ECE 551
System on Chip Design

Automated Layout Generation

Garrett S. Rose
Fall 2018
Overview

- Standard cells: a physical view of what they are how to use them
- ASIC/SOC Physical Design
  - Floorplanning
  - Place & Route
  - Closure
Overview

Standard cells: a physical view of what they are how to use them

- ASIC/SOC Physical Design
  - Floorplanning
  - Place & Route
  - Closure
Reasoning for Standard Cells

- Limit implementation effort by reusing limited library of cells
- Cells need to be designed and verified only once
  -- Reuse many times: amortization of design cost
- Trade-offs vs. full custom design
  - Reduced integration density
  - Lower performance
  - Can’t “fine-tune” a design at a low level
  - Options limited by quality of available library
Library Considerations

- Many options when determining library composition
- Design of a standard cell library can take long time
  -- Difference from full custom: amortization over a large number of designs due to reuse

- Which is better?
  - A small library with limited fan-in and fan-out
  - A large library with many versions of gates
    - Different fan-ins
    - Different sizing for different loads
Library Considerations

- Fan-out and wiring parasitics are unknown at the beginning of the design process.
- The design process can be simplified by ensuring each library element can drive a large fan-out:
  - Oversized cells that fit many conditions
  - Trade-offs:
    - Further degradation of performance
    - Further waste of area
Using the Library

- A design can be captured at schematic level where it is composed of only the cells in the library.
- Usually, a design is generated from a high level hardware description language (VHDL, Verilog, etc).
- The layout is then automatically generated.
- Synthesis tool must choose the right cells for given speed, power, and area requirements.
Placement of Cells

- Automation possible by restricting layout
- Cells placed in rows or tracks
- All cells in library are of same height
- Routing channel req’s reduced by more metal layers
Standard Cell Components

- Standard cells are building blocks for overall design
- Each cell in the standard cell library consists of more than just the layout
- Essential components of most standard cells:
  - Layout
  - Abstract view
  - Schematic
  - HDL representation (functional description)
  - Timing information
Standard Cell Layouts

- Major component of a standard cell is layout
- Layouts of all cells tend to be the same height or a multiple of that height
- These layout cells are used for final layout
- On right: Inverter and NAND layouts for a 0.18um library
Standard Cell Abstract

- When CAD tools route, only concern is metal
- An abstract view is a layout with only the metal layers
- The abstract view is used by tools for routing
- Used for LEF description
- Examples on left

Inverter

2-Input NAND
The Routing Grid

- Every standard cell must include conventions to determine dimension and placement of cell features.
- Some standard cell libraries use a conservative grid to give more flexibility in cell design.
- These grid rules must be inline with design rules.
Routing Grid Example

- Metal spacings are via to via
- Vias can be placed on two adj. grid points
- Routing made easier by wider grid points
- Illustration of grid rules to the right
- Any customized standard cell should follow these rules
Library Exchange Format (LEF)

- Placement tools must have a standard cell definition describing dimensions and port locations.
- Typically, this information is included in a Library Exchange Format (LEF) file.
- The LEF file also contains capacitance and resistance information for each metal layer.
- Cadence Abstract Generator can be used to generate such LEF files if needed.
- Any custom blocks to be used must be added to the LEF description.
Example LEF Description

- Some lines from a standard cell LEF file:

```
LAYER metal2
  TYPE ROUTING ;
  DIRECTION VERTICAL ;
  PITCH 0.8 ;
  WIDTH 0.3 ;
  SPACING 0.3 ;
  RESISTANCE RPERSQ 0.08 ;
  CAPACITANCE CPERSQDIST 1.9e-05 ;
  EDGECAPACITANCE 6.000000e-05 ;
END metal2
```

```
MACRO NAND2X1
  CLASS CORE ;
  FOREIGN NAND2X1 0.000 0.000 ;
  ORIGIN 0.000 0.000 ;
  SIZE 2.400 BY 10.000 ;
  SYMMETRY X Y ;
  SITE core ;
  PIN A
    DIRECTION INPUT ;
    PORT
      LAYER metal1 ;
        RECT 0.200 2.900 0.600 3.700 ;
    END
END A
```

...
The LIB file is also important (synthesis & routing):

```plaintext
... cell (INVX1) {
  cell_footprint : inv;
  area : 16;
  cell_leakage_power : 0.0221741;
  pin(A) {
    direction : input;
    capacitance : 0.00932456;
    rise_capacitance : 0.00932196;
    fall_capacitance : 0.00932456;
  }
  pin(Y) {
    direction : output;
    capacitance : 0;
    rise_capacitance : 0;
    fall_capacitance : 0;
    max_capacitance : 0.503808;
    function : "(!A)"
  }
... cell_fall(delay_template_5x5) {
  index_1 ("0.005, 0.0125, 0.025,
           0.075, 0.15");
  index_2 ("0.06, 0.18, 0.42, 0.6,
           1.2");
  values ( \
          "0.030906, 0.037434, 0.038584,
          0.039088, 0.030318",
          "0.04464, 0.057551, 0.073142,
          0.077841, 0.081003",
          "0.064368, 0.091076, 0.11557,
          0.126352, 0.144944",
          "0.139135, 0.174422, 0.232659,
          0.261317, 0.321043",
          "0.249412, 0.28434, 0.357694,
          0.406534, 0.51187");
...}
...
Abstract Generator

- Learn more about how to use Abstract Generator at: http://avatar.ecen.okstate.edu/projects/scells/flow/abstract/index.html

- Another useful tool is Cadence Encounter Library Characterizer
Overview

- Standard cells: a physical view of what they are how to use them
- ASIC/SOC Physical Design
  - Floorplanning
  - Place & Route
  - Closure
Place & Route Design Flow

Netlist

Library Description

Placement

DEF

Technology Constraints

Route

DEF

Technology Parameters

Parasitic Extract

ESPF

Timing Analysis

Library SDF

Timing OK

Noise and Reliability

OK

Tool error

Database to Manufacturer
Steps Toward Layout

- Placement
- Floorplanning
- Routing
- Parasitic Extraction
- Timing Analysis
- Noise Analysis
- Timing-driven Placement
- Clock-tree Routing
- Power Analysis
Data Exchange Formats

- Library Exchange Format (LEF) – contains timing information of standard cells and metal layers
- Design Exchange Format (DEF) – format (usually an ASCII file) passed between design flow stages
- Standard Delay Format (SDF) – IEEE standard for representing and interpreting timing data
- Parasitic extraction formats – used to represent parasitic information (R & C) for timing analysis
  - Extended Standard Parasitic Format (ESPF)
  - Reduced Standard Parasitic Format (RSPF)
  - Standard Parasitic Exchange Format (SPEF)
Placement

- Key is use of constant height, variable-width standard cells arrayed into rows across the chip
- Can also add application-specific custom blocks
- Typically, there is no separation between std. cell rows as routing occurs over the cells using multiple metal layers
- Placement makes use of LEF input
- Goal is usually to minimize the length of wires
Results of Placement

Bad Placement

Good Placement
Results of Placement

- Bad placement causes routing congestion:
  - Increases circuit area and cost
  - Longer wires leading to more capacitance
  - Longer delay and higher dynamic power

- Good placement:
  - Circuit area and wiring decreased
  - Shorter wires with less capacitance
  - Shorter delay and less dynamic power
Imagine the Following

- You are planning the transportation (i.e., roads & highways) for a new city the size of Brooklyn
- Many dwellings need direct roads that cannot be used by anyone else
- You can affect the layout of houses and neighborhoods but architects will complain
- Also, the time along any path can’t be longer than some fixed amount

- What are the considerations?
Some Considerations

- How many levels do my roads need to go?
  -- Remember: Higher is more expensive
- How do I avoid congestion?
- What basic structure do I want for my roads?
  - Manhattan?
  - Chicago?
  - Boston?

- Automated route tools have to solve problems of a comparable complexity on every leading edge chip
Routing Applications

Cell-based

Mixed Cell and Block

Block-based
Routing Algorithms

- Hard to tackle high-level issues like congestion and wire-planning and low level details simultaneously

- Global Routing
  - Identify routing resources to be used
  - Identify layers (and tracks) to be used
  - Assign particular nets to these resources

- Detail (Local) Routing
  - Define pin-to-pin connections
  - Must understand most or all design rules
  - May use a compactor to optimize results
Routing Rules – Part I

- Wiring/routing performed in layers (typically 5-9) only in “Manhattan” N/S E/W directions
  -- Example: Layer 1 N/S, Layer 2 E/W
- A segment cannot cross another on the same layer
- Wire segments can cross wires on other layers
- Power and ground may have their own layers
Routing Rules – Part II

- Routing can be on a fixed grid
- Case 1: Detailed routing only in channels
  - Wiring only over a row of cells where there is a free track – inserted with a “feedthrough”
  - Cells must bring signals out to channel through “ports” or “pins”
Routing Rules – Part III

- Routing can be fixed or gridless (aka, area routing)
- Case 2: Detailed routing over cells
  - Wiring can go over cells
  - Cell design must try to minimize obstacles
  - Cells do not need signals out to a channel
    -- the route will come to them
Timing Analysis

- Static Timing Analysis (STA) – method of computing expected timing of digital circuit without simulation
- Referred to as static because it does not depend on input vectors
- Each cell or module in design is accompanied by timing information (e.g., rise and fall times, delays)
- Timing analysis (usually via STA) must occur between multiple points in the design flow and is used to optimize synthesis, placement, & routing
Timing Directed Placement Design Flow

1. Library SDF
2. Technology Constraints DEF
3. Technology Attributes Wire Capacitance and Resistance
4. Library LEF

- Placement
  - Routing Engine
    - Timing Directed Placement Engine
      - Extract
        - Timing Analysis
          - Timing OK
            - Final Checks
              - OK
                - Finished
Clock-Tree Routing

- Central to modern high-speed designs is the clock distribution strategy.
- To minimize skew, it is often best to route the clock and its buffers before the main logic place & route.
- This is performed using a clock tree router.
H-Trees

- An H-tree is a fractal structure drawn with an H shape and further recursively drawn H shapes
- Goal is to distribute the clock to every endpoint on the chip with same wire length to center
Clock Spines

- As with a grid, clock buffers located in rows
- Spines drive length-matched serpentine wires to each small group of clocked elements
Summary

- Discussed the necessary elements of a standard cell
- Standard cell layouts must follow strict rules
- Timing, I/O, and dimension information must be included in LEF and LIB files
- Floorplanning, place and route are key to physical design
- Timing closure important final verification before tape-out