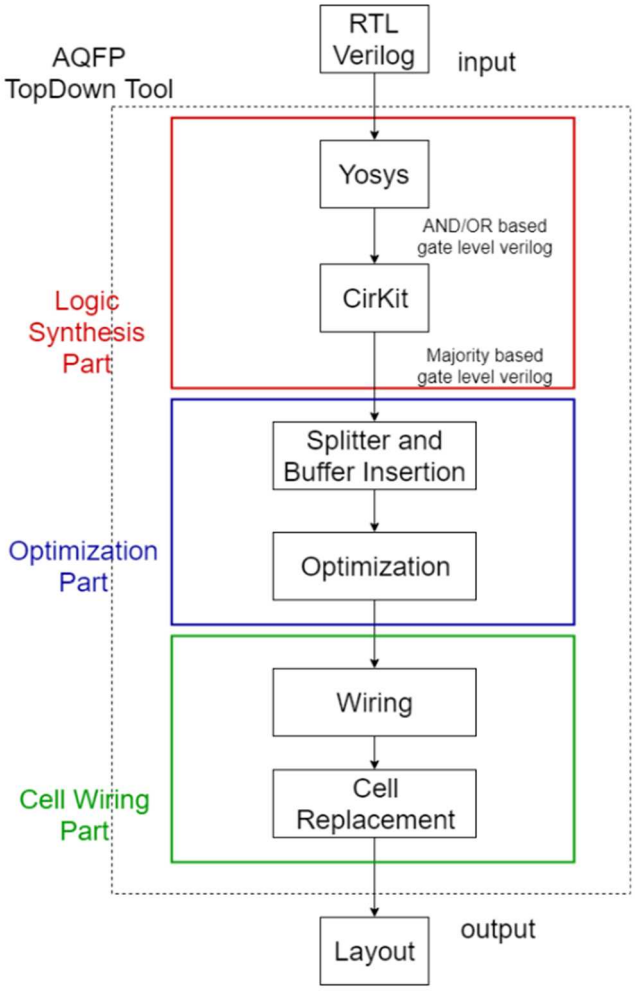
EDP2-5

Development of Majority-Logic-Based Top-Down Environment for Adiabatic Quantum-Flux-Parametron Circuits

*Ro Saito[1], Christopher L. Ayala[2], Olivia Chen[2], Tomoyuki Tanaka[1], Nobuyuki Yoshikawa[1]

Electrical and Computer Engineering, Yokohama National University[1]
Institute of Advanced Sciences, Yokohama National University[2]

1 Abstract Adiabatic quantum-flux-parametron (AQFP) logic is one kind of superconducting logic family spotlighted as a technological foundation for developing extremely low-energy computers. However, AQFP circuits have a disadvantage in design time when compared with modern CMOS circuits because AQFP logic lacks adequate EDA tools. This is one of the major problems that must be solved to push AQFP logic into practical use. In this thesis, we show our top-down development flow, how to take advantage of a CMOS logic synthesis tool and convert it into an AQFP circuit. 2 Outline of AQFP Top-Down Design Tool There are roughly three parts of the AQFP top-down design tool: (1) logic synthesis; (2) logic optimization; (3) cell wiring. The logic synthesis part of the flow uses open source tools, whereas the other parts use tools that we have developed ourselves. The top-down design tool is a program that integrates all three of these parts. First, an RTL (register transfer level) Verilog file which describes the circuit's behavior is synthesized by a logic synthesis tool such as Yosys. The logic synthesize tool converts it to an AND/OR-based gate-level description. Next, it is converted to a majority-based netlist using CirKit. Second, the optimization part uses some tools we developed before. The synthesized circuit can not be used as it is, because the AQFP circuit is a phase-driven circuit. Gates such as“ splitter ”and“ buffer ”are needed to be inserted to line up and synchronize phases. We already developed a program to insert the necessary number of buffers and splitters while also considering retiming optimization to reduce the necessary number of phase-aligning buffers. This completed netlist is given to the physical optimization part which is implemented using the genetic algorithm. However, all signal/bias lines are just virtual connections at this stage. Finally, these connections are automatically replaced by each physical cells that are suitable for the place, size, and direction during the cell wiring part of the top-down flow. The AQFP top-down environment can be built by integrating these steps. It enables us to export a chip layout from an RTL Verilog description.



Keywords: AQFP, topdown, logic synthesis, retiming