PENNSTATE

3DLAT: TSV-based 3D ICs crosstalk minimization utilizing Less Adjacent Transition Code

Qiaosha Zou

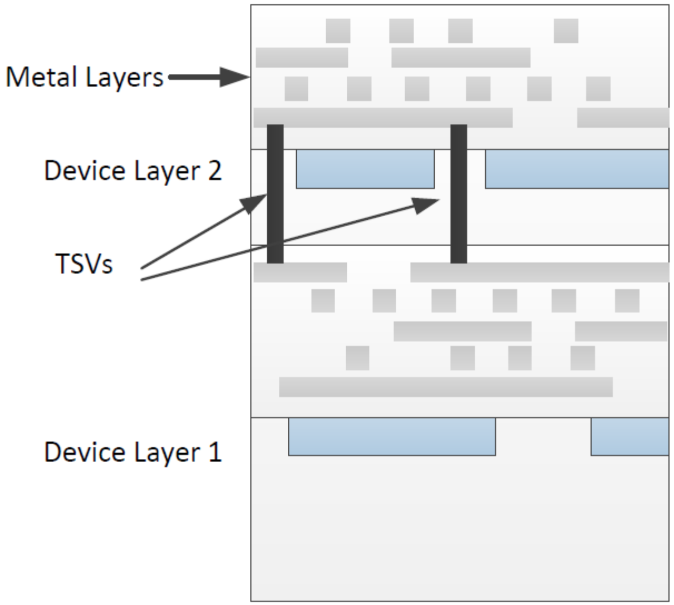
Dimin Niu

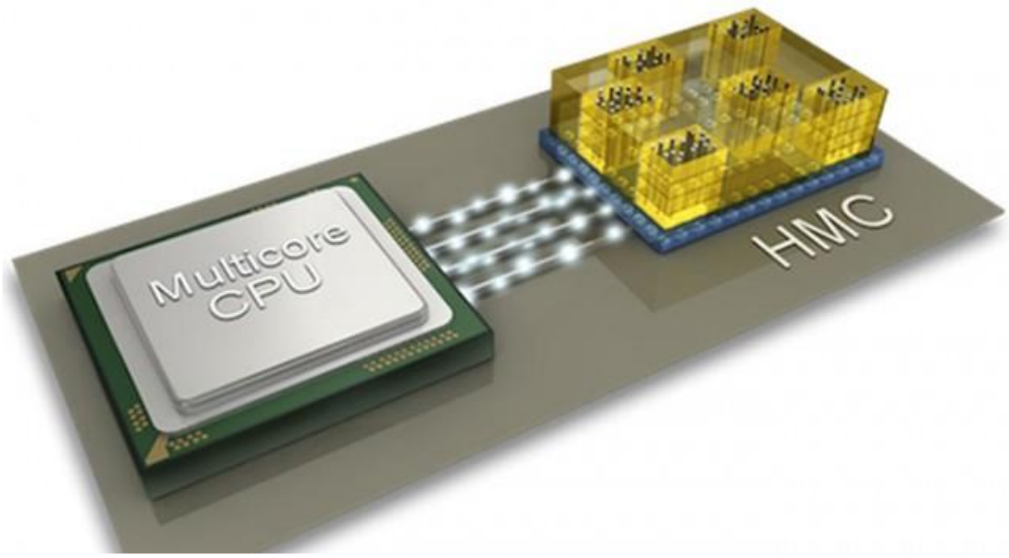
Yan Cao

Yuan Xie

The Pennsylvania State University

3D integration is a promising solution for interconnect crisis.





Source: Micron Hybrid Memory Cube

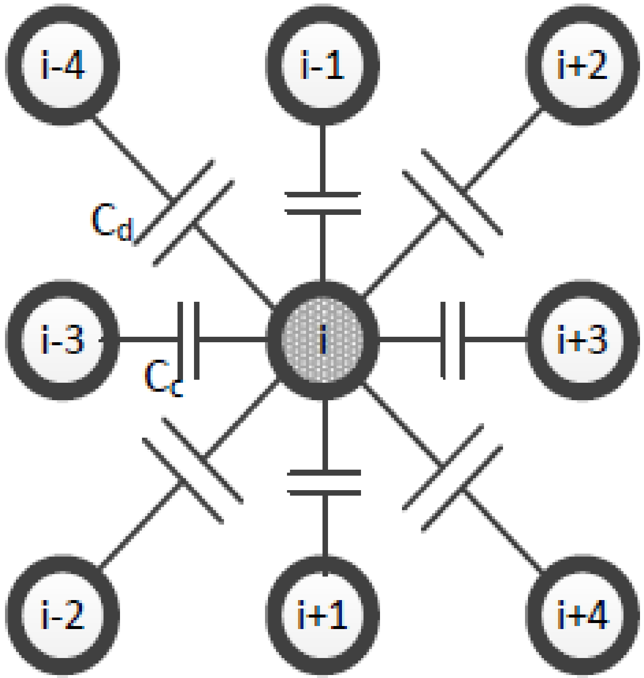
- Capacitance crosstalk in TSVs
  - Relatively large size of TSVs
  - Coupled deep inside the substrate

Outline

- Preliminaries
  - Backgrounds on crosstalk
  - 2DNAT (no adjacent transition) Code: Transition signaling and Limited Weighted Code
- 3D LAT Coding Mechanism
- Performance and power evaluation

Crosstalk in TSV arrays

- Analysis Complexity:
  - Increased number of neighbors
  - Each victim has 8 aggressors.
- Transition direction matters
  - $\Delta V_i = V_i(t^+) - V_i(t^-)$
    (0 to 1, or 1 to 0)
  - $\delta_{i,k} = abs(\frac{\Delta V_i - \Delta V_k}{V_{dd}})$
    (value of 0, 1, or 2)



Crosstalk in TSV arrays

- Effective crosstalk capacitance
  - $C_{eff,i} = C_L(1 + \lambda_1 \sum \delta_n + \lambda_2 \sum \delta_d)$
  - λ represents the capacitance ratio between coupling capacitance and self capacitance.
- Crosstalk classification
  - $\sum \delta_n$ can be any integer in [0, 8]
  - 0C to 8C without considering diagonal TSVs
  - Add 9C and 10C for four diagonal TSVs

Previous work on 3D crosstalk

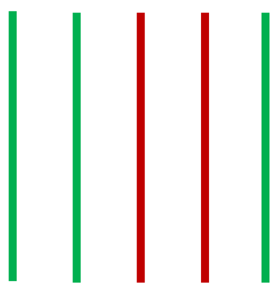
- 3D k-CAC: Crosstalk Avoidance Code (Kumar et al., DATE 2013)
  - Eliminate the transmission pattern that causes (k+1)C crosstalk.
  - Problems: large overhead and complexity
- ShieldUS (Chang et al., ASPDAC2013):
  - Use relatively stable data signals as shields
  - Problems: data mapping & unstable performance

How does 2D design handle crosstalk problem?

2D No Adjacent Transition Code

- Combine the transition signaling and the limited weighted code.
- Transition Signaling
  - Input bit is 1 => transition occurs
  - Assume signal is 10010, wire voltage is LLHHL
  - XOR previous and current wire value for input data

Input Signal 1 0 0 1 0



Limited Weighted Code & 2D NAT

- Limited Weighted Code
  - Weight: number of 1s in the data
  - Encode to limit the weight of each data input

- 2D NAT
  - No adjacent 1s are allowed in codeword

Avoidance Pattern: H L H

⇩

L H L

3D NAT is infeasible

- Imagine apply 2D NAT into 3D designs…

(assume weak coupling between diagonal TSVs)

X	$\bar{b}$	X
$\bar{b}$	b	$\bar{b}$
X	$\bar{b}$	X

b can be only 0 or 1

Codeword Cardinality (number of qualified codeword) is only 2^5 compared to 2^9.

Outline

- Preliminaries
- 3D LAT Coding Mechanism
  - LAT code design
  - LAT optimization
  - Heuristic CODEC design
- Performance and power evaluation

3D limited weighted LAT code

- Limit the number of 1s in adjacent nodes
  - Adjacent nodes include eight neighbors in the array
  - Target at TSV arrays with 3 rows.
  - Use ω for maximum allowed weight for each 3*3 TSVs
  - Limit the crosstalk within $(\omega - 1) * 2C$
    - Worst case consideration.
    - At most $\omega - 1$ neighbors are with the opposite transition direction.

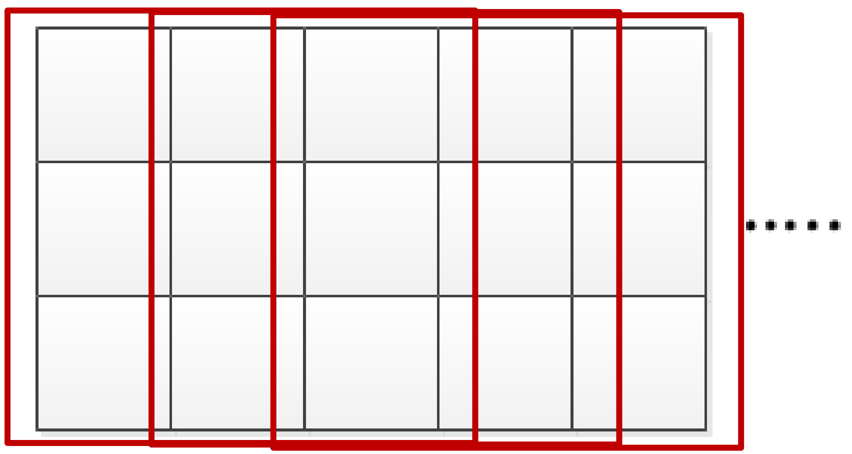
Code Cardinality Calculation

- The codeword overhead is determined by the code cardinality.
- The number of codeword should not be smaller than the number of data input ($T(\omega, N) \leq 2^d$)

$\omega_1 = 1$

$\omega_2 = 0$

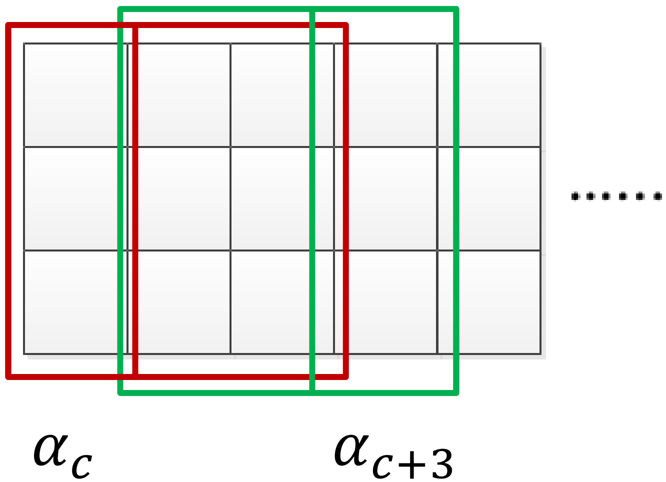
$\omega_3 = 2$



Impossible to calculate code cardinality with variable weights for each 3*3 TSV array.

Lower bound of the code cardinality is used instead. Each TSV subarray has exact the same weight.

Codeword Cardinality Induction



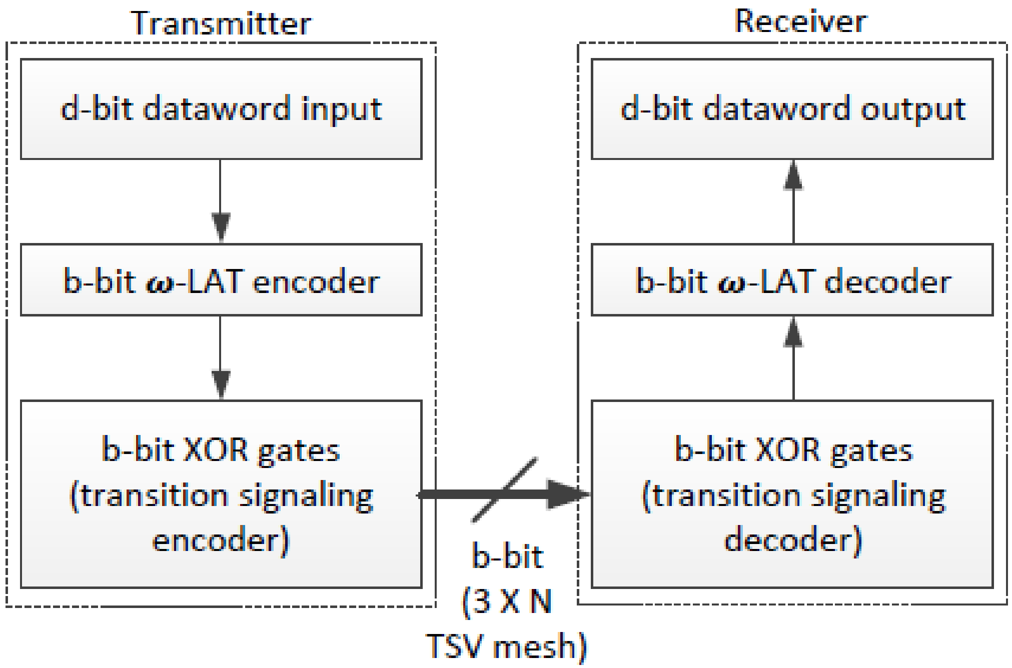
Every other three column has the same weight.

$$T(\beta, N) = \sum_{\alpha_1+\alpha_2+\alpha_3} \left[\binom{3}{\alpha_1} \binom{3}{\alpha_2} \binom{3}{\alpha_3} \right]^{N/3}$$

When value of N is small, enumation is used to get the code cardinality.
For large N, inductive method is used to calculate $T(\beta, N)$, until the minimum required N is found.

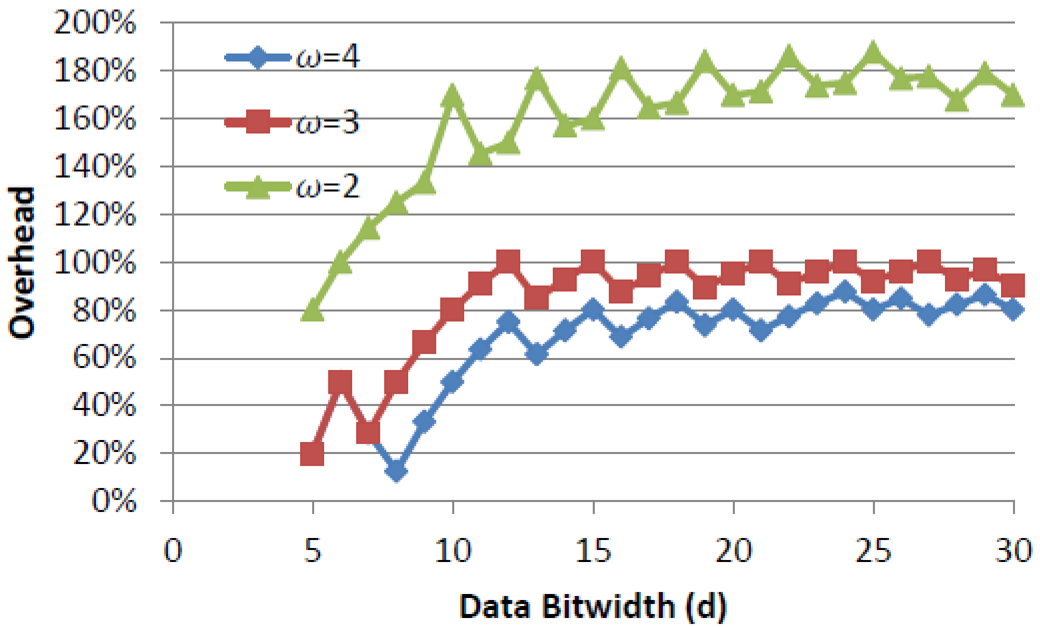
ω-LAT transmission framework

- Two level of encoder
  - LAT encoder
  - Transition signaling encoder



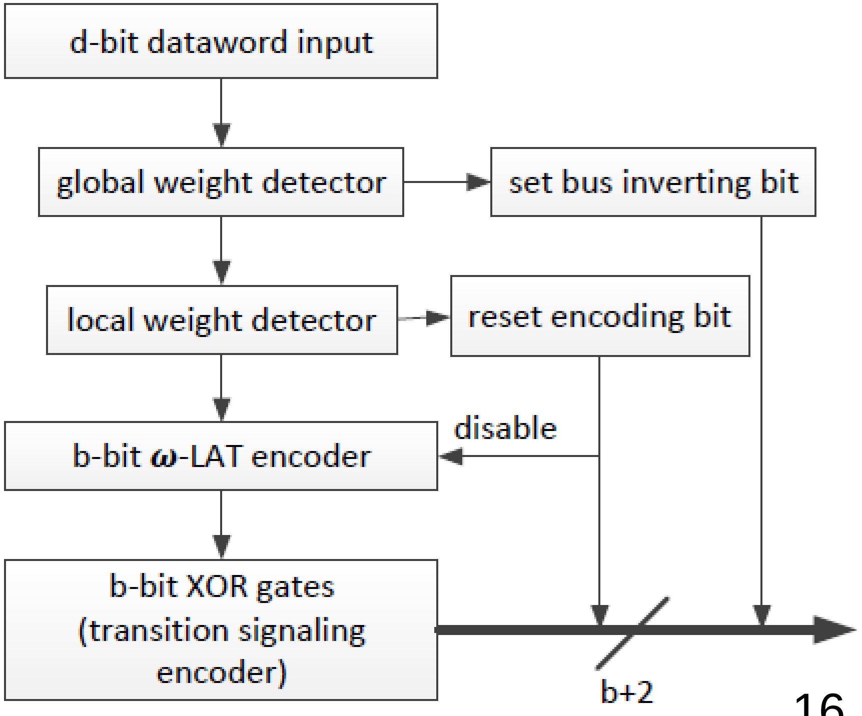
ω-LAT coding overhead

- ω is reduced, overhead is increased
- The overhead is the upper bound
- ω=2 has large overhead, but significantly smaller than 3D CAC (335% overhead)



LAT Code Optimization

- Only encode the data input that doesn't qualified.
  - For example, 00100 doesn't need to be encoded.
- Techniques:
  - Bus Inverting
  - Weight Detecting
- Limitations:
  - Timing overhead
  - Detector area overhead



Comparison of baseline and optimized scheme

Data Bitwidth	Optimized		original		reduced ratio	overhead reduction
	column	overhead (%)	column	overhead (%)	$(\frac{T_\omega(d/3)}{2^d})$	(%)
5	1	-40	2	20	25	60
10	3	-10	5	50	25	60
15	8	60	9	80	4.04	20
20	11	65	12	80	0.38	15
25	15	80	15	80	0.02	0

- With increased data bitwidth, the overhead reduction becomes marginal.
- The number of weight detectors increased with longer input.

Heuristic CODEC design

- No universal CODEC design due to the variation on ω.
- Option 1: Look Up Table based CODEC design.
- Option 2: Analyze the 3D LAT coding scheme.
- Two level of comparators are used in encoder
  - First level: TSV subarray weight
  - Second level: combination of α_1 to α_3
- Heuristic CODE design on case study
  - ω=4, data input 16 bits
  - Data input value 1024

CODEC design case study

- Codeword bitwidth is 27 and has 9 columns
- Decide ω based on the codeword cardinality.

ω	0	1	2	3	4
Cardinality	1	81	2268	24060	61398
value	1	82	2350	26410	87808

$$82 < 1024 < 2350$$

⇩

Subarray weight is 2

$$\alpha_1 + \alpha_2 + \alpha_3 = 2$$

CODEC design case study

- Calculate the codeword cardinality and determine the α combination
- 6 combinations: (0,1,1) (1,0,1) (1,1,0) (0,0,2) (0,2,0) (2,0,0)
- Determine code cardinality for each combination
- Find the combination according to the cardianlity
  - We choose to use (1,0,1) for value 1024.

CODEC design case study

- Determine the row position of the 1ss.
  - $k_0 * 3^0 + k_1 * 3^1 + k_2 * 3^2 + k_3 * 3^3 + k_4 * 3^4 + k_5 * 3^5$
- For 1024, the final codeword is:
  - $(k_0, k_1, k_2, k_3, k_4, k_5) = (0, 2, 2, 2, 1, 0)$

Codeword for 1024:

1	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0	0
0	0	1	1	0	1	0	0	0

Outline

- Preliminaries
- 3D LAT Coding Mechanism
- Performance and power evaluation
  - Analytical power evaluation
  - Performance simulation

Power Evaluation

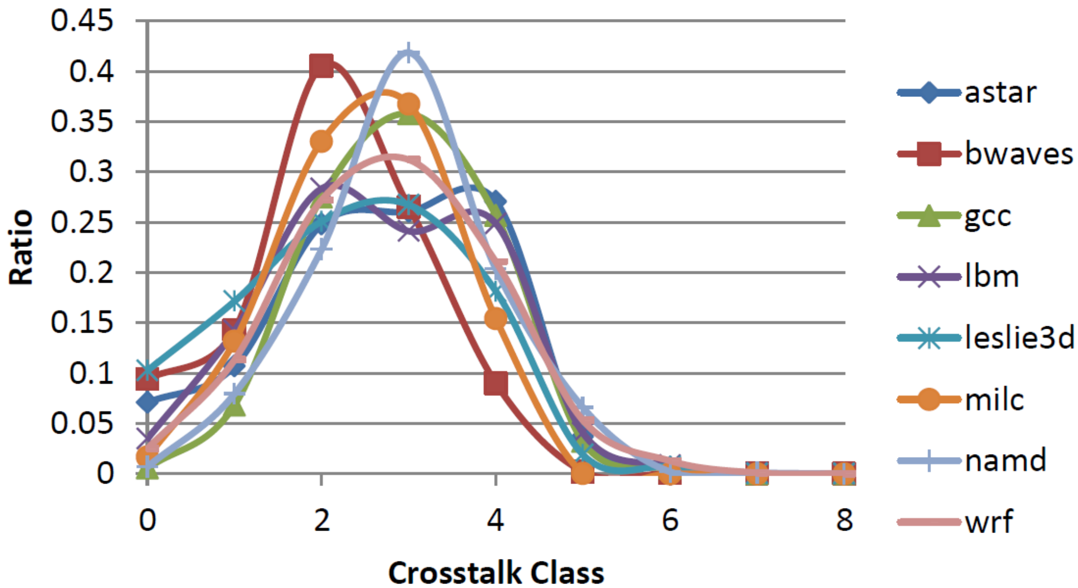
- Analytical Power Model
  - $P^s = \frac{1}{2} C_L {V_{DD}}^2 * \Pr(trans)$
  - $P^c = C_c {V_{DD}}^2 * \Pr(V_k(t^+) \neq V_{k+1}(t^+)) * E_t$

code	$Pr(trans)$	$Pr(V_k(t^+) \neq V_{k+1}(t^+))$	$E_t c(k, k+1)$
uncoded	0.5	0.5	1
ShieldUS	0.5	0.5	≤ 1
6C CAC	0.5	0.367	1
4-LAT	0.4079	0.5	0.8159

Assume λ_1 is 5.54, power consumption for uncoded cases is $8.56 C_L {V_{DD}}^2$, 4-LAT is $6.98 C_L {V_{DD}}^2$.

Benchmark Analysis

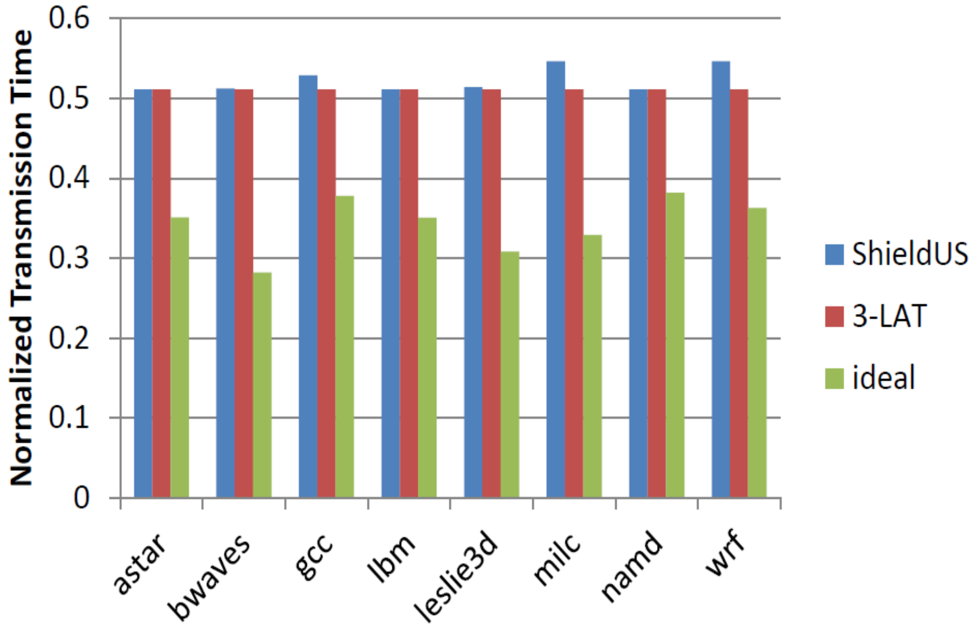
- Extract SPEC 2006 Benchmark memory trace and perform crosstalk class analysis
- Performance evaluation comparison with ShieldUS, 3-LAT, and ideal case.



- Most data transmission are within 5C crosstalk.

Performance Evaluation

- Ideal case: transmission time is flexible and determined by the crosstalk class.



- Ideal case always has the optimal performance.
- ShieldUS cannot guarantee the transmission time
- With determined value of ω, the proposed scheme can have stable performance.

Conclusion

- Due to the relatively large size and deep substrate coupling, 3D capacitive crosstalk minimization should be considered.
- ω-LAT (less adjacent transition) coding scheme is proposed to minimize crosstalk.
- The overhead is affordable with aggressive crosstalk minimization.
- Power consumption of each TSV is reduced and transmission delay can be guaranteed.

Thank you!

Q & A