

ASIC Design (7v81)

Spring 2000

Lecture 1 (1/21/2000)

- **General information**

General description

- We study the hardware structure, synthesis method, design methodology, and design flow from the application to ASIC chip.
- A project is developed to design an FIR digital filter. 7 students are required to start from a filter algorithm and system specifications to complete an ASIC design.

Instructor information

- ◆ Instructor D. Zhou
- ◆ Email: zhoud@utdallas.edu
- ◆ Phone: 972 883 4392
- ◆ Office: EC2.510
- ◆ Class room: EC 2.120

TA and office hours

- TA: Naveen Mysore
 - ◆ Phone : (214) 575-3246
 - ◆ Email : navmys@utdallas.edu
 - ◆ Office : EC: 2.322
- Office hour (TA) : Thursday 7pm - 8pm
- Office hour (instructor): after class meeting

Text book

- **Application-Specific Integrated Circuits, Michael John Sebastian Smith, Addison Wesley, ISBN: 0-201-50022**

Prerequisite

- VLSI design knowledge
- VHDL or similar hardware design language
- C language
- Physical layout, circuit simulation and logic verification

Project requirement

- Specification of the project
- Clear presentation
- Technique details of the designed system
- Well written report
- Maximum team size: 2

Grading

- homework 20%
- reading report 10%
- attendance and quiz 10%
- middle term project 20%
- final project 40%
- 85-100 ->A, 70-84 ->B, 60-69 ->C, and below 60 -> f

Lecture 2

- Introduction to ASIC design

PRODUCTION TO ASICs

Key concepts: The difference between full-custom and semicustom ICs • The difference between standard-cell, gate-array, and programmable ASICs • ASIC design flow • ASIC cell library

ASIC (“a-sick”) is an **Application-Specific Integrated Circuit**

Evolution of ASICs: the **standard parts**, initially used to design **microelectronic systems**, were gradually replaced with a combination of **combinational logic**, **custom ICs**, **dynamic random-access memory (DRAM)**, and **static RAM (SRAM)**

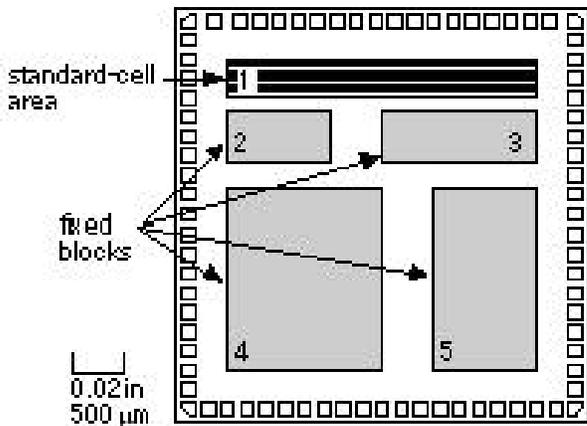
Application-specific standard products (ASSPs) are a cross between standard parts and ASICs

Types of ASICs

Full-Custom ASICs

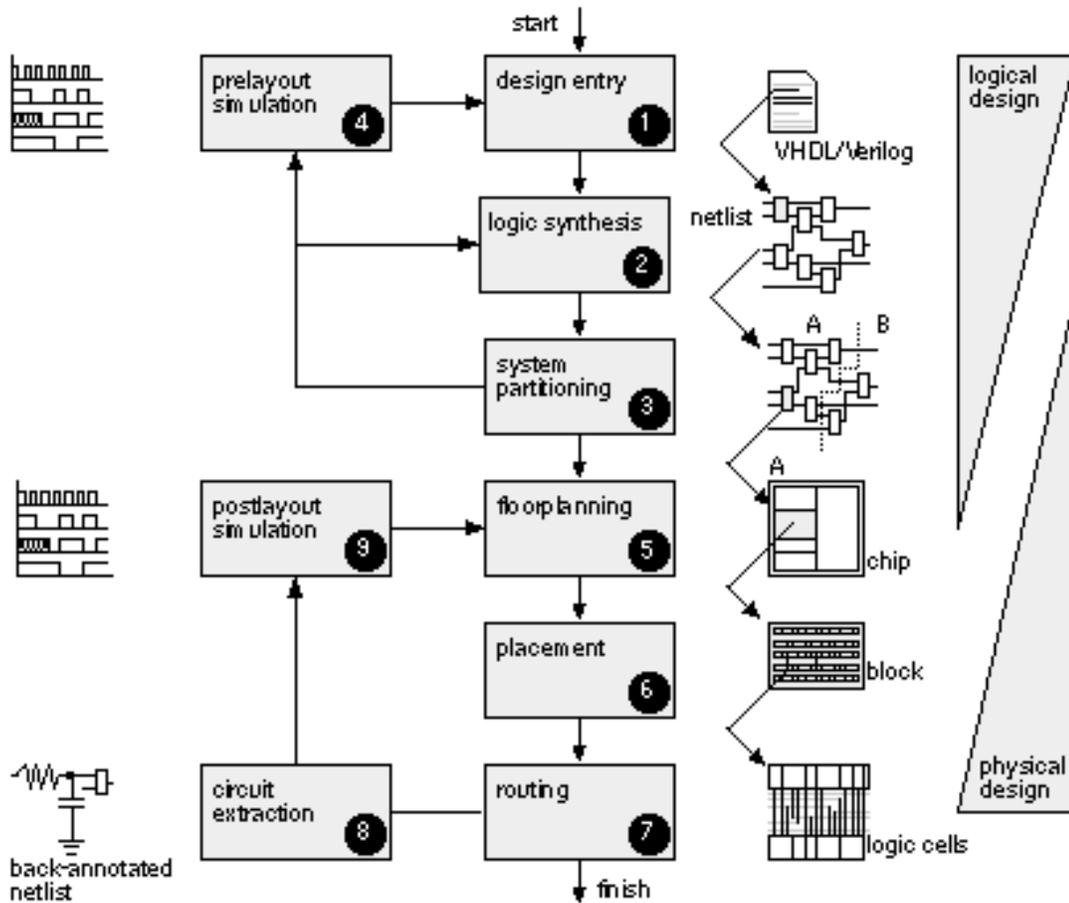
Full-custom offers the highest performance and lowest part cost (smallest die size) but has the disadvantages of increased design time, complexity, design expense, and highest risk. Examples of full-custom ICs or ASICs are requirements for high-voltage (automobile), analog/digital (communications), or sensors and actuators.

Standard-Cell-Based ASICs



A cell based ASIC die with a single cell area together with four fixed blocks

sign flow



SIC design flow. Steps 1–4 are **logical design** , and steps 5–9 are **physical design**

Full custom design and ASICs

- Full custom design
 - ◆ high performance and high design cost
 - ◆ examples: PC and workstation CPU
- ASICs
 - ◆ low design cost and compromised performance
 - ◆ examples: an I/O circuit or a special DSP chip

ASIC hardware architectures

- pre-design cells and functional blocks
- uniformed circuits and devices
- uniformed interconnect structures
- programmable interconnects

Cell-based ASIC (CBIC —“sea-bick”)

Standard cells

Possibly megacells , megafunctions , full-custom blocks , system level macros

(Slms), fixed blocks , cores , or functional standard blocks (fsbs)

All mask layers are customized—transistors and interconnect

Custom blocks can be embedded

Manufacturing lead time is about eight weeks

Gate-array-based ASICs

A gate array, masked gate array, MGA, or pre-diffused array uses macros (blocks) to reduce turnaround time and comprises a base array made from a base cell or primitive

cell

Types of gate array – based ASIC:

- Channeled gate arrays

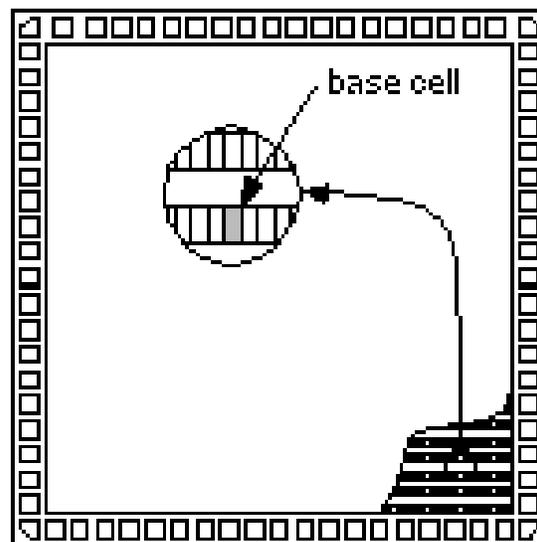
- Channel less gate arrays

- Structured gate arrays

channelled gate array

channelled gate array

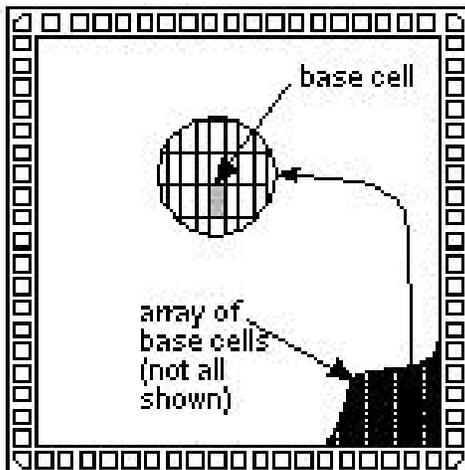
- Only the interconnect is customized
- The interconnect uses predefined spaces between rows of base cells
- Manufacturing lead time is between two days and two weeks



channelless gate array

channelless gate array (channel-free gate array , sea-of-gates array , or SOG array)

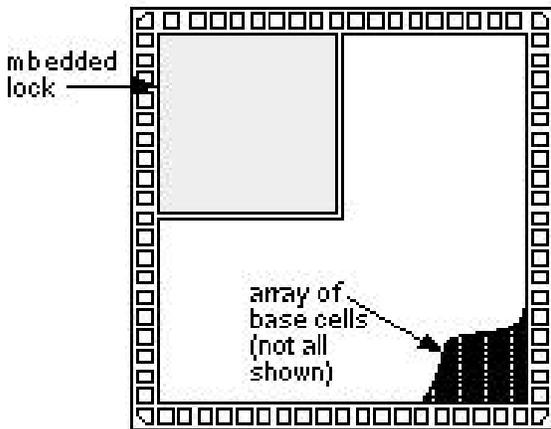
- Only some (the top few) mask layers are customized—the interconnect
- Manufacturing lead time is between two days and two weeks



Structured gate array

Embedded **gate array** or **structured gate array** (masterslice or masterimage)

- Only the interconnect is customized
- Custom blocks (the same for each design) can be embedded
- Manufacturing lead time is between two days and two weeks



programmable logic devices

programmable logic device (PLD)

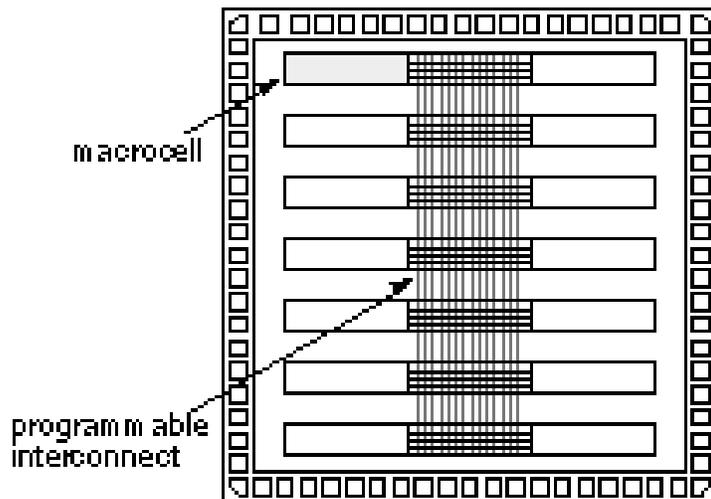
No customized mask layers or logic cells

Fast design turnaround

A single large block of programmable interconnect

A matrix of logic macro cells that usually consist of programmable array logic followed

by a flip-flop or latch



Field-programmable gate arrays

Field-programmable gate array (FPGA) or complex PLD

One of the mask layers are customized

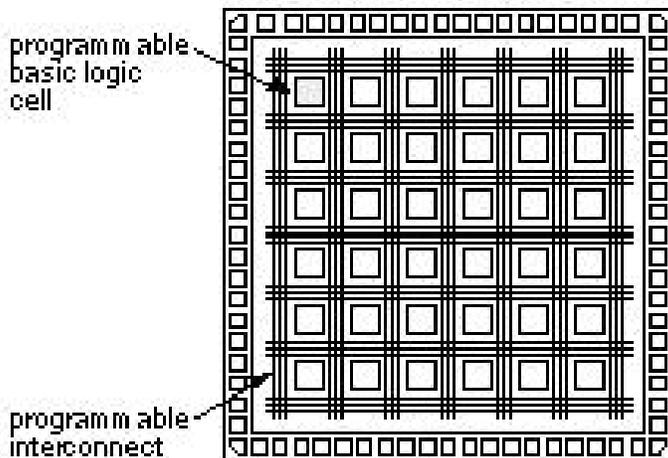
A method for programming the basic logic cells and the interconnect

The core is a regular array of programmable basic logic cells that can implement combinational as well as sequential logic (flip-flops)

A matrix of programmable interconnect surrounds the basic logic cells

Programmable I/O cells surround the core

Design turnaround is a few hours



Design entry . using a **hardware description language (VHDL)** or Schematic entry

Logic synthesis . produces a **netlist** —logic cells and their connections

System partitioning . divide a large system into functional blocks

Prelayout simulation . check to see if the design functions correctly

Floorplanning . arrange the blocks of the netlist on the chip

Placement . decide the locations of cells in a block

Routing . make the connections between cells and blocks

Extraction . determine the resistance and capacitance of the interconnect

Postlayout simulation . check to see the design still works with the added Loads of the interconnect

ASIC cell libraries

Design kit from the ASIC vendor

Usually a phantom library —the cells are empty boxes, or phantoms , you hand off your design to the ASIC vendor and they perform phantom instantiation
(Synopsys CBA)

ASIC-vendor library from a library vendor

Involves a buy-or-build decision

You need a qualified cell library (qualified by the ASIC foundry

If you own the masks (the tooling) you have a customer-owned tooling (COT , pronounced “see-oh-tee”) solution (which is becoming very popular)

Building own cell library

Involves a complex library development process:

Cell layout

Behavioral model

Verilog/VHDL model

Timing model

Test strategy

Characterization

Circuit extraction

Process control monitors (pcms) or drop-ins

Cell schematic

Cell icon

Layout versus schematic (LVS) check

Logic synthesis

Retargeting

Wire-load model

Routing model

Phantom

Reading assignment

- Chapter one
- Write a one page reading summary
- Due: in a week