Kai Huang, Xiao-xu Zhang, Si-wen Xiu, Dan-dan Zheng, Min Yu, De Ma, Kai Huang, Gang Chen, Xiao-lang Yan, 2015. Profiling and annotation combined method for multimedia application specific MPSoC performance estimation. *Frontiers of Information Technology & Electronic Engineering*, **16**(2):135-151. [doi:10.1631/FITEE.1400239]

Profiling and annotation combined method for multimedia application specific MPSoC performance estimation

Key words: MPSoC, Gradual refinement, Native simulation, Performance estimation, Profiling, Annotation, Gcov

Corresponding author: Si-wen Xiu
E-mail: siwen_xiu@126.com
ORCID: http://orcid.org/0000-0003-0400-8037

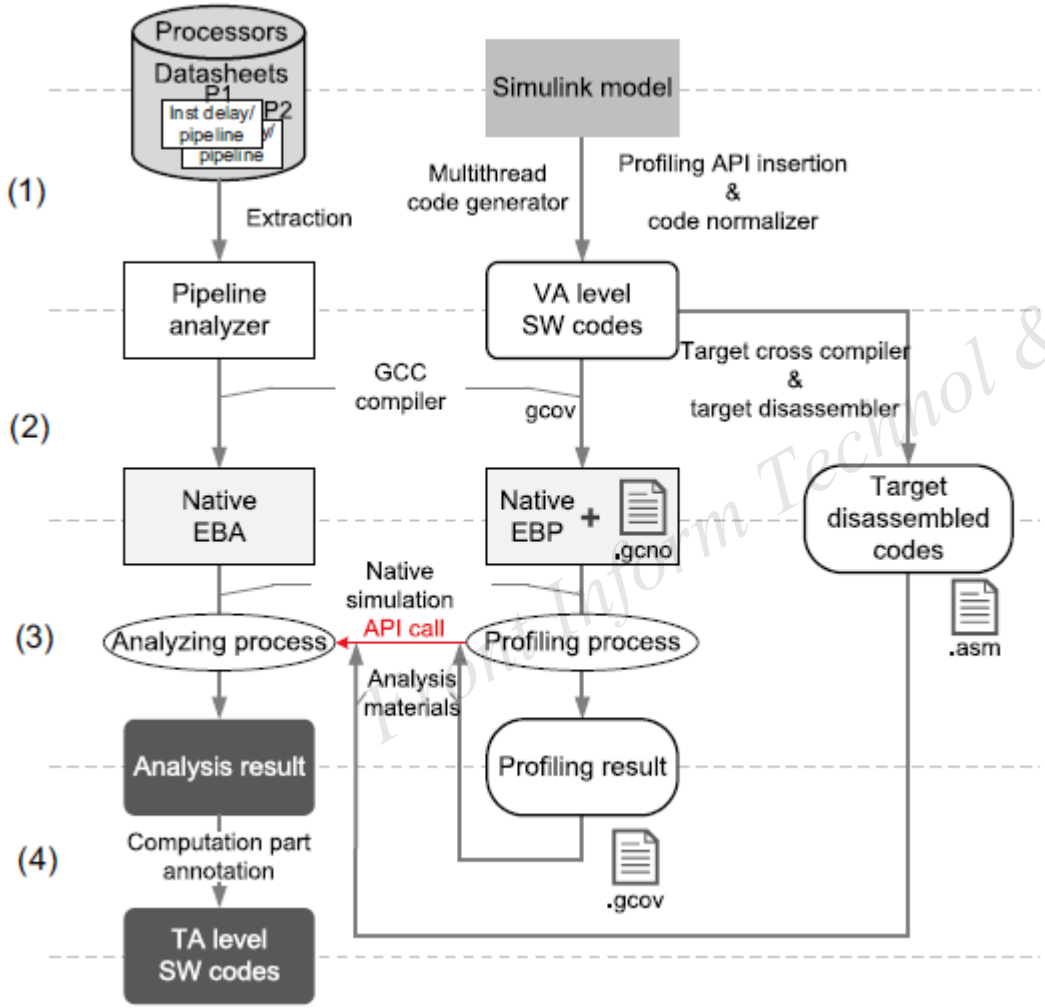
Motivations

- Accurate and fast performance estimation is necessary to drive design space exploration and thus support important design decisions.
- Fast performance estimation is especially important in the earlier stage of the design flow in order to fit tight time-to-market.
- Current performance analysis is expensive and cumbersome, thus difficult to use in any system design flow feedback loops.
- The attractive points of performance analysis in different abstraction levels are different. So, performance analysis should be gradually refined.
- Computation and communication should be evaluated separately according to the design orthogonalization.
- According to some experiments, simulation with VA models can achieve 100× the speed of the TA model with only a 15% accuracy loss.

Contributions

- A profiling and annotation combined flow for multimedia MPSoC performance estimation from the VA level to TA level.
- A profiling based method for VA-level native simulation.
  - GNU gcov is employed to profile the execution statistics of the given C code during native simulation.
  - Run-time analysis.
  - Accurate and fast.
- An annotation based simulation method for TA-level transaction accurate simulation.
  - Only the communication latency is refined.
  - More efficient.

Work flow



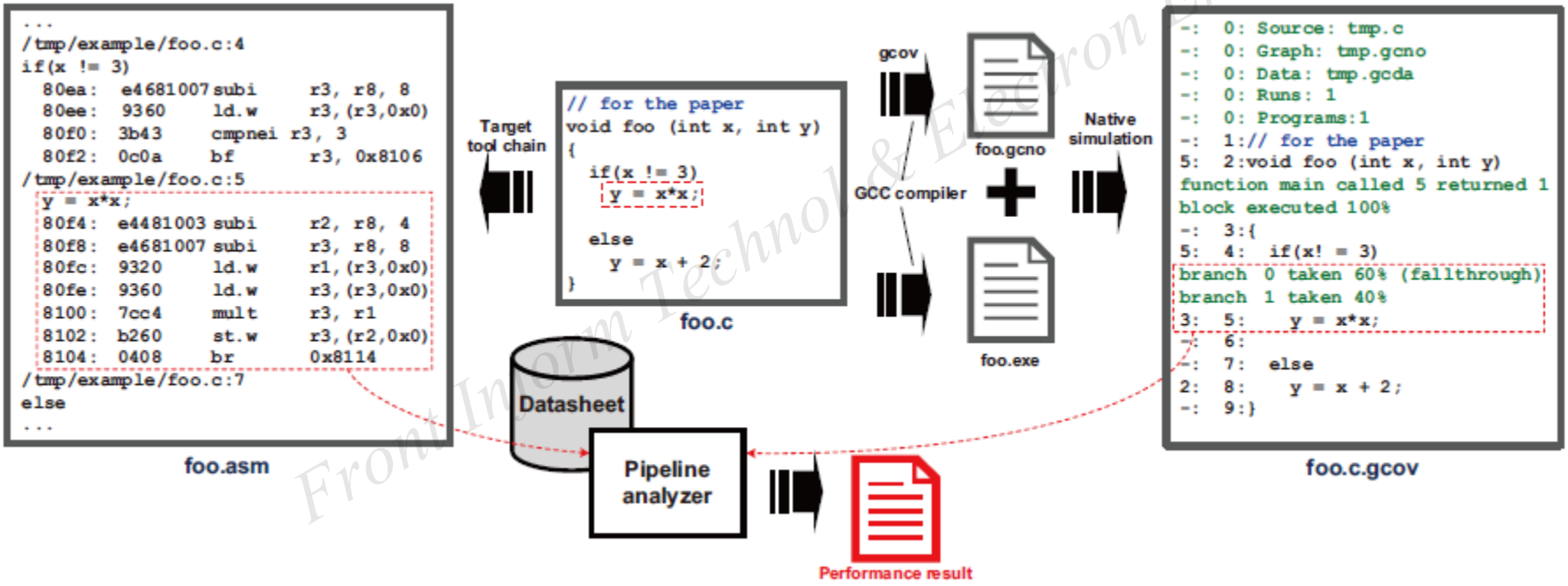
(1) Profiling API insertion

(2) Double compilation

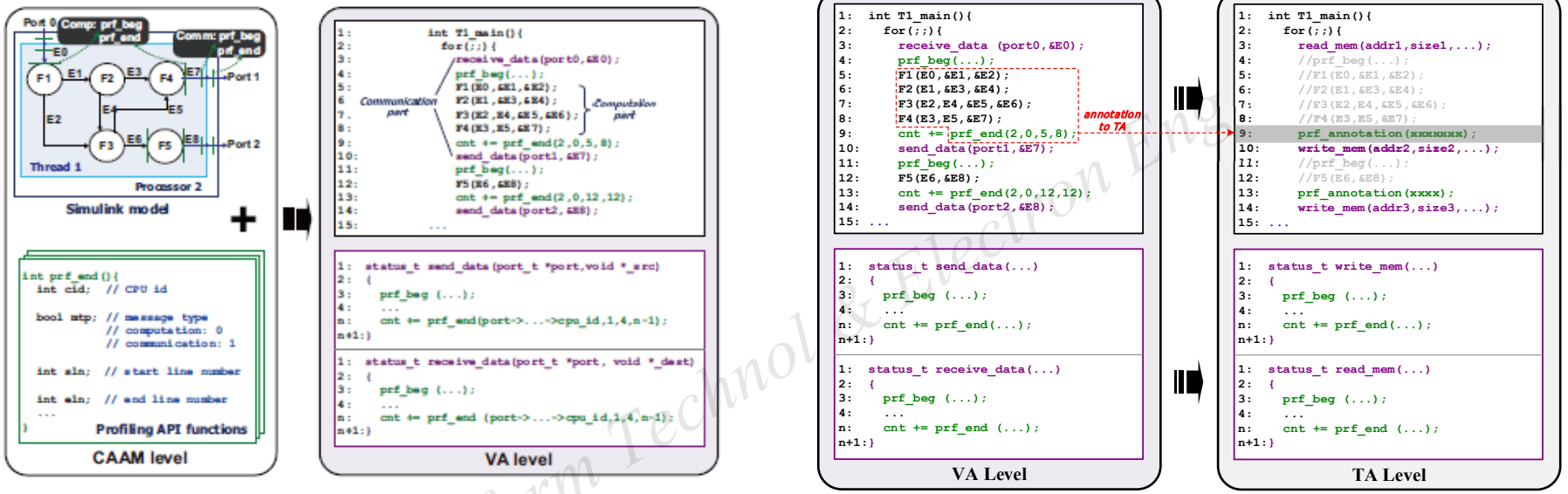
(3) Run-time performance analysis

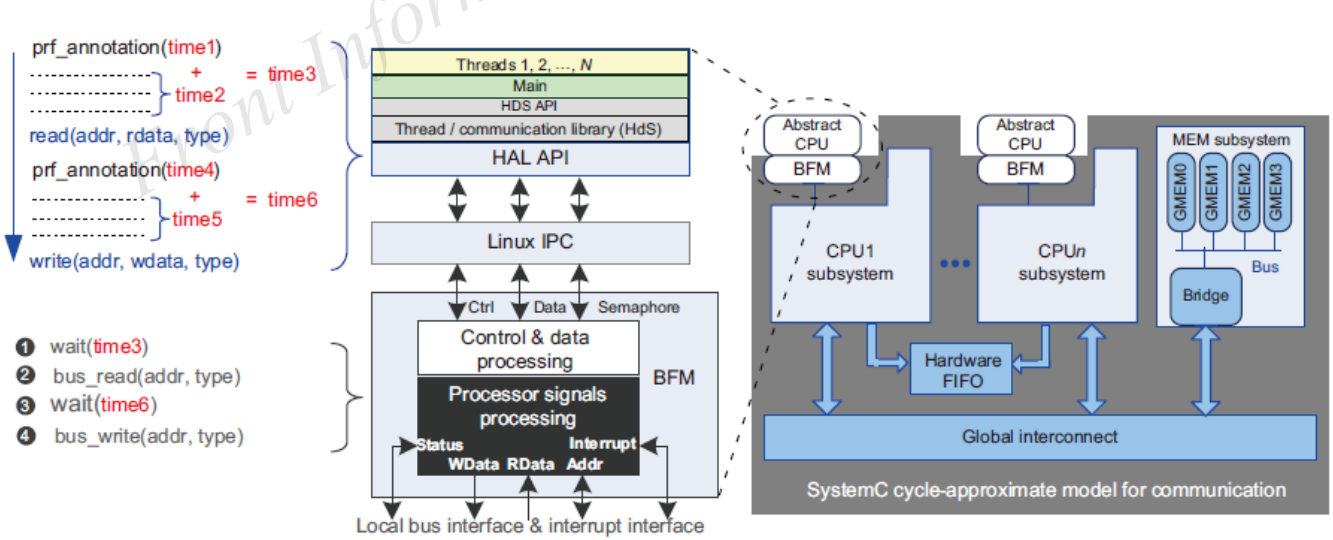
(4) Result annotation

GNU gcov based profiling and analyzing

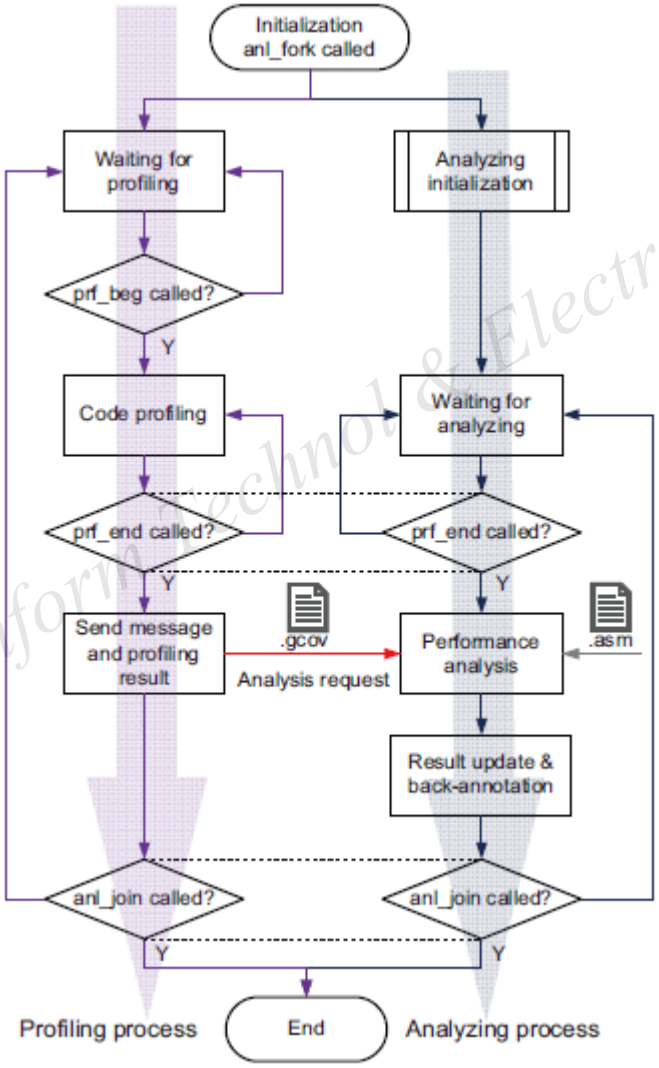


Profiling API insertion and annotation

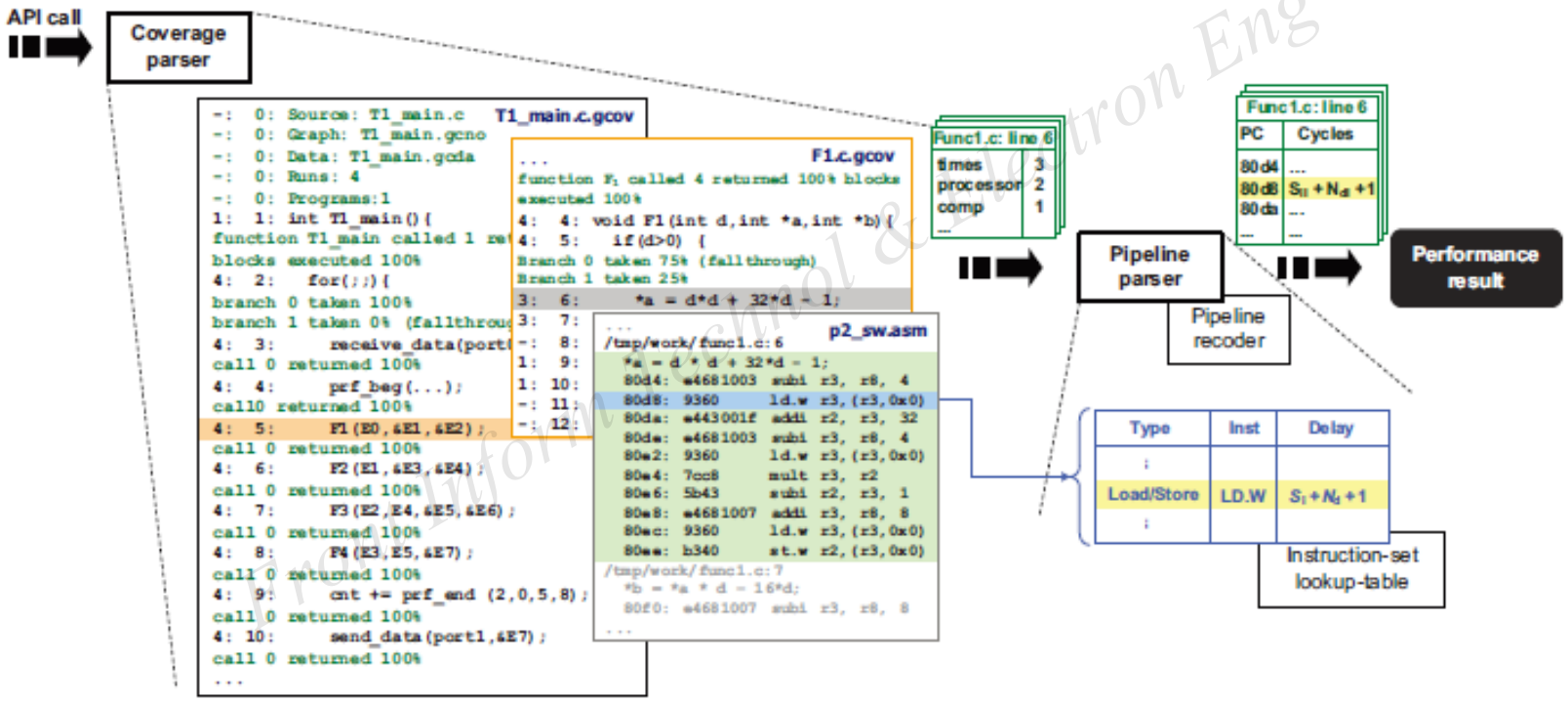




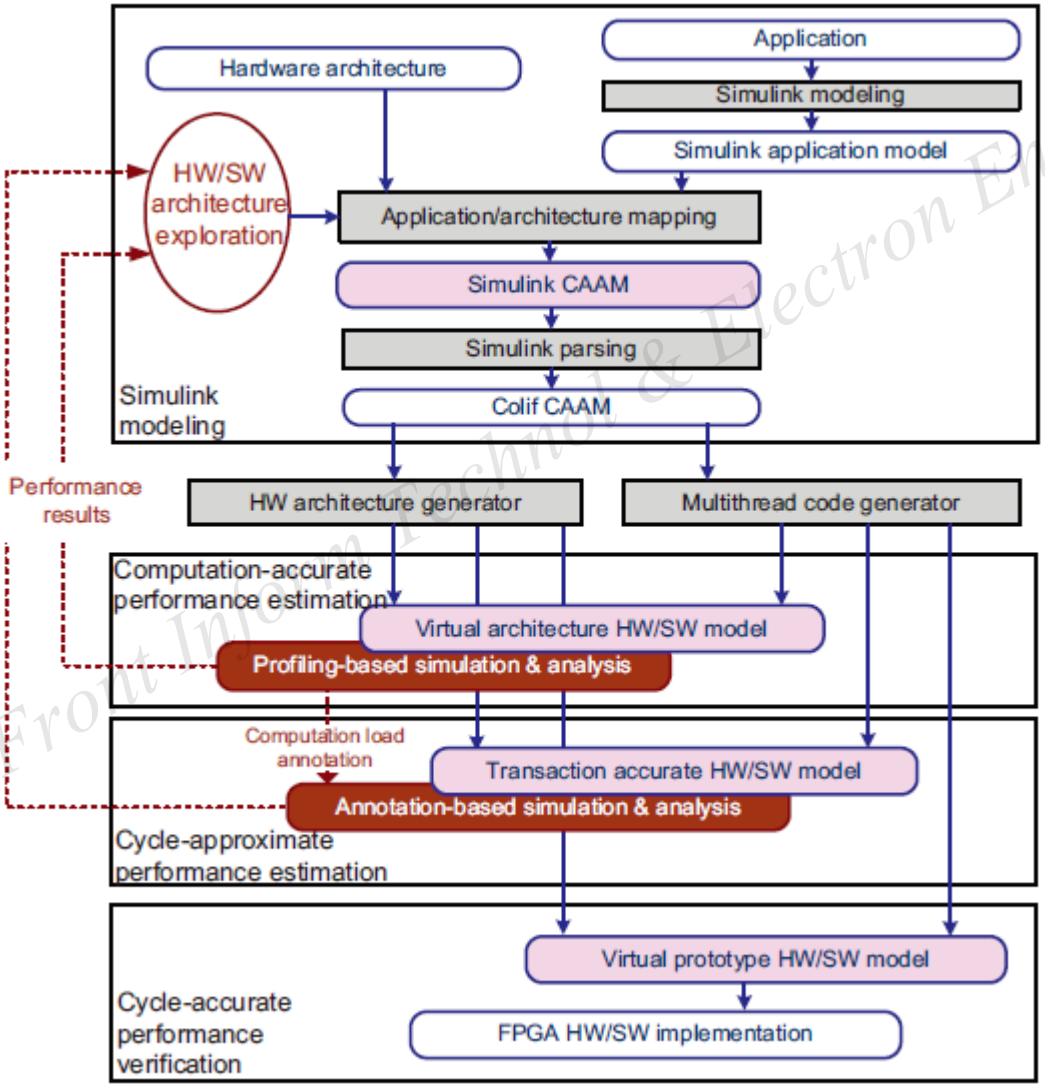
Run-time analysis



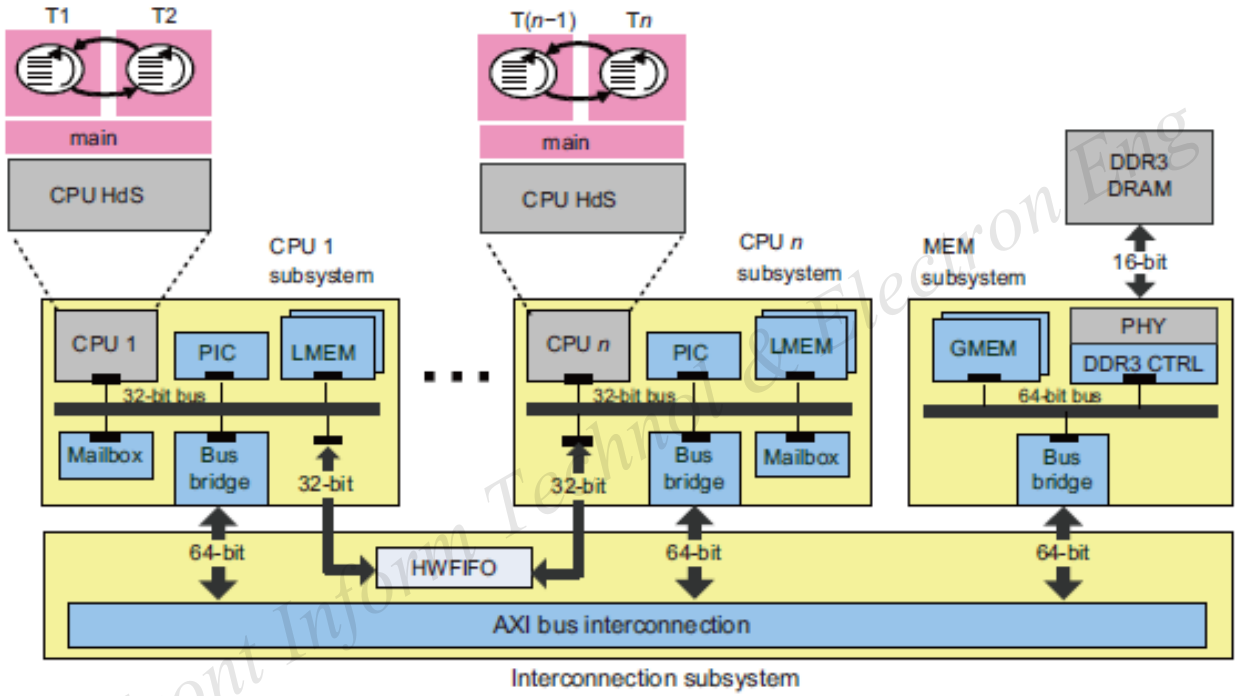
Pipeline analysis



Design and estimation framework



Experiment setup



Estimation scheme	Scheme	
	Early large-scale coarse-grained	Late small-scale fine-grained
TA	TA	TA
VA	VP	VP
TA+VP	TA	VP
VA+TA	VA+Profiling-based technique	TA+Annotation-based technique

Experimental results: speed

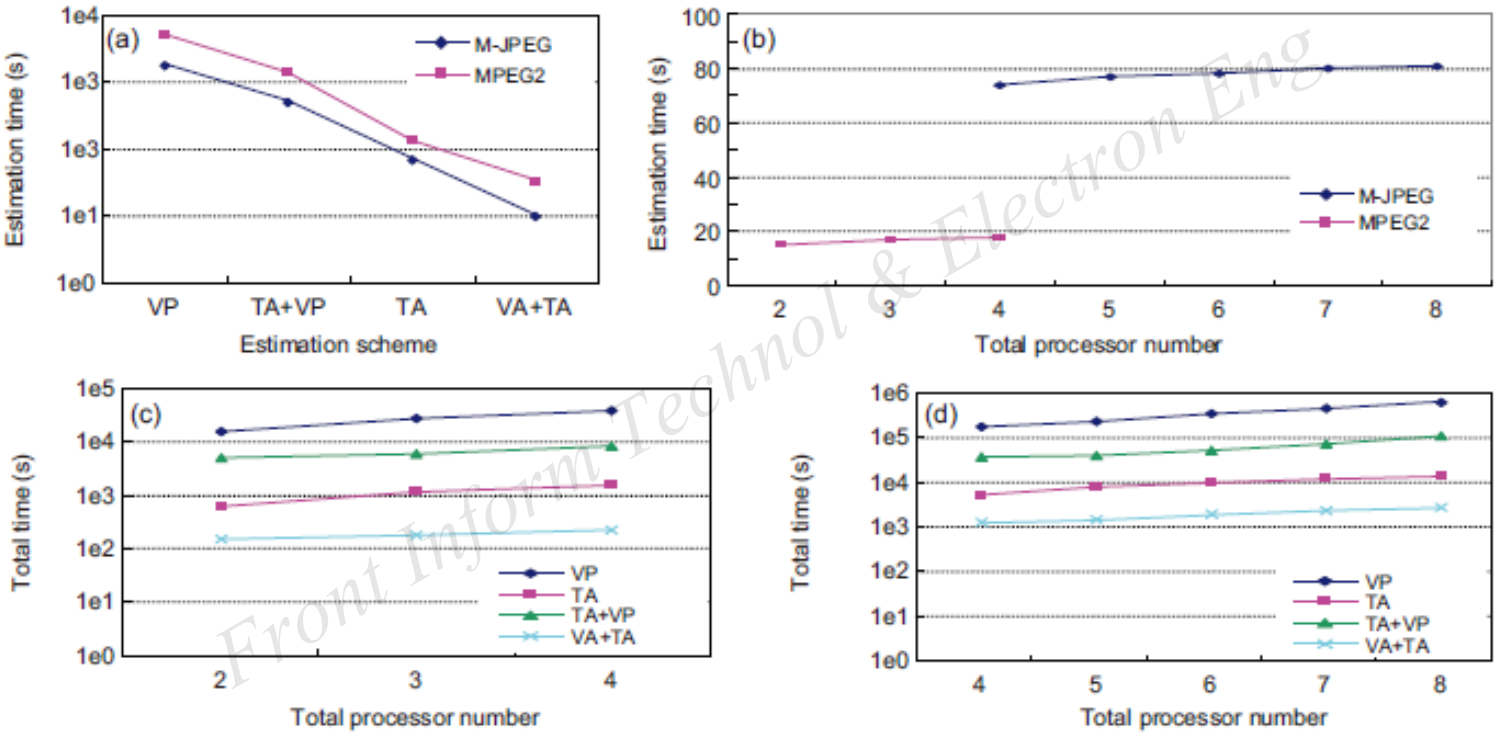


Fig. 14 Experimental results on estimation time: (a) estimation time of four different schemes for 10-frame M-JPEG and MPEG2 decoding; (b) estimation time of the VA+TA scheme given different numbers of processors in M-JPEG and MPEG2 decoder applications; (c) total time for four different schemes for MPSoC architecture exploration given 2, 3, and 4 processors in the M-JPEG decoder application; (d) total time for four different schemes for MPSoC architecture exploration given 4 to 8 processors in the MPEG2 decoder application

Experimental results: accuracy

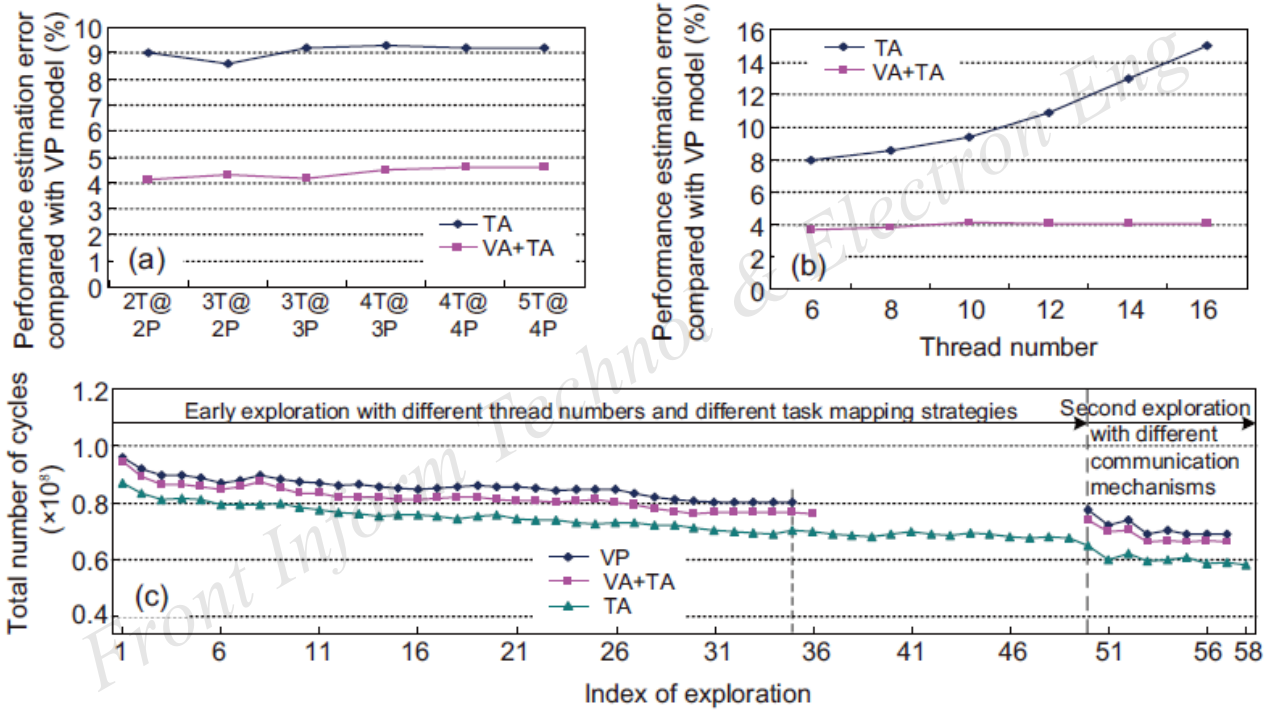


Fig. 15 Experimental results on estimation accuracy: (a) estimation error of TA and VA+TA with different numbers of processors and threads for the M-JPEG decoder application; (b) estimation error of TA and VA+TA given a 6-processor platform with 6 to 16 threads for the MPEG2 decoder application; (c) MPEG2 decoder architecture exploration on the 6-processor platform taking advantage of VP, VA+TA, and TA, respectively