Overview of This Course:
4 Main Components

■ Introduction to VLSI design

- ● Brief introduction to VLSI design and ASICs vs FPLDs

- ● Using Altera's Max-PLUS II for design entry and simulation

- ● Review of combinational and sequential circuits

■ IC technology

- ● Brief introduction to CMOS

- ● Comparison of various FPLD families

■ HDL-based design

- ● Design using AHDL and VHDL

- ● Large examples

■ Projects using schematic capture, AHDL and VHDL in Altera's MAX+PLUS II

Logic Synthesis Design Flow

■ Two alternative design entry methods:

  ● Manual design and schematic capture —
    draw and interconnect structural
    elements (gates, multliplexors, flip-flops,
    adders, ALUs, etc.)
    ■ Sequential or combinational design
    ■ CAD — manual design with automated
      bookkeeping & simulation / analysis

  ● HDL-based design — describe design in
    textual form using familiar programming
    constructs plus some additional ones
    ■ EDA — automated low-level decisions
      plus simulation / analysis

■ Compilation / Synthesis — compile
  modules into a flat netlist of gates,
  usually optimizing the design to
  minimize area, speed, power, etc.

■ Simulation and verification — make sure
  the design does what you think it does
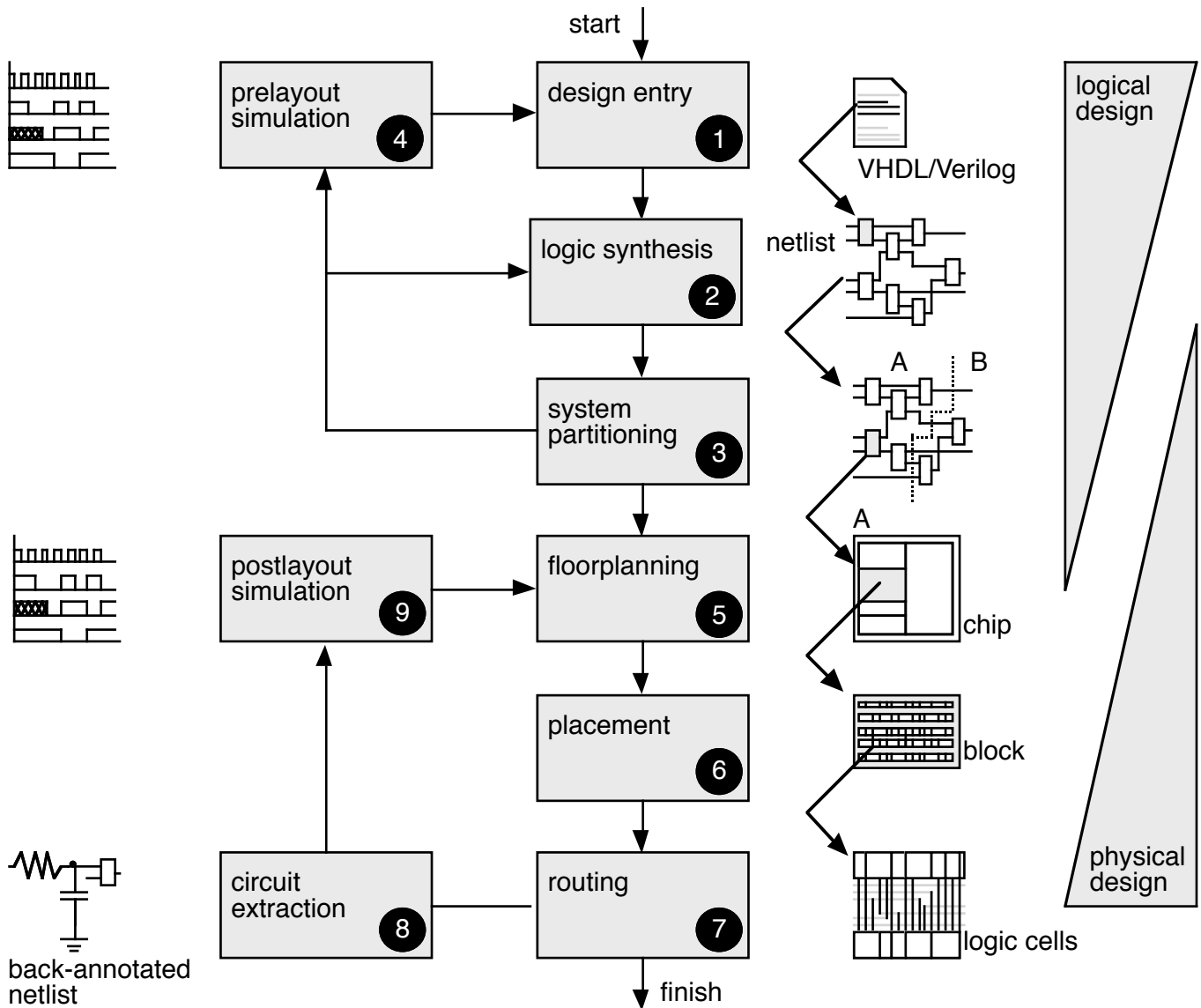
Logic Synthesis Design Flow
(cont.)

Figure from *Application-Specific Integrated Circuits,* Smith, Addison-Wesley, 1997

Logic Synthesis in a Larger Context

■ System synthesis — converts a task specification into processors, memories, ASICs, etc. plus software

- ● Hardware / software codesign

- ● Tradeoffs between hardware & software

■ Behavioral (high-level) synthesis — converts an algorithmic description of behavior into registers, adders, ALUs, busses, multiplexors, etc.

- ● Scheduling breaks design into states

- ● Data path synthesis produces interconnected set of functional units, registers, etc.

■ Logic synthesis — converts a structural description into gates and flip-flops

- ● Designer must specify all states