AHIR: from program to hardware

Sameer Sahasrabuddhe
Prof. Kavi Arya
Prof. Madhav P. Desai

Concept

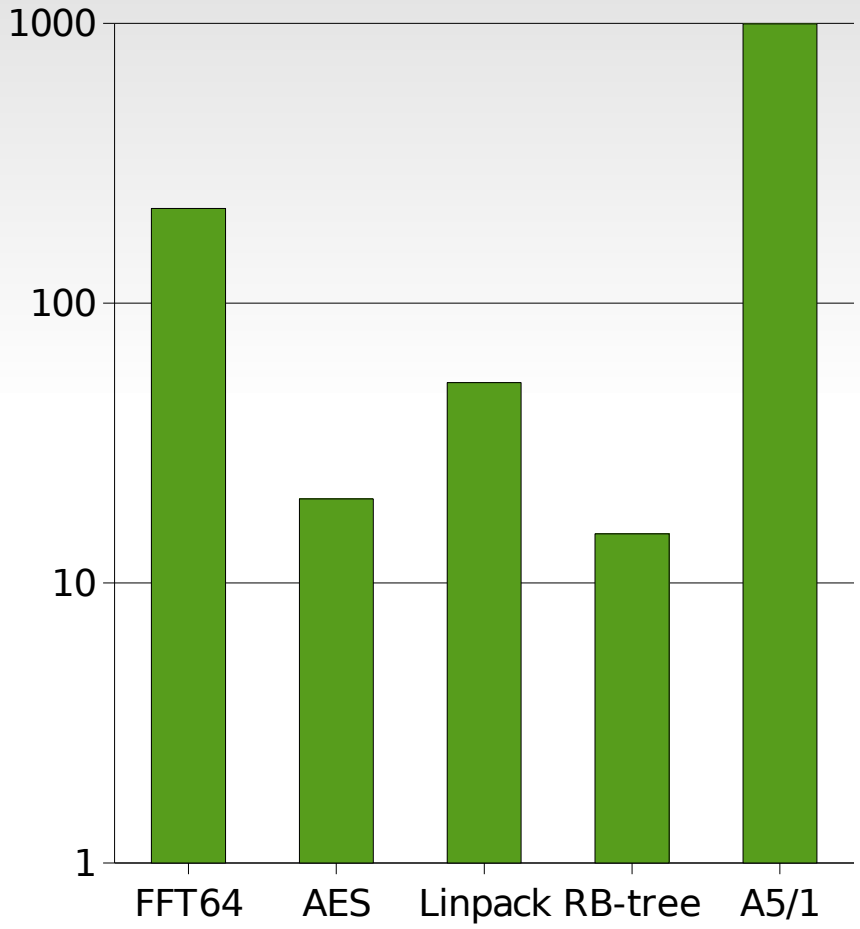
- Start with a program written in a “standard programming language” such as C.
  - Use software compiler to optimize program to extract parallelism etc.
- Transform the program into an intermediate representation (a virtual circuit)
  - AHIR
- Map the virtual circuit to an actual logic circuit
  - Optimize hardware by sharing resources etc.

Potential Value

- Replace use of processors in SoCs
  - Higher Performance/Watt (10X-1000X better relative to Pentium-4).
- Use as part of VLSI Design Flow
  - Fast path from algorithm specification to RTL.
  - Verification / Validation at the program level (specification) rather than the RTL level.
- Map algorithms to hardware for high-performance computing.

Normalized Performance/Watt

- Baseline: Pentium 4 (Northwood)
  - 130nm, 2.4GHz
  - Typical power: 70W
- AHIR circuits
  - 180nm TSMC (projected to 130nm)
  - Automatic flow using Synopsys DC and Cadence SoC Encounter



Data courtesy Kunal P. Ghosh and S. Sreenivas (MTech students, Dept. of EE)

Conclusion

- A complete flow from high-level programs to hardware
  - Currently demonstrated with C as starting point.
  - Scalable, applicable to complex software.
  - Energy efficient hardware relative to processor implementations.
  - Allows use of parallel programming languages as starting point for system level design.