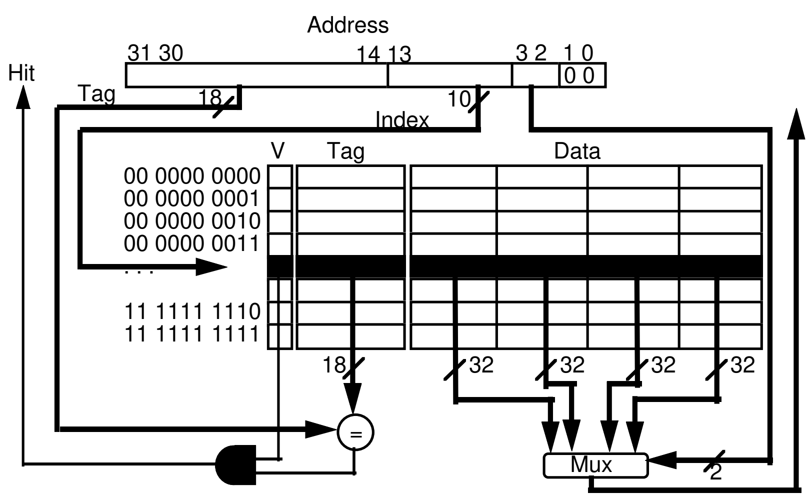
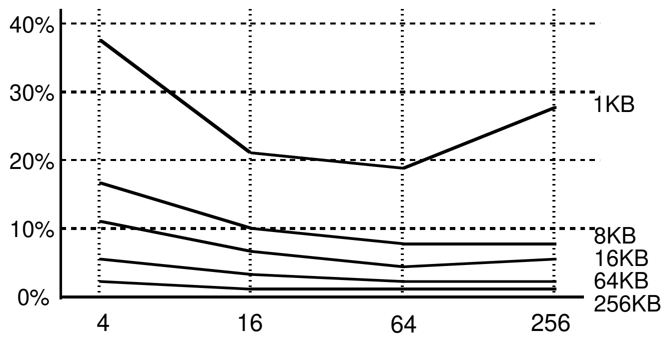
Caching

Contructing an effective cache requires the balancing of many properties. Bigger is always better, but how it is arranged is also important.

Direct Mapped Cache (4 word blks)



Miss rate vs block size



VAX traces (Agarwal,1987)

Benefits of Multiword Blocks

Increasing block size improves performance, to a point.

- Larger blocks increase benefits of spatial locality.
- Larger blocks = fewer blocks for a given cache size = greater likelihood a useful block is flushed when another block is brought in (conflict misses).

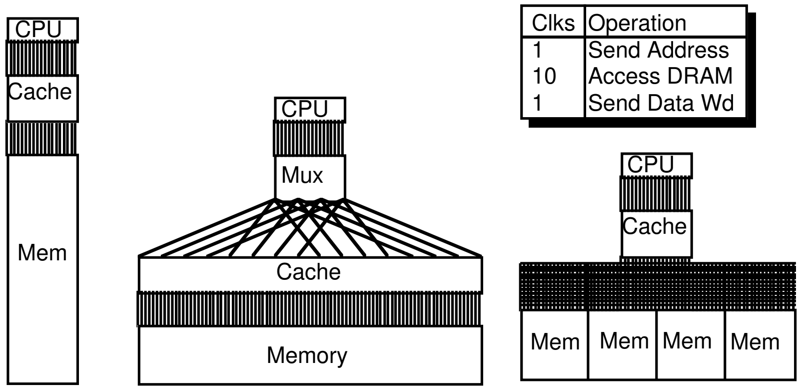
Memory request techniques --

- Early restart
- Requested word first

Pgm	Wd	Inst Miss	Data Miss	Effective Miss Ratio
GCC	1	6.1%	2.1%	5.4%
	4	2.0%	1.7%	1.9%
Spice	1	1.2%	1.3%	1.2%
	4	0.3%	0.6%	0.4%

Memory Organizations

Clks	Operation
1	Send Address
10	Access DRAM
1	Send Data Wd



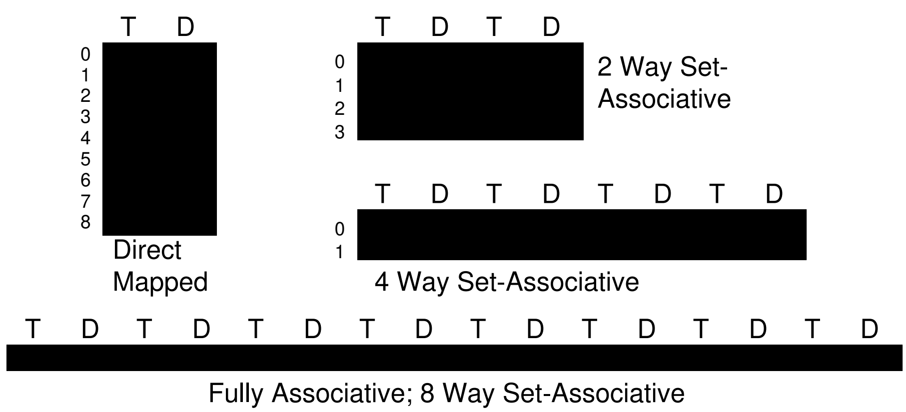
(a) (b) (c)

Miss Penalty: 1+4x10+4x1=45 1+1x10+1=12 1+1x10+4x1=15

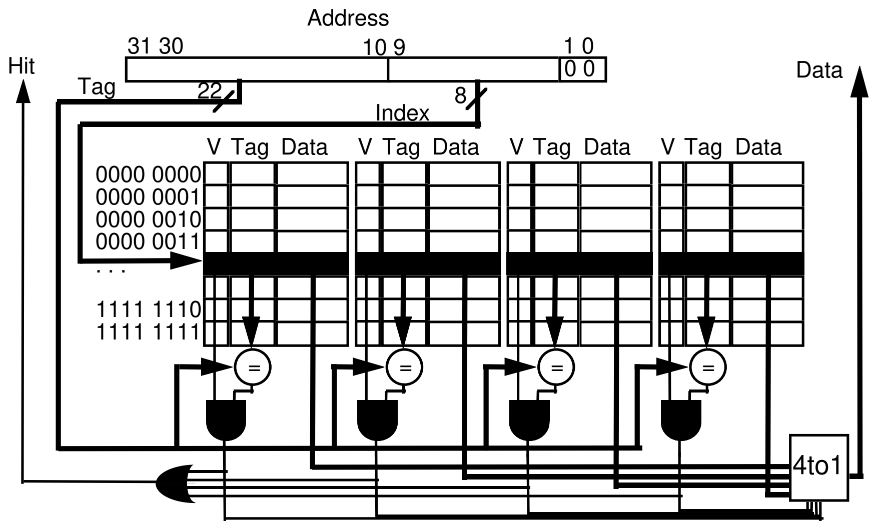
Bytes per Cycle: 4x4/45=0.35 4x4/12=1.33 4x4/15=1.0

Alternative Designs

Possible arrangements for cache elements



Set Associativity



Caching Performance ... a simple model

CPU time = (CPU_execution_clocks+Memory-stall_clocks) x Cycle_time

Memory-stall_clocks = Read-stall_clocks + Write-stall_clocks

Read-stall_clocks = Read/Program x Read_miss_rate x read_miss_penalty

Write-stall_clocks = (Write/Program x Write_miss_rate x write_miss penalty)
+ Writebuffer-stalls

Memory-stall_clocks = Memory_accesses/Program x Miss_rate x Miss_penalty

Memory-stall_clocks = Insts/Program x Misses/Inst x Miss_penalty

Example

Data:
- 5% instruction misses,
- 10% data misses,
- CPI without stalls = 4,
- miss penalty =12

What is the CPI including misses?

Inst_Miss_clocks = IC x 5% x 12 = 0.6x IC

Assume GCC numbers, 22% LW + 11% SW implies:

Data_Miss_clocks = IC x 33% x 10% x 12 = 0.4xIC

Memory stalls = 0.6IC + 0.4IC = 1.0IC Thus, CPI = 5 with stalls.

Cache Performance (continued)

CPI is 25% worse due to stalls ...

Consider doubling the clock rate without changing the memory system:

Miss_cycles/instruction = 5% x 24 + 33% x (10% x 24)
= 2.0 added to CPI

How much faster is the faster machine with the 2x clock?

$$\frac{\text{Performance}_{fast}}{\text{Performance}_{slow}} = \frac{\text{Ex\_time}_{slow}}{\text{Ex\_time}_{fast}} = \frac{\text{IC x CPI x cycle}}{\text{IC x CPI x (cycle/2)}} = \frac{5}{6\text{x}0.5} = \frac{5}{3}$$

Thus, the 2x machine is actually only 1.67 times faster