DESIGN AUTOMATION FOR REVERSIBLE AND ADIABATIC CIRCUITS

Robert Wille
Johannes Kepler University Linz
Software Competence Center Hagenberg GmbH (SCCH)

robert.wille@jku.at
https://iic.jku.at/eda/research/quantum/

SYNTHESIS OF REVERSIBLE CIRCUITS

$f: B\uparrow n \rightarrow B\uparrow m$

Embedding

$f: B\uparrow k \rightarrow B\uparrow k$

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

x	y	z	x'	y'	z'
0	**0**	**0**	**0**	0	0
0	**0**	**1**	**0**	0	1
0	**1**	**0**	**0**	1	0
0	**1**	**1**	**1**	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	0	1	1

SYNTHESIS OF REVERSIBLE CIRCUITS

$f: B\uparrow n \rightarrow B\uparrow m$

Embedding

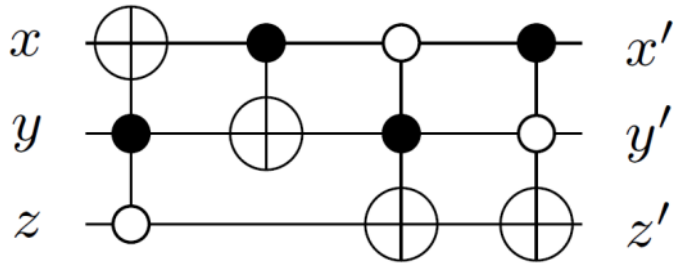
$f: B\uparrow k \rightarrow B\uparrow k$

Synthesis

Reversible Circuit

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

x	y	z	x'	y'	z'
0	**0**	**0**	**0**	0	0
0	**0**	**1**	**0**	0	1
0	**1**	**0**	**0**	1	0
0	**1**	**1**	**1**	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	0	1	1



JKU

THE EMBEDDING PROCESS

x	y	a	x'	y'	g
0	**0**	0	**0**	**0**	0
0	0	1	0	0	1
0	**1**	0	**0**	**1**	0
0	1	1	1	0	0
1	**0**	0	**0**	**1**	1
1	0	1	1	1	0
1	**1**	0	**1**	**0**	1
1	1	1	1	1	1

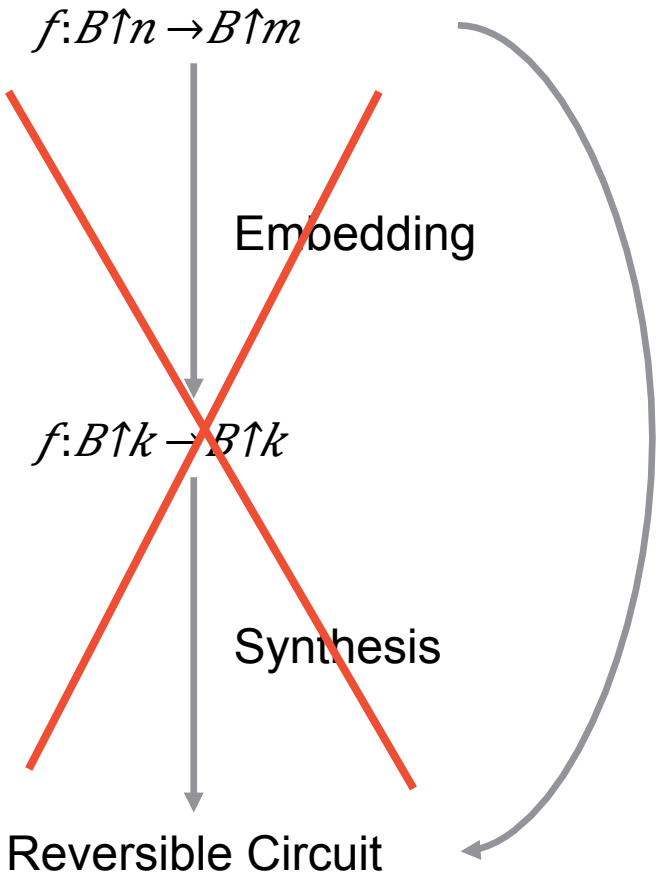
- Make output patterns distinguishable
  - Add 1 garbage output
- Adjust number of inputs and outputs
  - Insert 1 ancillary input
- Assign precise values
  - 192 possibilities
- **Embedding is a coNP-hard problem**

SYNTHESIS OF REVERSIBLE CIRCUITS

- Example: Transformation-based Synthesis
  - Transform outputs to inputs
  - Apply gates from right to left

line (i)	input xyz	output xyz
0	000	000 ✓
1	001	001 ✓
2	010	010 ✓
3	011	**0**11 ✓
4	100	100 ✓
5	101	101 ✓
6	110	110 ✓
7	111	**1**11 ✓

SYNTHESIS OF REVERSIBLE CIRCUITS



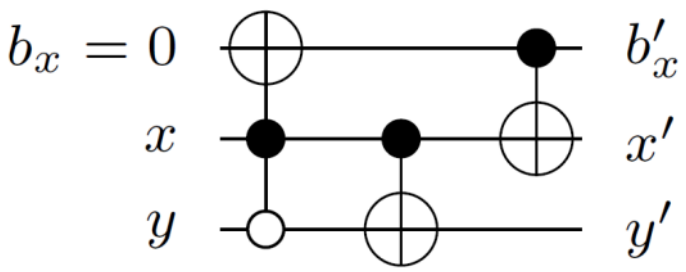
Solution:
Skipping embedding
→ One-pass synthesis

- Drawbacks:
  - ☐ Embedding is expensive
  - ☐ Degree of freedom is not exploited
  - ☐ Exponential growth of representation

ONE-PASS DESIGN FLOW

- Example: Transformation-based Synthesis

line (i)	input xy	output xy
0	00	00 ✓
1	01	01 ✓
2	10	10 ✓
3	11	11 ✓



- Start synthesis without embedding
- Modify function if problem occurs
  - ☐ Store changes on buffer line
- Complete synthesis with „wrong“ function
- Revert changes after synthesis
  - ☐ One gate for each buffer line

JKU

DESIGN AUTOMATION FOR ADIABATIC CIRCUITS

- Thus far:
  - Assumed full reversibility (e.g. mapping to Toffoli gates)
  - Caused overhead which is not necessarily needed
  - ➔Conditional reversibility is sufficient

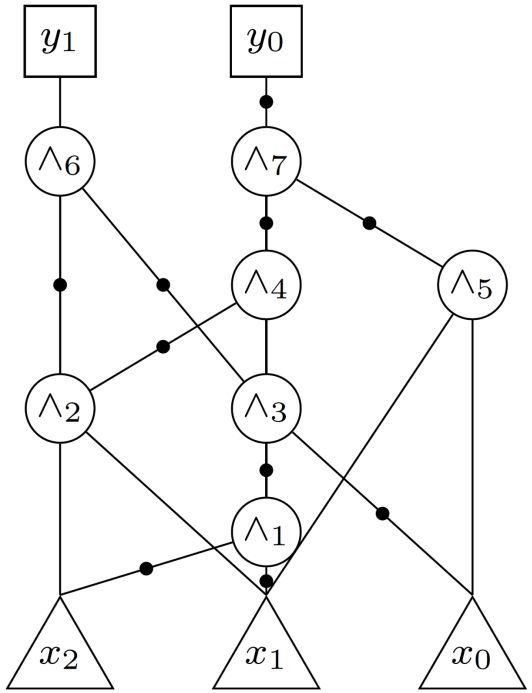
- Possible two-stage approach:
  1. Realize the function with respect to a certain logic gate library
  2. Map the resulting netlist to an adiabatic circuit satisfying switching rules

1ST STEP: UTILIZE AND-INVERTER GRAPHS (AIGS)

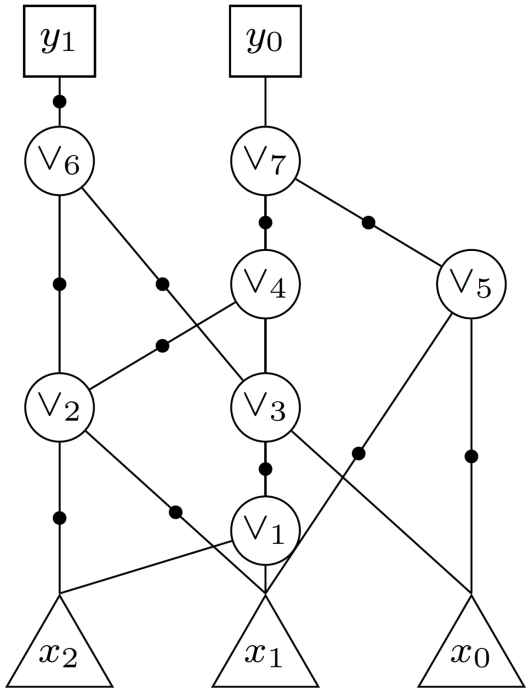
- Graph-based representation of Boolean functions
- Nodes represent AND operations; edges can be inverted (denoted by black circle)
- Can easily be mapped to NAND circuits and, using DeMorgan, to NOR circuits

$$y_0 = \overline{x}_2 x_1 + x_1 x_0 + x_2 \overline{x}_1 \overline{x}_0$$

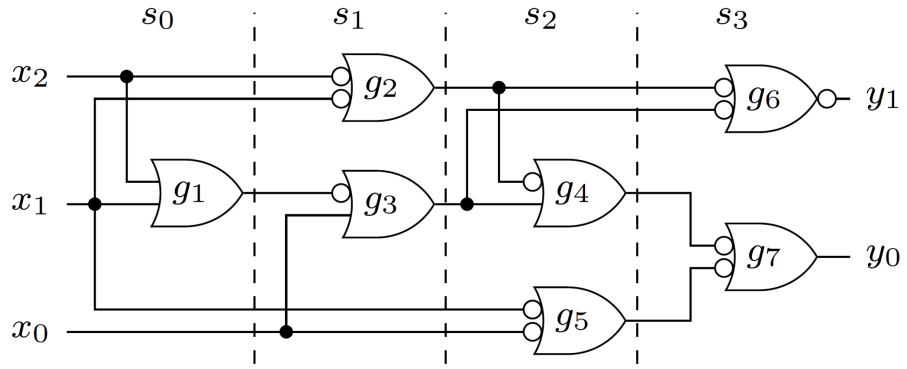
$$y_1 = \overline{x}_2 \overline{x}_1 + \overline{x}_2 x_0 + \overline{x}_1 x_0$$



Applying DeMorgan



Mapping to NOR circuit

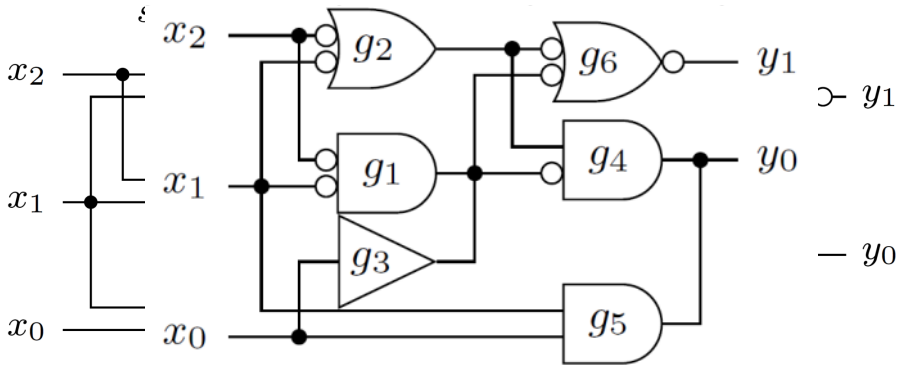


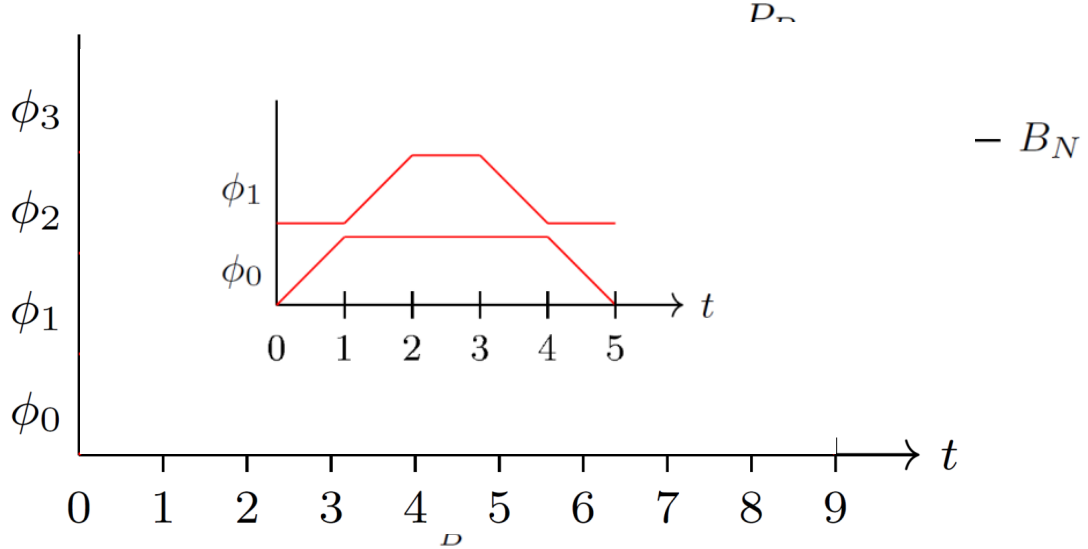
2ND STEP: MAP TO ADIABATIC CIRCUIT

- How to map gates?
- How to connect the gates to the power clocks?
- How to generate a corresponding waveform for these clocks?
- In all steps, switching rules need to be satisfied!

2ND STEP: USING RETRACTILE CIRCUITS

- 1:1 mapping of OR gates to transmission gates
- Circuit is composed of four stages → four clocks are needed
- Clock signals trigger the computations through the stages
- Once stable, clocks trigger decomputations in reverse order





FURTHER READING: BROADENING DESIGN

Synthesis of Reversible Circuits

- **One-pass Synthesis**
  One-pass Design for Reversible Circuits: Combining Embedding and Synthesis for Reversible Logic, TCAD 2017
- **Additionally Exploiting Coding Techniques**
  Exploiting Coding Techniques for Logic Synthesis of Reversible Circuits. ASP-DAC 2018

→ https://iic.jku.at/eda/research/one_pass_design_of_reversible_circuits

Design Automation for Adiabatic Circuits

- Design Automation for Adiabatic Circuits, ASP-DAC 2018
  https://arxiv.org/abs/1809.02421

Efficient Representation of Reversible Logic

- **Decision Diagrams**
  QMDDs: Efficient Quantum Function Representation and Manipulation, TCAD, 2016

→ http://iic.jku.at/eda/research/quantum_dd/