Laboratory for High Performance Computing

Cache Placement & Replacement Policies:
- Adaptive cache placement to reduce conflict misses.
- Heterogeneously Segmented Cache architectures for packet forwarding.
- Two-level mapping based placement for packet forwarding.
- Emulating the optimal replacement policy with a shepherd cache.

Prefetching:
- Performance Aware Prefetching.
- Identifying & Eliminating misses that cause pipeline stalls.

Power Performance Efficient Arch.
- Segmented Low Power Issue Queues.
- Scalable Energy Efficient Store Queues.
- Heterogeneous Width Register Files for Superscalar Architectures.

Programming Models:
- Ease Programmability of Heterogenous and Accelerator-based Architectures.
- IR-based framework for programming Accelerator Based Architectures.

- Design Space Exploration of Network Processors.
- Packet Reordering in Network Processors.
- Petri Net Models to Evaluate Packet Buffering in NPs.
- Distributed Packet Buffering & Dynamic Buffering Schemes for Network Processors.
- Improving Performance of Table Lookup Operations.
- Framework for Application Design Space Exploration.

Memory Arch. Exploration:
- Evolutionary Approach for Memory Architecture Exploration
- Integration of Architecture and Data Layout Exploration.
- Pareto Optimal Design Points for Power, Performance and Cost.
- Hybrid Memory Architectures Involving Cache & Scratchpads.

Compiler Optimizations:
- Code Size Aware Instruction Scheduling.
- Energy Aware Compilation techniques, exploiting DVFS.
- Energy Aware instruction Scheduling and S/W Pipelining.
- Instruction Sequencing to Improve Energy Efficiency.

Memory Hierarchy
General Purpose
Micro-architecture
Network Processors
Application Specific
Embedded Systems
ARCHITECTURE

Programmability
Performance
Power Efficiency

HPC
COMPILERS

Students: Graduated (Current)

ME/MTech	:	12 (1)
MSc(Engg)	:	15 (1)
PhD	:	3 (5)

Over 60 Publications
1 Best Thesis Award
2 Best Paper Award

- Multithreaded Architectures
  - Design and Performance Evaluation of Multithreaded Architectures.
- Distributed Shared Memory Architecture
  - Compiler and Software Assisted DSM.
- Clusters
  - Cluster Based Web Servers.
  - Cache Performance for Web Server Architectures.
  - Exploiting Communication Processors for Improving Application Performance
- Applications
  - Multi-channel Speech Recognition on General purpose Multi-Core and Accelerator Architectures.

Heterogenous High Performance Clusters:
- Synergistic Use of General Purpose and Accelerator Architectures.
- Single-source Compiler for GPUs.

Stream Compilers:
- Stream Program Compiler for GPUs.
- Coarse Grained S/W Pipelined Execution of Stream Programs on GPUs and Accelerators.
- Dynamic Scheduling Framework for Stream Programs on Multicores.

Compiler Analysis:
- Comprehensive Path-Sensitive Data Flow Analysis Techniques.
- Pointer Analysis:
  - Scalable Context-Sensitive Points-To Analysis.
  - Reducing Memory Requirement for Storing Points-to Pairs.

Compiler Optimizations:
- Instruction Scheduling, Register Allocation.
- Resource Usage Models for Instruction scheduling and S/W Pipelining.
- Software Pipelining:
  - Nested Loop SWP
  - Energy Aware SWP
  - Register Sensitive SWP
  - Enhanced Coscheduling
- Vectorizing Compilers for MMX.

Research Funding From:
DST, IBM, Intel
Microsoft and NVIDIA

Contact Faculty
Prof. R. Govindarajan
Mailing Address
Lab. for HPC, SERC, IISc,
Bangalore 560012
Website
http://hpc.serc.iisc.ernet.in

100 YEARS