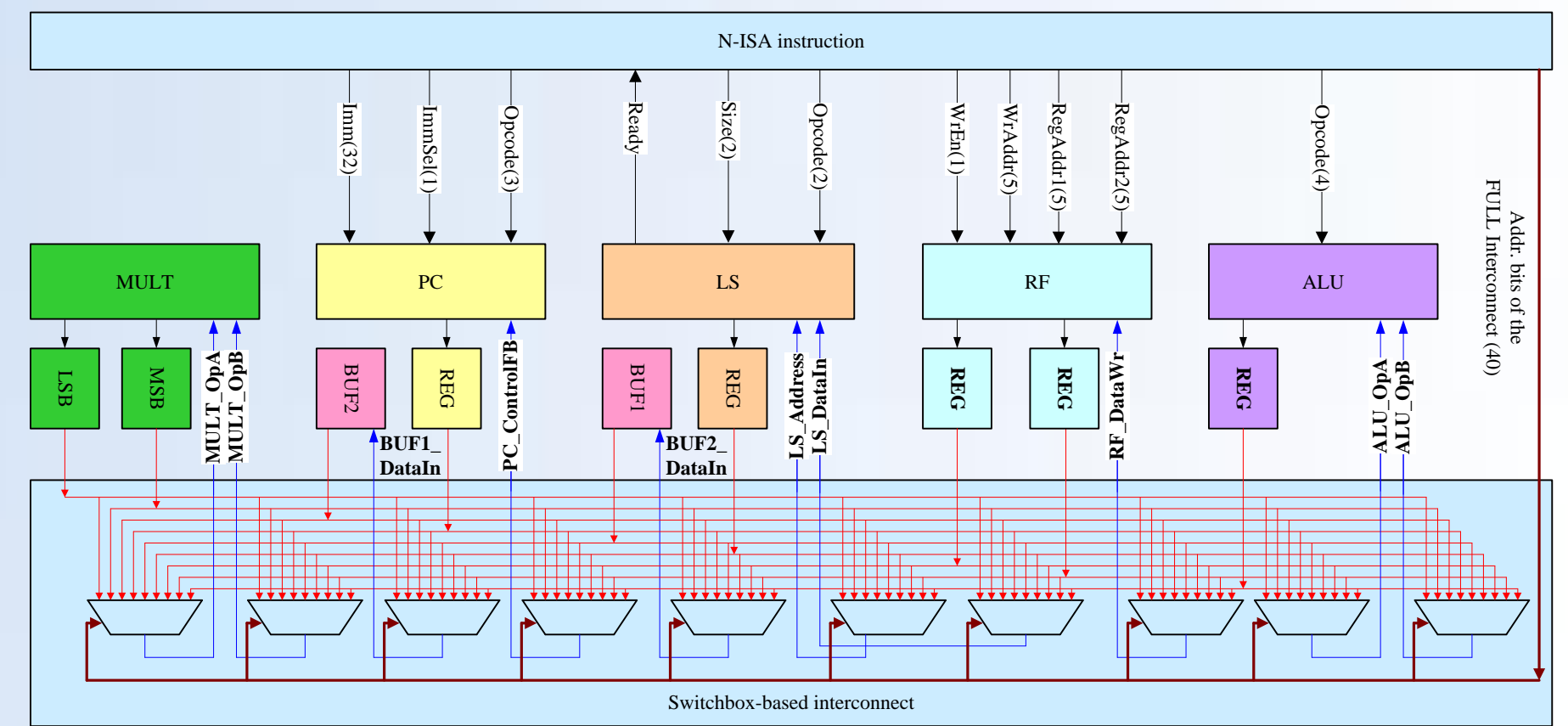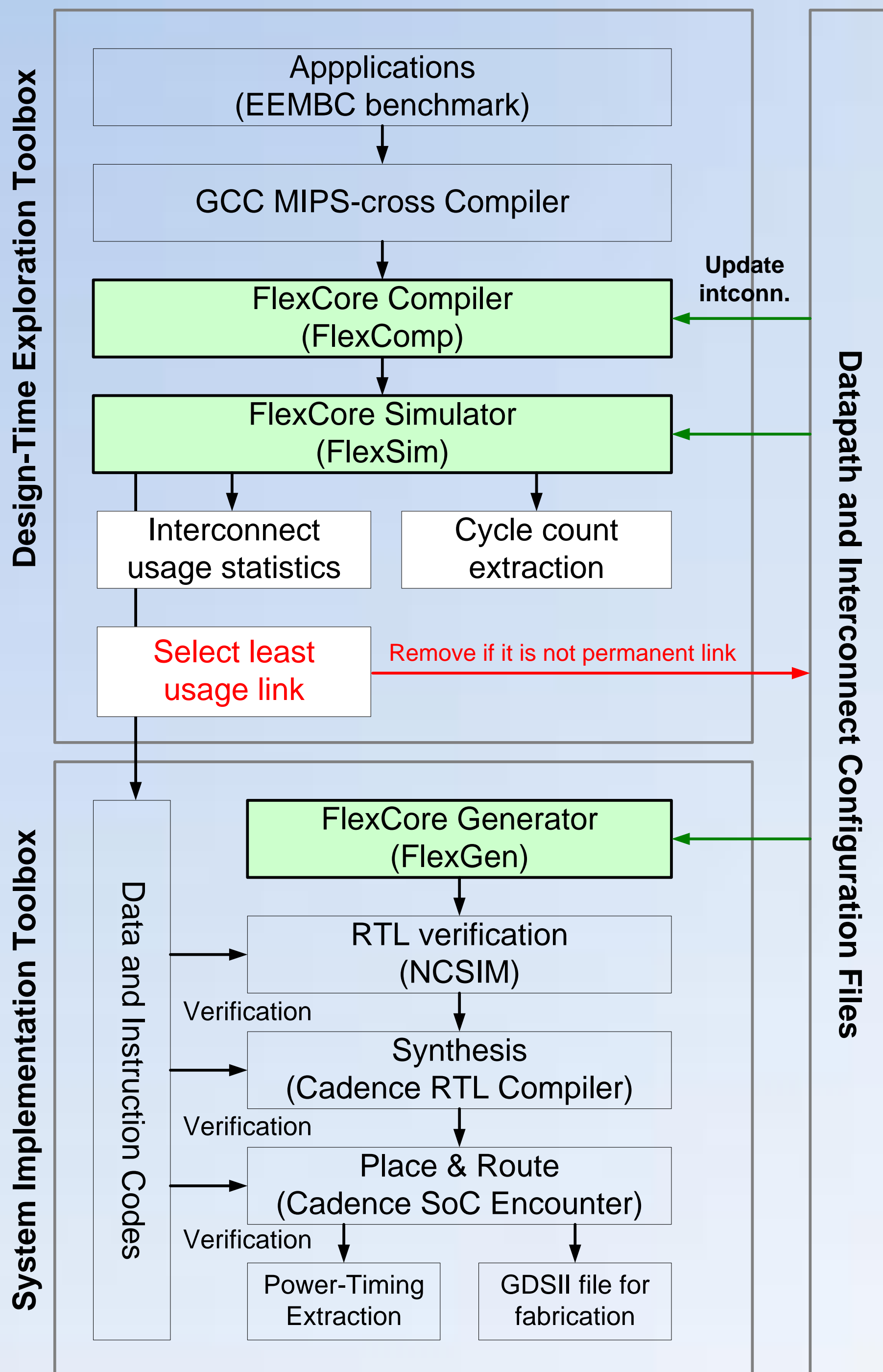# CHALMERS

# FlexTools: Design Space Exploration Tool Chain from C to Physical Implementation

Erik Ryman, Kasyab P. Subramaniyan, Tung T. Hoang,
Mafijul Islam, Magnus Själander, and Per Larsson-Edefors,
Chalmers University of Technology

**Design-Time Exploration Toolbox**

- Appplications (EEMBC benchmark)
- GCC MIPS-cross Compiler
- FlexCore Compiler (FlexComp) — Update intconn.
- FlexCore Simulator (FlexSim)
- Interconnect usage statistics
- Cycle count extraction
- Select least usage link — Remove if it is not permanent link

**System Implementation Toolbox**

- Data and Instruction Codes
- FlexCore Generator (FlexGen)
- RTL verification (NCSIM) — Verification
- Synthesis (Cadence RTL Compiler) — Verification
- Place & Route (Cadence SoC Encounter) — Verification
- Power-Timing Extraction
- GDSII file for fabrication

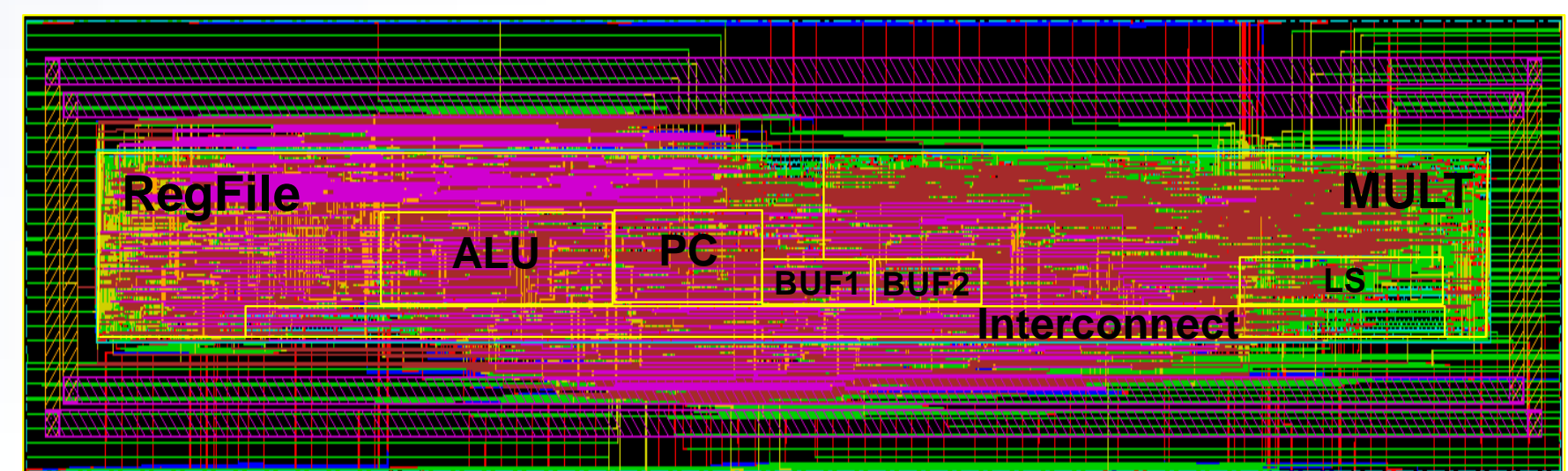**Datapath and Interconnect Configuration Files**



The FlexCore processor uses a flexible interconnect instead of a pipeline to connect all parts of the datapath. Each data path unit is linked to each other unit, this allows for a more flexible scheduling that leads to reduced cycle counts.
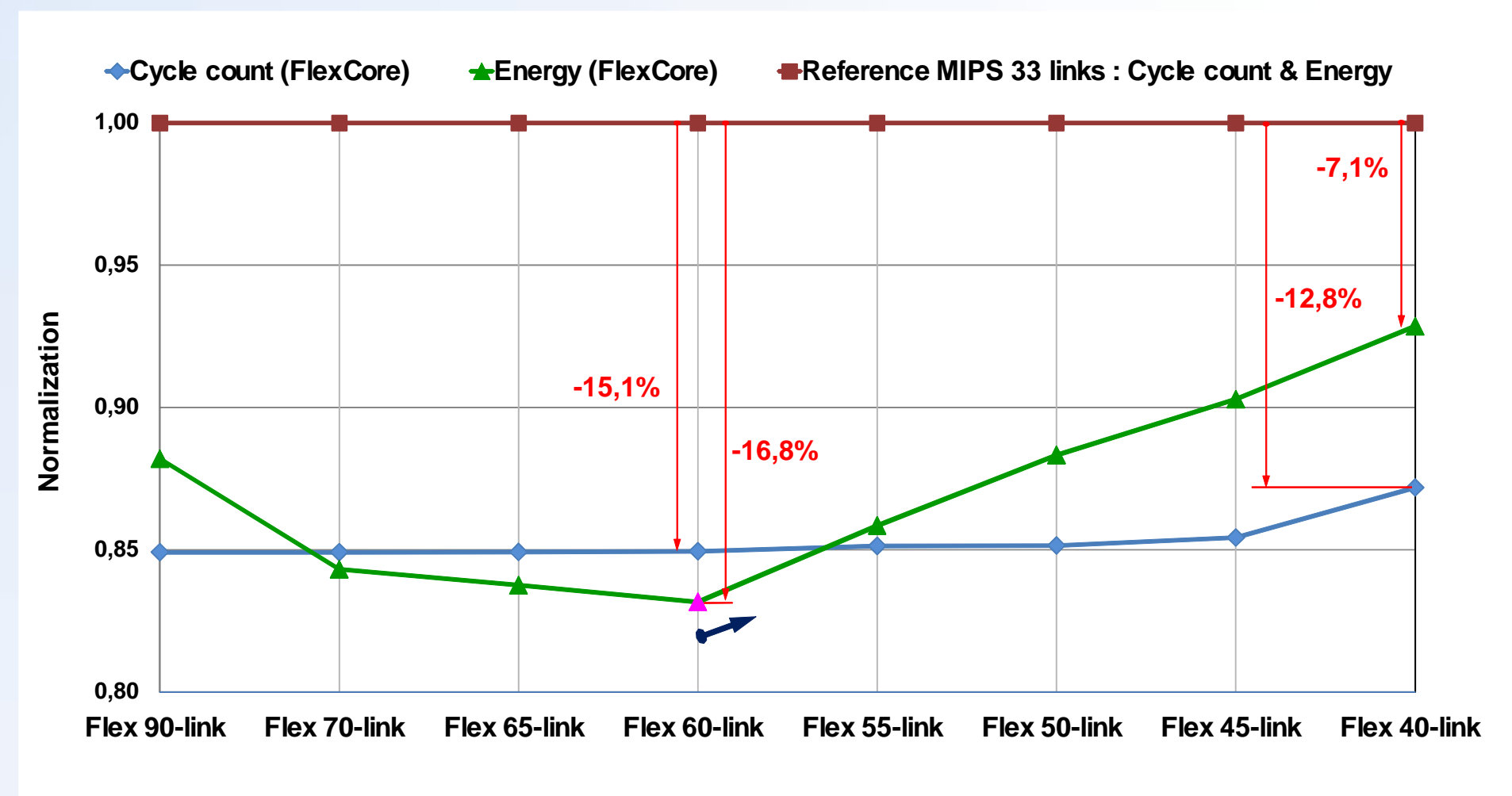


A full interconnect provides maximum flexibility but is costly in area and power. To optimize the interconnect, links that are unused or not frequently used can be removed. FlexTools allows for link usage, cycle count and energy statistics to be evaluated for specific applications and technologies. The graph above shows an average of the evaluation of 9 EEMBC benchmarks in a 65 nm technology.



Our proposed toolset facilitates design and evaluation of numerous prototypes of FlexCore processors. We currently focus on the design of a first test chip to evaluate a model of the FlexCore processor, optimized for a number of applications from the embedded benchmark suites. We plan to tape out this first test chip in June 2010. In the future, we expect to explore more design alternatives and also demonstrate more advanced aspects of the FlexTools chain.

- Integrated platform to take an application dependent design from concept to implementation.

- Allows adaptation of the FlexCore processor by usage of accelerators that can enhance computation capabilities and pruning of the interconnect to reduce area and power.

- Compiler reschedules the applications for the modified FlexCore to harvest maximum benefits.

- Enables evaluation of different performance criteria at different development phases to meet the design goals.

- Estimates in early phases are coarse and runtime is low, while accuracy and runtime increase in later phases.

- Cadence tools operate on the RTL code produced by Flex-Tools to complete the flow from specification to tape out.