Introduction to HDL-based design methodology

Prof. Chien-Nan Liu
TEL: 03-4227151 ext:34534
Email: jimmy@ee.ncu.edu.tw

Milestones for IC Industry

- **1947**: Bardeen, Brattain & Shockley invented the transistor, foundation of the IC industry
- **1952**: SONY introduced the first transistor-based radio
- **1958**: Kilby invented integrated circuits (ICs)
- **1965**: Moore's law
- **1968**: Noyce and Moore founded Intel
- **1971**: Intel announced 4-bit 4004 microprocessors (2300 transistors)
- **1976/81**: Apple / IBM PC
- **1985**: Intel began focusing on microprocessor products
- **1987**: TSMC was founded (implication: fabless design)

Milestones for IC Industry

- **1991**: ARM introduced its first embeddable RISC IP core (implication: chipless design)
- **1996**: Samsung introduced prototype 1G DRAM
- **1998**: IBM Austin Res. Lab announced 1GHz experimental microprocessor; Ericsson, etc. founded Bluetooth Special Interest Group -- designs go Systems-on-chip (SOC)!
- Today, Intel-PIII has > 10M transistors (up to 1.13 GHz; 0.18 um)
- Today we produce > 10M transistors/person (1 billion/person by 2008)
- Semiconductor/IC: #1 key field for advancing into Y2K (Business Week, Jan. 1995)

1-3

The First Transistor

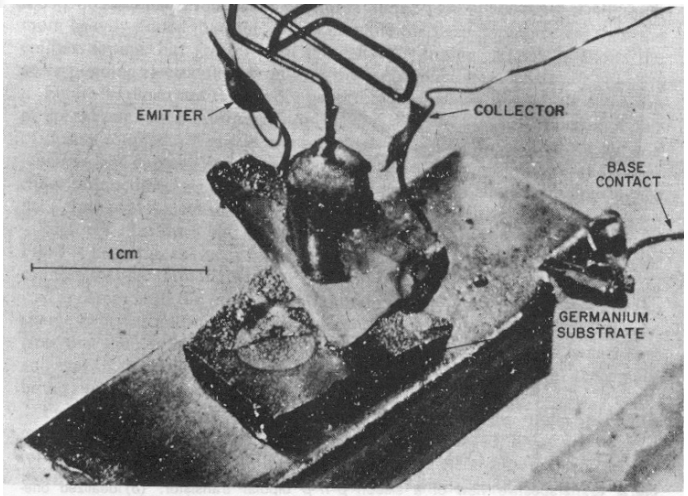


Fig. 1 The first transistor.[1]

1-4

Pioneers of the Electronic Age

[2] Gordon Moore [right] relaxes with fellow pioneers of the electronic age: Robert Noyce [center] and Andrew Grove [left]. Moore and Noyce contributed to the development of the planar IC. Grove is now president and chief executive officer of Intel Corp.

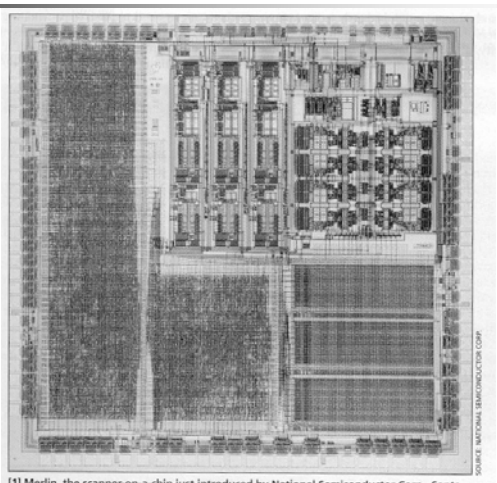
1-5

Silicon Wafers

A coalition of semiconductor makers, including Austin's Sematech, has released a report detailing where the industry is headed. Among the projects is increasing silicon wafer diameters from 8 inches to 12 inches. Such a move would greatly reduce the cost of making chips.

1-6

Scanner on a Chip



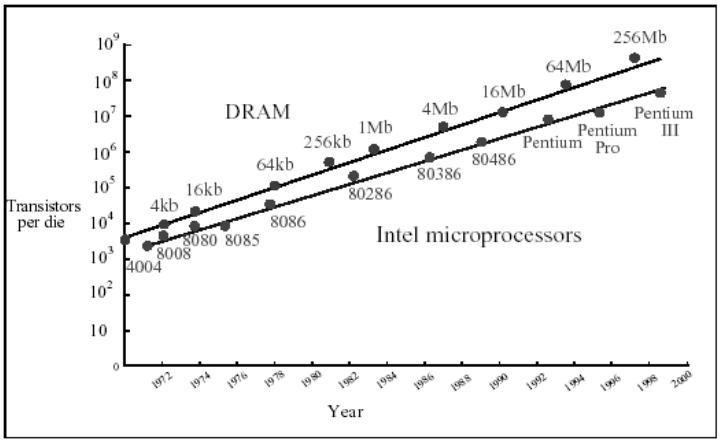
National Semiconductor
IEEE Spectrum
Jan. 1999, p.57

[1] Merlin, the scanner-on-a-chip just introduced by National Semiconductor Corp., Santa Clara, Calif., includes an analog front end, sensor clock generation, motor control, data buffering, and parallel port interface, all in a dime-sized area. Chips like this one, which load up with components in much the same way as boards load up with chips, are being dragged into existence by the demands of consumer electronics.

1-7

Moore's Law

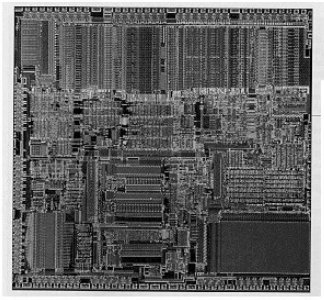
- Logic capacity doubles per IC per year at regular intervals (1965)
- Logic capacity doubles per IC every 18 months (1975)



1-8

The Dies of Intel CPUs

Pentium Pro



386

4004

1-9

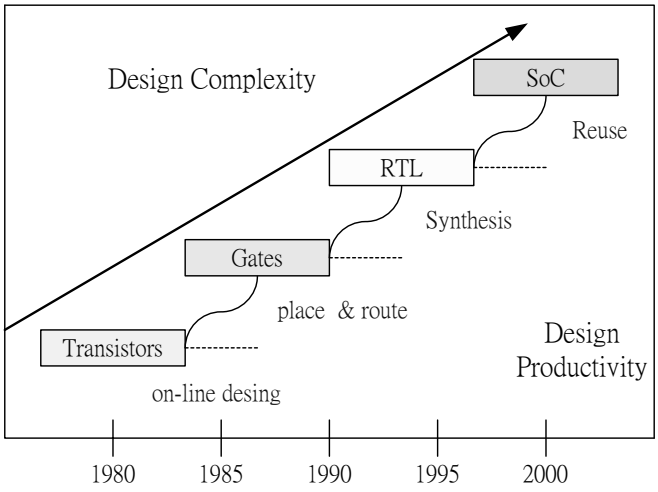
Semiconductor Technology Roadmap

Year	1997	1999	2002	2005	2008	2011	2014
Technology node (nm)	250	180	130	100	70	50	35
On-chip local clock (GHz)	0.75	1.25	2.1	3.5	6.0	10	16.9
Microprocessor chip size (mm^2)	300	340	430	520	620	750	901
Microprocessor transistor/chip	11M	21M	76M	200M	520M	1.40B	3.62B
Microprocessor cost/transistor ($x10^{-8}$ USD)	3000	1735	580	255	110	49	22
DRAM bits per chip	256M	1G	4G	16G	64G	256G	1T
Wiring level	6	6-7	7	7-8	8-9	9	10
Supply voltage (V)	1.8-2.5	1.5-1.8	1.2-1.5	0.9-1.2	0.6-0.9	0.5-0.6	0.37-0.42
Power (W)	70	90	130	160	170	175	183

Source: SIA99

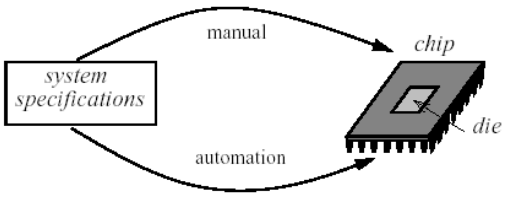
1-10

Trends of VLSI Design



1-11

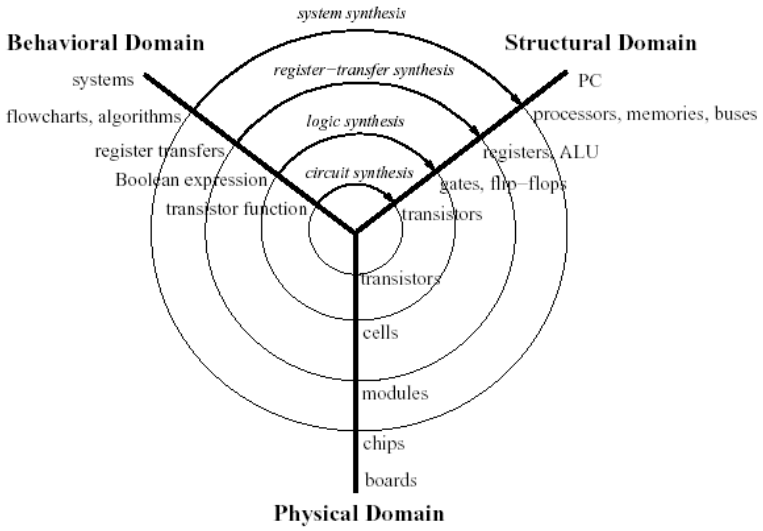
VLSI Design Considerations



- Several conflicting considerations
  - **Design complexity**: large number of devices/transistors
  - **Cost**: die area, packing, testing, etc.
  - **Performance**: optimization requirements for high performance
  - **Time-to-market**: about 15% gain for early birds
  - Others: power, noise, testability, reliability, manufacturability, etc.
- Keys: hierarchical design, abstraction, **CAD**

1-12

Design Representation



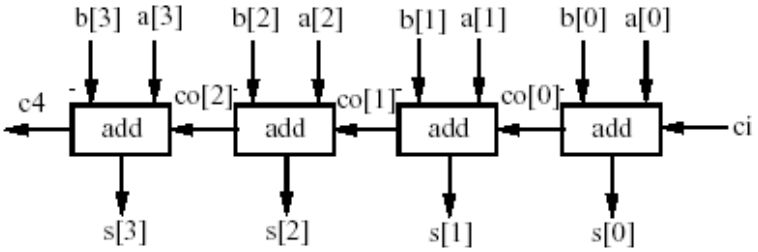
Three Different Domains

- Behavior
  - Functionality of components

```
module add4 (s, c4, ci, a, b);
        input [3:0] a, b;
        input ci;
        output [3:0] s;
        output c4;
        wire [2:0] co;
                add f0 (co[0], s[0], a[0], b[0], ci);
                add f1 (co[1], s[1], a[1], b[1], co[0]);
                add f2 (co[2], s[2], a[2], b[2], co[1]);
                add f3 (c4, s[3], a[3], b[3], co[2]);
endmodule
```

Three Different Domains

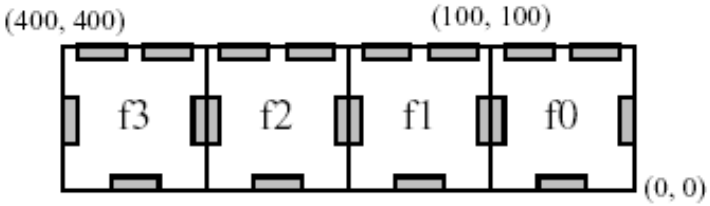
- Structural
  - Connectivity between components



1-15

Three Different Domains

- Physical
  - A layout description



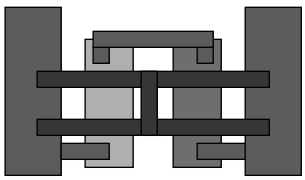
1-16

Hierarchy of Description

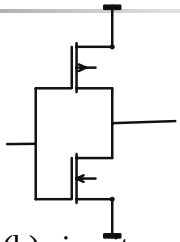
- System Level
- Algorithm Level
- Register Transfer Level
- Logic Gate Level
- Switch Level

1-17

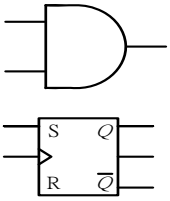
Design on Different levels



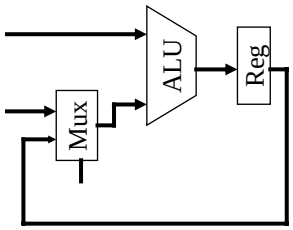
(a) silicon



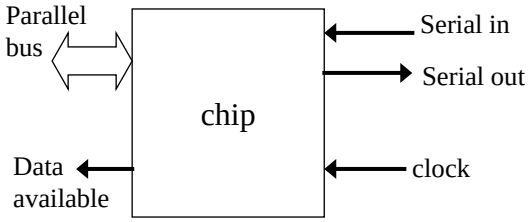
(b) circuit



(c) gate F-F



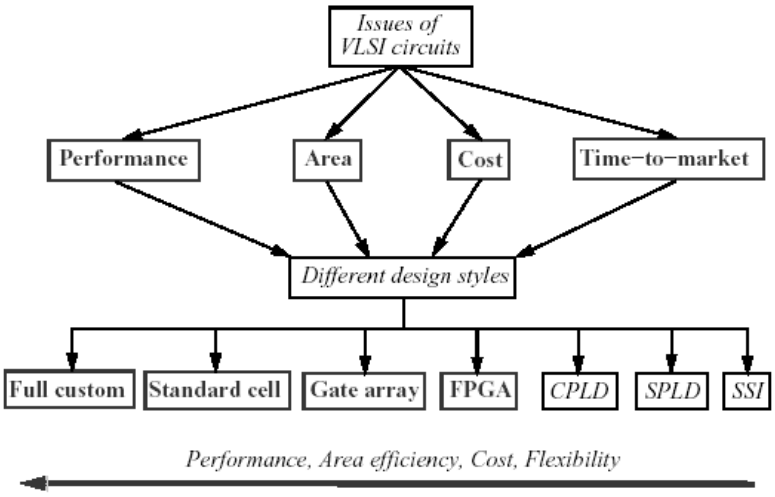
(d) Register



(e) Chip

1-18

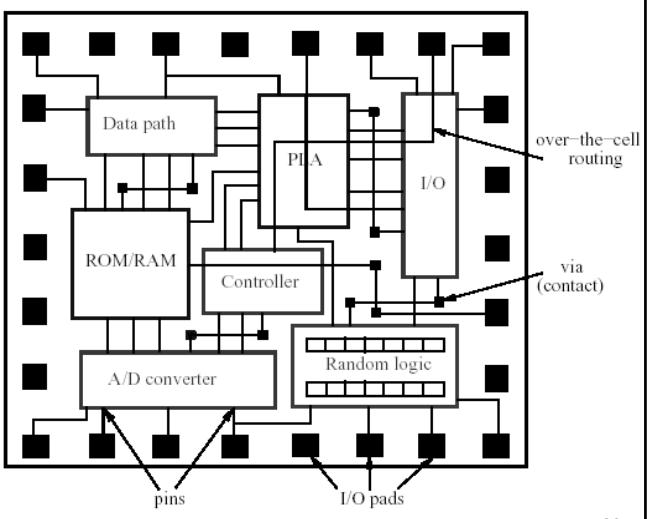
Design Styles



1-19

Full Custom Design Style

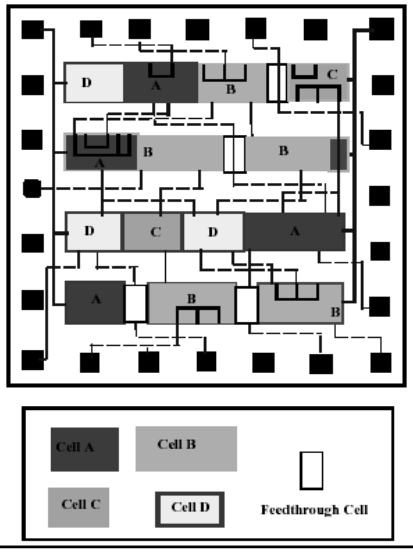
- Handcraft functional & physical designs
- Design efforts and cost are high
- Expect high quality and high volume



1-20

Cell-Based Design

- Cells are characterized and stored in library
- Need update when technology advances
- Compatible to custom designs
- Easier to develop CAD tools for design and optimization



·21

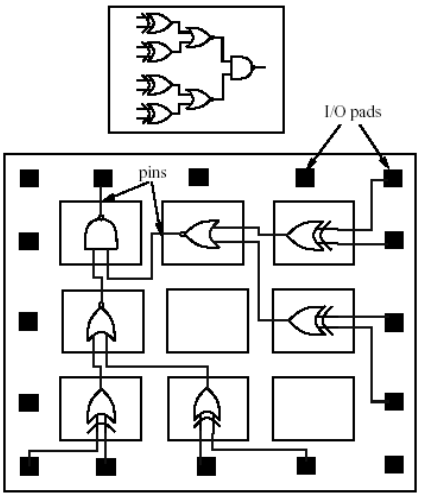
Standard Cell Design Style

- Cell-based: standard cells, macro cells
  - Standard cells: FFs, AND, OR..
  - Macro cells: Memory, PLA, ALU…
- Manufacturing process is not simplified
- Design process is simplified
- Need tremendous characterization effort
- Parameterized area and delay over ranges of temperatures and operating voltage

1-22

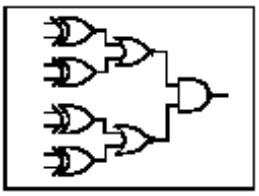
Gate Array Design Style

- Arrays of sites are pre-manufactured
- Metal and contact layers are used to program the chip
- Fewer manufacturing steps correlate to lower fabrication time and cost

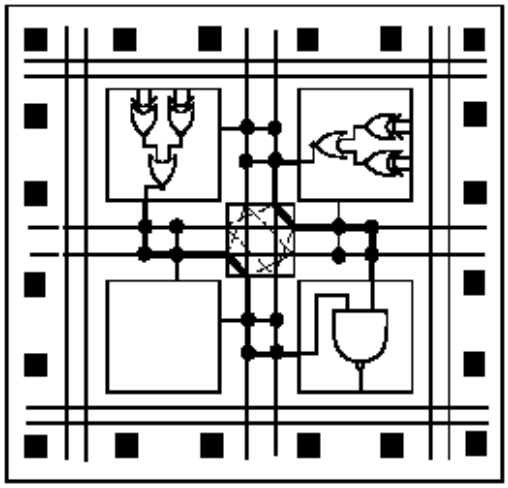


1-23

CPLD/FPGA Design Style



- Illustrated by a symmetric array-based FPGA



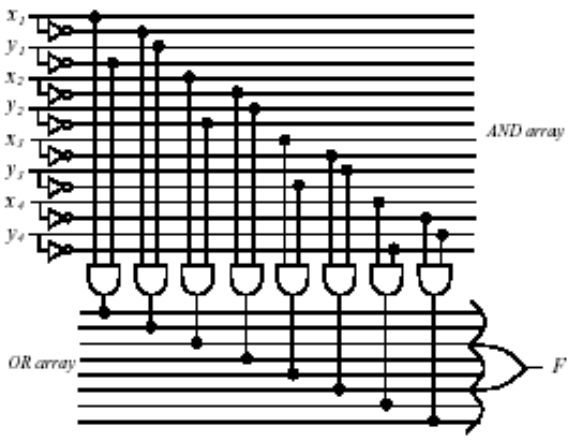
1-24

Field Programmable Gate Array

- Arrays of programmable modules with the capability of implementing a generic logic function
- Wires can be connected by programming antifuses
- Reduce development and production time
- Low cost prototyping

1-25

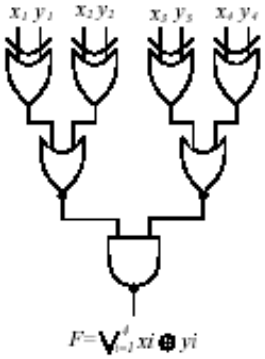
SPLD Design Style



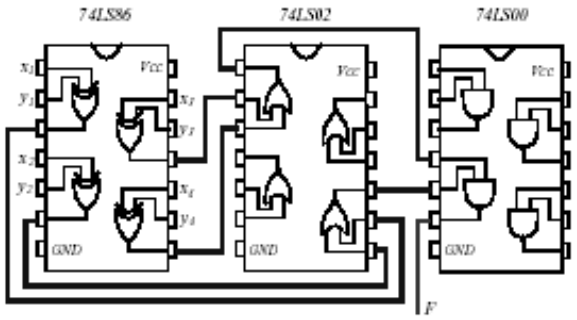
SPLD (PLA) implementation

1-26

SSI Design Style



(a) 4−bit comparator.



(b) SSI implementation.

* SSI = Small Scaled Integrated circuit

1-27

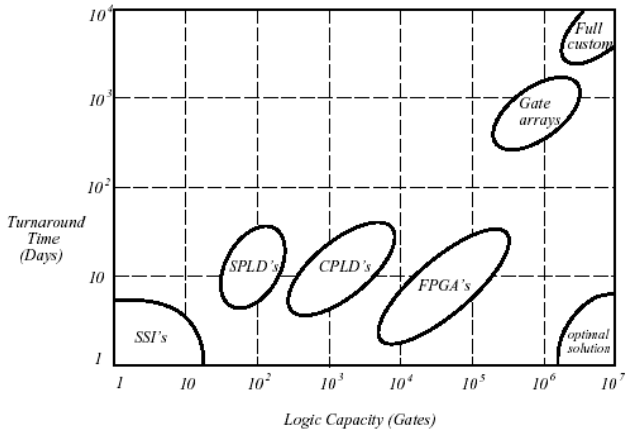
Comparison of Design Styles

	Full custom	*Standard cell*	*Gate array*	*FPGA*	*SPLD*
Density	Very high	High	High	Medium	Low
Performance	Very high	High	High	Medium	Low
Flexibility	Very high	High	Medium	Low	Low
Design time	Very long	Short	Short	Very short	Very short
Manufacturing time	Medium	Medium	Short	Very short	Very short
Unit cost -- small quantity	Very high	High	High	Low	Very Low
Unit cost – large quantity	Low	Low	Low	High	Very High

1-28

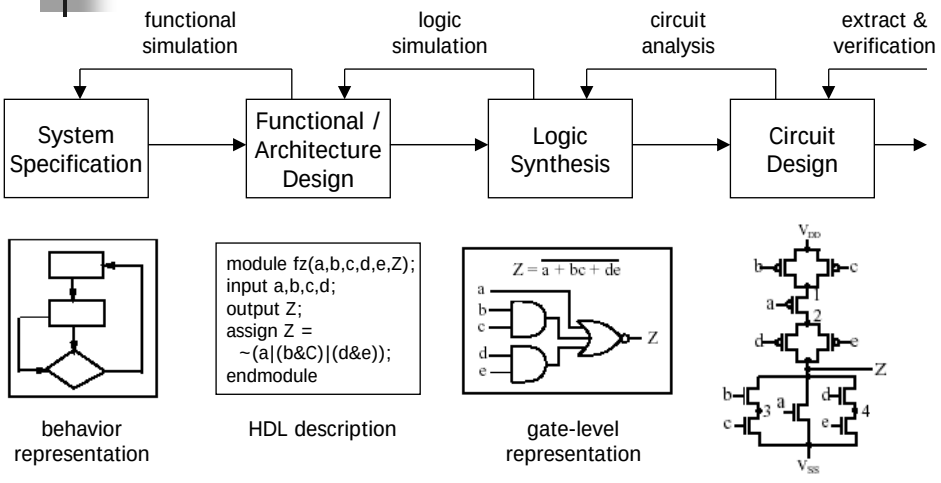
Design-Style Trade-offs

- Performance/cost tradeoff
  - Customize design to achieve desired performance



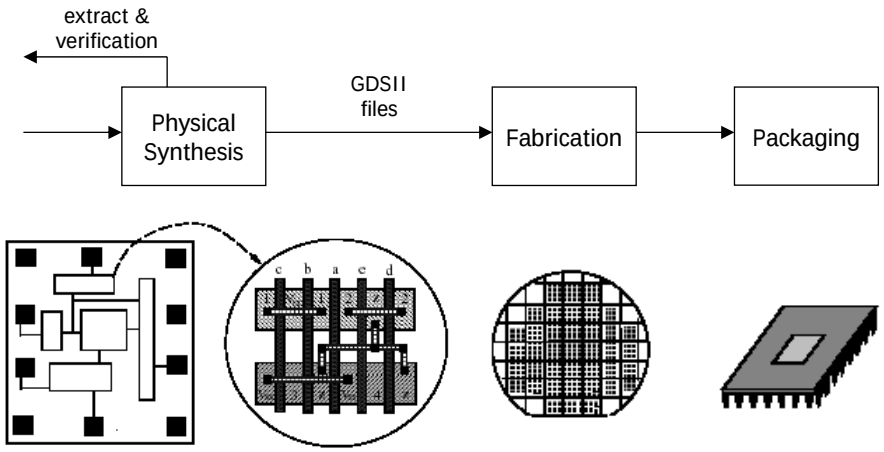
1-29

Traditional VLSI Design Flow



1-30

Traditional VLSI Design Flow



1-31

Specification to Circuits

- Specification: Specify *what* the system does
  - The functional appearance of a system to its users
- Design: Determine
  - The structure of a system
  - Different methods to achieve a function
  - Logical structures that perform the architecture
- Realization: Materialize the physical structures
  - Technology decision (CMOS, ECL)
  - Design style decision (Full Custom, Semi- Custom ...)

1-32

Design Specification

- Informal description by English
  - These full adders perform the addition of two 2-bit binary numbers. The sum output is provided for each bit and the resultant carry is obtained from the second bit. Design for medium-to-high speed, multiple bits, parallel-add/serial carry applications.

1-33

Design Specification

- Formal description using HDL
  - Verify the specification through simulation or verification
  - Easy to change
  - Enable automatic synthesis

1-34

Hardware Description Language

- Can describe a design at some levels of abstraction
  - Behavioral, RTL, Gate-level
- Can be used to document the complete system design tasks
  - testing, simulation … related activities
- Comprehensive and easy to learn

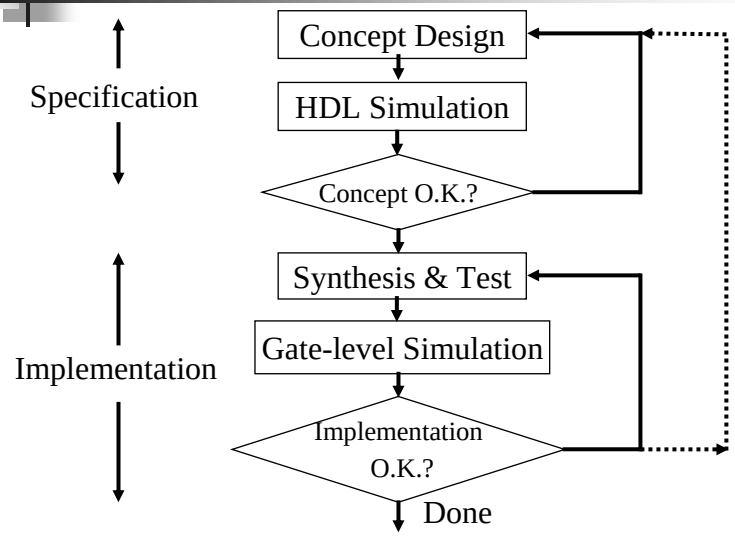
1-35

HDL-Based Design Flow

- Describe desired functionality and timing using HDL such as VHDL, Verilog
- Use high-level synthesis tool to obtain structural level design
- Using placement and routing tools to obtain physical level design
- Very popular design approach for standard cell and gate array design

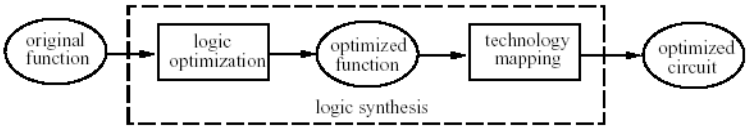
1-36

HDL-Based Design Flow



1-37

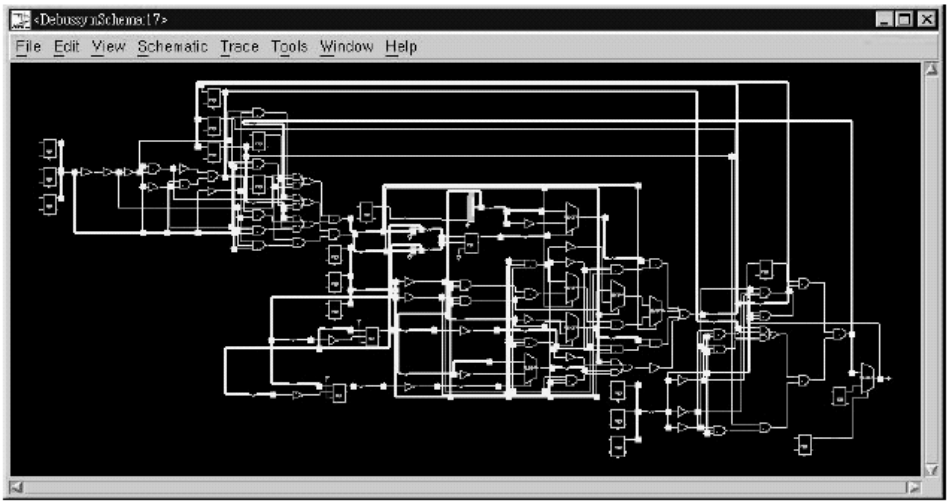
Logic Synthesis



- Logic synthesis programs transform HDL descriptions into logic gate networks in a particular library
  - Involve translation and optimization
- Optimization criteria: area, performance, power, and testability
- Logic optimization: technology independent
  - Optimize Boolean expressions but still keep their equivalence
- Technology mapping: technology dependent
  - Map Boolean expressions into a particular cell library

1-38

Gate-Level Netlist



1-39

Drivers for Synthesis: Competition

- Time to market
  - Delayed market entry means lost revenue
- Reducing costs
  - 2-4X more complex, 2-4X higher quality for 50% of the cost as the last system
- Higher quality: shipping fewer defective products
- Managing the complexity
- Technology independent

1-40

Synthesis v.s. Schematic Capture

- Traditional way of design
  - Architecture design
  - Functional design with logic
  - Design capture using schematic capture tools
- HDL-based design
  - Architecture design
  - Functional capture using HDL
  - Implementation using synthesis tools

1-41

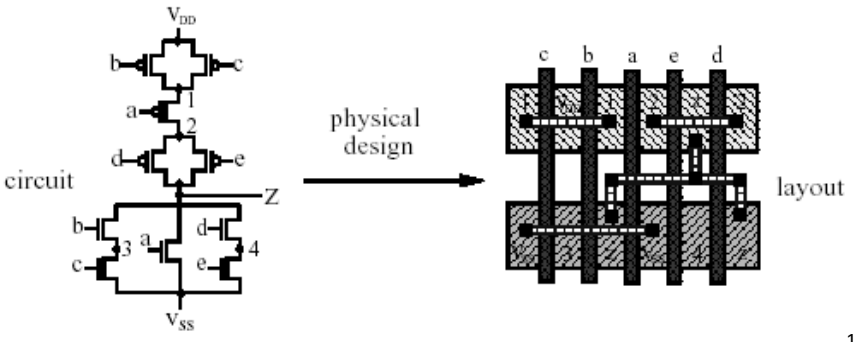
How HDL-Based Design Helps

- Shortens the design verification loop
- Allows architectural tradeoffs with short turnaround
- Reduces time for design capture
- Encourages focus on functionality
- Ensures design documentation consistency
- Facilitates design modification

1-42

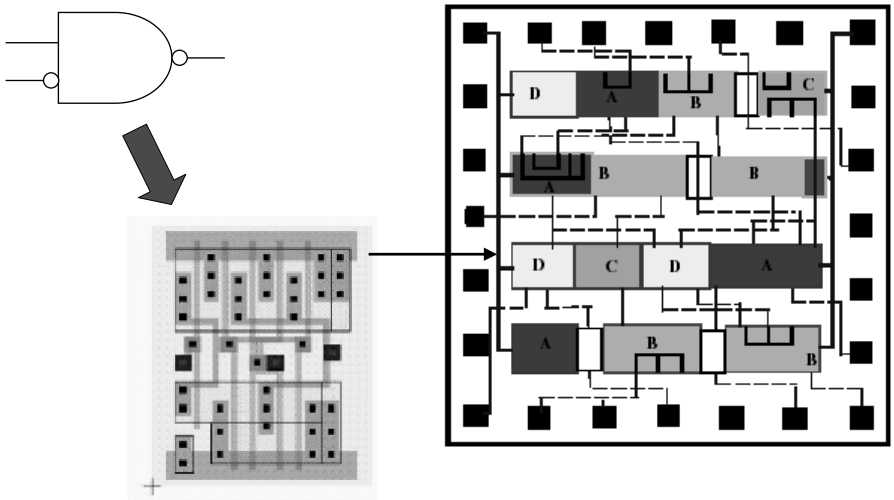
Physical Design

- Physical design converts a circuit description into a geometric description (GDSII file)
- The description is used to manufacture a chip



1-43

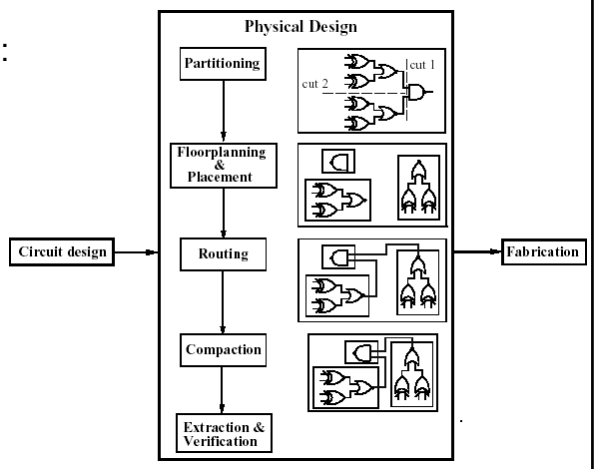
Standard Cells



1-44

Physical Design Flow

- Physical design cycle:
  1. Logic partitioning
  2. Floorplanning, placement, and pin assignment
  3. Routing (global and detailed)
  4. Compaction
  5. RLC extraction & verification



1-45

PowerPC Chip Floorplan

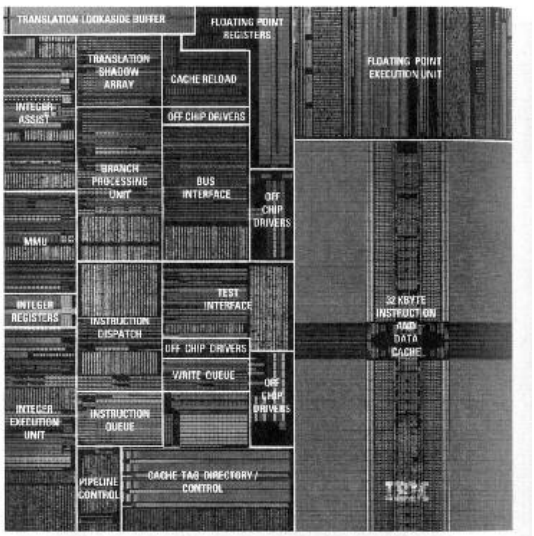
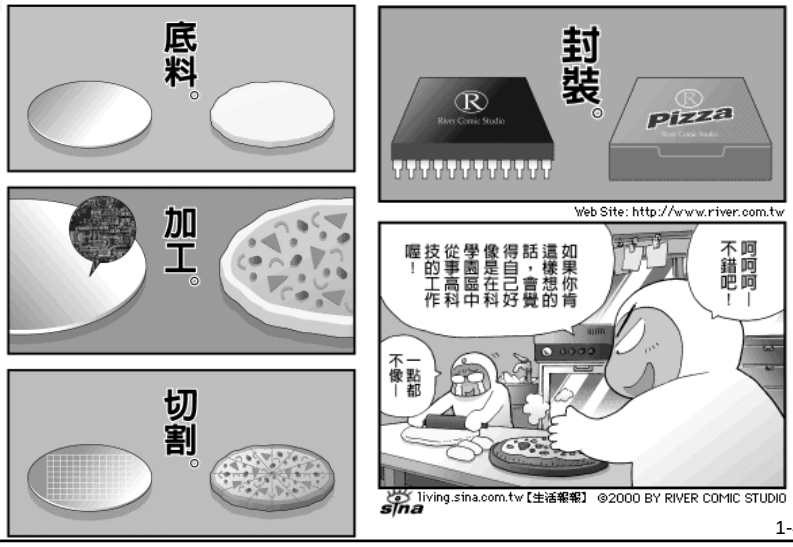


Figure 9-30 A Power PC microprocessor chip with various functional areas delineated. This microprocessor uses the *reduced instruction set computing (RISC)* architecture. (Photograph courtesy of IBM Corp.)

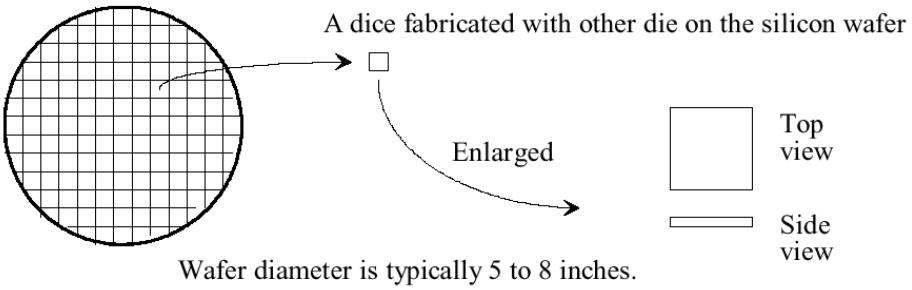
1-46

IC Fabrication



1-47

Wafer and Dices



Source: http://cmosedu.com/cmos1/doslasi/doslasi.pdf

1-48

Sawing a Wafer into Chips

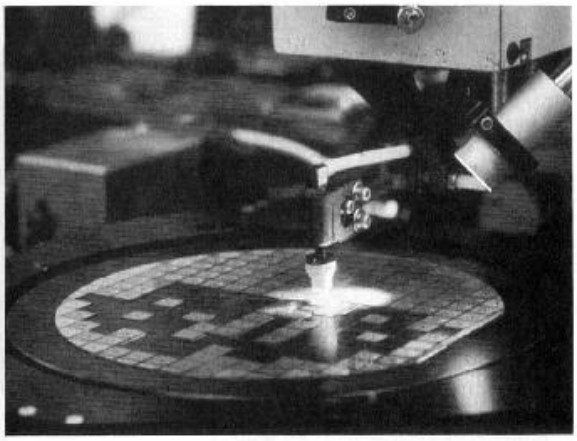
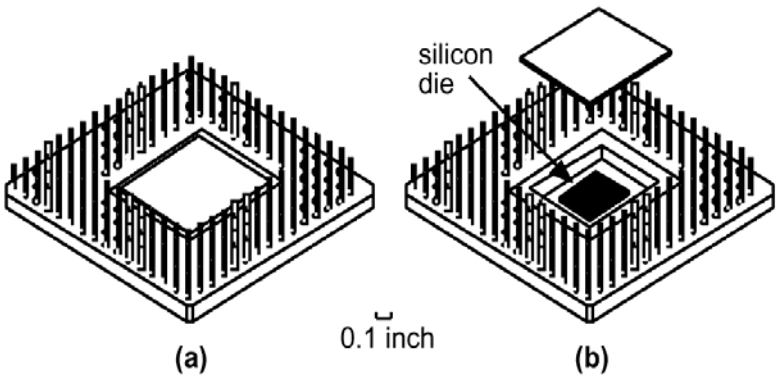


Figure 9-33
After testing and sawing, the individual chips are picked up by a robotic arm and placed in the package for die bonding. (Photograph courtesy of Intel Corp.)

1-49

IC and Die



1-50

Chip Packing

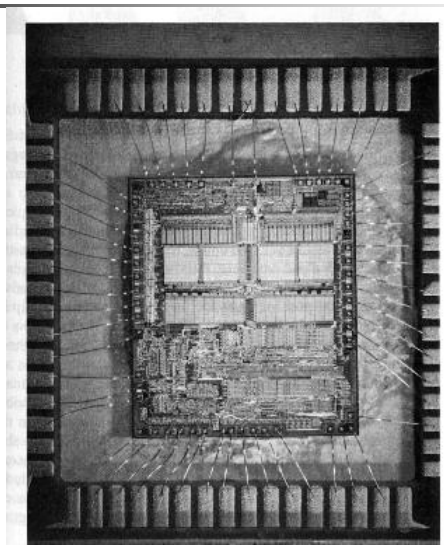
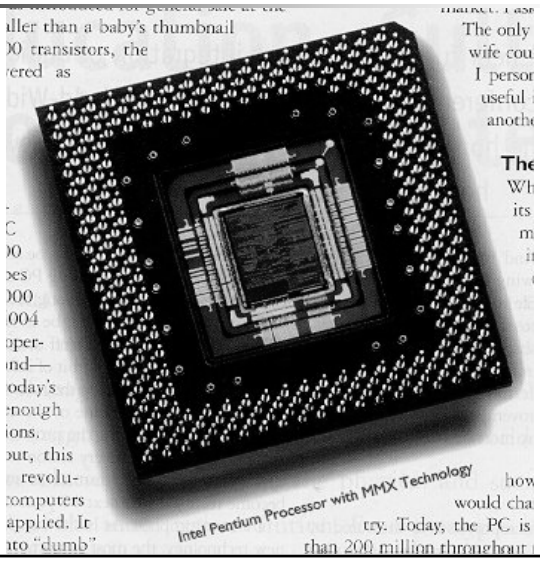


Figure 9-34
Attachment of leads from the Al pads on the periphery of the chip to posts on the package. (Photograph courtesy of Motorola, Inc.)

1-51

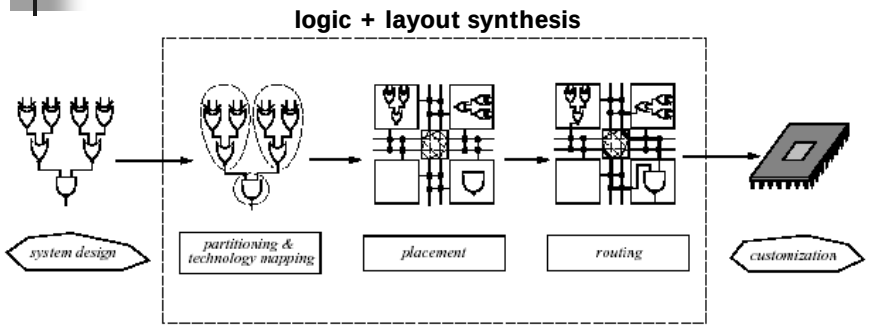
Pentium-MMX with PGA Packging



Intel Pentium Processor with MMX Technology

1-52

FPGA Design Flow



- Advantages: Fast and reusable prototyping
  - Can be reprogrammed and reused
  - Implementation time is very short
- Disadvantages: Expensive and high volume

1-53

FPGA Structure (Altera)

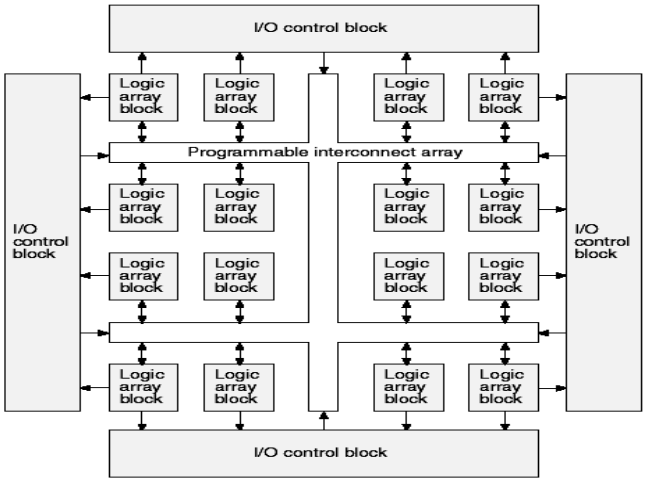


Fig. 6-28 Altera® MAX 7000™ Structure (Reprinted with Permission of Altera Corporation,© Altera Corp., 1991)

1-54

FPGA Structure (Xilinx)

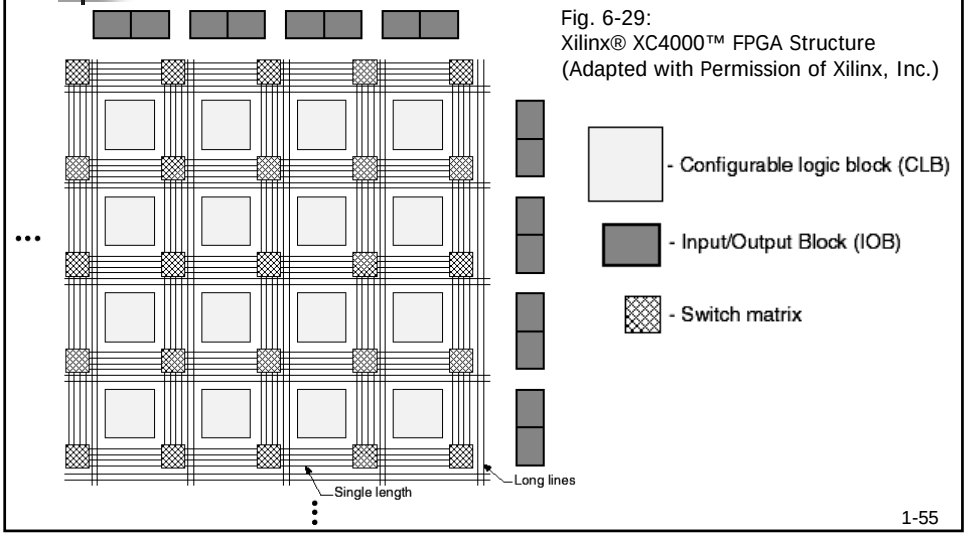
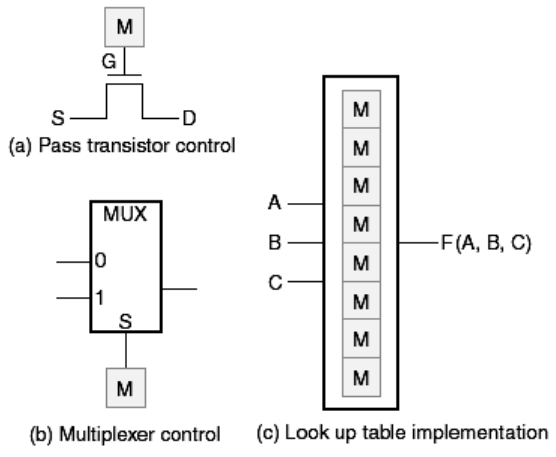


Fig. 6-29:
Xilinx® XC4000™ FPGA Structure
(Adapted with Permission of Xilinx, Inc.)

1-55

Store the Programming Info.

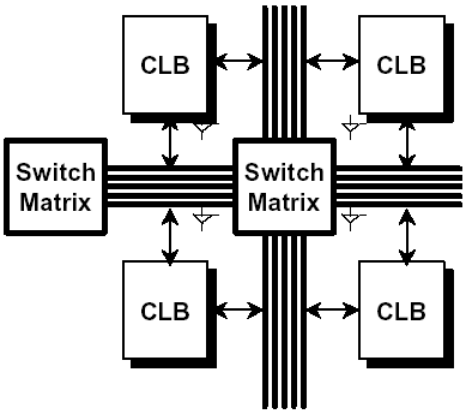
- SRAM technology is used
  - M = 1-bit SRAM
  - Loaded from the PROM after power on
- Store control values
  - Control pass transistor
  - Control multiplexer
- Store logic functions
  - Store the value of each minterm in the truth table



1-56

Xilinx FPGA Routing

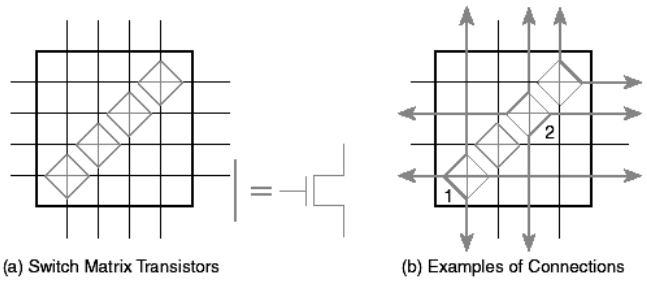
- Fast direct interconnect
  - Adjacent CLBs
- General purpose interconnect
  - CLB – CLB or CLB – IOB
  - Through switch matrix
- Long lines
  - Across whole chip
  - High fan-out, low skew
  - Suitable for global signals (CLK) and buses
  - 2 tri-states per CLB for busses



1-57

Xilinx Switch Matrix

- Six pass transistors to control each switch node
- The two lines at point 1 are joined together
- At point 2, two distinct signal paths pass through one switch node



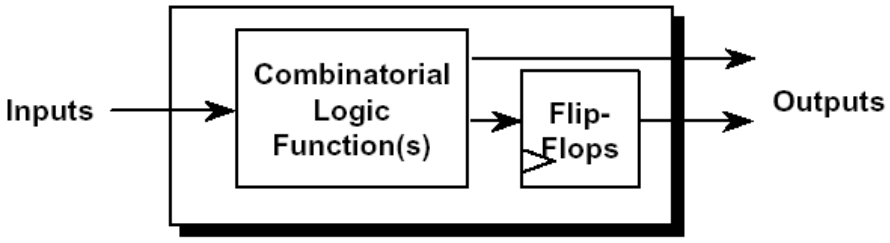
(a) Switch Matrix Transistors (b) Examples of Connections

Fig. 6-31 Example of Xilinx® Switch Matrix (Adapted with Permission of Xilinx®, Inc.)

1-58

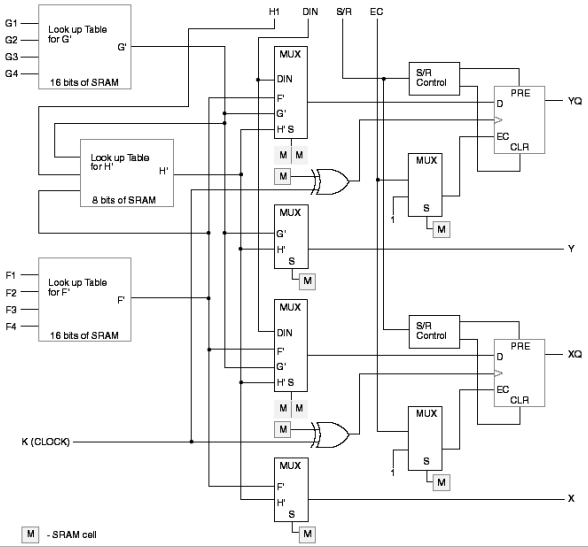
Configurable Logic Block (CLB)

- Combinational logic via lookup table
  - Any function(s) of available inputs
- Output registered and/or combinational



1-59

Simplified CLB Structure



1-60

Internal Functions of a CLB

- Two 4-input tables implement two distinct functions (F' and G')
- F' and G' with another control (H1) feed into a third lookup table (H')
- Two arbitrary functions of up to four variables and selected functions of up to nine variables can be implemented
- Properly setting the two MUXes can assign any pair of F', G', and H' to the two combinational outputs (X and Y)

1-61

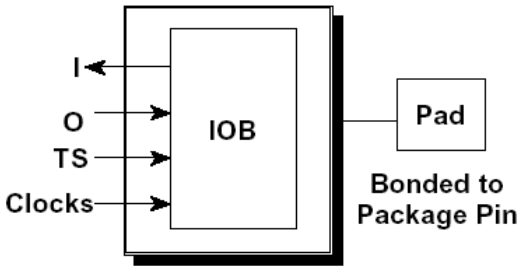
Internal Functions of a CLB

- Two D flip-flops directly drive outputs XQ and YQ
- Each of the D inputs can be selected from F', G', H' and input DIN
- Two XORs select each flip-flop individually to be positive or negative edge triggered
- Two SR controls select the signal S/R to be an asynchronous Set or Reset for the flip-flops
- Two multiplexers allow the input EC to optionally act as a clock ENABLE signal for each flip-flop

1-62

I/O Block (IOB)

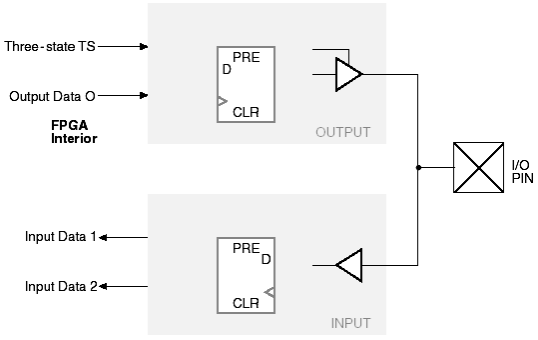
- Periphery of identical I/O blocks
  - Input, output, or bidirectional
  - Registered, latched, or combinational
  - Three-state output
  - Programmable output slew rate



1-63

Input/Output Mode of an IOB

- Input
  - 3-state control places the output buffer into high impedance
  - Direct in and/or registered in
- Output
  - 3-state driver should be enabled by TS signal
  - Direct output or registered output



1-64

Design with FPGA

- Using HDL, schematic editor, SM chart or FSM diagram to capture the design
- Simulate and debug HDL codes
- Work out detail logic manually or by using synthesis tools
- Simulate and debug again at logic level
- Feed the logic into CLBs and IOBs
  - Completed by a CAD tool

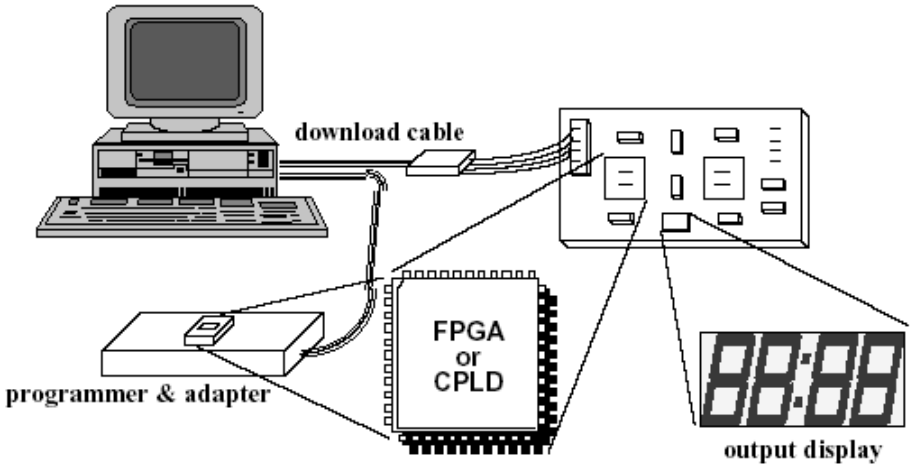
1-65

Design with FPGA (cont'd)

- Partitioning (Technology mapping)
  - Chip-level
  - CLB-level
- Place and route
- Generate bit pattern for programming the FPGA
- Download the bit pattern into the internal configurable memory cells and test the operations

1-66

Download to a FPGA demo board



1-67

Acknowledgement

- Some of the presentation materials were modified from the previous works of
  - Prof. Jing-Yang Jou (周景揚, Dept. of EE, NCTU)
  - Prof. Yao-Wen Chang (張耀文, Dept. of EE, NTU)
  - Ms. I-Ling Chen (陳怡伶, ITRI/STC)
- Their supports are greatly appreciated

1-68