**High Performance Computing**

**Chapter1. Introduction**

1. **Hardware**

- The Von Neumann machine

CPU { CU, ALU } 🡨 cache memory 🡪 memory { program, data }

The control unit is responsible for deciding which instructions in a program should be executed

The ALU is responsible for executing the actual instructions.

- Flynn’s taxonomy (1966)

#instructions vs. #data streams

SISD – typical computer

MISD – X

{ SIMD – old version }

{ MIMD – new version } 🡪 parallel computer

1. **Parallel computers**

-SIMD (Single Instruction Multiple Data)

SIMD systems operate on multiple data streams by applying the same instruction

SIMD scales well

1 CU means program

-MIMD (Multiple Instructions Multiple Data)

Multiple CPUs : each processor(core) executes its own program

1. Shared memory : Multi-core computers
2. Distributed memory : Network 🡪 difficult programming

No OS, compiler

1. Software for MIMD computers

-shared memory programming : OS, compiler

e.g. semaphore

shared int S = 1; // S is semaphore

while( !S ) ;

S = 0;

sum += private\_x;

S = 1;

-distributed memory programming : Network

e.g. Message passing

if ( pid is 0 )

MPI\_Send(x);

Else MPI.Recv(x);