A Transduction-based Framework to Synthesize RSFQ Circuits

Shigeru Yamashita (NAIST), Katsunori Tanaka (NEC),
Hideyuki Takada (Kyoto Univ.),
Koji Obata and Kazuyoshi Takagi (Nagoya Univ.)

Outline of the Talk

A Transduction-based Framework to Synthesize RSFQ Circuits

- Background & Preliminaries
  - RSFQ Logic Circuits
  - RSFQ Logic Elements
- Framework for logic design of RSFQ circuits
  - Transduction Method
- Experimental Results
- Concluding Remarks

Background: Next Generation Tech.

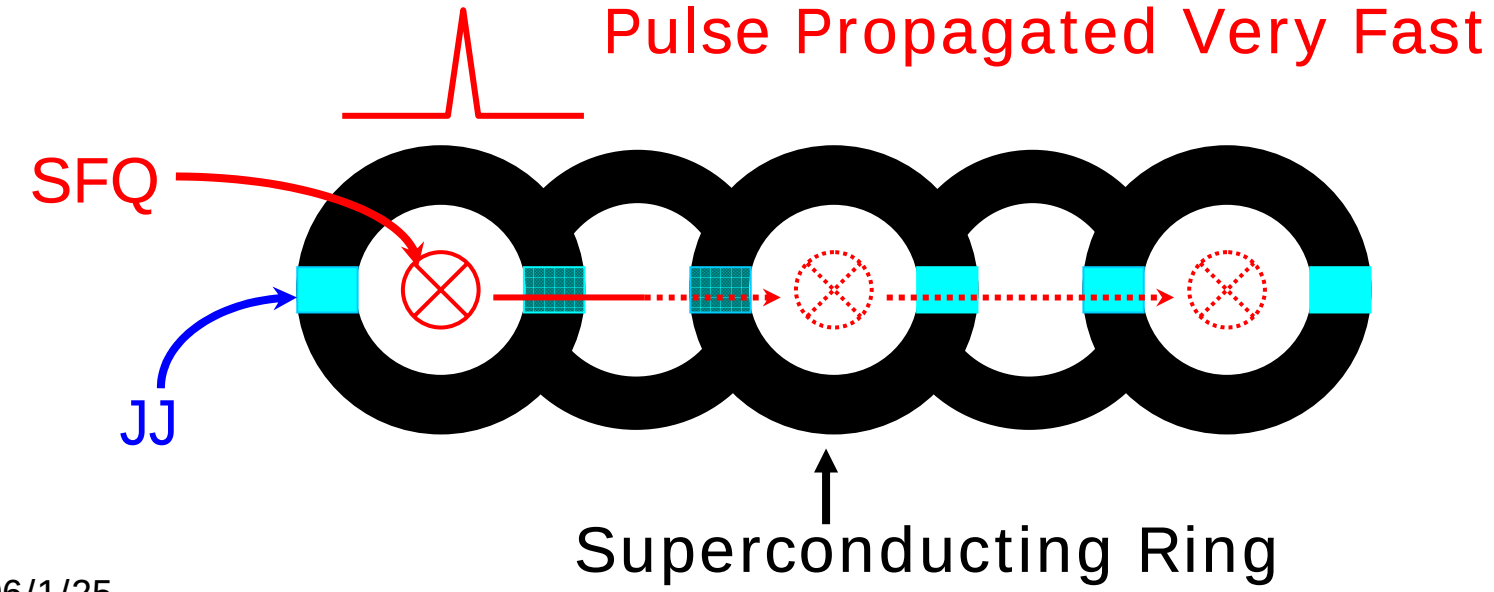
- Reaching the Limit of Conventional Semiconductor Technology.
  - Increasing Power Dissipation
  - Increasing Relative Delay
  - Difficult Implementation, etc.
- Beyond Conventional Semiconductor Technology.
  - SET: Single Electron Transistor
  - RSFQ: Rapid Single Flux Quantum
  - QC: Quantum Computer

Background: RSFQ Circuits (1/2)

- Based on Superconductor Technology
- Very Low Energy for Representing a Single Bit
- Very High Speed
  - Near to the Light Speed
- Proven Technology
  - Experimental Implementation of a Network Switch Circuit
- High Cost for Cooling

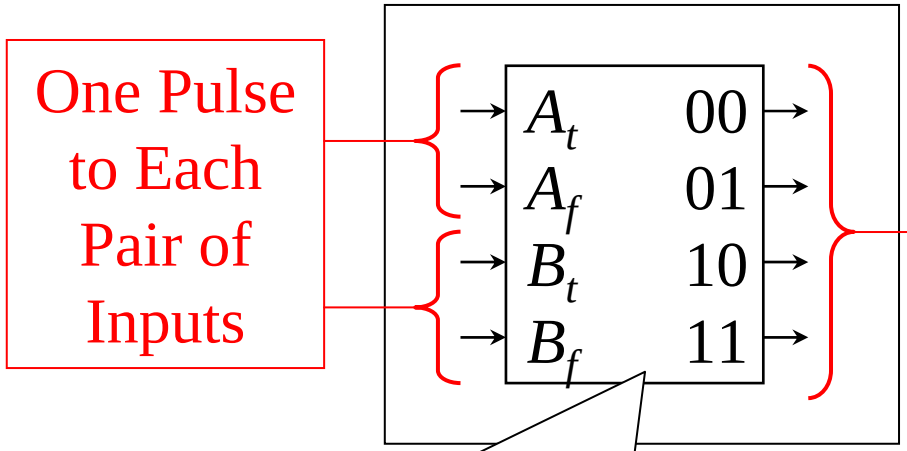
Background: RSFQ Circuits (2/2)

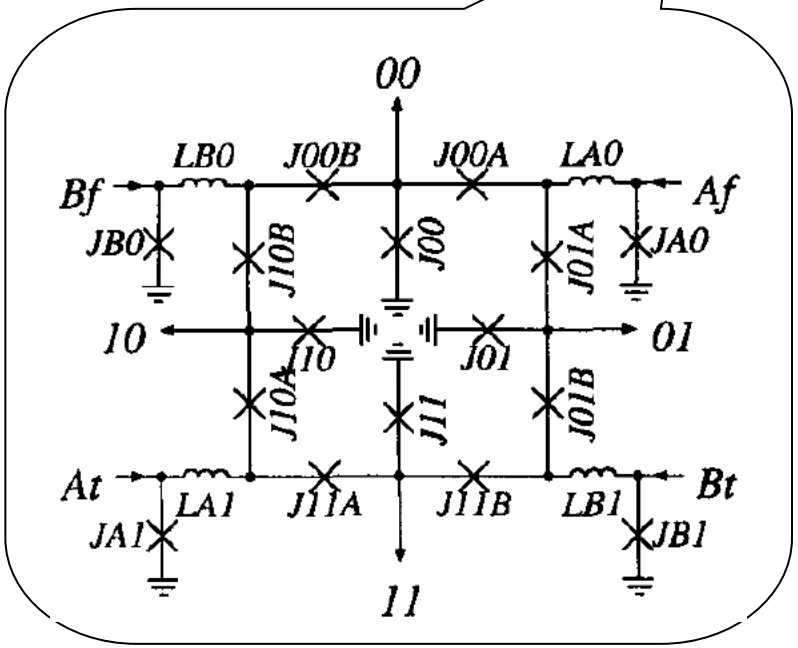
- Pulse Representation for Information Propagated
  - Single Flux Quantum = Minimal Unit of Flux
- Basic Element = Superconducting Ring + Josephson Junction (JJ)
- JJ: Switch to Bring a Pulse Into or Out of the Ring.



Background: RSFQ Logic Primitive (1/2)

2x2-Join

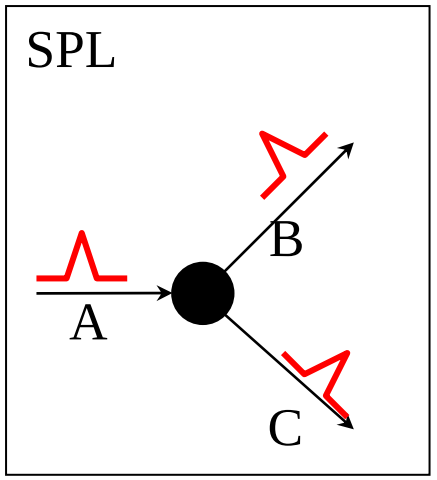


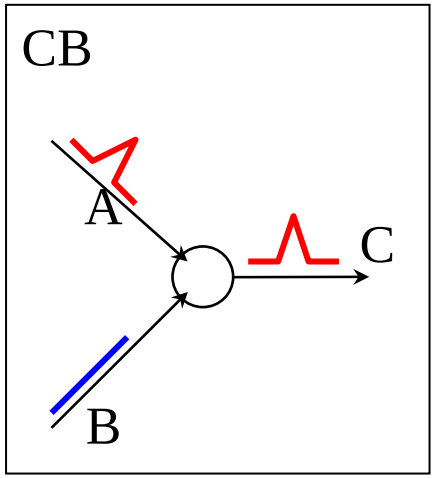


Inputs				Outputs			
A_t	A_f	B_t	B_f	00	01	10	11
⎍	_	⎍	_	_	_	_	⎍
⎍	_	_	⎍	_	_	⎍	_
_	⎍	⎍	_	_	⎍	_	_
_	⎍	_	⎍	⎍	_	_	_

Background: RSFQ Logic Primitive (2/2)

- In Our Framework:
  - SPL (Splitter)
    - Split One Pulse to Two
  - CB (Confluence Buffer)
    - Converging Two Paths
    - No Simultaneous Input of Two Pulses
  - 2x2-Join
    - Dual-Rail, Two-Input
    - One-of-Four Output





Our Motivation

How to design logic circuits by these primitives?

Proposed Framework

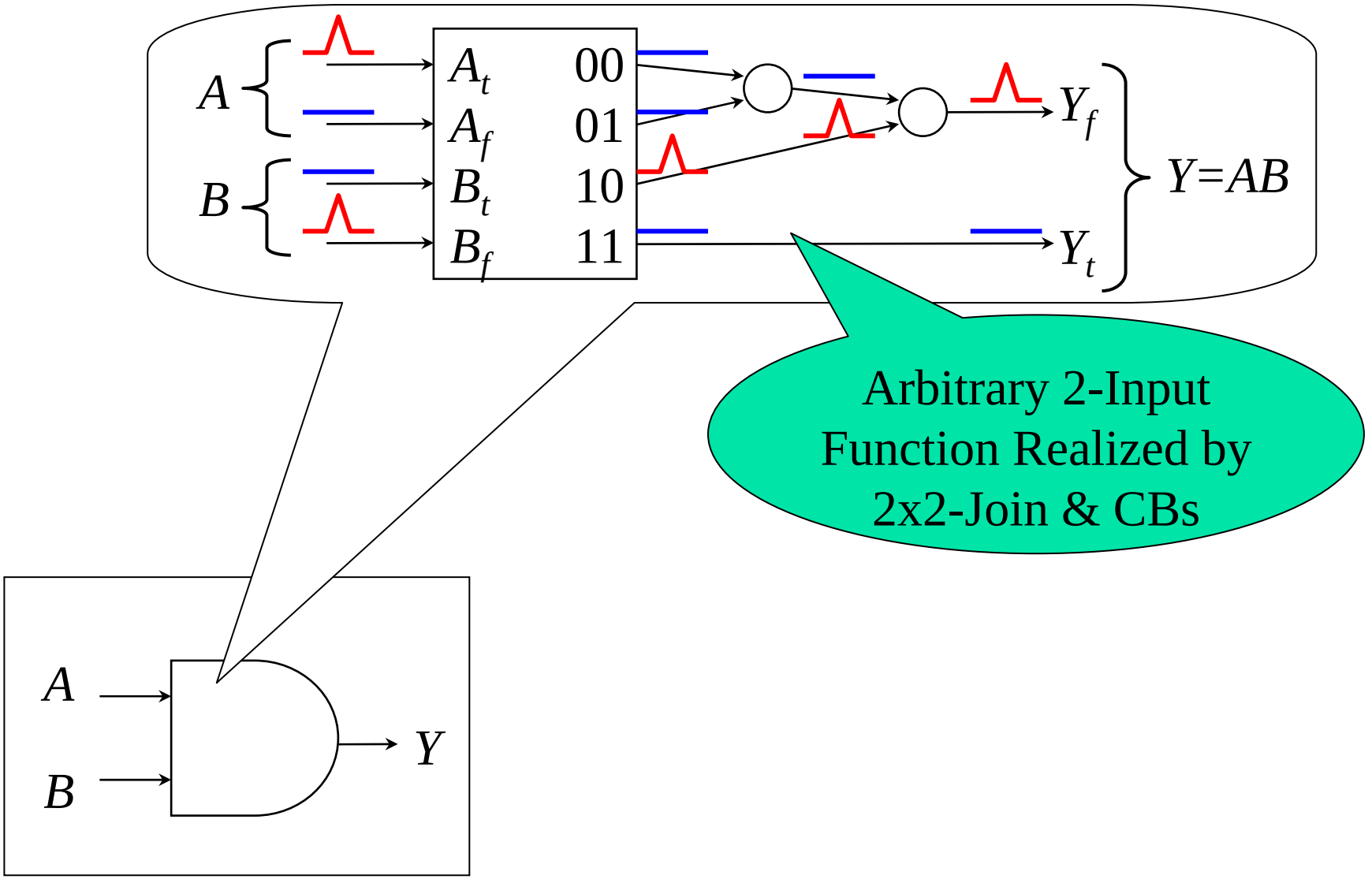
- Our Policies
  - Modification of Conventional Logic Design
  - Dual-Rail

- Our Framework
  - Application of Conventional 2-Input Mapper
  - 2x2-Join+CB+SPL → 2-AND/XOR Cell
  - Optimization by the Transduction Method

Dual-Rail Logic Design

- Pulse Representation of bits
  - Single-Rail is Insufficient.
- Dual-Rail
  - Data Line + Clock Line
    - Difficulty on Timing
  - 0-Data Line + 1-Data Line
    - Paired Data Lines (0-Data, 1-Data)
    - (Pulse, No) = Valid-0
    - (No, Pulse) = Valid-1
    - (No, No) = Invalid
    - (Pulse, Pulse) = Prohibited.
    - Can be realized easily by 2x2-Joins

Dual-Rail, 2-Input Cell

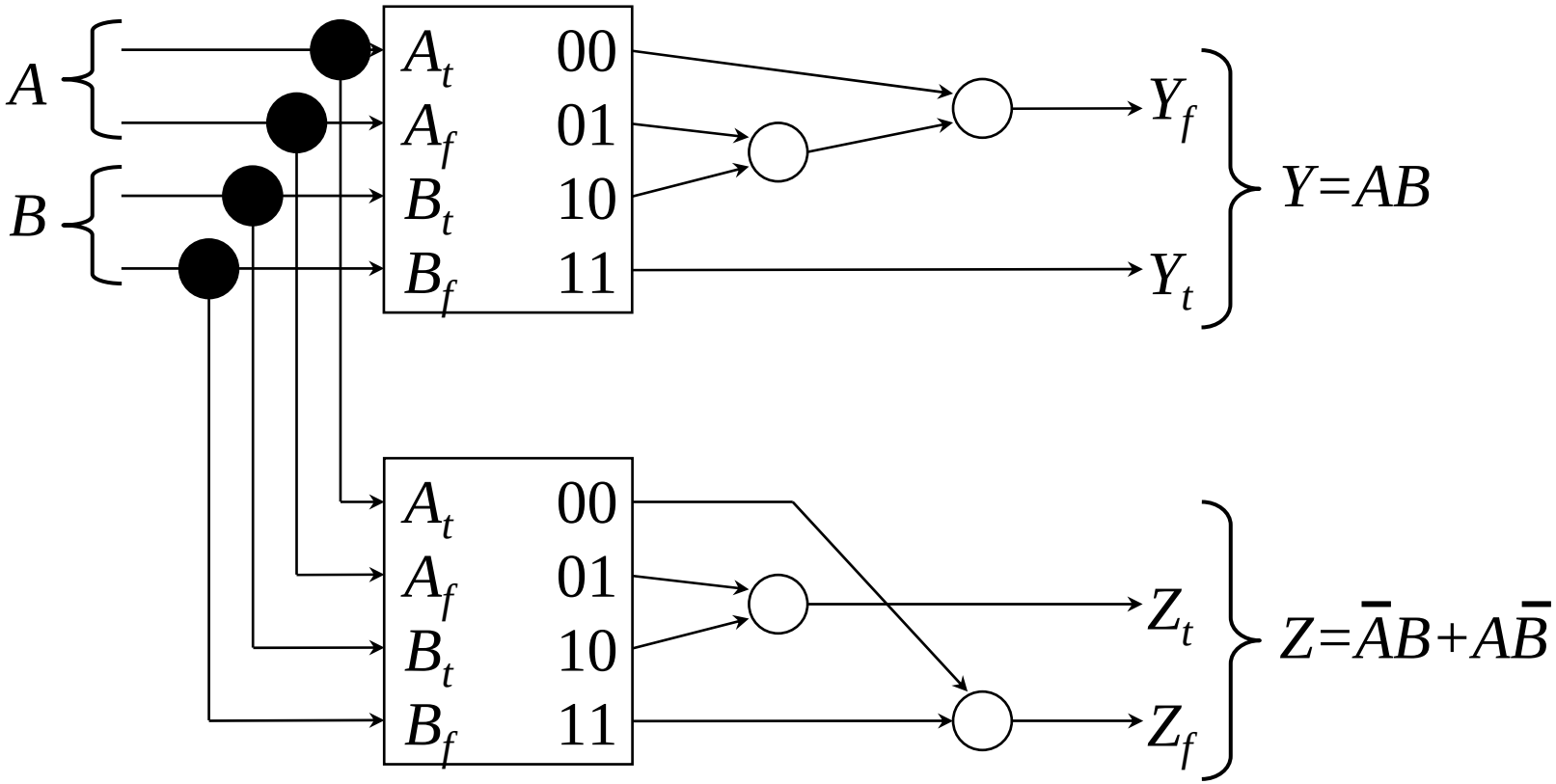


RSFQ Logic Design Framework

1. Synthesize 2-input node (= 2x2-Join+CB) Circuit
   - By Conventional Method
   - With 2-input Mapping
2. Logic Optimization by Transduction Method

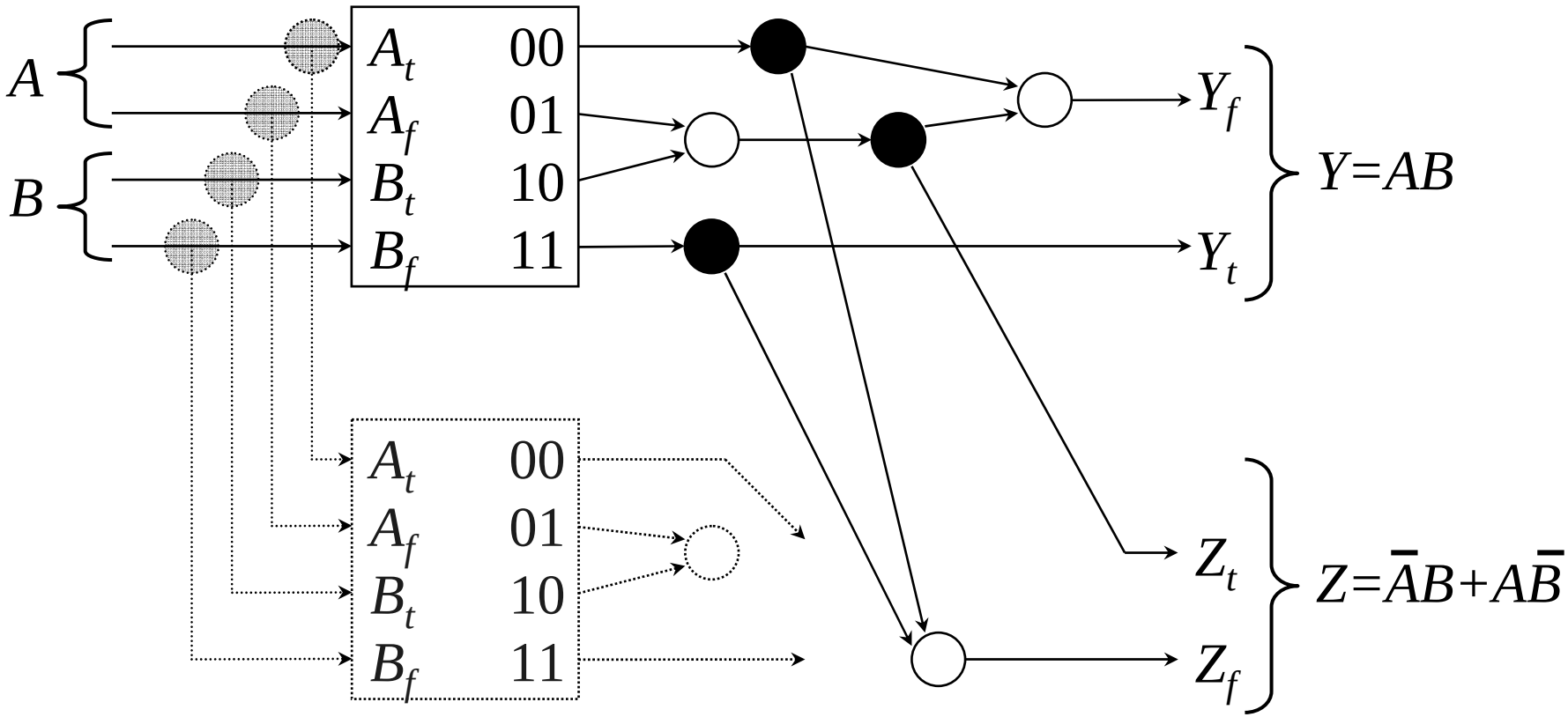
- RSFQ Logic Design
  - Efficient Sharing 2x2-Joins in Logic Optimization
  - Multi-Output Cell with SPLs

Multi-Output 2-Input Cell (1/2)



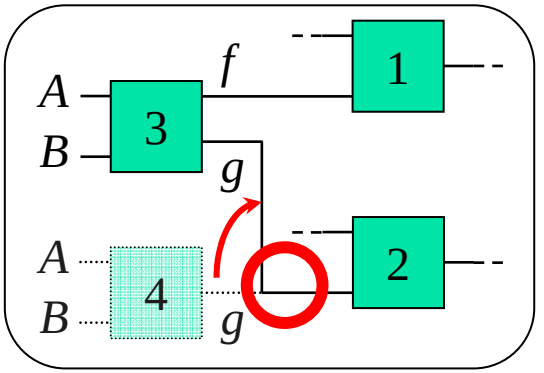
Multi-Output 2-Input Cell (2/2)

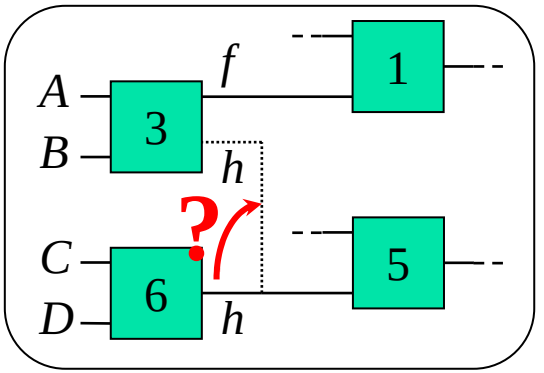
Element Reduction by Sharing:
2x2-Join: -1, CB: -1, SPL: -1



Generalized Sharing

- Many Sharings → Large Reduction
- Determine whether Possible to Share.
  - Completely Common Inputs → Easy
  - One or Two Different Inputs → ?





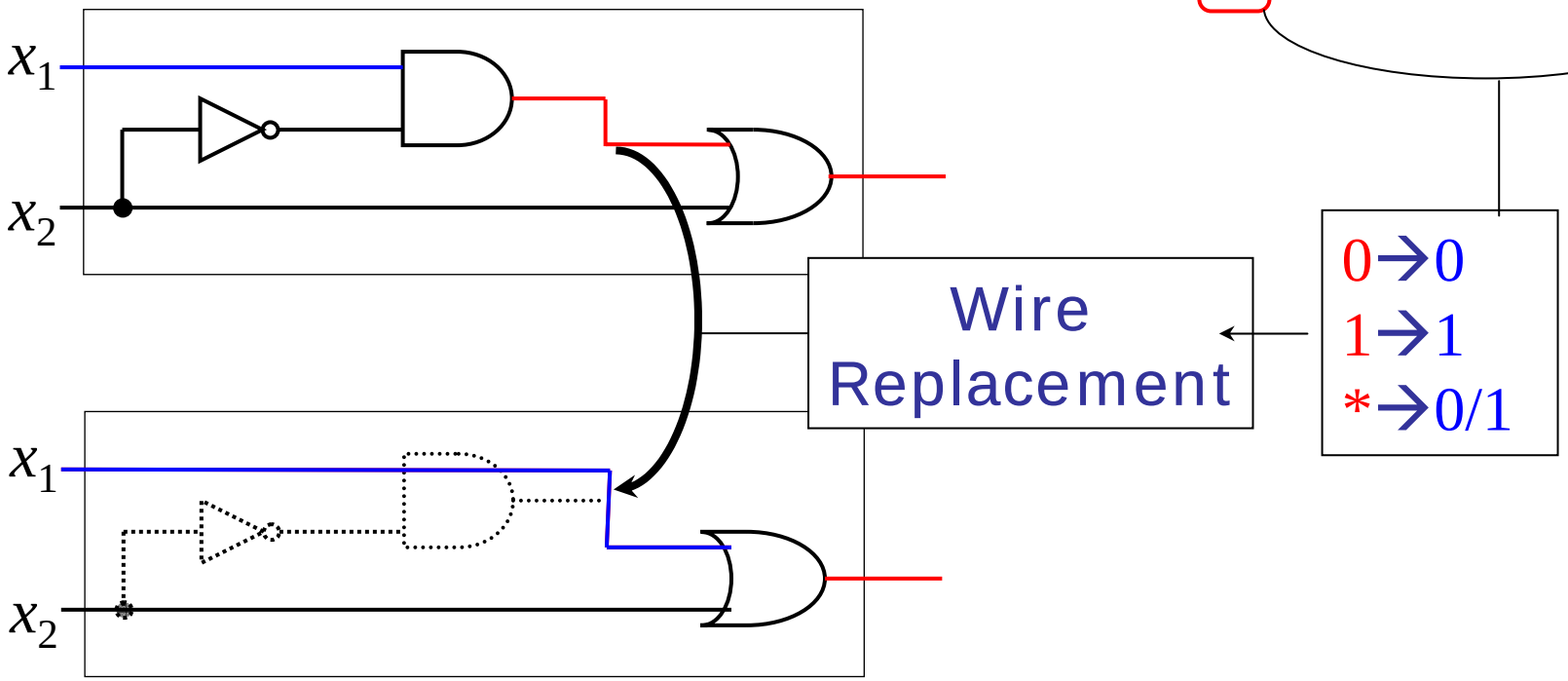
- Use of the Transduction Method
- Don't-care-based Logic Optimization
- Useful for Finding Sharings of Cells with Uncommon Inputs

The Transduction Method (Permissible Functions)

x_1x_2	$x_1.\overline{x_2}$	x_2	
0 0	0	0	0
0 1	0	1	1
1 0	1	0	1
1 1	0	1	1

x_1x_2		x_2	
0 0	0	0	0
0 1	*	1	1
1 0	1	0	1
1 1	*	1	1

x_1x_2	x_1
0 0	0
0 1	0
1 0	1
1 1	1



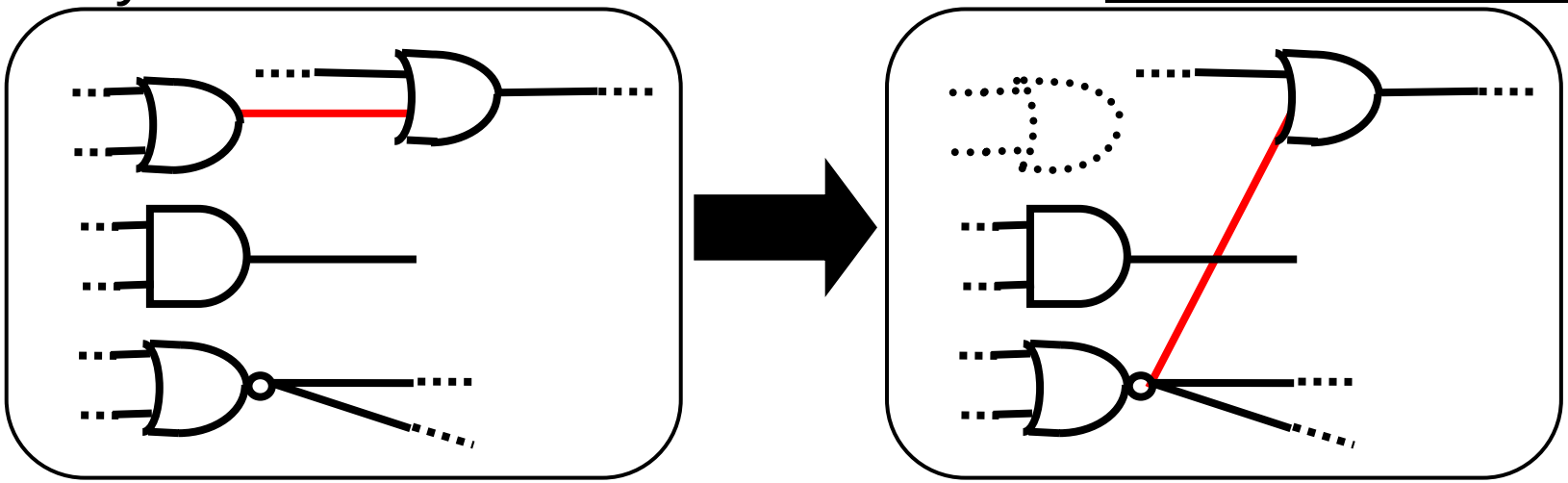
The Transduction Method (Wire Replacement)

Repeat until No Reduction

1. Extracting PFs
   - Don't-Care condition for alternative wire
2. Replacement of Wires Satisfying the Conditions Extracted in 1.

Gate Count Reduction by Wire Replacement:

Many Candidates: the Gate with the **Maximal Fanout**

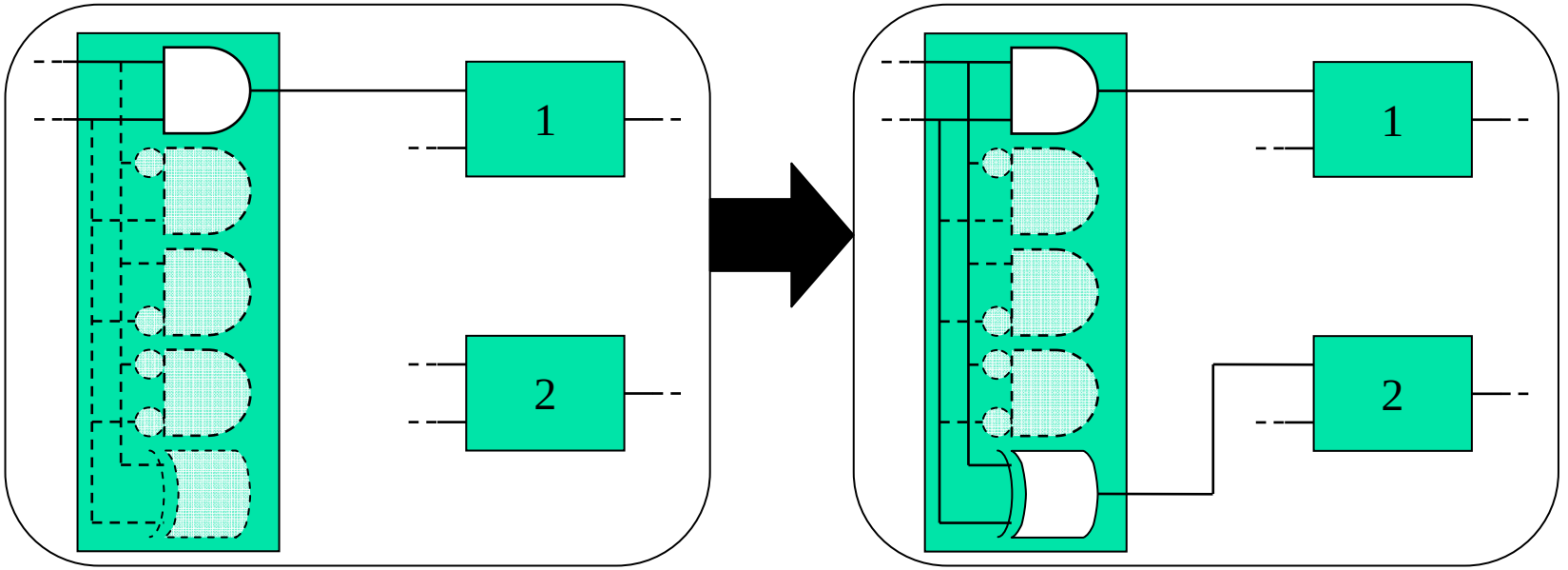


The Transduction Method for RSFQ (1/2)

- Initial circuit: 2-input gate circuit
- The same input gates can be considered as a group

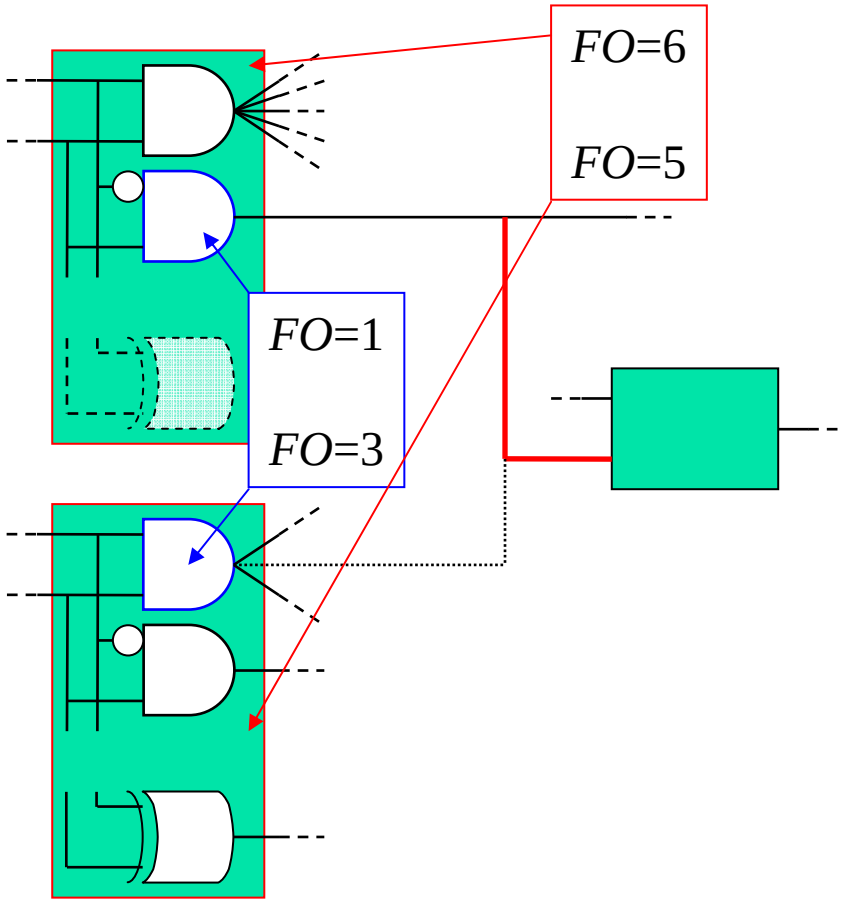
→ 2-AND/XOR Cell

  - Can realize arbitrary 2-input function
  - 4×AND (w/ NOTs)
  - 1×XOR



The Transduction Method for RSFQ (2/2)

- Removing a cell if all the gates has lost fan-out.
- Total # of fanouts of the virtual gates are used for the wire selection priority



Experimental Results

- Initial Circuit: by SIS as 2-input node circuit
- 32 % Cell Count Reduced by Proposed Method

			Initial			After Trans.					
Circuits	PI	PO	2x2	Conn.	Lev.	2x2	Conn.	Lev.	Time (s)	Shared	(%)
C1355	41	32	234	468	17	174	348	14	1.17	6	3.4
C7552	207	108	1504	3059	32	994	2049	32	122.14	136	13.7
alu2	10	6	386	773	42	236	473	28	3.35	41	17.3
alu4	14	8	667	1334	43	479	958	36	11.26	84	17.5
cmb	16	4	44	88	6	25	51	5	0.01	4	16
dalu	75	16	1172	2344	36	809	1618	18	36.59	84	10.4
f51m	8	8	113	227	10	64	129	9	0.11	10	15.6
i8	133	81	1260	2520	19	971	1942	15	155.37	127	13.1
lal	26	19	88	177	8	61	123	8	0.06	7	11.5
adder	33	17	96	192	48	64	128	48	0.08	16	25
t481	16	1	1690	3380	20	1073	2146	19	110.66	57	5.3
term1	34	10	259	519	16	116	234	11	0.68	15	12.9
ttt2	24	21	182	364	10	127	254	10	0.41	19	15
x3	135	99	724	1448	14	548	1096	12	14.19	41	7.5
z4ml	7	4	44	88	9	12	25	8	0.01	6	50
Average			100%	100%	100%	68.0%	68.2%	82.7%	6.08		11

Concluding Remarks

- RSFQ Logic Design Framework
  - Initial circuit: Synthesize a 2-input node Circuit
  - Optimize by the Transduction Method for Multi-Output Cells
  - 2-AND/XOR → 2x2-Join+CB+SPL
- Conventional Logic Design Methods are applicable.
- Size Reduction by Sharing 2x2-Joins: 32.0% on Avg.
- Future work
  - Combination with BDD-based Synthesis
  - Extension for Other Types of RSFQ Logic Element