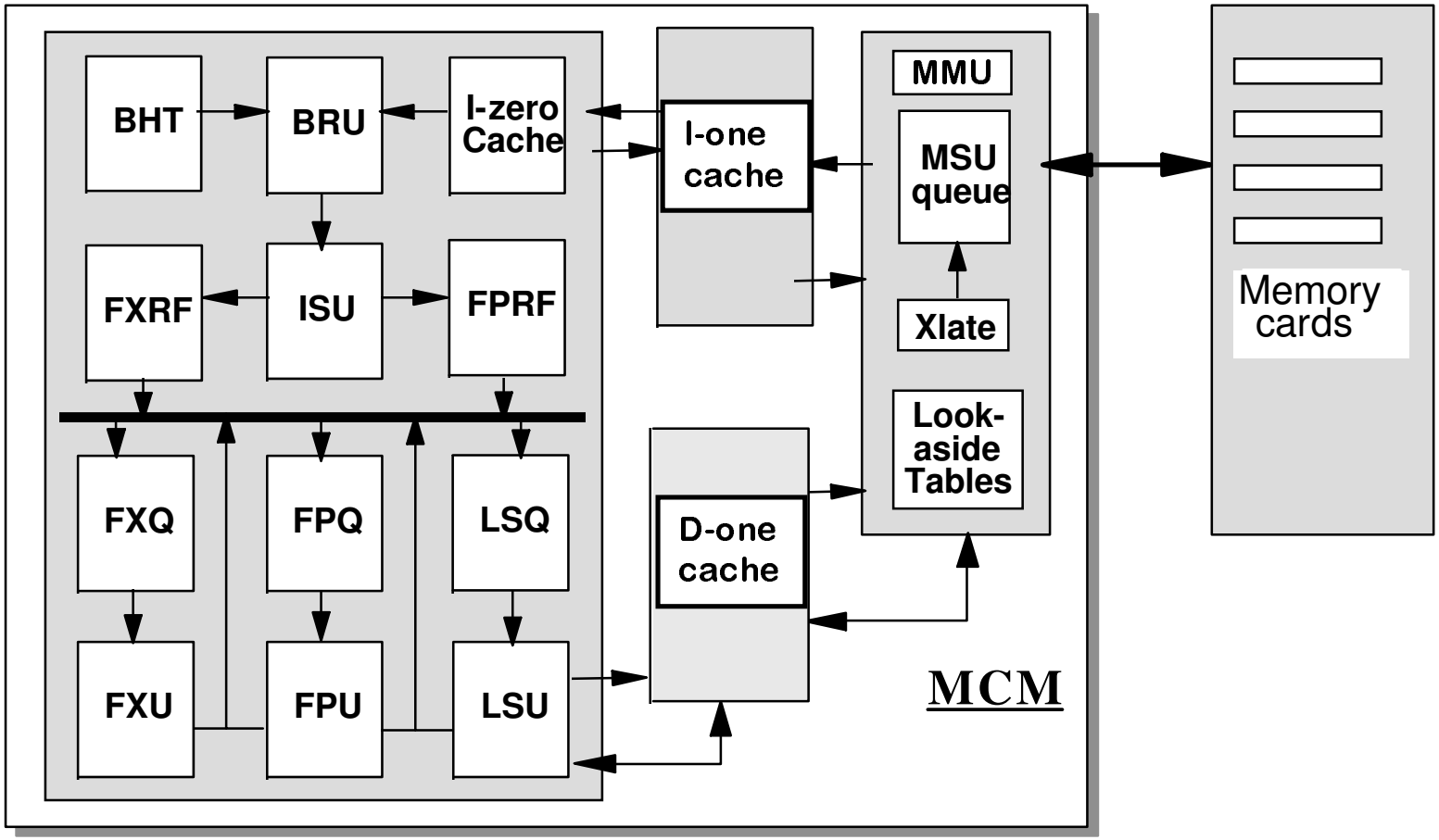
Processor Details



HAL COMPUTER SYSTEMS
A Fujitsu Company

Performance Estimates

Elapsed Time = CPI x Cycle time x Inst. count

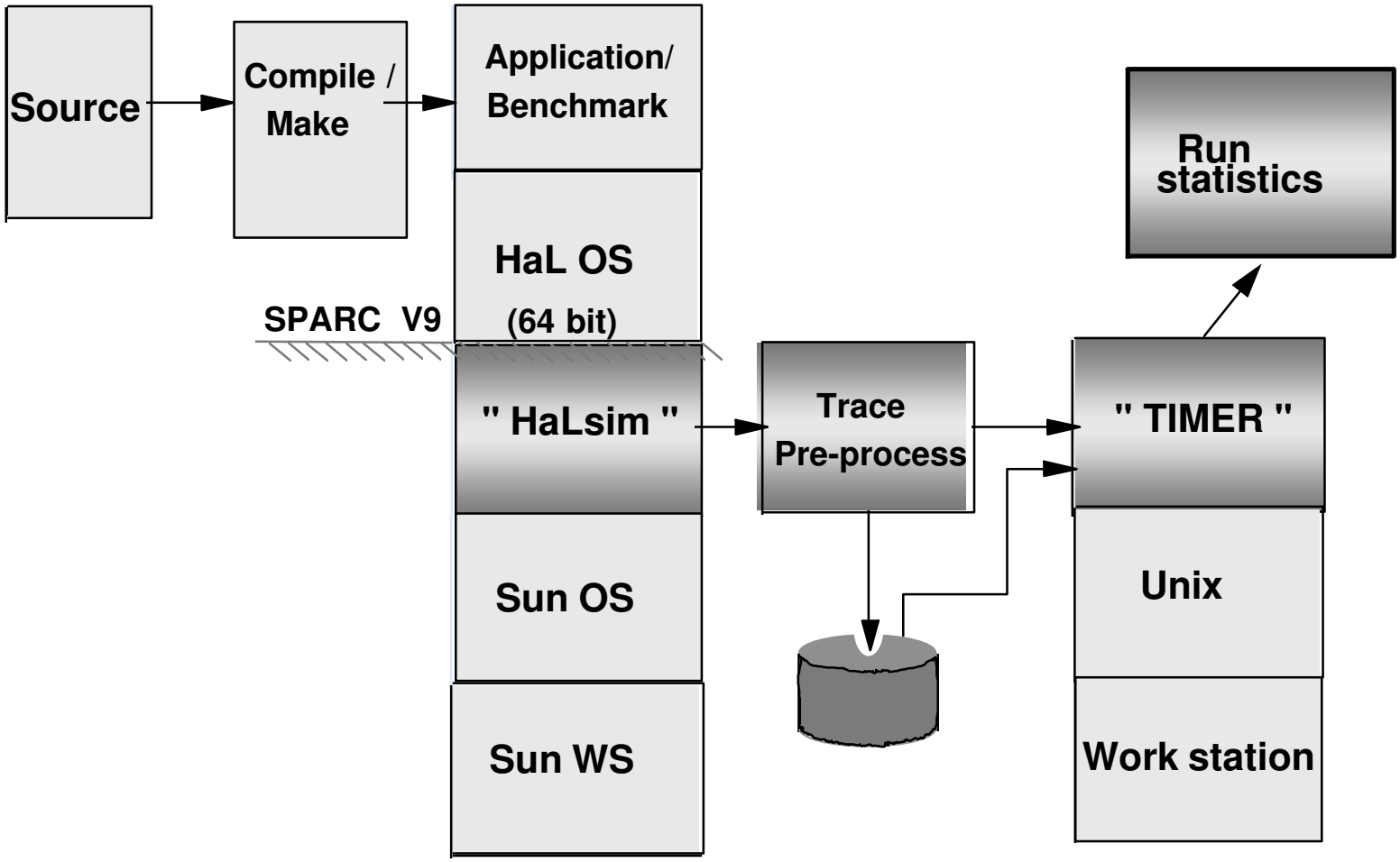
- **Cycle per Inst*: Predicted by the Timer software model**
  Input from HaLsim traces
  Statistical sampling techniques were used
- **Cycle time: Determined by technology and logic design**
- **Inst. count: Measured with applications run on HaLsim**

*Sometimes it's more convenient to use Instructions per Cycle: IPC = 1 / CPI

A Fujitsu Company

HAL COMPUTER SYSTEMS

Simulation Process



Timer is a Flexible Tool

A Fujitsu Company

HAL COMPUTER SYSTEMS

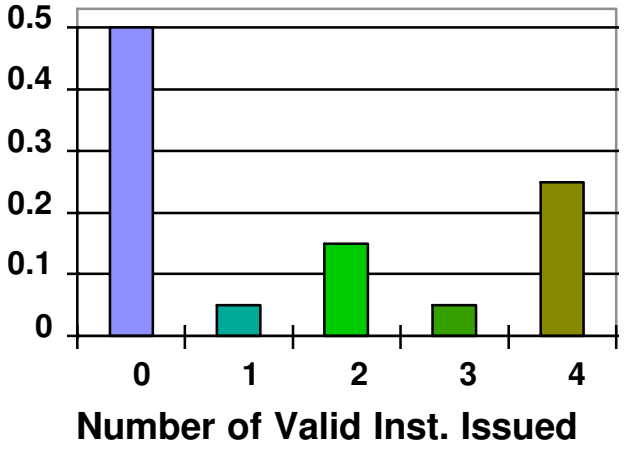
200+ Independent Options Handled as Switches in Timer

- Functional unit Latency, Number of, and Queue depth
- Cache sizes, Organizations and Levels
- Branch prediction, Fetch policies
- Issue rules, Peak Issue rate, and Instruction window size
- Translation and Memory latency and bandwidth (incl. TLB miss modeling)
- Magic Options (e.g. perfect caches, perfect branch predict, . . .)
- . . .

Options Not Easily Handled in Timer

- Speculative side effects
- I/O -- contention from DMA
- Major changes in pipeline stages

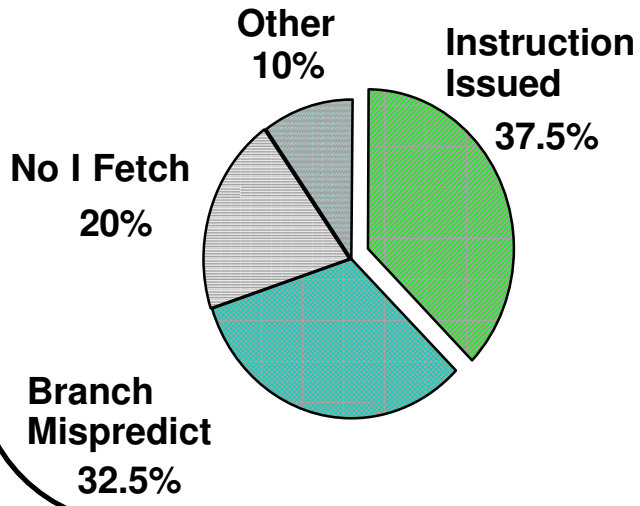
IPC Calculations



$$IPC = \sum_{n=0}^{4} P(n\text{-issue cycles}) \times n$$

For this example:

IPC =.5 (0) + .05(1) + .15(2) + .05(3) + .25(4)
IPC = 1.50



IPC = % inst. issued x opportunities/cycle

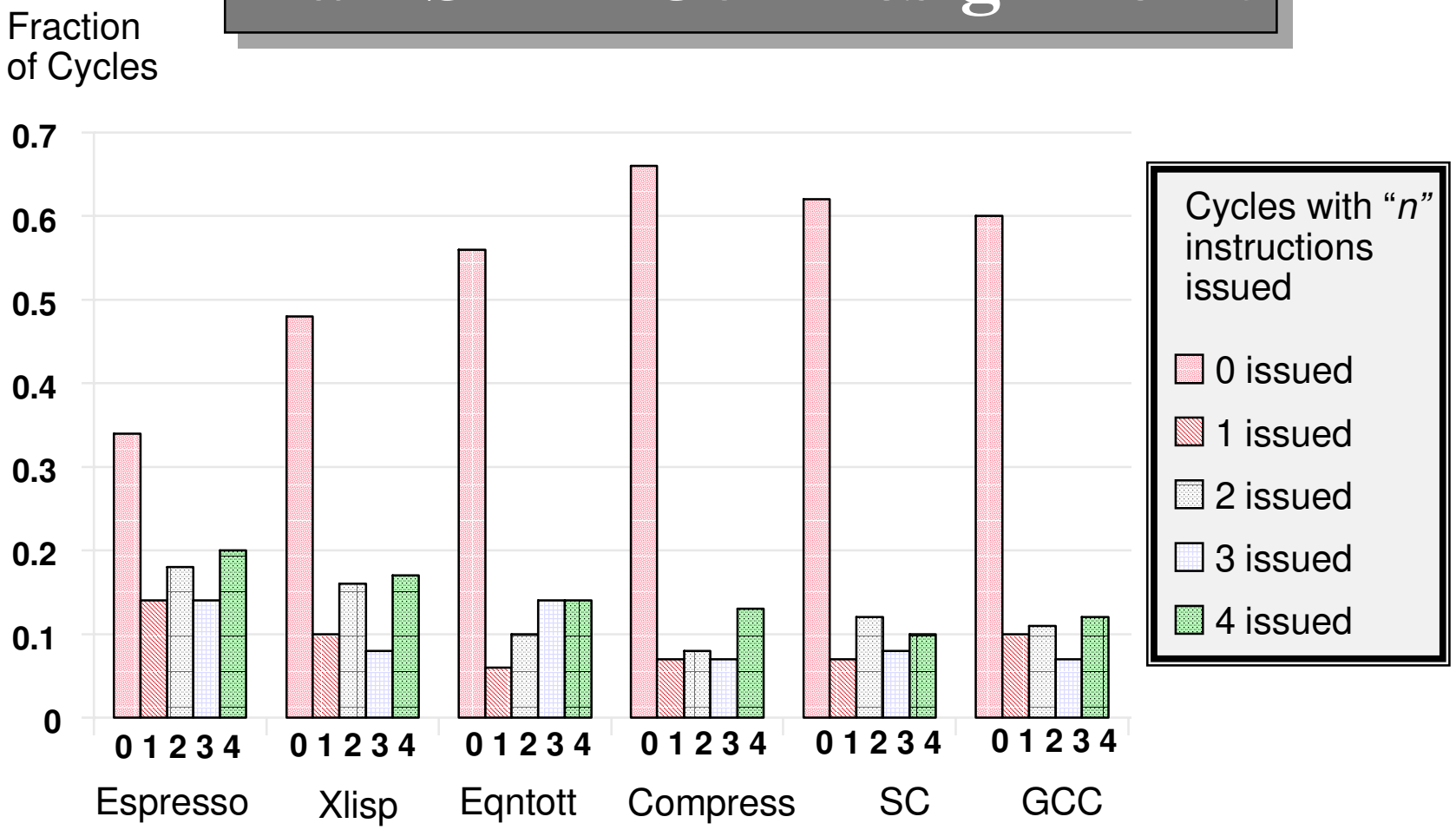
For this example:

IPC = 37.5% x 4
IPC = 1.50

A Fujitsu Company

COMPUTER SYSTEMS

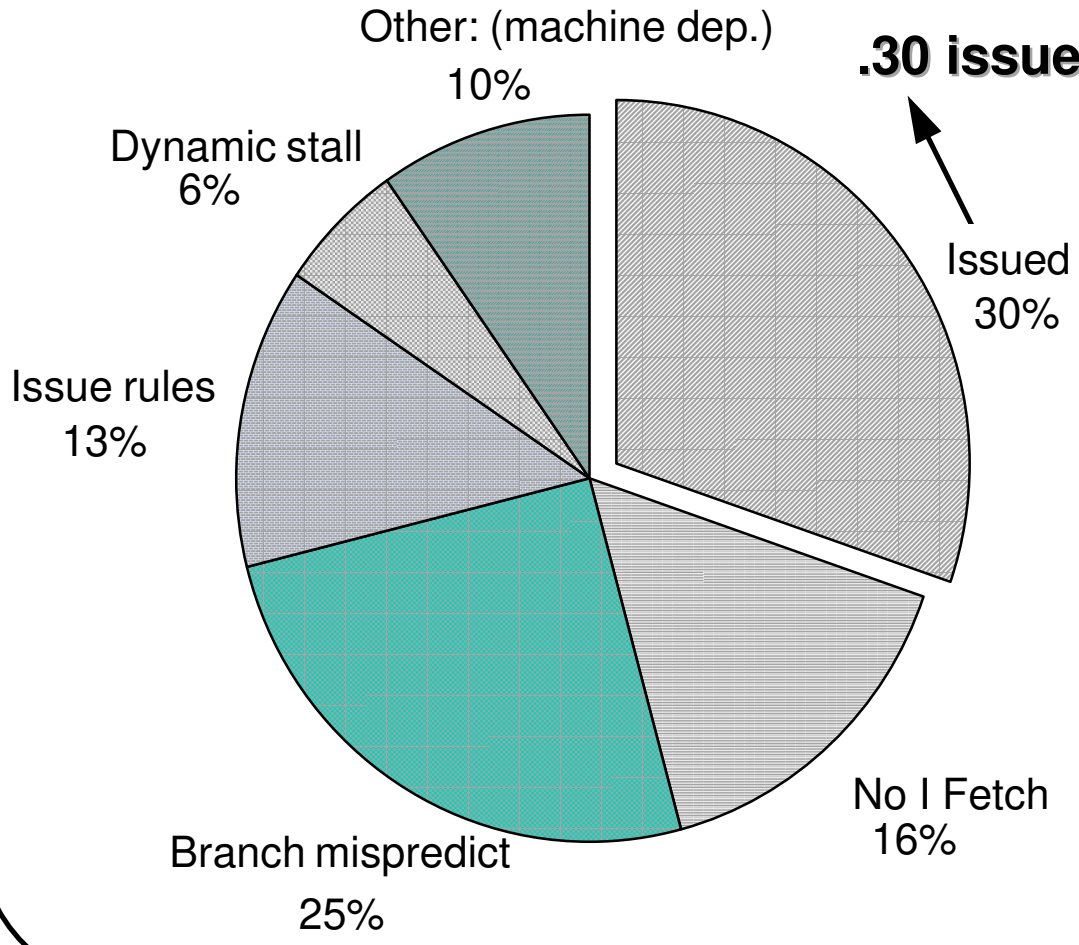
HaL SPARC64 Design Point



Benchmark results estimated with Timer

HAL COMPUTER SYSTEMS
A Fujitsu Company

HaL SPARC64 Design Point



.30 issued/slot x 4 slot/cyc = 1.2 IPC

Estimate for SPECint92
- **Avg. of 6 benchmarks**
- **Each benchmark equal wt.**

HAL COMPUTER SYSTEMS
A Fujitsu Company

"Back of the Envelope" Sanity Check

$$IPC^{-1} = CPI = .25 + 8 \times MR(Br) + 4 \times MR(I_0) + 28 \times MR(L_1)$$

4 issue/cyc	8 cyc/miss	4 cyc/ I-zero miss	28 cyc to DRAM
	.025 miss/inst	.015 miss/inst	.0056 miss/inst

Rough Estimate

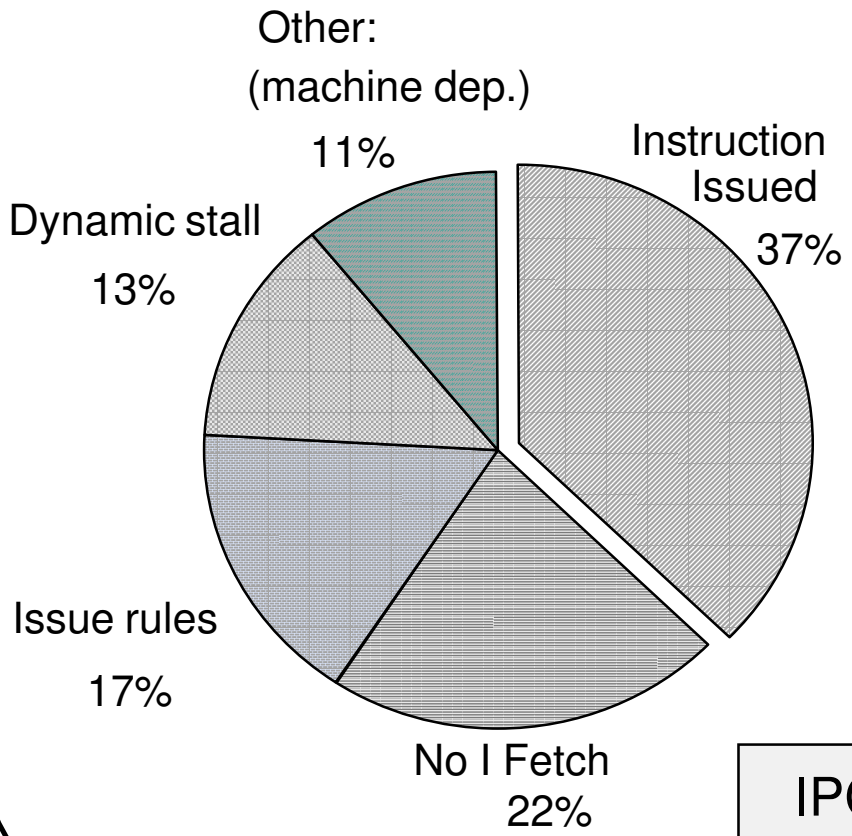
CPI = .25 +.42 = 0.67

or

IPC = 1.50

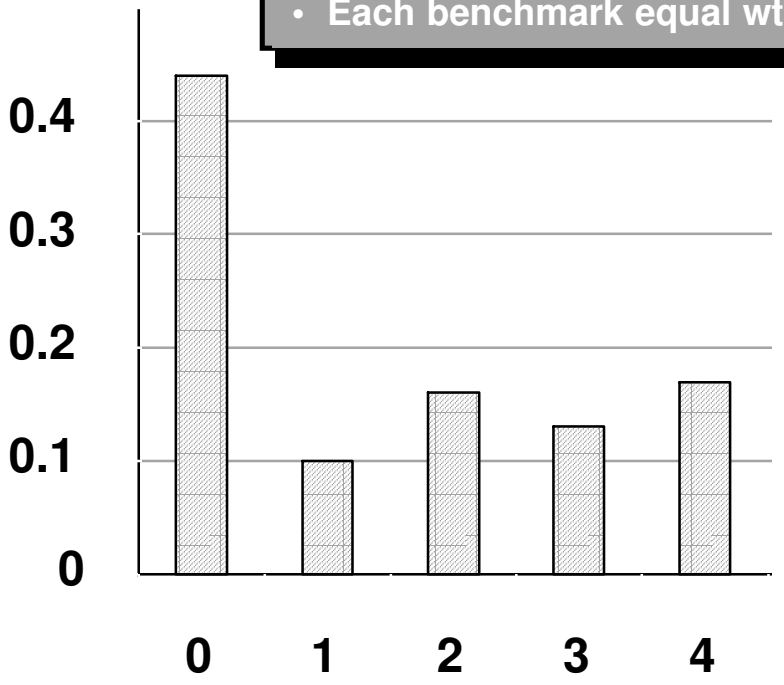
MR is Miss Rate expressed as Misses per Instruction Retired

Perfect Branch Predict



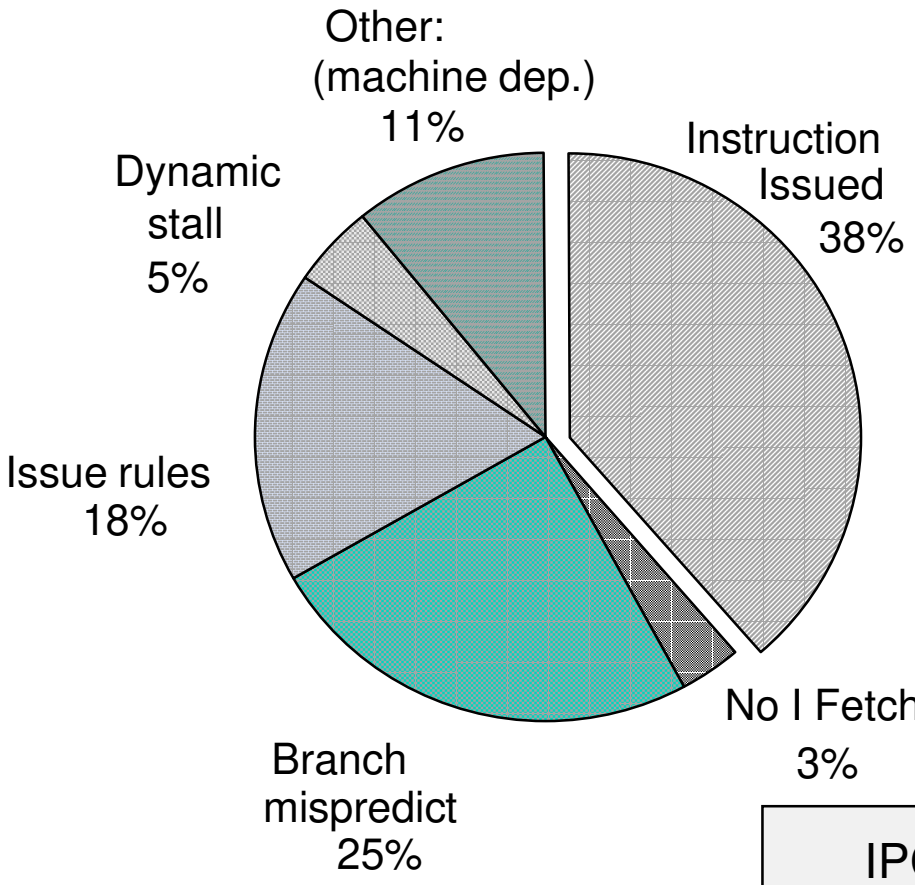
Branch mispredict = 0%

Estimate for SPECint92
- **Avg. of 6 benchmarks**
- **Each benchmark equal wt.**

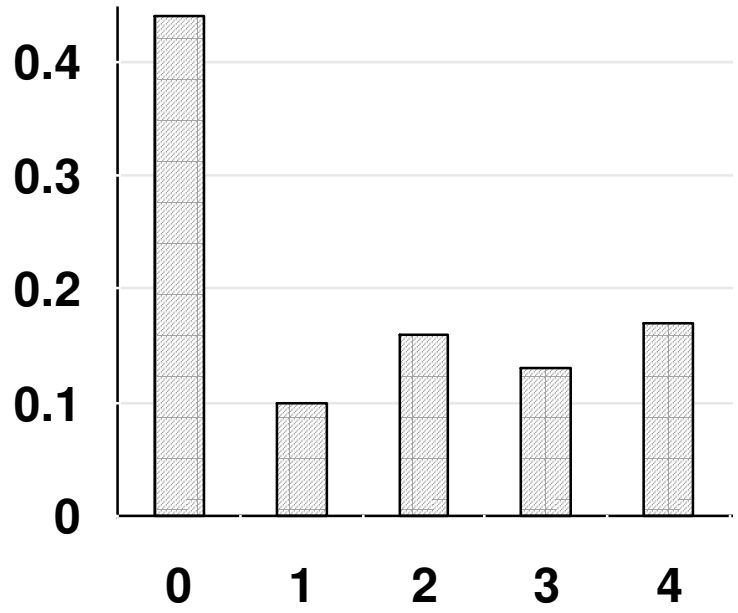


IPC = 1.47 or +22%

Perfect Caches

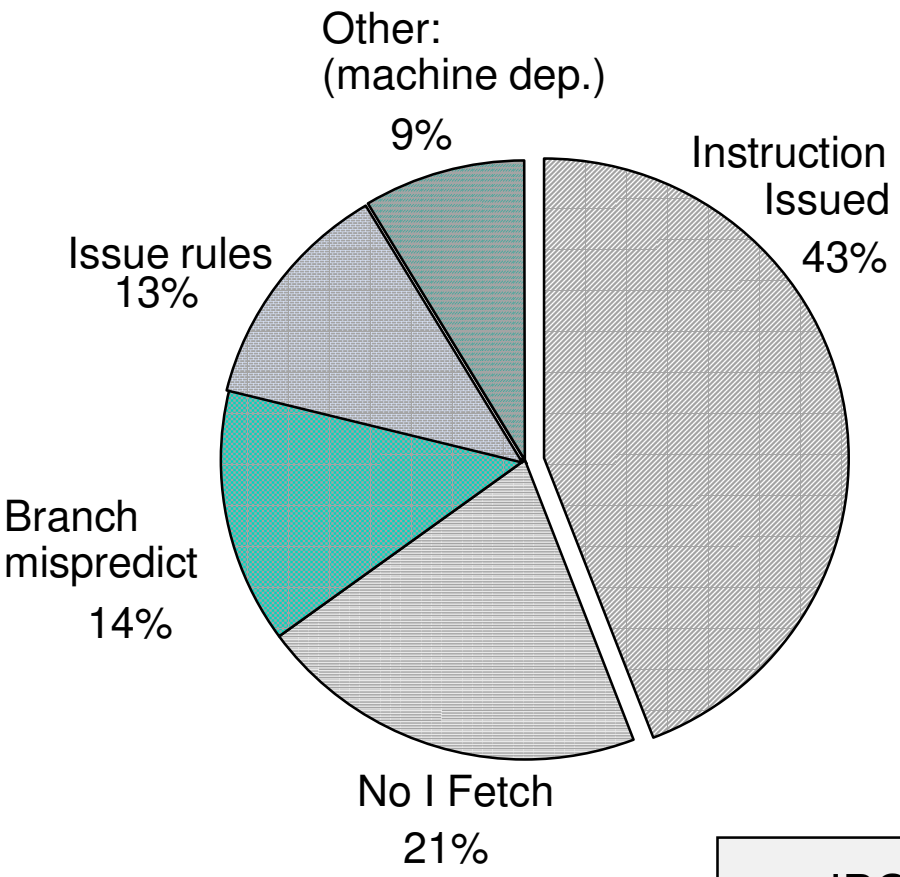


Estimate for SPECint92
- **Avg. of 6 benchmarks**
- **Each benchmark equal wt.**



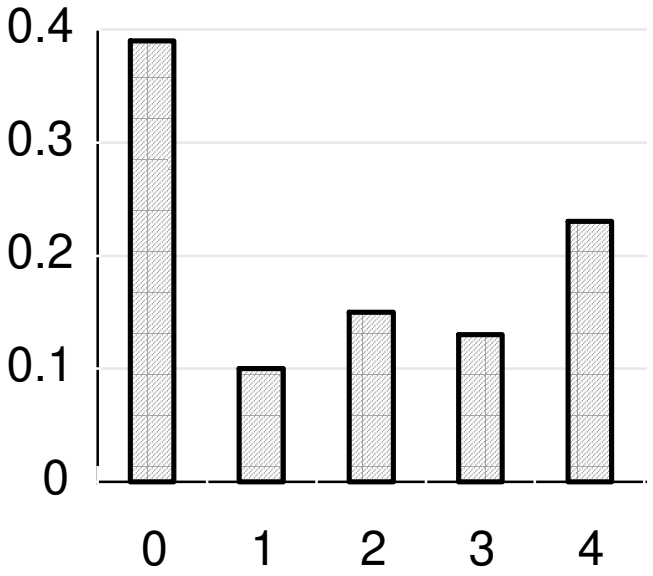
IPC = 1.52 or +27%

All execution in one cycle



Dynamic stall = 0%

- Implies perfect D cache
- Minimizes losses due to branch mispredict



IPC = 1.72 or +43%

HAL COMPUTER SYSTEMS
A Fujitsu Company

SPECint92 / MHz

- **SPECint92 / MHz and SPECint92 are good metrics**
- **Broader comparisons allowed**
  - IPC's are seldom published
  - SPECint results are publicly available
- **Directly comparing IPC's has "the MIPS problem"**
  - Different architectures make IPC comparisons meaningless across vendors
  - A machine running the same benchmark with different binaries can give misleading IPC results