REVERSIBLE LOGIC CIRCUITS

Pawel Kerntopf
Institute of Computer Science
Warsaw University of Technology
Warsaw, Poland

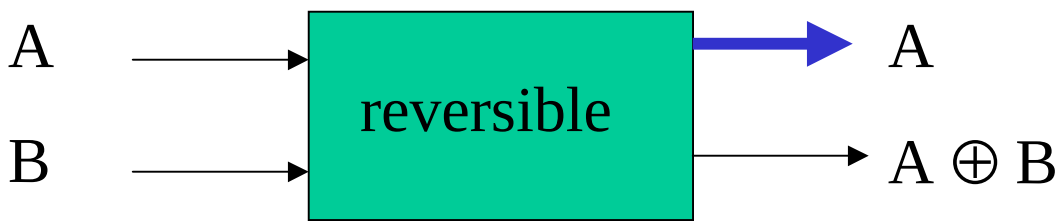
OUTLINE

- General issues
- Basic notions
- Reversible gates
- Mathematical results
- Synthesis
- Open problems
- Conclusions

Information is Physical

- **Is some minimum amount of energy required <u>per one computation step</u>?**

- Rolf Landauer, 1961. Whenever we use a logically irreversible gate we dissipate energy into the environment.



Information loss = energy loss

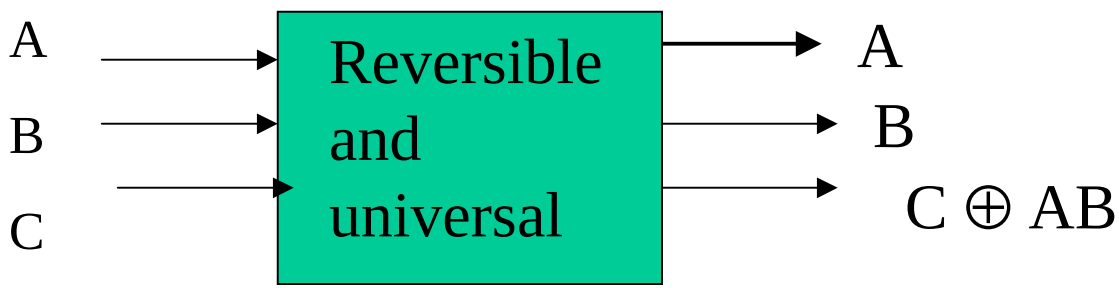
- The loss of information is associated with laws of physics requiring that *one bit of information lost* dissipates k T ln 2 of energy,
  - where k is ***Boltzmann' constant***
  - and T is the temperature of the system.
- Interest in **reversible computation** arises from the desire to *reduce heat dissipation*, thereby allowing:
  - higher densities
  - higher speed

R. Landauer, "Irreversibility and Heat Generation in the Computing Process", IBM J. Res. & Dev., 1961.

Solution = Reversibility

- **Charles Bennett, 1973:** There are no unavoidable energy consumption requirements per step in a computer.
- Power dissipation of reversible circuit, under ideal physical circumstances, **is zero**.

- **Tomasso Toffoli, 1980:** There exists a reversible gate which could play a role of a universal gate for reversible circuits.



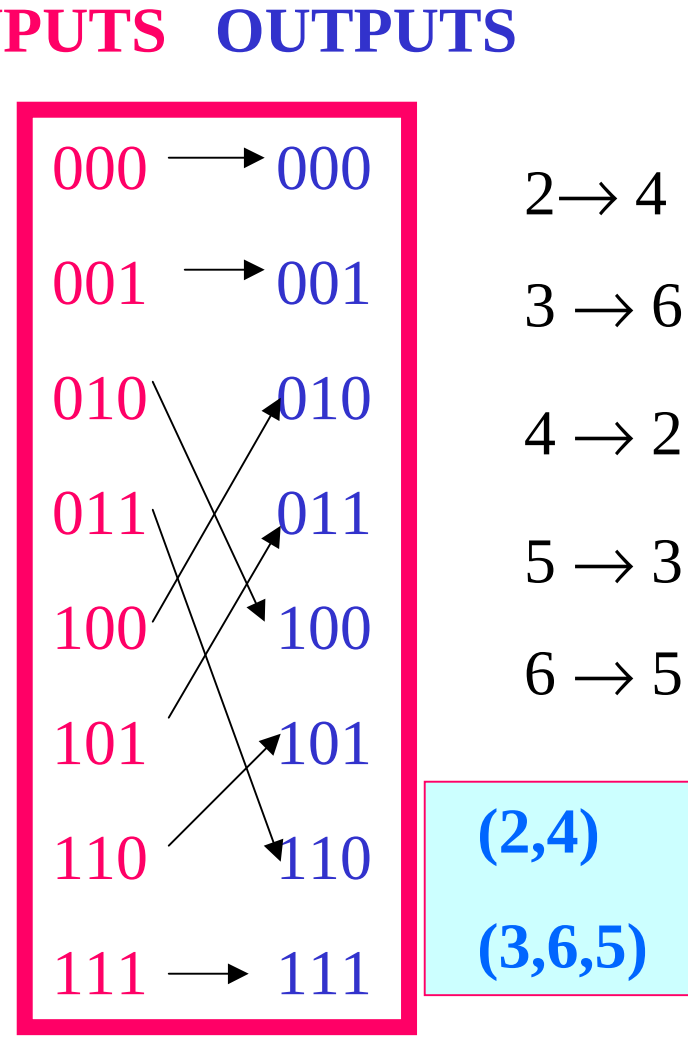
Reversible computation

- Landauer/Bennett: **all operations required in computation** could be performed in a reversible manner, thus dissipating no heat!
- **The first condition** for any deterministic device to be reversible is that its input and output be uniquely retrievable from each other - then it is called logically reversible.
- **The second condition:** a device can actually run backwards - then it is called physically reversible.
- and the second law of thermodynamics guarantees that **it dissipates no heat.**

Billiard Ball Model

Reversible logic

Reversible are circuits (gates) that have *one-to-one* ***mapping*** *between vectors of inputs and outputs; thus the vector of input states can be always reconstructed from the vector of output states.*



Balanced Functions

- 10 out of 20 permutation equivalence classes of 3-valued balanced functions (70 functions altogether)

Class	**# functions**	**Representative**
1	*3*	x
2	*3*	$x \oplus y = x \ XOR \ y$
3	*3*	$x \oplus yz$
4	*1*	$x \oplus y \oplus z$
5	*6*	$x \oplus y \oplus xz$
6	*6*	$x \oplus xy \oplus xz$
7	*1*	$xy \oplus xz \oplus yz$
8	*3*	$x \oplus y \oplus z \oplus xy$
9	*6*	$x \oplus y \oplus xy \oplus xz$
10	*3*	$x \oplus y \oplus xy \oplus xz \oplus yz$

Reversible Gates versus Balanced Functions

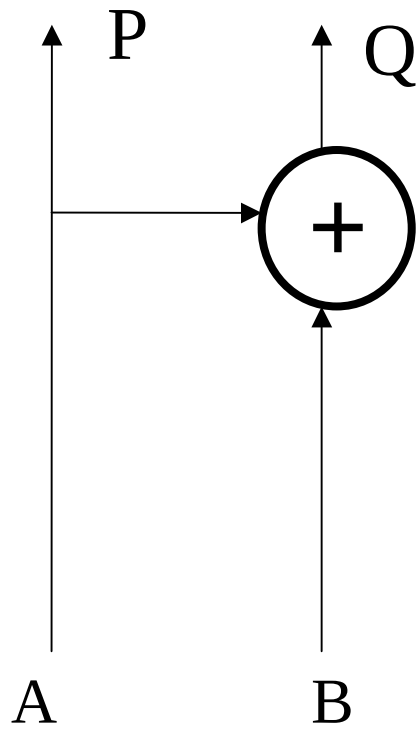
- There exist 2^{24} = 16,777,216 different truth tables with 3 inputs and 3 outputs.
- The number of triples of **balanced functions** is equal to 70 * 70 * 70 = 343 000
- However, the number of reversible (3,3)-gates is much smaller: 8! = 40320
  - not every pair of balanced functions of 3 variables may appear in a reversible (3,3)-gate

Extension of the table

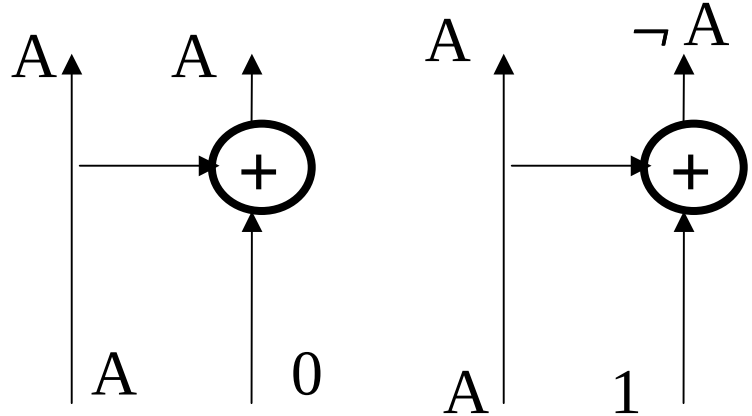
A	B	C	D	P	Q	R	S
0	0	0	0	0	0		
0	0	0	1	1	0		
0	0	1	0	1	0		
0	0	1	1	0	1		
0	1	0	0	1	0		
0	1	0	1	0	1		
0	1	1	0	0	1		
0	1	1	1	1	1		
1	0	0	0				
1	0	0	1				
1	0	1	0				
1	0	1	1				
1	1	0	0				
1	1	0	1				
1	1	1	0				
1	1	1	1				

- **Balanced functions must be used**
- **We want to extend the table to make all its output rows to be permutations of input rows**
- **This sets certain constraints on selection of entries leading to *garbage outputs***

Feynman Gate



- When ***A = 0*** then ***Q = B,*** when ***A = 1*** then ***Q = B'***.
- Every linear reversible function can be built by composing only 2*2 Feynman gates and inverters
- With ***B=0*** Feynman gate is used as a **fan-out gate.** (**Copying gate**)



Fredkin Gate

- Fredkin Gate is a fundamental concept in *reversible and quantum computing*.
- Every Boolean function can be build from 3 * 3 Fredkin gates:

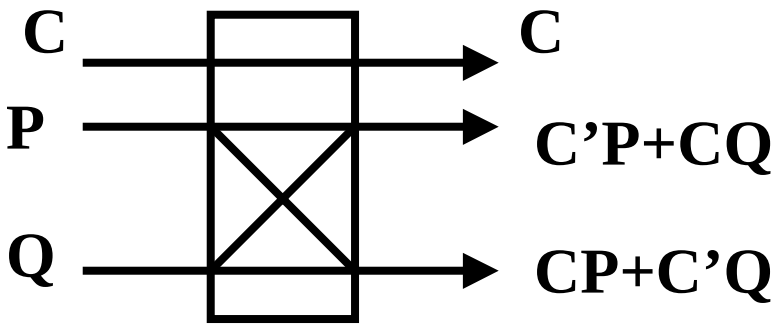
  P = A,

  Q = if A then C else B,

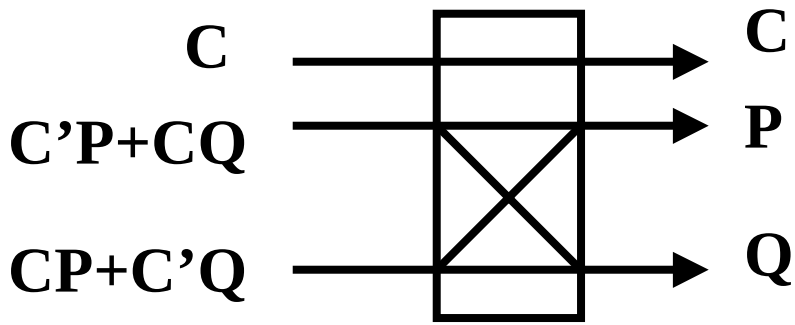
  R = if A then B else C.

Useful Notation for Fredkin Gate

Fredkin Gate



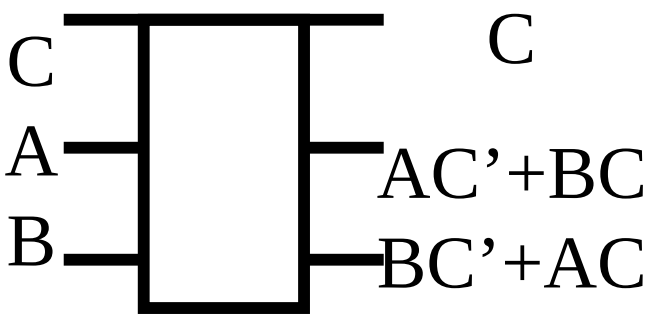
Inverse Fredkin Gate

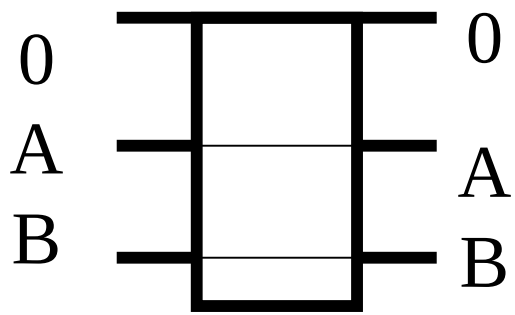


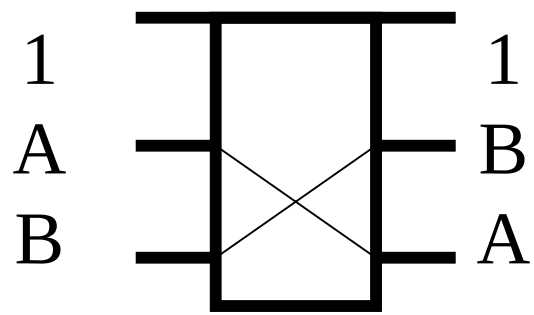
In this gate the input signals P and Q are routed to the same or exchanged output ports depending on the value of control signal C

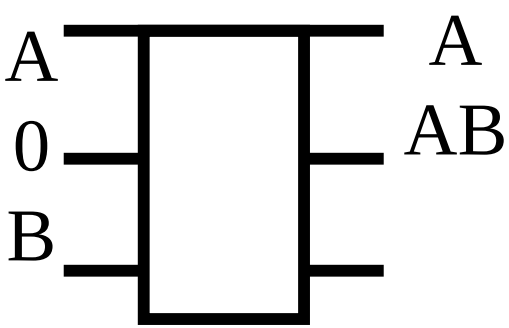
Fredkin gate is conservative and it is its own inverse

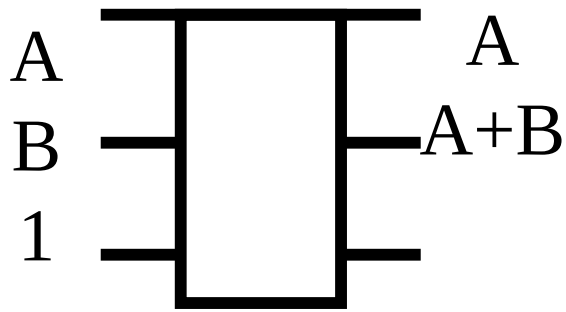
Operation of the Fredkin gate

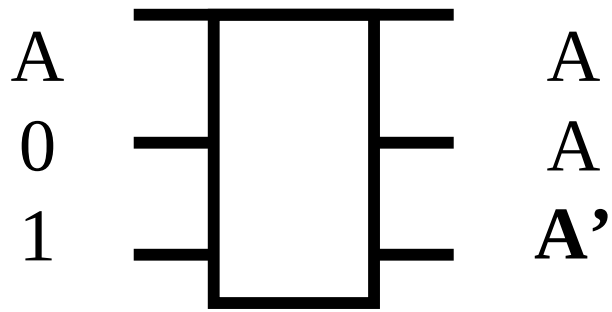




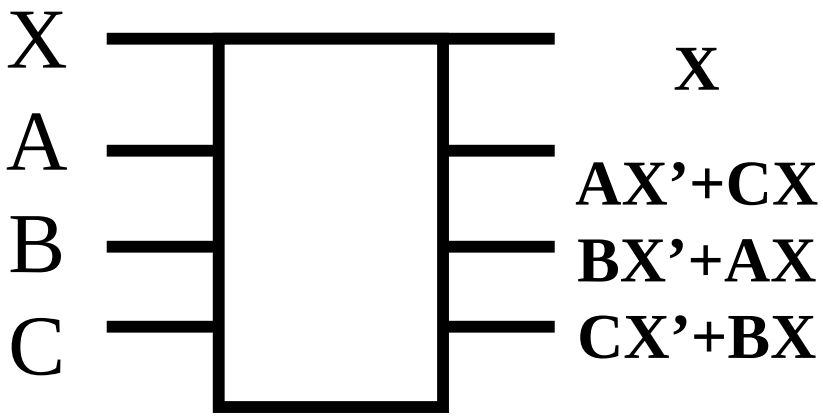


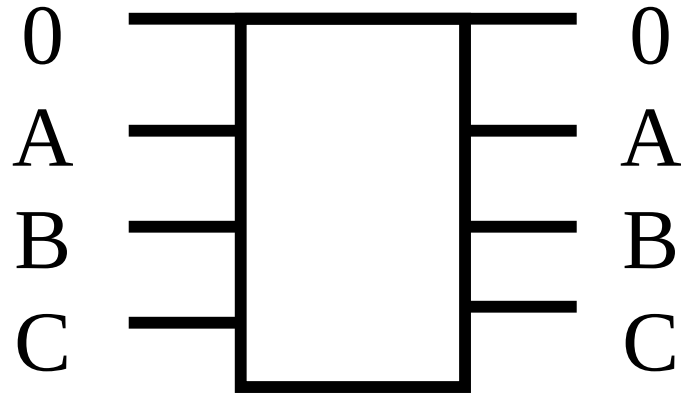


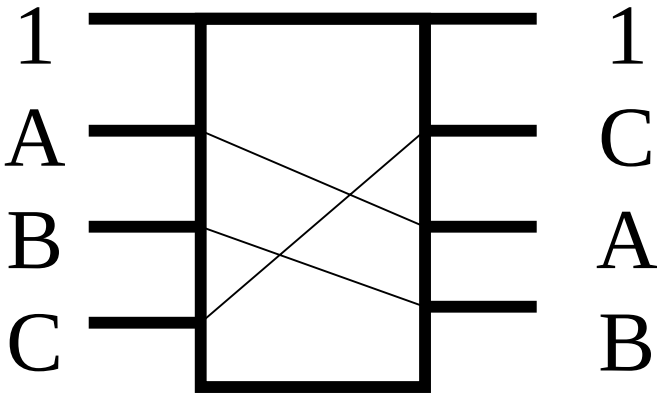


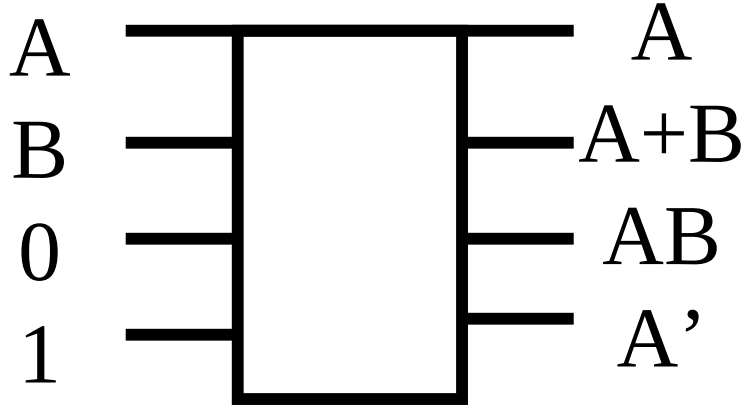


A 4-input Fredkin gate





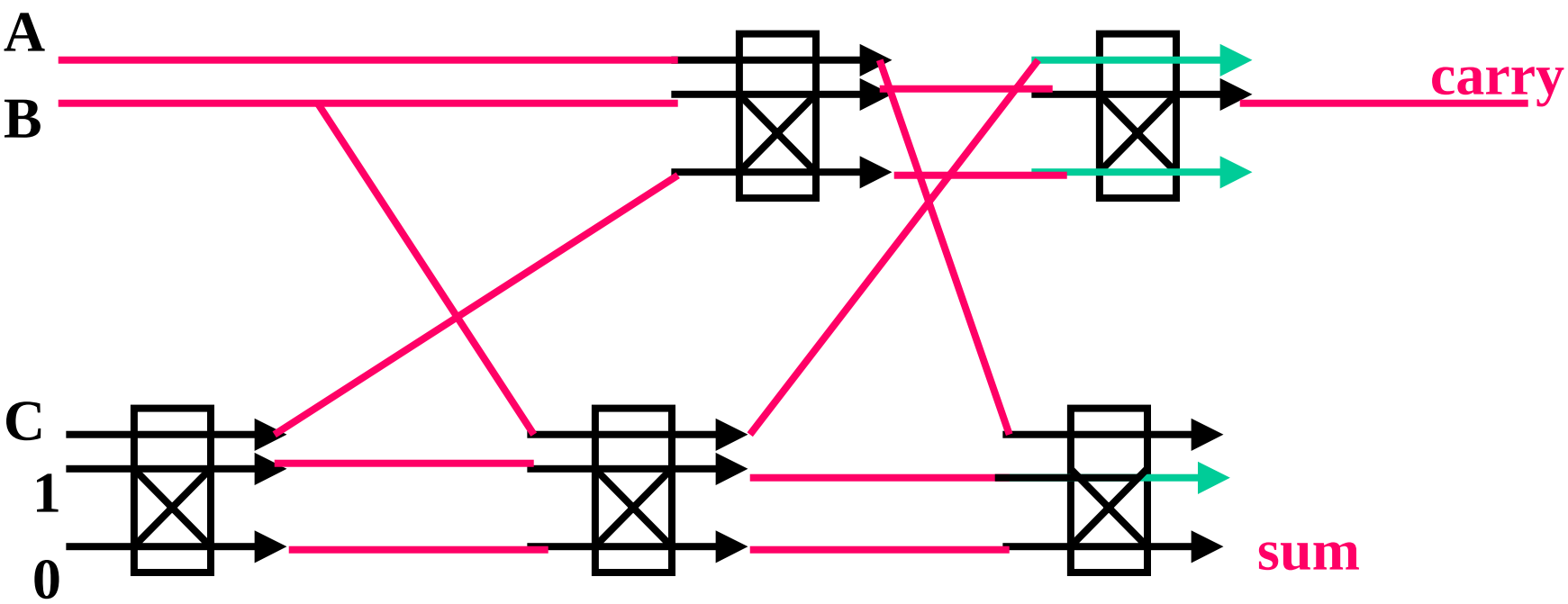




Reversible logic: Garbage

- A reversible circuit without constants on inputs realizes on all outputs only balanced functions.
- Therefore, reversible circuit can realize unbalanced functions only with additional inputs and ***garbage*** outputs.

Minimal Full Adder Using Fredkin Gates

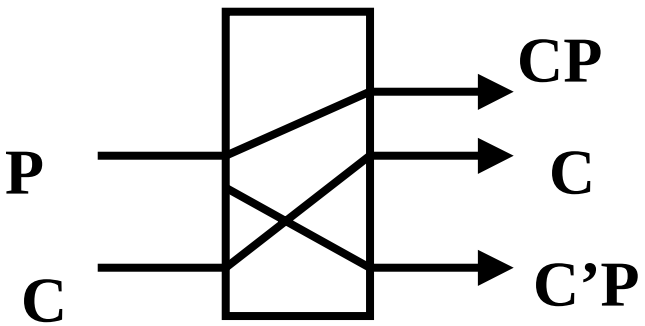


In this gate the input signals P and Q are routed to the same or exchanged output ports depending on the value of control signal C

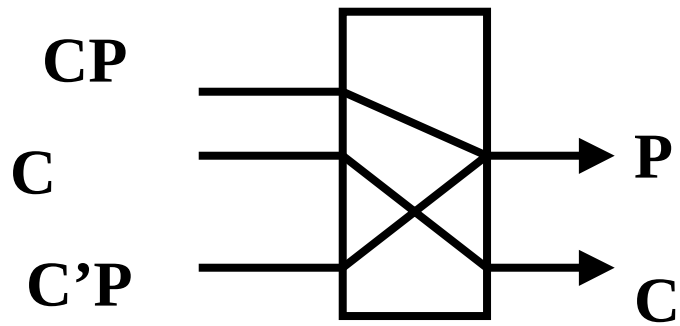
3 garbage bits

Switch Gate

Switch Gate

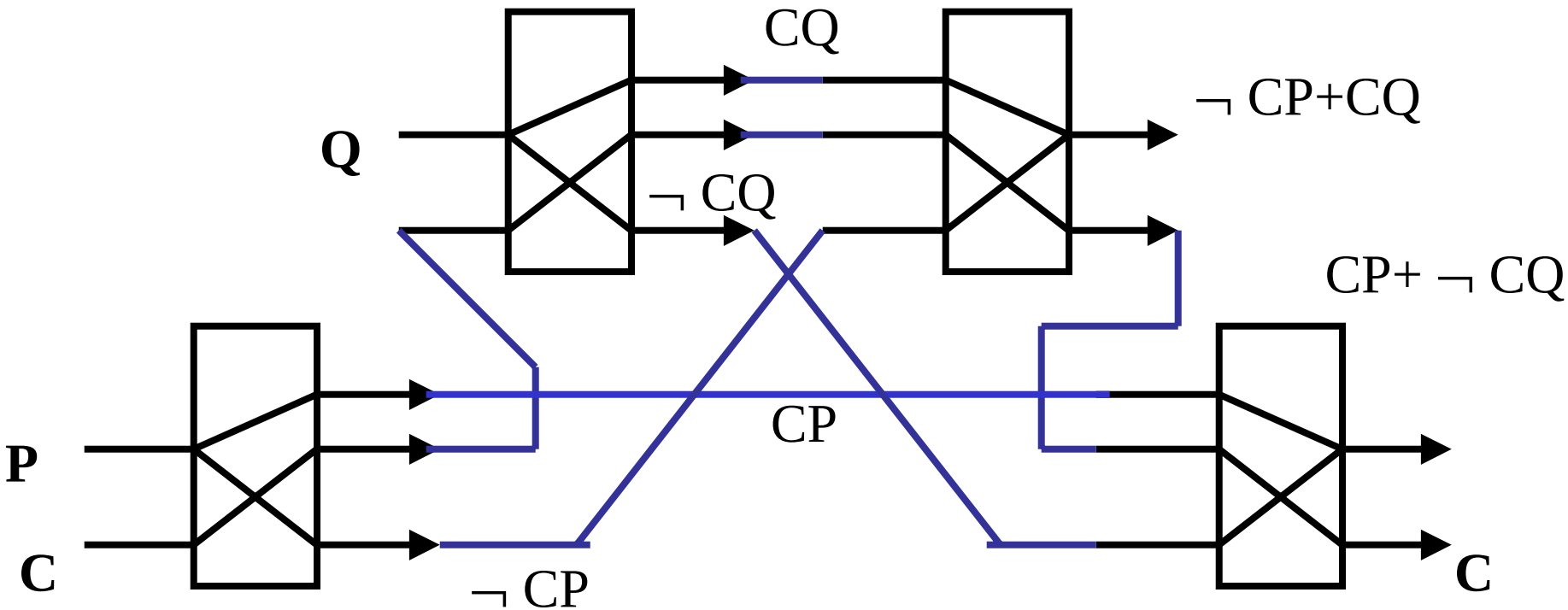


Inverse Switch Gate



In this gate the input signal P is routed to one of two output ports depending on the value of control signal C

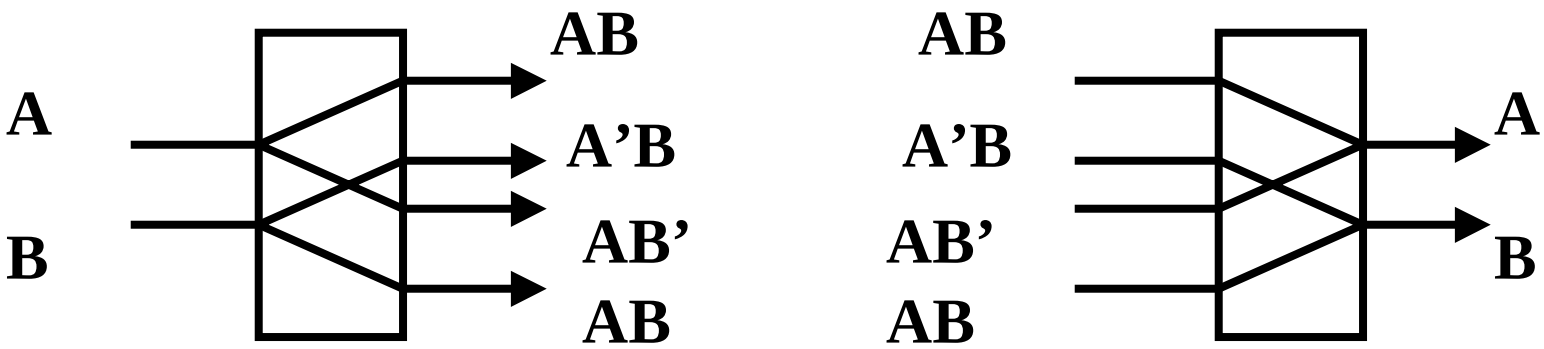
Fredkin Gate from Switch Gates



Interaction Gate

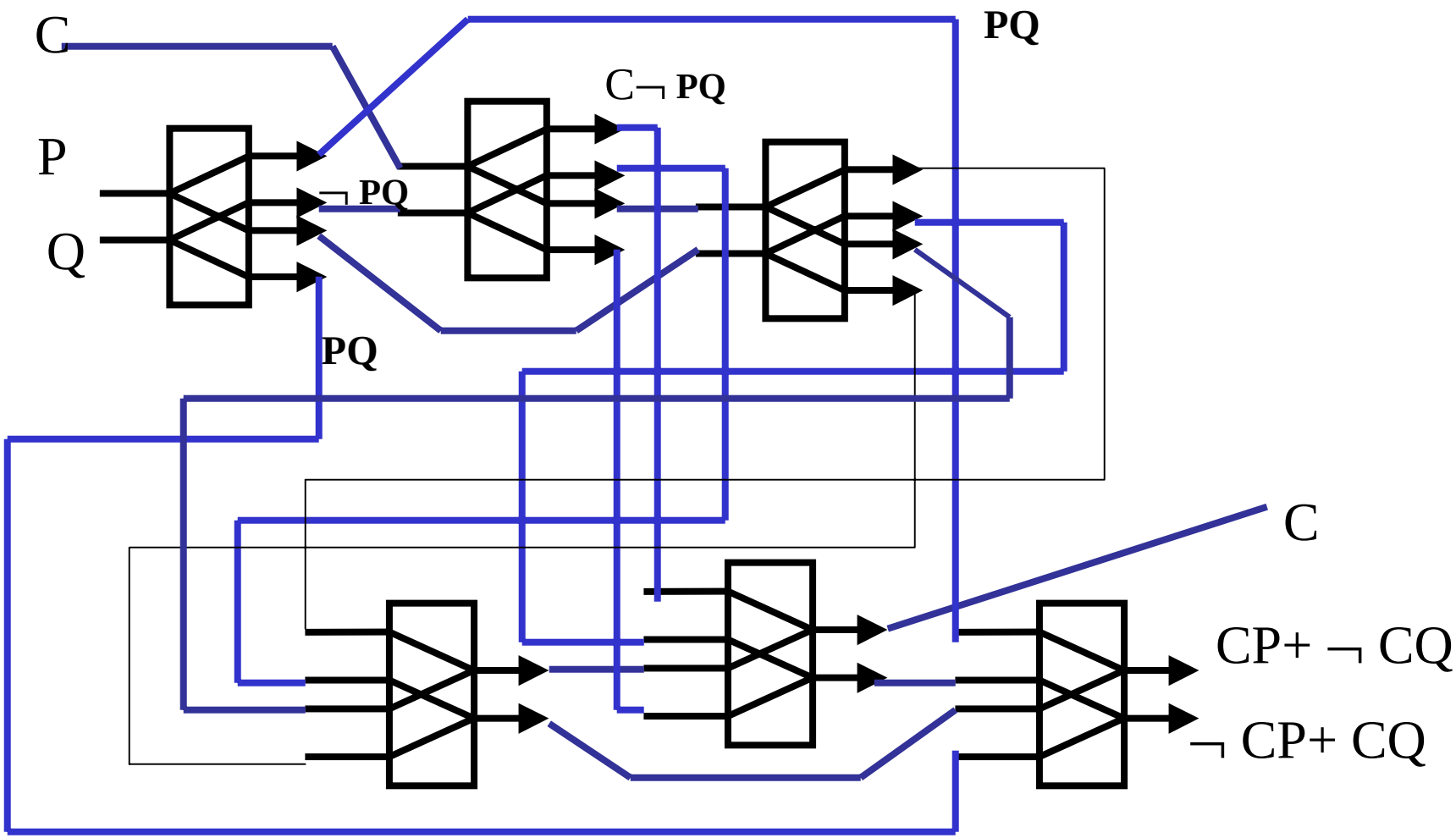
Interaction Gate

Inverse interaction Gate

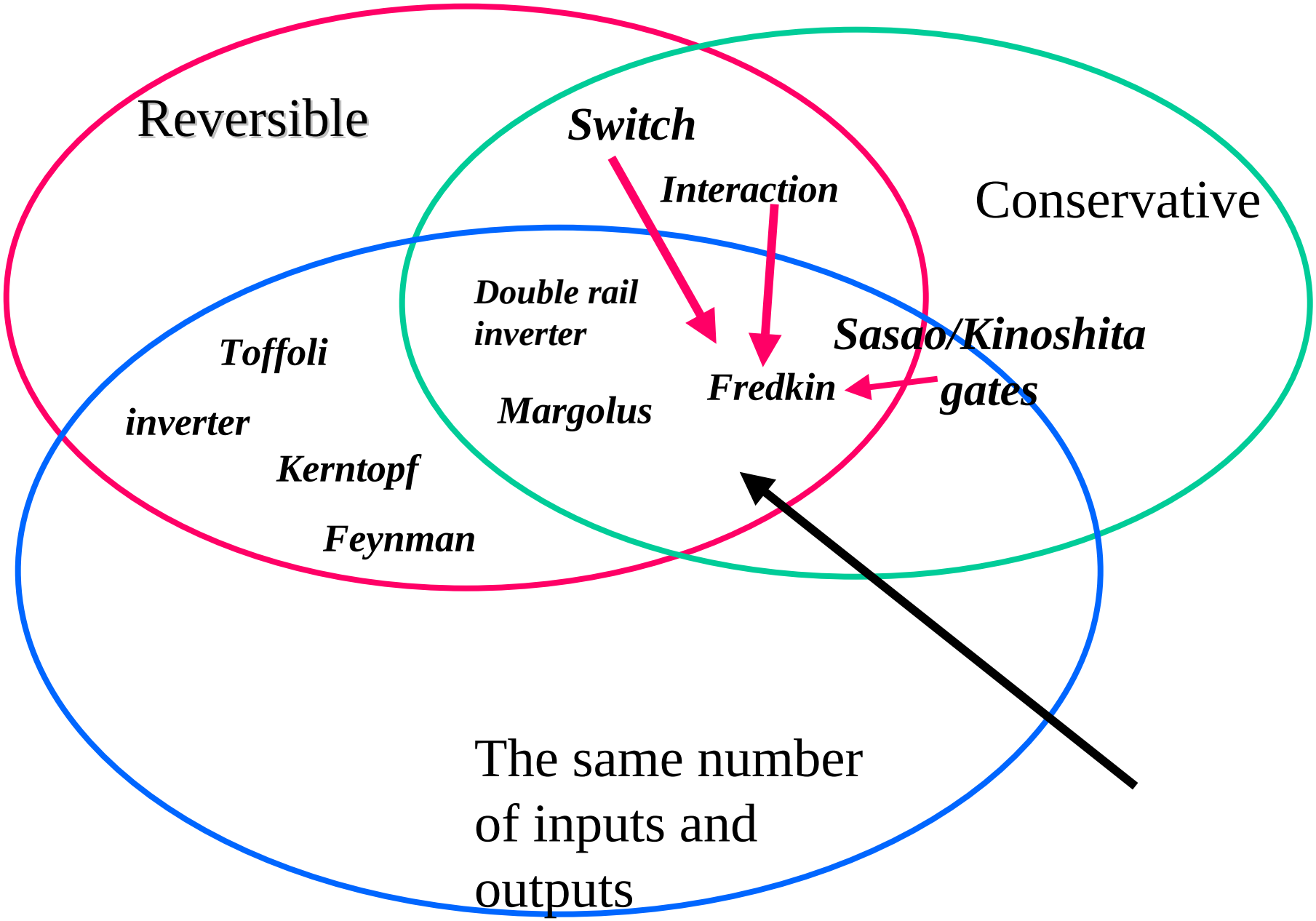


In this gate the input signals are routed to one of two output ports depending on the values of A and B

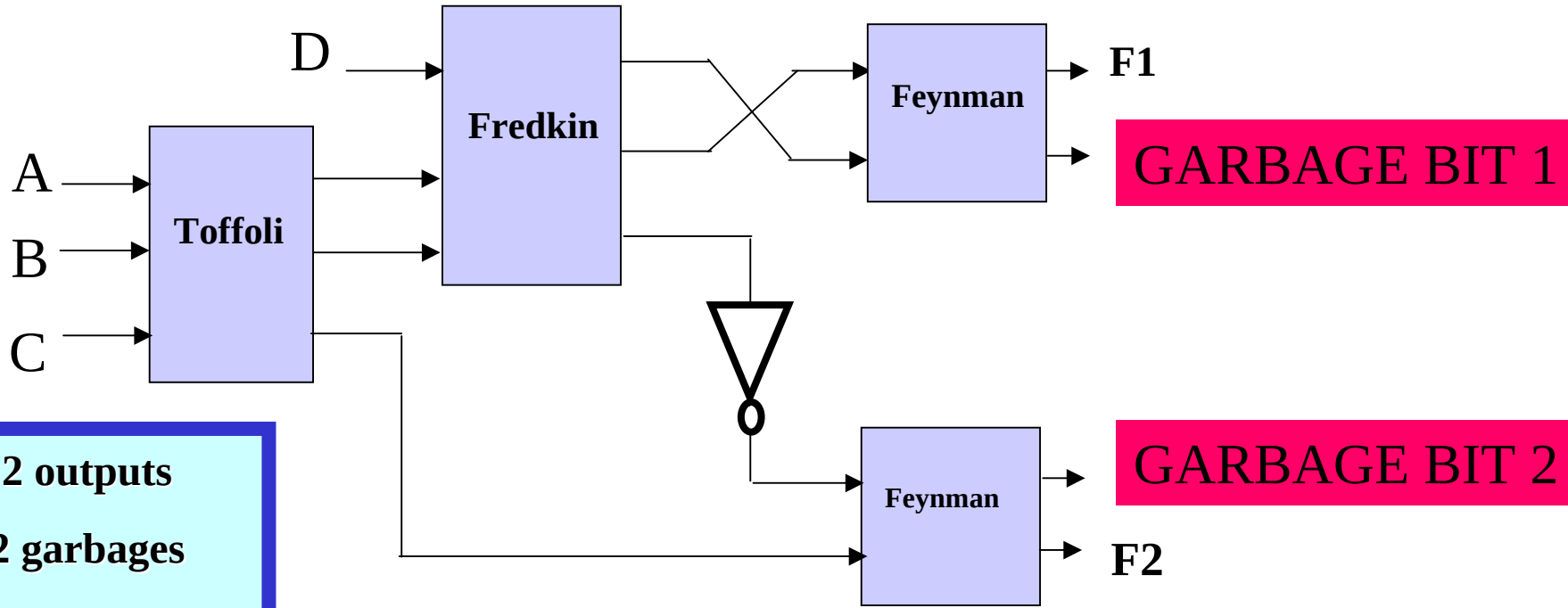
Fredkin Gate from Interaction Gates



Types of reversible logic



How to build garbage-less circuits



2 outputs

2 garbages

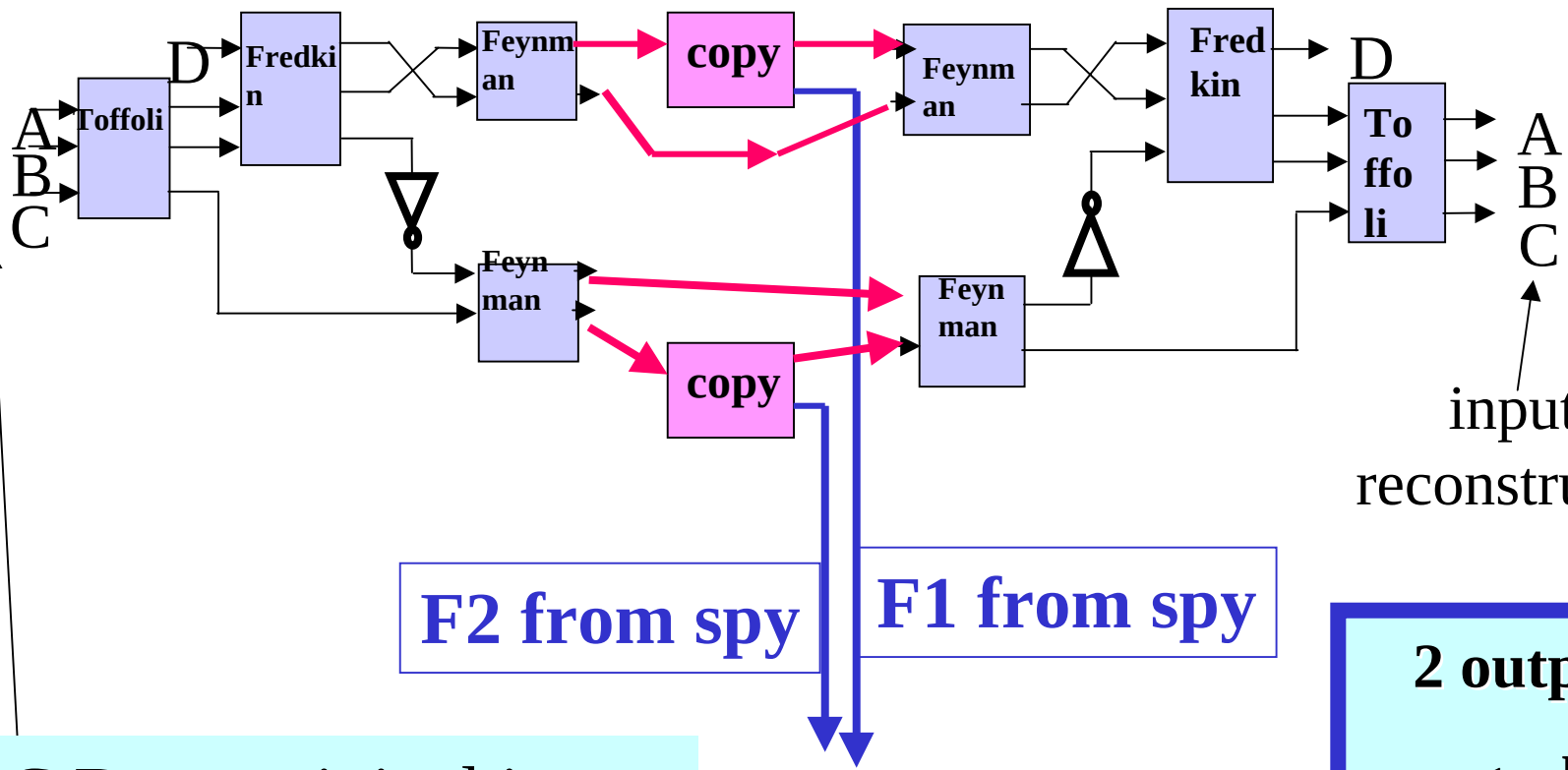
width = 4

delay = 4

We can decrease garbage at the cost of delay and number of gates

We create inverse circuit and add spies for all outputs

How to build garbage-less circuits



A,B,C,D are original inputs

2 outputs

no garbage

width = 4

delay = 9

This process is informationally reversible

It can be in addition thermodynamically reversible

Efficiency of gates (definitions)

- ***Definition***. A gate is **universal in n arguments** (is **ULM-n**) if every Boolean function of *n* variables can be implemented at one of its outputs using this gate (allowing constant signals at some inputs).
- ***Definition***. A gate is **two-level universal in n arguments** if it is possible to implement every Boolean function of *n* variables with a two-level circuit using this gate (allowing constant signals at some inputs).
- ***Definition***. A gate is **cascade-universal in n arguments** if it is possible to realize and arbitrary *n***n*-gate with a cascade circuit using this gate (allowing constant signals at some inputs).

Earlier work on Efficiency of gates

- Yale N. Patt (AFIPS Spring Joint Comp. Conf., 1967) established that the 3*1-gate implementing the following function

  $F = 1 \oplus x1 \oplus x3 \oplus x1*x2$

  is universal in three arguments with no more than three gates.
- George I. Opsahl (IEEE Trans.on Comp., 1972) showed that F is two-level universal in three arguments and that the following generalization of F:
  $G=1 \oplus x1 \oplus x3 \oplus x4 \oplus x1*x4 \oplus x2*x3 \oplus x1*x2*x4 \oplus x2*x3*x4$
  is two-level universal in four arguments. He also defined a sequence of functions having similar properties to HWB_n.
- It was also shown that functions with the best compositional properties have the number of cofactors close to the maximum (P. Kerntopf, IEEE Symp. on Switching and Automata Theory, 1974).

Statement of the Problems

- We will be concerned with searching for optimal gates.
- Let us try to find answers to the following questions
  - (1) Is there a reversible 3*3-gate for which all cofactors of the output functions obtained by replacements of one variable by constant 0 and 1 are distinct?
  - (2) Does there exist a reversible 3*3-gate universal in two arguments?
  - (3) Does there exist a reversible 3*3-gate two-level universal in three arguments?
  - (4) Does there exist a reversible 3*3-gate cascade-universal in three arguments?

Despite reversibility constraint the answers to all the above questions are positive.

Gate Having 18 Distinct Cofactors

$$P = 1 \oplus AB \oplus AC \oplus BC$$

$$Q = A \oplus C \oplus AB \oplus AC \oplus BC$$

$$R = A \oplus B \oplus AB \oplus AC \oplus BC$$

if A=0 then

$P = 1 \oplus BC$

$Q = C \oplus BC$

$R = B \oplus BC$

if A=1 then

$P = 1 \oplus B \oplus C \oplus BC$

$Q = 1 \oplus B \oplus BC$

$R = 1 \oplus C \oplus BC$

if B=0 then

$P = 1 \oplus AC$

$Q = A \oplus C \oplus AC$

$R = A \oplus AC$

if B=1 then

$P = 1 \oplus A \oplus C \oplus AC$

$Q = AC$

$R = 1 \oplus C \oplus AC$

if C=0 then

$P = 1 \oplus AB$

$Q = A \oplus AB$

$R = A \oplus B \oplus AB$

if C=1 then

$P = 1 \oplus A \oplus B \oplus AB$

$Q = 1 \oplus B \oplus AB$

$R = AB$

3*3-gate, universal in two arguments (ULM-2)

A	B	C	P	Q	R
0	0	0	1	1	0
0	0	1	1	0	1
0	1	0	1	0	0
0	1	1	0	1	1
1	0	0	0	1	0
1	0	1	0	0	0
1	1	0	1	1	1
1	1	1	0	0	1

Inputs	Output
A=1, B=0, C=y	P=0
A=x, B=y, C=1	P=x’y’
A=x, B=y, C=1	Q=x’y
A=x, B=0, C=y	P=x’
A=1, B=x, C=y	P=xy’
A=x, B=1,C=y	P=y’
A=x, B=1, C=y	Q=x ⊕ y
A=0, B=x, C=y	P=x’+y’
A=x, B=y, C=0	R=xy
A=0, B=x, C=y	Q=(x ⊕ y)’
A=0, B=x, C=y	R=y
A=x, B=y, C=0	P=x’+y
A=1, B=x, C=y	R=x
A=x, B=y, C=0	Q=x+y’
A=x, B=1, C=y	R=x+y
A=1, B=1, C=y	R=1

Experimental Results

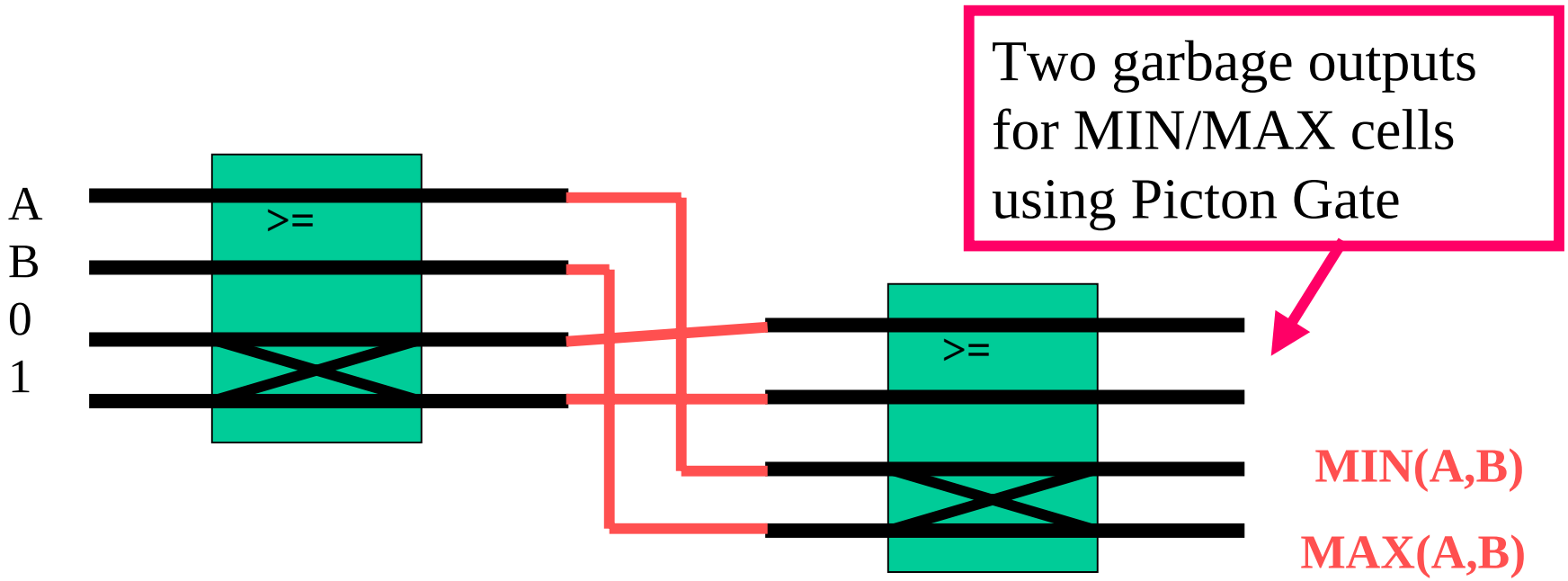
- Program was run constructing all two-gate circuits made of identical reversible 3*3-gates:
  - (3,3)-circuits,
  - (4,4)-circuits with one additional input to which only one constant signal was applied,
  - (5,5)-circuit with two additional inputs to which two identical constant signals are applied (00 or 11),
  - (5,5)-circuit with two additional inputs to which different constant signals are applied (00, 01, 10, 11).
  - There exist reversible 3*3-gates two-level universal in 3 arguments and cascade-universal in 3 arguments.

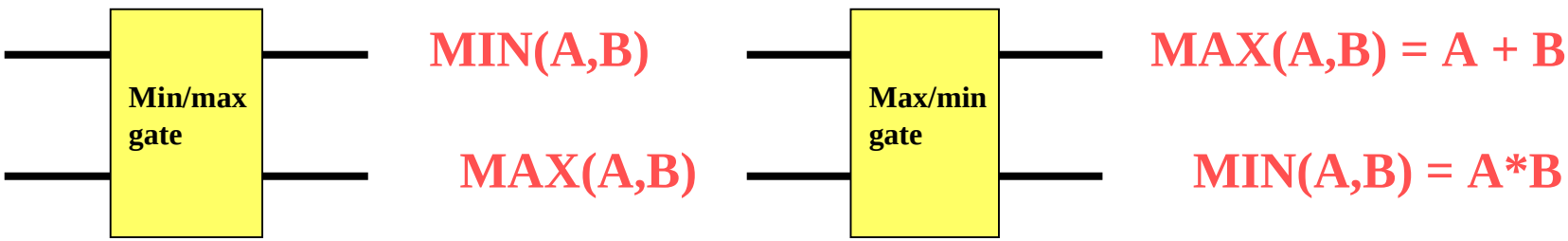
Goals of reversible logic synthesis

1. Minimize the garbage
2. Minimize the width of the circuit

   (the number of additional inputs)
3. Minimize the total number of gates
4. Minimize the delay

Use of two Multi-valued Fredkin (Picton) Gates to create MIN/MAX gate





Complex Gate

- Let us define a gate by the following equations:

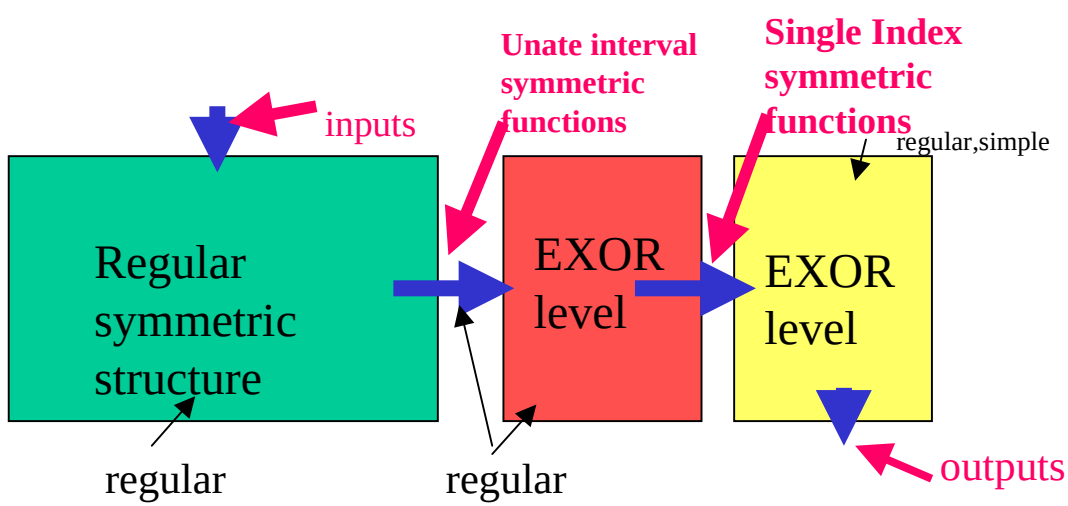
  $P = 1 \oplus A \oplus B \oplus C \oplus AB$

  $Q = 1 \oplus AB \oplus B \oplus C \oplus BC$

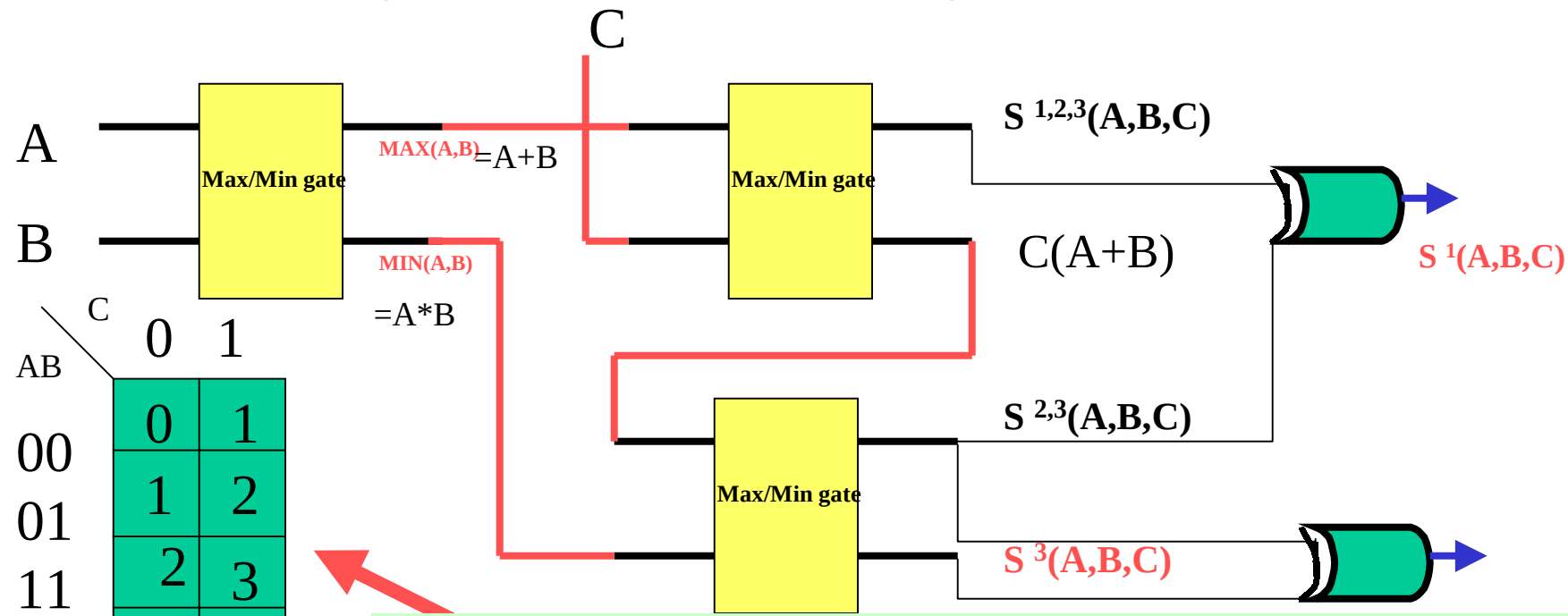
  $R = 1 \oplus A \oplus B \oplus AC$

- When $\underline{C = 1}$ then $P = A+B$, $Q = A*B$, $R = B'$, so operators ***AND/OR/NOT*** are realized on outputs P and Q with C as the controlling input value.
- When $\underline{C = 0}$ then $P = (A+B)'$, $Q = A+B'$, $R = (A \oplus B)'$.

Regular Structure for Symmetric Functions

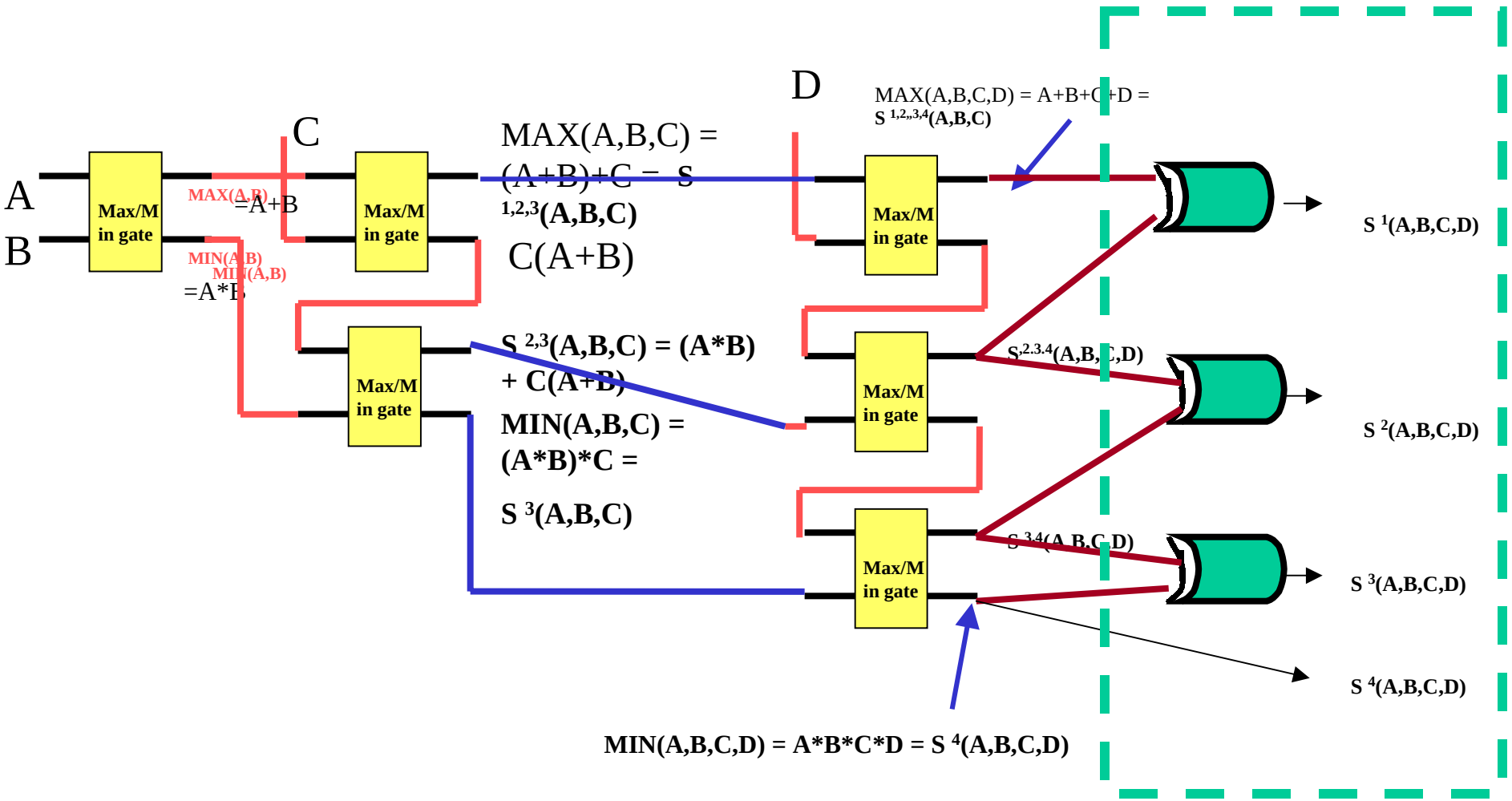


Every single index Symmetric Function can be created by EXOR-ing last level gates of the previous regular expansion structure



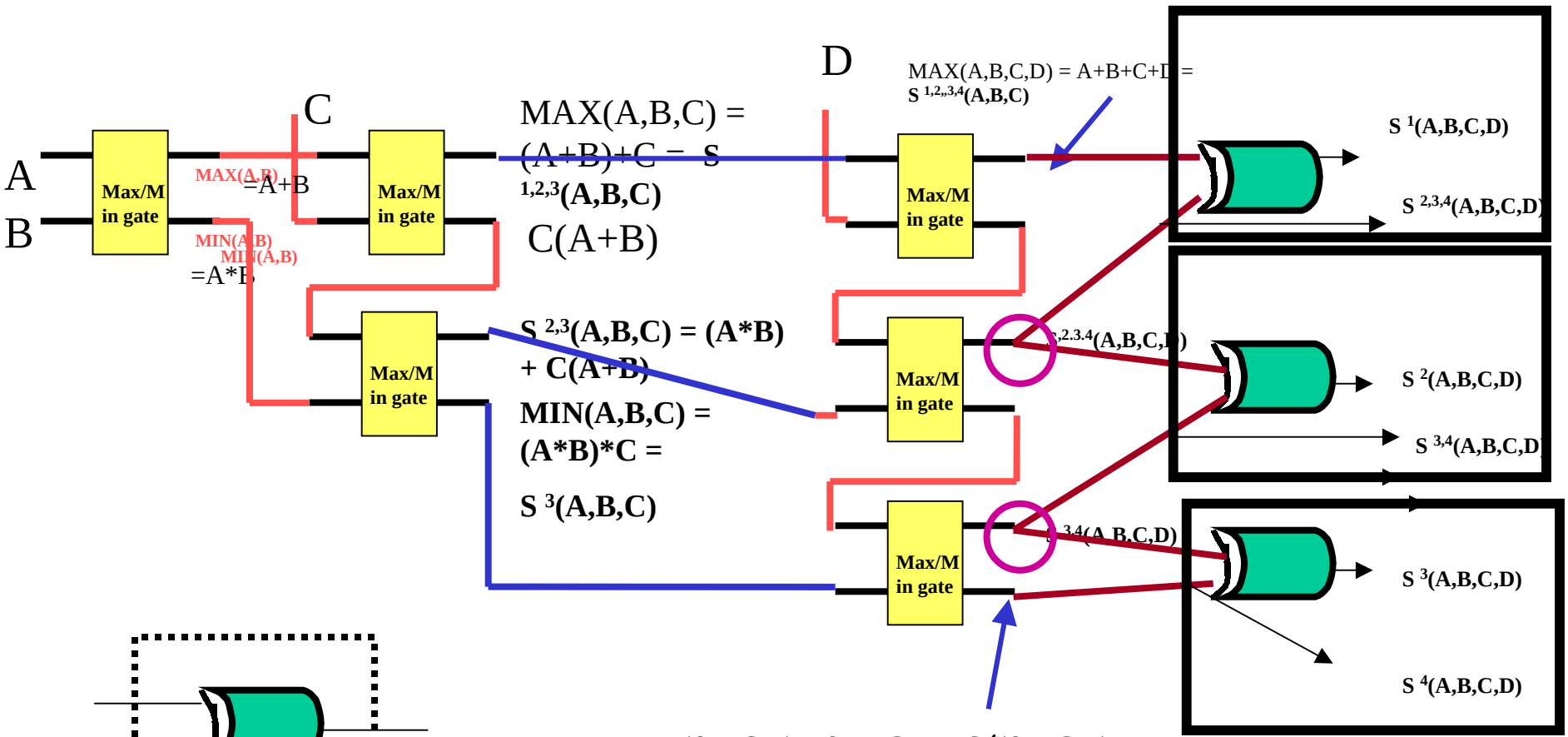
Indices of symmetric binary functions of 3 variables

Example for four variables, EXOR level added



It is obvious that any multi-output function can be created by OR-ing the outputs of **EXOR level**

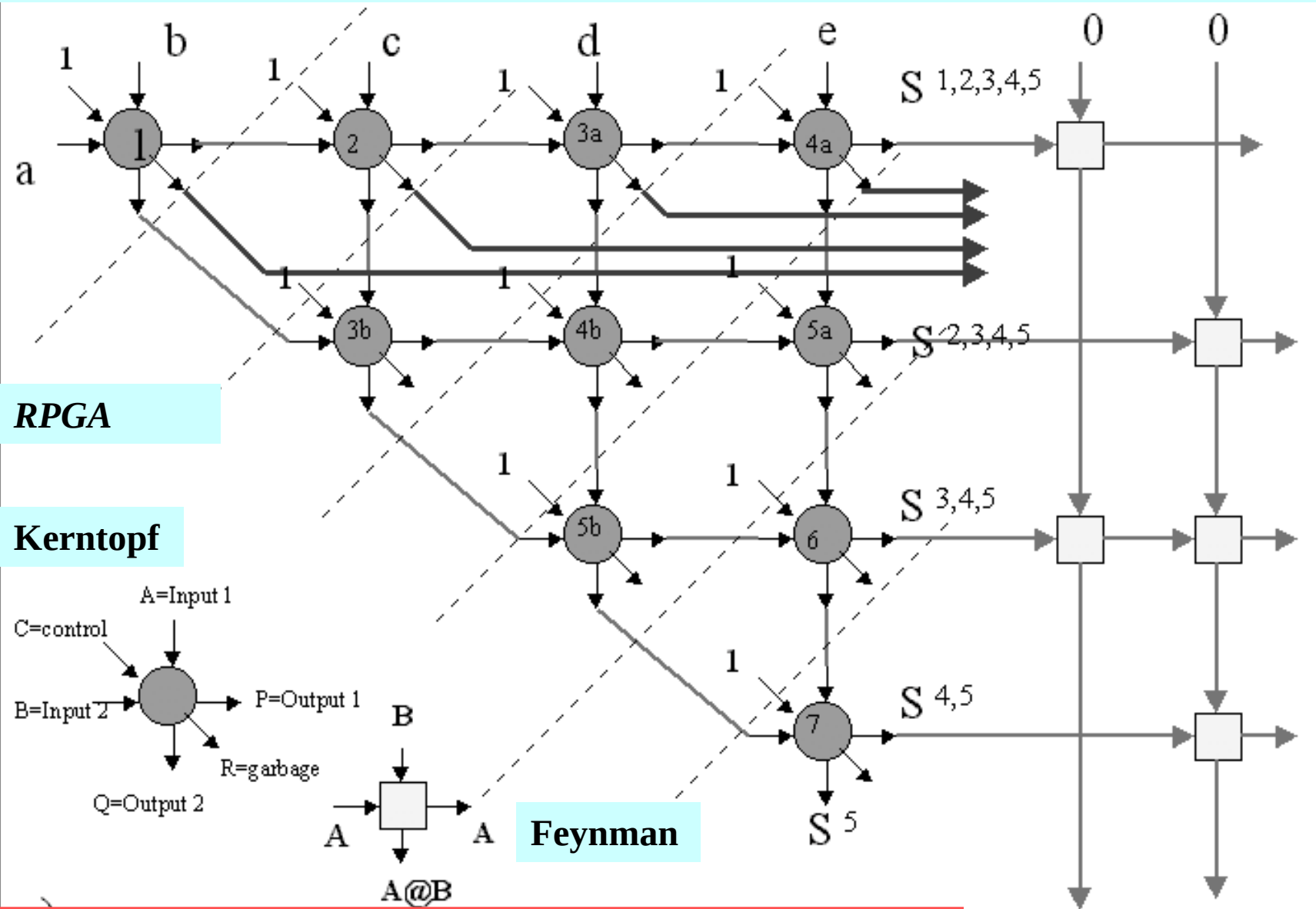
Now we extend to Reversible Logic



Denotes Feynman (controlled NOT) gate

Denotes fan-out gate

Using Kerntopf and Feynman Gates in **Reversible *Programmable Gate Array***



Arbitrary symmetric function can be created by exoring single indices

GENERALIZATIONS

- Arbitrary symmetric function can be realized in a net without repeated variables.
- Arbitrary (non-symmetric) function can be realized in a net with ***repeated variables*** (so-called ***symmetrization***).
- Many non-symmetric functions can be realized in a net without ***repeated variables***.

In a similar way we can obtain very many new circuit types, which are reversible and multi-valued generalizations of Shannon Lattices, Kronecker Lattices, and other regular structures introduced in the past.

General characteristic of logic synthesis methods for reversible logic

Very little has been published

Sasao and Kinoshita - cascade circuits, ***small garbage , high delay***

Picton - binary and multiple-valued PLAs, ***high garbage, high delay, high gate cost***

Toffoli, Fredkin, Margolus - ***examples of good circuits, no systematic methods***

De Vos, Kerntopf - new gates and their properties, ***no systematic methods***

Knight, Frank, Vieri (MIT); Athas et al. (USC) - circuit design, ***no systematic methods***

Joonho Lim, Dong-Gyu Kim and Soo-Ik Chae
School of Electrical Engineering, Seoul National University
- circuit design, ***no systematic methods***

•PQLG (Portland Quantum Logic Group) - Design methods for regular structures (including multiple-valued and three-dimensional)

Selection of good building blocks (another approach)

- Binary reversible logic gates with **three inputs and three outputs** have a ***privileged position:*** they are sufficient for constructing arbitrary **binary reversible networks** and therefore are the key to ***reversible digital computers.***
- There exist as many as 8! = 40,320 different 3-bit reversible gates.
- **The question**: ***which ones to choose as building blocks***.
- Because these gates form a **group** with respect to the operation 'cascading', it is possible to apply **group theoretical** tools, in order to make such a choice.
- **Leo Storme, Alexis De Vos, Gerald Jacobs (Journal of Universal Computer Science, 1999)**

R = the group of all reversible 3*3 gates (isomorphic to S_8)

- When a reversible 3*3 gate ***x*** is cascaded by a reversible 3*3 gate ***y*** then a new reversible 3*3 gate ***xy*** is formed.
- The subgroup of permutation and negation gates partitions R into 52 double cosets.
- PROBLEMS:
- 1. Find generators of group R ($r = s_1 g s_2 \ldots s_n g s_{n+1}$).
- 2. Investigate the effectiveness of these generators, it means the average number of cascade levels needed to generate an arbitrary circuit from this type of generator.
- 3. Investigate small sets of generators as candidates for a library of cells.

k_i	n_{max}	n_{ave}
k_{14}	4	2.46
k_{44}	4	2.46
k_{47}	4	2.46
k_{22}	4	2.88
k_{17}	4	2.92
k_{39}	4	2.92
k_{16}	5	3.44
k_6	5	3.54
k_{10}	5	3.54
k_{20}	5	3.54
k_{35}	5	3.54
k_{24}	5	3.58
k_{26}	5	3.58
k_{29}	5	3.58
k_{32}	5	3.58
k_{37}	5	3.58
k_{51}	5	3.58
k_4	6	3.73

Best gates

cascade-universal gates

k_4	6	3.73
k_{30}	6	4.23
k_{52}	6	4.23
k_{42}	7	4.35
k_{27}	7	4.77
k_{12}	8	5.71

Circuits with Toffoli gates need 0 <n< 6 levels (with average value 97/26 = 3.73) in order to generate R.

Toffoli gate

Cascade-universal gates (cont'd)

- If we consider depth $n = 4$ as too deep a cascade (too much silicon surface area/delay), we can construct a larger library.
- If we choose an $p = 2$ library, there are four equivalent optimal combinations:
  - r_{14} together with r_{18},
  - r_{14} together with r_{41},
  - r_{44} together with r_{48} , and
  - r_{44} together with r_{50} .
- Now we have n = 3, with expectation value 101/52 = 1.94.
- Enlarging the library to $p = 3$ yields $n = 2$ and average cascade depth 99/52 = 1.90.

Minimal number of constant inputs

- An arbitrary Boolean function of *n* variables can be implemented using Fredkin gates by a circuit with three constant inputs (Tsutomu Sasao, Kozo Kinoshita, “Conservative Logic Elements and Their Universality”, IEEE Trans. on Computers, 1979 - based on the paper by Bernard Elspas, Harald S. Stone “Decomposition of group functions and the synthesis of multirail cascades”, IEEE Symposium on Switching and Automata Theory, 1967).
- For n = 3 there exist reversible 3*3 gates that using them it is possible to implement each function with at most two constant inputs (P. Kerntopf, IEEE Workshop on Logic Synthesis, 2000)

Rich Ability of Computing

- Reversible circuits have relatively rich ability of computing in spite of reversibility constraint.
- **Reversible Turing Machines have computation universality:**
- **Lecerf (1963)** defined a reversible Turing Machine (TM) and proved that an irreversible TM can be simulated by a reversible one at the expense of a linear space-time slowdown.
- **Bennett (1973)** independently showed that irreversible TM can be simulated by an equivalent reversible TM.

Rich Ability of Computing (2)

- **Toffoli (1977)** showed that any *k*-dimensional cellular automaton can be simulated by a (k+1)-dimensional reversible cellular automaton (RCA).
- From this computation universality of 2-dimensional RCAs can be derived.
- **Morita, Harao (1989)** proved that 1-dimensional RCAs are computation universal in the sense that for any given RTM we can construct a 1-dimensional RCA that simulates it.
- **Morita (1990)** proved that any sequential circuit, reversible finite automaton and reversible cellular automaton (hence reversible TM) can be constructed only from Fredkin gates and delays without generating garbage signals.

Conclusions

- Reversible Computing is an attractive research area.
- ***Try to solve reversible problems:***
- **YOU'LL LIKE THEM!**