CS 451 / 551 / ECE 541

ADVANCED
COMPUTER ARCHITECTURE

SESSION no. 14

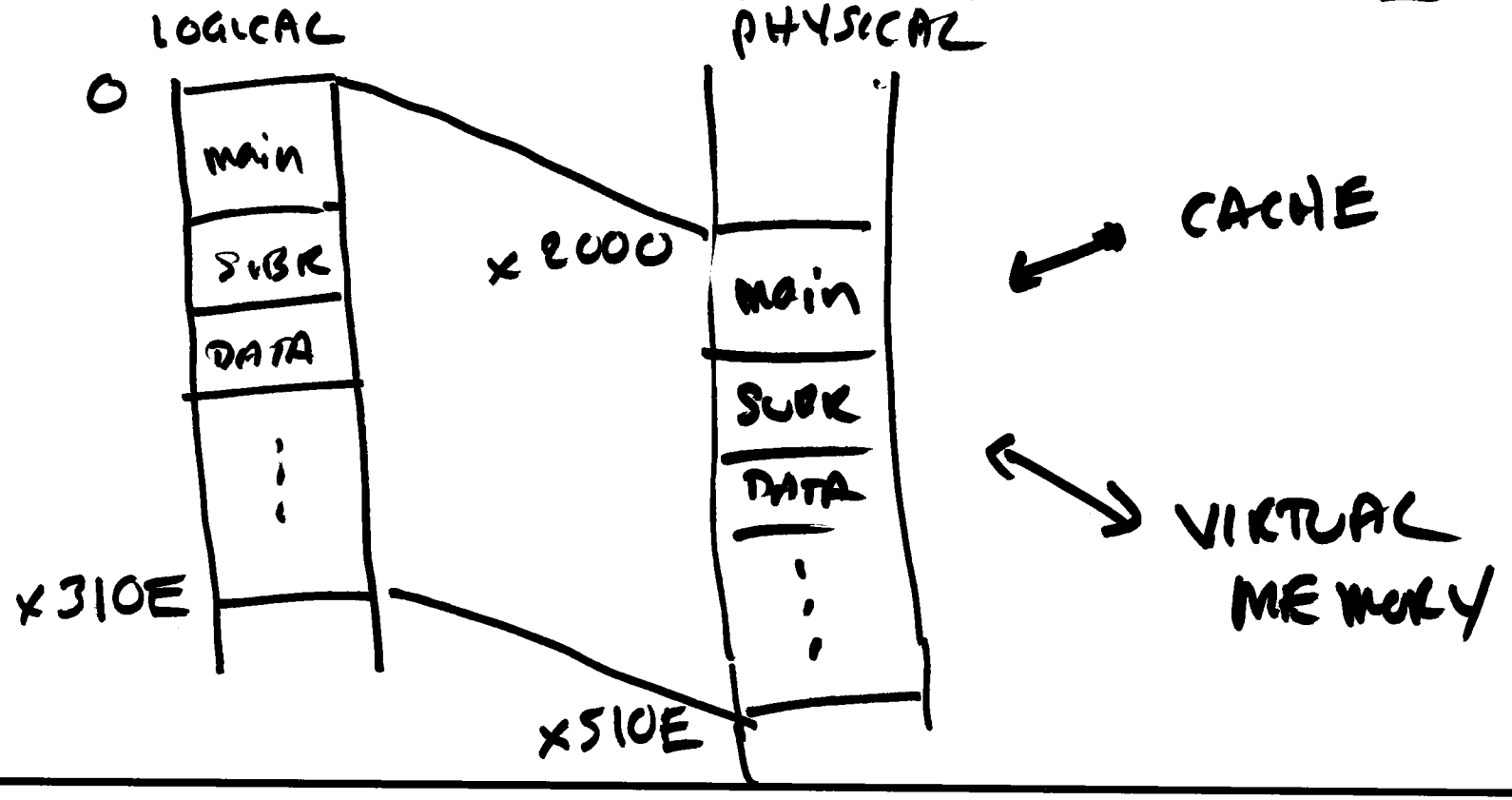
University of Idaho

EXAM #1 ON THURSDAY OCT 17

- IN CLASS
- OPEN BOOK, OPEN NOTES, NO WIFI!
- 100 PTS 1HR + 15 MIN
- CONCEPTS - SHORT ANSWER

LOGICAL VS PHYSICAL ADDRESS SPACE

- LOGICAL - ADDRESSES DEFINED BY # OF ADDRESS BITS.
- PHYSICAL - ACTUAL MEMORY (DRAM) < LOGICAL ADDRESS SPACE



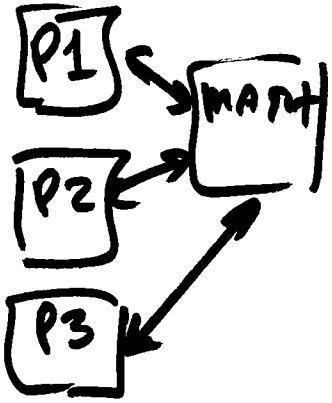
ADDRESS TRANSLATION

(LINKING) LOADER

c/s

LOGICAL → PHYSICAL

DLL



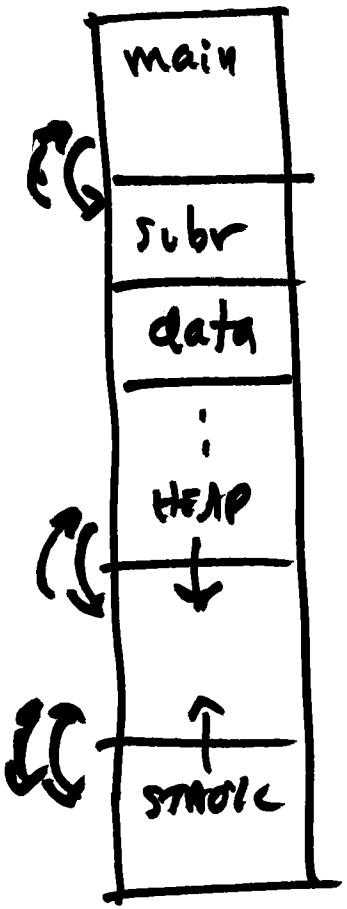
MEMORY MANAGEMENT UNIT (MMU)

HARDWARE ASSISTANCE

POSITION INDEPENDENT CODE

- ALL MEMORY REFERENCES RELATIVE TO PC
- EASILY RELOCATEABLE

HOW MANY CACHES? BLOCKS? SIZE? SETS?



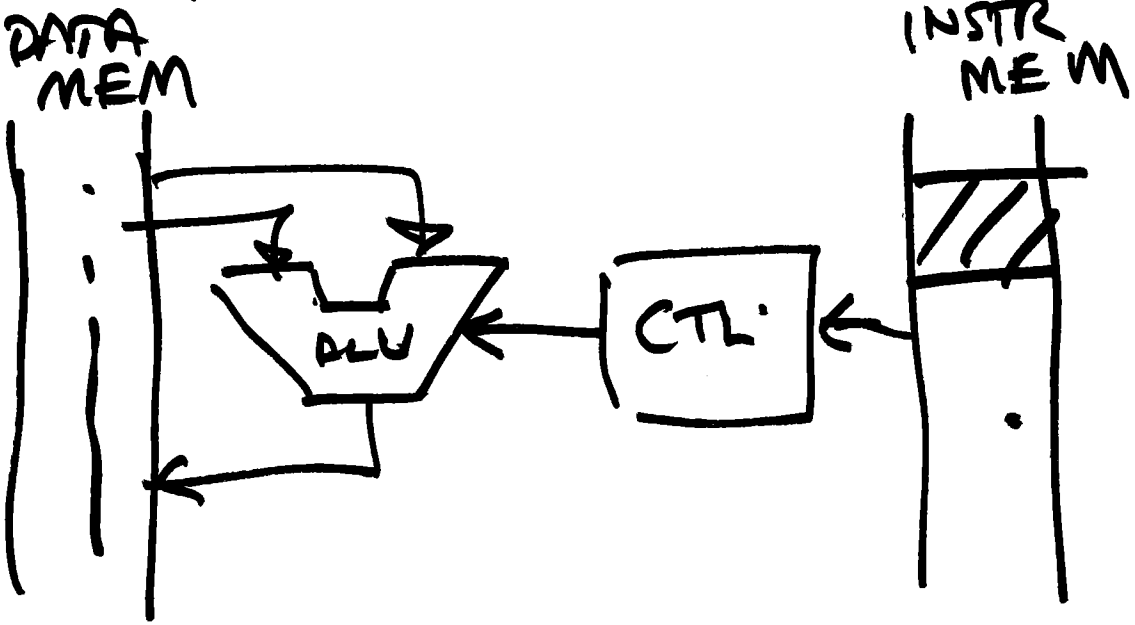
- HOW MANY "HOT SPOTS" OF ACTIVITY?
- PROFILE
  - CODE
  - SYSTEM

SIMULATE
- VERIFY MODEL
- ESTIMATE PERFORMANCE

"SPEC" BENCHMARKS

OLD DAYS: HARVARD ARCHITECTURE

- SEPARATE INSTR & DATA MEM
- "I AND D SPACE"



ADVANTAGES:

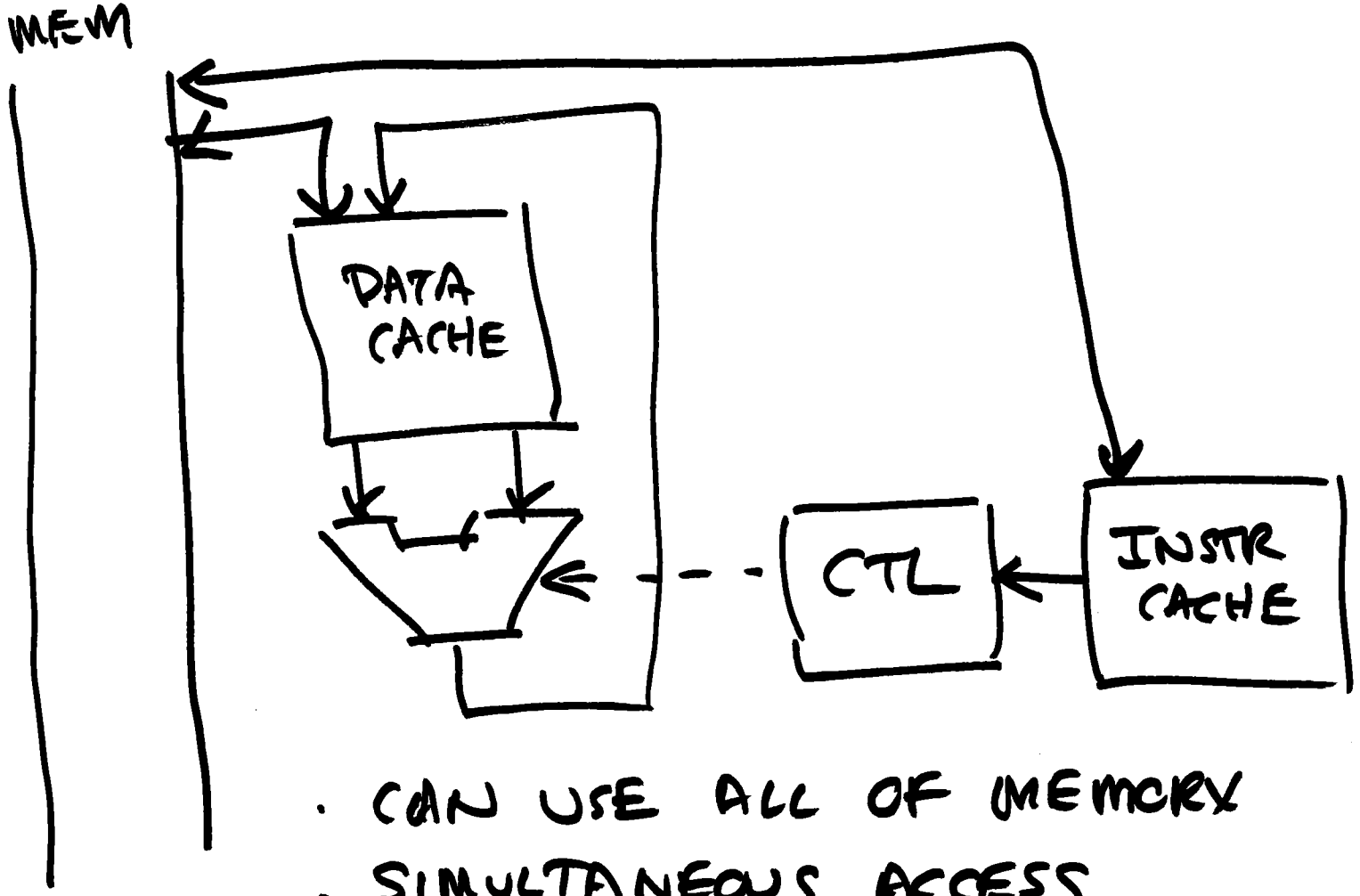
- PARALLEL ACCESS

DISADVANTAGE:

- WASTED MEMORY

University of Idaho

SEPARATE I & D CACHE



- CAN USE ALL OF MEMORY
- SIMULTANEOUS ACCESS THRU CACHE
- EXTRA COST, COMPLEXITY

CACHE - FRIENDLY CODE

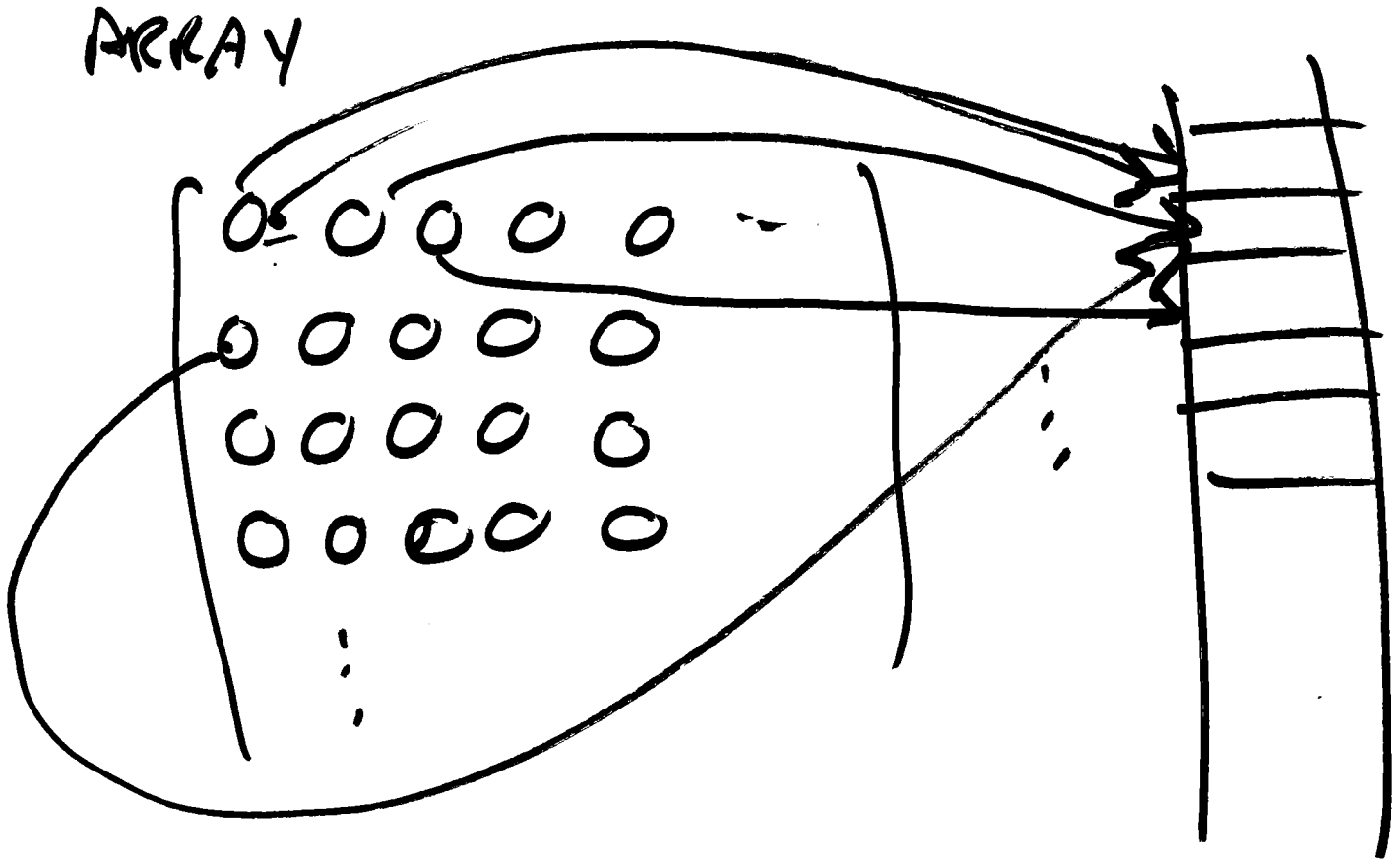
JOB INTERVIEW ALERT!

- ~~TEMOR~~ ENHANCE TEMPORAL & SPATIAL LOCALITY

1. BREAK PROGRAM UP INTO SMALL MODULES
   - SMALL COMPUTATIONAL BLOCKS
   - FUNCTIONS

2. DATA ORDERING

3. CHOICE OF DATA STRUCTURE ☑
   - ✓ ARRAY? — CACHE FRIENDLY
   - LINKED LIST?



ROW ORDERING - BETTER SPATIAL LOCALITY.

. CACHE- FRIENDLY.

```
for (i=0; i<1000000; i++)
    for (j=0; j<1000000; j++)
```

inner loop - adjacent memory storage

University of Idaho

SISD - CPU, CORE

SIMD - DATA PIPELINES, VECTOR MACHINES

MISD - X

MIMD - MULTICORE - GPGPU

GRAPHICS

PARALLELISM

- INSTRUCTION LEVEL - ILP } EASY!
  - INSTR PIPELINING
- DATA PARALLELISM - SIMD, VECTOR MACHINES ~~HARD~~
- THREAD PARALLELISM - MULTICORE, MIMD
  PROGRAMMER AWARENESS - HARDER

University of Idaho

HAZARDS

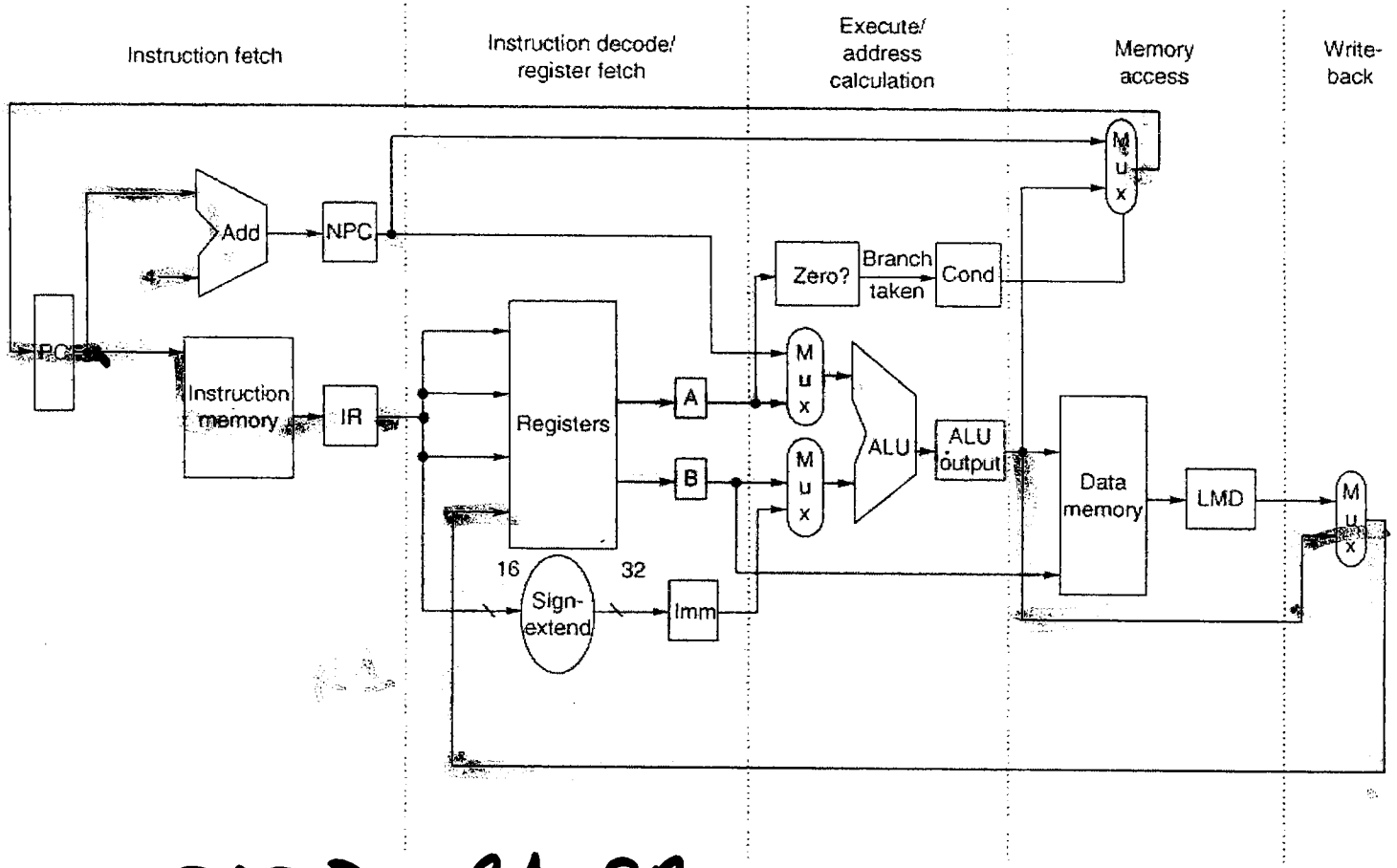
1. DATA - CALCULATION DEPENDS ON RESULTS OF A PREVIOUS CALCULATION WHICH IS STILL IN THE PIPELINE.
   - CODE RE-ORDERING
   - FORWARDING
2. CONTROL - BRANCH (CONDITIONAL) TAKEN OR NOT? PREFETCH WRONG INSTRUCTION
   - BRANCH PREDICTION
     - STATIC - "COMPILE TIME GUESS"
     - DYNAMIC - RUN TIME, BASED ON PAST

3. STRUCTURAL — COMPETITION FOR RESOURCES

- CALCULATION UNIT
  - ARITH INSTR } ADDER
  - UPDATE PC }
- SIGNAL ROUTING PATHS

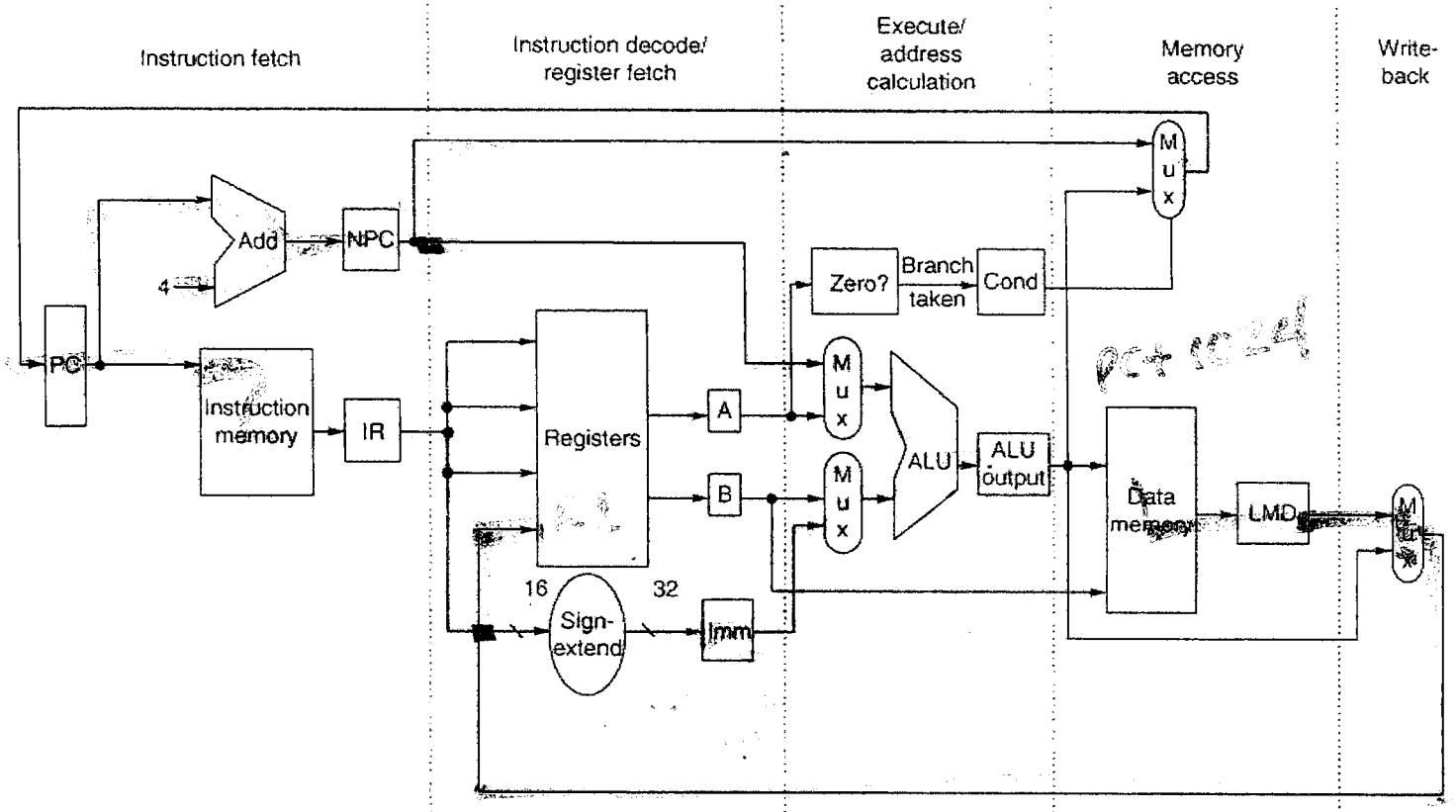
CACHE OPTIMIZATION

1. Minimize miss rate
2. Minimize miss penalty



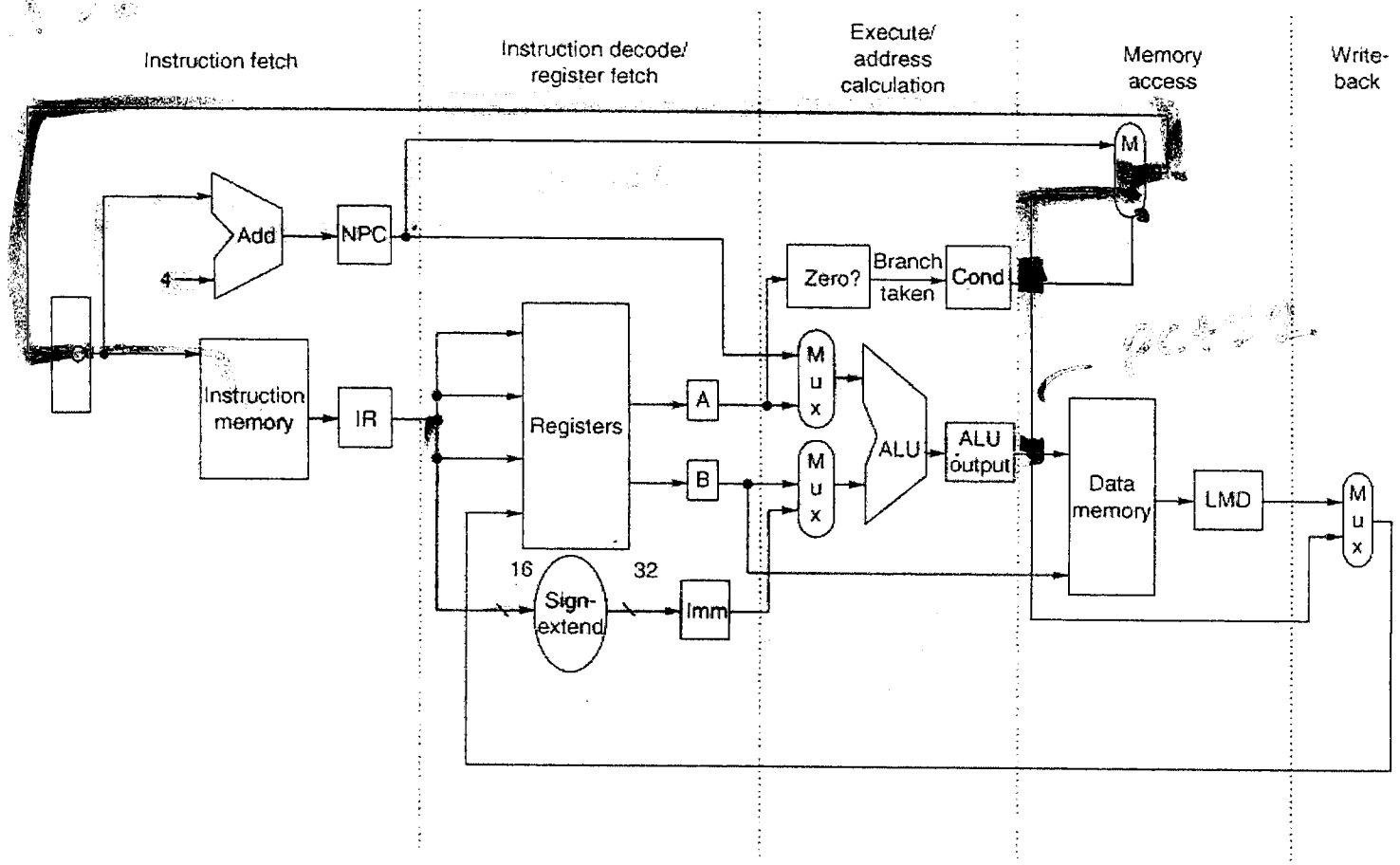
DADD R1, R2

R1 ← R1 + R2



LD R1, PC+1024

R1 ← MEM[PC+1024]



BEQ PC+32

if COND == 0

PC ← PC+32 = PC+4+28