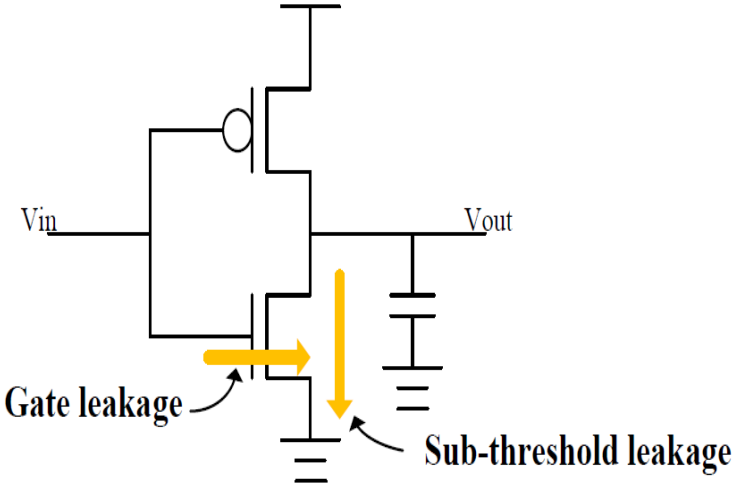
State Retention for Power Gated Circuits with Retention Registers

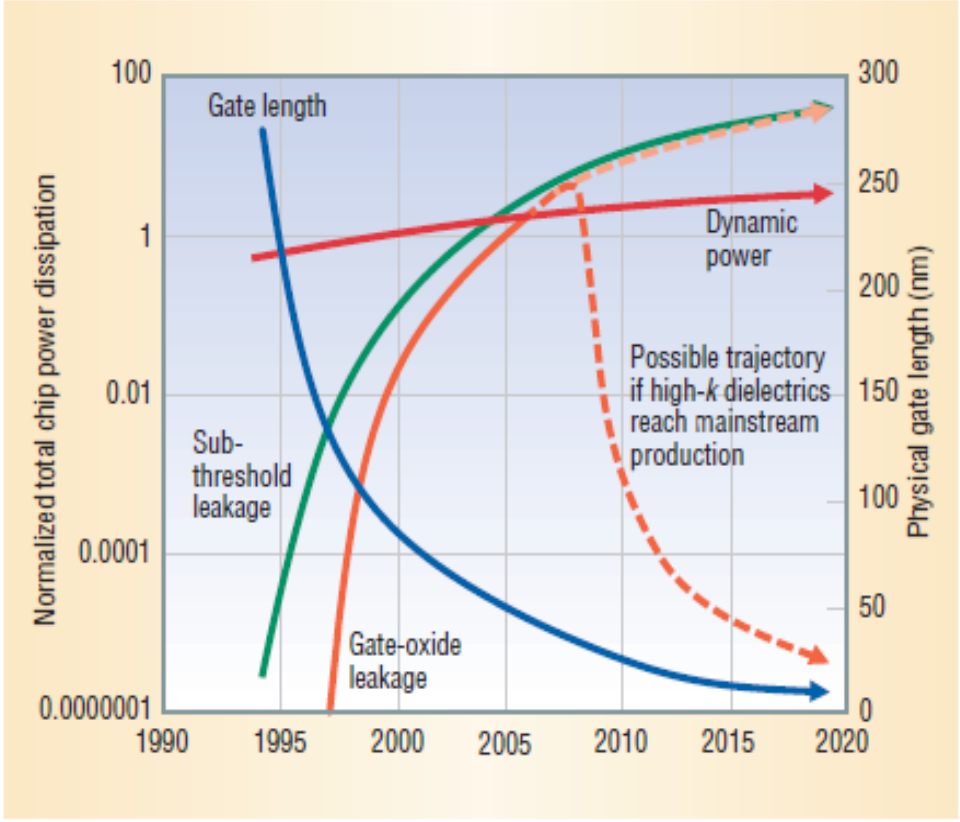
Mark Po-Hung Lin

Department of Electrical Engineering
National Chung Cheng University
Chiayi, Taiwan

The Growth of Leakage Power

- Dramatic increase of leakage power in nano-scale design.

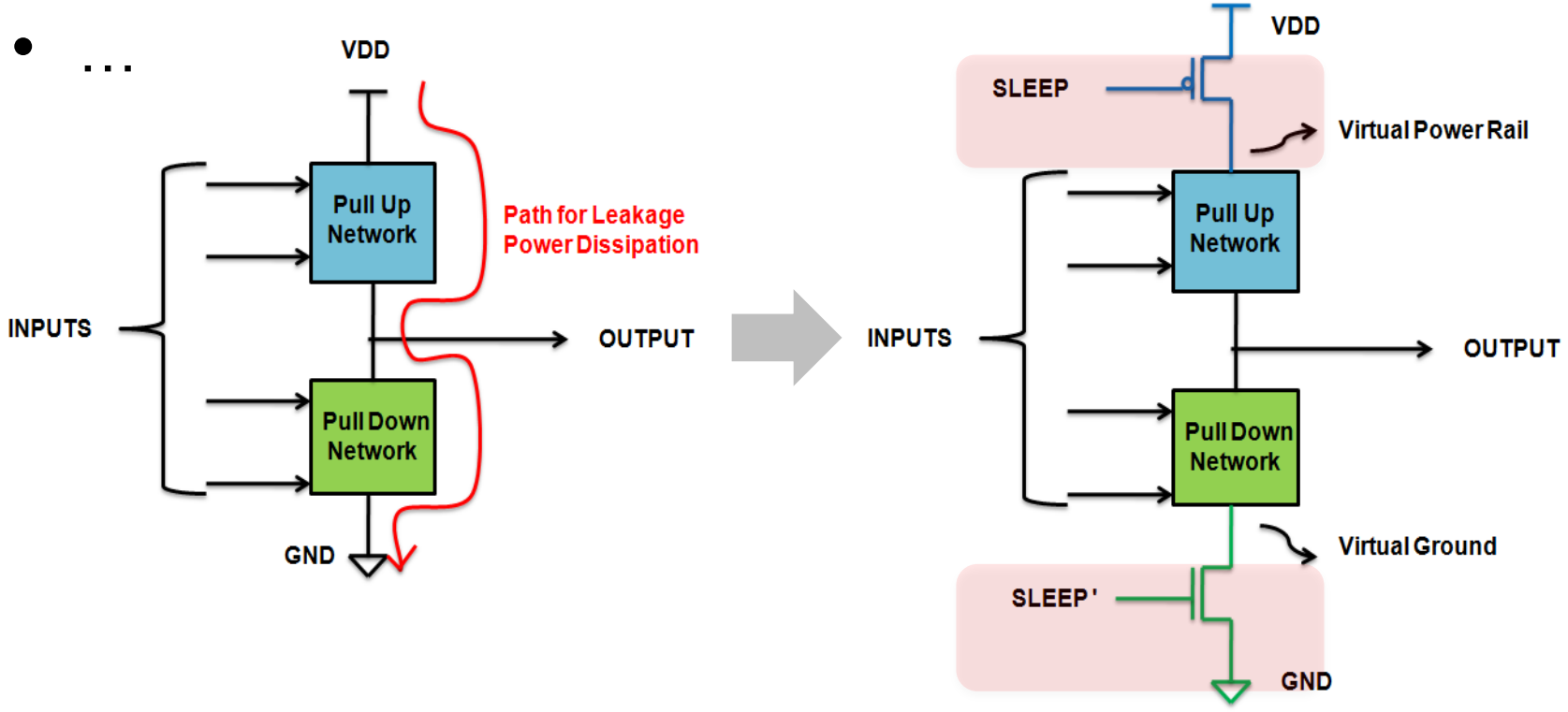




[Kim et al., IEEE Computer 2003]

Leakage Power Reduction

- Multi-Vt circuit optimization
- Reverse body biasing
- Power gating with sleep transistors
  - Surpress leakage power, but may lose flip-flop states
- …



Literature Review for State Retention

- Selective state retention
  - Darbari et al., “Selective state retention design using symbolic simulation”, **DATE-2009**
  - Greenberg et al., “Selective state retention power gating based on gate-level analysis”, **TCAS-I, 2014**
  - Greenberg et al., “Selective state retention power gating based on formal verification”, **TCAS-I, 2015**
  - Chiang et al., “Scalable sequence-constrained retention register minimization in power gating design”, **DAC-2015**
- State retention methods
  - Chen et al., “Efficient multiple-bit retention register assignment for power gated design: Concept and algorithms,”, **ICCAD-2012**
  - Chen et al., “Multi-bit retention registers for power gated designs: Concept, design, and deployment,” **TCAD, 2014**
  - Lin & Lin, "More effective power-gated circuit optimization with multi-bit retention registers," **ICCAD-2014**

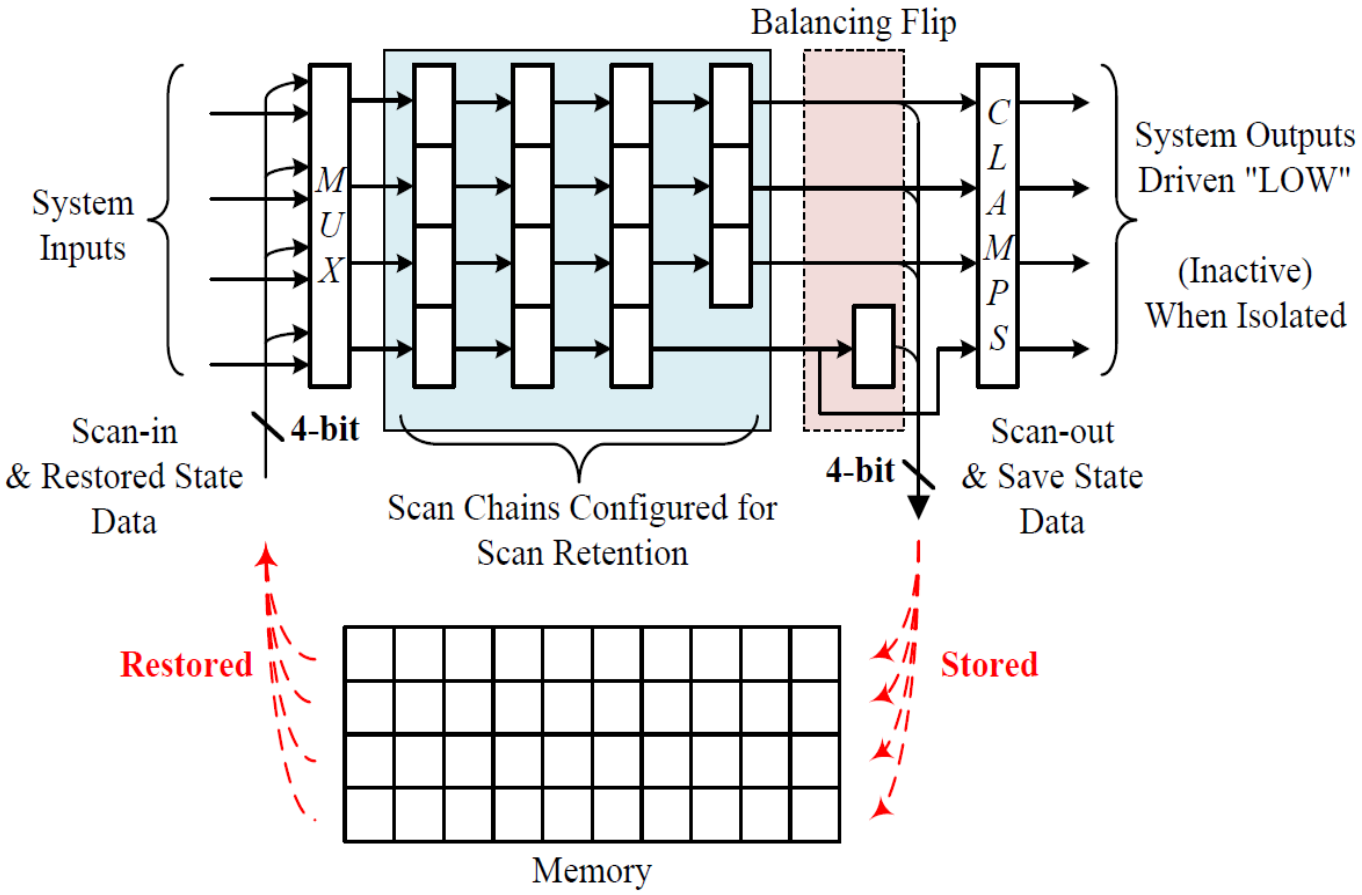
State Retention Approaches

- Memory-based approach
- Register/latch-based approaches
  - Single-bit retention registers
  - Uniform multi-bit retention registers/latches
    - [Chen et al., ICCAD'12 & TCAD'14]
    - [Lin et al., ICCAD'14]
  - Non-uniform multi-bit retention registers/latches
    - [Fan & Lin, ICCAD'17]

	Memory-based	SBRR	MBRR
Wakeup Latency			
Area Overhead			
Power Consumption			

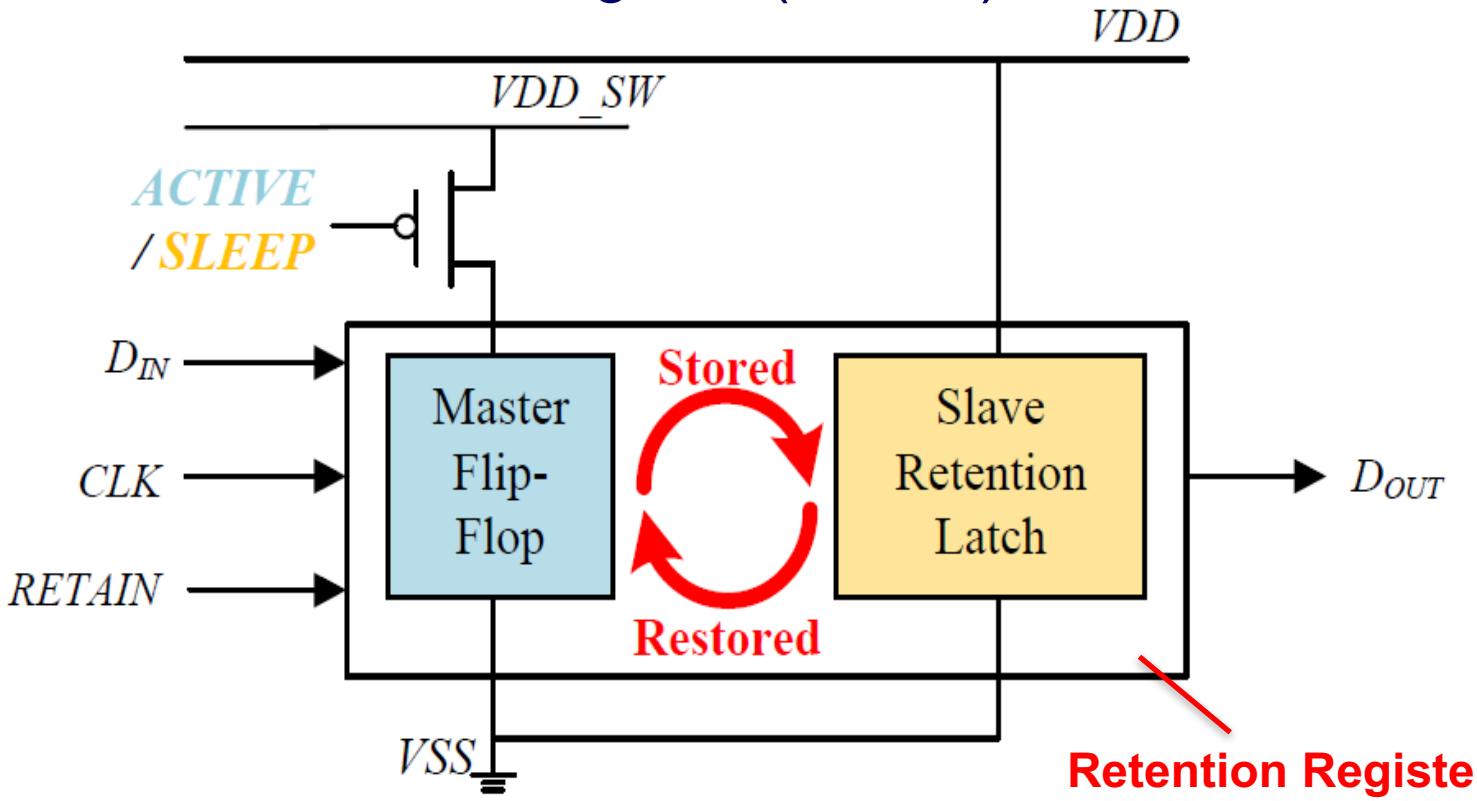
Memory-based Approach

- Scan out flip-flop states into memory ➔ sleeping
- Scan back flip-flops states from memory ➔ active



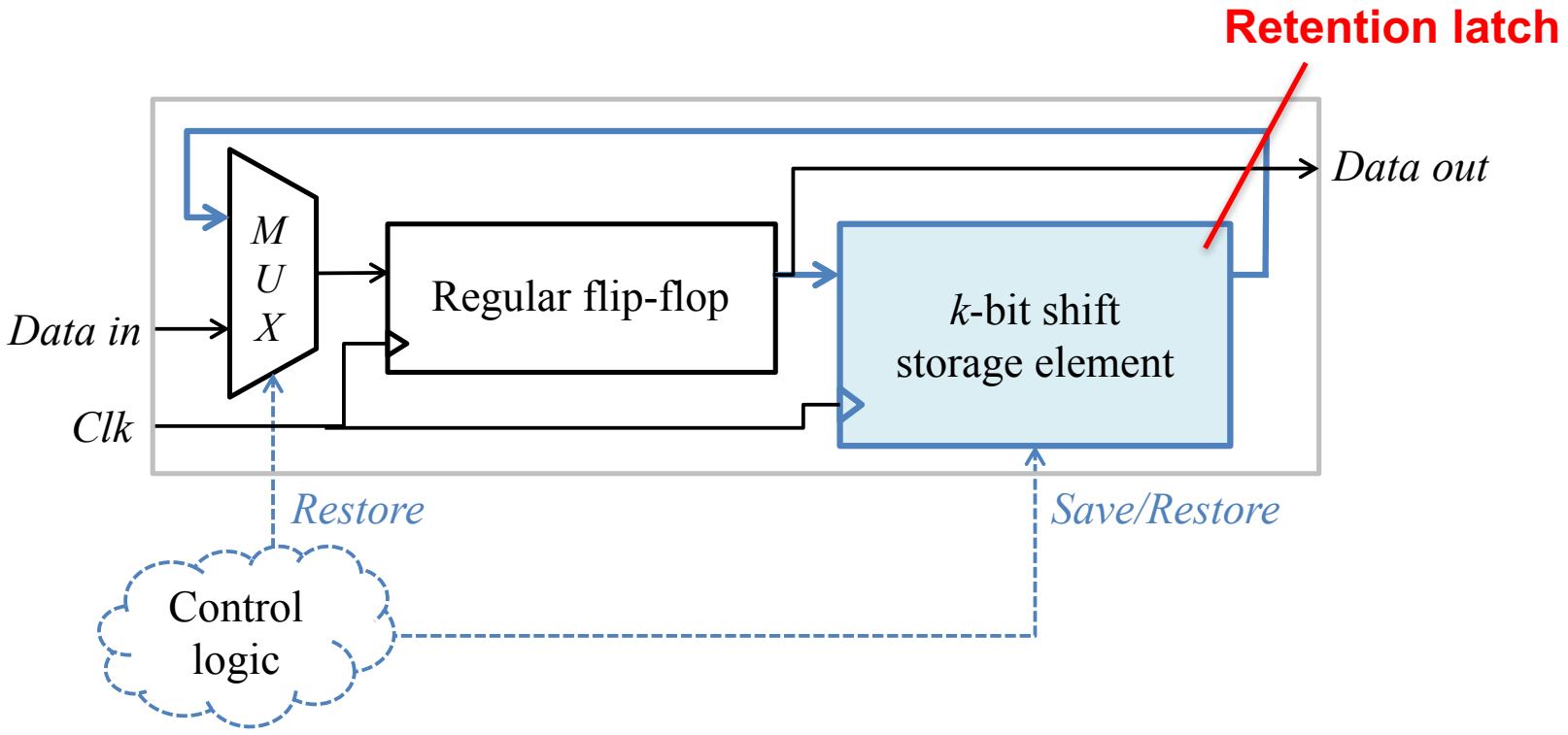
Register/Latch-based Approach

- Keep flip-flop states in retention latches
  - Single-bit retention register (SBRR)
  - Multi-bit retention register (MBRR)



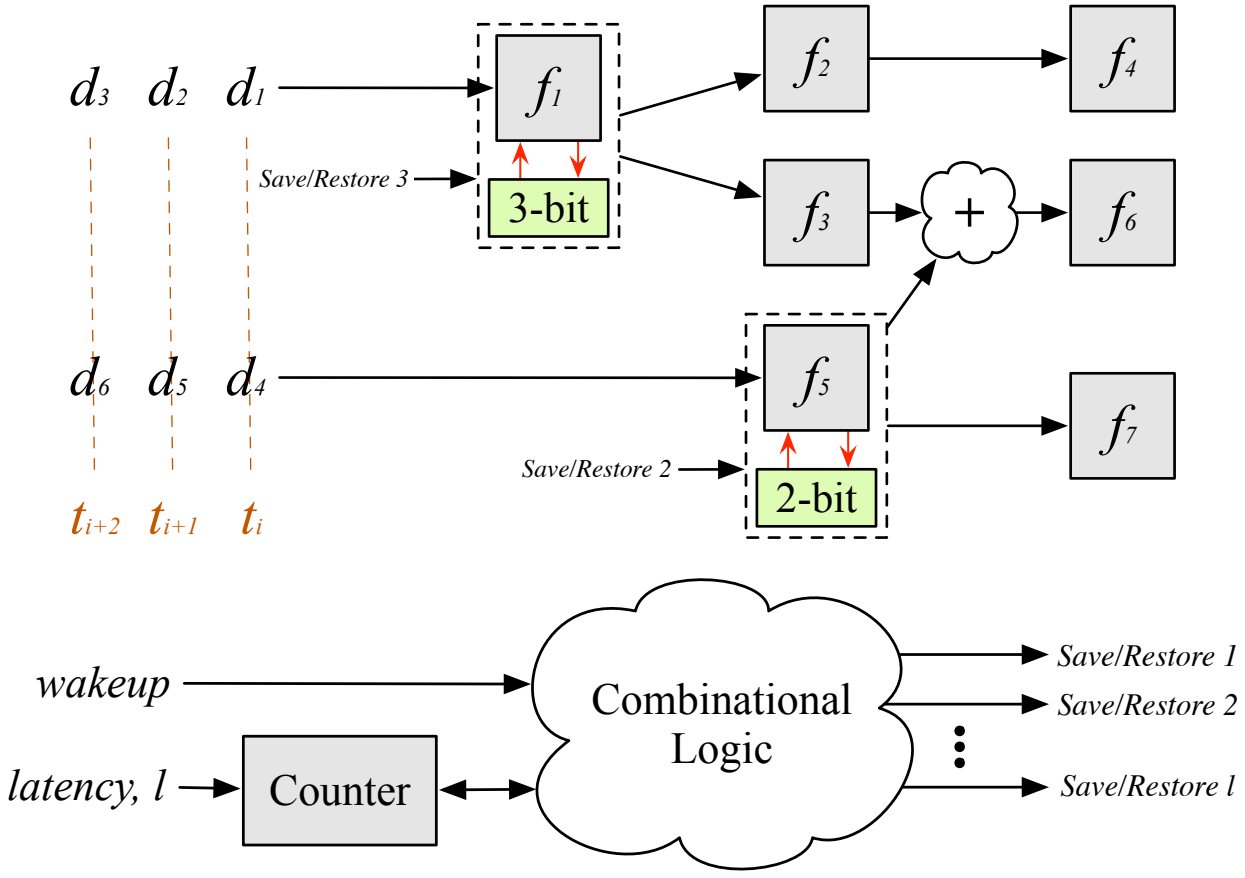
k-bit Retention Register

- Save/restore *k*-cycle flip-flop states during sleep/active mode switching



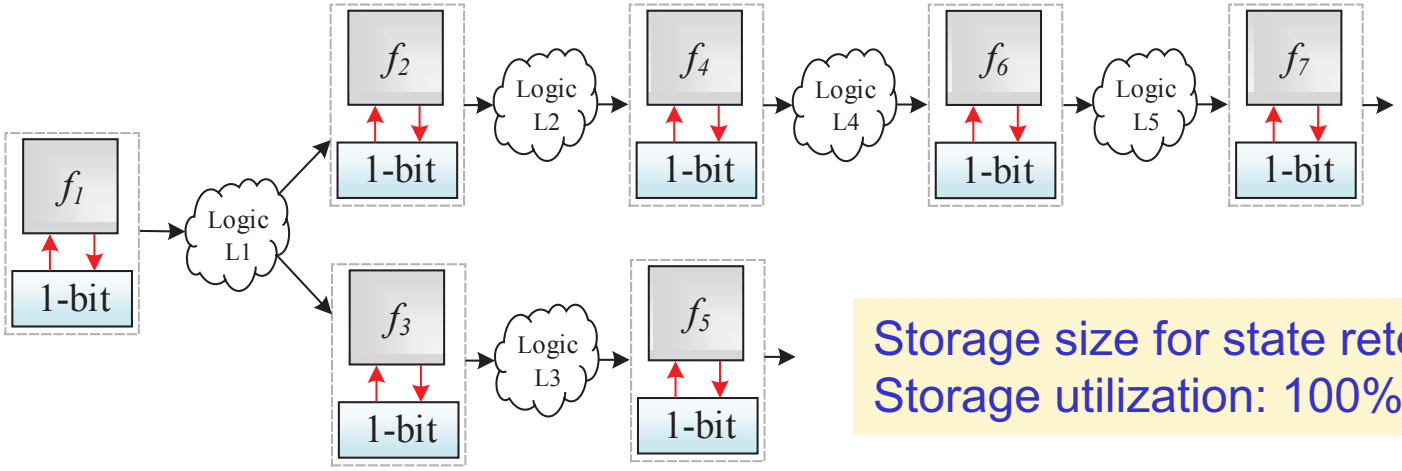
Non-uniform MBRR

- Different save/restore signals for different MBRRs with different bit numbers



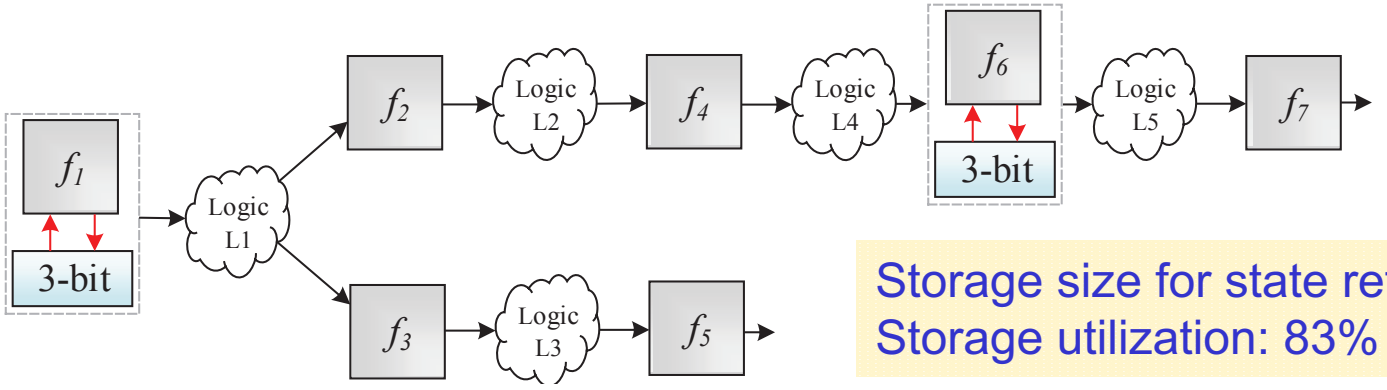
SBRR v.s. MBRR

- Single-bit retention register (SBRR)



Storage size for state retention: 7 bits
Storage utilization: 100%

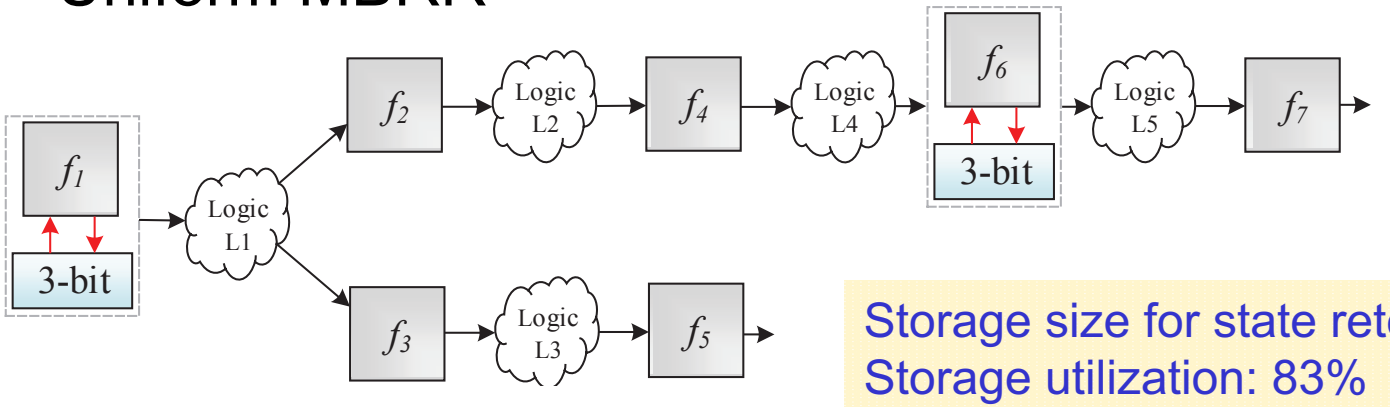
- Multi-bit retention register (MBRR)



Storage size for state retention: 6 bits
Storage utilization: 83%

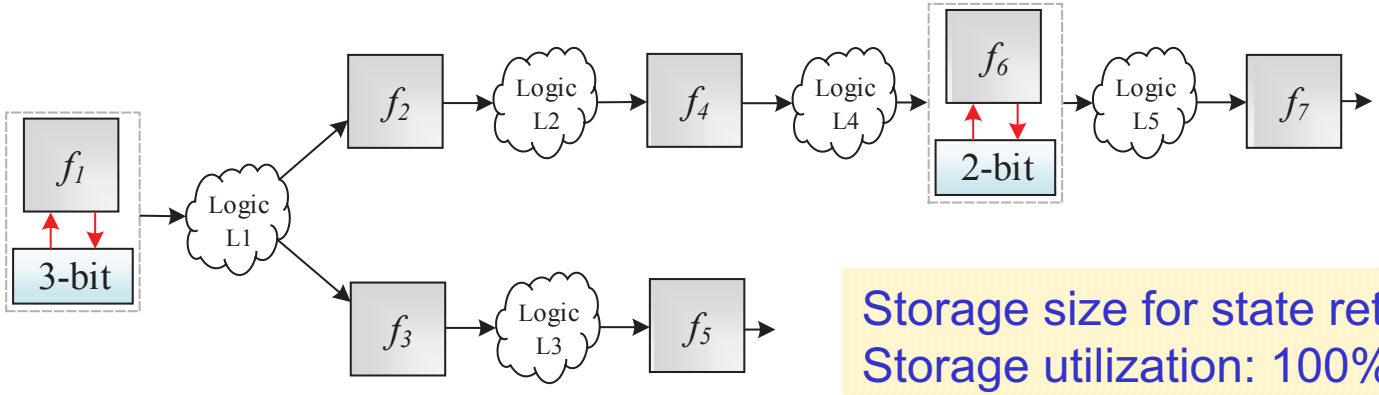
Uniform v.s. Non-Uniform MBRR

- Uniform MBRR



Storage size for state retention: 6 bits
Storage utilization: 83%

- Non-uniform MBRR



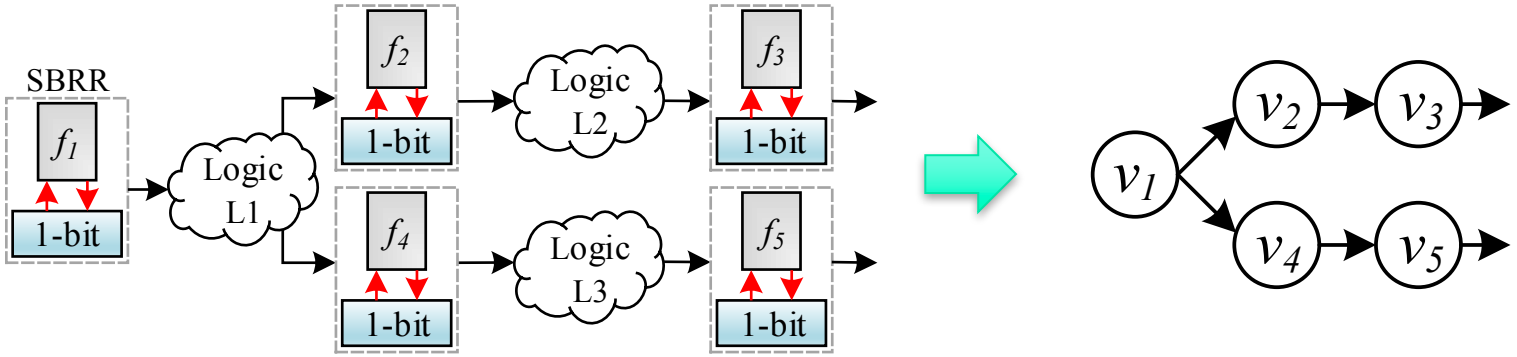
Storage size for state retention: 5 bits
Storage utilization: 100%

Problem 1: Uniform MBRR

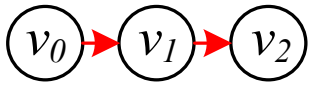
- Input:
  - A power-gated circuit, ***C***
  - A **fixed** mode transition latency of ***k*** clock cycles
- Objective:
  - Replace a minimum set of flip-flops in ***C*** with ***k*-bit retention registers** such that all flip-flop states in ***C*** can be retained while total storage size is minimized
- Solutions:
  - Greedy method **[Chen et al., ICCAD'12 & TCAD'14]**
  - Integer-linear-programming (ILP) method **[Lin & Lin, ICCAD'14]**

Terminologies

- Data dependency graph (DDG)

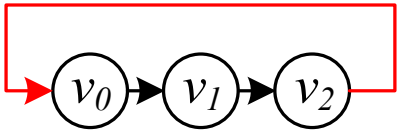


- Path length of *k*
  - A directed path between two vertices containing *k* directed edges

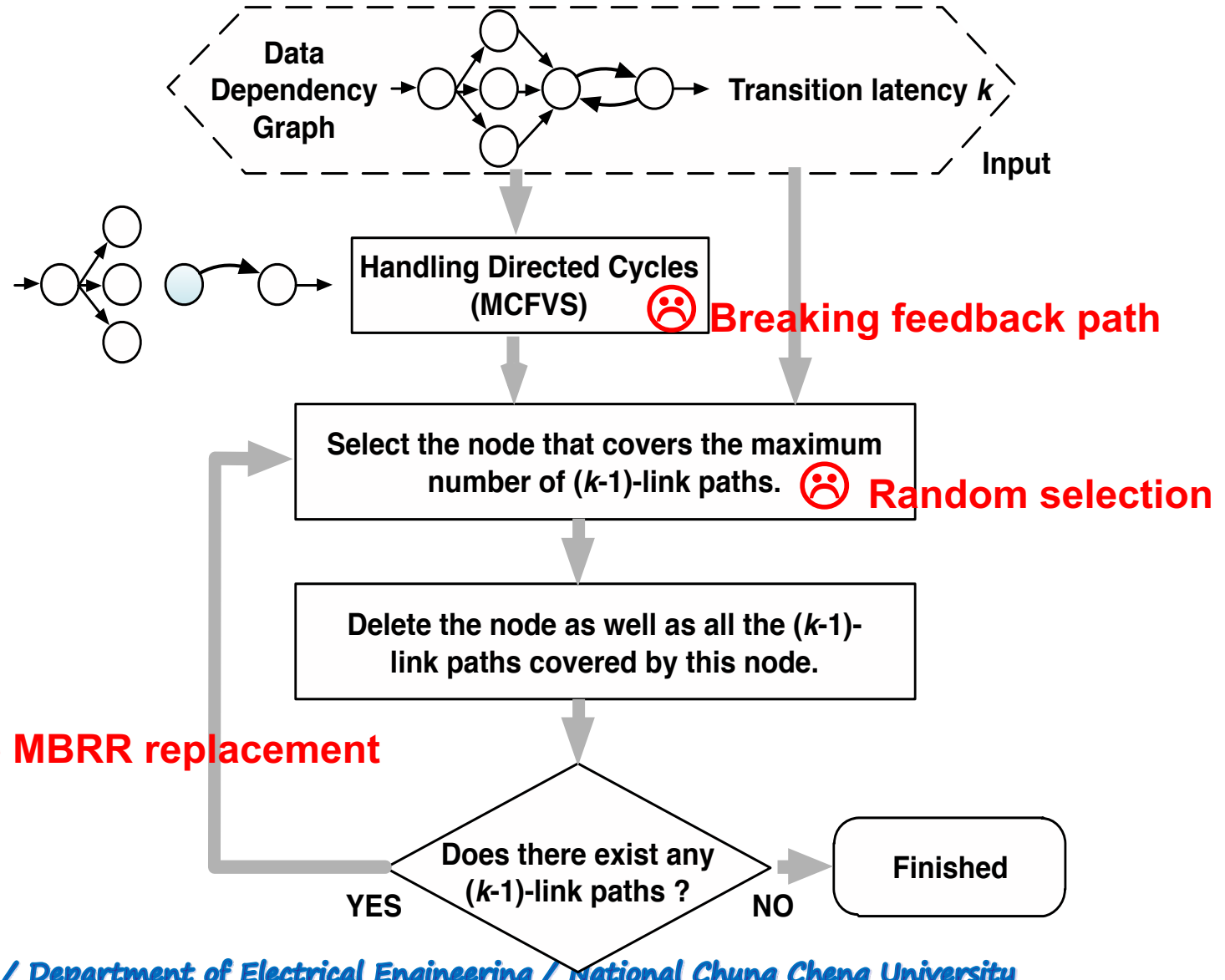


Path length of 2

- Feedback path
  - A path forming a directed cycle

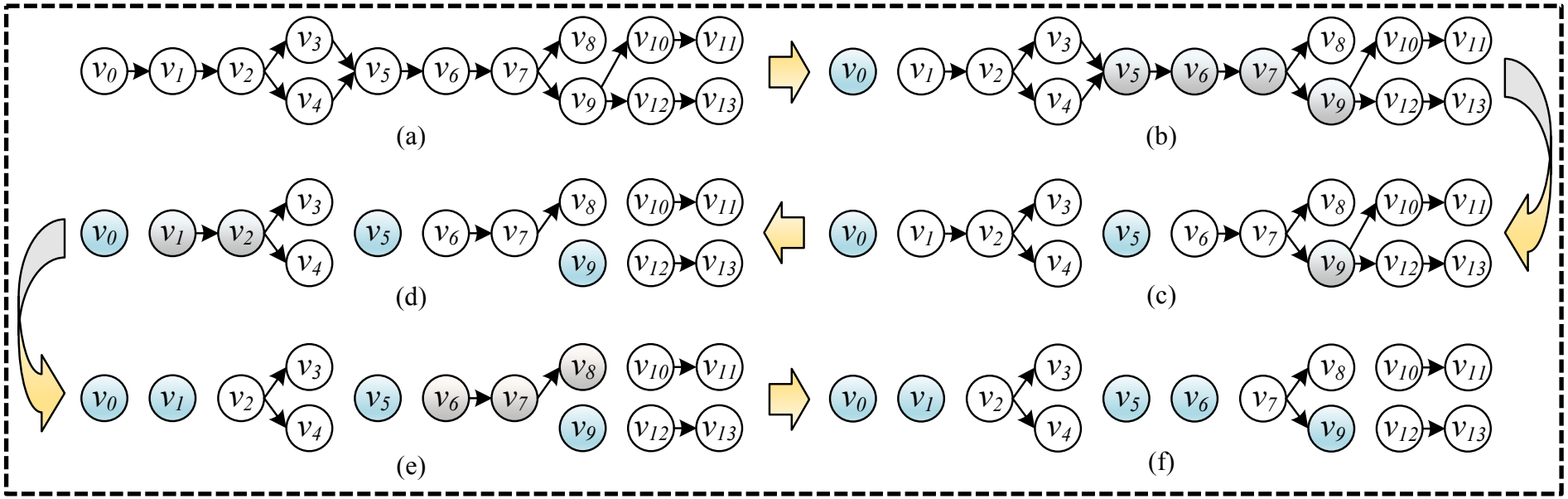


Chen et al.'s Heuristics [ICCAD'12 & TCAD'14]

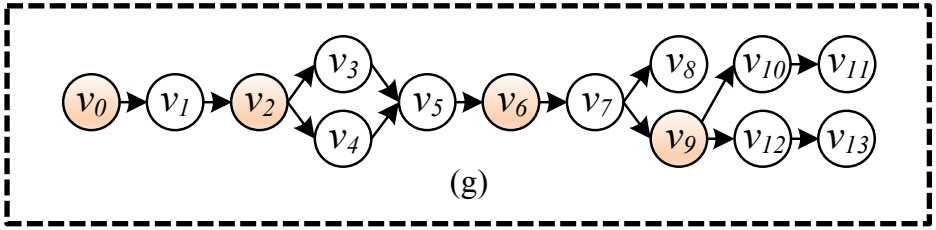


Iterative/Random MBRR Replacement

Chen et al.'s approach ⇨ **Five 3-bit retention registers**



Optimal Solution ⇨ **Four 3-bit retention registers**

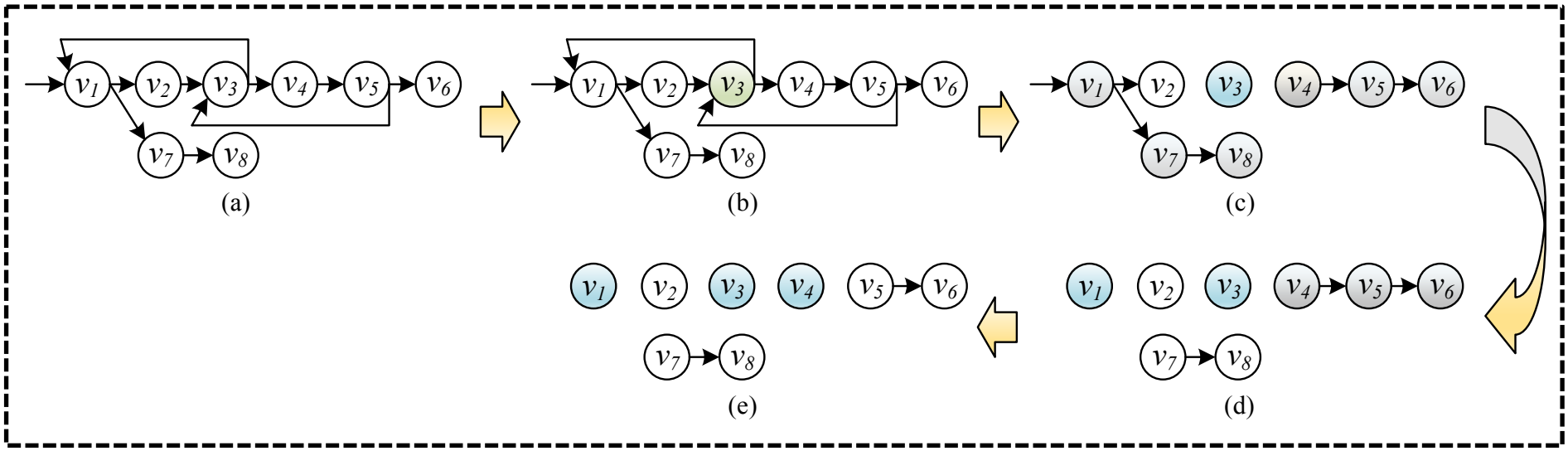


Vertex covering maximum paths

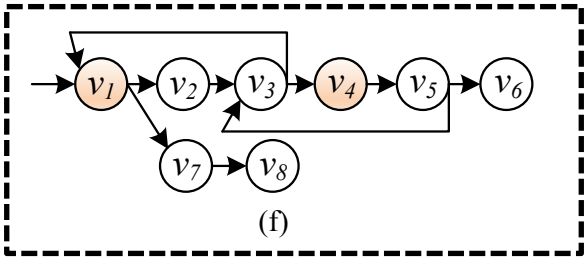
Selected Vertex

Breaking Feedback Path

Chen et al.'s approach ⇨ **Three 3-bit retention registers**



Optimal Solution ⇨ **Two 3-bit retention registers**

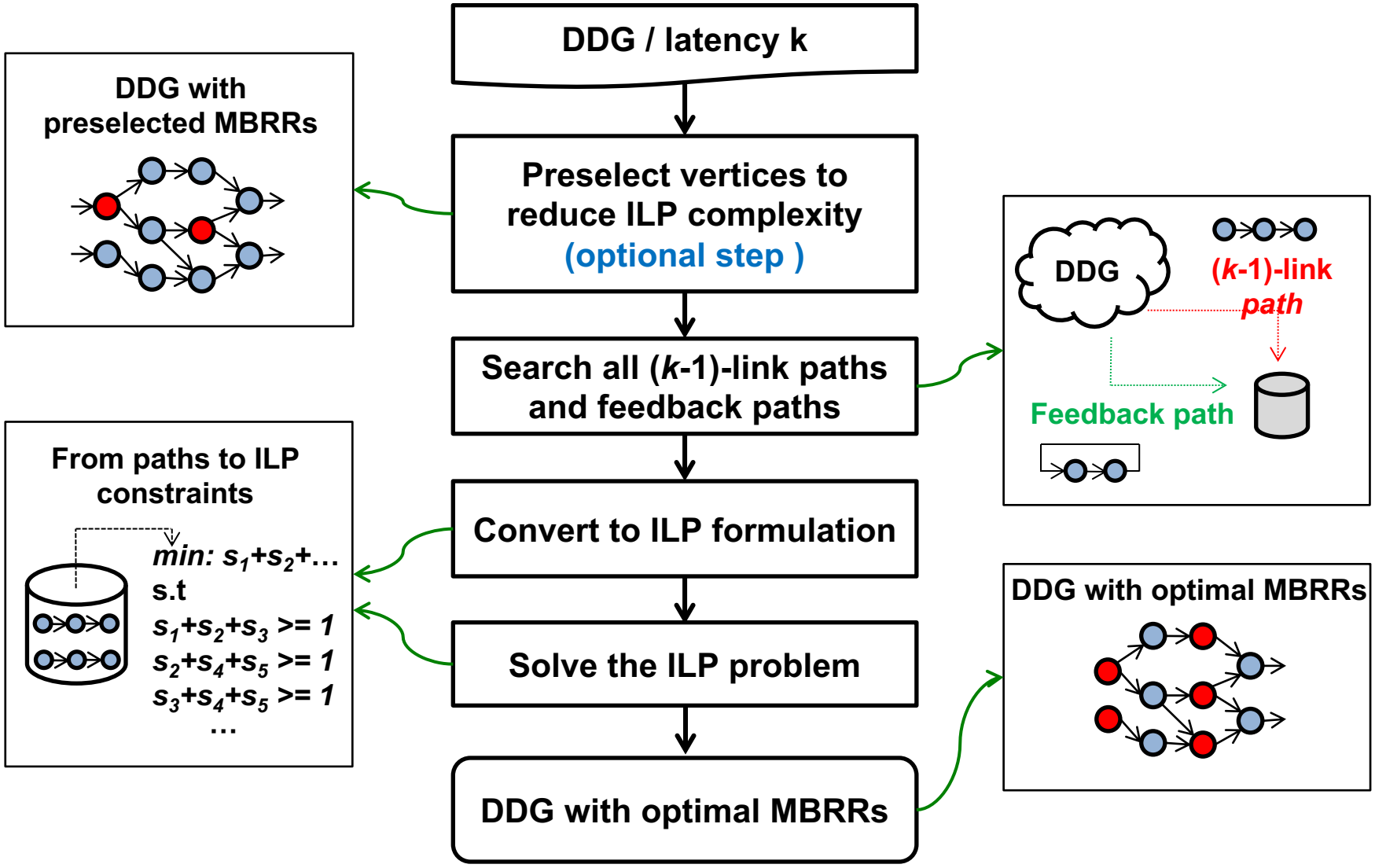


Vertex for breaking feedback paths

Vertex covering maximum paths

Selected vertex

The Proposed ILP-based Approach



ILP Formulation

- Minimize # of flip-flops which need to be replaced with MBRRs for state retention
  - **Vertex selection problem in a DDG**
- Objective:

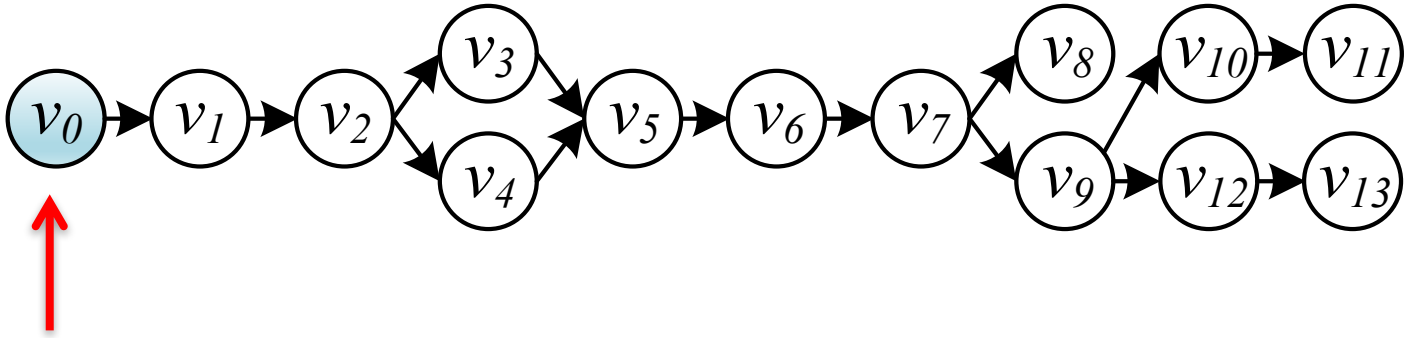
$$Minimize : \sum_{i=1}^{|V|} s_i, \quad \forall s_i \in \{0, 1\}.$$

  - |V| denotes the number of vertices in a DDG.
  - s_i denotes whether a vertex will be replaced with an MBRR.
- Constraints:
  - Zero-indegree constraint
  - Path-length constraint
  - Cycle constraint

Zero-indegree Constraint

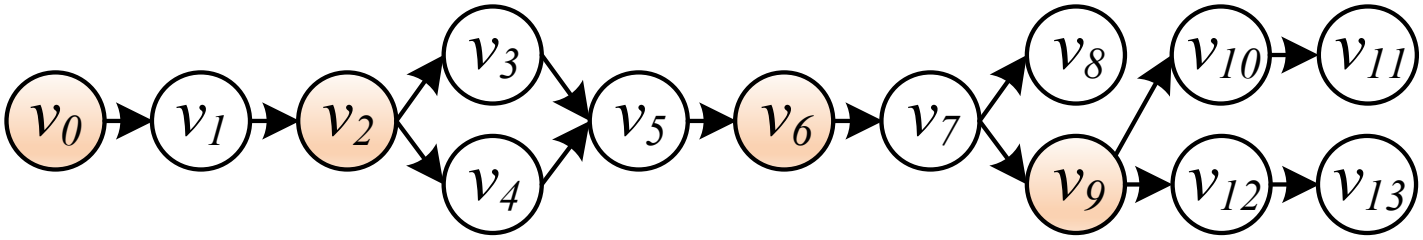
- A vertex in a DDG, which has no indegree, must be selected.

$$s_i = 1, \quad \forall v_i \in V \ \cap \ deg^-(v_i) = 0.$$



Path-length Constraint

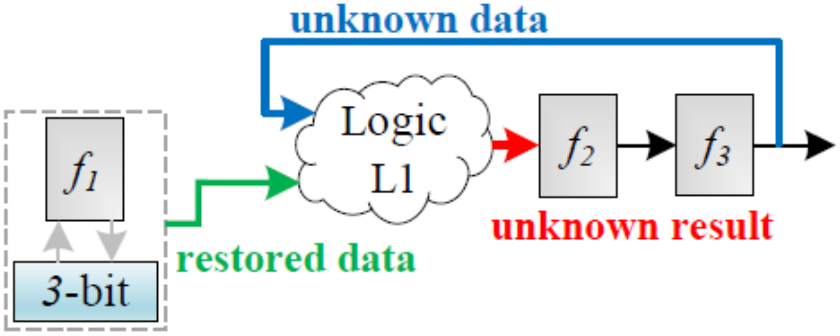
- Any directed path, p_i, in a DDG whose length is equal to k-1 must contain at least one selected vertex.
  - Example:



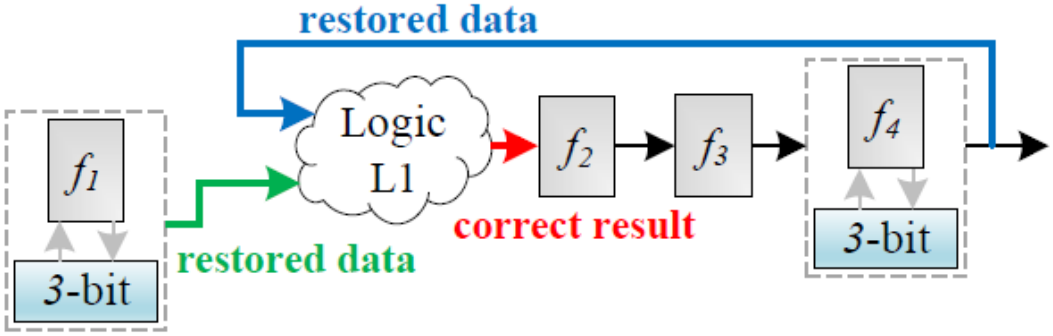
  - Each 2-link path contains at least one selected vertex.

Cycle Constraint

- Every feedback path must contain at least one *k*-bit retention register.
  - A feedback path without MBRRs

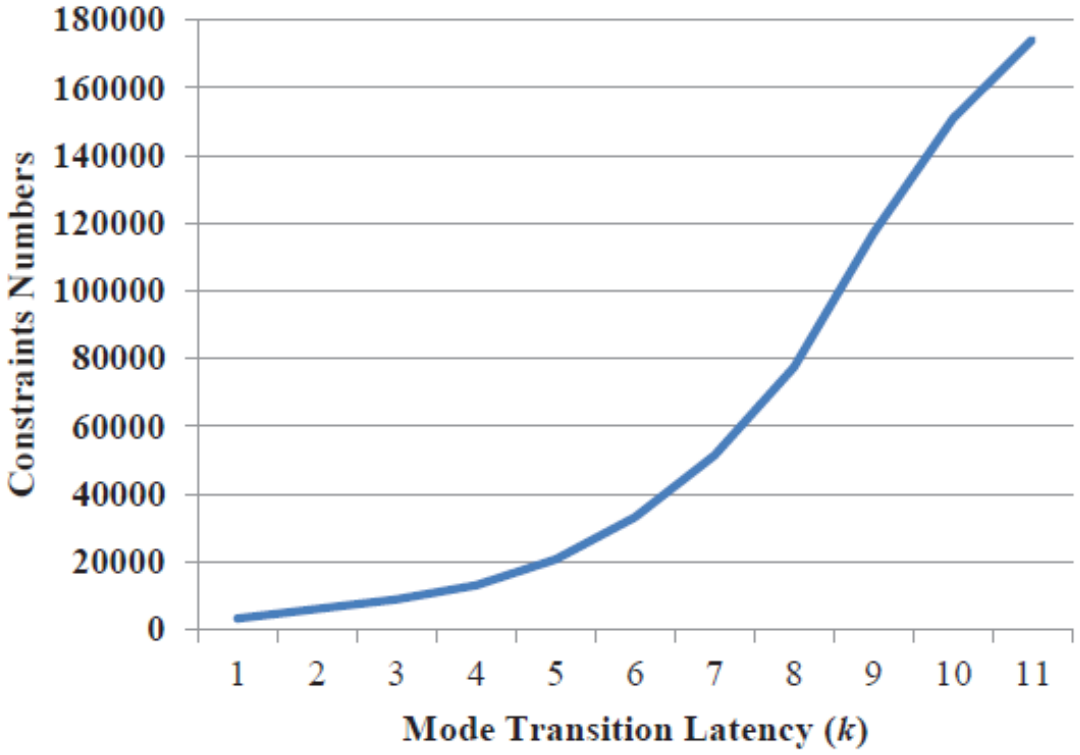


  - A feedback path with an MBRR



ILP Complexity Issue

- The path-length constraints will exponentially grow with respect to the number of flip-flops and mode transition latency.



ILP Constraint Reduction

- Perform greedy selection before ILP
  1) Iterative select a vertex which is contained in the most constraints
  2) Remove the corresponding constraints to gain the most constraints reduction
  3) Until the number of constraints is acceptable

Problem 2: Non-Uniform MBRR

- Input:
  - A power-gated circuit, ***C***
  - A **maximum** mode transition latency of l clock cycles
- Objective:
  - Replace a minimum set of flip-flops in ***C*** with MBRRs **ranging from 1 to l bits** such that all flip-flop states in ***C*** can be retained while total storage size is minimized with **100% storage utilization**
- Solutions
  - Greedy method **[Fan & Lin, ICCAD'17]**
  - ILP-based method **[Fan & Lin, ICCAD'17]**

Greedy Method

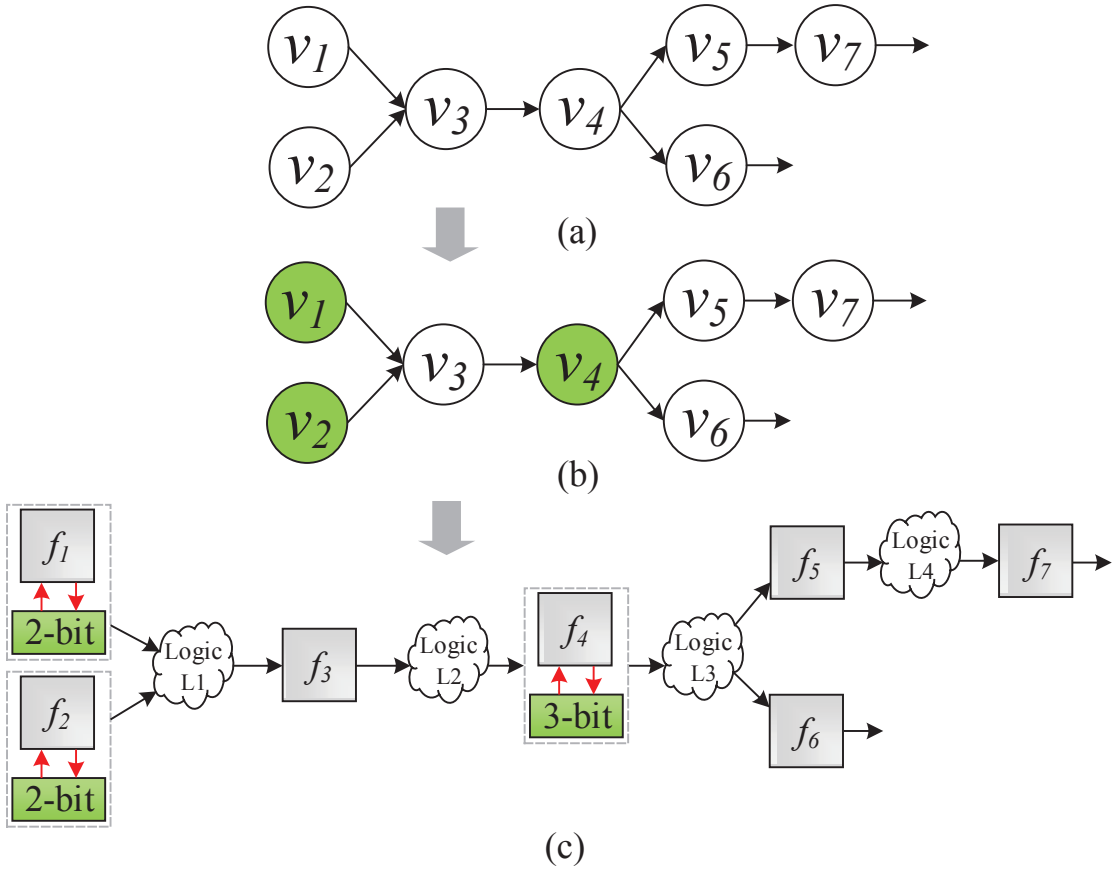
power-gated circuit, ***C***
max mode transition latency, ***l***

Construct DDG

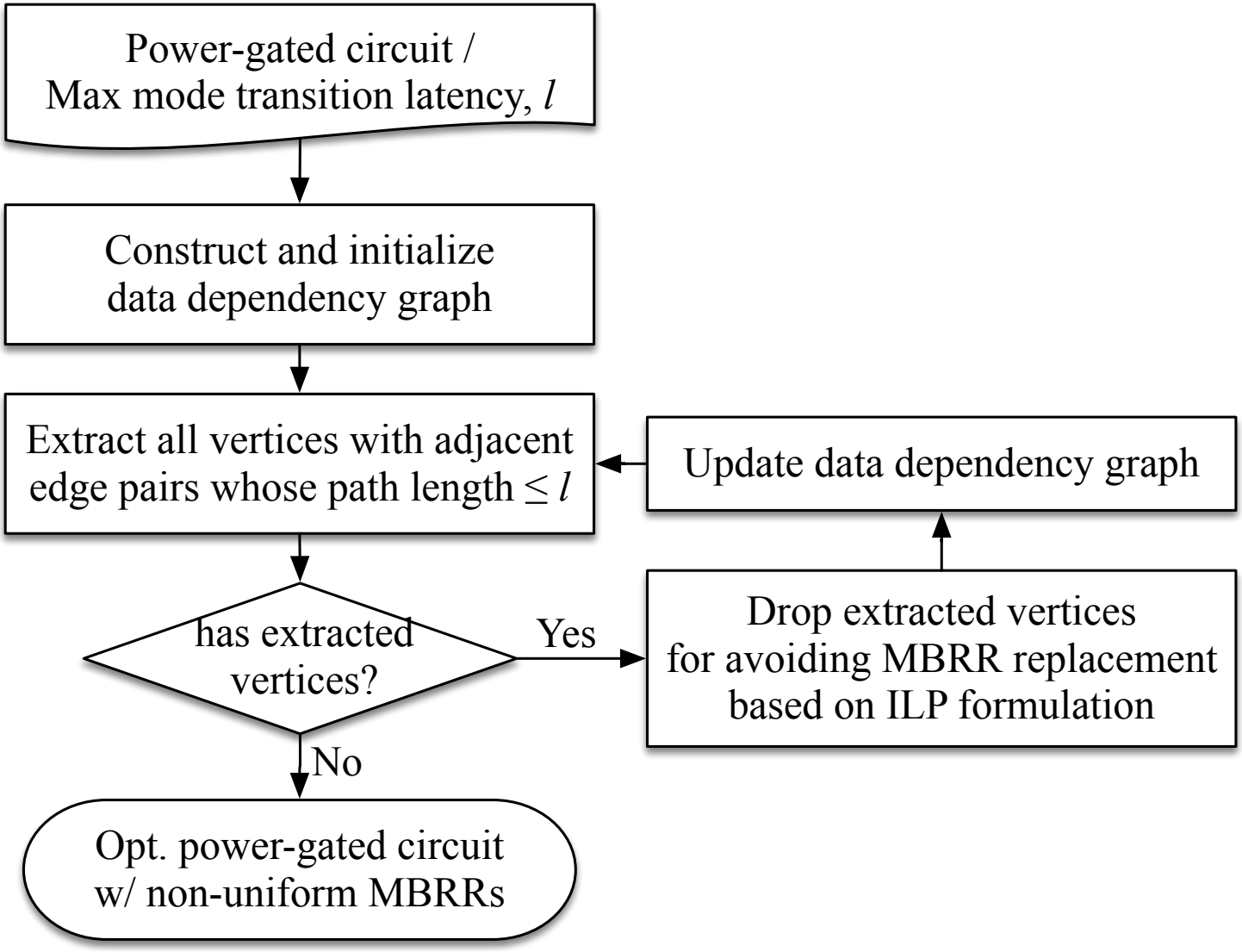
Perform ILP-based method w/ uniform ***l***-bit retention registers [Lin & Lin, ICCAD'14]

Remove unused bits in each MBRR

power-gated circuit w/ non-uniform MBRRs

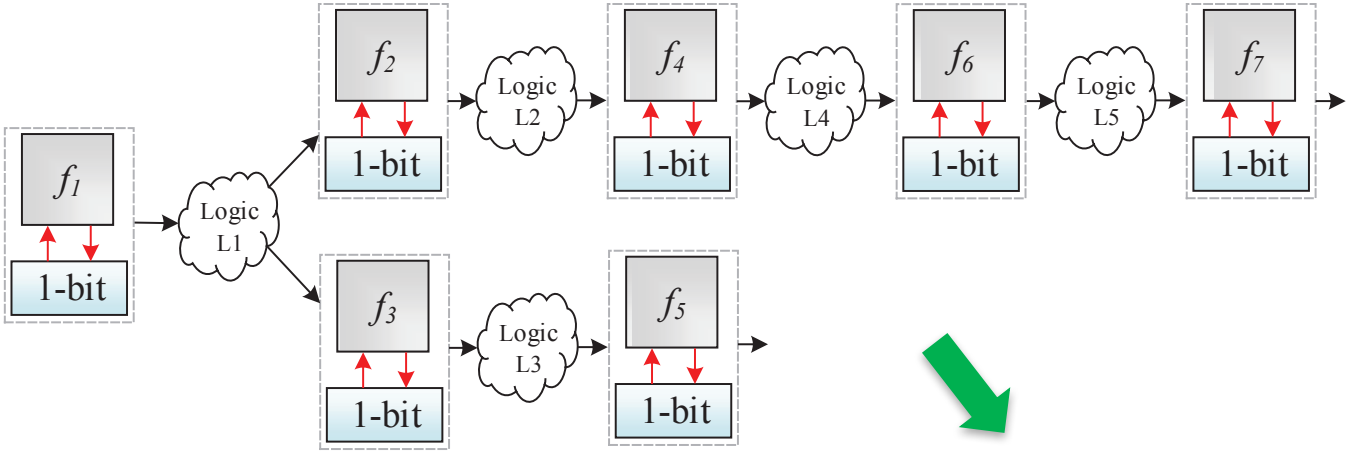


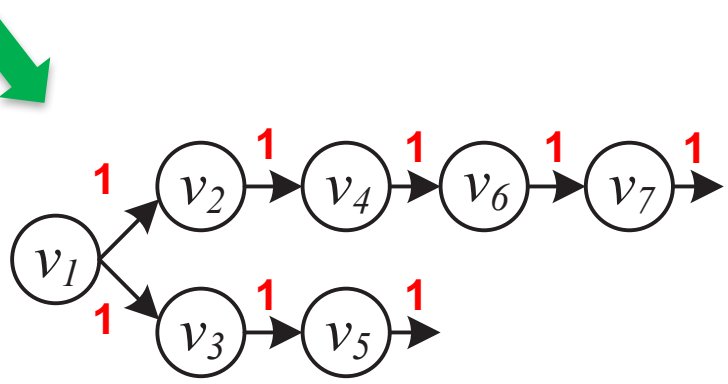
ILP-based Method



DDG Initialization

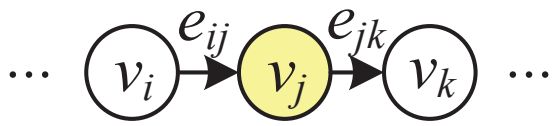
- Assign all edge weights in DDG equal to 1, which assumes that all flip-flops are replaced with 1-bit retention registers





Vertex Extraction from DDG

- Extract each vertex in DDG if the summation of its connected edge weights is less than ***l***
  - v_j is extracted if $w_{e_{ij}} + w_{e_{jk}} \leq l$
  - V^e contains all extracted vertices
  - E^e contains all edges connected to the extracted vertices



- Drop a set of vertices in V^e, which are not going to be replaced with MBRRs

ILP Formulation

- Objective: Drop a set of vertices in V^e, while minimizing the weighted summation of
  - Total storage size
  - Total number of MBRRs

$$\min \sum_{i=1}^{|V^e|} s_i + \alpha \sum_{i=1}^{|V^e|} x_i, \quad \forall x_i \in \{0,1\}.$$

  - s_i represents the expected weight of v_i, which corresponds to the expected storage size of the respective MBRR, after dropping a set of vertices in V^e
  - x_i is a 0–1 variable, which determines whether v_i will be selected or dropped.

ILP Formulation

- Fixed selection constraints
  - A vertex in DDG, which has no in-degree, must be selected (i.e. must not be dropped).

$$x_i = 1, \quad \forall (v_i \in V^e) \cap \left(deg^-(v_i) = 0\right).$$
$$x_j = 1, \quad \forall v_j \in V^R.$$

- Alternative selection constraints
  - For each edge in E^e, at least one of its connected vertices must be selected.

x_i	x_j	s_i	s_j
0	0	(forbidden case)	
0	1	0	w_{v_j}
1	0	$w_{v_i} + w_{v_j}$	0
1	1	w_{v_i}	w_{v_j}

$$x_i + x_j \geq 1, \quad \forall e_{ij} \in E^e.$$
$$s_i \geq x_i w_{v_i} + (1 - x_j) w_{v_j}, \quad \forall e_{ij} \in E^e.$$
$$s_j \geq x_j w_{v_j}, \quad \forall e_{ij} \in E^e.$$

DDG Update after Solving ILP

- After solving the ILP problem, a set of vertices in DDG will be dropped.

- Update DDG with new vertex and edge weights

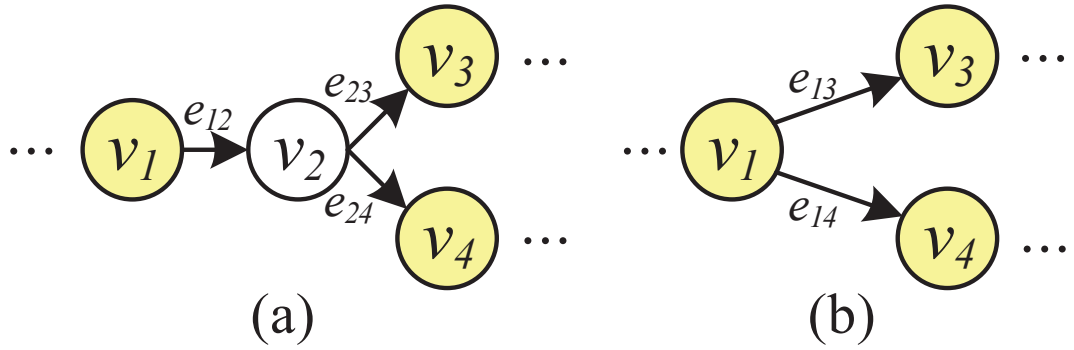


Fig. 9. A partial data dependency graph (a) before and (b) after dropping the vertex, v_2, where the new edge weights, $w_{e13} = w_{e12} + w_{e23}$ and $w_{e14} = w_{e12} + w_{e24}$, and the new vertex weight, $w_{v1} = \max\{w_{e13}, w_{e14}\}$.

Experimental Setup

- Programming language: C/C++
- Platform: 3.4GHz Intel CPU, 16GB memory
- ILP solver: Gurobi Optimizer
- Benchmark Circuits: [Chen et al., TCAD'14]

Circuit	# vertices	# edges	# directed cycles	latency
idtc	48	50	0	11
usb_phy	98	98	2	7
dist	128	174	0	11
DMA	2192	3138	8	5
pci_pridge	3559	6538	11	7
b19	6594	16612	19	8
des_perf	8802	12387	0	11
ECC decoder	6589	49759	76	6

SBRR v.s. MBRR [Chen et al. TCAD'14]

- Power consumption

circuit name	active mode(μW)									sleep mode(nW)		
	original			*SBRR*			*MBRR*					
	total	/	register	total	/	register	total	(reduction)	/ register	(reduction)	*SBRR*	*MBRR*	(reduction)
idtc	208.6	/	68.4	208.8	/	68.6	208.7	(0.07%)	/ 68.4	(0.21%)	478.6	51.5	(89.2%)
usb_phy	93.4	/	84.1	93.6	/	84.3	93.4	(0.19%)	/ 84.1	(0.21%)	977.1	208.9	(78.6%)
dist	558.5	/	182.4	558.9	/	182.9	558.6	(0.06%)	/ 182.5	(0.20%)	1276.2	257.5	(79.8%)
DMA	982.2	/	732.1	983.9	/	733.8	982.2	(0.17%)	/ 732.2	(0.23%)	21854.2	607.2	(97.2%)
pic_bridge	1680.2	/	1549.6	1683.9	/	1553.2	1680.3	(0.21%)	/ 1549.6	(0.23%)	33489.2	1183.5	(96.5%)
b19	1673.5	/	1274.1	1676.5	/	1277.1	1673.6	(0.18%)	/ 1274.1	(0.23%)	65742.2	2172.2	(96.7%)
des_perf	12839.6	/	10035.5	12863.2	/	10059.1	12847.9	(0.12%)	/ 10043.8	(0.15%)	87755.9	37698.0	(57.0%)
ECC decoder	82345.1	/	6696.2	82360.9	/	6712.0	82345.5	(0.02%)	/ 6696.6	(0.23%)	65692.3	17614.1	(73.2%)
average	—			—			—	(0.13%)	/ —	(0.21%)	—	—	(83.5%)

- Chip area

circuit name	original	*SBRR*			*MBRR*			
	total	total	/	retention	total	(reduction)	/ retention	(reduction)
idtc	1749.9	1944.2	/	194.3	1840.2	(5.4%)	/ 90.3	(53.5%)
usb_phy	1559.4	1895.4	/	336.0	1851.3	(2.3%)	/ 291.9	(13.1%)
dist	3894.9	4413.0	/	518.1	4346.3	(1.5%)	/ 451.4	(12.9%)
DMA	18606.0	21524.5	/	2918.5	18916.4	(12.1%)	/ 310.4	(89.4%)
pic_bridge	35656.1	42440.4	/	6784.3	36630.7	(13.7%)	/ 974.6	(85.6%)
b19	39633.6	44402.0	/	4768.4	40290.4	(9.3%)	/ 656.9	(86.2%)
des_perf	160840.1	196073.2	/	35233.1	226189.0	(-15.4%)	/ 65348.9	(-85.5%)
ECC decoder	258856.4	285520.7	/	26664.3	287212.8	(- 0.6%)	/ 28356.3	(- 6.3%)
average	—	—			—	(3.5%)	/ —	(31.1%)

Uniform MBRR

- Chen et al.'s approach **[Chen et al., ICCAD'12 & TCAD'14]**
  - 20~46% more MBRRs compared with our approaches
- The proposed ILP approach **[Lin & Lin, ICCAD'14]**
  - fewest MBRRs; longest runtime
- The proposed hybrid approach (Pre-selection + ILP)
  - shortest runtime; slightly more MBRRs

Circuit	Chen *et al.*'s approach [8]			the proposed hybrid approach			the proposed pure ILP approach		
	# MBRRs	# bit stored	runtime (sec)	# MBRRs	# bit stored /(reduction)	runtime (sec)	# MBRRs	# bit stored /(reduction)	runtime (sec)
idtc	1	11	< 0.1	1	11 / (00.0%)	< 0.1	1	11 / (00.0%)	< 0.1
usb_phy	6	42	< 0.1	4	28 / (33.3%)	< 0.1	4	28 / (33.3%)	< 0.1
dist	5	55	< 0.1	4	44 / (20.0%)	< 0.1	4	44 / (20.0%)	< 0.1
DMA	23	115	0.7	18	90 / (21.7%)	1.0	18	90 / (21.7%)	1.0
pci_bridge	34	238	4.0	23	161 / (32.4%)	1.1	23	161 / (32.4%)	1.1
b19	56	448	24.8	30	240 / (46.4%)	23.7	30	240 / (46.4%)	59.7
des_perf	732	8052	1190.4	549	6039 / (25.0%)	984.1	544	5984 / (25.7%)	22173.0
ECC decoder	576	3456	197.8	445	2670 / (22.7%)	134.8	N/A	N/A / (N/A)	N/A

Uniform v.s. Non-Uniform MBRR

- Compared with "Uniform MBRR" **[Lin & Lin, ICCAD'14]**, "Non-uniform MBRR" results in
  - **34%** storage size reduction based on greedy method;
  - **41%** storage size reduction based on ILP-based method;
  - **34%** storage utilization improvement.

Circuit	Uniform MBRR replacement [8]			Non-uniform MBRR replacement (The greedy method in Section III)			Non-uniform MBRR replacement (The ILP-based method in Section IV)		
	# bit stored	# bit used / utilization	Time (s)	# bit stored / (reduction)	# bit used / utilization	Time (s)	# bit stored / (reduction)	# bit used / utilization	Time (s)
dtc	13	78 / 54.55%	< 0.1	78 / (45.45%)	78 / 100%	< 0.1	48 / (66.43%)	48 / 100%	0.9
usb_ phy	18	77 / 61.11%	< 0.1	77 / (38.89%)	77 / 100%	< 0.1	67 / (46.83%)	67 / 100%	0.7
dist	20	158 / 71.82%	< 0.1	158 / (28.18%)	158 / 100%	0.1	111 / (49.55%)	111 / 100%	1.3
DMA	77	206 / 53.51%	1.2	206 / (46.49%)	206 / 100%	1.3	201 / (47.79%)	201 / 100%	11.7
pci_ pridge	104	357 / 49.04%	1.4	357 / (50.96%)	357 / 100%	1.4	351 / (51.78%)	351 / 100%	23.5
b19	46	303 / 82.34%	23.2	303 / (17.66%)	303 / 100%	23.4	286 / (22.28%)	286 / 100%	88.7
des_ perf	787	7858 / 90.77%	575.0	7858 / (9.23%)	7858 / 100%	579.6	7702 / (11.03%)	7702 / 100%	1000.1
ECC decoder	486	1911 / 65.53%	155.3	1911 / (34.47%)	1911 / 100%	155.6	1857 / (36.31%)	1857 / 100%	448.9
Average		– / 66.08%		– / (33.92%)	– / 100%		– / (41.05%)	– / 100%	

Experiment on Different Latencies

- The best mode transition latency of each circuit with the smallest storage size of retention latches

Circuit	Storage size (bits) resulting from [8] with different mode transition latencies									
	2 cycles	3 cycles	4 cycles	5 cycles	6 cycles	7 cycles	8 cycles	9 cycles	10 cycles	11 cycles
idtc	56	66	80	85	102	112	120	126	130	143
usb_ phy	96	90	96	100	114	126	144	162	160	176
dist	128	141	160	160	186	168	184	189	210	220
DMA	408	354	328	385	432	490	560	621	690	748
pci_bridge	886	594	584	605	654	728	816	882	970	1056
b19	1772	1386	756	540	450	399	368	378	360	396
des_perf	8910	8799	8808	8795	8616	8729	8776	8622	9080	8657
ECC decoder	4688	4410	3284	3125	2910	2968	2984	3213	4190	6072

Circuit	Storage size (bits) resulting from our non-uniform MBRR replacement approach with different mode transition latencies									
	2 cycles	3 cycles	4 cycles	5 cycles	6 cycles	7 cycles	8 cycles	9 cycles	10 cycles	11 cycles
idtc	48	48	48	48	48	48	48	48	48	48
usb_ phy	76	67	67	67	67	67	66	66	66	66
dist	128	111	111	111	111	111	111	111	111	111
DMA	304	248	201	201	201	201	201	201	201	201
pci_bridge	675	445	400	365	352	351	337	337	337	337
b19	1419	894	606	482	360	328	286	286	286	286
des_ perf	8247	8086	7877	7817	7785	7759	7724	7714	7709	7702
ECC decoder	3348	2759	2090	2090	1857	1857	1857	1857	1857	1857

Conclusions

- Advantages of MBRR for state recovery
  - Short wake-up time
  - Small area overhead
  - Low power consumption
- Uniform v.s. Non-uniform MBRR
  - Uniform MBRRs
    - Simpler control circuit
    - Larger storage size
    - Lower storage utilization
  - Non-uniform MBRRs
    - 100% storage utilization
    - Greedy method: fast, 34% storage size reduction
    - ILP-based method: slower, 41% storage size reduction

Thanks for Your Attention!

Mark Po-Hung Lin
marklin@ccu.edu.tw
http://eda.ee.ccu.edu.tw/~marklin
National Chung Cheng University (CCU)
Chiayi, Taiwan