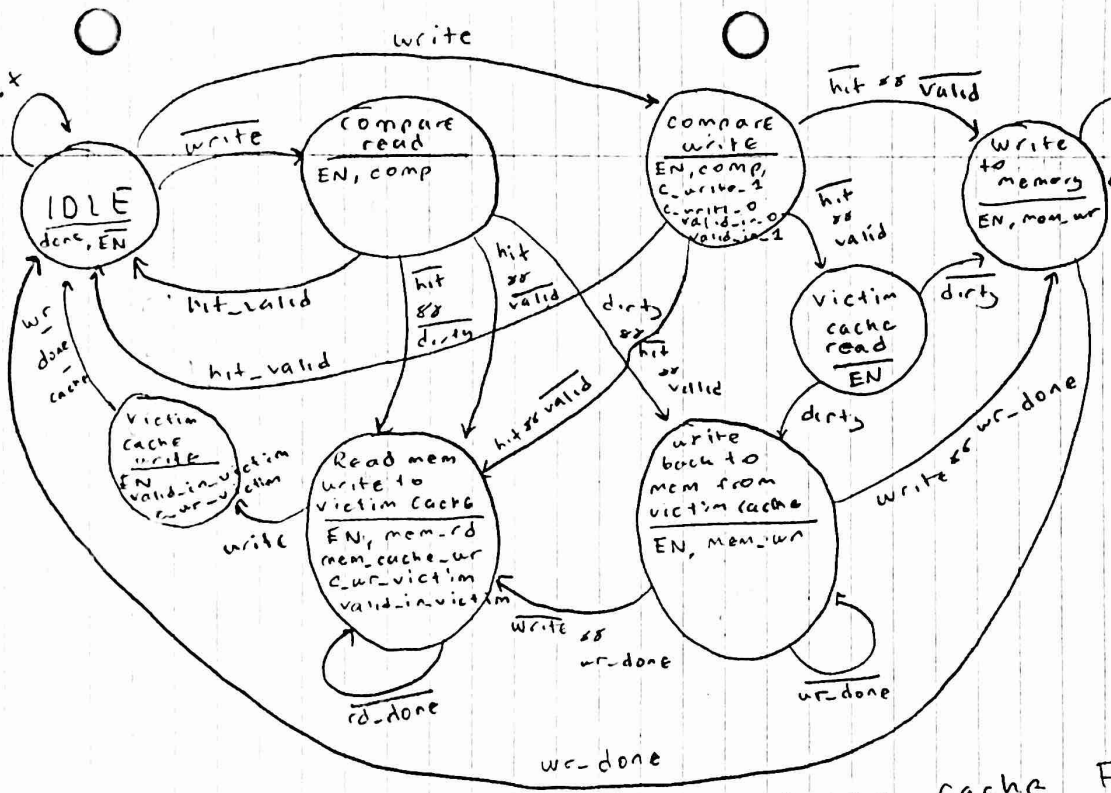

ECE 552 HW5

Kshitij Wavre
David Waltz

state output	description
enable	cache is enabled
mem_rd	read from memory
mem_wr	write to memory
comp	enables comparison tags
c_write_0	write to cache 0
c_write_1	write to cache 1
valid_in_0	input to cache 0 is valid
valid_in_1	input to cache 1 is valid
mem_cache_write	write to cache data from mem
done	operation is complete or in IDLE

Problem 2: two-way set assoc. cache FSM

transition signal description

hit = hit_0 · vic_flop + hit_1 · vic_flop

hit_valid = (hit_1 && valid_1) || (hit_0 && valid_0)

dirty = dirty_0 · $\overline{\text{vic\_flop}}$ + dirty_1 · vic_flop

valid = valid_0 · $\overline{\text{vic\_flop}}$ + valid_1 · vic_flop

wr_done : writing to mem is done

rd_done : reading to mem is done

victimizeflop = 0 or 1. Invert on read or write. 0 victimizes cache 0. 1 victimizes cache 1.

wr_done_cache : done writing to cache

rd_done_cache : done reading from cache

state description

- compare read & compare write will victimize && and find the victim blk in the victim cache accordingly
- victim cache can be cache_0 or cache_1 due to replacement policy