

CREST DVLSI領域会議
「アーキテクチャと形式的検証の協調による
超ディペンダブルVLSI」
外部討論資料

株式会社富士通研究所
吉川隆英

1. 問題箇所絞り込み

A) 再現性の確認

B) ソフトウェア命令列入れ替え・ハードウェア設定変更による再現性変化の確認

C) 問題箇所の特定

エラーを見つけた箇所ではなく、誤動作の根源を捕まえる

2. 修正方針の決定

A) 修正方法の検討(論理)

考慮漏れが無いのか？ 修正により他の論理に影響がないか？

B) 修正方法の検討(実装)

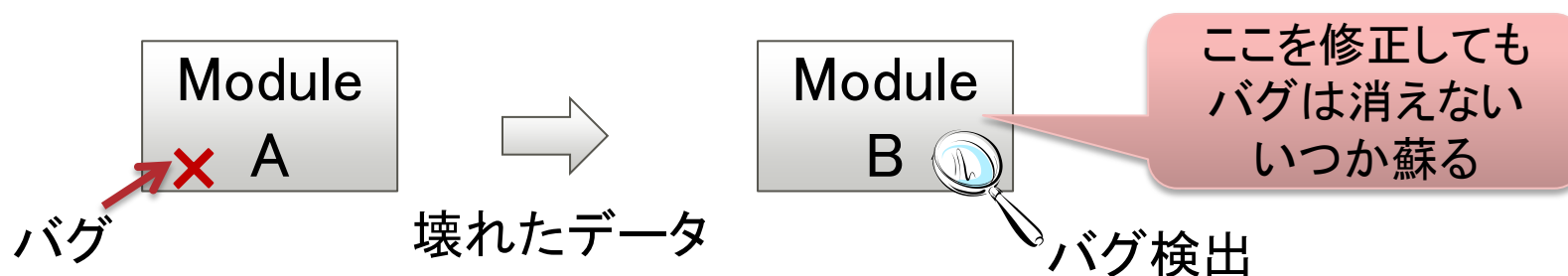
面積、タイミング、電力などの制約

■ 修正方法の検討(論理)の自動化

- 形式的に行えるようになると
 - (探索対象範囲の中では)最適な修正論理提示
 - 仕様との等価確認により、劣化障害の危険性減少が実現されるため、より正確な修正が出来る可能性がある
- さらに本研究では
 - LUTへの入力変数が追加可能という性質を取り込んだため、いくつかの実製品に対してLUTによる自動修正が適用可能となった

■高精度なバグ位置特定ツールとの組み合わせが必要

- 入力パタンが適切に振られなければバグ位置を推定できない
- エラーを見つけた場所がバグ位置とは限らない




- 適切な位置を修正出来なければ、対症療法の積み上げとなる

■全ての入出力に対する正解を提示しきれない

- 64bit の入力全部が例えば問い合わせられると対応しきれない

■まだ自動修正可能な論理は一部。それはどれくらいか？ 残りはどう対応していくか？



FUJITSU

shaping tomorrow with you