

Download
manuscript PDF



Synthesis of SFQ Circuits with Compound Gates

Rassul Bairamkulov, Alessandro Tempia Calvino, and Giovanni De Micheli
Integrated Systems Laboratory, EPFL



OUTLINE

Motivation

Rapid Single-Flux Quantum – Overview

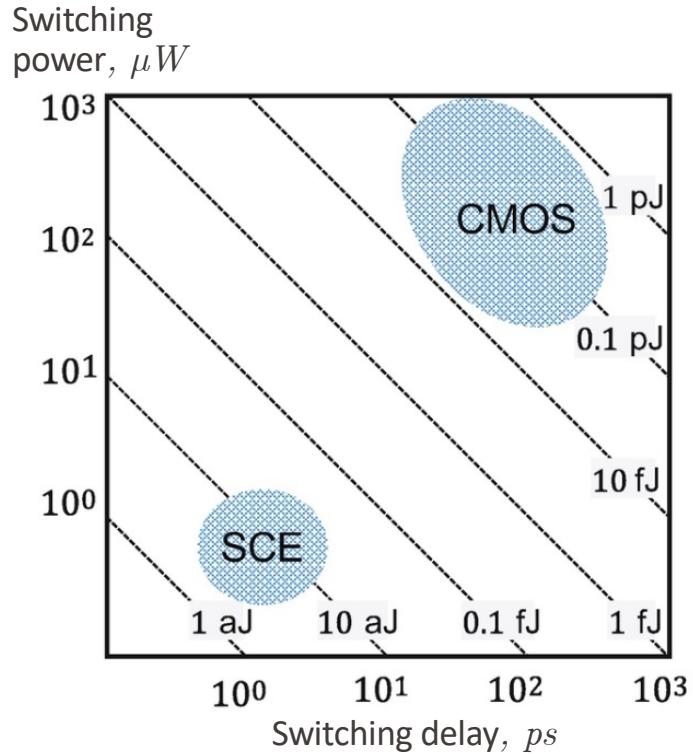
Gate Compounding Technique

Technology Mapping

Results

Rapid Single-Flux Quantum

- Rapid Single-Flux Quantum
 - Invented in 1985 by K. Likharev and V. Semenov [1]
- Superconductive technology
 - Cryogenic, 4.2 K
 - Based on SFQ pulses
- Low-power
 - Hence low heat
 - Even considering refrigeration
- World's fastest digital technology
 - Record speed of 770 GHz [2]
 - Consistently achieving speeds above 10 GHz



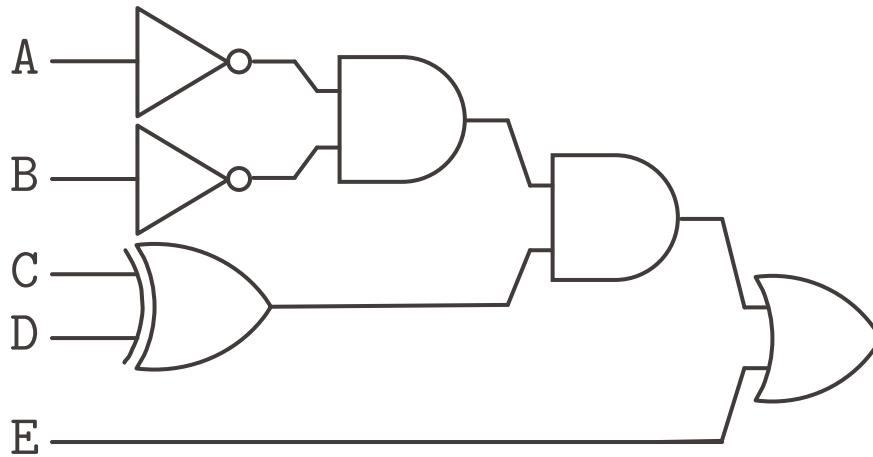
[1] K. K. Likharev and V. K. Semenov, RSFQ logic/memory family: a new Josephson-junction technology for sub-terahertz-clock-frequency digital systems, in IEEE TASC 1991

[2] W. Chen, et al. Rapid single flux quantum T-flip flop operating up to 770 GHz. IEEE TASC 1999.

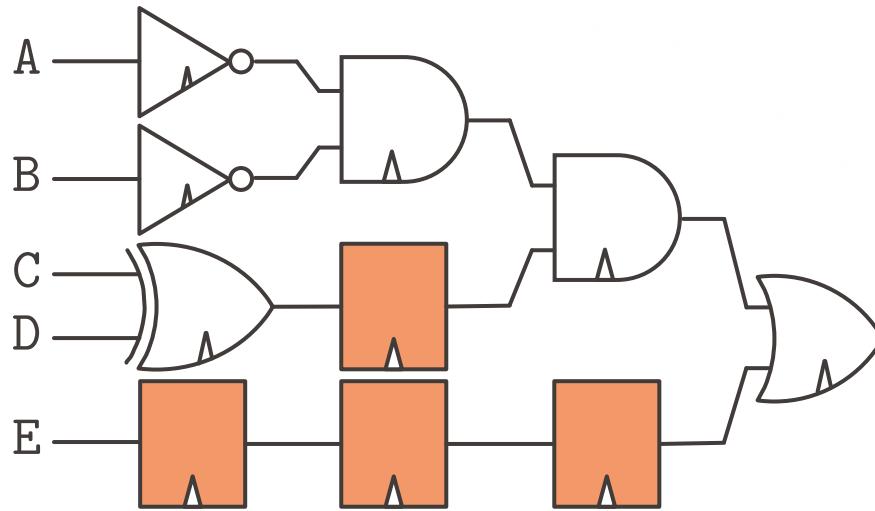
[3] D. S. Holmes, et al. Energy-efficient superconducting computing—Power budgets and requirements. IEEE TASC 2013.

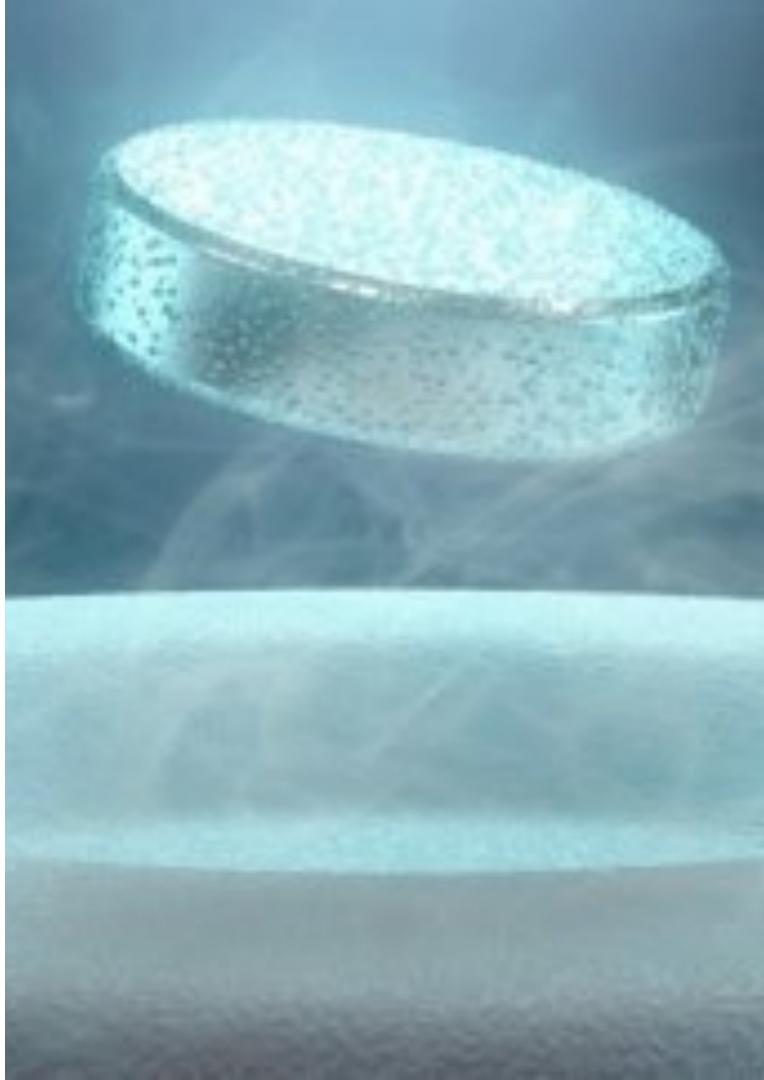
- Fabrication technology
 - State of the art : 6k JJ/mm²
 - Compare to 90M transistors/mm²
(TSMC 7nm)
- Expensive memory
 - Poor density of inductor-based memory
- Architectural differences
 - **Gate-level pipelining**

- Ensure correct order of signal arrival
- Fundamental issue in superconductive electronics



- Ensure correct order of signal arrival
- Fundamental issue in superconductive electronics
 - Dominates the IC area





OUTLINE

Motivation

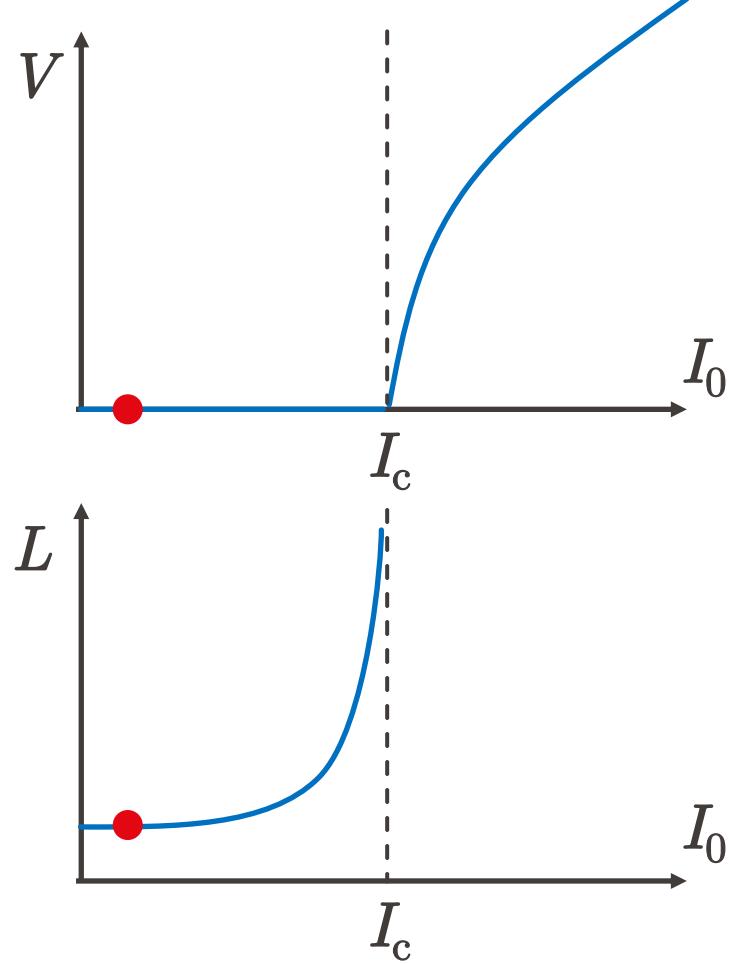
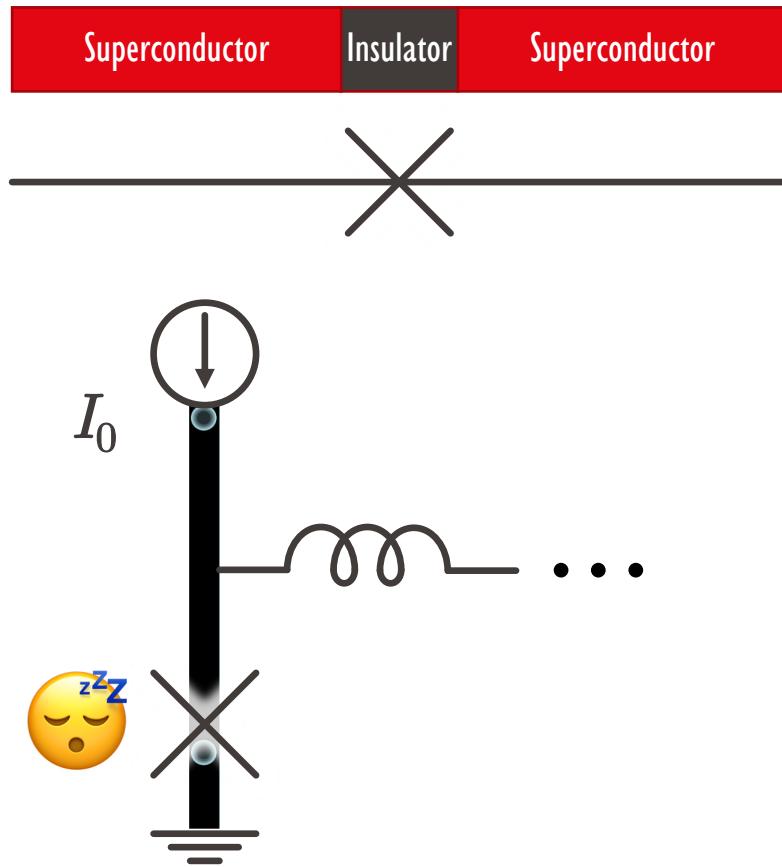
Rapid Single-Flux Quantum - Overview

Gate Compounding Technique

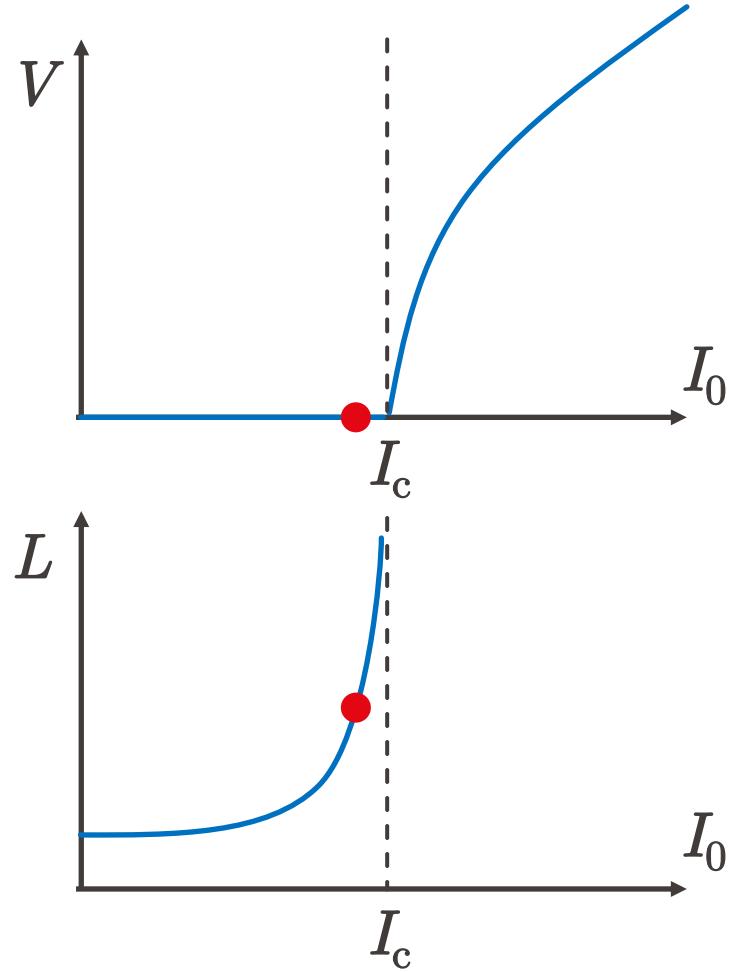
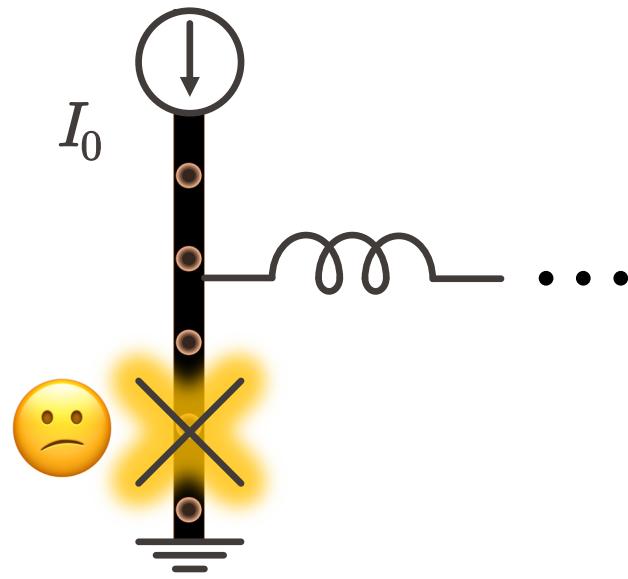
Technology Mapping

Results

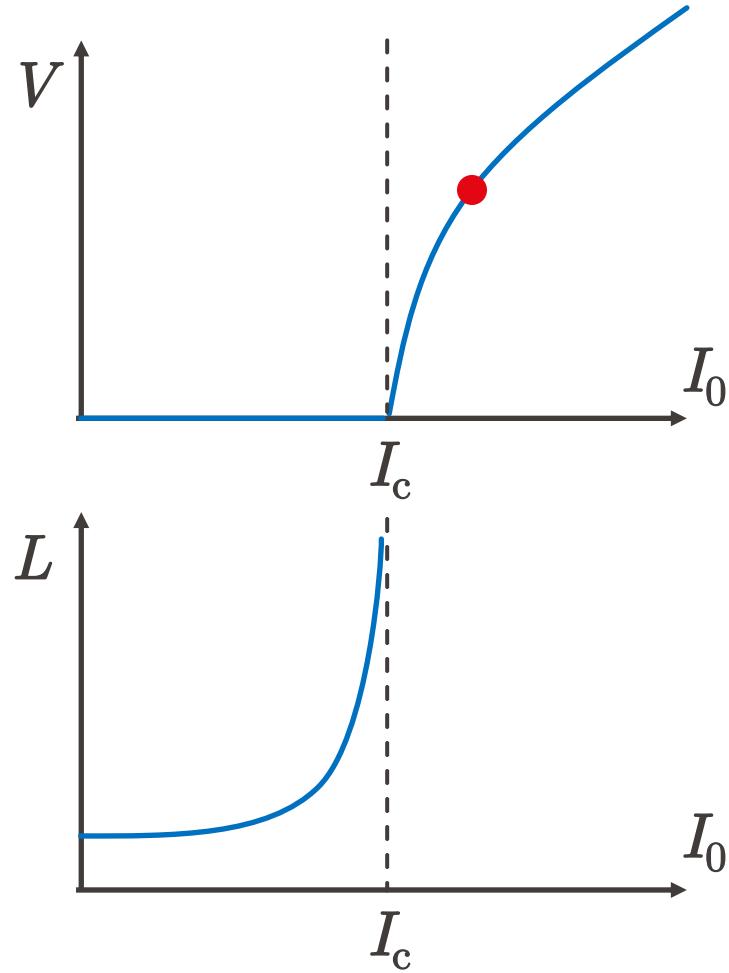
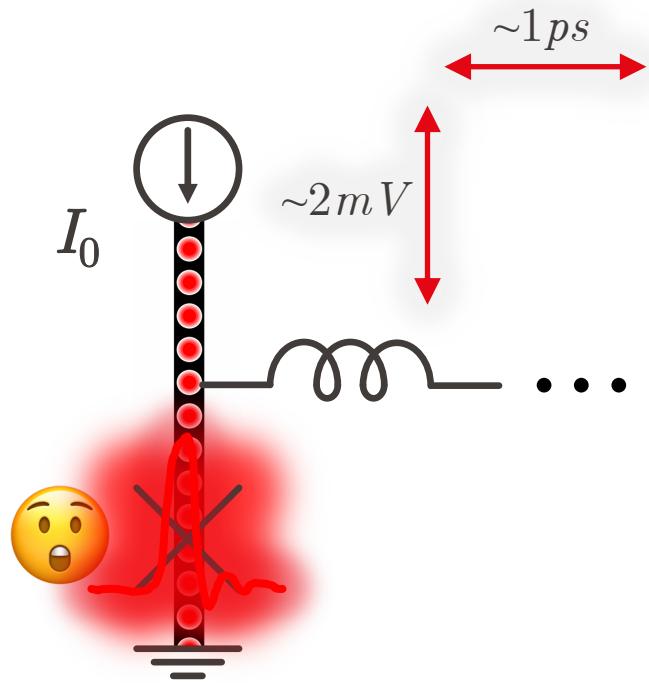
Josephson Junction (JJ)



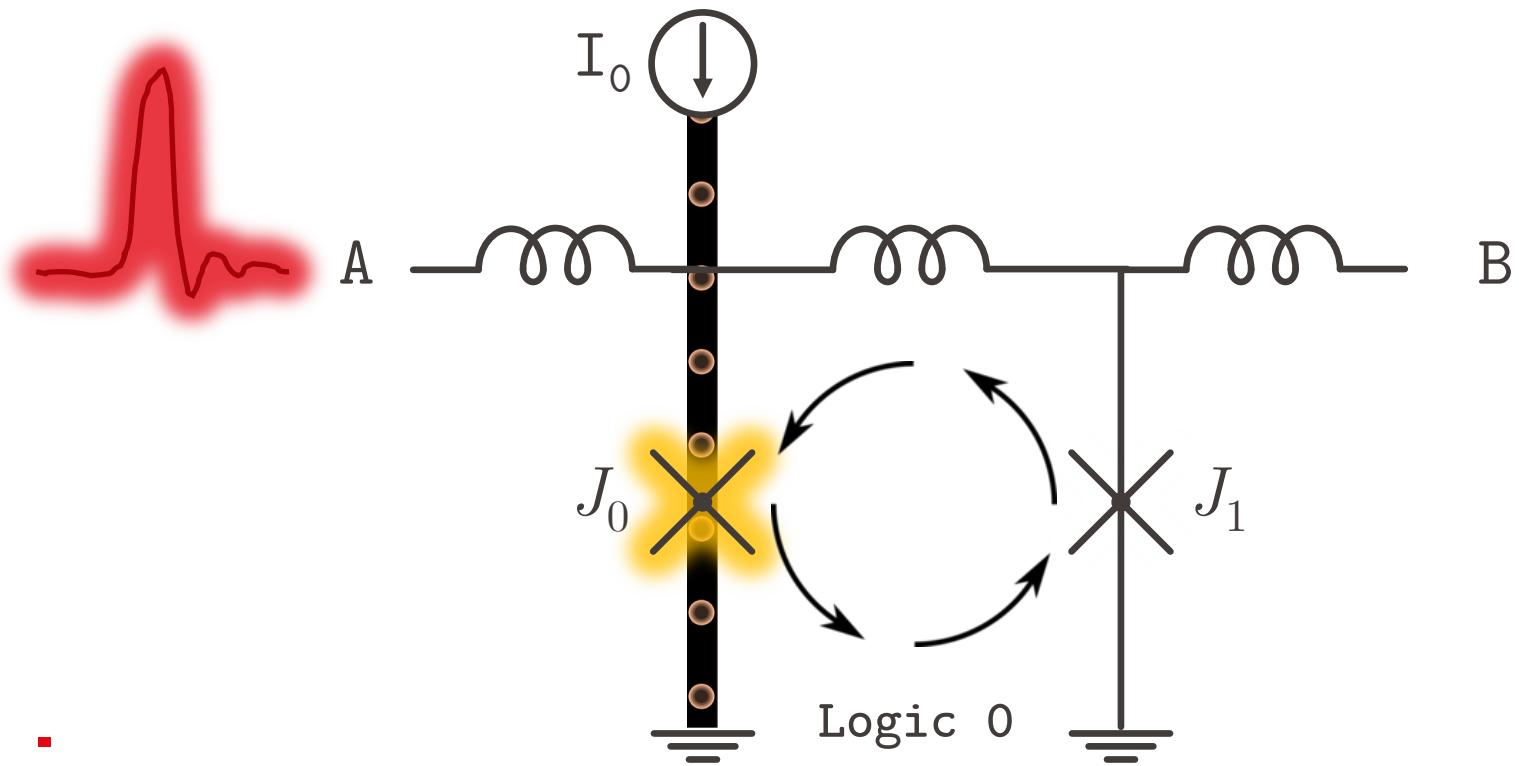
Josephson Junction (JJ)



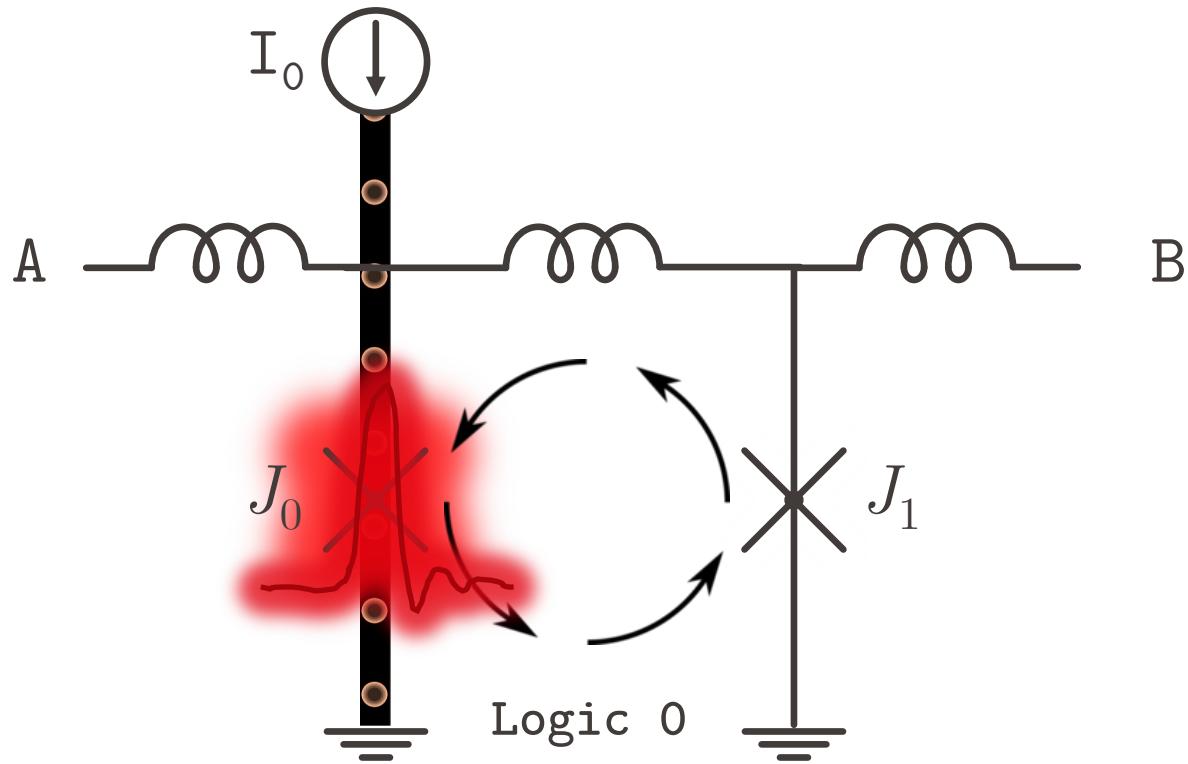
Josephson Junction (JJ)



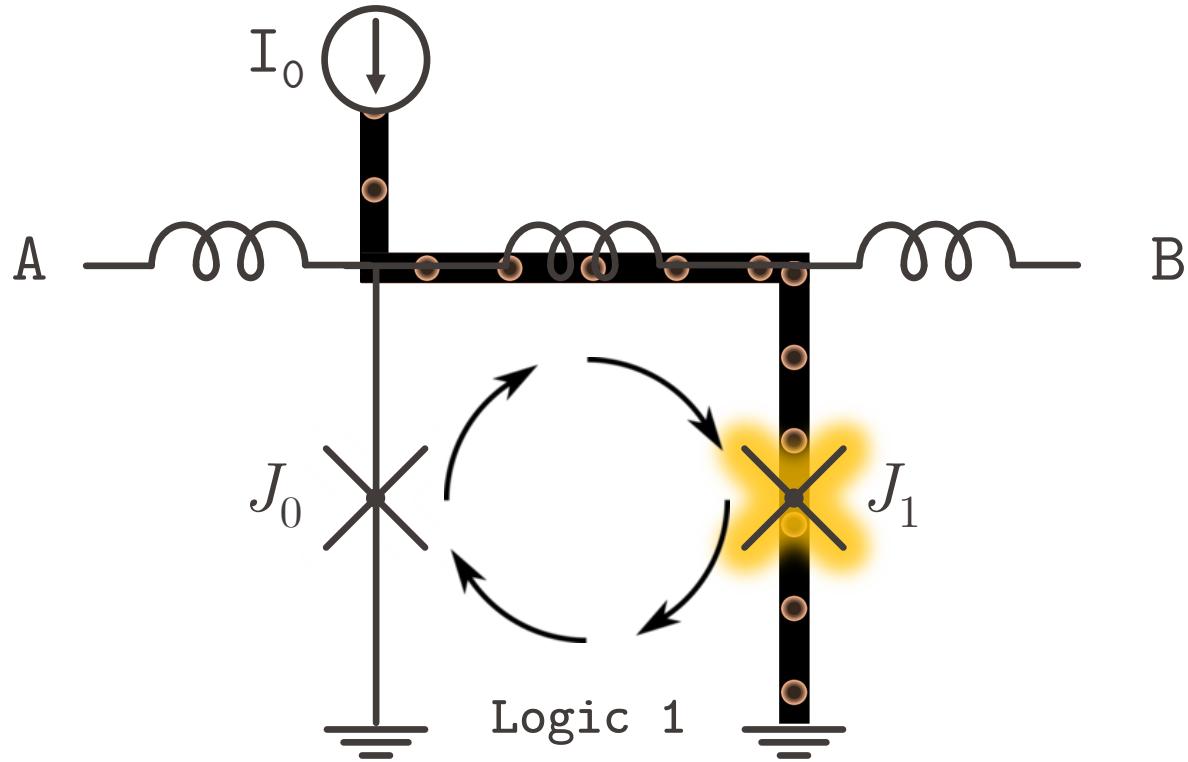
Basic JJ SQUID Loop



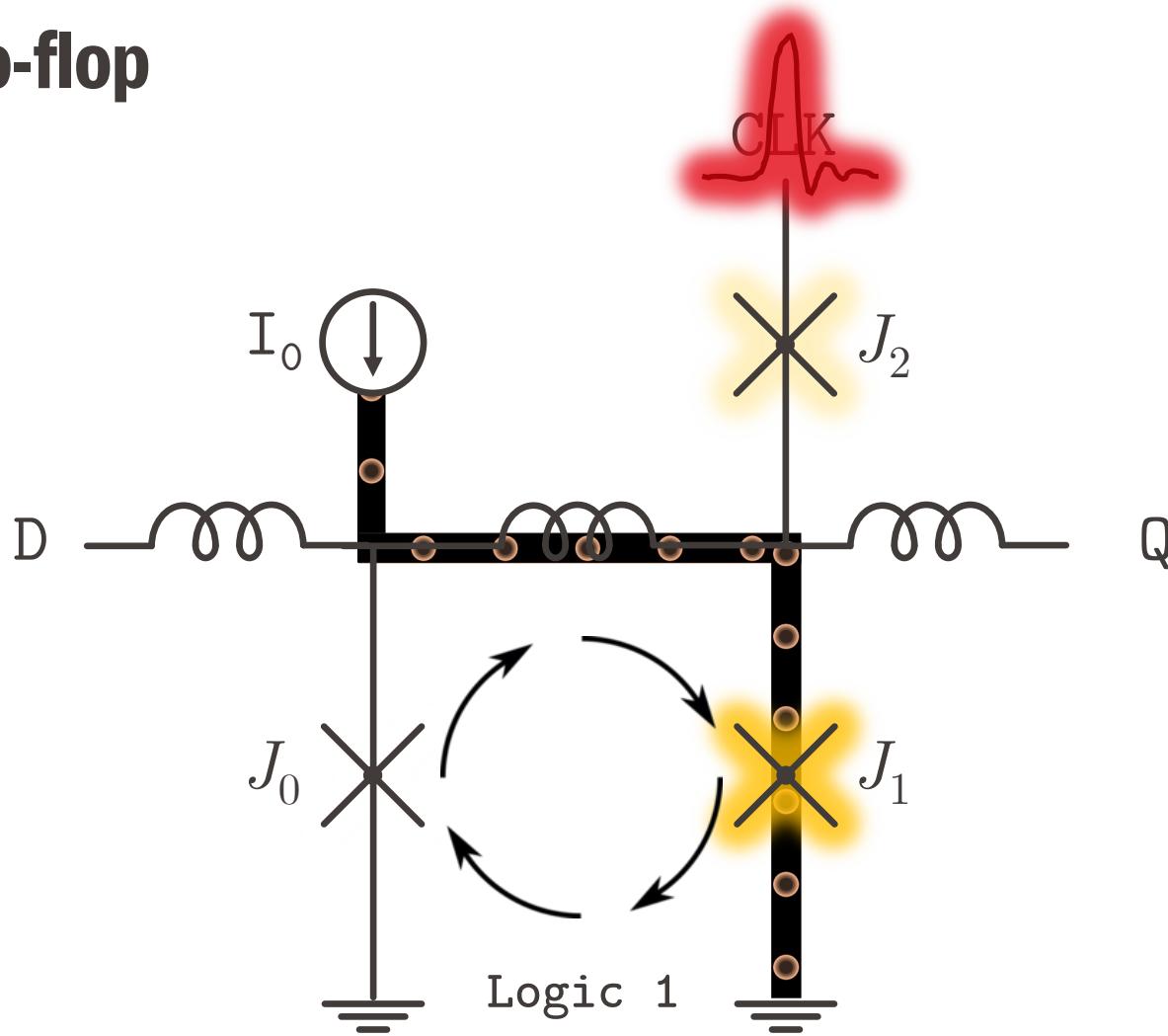
Basic JJ Loop



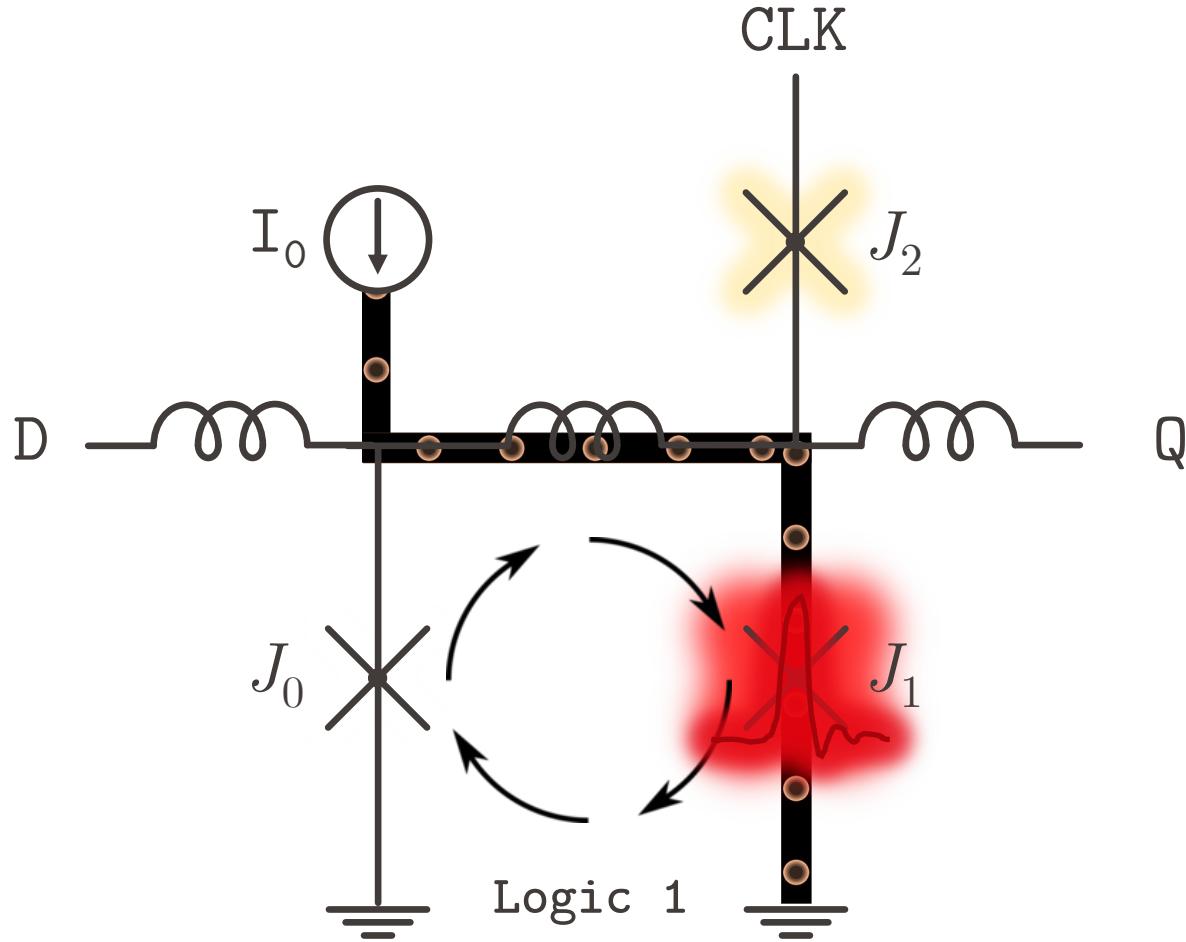
Basic JJ Loop



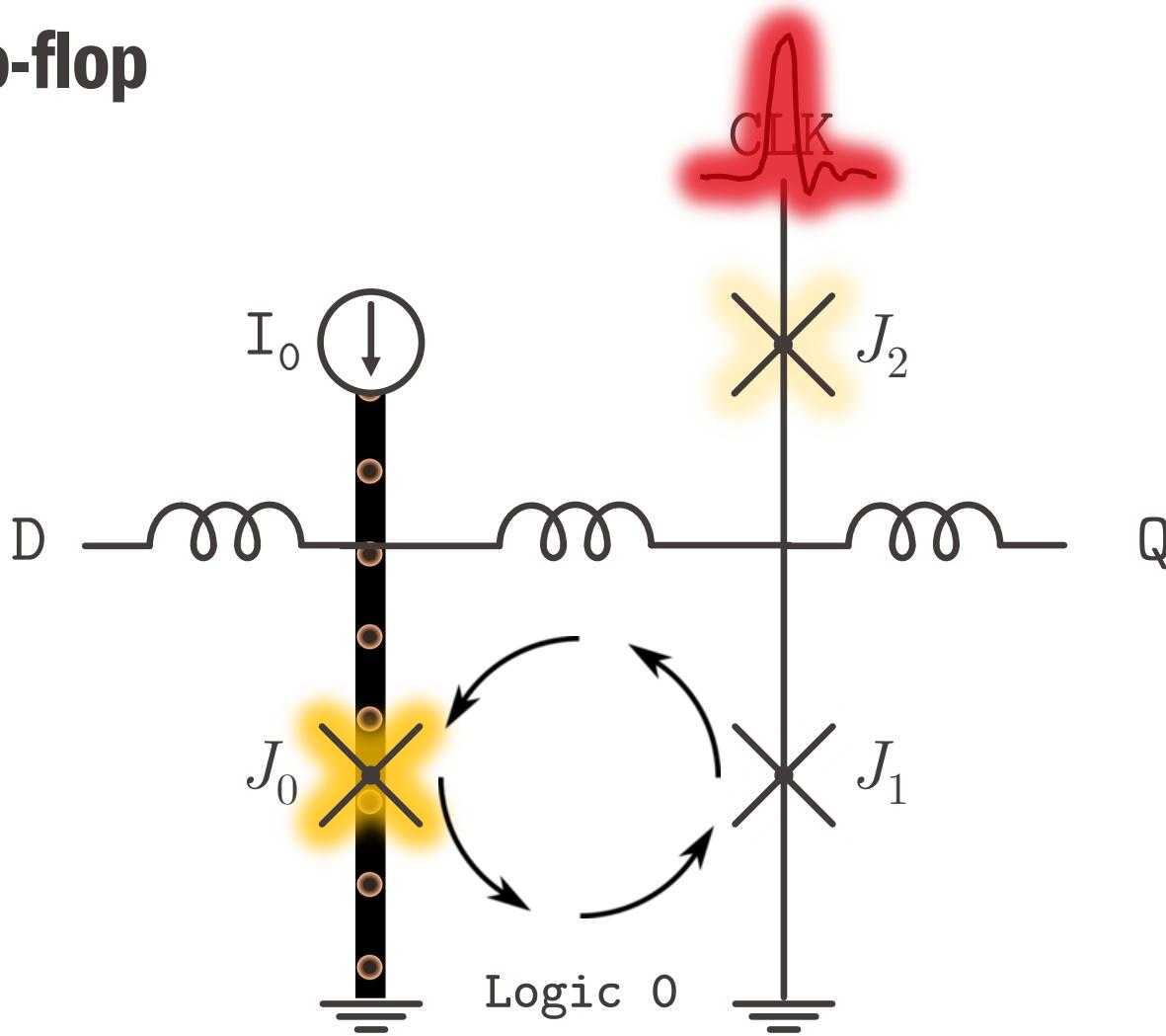
D flip-flop



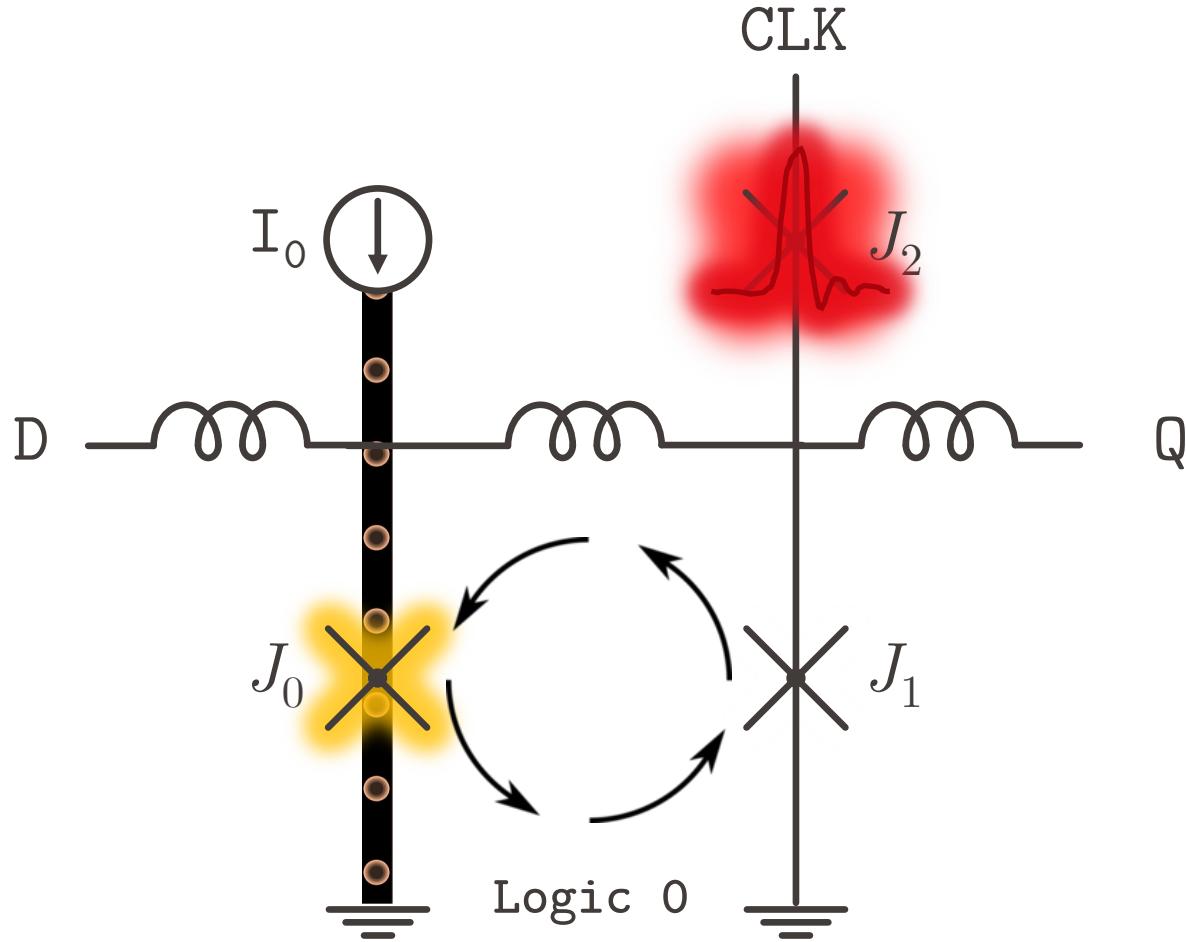
D flip-flop

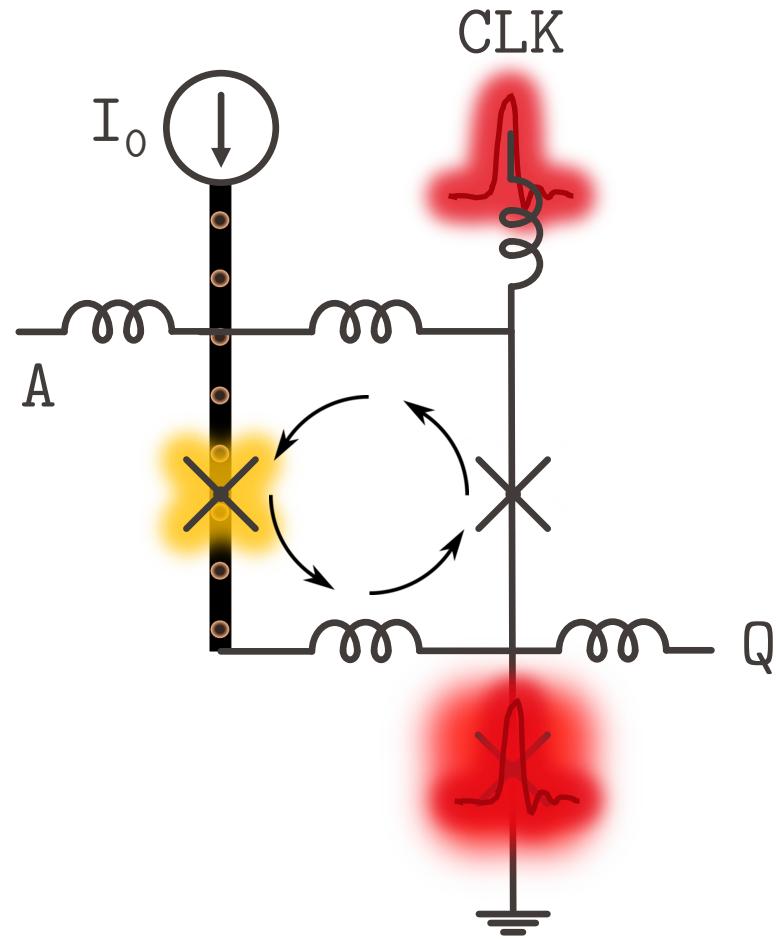


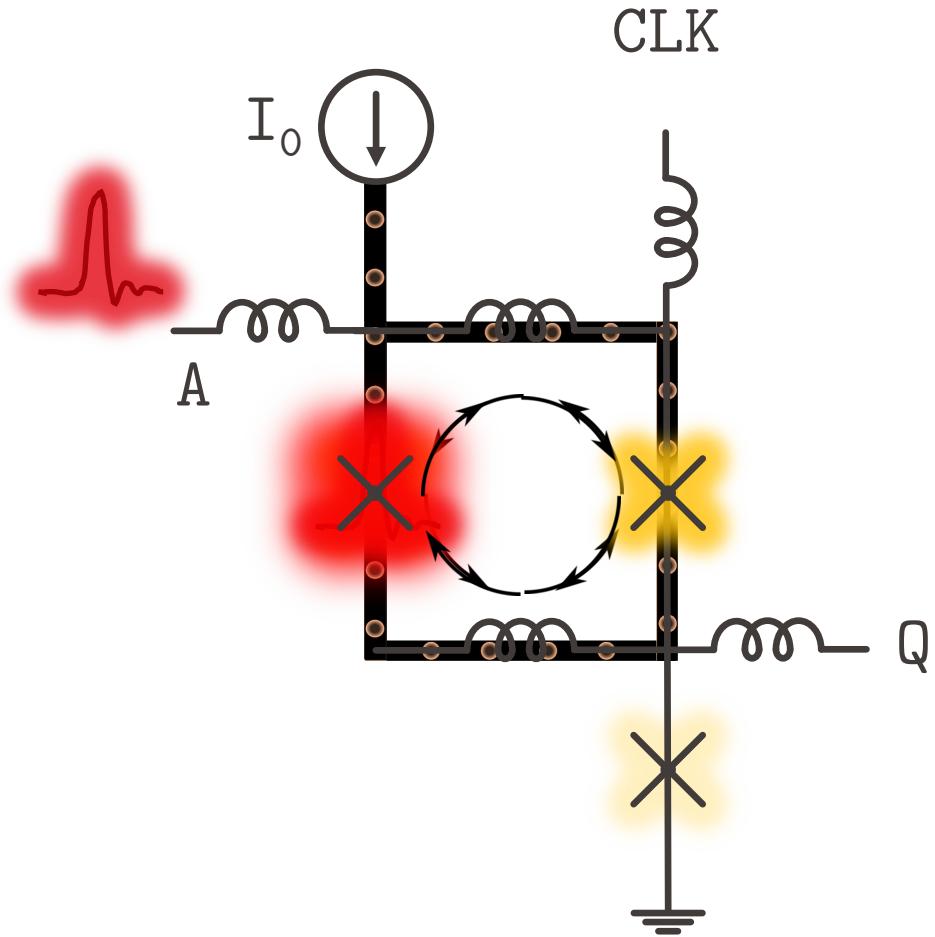
D flip-flop

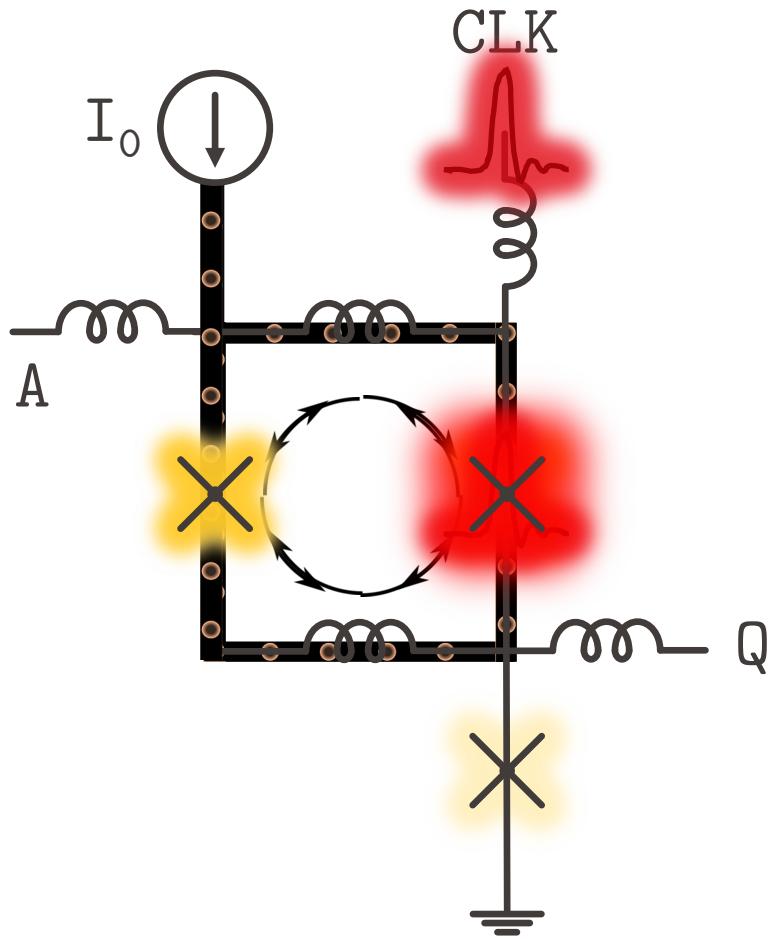


D flip-flop

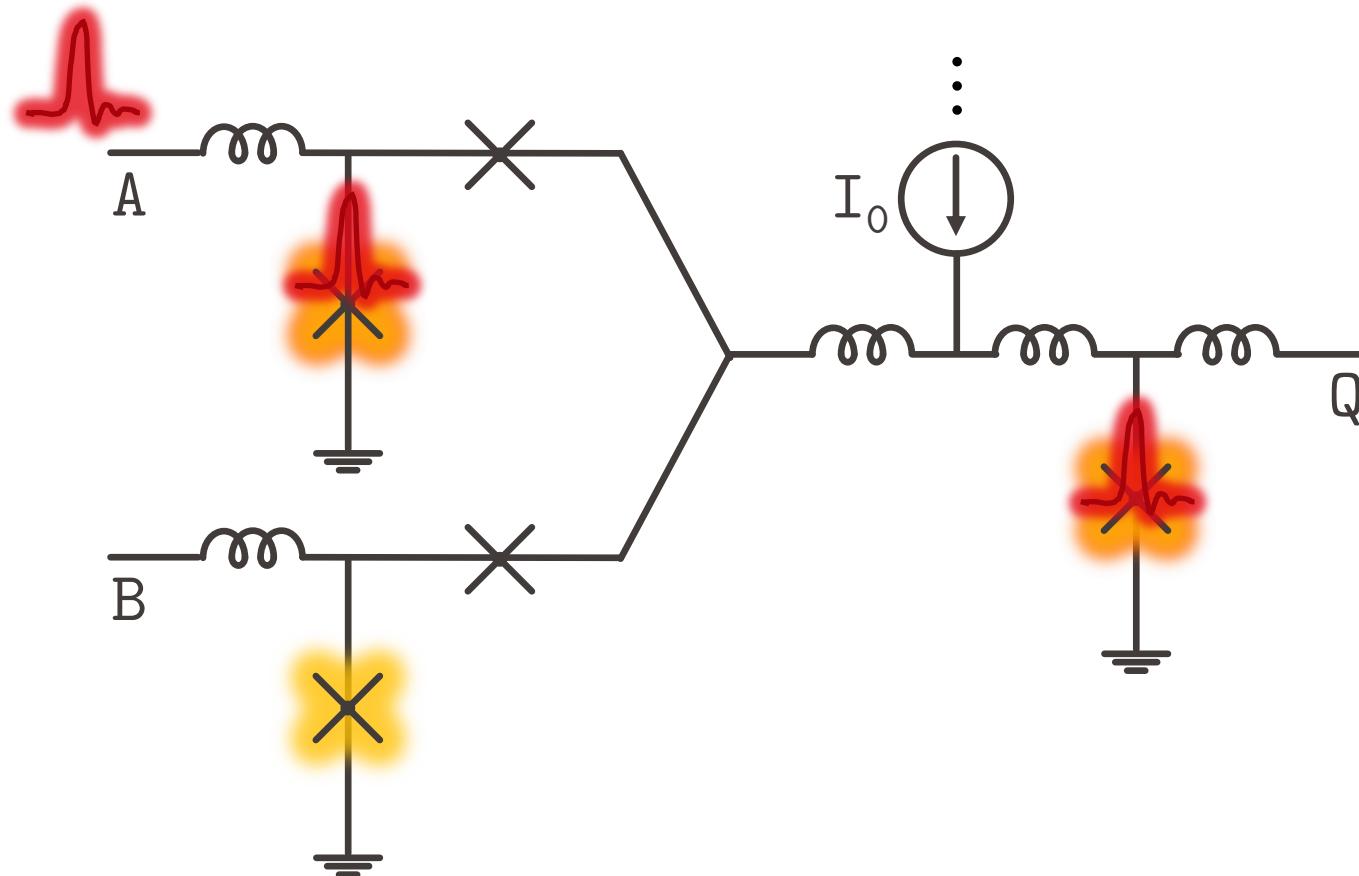




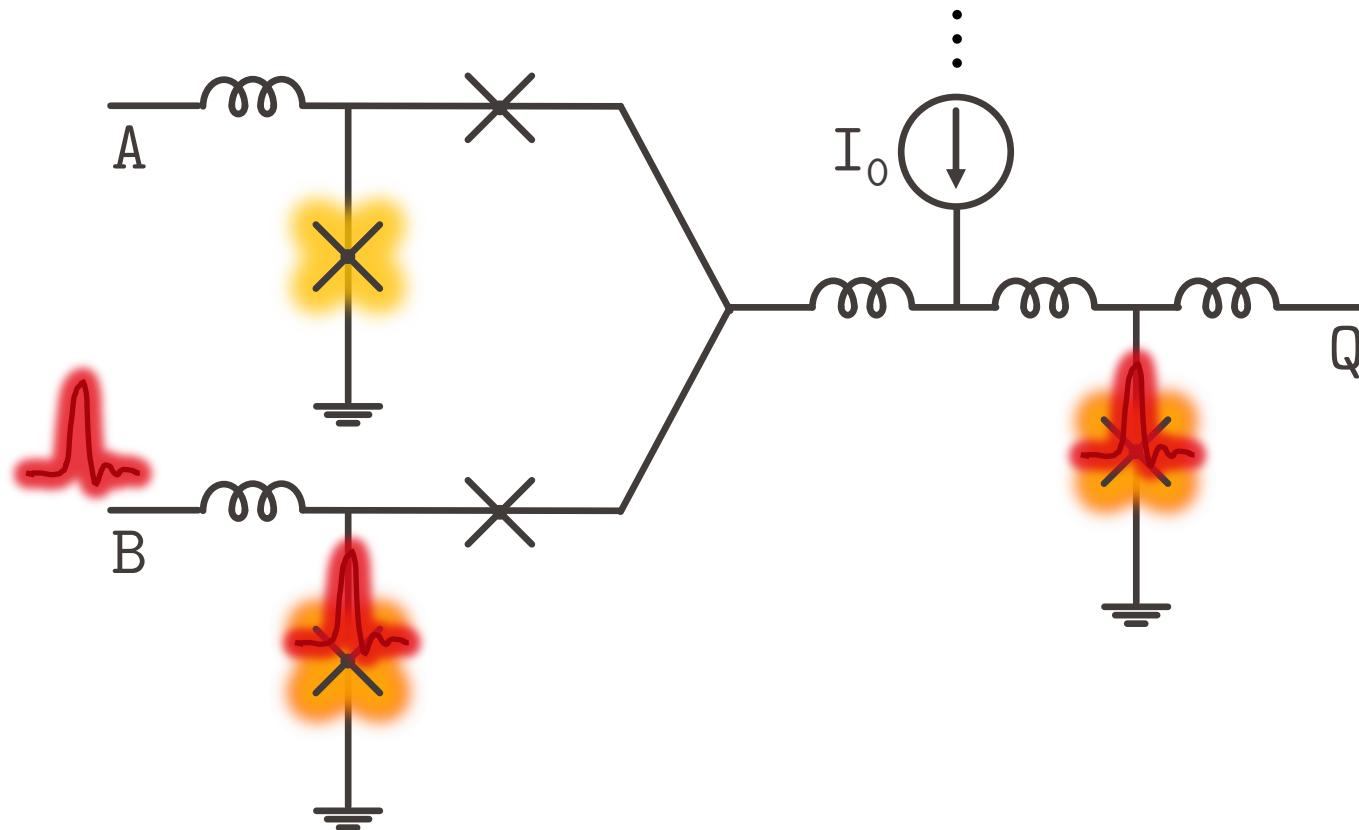




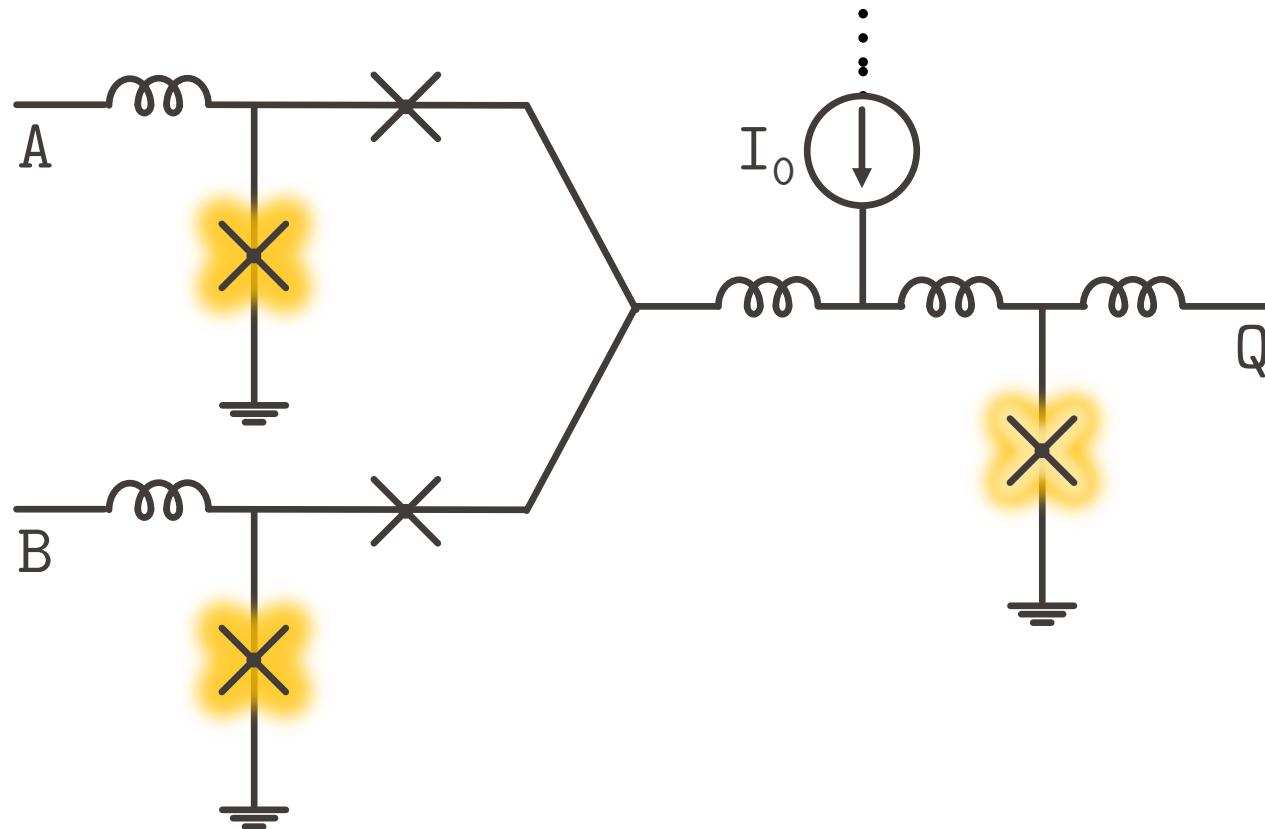
Merger (Asynchronous OR)



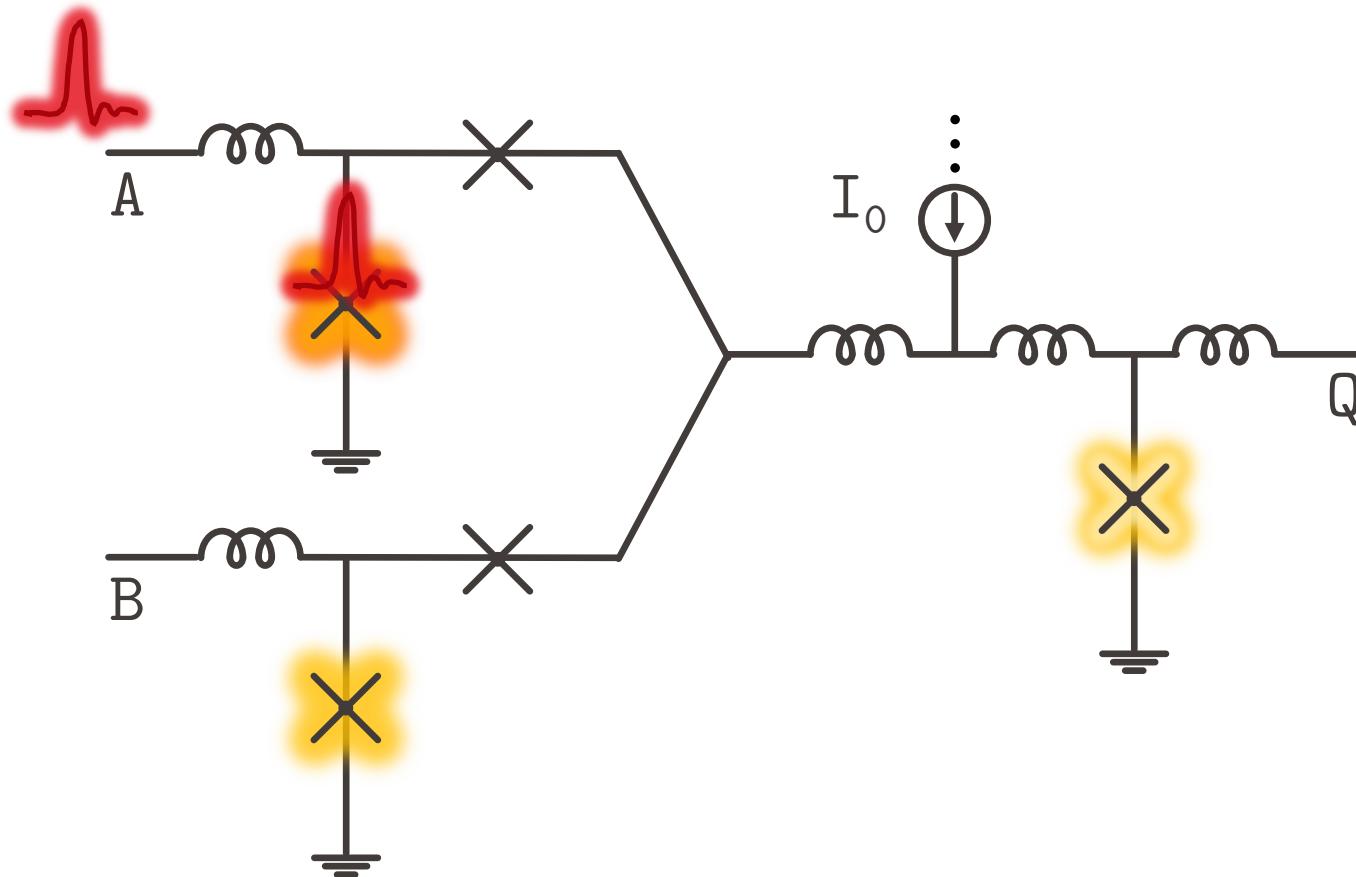
Merger (Asynchronous OR)



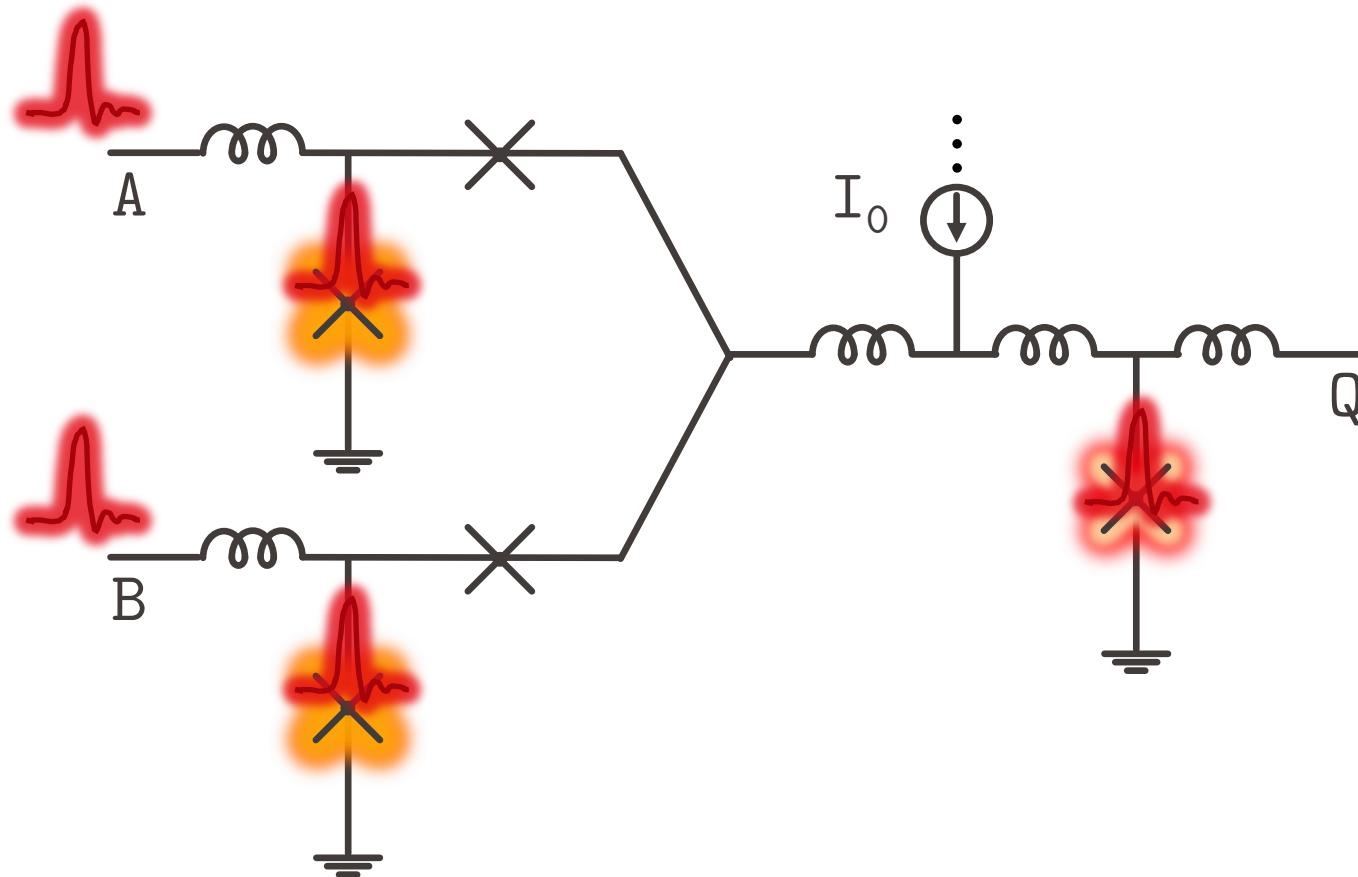
AND Gate



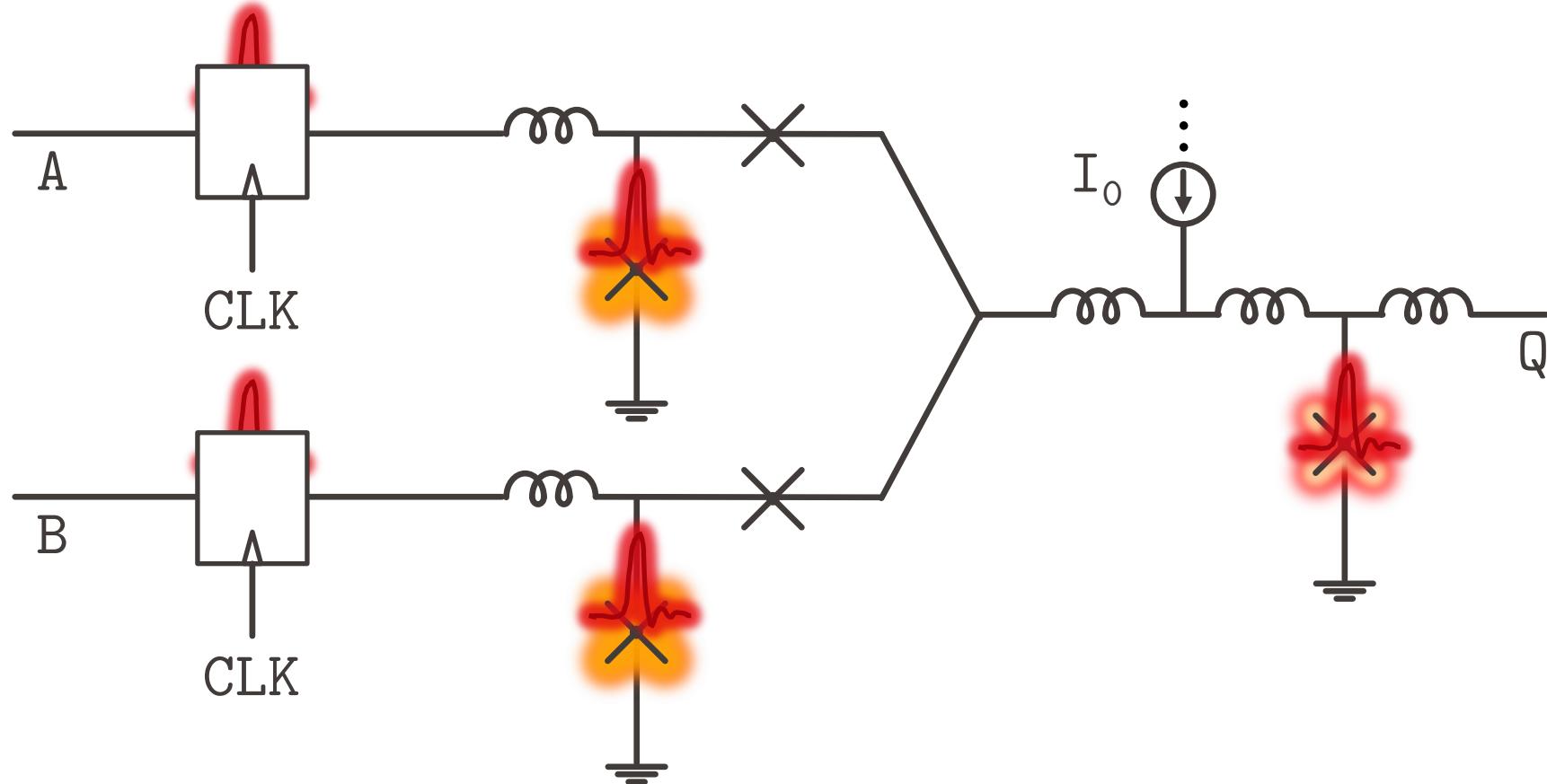
AND Gate

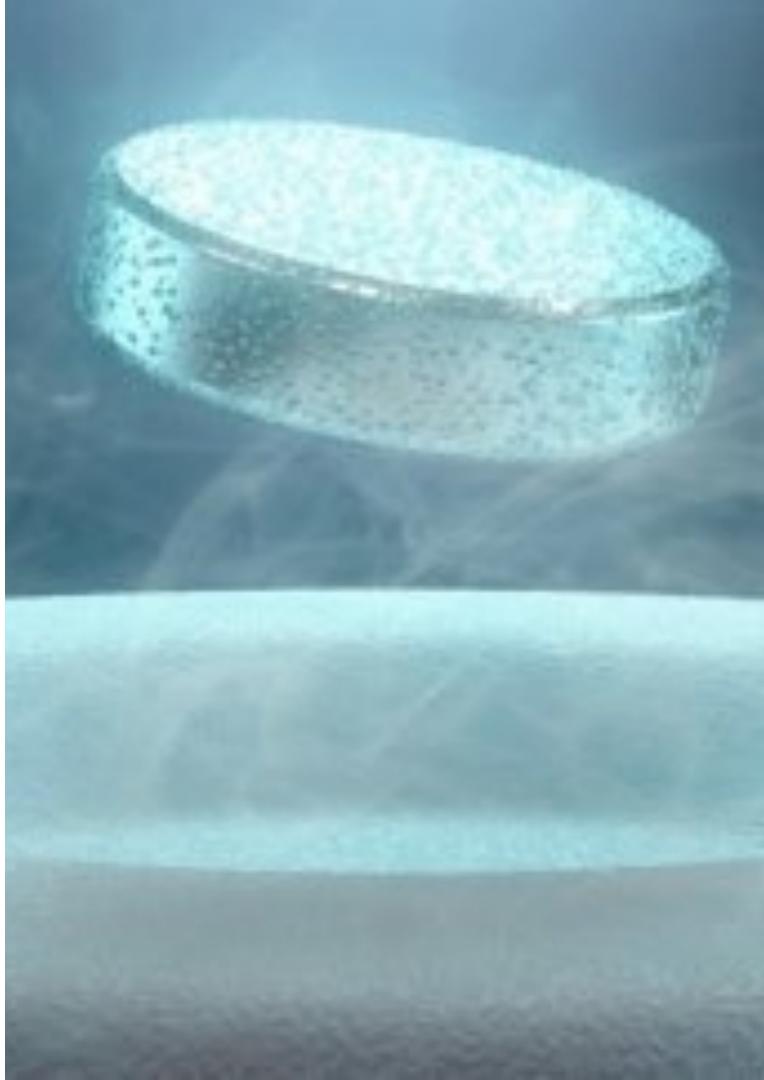


AND Gate



AND Gate





OUTLINE

Motivation

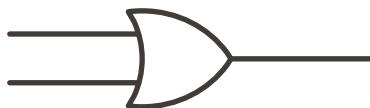
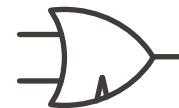
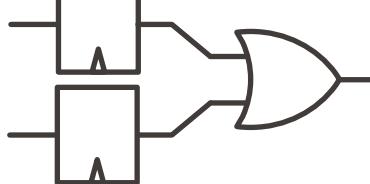
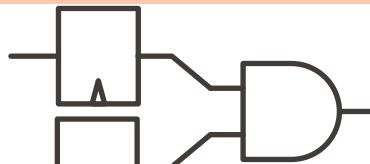
Rapid Single-Flux Quantum – Overview

Gate Compounding Technique

Technology Mapping

Results

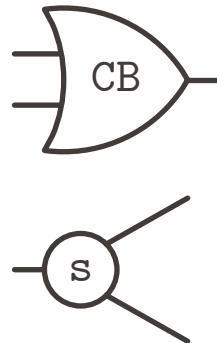
Function	Behavior in RSFQ	RSFQ Symbol
NOT		
XOR		
AND		
OR		
OR*		



Three Types of SFQ Devices

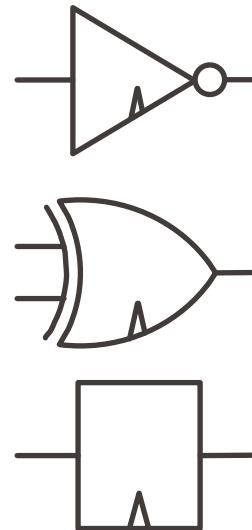
- Asynchronous

- Asynchronous input
 - Asynchronous output



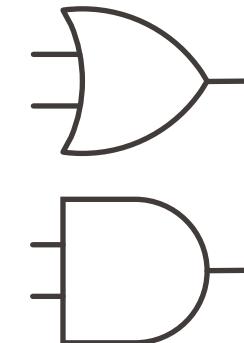
- Synchronizers

- Asynchronous input
 - **Synchronized** output



- Synchronized

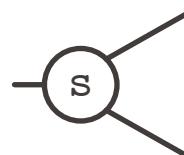
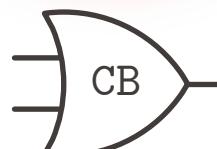
- **Synchronized** input
 - Asynchronous output



Three Types of SFQ Devices

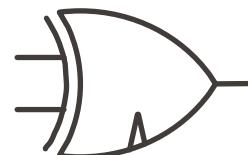
- Asynchronous

- Asynchronous input
- Asynchronous output



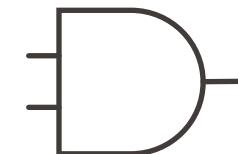
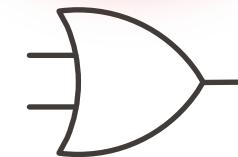
- Synchronizers

- Asynchronous input
- Synchronized output



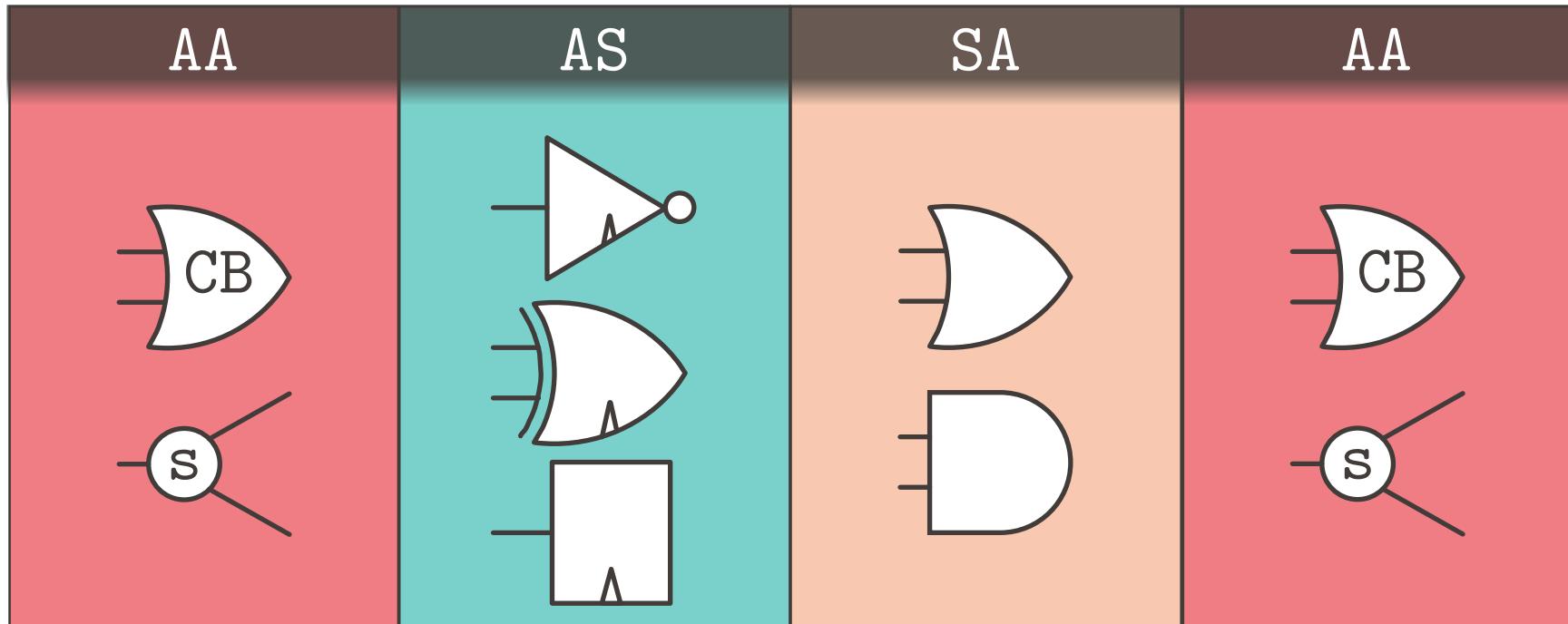
- Synchronized

- Synchronized input
- Asynchronous output

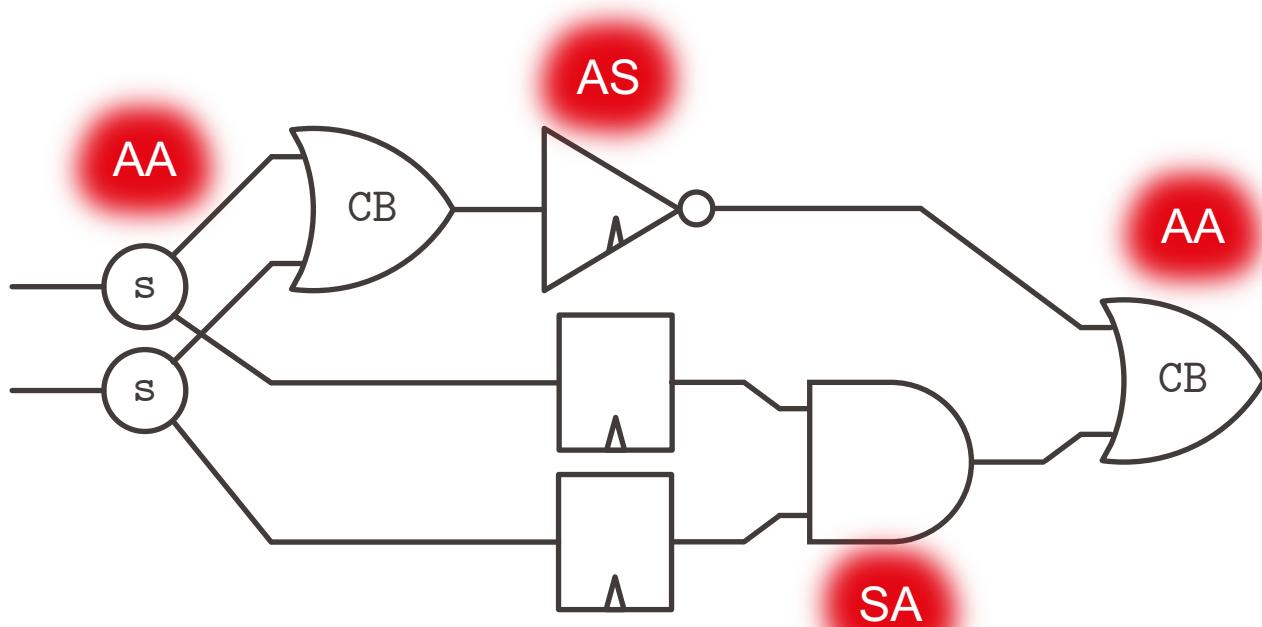


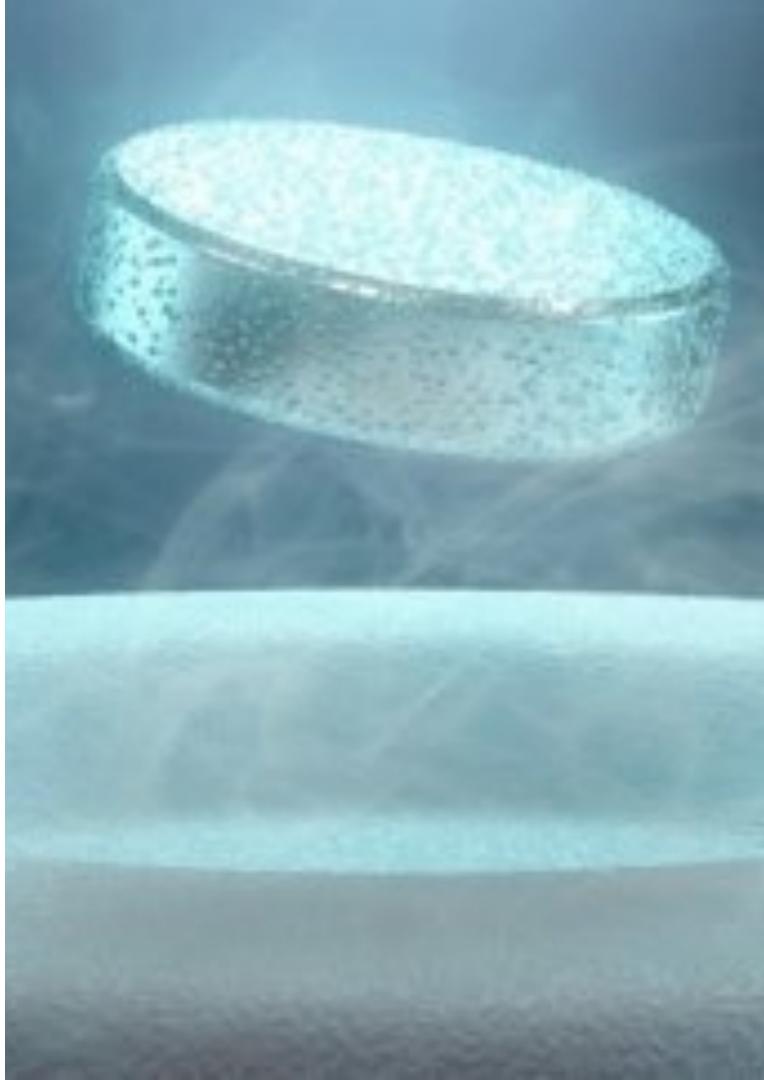
Gate Compounding Technique

- AS right before SA
 - Synchronize inputs to SA gates
- AA gates at periphery
 - Enrich functionality without extra cycles



Gate Compounding: XNOR





OUTLINE

Motivation

Rapid Single-Flux Quantum – Overview

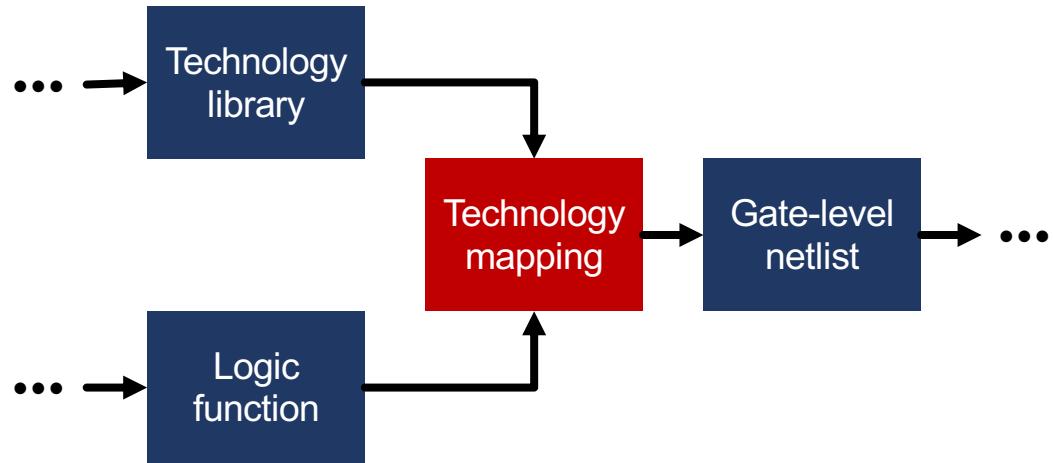
Gate Compounding Technique

Technology Mapping

Results

Technology Mapping 101

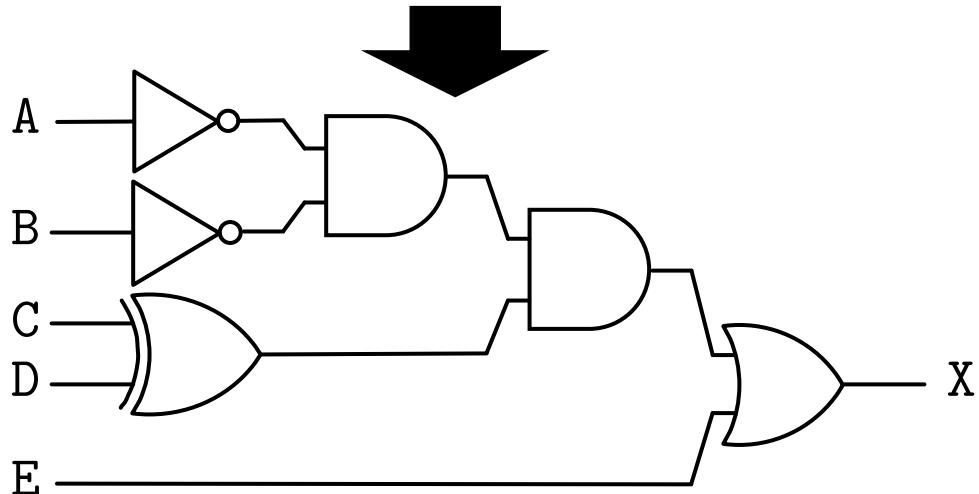
- Convert logic function into gate-level netlist
- Comply with technological constraints
 - Path balancing
 - Limited fanout
 - Timing constraints
- Optimize target metrics
 - Area
 - Delay



Technology Mapping 101

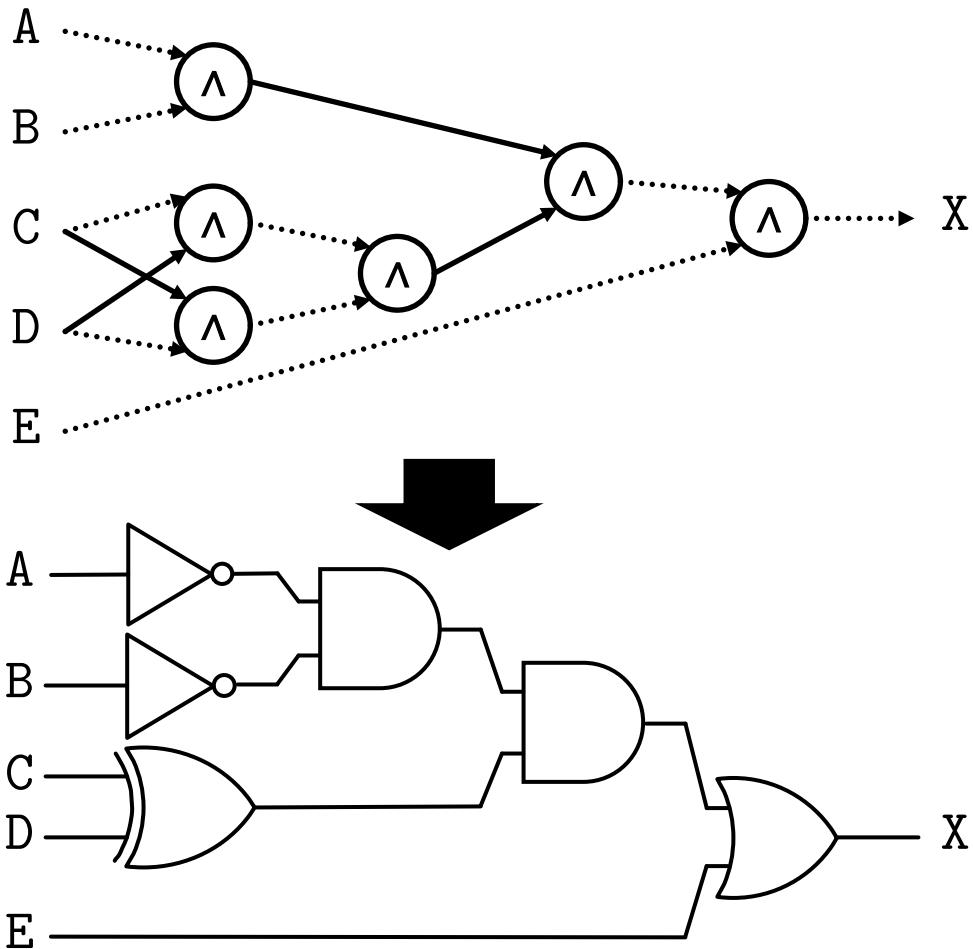
- Convert logic function into gate-level netlist
- Comply with technological constraints
 - Path balancing
 - Limited fanout
 - Timing constraints
- Optimize target metrics
 - Area
 - Delay

$$X = \overline{A}\overline{B}(C \oplus D)E$$



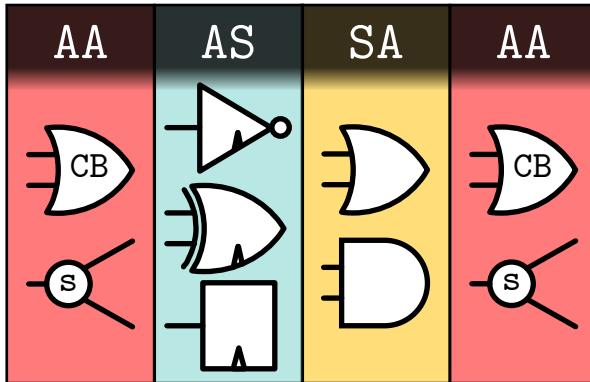
Technology Mapping 101

- Convert logic function into gate-level netlist
- Comply with technological constraints
 - Path balancing
 - Limited fanout
 - Timing constraints
- Optimize target metrics
 - Area
 - Delay



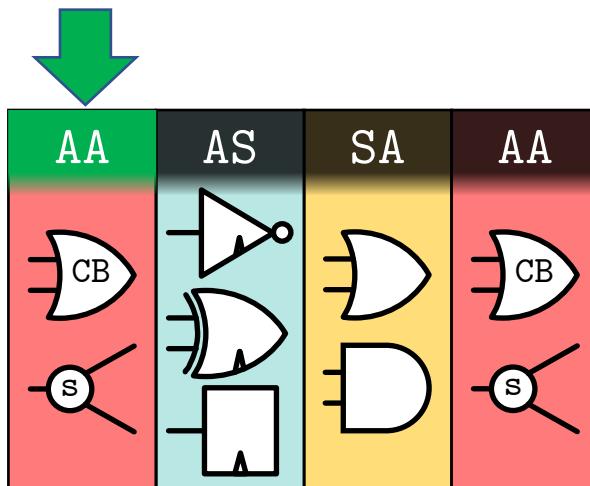
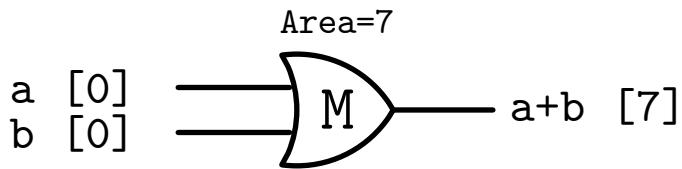
Precompute Technology Library

a [0]
b [0]



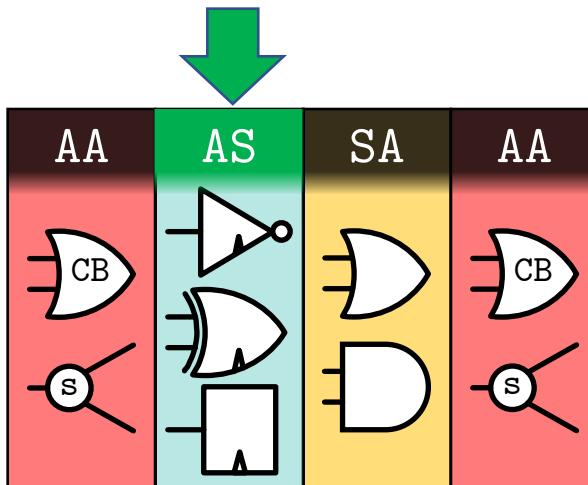
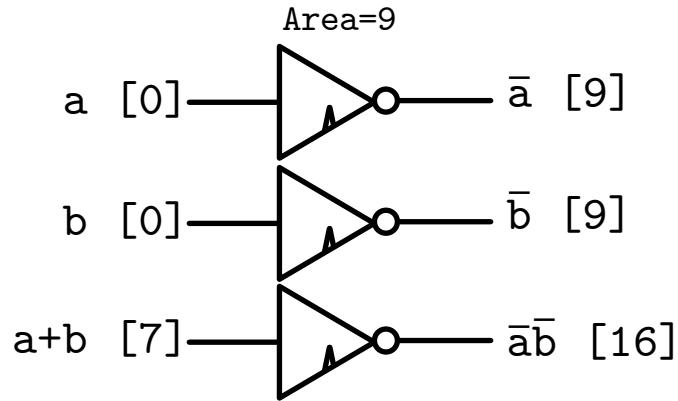
Function	Area	Delay
\bar{a}		
\bar{b}		
$\bar{a}\bar{b}$		
$a\bar{b}$		
$\bar{a}b$		
ab		
$a+b$		
$\bar{a}+b$		
$a+\bar{b}$		
$\bar{a}+\bar{b}$		
$ab+\bar{a}\bar{b}$		
$\bar{a}b+a\bar{b}$		

Precompute Technology Library: AA



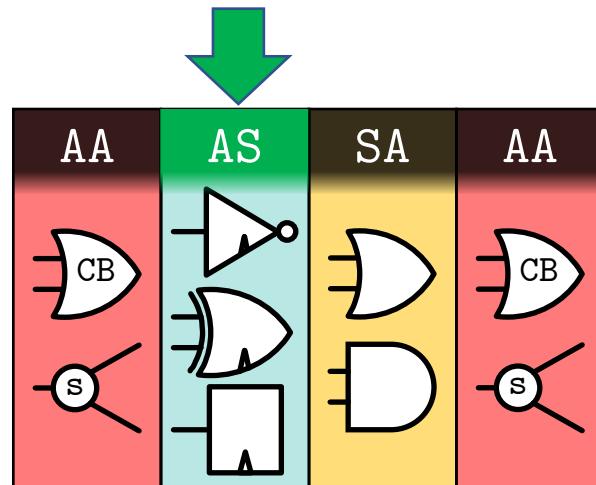
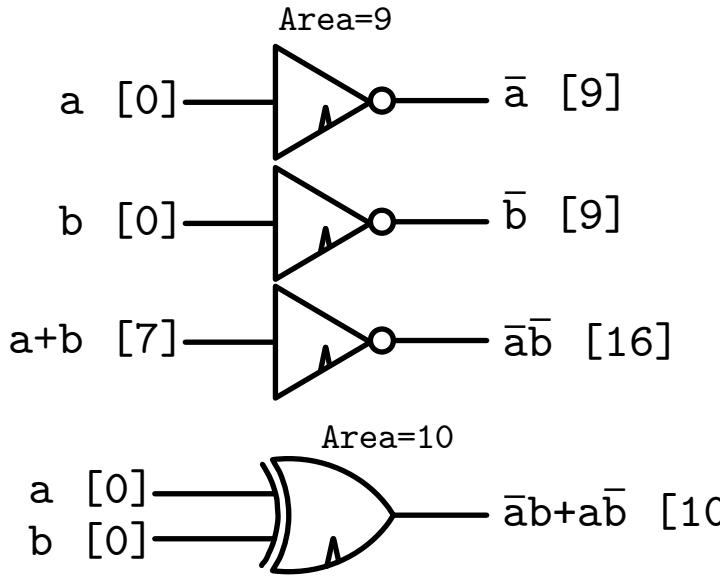
Function	Area	Delay
\bar{a}		
\bar{b}		
$\bar{a}\bar{b}$		
$a\bar{b}$		
$\bar{a}b$		
ab		
$a+b$	7	0
$\bar{a}+b$		
$a+\bar{b}$		
$\bar{a}+\bar{b}$		
$ab+\bar{a}\bar{b}$		
$\bar{a}b+a\bar{b}$		

Precompute Technology Library: AS



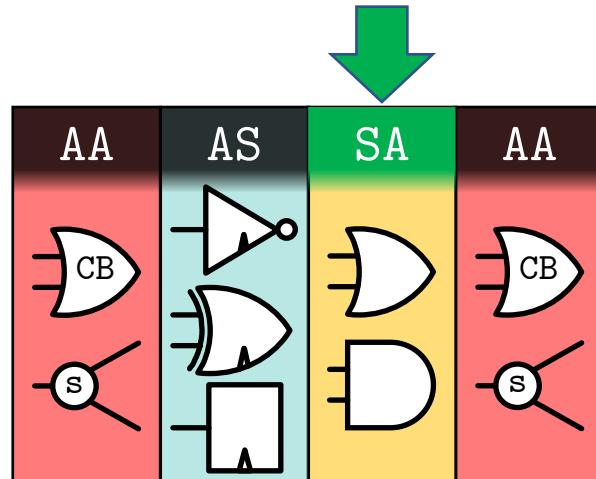
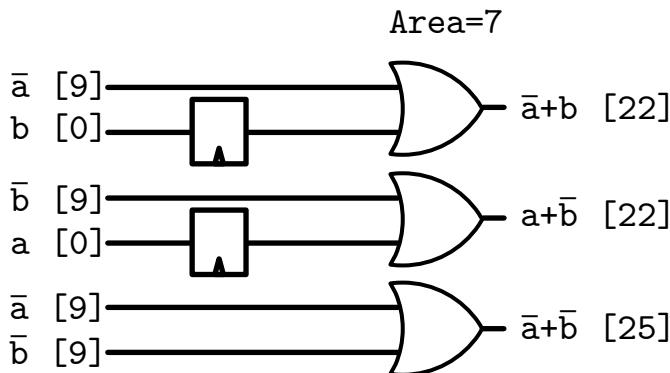
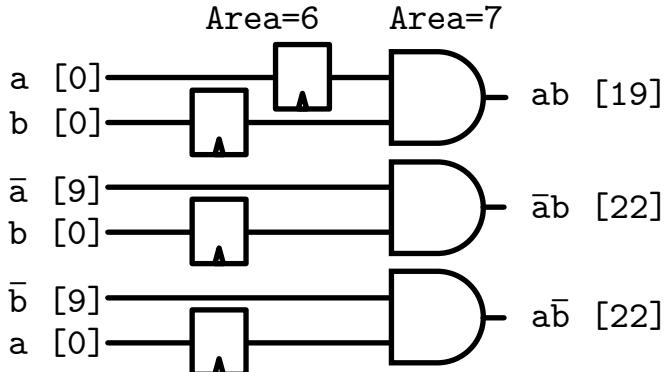
Function	Area	Delay
\bar{a}	9	1
\bar{b}	9	1
$\bar{a}\bar{b}$	16	1
$a\bar{b}$		
$\bar{a}b$		
ab		
$a+b$	7	0
$\bar{a}+b$		
$a+\bar{b}$		
$\bar{a}+\bar{b}$		
$ab+\bar{a}\bar{b}$		
$\bar{a}b+a\bar{b}$		

Precompute Technology Library: AS



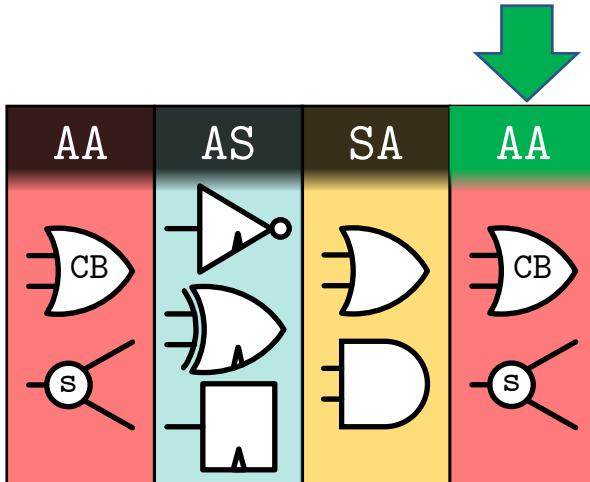
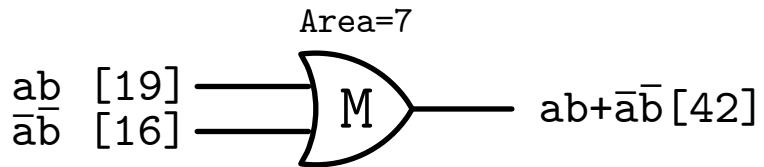
Function	Area	Delay
\bar{a}	9	1
\bar{b}	9	1
$\bar{a}\bar{b}$	16	1
$a\bar{b}$		
$\bar{a}b$		
ab		
$a+b$	7	0
$\bar{a}+b$		
$a+\bar{b}$		
$\bar{a}+\bar{b}$		
$ab+\bar{a}\bar{b}$		
$\bar{a}b+a\bar{b}$	10	1

Precompute Technology Library: SA



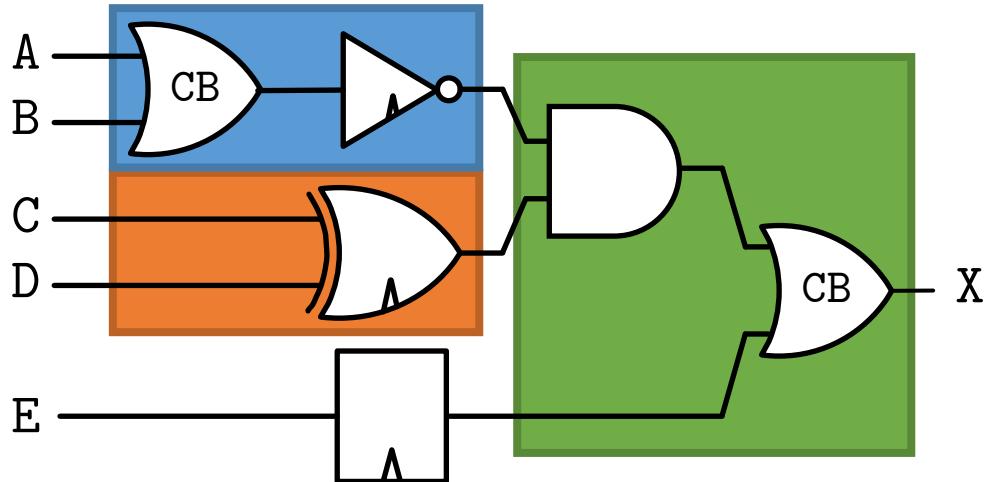
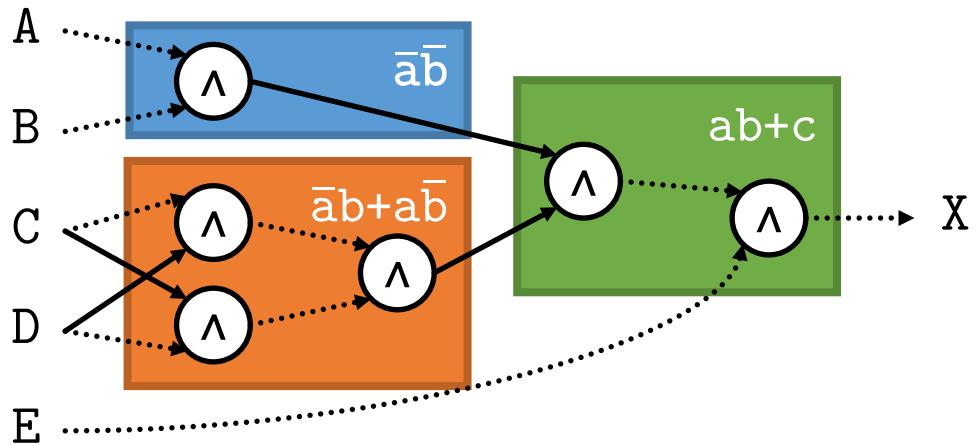
Function	Area	Delay
\bar{a}	9	1
\bar{b}	9	1
$\bar{a}\bar{b}$	16	1
$a\bar{b}$	22	1
$\bar{a}b$	22	1
ab	19	1
$a+b$	7	0
$\bar{a}+b$	22	1
$a+\bar{b}$	22	1
$\bar{a}+\bar{b}$	25	1
$ab+\bar{a}\bar{b}$		
$\bar{a}b+a\bar{b}$	10	1

Precompute Technology Library: AA



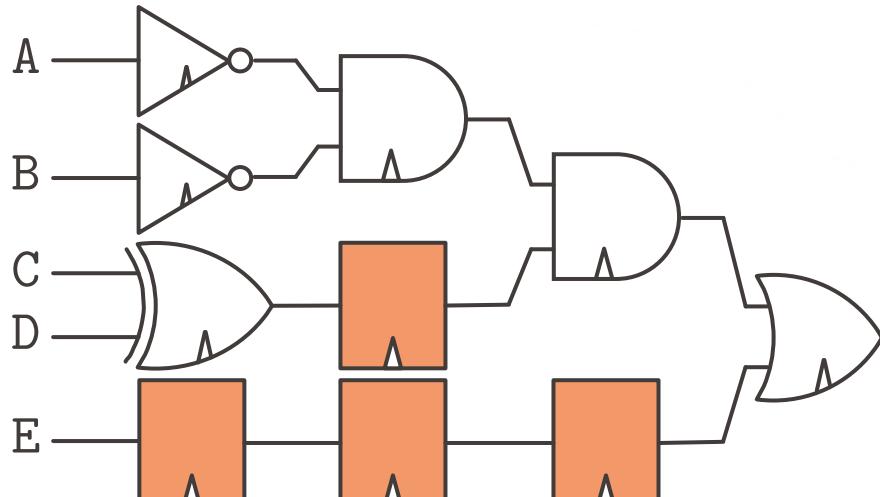
Technology Mapping

- Precompute library of optimal structures
 - Only done once
 - Can be reused
- Apply library to create gate-level netlist

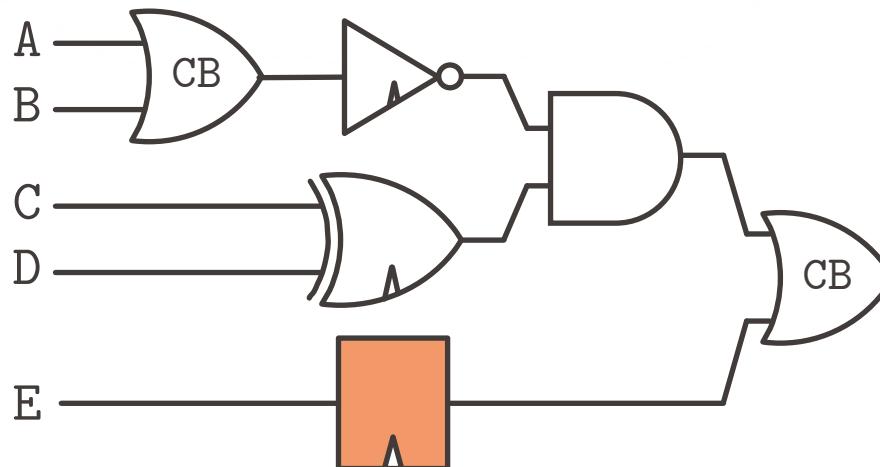


Depth Reduction

Conventional RSFQ
10 elements
4 cycles



Gate compounding RSFQ
6 elements
1 cycle



Results

- Enumerated all 65,536 4-input functions
- 48x 2.5GHz Intel Xeon E5-2680 CPU
- 256GB of RAM
- 52 hour runtime
- Mapped with `mockturtle` library [5]
- EPFL and ISCAS benchmarks
- On average vs state of the art [6]
 - 24% smaller area
 - 33% smaller depth

Benchmark	#DFF			#JJ			Delay			Runtime, s
	Baseline	Our work	Ratio	Baseline	Our work	Ratio	Baseline	Our work	Ratio	
sin	13,666	17,627	1.29	215,318	126,694	0.59	182	86	0.47	0.399
cavlc	522	987	1.89	16,339	15,098	0.92	17	11	0.65	0.009
dec	8	16	2.00	5,469	6,324	1.16	4	4	1.00	0.006
int2float	270	443	1.64	6,432	5,616	0.87	16	10	0.63	0.004
priority	9,064	14,754	1.63	102,085	95,370	0.93	127	125	0.98	0.013
c499	476	512	1.08	7,758	5,593	0.72	13	8	0.62	0.040
c880	774	1,179	1.52	12,909	8,359	0.65	22	13	0.59	0.013
c1908	696	799	1.15	12,013	5,553	0.46	20	11	0.55	0.025
c3540	1,159	1,556	1.34	28,300	22,231	0.79	31	18	0.58	0.034
c5315	2,908	3,727	1.28	52,033	33,524	0.64	23	13	0.57	0.091
c7552	2,429	4,744	1.95	48,482	28,900	0.60	19	13	0.68	0.115
Average			1.53			0.76			0.67	

[5] <https://github.com/lsils/mockturtle>

[6] G. Pasandi and M. Pedram, "PBMMap: A Path Balancing Technology Mapping Algorithm for Single Flux Quantum Logic Circuits," in *IEEE TASC* 2019

- RSFQ can bring enormous improvements in power and speed
- Gate compounding simplifies path balancing of RSFQ circuits
 - Smaller logic depth
 - More functionality in fewer clock cycles
- Enumerate database of optimal small-scale structures up to 4 inputs
 - Apply to construct large scale networks
- Smaller area and depth w.r.t. state-of-the-art

Thank you!