Performance Modeling and Noise Reduction in VLSI Packaging

Ph.D. Defense

Brock J. LaMeres
University of Colorado
October 7, 2005

Problem Statement

- **VLSI Packaging Limits System Performance**

1) Supply Bounce
2) Signal Coupling
3) Bandwidth Limitation
4) Impedance Discontinuities
5) Cost & Scaling

Agenda

1) **Problem Motivation**

2) **Research Overview**

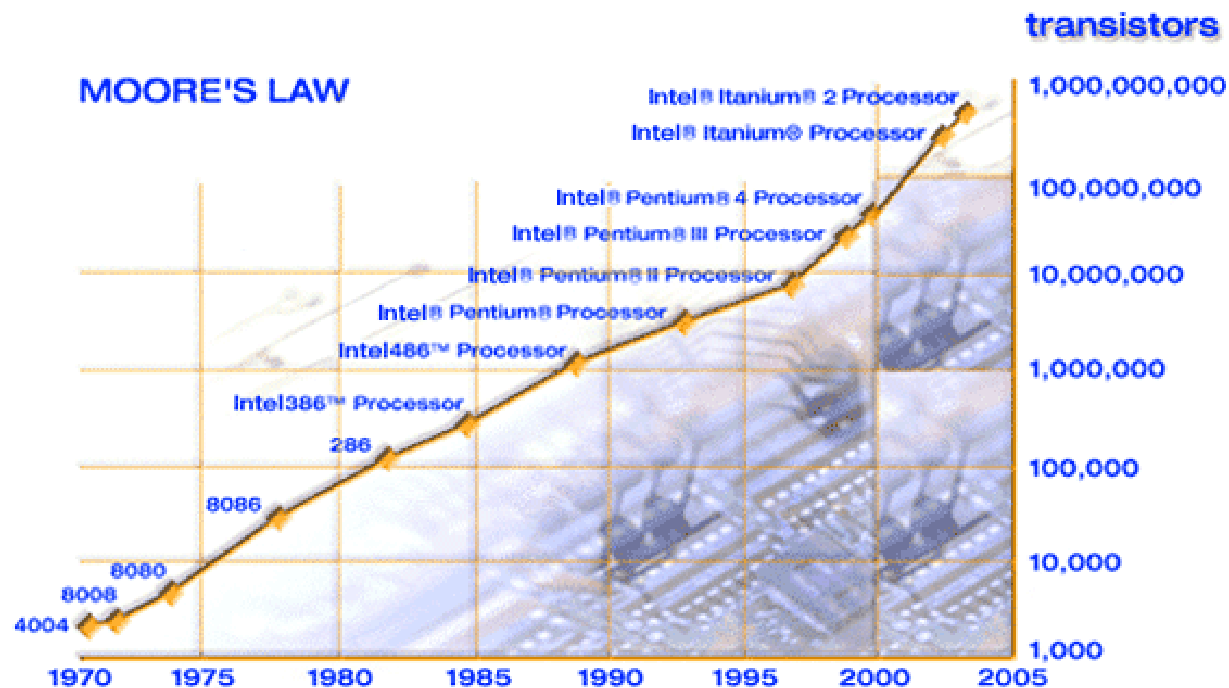
3) **Advantages Over Prior Techniques**

4) **Broader Impact of this Work**

1) Problem Motivation

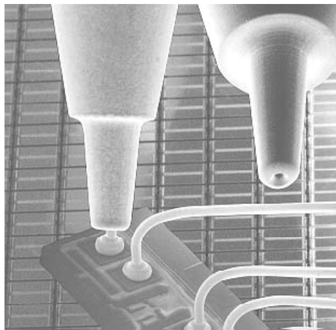
Why is packaging limiting performance?

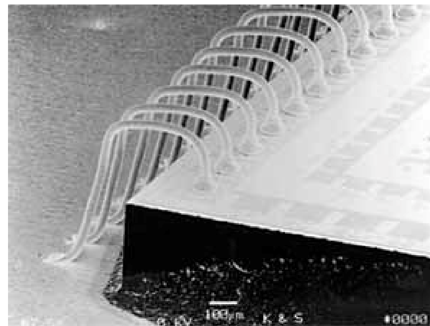
- **IC Design/Fabrication is Outpacing Package Technology**
  - **- We're seeing exponential increase in IC transistor performance**
  - **- >1.3 Billion transistors on 1 die [Fall IDF-05]**

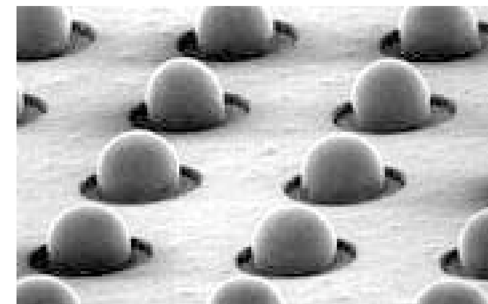


Why is packaging limiting performance?

- **Packages Have Been Designed for Mechanical Performance**
  - **- Electrical performance was not primary consideration**
  - **- IC's limited electrical performance**
  - **- Package performance was not the bottleneck**

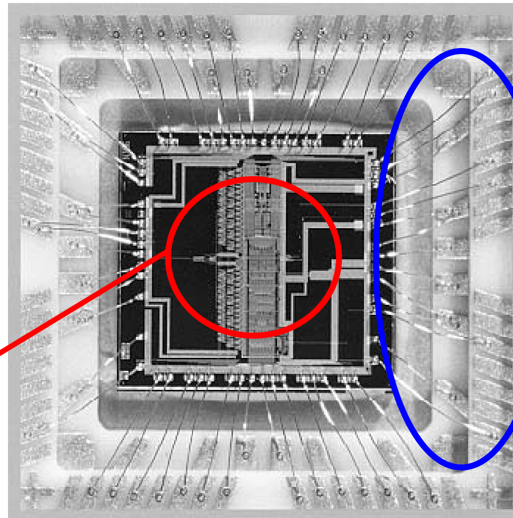






Why is packaging limiting performance?

- **VLSI Performance Exceeds Package Performance**
  - **Packages optimized for mechanical reliability, but still used due to cost**
  - **IC performance far exceeds package performance**



On-Chip

- $f_{IC} > 4GHz$
- large signal counts
- exponential scaling

Package

- $f_{pkg} < 2GHz$
- limited signal counts
- linear scaling

Why is packaging limiting performance?

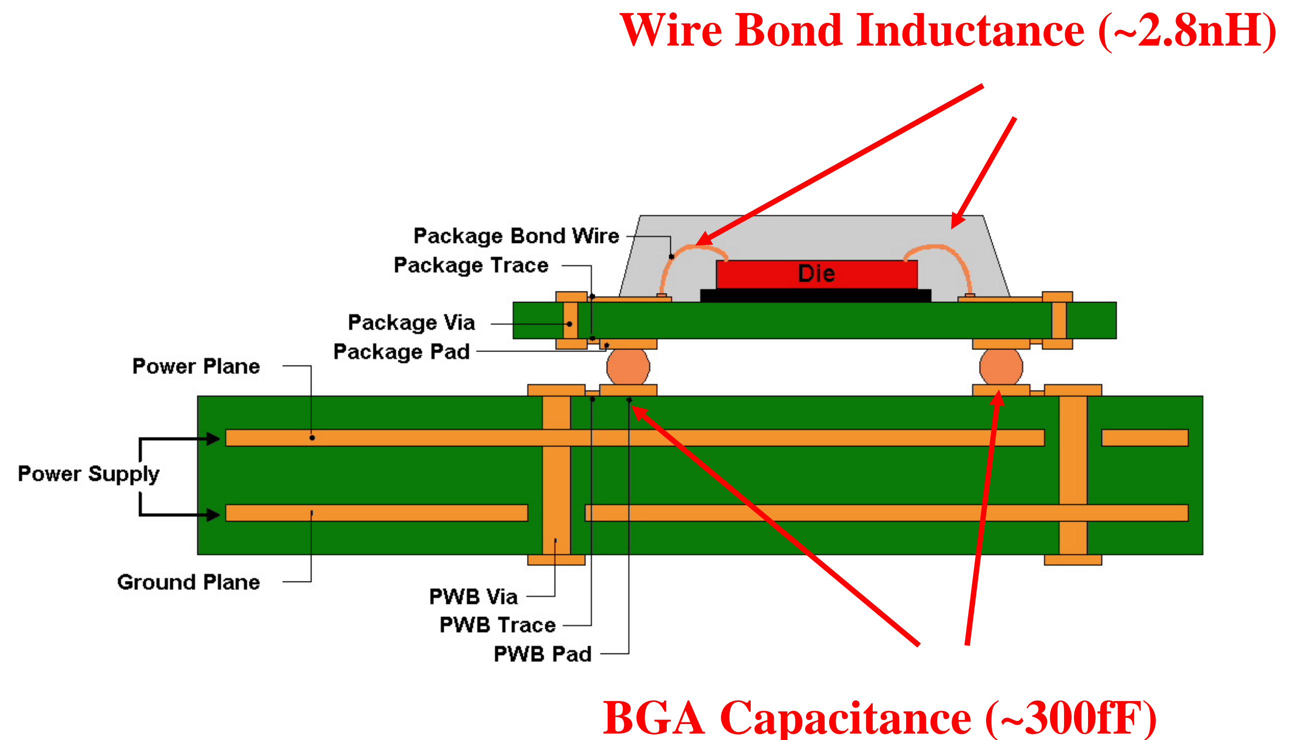
- **Package Interconnect Contains Parasitic Inductance and Capacitance**

 - Long interconnect paths

 - Large return loops

 - $L = \frac{\Phi}{I}$

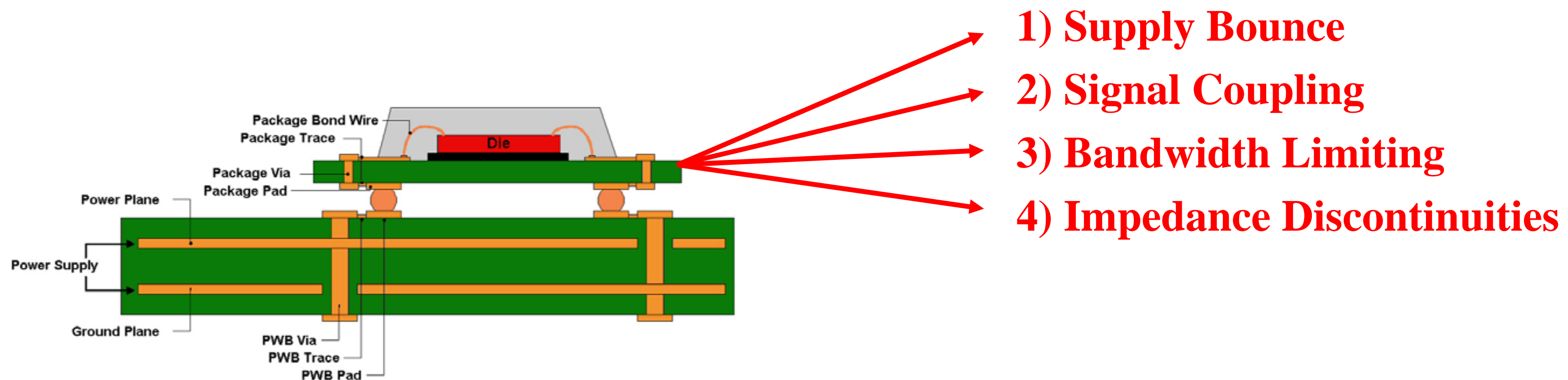
 - $C = \frac{Q}{V} = \frac{A\varepsilon}{t}$



Why is packaging limiting performance?

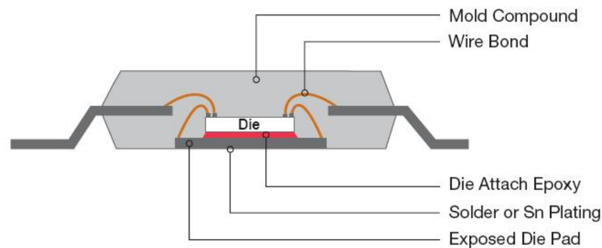
- **Package Parasitics Limit Performance**

  - Excess L and C causes package noise
  - Noise limits how fast the package can transmit date

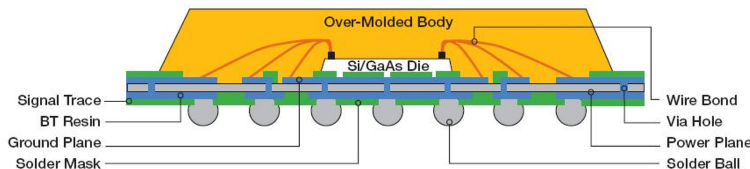


Why is packaging limiting performance?

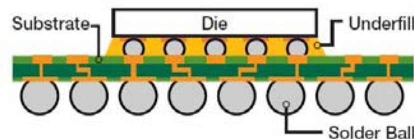
- **Aggressive Package Design Helps, but is expensive…**
  - **- 95% of VLSI design-starts are wire bonded**
  - **- Goal: Extend the life of current packages**



QFP – Wire Bond : 4.5nH → $0.22 / pin



BGA – Wire Bond : 3.7nH → $0.34 / pin *



BGA – Flip-Chip : 1.2nH → $0.63 / pin

2) Research Overview

Research Overview

- **Performance Modeling & Bus Sizing**
  - algebraic model to predict performance and cost-effectiveness

- **Bus Expansion CODEC**
  - encoding data to avoid patterns on bus which cause excessive noise

- **Bus Stuttering CODEC**
  - encoding data to avoid patterns on bus which cause excessive noise

- **Impedance Compensation**
  - adding C or L near package to match impedance to system

Publications: Performance Modeling and Bus Sizing

- **"FPGA I/O – When to go serial",**
  ***IEE Electronic Systems and Software*, 2004**

- **"Performance Model for Inter-Chip Busses Considering Bandwidth and Cost"**
  ***DesignCon*, 2005**
  - ***Best Paper Award***

- **"Performance Model for Inter-chip Com Considering Inductive Cross-talk and Cost",**
  ***ISCAS*, 2005**

- **"Performance Model for Inter-Chip Busses Considering Bandwidth and Cost",**
  ***DesignConEast*, 2005**

- **"Package Performance Model for Off-chip Busses Considering Bandwidth and Cost",**
  IEE Journal on Computers and Digital Techniques (accepted for publication)

Publications: Bus CODECs to Avoid Package Noise

- **"Encoding-based Minimization of Inductive Cross-talk for Off-chip Data Transmission", *DATE*, 2005**

- **"Controlling Inductive Cross-talk and Power in Off-chip Buses using CODECS", ASP-DAC 2006** (accepted for publication)

- **"Bus Stuttering: An Encoding Technique to Reduce Inductive Noise in Off-Chip Data", DATE 2006** (submitted)

Publications: Impedance Compensation

- **"Time Domain Analysis of a Printed Circuit Board Via",**
  Microwave Journal, **2000**

- **"The Effect of Ground Vias on Changing Signal Layers in Multi-Layered PCBs",**
  Microwave and Optical Technology Letters, **2001**

- **"Broadband Impedance Matching for Inductive Interconnect in VLSI Packages",**
  ICCD**, 2005**
  - ***Best Paper Award***

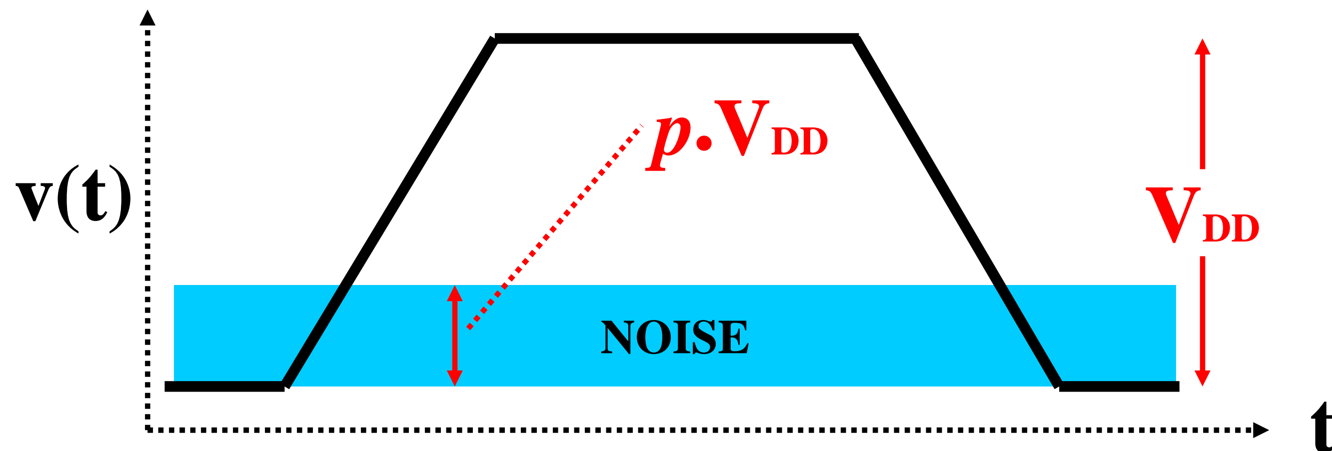
- **"Impedance Matching Techniques for VLSI Packaging",**
  DesignCon**, 2006** (accepted for publication)

New

Performance Modeling

- **Analytical Model To Predict Bus Performance**
  - **VLSI/CAD integration**
  - **Quick hand calculations**

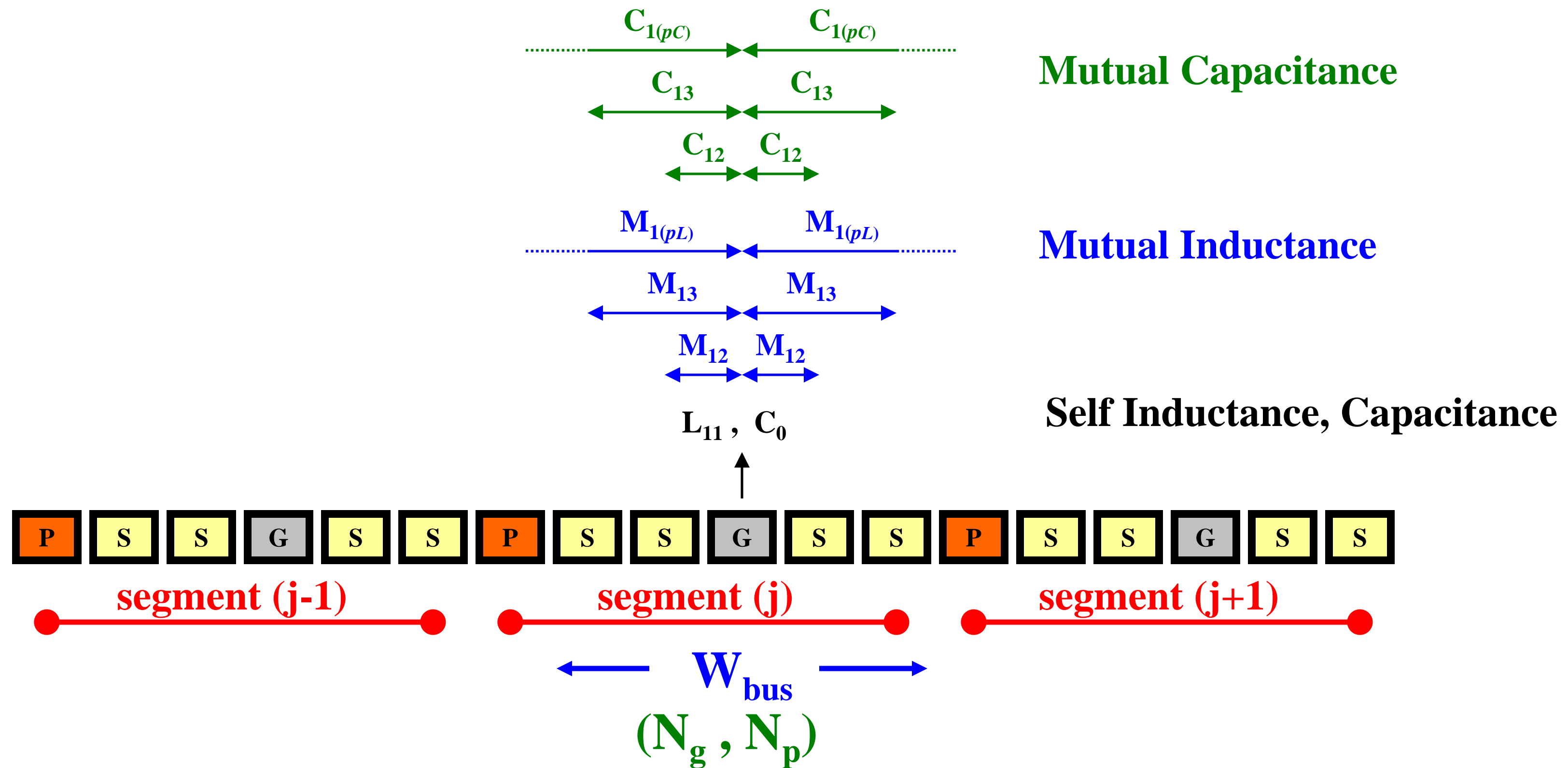
- **We Can Use Package Noise As the Failure Parameter**
  - **Any noise source can be used as limit**
  - **Max ($^{di}/_{dt}$) or ($^{dv}/_{dt}$) is extracted and converted to bus throughput**



New

Performance Modeling

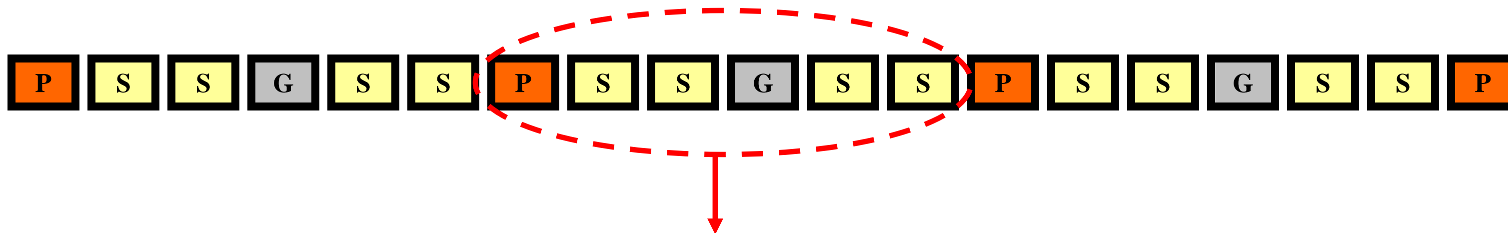
- **Bus Notation**
  - Analysis performed on repetitive *segment*, reducing computation time
  - A scalable framework is used to represent the bus configuration



New

Performance Modeling

- **Use Ground Bounce as Failure Mechanism**



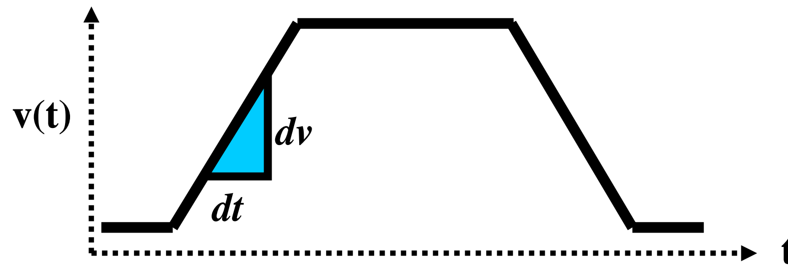
$$V_{gnd-bnc} = \left(\frac{L_{11} \cdot W_{bus}}{N_g}\right)\left(\frac{di}{dt}\right)_k + \sum_{k=-p_L}^{p_L}\left(M_{1|(|k|+1)}\right)\cdot\left(\frac{di}{dt}\right)_k + \sum_{k=-p_C}^{p_C}\left(\frac{C_{1|(|k|+1)} \cdot Z_0^2}{0.8}\right)\cdot\left(\frac{di}{dt}\right)_k = p \cdot V_{DD}$$

Self Contribution — *Mutual Inductive Contribution* — *Mutual Capacitive Contribution* — *Noise Limit*

New

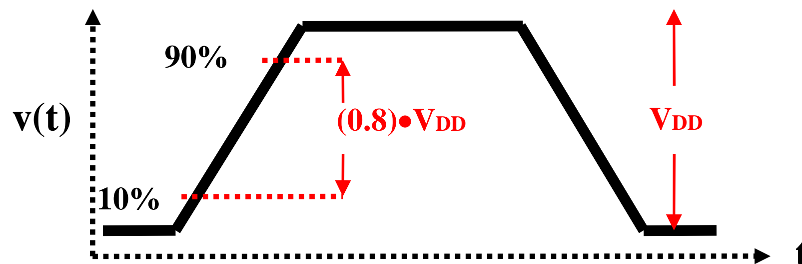
Performance Modeling

∴ Slewrate



$$slewrate = \left(\frac{dv}{dt}\right) = \left(\frac{di}{dt}\right) \cdot Z_{load}$$

∴ Risetime

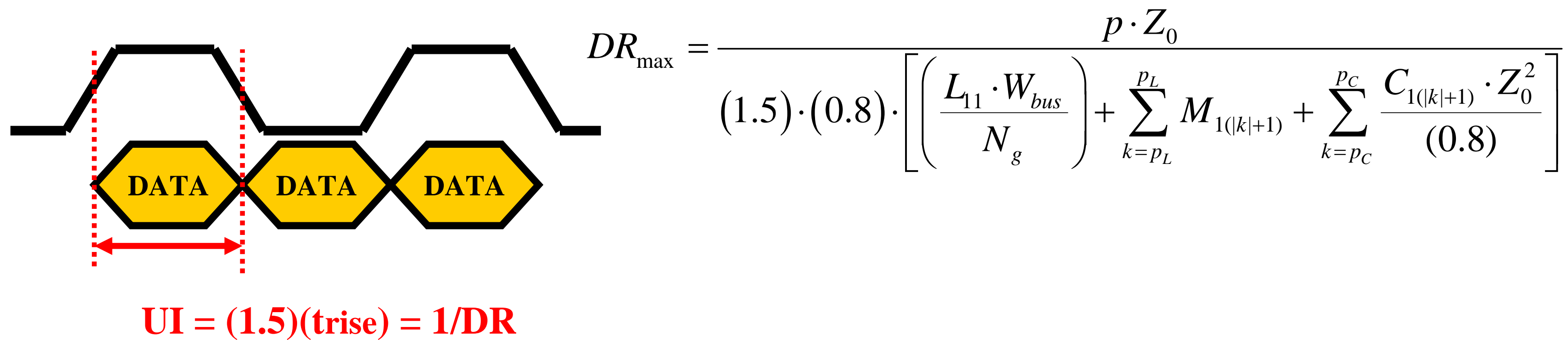


$$t_{rise} = \frac{(0.8) \cdot V_{DD}}{slewrate}$$

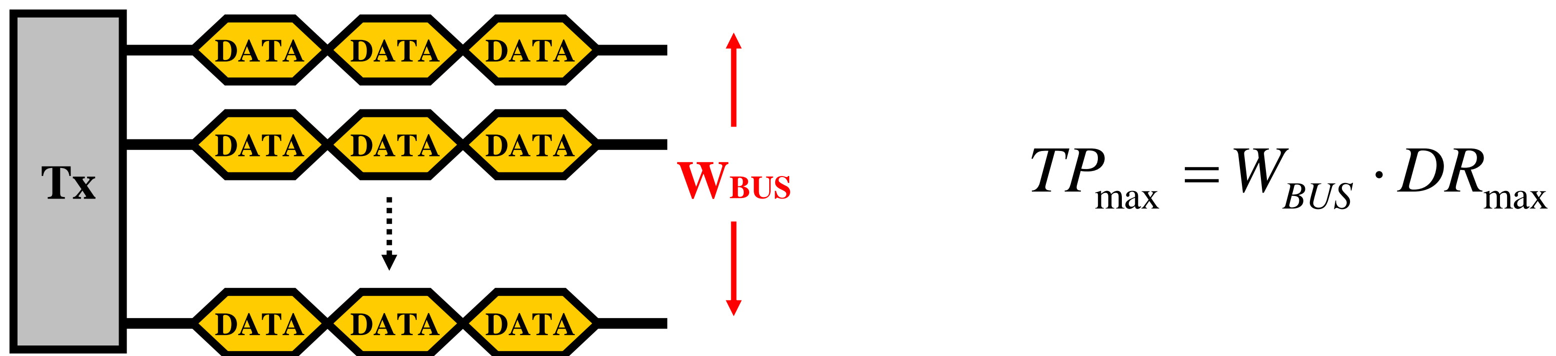
New

Performance Modeling

∴ Datarate



∴ Throughput

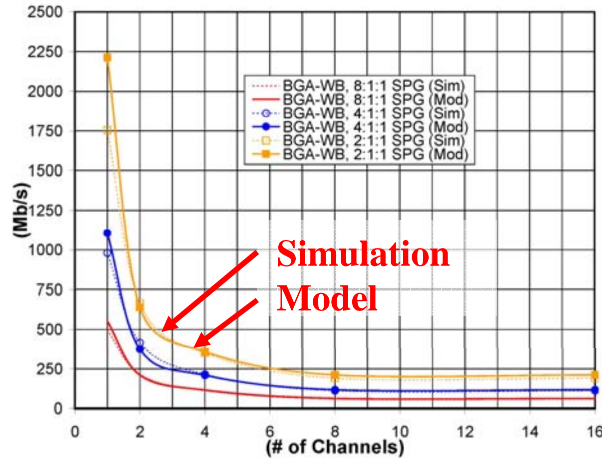


New

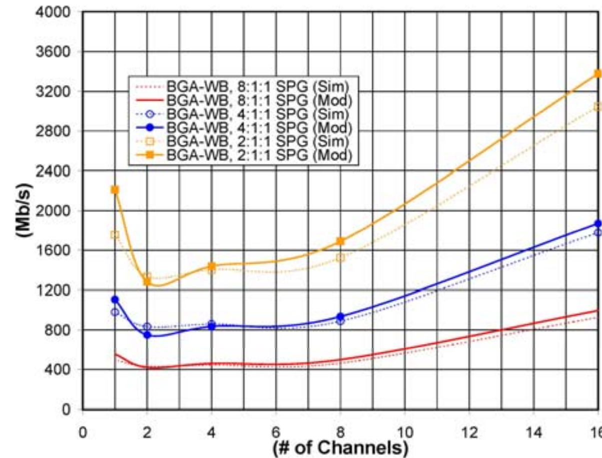
Performance Modeling

• BGA Wire-Bond Package Simulations

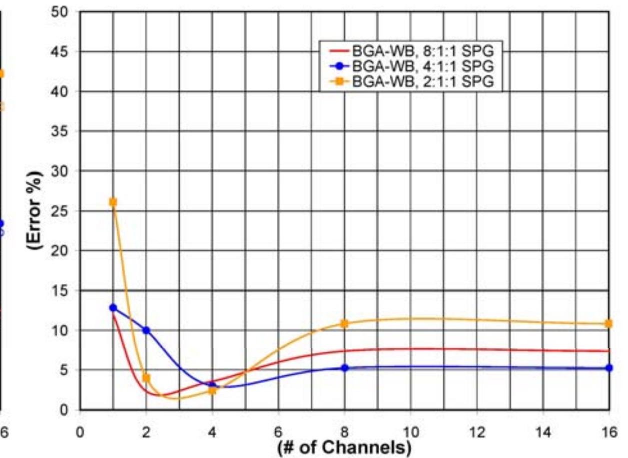
Per-Pin Data-Rate



Bus Throughput



Model Error

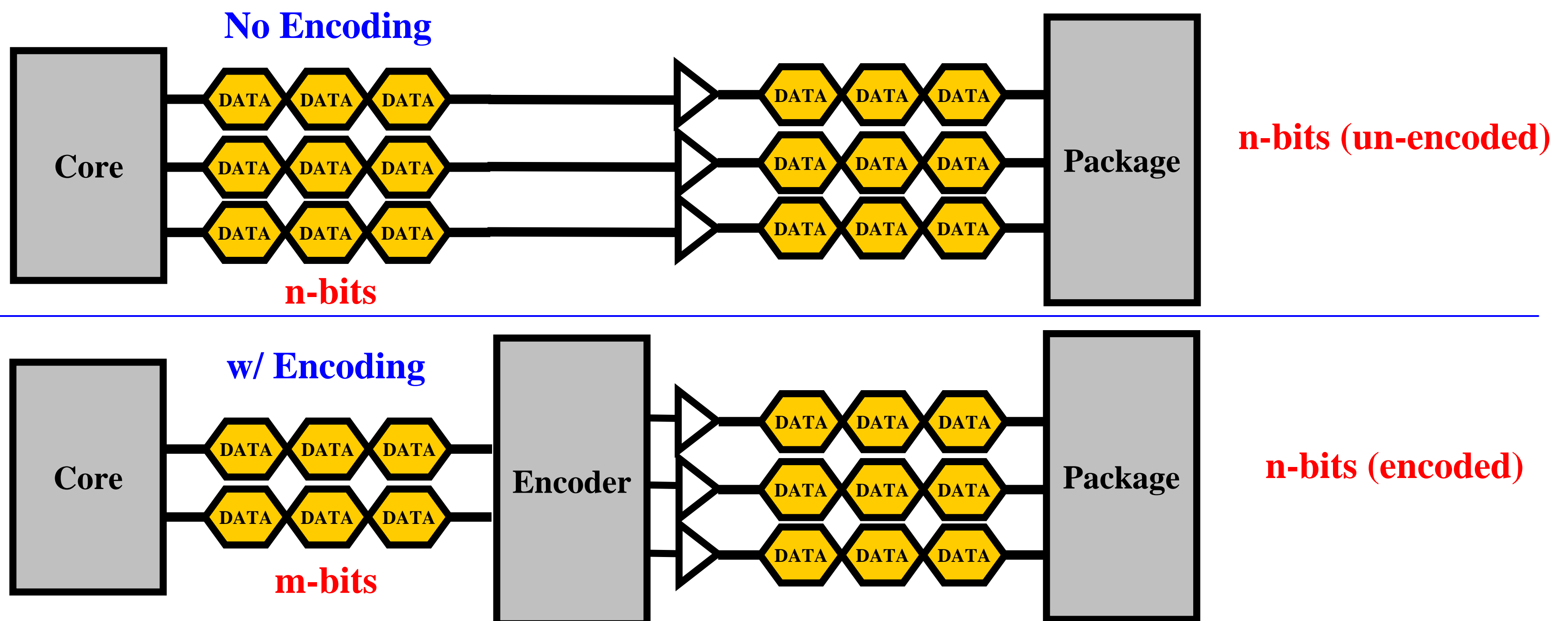


- **Model Matches Simulations to 11% for segments greater than 1 bit**
- **Throughput does not increase linearly as channels are added**

Covered in Comps

Bus Expansion CODEC

- **Encode the Data To Avoid Noise Causing Vector Sequences**
  - **Reducing noise allows faster per-pin datarate**
  - **Throughput is increased even after considering Overhead**
  - **Bus Expansion CODEC maps on-chip bus size (m) into off-chip bus size (n)**

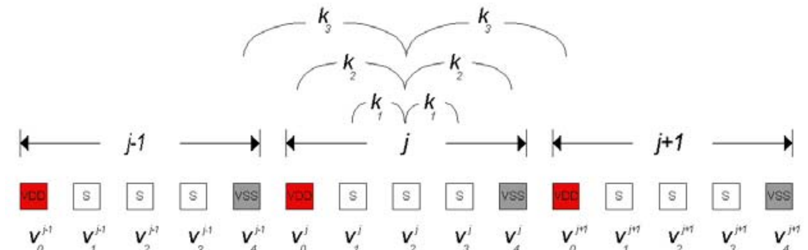


Covered in Comps

Bus Expansion CODEC - Constraints

- **For Each Possible Noise Source on the Bus, a Constraint is written**

1) $v_0^j = VDD \rightarrow -P_{bnc} \geq (L/2)\cdot(\#\text{ of } v_i^j \text{ pins} = 1) \leq Pbnc$
2) $v_1^j = 1 \rightarrow k1\cdot(v_2^j) + k2\cdot(v_3^j) \geq P_1$
3) $v_1^j = -1 \rightarrow k1\cdot(v_2^j) + k2\cdot(v_3^j) \leq P_{-1}$
4) $v_1^j = 0 \rightarrow -P_0 \leq k1\cdot(v_2^j) + k2\cdot(v_3^j) \leq P_0$



- **Each Constraint is Evaluated to Find Illegal Transitions:**

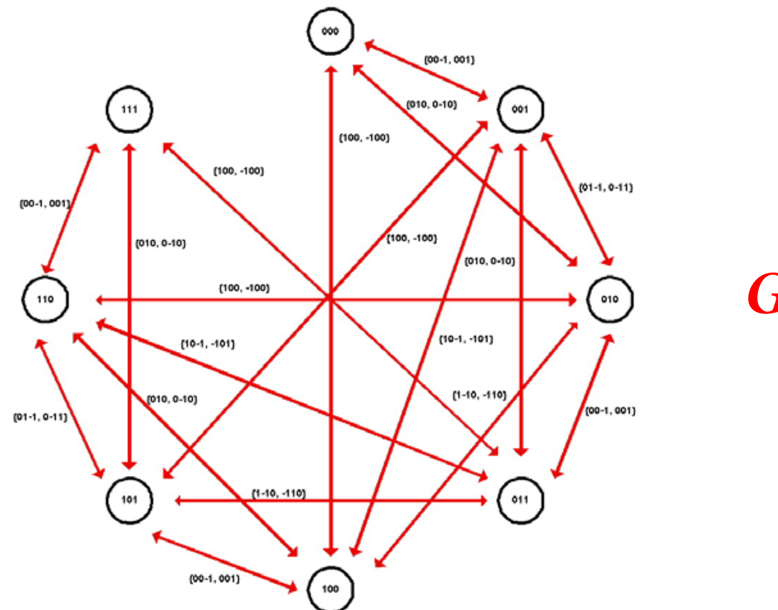
$v_1^j = 1$ = rising
$v_1^j = 0$ = static
$v_1^j = -1$ = falling

v_1^j	v_2^j	v_3^j	
1	0	1	
1	-1	0	
~~1~~	~~-1~~	~~-1~~	violates user-defined "*glitch*" parameter
1	-1	1	
1	1	0	
~~1~~	~~1~~	~~-1~~	violates user-defined "*supply*" bounce parameter
1	1	1	

Covered in Comps

Bus Expansion CODEC - Algorithm

- **The Remaining *Legal* Transitions Construct a Directed Graph**



G

- **The Directed Graph is evaluated to see if an *m*-bit bus can be encoded**

- A closed set of nodes *S* must exist such that:

- **$|S| \geq 2^m$**
- **each vertex *s* in *S* has at least 2^m outgoing edges to vertices *s'* in *S***

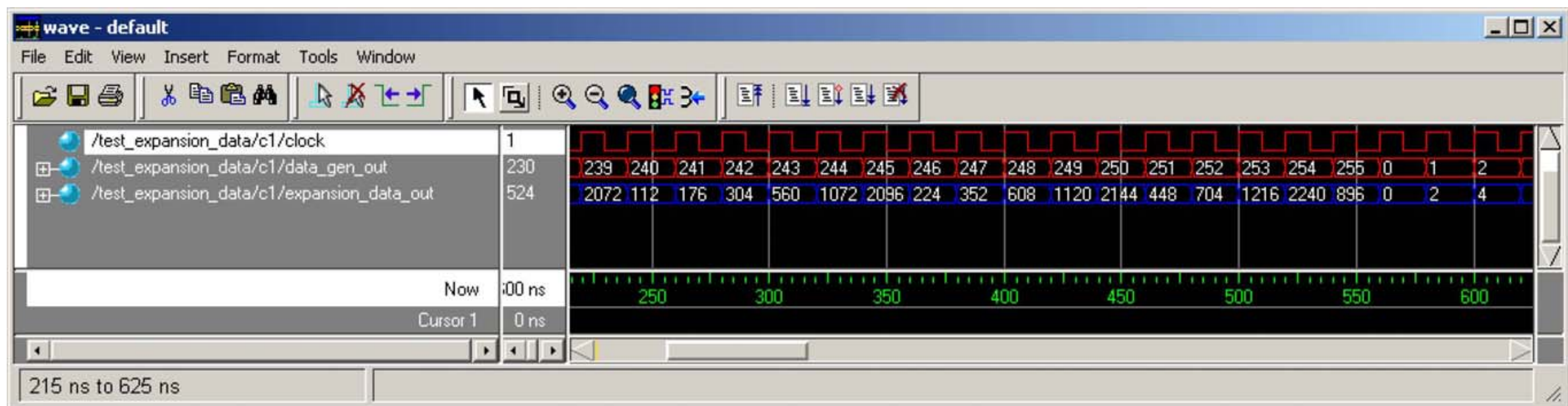
New

Bus Expansion CODEC – Physical Results

• TSMC 0.13um Synthesis Results

- RTL design, synthesized and mapped
- Segment sizes 2 → 8 implemented
- Logic, delay, and area evaluated

	Bus Size (m)	Noise Limit	
	-	5% (aggressive)	10% (non-aggressive)
Delay (ns)	2	0.170	encoder not required
	4	0.670	0.503
	6	1.150	0.955
	8	1.310	0.983
Area (um^2)	2	22	encoder not required
	4	152	114
	6	614	509
	8	1,181	886



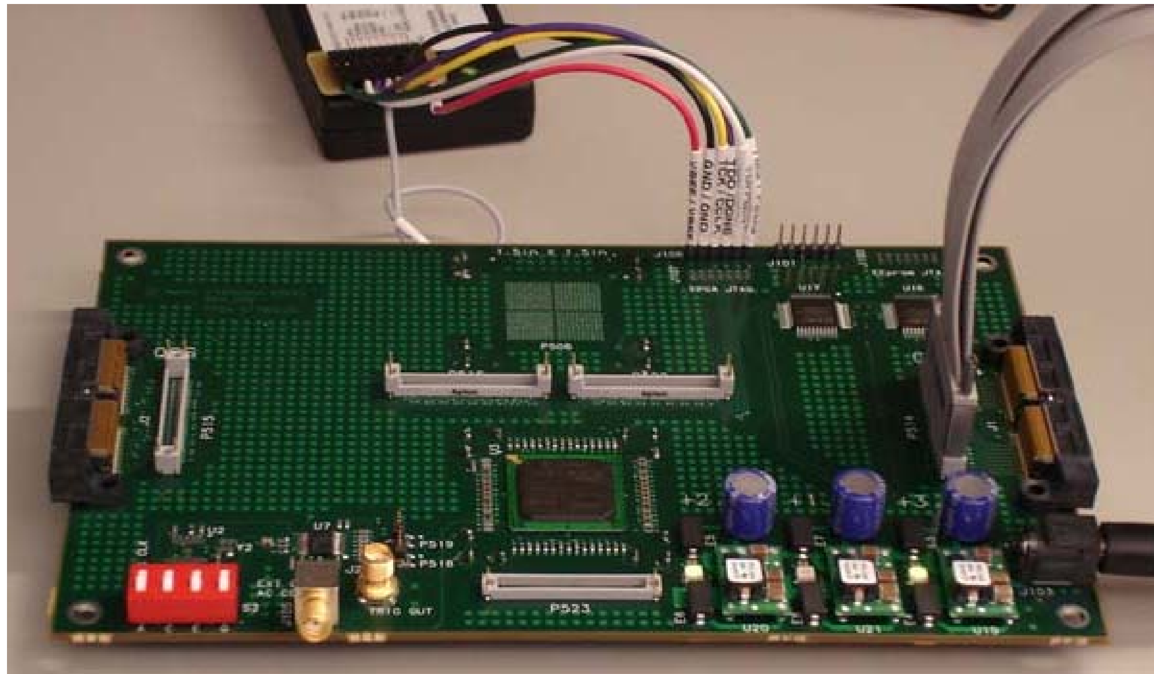
New

Bus Expansion CODEC – Physical Results

- **Xilinx FPGA, 0.35um Implementation Results**

- RTL design implemented

- Xilinx, VirtexIIPro, FPGA



New

Bus Expansion CODEC – Physical Results

• Xilinx FPGA, 0.35um Implementation Results

- RTL design, implemented
- Segment sizes 2 → 8 measured
- Logic operation verified
- Noise Reduced from 16% to 4%

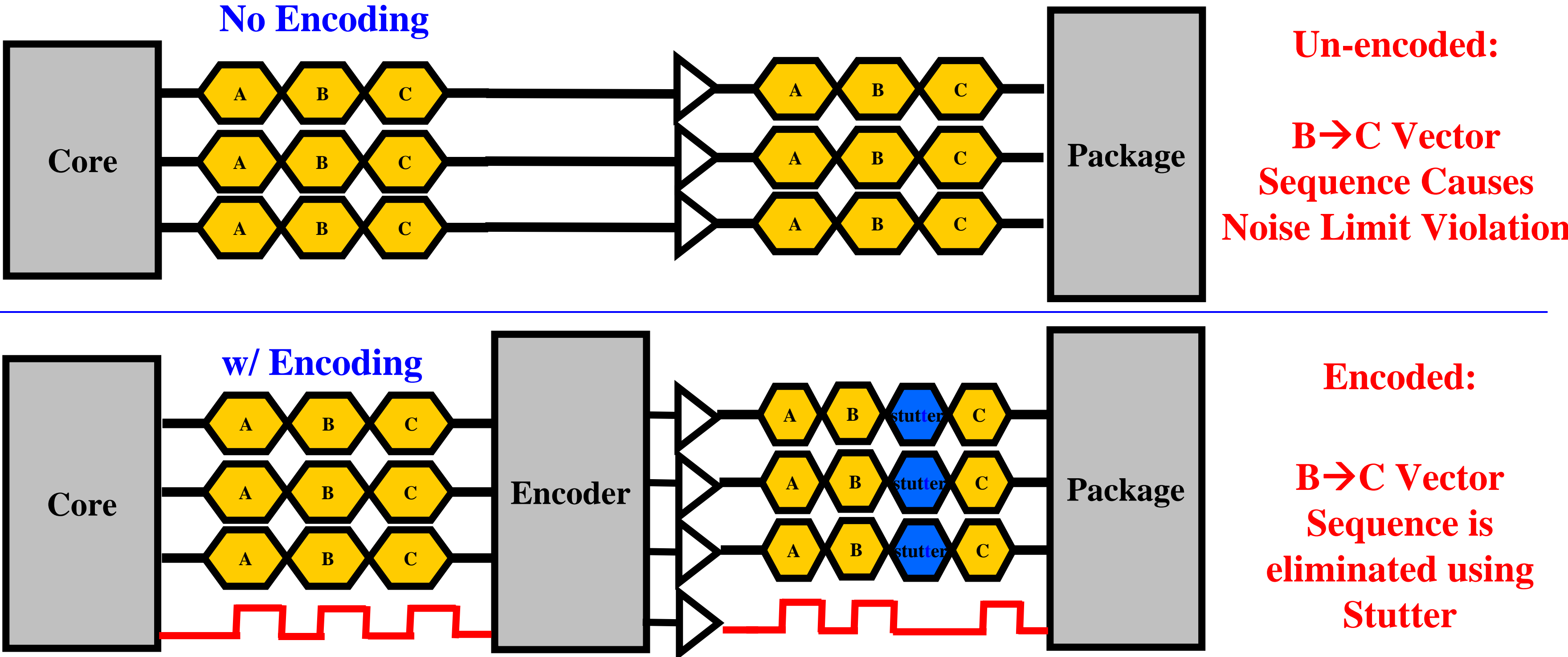
(3 bit, SPG=4:1:1)

	Bus Size (m)	Noise Limit
	-	5% (aggressive) & 10% (non-aggressive)
Delay (ns)	2	0.351
	4	1.020
	6	1.450
	8	1.610
FPGA Usage	2	$< 1\%$
	4	$< 1\%$
	6	$< 1\%$
	8	$< 1\%$
FPGA Implementation	2	3x, 2-Input FG's
	4	6x, 4-Input FG's
	6	9x, 6-Input FG's
	8	12x, 8-Input FG's

New

Bus Stuttering CODEC

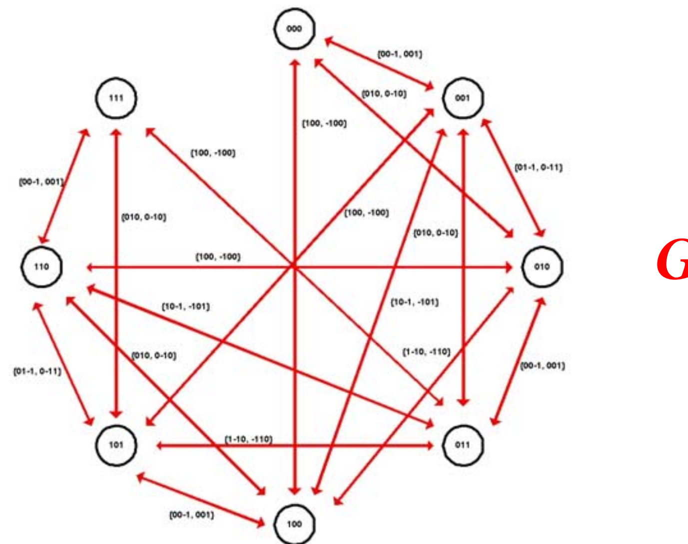
- **Intermediate States are Inserted Between Noise Causing Transitions**
  - *Stutter* states limit the number of simultaneously switching signals
  - The source synchronous clock is gated during stutter state



New

Bus Stuttering CODEC - Algorithm

- **Constraints are Evaluated and a Legal Directed Graph is Created**



G

- **Directed Graph is Used to Map Transitions Between any Two Vectors**
  - **A transition path (which may include stutters) exists between any two vectors if:**
    - **There exists at least two outgoing edges for each vector $v_s \in G$ (including self-edge)**
    - **There exists at least two incoming edges for each vector $v_d \in G$ (including self-edge)**

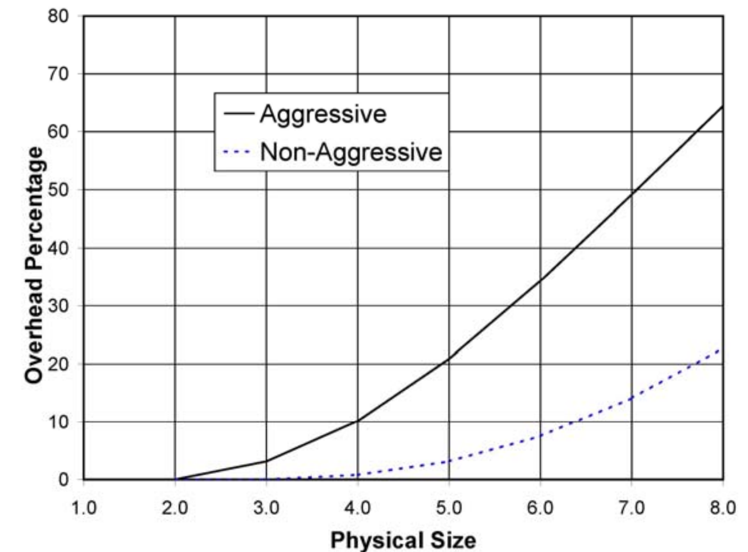
New

Bus Stuttering CODEC - Construction

- **Multiple Stutter States can be used**
  - between 0 and $2^{(W_{bus}-1)}$ stutters can be inserted between any two vectors
  - experimental results show that for segments up to 8 bits, more than 3 stutters is rare

- **Overhead**
  - Overhead increases as segments sizes increase
  - Still useful since segments greater than 8 bits are rarely used (SPG=8:1:1)

$$Overhead = \frac{\sum_{k=1}^{2^{(W_{bus}-1)}} (\#_Trans_Requiring_k_stutters) \cdot k}{2^{(2 \cdot W_{bus})}}$$

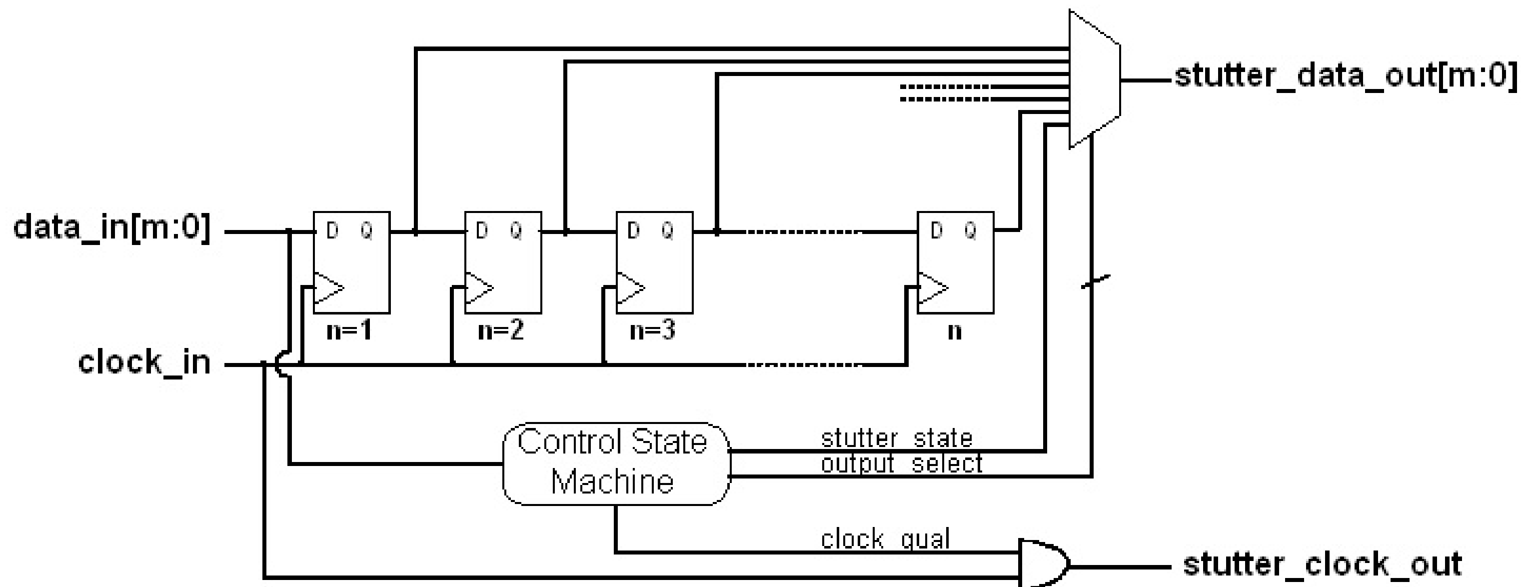


New

Bus Stuttering CODEC – Physical Results

• Circuit Implementation

- 32 pipeline stages used
- pipeline reset after 32 idle states (similar to SRIO, HT, and PCI Express)
- protocol inherently handles pipeline overflow



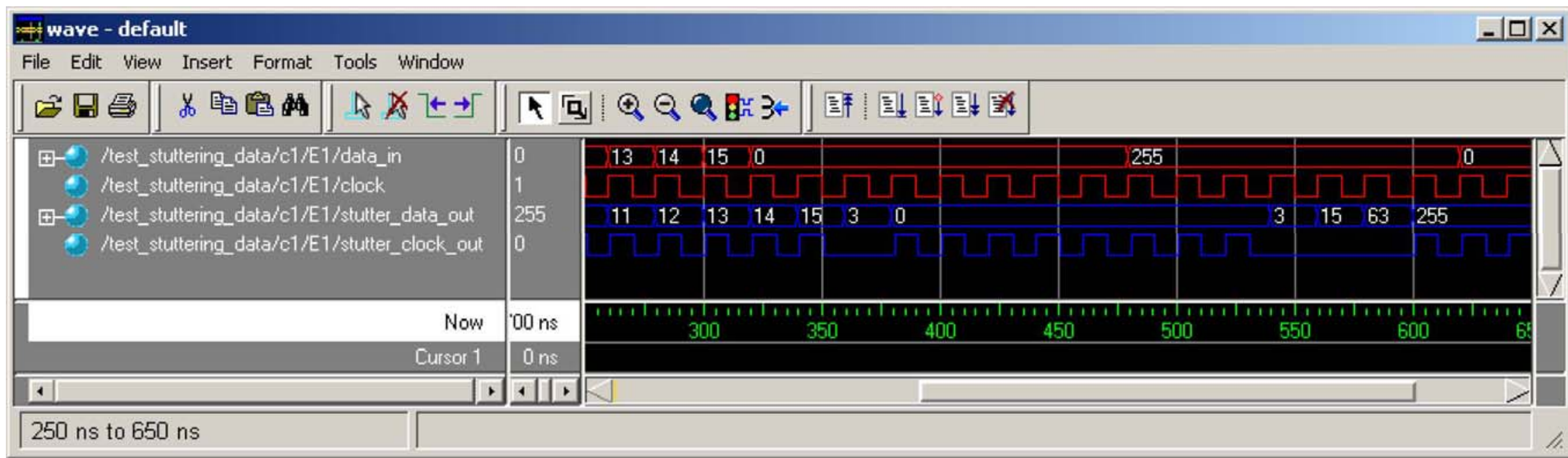
New

Bus Stuttering CODEC – Physical Results

• TSMC 0.13um Synthesis Results

- RTL design, synthesized and mapped
- Segment sizes 2 → 8 implemented
- Logic, delay, and area evaluated

	Bus Size	Noise Limit	
	-	5% (aggressive)	10% (non-aggressive)
Delay (ns)	4	2.02	1.99
	6	2.42	2.38
	8	2.85	2.79
Area (um^2)	4	311k	310k
	6	362k	345k
	8	382k	368k



New

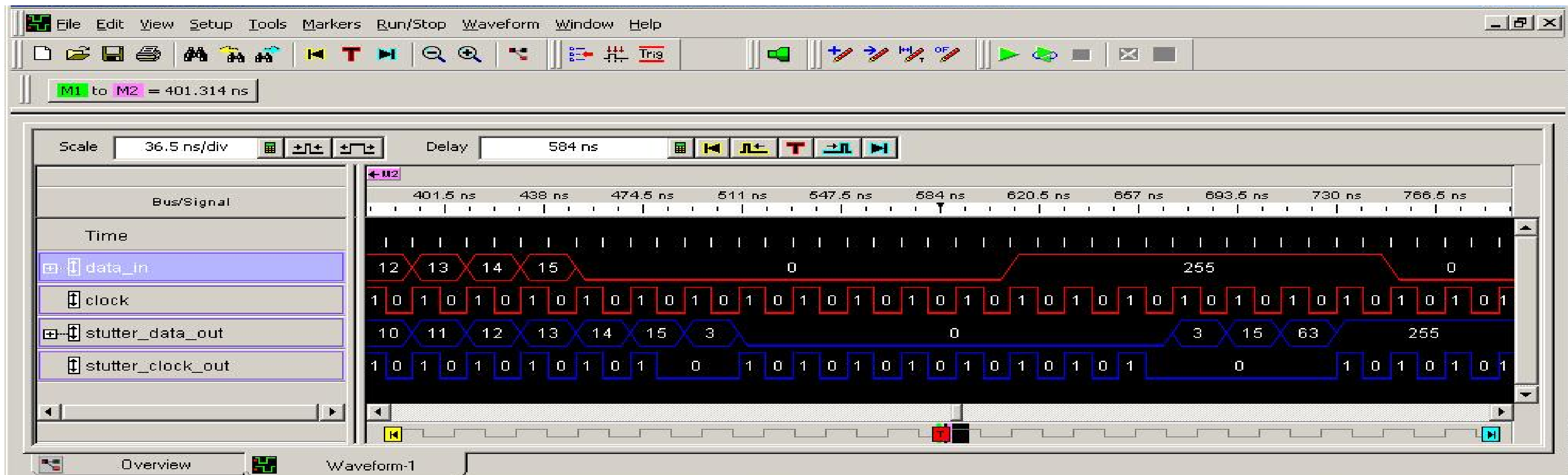
Bus Stuttering CODEC – Physical Results

• Xilinx FPGA, 0.35um Implementation Results

- RTL design, implemented
- Segment sizes 2 → 8 measured
- Logic operation verified
- Noise Reduced from 16% to 4%

(3 bit, SPG=4:1:1)

	Bus Size	Noise Limit
	-	5% (aggressive) & 10% (non-aggressive)
Delay (ns)	4 6 8	4.78 5.29 5.89
FPGA Usage	4 6 8	$< 1\%$ $< 1\%$ $< 1.5\%$



Covered in Comps

Impedance Compensation

- **Add Capacitance Near Bond Wire to Reduce Impedance**

- Adding additional capacitance lowers the wire bond impedance
- Impedance can be matched to system, reducing reflections

$$Z_{WireBond} = \sqrt{\frac{L_{WireBond}}{C_{WireBond}}}$$

Add Capacitance to lower Z

$$\Gamma = \frac{Z_L - Z_0}{Z_L + Z_0}$$

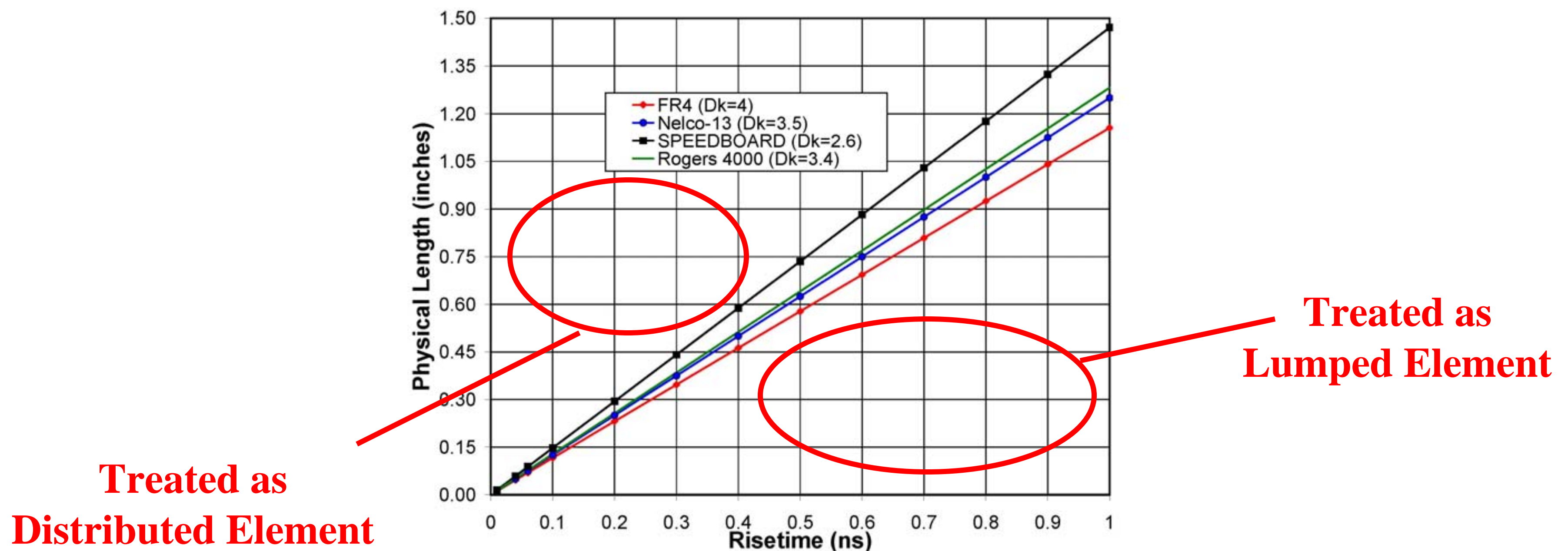
Better Impedance Match results in less reflections

Covered in Comps

Impedance Compensation

- **If the capacitance is close to the wire bond, it will alter its impedance**

- Electrical lengths less than 20% of risetime are treated as lumped elements
- For modern dielectrics, anything within 0.15" of wire bond is lumped

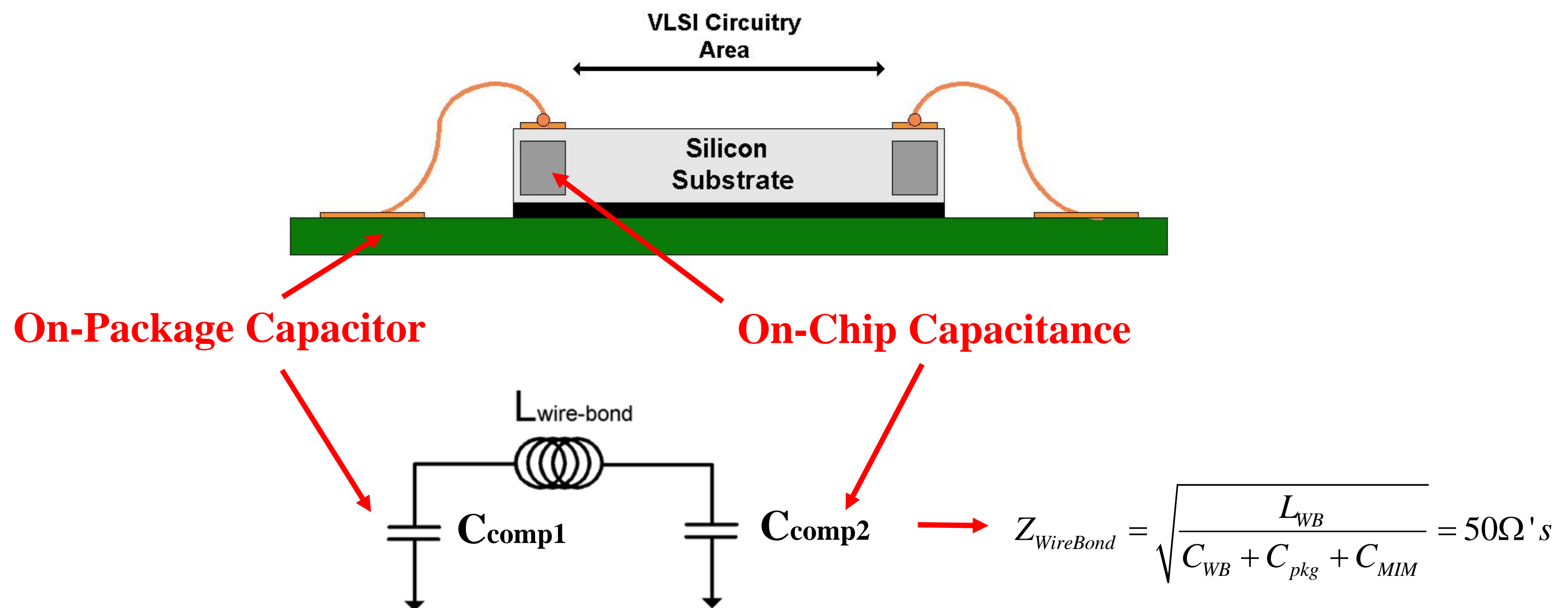


Covered in Comps

Static Impedance Compensator

- **Capacitor values chosen prior to fabrication**

  - Equal amounts of capacitance are used on-chip and on-package



$$Z_{WireBond} = \sqrt{\frac{L_{WB}}{C_{WB} + C_{pkg} + C_{MIM}}} = 50\Omega's$$

Static Impedance Compensator

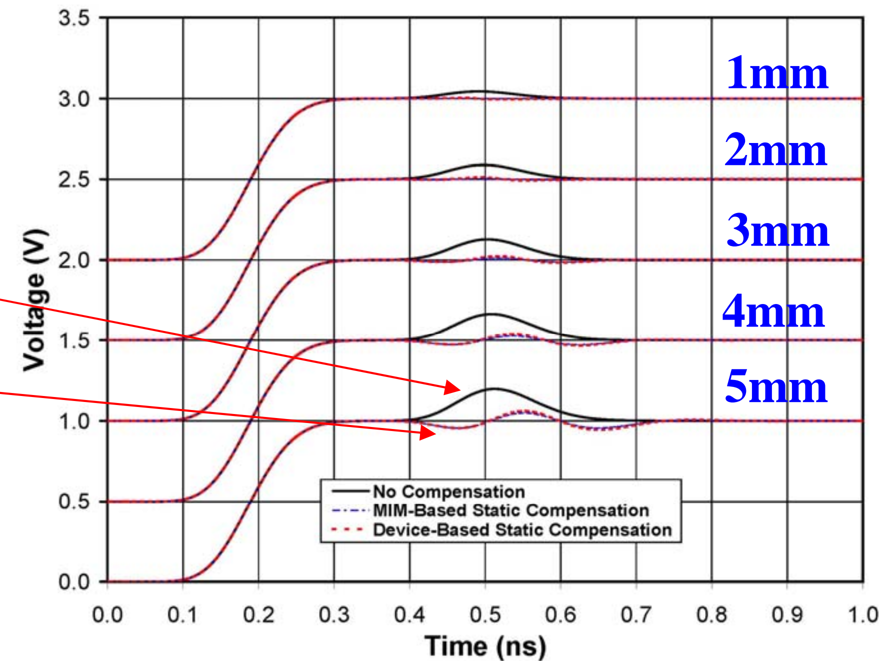
- **Time Domain Analysis (TDR)**

$Length_{wb}$	$\Gamma_{No-Comp}$	$\Gamma_{MIM-Comp}$	$\Gamma_{Device-Comp}$
1 mm	4.5%	0.05%	0.5%
2 mm	8.7%	0.4%	1.2%
3 mm	12.7%	1.3%	2.4%
4 mm	16.4%	2.7%	4.1%
5 mm	19.8%	4.8%	6.0%

Worst Case : 5mm

No Static Capacitance = 19.8%

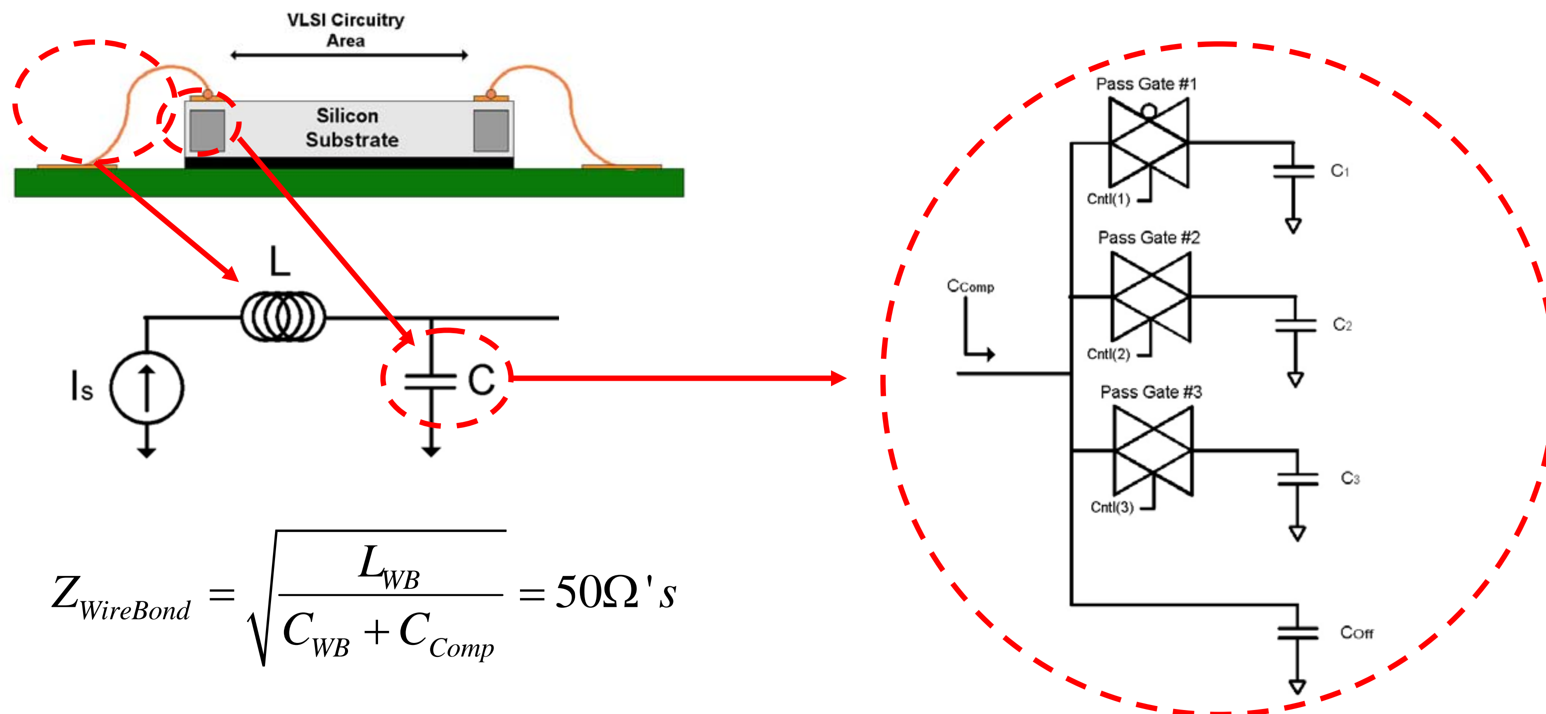
w/ Static Capacitance = 4.8%



Dynamic Impedance Compensator

- **Pass Gates are used to switch in on-chip capacitors**

  - Pass gates connect on-chip capacitance to the wire bond inductance
  - Pass gates have control signals which can be programmed after fabrication



Covered in Comps

Dynamic Impedance Compensator

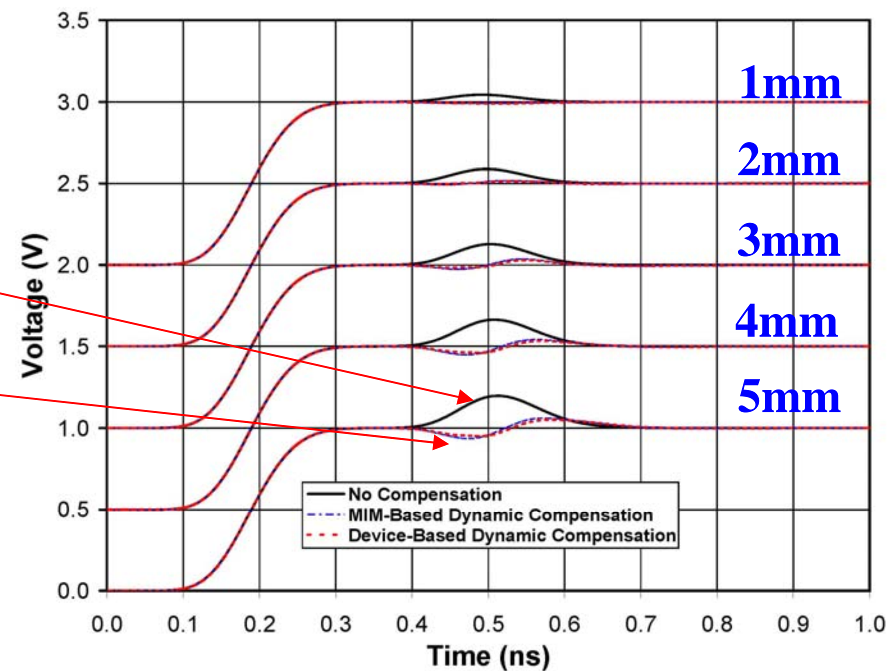
- **Time Domain Analysis (TDR)**

$Length_{wb}$	$\Gamma_{No-Comp}$	$\Gamma_{MIM-Comp}$	$\Gamma_{Device-Comp}$	Setting
1 mm	4.5%	1.0%	1.0%	001
2 mm	8.7%	1.8%	1.3%	011
3 mm	12.7%	3.6%	3.0%	100
4 mm	16.4%	4.3%	3.3%	110
5 mm	19.8%	6.0%	5.0%	111

Worst Case : 5mm

No Dynamic Capacitance = 19.8%

w/ Dynamic Capacitance = 6.0%



3) Advantages Over Prior Techniques

Performance Modeling and Bus Sizing

- **Currently Packages are Modeled Using SPICE**
  - Analog simulators are computationally expensive [BSIM, BPTM}
  - Time of simulation reduces the number of configurations to be evaluated [Agilent Ft. Collins]

- **Model is Linear in the size of the bus**
  - Fast computation is enabled using key assumptions
  - More configurations can be evaluated, which expands usefulness
  - Narrows hundreds of configurations into 2 or 3 for SPICE evaluation

- **Cost is Considered**
  - Analog simulators do not account for cost
  - This adds even more time to analysis

Bus CODECs to Avoid Package Noise

- **Current Approaches Have Physical Limitations**
  - Operate by reducing ($^{di}/_{dt}$) or skewing transitions [pipeline_damping, Multi-Level]
  - Reducing ($^{di}/_{dt}$) will ultimately limit performance
  - Skewing data increases data invalid window, will ultimately limit performance

- **Our CODECs operate above the physical layer**
  - Only data vectors are altered
  - Off-chip drivers are left unchanged, no skewing is necessary
  - This allows usefulness up to higher frequencies
  - This also allows implementation in various process and package technologies

Impedance Compensation

- **Currently, Package Interconnect is Not Addressed**
  - **Only primary impedance is terminated (i.e., the PCB T-line)** [HS_Design, MGT]
  - **No broadband solution exists**

- **Our Techniques Target Package Directly**
  - **Impedance of wire bond or bumping can be addressed**
  - **Broadband operation suited well for digital VLSI**

- **Static Compensator**
  - **Developed using embedded construction, no cost**
  - **Simple and requires no active circuitry**

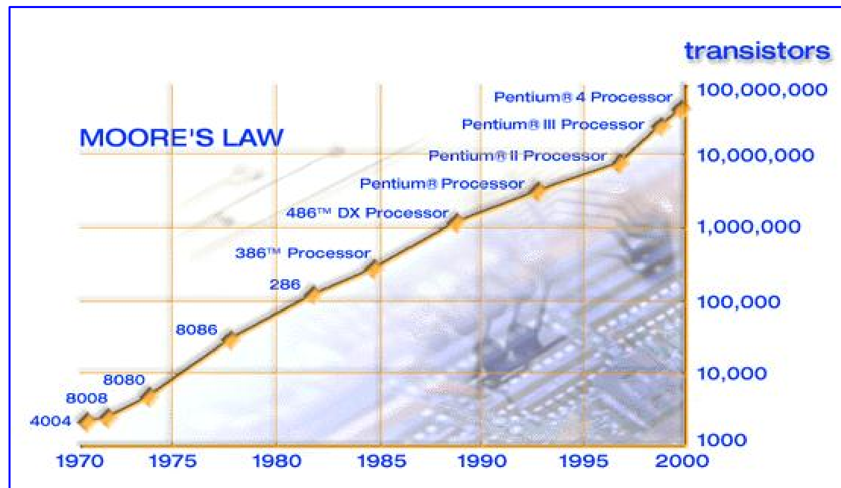
- **Dynamic Compensator**
  - **Accounts for process variation by allowing programmability after fabrication**

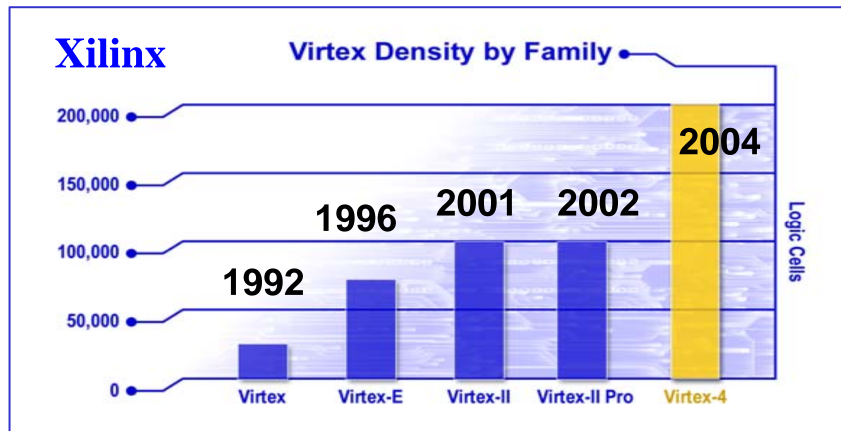
Comp Question

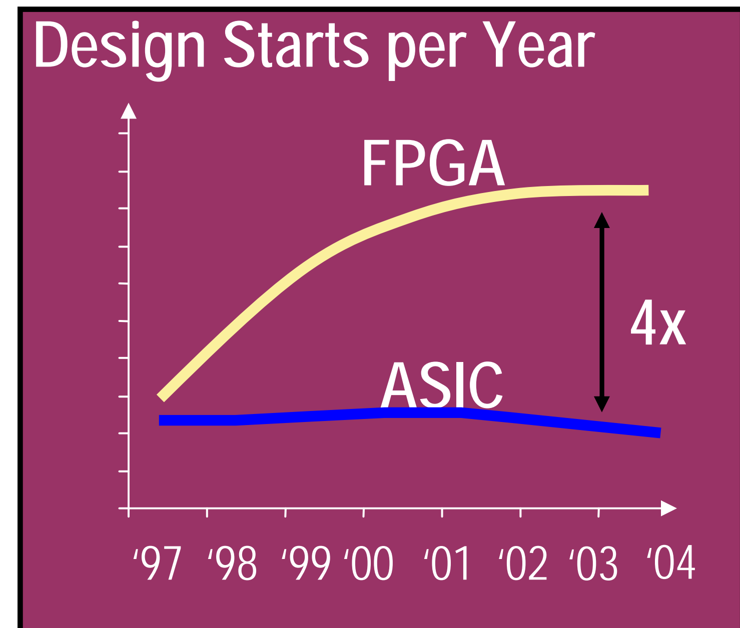
4) Broader Applications of this Work

The Move Toward FPGAs

- **80% of Design Starts Have FPGAs**

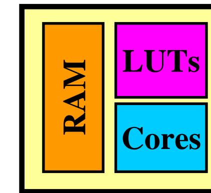






The Move Toward FPGAs

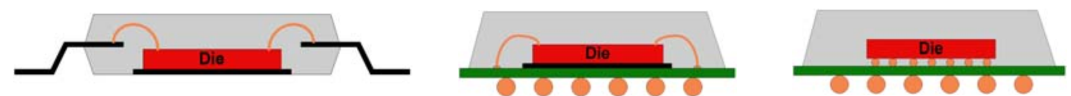
- **FPGA Business Model**
  - Single design is packaged in multiple technologies
  - This enables multiple performance price-points
  - Designer cannot optimize for particular package



Wire-Bond

Flip-Chip

- Performance Modeling
- Noise Reduction CODECs
- Impedance Compensators

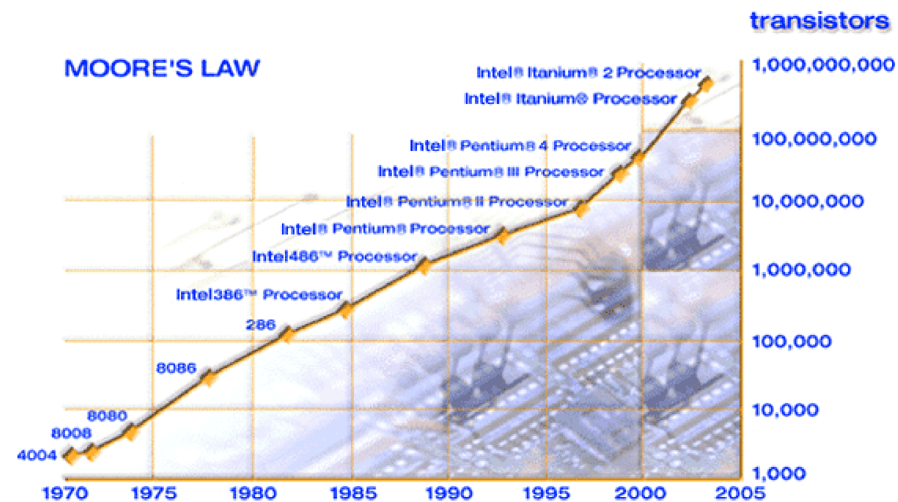


Cost / Performance

Power Minimization

- **Power is Predicted to Limit Moore’s Law**

  - Large amounts of power are consumed in the off-chip drivers
  - CODECs can remove patterns which result in noise violations
  - CODECs can also remove patterns with high power consumption

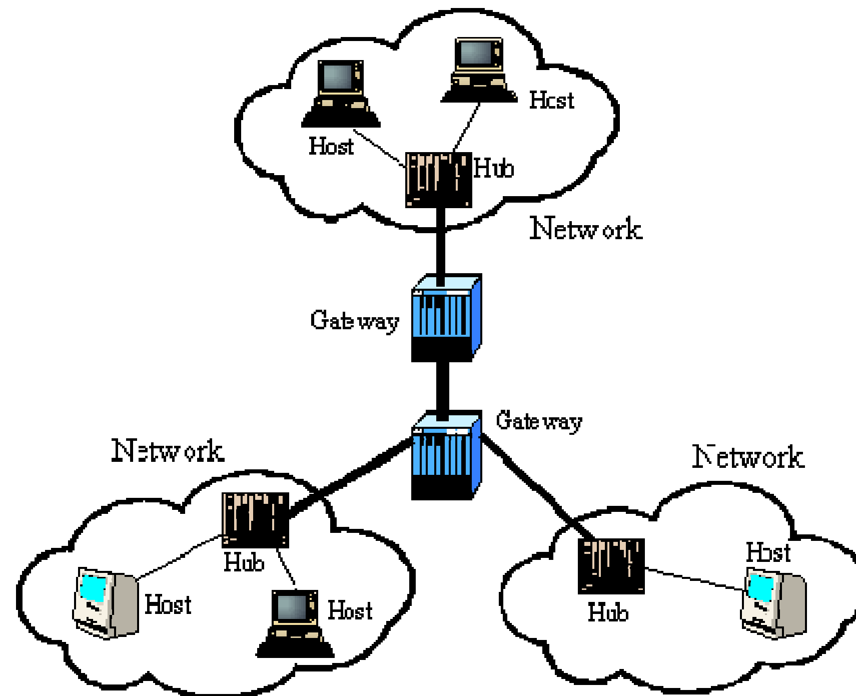


> 100W [ITRS]

Internet Fabric

- **Network Congestion Slows Internet Performance**

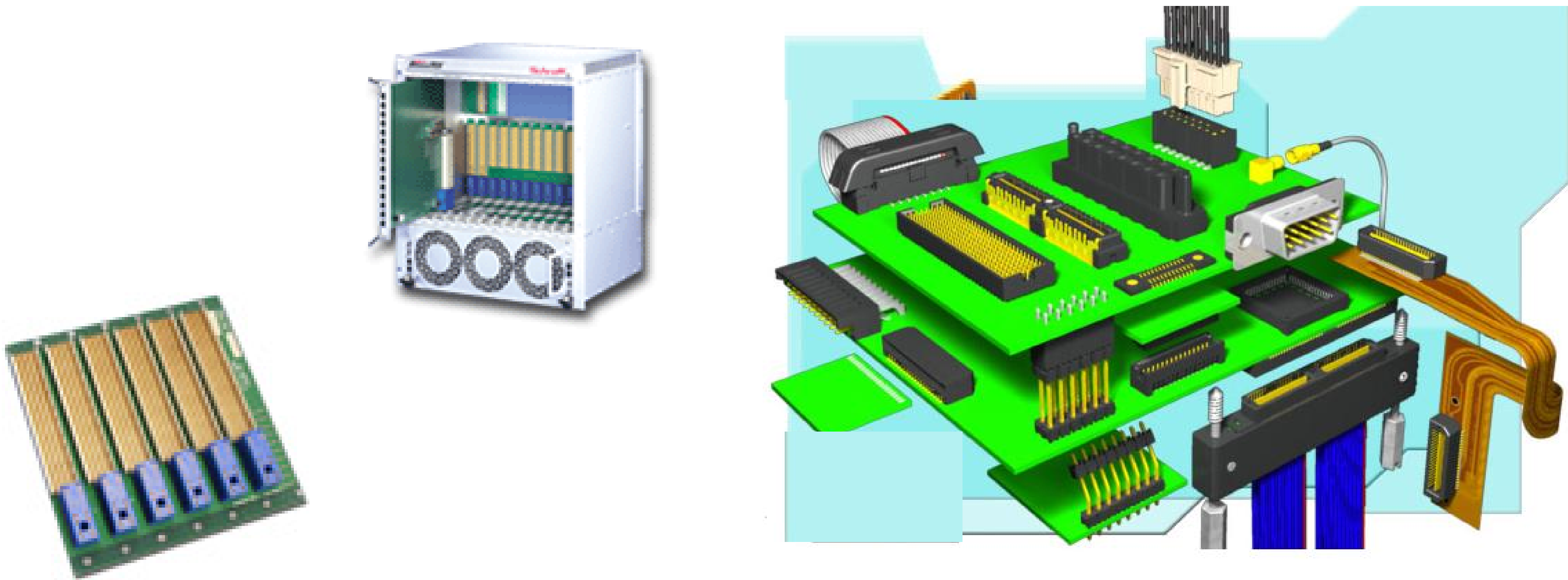
  - **CODECs can remove patterns which result in noise violations**
  - **Can extend CODECs to remove redundant patterns in streaming A/V**



Backplanes and Connectors

• **All Interconnect Has Parasitic Inductance and Capacitance**

- **Backplanes are popular to provide design segmentation and scalability**
- **Connectors are present in all digital designs**
- **Modeling, CODECs, and Compensation can be applied to backplanes/connectors**



Questions?