Introduction to Logic Synthesis

Adnan Aziz
E.C.E.
University of Texas at Austin

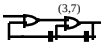
Todays lecture

- Design automation
  - history
  - significance
- Course outline
  - applications
- Administrivia
- Boolean functions

VLSI design automation

- Need for VLSI design assistants fueled by
  - complexity of VLSI - 10^7 transistors on a chip
  - profitability strongly linked to time to market
- CAD tools are an integral part of hardware design efforts

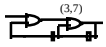
VLSI Design - Big Picture

RTL level	reg x[0:4]; x = (y+z) mod 6;	Design verification, High level synthesis
Gate level		Logic optimization, test generation
Transistors		Analog simulation
Layout		Place and route

CalTech experiment

- CalTech experiment - “silicon compilation” 197x
  - + framework - pioneering effort
  - - poor quality silicon
- Berkeley approach
  - + bottom up, rooted in optimization
  - + competitive with designers

Models and Algorithms

LEVEL	MODEL	ALGORITHMS
reg x[0:4]; x = (y+z) mod 6;	FSM network, Data Flow Graph	Graph traversal, Integer programming
	Graph of logical operators and delay elements	Boolean satisfiability, graph covering
	Differential equations	Numerical analysis
	Geometric objects (polygons)	min spanning tree, convex hull

Commercial significance

- semiconductor market 1996
  - 120 Billion US $ - 85% digital
- microelectronics cad 1996
  - 1.2 Billion US $
    - system level = 75 million US $
    - RT level = 375 million US $
    - Gate level = 275 million US $
    - Misc = 275 million US $

Logic synthesis - technology

- many ways of implementing design in VLSI
  - custom
  - semicustom: cell based and array based
    - cell based - standard cell/macro cell
    - array based - MPGAs/FPGAs
- choice based on various factors
  - density, performance, flexibility, design time, manufacturing time, cost(volume)
- logic synthesis - largely technology independent

What will you get from this class?

Study automatic generation of gate level implementations from HDL specifications

- Fundamental CAD algorithms + data structures:
  - graph covering, SAT heuristics, shortest path,...
  - Boolean logic, BDDs, TBFs, FSMs,...
- Tools: end user + programming
  - VIS, SIS, Synopsys
  - Verilog, BLIF, BLIF-mv

Course summary

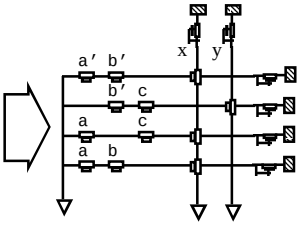
- 2 level logic minimization
  - Boolean functions, PLAs
- Multi-level logic minimization
  - share common logic, node simplification
- Topics in logic synthesis
  - timing analysis & optimization, testing, FPGAs
- Sequential synthesis
  - sequential functions, encoding, state minimization, retiming, verification

2 level minimization

- Reduce size of PLA implementing logic function

a	b	c	x	y
0	0	0	1	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	-	-
1	0	1	1	1
1	1	0	1	0
1	1	1	1	0

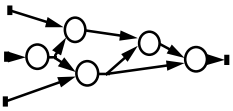
Designer specification



VLSI implementation

Multi-level minimization

- Logic network

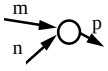


1. factoring

f = ab + c’ + d’e’;
g = ab + c’ + de;

h = ab + c’;
f = h + d’e’;
g = h + de;

2. node simplification

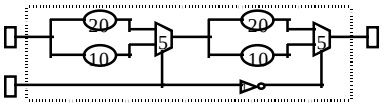


p = n ^ m;

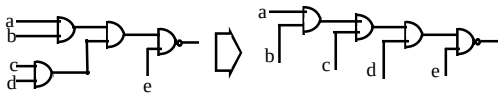
p = n + m;

Timing analysis & optimization

- how fast is design?

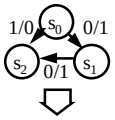


- make design faster

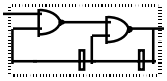


Sequential synthesis

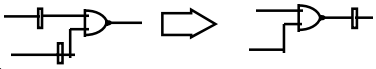
- representation: FSM



- encoding:



- retiming:



Administrivia - I

- Prerequisites
  - Digital design, C, mathematical maturity
- Book: "Synthesis and Optimization of Digital Systems"
  - DeMicheli, McGraw Hill 1994
- Format/Evaluation:
  - 8 homeworks = 30%
  - Midterm 1 + Midterm 2 = 35 %
  - Project = 35%