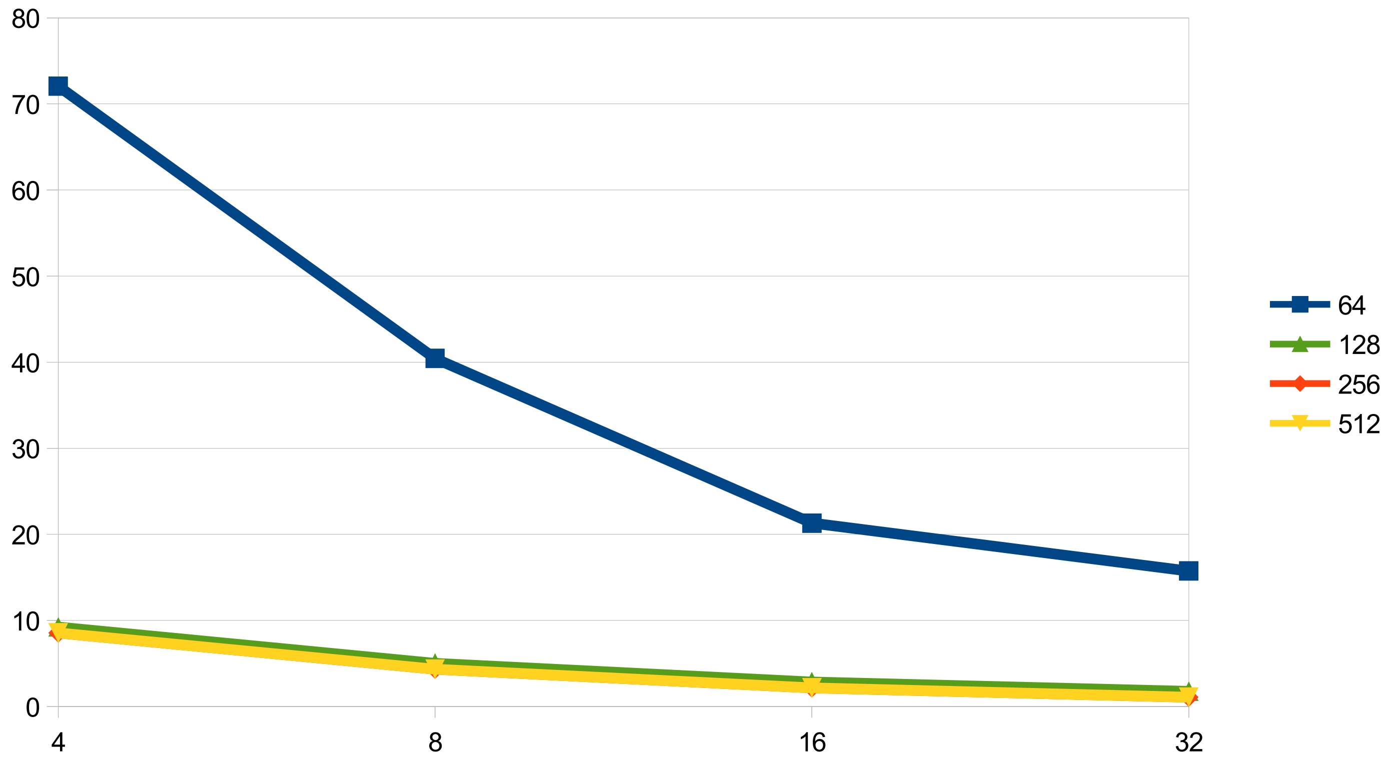
Computer Organization Lab 6 Report

by 0016014, 0016045(group 4)

1.BASIC

ICACHE:

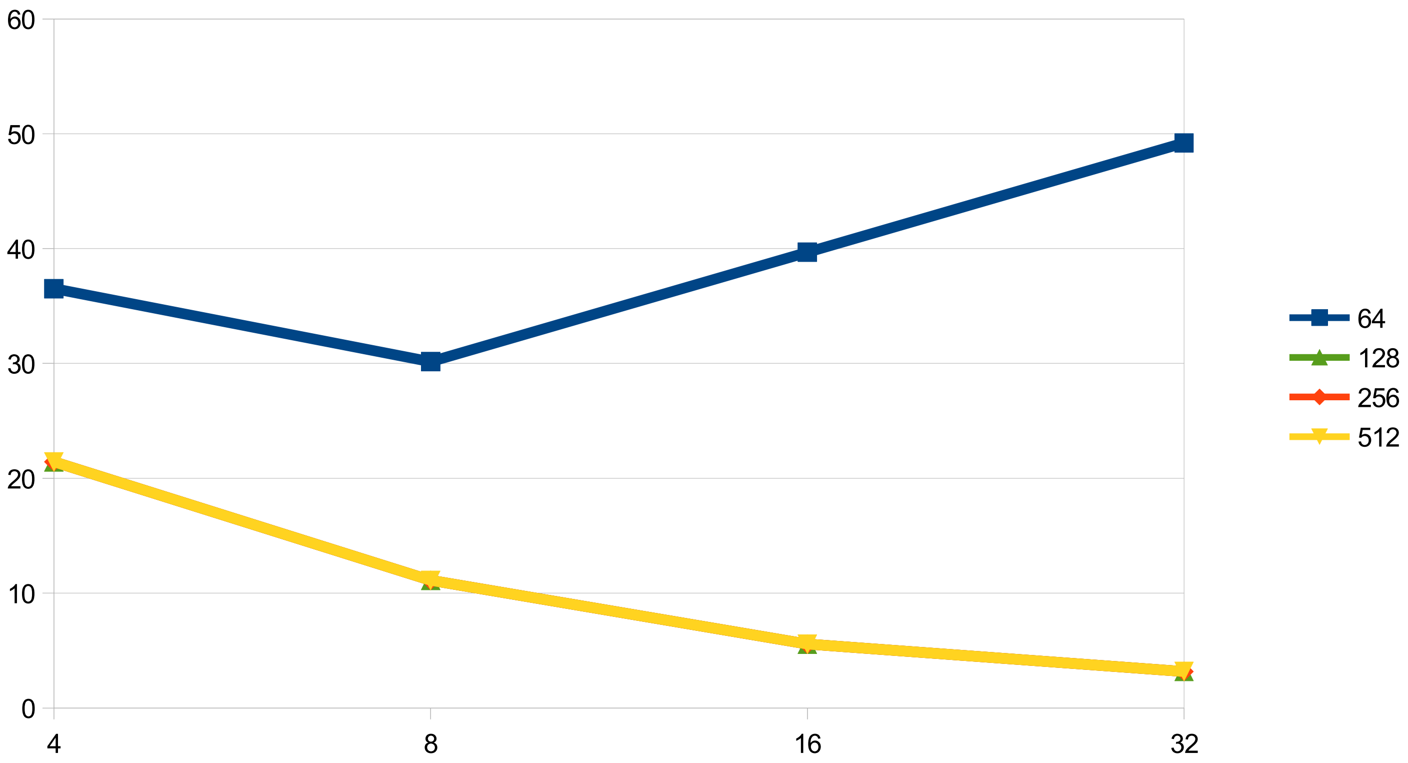


(x = block size in B, y = miss rate in %. 4 different cases of cache size in B)

Larger cache size -> More blocks -> Less collisions, but since the first 16 bits of address used in ICAHCE are all zero, the effect between128, 256, and 512 is quite limited.

Larger block size -> More spatiality. Since address used in ICACHE are very linear, the more spatiality is quite effective in this case and surpasses the less-block disavantage.

DCACHE:

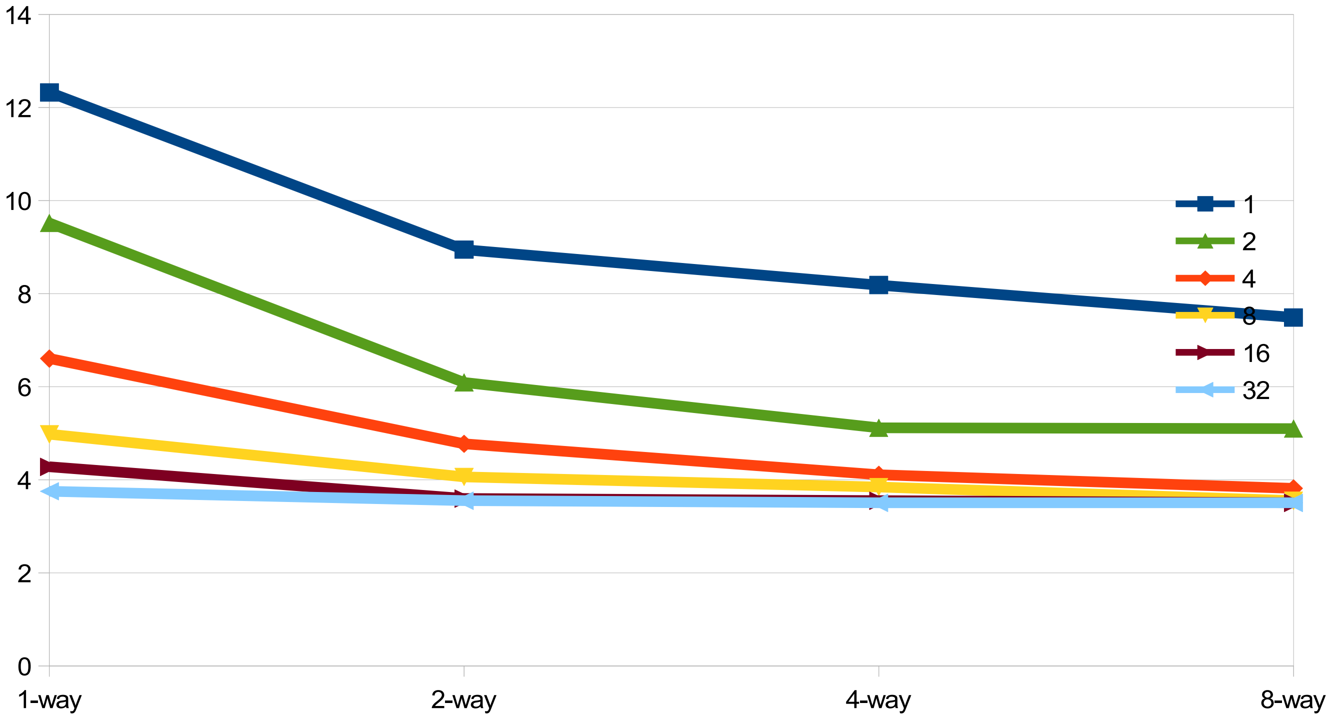


Larger cache size is quite like the case in ICACHE.

Larger block size -> More spatiality. Unlike ICACHE, The address used in DCACHE is more messy, so the less-block disavantage is more powerful when cache is small enough.

2. Advance a

LU:



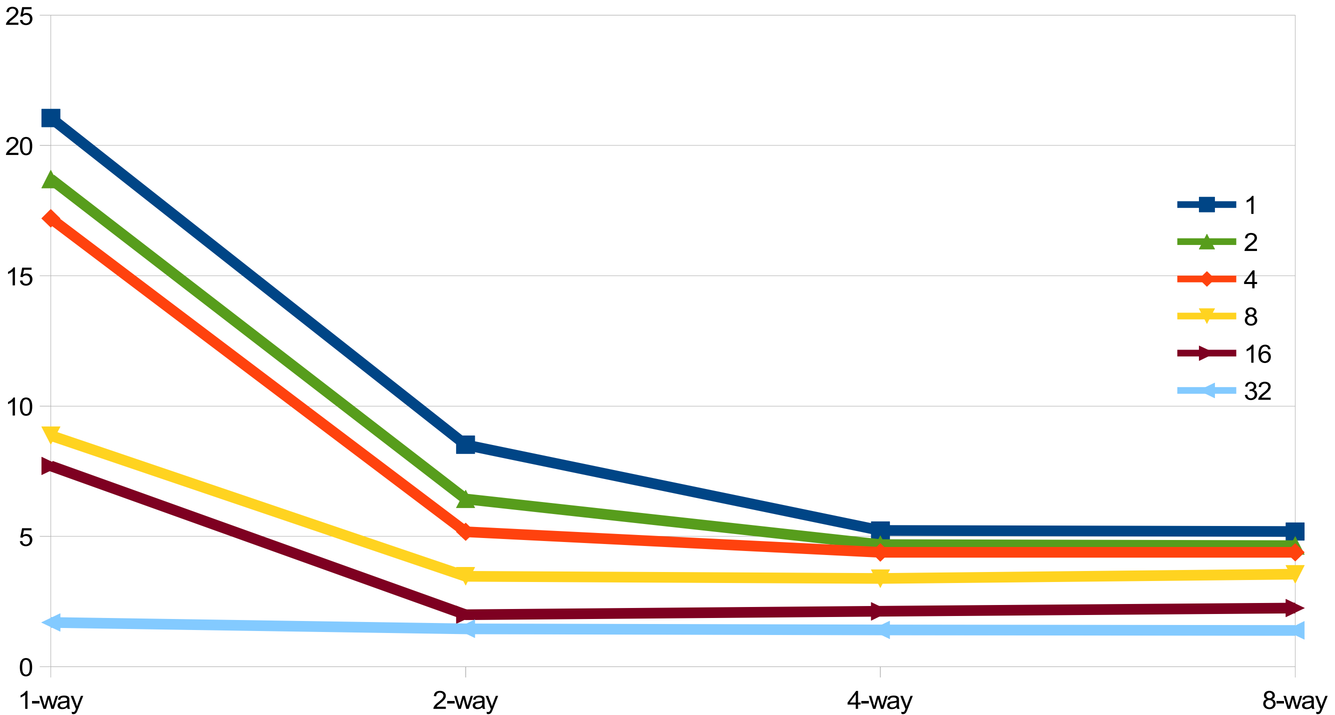
(x = ways, y = miss rate in %. 6 different cases of cache size in KB)

More associativity -> Less collisions for a index

Larger cache size -> More blocks -> Less collisions

Limited variaty in address causes little improvement between large cases.

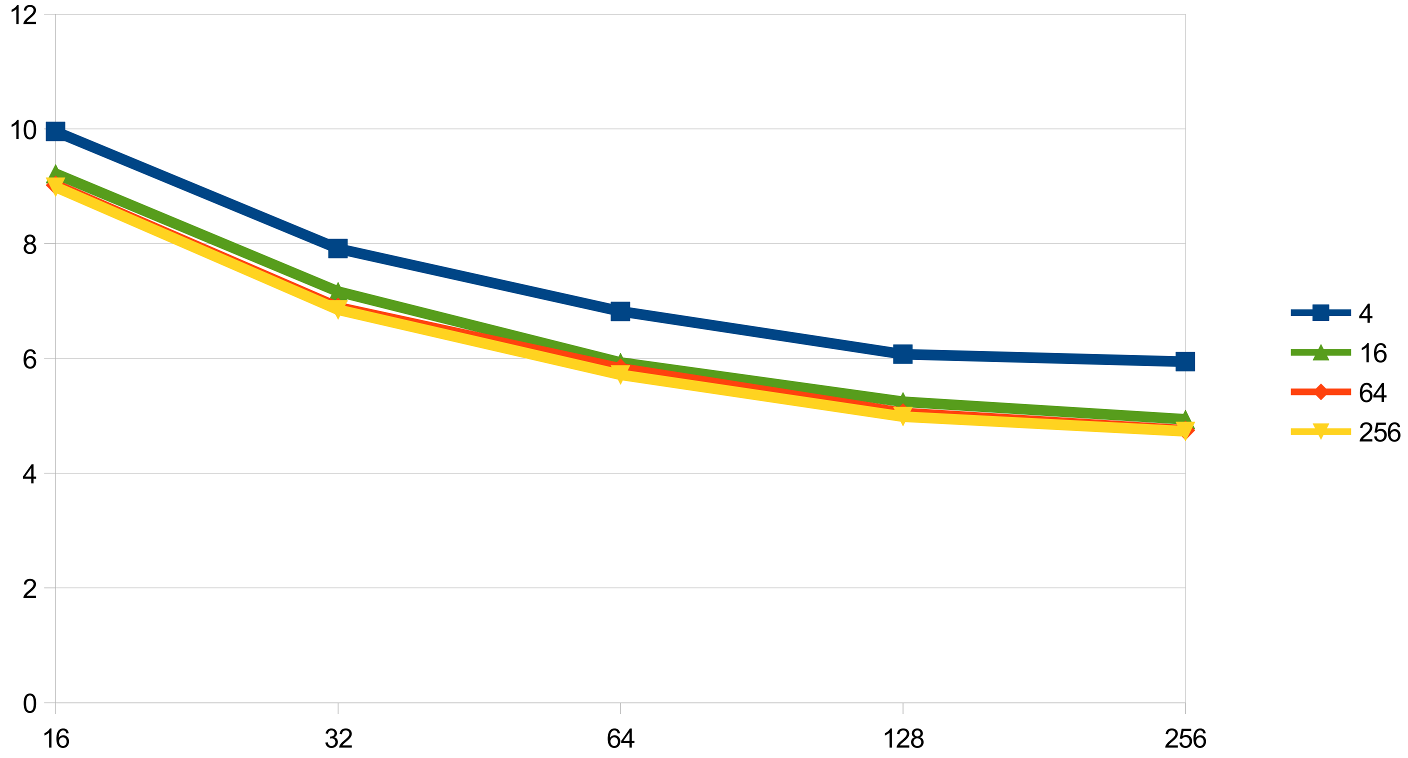
RADIX:



More associativity and Larger cache size are quite like the case in LU.

3. Advance b

LU:



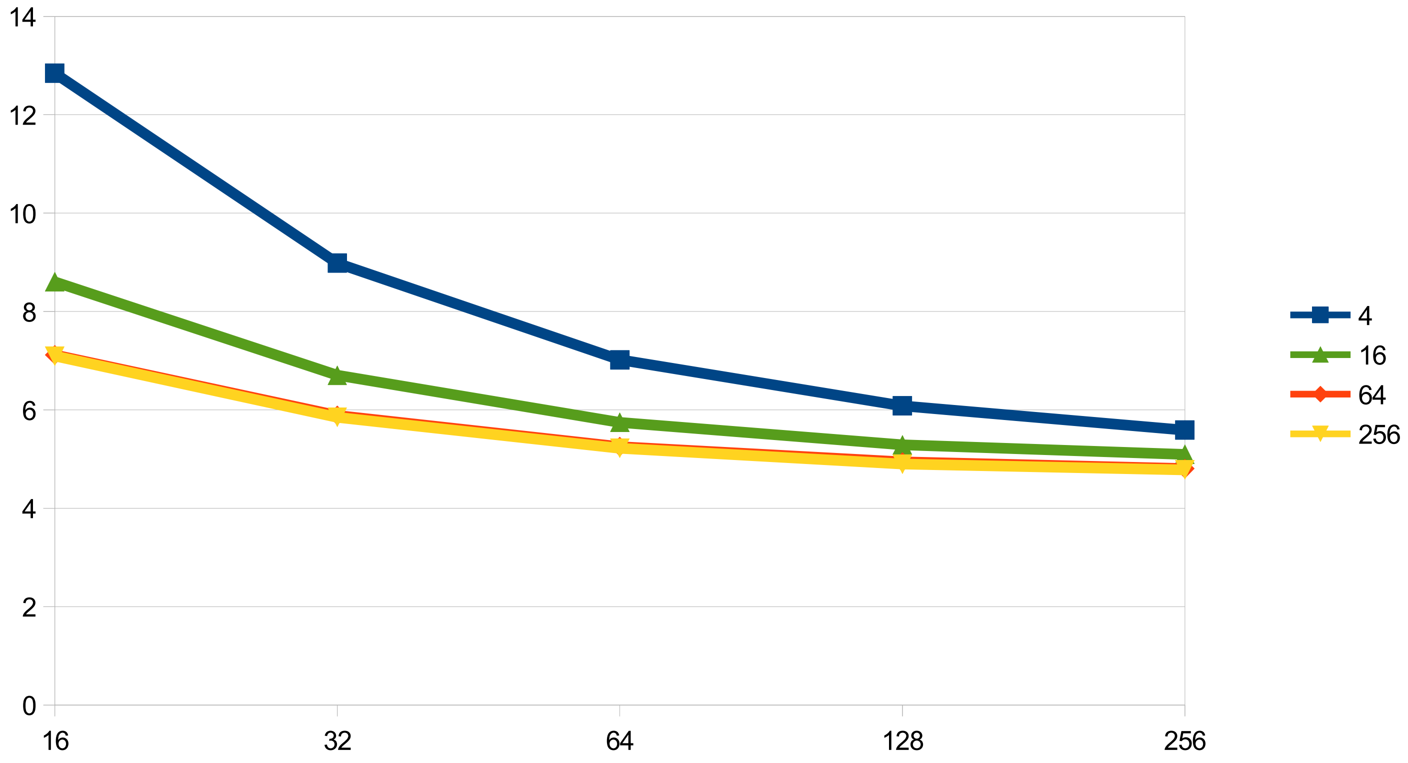
(x = block size, y = AMAT in cycles. 4 different cases of cache size in KB)

Larger block size -> More spatiality

Larger cache size -> More blocks -> Less collisions

Limited variaty in address causes little improvement between large cache size cases.

RADIX:



Larger block size and Larger cache size are quite like the case in LU.