Encoding-based Minimization of Inductive Cross-talk for Off-Chip Data Transmission

Motivation

- **Power delivery is the biggest challenge facing designers entering DSM**
  - The IC core current continues to increases (P4 = 80Amps).
  - The package interconnect inductance limits instantaneous current delivery.
  - The inductance leads to ground and power supply bounce.

- **SSN on signal pins is the leading cause of inter-chip bus failure**
  - Ground/power supply bounce causes unwanted switching.
  - Mutual Inductive cross-talk causes edge degradation which limits speed.
  - Mutual Inductive cross-talk causes glitches which results in unwanted switching.

- **Aggressive package design helps, but is too expensive:**
  - Flip-Chip technology can reduce the interconnect inductance.
  - Flip-Chip requires a unique package design for each ASIC.
  - This leads to longer process time which equals cost.
  - 90% of ASIC design starts use wire-bonding due to its low cost.
  - Wire-bonding has large parasitic inductance that must be addressed.

Our Solution

“Encode Off-Chip Data to Avoid Inductive Cross-talk”

- **Avoid the following cases:**

1) **Excessive switching in the same direction** **= reduce ground/power bounce**

2) **Excessive X-talk on a signal when switching** **= reduce edge degradation**

3) **Excessive X-talk on signal when static** **= reduce glitching**

Our Solution

- **This results in:**

1) A subset of vectors is transmitted that avoids inductive X-talk.

2) The off-chip bus can now be ran at a higher data rate.

3) The subset of vectors running faster can achieve a higher throughput over the original set of vectors running slower.

(Throughput of less vectors at higher data-rate) > (Throughput of more vectors at lower data-rate)

Agenda

1) Inductive X-talk:	5%
2) Terminology:	5%
3) Methodology:	50%
4) Experimental Results:	30%
5) Conclusion:	10%

1) Inductive X-Talk

<u>Supply Bounce</u>

•The instantaneous current that flows when signals switch induces a voltage across the inductance of the power supply interconnect following:

$$V_{bnc} = L \cdot \left(\frac{di}{dt} \right)$$

•When more than one signal returns current through one supply pin, the expression becomes:

$$V_{bnc} = L \cdot \sum_{i} \left(\frac{di}{dt} \right)$$

NOTE: Reducing the number of signals switching in the same direction at the same time will reduce the supply bounce.

1) Inductive X-Talk

Glitching

- **Mutual Inductive coupling from neighboring signals that are switching cause a voltage to induce on the victim that is static:**

$$V_{glitch} = M_{1k} \cdot \left(\frac{di_k}{dt} \right)$$

- **The net coupling is the summation from all neighboring signals (k1, k2, k3, …) that are switching:**

$$V_{glitch} = \sum_{1}^{k} M_{1k} \cdot \left(\frac{di_k}{dt} \right)$$

NOTE: The mutual inductive coupling can be canceled out when two neighbors of equal *k* switch in opposite directions.

1) Inductive X-Talk

Edge Degradation

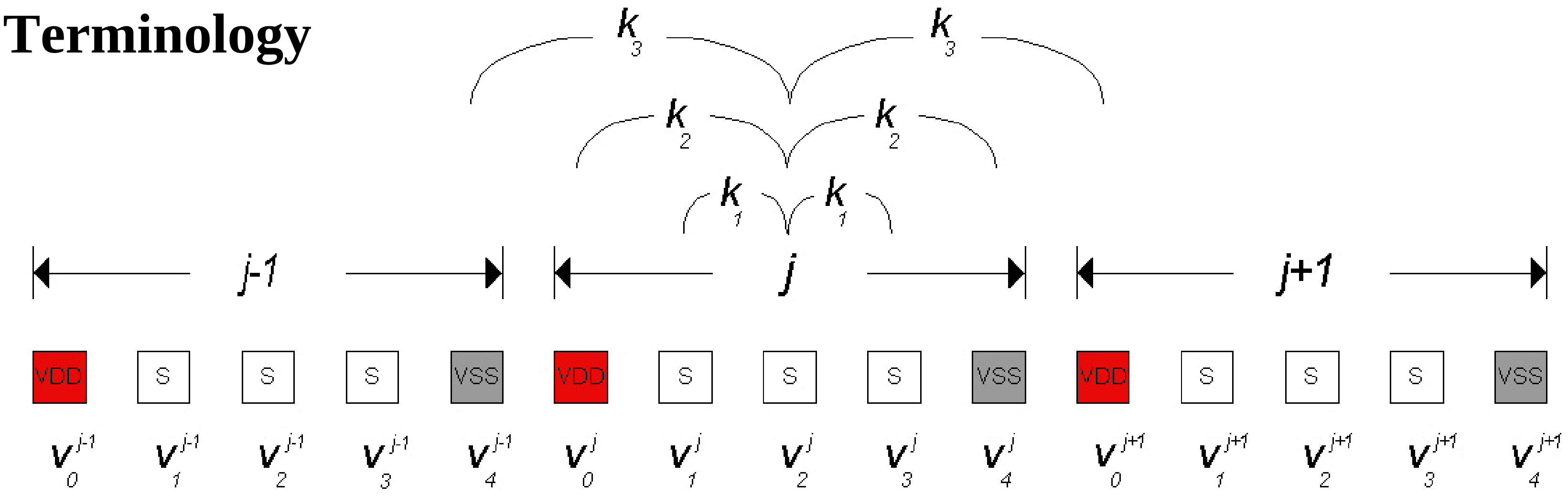
- **Mutual Inductive coupling from neighboring signals that are switching cause a voltage to induce on the victim that is also switching. This follows the same expression as glitch coupling:**

$$V_{glitch} = \sum_{1}^{k} M_{1k} \cdot \left(\frac{di_k}{dt} \right)$$

NOTE: The mutual inductive coupling can be canceled out when two neighbors of equal *k* switch in opposite directions.

NOTE: Mutual Coupling can be encoded so as to *help* the transition resulting in a faster rise-time.

2) Terminology

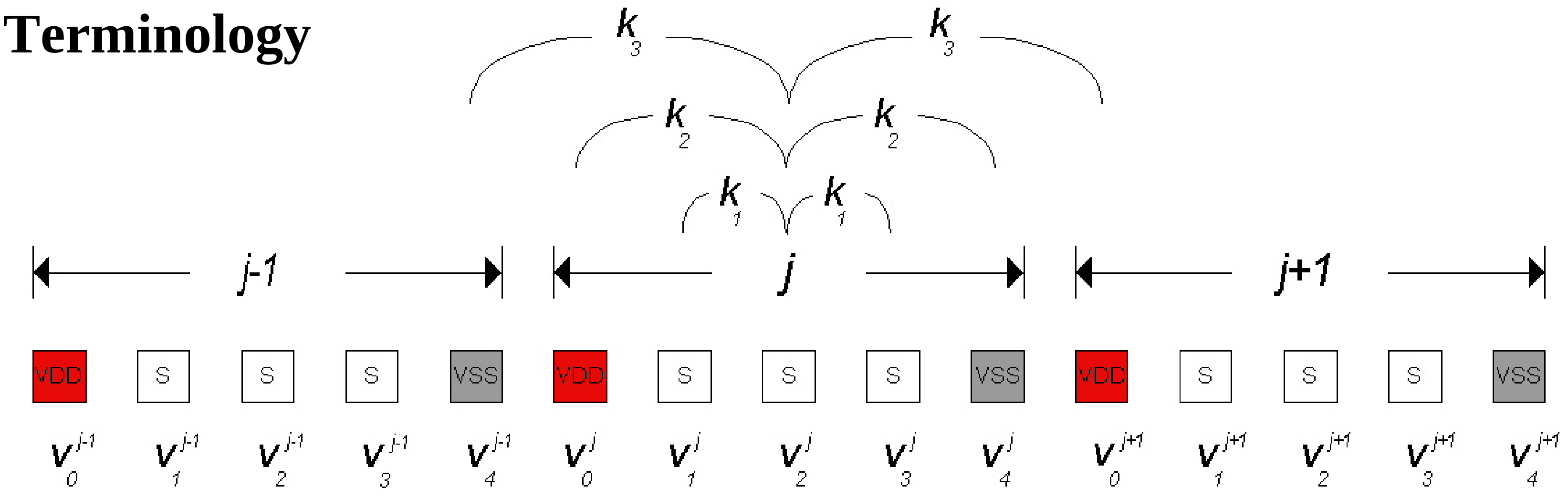


Define the following:

n = **width of the bus**
where each bus consists of n-2 signals
and 1 VDD and 1 VSS.

j = **the segment consisting of an n-bit bus.**
j_i is the segment under consideration.
j_{i-1} is the segment to the immediate left.
j_{i+1} is the segment to the immediate right.
each j segment has the same VDD/VSS placement.

2) Terminology



Define the following:

v_i = **the transition (vector) that the signal is undergoing.**

where

v_i = 1 = rising edge
v_i = -1 = falling edge
v_i = 0 = static edge

2) Terminology

Define the following coding constraints:

Supply Bounce

if v_i is a supply pin, the total bounce on this pin is bounded by P_{bnc}. P_{bnc} is a user defined constant.

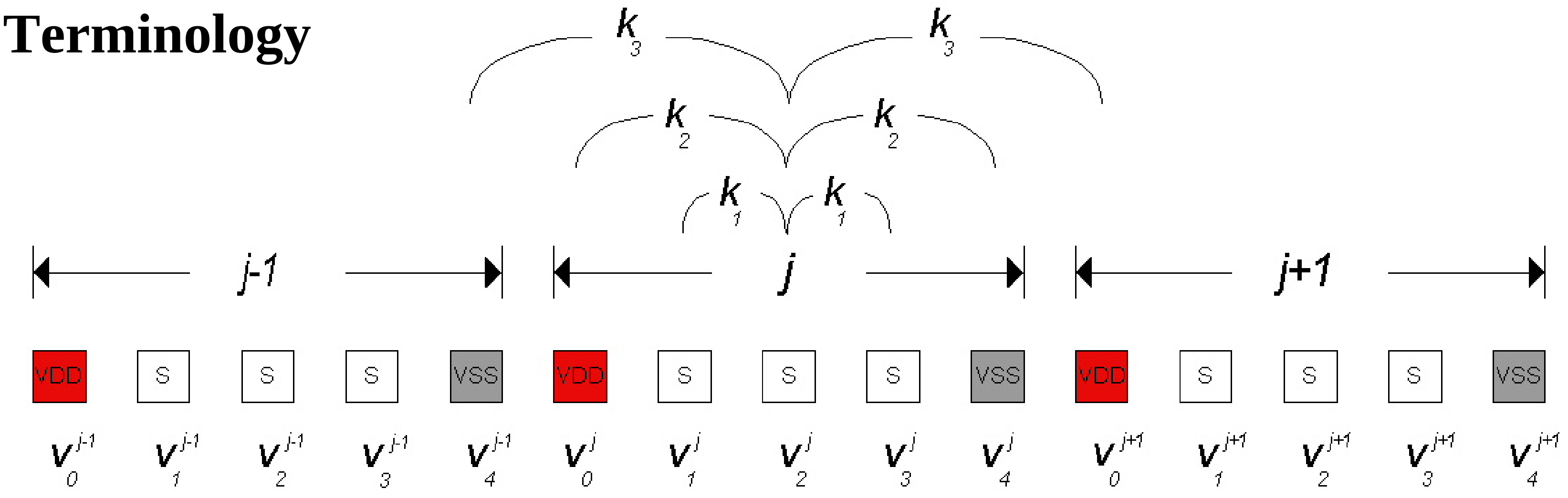
Glitching

if v_i is a signal pin and is static (v_i=0), the total magnitude of the glitch from switching neighbors should be less than P_0. P_0 is a user defined constant.

Edge Degradation

if v_i is a signal pin and is switching (v_i=1/-1), the total magnitude of the coupling from switching neighbors should be greater than P_1. This coupling should not hurt (should aid) the transition. P_1 is a user defined constant.

2) Terminology


k3 k3
k2 k2
k1 k1
j-1
j
j+1
VDD S S S VSS VDD S S S VSS VDD S S S VSS
V0 j-1, V1 j-1, V2 j-1, V3 j-1, V4 j-1
V0 j, V1 j, V2 j, V3 j, V4 j
V0 j+1, V1 j+1, V2 j+1, V3 j+1, V4 j+1

Define the following:

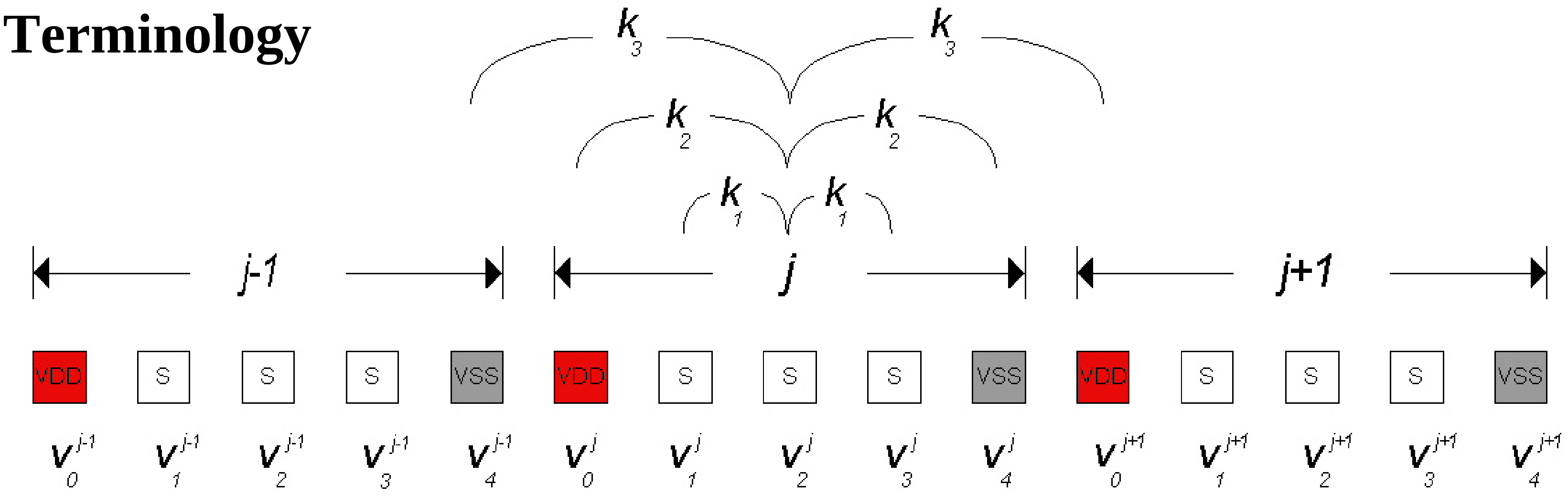
k

k_i = **the mutual inductive coupling coefficient**

$$M_{1k} = k_{1k} \cdot \sqrt{L_1 \cdot L_k}$$

p = **how far away to consider coupling**
(ex., p=3, consider k_1, k_2, and k_3 on each side of the victim)

2) Terminology

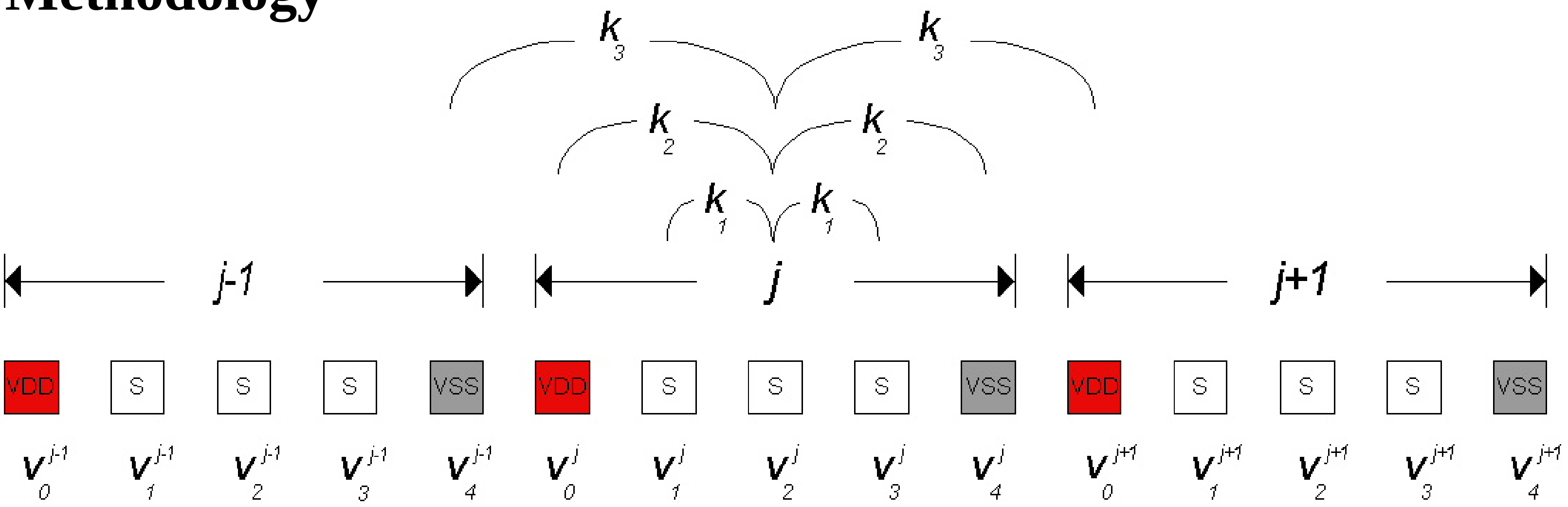


Define the following:

k = the number of j segments in the total bus.

α = Supply / Signal Ratio
(ex., *n*=5 with 1 VDD and 1 VSS, this would have α =5/2)

3) Methodology



- **For each pin v_i^j within segment *j*, we will write a series of constraints that will bound the inductive cross-talk magnitude.**

- **The constraints will differ depending on whether v_i^j is a signal or power pin.**

- **The coupling constraints will consider signals in adjacent segments (*j+1*, *j-1*) depending on *p*.**

3) Methodology – Signal Pin Constraints

Glitching : coupling is bounded by P_0

Example:

$v_2^j = 0$, and $p=3$. This means the three adjacent neighbors on either side of v_2^j need to be considered (v_4^{j-1}, v_0^j, v_1^j, v_3^j, v_4^j, v_0^{j+1}).

Note the *modulo n* arithmetic allows consideration of adjacent segments with one mathematical framework.

$v_2^j = 0$ (static)

$$-P_0 \le k_3 \cdot (v_4^{j-1}) + k_2 \cdot (v_0^j) + k_1 \cdot (v_1^j) + k_1 \cdot (v_3^j) + k_2 \cdot (v_4^j) + k_3 \cdot (v_0^{j+1}) \le P_0$$

Now the constraint equation is evaluated for each possible transition and the transitions that violate the constraint are eliminated.

3) Methodology – Signal Pin Constraints

Edge Degradation : coupling is bounded by P_1 and P_{-1}

Example:

v_2^j =1 or -1, and p=3. This means the three adjacent neighbors on either side of v_2^j need to be considered (v_4^{j-1}, v_0^j, v_1^j, v_3^j, v_4^j, v_0^{j+1}).

v_2^j =1 (rising)

$$k_3 \cdot (v_4^{j-1}) + k_2 \cdot (v_0^j) + k_1 \cdot (v_1^j) + k_1 \cdot (v_3^j) + k_2 \cdot (v_4^j) + k_3 \cdot (v_0^{j+1}) \geq P_1$$

v_2^j =-1 (falling)

$$k_3 \cdot (v_4^{j-1}) + k_2 \cdot (v_0^j) + k_1 \cdot (v_1^j) + k_1 \cdot (v_3^j) + k_2 \cdot (v_4^j) + k_3 \cdot (v_0^{j+1}) \leq P_{-1}$$

Again, the constraint equations are evaluated for each possible transition and the transitions that violate the constraints are eliminated.

3) Methodology – Power Pin Constraints

Supply Bounce : coupling is bounded by P_{bnc}

Example:

v_0^j =VDD or VSS. The total number of switching signals that use v_0^j to return current must be considered. Due to symmetry of the bus definition, signal pins will always return current through two supply pins. i.e., (v_0^{j-1} and v_0^j) or (v_0^j and v_0^{j+1}). This results in the self inductance of the return path being divided by 2.

v_0^j =VDD

$$(L/2)\cdot(\#\text{ of } v_i^j \text{ pins that are } 1) \leq P_{bnc}$$

v_4^j =VSS

$$(L/2)\cdot(\#\text{ of } v_i^j \text{ pins that are } -1) \leq P_{bnc}$$

3) Methodology – Constructing Legal Vectors Sequences

- **For each bit in the j segment bus, constraints are written.**

- **If the pin is a signal, 3 constraint equations are written;**
  - **$v_0^j = 0$, the bit is static and a *glitching constraint* is written**
  - **$v_0^j = 1$, the bit is rising and an *edge degradation* constraint is written.**
  - **$v_0^j = -1$, the bit is falling and an *edge degradation* constraint is written.**

- **If the pin is V_{DD}, 1 constraint equation is written to avoid *supply bounce*.**

- **If the pin is V_{SS}, 1 constraint equation is written to avoid *ground bounce*.**

3) Methodology – Constructing Legal Vectors Sequences

- This results in the total number of constraint equations being written is:

$$(3 \cdot n - 4)$$

- Each equation must be evaluated for each possible transition to verify if the transition meets the constraints. The total number of transitions that are evaluated depends on *n* and *p:*

$$3^{(n+2 \cdot p - 6)}$$

3) Methodology – Constructing the CODEC

- **The remaining legal transitions are used to create the CODEC.**

- **Each of the 2^n States will potentially have a set of legal *outgoing* transitions that it may take to reach another State.**

- **Each of the 2^n States will potentially have a set of legal *incoming* transitions that other States use to reach it.**

- **The total number of remaining legal transitions will depend on how aggressive the user-defined constants are chosen (P_0, P_1, P_{-1}, P_{bnc})**

3) Methodology – Constructing the CODEC

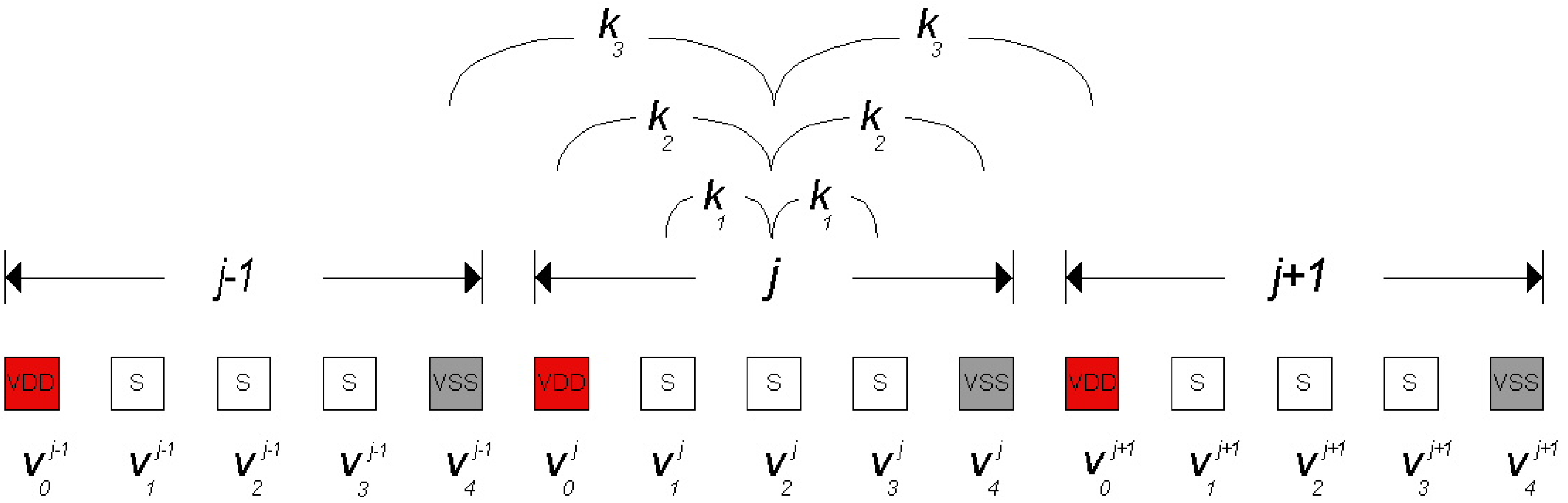
- **For each State, the legal transitions are used to create a subset of possible states that it can reach and also return from.**

- **The States that form the largest possible effective bus size with the largest number of transitions within the group form the final encoded bus.**

- **The circuitry to map the original possible states into the new subset of states is synthesized to implement the encoder/decoder.**

4) Experimental Results – 3 Signal Pins

Example Bus: $n=5, k=3, \alpha=5/2, p=2$



	P_0, P_1, P_{-1}, P_{bnc}
Aggressive Encoding	**5% of V_{DD}**
Non-Aggressive Encoding	**10% of V_{DD}**

4) Experimental Results – Possible Transitions

Possible Transitions = $3^{(n+2p-6)}$ = 27

Transition	v_1	v_2	v_3
1	0	0	0
2	0	0	1
3	0	0	-1
4	0	1	0
5	0	1	1
6	0	1	-1
7	0	-1	0
8	0	-1	1
9	0	-1	-1
10	1	0	0
11	1	0	1
12	1	0	-1
13	1	1	0
14	1	1	1
15	1	1	-1
16	1	-1	0
17	1	-1	1
18	1	-1	-1
19	-1	0	0
20	-1	0	1
21	-1	0	-1
22	-1	1	0
23	-1	1	1
24	-1	1	-1
25	-1	-1	0
26	-1	-1	1
27	-1	-1	-1

4) Experimental Results – Constraint Equations

# of Constraints = (3n – 4) = 11

1) $v_0^j = V_{DD} \quad \rightarrow \quad (L/2)\cdot(\#\text{ of } v_i^j \text{ pins that are 1}) \leq P_{bnc}$

2) $v_1^j = 1 \quad \rightarrow \quad k_1\cdot(v_2^j) + k_2\cdot(v_3^j) \geq P_1$

3) $v_1^j = -1 \quad \rightarrow \quad k_1\cdot(v_2^j) + k_2\cdot(v_3^j) \leq P_{-1}$

4) $v_1^j = 0 \quad \rightarrow \quad -P_0 \leq k_1\cdot(v_2^j) + k_2\cdot(v_3^j) \leq P_0$

5) $v_2^j = 1 \quad \rightarrow \quad k_1\cdot(v_1^j) + k_1\cdot(v_3^j) \geq P_1$

6) $v_2^j = -1 \quad \rightarrow \quad k_1\cdot(v_1^j) + k_1\cdot(v_3^j) \leq P_{-1}$

7) $v_2^j = 0 \quad \rightarrow \quad -P_0 \leq k_1\cdot(v_1^j) + k_1\cdot(v_3^j) \leq P_0$

8) $v_3^j = 1 \quad \rightarrow \quad k_2\cdot(v_1^j) + k_1\cdot(v_2^j) \geq P_1$

9) $v_3^j = -1 \quad \rightarrow \quad k_2\cdot(v_1^j) + k_1\cdot(v_2^j) \leq P_{-1}$

10) $v_3^j = 0 \quad \rightarrow \quad -P_0 \leq k_2\cdot(v_1^j) + k_1\cdot(v_2^j) \leq P_0$

11) $v_4^j = V_{SS} \quad \rightarrow \quad (L/2)\cdot(\#\text{ of } v_i^j \text{ pins that are -1}) \leq P_{bnc}$

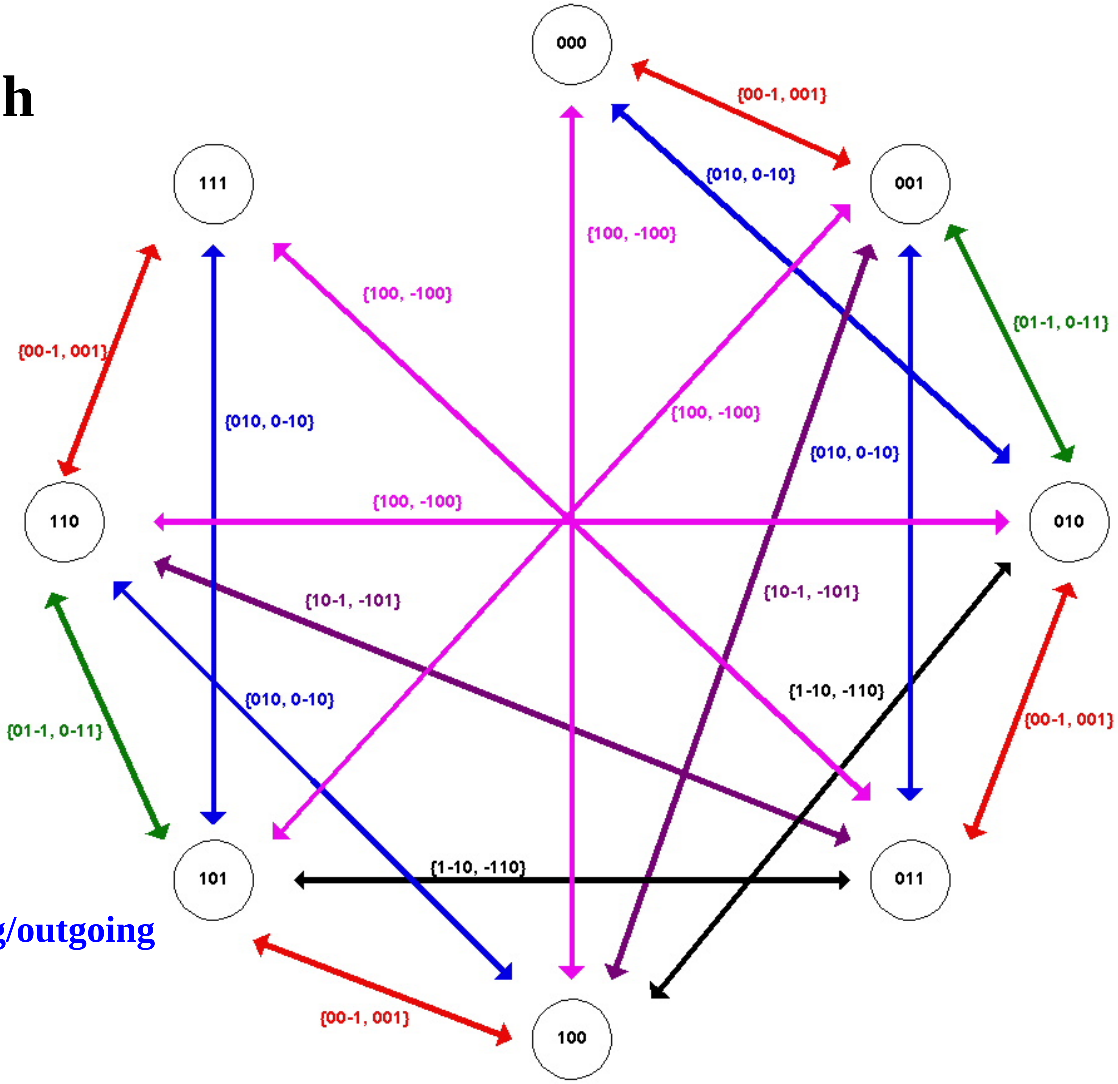
4) Experimental Results – CASE 1: Fixed di/dt

Transitions Eliminated due to Rule Violations

	Rule(s) Violated	
Transition	**Aggressive**	**Non Aggressive**
011	violates 1,4	-
0-1-1	violates 4,11	-
101	violates 1,7	-
110	violates 1,10	-
111	violates 1,2,5,8	violates 11
11-1	violates 1	-
1-11	violates 1	-
1-1-1	violates 11	-
-10-1	violates 7,11	-
-111	violates 1	-
-11-1	violates 11	-
-1-10	violates 10,11	-
-1-11	violates 11	-
-1-1-1	violates 3,6,9,11	violates 1

4) Experimental Results – CASE 1: Fixed di/dt

Directed Acyclic Graph

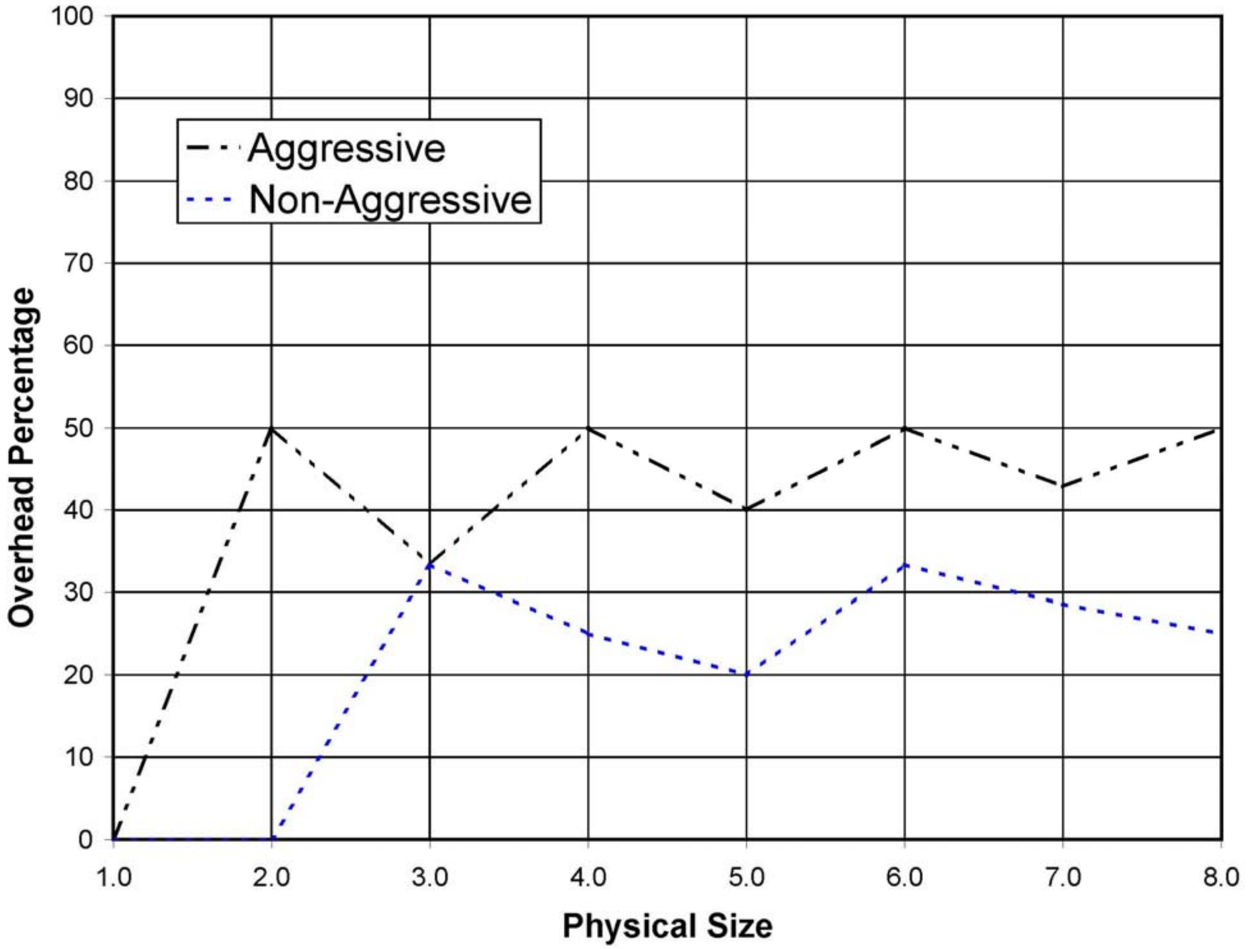


- remaining vectors are shown for each state.

- states without sufficient incoming/outgoing vectors are eliminated

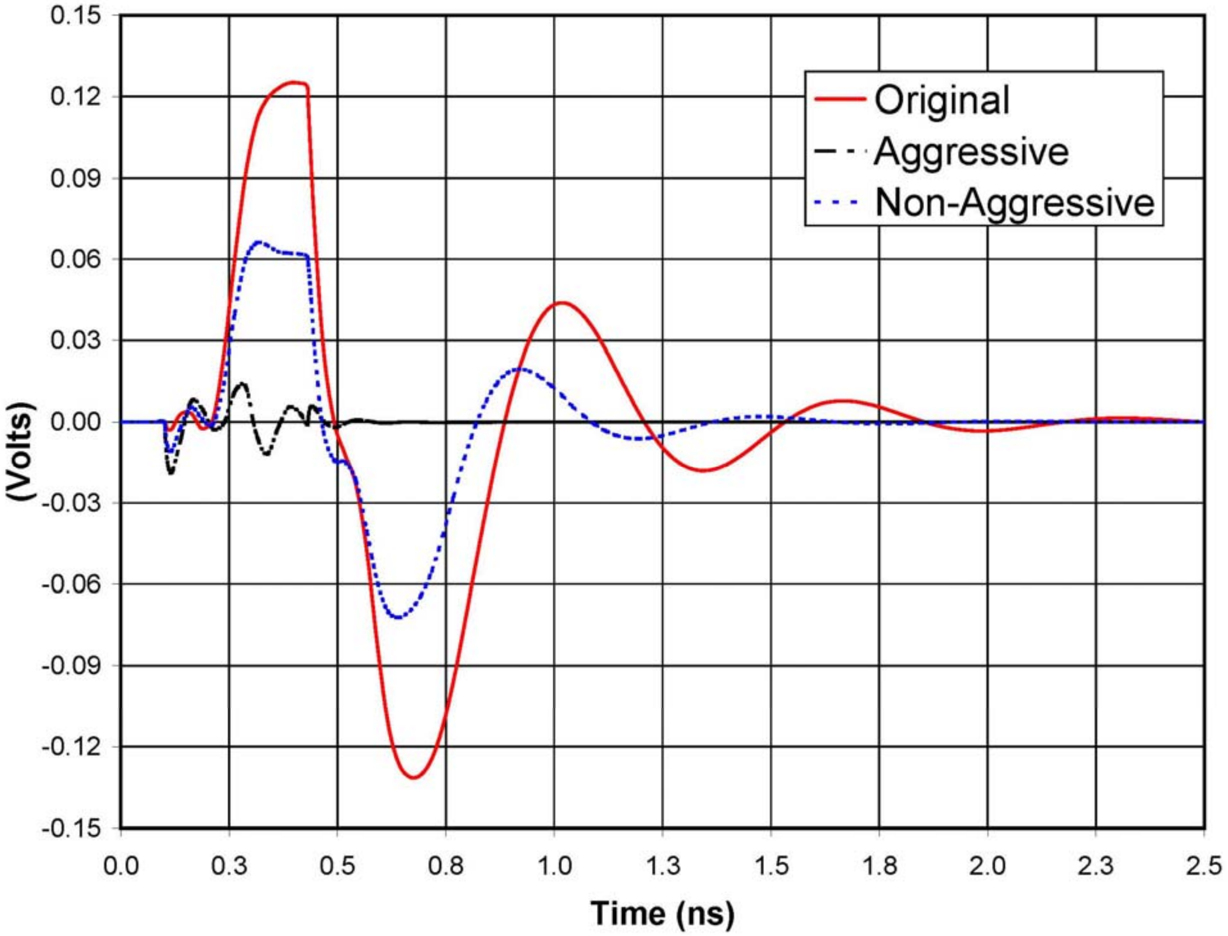
4) Experimental Results – CASE 1: Fixed di/dt

DAG Efficiency



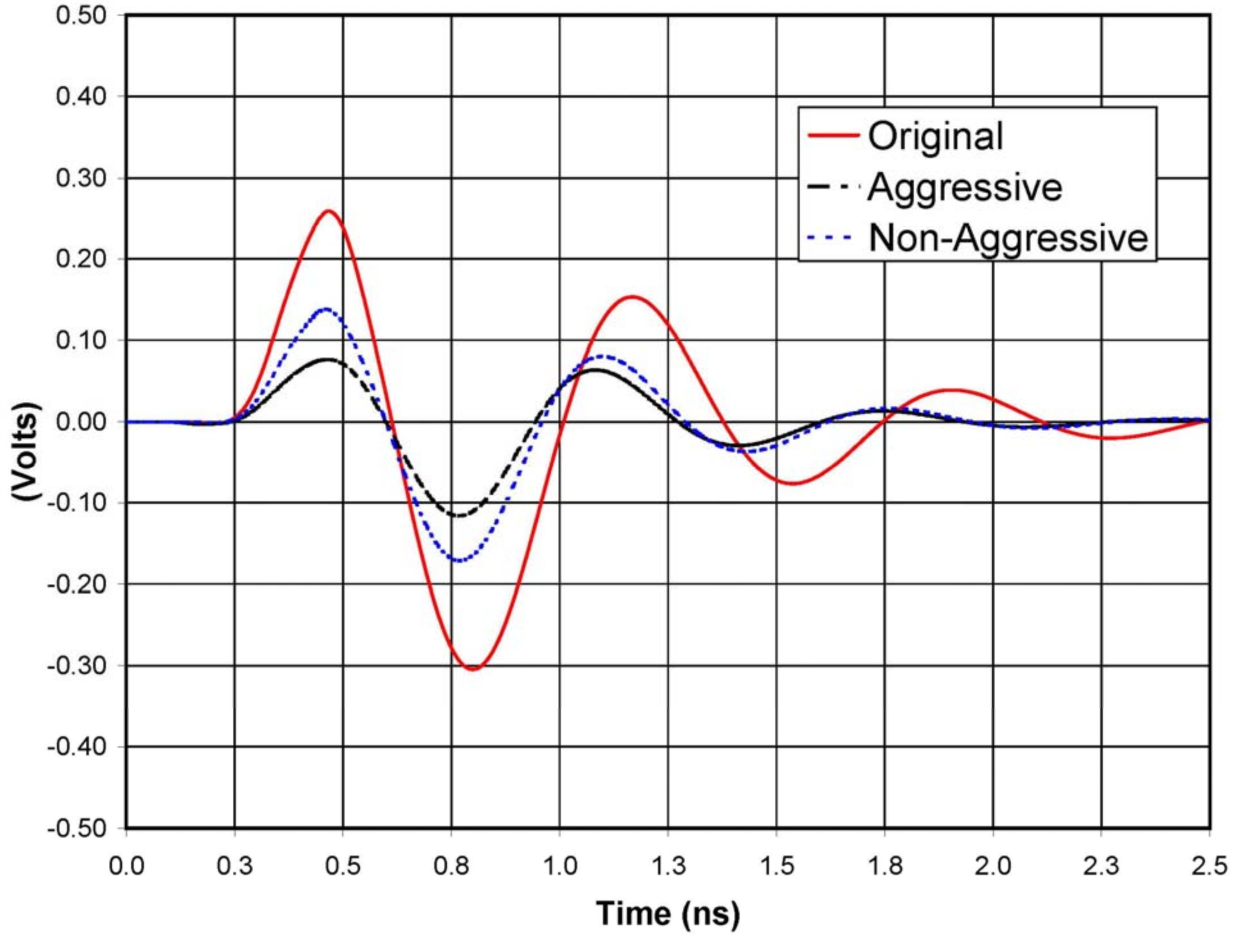
4) Experimental Results – CASE 1: Fixed di/dt

Ground Bounce Simulation



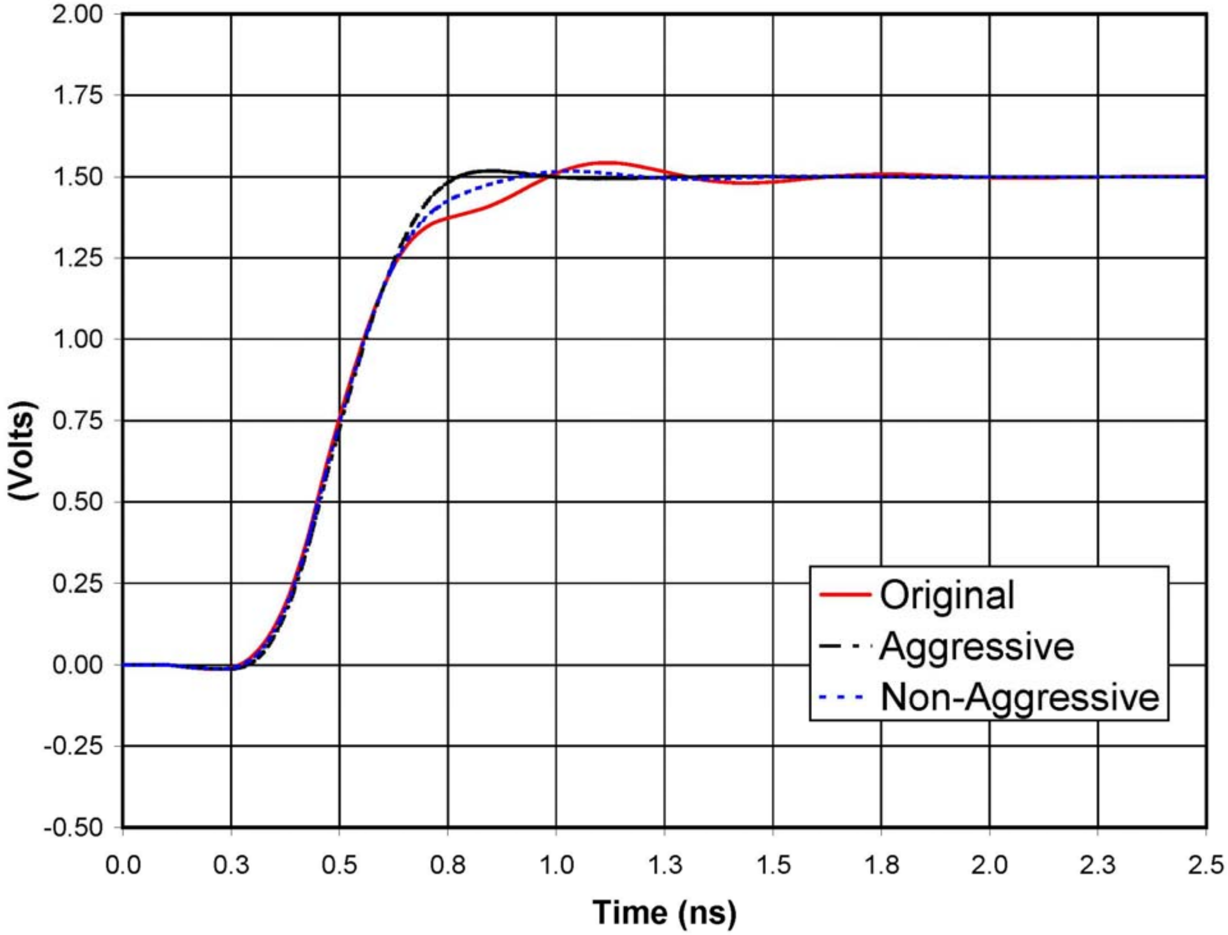
4) Experimental Results – CASE 1: Fixed di/dt

Glitch Simulation



4) Experimental Results – CASE 1: Fixed di/dt

Edge Degradation Simulation



4) Experimental Results – CASE 2: Variable di/dt

- di/dt was swept for both the non-encoded and encoded configuration.
- the maximum di/dt was recorded that resulted in a failure.
- a failure was defined as 5% of VDD
- the maximum di/dt was converted to data rate and throughput.

	Non-Encoded	Encoded
Maximum di/dt:	13.3 MA/s	37 MA/s
Maximum data-rate per pin:	222 Mb/s	617 Mb/s
Effective bus width:	3	2
Total Throughput:	666 Mb/s	1234 Mb/s
Improvement	-	85%
Encoder Overhead	-	33%

5) Conclusion

- **Using a single mathematical framework, inductive X-talk constraints can be written that consider supply bounce, glitching, and edge degradation.**

- **This technique can be used to encode off-chip data transmission to reduce inductive X-talk to acceptable levels.**

- **It was demonstrated that even after reducing the effective bus size, the improvement in per pin data-rate resulted in an *increase* in throughput compared to a non-encoded bus.**

Future Work

1) Power Reduction

- A large percentage of the power (25%-50%) is consumed in the output stages.
- this technique can be used to limit the amount of simultaneous switching to reduce power.

2) Programmable CODECs

- This CODEC could be implemented as a *programmable* coding circuit prior to the tapered output drivers.
- This would allow one generic circuit to reside on the die and compensates for any style of package that is used.