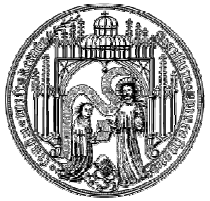
Utilizing Parallelism of TMR to Enhance Power Efficiency of Reliable ASIC Designs

Hagen Sämrow, Claas Cornelius, Jakob Salzmann,
Andreas Tockhorn, Dirk Timmermann

30/11/2010, Cairo

Institute of Applied Microelectronics
and Computer Engineering



University of Rostock

Outline

- Motivation
- Power-efficient TMR approach
  - Design idea
  - Design structure
  - Results
- Conclusion / Outlook

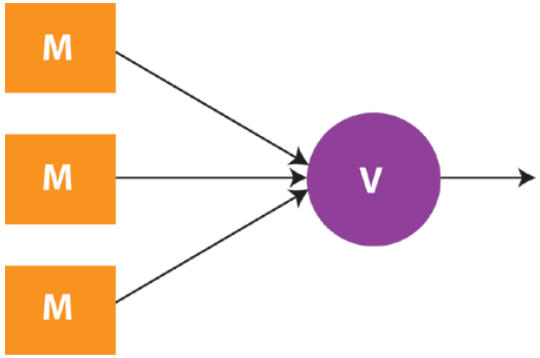
Motivation

- Power reduction for mobile applications

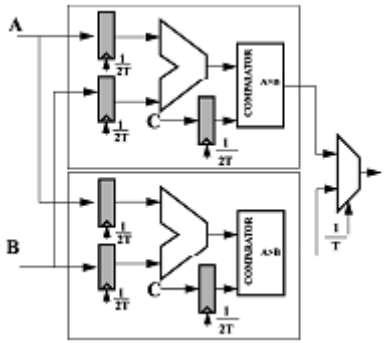
- Reliability suffers with every new technology and becomes more and more a concern of VLSI designers

- Redundancy – traditional approach to enhance reliability
  - At the cost of area and power consumption

Design Idea



- TMR – Triple modular redundancy
  - Three identical units
  - Same input for every module
  - Results are merged by a voting unit
  - Threefold power consumption and area
- Parallelized data paths
  - Reduced operating frequency and supply voltage
  - Same throughput
  - Larger design area

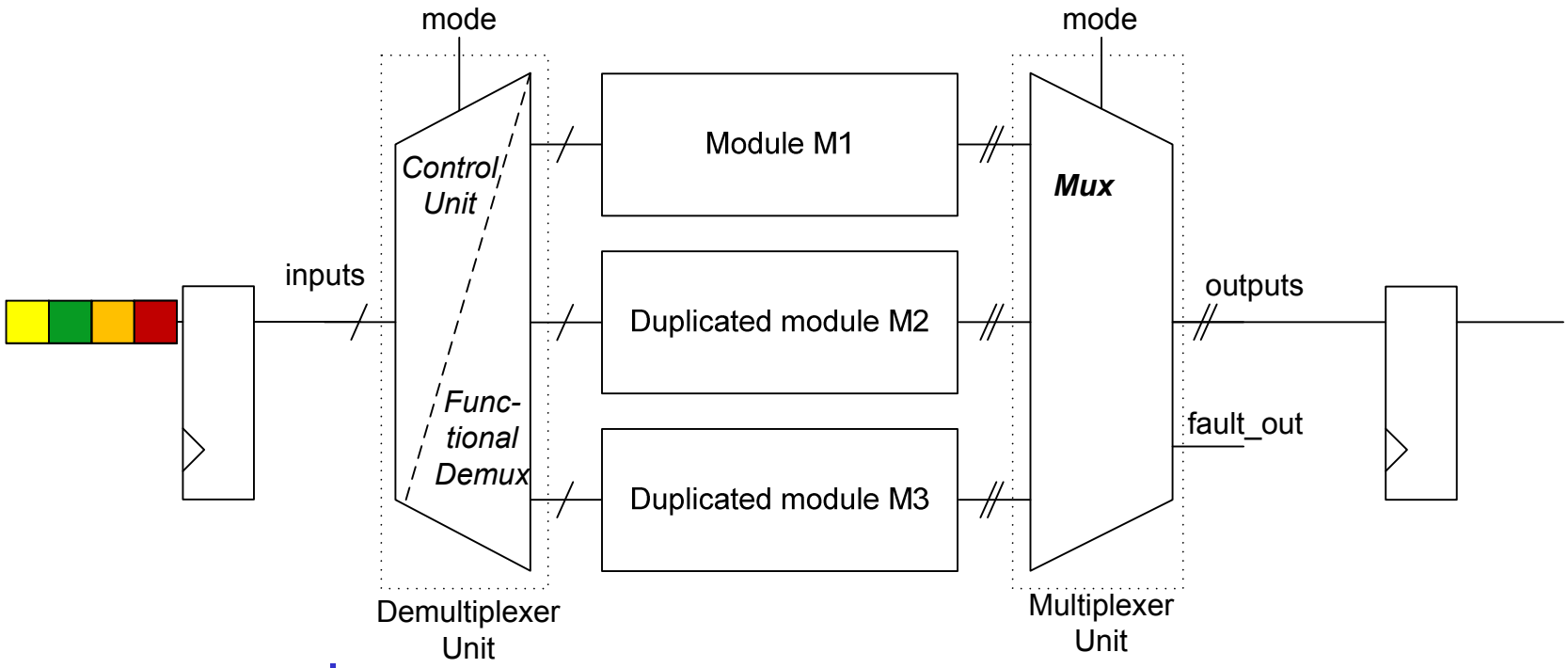


- Inherent parallel data paths of TMR

→ Power reduction possible

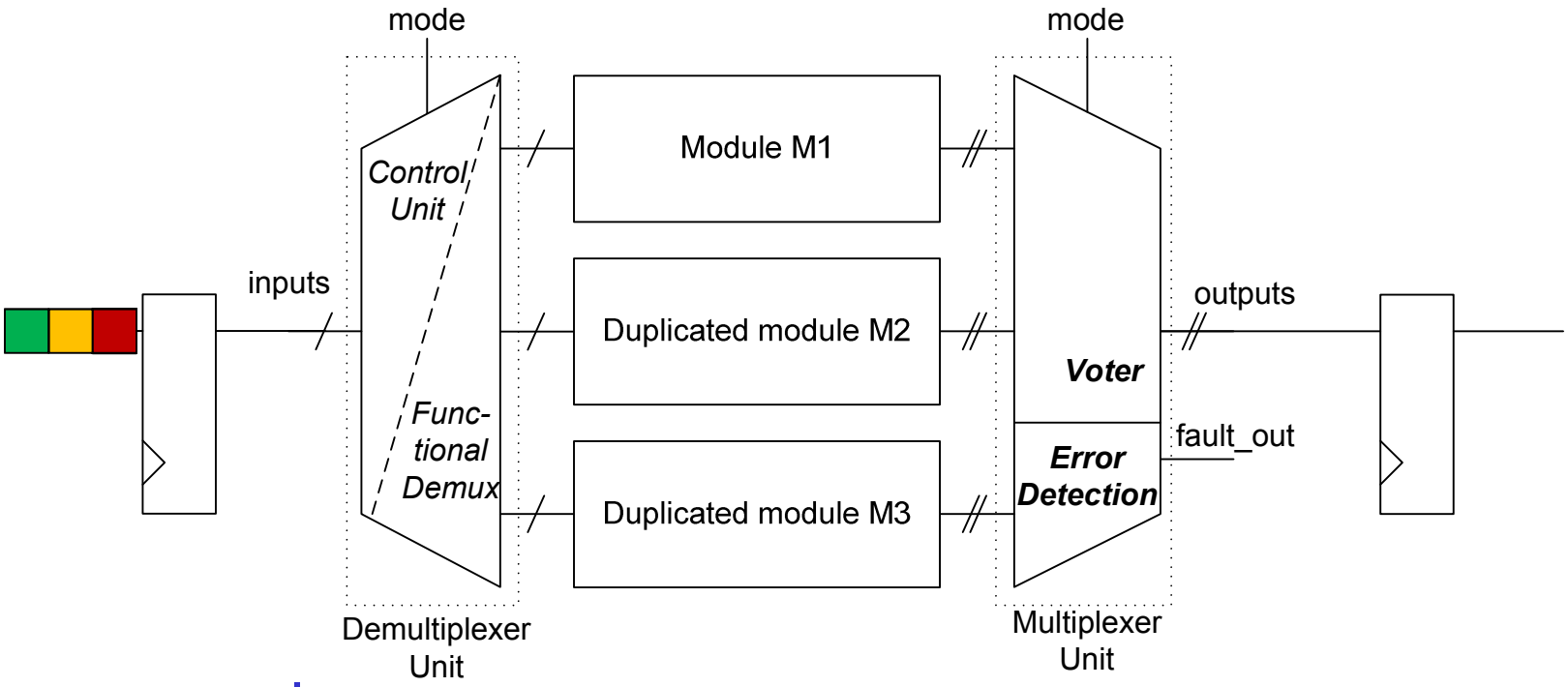
Design idea – NTMR operation mode

Power efficient operation mode
Every module executes the original function on different data sets
→ Creation of a 3-fold parallel system → $f_{MODULEx} = 1/3\ f_{OPERATION}$

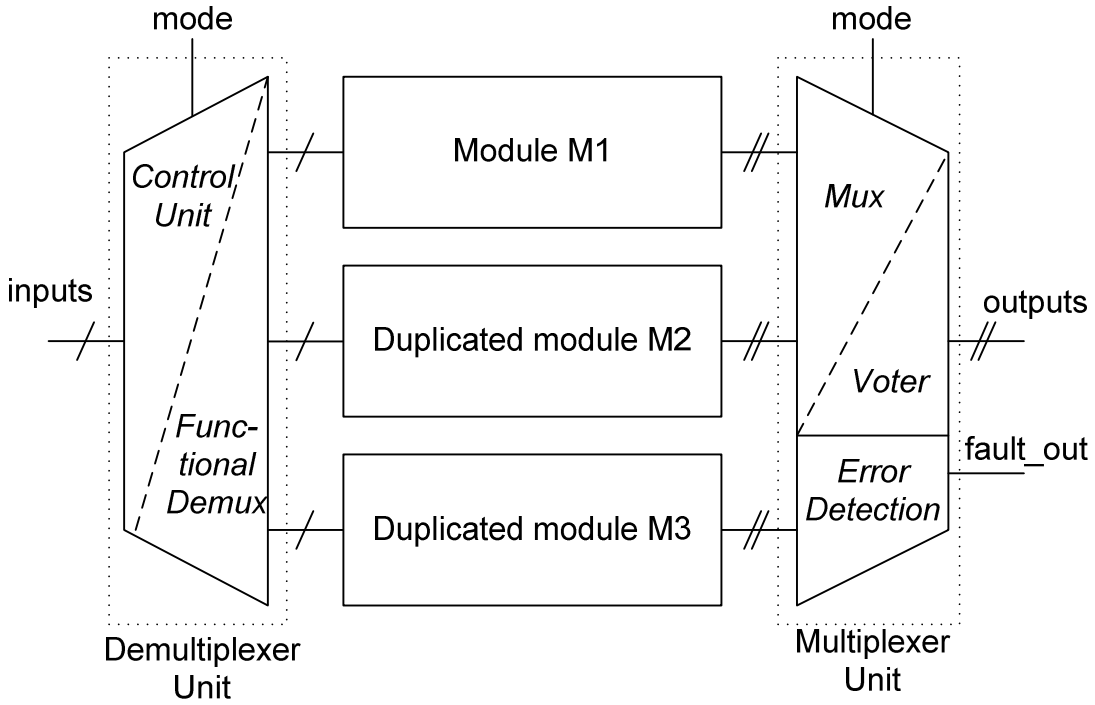


Design idea – TMR operation mode

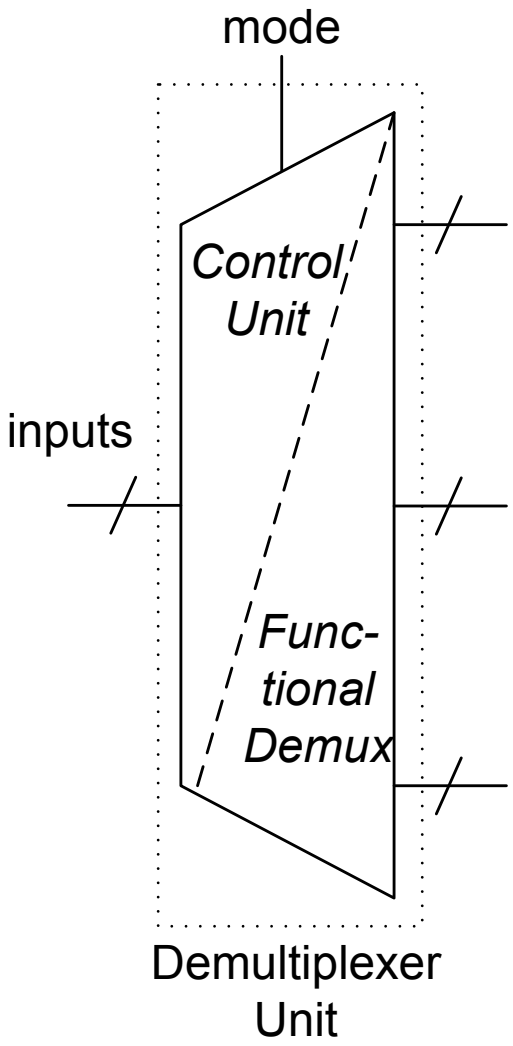
For error detection – Usual TMR
Same inputs for all modules



Design Structure



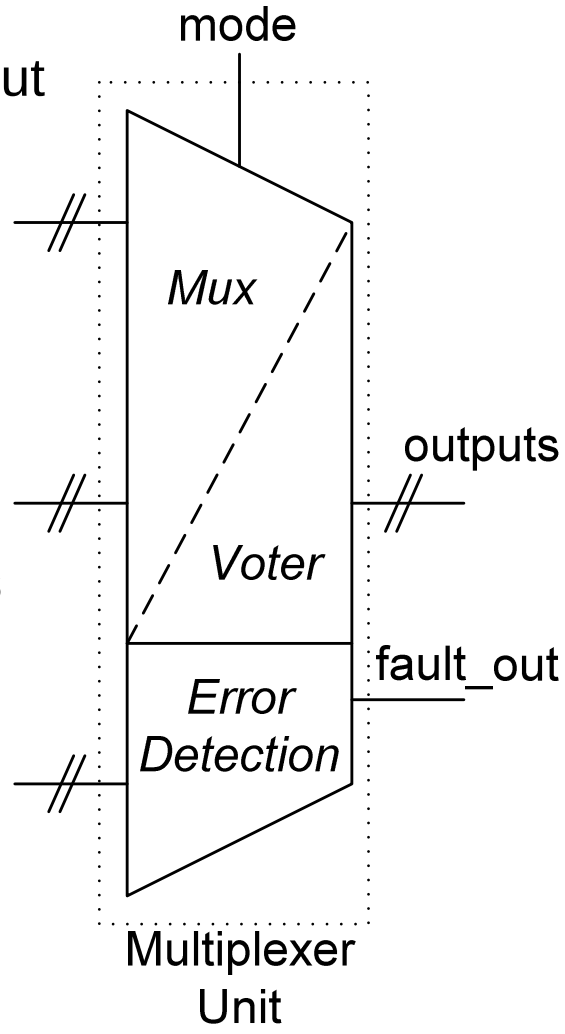
Design Structure – Demultiplexer Unit



- Assignment of the correct data inputs to the appropriate modules
- TMR mode
  - Enabling all inputs concurrently → $f_{ORIGINAL}$
- NTMR mode
  - Enabling the inputs of ONE module during a clock phase (disabling of the two other ones)
    → 1/3 $f_{ORIGINAL}$

Design Structure – Multiplexer Unit

- Assignment of the correct data output
- TMR mode
  - Voting of the module outputs
  - Voting result as design output
- NTMR mode
  - The appropriate module output has to be routed to the design output
- Error detection during TMR mode
  - XOR implementation + OR-tree
  - Small amount of inherent reliability due to 3 XORs per output



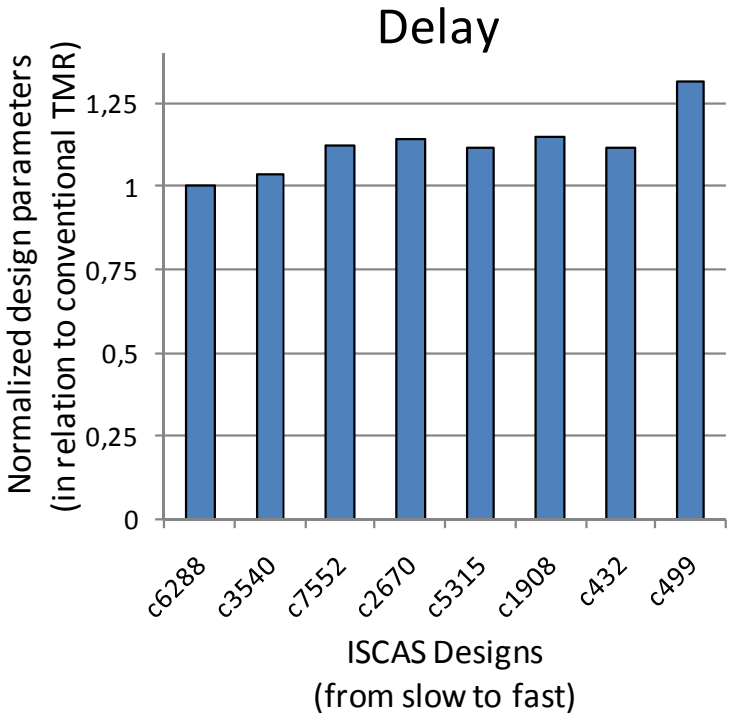
Simulation Setup

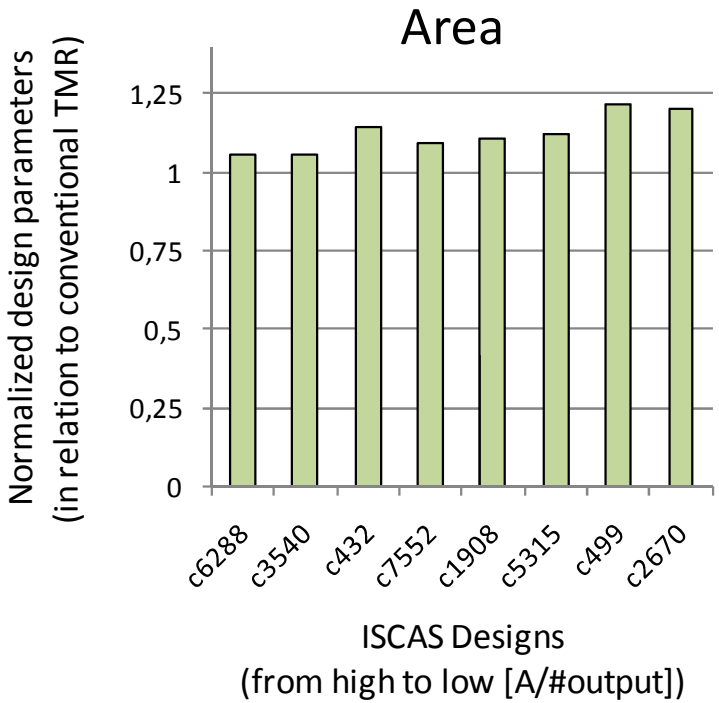
- Industrial 65 nm technology
- Gate level simulation
- Toggle rate analysis for power calculation

- Stuck-at faults with randomly chosen locations (incl. periphery)
- Constant failure rate for every net of the system

  → Area dependent system failure rate

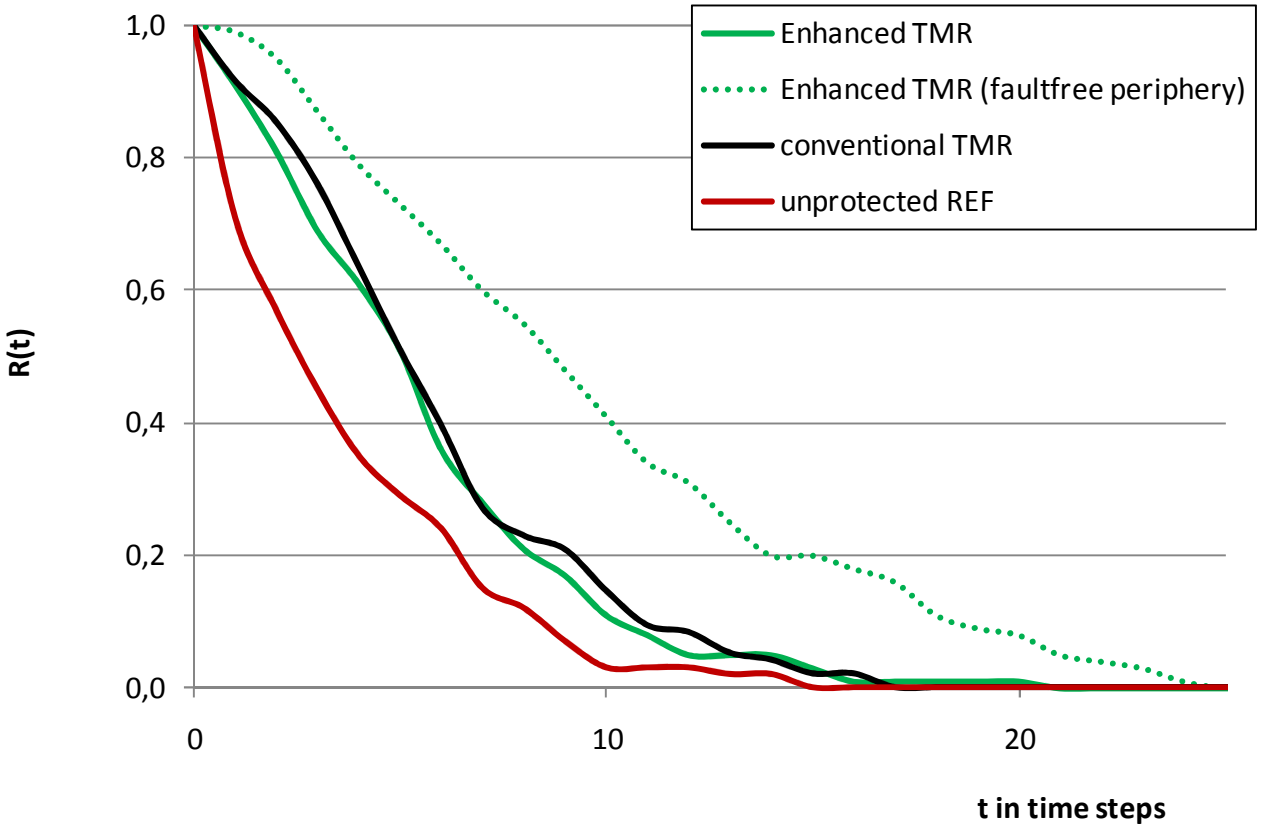
Results – Design overhead





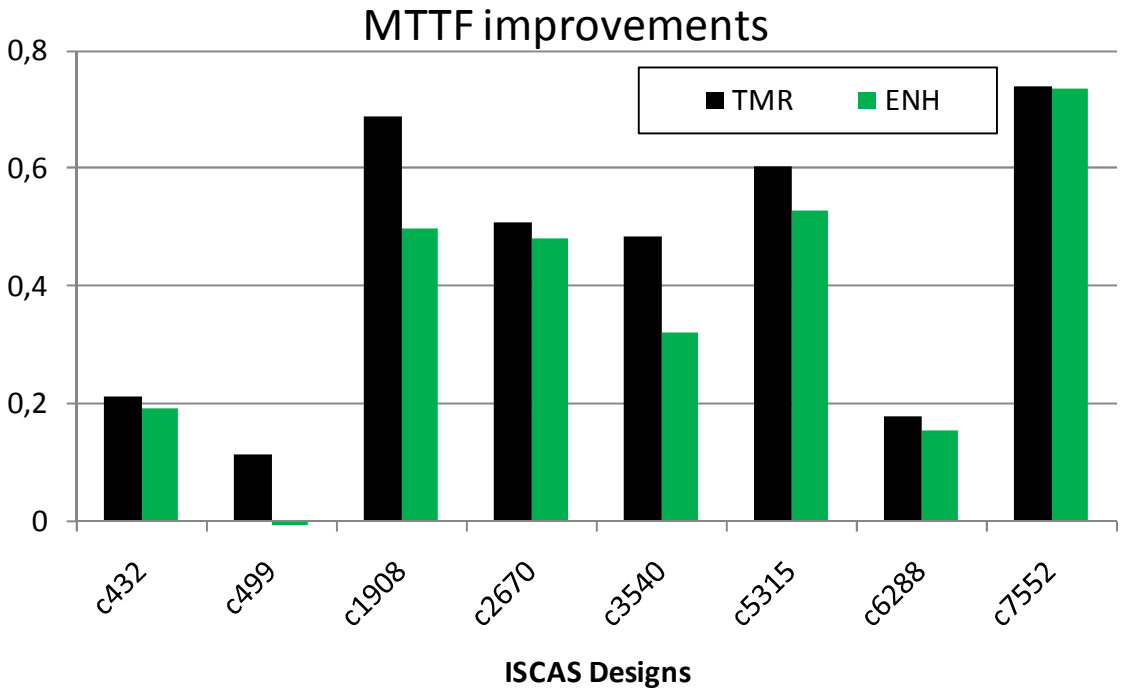
- Lowest overhead for large and slower designs with few output bits

Results – Reliability



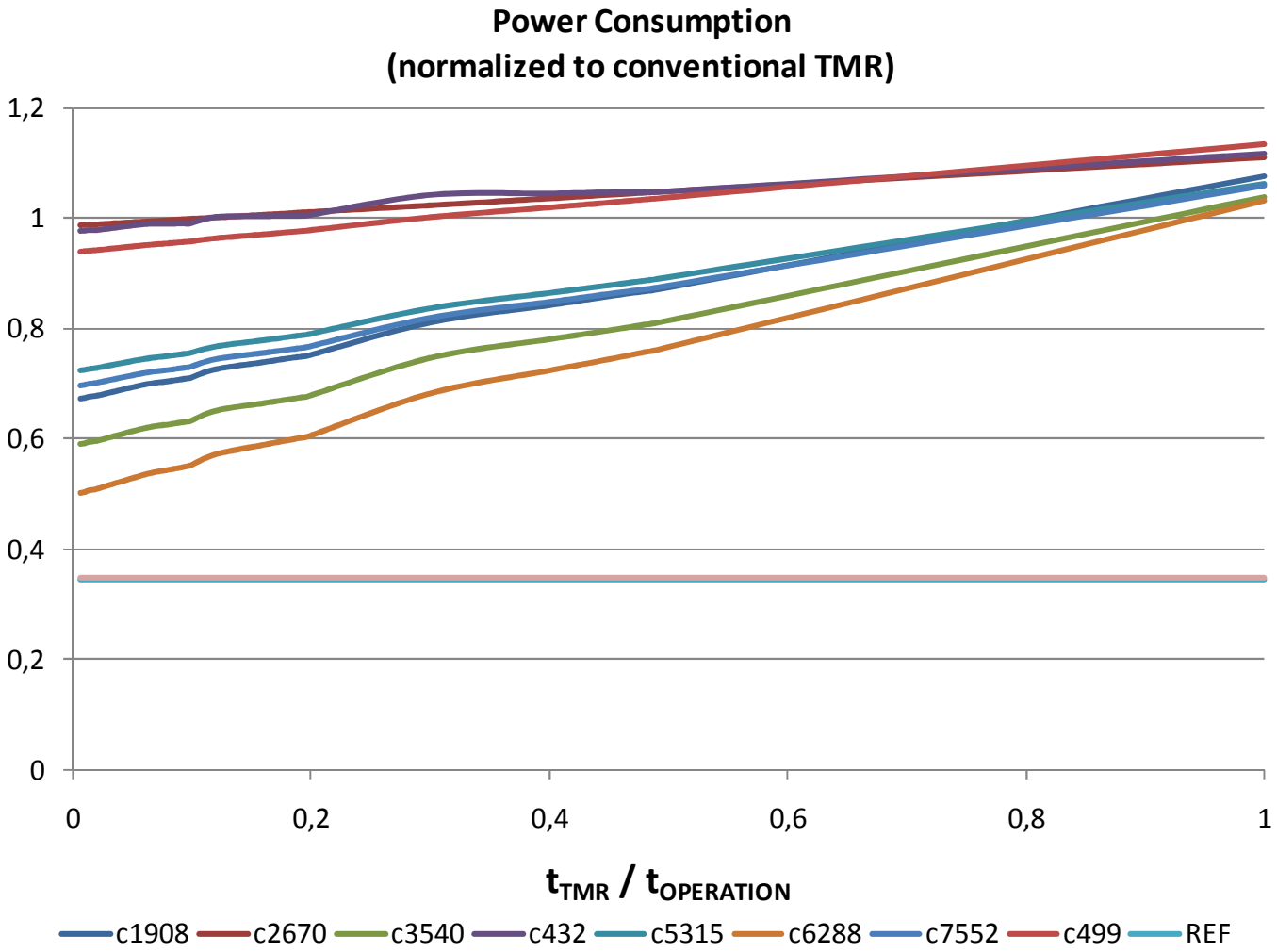
c5315

Results – MTTF improvements

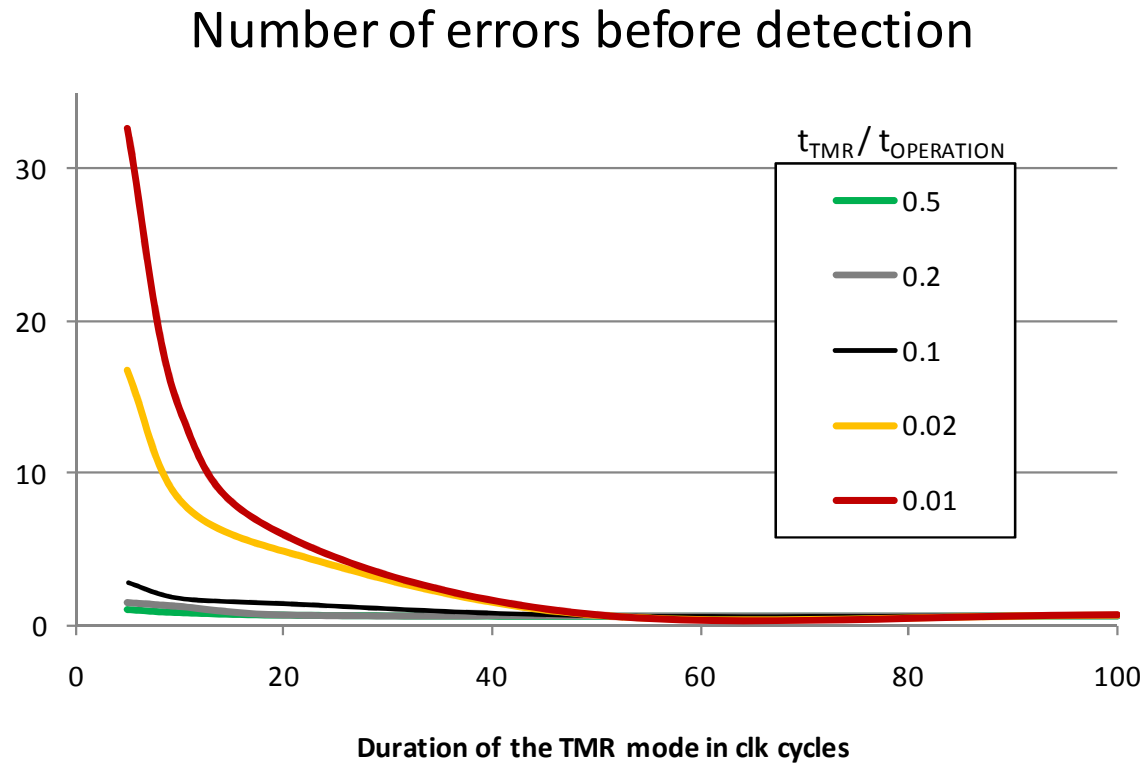


- Larger designs suffer less from area overhead

Results – Power consumption



Results – Switching Strategies



- Fast detection of faults in the module with
  - Sufficient high t_{TMR} / $t_{OPERATION}$ rate
  - Sufficient long TMR mode duration → possible low t_{TMR} / $t_{OPERATION}$ rate

Conclusion / Outlook

- Lifetime reliability will be a major issue in ciruit design
- Design of an enhanced power-efficient TMR design using the inherent parallelism of TMR
- Power savings up to 50 % compared to conventional TMR are achievable with a slightly overhead in area, latency

- Further enhancements
  - *Reduced Supply voltage*
  - *More reliable periphery*
  - *Implementation of further operation modes*