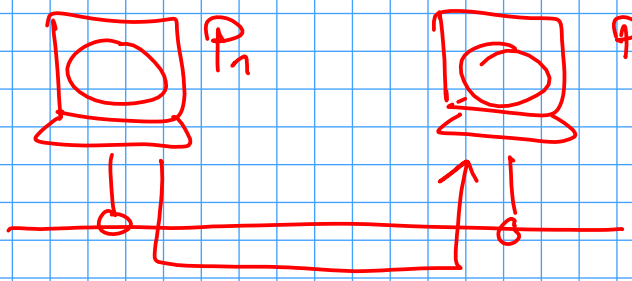
DISTRIBUTED ARCHITECTURES (NOW) (COW)

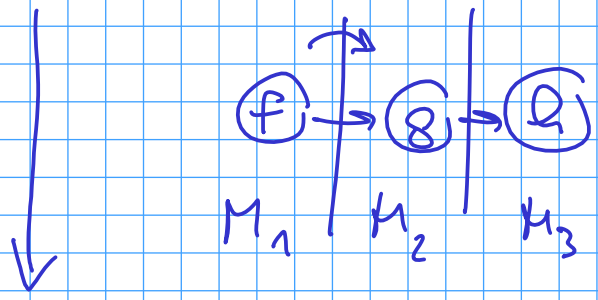


taille of msg

$$t_{comm} = (t_{setup} + d/B)$$

"latency"

1) main goal ⇒ keep computations local to data



2) "coarse grain"

$$\frac{T_{compute}}{T_{communicate}}$$

BUFFERING of comms

↳ if there is possibility to overlap comp & comms

double buffering

triple

Bare minimum : NCI hw with DMA

↳ send

CPU → dir(send/recv) addr, len → NIC

More "powerful" NIC

↳ on board processor

↓

INFI BAND

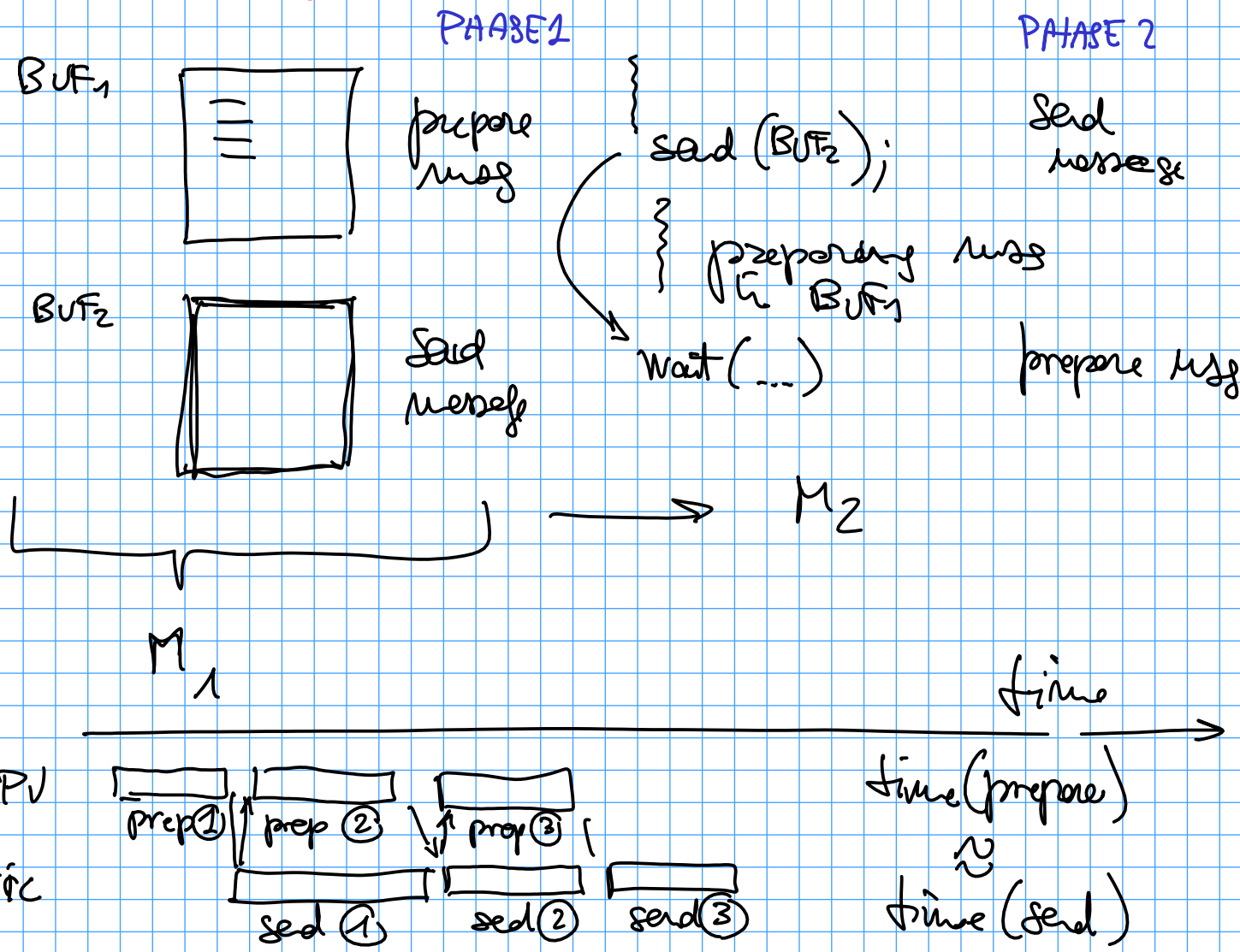
↙ latency (+setup) → < μsec

↘ bandwidth (Gbit/sec) → ≥ 54 Gbit/sec

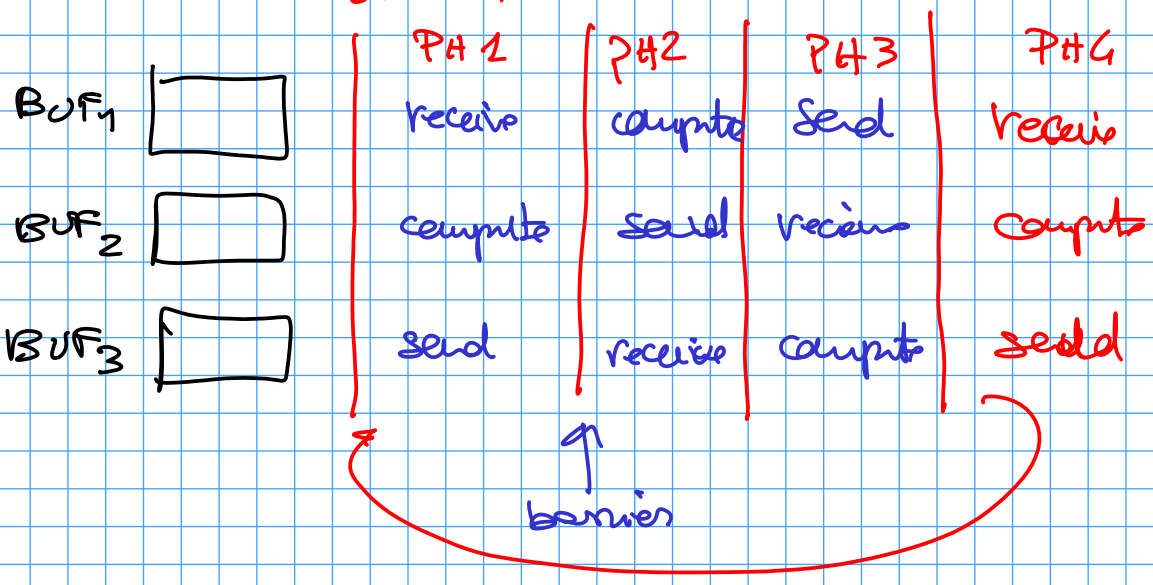
INT ↙

does core of the harmonies directly access Main Memory (same of the CPU

DOUBLE BUFFERING



TRIPLE BUFFERING



$$time(compute) \approx time(send) \approx time(receive)$$

$$\gtrapprox$$

COPROCESSORS

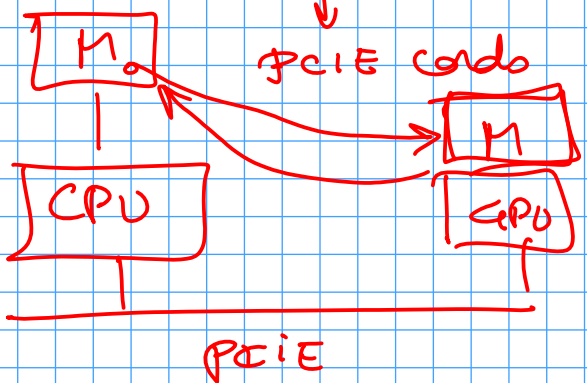
↳ GP-GPU

↓

COPROCESSOR

↓

PCIE cards



(O(1000) cores)

SIMD

nVidia GPUs

high end (C2050 tesla ~500 cores

2 DMA engines

transfer host2device

H2D

kernel (comp)

D2H

<<< ... >>>

CUDA (nVidia)

OpenCL (open standard Apple, Intel, ...)

↳ only languages for GPUs

cheaper gpu

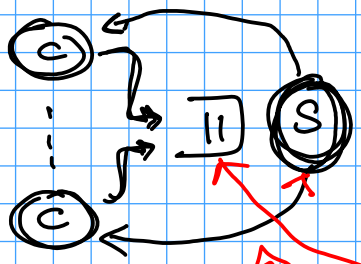
GTX 660 (Kepler)

higher number of cores

960 cores

1 DMA engine

CLIENT / SERVER



Sequential server

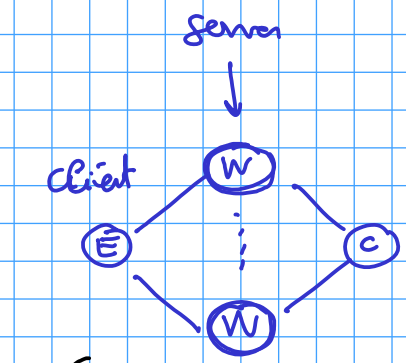
server socket

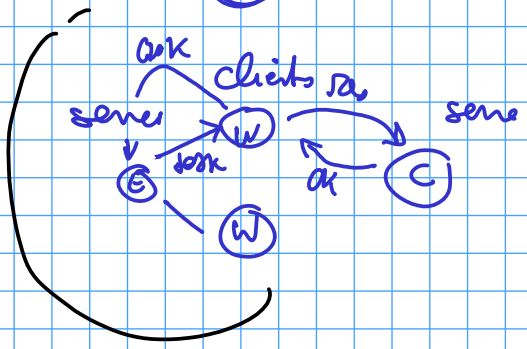
```
socket(   )
bind (addr  )
listen(   )
```

client socket

```
cs = accept(   )   blocking call
```

```
process request
(addr cs)
send result
(on cs)
```



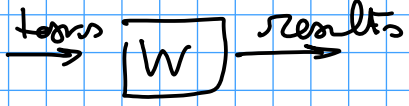



```
while(  ) {
    (cs) = accept(    );
    req  = read(cs ...  );
    answ = solve(req);
    write(cs, answ ...)
}
```

multithreaded server

```
while (...) {
    cs = accept( ... );
    < Fork a CA(cs) dealing with the request
      coming from cs (A to Z) >
}
```

```
req = read(cs ...);
answ = solve(req ...);
write(cs, answ ...)
close(cs);
exit;
```



(triple buffering) → good throughput

(multi threaded server) → [2]

↓ at a price of?

(time (by os) to solve contention)

excess parallelism

↓

popular for distributed memory (cow & now)

(may pose extra requirement on some resources (FP-ALU, memory subsystem))