Interconnect Design Methods for Memory

Chanseok Hwang, Massoud Pedram

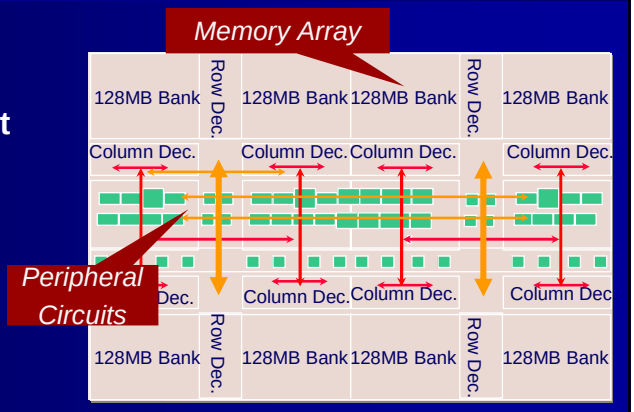
University of Southern California
Department of Electrical Engineering
Los Angeles CA, USA

Outline

- Issues in the layout design of modern memory devices
- Our solution
  - Fixing the net topologies
  - Crosstalk-aware over-the-cell channel routing
- Experimental results
- Conclusion

An Example Memory Device Layout

- **Memory Array**
  - **Regular layout structures**
  - **Full-custom design**
- **Peripheral Circuits**
  - **Random logic**
  - **Semi-custom design**



An example of 1GByte DRAM Architecture

Layout Design of Peripheral Circuits: Features

- High layout complexity
  - Error-prone task with low productivity requiring full-custom approaches
- Layout area minimization is the goal
- High-speed and crosstalk-immune signal designs are the constraints
  - Mixed analog and digital signals
- Lack of adequate design automation

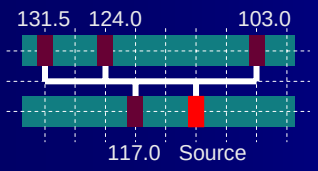
Our Solution

- Fixing the net topologies
  - Pre-define net topology for all nets
  - Well-known design (timing-critical signals, noise-sensitive signals)
  - Quality comparable to handcrafted layout
- Crosstalk-aware over-the-cell channel routing
  - Satisfy peak crosstalk noise constraints for noise-sensitive signals
  - Minimize channel height by using the over-the-cell area efficiently

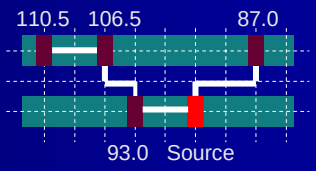
Fixed Net Topologies

- We define four types of routing topologies according to the performance constraints:
  - Delay critical and crosstalk noise sensitive nets, N_C, for address and data signal nets
  - Crosstalk noise sensitive nets, N_S, such as analog signal nets
  - Delay critical nets, N_T, for signals that lie on the timing-critical paths
  - Non-critical digital nets, N_B, for signals with enough noise margin and timing slack

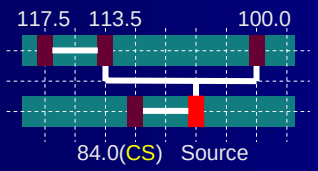
Routing Solutions for Different Fixed Net Topologies



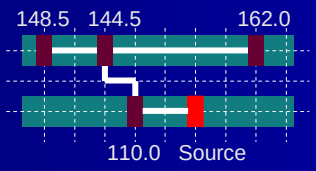
(a) Topology-I for N_C



(b) Topology-II for N_S



(c) Topology-III for N_T



(d) Topology-IV for N_B

Example Comparison of the Fixed Net Topologies

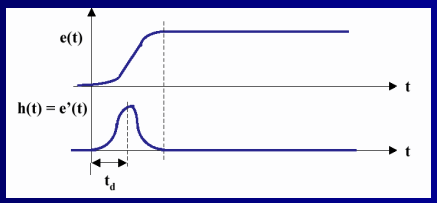
Topology type	I	II	III	IV
Critical sink delay	117 (39%)	93 (10%)	84	110 (30%)
Total routing length	12	11	12	12
Consumed channel area	12	7	8	3

Crosstalk-aware Over-the-Cell Channel Routing

- Propose practical models for delay, crosstalk effect, and the cell layout
- Performance-driven net partitioning (PDNP)
- Performance-driven track assignment (PDTA)

Delay Model

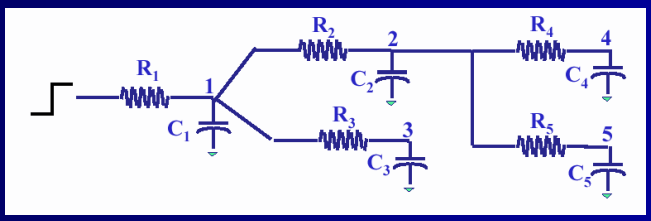
- We use Elmore delay for computing the interconnect delay



$$T_D = \int_0^\infty t \cdot h(t) dt$$

Elmore Delay for RC-tree

$$T_{delay} = \sum R_i C_{downstream,i}$$



$T_{delay,4}=R_1(C_1+C_2+C_3+C_4+C_5)+R_2(C_2+C_4+C_5)+R_4C_4$

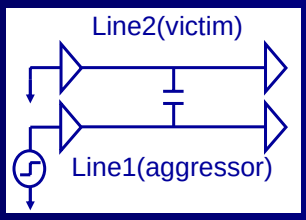
Crosstalk Noise Model

- We compute the peak crosstalk noise amplitude by using the equation proposed by Vittal et al
  - Easily computable expressions for crosstalk amplitude in RC coupled lines with the average error of about 10% and maximum error of less than 20%

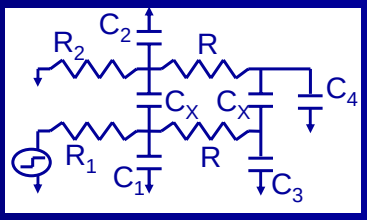
$$PN = \frac{\sum_{R_i \in \Omega} C_{xi} R_i}{\sum_{C_i \in \Lambda} C_i R_{ii}}$$

Ω: the union of the victim driver res. and the set of resistances in the path from the root to the node *i*
C_{xi}: the sum of downstream coupling cap. at node *i*
R_i: the upstream resistance along the path
Λ: the set of all capacitances
C_i: the *i* th capacitance
R_{ii}: the resistance seen across C_i when all other capacitors are open

An Example of Computing the Peak Crosstalk Noise



(a) Circuit configuration



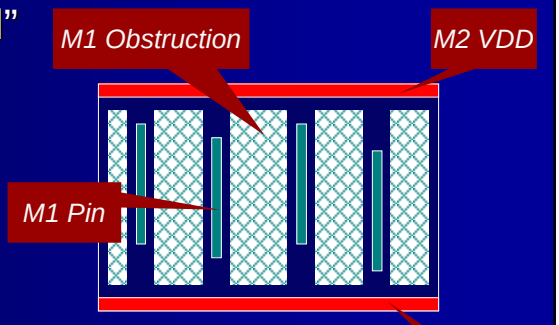
(b) Equivalent circuits

$$PN = \frac{(2R_2 + R)\,C_X}{R_1(C_1+C_3+2C_X) + R_2(C_2+C_4+2C_X) + R(C_3+C_4+2C_X)}$$

(c) Equation of computing peak crosstalk noise

Physical Cell Model

- "Target-based cell"
  - Efficiently use the over-the-cell area as a routing resource
  - Easy to apply automatic routing tools



Target based cell model

Crosstalk-aware Over-the-Cell Channel Routing

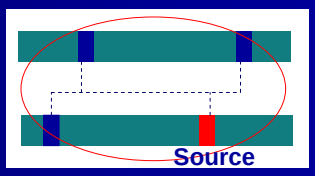
- Propose practical models for delay, crosstalk effect, and the cell layout
- Performance-driven net partitioning (PDNP)
- Performance-driven track assignment (PDTA)

Performance-Driven Net Partitioning (PDNP)

- Create a performance-driven tree for each net type
  - BRT for N_C, MST for N_S, CSRT for N_T, and MCAT for N_B
  - Split a net into subnets
  - Preserve the fixed-topologies during the routing process

PDNP (Cont'd)

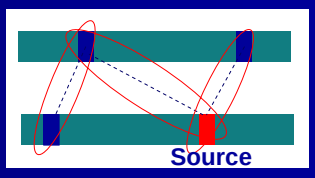
- Bus Routing Tree (BRT) for N_C
  - Bus signals such as address and data signal nets need to be routed in a similar routing topology
  - Obtain a steiner tree on a single track (nets are not split into subnets)



Forming BRT for N_C

PDNP (Cont'd)

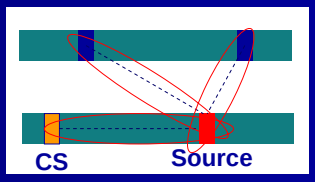
- Minimum Spanning Tree (MST) for N_S
  - Crosstalk noise is caused by the coupling capacitance, which is proportional to the coupling length
  - Obtain a tree with the minimum total length (coupling capacitance) with Prim's algorithm



Forming MST for N_S

PDNP (Cont'd)

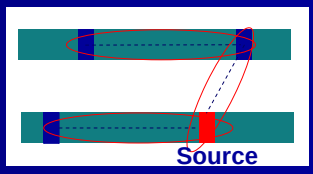
- Critical Sink Routing Tree (CSRT) for N_T
  - Minimize the delay from the source to the critical sink (CS)
  - Based on the ERT algorithm
  - Create a MST for (T – {cs}) and then connect cs to T at the right point
  - Obtain a tree with the minimum critical sink delay from the source



Forming CSRT for N_T

PDNP (Cont'd)

- Minimum Channel Area Tree (MCAT) for N_B
  - Cell height in peripheral circuits tends to be 100um(5~10X in ASIC)
  - First, form two subnets with pins in the top-row and bottom-row, respectively, then create another subnet connecting the two subnets with the minimum length
  - Obtain a spanning tree T that uses the minimum channel area



Forming MCAT for N_B

Crosstalk-aware Over-the-Cell Channel Routing

- Propose practical models for delay, crosstalk effect, and the cell layout
- Performance-driven net partitioning (PDNP)
- Performance-driven track assignment (PDTA)

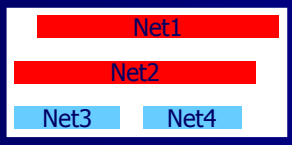
Performance-Driven Track Assignment (PDTA)

- Assign nets or sub-nets to tracks while meeting the following constraints
  - Horizontal constraints (HC) for all nets
  - Vertical constraints (VC) for all nets
  - Noise constraints (NC)
    - $PN(n_i) \leq AN(n_i)$ for a net $n_i \in \{N_C, N_S\}$, where

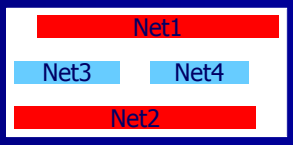
      $PN(n_i)$: Peak crosstalk noise for n_i

      $AN(n_i)$: Allowable peak crosstalk noise for n_i

PDTA (Cont'd)



(a) *PN(Net2) > AN(Net2)*

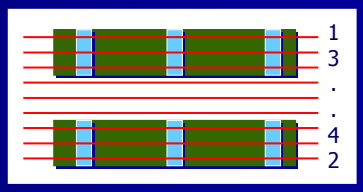


(b) *PN(Net2) < AN(Net2)*

An Example of the track assignment satisfying noise constraints where Net1 and Net2 $\in N_S$ *or* N_C, and Net3 and Net4 $\in N_B$

PDTA (Cont'd)

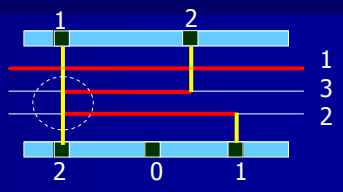
- Track ordering
  - PDTA proceeds track by track from the over-the-cell area towards the middle of channel
  - Over-the-cell area can be efficiently used
  - Vertical constraint violation (VCV) can be easily checked



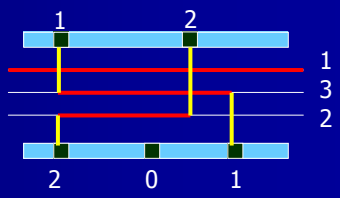
Track Ordering

VCV Checking Rules

- Efficient determination of whether a VCV occurs
  - If the track has been selected from the bottom part of the channel, then a VCV will occur exactly if
    - net enters from the top, and
    - the net that is entering from the bottom in the same column has not yet been assigned to the track
- No vertical constraint graph is needed to check the VCV's



(a) A VCV occurs when assigning net1 on track2



(b) No VCV occurs when assigning net2 on track2

TANAR Algorithm

1. All nets are classified into four types and split into subnets according to their types using PDNP;
2. Sort subnets in the increasing order of their leftmost end positions;
3. Select a track *T* from the over-the-cell region toward the middle of a channel;
4. Select the next subnet *n* with the lowest left-end position;

   If *n* satisfies *HC*, *VC* and *NC*

   Place *n* on the current track *T*;

   Delete *n* from the sorted list;

   Else Goto 4 if there is an untried subnet in the list

   Repeat step 3 and 4 until all subnets are assigned

Experimental Results

- We randomly assigned net types for the example circuits
  - N_C (10%), N_S (15%), N_T (25%) and N_B (50%)
- We assigned the maximum allowable peak noise *AN*, ranging from 10% to 30% of supply power voltage, for nets in N_C or N_S

The characteristics of benchmark circuits

Circuits	# of Nets	# of Columns	Channel Density
Deutsch	72	174	19
Rand100	100	220	20
Rand200	200	480	39
Rand300	300	720	60
Rand400	400	870	81
Rand500	500	1200	90

Experimental Results (Cont'd)

Test Circuits	TANAR				CONV			
	Peak Noise (V) for N_S and N_C	CS-Delay (ps) for N_T	# of Used Tracks		Peak Noise (V) for N_S and N_C	CS-Delay (ps) for N_T	# of Used Tracks	
			In-the-Channel	Over-the-Cell			In-the-Channel	Over-the-Cell
Deutsch	0.17 (-67.0%)	28.9(-4.0%)	13	24	0.53	30.4	16	15
Rand100	0.06 (-40.9%)	26.5(-5.6%)	16	27	0.10	28.1	17	14
Rand200	0.20 (-67.7%)	33.8(-7.3%)	25	45	0.62	36.5	26	28
Rand300	0.36 (-60.0%)	36.9(-7.7%)	44	77	0.90	40.0	46	40
Rand400	0.41 (-62.0%)	40.7(-13.5%)	41	86	1.08	47.1	45	47
Rand500	0.52 (-57.4%)	46.4(-16.5%)	54	92	1.22	55.6	56	56

Conclusion

- We solved the automatic routing problem of memory peripheral circuits as an over-the-cell channel routing problem under performance constraints
- Our approach handles noise sensitive nets and timing critical nets at the same time while efficiently utilizing the over-the-cell area
- Experimental showed that the proposed routing algorithm reduces the peak crosstalk noise by 40%~67% and the critical-sink delay by 4%~16% while reducing (or preserving) the channel height