# COE 561
# Digital System Design & Synthesis
# Introduction

**Dr. Aiman H. El-Maleh**
**Computer Engineering Department**
**King Fahd University of Petroleum & Minerals**

## Outline

- Course Topics
- Microelectronics
- Design Styles
- Design Domains and Levels of Abstractions
- Digital System Design
- Synthesis Process
- Design Optimization

# Course Topics …

- **INTRODUCTION**
  - Microelectronics, semiconductor technologies, microelectronic design styles, design representations, levels of abstraction & domains, Y-chart, system synthesis and optimization, issues in system synthesis. **0.5 week**
- **MODELING OF DIGITAL SYSTEMS**
  - **Introduction to Hardware description languages(HDLs). Hardware Description and design using VHDL. Basic modeling concepts, Language elements, Behavioral modeling, Dataflow modeling, Structural modeling, some hardware modeling examples. 1.5 weeks**
- **LOGIC SYNTHESIS 6.5 weeks**
- Introduction to logic synthesis
  - **Boolean functions representation, Binary Decision Diagrams, Satisfiability and Cover problems 0.5 week**

1-3

# … Course Topics …

- Two-level logic synthesis and optimization
  - **Logic minimization principles, Exact logic minimization, Heuristic logic minimization, The Espresso minimizer, Testability properties of two-level circuits. 1 week**
- Multi-level logic synthesis and optimization
  - **Models and transformations of combinational networks: elimination, decomposition, extraction.**
  - **The algebraic model: algebraic divisors, kernel set computation, algebraic extraction and decomposition.**
  - **The Boolean model: Don't care conditions and their computations, input controllability and output observability don't care sets, Boolean simplification and substitution.**
  - **Optimization based on redundancy addition and removal. Transduction, Global flow.**
  - **Testability properties of multilevel circuits.**
  - **Synthesis of minimal delay circuits. Rule-based systems for logic optimization. 2 weeks**

1-4

## … Course Topics …

- Sequential Logic Synthesis
  - **Introduction to FSM Networks, Finite state minimization, state encoding: state encoding for two-level circuits, state encoding for multilevel circuits, Finite state machine decomposition, Retiming, Implicit finite state machine traversal methods, and Testability consideration for synchronous sequential circuits. 2 weeks**
- Technology Mapping
  - **Problem formulation and analysis, Library binding approaches – Structural matching, Boolean matching, Covering & Rule based approach, Case studies – Mapping the design onto FPGAs. 1 week**

1-5

## … Course Topics …

- **HIGH LEVEL SYNTHESIS 6.5 weeks**
- **Introduction to High level synthesis**
- **Design representation and transformations**
  - Design flow in high level synthesis, HDL compilation, internal representation (CDFG), data flow and control sequencing graphs, data-flow based transformations. **0.5 week**
- **Architectural Synthesis**
  - Circuit specifications: resources and constraints, scheduling, binding, area and performance optimization, datapath synthesis, control unit synthesis, synthesis of pipelined circuits. **2 weeks**

1-6

## … Course Topics

- **Scheduling**
  - Unconstrained scheduling: ASAP scheduling, Latency-constrained scheduling: ALAP scheduling, time-constrained scheduling, resource constrained scheduling, Heuristic scheduling algorithms: List scheduling, force-directed scheduling. **2 weeks**
- **Allocation and Binding**
  - resource sharing, register sharing, multi-port memory binding, bus sharing and binding, unconstrained minimum-performance-constrained binding, concurrent binding and scheduling. **2 weeks**

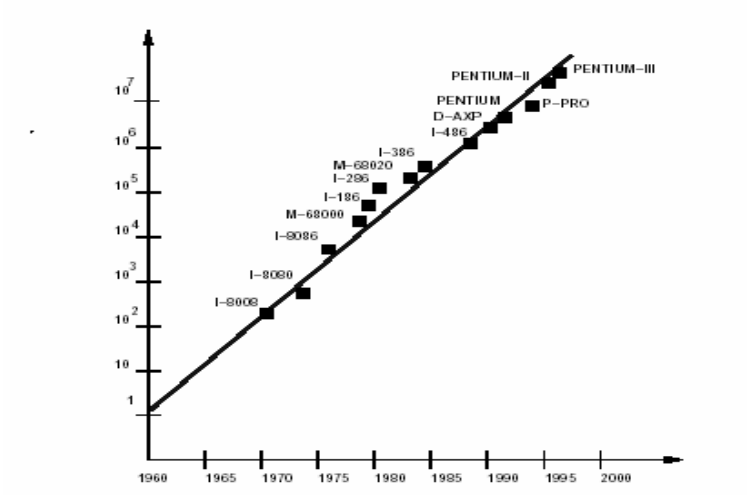
1-7

## Microelectronics

- **Enabling and strategic technology for development of hardware and software**
- **Primary markets:**
  - Information systems.
  - Telecommunications.
  - Consumer.
- **Trends in microelectronics**
  - Improvements in device technology:
    - Smaller circuits.
    - Higher performance.
    - More devices on a chip.
  - Higher degree of integration.
    - More complex systems.
    - Lower cost in packaging and interconnect.
    - Higher performance.
    - Higher reliability.

1-8

# Moore's Law




1-9

# Microelectronic Design Problems

- **Use most recent technologies: to be competitive in performance.**
- **Reduce design cost: to be competitive in price.**
- **Speed-up design time: Time-to-market is critical.**
- **Design Cost**
  - Design time and fabrication cost.
  - Large capital investment.
  - Near impossibility to repair.
- **Recapture costs:**
  - Large volume production is beneficial.
  - Zero-defect designs are essential.
  - Follow market evolution.

1-10

## Microelectronic Circuits

- General-purpose processors:
  - **High-volume sales.**
  - **High performance.**
- Application-Specific Integrated Circuits (ASICs):
  - **Varying volumes and performances.**
  - **Large market share.**
- Prototypes.
- Special applications (e.g. space).

1-11

## Computer-Aided Design

- **Enabling design methodology.**
- **Makes electronic design possible:**
  - Large scale design management.
  - Design optimization.
    - Feasible implementation choices grow rapidly with circuit size
  - Reduced design time.
- **CAD tools have reached good level of maturity.**
- **Continuous grows in circuit size and advances in technology requires CAD tools with increased capability.**
- **CAD tools affected by**
  - Semiconductor technology
  - Circuit type

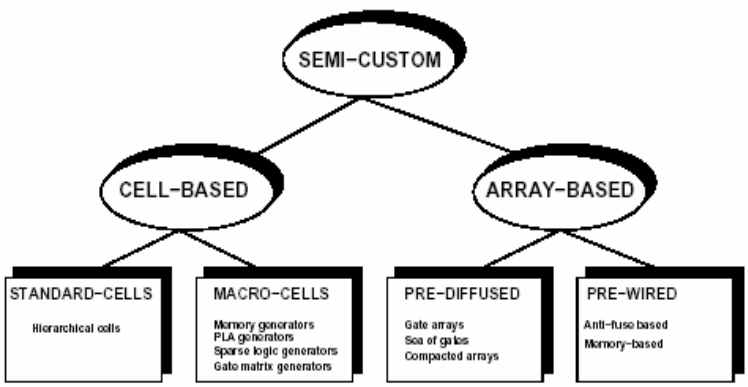
1-12

# Microelectronics Design Styles

- Adapt circuit design style to market requirements
- Parameters:
  - **Cost.**
  - **Performance.**
  - **Volume.**
- **Full custom**
  - Maximal freedom
  - High performance blocks
  - Slow
- **Semi-custom**
  - Standard Cells
  - Gate Arrays
    - Mask Programmable (MPGAs)
    - Field Programmable (FPGAs))
  - Silicon Compilers & Parametrizable Modules (adder, multiplier, memories)

1-13

# Semi-Custom Design Styles




1-14

## Standard Cells

- Cell library:
  - **Cells are designed once.**
  - **Cells are highly optimized.**
- Layout style:
  - **Cells are placed in rows.**
  - **Channels are used for wiring.**
  - **Over the cell routing.**
- Compatible with macro-cells (e.g. RAMs).

1-15

## Macro-cells

- **Module generators:**
  - Synthesized layout.
  - Variable area and aspect-ratio.
- **Examples:**
  - RAMs, ROMs, PLAs, general logic blocks.
- **Features:**
  - Layout can be highly optimized.
  - Structured-custom design.

1-16

## Array-based design

- **Pre-diffused arrays:**
  - Personalization by metalization/contacts.
  - Mask-Programmable Gate-Arrays.
- **Pre-wired arrays:**
  - Personalization on the field.
  - Field-Programmable Gate-Arrays.

1-17

## MPGAs & FPGAs

- **MPGAs:**
  - Array of sites:
    - Each site is a set of transistors.
  - Batches of wafers can be pre-fabricated.
  - Few masks to personalize chip.
  - Lower cost than cell-based design.
- **FPGAs:**
  - Array of cells:
    - Each cell performs a logic function.
  - Personalization:
    - Soft: memory cell (e.g. Xilinx).
    - Hard: Anti-fuse (e.g. Actel).
  - Immediate turn-around (for low volumes).
  - Inferior performances and density.
  - Good for prototyping.

1-18

## Semi-custom style trade-off

| | *Custom* | *Cell-based* | *Pre-diff.* | *Pre-wired* |
|---|---|---|---|---|
| Density | Very High | High | High | Medium-Low |
| Performance | Very High | High | High | Medium-Low |
| Flexibility | Very High | High | Medium | Low |
| Design time | Very Long | Short | Short | Very Short |
| Man. time | Medium | Medium | Short | Very Short |
| Cost - lv | Very High | High | High | Low |
| Cost - hv | Low | Low | Low | Medium-High |

1-19

## Example: AT&T ASIC Chip

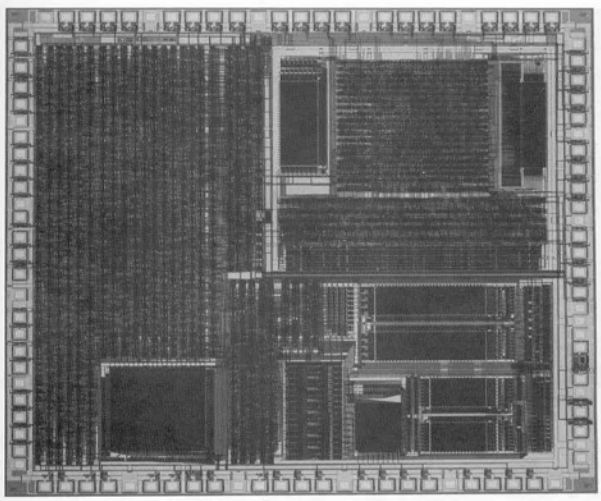


FIGURE 1.3
One of AT&T's Application Specific Standard Product chips. The chip was designed and laid out using AT&T CAD tools, with a standard cell design style. (Courtesy of AT&T.)

1-20

## Example: DEC AXP Chip designed using Macro Cells

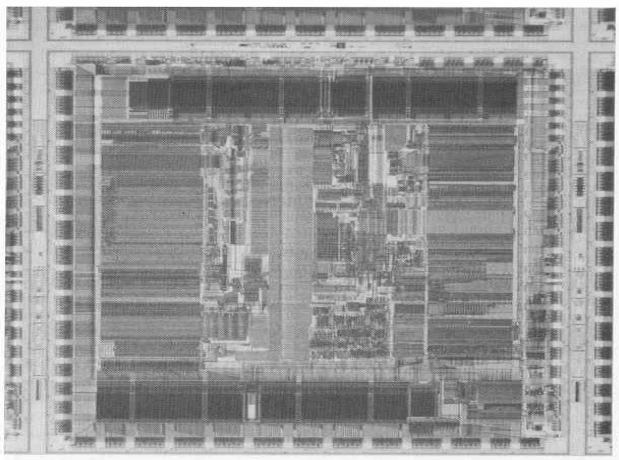


FIGURE 1.4
Microphotograph of the Alpha AXP chip by Digital Equipment Corporation, using several macr
designed with proprietary CAD tools. (Courtesy of Digital Equipment Corporation.)

1-21

## Example: Mask Programmable Gate Array from IBM Enterprise System 9000

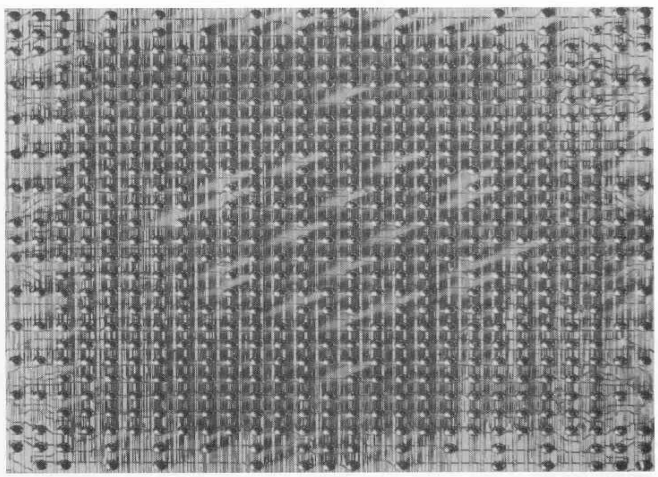
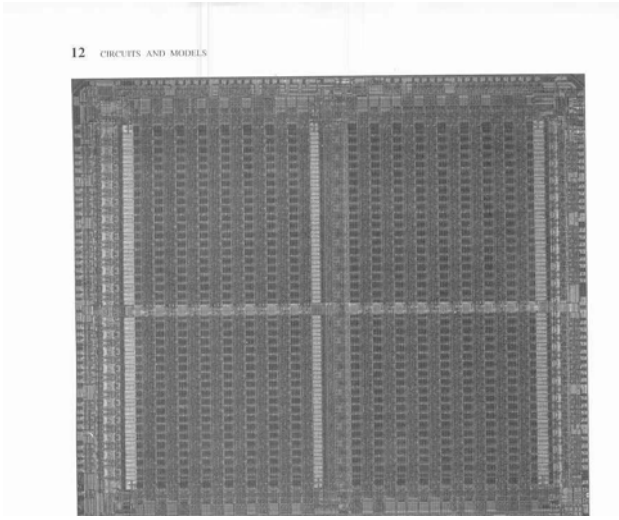


FIGURE 1.5
A mask programmable gate array from the IBM Enterprise System/9000 air-cooled processor technology.
(Courtesy and copyright of IBM, reprinted with permission.)

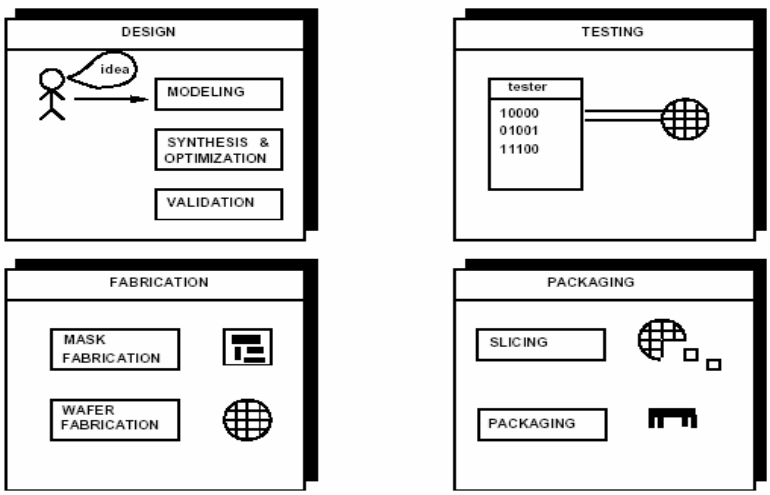
1-22

## Example: Field Programmable Gate Array from Actel




1-23

## Microelectronic Circuit Design and Production



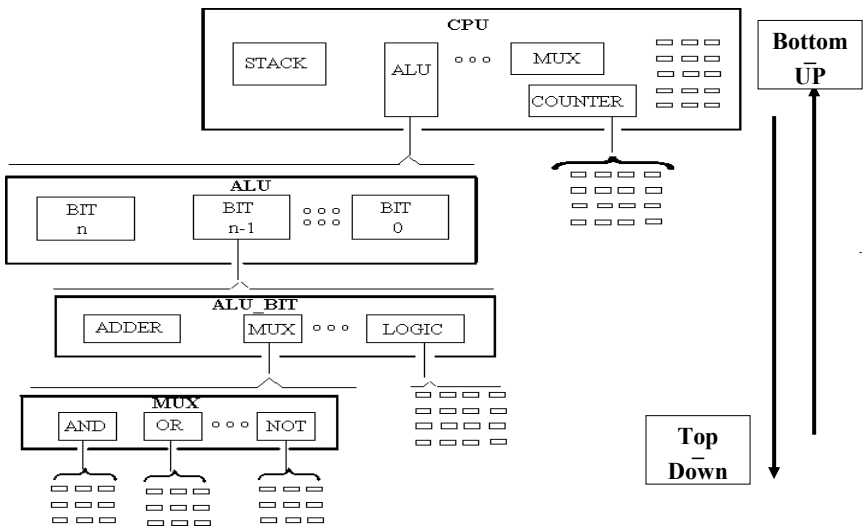

1-24

# How to Deal with Design Complexity?

- Moore's Law: Number of transistors that can be packed on a chip doubles every 18 months while the price stays the same.
- Hierarchy: structure of a design at different levels of description
- Abstraction: hiding the lower level details.

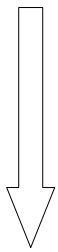


# Design Hierarchy



# Abstractions

- **An *Abstraction* is a simplified model of some Entity which *hides certain amount of the Internal details of this Entity***
- **Lower Level abstractions give more details of the modeled Entity.**
- **Several levels of abstractions (*details*) are commonly used:**
  - System Level
  - Chip Level
  - Register Level
  - Gate Level
  - Circuit (Transistor) Level
  - Layout (Geometric) Level



**More Details**

**(Less Abstract)**

1-27

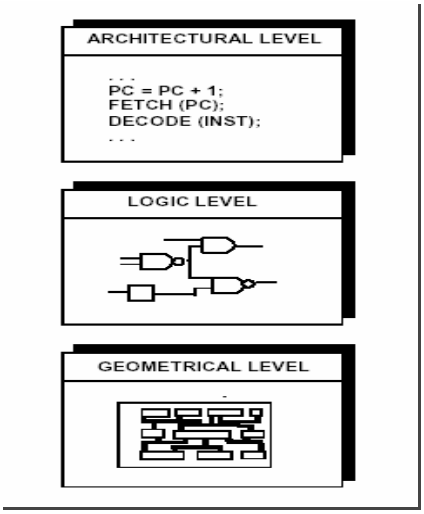
# Design Domains & Levels of Abstraction

- **Designs can be expressed / viewed in one of three possible domains**
  - Behavioral Domain (***Behavioral View***)
  - Structural/Component Domain (***Structural View***)
  - Physical Domain (***Physical View***)
- **A design modeled in a given domain can be represented at several levels of abstraction (*Details*)**

1-28

## Three Abstraction Levels of Circuit Representation

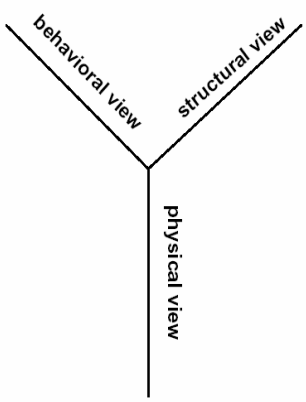
- Architectural level:
  - **Operations implemented by resources.**
- Logic level:
  - **Logic functions implemented by gates.**
- Geometrical level:
  - **devices are geometrical objects.**




1-29

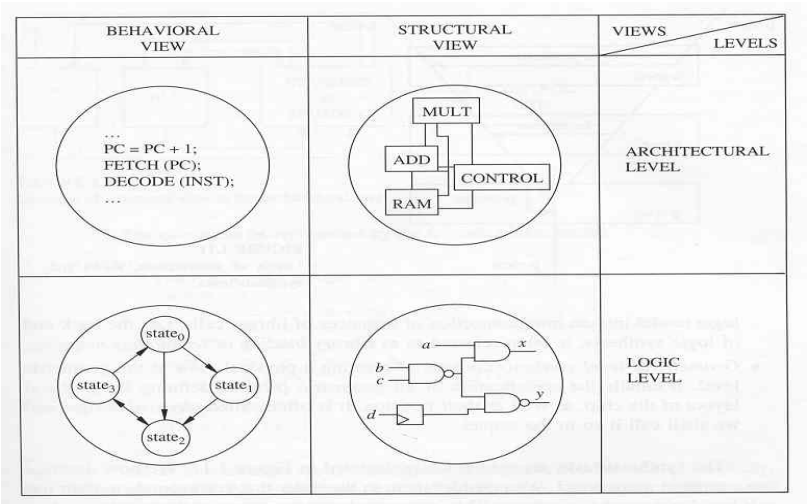
## Modeling Views

- Behavioral view:
  - **Abstract function.**
- Structural view:
  - **An interconnection of parts.**
- Physical view:
  - **Physical objects with size and positions.**



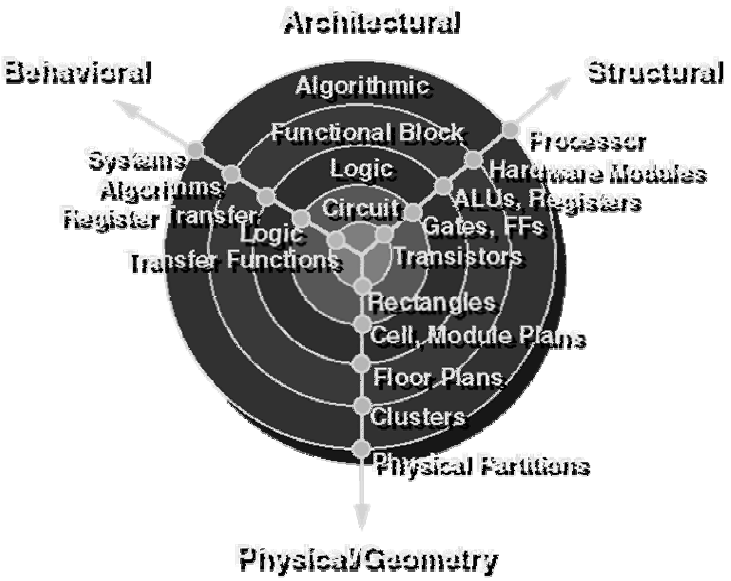

1-30

## Levels of Abstractions & Corresponding Views




1-31

## Gajski and Kuhn's Y Chart




1-32

# Design Domains & Levels of Abstraction

*Design Domain*

| *Abstraction Level* | *Behavioral* | *Structural* | *Physical* |
|---|---|---|---|
| **System** | **English Specs** | **Computer, Disk Units, Radar, etc.** | **Boards, MCMs, Cabinets, Physical Partitions** |
| **Chip** | **Algorithms, Flow Charts** | **Processors, RAMs, ROMs** | **Clusters, Chips, PCBs** |
| **Register** | **Data Flow, Reg. Transfer** | **Registers, ALUs, Counters, MUX, Buses** | **Std. Cells, Floor Plans** |
| **Gate** | **Boolean Equations** | **AND, OR, XOR, FFs, etc** | **Cells, Module Plans** |
| **Circuit (Tr)** | **Diff, and element Equations** | **Transistors, R, C, etc …** | **Mask Geometry (Layout)** |

1-33

# Digital System Design

- **Realization of a specification subject to the optimization of**
  - Area (Chip, PCB)
    - Lower manufacturing cost
    - Increase manufacturing yield
    - Reduce packaging cost
  - Performance
    - Propagation delay (combinational circuits)
    - Cycle time and latency (sequential circuits)
    - Throughput (pipelined circuits)
  - Power dissipation
  - Testability
    - Earlier detection of manufacturing defects lowers overall cost
  - Design time (time-to-market)
    - Cost reduction
    - Be competitive

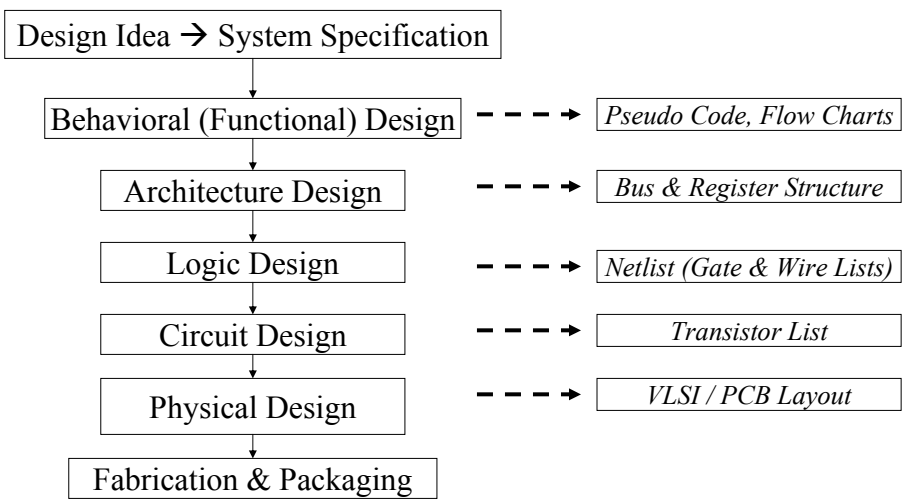
1-34

# Design vs. Synthesis

- **Design:**
  - A Sequence of synthesis steps down to a level of abstraction which is manufacturable
- **Synthesis:**
  - Process of transforming H/W from one level of abstraction to a ***lower*** one
- **Synthesis may occur at many different levels of abstraction**
  - Behavioral or High-level synthesis
  - Logic synthesis
  - Layout synthesis

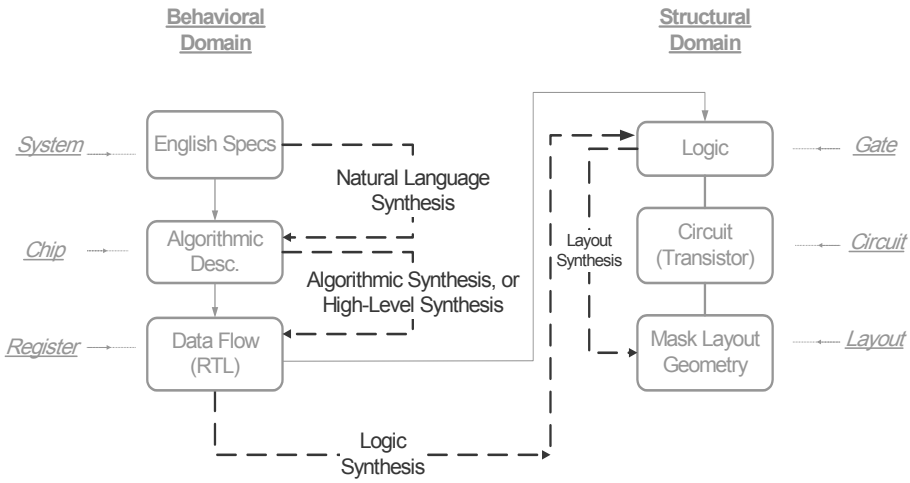
1-35

# Digital System Design Cycle




1-36

# Synthesis Process



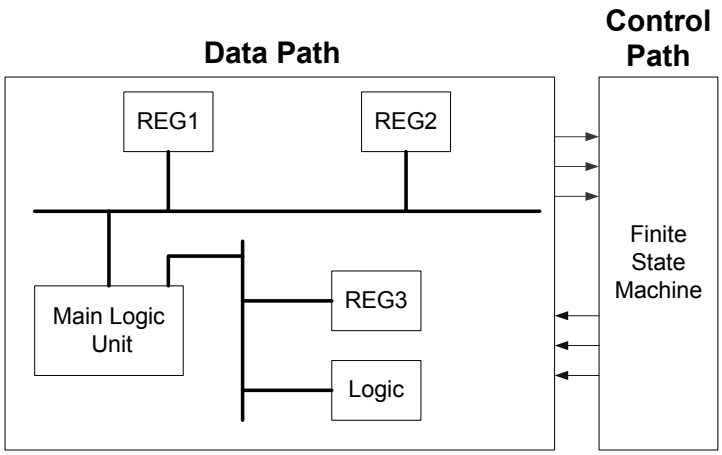

1-37

# Circuit Synthesis

- **Architectural-level synthesis:**
  - Determine the macroscopic structure:
    - Interconnection of major building blocks.
- **Logic-level synthesis:**
  - Determine the microscopic structure:
    - Interconnection of logic gates.
- **Geometrical-level synthesis:**
  - (Physical design): placement and routing
  - Determine positions and connections.

1-38

# Architecture Design



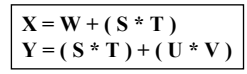

1-39

# Behavioral or High-Level Synthesis

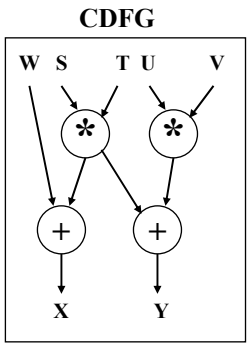
- **The automatic generation of data path and control unit is known as *high-level synthesis.***
- **Tasks involved in HLS are scheduling and allocation**
- **Scheduling distributes the execution of operations throughout time steps**
- **Allocation assigns hardware to operations and values.**
  - Allocation of hardware cells include functional unit allocation, register allocation and bus allocation.
  - Allocation determines the interconnections required.

1-40

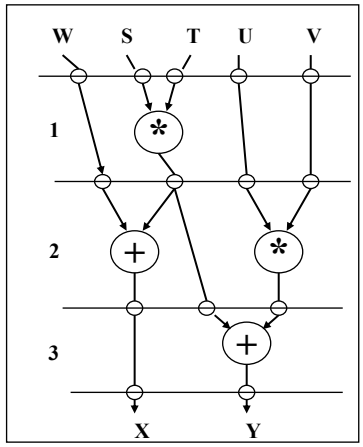
# Behavioral Description and its Control Data Flow Graph (CDFG)




(a)



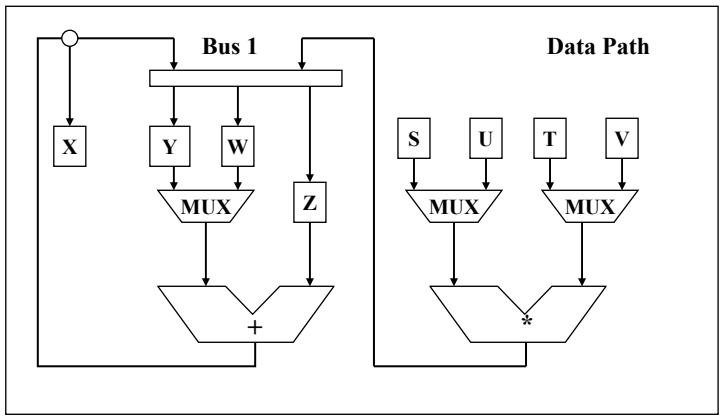

(b)




(c)

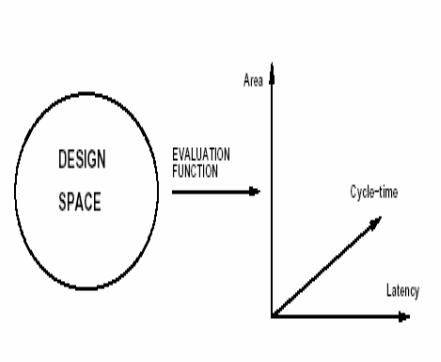


# Resulting Architecture Design



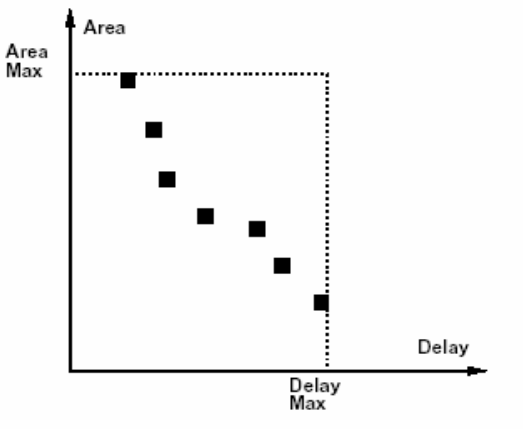
# Design Space and Evaluation Space

- **All feasible implementations of a circuit define its design space.**
- **Each design point has values for objective evaluation functions e.g. area**
- **The multidimensional space spanned by the different objectives is called design evaluation space**



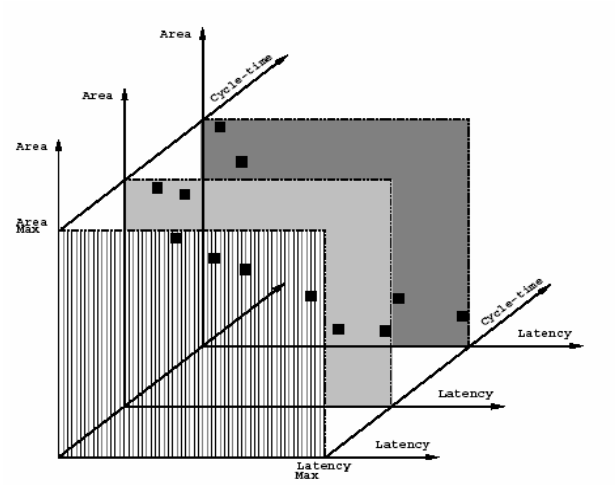

1-43

# Optimization Trade-Off in Combinational Circuits




1-44

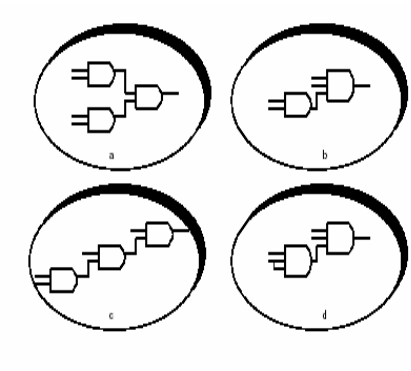
# Optimization Trade-Off in Sequential Circuits

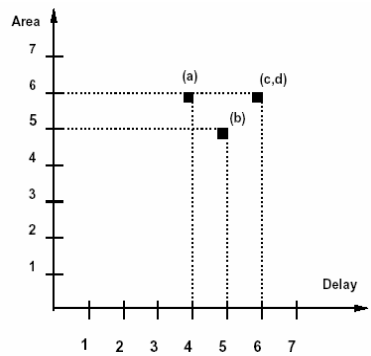



1-45

# Combinational Circuit Design Space Example

- Implement f = p q r s with 2-input or 3-input AND gates.
- Area and delay proportional to number of inputs.

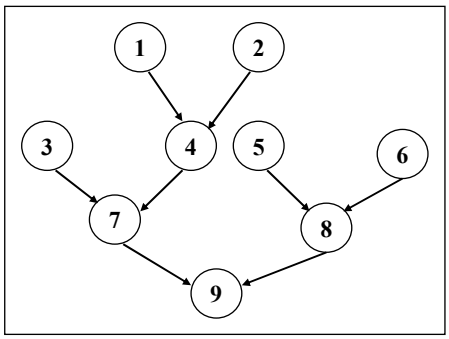



1-46

## Architectural Design Space Example …

- A CDFG and 3 Solutions



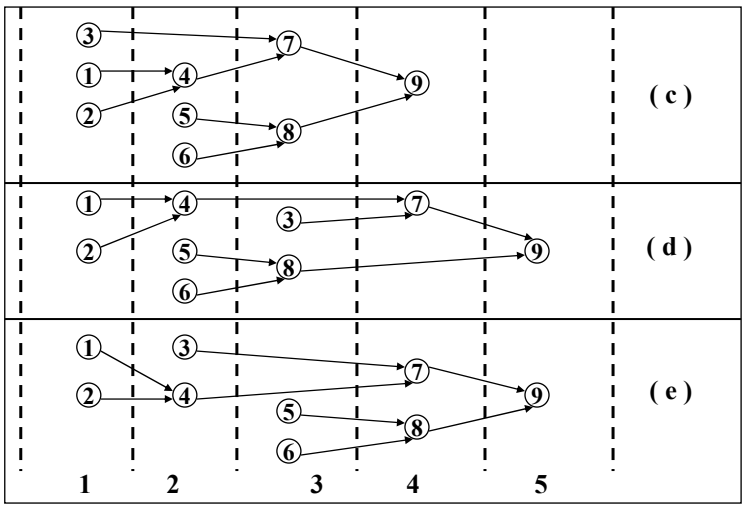

( a )

| | CSs | REGs | FUs |
|---|---|---|---|
| ( c ) | 4 | 4 | 3 |
| ( d ) | 5 | 3 | 3 |
| ( e ) | 5 | 4 | 2 |

( b )

1-47

## …Architectural Design Space Example




1-48

# Design Automation & CAD Tools

- **Design Entry (Description) Tools**
  - Schematic Capture
  - Hardware Description Language (HDL)
- **Simulation (Design Verification) Tools**
  - Simulators (Logic level, Transistor Level, High Level Language “HLL”)
- **Synthesis Tools**
- **Formal Verification Tools**
- **Test Vector Generation Tools**

1-49