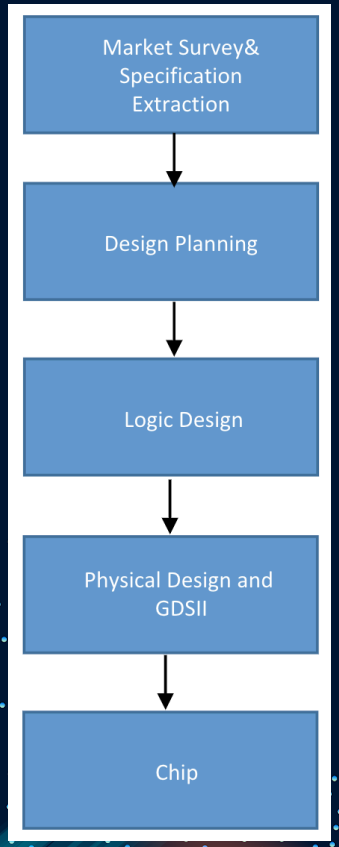
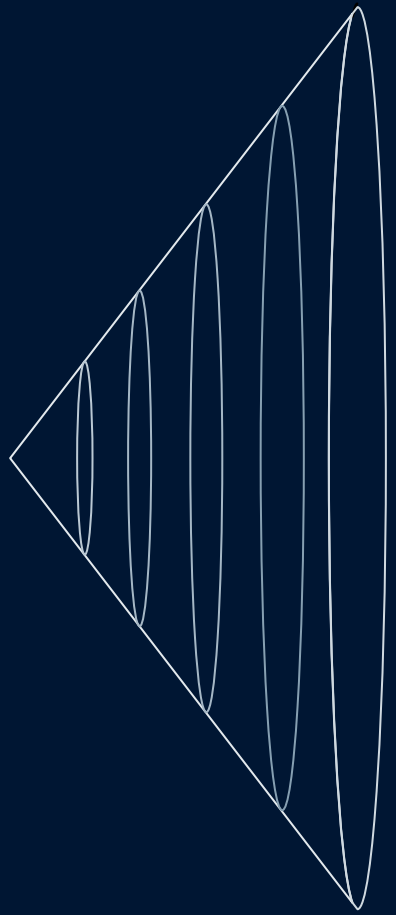
Logic Synthesis in VLSI Design

Embedded Electronic Engineering B - Winter Semester 2022/23

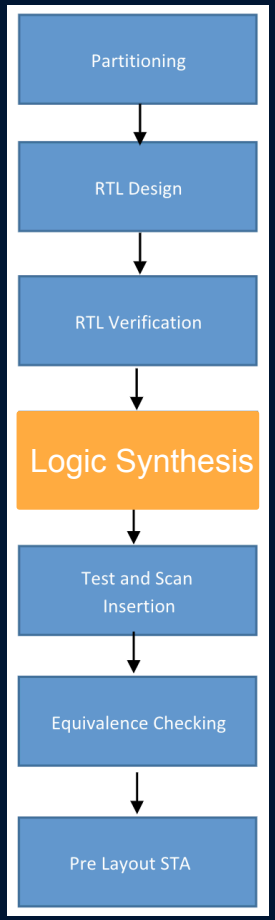
Vinícius Carvalho Gomes

ASIC Design Flow





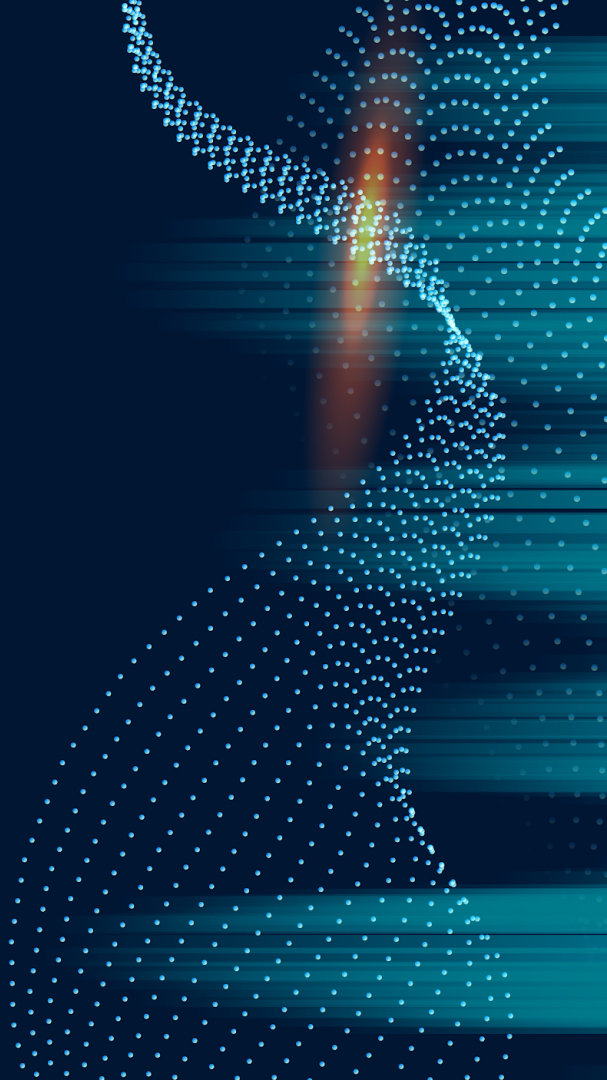
Logic Design Flow



What is Logic Synthesis?

Logic Synthesis converts RTL description into gate-level netlist suitable for the chosen technology.

- Input: Higher Level, RTL design, standard cell library (technology), design constraints.
- Output: Lower level, gate-level netlist, technology specific



Logic Synthesis Steps

01

Translation

Input: RTL circuit description

Output: Technology independent netlist

02

Logic Optimization

Input: Unoptimized boolean description

Output: Minimized boolean description

03

Technology Mapping

Input: Technology independent optimized netlist

Output: Optimized technology specific netlist

Logic Optimization

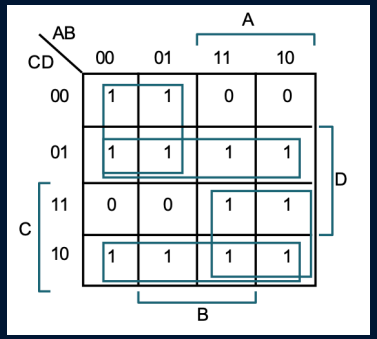
The goal of logic optimization is to reduce the number of literals in the boolean description of the design, removing redundancies. The main output of this process should be an optimal structure for the circuit independent of the gates available in the technology chosen to be used later on to implement the system.

Two main types of Boolean description:

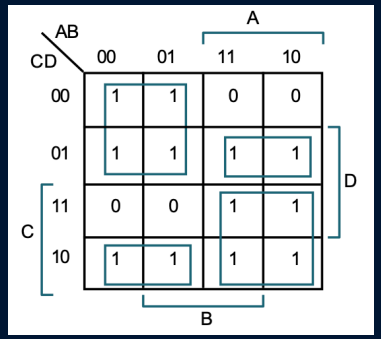
- Two-level logic: Boolean description as SOP, two levels of logic
- Multi-level logic: more than two levels of logic

Two-level Logic Minimization

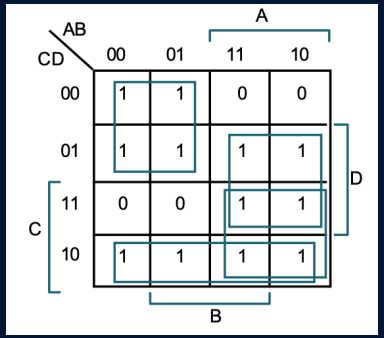
Find the minimum SOP that covers the output Boolean function. The most used heuristic method is the Espresso algorithm. Taking for example the expression
f = NOT(AC) + AD + AC + NOT(CD)



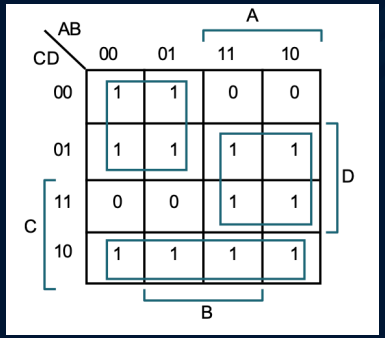
Starting SOP
f = (AC)' + C'D + AC + CD'
8 Literals, 4 primes



Reduce
f = (AC)' + AC'D + AC + A'CD'
10 Literals, 4 primes



Expand
f = (AC)' + AD + AC + CD'
10 Literals, 4 primes



Irredundant
f = (AC)'+AD+CD'
6 Literals, 3 primes

Multi-level Logic Minimization

Often is better to use more than two levels of logic. Some operations that can be used in multi-level minimization are:

- Extraction operations: identify common sub expressions and replace them with one variable.
- Collapsing operations: eliminate intermediate variables that do not affect the final result.
- Simplification operations: use two level minimization algorithms to simplify smaller parts of the expression.

ASIC Technology Mapping

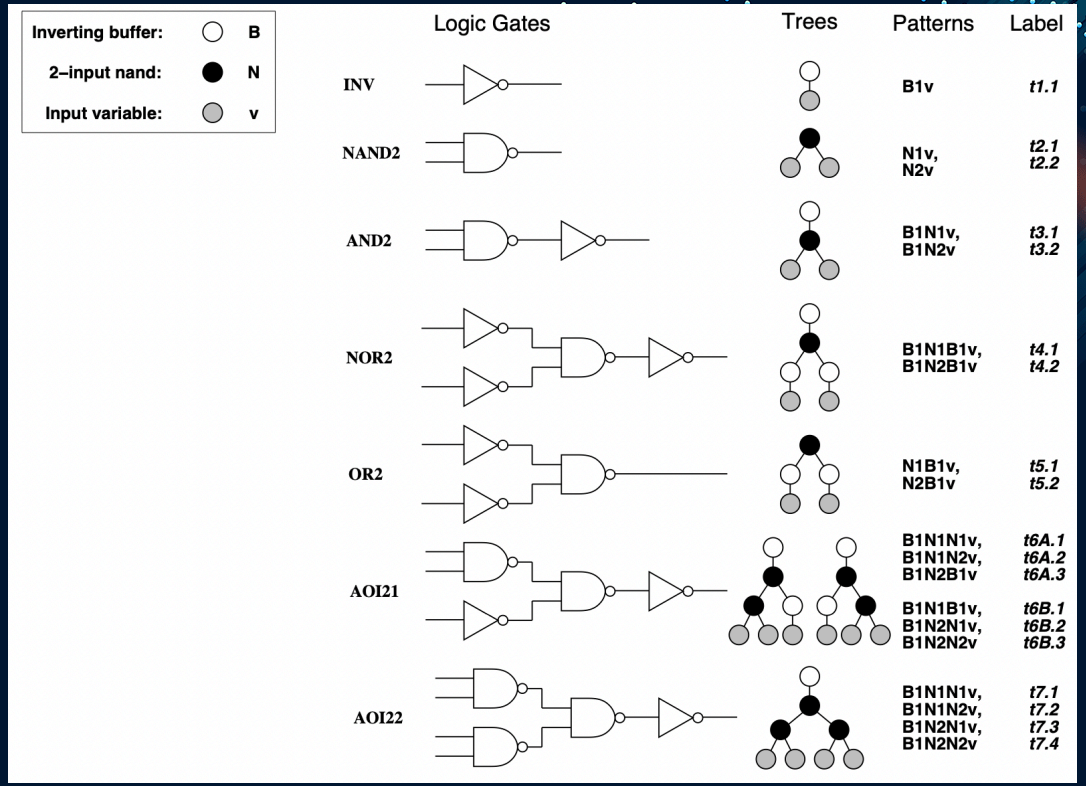
In ASIC design, each technology library has a set of specific gates that can be used and each gate is associated with a specific cost. The main focus on technology mapping will be to find the set of gates included in the library that can represent the Boolean function with the minimum cost.

The most famous model of technology mapping is called tree covering. The tree covering process can be separated in three parts:

- Simple Gate Mapping
- Tree-ifying
- Minimum Tree Covering

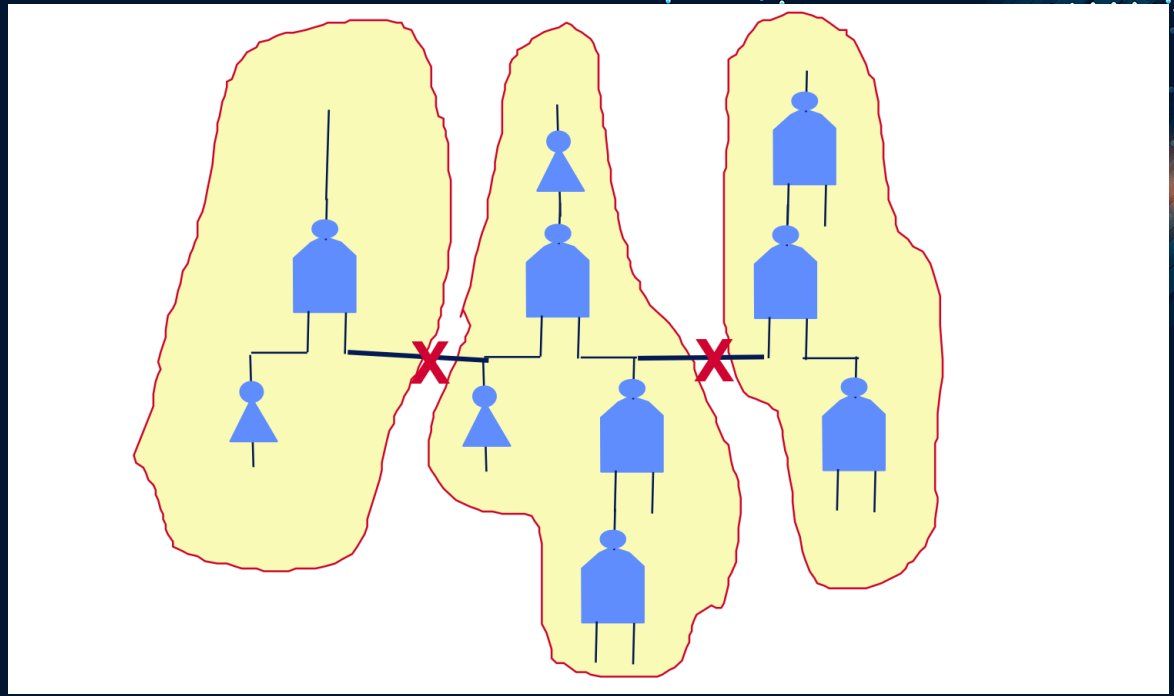
Simple Gate Mapping

Represent the Boolean function found in logic optimization and the standard library to be used as a collection of NOT and NAND2 gates.



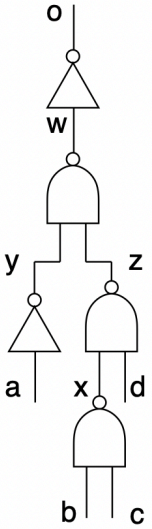
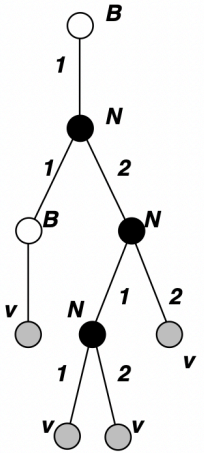
Tree-ifying

A given logic network is not a tree if there is a fanout greater than 1 at any gate output. Any nodes that have a fanout greater than 1 must be cut, generating two separate trees.



Minimum Tree Covering

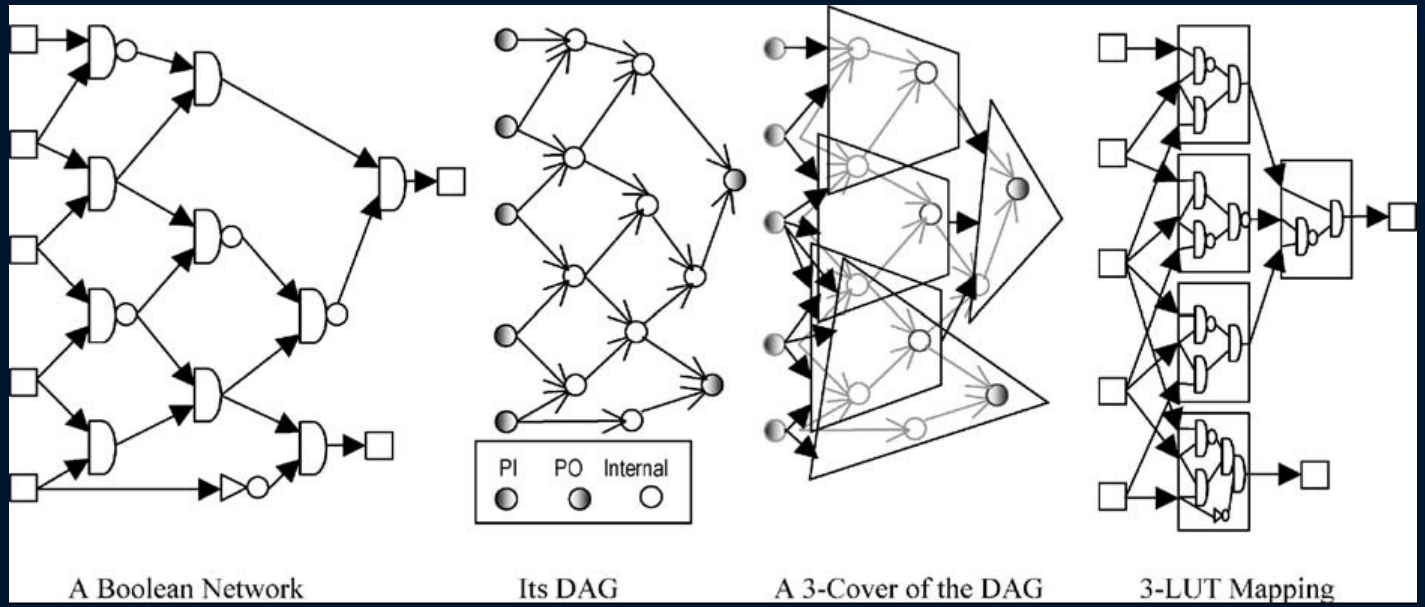
Using a recursive algorithm. Starting at the output of the tree, it is necessary to find all the gates available at the library that match a sequence from the top node of the tree then choose the lowest cost possible.

Network	Subject graph	Vertex	Match	Gate	Cost
		x	*t2*	NAND2(b,c)	NAND2
		y	*t1*	INV(a)	INV
		z	*t2*	NAND2(x,d)	2 NAND2
		w	*t2*	NAND2(y,z)	3 NAND2 + INV
		o	*t1*	INV(w)	3 NAND2 + 2 INV
			t3	AND2(y,z)	2 NAND2 + AND2 + INV
			t6B	AOI21(x,d,a)	NAND2 + AOI21

LUT Mapping (FPGA)

- For Field Programmable Gate Array (FPGA), the basic logic elements are usually lookup-tables (LUT).
- A LUT of K logic inputs (K-LUT) can implement any Boolean function of K variables and one output.
- The first step is to define possible cuts and then select the cuts according to the desired optimization criteria.
- Area of a mapping is the total number of LUT used.
- The maximum delay of a mapping is the maximum amount of LUT in a path from an input to the output.

LUT Mapping (FPGA) - Example



Logic Synthesis Solutions

- Most logic synthesis solutions are proprietary.
- For FPGA the most commonly used tools are Xilinx Vivado and Intel Quartus ii.
- For ASIC design the market leader is Synopsis
- Open source tools are emerging, offering non proprietary synthesis solutions for FPGA and ASIC (such as Yosys).

References

[1] Jason Cong. FPGA technology mapping, 1992.
[2] Frederic Mailhot. Technology mapping for VLSI circuits exploiting boolean properties and operations, 1991.
[3] Richard L Rudell. Logic synthesis for VLSI design, 1989.
[4] Rob A. Rutenbar. Technology mapping. https://course.ece.cmu.edu/ee760/760docs/lec10.pdf, 2001.
[5] Vaibbhav Taraate. *ASIC Design and Synthesis*. Springer Singapore, 2021. [6] Dr. Adam Teman. Digital vlsi design: Logic synthesis. https://www.eng.biu.ac.il/temanad/files/2017/02/Lecture-5-Synthesis.pdf, 2016.