- Into the Wild: Studying Real User Activity Patterns to Guide Power Optimizations for Mobile Architectures
  - Android G1
- SHARK: Architectural Support for Autonomic Protection Against Stealth by Rootkit Exploits
  - Modifying Linux kernel
- Cache Bursts: A New Approach for Eliminating Dead Blocks and Increasing Cache Efficiency
  - Improving the L1 and L2 cache performance
- Characterizing Flash Memory: Anomalies, Observations, and Applications
  - Performance, power, reliability for 5 vendors

Sequential Y86 Implementation

1. Fetch
   - From PC
   - icode (4 bits) & ifun (4 bits) [valC (4bytes)]
   - Calc valP
2. Decode: get rA [rB] [%esp]
3. Execute
   - ALU → valE
   - Conditions set
   - Memory calculations
   - Stack pointer update
4. Memory: valM
5. Write back: registers)
6. PC Update: Set via valP

Why so much processing?