Topics in computer architecture

SPARC in ECL

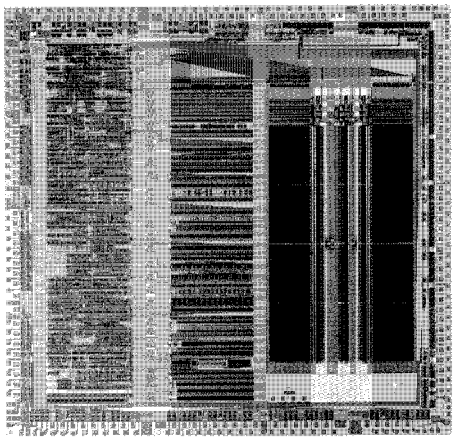
P.J. Drongowski
SandSoftwareSound.net

SPARC in ECL

- "Implementing SPARC in ECL," E.W. Brown, et al., IEEE Micro, February 1990, pg. 10 -22.
- Joint project with Bipolar Integrated Technology (BIT)
- Design goals
  - Product
    - Clock cycle time of 12.5 nanoseconds (80 MHz)
    - Benchmark performance of 60 MIPS, 12 MFLOPS
    - $100,000 entry level price
    - One conventional circuit card
    - Air cooling with pin grid array and DIP packaging
    - Standard ECL 10K glue logic
  - IC Process
    - Emitter Coupled Logic (ECL)
    - Switching speed over low cost/power dissipation
    - Lower density than CMOS due to large bipolar Q's
    - Gates biased to drive long, capacitive signal lines
  - BIT process
    - Unloaded gate delay of 375 picoseconds
    - Three layers of interconnect metalization
    - Power
      - Distributed on all three layers
      - Copper-tungsten slug with high conductivity
      - Die bonds directly to slug
      - Slug transfers heat to top of package
      - Heat sink in forced air dissipates the heat
    - ECL inverter
      - Differential pair, load resistor, output driver
      - Transistors in parallel to input - OR
      - Differential pairs in series - AND
      - Series gating - (A+B)(C+D)(E+F)
      - Series gating useful for diagnostic scan chain
      - Multiplex and latch combine into one gate
  - 25-ohm transmission lines (match impedences)
  - ECL 10K has no temperature compensation
  - Must control temperature to avoid switching skew

B5000 microprocessor

- Custom, cell-based approach (600 unique cell types)
  - 375 x 387 mil
  - 122,000 transistors and 77,000 resistors
  - 279-pin grid array package
- Internal datapath is 32-bits wide
- Register file
  - Three port RAM (two read, one write)
  - 4,352 bits (70,000 transistors)
  - 10 nanosecond read access time
- Input / output connections
  - 213 signal wires
  - 87 power bond wires
- Parity
  - Parity checking on each incoming byte
  - Two parity bits for each register
  - Synchronous trap on parity error
- Scan mode
  - Most registers can be connected into one shift register
  - Entire shift register can serially loaded and read
  - Essential for testing, debugging and diagnosis
- Redundancy
  - Redundant block of eight general purpose registers
  - Substitute for one of 14 blocks of window registers
  - Enhances yield
  - IC tester determines the bad block
  - Firmware routine performs substitution in system



Cache design considerations

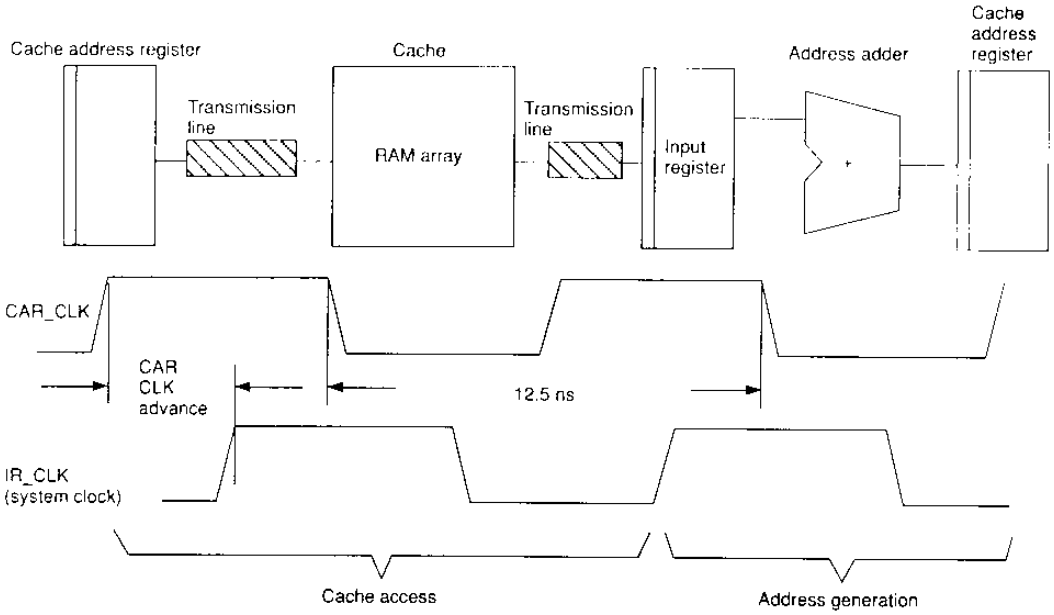
- Set associative (rejected)
  - Set elements must be multiplexed
  - Multiplexing will increase cache cycle time
- Non-pipelined, two-cycle access cache (rejected)
  - Branches must be two-cycle instructions
  - Performance would be slowed by 15 percent
- Separate instruction and data caches (rejected)
  - Each cache requires its own 32-bit port
  - Lack of 64-bit paths increases cache miss penalty
  - Double precision performance also affected
- Direct-map, write-back cache (chosen approach)
  - 72-bit paths (including parity)
  - One cycle access
  - Combined instruction and data cache
  - Fetch two instructions per cycle
  - Achieves 80 to 90% performance of separate caches
  - Static RAM (SRAM) technology
    - Composition of 16K x 4 and 4K x 4 bit RAM's
    - Write cycle is longer than read access time
    - Address/data must bbe set-up before write pulse

Electrical and physical considerations

- Signal reflection and crosstalk analyzed and minimized
- Treat signal lines as transmission lines
  - Driven from one end of wire
  - Daisy chain wire from receiver to receiver
  - Terminate with matched DC impedance
- Differential transimission used only on critical clocks
- Crosstalk reduction
  - Place power planes between signal planes
  - Interleave vertical and horizontal signal layers

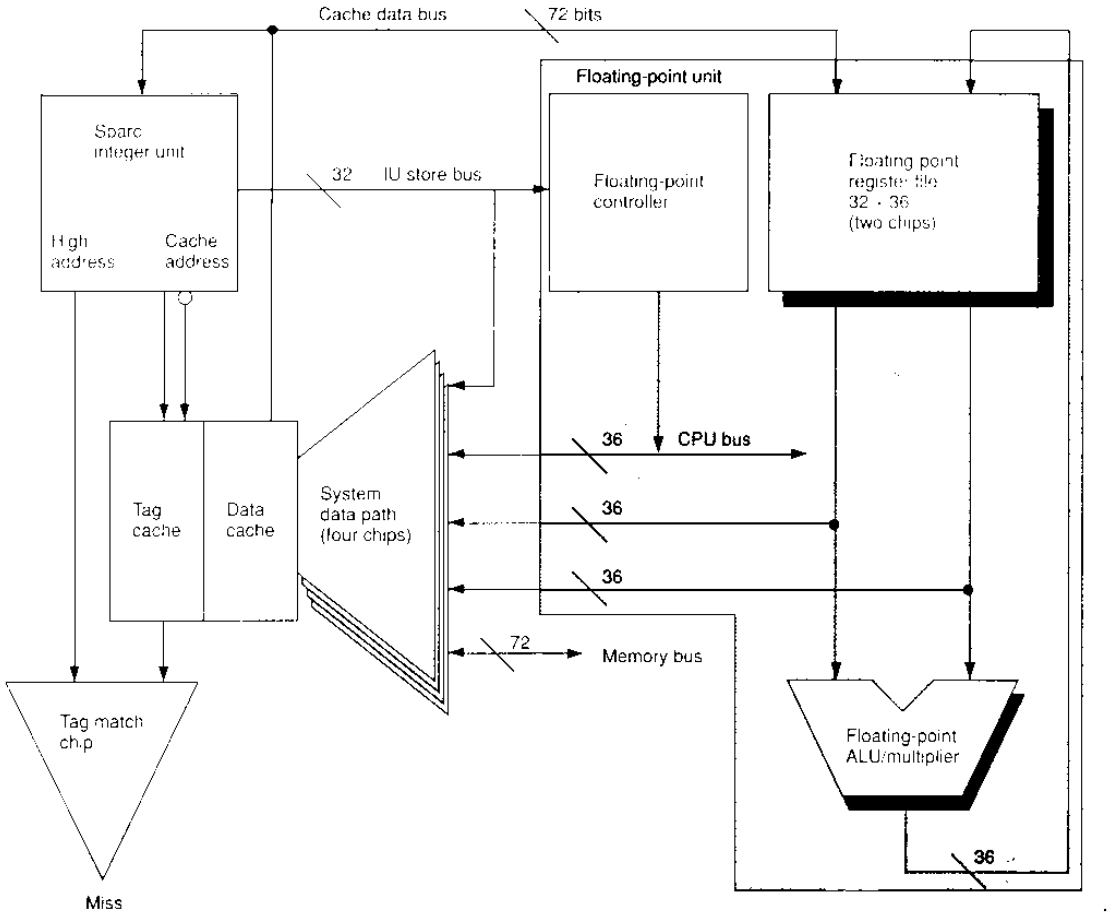
Cache design

- Driving addresses into the array
  - Loading on address is pin is 5 to 7 picofarads
  - AC impedance of address bus is about 25 ohms
  - Delay on bus transmission line is 10 nsec
  - Split cache into two banks to reduce delay to 5 nsec
  - IU has internal 25 ohms differential drivers
  - Further splitting would require external drivers
- Speeding tag-access and comparison
  - Tag RAM's are 4 times smaller than data RAM's
  - Thus, tag RAM has a shorter access time
  - Developed dedicated tag match chip (gate array)
  - Place tag RAM's to receive IU address first
- SRAM cache timing
  - 80MHz SRAM unavailable, too expensive, too small
  - Change pipeline to allow extra time in cache access
  - Borrow time from faster address generation stage
  - Generate early cache address clock
  - Transmission lines and SRAM provide hold time
- Store operation
  - Tag check (1 cycle) followed by actual write (2 cycles)
  - If next instruction is non-memory, overlap last cycle



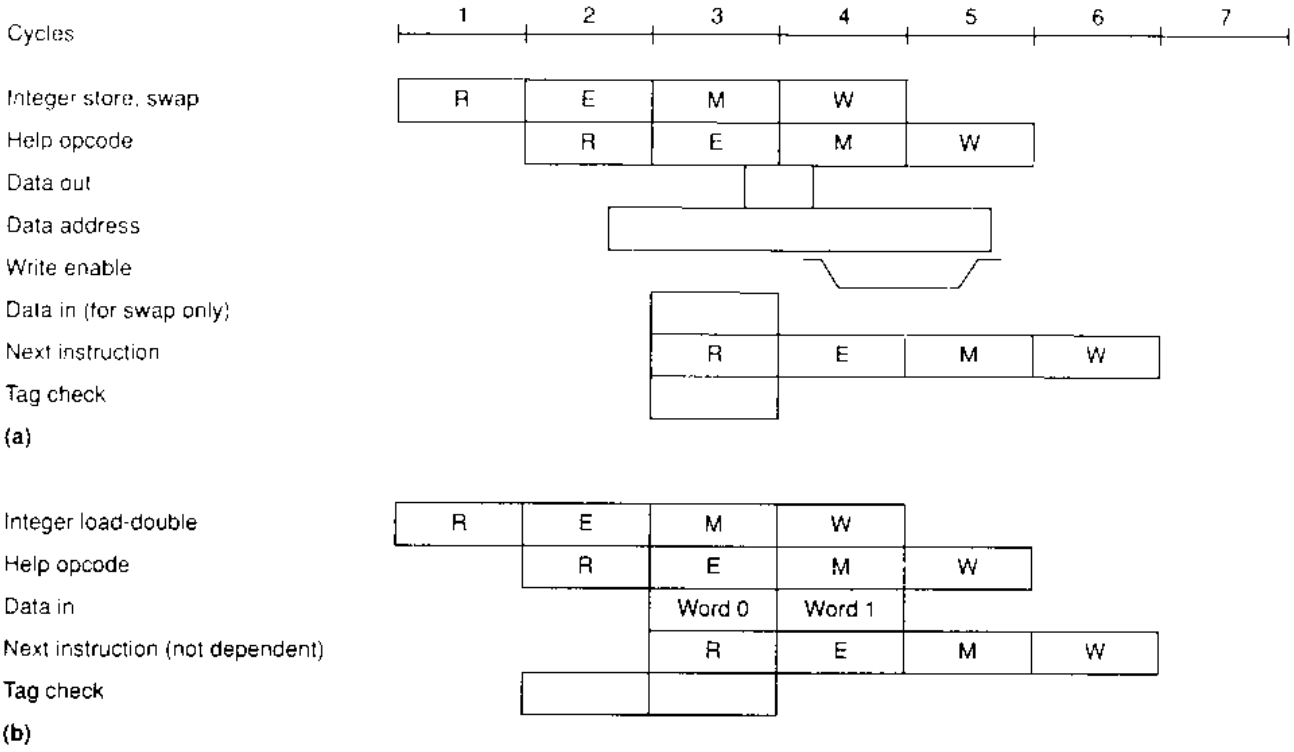
CPU core

- SPARC Integer Unit (IU)
- Floating Point Unit (FPU)
  - Five chips total
  - Controller
  - Two register file chips
  - Double precision ALU
  - Double precision multiplier
- Cache RAM array
- Tag match chip
  - Cache miss logic
  - Tag portion of four-entry translation lookaside buffer
- System datatpath
  - Four gate arrays
  - Contains data portion of TLB
  - Provides interconnection between major units
  - Contains memory bus interface



IU pipeline

- Five pipeline stages
  - Fetch (F) - move instruction from memory to IU
  - Read (R)
    - Read operands from register file
    - Decode op-code
    - Detect instruction dependencies
  - Execute (E) - perform ALU or shift operation
  - Memory (M)
    - Fetch data operand from memory (load)
    - Move arithmetic result to write port of register file
  - Write (W)
    - Write memory data into register file (load)
    - Write ALU result into file
- Data dependencies
  - Register forwarding
  - Execute load in one cycle when dependency is absent
  - Hardware detects load dependency: takes two cycles



Instruction queue

- 64-bit data input bus
- Fetch two 32-bit instructions in parallel
- Store one or both into four (max) instruction queue
- Load/store instructions must use memory ports
- Fetch from queue permits pipe to proceed after load/store
- Queue depth
  - Determines amount of load/store traffic without penalty
  - Depth of 4 permits three back-to-back loads
  - Greater queue depth not practical
    - Branches occur as frequently as every six cycles
    - Chip area budget did not permit bigger queue
    - Problem analyzed through simulation

Conditional branching

- Effect of branch is delayed one cycle
- IU assumes branch will *always* be taken
- Always issue target address late in the Read stage
- If *not* taken, "fall through" instruction is usually in queue
- Case where instruction is absent complicates logic design

FPU/coprocessor interfaces

- FPU and coprocessor can operate concurrently
- Units maintain queue of pending instructions and PC's
- Queue is used to identify instruction causing exception
- FPU carries condition codes and validation bit to IU
- Permits IU to execute FP conditional branch instructions
- Validation bit forces IU to wait for valid condition codes

Performance

- 103,000 Dhrystones per second
- 1.29 cycles per instruction
- 62 million instructions per second
- 14 million FLOPS (inner-loop routine of LINPACK)