Design and Allocation of Loosely Coupled Multi-bit Flip-flops for Power Reduction in Post-Placement Optimization
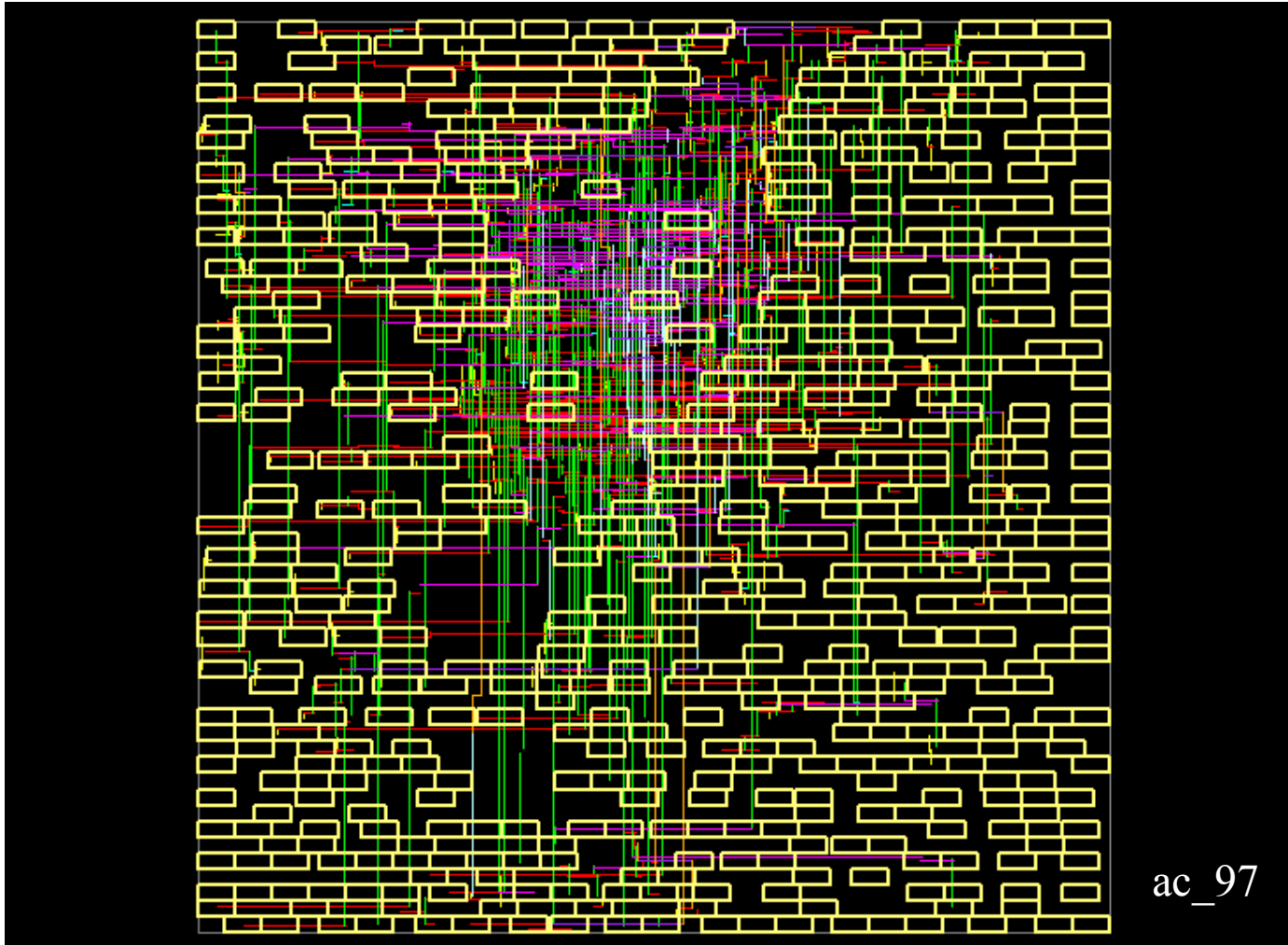
Hyoungseok Moon & Taewhan Kim
**Seoul National University**

Outline

- **Introduction**
  - Multi-bit flip-flop and related works
- **New Style of Multi-bit Flip-flops**
- **Allocation Algorithm**
  - Minimizing power consumption
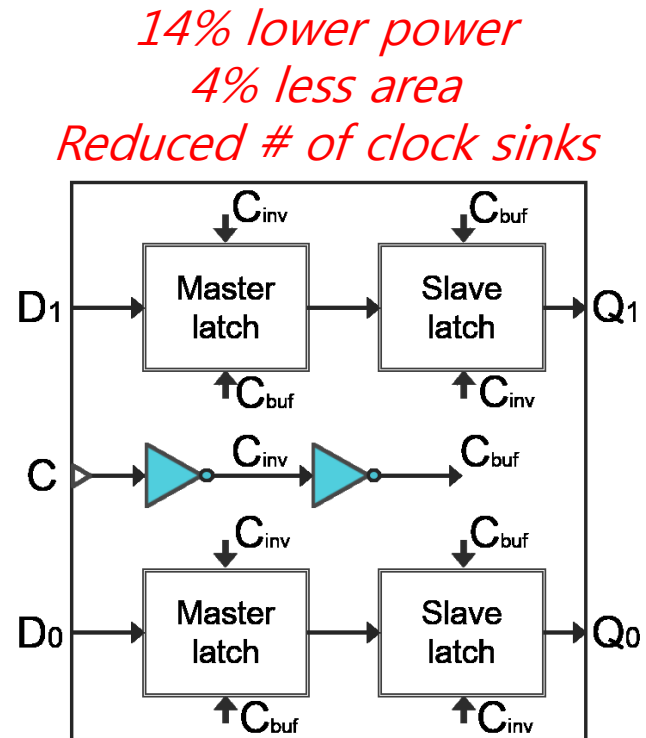 - Awareness of clock network
- **Experimental Results**
- **Conclusion**

So Many Flip-flops in SoC

ac_97

Conventional Structure of Multi-bit Flip-flops

14% lower power
4% less area
Reduced # of clock sinks
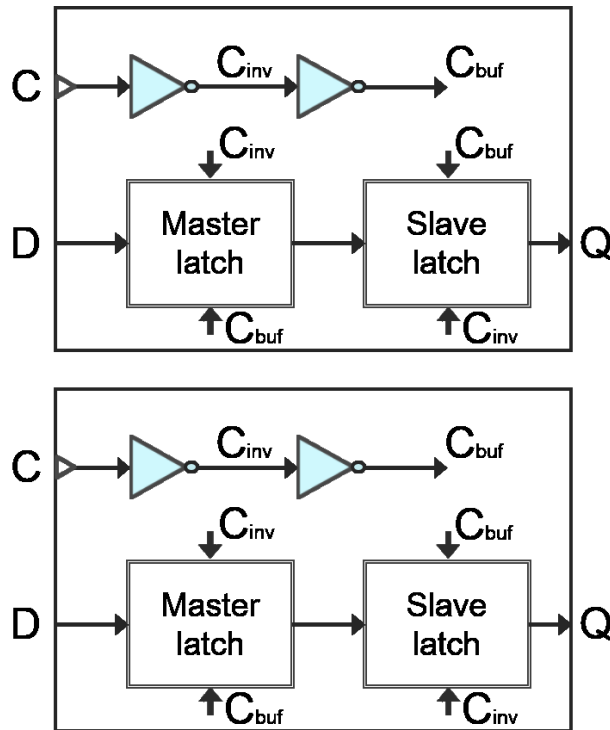
(a) Two 1-bit flip-flops

(b) 2-bit flip-flop

Related Works and their Limitations

Pre-placement

Kretchmer, SNUG'99

Hard to estimate timing/placement

In-placement

Hsu, ICCAD'13

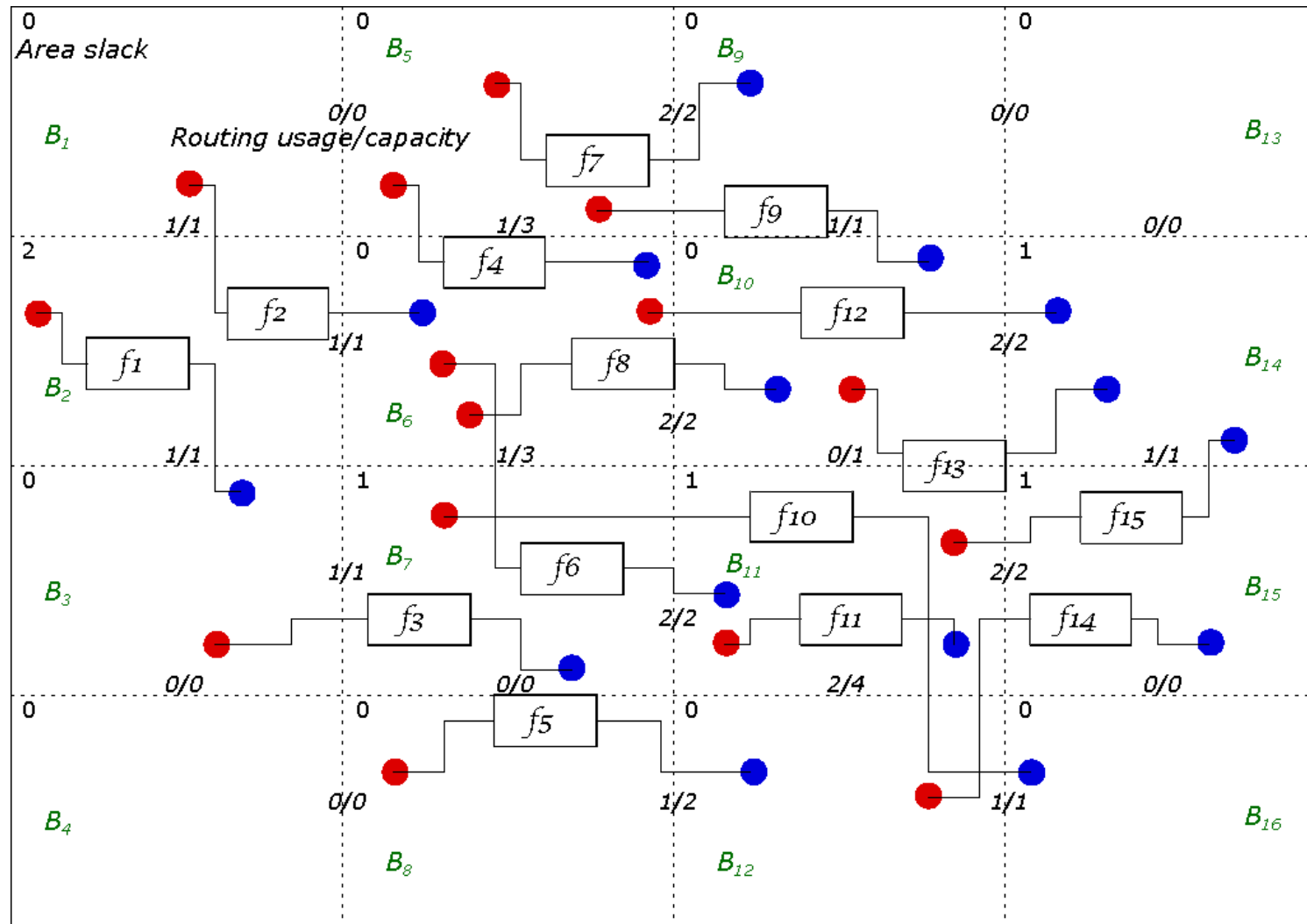1. Merging FFs to allocate MBFFs
2. Placing MBFFs

Post-placement

Yan, ICGCS'10
Chang, ICCAD'10
Jiang, TCAD'12
Wang, TCAD'12
Chen, Integration'13
Shyu, TVLSI'13

Constraints
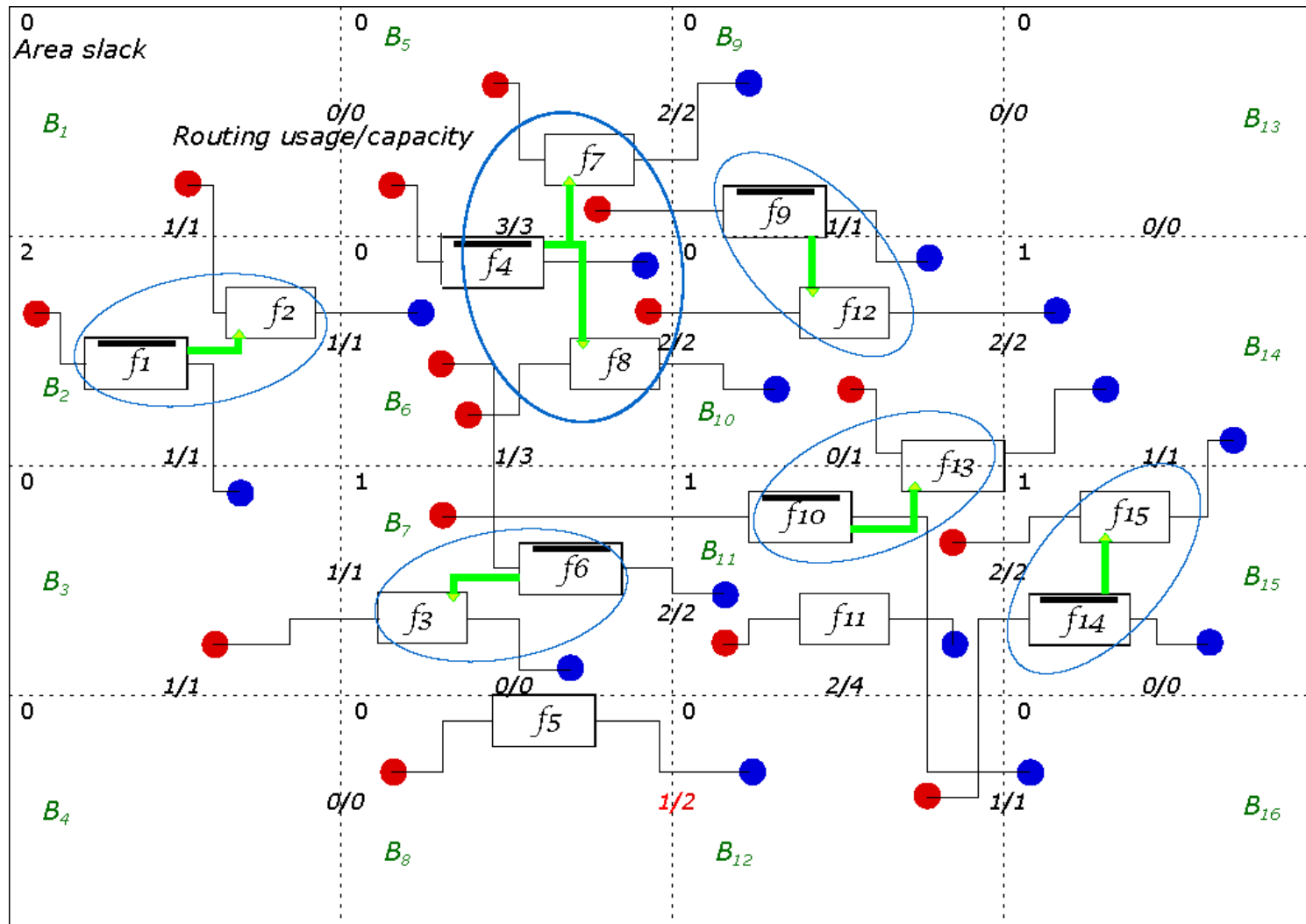1. Timing
2. Space
3. Routability
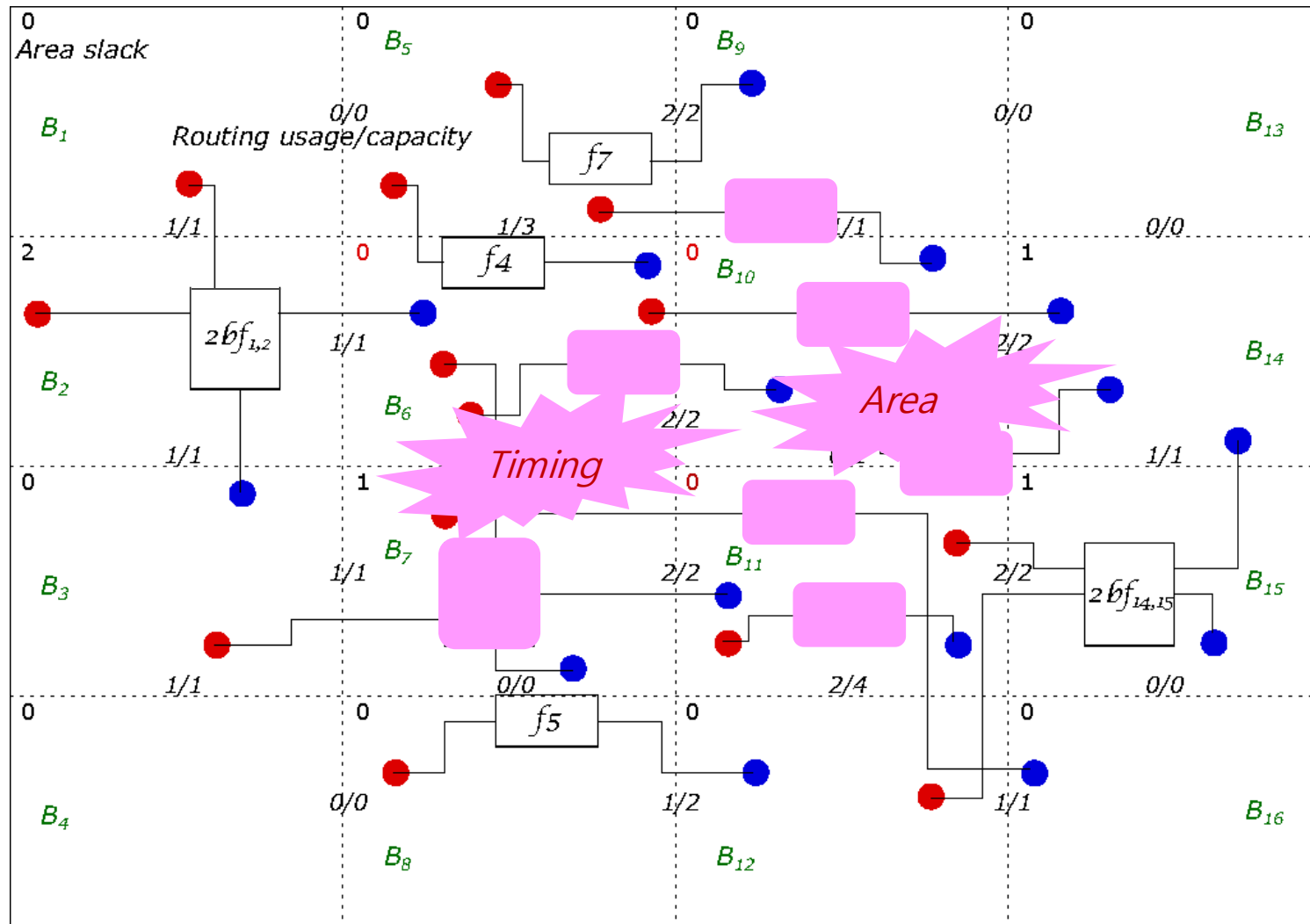
Loosely Coupled Multi-bit Flip-flop

■ Structure

❑ Flip-flops are merged via *"Sharing nets"*
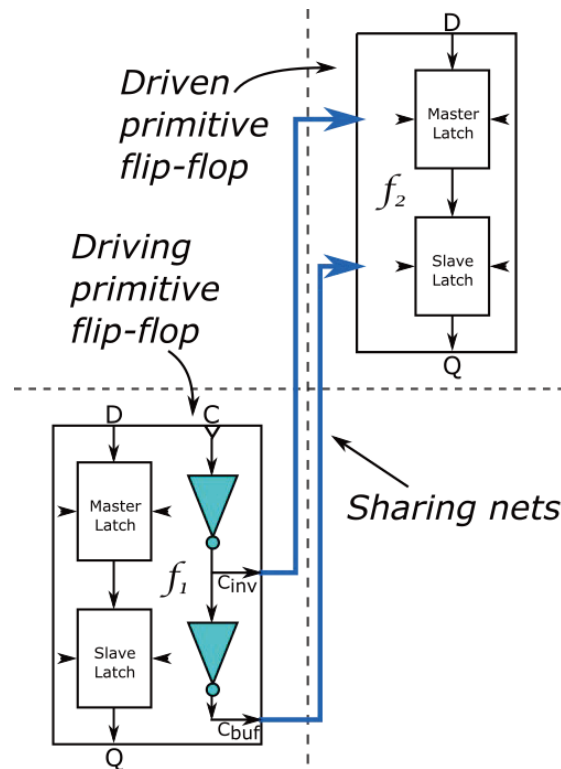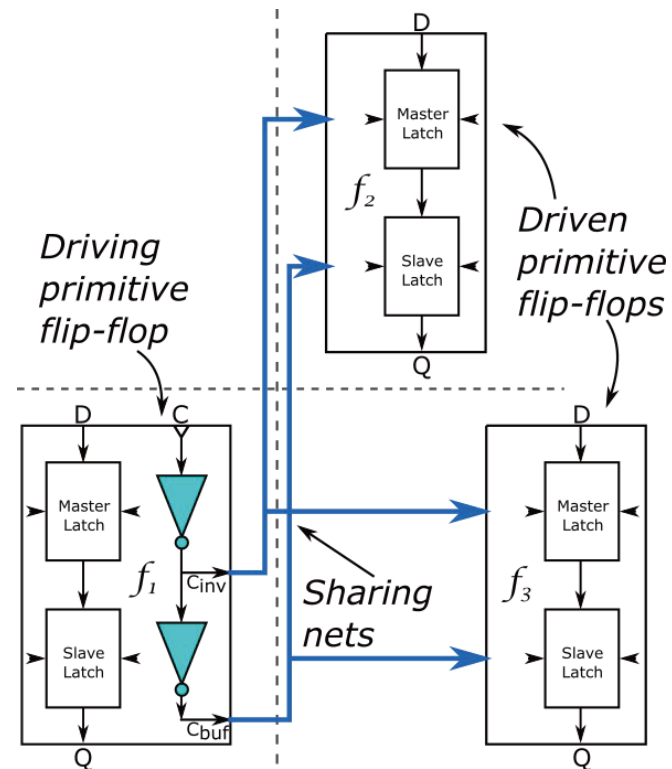
(a) 2-bit LC-MBFF   (b) 3-bit LC-MBFF

- **The shorter, the better**

(a) (b) (c)

The Shortest!
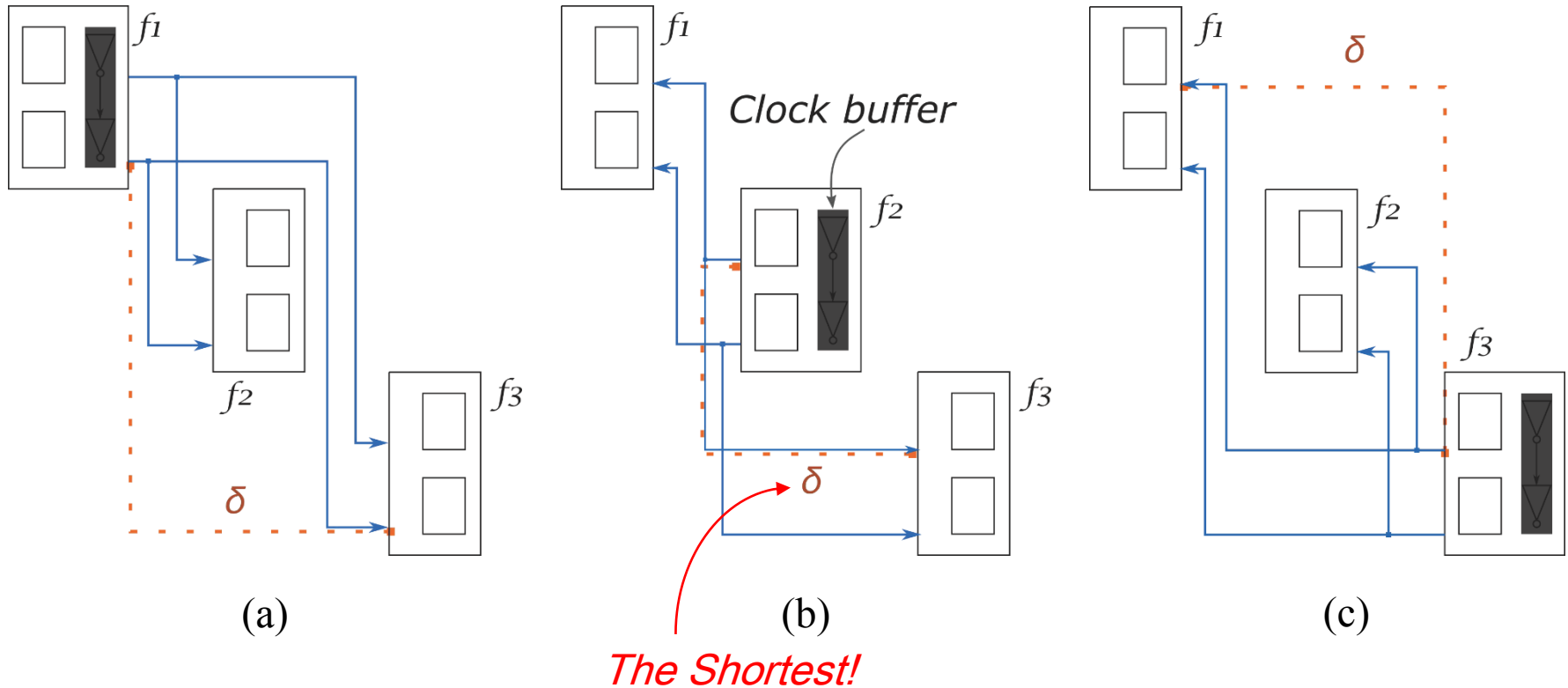
Feasibility Analyses of LC-MBFF

- **2-bit & 3-bit LC-MBFF libraries implemented**
  - Wires for clock sharing net are modeled with PTM interconnect structure and 45nm Open Cell Library

Dimension	Value
Width	0.08um
Space	0.08um
Thickness	0.20um
Height	0.20um
Length (δ)	4um ~ 30um

  - HSPICE simulations with 500MHz operating clock

Time Delay of LC-MBFF

- **Clock skew is very negligible**
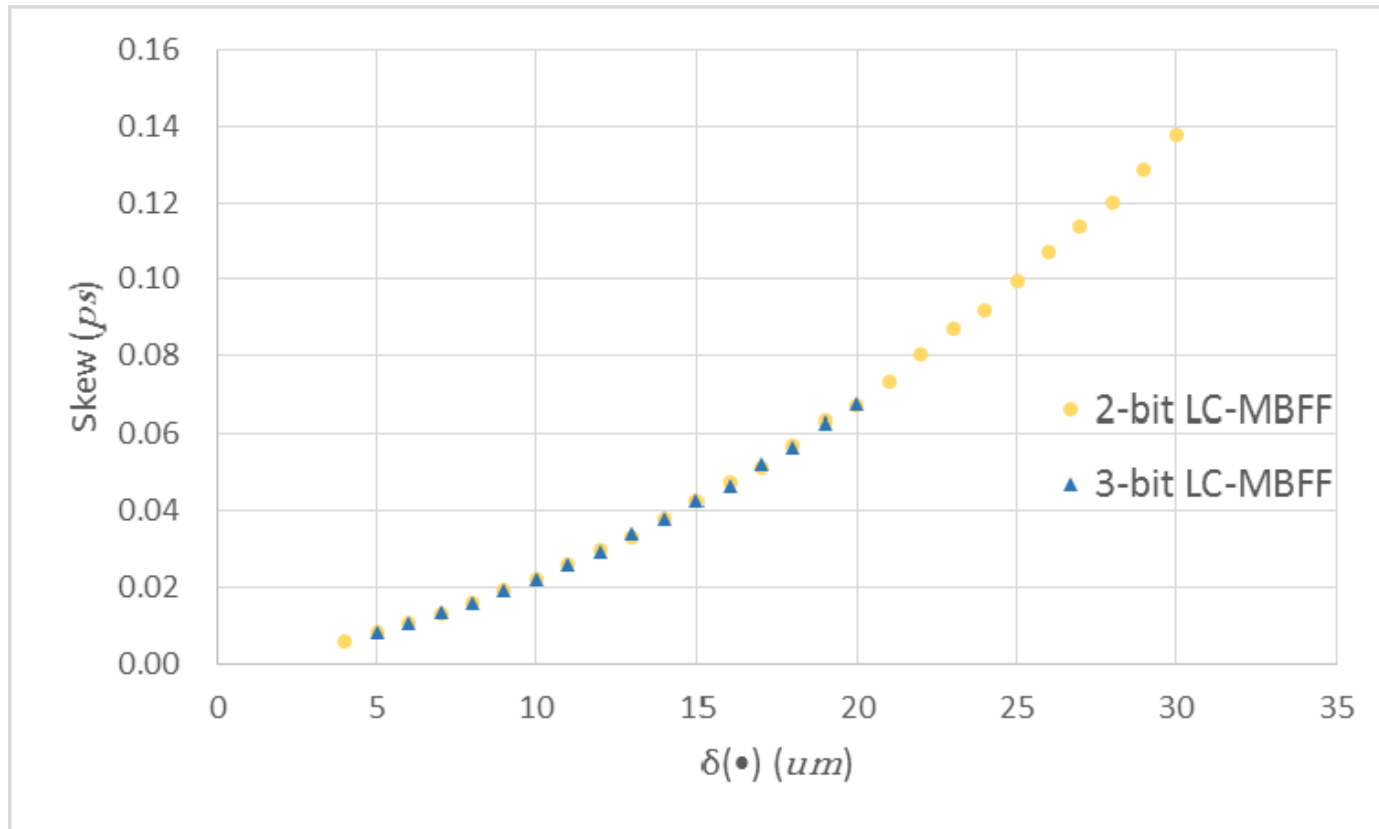
Power Consumption

- **More power saving with closer flip-flops**
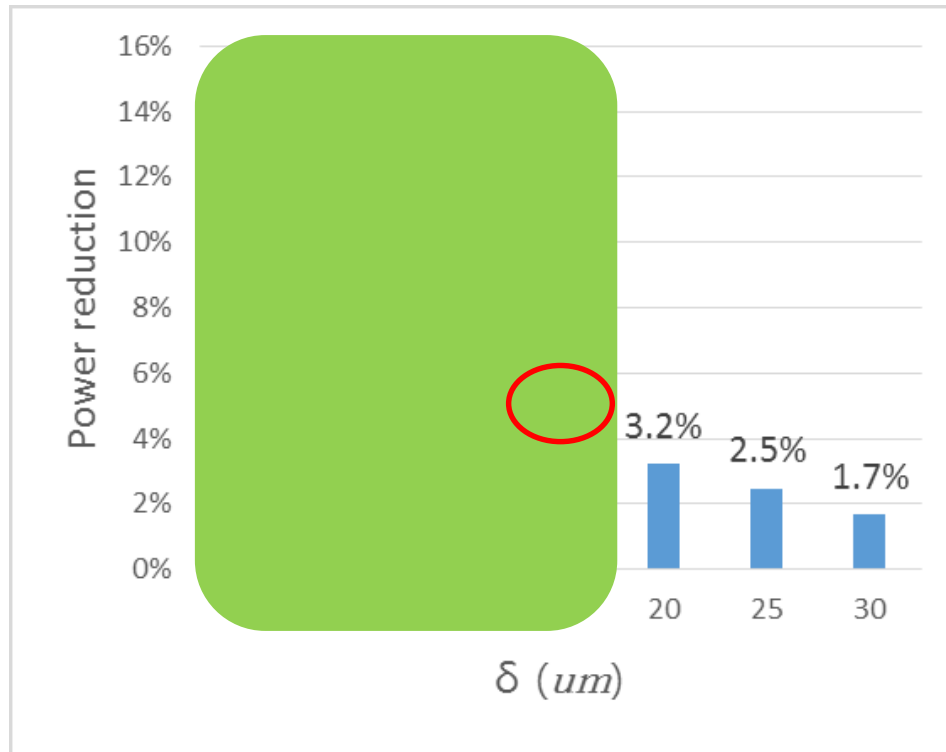
(a) 2-bit LC-MBFF
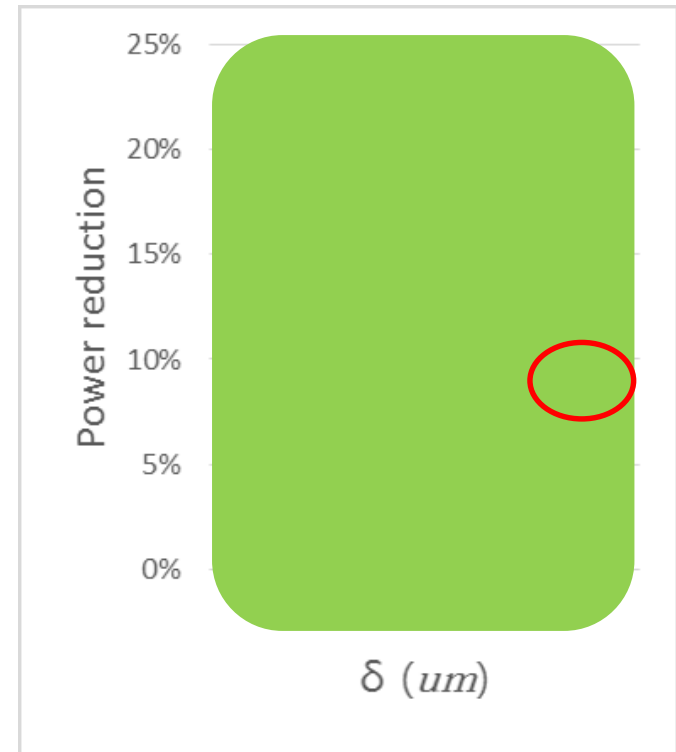
(b) 3-bit LC-MBFF

LC-MBFF Allocation Rules

- **Flip-flops in a close distance**

$$\delta(f) \leq D_{max}^{k-bit}$$

- **Flip-flops in the same level of clock tree**
  - Simpler resulting clock tree
- **Routability of sharing nets**

LC-MBFF Allocation Flow
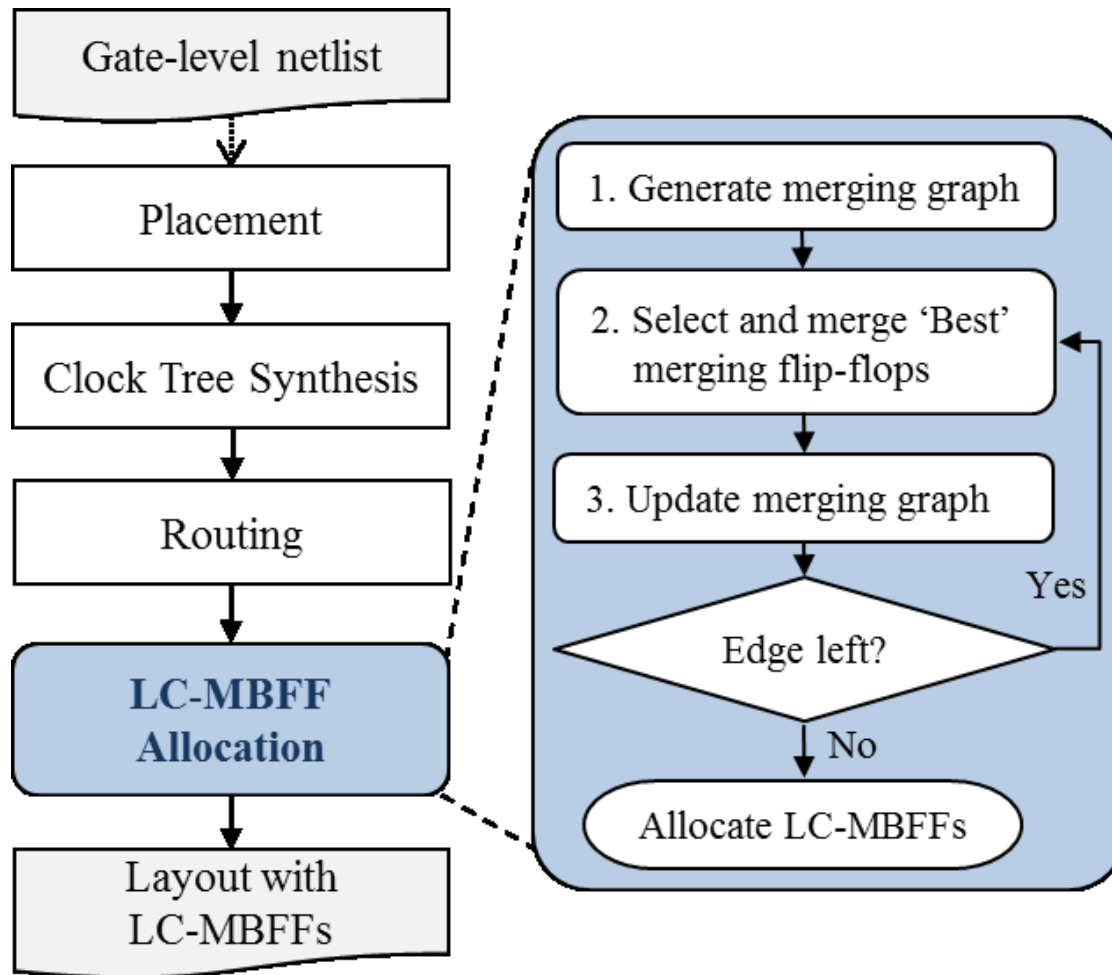
Generate Merging Graph

1. Generate merging graph

2. Select and merge 'Best' merging flip-flops

3. Update merging graph

Edge left? Yes

No

Allocate LC-MBFFs

$\overline{\langle f_1, f_2 \rangle}$

$\langle f_7 \rangle$

$\langle f_4 \rangle$

$\langle f_8 \rangle$

$\langle f_2 \rangle$

$\langle f_1 \rangle$

$\langle f_6 \rangle$

$\langle f_3 \rangle$

$\langle f_5 \rangle$

16

10

15

19

1st best merging

8

12

17

2nd best merging

Iterate if Any Edge Left

1. Generate merging graph
2. Select and merge 'Best' merging flip-flops
3. Update merging graph
Edge left? → Yes
No → Allocate LC-MBFFs

3rd best merging
$\overline{\langle f_4, f_7 \rangle}$ **16**

$\langle f_7 \rangle$
$\langle f_4 \rangle$ 10

$\overline{\langle f_1, f_2 \rangle}$
$\langle f_2 \rangle$ 15
19 $\langle f_8 \rangle$
$\langle f_1 \rangle$

Removed during update

13 Updated weight

$\langle f_6 \rangle$
$\langle f_3 \rangle$ 17

$\overline{\langle f_3, f_6 \rangle}$

$\langle f_5 \rangle$
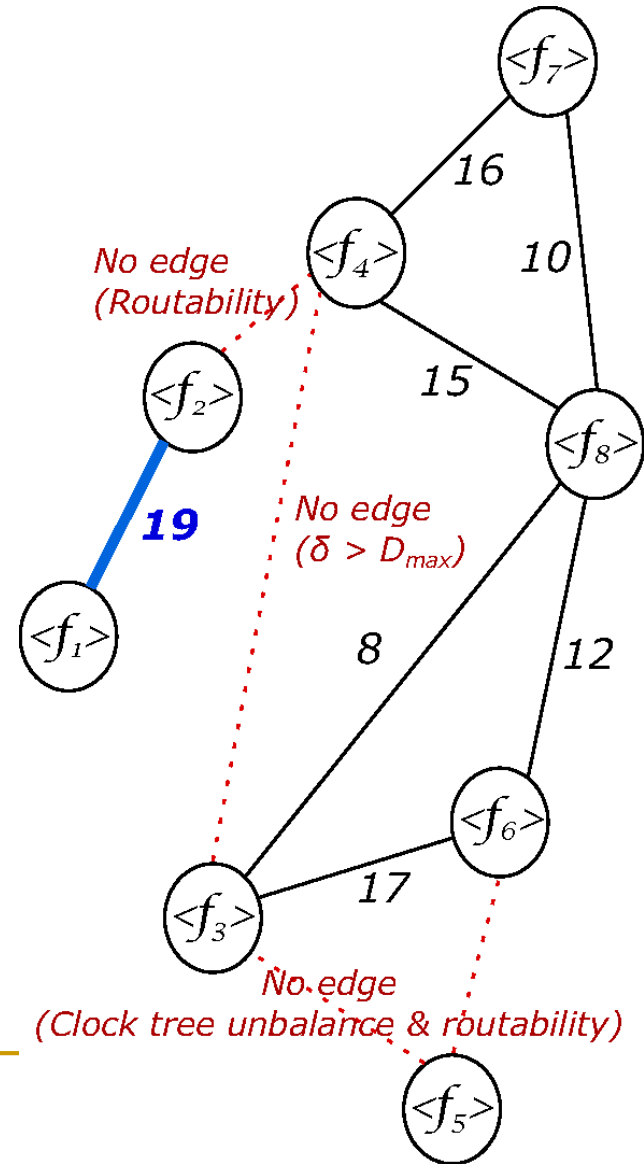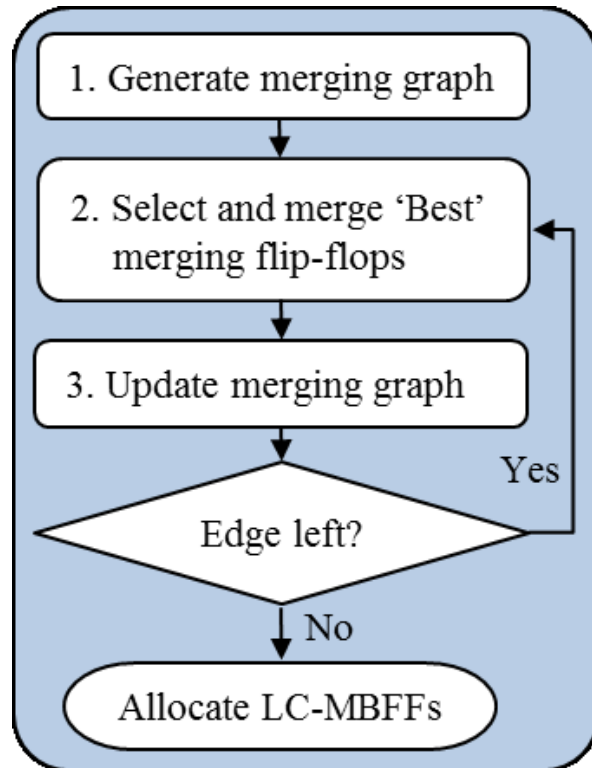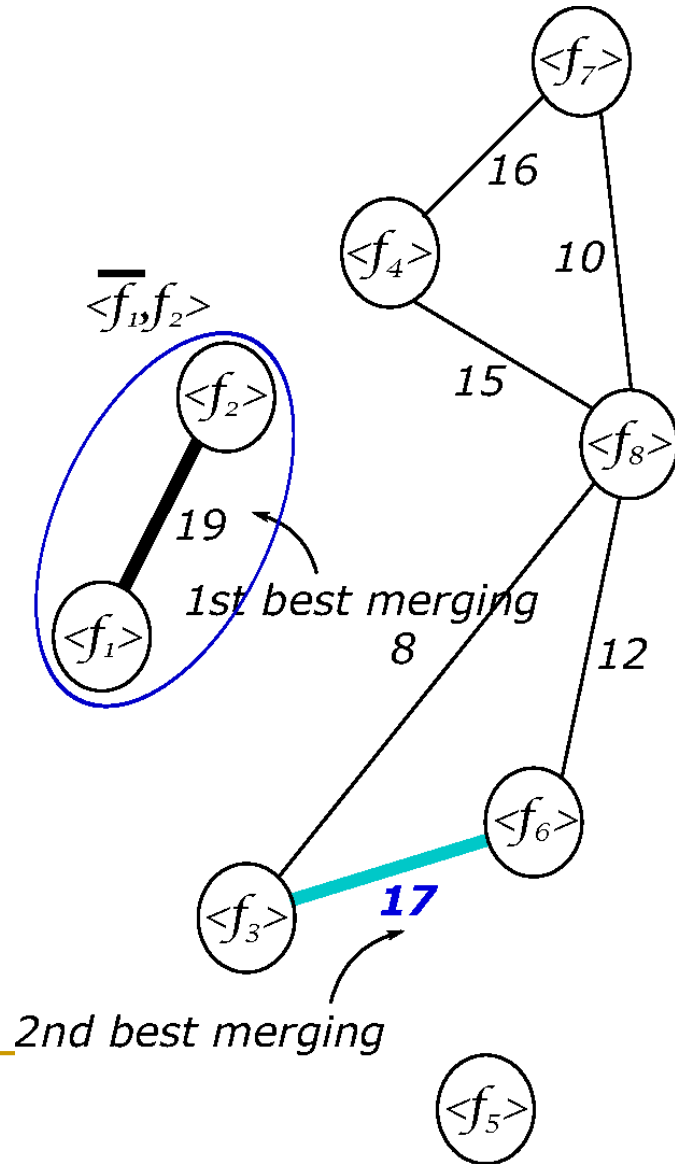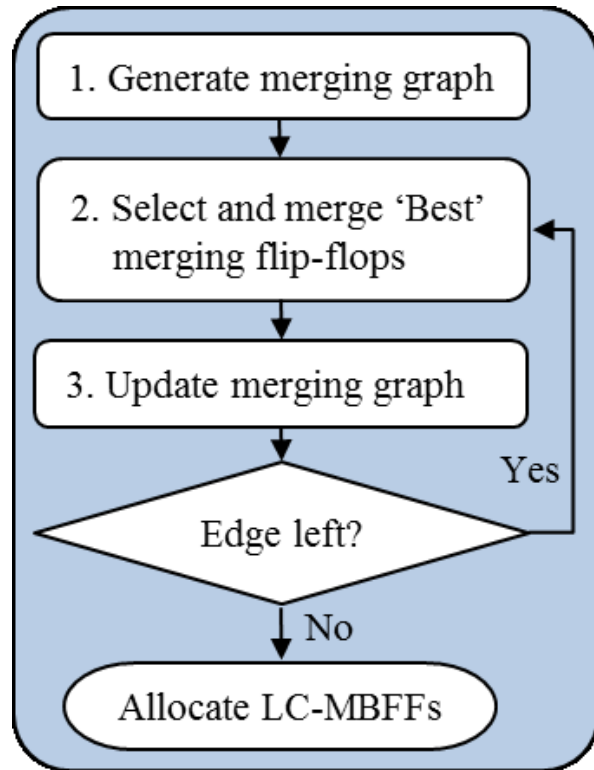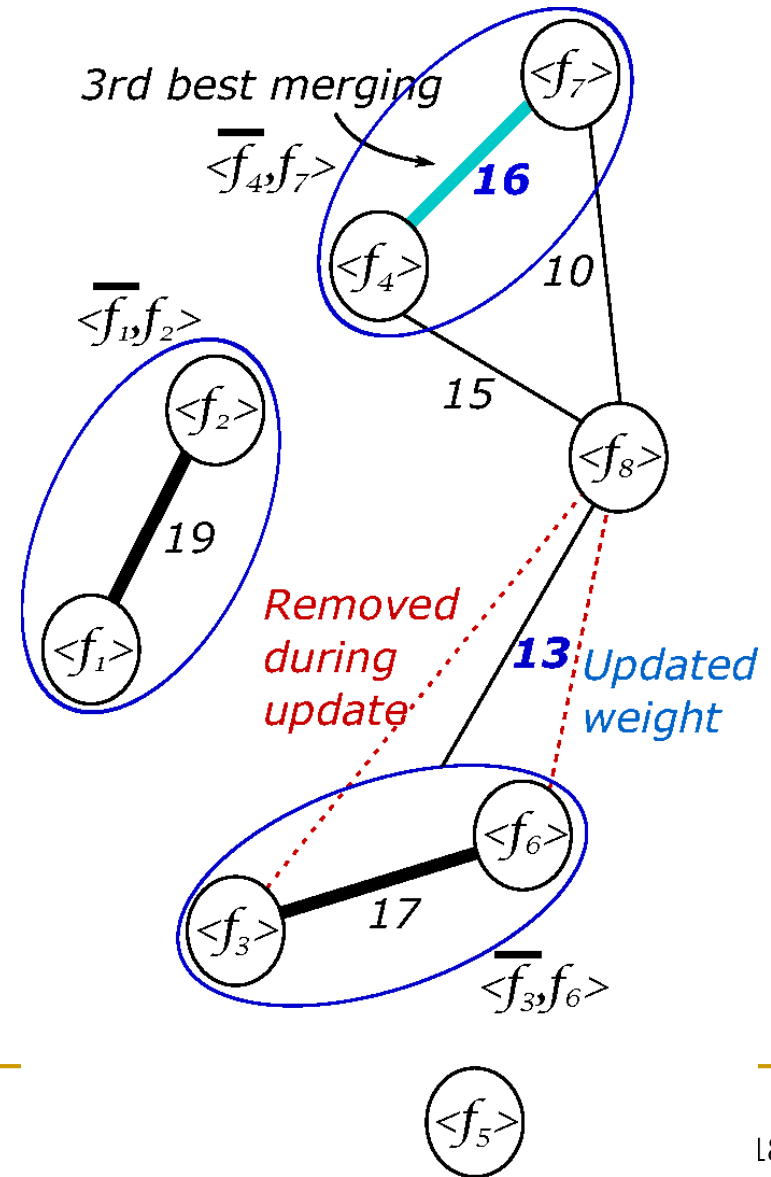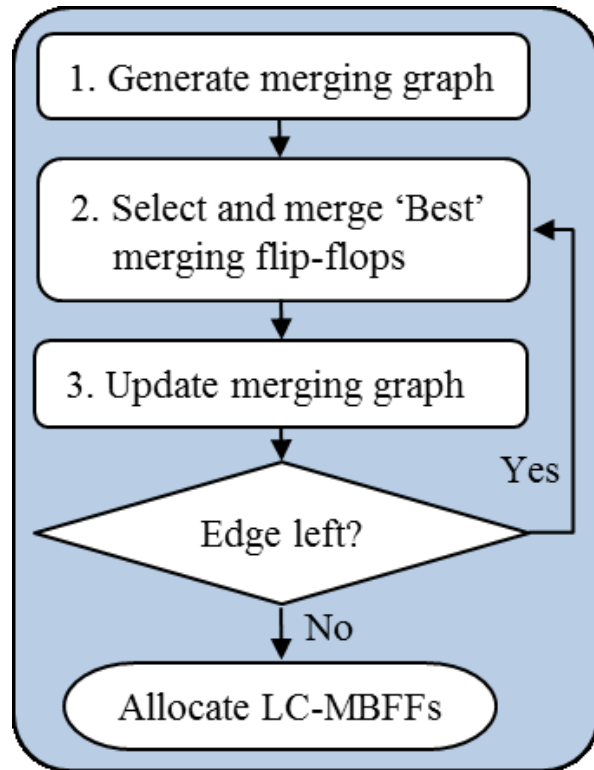
Example after the Final Update

1. Generate merging graph

2. Select and merge 'Best' merging flip-flops

3. Update merging graph

Edge left? — Yes

No

Allocate LC-MBFFs

$\overline{\langle f_4, f_7, f_8\rangle}$

$\langle f_7\rangle$

16

$\langle f_4\rangle$

Removed during update

$\overline{\langle f_1, f_2\rangle}$

$\langle f_2\rangle$

15

$\langle f_8\rangle$

19

$\langle f_1\rangle$

4th best merging

Removed during final update

$\langle f_6\rangle$

$\langle f_3\rangle$ 17

$\overline{\langle f_3, f_6\rangle}$

$\langle f_5\rangle$
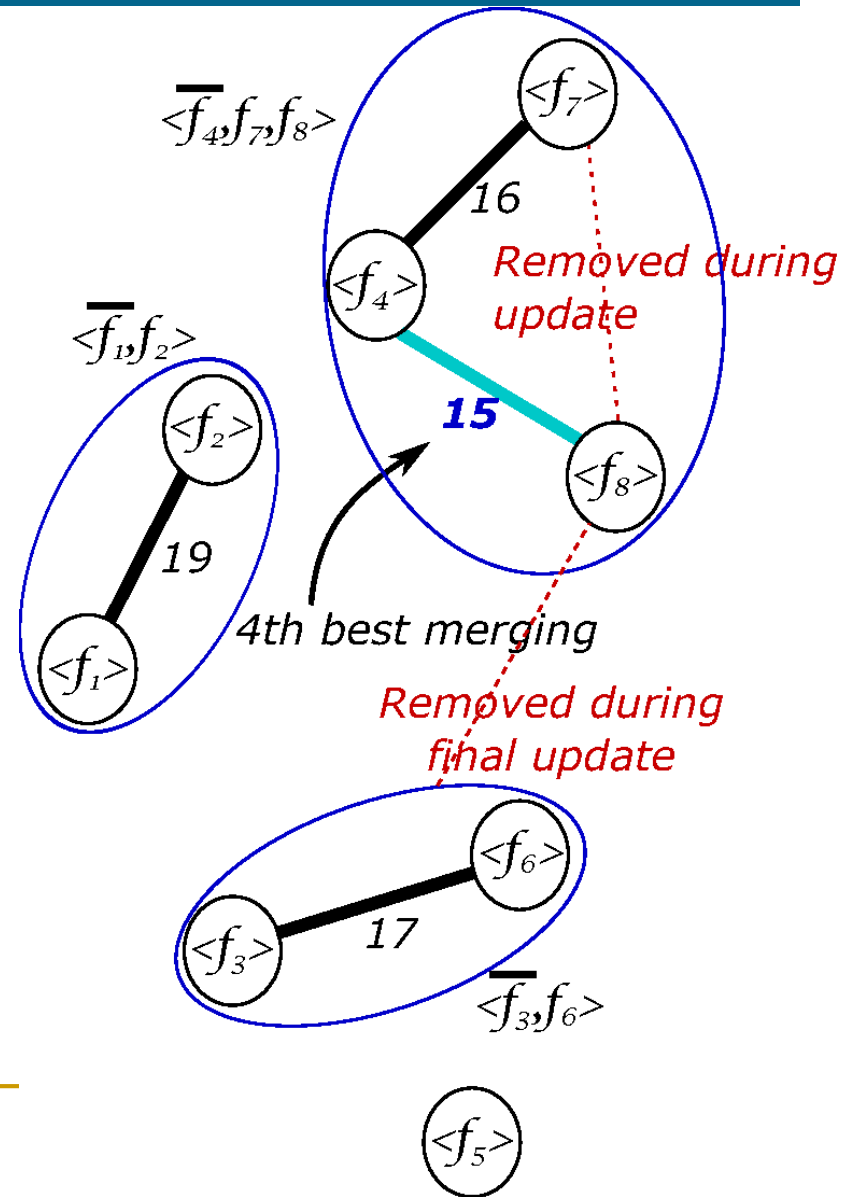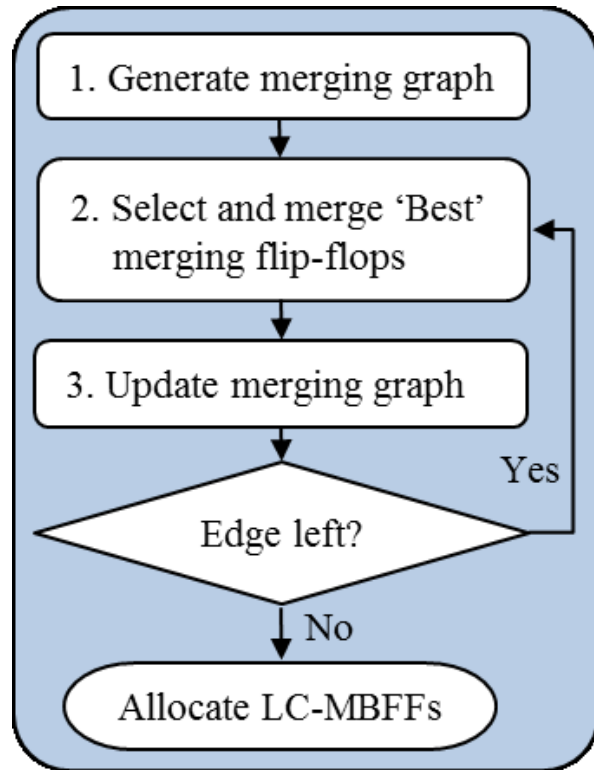
Experimental Setup

- **Algorithm implementation**
  - *C++* & *GCC* on *Intel* 64-bit 2.6GHz machine
- **Physical design environment**
  - *45nm Open Cell Library* & *PTM interconnect structure*
  - *Synopsys Design Compiler* & *IC Compiler*
- **11 *ISCAS89* & *IWLS2005* benchmark circuits**
  - 500MHz operating clock frequency
- D_{max}^{2-bit} **set as 30um and** D_{max}^{3-bit} **as 20um**

Experimental Results

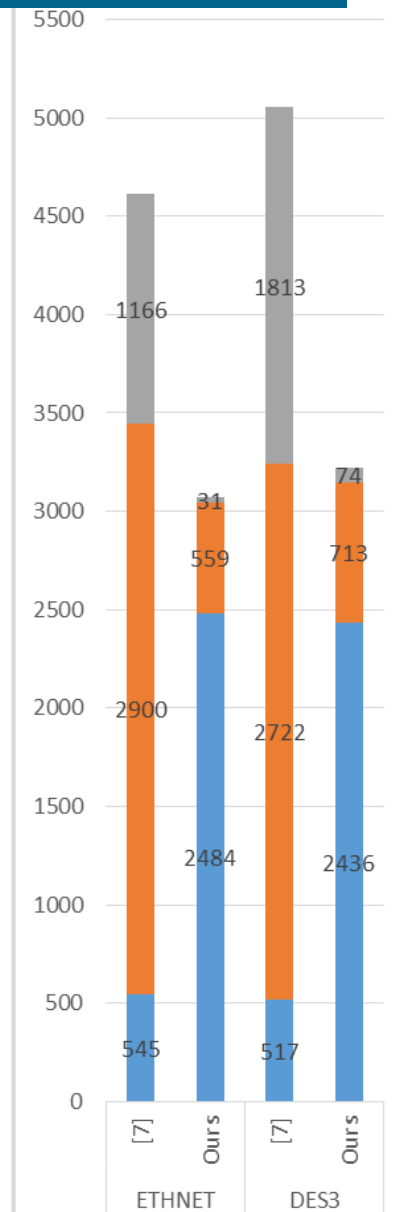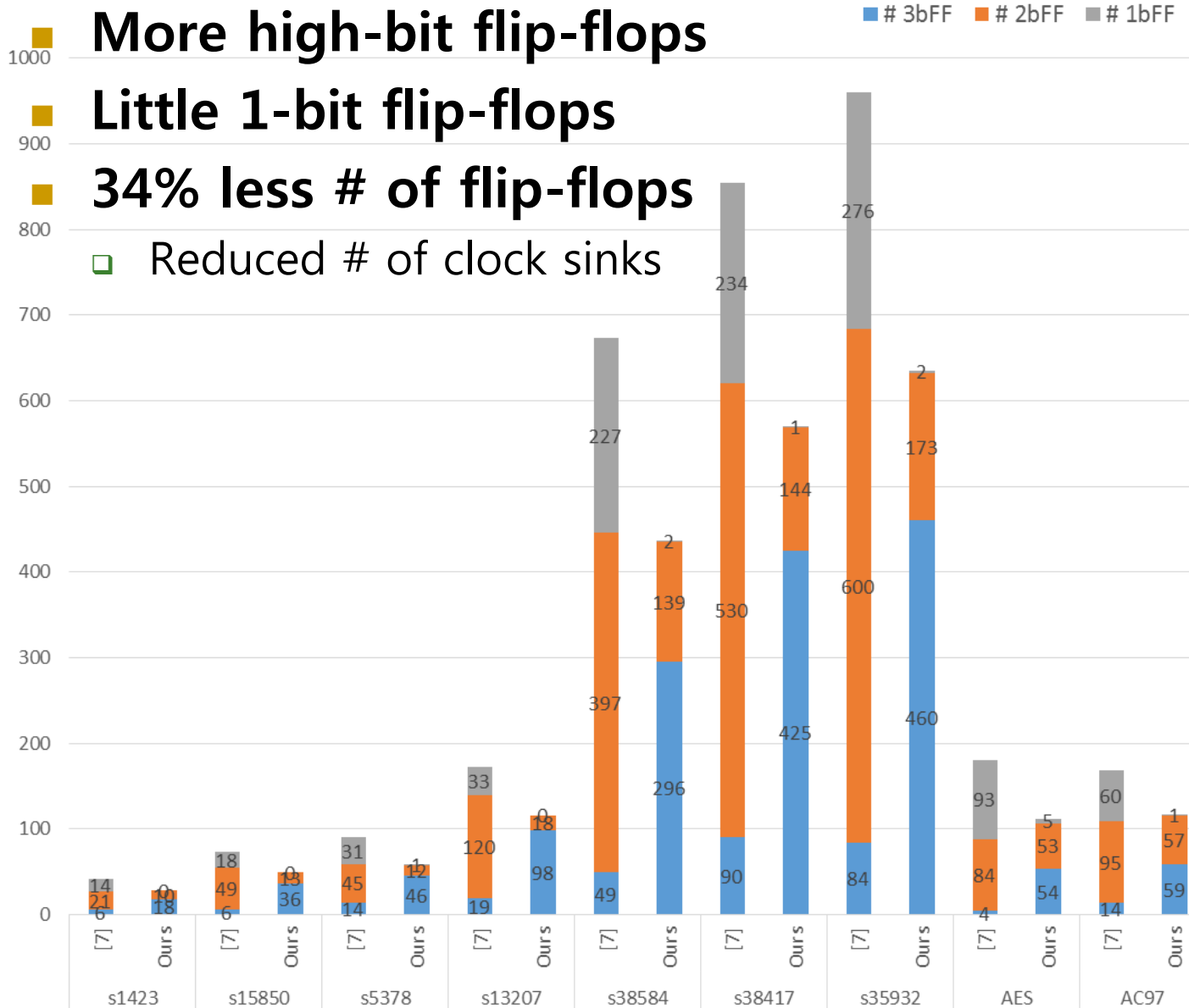| Circuit	# of 1-bit FFs	Chen-Yan [7]				Ours (LC-MBFF allocation)				Area Impact (%)	Runtime (sec.)
		# of 3-bit MBFFs	# of 2-bit MBFFs	# of 1-bit FFs		# of 3-bit LC-MBFFs	# of 2-bit LC-MBFFs	# of 1-bit FFs			
s1423	74	6	21	14		18	10	0		1.4%	0.01
s15850	134	6	49	18		36	13	0		0.9%	0.04
S5378	163	14	45	31		46	12	1		1.1%	0.04
S13207	330	19	120	33		98	18	0		1.1%	0.32
S38584	1168	49	397	227		296	139	2		1.2%	2.66
S38417	1564	90	530	234		425	144	1		1.2%	5.78
S35932	1728	84	600	276		460	173	2		1.4%	7.25
AES	273	4	84	93		54	53	5		0.3%	0.09
AC97	292	14	95	60		59	57	1		0.6%	0.27
ETHNET	8601	545	2900	1166		2484	559	31		0.3%	242.02
DES3	8808	517	2722	1813		2436	713	74		0.4%	143.97
										0.91%	

- **3.13% more power saving on average**
  - With less than 1% interconnect area

---

[7] Z.-W. Chen and J.-T. Yan, "Routability-constrained multi-bit flip-flop construction for clock power reduction," Integration, the VLSI Journal, Jun. 2013.

Distribution of LC-MBFFs after Allocation

- **More high-bit flip-flops**
- **Little 1-bit flip-flops**
- **34% less # of flip-flops**
  - Reduced # of clock sinks

Impact of Additional Wire Length

Circuit	Total Wire Length	Added Wire Length	Added WL /Total WL	Total Wire Area	Total Circuit Area	Impacted Area
s1423	5238	386	7.4%	200.12	1064	1.4%
s15850	8761	502	5.7%	237.01	1549	0.9%
s5378	12844	752	5.9%	415.13	2168	1.1%
s13207	17052	1288	7.6%	529.46	3648	1.1%
s38584	84200	5254	6.2%	3075.19	16266	1.2%
s38417	101362	6820	6.7%	3605.81	19710	1.2%
s35932	86014	7756	9.0%	3508.73	22049	1.4%
AES	32882	1636	5.0%	540.15	8893	0.3%
AC97	28321	1334	4.7%	878.29	6368	0.6%
ETHNET	2325506	38688	1.7%	21920.98	121539	0.3%
DES3	2351961	42130	1.8%	34495.96	160824	0.4%

- **With simplified clock tree, the actual area impact is much less**

Conclusion

- **Loosely-coupled Multi-bit Flip-flop**
  - New structure of multi-bit flip-flop
  - No timing/area constraints
- **LC-MBFF Allocation Algorithm**
  - Considering power, clock tree & routability
- **More clock power saving**
  - 3.13% more clock power saving than the existing work

Thank you for your attention!

A1. Merging Distance Limit

- D_{max}^{2-bit} set as 30um and D_{max}^{3-bit} as 20um