An Interconnect-Centric Approach to Cyclic Shifter Design

Haikun Zhu, Yi Zhu
C.-K. Cheng
Harvey Mudd College.

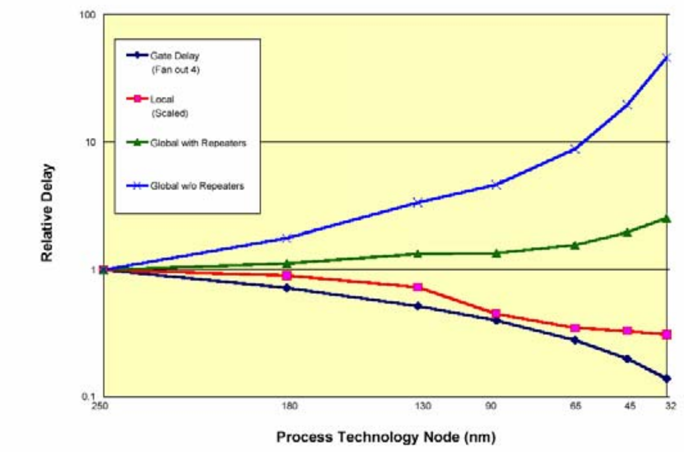
David M. Harris
Harvey Mudd College.

Outline

- Motivation
- Previous Work
- Approaches
  - Fanout-Splitting
  - Cell order optimization by ILP
- Conclusions

Motivation

- Interconnect dominates gate in present process technology
  - Delay, power, reliability, process variation, etc.



Source: ITRS roadmap 2005

- Conventional datapath design focuses on logic depth minimization

Technology Trends

- Device (ITRS roadmap 2005, Table 40a)

Gate length (nm)	90	65	% decreasing
Vdd (V)	1.1	1.1	-
Vth (V)	0.195	0.165	-
NMOS gate Cap (fF/μm)	0.573	0.469	18.2%
NMOS intrinsic delay (ps)	0.870	0.640	26.4%

- Updated Berkeley Predictive Interconnect Model

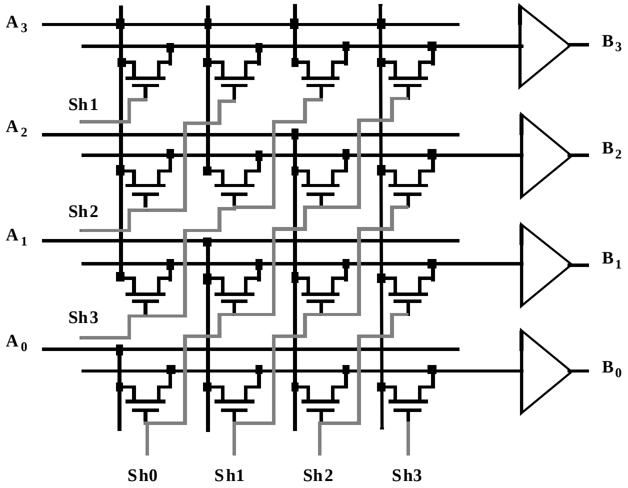
Gate length (nm)	90	65	% decreasing
Inter-layer dielectric constant	2.8	2.6	-
Capacitance of local interconnect (fF/um)	0.186	0.173	7.0%

Shifter Taxonomy

- Functionality
  - Logical Shift: MSBs stuffed with 0’s
  - Arithmetic Shift: Extend original MSB
  - Cyclic Shift (rotation)
  - Bidirectional Shift
- Circuit Topology
  - Barrel Shifter
  - Logarithmic Shifter

Barrel Shifter

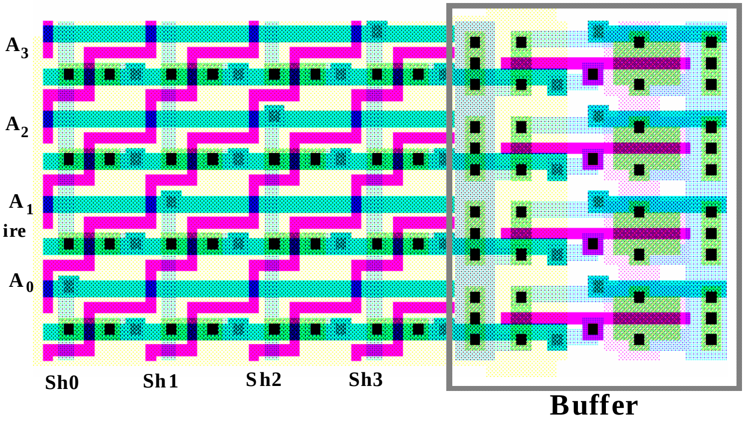
Schematic



— : Data Wire

— : Control Wire

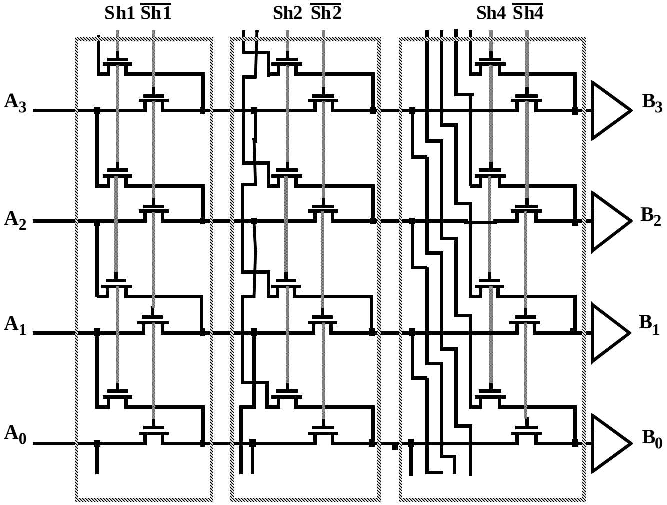
layout



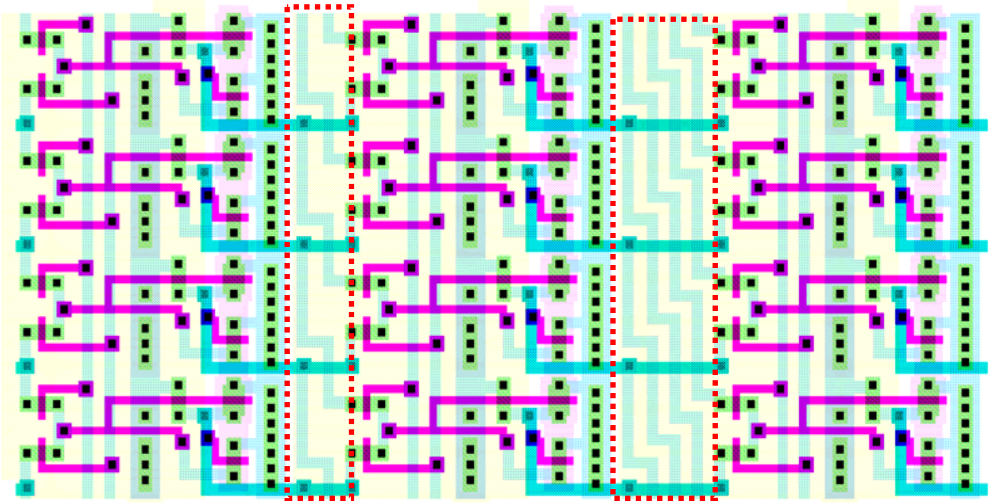
- Pros
  - Every data signal pass only one transmission gate
- Cons
  - Input capacitance is $O(N)$
  - # transistors = $O(N^2)$
  - Requires additional decoder for control signals

Logarithmic Shifter

Schematic


Sh1 $\overline{\text{Sh1}}$
Sh2 $\overline{\text{Sh2}}$
Sh4 $\overline{\text{Sh4}}$
A_3
A_2
A_1
A_0
B_3
B_2
B_1
B_0

layout



- Pros
  - # transistors = $O(N \log_2 N)$
- Cons
  - Long inter-stage wires, especially for cyclic shifters

Target of Optimization

Cyclic Shifter -- Applications

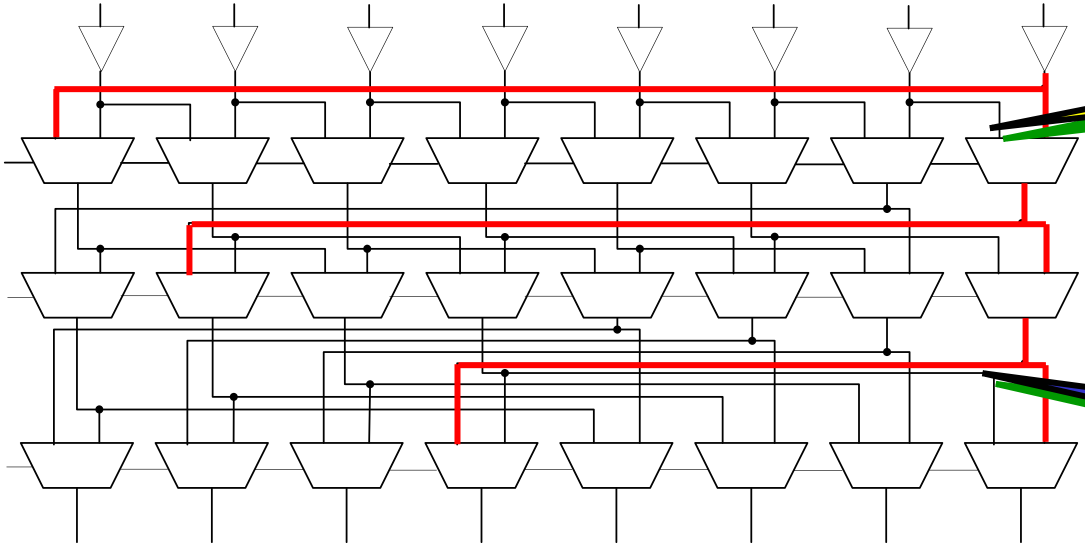
- Finite Field Arithmetic
  - In normal basis, squaring is done by cyclic shifting.
- Encryption
  - ShiftRows operation in Rijndael algorithm.
- DCT processing unit
  - Address generator
- Bidirectional shifting
  - Can be implemented as a cyclic shifter with additional masking logic
- CORDIC algorithm
- etc ...

Previous Work

- Bit interleave
- Two dimensional folding strategy
- Gate duplicating
- Ternary shifting
- Comparison between barrel shifter and log shifter

Cyclic Shifter – Traditional Design

- MUX-based



Timing

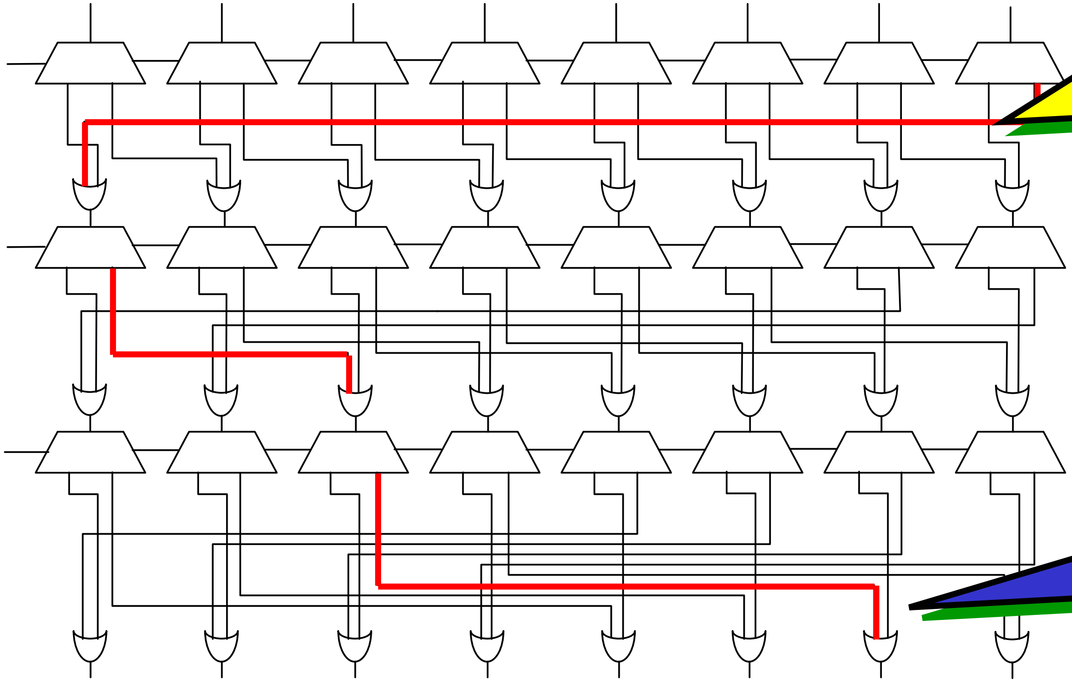
- Shifting & non-shifting paths are intertwined together.
- Wire load on the critical path is $O(N \log_2 N)$

Power

- When configured to pass through, the non-shifting paths have to be switched as well.

Fanout Splitting Shifter

- Use DEMUXes instead of MUXes



Timing

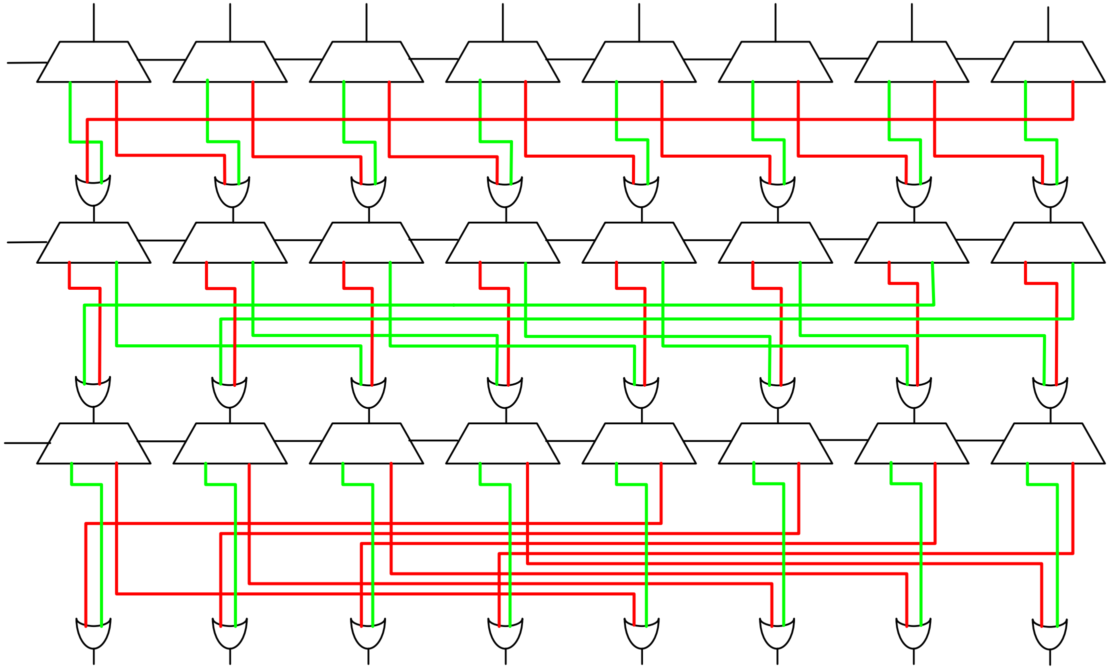
- Shifting & non-shifting paths are now decoupled.
- Wire load on the longest path is $O(N)$

Power

- When shifting, non-shifting paths are at rest.

Example

Right rotate 5 bits



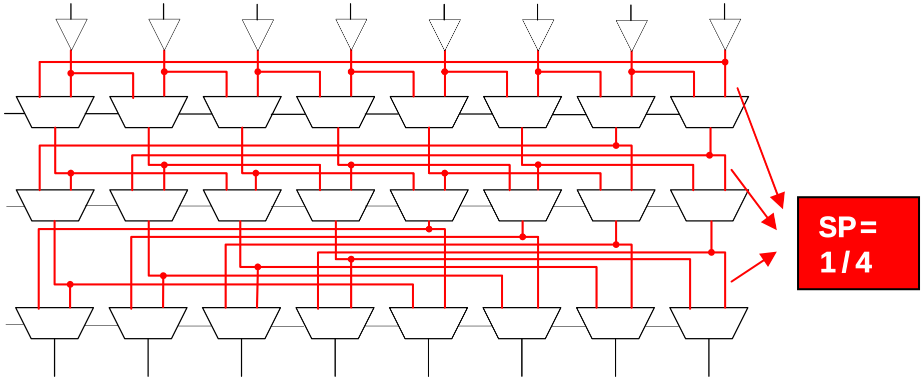
Red lines are signal lines

Green lines are quiet lines

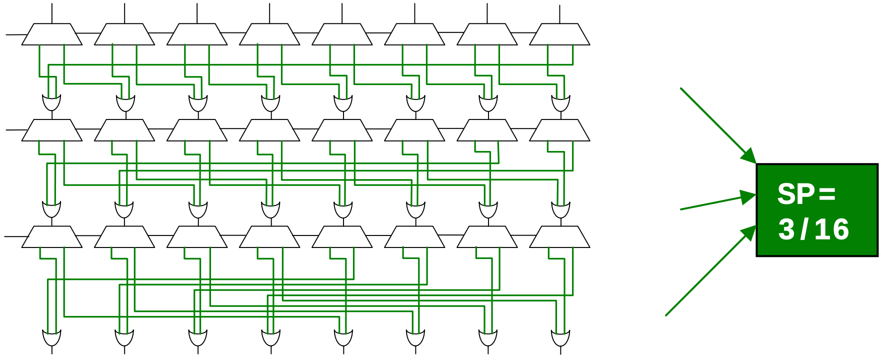
Dynamic Power Consumption

- Dynamic Power $P_{\text{dynamic}} = \frac{C_{\text{eff}} V_{\text{supply}}^2}{2} = \frac{(C_{\text{load}} P_{0 \to 1}) V_{\text{supply}}^2}{2}$
- Switching Probability $P_{X|0->1} = P_{X=0}(1 - P_{X=0})$

MUX based design



DEMUX based design

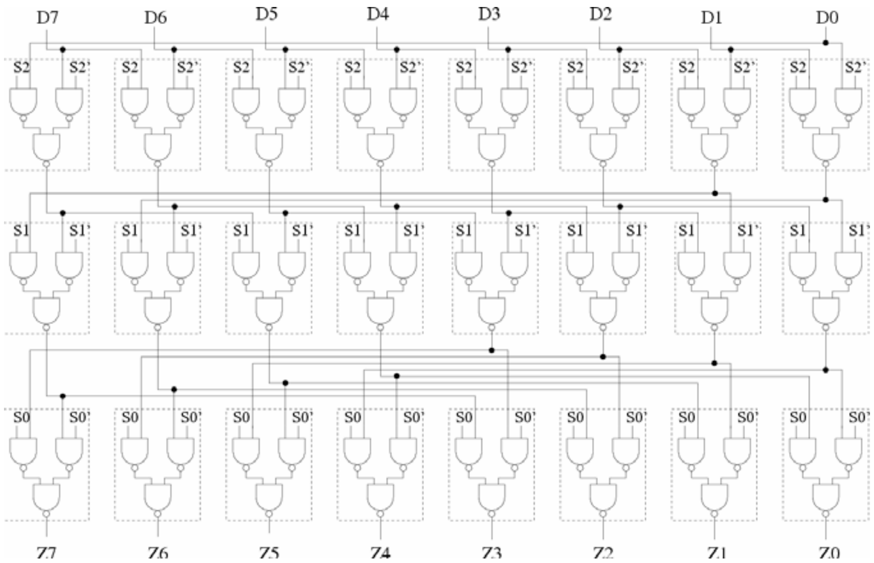


SP = Switching Probability

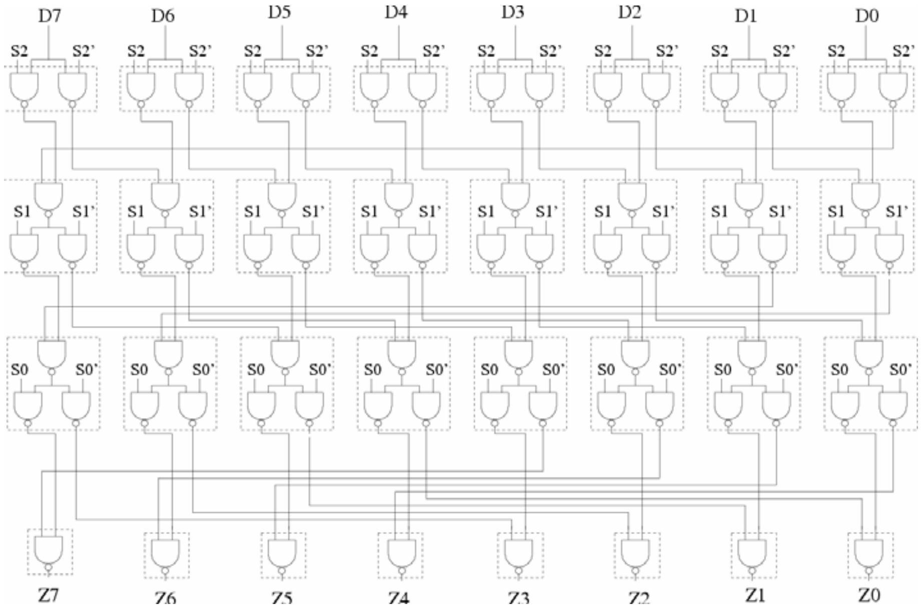
Gate Complexity

- Re-factoring design
  - No extra complexity at gate level, both are $O(N \log_2 N)$

MUX-based

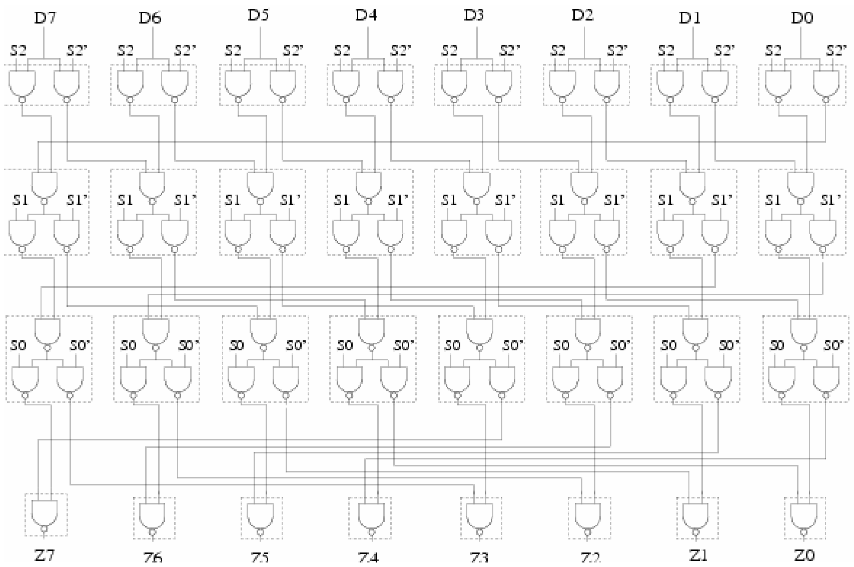


DEMUX-based

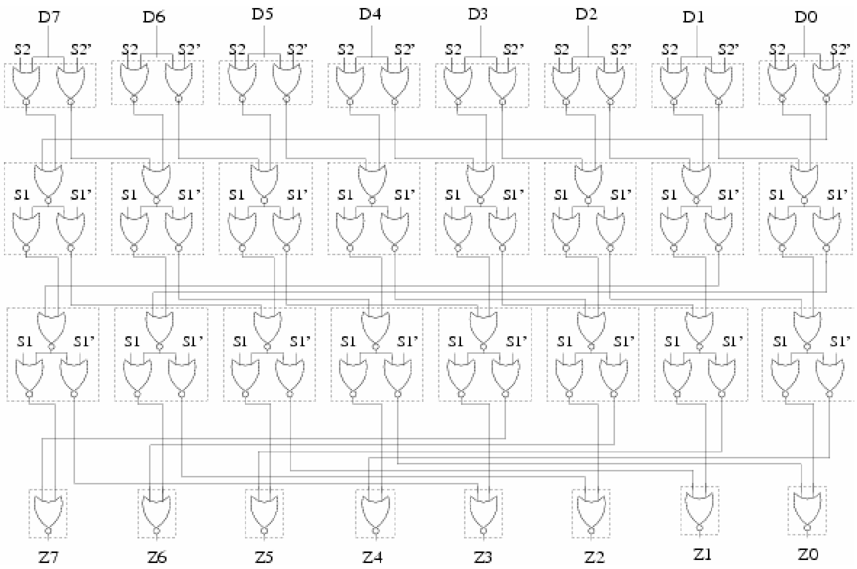


Duality

NAND gates network



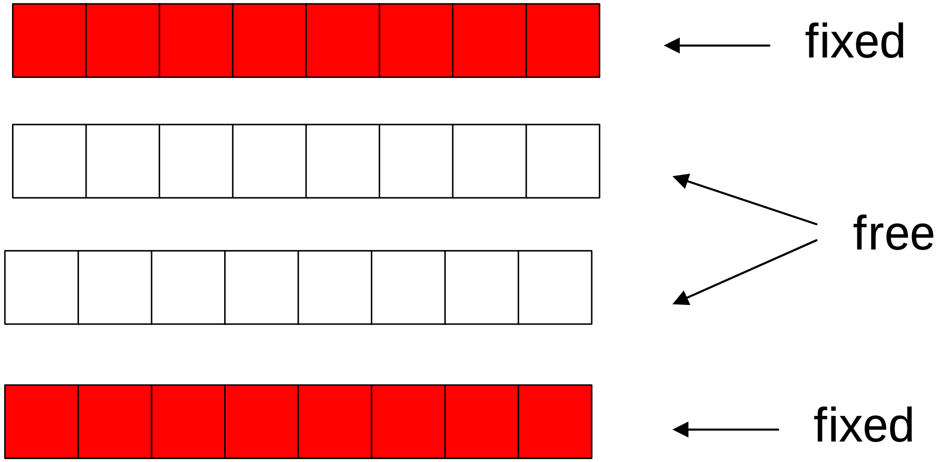
NOR gates network



- Duality provides flexibility for low level implementation
  - NAND gates are good for static CMOS.
  - NOR gates are good for dynamic circuits.

Cell Permutation

- Datapath usually assumes bit-slice structure
  - The cell order of the input/output stages must be fixed
  - However, the cells in the intermediate stages are free to permute.



Problem Statement

- Given
  - A N-bit rotator
  - Fixed linear order of the input/output stages
- Find
  - An optimal permutation scheme of the intermediate stages such that the longest path is minimized (or, the total wire length s.t. delay constraint).

ILP Formulation

- Introduce a set of binary decision variables $x^{l}_{ij} \in \{0, 1\}$
  - $x^{l}_{ij} = 1$ if and only if logic cell i is at physical location j on level l
- The solution space is fully defined by constraints

$$\sum_{i=0}^{N-1} x^{l}_{ij} = 1 \quad (0 \le j \le N-1, 1 \le l \le n-1)$$

$$\sum_{j=0}^{N-1} x^{l}_{ij} = 1 \quad (0 \le i \le N-1, 1 \le l \le n-1)$$

ILP Formulation (cont')

- Minimum delay formulation

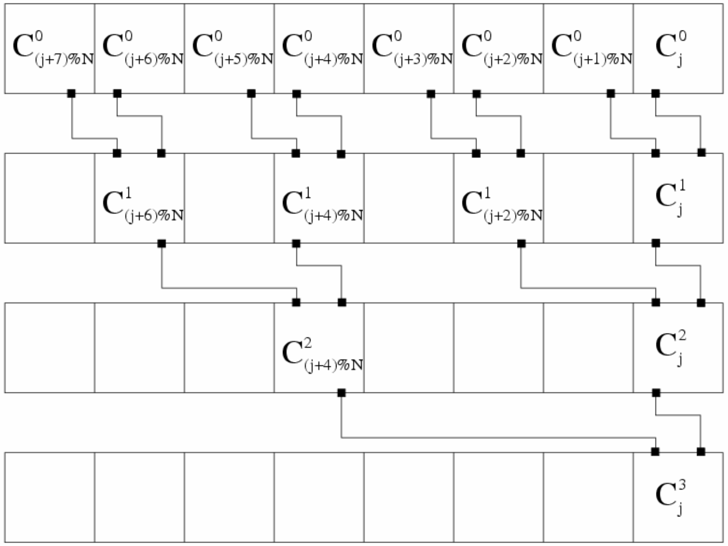
$$T_{\mathrm{max}} = \max\{\text{length of the delay path from } C_i^0 \text{ to } C_j^n,\ 0 \le i,j \le N-1\}$$

Which can be expanded into

$$T_{\mathrm{max}} \ge \text{length of the delay path from } C_i^0 \text{ to } C_j^n \text{ for } 0 \le i,j \le N-1$$

objective

minimize T_{max}



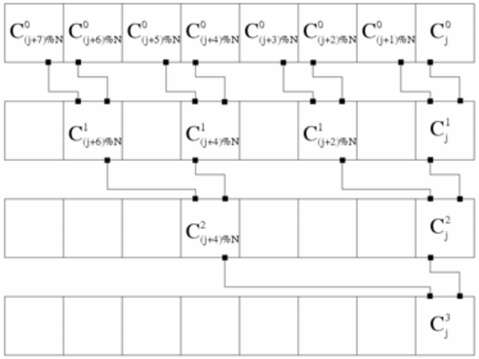
- Minimum power formulation

minimize T_{total}

s.t. $T_{\mathrm{max}} <$ const

ILP Formulation (cont')

- Represent the length of a single wire segment



$$d = \Big| \sum_{j=1}^{N-1} j \cdot x^0_{i_1 j} - \sum_{j=2}^{N-1} j \cdot x^1_{i_2 j} \Big|$$

- Formulating absolute operation

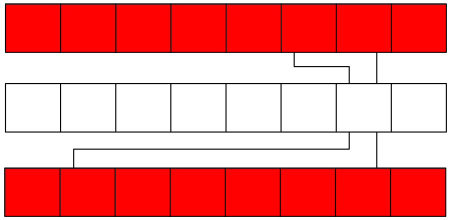
$$d = |D| \quad \Rightarrow d = \max\{D, -D\}$$

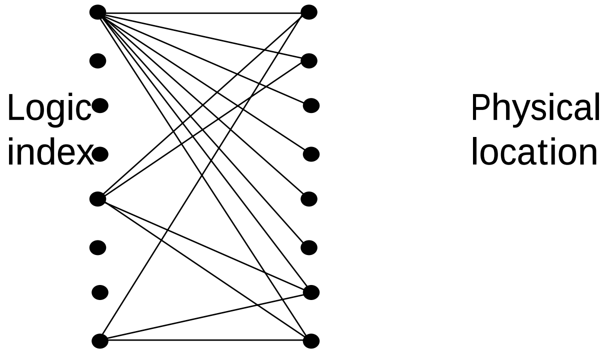
$$\Rightarrow \begin{cases} d \geq D \\ d \geq -D \\ \cancel{d \leq D + \mathrm{INF} \cdot g} \\ \cancel{d \leq -D + \mathrm{INF} \cdot (1-g)} \end{cases}$$

Psuedo-linear constraints discarded because we're trying to minimize

Complexity

- Minimum total wire length formulation
  - The case of one level of free cells is a minimum weight bipartite matching problem



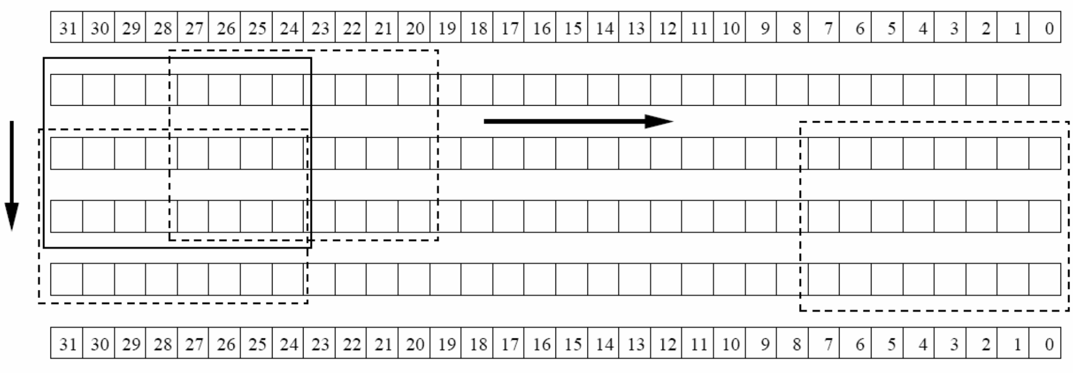


  - For the case of two or more levels of free cells, optimal polynomial algorithm is unknown
- Hardness of minimum delay formulation un-established.

ILP Complexity

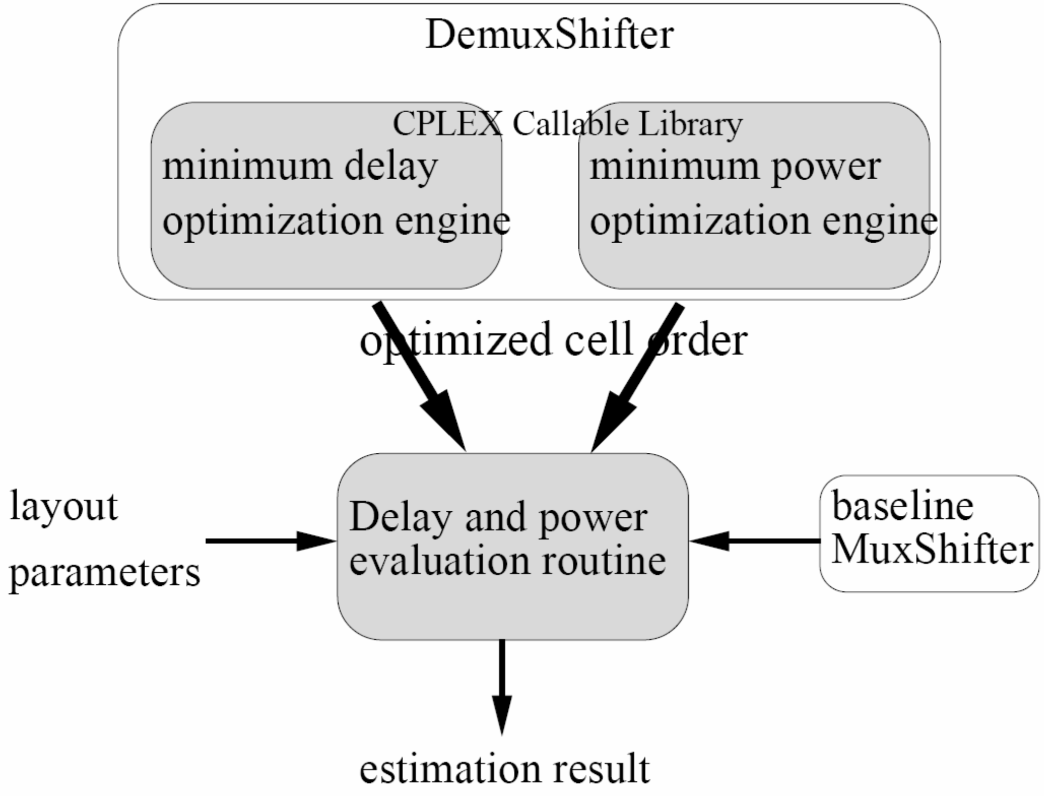
- The ILP formulation does not scale well
  - Both #integer variables and #constraints are $O(N^2)$
  - CPLEX uses on branch & bound – exponential growth
- Sliding window scheme
  - Only cells in the window are allowed to permute
  - Consists of multiple passes; terminate when there is no improvement between passes

WW = 8 columns
WH = 3 rows
HS = 4 columns
VS = 1 column



Power & Delay Evaluation

- Overall Flow



Optimal solution for 8-bit case

7	6	5	4	3	2	1	0

7	6	5	4	3	2	1	0

7	6	5	4	3	2	1	0

7	6	5	4	3	2	1	0

7	6	5	4	3	2	1	0

6	5	4	3	7	2	1	0

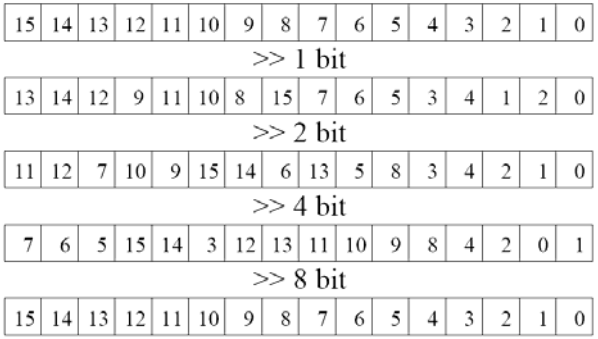
3	4	2	6	7	5	1	0

7	6	5	4	3	2	1	0

A global optimal solution

16-bit and 32-bit cases

16-bit, global optimal solution



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

>> 1 bit

13	14	12	9	11	10	8	15	7	6	5	3	4	1	2	0

>> 2 bit

11	12	7	10	9	15	14	6	13	5	8	3	4	2	1	0

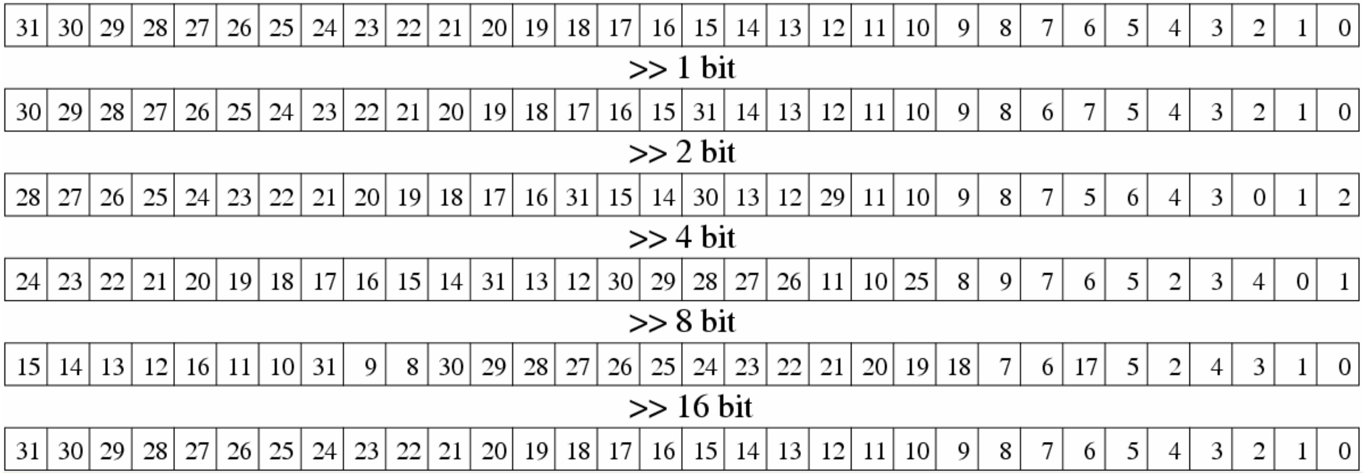
>> 4 bit

7	6	5	15	14	3	12	13	11	10	9	8	4	2	0	1

>> 8 bit

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

32-bit, suboptimal solution by sliding window method



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

>> 1 bit

30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	31	14	13	12	11	10	9	8	6	7	5	4	3	2	1	0

>> 2 bit

28	27	26	25	24	23	22	21	20	19	18	17	16	31	15	14	30	13	12	29	11	10	9	8	7	5	6	4	3	0	1	2

>> 4 bit

24	23	22	21	20	19	18	17	16	15	14	31	13	12	30	29	28	27	26	11	10	25	8	9	7	6	5	2	3	4	0	1

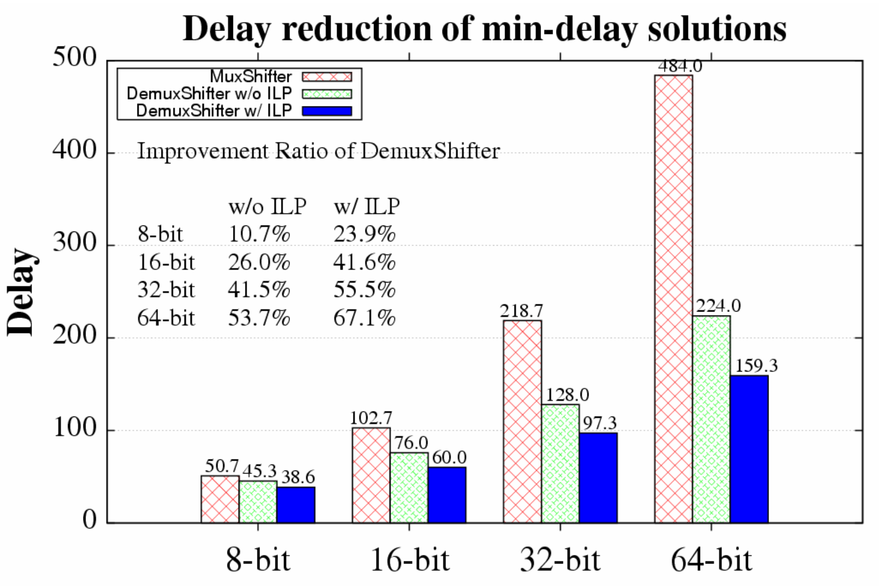
>> 8 bit

15	14	13	12	16	11	10	31	9	8	30	29	28	27	26	25	24	23	22	21	20	19	18	7	6	17	5	2	4	3	1	0

>> 16 bit

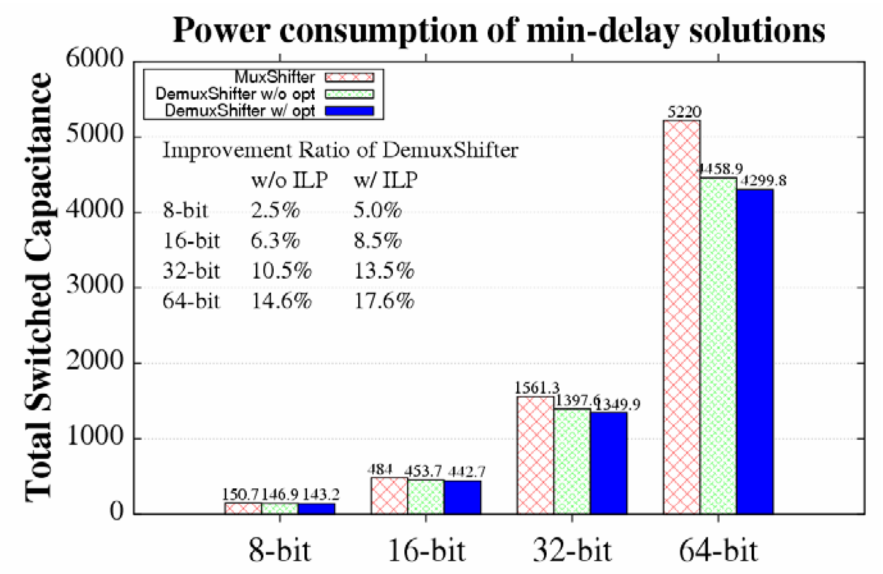
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Analysis Results



Improvement Ratio of DemuxShifter

	w/o ILP	w/ ILP
8-bit	10.7%	23.9%
16-bit	26.0%	41.6%
32-bit	41.5%	55.5%
64-bit	53.7%	67.1%



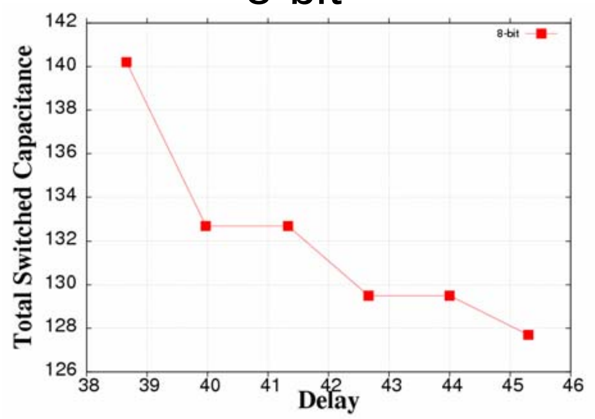
Improvement Ratio of DemuxShifter

	w/o ILP	w/ ILP
8-bit	2.5%	5.0%
16-bit	6.3%	8.5%
32-bit	10.5%	13.5%
64-bit	14.6%	17.6%

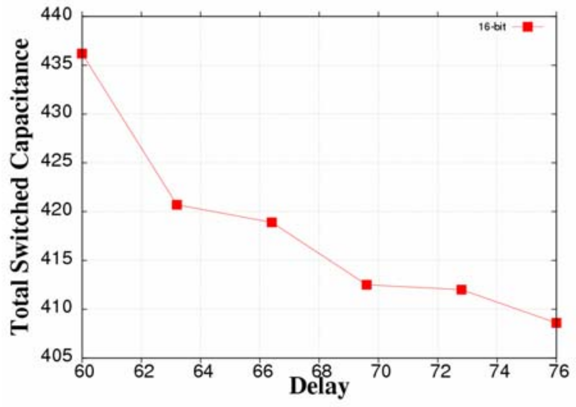
Power-Delay Tradeoff

- Given T_{max} constraint, optimize T_{total}

8-bit

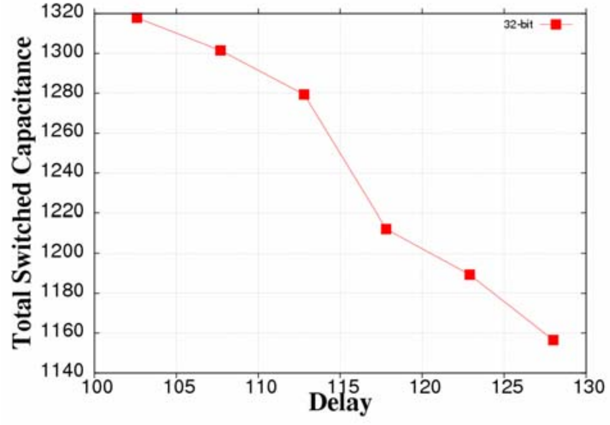


16-bit

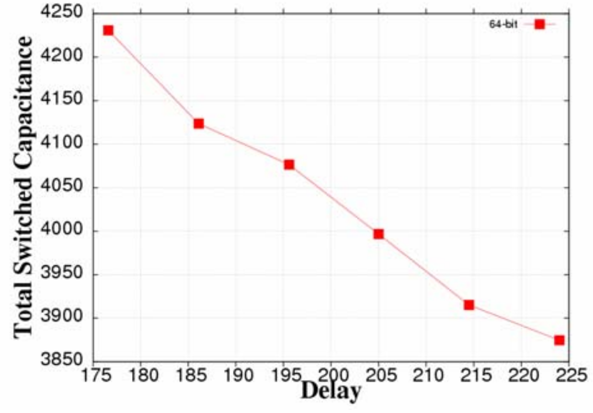


8-bit & 16-bit are global optimum by cplex

32-bit



64-bit



32-bit & 64-bit are suboptimal result by sliding window scheme

Implementation Results

- Implementation methodology
  - Standard cell based design using relative placement (by synopsys Physical Compiler)
  - Routing and timing analysis in synopsys Astro
  - Power estimation by synopsys PrimePower
  - Control signals manually buffered following FO4 rule
  - TSMC 90nm technology
  - Three types of designs investigated
    - Mux shifter using NAND2X2 gates
    - Mux shifter using MX2X2 gates
    - Demux shifter using NAND2X2 gates

Implementation Results – Cont’

- 64-bit results
  - Most improvement comes from the interconnect

	Delay Components			**muxsft64** ^{A} **(NAND2X2)**	**muxsft64** ^{B} **(MX2X2)**	**Demuxsft64** ^{C} **(NAND2X2)**	**Imp. (C/A)**
	Gate	Wire load	Xtalk				
Global critical path delay (ns)	√			0.7243	0.8731	0.7243	0%
	√	√		1.5534	1.5534	1.1135	28%
	√	√	√	1.9599	2.073	1.4503	26%
Critical data-in/data-out path delay (ns)*	√			0.4839	0.6120	0.4717	2.5%
	√	√		1.2520	1.1745	0.8473	32.3%
	√	√	√	1.5986	1.5578	1.0882	31.9%
Power (w) @ 500M	Cell Power			5.165e-04	7.707e-04	5.129e-04	0.7%
	Net Power			1.112e-03	9.907e-04	9.833e-04	11.6%
	Total			1.629e-03	1.761e-03	1.496e-03	8.2%

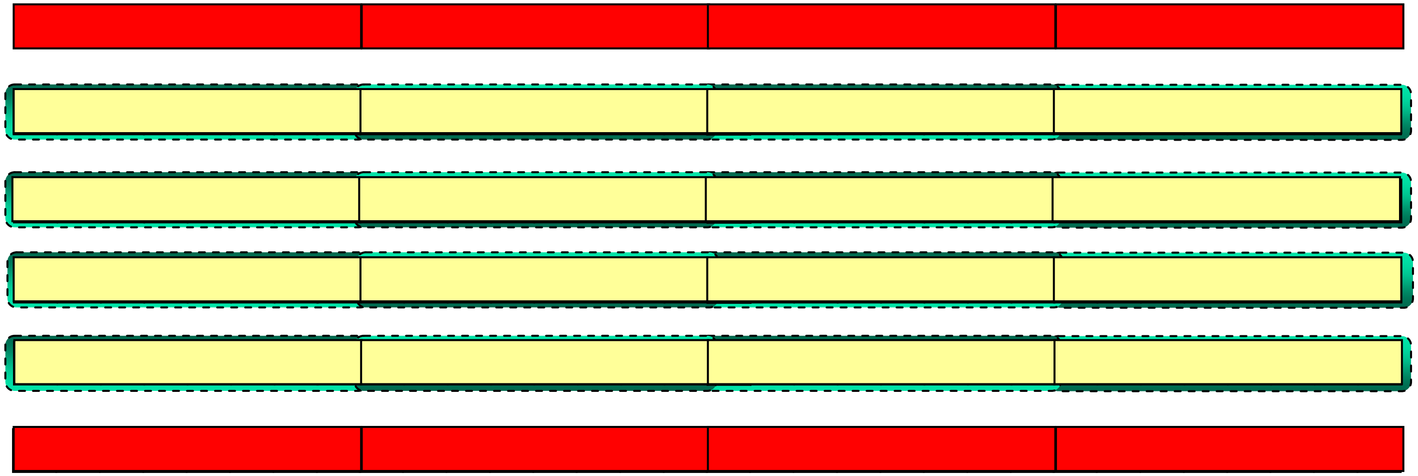
* For mux shifters, this is d[0]->z[0] while for demux shifter it is d[0]->z[1]

Outline

- Motivation
- Previous Work
- Approaches
  - Fanout-Splitting
  - Cell order optimization by ILP
- Conclusions and future work

Conclusions & Future Work

- We have proposed
  - Fanout-splitting design
  - ILP based layout optimization
- Future directions
  - Extend the fanout splitting idea and ILP formulation to ternary shifter
  - Try alternative hierarchical approach to tackle the ILP complexity issue



The End

Thank you!