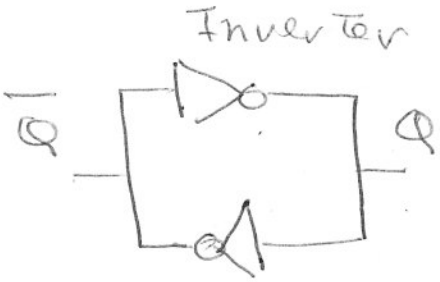
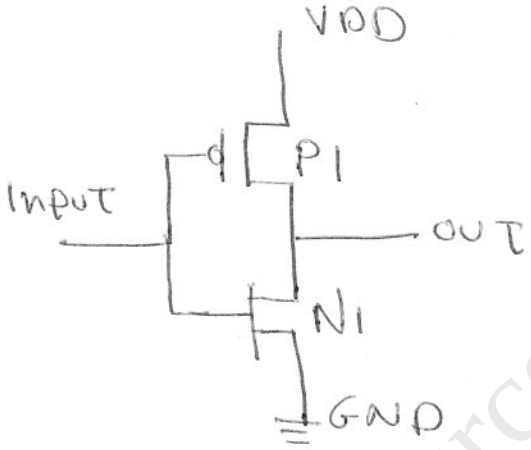
SRAM organization

Latch

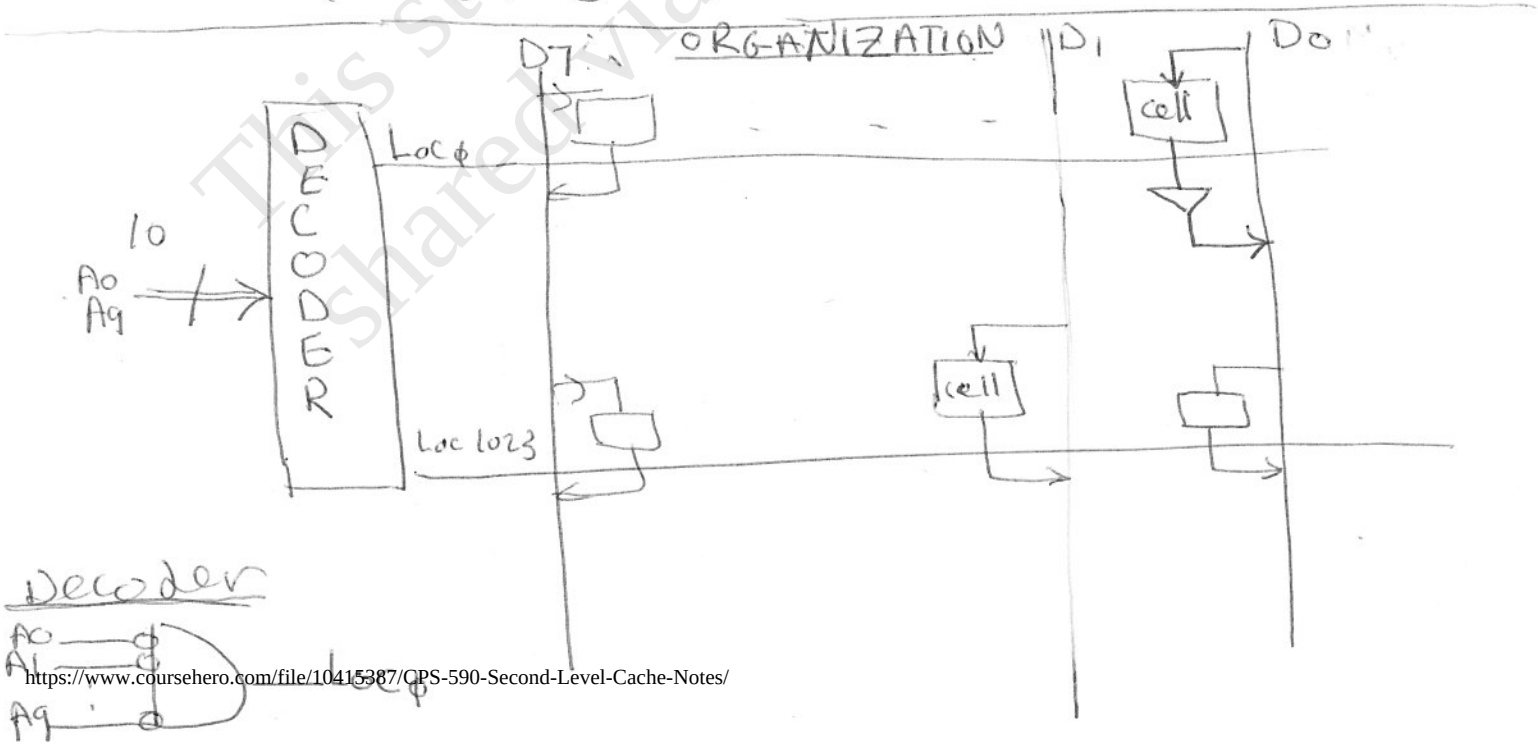


Inverter



Input = 0
P1 conducts, N1 off
OUT = VDD = 1

Input = 1
N1 conducts, P1 off
OUT = GND = 0



Decoder

Method 4: Non Blocking Caches

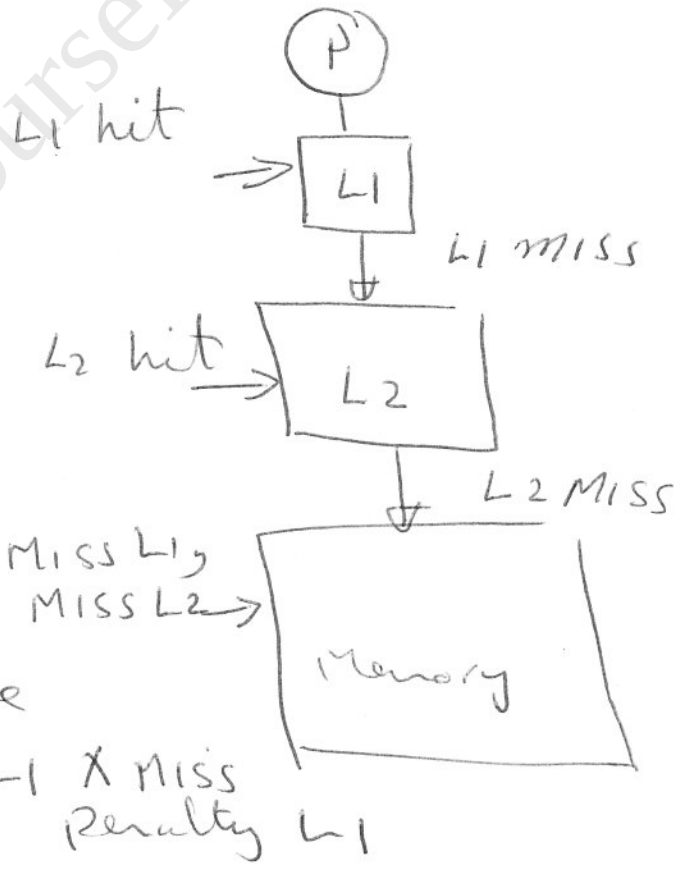
For pipelined machines that allows out of order execution and completion, CPU could continue fetching instructions from instruction cache while waiting for data cache miss. It also allows Data cache to supply data while waiting for a miss data to arrive from memory.

- Need Scoreboard or Tomasulo control.

Method 5: Second Level Caches

on a first level cache miss, can use a faster level 2 cache and not the memory.

only misses from L1 could be found in L2.



Average memory access time
= Hit time L1 + Miss L1 X Miss Penalty L1

Miss Penalty L1 = Hit Time L2 + Miss L2 X Miss Penalty L2

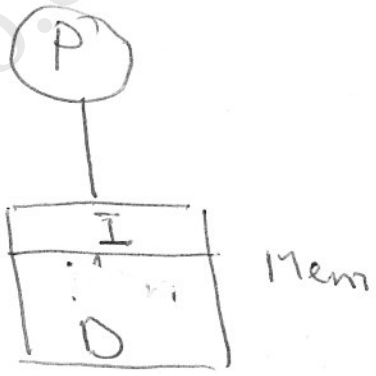
Main Memory

- Introduction (cpu and Mem)
- SRAM (organization, Timing, Expansion)
- DRAM (organization, Timing, Access modes, Interleaving, EXPANSION)

Introduction

Processor and Memory model

- Instructions stored in Memory. Processor uses Fetch
- Data stored in Mem Processor uses load/store



Each single instruction might access memory twice (Fetch, and load/Store)

Example lw R1, 100(R2)
R1 ← M[100+R2]

- Using RAM = Radom Access memory. All accesses have same access time
- Computer system performance depends on memory speed (or cpu??)