論 文 の 内 容 の 要 旨

論文題目　　　　　論理デバッグ・ECO のための自動修正技術に関する研究
　　　　　　　　　Automatic Rectification Methods for Logic Debugging and ECO

氏　　名　　　木村　悠介

The design of digital circuits may go through different abstraction levels like high-level (e.g. C-based) design, RTL design, gate-level design and so on. Sometimes, a logical bug-fix or engineering change order (ECO) may happen in the later phases of the design, and those changes need to be reflected in all the design descriptions of the target design. This research mainly deals with two situations: (1) C or RTL design is modified, and ECO should be done at gate-level in an automatical manner, (2) gate-level circuits are modified directly, and high-level C descriptions remains unchanged.

In the first situation, the process to modify gate-level circuits generally consists of three steps: (1) Finding the target gates to be modified, (2) Selecting internal signals for correcting the gates, (3) Generating new logic for fan-ins of the target gates (patch). My proposed methods focus on step (2). This research direction is common recently and essentially the same as ICCAD'17 contest. I propose a new formulation to solve multiple targets at the same time. It gets 10-100 times better results in terms of the contest criteria compared to the existing research.

Moreover, the proposed method can be applied to sequential circuits, by enumerating all the reachable states. The method can find correct solutions within practical time for all ITC99 benchmark circuits.

In the second situation, C-based design is assumed. The C description should be modified to become equivalent to the modified gate-level circuit. In this research, it is assumed that designers know which line(s) to be modified in the C description and have rough idea about how to modify them. Each target line is converted to a vacant place, and my proposed method synthesizes program expressions by utilizing the existing program synthesis techniques. It can rectify the C description of AES256 encryption successfully, and such a large program is not handled in the existing researches.