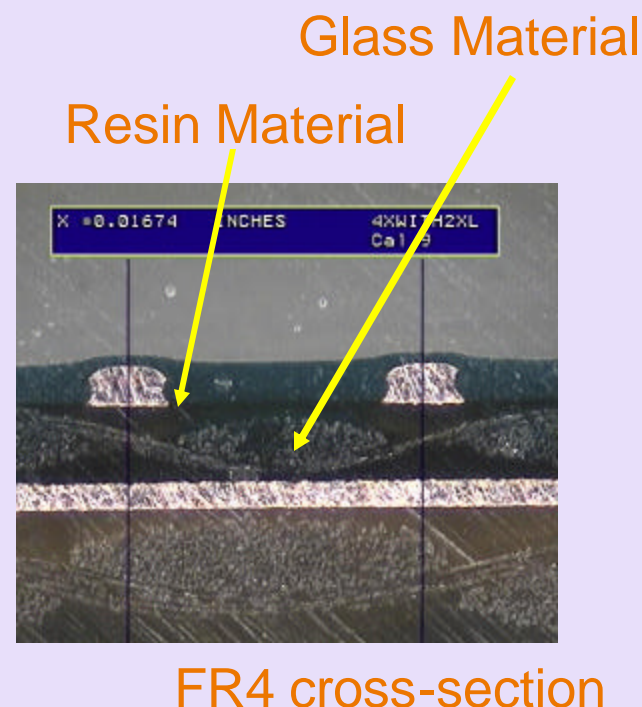
Non-interleaved Topology example



Interleaved Topology example

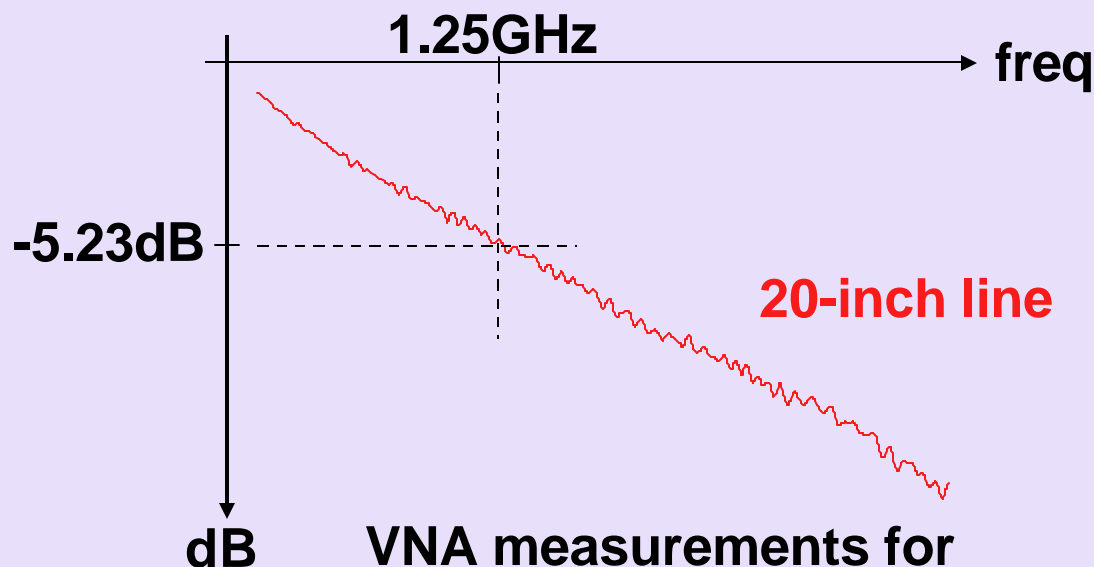
# PCB material dominates loss

- Stackup FR4 material
  - ✓ Copper roughness  $\Rightarrow$  loss  $\uparrow$
  - ✓ Thinner dielectrics  $\Rightarrow$  loss  $\uparrow$
- Non-homogeneous dielectric
  - ✓ Localized  $Z_0$  variation due to material weave  $\Rightarrow$  loss  $\uparrow$
- Wide differential Impedance variation on  $\mu$ strip
  - ✓ Etching and Plating process  $\Rightarrow$  loss  $\uparrow$



# Trace length

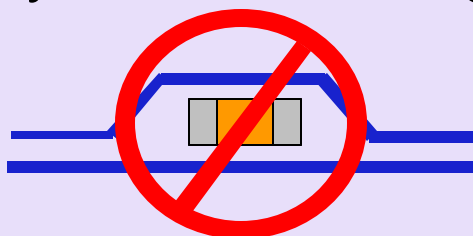
- Longer trace length  $\Rightarrow$  loss  $\uparrow$
- 0.25 to 0.35 dB inherent loss per inch for FR4 microstrip traces
- Limit motherboard trace to  $< 12$  inches and add-in card trace to  $< 3$  inches



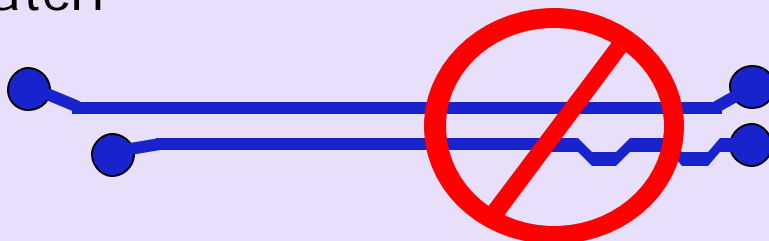
VNA measurements for  
trace insertion loss

# Trace Symmetry & Matching

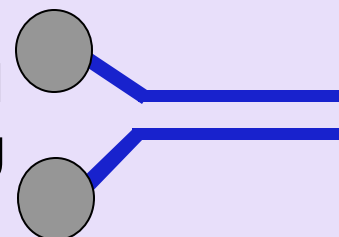
- Match each differential pair per segment
  - ✓ Match overall length  $\leq 5$  mils
  - ✓ Symmetric routing for each pair



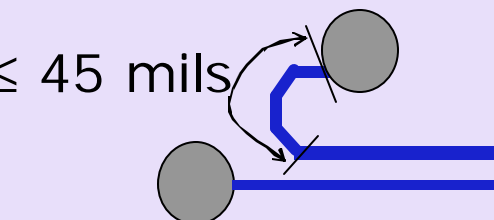
Match  
near  
mismatch



Preferred  
matching



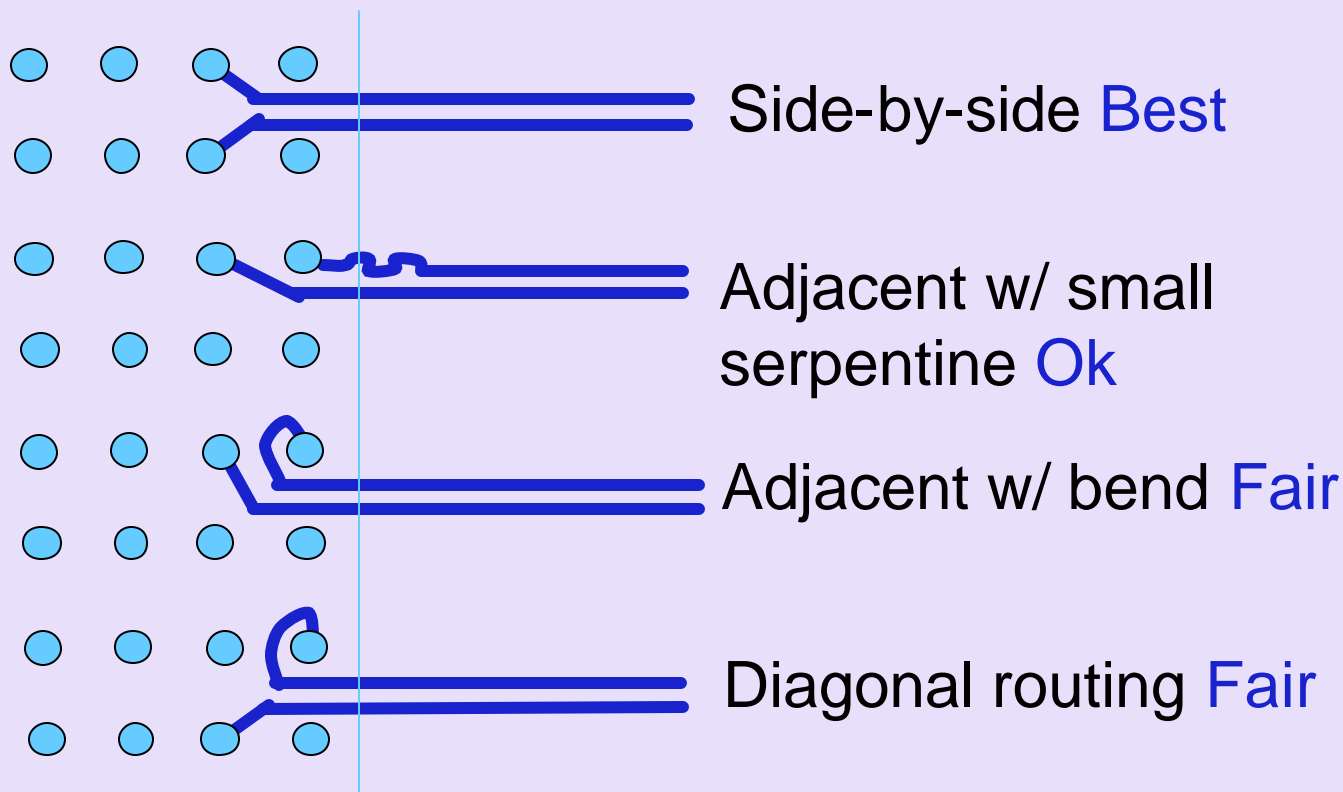
$\leq 45$  mils



Alternative  
matching

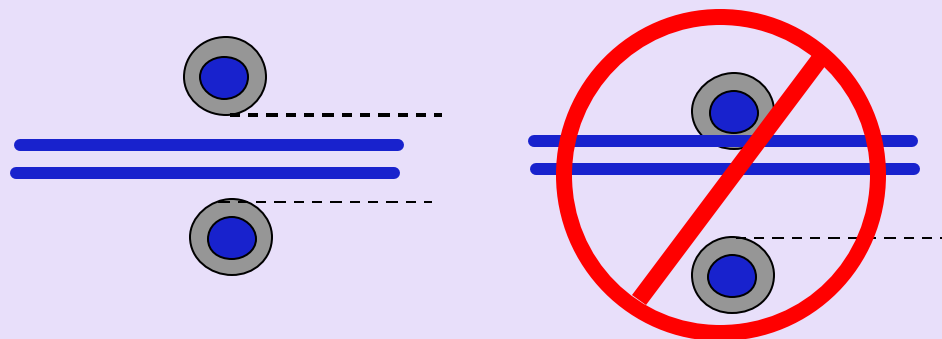
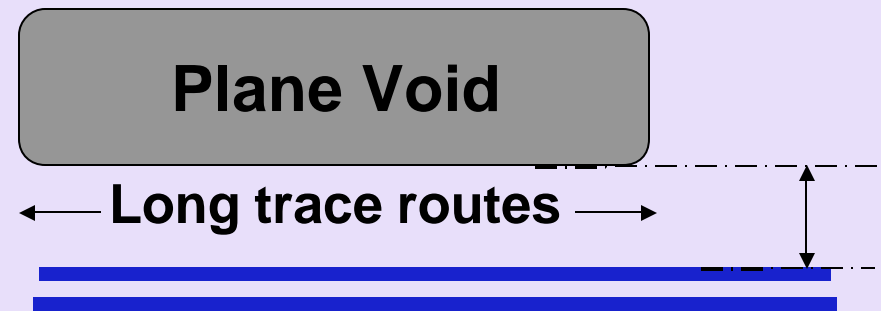
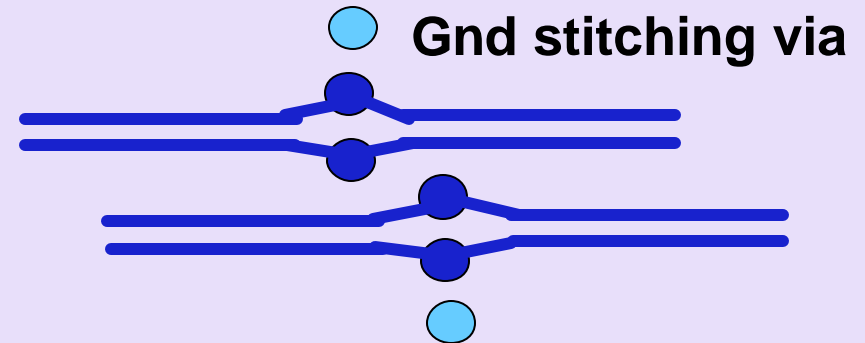
# Pin field breakout

- Use side-by-side breakout for package to maintain symmetry
- Avoid tight bends



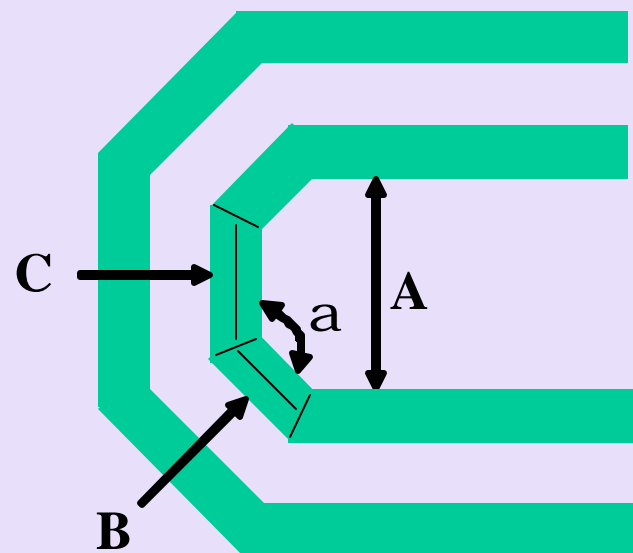
# Reference plane

- Full ground plane reference
- Stitching vias required for layer transition
- Clearance near plane void
- Avoid trace over anti-pad



# Bend Guidelines

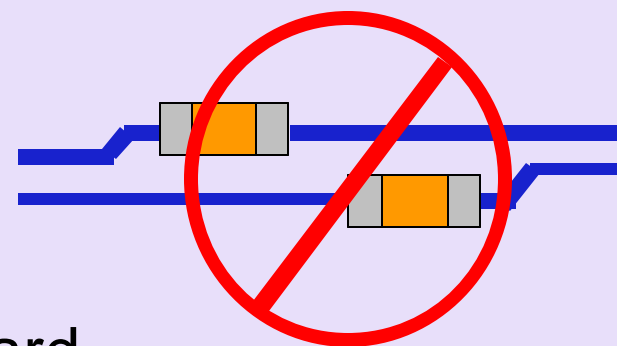
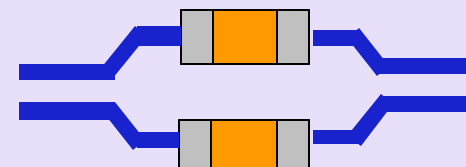
- Avoid tight bends
  - ✓ No 90° bends; impact to loss and jitter budgets
- Keep angles  $\geq 135^\circ$  (a)
- Keep minimum air gap
  - ✓  $A \geq 3x$  the trace width
- Length of B and C  $\geq 1.5x$  the width of the trace





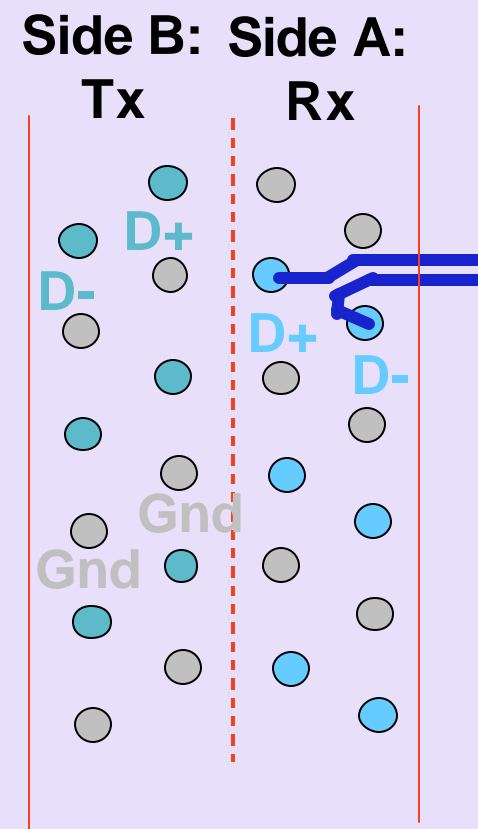
# AC coupling caps

- Size: 0402 **best**, 0603 **ok**
  - No 0805 size or C-packs
  - Symmetric placement
- 
- Cap Size: 0.1  $\mu$ F **best**
  - Cap location:
    - ✓ Along Tx pairs on Motherboard
    - ✓ Along Tx pairs on Add-in card



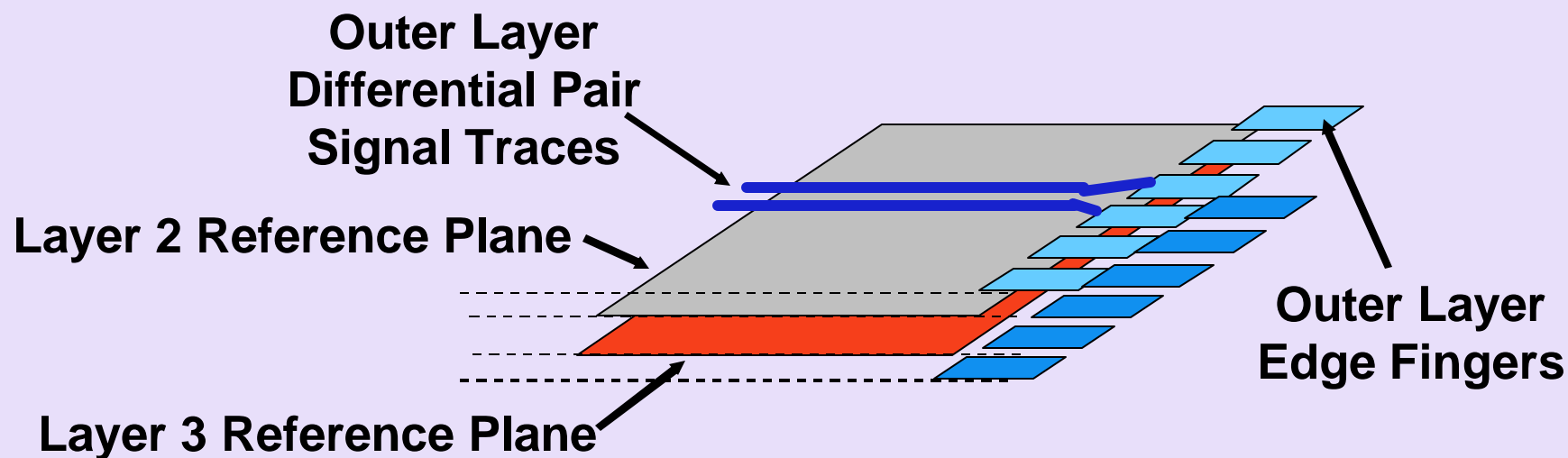
# Connectors

- New connector with standard PTH
- Pinout optimized for differential routing
- Loss & crosstalk part of baseboard budget
- Connector sizes: x1, x4, x8, x16



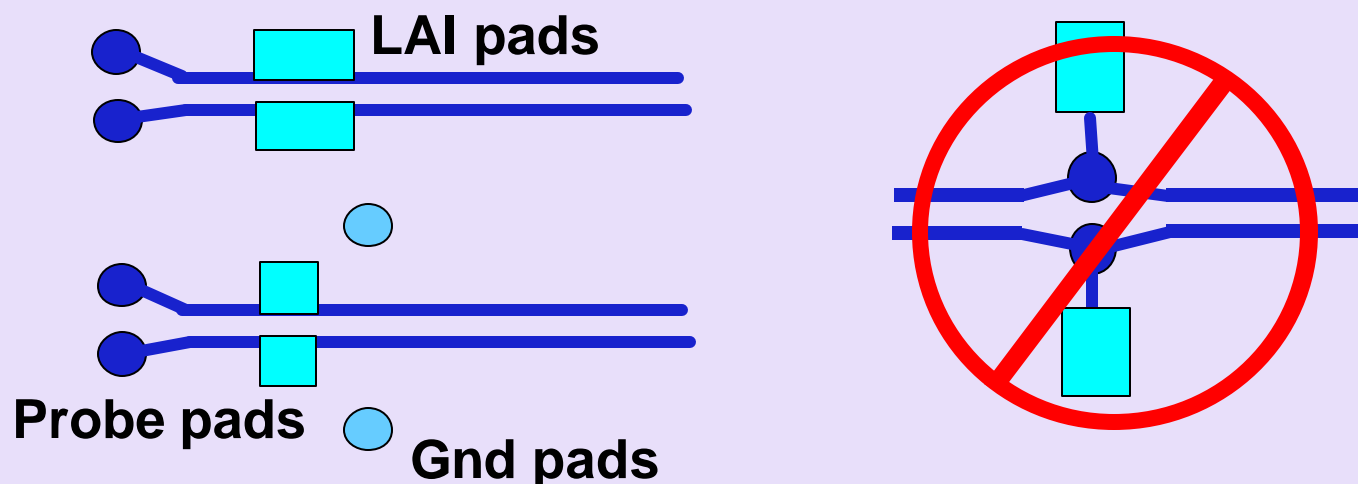
# Card edge fingers

- Remove ref plane under edge fingers pads
  - ✓ For better impedance/loss performance



# Test points & Vias

- Minimize Vias usage
  - ✓ Up to 0.25 dB loss per via
  - ✓ Via pad size  $\leq 25$  mil, hole size  $\leq 14$  mil
- Put test points or LAI pads in series
  - ✓ No stubs
  - ✓ Provide Gnd pads for single-ended probing



# Board Design

- ✓ Background
- ✓ Layout considerations
- Simulations
- Summary

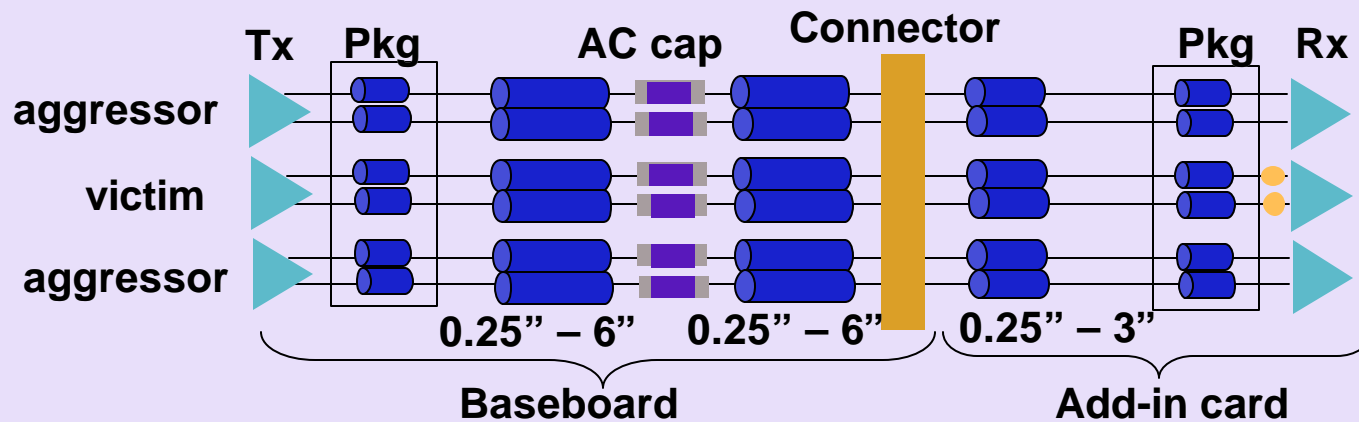
# Simulations

- Simulations to maximize board solution space
  - ✓ Simulation analysis using HSPICE, etc
  - ✓ Dielectric and conductor loss must be modeled
- Simulate spec parameters
  - ✓ Compliance Eye (Loss/Jitter)
  - ✓ AC & DC common mode
  - ✓ Return Loss (for buffer/package)
- Models
  - ✓ Buffer, package, PCB (trace, via), connector
  - ✓ Worst case ref channel

**Perform simulations to maximize solution space**

# Topology & modeling

- Multi-pair (2 aggressors, 1 victim) coupled models

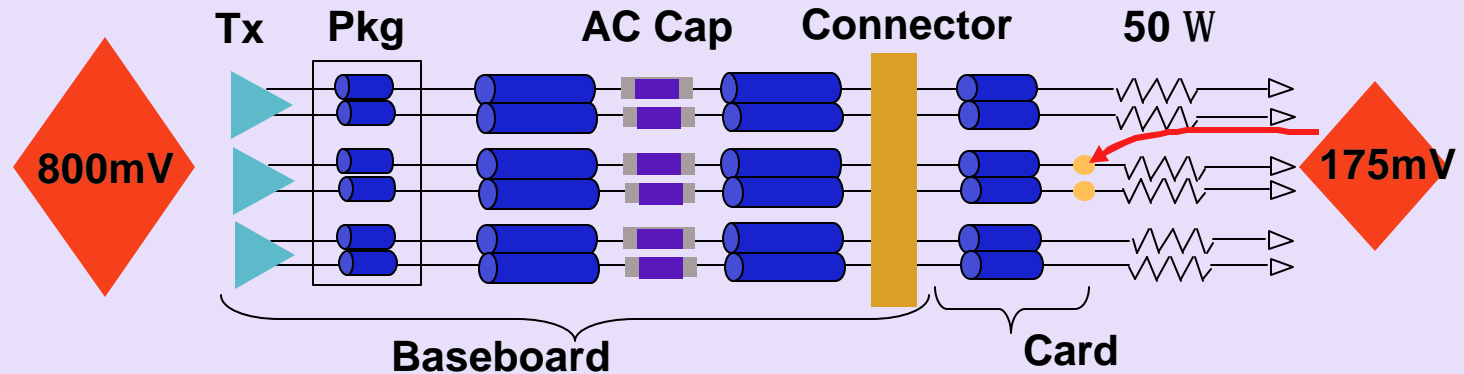


- Corner case PCB: impedance variations and non-homogenous effects
- 8b/10b compliance data pattern in Spec

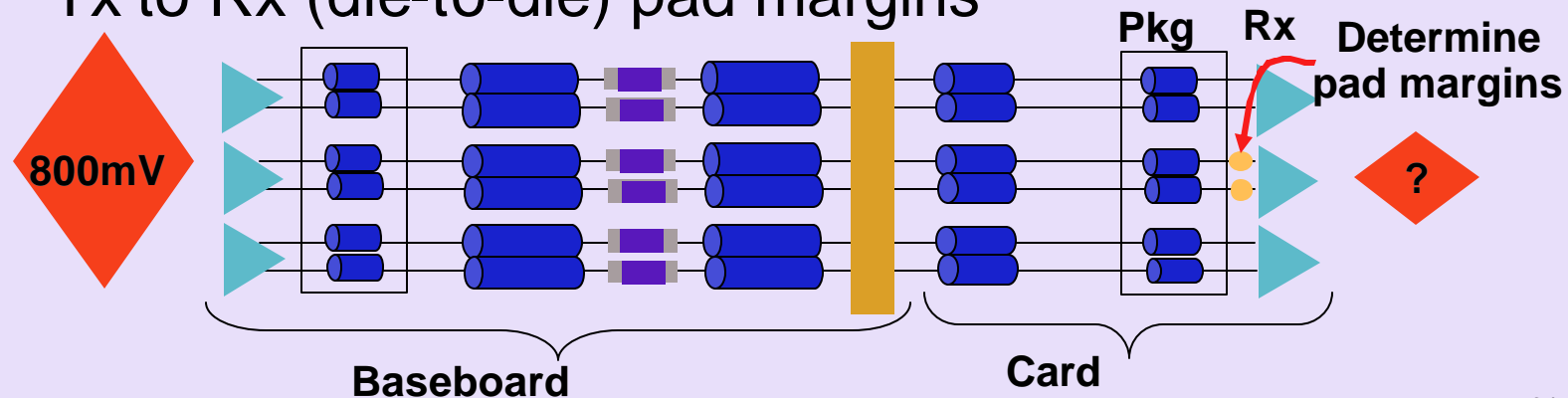
Aggressor	1100000101	0011111010	1100000101	0011111010	1100000101	0011111010
Victim	1100000101	0011111010	1100000101	1100000101	0011111010	1100000101
Aggressor	1100000101	0011111010	1100000101	0011111010	1100000101	0011111010

# Spec vs. Product Simulations

- Spec simulations for generic board solution
  - ✓ Tx to 50Ω load ~175mV



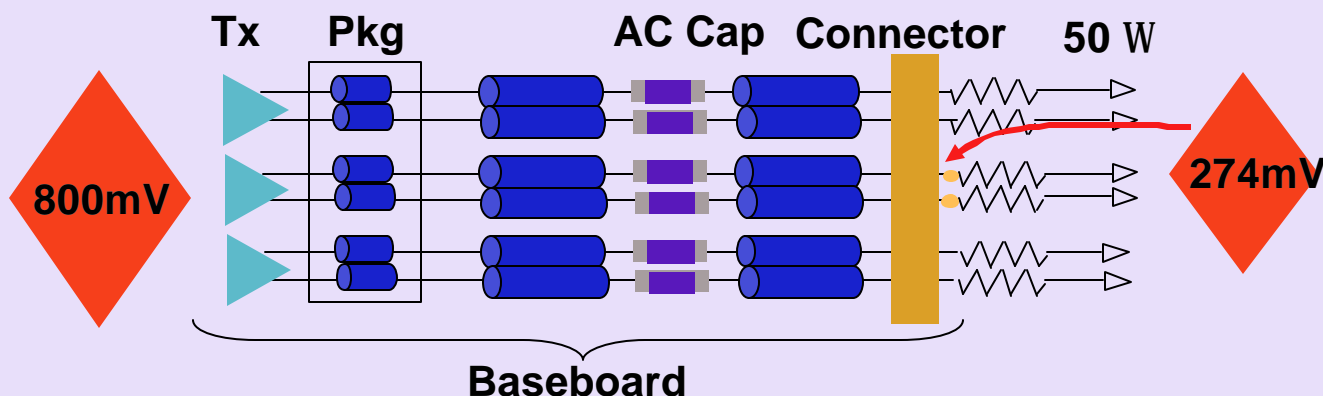
- Product simulations for specific package
  - ✓ Tx to Rx (die-to-die) pad margins



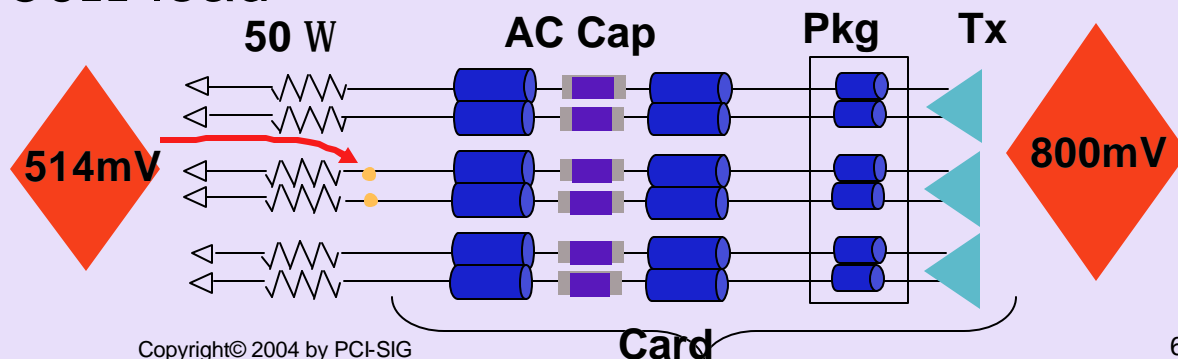


# Baseboard vs Card TX Eyes

- CEM Spec for separate Baseboard vs Card budget
- ✓ Baseboard Tx to 50Ω load



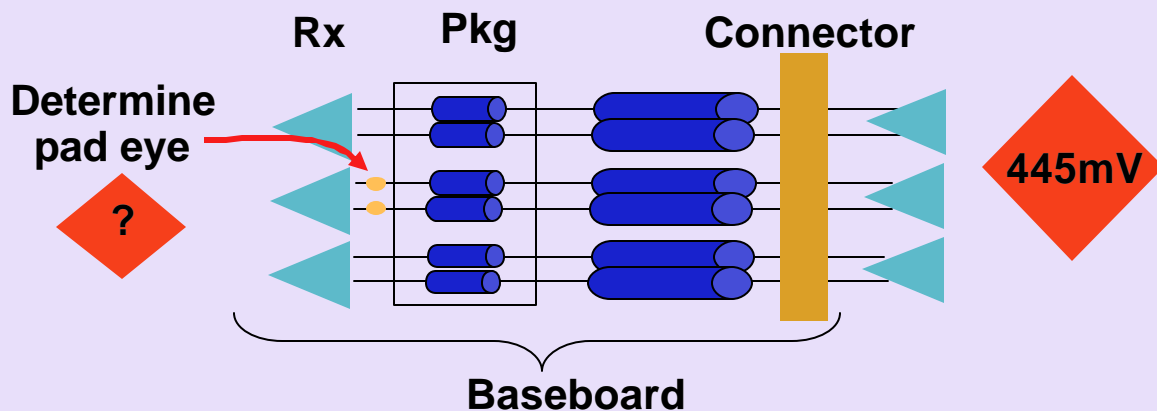
- ✓ Card Tx to 50Ω load



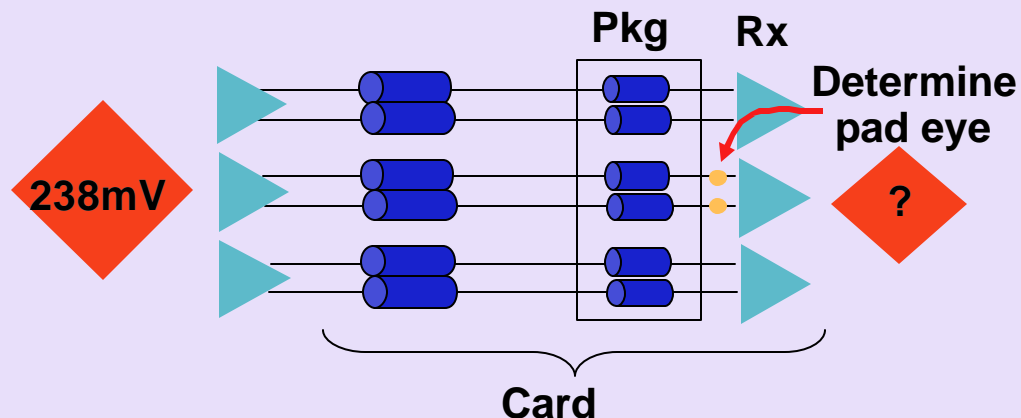
# Baseboard vs Card RX Eyes

- CEM Spec defines Baseboard vs Card input requirements

✓ Eye for Baseboard Rx

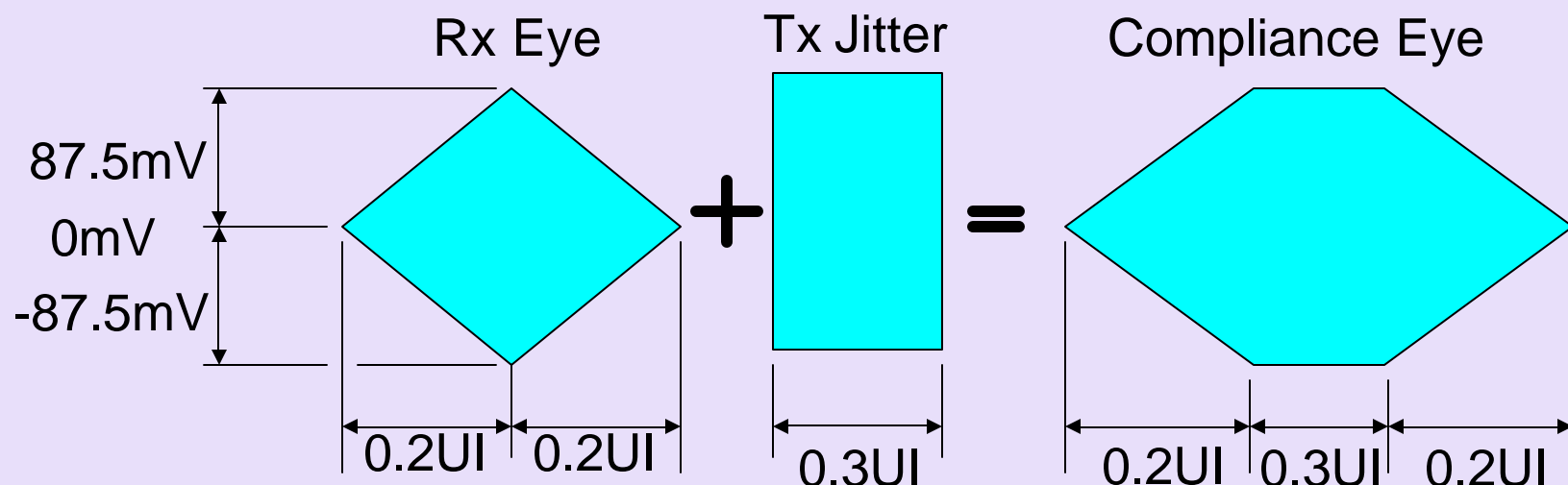


✓ Eye for Card Rx



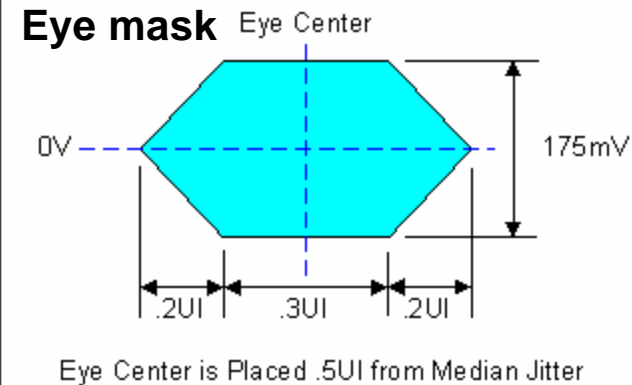
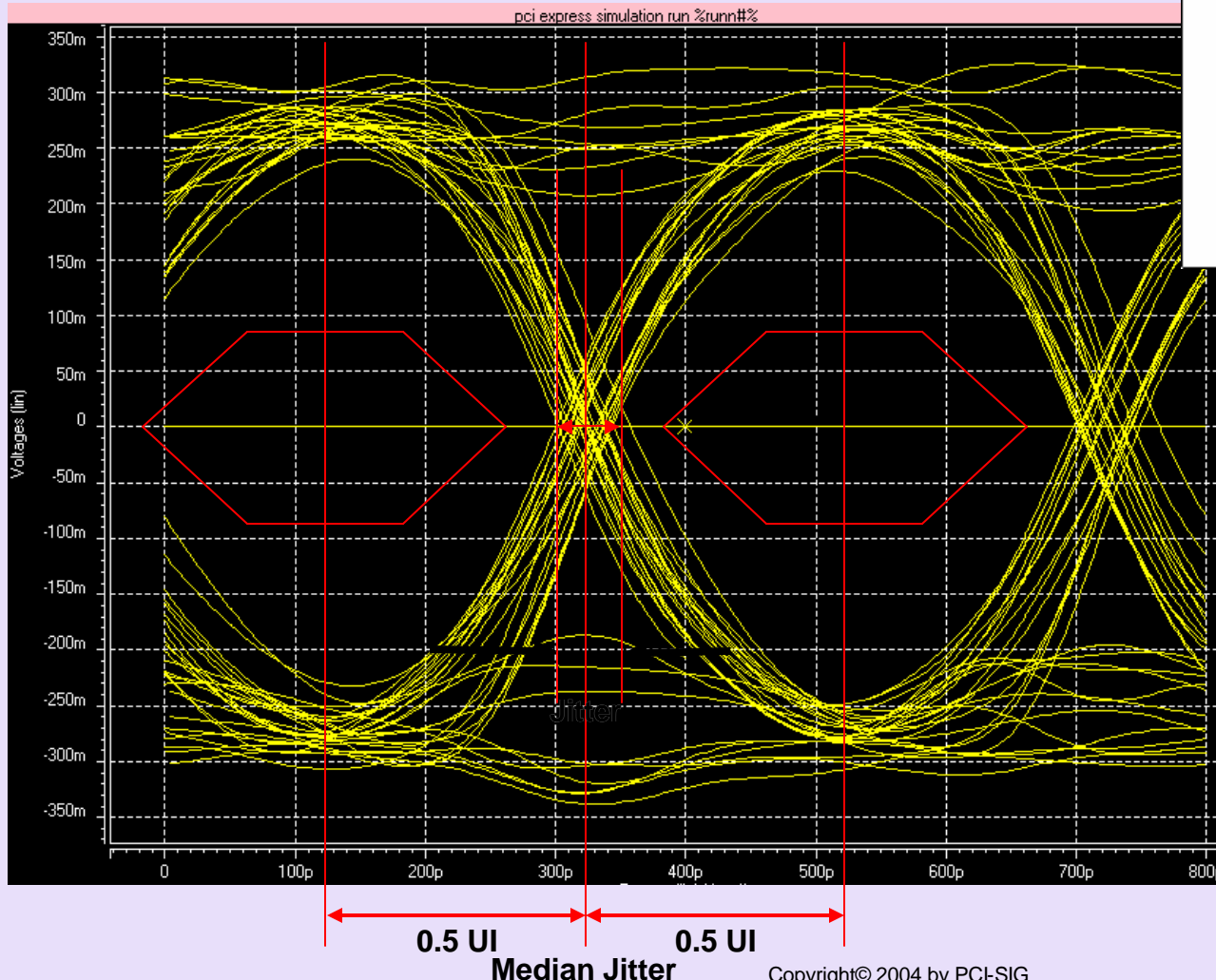
# Compliance eye mask

- Must meet compliance eye @ Rx pins
  - ✓ Min Rx eye  $\Rightarrow$  0.4 UI at 0 mV differential
  - ✓ Min  $V_{diff-p-p}$  at Rx  $\Rightarrow$  175 mV
  - ✓ Add any Tx jitter not included in modeling



# Data Analysis

## ■ Example eye diagrams at RX pin



### Example parameters for worst-case eye:

RX cap = 1.3 pF  
 TX cap = 1.3 pF  
 RX res = 53  
 TX res = 53  
 MB length = 12" (with 250mil BO)  
 Card length = 4" (with 250mil BO)  
 AC cap = 200 nF  
 MB Zo = High ~113 diff Z  
 Card Zo = High ~113 diff Z  
 De-emph = 3 dB  
 Swing = 800 mV  
 Edge rate = slow  
 Vias = 6  
 Driving direction = card Tx, MB Rx

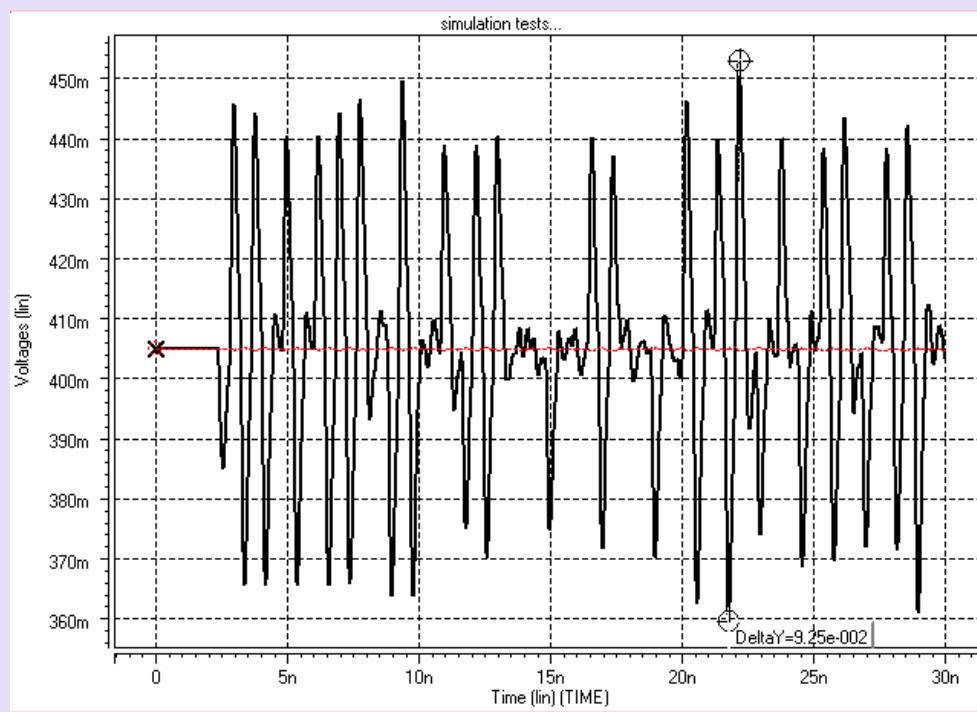
30% guard band for AC common mode (for fiber weave effects)  
 20% guard band otherwise

# AC common mode

- Max AC common mode @ Rx < 150 mV peak

✓  $V_{AC-cm} = |V_{D+} + V_{D-}| \div 2 - V_{DC-cm}$

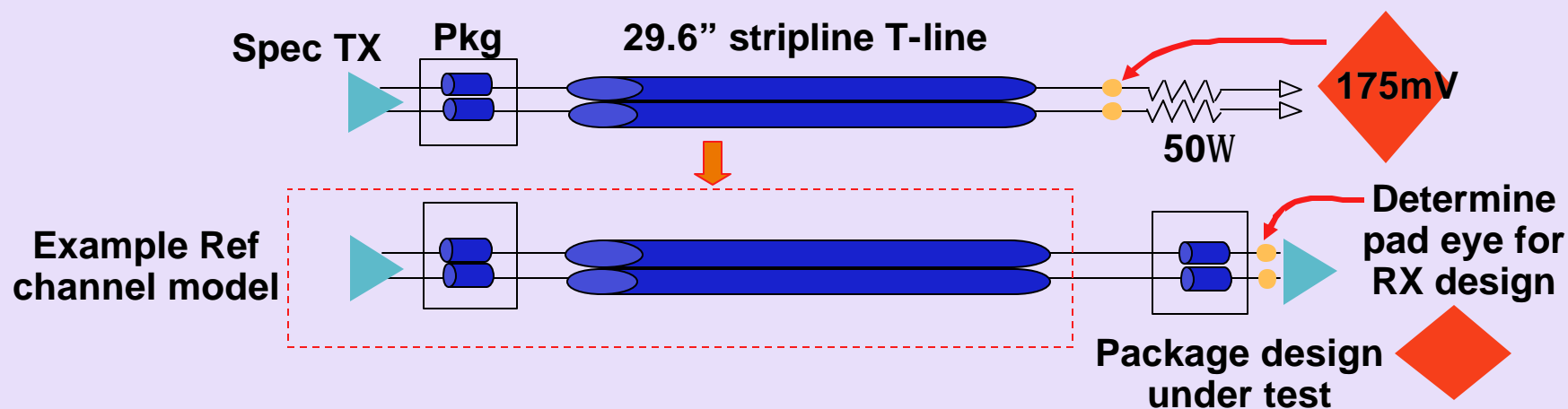
✓  $V_{DC-cm} = DC(avg) \text{ of } |V_{D+} + V_{D-}| \div 2$



**Example:  
50 mV  
peak**

# Ref channel for RX simulations

- Return Loss spec is **not** sufficient to guarantee RX operability
- Use Ref channel model for RX simulations
  - ✓ Calibrate T-line with  $\sim 13.2\text{dB}$  loss to  $175\text{mV}$  at  $50\Omega$  load
  - ✓ Use Spec TX (800mV swing,  $-3.5\text{dB}$  de-emphasis,  $0.3\text{UI}$  TX jitter)



# Board Design Summary

- PCI Express point-to-point layout is straightforward
- Manage loss and symmetry in PCB to meet interconnect budget
- Follow basic layout rules and design tradeoffs to implement typical topologies
- Perform simulations to maximize solution space for complex topologies

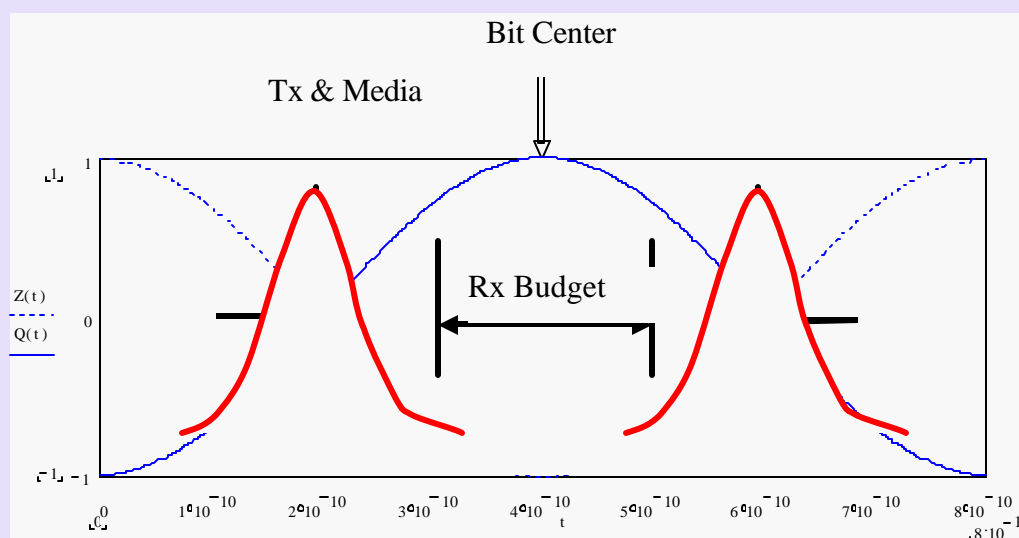
# Jitter

- Definitions
- Transfer Functions
- System Models
- Application of system models



# Bit Errors

- Data Communication is a sampled system
  - ✓ The receiver samples the data using a clock
  - ✓ Bit Errors occur at the receiver when it incorrectly samples the data
- Noise affects the sample
  - ✓ Noise is mainly crosstalk, reflections, thermal, etc
- The sample position affects the quality of the sample
  - ✓ The goal is to sample at the time of the highest signal
  - ✓ Budgets are established by the PCI Express Specification for the Tx, Media and the Receiver



# Units of Jitter

- Time and Phase are used interchangeably to quantify jitter
  - ✓  $T_{ideal}$  is the ideal period
  - ✓  $T_{measured}$  is the jittered sample

- Difference in time is:

$$\Delta T = T_{ideal} - T_{measured}$$

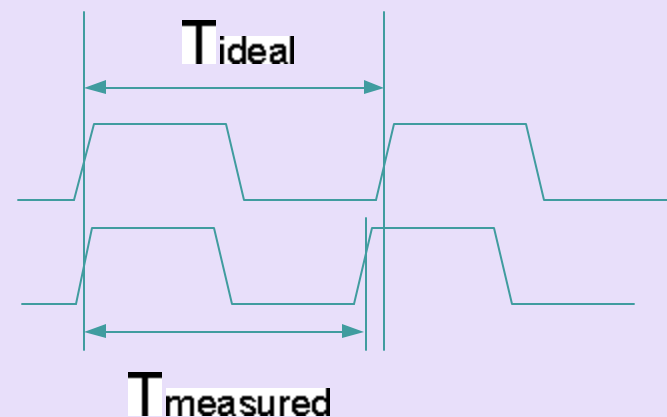
- Difference in phase is:

$$\Delta Phase = 2\pi - 2\pi \frac{T_{measured}}{T_{ideal}}$$

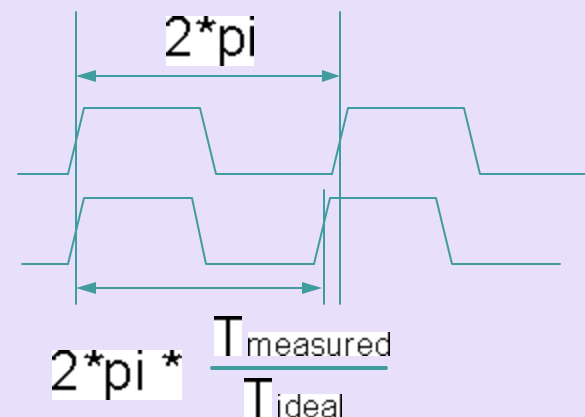
- The Conversion is simply:

$$\Delta Phase = \frac{\Delta T}{T_{ideal}} * 2\pi$$

Time:



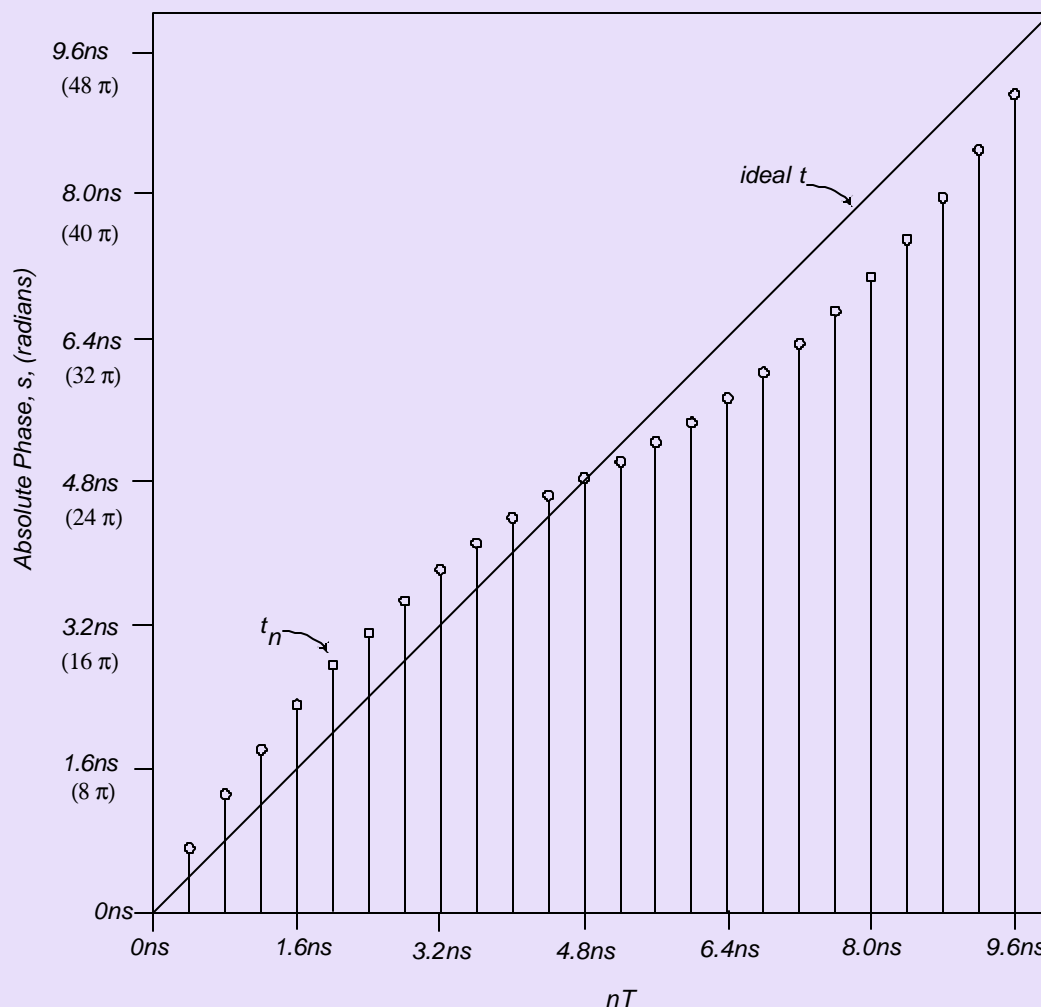
Phase:



# Measuring the Phase, $P(t)$

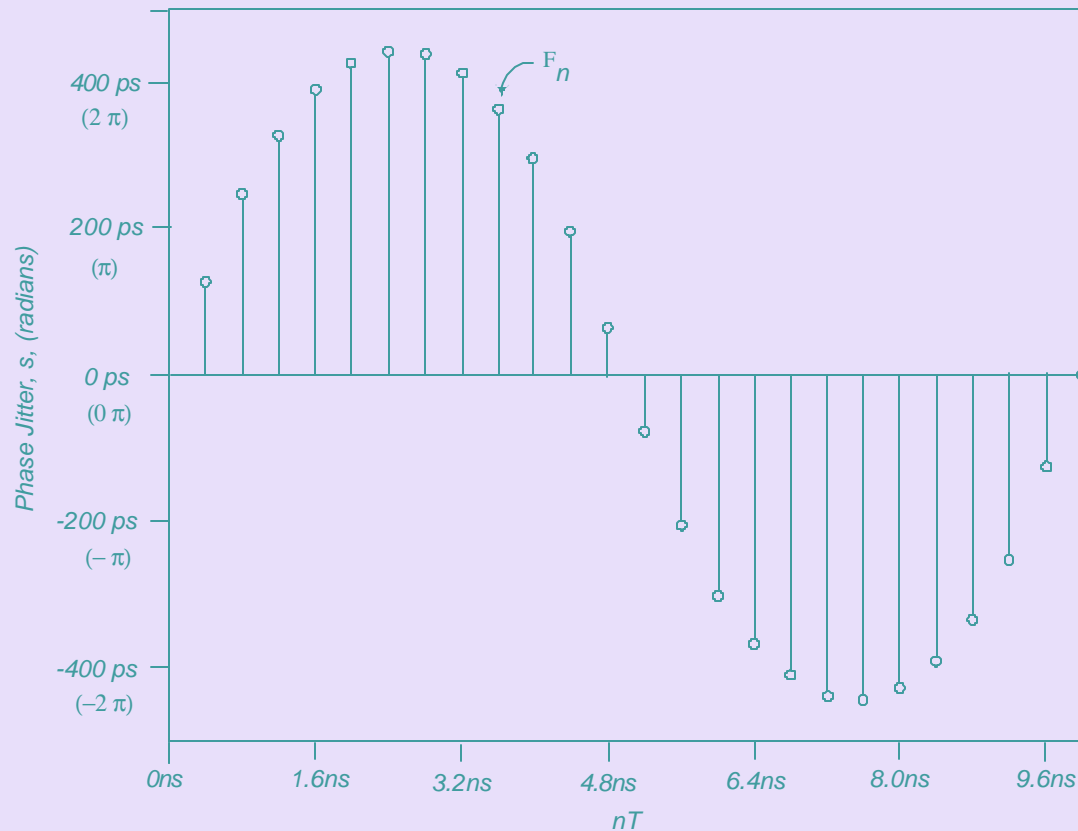
- Start by measuring the time intervals of the zero crossings of the differential signal
- $t_n$  has sinusoidal jitter added
  - ✓ Phase is  $n \cdot 2\pi$  plus a sinusoidal term
- Ideal "T" is the perfect clock
  - ✓ The phase is a multiple of  $2\pi$

$$\Theta_n = 2n\pi, \quad n = 0, 1, \dots, \infty$$



# Phase Jitter (F)

- Also known as phase noise and accumulated Jitter
  - ✓ Can be seen in the frequency domain by a phase noise plot.



$$\Phi_n = t_n - nT, \quad n = 1, 2, \dots, \infty$$

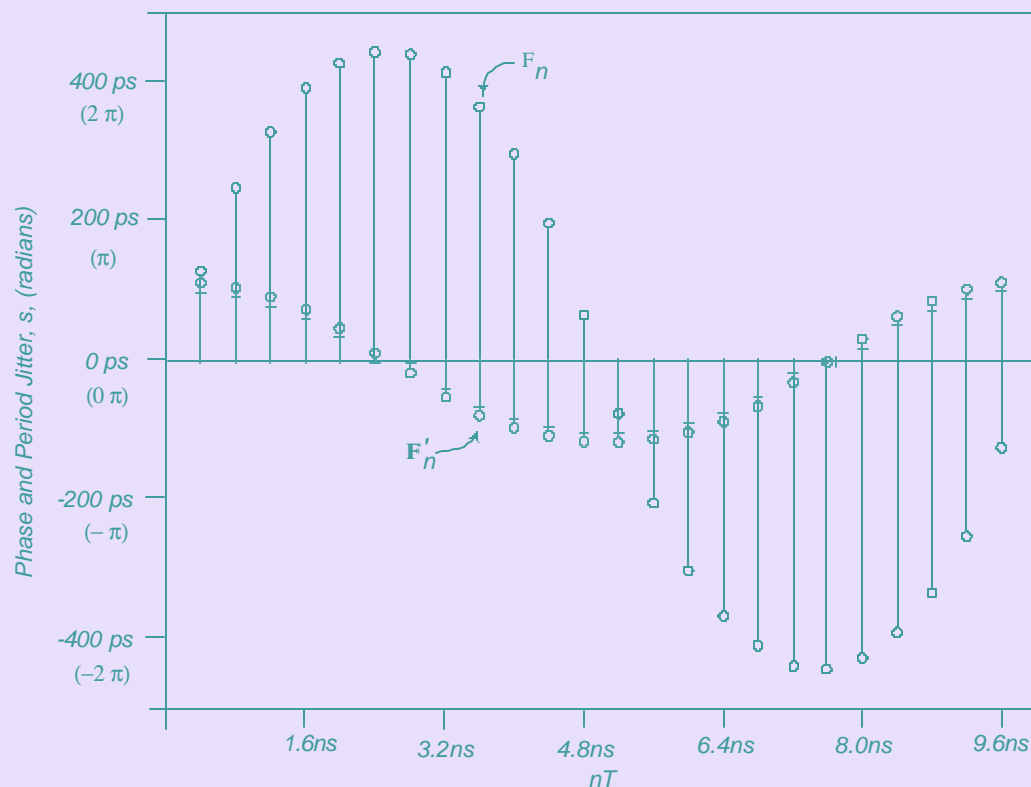
## ■ Period Jitter

- ✓ The period Jitter ( $\Phi'$ ) is the difference between the measured period and the ideal period

$$\Phi'_n = (t_n - t_{n-1}) - T, \quad n=1,2,\dots,N$$

## ■ Also is:

$$\Phi'_n = \Phi_n - \Phi_{n-1}$$



# Cycle to Cycle Jitter ( $F''$ )

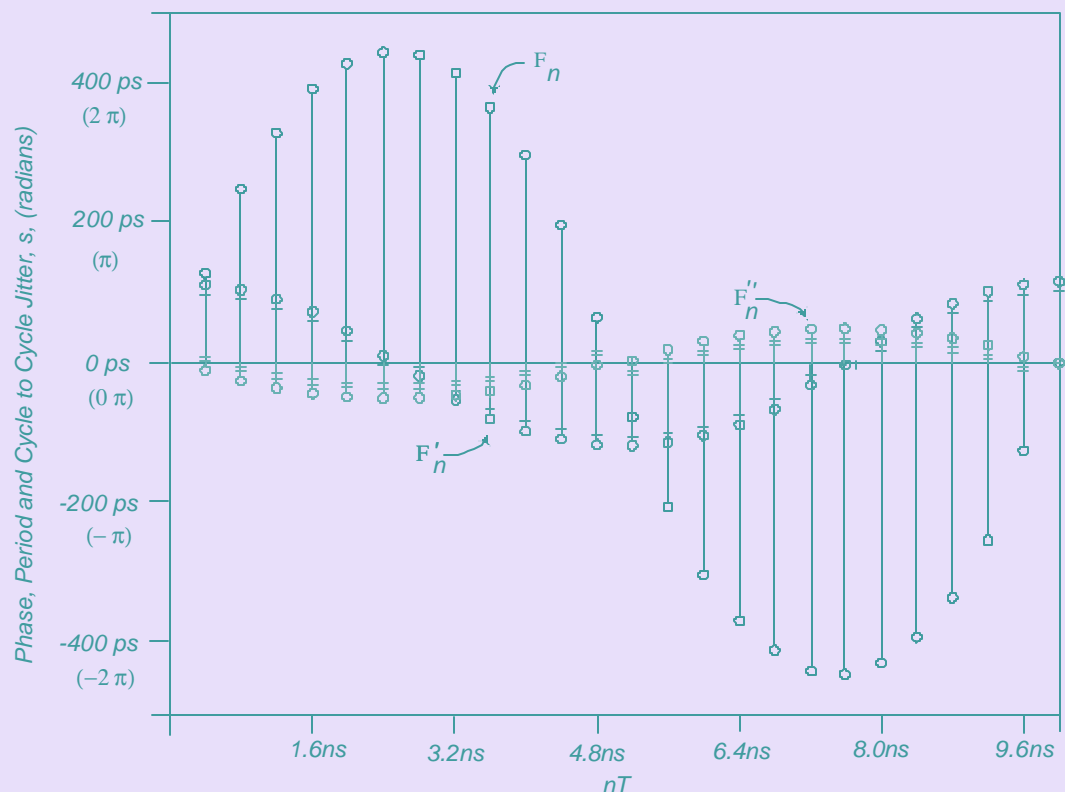
## ■ Cycle to cycle

- ✓ The difference between consecutive bit periods

$$\Phi''_n = (t_n - t_{n-1}) - (t_{n-1} - t_{n-2}), \quad n=1,2,\dots,N$$

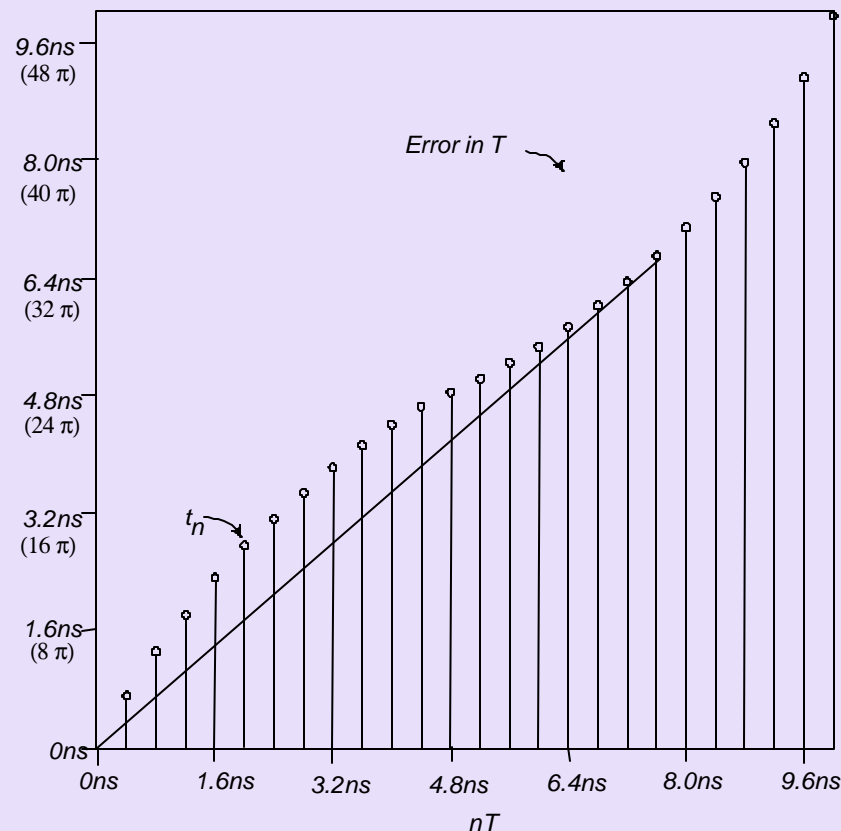
## ■ Also is:

$$\Phi''_n = \Phi'_n - \Phi'_{n-1}, \quad n=1,2,\dots,N$$



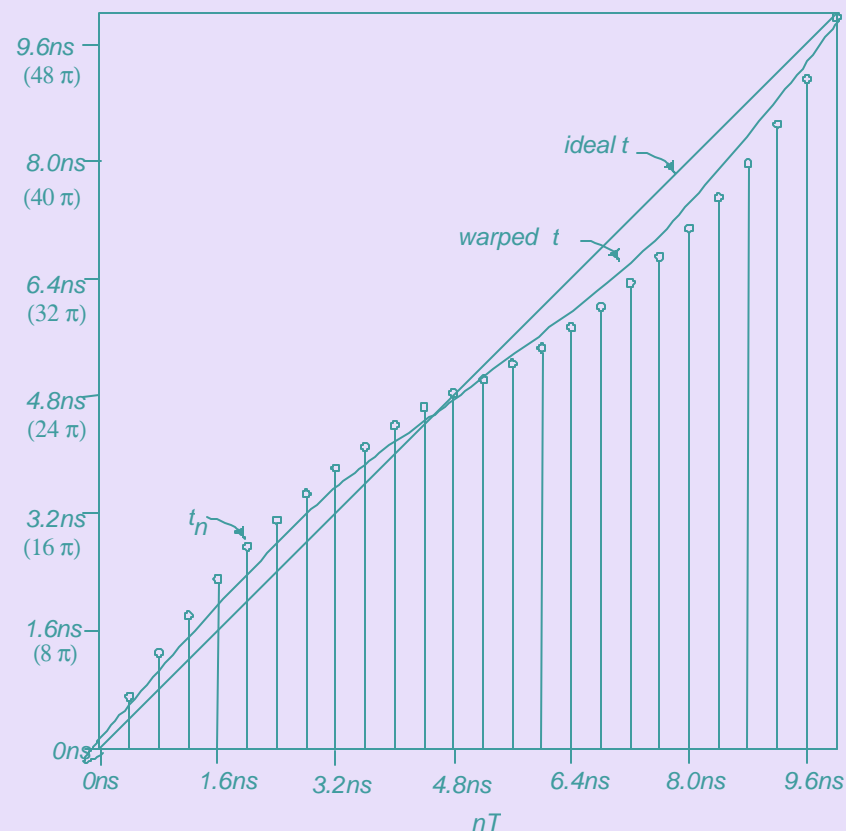
***Jitter types are different representations of the same data  
Be specific about the representation you are using***

- Estimating the ideal period is important
- A slight error in  $T$  causes an error
- Picture shows why an average over a small time is inaccurate
- Record length must be much longer than lowest frequency to average the error out
  - ✓ Spread Spectrum Clock (SSC) is a dominant component at  $\sim 10 \text{ ns} * \sin(2 * \pi * 33 \text{ KHz} * t)$
- Sufficient UI are required for calculating the ideal clock  $T$ 
  - ✓ Other methods like windowing are effective but beyond the scope of this document



# The Ideal Clock T

- Calculating Phase Jitter against a recovered clock gives a warping of the ideal clock
  - ✓ This reduces the amount of phase jitter calculated
  - ✓ The amount of warping depends on the recovered clock's transfer function
- As we shall see later, this follows the system behavior
  - ✓ Low frequency phase jitter is rejected by the recovered clock function



***The Clock Recovery Function Changes  
the Phase Jitter Measurement***



# Phase Transfer Functions of a PLL

- Ignoring harmonics, the input signal to a PLL is

$$V_{in}(t) = A_{in} \sin(\omega_{in}t + P_{in}(t))$$

where  $P_{in}(t)$  is the input phase signal as previously described and is the state variable of the PLL

- The output from a PLL is

$$V_{out}(t) = A_{out} \sin(\omega_{out}t + P_{out}(t))$$

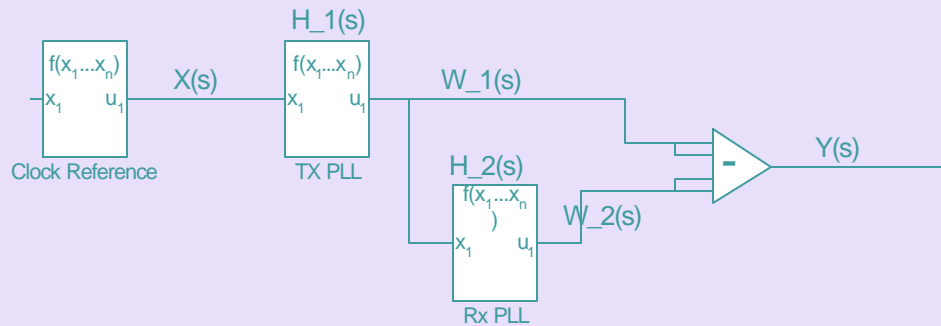
where  $P_{out}(t)$  is the output phase.

- The PLL has a phase transfer response,  $h(t)$ , given by

$$P_{out}(t) = h(t) \otimes P_{in}(t) \quad \text{Laplace Transform} \rightarrow \quad H(s) = \frac{P_{out}(s)}{P_{in}(s)}$$

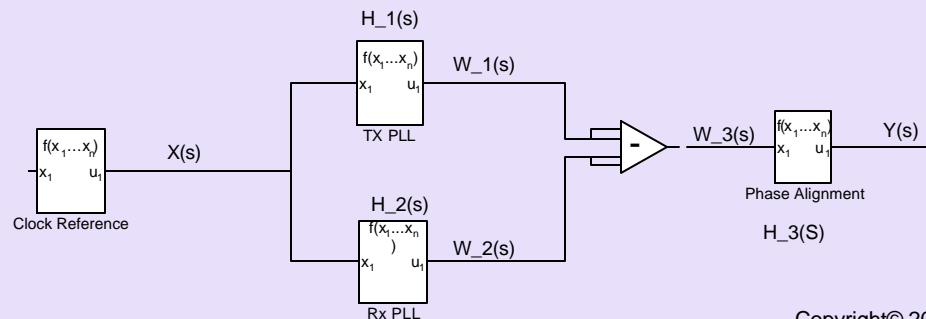
# System Math Models

## PLL Based DRC Model



$$H_t(s) = H_1(s)(1 - H_2(s))$$

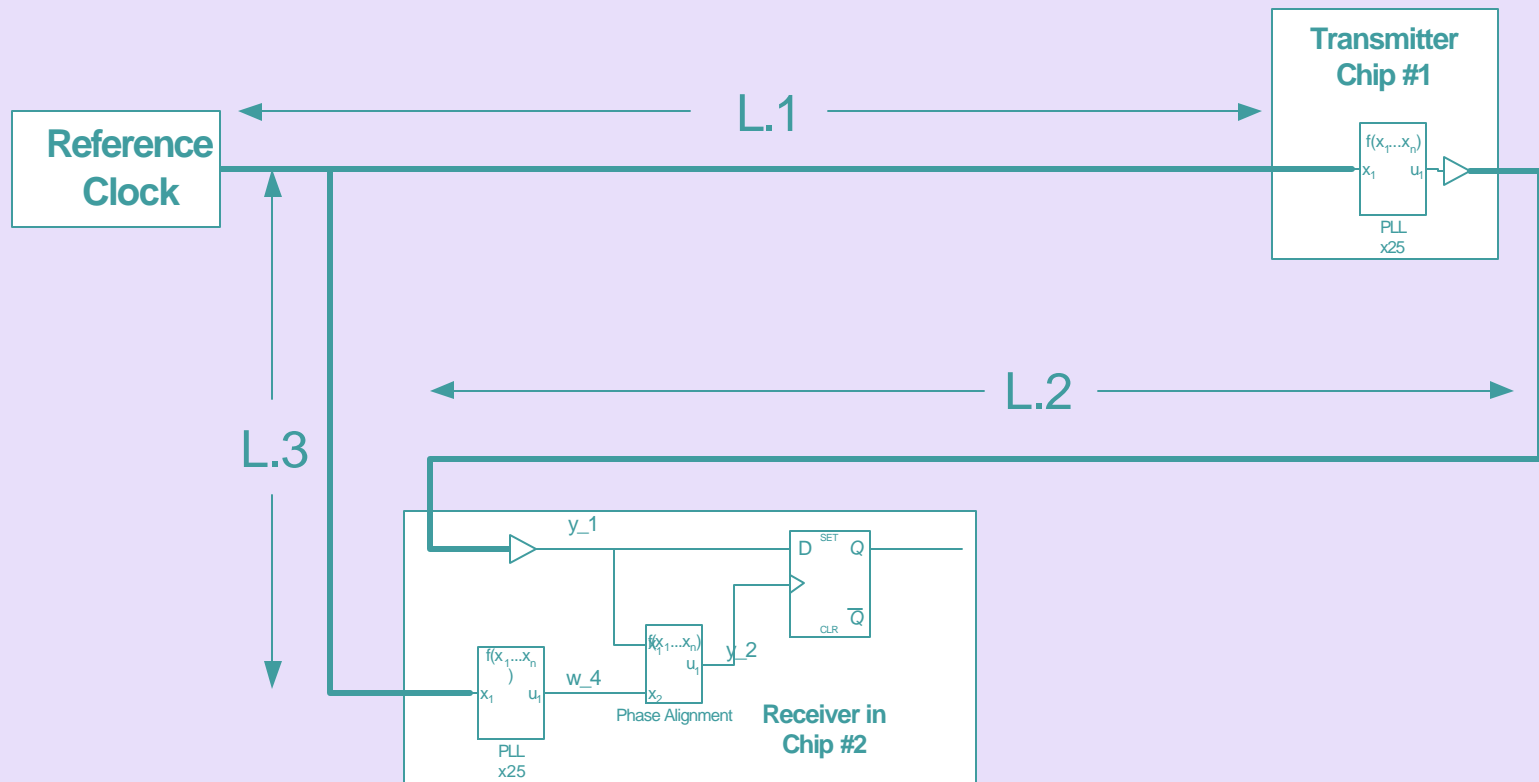
## Digital DRC System Model



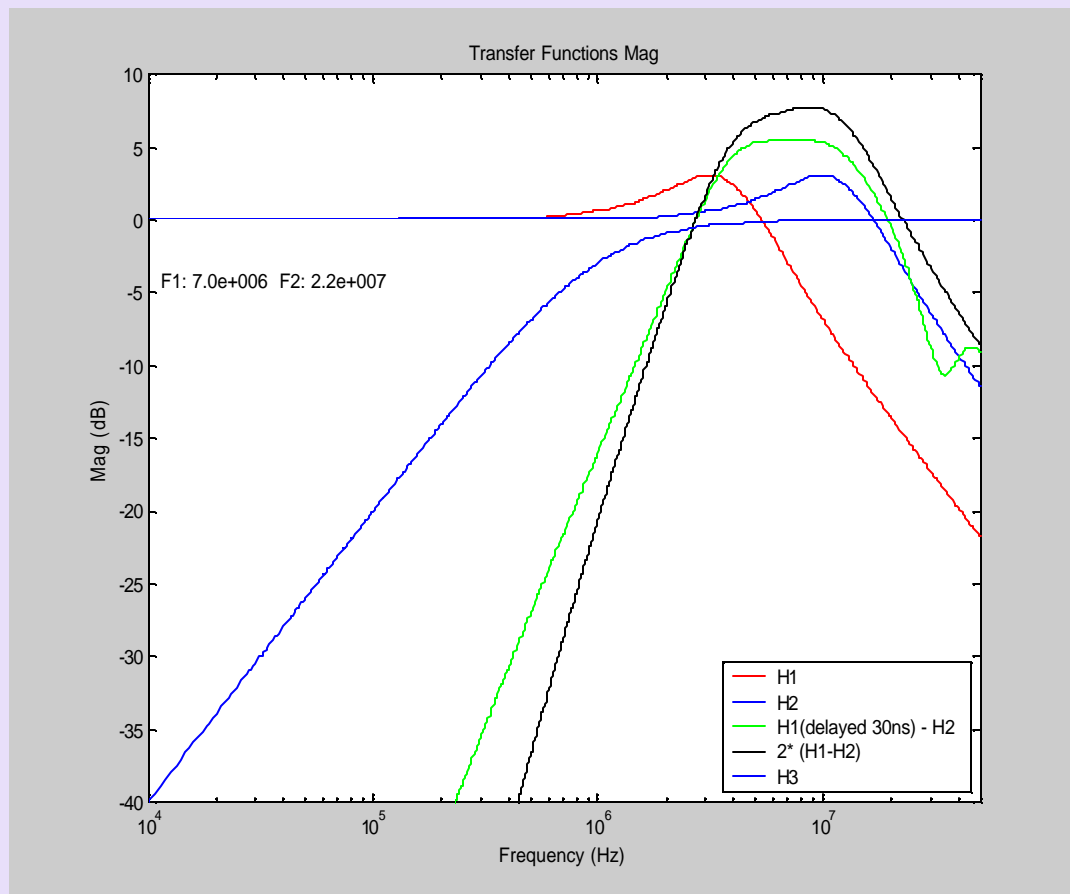
$$H_t(s) = (H_1(s) - H_2(s))H_3(s)$$

# Delay Model

- Delay transfer function models into the system
  - ✓ The phase delay is given by  $(L1 + L2) - L3$
  - ✓ This is modeled by  $\exp(-s * t_{\text{delay}})$



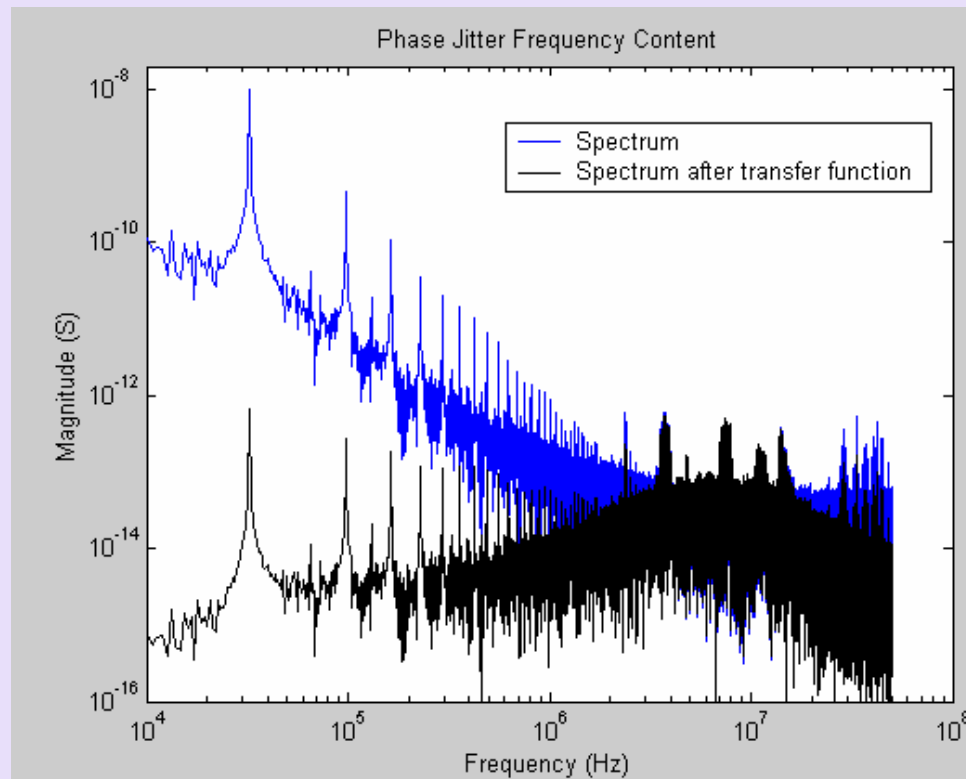
- The difference function includes an arbitrary phase delay not to exceed 30 ns
  - ✓ The delay “un-correlates” the clock/data and closes the eye
- This can be estimated with 2x multiplier of  $H_1(s) - H_2(s)$



$$H_t(s) = 2 [H_1(s) - H_2(s)] H_3(s)$$

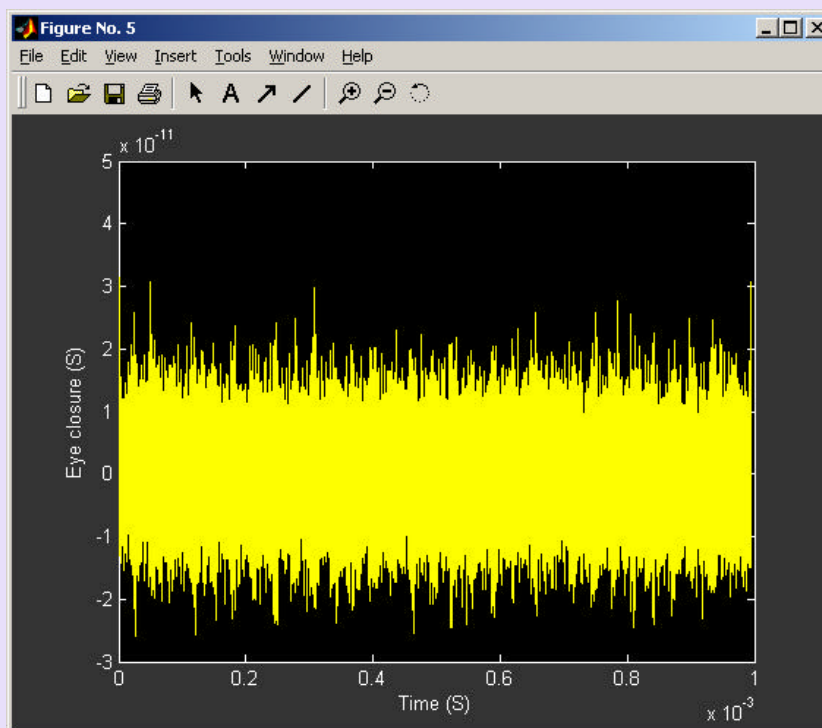
# Model Applied to Measured Data

- $X(s)$ 
  - ✓ The input spectrum can be measured by measuring a clock and applying the jitter definitions to get the phase jitter
  - ✓ The phase jitter signal is then transformed into the phase jitter spectrum,  $X(s)$
- $H(s)$ 
  - ✓ The models presented here give  $H_t(s)$ , the bounding function, including the delay
- $Y(s)$ 
  - ✓  $Y(s) = X(s) * H(s)$



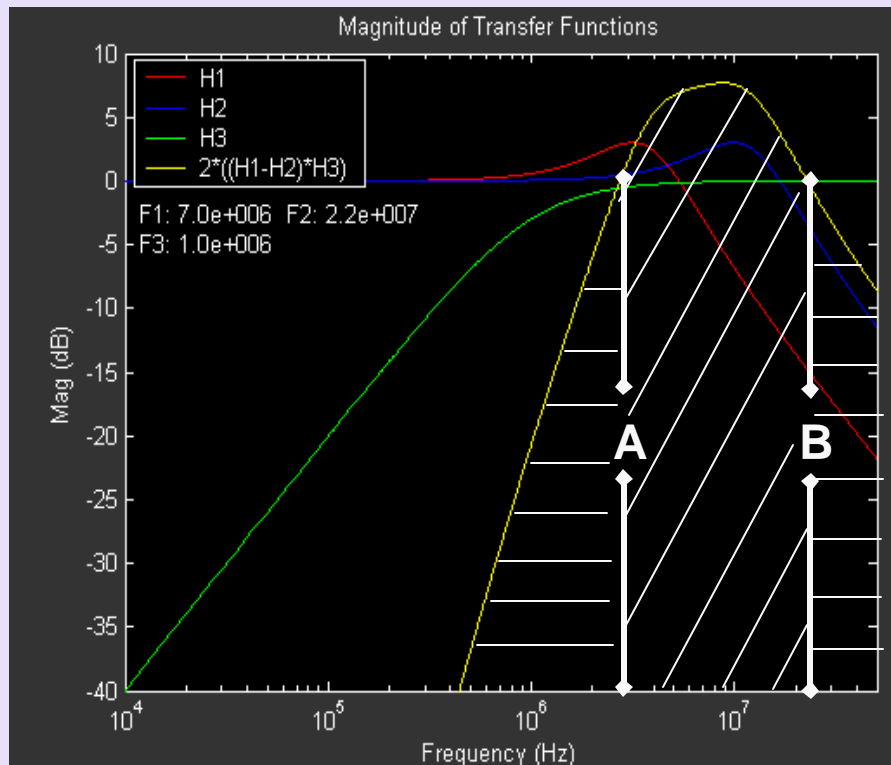
# Eye Closure at the Sampling Flop – One Method

- The inverse transform of  $Y(s)$  gives the eye closure in the time domain,  $y(t)$



# Multiple Variable Problem

- Difference function dominated by Tx, Rx and Phase Interpolator
  - ✓ PLL bandwidth difference
  - ✓ PLL peaking
- Reference clock noise spectrum as seen at Tx and Rx
- Any noise under the difference function can cause phase jitter



- **3 Danger Areas (7-22 MHz shown)**
  - ✓ Between "A" and "B" ref clock jitter gets multiplied
  - ✓ Left of "A" rolls off
  - ✓ Right of "B" rolls off
- **Changing Tx, Rx below 7 MHz increases width of danger areas**
  - ✓ Width and Amplitude
  - ✓ Impact varies with clock spectrum

# Design Checklist

- ✓ Understand and control the noise in the sensitive clock area
- ✓ Select the appropriate clock source to minimize channel jitter for your design



# 3 Stages of Serial Link Physical Layer Validation

## 1. Electrical Specification Compliance

- ✓ Measure timing and voltage margins, and compare against spec.
  - Primary Tool – Measurement Equipment
  - Failing Symptoms - Link won't train reliably, failing Loopback. Master tests through a Loopback. Slave, and the Link should be extremely flakey.

## 2. Link Stress Link Validation

- ✓ Look for intermittent or functional failures during system stress while testing all Link Features
  - Primary Tool – Logic Analyzer and Config Registers
  - Failing Symptoms – Periodic Link failures, Link Retraining, Link Recovery.

## 3. Functional BER EYE Margining Validation

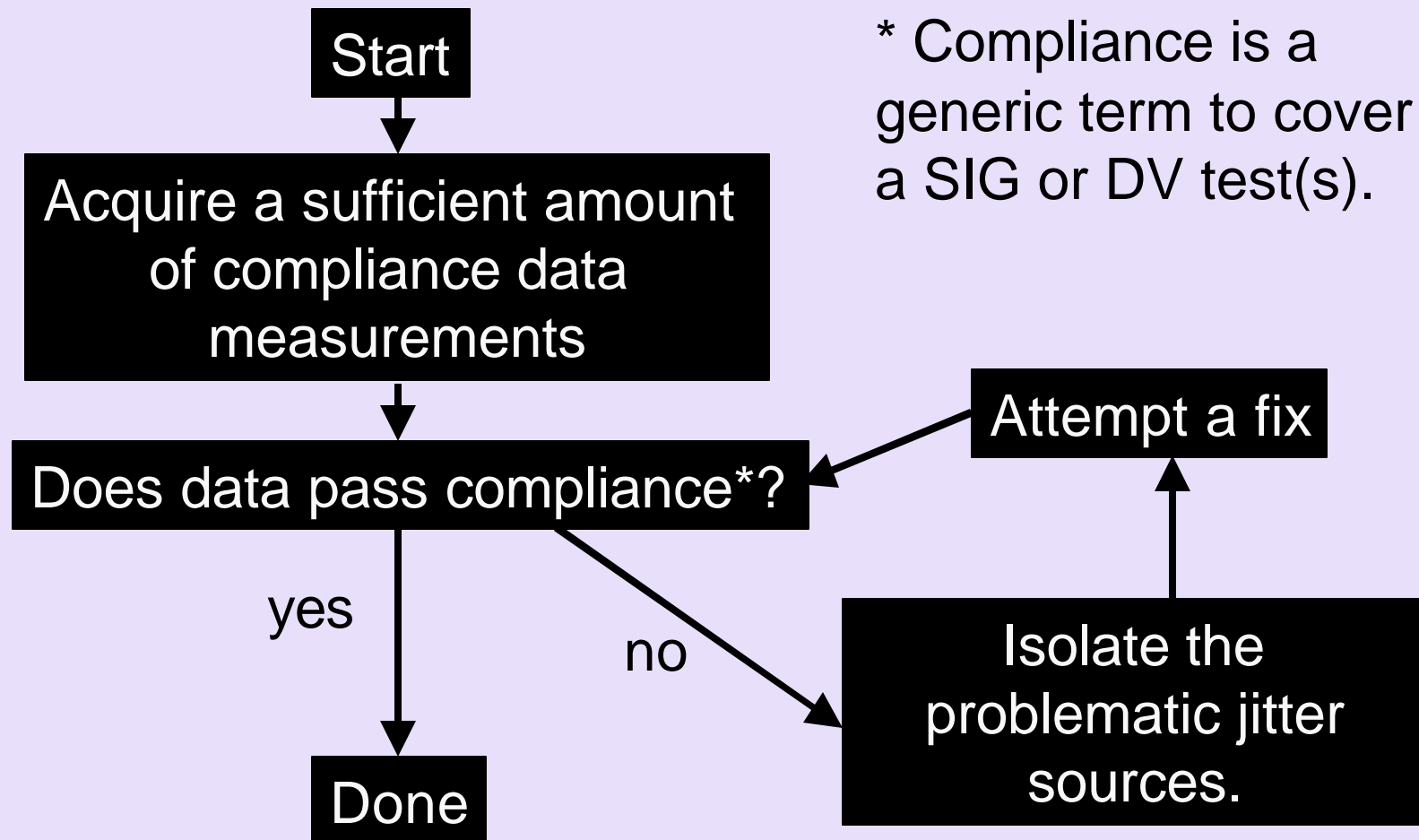
- ✓ Validate the Link has necessary timing and voltage margin as measured by BER
  - Primary Tool – Functional Loopback Applications
  - Failing Symptoms – Doesn't Pass required BER mask.

**This presentation focuses on step 1 only**

# Electrical Validation Key Messages

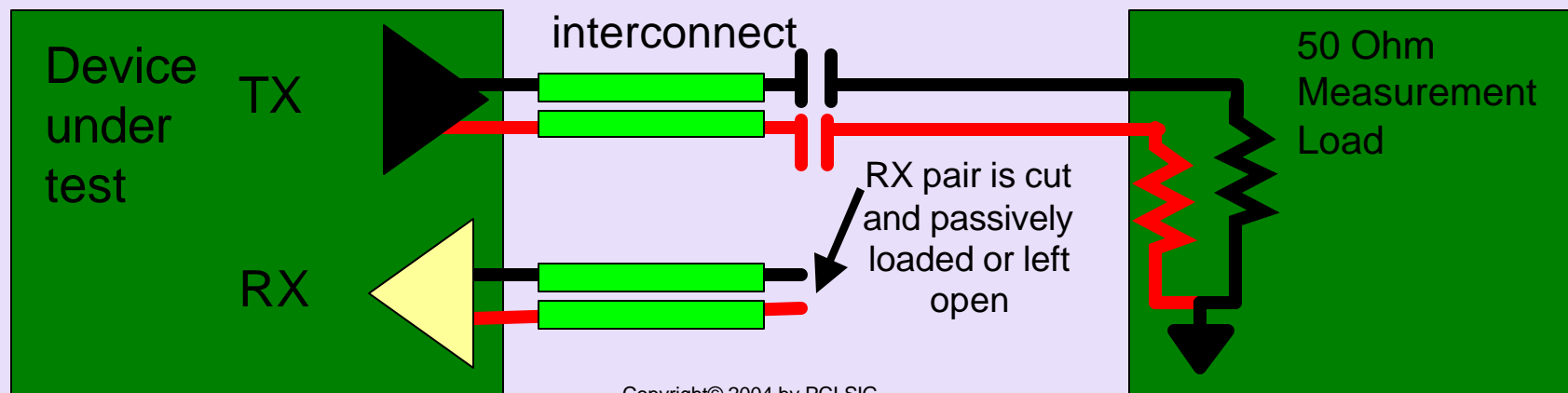
- Electrical Validation requires recovering a clock.
  - ✓ No strobe to compare the data against since the clock is embedded.
  - ✓ The recovered clock model can be made to look like the RX clock recovery.
- Any serial data measurements without explicitly listing the recovered clock model are meaningless!
  - ✓ Measurements on the same data can vary by >>2x.
  - ✓ Currently not all scopes provide this data readily.
- Electrical Measurement need to be close to the first validation test performed.
  - ✓ PCI Express Links can hobble along functionally due to error correction with out of spec electrical parameters.
  - ✓ Product validation efforts can go on for weeks chasing logical bugs when the real problem is jitter.

# General Electrical Validation Flowchart



# Electrical Validation Setup

- Measure Electrical Parameters Using Compliance Pattern and a 50ohm measurement equipment
  - ✓ The PCI Express specification requires that all devices output the compliance pattern for EV.
    - The compliance pattern is transmitted by simply Connecting 50 Ohm Probes to one of the TX pairs while making sure a corresponding RX pair stays quiet.



# Electrical Validation Setup

- Two Electrical Validation Components are available through the PCI SIG to simplify measurements\*.
  1. Compliance base board – Allows for testing of add in cards at the connector (upstream devices).
  2. Compliance add in card – Allows for testing of motherboard devices (downstream devices).
- Components can be ordered from PCI SIG website at [http://www.pcisig.com/specifications/pciexpress/compliance/compliance\\_library](http://www.pcisig.com/specifications/pciexpress/compliance/compliance_library)

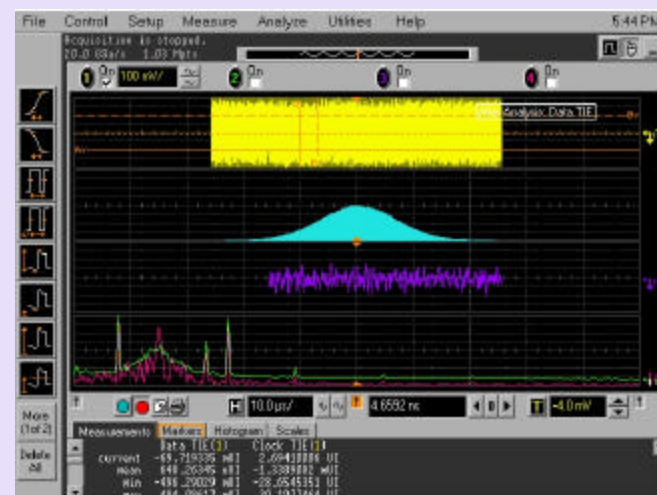
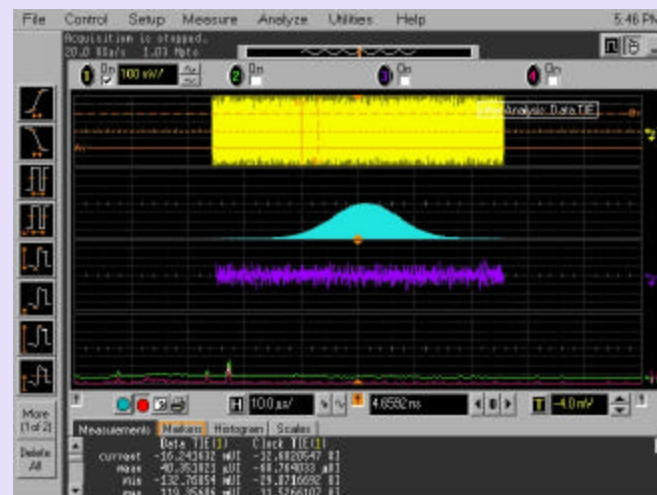
\* - Currently applies only to the CEM form factor.

# Different Clock Recovery Models Change the Results

Pk to Pk Jitter = 100ps

- Using the same raw data 2 jitter measurements using 2 different clock recovery models show a 4x difference.
  - ✓ Both are correct!

Pk to Pk Jitter = 394ps



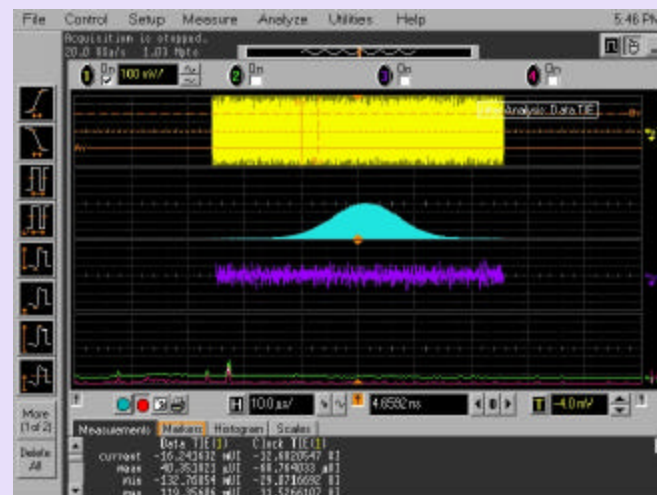
# The Recovered Clock Acts Like a High Pass Jitter Filter On the Data

- Voltage and timing information above the recovered clock cutoff frequency is included in the measurement
- Voltage and timing information below the recovered clock frequency is attenuated

# Example of how the Recovered Clock Used Changes the Data.

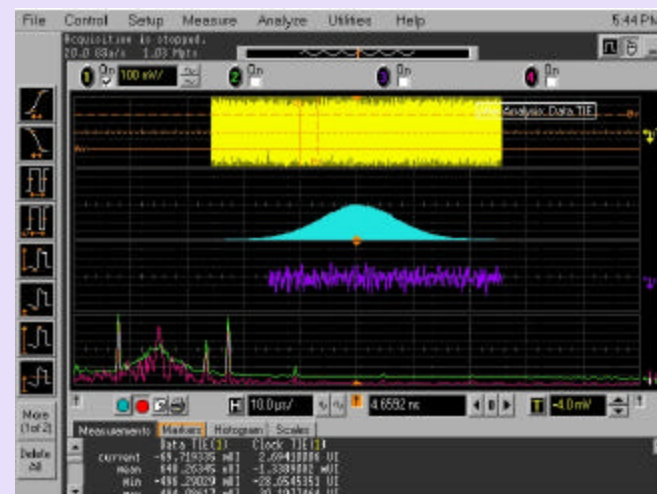
PLL Recovered Clock model  
10Mhz cutoff with a  
20db/dec rolloff

Pk to Pk Jitter = 100ps



PLL Recovered Clock model  
500Khz cutoff with a  
20db/dec rolloff

Pk to Pk Jitter = 394ps





# How do we test a component with the PCI Express Clock Recovery Model?

- There is free PCI Express software to test for compliance, and some scopes are also adopting this model of clock recovery
  - ✓ Software can be obtained from the PCI SIG website
  - ✓ Some scope manufacturers are building the compliance check into their hardware.

# How do we test a component with the PCI Express Clock Recovery Model?

- There is free PCI Express software to test for compliance, and some scopes are also adopting this model of clock recovery.
  - ✓ Software can be obtained from the PCI SIG website
  - ✓ Some measurement manufacturers are building the compliance check into their hardware.

## **Need to Continue to Work with Measurement Equipment Manufacturers**

- It absolutely necessary for us to have all measurement equipment to be able to allow the user to control and understand the clock recovery models being used in processing the measured data.

# Electrical Compliance Conclusions

- Electrical Validation requires recovering a clock before any measurements are made.
  - ✓ One must understand the clock recovery model to understand the data.
- Compliance Boards and Software is available at [http://www.pcisig.com/specifications/pciexpress/compliance/compliance\\_library](http://www.pcisig.com/specifications/pciexpress/compliance/compliance_library)

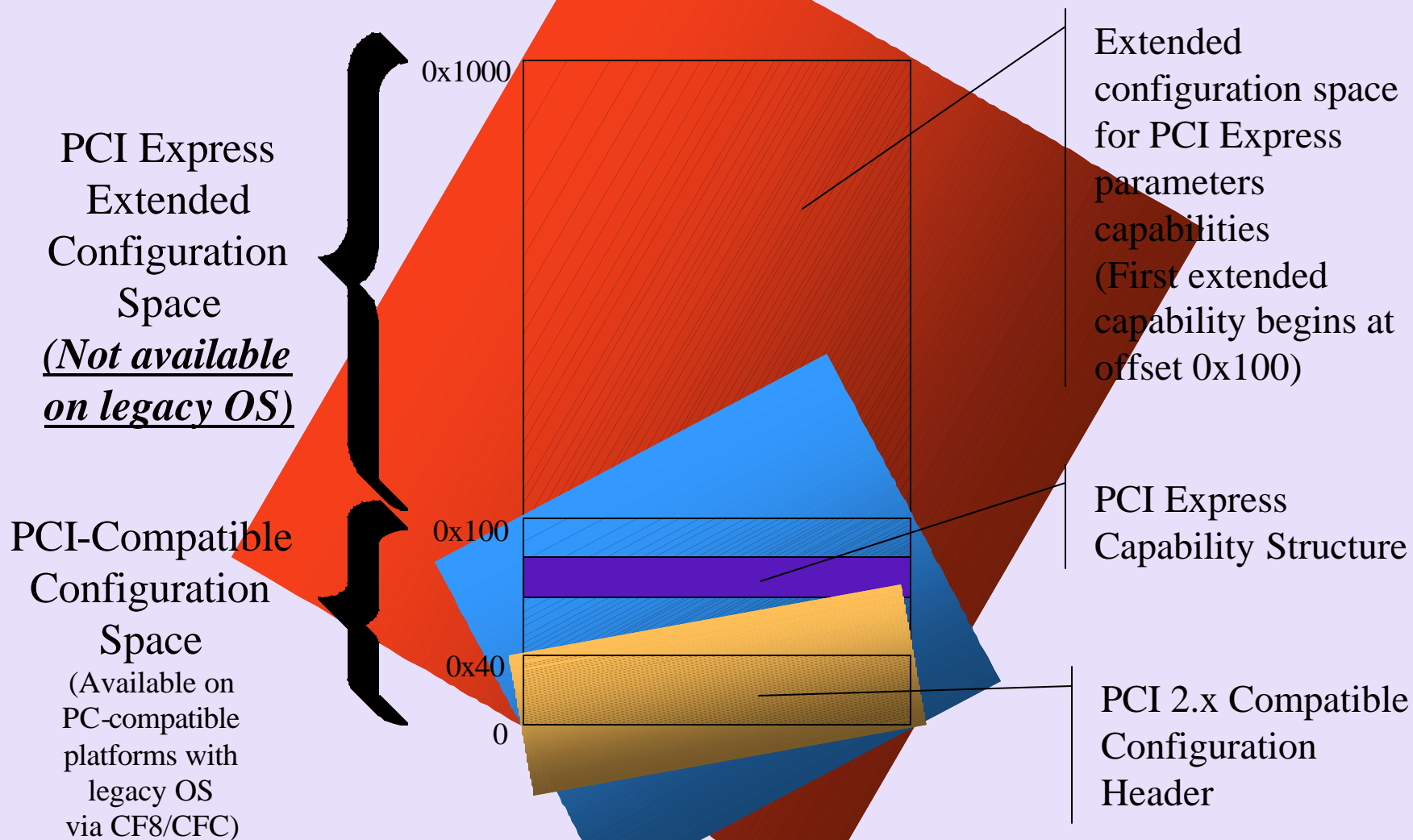
# Agenda

9:00	PCIe Overview	<i>Neshati</i>
9:45	PCIe Physical Layer	<i>Schoenborn</i>
12:00	LUNCH	
1:00	<b>PCIe Configuration &amp; Software</b>	<i>Cowan</i>
2:45	BREAK	
3:00	PCIe Compliance Tools	<i>Choate/Froelich</i>
5:00	Q&A	
5:15	BREAK	
5:30	Compliance Workshop Orientation	<i>Neal/Kelley, et al</i>

# Getting The Fundamentals Right...

- Configuration Space
- Interrupts
- Power Management
- Power Negotiation
- Boot Considerations
- Robustness Guidelines

# Configuration Space



# Configuration Space – Key Points

- PCI-Compatible and Extended Config Space
- Do not rely on Extended Config space to be available in legacy environments
  - ✓ **Extended Configuration Space access may not be available in legacy OS scenarios**
  - ✓ **If access to Extended Configuration Space elements is really needed, design for aliasing elements through a BAR or PCI-Compatible Configuration Space region**
- Note that registers critical to device functionality are all located in PCI-Compatible Configuration Space
  - ✓ **PCI Express Capability Structure located below 256 bytes**



# Configuration Space – Key Points

- Specification defined capabilities are for system software use
  - ✓ **Device-specific software should rely on OS services for access to these registers**
  - ✓ **Writes to spec-defined capabilities are not recommended for software other than system software (firmware / OS)**
    - Improper use can cause upgrade problems during OS migration
- Recommend that device-specific registers be located in BAR regions

# Interrupts

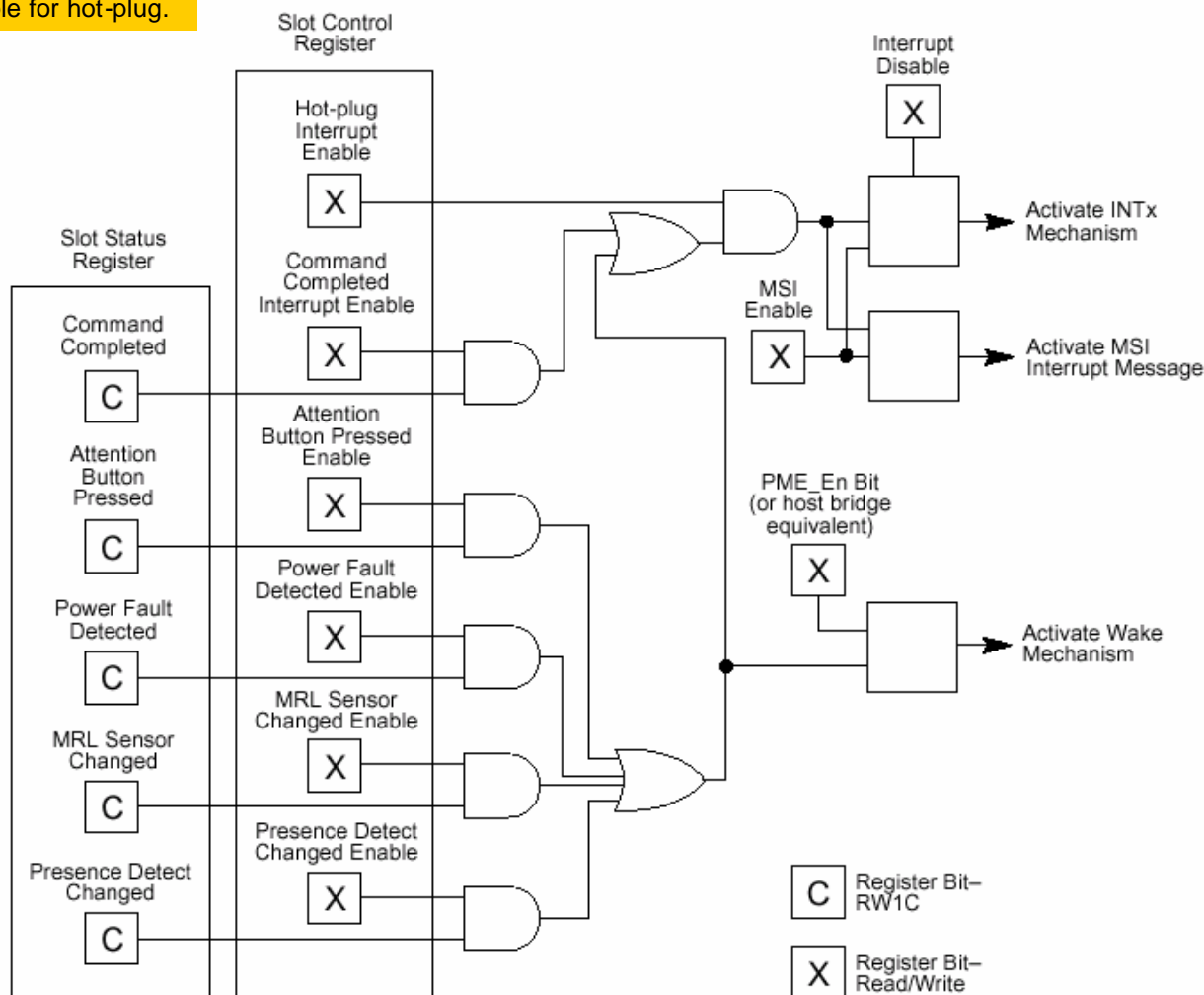
- PCI Express supports three interrupt mechs:
  - ✓ INTx: level triggered
  - ✓ MSI: edge triggered
  - ✓ MSI-X: edge triggered; newly added via ECN
  
- INTx, MSI, & MSI-X are mutually exclusive
  - ✓ Enabling MSI or MSI-X disables INTx
  - ✓ SW is prohibited from enabling MSI & MSI-X concurrently
  - ✓ MSI/MSI-X are not controlled by (INTx) interrupt disable bit
  - ✓ MSI/MSI-X messages are memory write requests and can be disabled by clearing the BME (Bus Master Enable) bit
  - ✓ MSI/MSI-X/INTx interrupts can only be signaled in D0 state

# Interrupts

- Level triggered interrupts can result in interrupt storms
  - ✓ **Especially high risk in virtual wire scenarios if de-assert messages not sent correctly**
  - ✓ **Be sure to:**
    - De-assert INTx in low-power states
    - De-assert INTx when source becomes masked
    - De-assert INTx when interrupts become disabled
  - ✓ **Asserts / De-asserts must be sent in pairs**
- Switches must synthesize de-asserts as necessary
  - ✓ **For example, if attached device is surprise removed**

# Interrupts: Design Example

**NOTE:** Example intended to illustrate device interrupt logic. Do not use example for hot-plug.



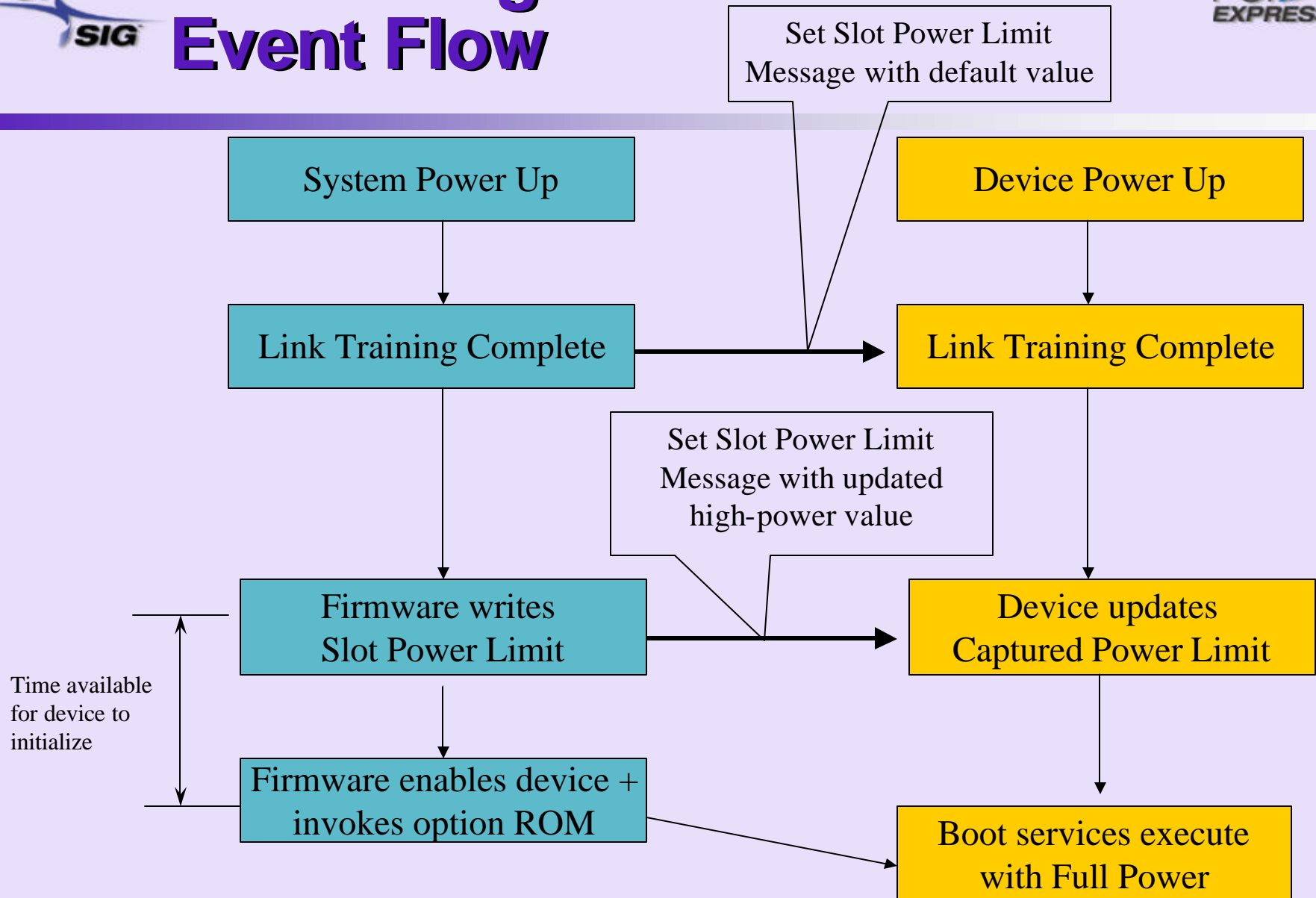
# Power Management Considerations

- Devices cannot source memory on IO requests when in a non-D0 state
  - ✓ Remember to de-assert any pending INTx interrupts when transitioning out of D0 to a low-power state
- PME\_Turn\_Off message can be received at any time
  - ✓ Not just in non-D0 states
  - ✓ Simply indicates to device that power and clocks are going to be removed
    - For example, can be received on system shutdown
- PME sequence is a two-stage process
  - ✓ Two-stage process:
    - Wakeup is separated from request for service
  - ✓ Wakeup is responsible for getting clocks and power
  - ✓ Service is requested through PME message

# Power Negotiation

- Slot Power Notification mechanism
  - ✓ Flexible and scalable mechanism to notify a PCI Express adapter of additional available slot power
- Conditions that trigger the message:
  - ✓ On a Configuration Write to the Slot Capabilities register (when the Data Link Layer reports DL\_Up status)
  - ✓ Anytime when a Link transitions from a non-DL\_Up status to a DL\_Up status

# Power Negotiation: Event Flow



# Power Negotiation Considerations

- Devices strongly encouraged to provide minimum level of functionality within the CEM specification power
  - ✓ **Driver or Option ROM can detect available power through captured power limit registers**
- Power Budgeting Extended Capability can allow flexibility in power redistribution
  - ✓ **Implementation specific dynamic power redistribution**
  - ✓ **Devices are strongly encouraged to implement this capability**



# Other Considerations and Robustness Guidelines

- Don't rely on access to extended configuration space for PCI compatible device operation
- IO BARs can be closed by system software for native PCI Express devices
  - ✓ Device may support an IO BAR for booting in legacy environments
  - ✓ But... Native PCI Express device must alias IO resources through MMIO as per PCI Express specification
- Do not hang on unexpected TLPs
  - ✓ For example, power changes or hot-plug messages
- Do not assume that TLPs can only arrive in certain states
  - ✓ For example, PME\_Turn\_Off
- Not sending de-asserts correctly for INTx can cause interrupt storms

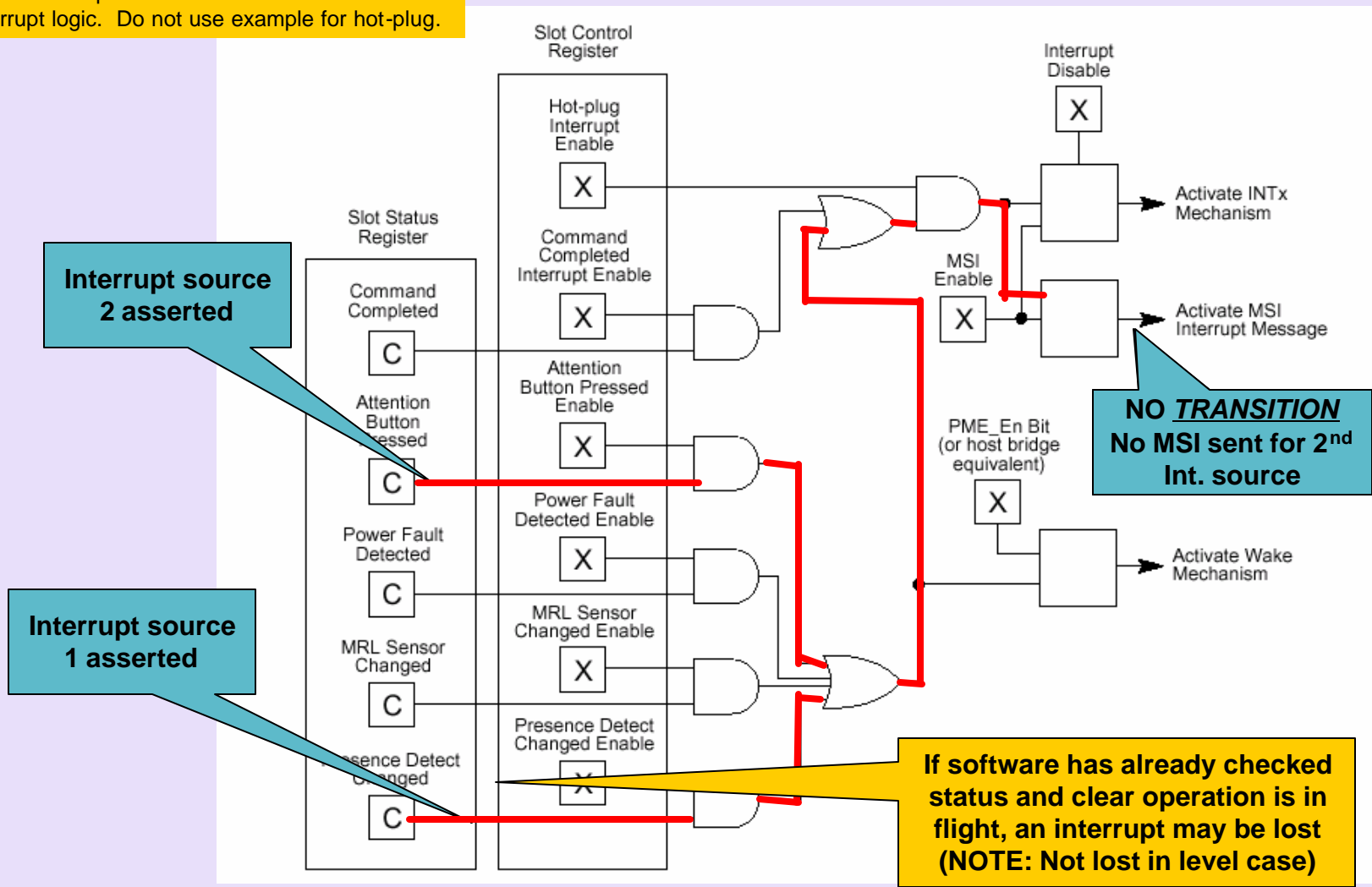
**Legacy software compatibility and transition strategy to PCI Express aware software environment must be considered at all times**

# Designing for MSI/MSI-X

- MSI/MSI-X Gotchas: Losing interrupts
- How not to lose interrupts?
- MSI/MSI-X Benefits and Optimizations
- MSI-X Advantages & Disadvantages over MSI

# MSI Edge-Triggered Interrupts Problems w/ Simple Wire-OR

**NOTE:** Example intended to illustrate device interrupt logic. Do not use example for hot-plug.



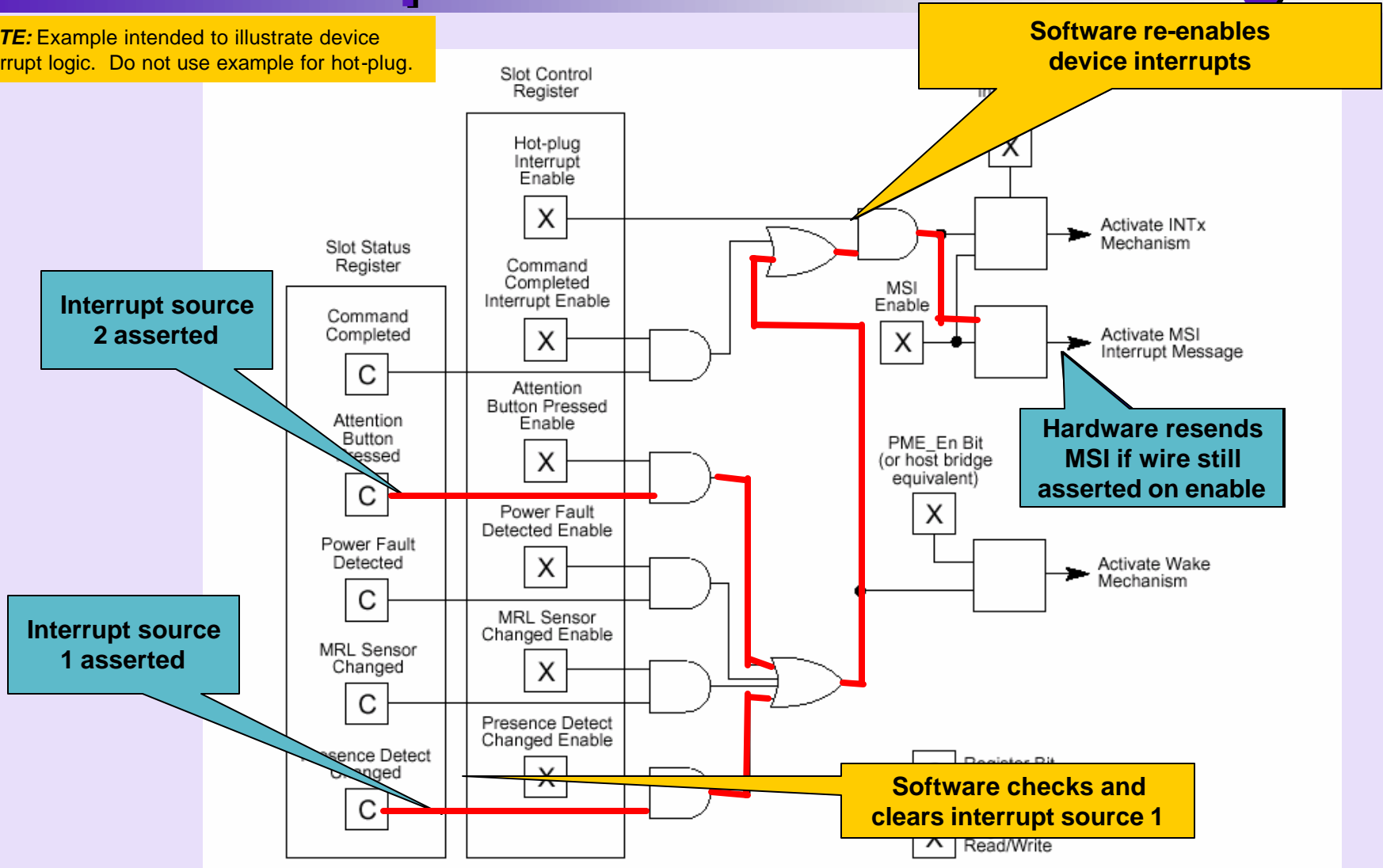
# Avoid Losing Interrupts

- Existing design wire-OR signals cannot be simply converted to edge at wire-OR output
- Simplest fix requires both hardware and software to be carefully designed
- Device must evaluate internal line status when interrupts are masked/disabled; this can be via:
  - ✓ The various mask bits defined by the MSI-X ECN for PCI 2.3
    - Optional per-vector mask (PVM) bits for MSI
    - Mandatory per-vector mask (PVM) bits for MSI-X
    - Mandatory function mask bit for MSI-X
  - ✓ A device-specific mask or enable bit
  - ✓ But NOT the “Enable” bit in the MSI or MSI-X Cap Structure
- Device (re)sends any message on interrupt unmask if status (or wire-OR status) still set
- ISR must be designed correctly

# MSI Edge Triggered Interrupts

## Example Solution via Masking

**NOTE:** Example intended to illustrate device interrupt logic. Do not use example for hot-plug.



# Representative MSI ISR

```
Device_MSI_ISR()  
{  
    MaskInterrupts();  
    CheckAndClearSources();  
    // Clear only if status set for given source  
  
    ServiceSources();  
    // Or alternatively queue for deferred  
    // processing  
  
    UnmaskInterrupts()  
    // Hardware must re-evaluate and resend any  
    // pending interrupts  
}
```

# MSI / MSI-X Benefits and Optimizations

- Possibility of interrupt storms is greatly reduced
- MSI/MSI-X messages are memory write transactions, pushing data on the same VC
  - ✓ **PIO read status checks can be avoided in many cases if using exclusive unshared interrupts**
- Multiple MSI/MSI-X vector usages
  - ✓ **Allows dedicated interrupt service routines for internal device sources**
    - For example, separate reader/writer threads
    - Dedicated (non-shared) ISR does not have to check possible interrupt sources prior to servicing
  - ✓ **Different MSI/MSI-X vectors can target different processors for interrupt load balancing or processor affinity**

**MSI/MSI-X have major advantages, but correct interrupt handling to avoid loss of edge-triggered interrupts is required**

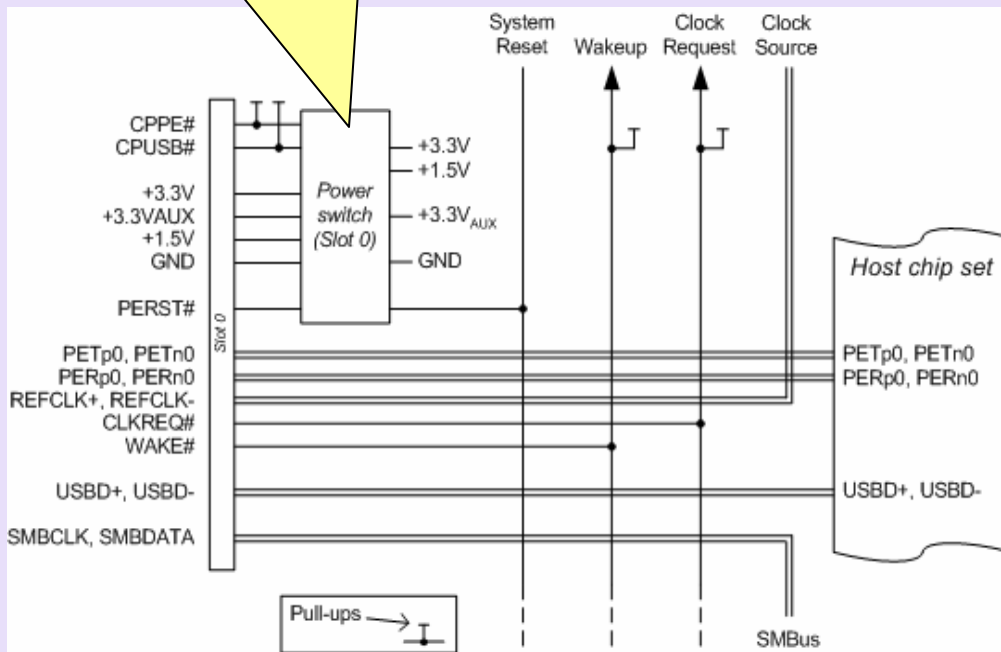
# MSI-X Advantages & Disadvantages over MSI

- MSI-X Advantages
  - ✓ Supports arbitrary address/data values for each vector
    - **All MSI vectors on a given function use the same address, which on some platforms prevents targeting different processors**
    - Only LSBs in the data value are used to distinguish vectors
  - ✓ Supports up to 2048 vectors per function vs 32 for MSI
  - ✓ Supports a new paradigm, where the driver determines how many vectors to request from the system
  - ✓ Supports assignment of vectors via architected mechanism when fewer are allocated by the system than requested
- MSI-X Disadvantages
  - ✓ Additional complexity, notably MSI-X structures in memory space
- Recommendations
  - ✓ Implement MSI w/ PVM when only one vector is merited
  - ✓ Implement MSI-X when multiple vectors are merited
  - ✓ Implement both only if multiple vectors are merited, **and** backwards compatibility with MSI software is also required



# Designing Software for ExpressCard\*

- ExpressCard\* power is automatically switched by hardware
- No software handshake is required for device removal



- Software must correctly handle device surprise removal
- Key Considerations:
  - ✓ **Write caching of data needs particular attention**
    - May need to be disabled to avoid loss of data
  - ✓ **Software should check device availability before issuing commands**
    - “Bit cleared” if available type of test
    - Implement timeouts

# Other ExpressCard\* Considerations

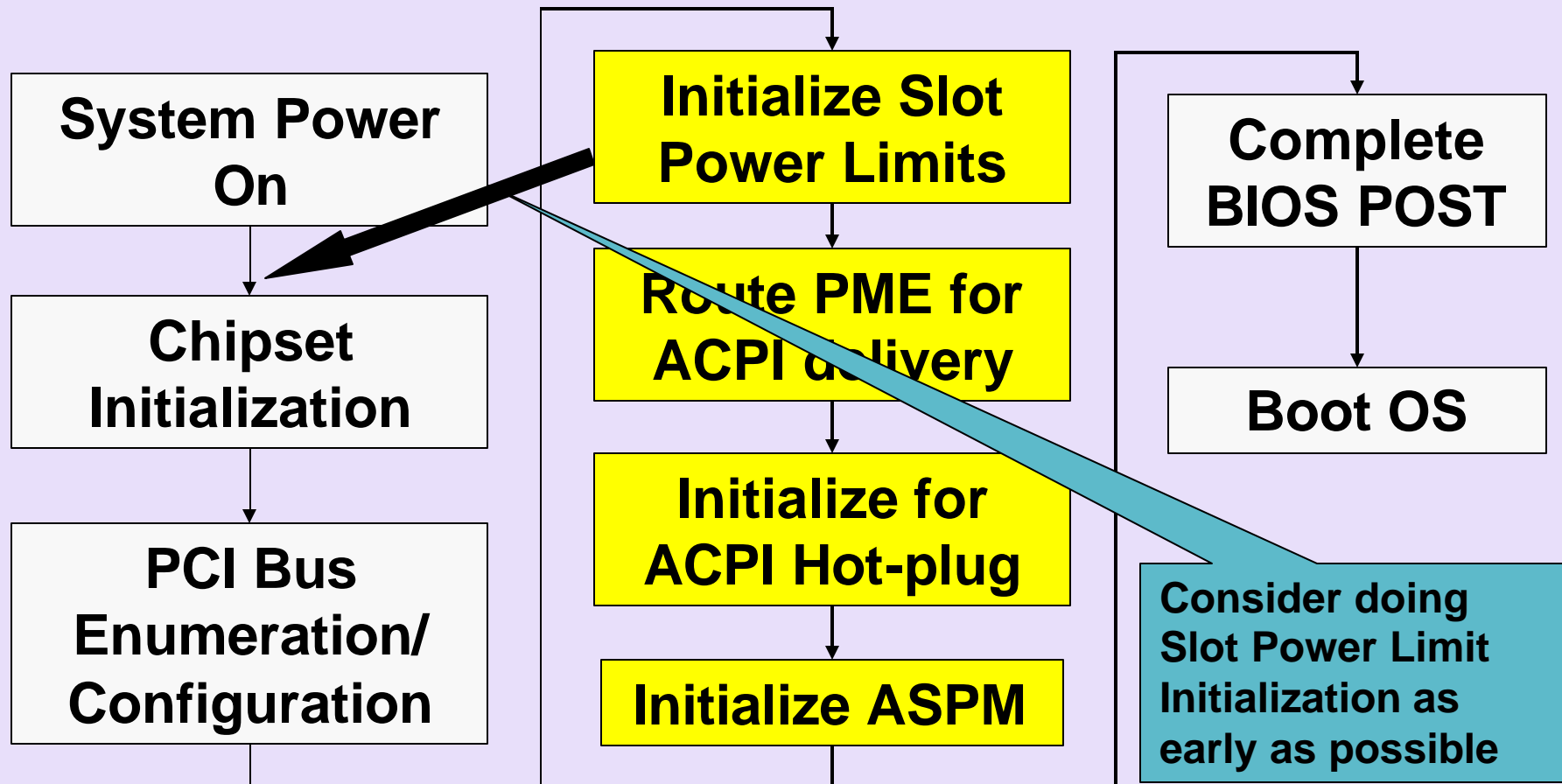
- Hot-plug messages must be handled correctly even if corresponding functionality not supported by device
  - ✓ **Don't hang on unsupported TLPs**
- Consider enabling ASPM by default if latency tolerable for device functionality
  - ✓ **Mobile systems benefit from increased battery life**
- Clocks can be removed in mobile environments to conserve power
- Devices should update NFTS values correctly based on common clock configuration
  - ✓ **Power conservation for mobile systems**
- Switch vendors need to synthesize INTx de-asserts for surprise removal of attached devices

**ExpressCard\* implementations must pay particular attention to power management and surprise removal strategies**

# Firmware Guidelines

- Firmware must correctly communicate configuration space information to operating system
- Firmware support required for legacy compatible support in the absence of a PCI Express-aware OS
  - ✓ **ExpressCard\* hot-plug through ACPI**
  - ✓ **ACPI PME handling**
- Firmware must implement functionality for transition of services to PCI Express aware OS
  - ✓ **Native OS PME handling**
  - ✓ **Native OS hot-plug**
- Firmware support required for correct power negotiation
  - ✓ **Power budgeting can enable firmware to optimize power distribution at boot**

# Firmware Initialization Example



**Firmware plays key role in enabling new PCI Express features**

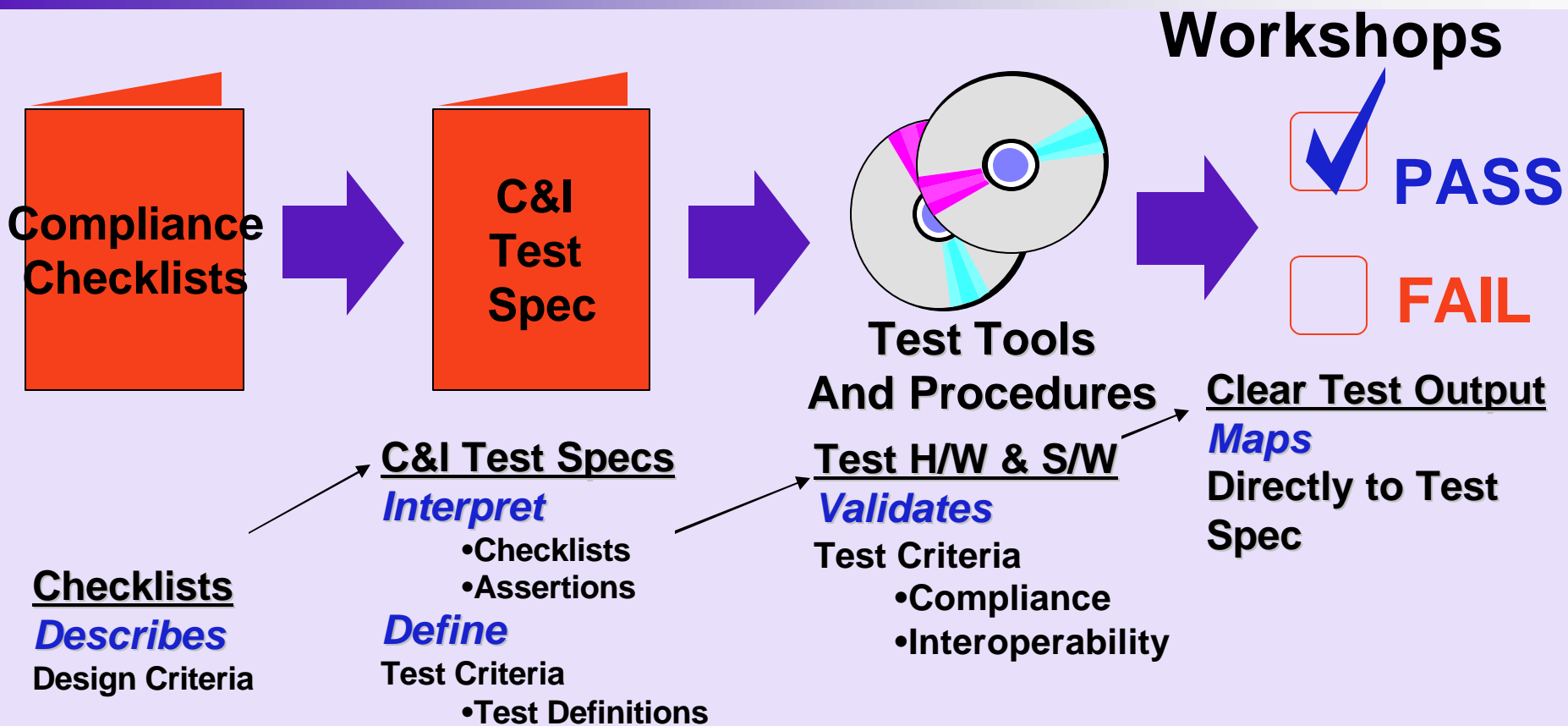
# Summary

- Legacy software compatibility and transition strategy to PCI Express-aware software environment must be considered at all times
- MSI/MSI-X have major advantages over INTx, but correct interrupt handling to avoid loss of edge-triggered interrupts is not intuitive
- Implement MSI-X instead of MSI when multiple vectors are merited
- ExpressCard\* implementations must pay particular attention to power management and surprise removal strategies
- Firmware plays key role in enabling new PCI Express features

# Agenda

9:00	PCIe Overview	<i>Neshati</i>
9:45	PCIe Physical Layer	<i>Schoenborn</i>
12:00	LUNCH	
1:00	PCIe Configuration & Software	<i>Cowan</i>
2:45	BREAK	
<b>3:00</b>	<b>PCIe Compliance Tools</b>	<i>Choate/Froelich</i>
5:00	Q&A	
5:15	BREAK	
5:30	Compliance Workshop Orientation	<i>Neal/Kelley, et al</i>

# PCI Express Compliance Process Components



**Predictable Path to Design Compliance**

# PCI Express Compliance Checklists

- Provide design-time 'rules' that implementations should follow
  - ✓ Checklists for Root Complex, Endpoint, Switch, addin card, and motherboard
- Simple set of 'yes/no' questions
- Checklists available on PCI SIG website
  - ✓ [http://www.pcisig.com/specifications/pciexpress/technical\\_library](http://www.pcisig.com/specifications/pciexpress/technical_library)

PHY.2.6#1	Training sequence ordered-sets are never scrambled but always 8b/10b encoded.	yes ____ no ____
PHY.3.1#25	The receiver terminations must remain enabled in Electrical Idle.	yes ____ no ____
PHY.3.2#5	The Beacon signal must contain minimum width pulses $\geq 2$ ns.	yes ____ no ____



# PCI Express Compliance Test Specifications

- Clear description of what is being tested
- Contains:
  - ✓ Assertions
    - Prioritized set pulled from checklists
  - ✓ Test Descriptions
    - What a test does, and what assertions it checks

# PCI Express Compliance Test Procedures

- Describes how to run tests
  - ✓ Required equipment
  - ✓ Step-by-step instructions
- Detailed procedures help ensure repeatability
  - ✓ At Compliance Workshops
  - ✓ In development labs

# Compliance Tools Goals

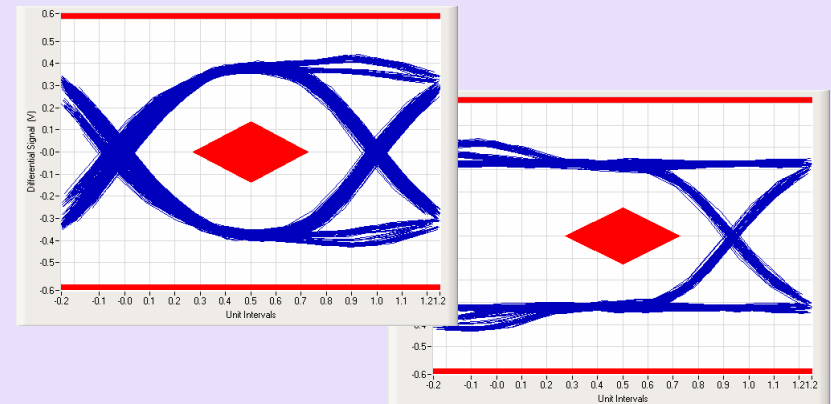
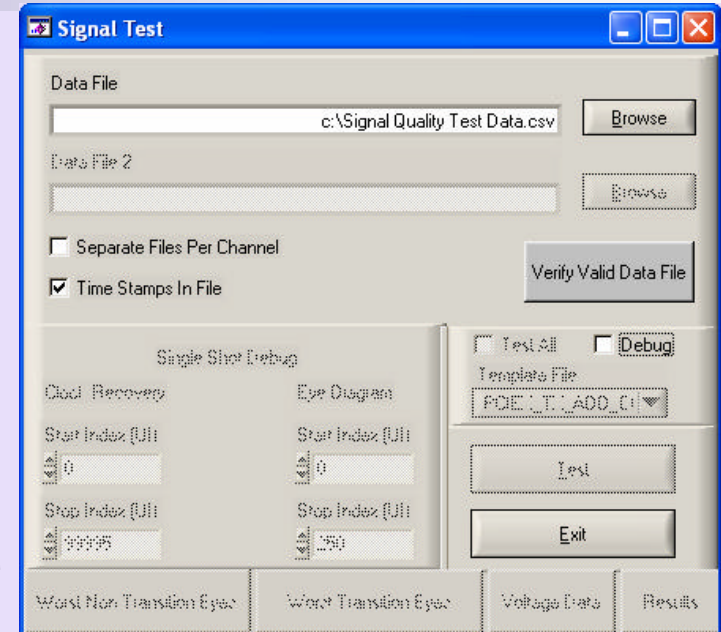
- Wide deployment in development labs
  - ✓ Tools and procedures should be useful and valuable in a development lab setting
- Easy to use
  - ✓ Clear documentation, clear procedures
- Reasonable equipment cost
  - ✓ ~\$50K
  - ✓ Ideally, equipment is useful for many other things besides PCI Express compliance testing

# Compliance Test Areas

- Physical layer
  - ✓ Examine electrical signaling
- Configuration Space
  - ✓ Verify required fields and values
- Link & Transaction layer (2 areas)
  - ✓ Exercise protocol boundary conditions
  - ✓ Inject errors and check error handling
- Platform Configuration
  - ✓ Check BIOS handling of PCI Express devices

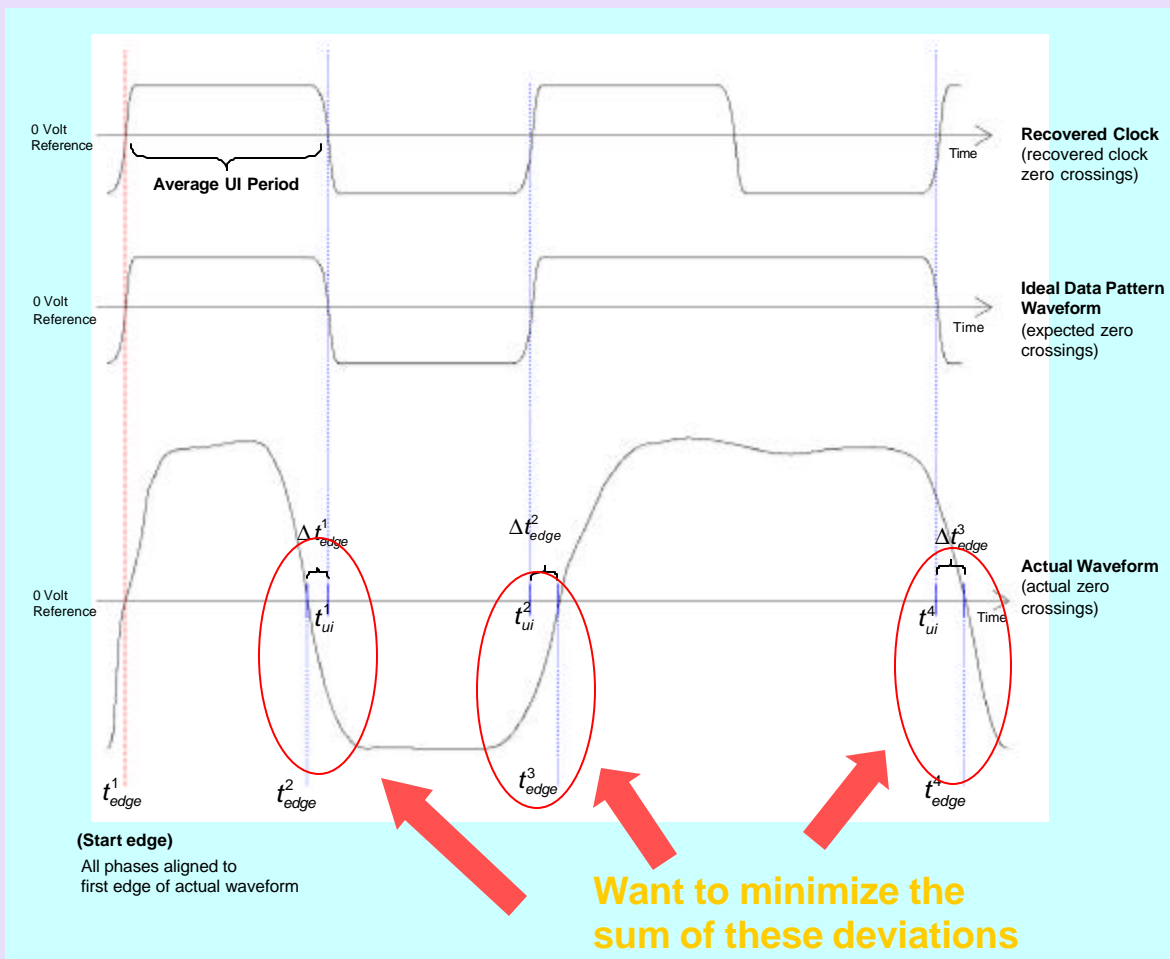
# Electrical Tests and Tools

- Signal Quality Analysis H/W and S/W
  - ✓ Eye pattern, jitter and bit rate analysis
  - ✓ Upstream and downstream signaling
  - ✓ Electrical compliance base board
  - ✓ Electrical compliance load board
  - ✓ Stand-alone Windows-based eye diagram analysis S/W
  - ✓ Electrical test procedures and Oscilloscope setup files
- Jitter Analysis DLL
  - ✓ Clock Recovery
  - ✓ Interpolation
  - ✓ Transition/non-transition eye points
  - ✓ Goal - Promote consistent solutions



<http://www.pcisig.com/specifications/pciexpress/compliance>

# Tx Clock Recovery



The phase and frequency of the recovered clock is determined by minimizing the sum of the absolute deviations between the actual edges of the waveform and the edges based on the recovered clock.

Minimize Absolute Deviation Algorithm

$$t^n_{edge[recovered]} = f + \frac{1}{f} i$$

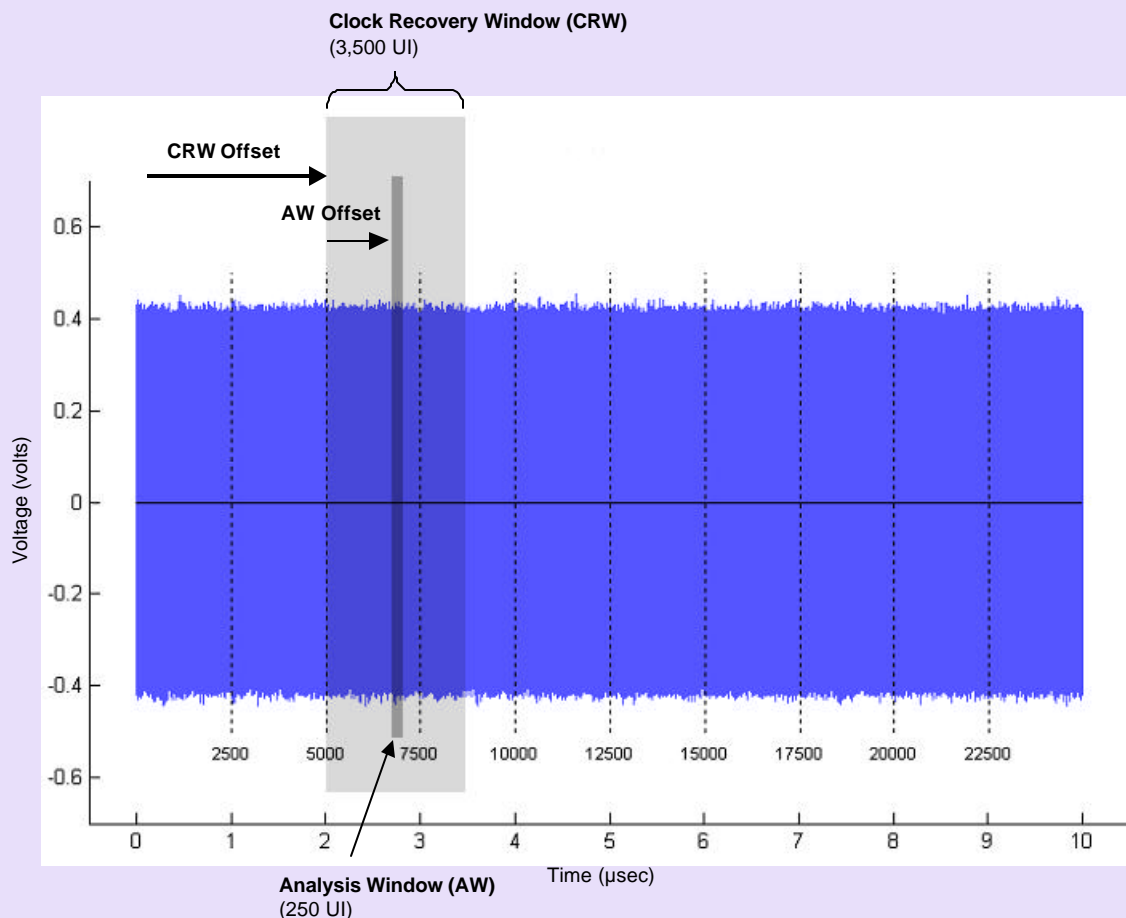
$$f = \text{median} \left\{ t^n_{edge[actual]} - \frac{1}{f} i \right\}$$

$$\sum_{n=1}^N \left| t^n_{edge[actual]} - f - \frac{1}{f} i \right|$$

$$0 = \sum_{n=1}^N i \text{sgn}(t^n_{edge[actual]} - f - \frac{1}{f} i)$$

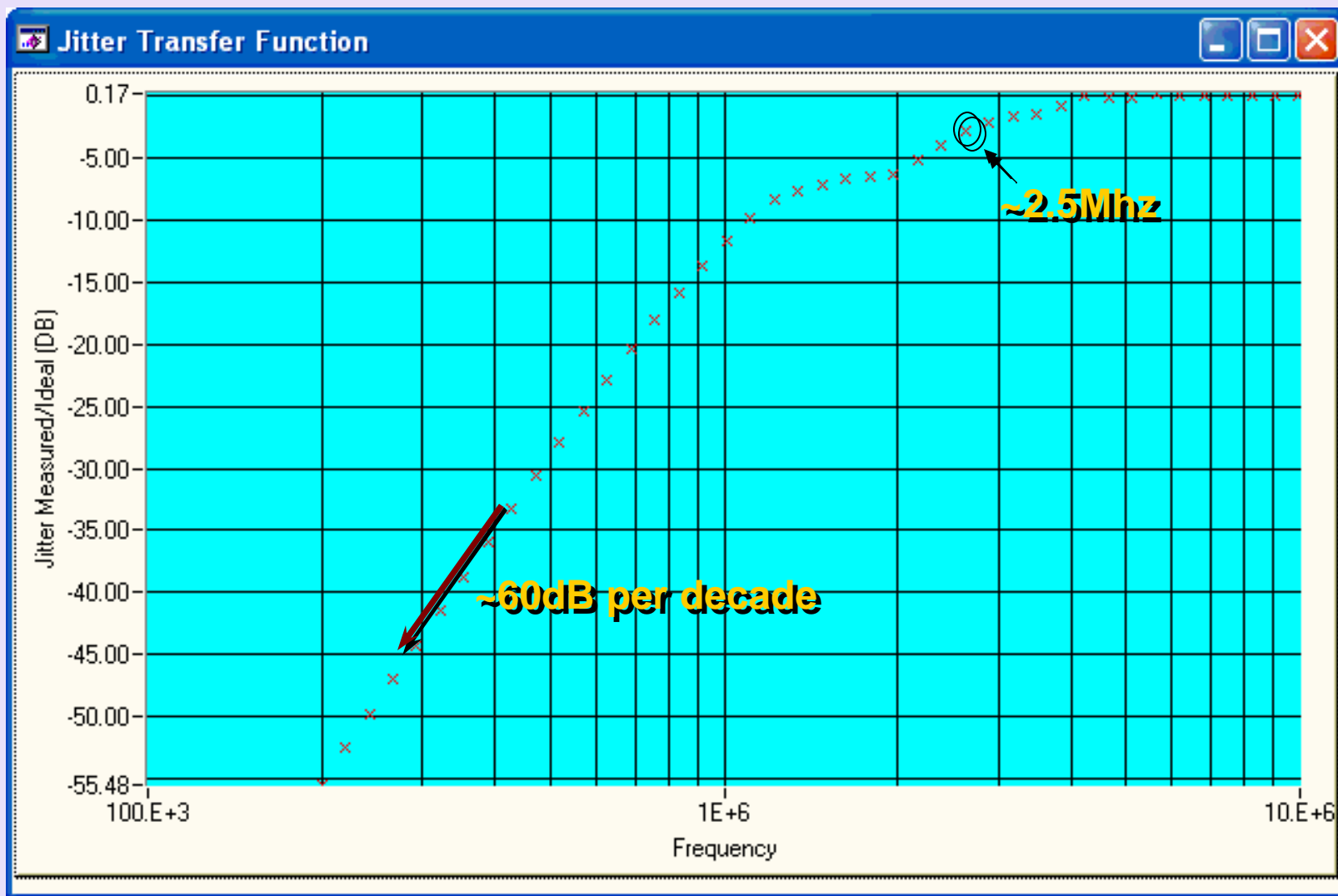
The 'minimize absolute deviation' algorithm is more robust (less sensitive to outliers) than a  $\chi^2$  or average method.

# Clock Recovery Window



- A clock recovery window (CRW), which is 3,500 unit intervals (UI) in length, is swept across the waveform. A window length of 3,500 UI equals approximately  $1/20^{\text{th}}$  the allowable spread spectrum modulation period. This allows the recovered clock to track any frequency variation due to spread spectrum clocking.
- All edges within the CRW are used to calculate the recovered clock.
- Jitter is calculated relative to the recovered clock over the middle 250 UI subset of the CRW.

# SigTest Jitter DLL Transfer Function





# Signal Analysis Methodology Summary

1. At least 3,500 consecutive UI of data under test are obtained from an oscilloscope sampling at 50 ps or less.
2. Sinc interpolation is used to produce sample data at 17 ps intervals or less. (2 interpolated points @ 50ps sample rate)
3. Linear interpolation is used to estimate the crossover points (edges)
4. A minimize deviation fit algorithm is used to obtain the recovered clock using the clock recovery window of the first 3,500 unit intervals.
5. The data skew (jitter) relative to the recovered clock is extracted.
6. The jitter values are normalized to the median jitter value.
7. Eye diagrams are plotted using the recovered clock period and measured against the appropriate eye mask.
8. The clock recovery window is incremented and steps 4-7 are repeated.
9. Step 8 is repeated until the clock recovery window reaches the end of the data set.

# Motherboard Electrical Tools

- Compliance Load Board
  - ✓ Root Complex electrical signal quality
  - ✓ Systems & Motherboards
  - ✓ Terminates Transmitters to utilize spec required compliance mode
  - ✓ X1, X4, X8, X16 test configurations
  - ✓ Improved active probing geometries
  - ✓ SMAs added to x8 & x16, first-middle-last
  - ✓ Analog loopback (Tx to Rx stuff option)

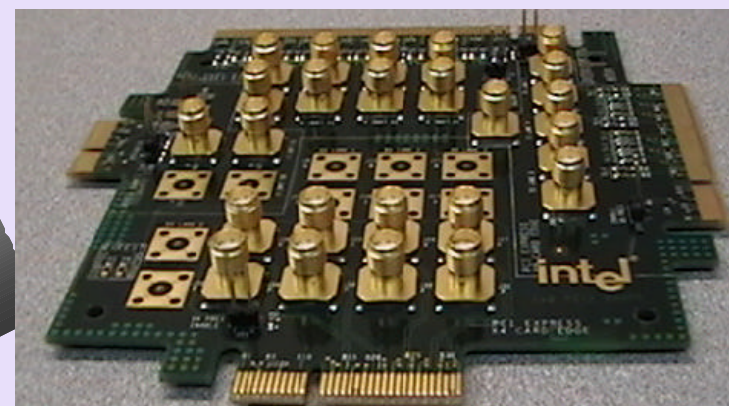
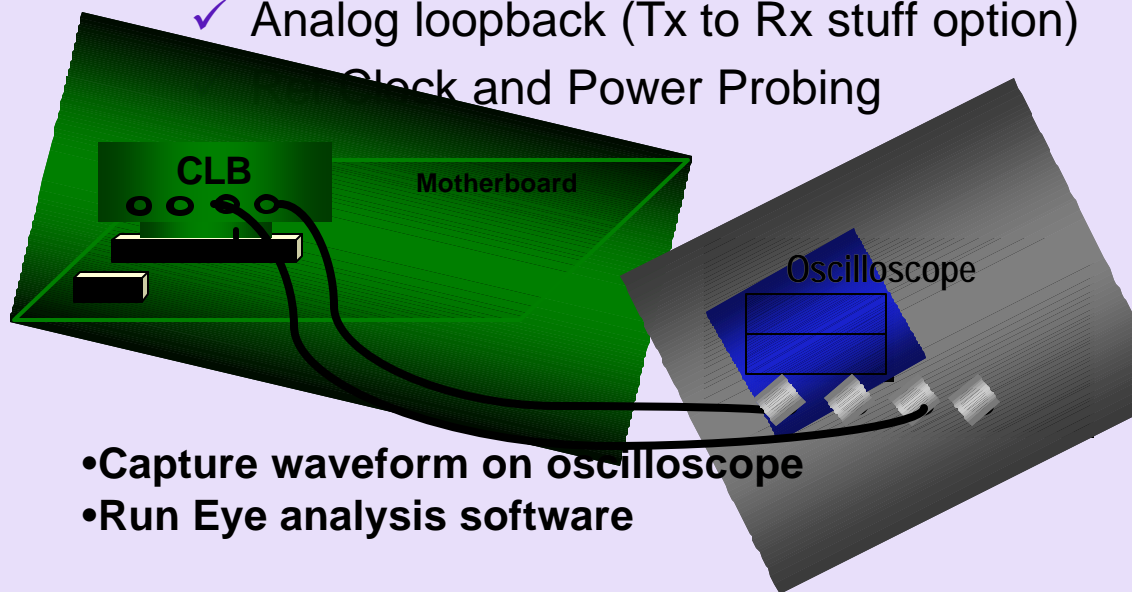
*Available thru*

*PCI SIG website*

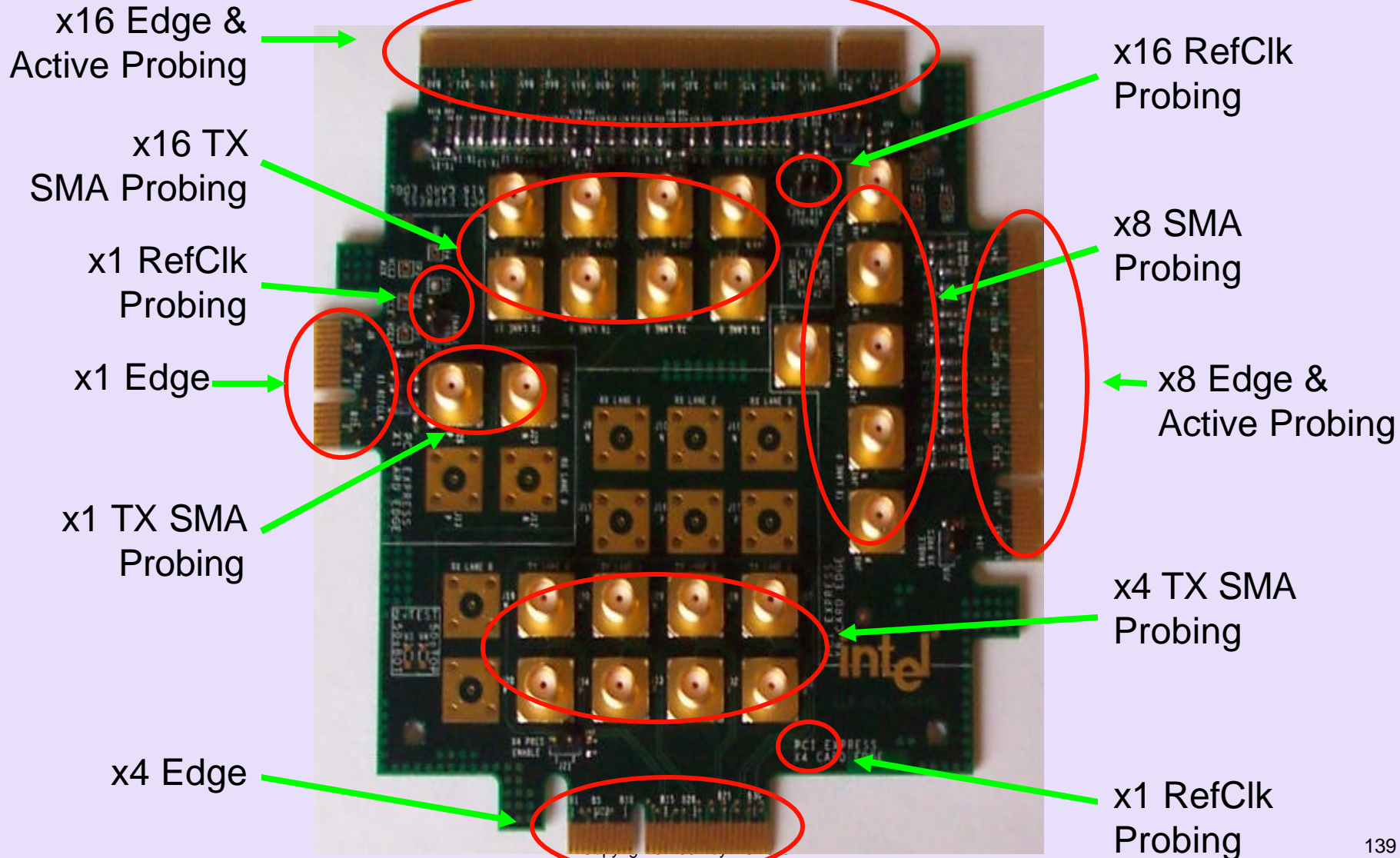
<http://www.pcisig.com>

[/specifications/pciexpress/compliance](http://www.pcisig.com/specifications/pciexpress/compliance)

Clock and Power Probing



# Compliance Load Board (CLB)



# Add-in Card Electrical Tools

## ■ Compliance Base Board

✓ X1, X16 connectors

– X16 provides termination X16 through X1

✓ Power adaptor + current measurement

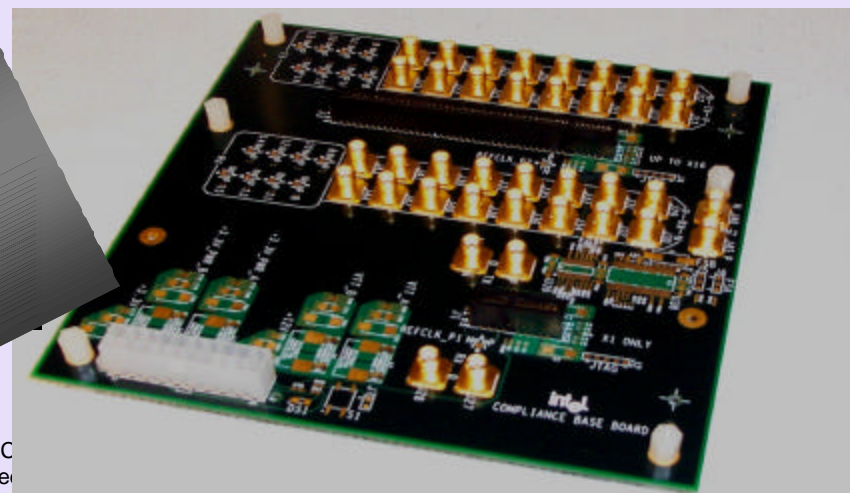
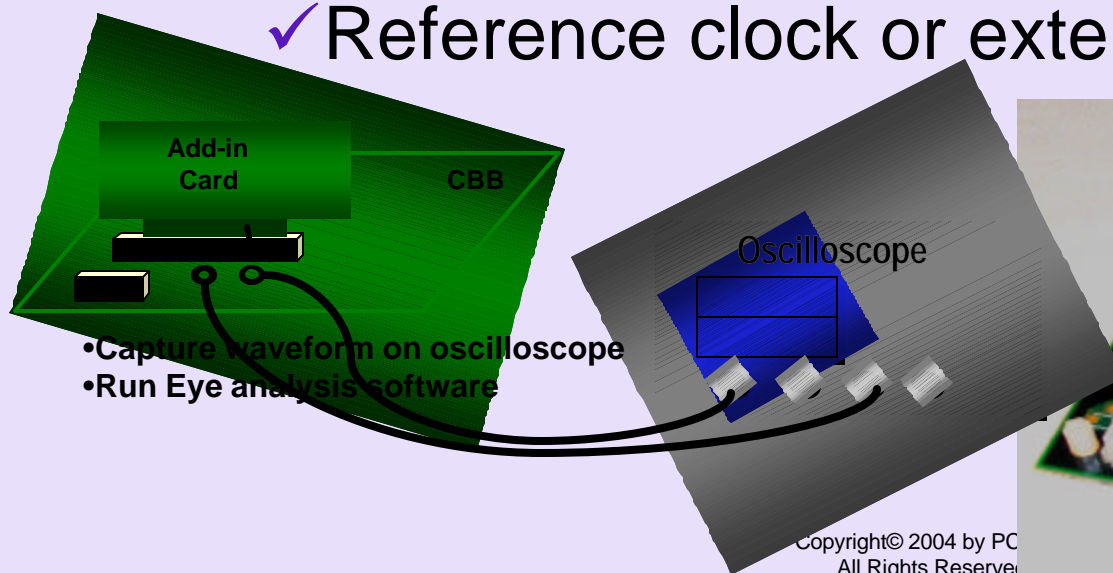
✓ RX for X1 up to X8

✓ Reference clock or external input

*Available thru  
PCI SIG website*

<http://www.pcisig.com>

[/specifications/pciexpress/compliance](http://www.pcisig.com/specifications/pciexpress/compliance)





x16 RX  
Inputs

x16 RefClk  
Probing

x16 TX  
Active  
Probing

x16 TX  
SMA  
Probing

Current  
Probing

ATX PS

x16 Test Slot

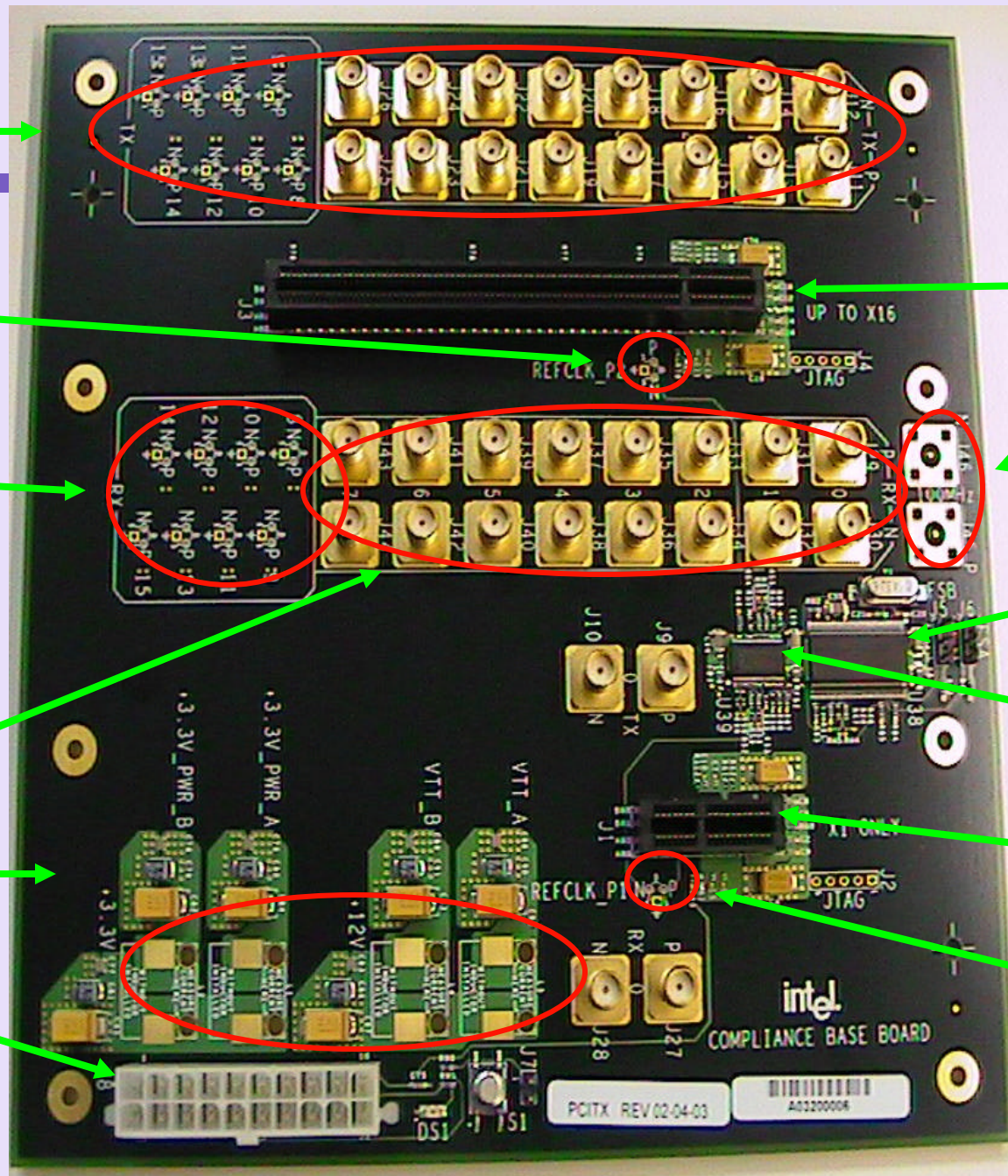
External  
Clock Input

100Mhz Clk  
(CK409)

Clk Buffer  
(DB400)

x1 test slot

x1 RefClk  
Probing



# Electrical Test Tool Demo

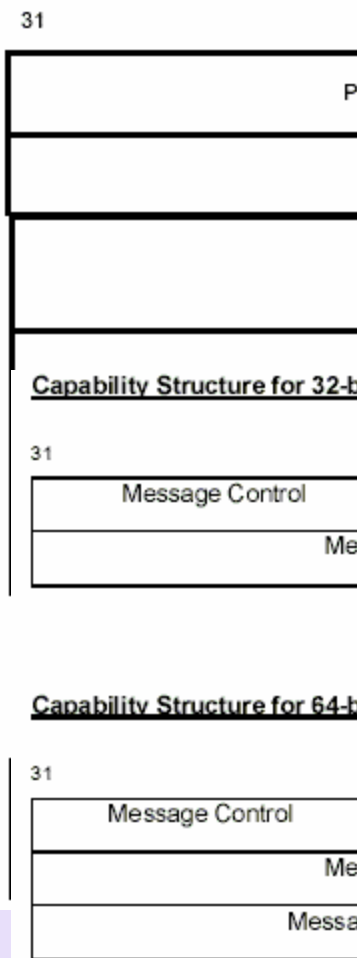


Figure 6-11: M

**4 Specs**  
**20+ ECRS**  
**14+ Capabilities**  
**100's Of Registers**  
**1000's Of Fields**  
**1000 Companies**  
**Implementing**

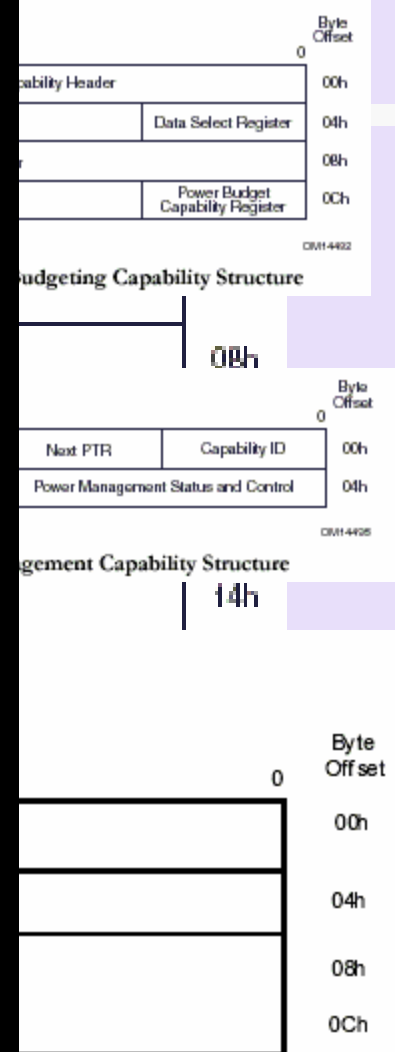


Figure 7-26: Link Entry



Figure 7-26: PCI Express Advanced Error Reporting Extended Capability Structure

# Config Space Tests

- Registers and Capabilities
  - ✓ Default Values
  - ✓ Characteristics
  - ✓ Required registers and capabilities
- Functional
  - ✓ Configuration Stress
    - Earliest Allowed Requests
    - Different Upstream/Downstream ASPM Combinations
    - Power Indicator/Control Messages
- Available on SIG website
- <http://www.pcisig.com/specifications/pciexpress/compliance>



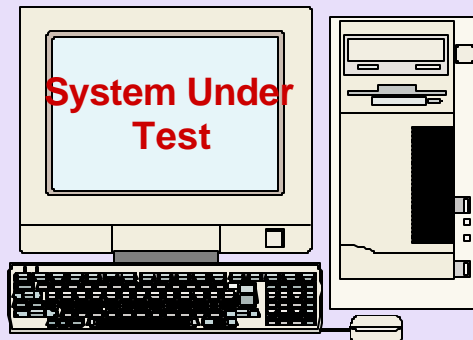
# Configuration: Register Only Rule Checks

Check register characteristics and default values..

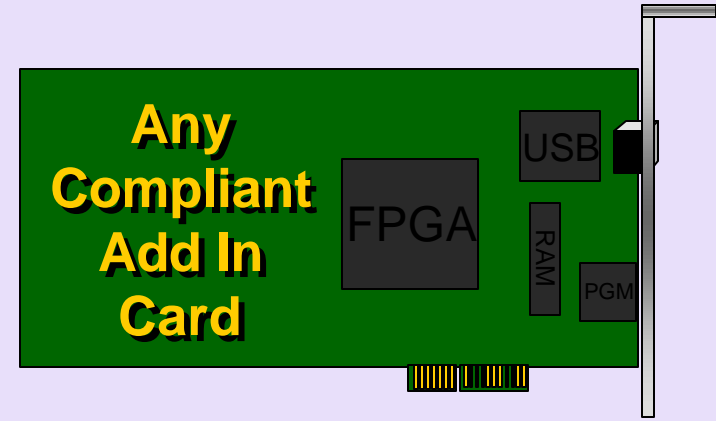
- One test for each new capability
  - ✓ PCI Express Capability
  - ✓ Device Capabilities, Control, and Status
  - ✓ Link Capabilities, Control, and Status
  - ✓ MSI
  - ✓ Advanced Error Reporting
  - ✓ Virtual Channel
  - ✓ Device Serial Number
  - ✓ Power Budgeting

# Configuration Tests - Diagram

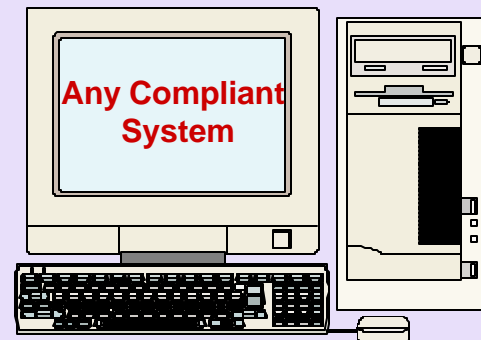
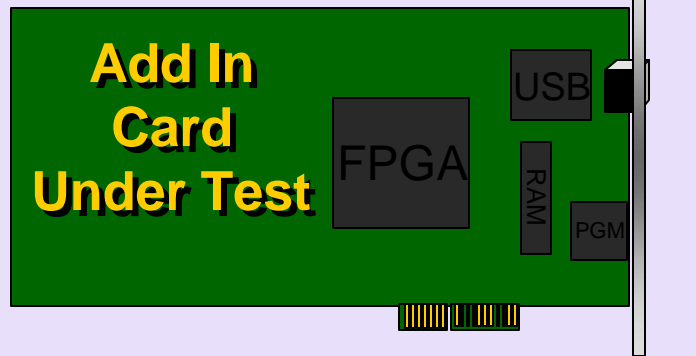
**System  
Test**



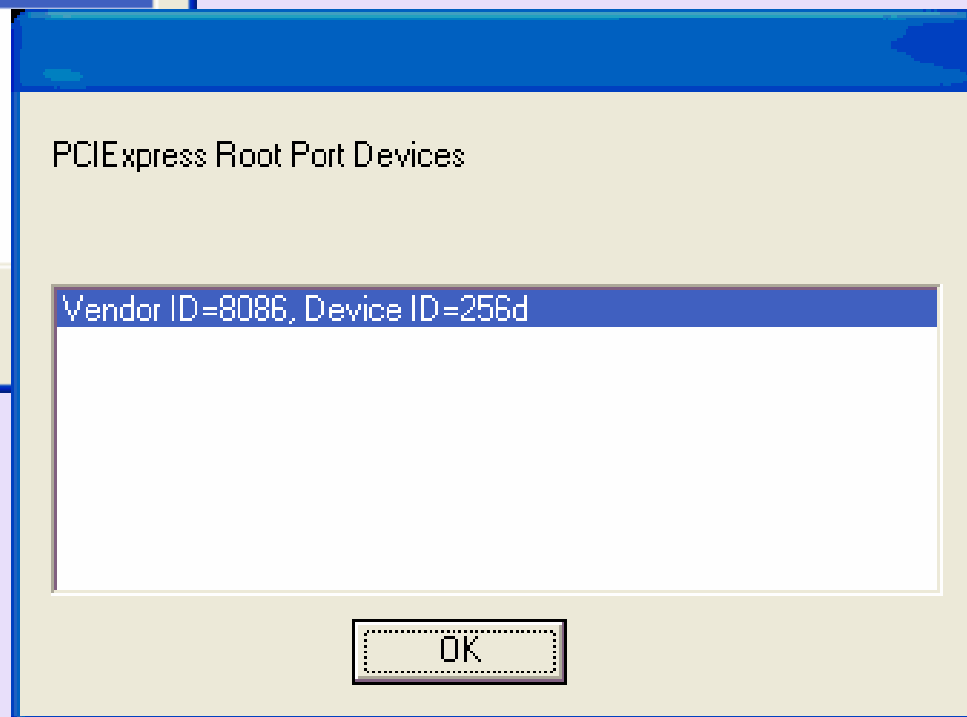
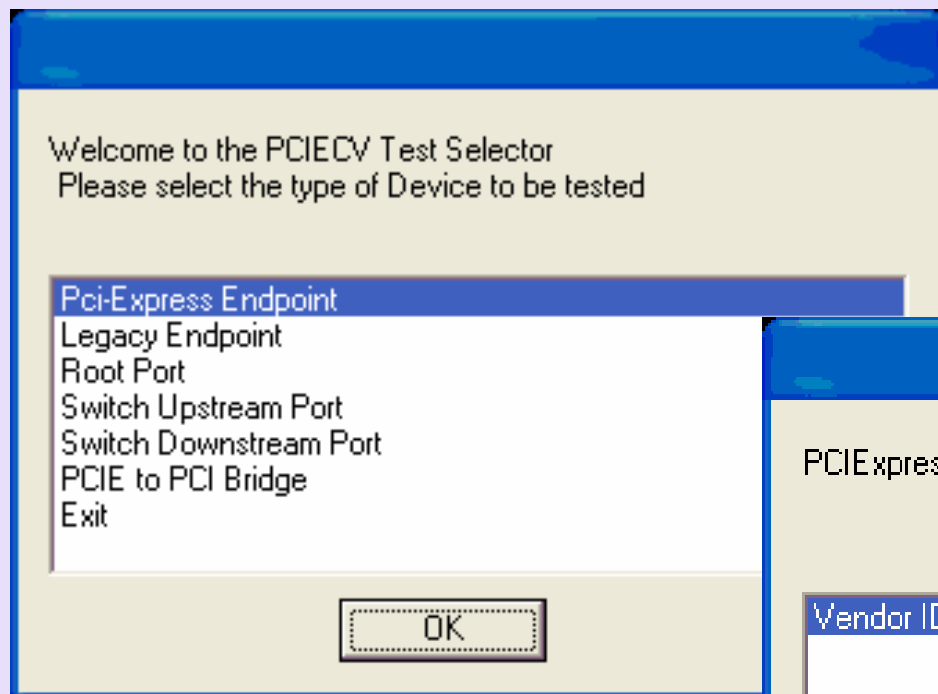
**Any  
Compliant  
Add In  
Card**



**Add in  
Card  
Test**



# PCIECV – Device Type Selection



```
INFO Test End Point was selected..
INFO Device selected: vendor ID= 11c2, Device ID= beef
          Bus number= 0001, Device Number= 0000, Function= 0000
INFO Beginning Register and Capability Tests
INFO TD_1_2_PCIExpressCapabilityStructureTest Selected.
```

```
INFO Starting Test: TD_1_2 PCIExpress Capability Structure Test
*****
```

```
INFO PCI Capability ID value: 0x10
INFO Capability offset value: 0x60
INFO Next Capability Pointer value: 0x84
INFO Device Capability Register value: 0xfa5
INFO Device Control value: 0x2810
INFO Device Status value: 0x0
INFO Link Capabilities value: 0x3f411
INFO Link Control value: 0x0
INFO Link Status value: 0x11
INFO Slot Implemented bit is not set.
```

```
INFO Stopping Test: TD_1_2 PCIExpress Capability Structure Test
INFO Number of: Fails (0); Aborts (0); warnings (0)
```

```
INFO TD_1_3_PCIExpressCapabilitiesRegisterTest Selected.
```

```
INFO Starting Test: TD_1_3 PCI Express Capabilities Register Test
*****
```

```
INFO PCI Express Port/Device Type value: 0x0
INFO PCI Config Space Header Type: 0x0
INFO PCIe Slot Implemented value : 0x0
```

```
INFO Testing Register: PCIExpress Capability List Register
```

```
INFO Beginning RO test: Capability ID
INFO Initial value of the field is 0x10
INFO Testing with inverse of original value.
INFO Value read back from the field was: 0x10
INFO Testing field with all ones.
INFO Value read back from the field was: 0x10
```

```
INFO Beginning RO test: Next Capability Pointer
INFO Initial value of the field is 0x84
INFO Testing with inverse of original value.
INFO Value read back from the field was: 0x84
INFO Testing field with all ones.
INFO Value read back from the field was: 0x84
```

```
INFO Testing Register: PCI Express Capabilities Register
```

```
INFO Beginning RO test: Capability Version
INFO Initial value of the field is 0x1
INFO Testing with inverse of original value.
INFO Value read back from the field was: 0x1
```

#### PCIExpress End Point tests

##### Run all Tests

```
Run all Register and Capability Tests
Run Register and Capability Tests in Debug Mode
Run all Functional Tests
Run Functional Tests in Debug Mode
Exit
```

#### Test Results

```
Passed: TD_1_2_PCIExpressCapabilityStructureTest
Passed: TD_1_3_PCIExpressCapabilitiesRegisterTest
Failed: TD_1_4_DevCapControlStatusReg
Passed: TD_1_5_LinkCapControlStatusReg
Passed: TD_1_6_MSICapabilityStructureTest
Failed: TD_1_7_AdvancedErrorReportingCap
Passed: TD_1_8_VirtualChannelCap
Passed: TD_1_9_SerialNumberCap
Passed: TD_1_10_PowerBudgetingCap
Passed: TD_1_11_CommandStatusRegTest
Passed: TD_1_12_CacheLnSzMasterLatTimerMinGntMaxLatReg
Failed: TD_1_13_InterruptPinInterruptLine
Not Run: TD_1_14_SecondaryLatTimerSecondaryStatusReg
Not Run: TD_1_15_BridgeControlReg
Failed: TD_1_16_PowerManagementCap
Not Run: TD_1_17_MSIXCapabilityStructureTest
Passed: TD_1_18_BaseAddressRegistersTest
```

OK

# PCIECV Software Status

## ■ Beta Release

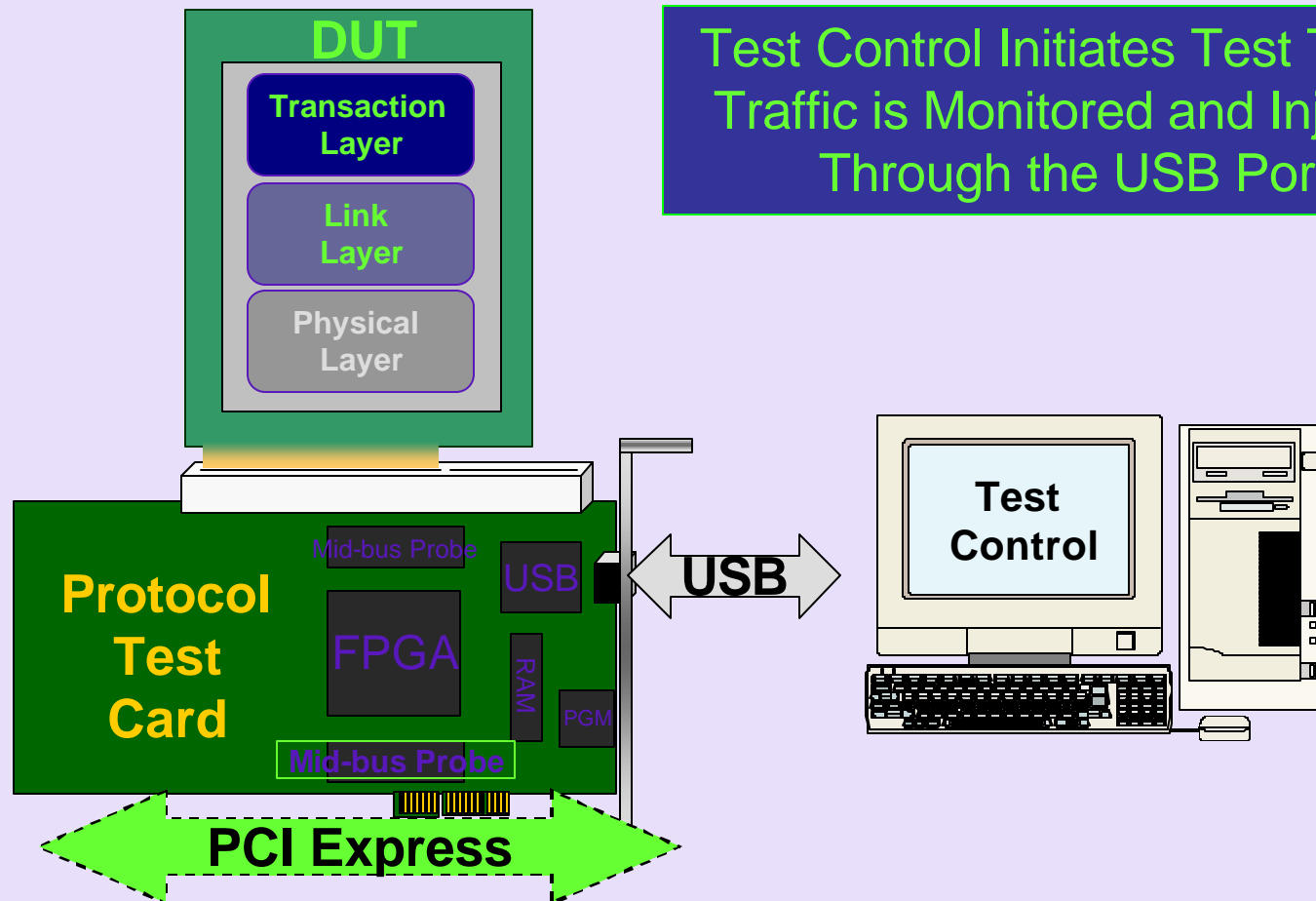
- ✓ Plug In allows flexible use
  - Test Equipment Exercisers (NITAL Shipping Plug In)
  - Simulation environments
- ✓ All Common Test Specified Through .ini Files
  - Easy reaction to spec changes and additions with recompilation
  - Easy debugging
- ✓ Coverage of chapter 7 in 1.0a PCI Express Specification

## ■ 1.0 Release

- ✓ Functional tests
  - Configuration stress testing
  - Default values
- ✓ General system level tests
  - Port Reporting
  - Configuration Stress

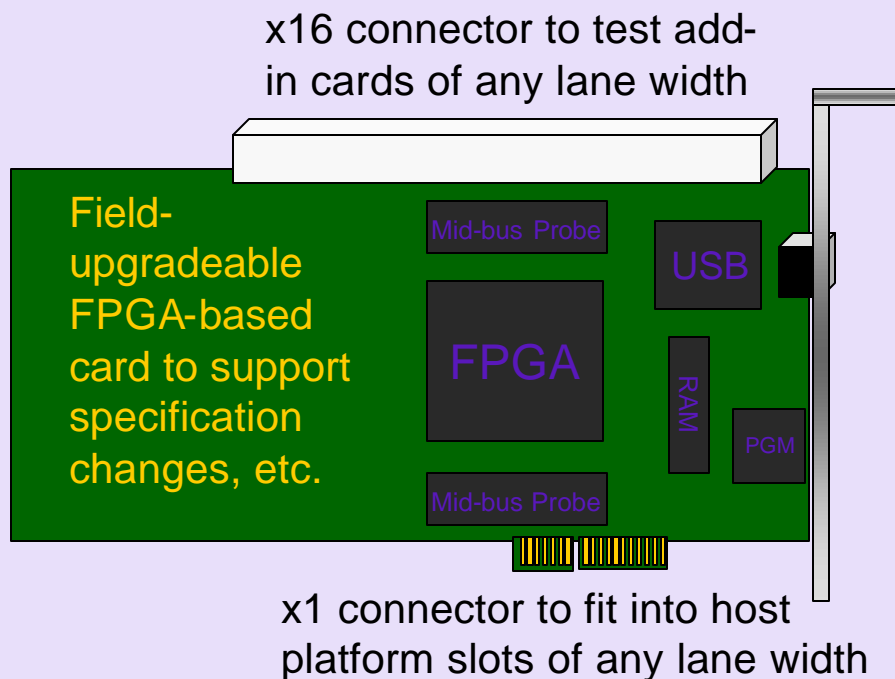
## ■ 1.1 Release

- ✓ Full PCI Express 1.0a Bridge Spec config coverage.
- ✓ PCI 2.3/3.0 requirements that apply to PCI Express devices.
- ✓ PCI 1.1/1.2 Bridge Spec requirements that apply to PCI Express bridges.
- ✓ Planned for Q2 2004



# Protocol Test Card

Host and add-in card mid-bus probe points for connection to any PCI Express protocol analyzer



USB interface For Test Monitoring/Control and for FPGA programming/debug

- x1 operation only
  - ✓ Sufficient for most protocol-oriented tests

# Link Compliance Focus

- Basic
  - ✓ Link Initialization, DLLP and TLP packet Formats
- Reliability
  - ✓ Replay Mechanism
    - Nak Response, Replay Timer countdown, Retry counts, Link Retraining
  - ✓ CRC
    - LCRC and DLLP CRC calculation checks, ECRC when applicable
  - ✓ Protocol
    - Missing and duplicate TLPs, Wrong Sequence Numbers
    - Response to Undefined TLP and DLLPs
    - Reset Propagation across secondary Buses
  - ✓ Timing
    - Ack/Nak Timeouts
- Serviceability
  - ✓ Advanced Error Logging and Reporting

**See Test Spec  
for Details**

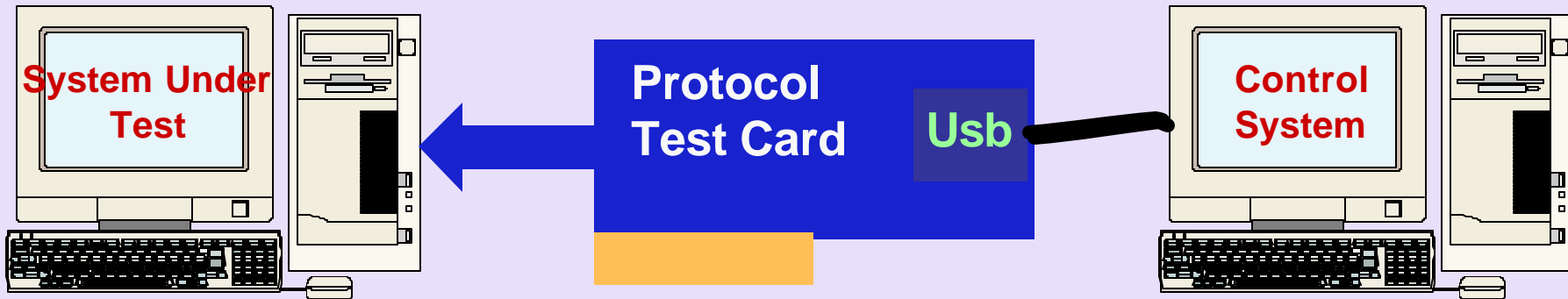


# Transaction Compliance Focus

- Basic Functional
  - ✓ Completion request
  - ✓ Completion timeout
  - ✓ Read data
- Baseline Messaging
  - ✓ Legacy interrupts
  - ✓ Native power management
  - ✓ Native Hot Plug
  - ✓ Error Signaling
- Flow Control
  - ✓ Initialization
  - ✓ Transmit and Receive states
  - ✓ Negotiated link width
- Virtual Channel

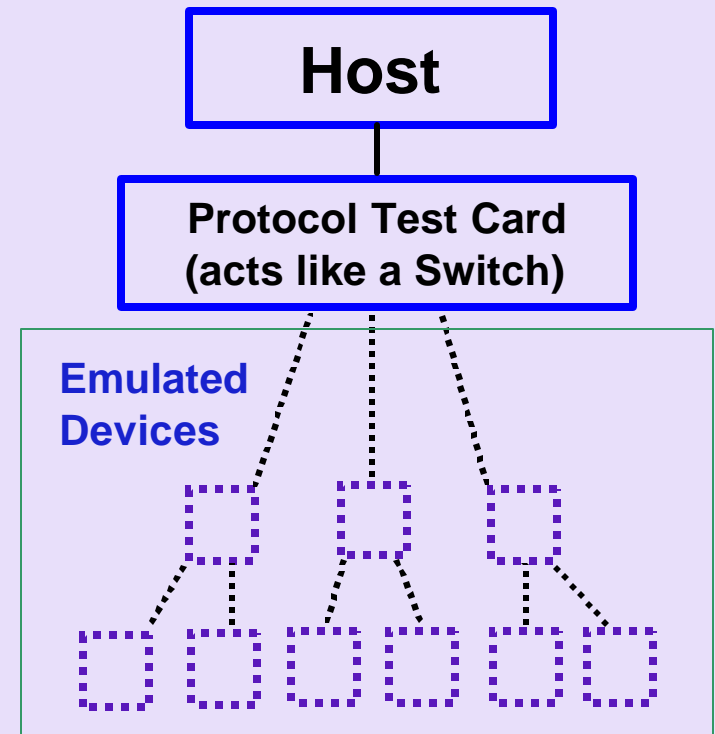
**See Test Spec  
for Details**

# Platform BIOS Testing



**Protocol Test Card Can  
Represent Any Hierarchical  
Multi Device/Bridge Topology**

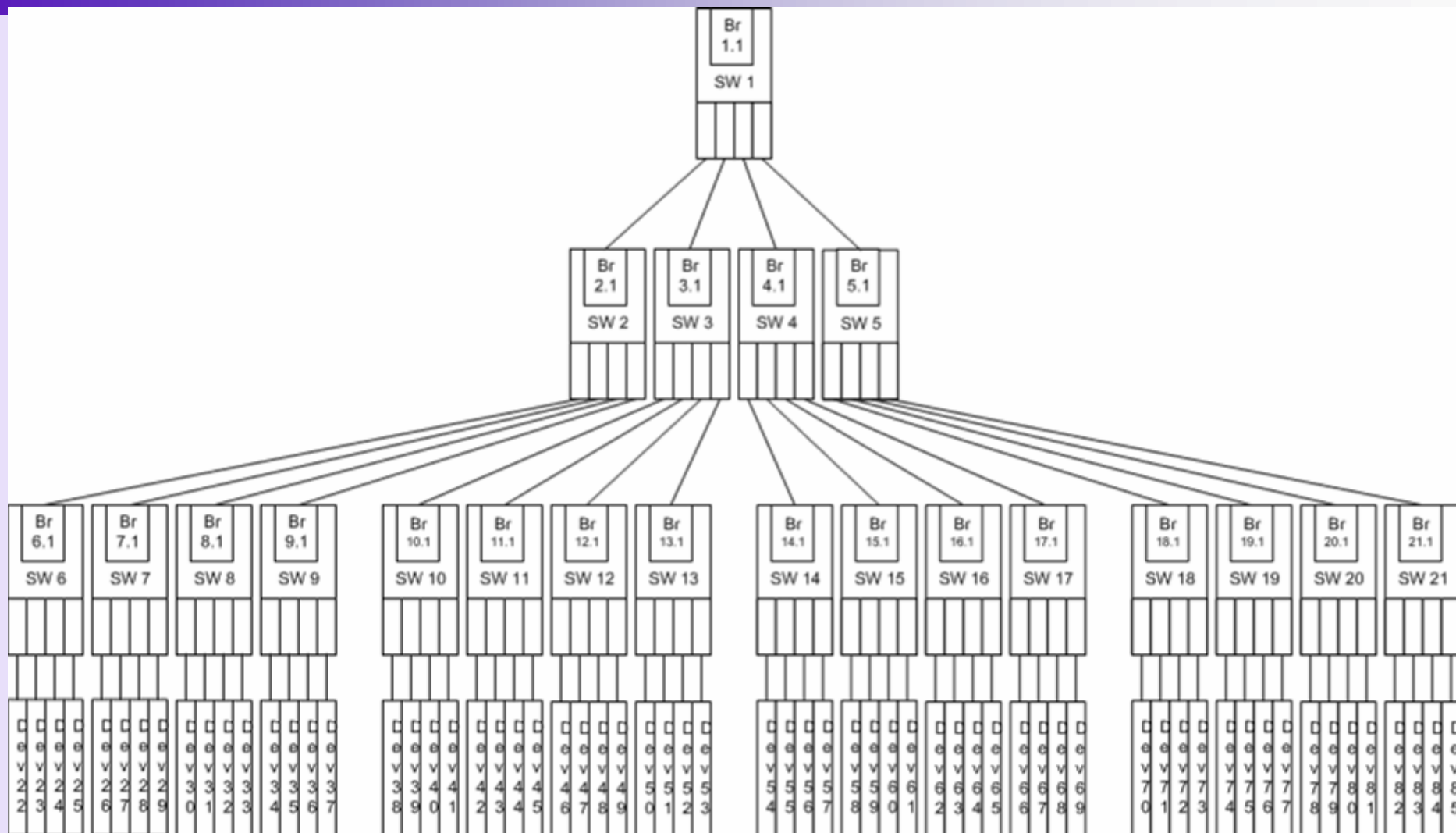
**Device Decodes All Type 0 and  
Type 1 Configuration Cycles.**



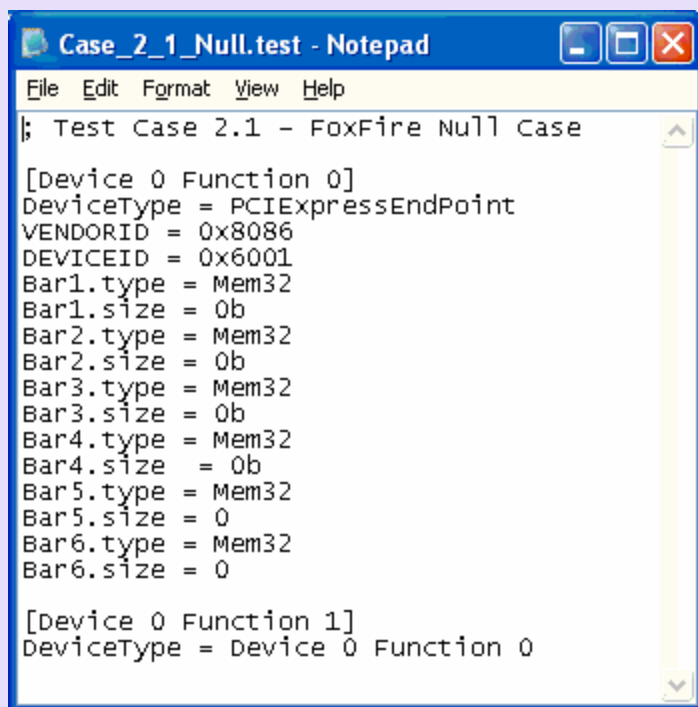
# Planned Platform Firmware Tests

- Complex Topology Configuration
  - ✓ Complex Memory Assignments
  - ✓ Level of Bridge Support
  - ✓ PME
  - ✓ ASPM Support
- Device Configuration Corner Cases
  - ✓ Non-responsive “bad” device
  - ✓ Worst case legal device response after reset
  - ✓ Config retry
  - ✓ Device doesn’t respond to shutdown requests

# Worst Case Topology



# PCIEPT Scripting Language



```

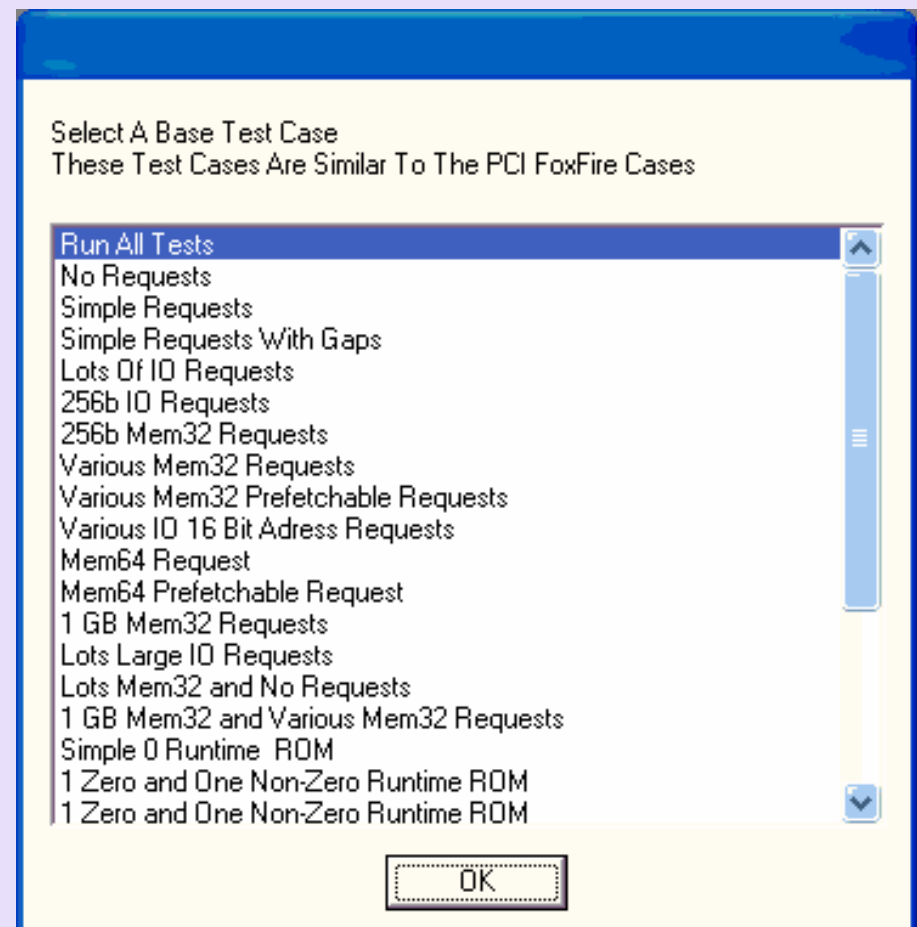
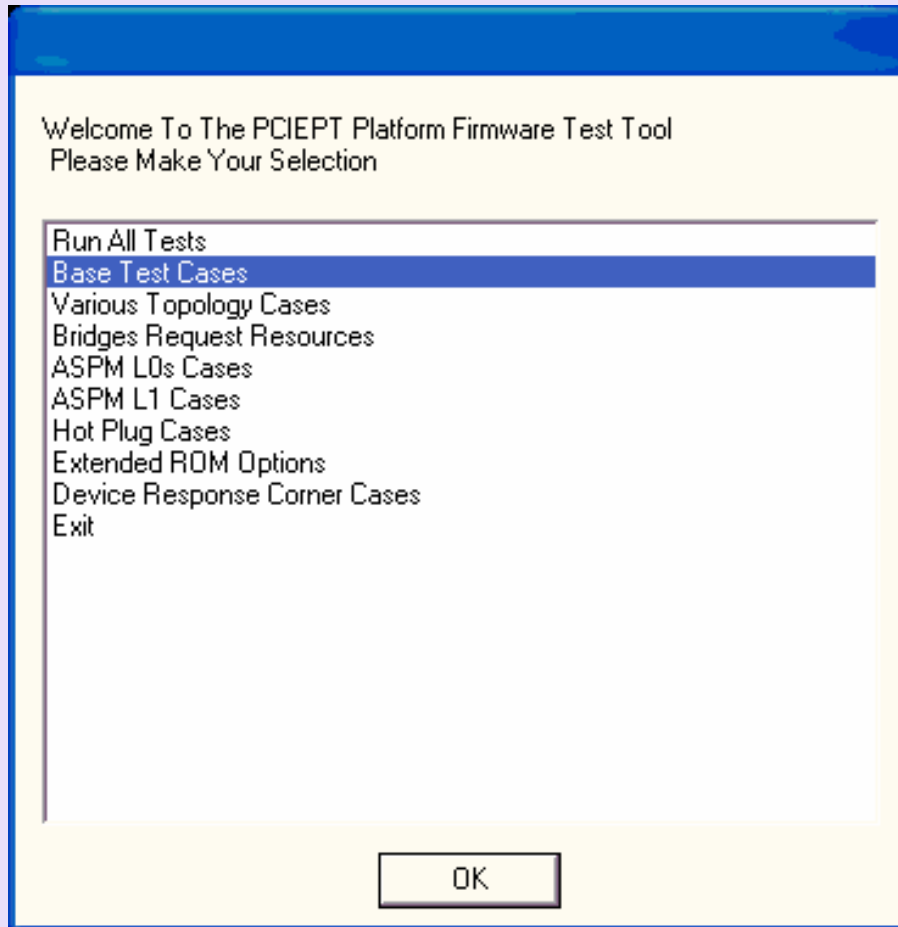
File Edit Format View Help
; Test Case 2.1 - FoxFire Null Case

[Device 0 Function 0]
DeviceType = PCIExpressEndPoint
VENDORID = 0x8086
DEVICEID = 0x6001
Bar1.type = Mem32
Bar1.size = 0b
Bar2.type = Mem32
Bar2.size = 0b
Bar3.type = Mem32
Bar3.size = 0b
Bar4.type = Mem32
Bar4.size = 0b
Bar5.type = Mem32
Bar5.size = 0
Bar6.type = Mem32
Bar6.size = 0

[Device 0 Function 1]
DeviceType = Device 0 Function 0
    
```

- Add Components
  - ✓ Switch
  - ✓ Endpoint
- Add Capabilities
- Control At Bit Level
  - ✓ Default Values
  - ✓ RW, RO, etc.

# PCIEPT Application



# Summary

- PCI Express Compliance tools are available
  - ✓ <http://www.pcisig.com/specifications/pciexpress/compliance>
- Get the Electrical and Configuration tests
  - ✓ Test specs, procedures, and fixtures are available
  - ✓ Validate in your labs
- Visit the PCI-SIG Website for Compliance Related Announcements

# Agenda

9:00	PCIe Overview	<i>Neshati</i>
9:30	PCIe Physical Layer	<i>Schoenborn</i>
12:00	LUNCH	
1:00	PCIe Configuration & Software	<i>Cowan</i>
2:45	BREAK	
3:00	PCIe Compliance Tools	<i>Choate/Froelich</i>
5:00	Q&A	
5:15	BREAK	
5:30	Compliance Workshop Orientation	<i>Neal/Kelley, et al</i>