Schematic Entry

- Computer Aided Design (CAD) tools typically support both graphical schematic capture as well as textual design entry (e.g., AHDL, VHDL)
  - Documentation, design, simulation, verification

- A circuit *schematic* shows the interconnection of structural elements that make up a circuit
  - Captures only interconnection; behavior specified separately
  - The electronic (usually ASCII) version of that schematic is called a netlist

- Schematic capture
  - Direct entry of the circuit schematic
  - More “bookkeeping” than “automation”

![Figure from Altera technical literature](image)

Design Entry

- Enter cells from various predefined component libraries, or user-defined cells
- Connect cells using nets, buses, or name
- “Smart” selection tool — automatically becomes proper tool for task at hand

Cell Library

- **Components** (sometimes called *modules*) in a schematic are chosen from a library of cells
  - ASIC vendors provide a library of primitive gates for schematic entry
  - Users can define their own components and symbols

- Problem — no standard exists
  - Individual vendors might use different names to refer to a 2-input or gate
    - May be TTL 7400-series names:
      - 2-input NAND = 7400
      - 2-input AND = 7408
      - 2-input OR = 7432
    - May be more descriptive:
      - nand2, xor3, ...
  - Behavior may vary
    - Which input does 2-input multiplexor select when select input S = 0?
Hierarchical Design

- **Hierarchy** is used to reduce the size and complexity of the schematic
  - The alternative — drawing all symbols on one giant schematic with no hierarchy — is called a flat schematic
- Flat schematics are impractical to work with for even thousands of components
- Flat netlists, however, are occasionally used when the hierarchy isn’t relevant

![Diagram of hierarchical design](image)

**Nets**

- Cell instances have terminals, also known as pins, connectors, or signals, that are the inputs and outputs of the cell
- Cell instances are connected by wire segments, commonly called nets
  - A **local** (internal) net is internal to a cell
  - An **external** net connects to the inputs and/or outputs of the cell
- Nets may sometimes be collected together into buses for convenience
  - May be represented by a thicker line on the schematic, with some indication of number of nets involved
  - Individual nets can still be accessed when necessary

![Diagram of nets](image)
The “Chiptrip” Tutorial Example

- Simulates an auto driving around town
  - auto_max — AHDL state machine that keeps track of location of auto and acceleration at that point in time, gives ticket if you accelerate on small roads
  - speed_ch — waveform state machine that gives ticket if you accelerate for a second time
  - tick_cnt — counter that counts tickets
  - time_cnt — AHDL counter that keeps track of time taken to reach Altera

Graphic Editor

- Enter cells from various predefined component libraries, or user-defined cells
- Connect cells using nets, buses, or name
- “Smart” selection tool — automatically becomes proper tool for task at hand

Waveform Editor (for Design Entry)

- Can contain logical and state machine inputs; combinational, registered, and state machine outputs; and “buried” nodes to help define desired outputs
  - Can specify state names for state machines
  - Can compare desired and actual outputs

Compiler

- Checks for design entry errors, builds a single large flat database
- Logic synthesis to minimize resource usage (see Assign/Global Project Logic Synthesis), partitioner and fitter to match to available devices
Waveform Editor + Simulation

- Use waveform editor to specify simulation inputs
- Simulate, then view results in waveform editor (as shown above)
  - Simulate individual or grouped nodes (particularly good for state machines)

Floorplan Editor

- Device view shows pins, LAB view shows LABs, equations, I/O, and routing
- Can use to edit assignments
- After compilation, get information on most congested area of chip, number of expanders used within each LAB