RAEF: A Power Normalized System-level Reliability Analysis and Estimation Framework

R.A. Shafik[1], Bashir M. Al-Hashimi[2], J. Mathew[1], D.K. Pradhan[1] and S.P. Mohanty[3]

[1]Dept of Computer Science, University of Bristol, UK
[2]School of Electronics and Computer Science, University of Southampton, UK
[3]Dept of Computer Science and Engineering, University of North Texas, USA

Presenter:
Geng Zheng
University of North Texas
Email: gengzheng@my.unt.edu

University of BRISTOL

UNIVERSITY OF Southampton

UNT UNIVERSITY OF NORTH TEXAS Discover the power of ideas

Outline

- Introduction: System-level Reliable Design
- Motivations and Contributions
- Power Normalized Reliability Metric
- RAEF: Proposed Reliability Estimation Framework
- Results: MPEG-2 Decoder and Other Applications
- Conclusions

Introduction

- Reliability in the presence of soft errors is an emerging design challenge
  - Further exacerbated by continued technology scaling
- System-level design is highly effective in reliable multiprocessor system-on-chip (MPSoC) design
- A crucial aspect in such design is to estimate the reliability in early design phase
  - with an aim to assess the comparative component reliabilities and design low-cost reliable system

Previous Work

- To estimate reliability at system-level various approaches have been used over the years
  - Hierarchical Monte-Carlo based estimation from component-level to MPSoC system [Xiang *et al*]
  - MTTF based MPSoC reliability model based on statistical modelling [Coskun *et al*]
  - Mean Error Impact based MPSoC reliability model [Wu and Marculescu]
  - Reliability metric based system-level analysis and estimation [Zhao *et al*]

Motivation

- Currently power and reliability estimations are carried out separately
- Since power minimization directly affects reliability, component reliability comparisons do not signify joint consideration of power and reliability
  - Which is much needed for a system where low power and high reliability are joint objectives
- Hence, a composite metric is much needed highlighting power and reliability trade-offs

Motivation

[Examples of separate measurements]

Component	Scaling	Reliability	Power, mW
comp1	66.7MHz @ 0.55V	0.96	0.98
comp2	100MHz @ 0.6V	0.98	1.96
comp3	200MHz @ 1V	0.99	9.24

- *comp1* has low power but low reliability; achieved through aggressive voltage scaling
- *comp3* has high reliability at the expense of high power; no voltage scaling applied
- *comp2* has reliability of 0.98 and a power consumption of 1.98mW
- With separate power and reliability measurements, it is hard to comparatively assess the system reliability

Contribution

- We propose a novel composite metric, called power normalized reliability (PNR)
  - Expressed as a ratio of system reliability and power
  - Aim is to highlight power and reliability trade-offs at system- and component-level

- Underpinning this metric, a novel estimation framework, reliability analysis and estimation framework (RAEF) is presented
  - For effective fault injection, analysis and estimation at various architectural hierarchies: register-level, core-level and system-level

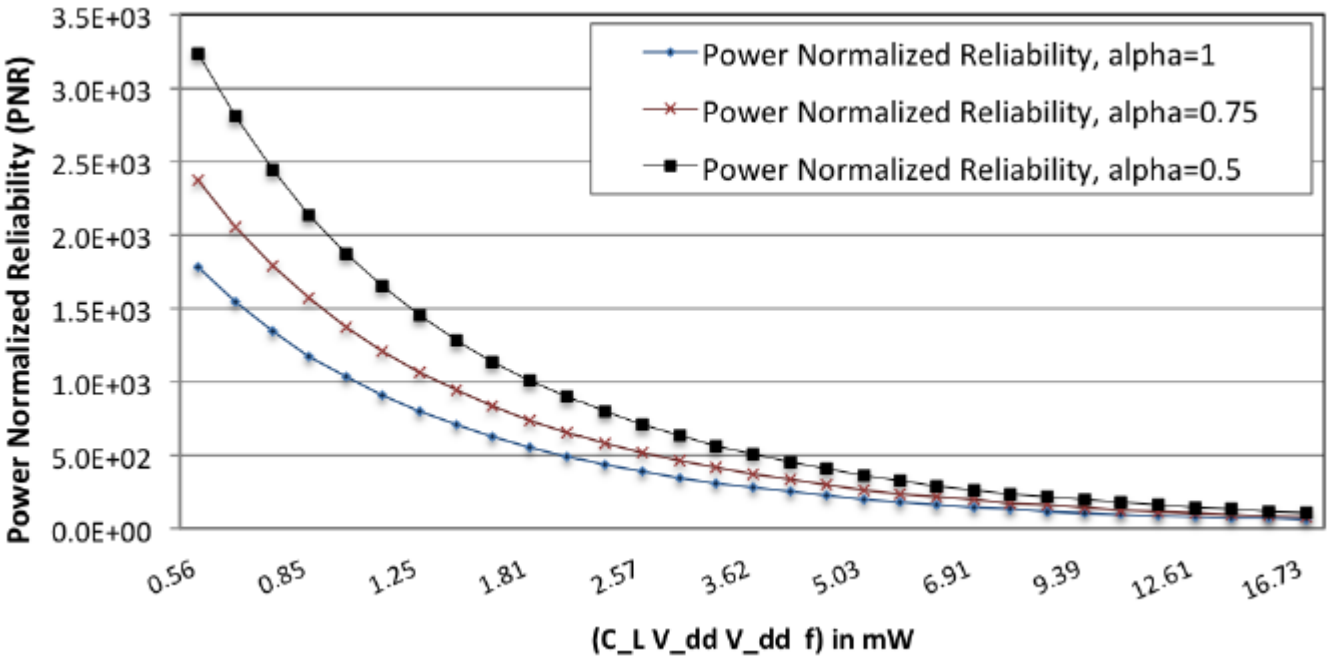
Power Normalized Reliability (PNR)

$$PNR = \frac{R}{P} = \frac{exp\left[-\lambda_0\ k\ v\ t\right]}{\alpha\ C_L\ V_{dd}^2 f}$$

R is reliability expressed as a function of basic soft error rate (λ_o), factor of soft error increase with voltage scaling (k), architectural vulnerability factor (v) and time (t)

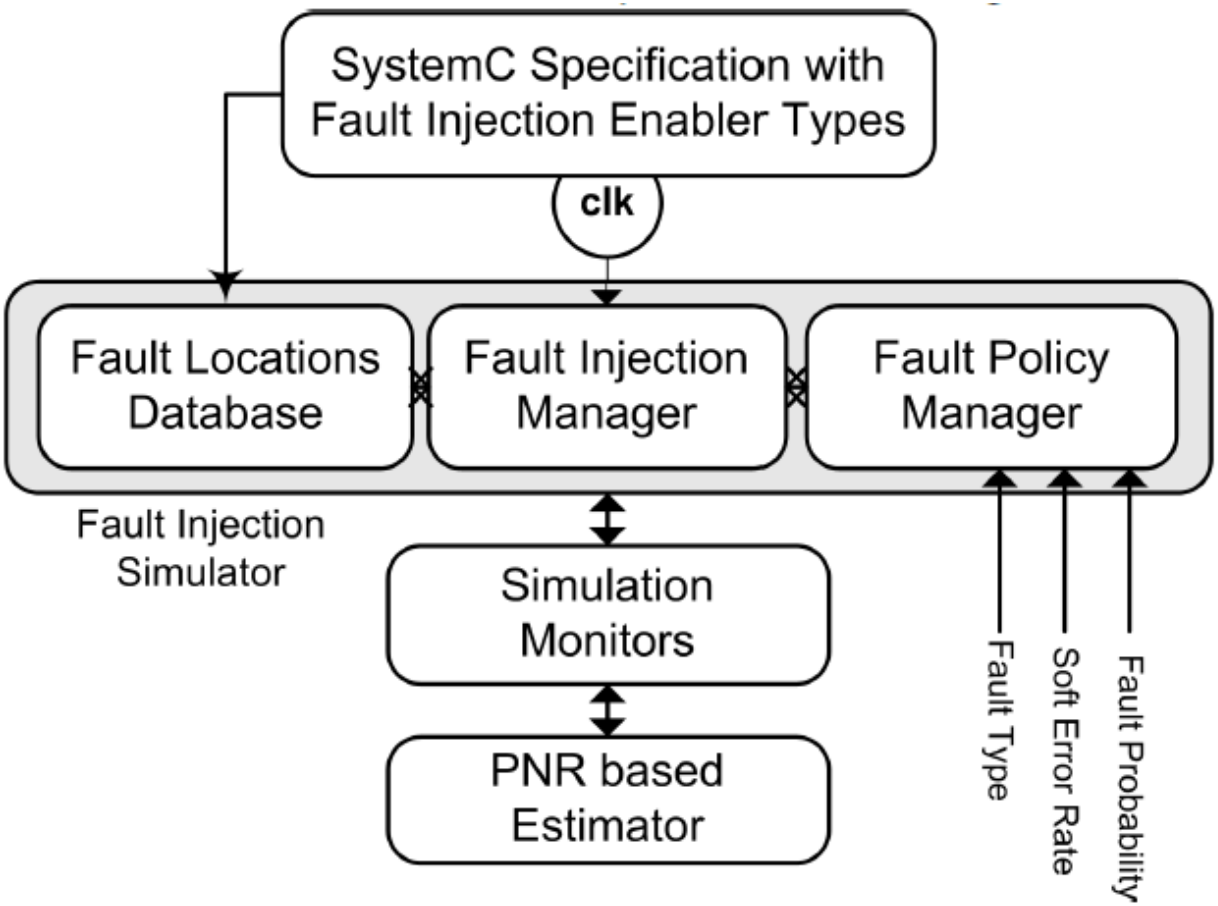
P is dynamic power expressed as a function of activity factor (α), load capacitance (C_L), supply voltage (V_{dd}) and operating frequency (f)

Power versus PNR

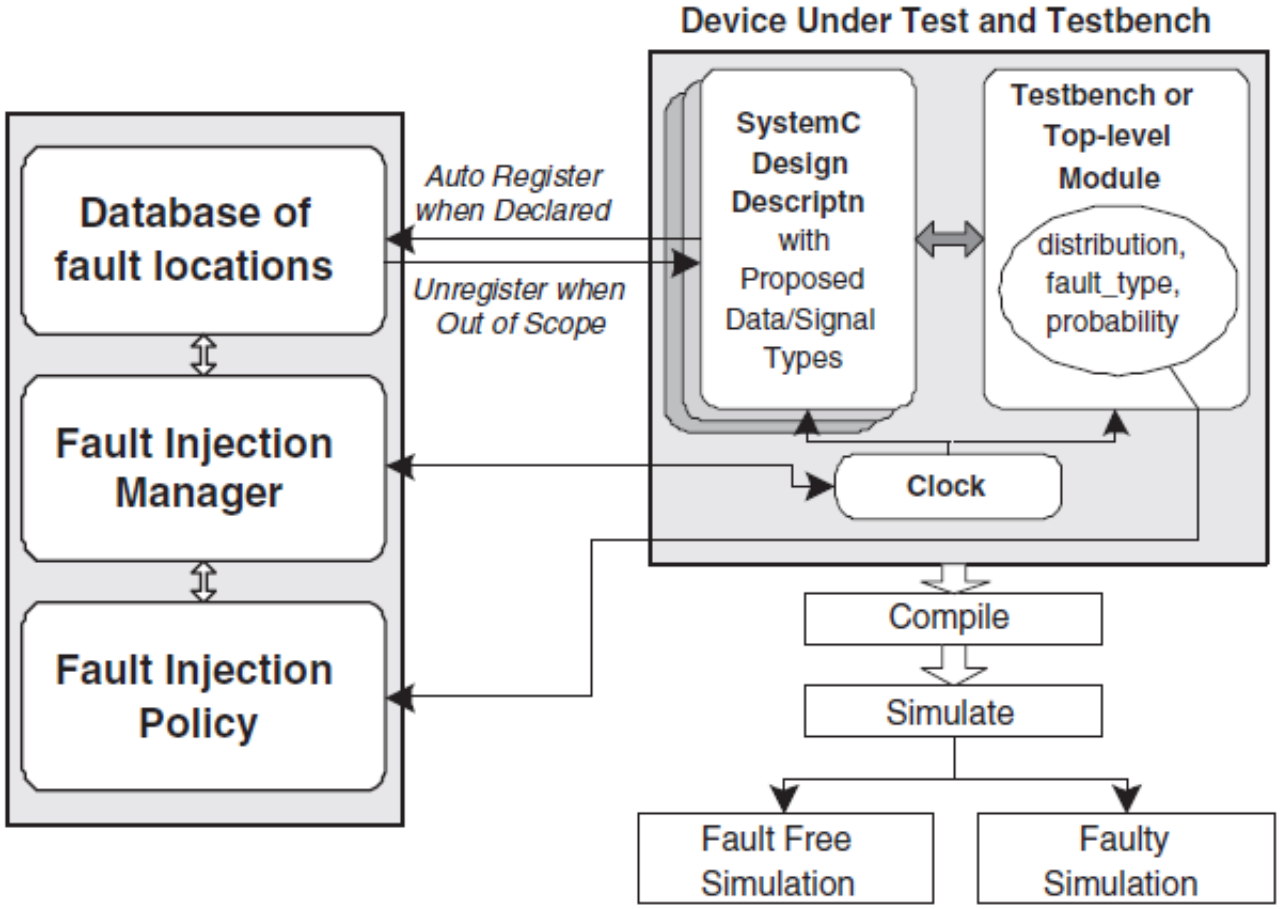


- PNR is high at lower power and low at higher power
- It varies significantly due to different activity factors

RAEF: Proposed Estimation Framework



Fault Injection in RAEF



- Fault injection enabler types form a fault locations database
- Faults are injected at random times/locations based on fault policy

Hierarchical PNR Estimation
[@ Register-level]

- PNR is estimated for each register in design specification

$$PNR_{i,c} = \frac{R_{i,c}}{P_{i,c}} = \frac{\exp\left[-g_{i,c}k_c\lambda_b t\right]}{\frac{t^b_{i,c}}{t}C_L V_{dd}^2 f}$$

$R_{i,c}$ is reliability (ı function of soft error rate (λ_b, per bit per unit time), factor of soft error increase with voltage scaling on c-th core (k_c), size of the register ($g_{i,c}$) and time (t)

$P_{i,c}$ is dynamic power of i-th register in c-th core expressed as a function of register activity factor ($\alpha_{i,c}$: ratio of busy cycles, $t^b_{i,c}$ and time, t), load capacitance (C_L), supply voltage (V_{dd}) and operating frequency (f)

NOTE: at this level vulnerability does not affect component reliability

Hierarchical PNR Estimation
[@ Processing Core-level]

- At processing core-level PNR is estimated for each core as

$$PNR_c = \frac{R_c}{P_c} = \frac{\exp\left[-\sum_{i=1}^{G_c} \lambda_{i,c} \left(\Gamma^V_{i,c}/\Gamma^A_{i,c}\right) t\right]}{\frac{t^b_c}{t} C_L V^2_{dd} f}$$

R_c is reliability of c-th core expressed as a function of component soft error rate ($\lambda_{i,c}$, per bit per unit time), component vulnerability factor (υ: ratio of visible fault, $\Gamma^V_{i,c}$ and actual number of fault injected, $\Gamma^A_{i,c}$), and time (t)

P_c is dynamic power of c-th core expressed as a function of core activity factor ($\alpha_{i,c}$: ratio of busy cycles, t^b_c and time, t), load capacitance (C_L), supply voltage (V_{dd}) and operating frequency (f)

NOTE: At higher architectural level, vulnerability becomes important.

Hierarchical PNR Estimation
[@ System-level]

- At system-level, PNR is estimated for the overall MPSoC system as

$$PNR = \exp\left[-\sum_{c=1}^{C} \lambda_c \left(\Gamma_c^V / \Gamma_c^A\right) t\right] \Big/ \sum_{c=1}^{C} P_c$$

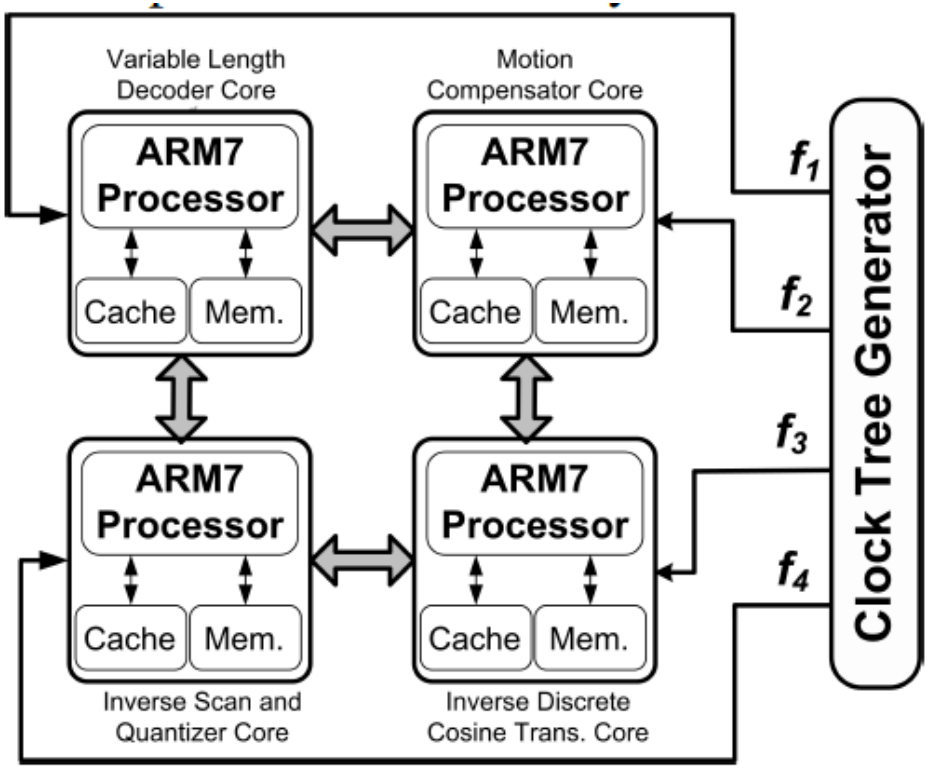
Reliability of MPSoC system is expressed as a function of per core core soft error rate (λ_c, per bit per unit time), component vulnerability factor (v: ratio of visible fault per core, Γ^V_c and actual number of fault injected per core, Γ^A_c), and time (t)

Overall power *is* expressed as a sum of core dynamic powers

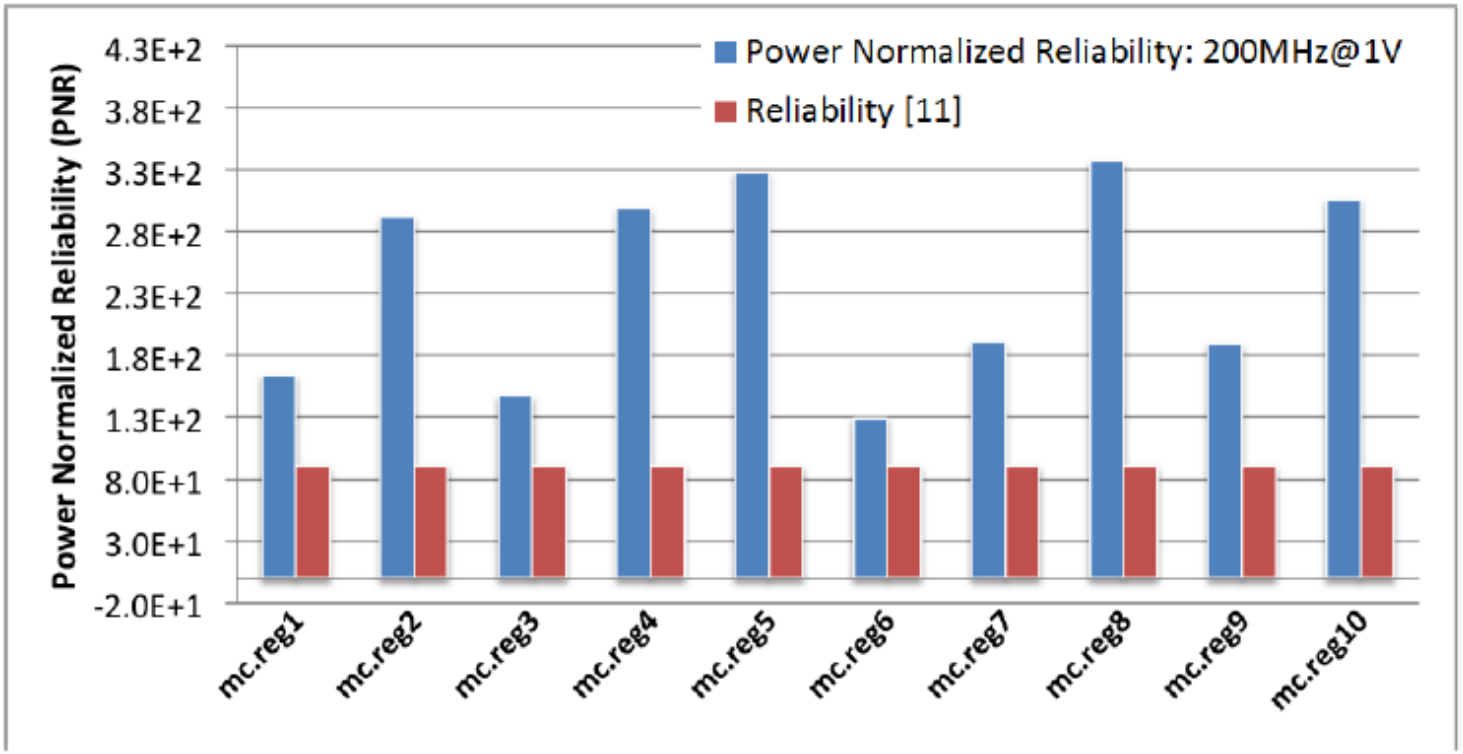
NOTE: At system-level, PNR is determined by various factors including component vulnerability, activity factor and observation time.

Case Study: MPEG-2 Decoer

- MPSoC with four decoder cores
- RAEF will be used to examine PNR based reliability at
  - register-level
  - core-level and
  - overall system-level.

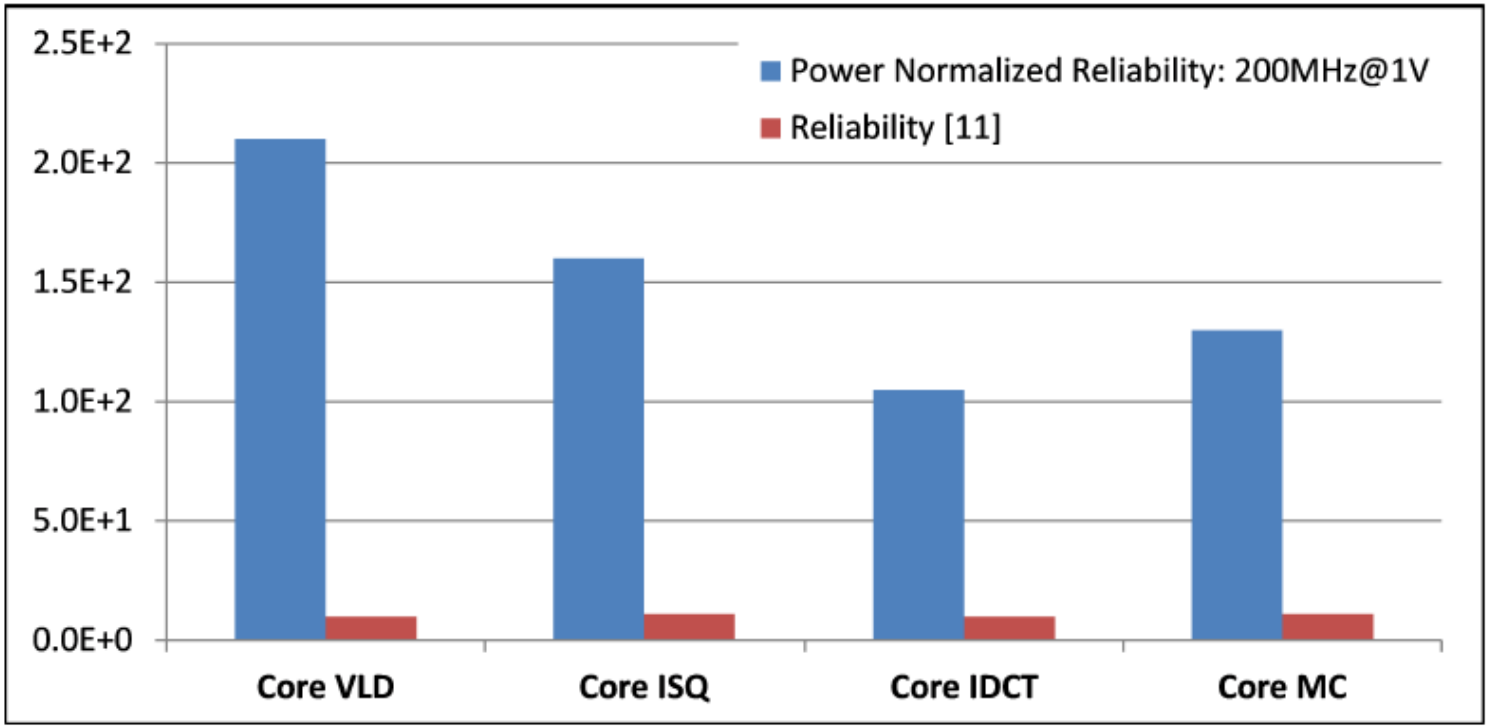


PNR Estimates at Register-level



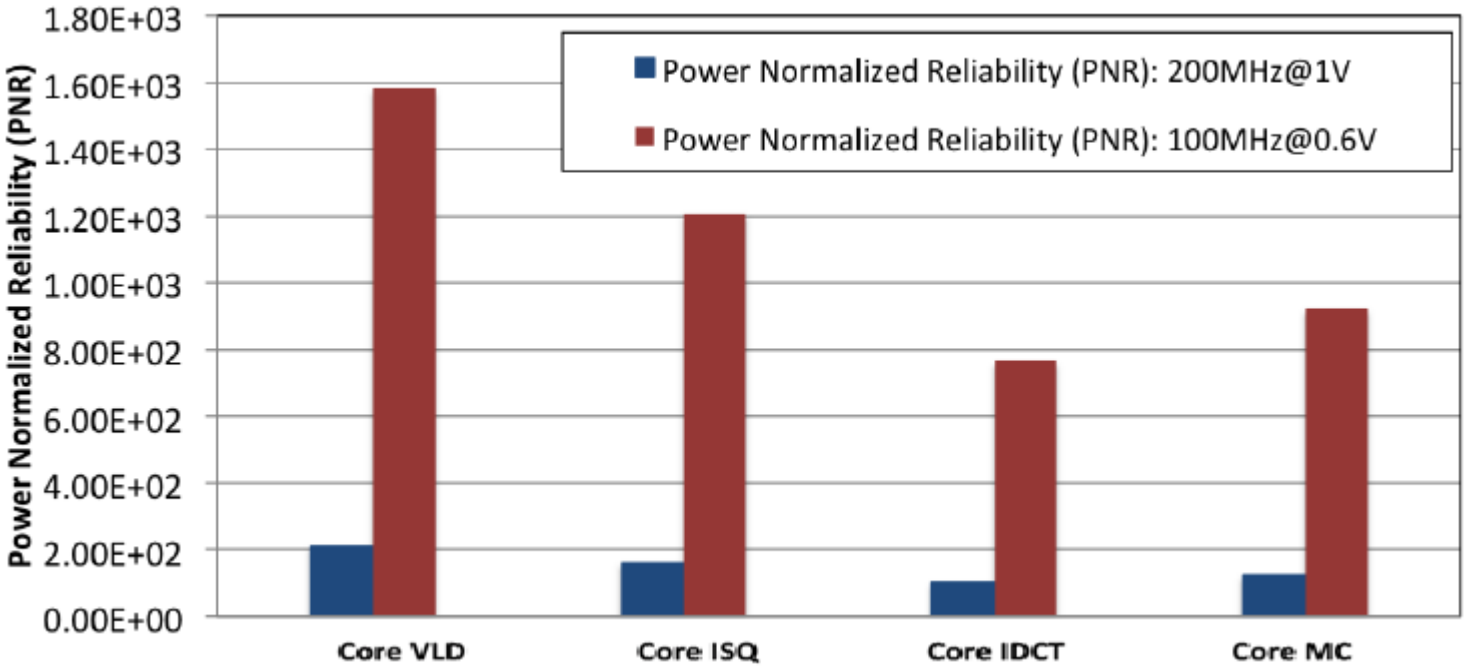
- Longer registers have low reliability and hence lower PNR
- Registers with low activity factor have high PNR
- COMMENT: PNR clearly signifies the trade-off, while traditional reliability fails to do so

PNR Estimates at Core-level



- Processing cores with low activity & vulnerability factor have high PNR
- COMMENT: At core level, reliability metric has minor variations and does not highlight power trade-off; PNR overcomes this.

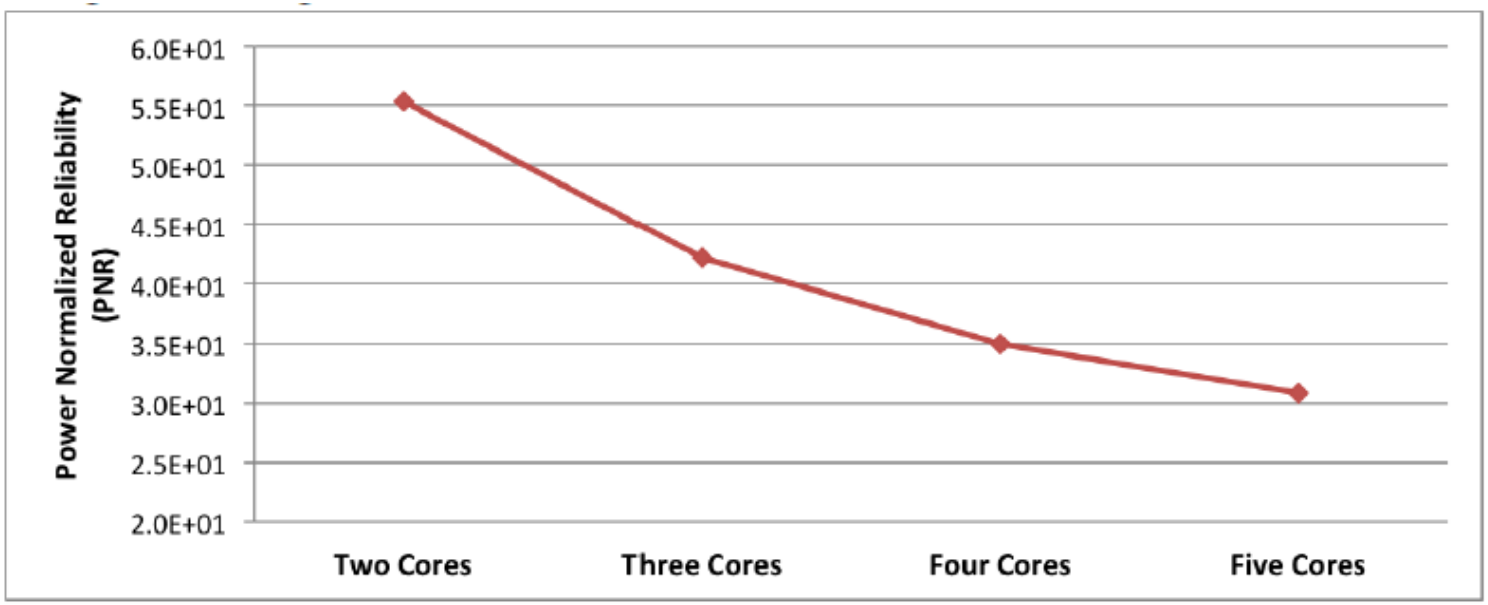
Impact of Voltage Scaling
[and activity factor]



Processing cores with low activity factor has low power and hence higher PNR; Voltage scaling significantly improves PNR

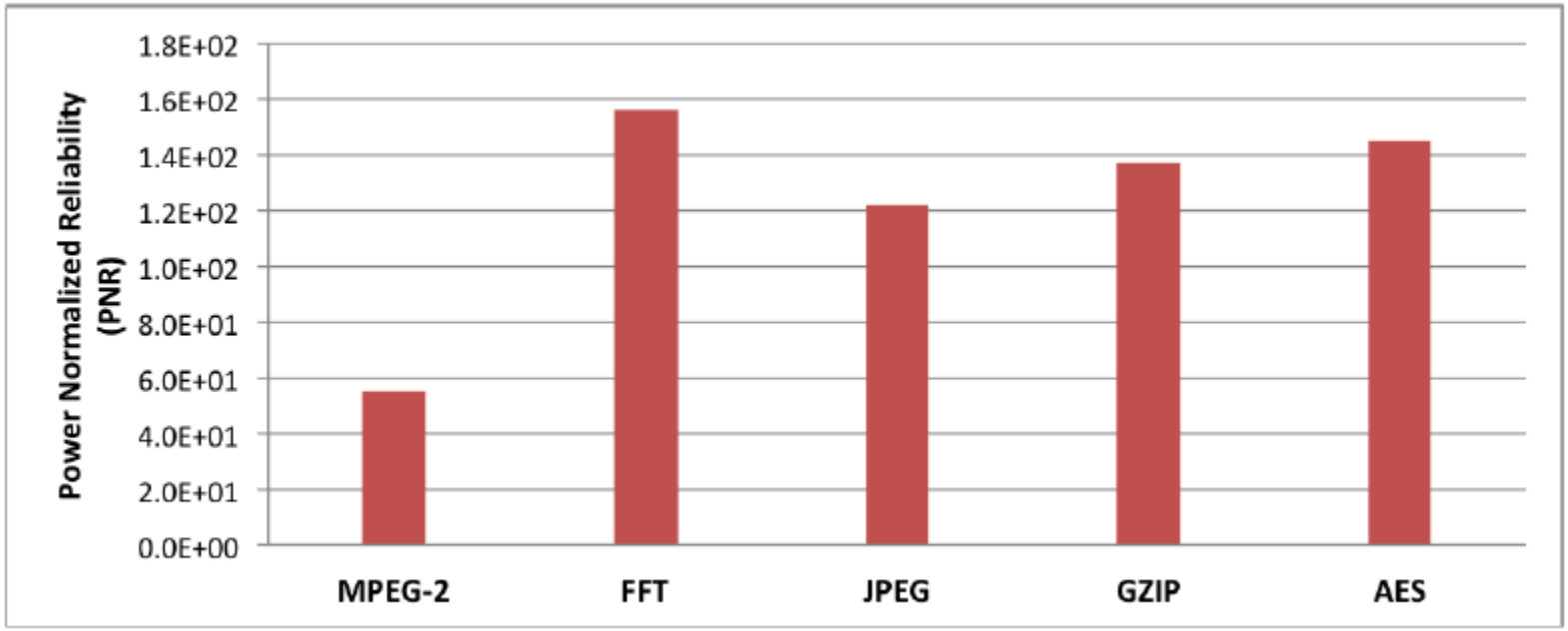
NOTICE: Core VLD (which has lowest overall activity factor) has the highest PNR, while core MC (which has the highest overall activity factor) has the lowest PNR

PNR Estimates at System-level
[Impact of Architecture Allocations]



With higher architecture allocation, register usage increases, which also degrades core reliabilities and hence overall PNR

PNR Estimates at System-level
[Difference between various applications]



Applications with higher activity factor and higher register usage had lower overall PNR; hence, MPEG-2 is outperformed by other applications shown

Conclusions

- Proposed PNR based estimation is highly effective
  - It highlights the comparative reliability between components, taking into consideration power and reliability jointly
- Using PNR, an analysis and estimation framework (RAEF) was shown
  - Enables PNR based insightful analysis and estimation using system-level simulation techniques
- The effectiveness of RAEF using PNR was evaluated
  - Using MPEG-2 and various other applications
  - Considering the impact of voltage scaling, architecture allocation and observation times on PNR

THANK YOU