



Technology Mapping

Introduction

- ◆ **Technology mapping** is a step of
 - taking **a logic netlist** as the input and
 - expressing the netlist using a set of gates from a **technology library**
- ◆ In the technology library, a gate may have different configurations, resulting in different **area**, **delay**, and **power**
- ◆ In this assignment, you are going to write a program to determine the gate configurations to **minimize the power** subject to a **delay constraint**

Problem Formulation

◆ Inputs

- A netlist in the AIG format
- A technology library

◆ Output

- A functionally equivalent netlist implemented with gates in the technology library

◆ Objective

- Optimize the **total power** with the **minimum delay**

Input 1 – Netlist (.blif)

- ◆ Netlist are given in the **blif** format
- ◆ Use ABC to parse the netlist and transfer it into the **AIG** format
- ◆ **Don't optimize the given netlist**

Input 2 - Technology Library (.lib)

◆ INV

Pattern Type	Timing (n:#FOs)	Power
INV1	$0.28+0.72*n$	20.92
INV2	$1.03+2.64*n$	1
INV3	$0.47+1.20*n$	2.75
INV4	$0.82+2.10*n$	19.25

◆ NAND

Pattern Type	Timing (n:#FOs)	Power
NAND1	$0.56+1.44*n$	25.76
NAND2	$2.31+5.95*n$	1.4
NAND3	$1.16+3.00*n$	10.82
NAND4	$1.79+4.61*n$	25.38

Output – Netlist (.mbench)

- ◆ Output the mapped netlist in the **modified bench** format
- ◆ You cannot modify PI and PO names

INPUT(x1)

INPUT(x2)

..

OUTPUT(y1)

OUTPUT(y2)

...

G1 = INV1(x1)

G2 = AND2(x2, G1)

...

Technology Mapping

- ◆ An **AND** node is mapped to **an NAND + an INV**
- ◆ A complemented fanin (phase==1) is mapped as an **INV**
- ◆ **Two INVs** on a wire are redundant and need to be removed
- ◆ **Minimum delay**
 - Circuit delay under which all the nodes are mapped to the **fastest** gates
 - **Topological timing analysis**

Delivery & Due Date

- ◆ A zip file including
 - Your source code and a ReadMe describing how to compile and run your program
 - Your mapping results of ISCAS'85 benchmarks
c432 ~ c7552
- ◆ Due on 2022/6/12 before lecture