UCLA

FPGA Placement and Routing

Santiago Mok
smok@ucla.edu

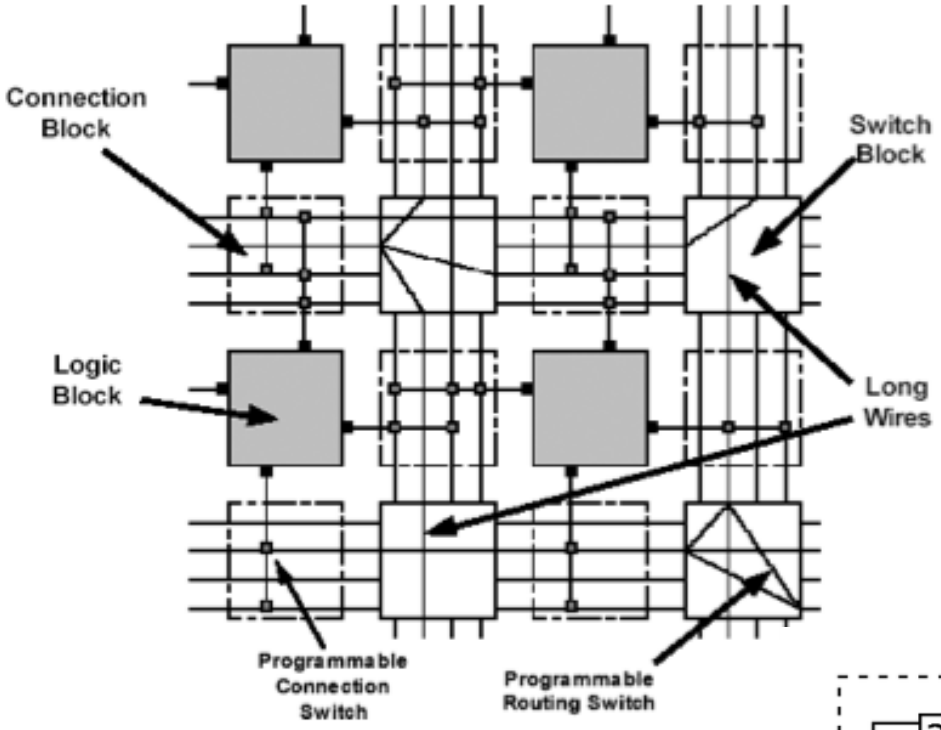
Outline

- Overview of FPGA
- FPGA Architecture
- FPGA CAD Flow
- Clustering and Placement
  - Simulated Annealing-based method
- Physical Synthesis Optimization
- Routing

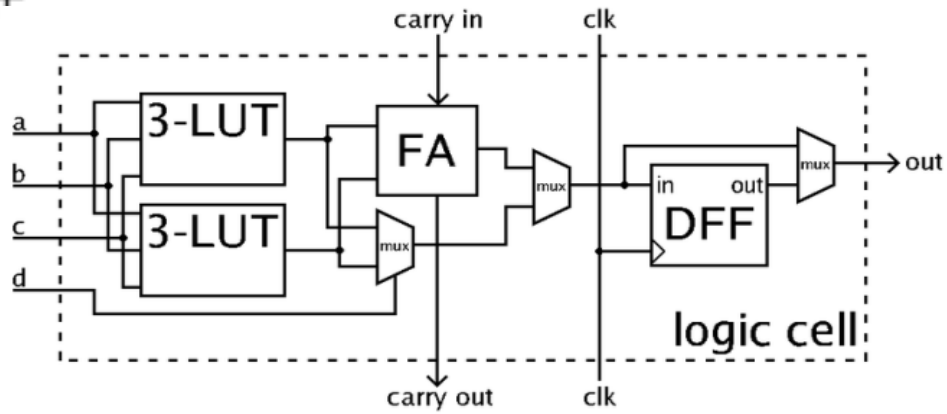
FPGA Overview

- Field-Programmable Gate Arrays
  - Pre-fabricated silicon devices that comprise of an array of uncommitted circuit elements (logic blocks) and interconnect resources
  - An IC designed to be configured by end-user after manufacturing
- Implement any logical function that ASIC can perform
- Applications:
  - DSP
  - Device controllers
  - Medical imaging

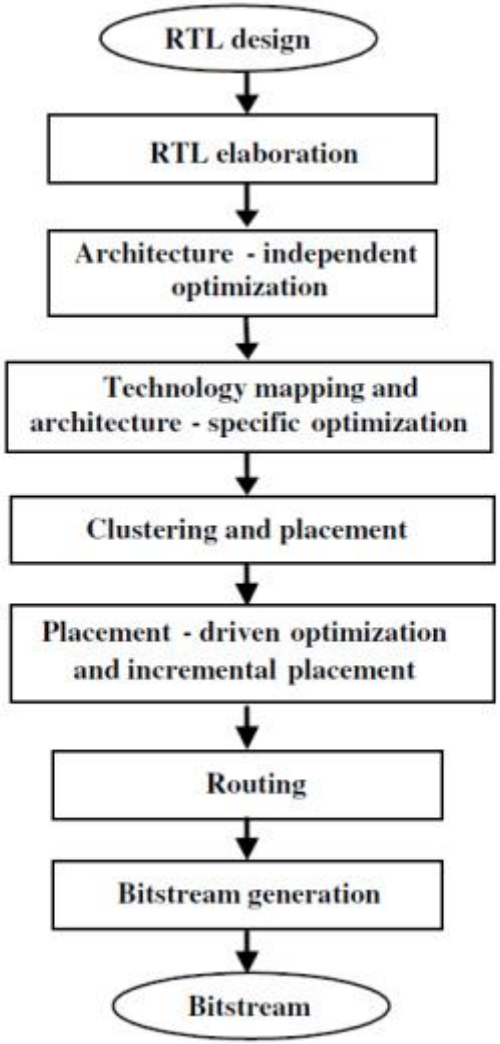
FPGA Architecture

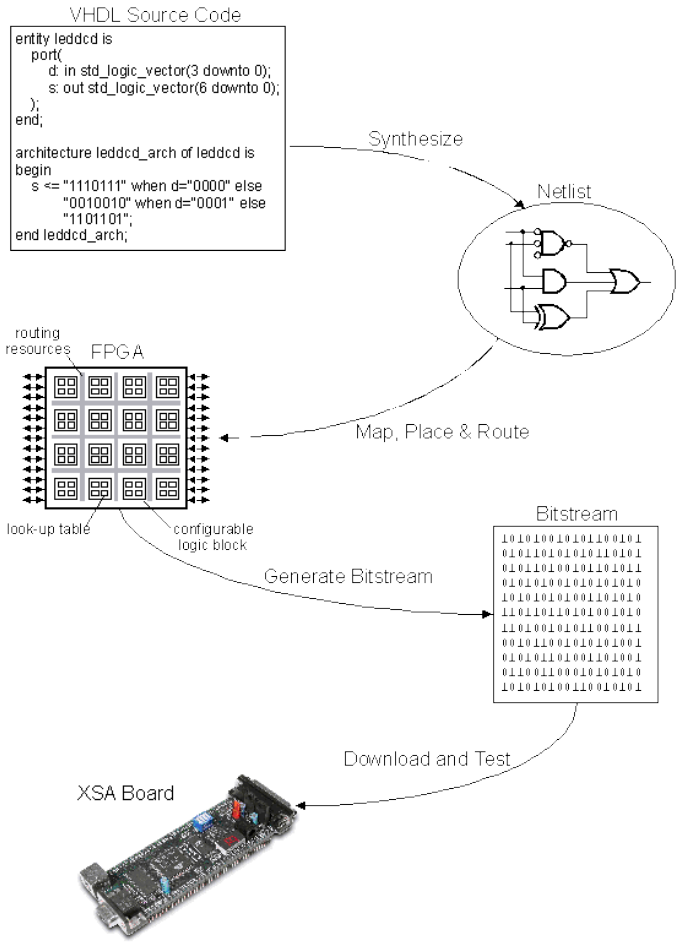


- I/O blocks and core programmable fabric
- Switch block
- Connection block
- Routing channels



FPGA CAD Flow





Clustering

- Group logic elements into logic blocks
- Separate clustering step may be performed prior to placement
- Reduce the number of logic blocks to be placed
- Simplify legality checking for main placement
- Algorithm: greedily packs LE with the highest attraction to the current cluster

Placement

- Slot assignment problem
- Placement has significant impact on the performance and routability of circuit design

- Existing approaches to FPGA placement:
  1. Simulated Annealing-based placement
  2. Partitioning-based placement
  3. Analytical method-based placement

Simulated Annealing-based Placement

- Placement optimization engine for placement used in the well-known VPR package for FPGA

```
P = InitialPlacement ();
T = InitialTemperature ();

while (ExitCriterion () == False) {
    while (InnerLoopCriterion () == False) {  /* "Inner Loop" */
        P_new = PerturbPlacementViaMove (P);
        ΔCost = Cost (P_new) – Cost (P);
        r = random (0,1);
        if (r < e^(-ΔCost/T)) {
            P = P_new;      /* Move Accepted */
        }
    }  /* End "Inner Loop" */
    T = UpdateTemp (T);
}
```

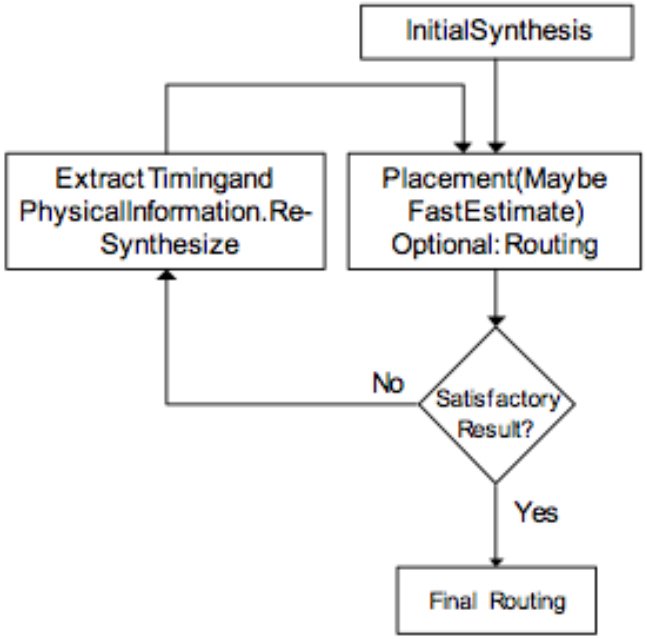
Simulated Annealing-based Placement

- The cost function penalizes placement which require more routing in the narrower channels

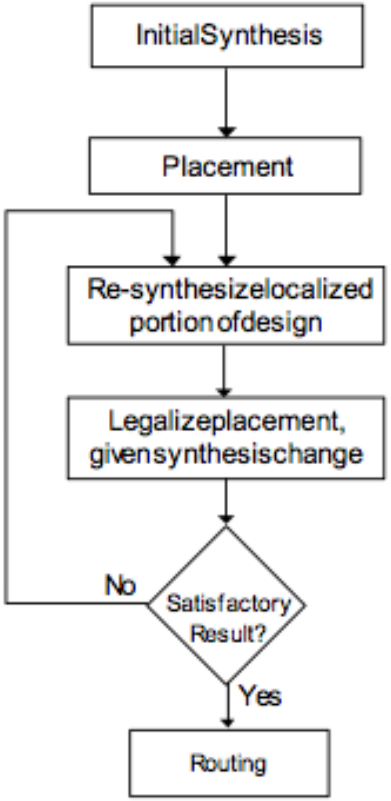
$$Cost = \sum_{n=1}^{N_{nets}} q(n)\left[\frac{bb_x(n)}{C_{av,x}(n)} + \frac{bb_y(n)}{C_{av,y}(n)}\right]$$

- Key strengths that SA possess:
  - Possible to enforce all the legality constraints imposed by the FPGA architecture
  - Possible to model the impact of the FPGA routing architecture on circuit delay and routing congestion

Physical Synthesis Optimizations



(a) Example "iterative" physical synthesis flow.



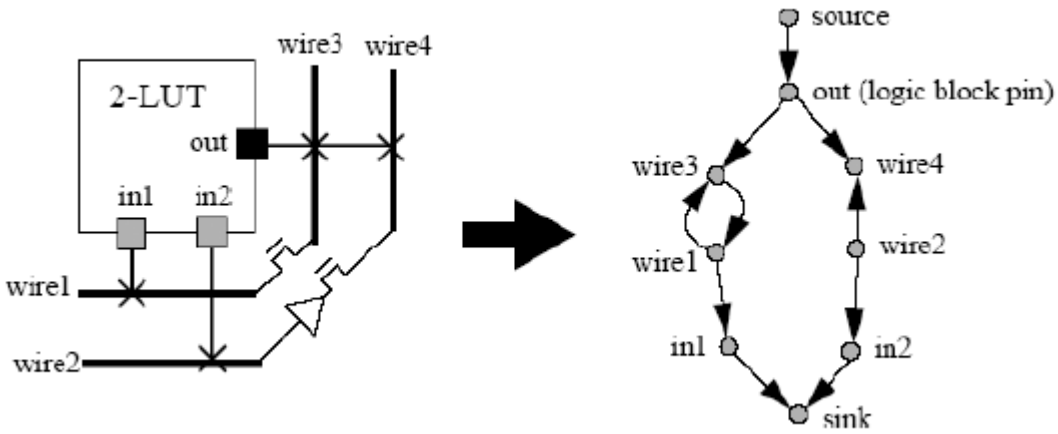
(b) Example "incremental" physical synthesis flow.

Routing

- FPGA routing consists of pre-fabricated metal wires and programmable switches
- Interconnect between wire and CLB I/O blocks
- FPGA routing typically goes through:
  - Routing-resource graph generation
  - Global routing
  - Detailed routing
- Two-Step Routing
- Single-Step Routers

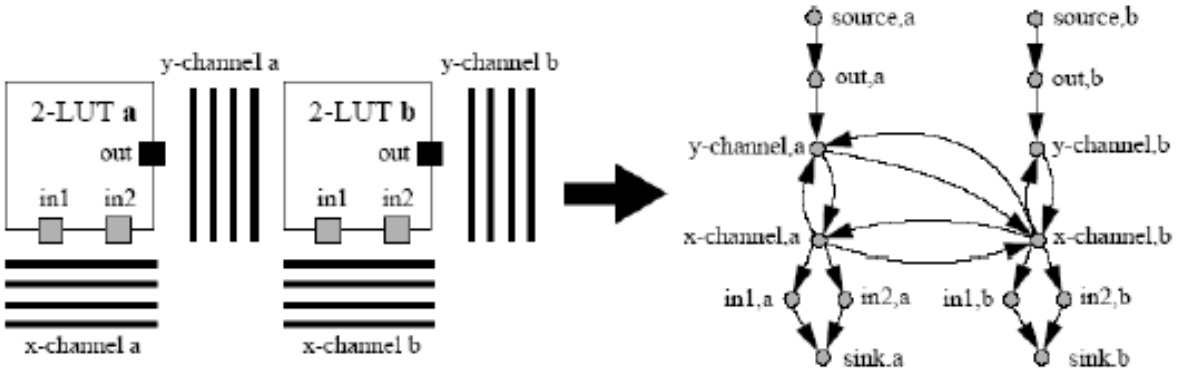
Routing-resource graph

- An abstract data representation for global and detailed routers
- Vertices: I/O pins of logic blocks and wire segment in the routing channels
- Edges: programmable switches that connect two vertices



Global Routing

- Uses a simplified routing resource graph
- To determine the routing of each net on the graph such that all the channel capacity constraints are met
    1. Each connection is initially routed using minimum cost with little regards to congestion
    2. Routing iterations to reduce wire overuse

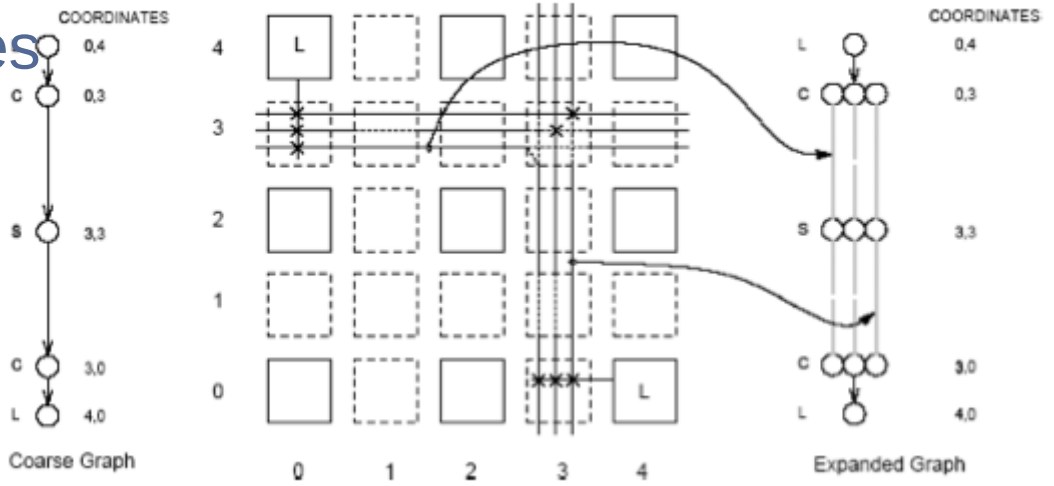


$$CongestionCost(n) = \left[b(n) + h(n)\right] \cdot p(n)$$

Detailed Routing

- Given a global routing solution, the detailed router step implements each step in the coarse routing-resource graph to eliminate resource conflict
- Two phases:

1. Enumerate all the possible detailed routes and add to expansion graph
2. Iteratively refine route with lowest cost
   - Essential routes



Single-step router

- Avoid possible mismatch between global and detailed routing
- These routers differ primarily in their costing of various routing alternatives, search techniques, and congestion resolution
- Various single-step routers yield better result than two-step routing

Global R.	LocusRoute [17]		GBP [20]	OGC [21]	IKMB [22]	VPR	TRACER [24]	VPR
Detail R.	CGE [18]	SEGA [19]				SEGA [23]		
9symml	9	9	9	9	8	7	6	**6**
alu2	12	10	11	9	9	8	9	**8**
alu4	15	13	14	12	11	10	11	**9**
apex7	13	13	11	10	10	10	8	**8**
example2	18	17	13	12	11	10	10	**9**
k2	19	16	17	16	15	14	14	**12**
term1	10	9	10	9	8	8	7	**7**
too_large	13	11	12	11	10	10	9	**8**
vda	14	14	13	11	12	12	11	**10**
Total	123	112	110	99	94	89	85	**77**

Future Challenges

- Need for more scalable and efficient placement and routing algorithm
- Novel PD algorithms with considerations of process variability and be able to perform statistical optimization

References

- D. Chen, J. Cong and P. Pan, "FPGA Design Automation: A Survey," *Foundations and Trends in Electronic Design Automation,* vol. 1, no. 3, pp. 195-330, Nov 2006.
- V. Betz and J. Rose. VPR: a new packing, placement and routing tool for
- FPGA research. In *International Workshop on Field-Programmable Logic and*
- *Applications, pages 213–222, 1997.*
- M. Hutton and V. Betz, "FPGA Synthesis and Physical Design", Chapter 13 in CRC Press "Electronic Design Automation for Integrated Circuits", 2006.
- J. Cong, T. Kong, J. Shinnerl, M. Xie, and X. Yuan, "Large Scale Circuit Placement," *ACM Transaction on Design Automation of Electronic Systems,* vol. 10, no. 2, pp. 389-430, April 2005.
- http://www.eecg.toronto.edu/~jayar/pubs/brown/survey.pdf
- http://www.xess.com/appnotes/fpga_tut.php