Chip Physical Design EDA Tool on PYNQ

Team NO. 11
Team member：Shuai Ye, Wei Wu, Qiwei Sun
Tutor：Zhixiong Di
Affiliation: Southwest Jiaotong University, Chengdu 611756, Sichuan Province China

Overview:

- A congestion-aware and area&timing oriented placement method which is called LC-KO is proposed. Experimental results show the feasibility and effectiveness in minimizing congestion with less placement area and better STA results.
- The proposed method has been successfully used in tape out procedure of several chips, which is based on 55, 40, to 28nm standard cell libraries.
- Build an embedded server on ARM(PYNQ) to transmission the data between the Linux workstation and FPGA.
- Completed the 4×4 NoC and routing algorithms hardware IP. Based on this hardware architecture, we can enhance openness, scalability and flexibility of this project.

Conclusion:

- FPGA acceleration can reduce the computation time of the routing algorithms from several hours to minutes. The conclusion has important significance for the SoC chip design.
- The routing algorithm has no precision dependence, so, it is very suitable for using FPGA hardware acceleration.
- The main challenge of the EDA algorithms is enormous data and to scan it. “FPGA on Cloud” can exerts resources and performance advantages to solve this problem.

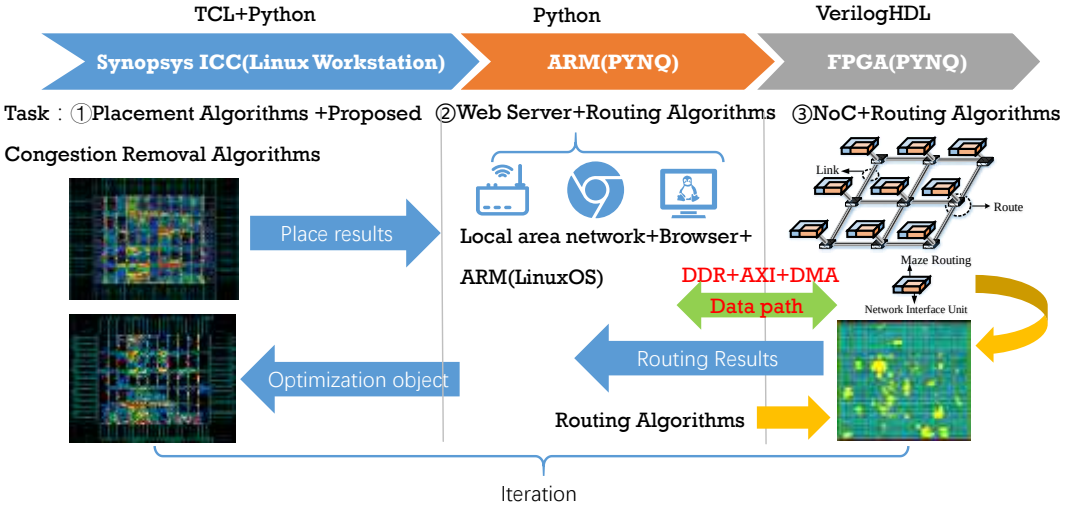


Figure 1-Overall Architecture

Table 1-Congestion Removal Results

Process	TSMC-28nm		UMC-40nm		UMC-5nm	
	Instance	Layer	Instance	Layer	Instance	Layer
	1.5m	8lm	1m	6lm	5k	5lm
Metric	-congestion	LC-KO	-congestion	LC-KO	-congestion	LC-KO
Shorts	2450	567	1327	231	2943	181
WNS	-0.3	-0.3	-0.27	-0.27	-0.34	-0.34
TNS	-1.2ns	-1.3ns	-3.8ns	-3.8	-5.4ns	-5.4ns
Std.cell area	2170920	2170920	3101324	3101345	1137	1137
Utilization	85%	85%	79%	79%	59%	59%
Total wire length	1996977	1994238	1239943	110365	702785	694755
Run time	13hours	8hours	8hours	4hours	1hours	45mins

Table 2-Overflow Results

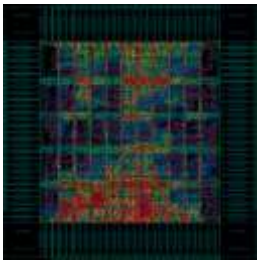
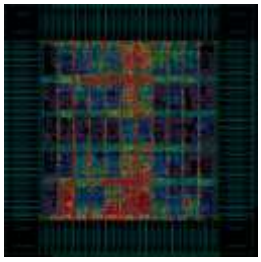
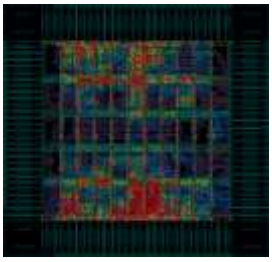
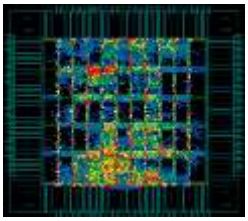
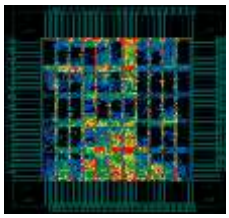
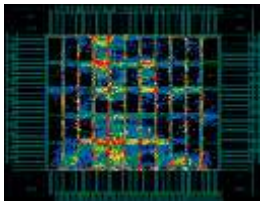
Initial placement	Synopsys ICC	Proposed
		
		

Figure 2-PYNQ Hardware resource and performance