COMP. ARCH. III : MULTI-CORES ETC.

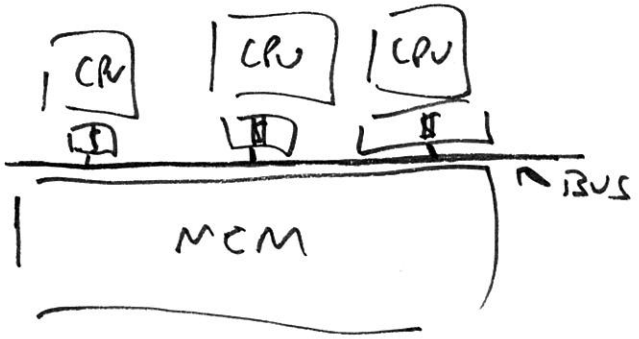
- MOORE'S LAW'S PEMISE → NO MORE FREE LUNCH
  (TRANSISTORS STOP SCALING AGGRESSIVELY)

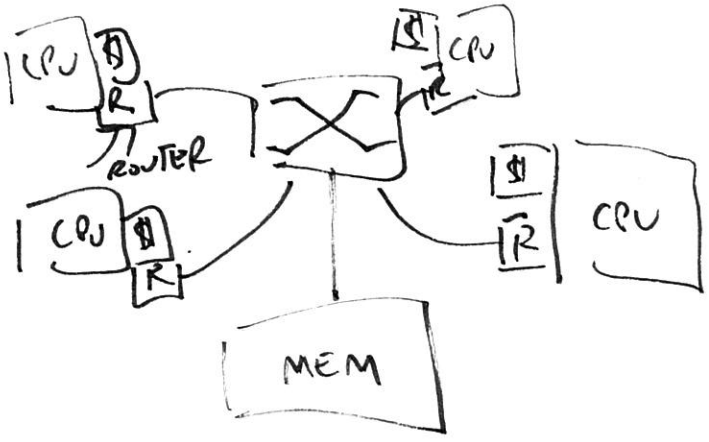
- THE BEST WAY TO GET BETTER PERF?
  PARALLELISM! (RUN MORE THAN ONE THING AT A TIME)

- 1 CPU NOW HAS MULTIPLE CORES ← JUST MORE CPUS! (CMPs)

- HOW DOES THIS WORK?

  ↳ SMP (SYMMETRIC MULTIPROCESSING)





- SAME OS RUNS ON ALL CORES

  (OTHER MODELS POSSIBLE → THIS IS RESEARCH)

- CACHES ARE KEPT "COHERENT". THIS IS HOW THE "CORES" COMMUNICATE

[redacted]

[redacted]

SUPER SCALAR ARCHITECTURES

- DONT JUST FETCH/EXEC ONE INSTRUCTION AT A TIME. FETCH BLOCKS OF INSTRUCTIONS.
- REPLICATE FUNCTIONAL UNITS (ALUS, FPUs, etc.)
- BLOCKS USUALLY DELIMITED BY BRANCHES CALLED BASIC BLOCKS

- LESS COMMON THESE DAYS → CONSUMES A LOT OF ENERGY, DIE AREA

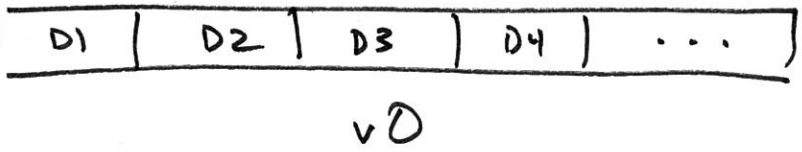
SIMD (SINGLE INSTRUCTION MULTIPLE DATA)

- OFTEN, WE APPLY ONE OPERATION TO MANY PIECES OF DATA. THINK: SCALAR × VECTOR.

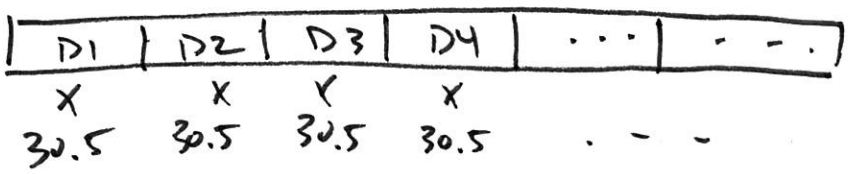
OR, MAP(f, VECTOR).

Ex of DATA PARALLELISM

ADD "VECTOR REGISTERS", e.g. v0



Ex: DMUL #30.5, v0



MOST CPUS THESE DAYS HAVE SIMD UNITS AND INSTRUCTIONS.

GPUS: GRAPHICS GREAT USE CASE FOR SIMD. THINK: A BUNCH OF POINTS THAT REPRESENT AN OBJECT.

TO SCALE THE OBJECT, WE CAN MULTIPLY ITS VECTOR REPRESENTATION BY × MATRIX. DATA PARALLEL

FUTURE HORIZONS

- FPGAS → BECOMING EASIER TO USE
  (AROUND SINCE AT LEAST 80S THOUGH)

- "NEUROMORPHIC COMPUTING" → MAKE CHIPS LIKE BRAINS
  (E.G. IBM'S TRUE NORTH)

- QUANTUM COMPUTING → BITS TO QUBITS

- 1000-CORE CHIPS

- OPTICAL TECH. WITHIN CHIPS

- 3D INTEGRATION

- CUSTOM "ACCELERATORS"

- "WET" COMPUTING: COMPUTING WITH ORGANIC MOLECULES
  (E.G. DNA)