Discussion 15

12/3/02 Tue

Summary of important topics in CS61C

- **C program**: you need to understand how C program works.
- **MIPS**: need to understand MIPS assembly program, especially procedure calling convention.
- **Memory allocation**: need to understand best fit, first fit and next fit.
- **Logic design**:
  - You are given a description. Then, you need to express it into a logic table. In case of sequential logic, you also need to express it in a state diagram.
  - It is straightforward to transform the logic table into logic gates and Verilog code.
- **Virtual memory**:
  - some analytic problems: tag, index and offset size. Total address space that can be represented or page size to represent an address space.
- **Interrupts**:
  - interrupts have priority.
  - User process < low priority interrupt < high priority interrupt
  - Lower priority process resume after higher priority process finishes.
- **Pipeline**: need to know when data and control hazards happen and find them in a sequence of code
- **Cache**:
  - AMAT = hit time + miss rate * miss penalty
  - Set associativity helps miss rate. Direct mapped cache helps faster hit time. Larger block size causes higher miss penalty but helps lower miss rate.
  - Various replacement policy: Random, LRU and MRU
  - Split cache / unified cache: split cache has separate cache for instruction and data cache. It increases bandwidth.
  - Unified cache: inst and data share the memory space. Can use memory space economically.
- **Hard disk**.
  - Factors of performance: seek time + rotational delay + transfer time + controller overhead
- **CPU organization & performance**
  - CPU time = CPI * Number of instructions * clock cycle time = CPI * Number of Instructions / clock rate
  - PerfA/PerfB = CPU timeB / CPU timeA

Back to Top

Revised: .