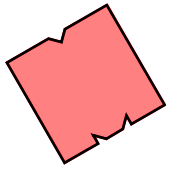
CS-590.26, Spring 2014

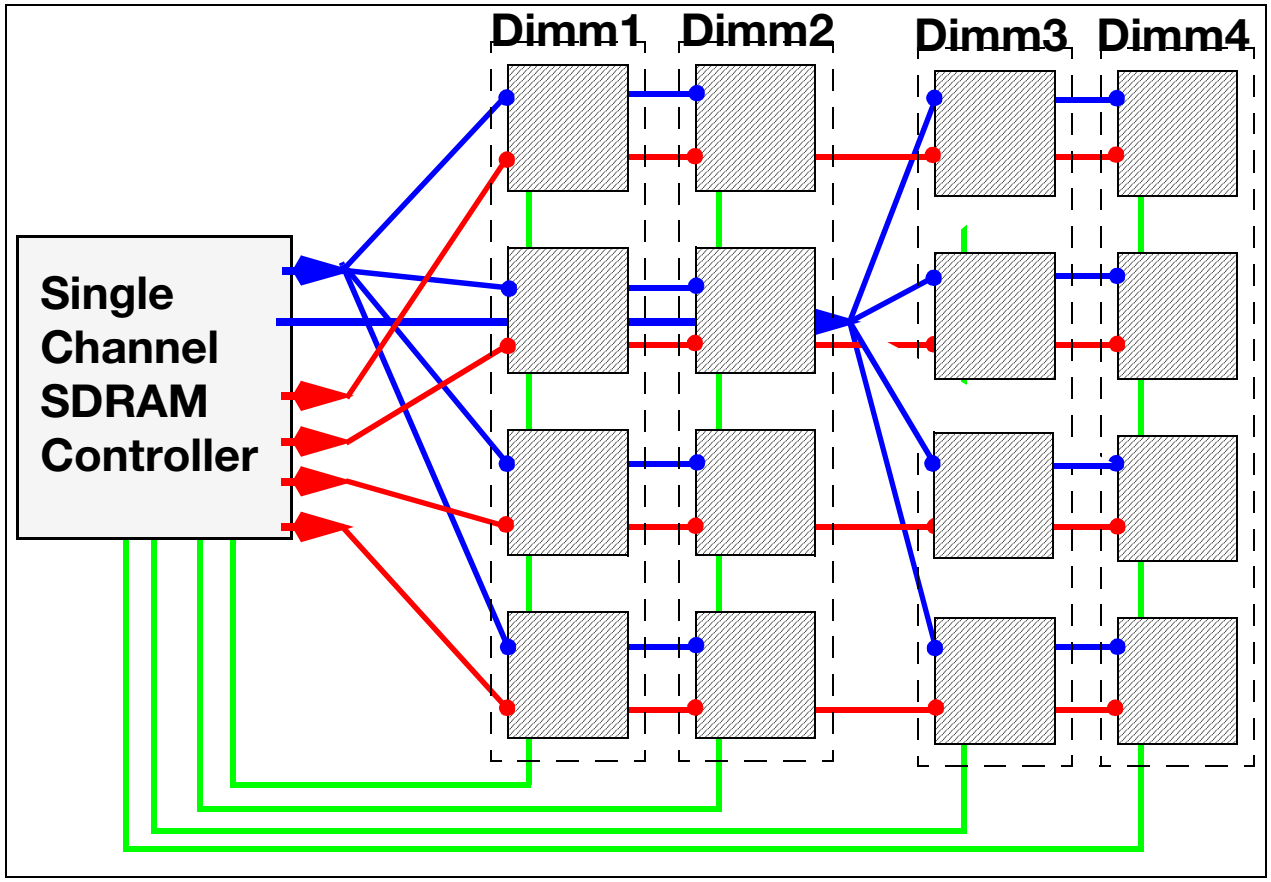
High Speed Memory Systems: Architecture and Performance Analysis

Memory System Organization and System Controller

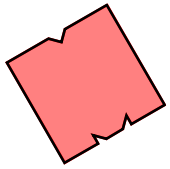
Credit where credit is due:
Slides contain original artwork (© Jacob, Wang 2005)



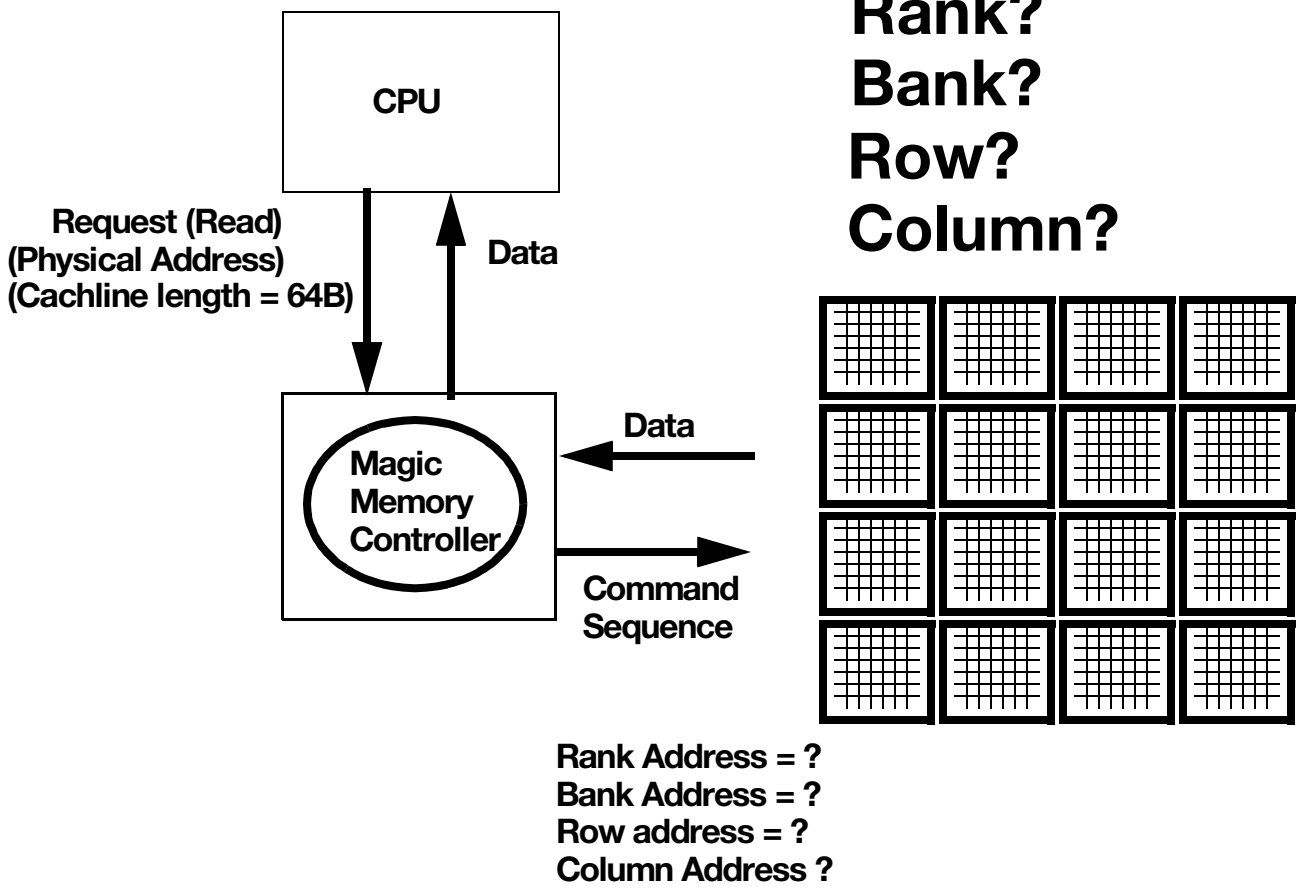
Memory System Organization



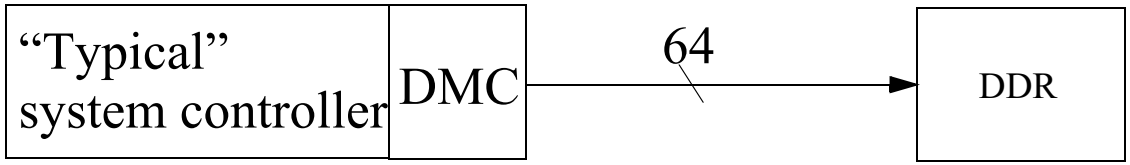
"Mesh Topology"



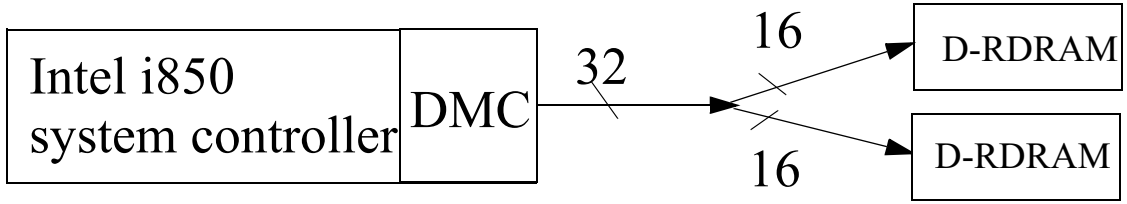
Where is the data?



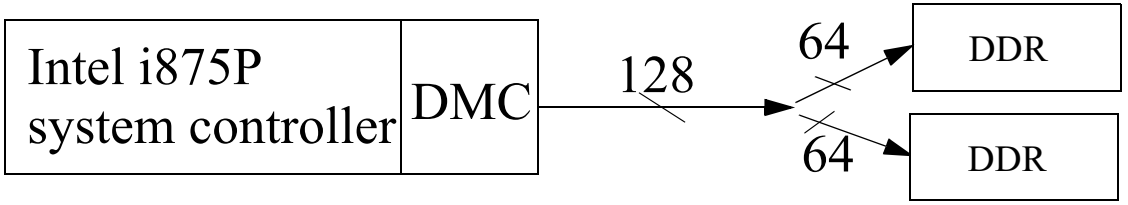
Channel I



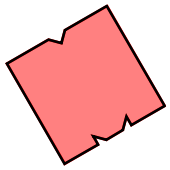
"PC Class" memory system.
1 physical channel of DDR SDRAM



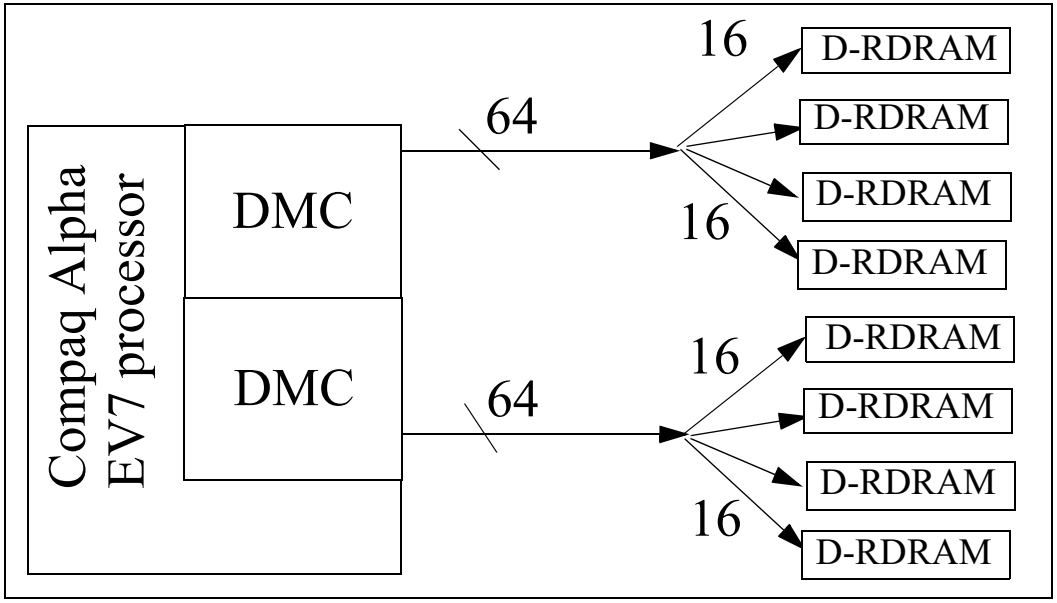
Intel i850 DRDRAM memory system.
2 physical channel. 1 logical channel



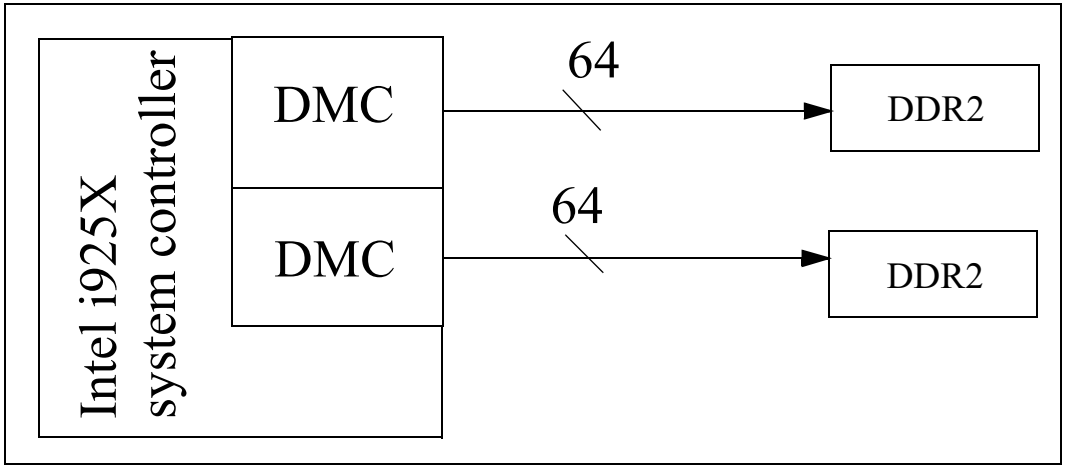
Intel 875P DDR SDRAM memory system.
2 physical channel. 1 logical channel



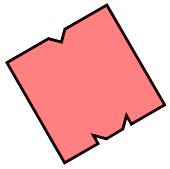
Channel II



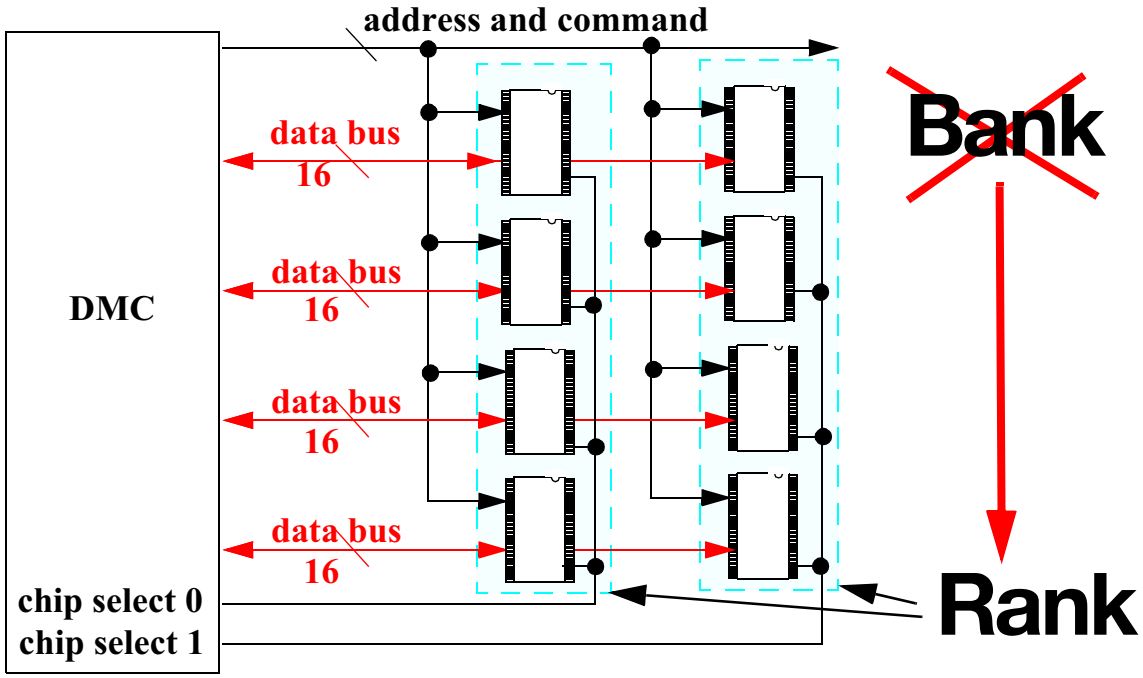
Two Channels: 64 bit wide per channel



Two Channels: 64 bit wide per channel



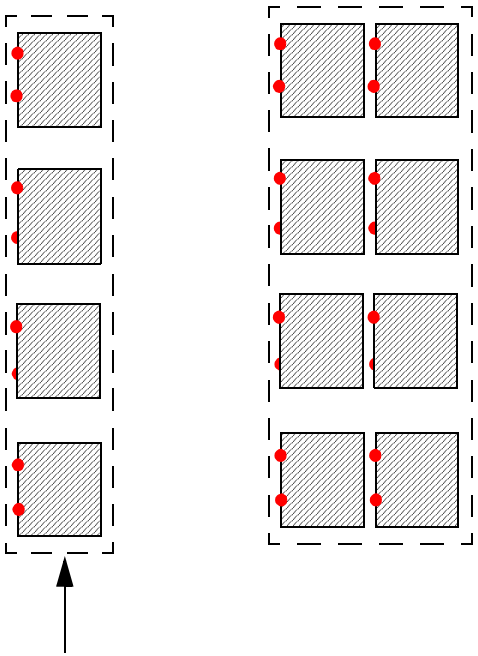
Rank I



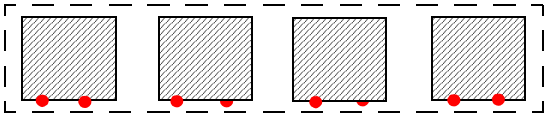
It's a "bank" of chips that responds to a single command and returns data.

"Bank" terminology already used.

Rank II



SDRAM
Double Sided Dimm
Two Ranks



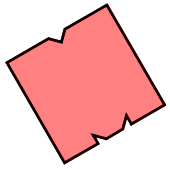
Rambus RIMM

Rank Count is Number of Devices

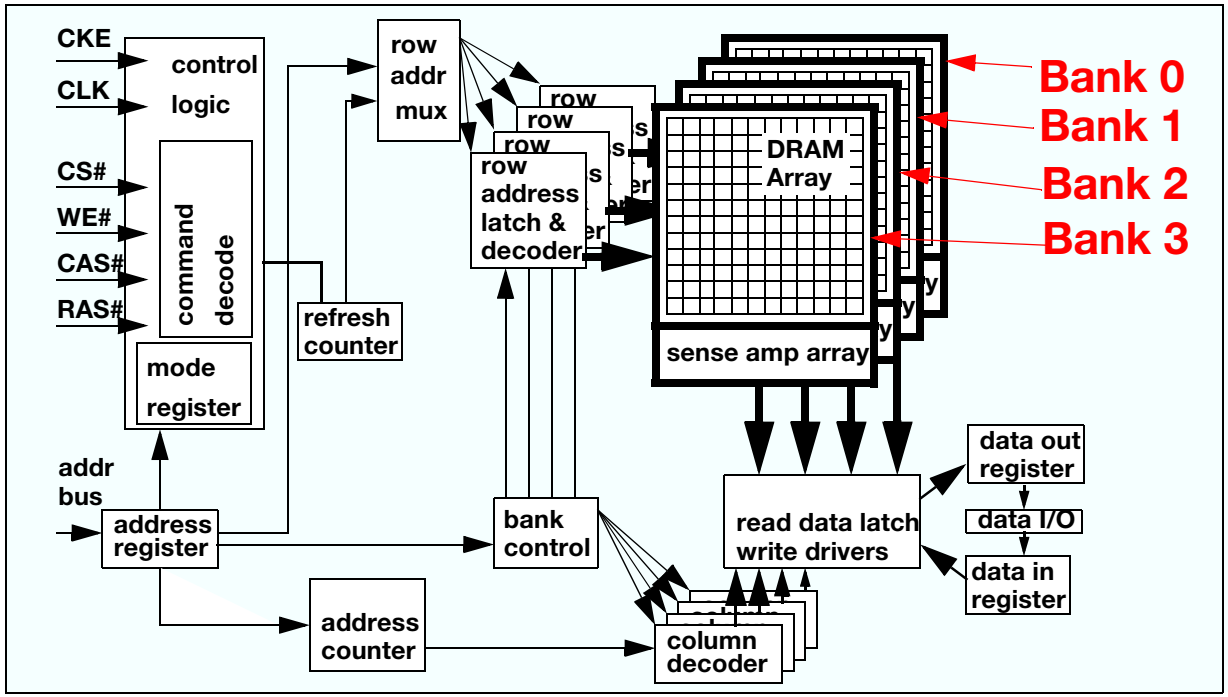
SDRAM
Single Sided Dimm
One Rank

SDRAM/DDR SDRAM system: 1~6 ranks

RDRAM system: <= 32 ranks



Bank

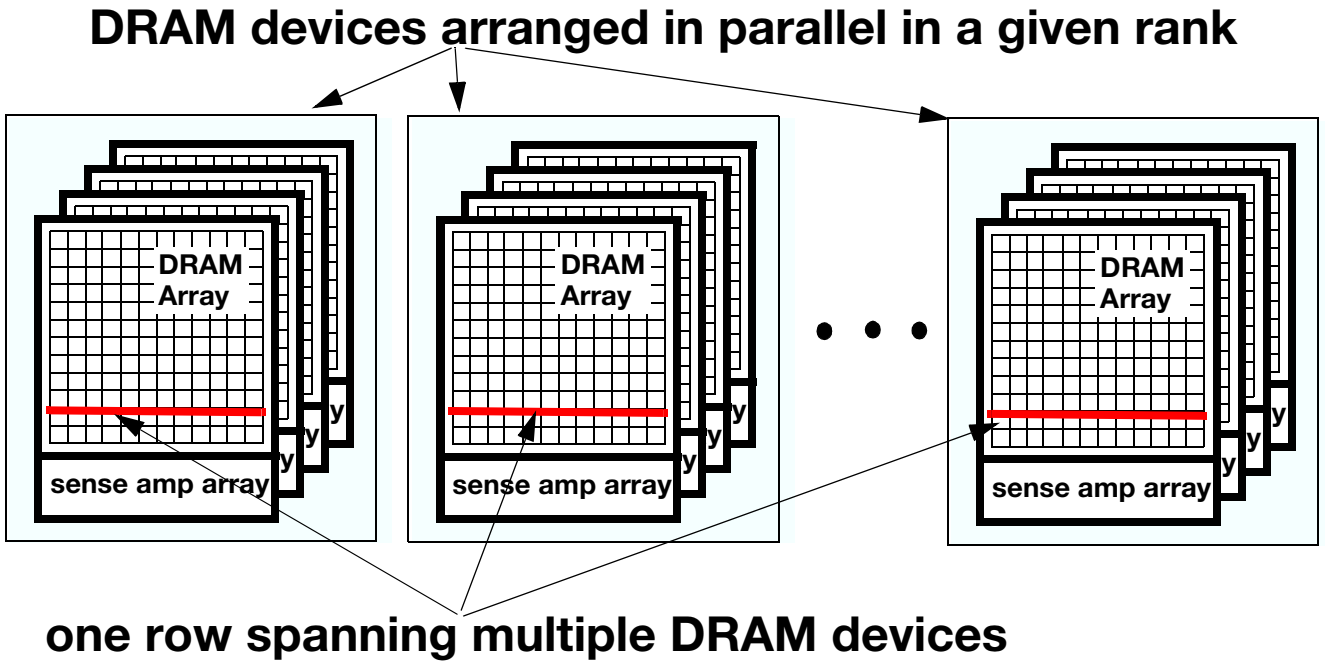


"Banks" of indepedent memory arrays inside of a DRAM Chip

SDRAM/DDR SDRAM system: 4 banks

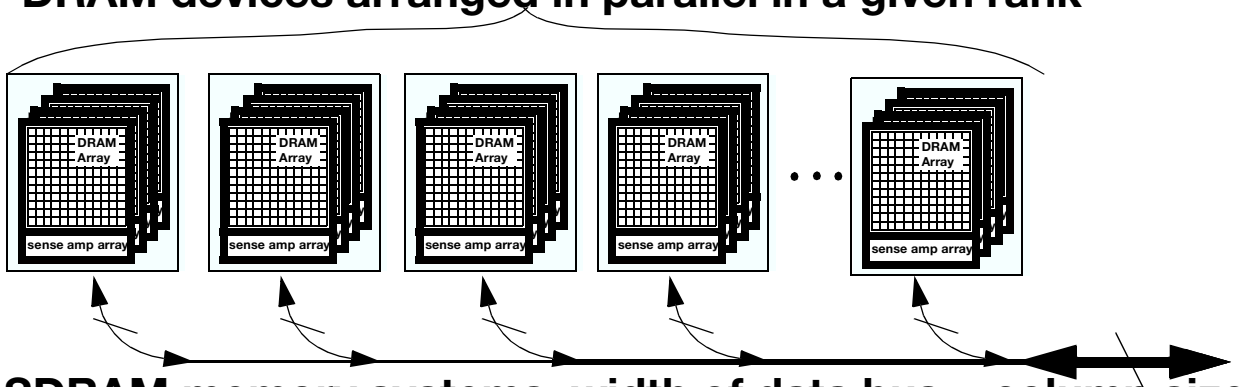
RDRAM system: "32" split or 16 full banks

Row


DRAM devices arranged in parallel in a given rank
DRAM
Array
sense amp array
DRAM
Array
sense amp array
DRAM
Array
sense amp array
one row spanning multiple DRAM devices

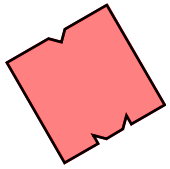
Column

DRAM devices arranged in parallel in a given rank



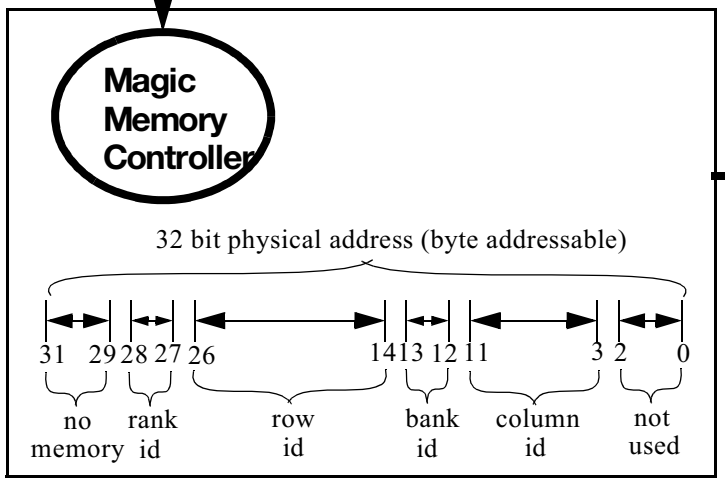
SDRAM memory systems: width of data bus = column size

Column = Smallest unit of data moved in memory system

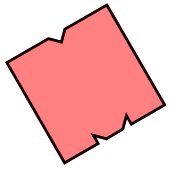


Where's the data? Part 1

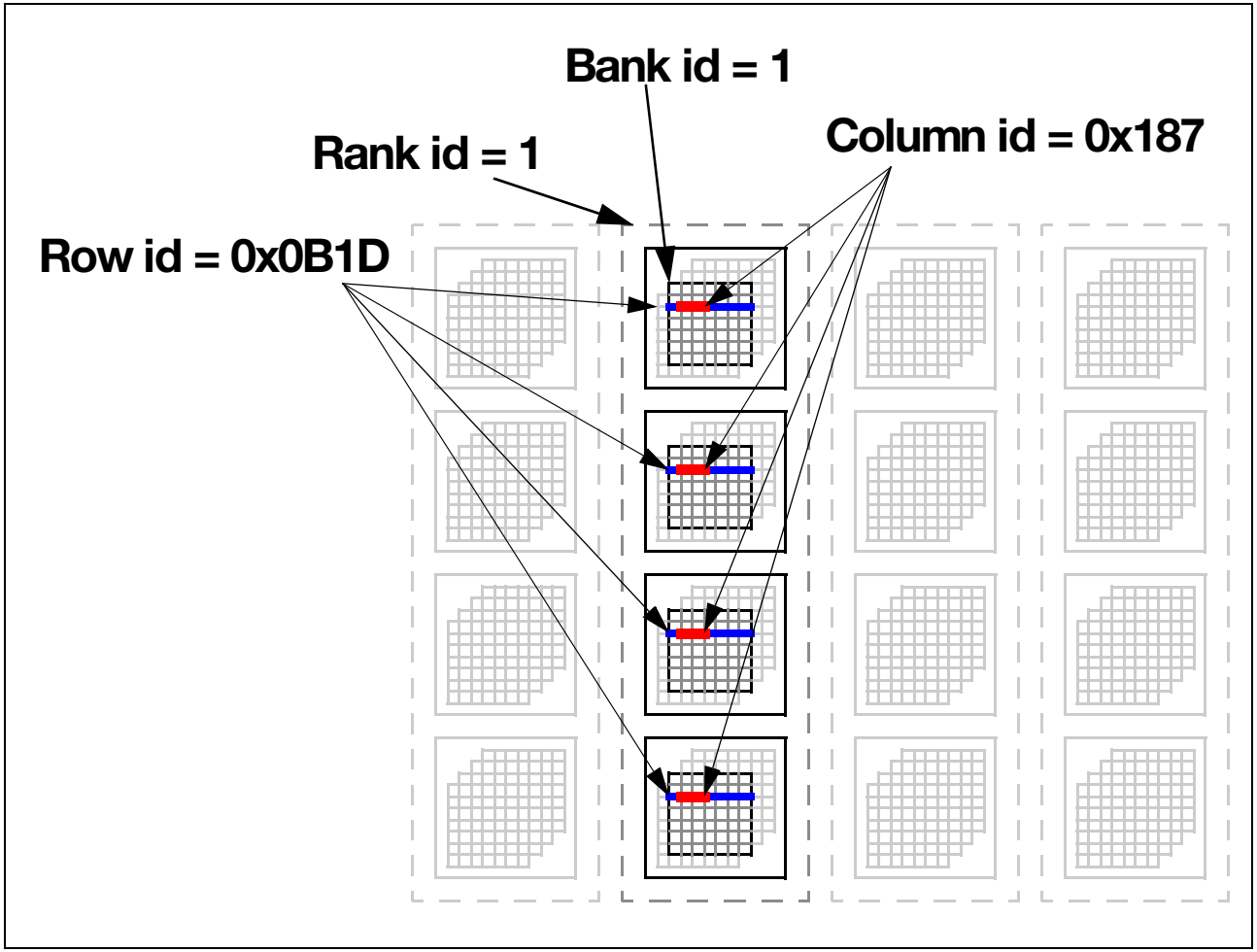
Read Request
Physical Address:
0x0AC75C38 = 0b 0000 1010 1100 0111 0101 1100 0011 1000
= 000 01 0101100011101 01 110000111 000
rank row bank column



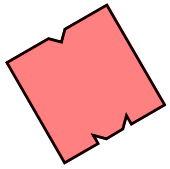
Rank id = 1
Bank id = 1
Row id = 0x0B1D
Column id = 0x187



Where's the data? Part 2

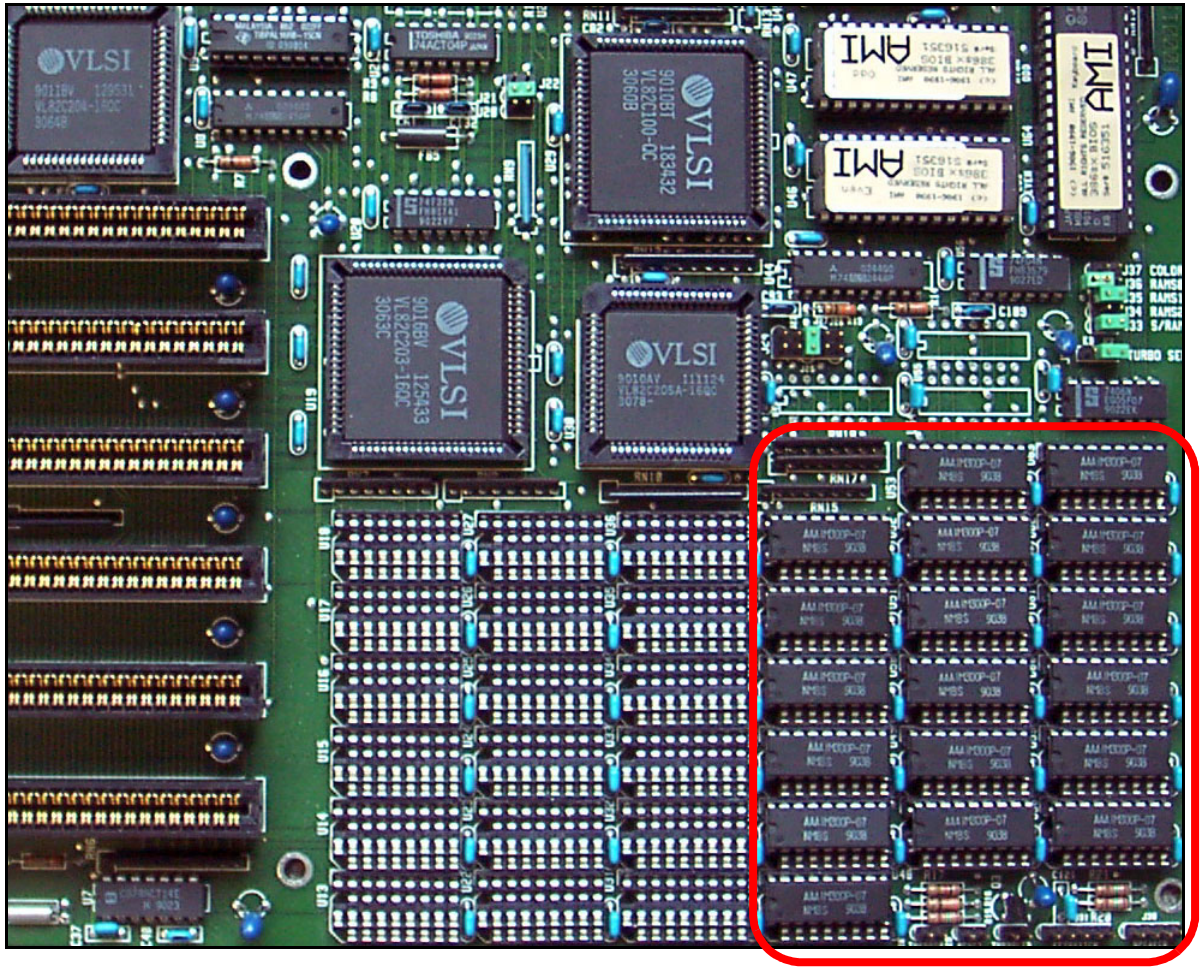


FPM / EDO / SDRAM / etc.



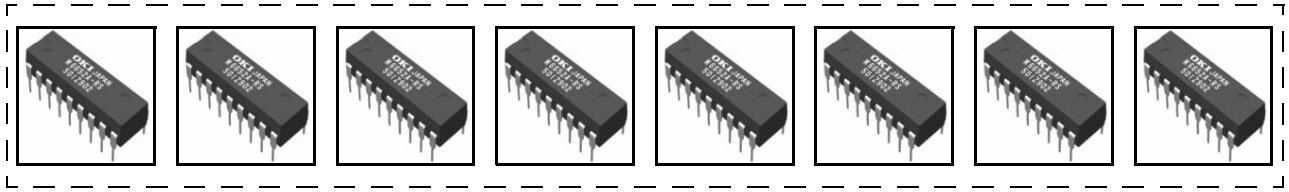
Bare Chips

Bare DIP's shoved into sockets
18 Chips, each x1, 18 bit wide data bus

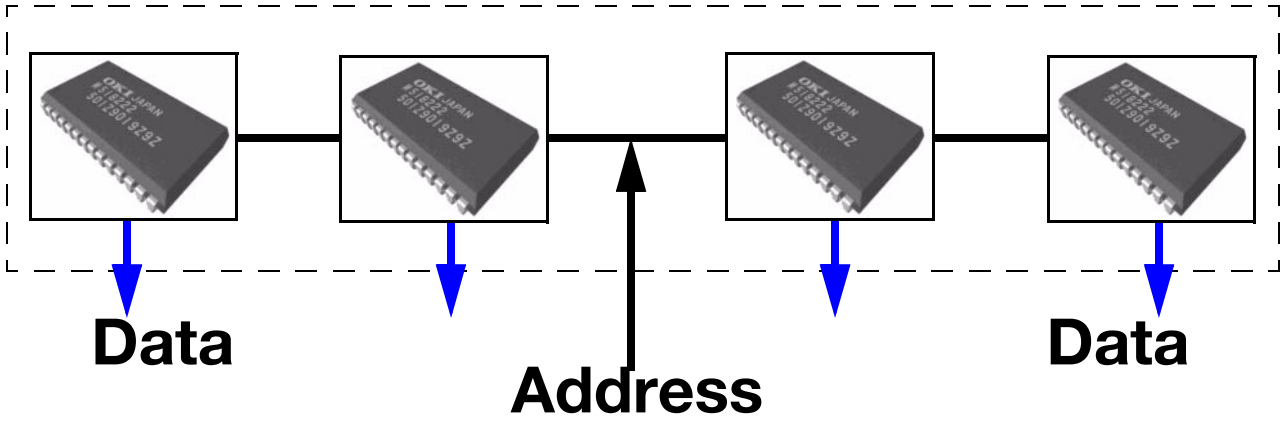


Memory Modules I

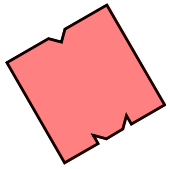
Organizing chips into modules



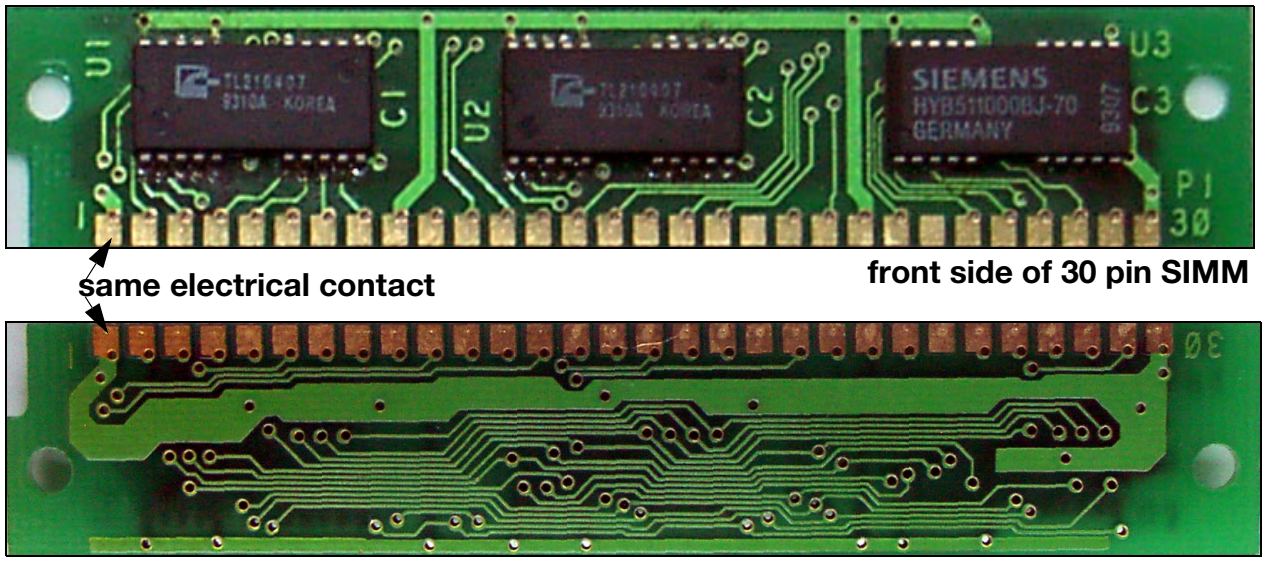
Put chips on PCB, make a module



FPM / EDO / SDRAM / etc.

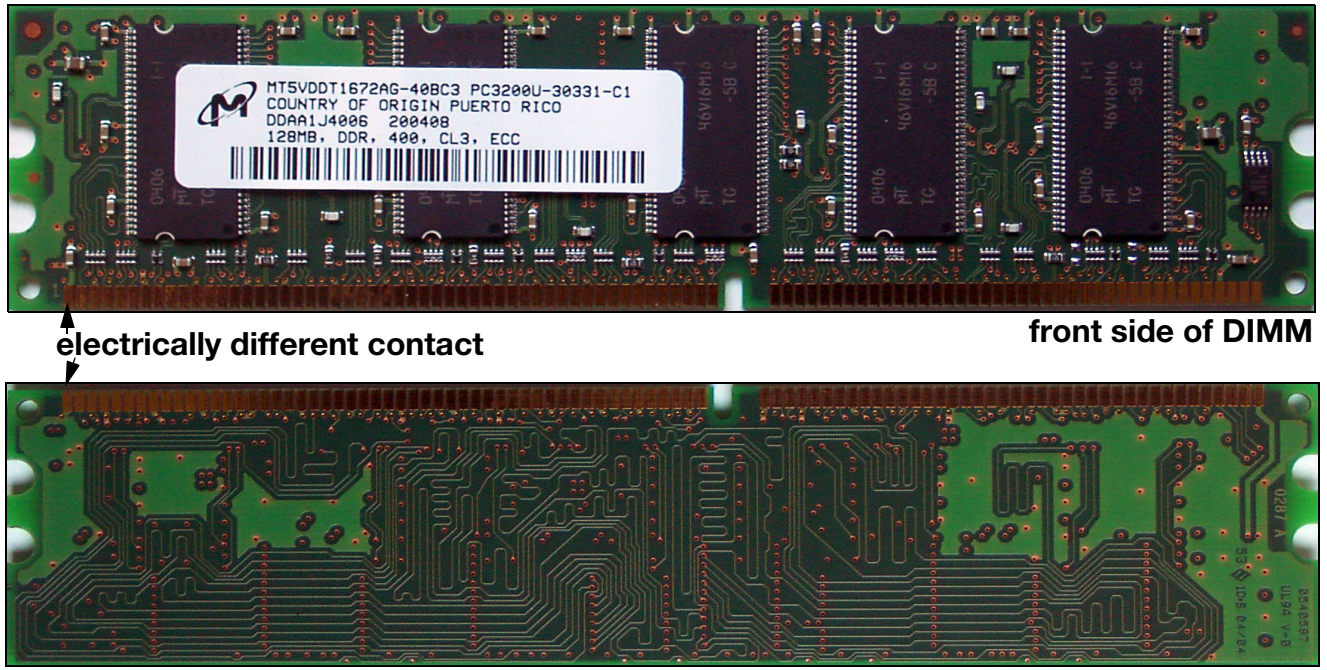


Memory Modules II



Single Inline Memory Module

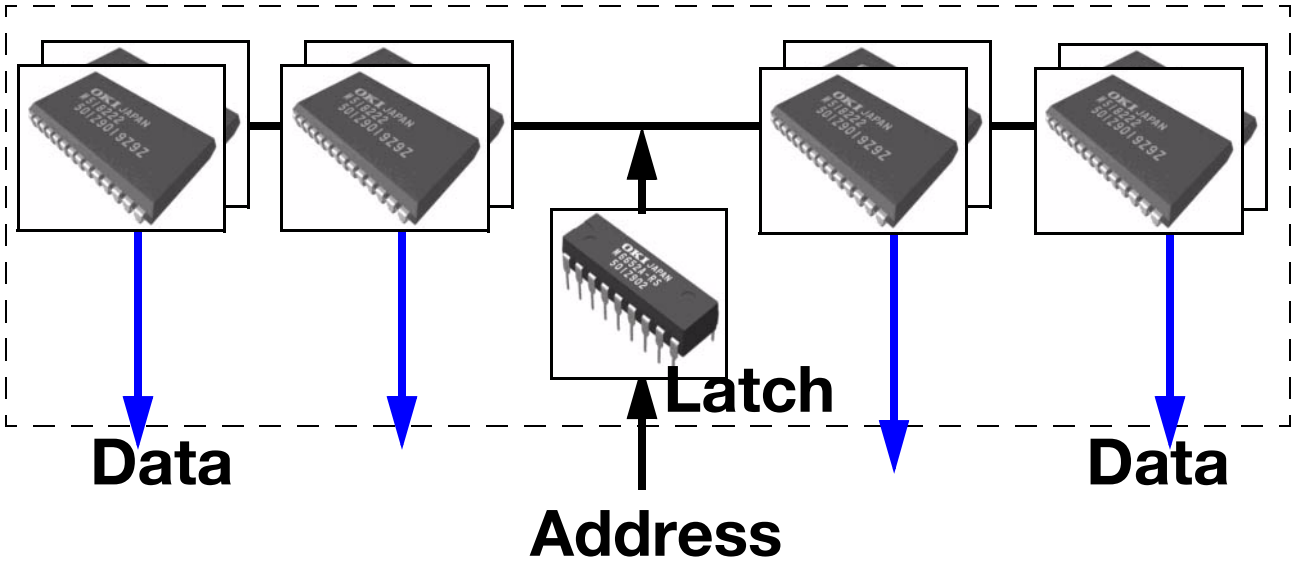
Memory Modules III



Dual Inline Memory Module

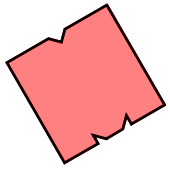
Memory Modules IV

Registered DIMM



One extra cycle to buffer and distribute address.

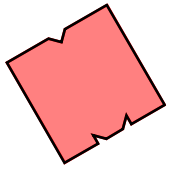
More chips (load) can be placed on module



Memory Modules V

Capacity	device density	number of ranks	devices per rank	device width	number of banks	number of rows	number of columns
128 MB	64 Mbit	1	16	x4	4	4096	1024
128 MB	64 Mbit	2	8	x8	4	4096	512
128 MB	128 Mbit	1	8	x8	4	4096	1024
128 MB	256 Mbit	1	4	x16	4	8192	512

Four different configurations for a 128 MB SDRAM DIMM

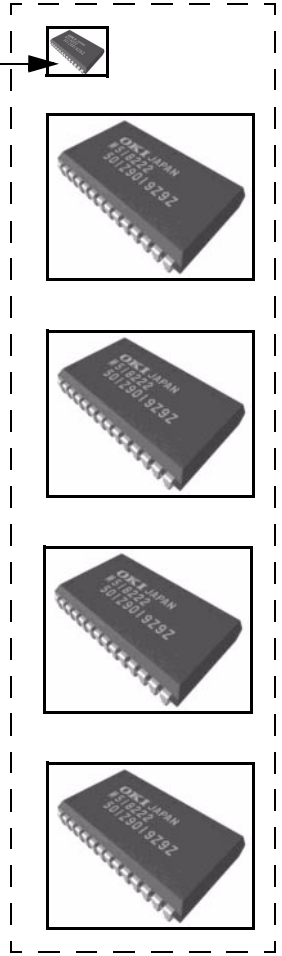


SPD: Serial Presence Detect

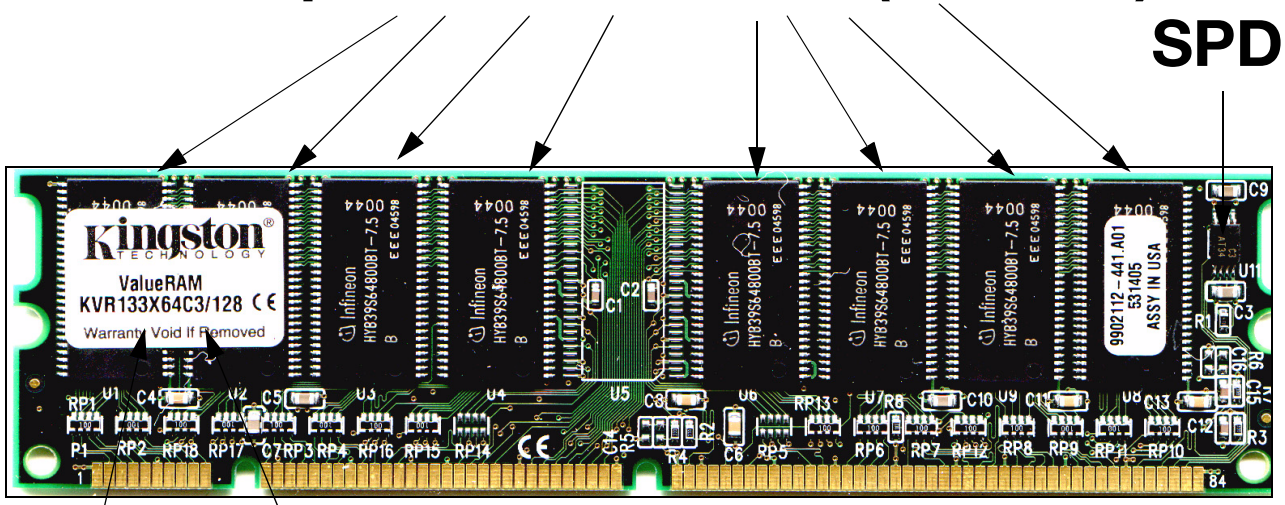
SPD: Tiny EEPROM

Contains Parameters

- **Speed settings**
- **Configurations**
- **Programmed by module maker**

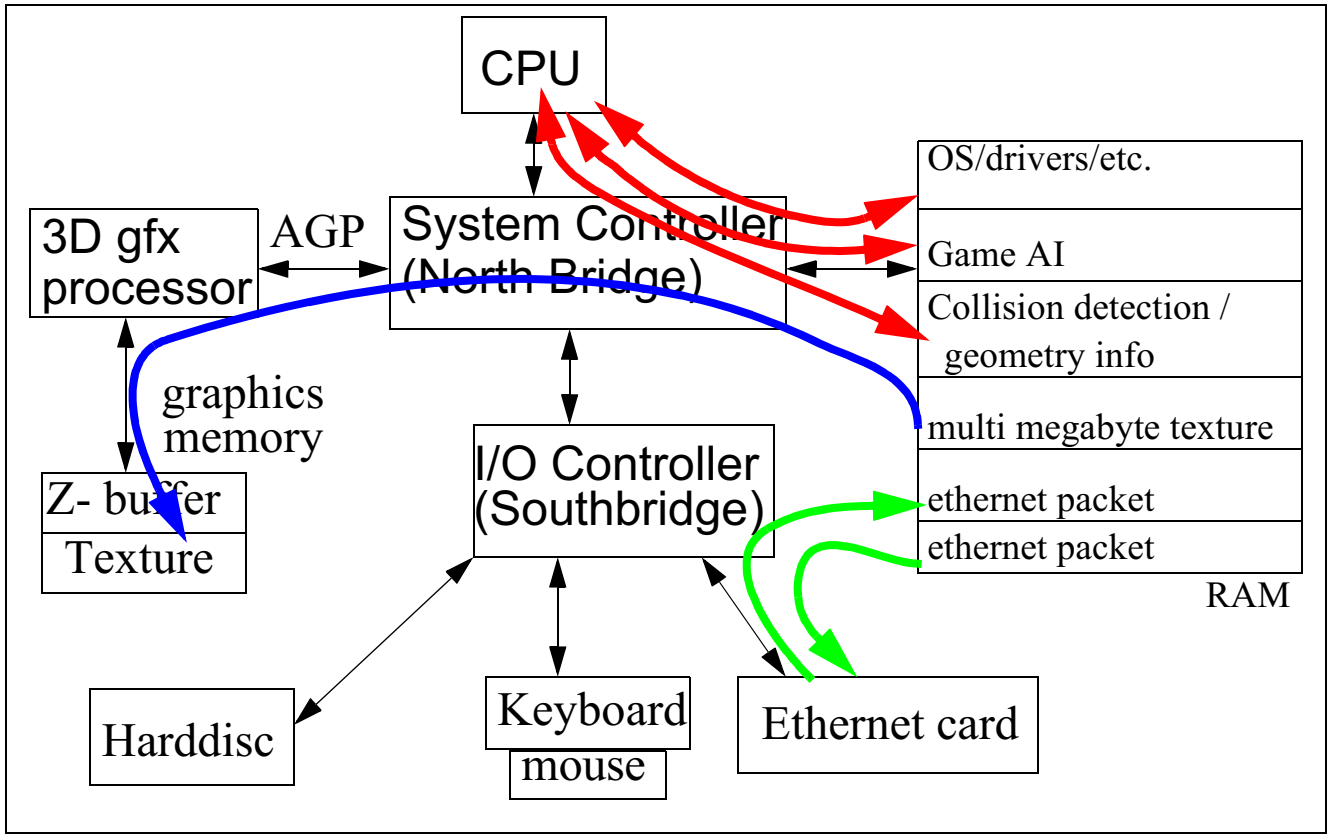


Kingston SDRAM DIMM



Dual Inline Memory Module

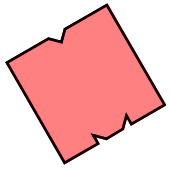
System Controller



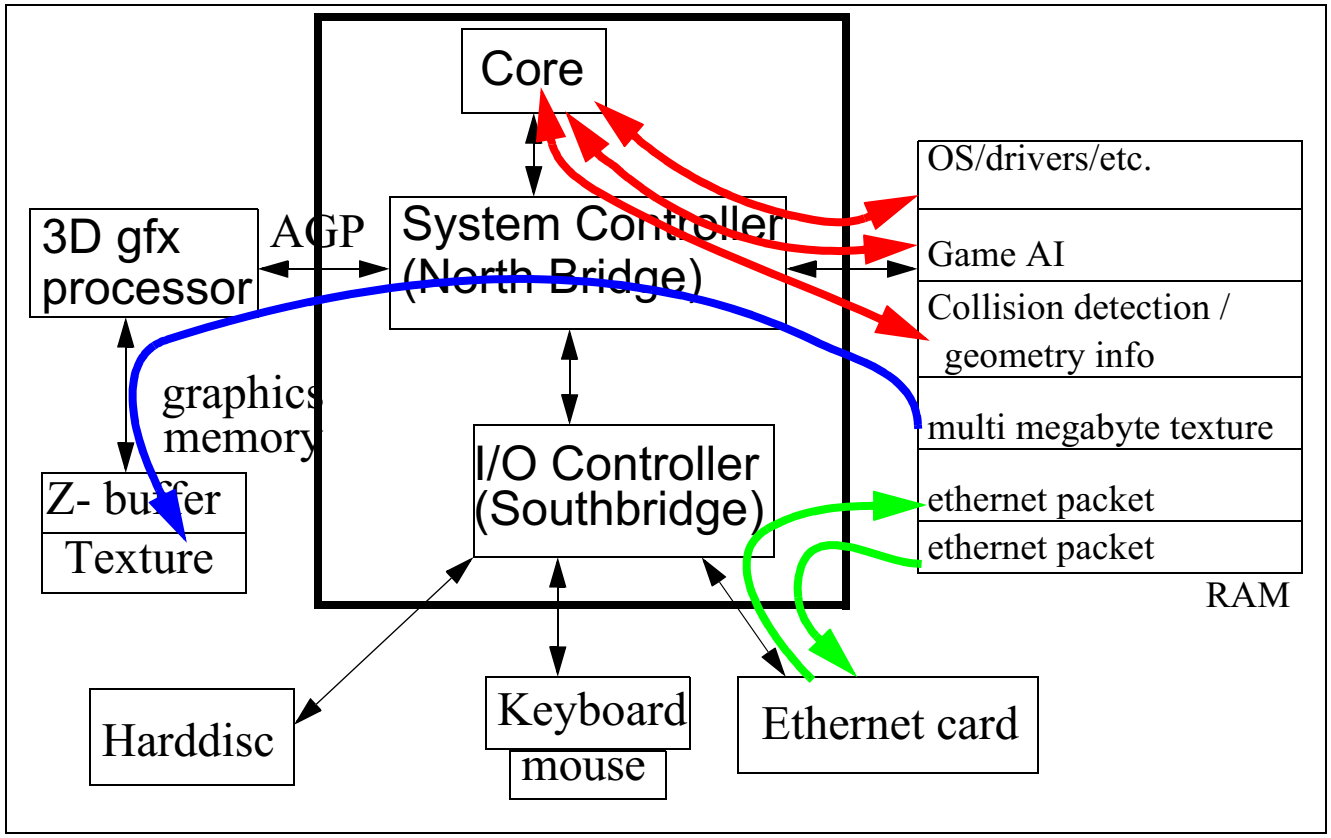
Heavy demand placed on memory system

Heavier still in SMP/SMT/CMP system

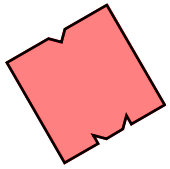
System Controller == System traffic cop



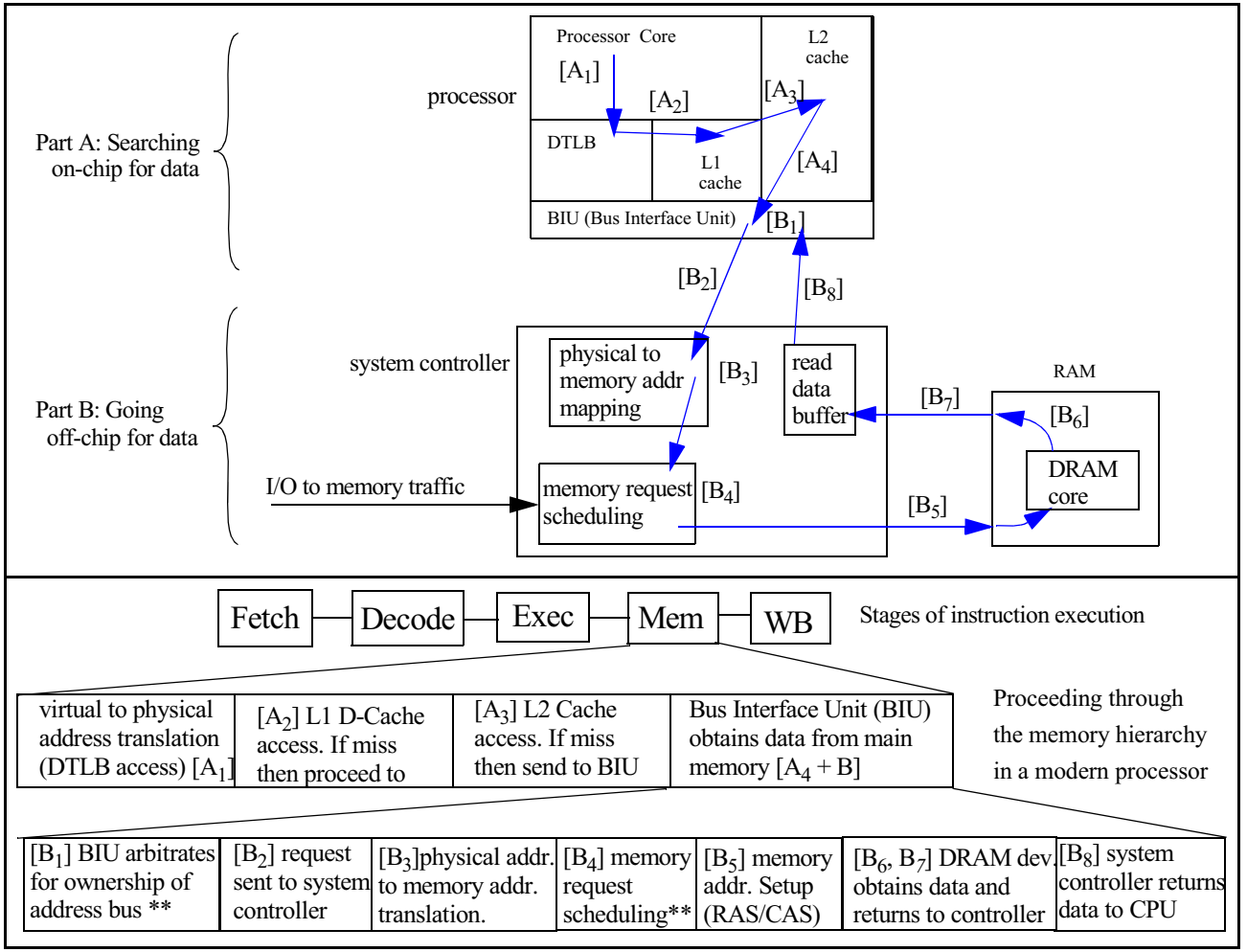
System Controller



Problem remains (exacerbated?) even if controller integrated onto CPU

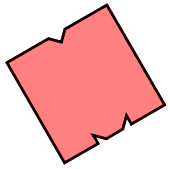


Memory Request Overview

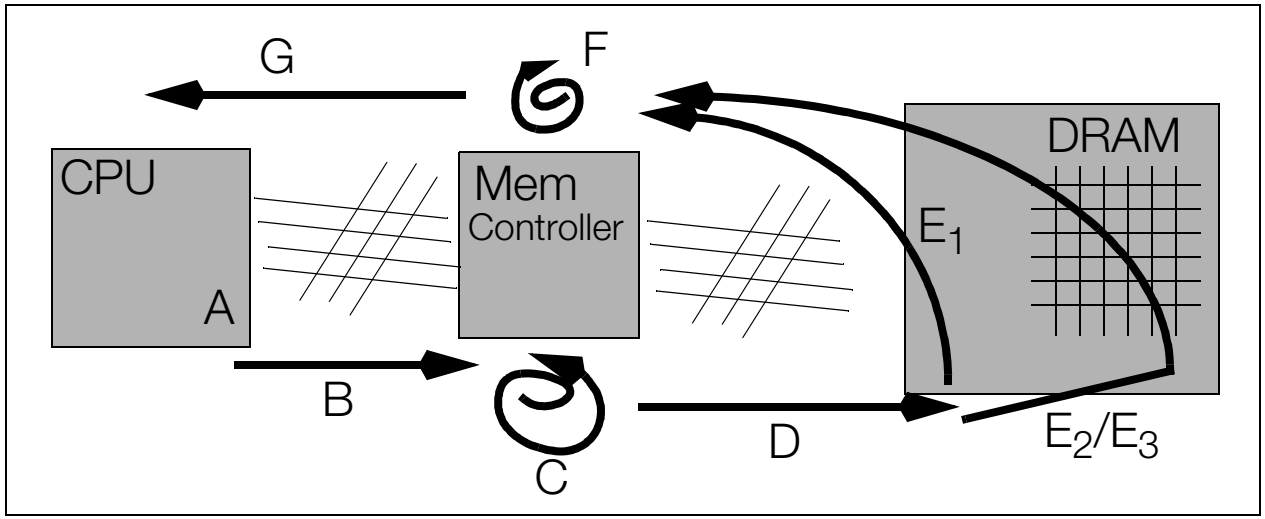


** Steps not required for some processor/system controllers. protocol dependant.

Progression of a Memory Read Transaction Request Through Memory System

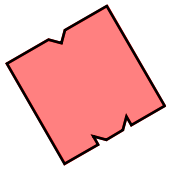


"Memory Latency"

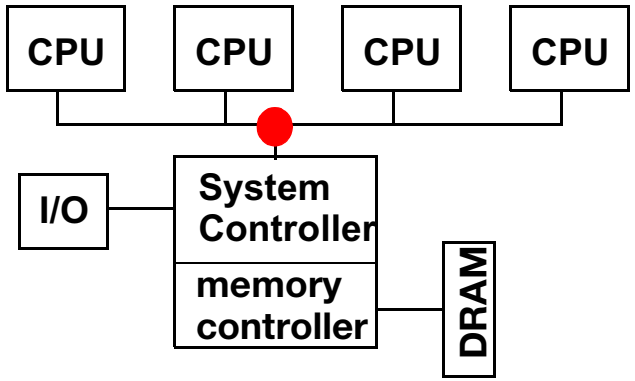


A: Transaction request may be delayed in Queue
B: Transaction request sent to Memory Controller
C: Transaction converted to Command Sequences
(may be queued)
D: Command/s Sent to DRAM
E_1: Requires only a **CAS** or
E_2: Requires **RAS + CAS** or
E_3: Requires **PRE + RAS + CAS**
F: Data is staged at controller
G: Transaction sent back to CPU

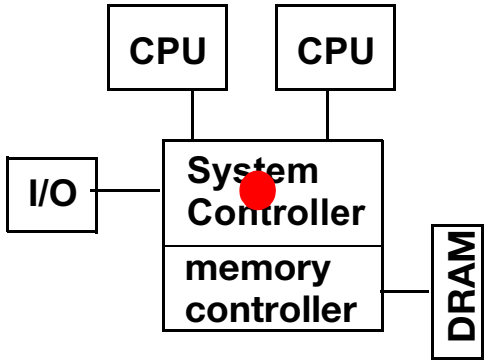
"DRAM Latency" = A + B + C + D + E + F + G



Small System Topologies

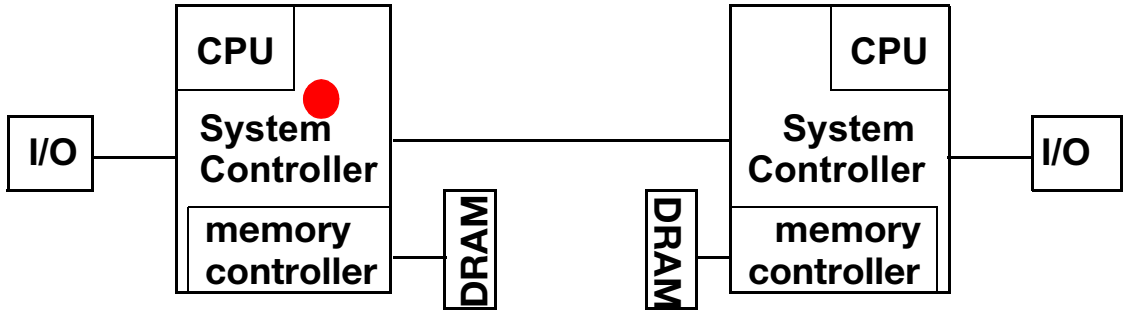


Classic small system topology
(Lots of systems)
(including multicore + on-chip MC)



Point-to-point processor-controller system topology

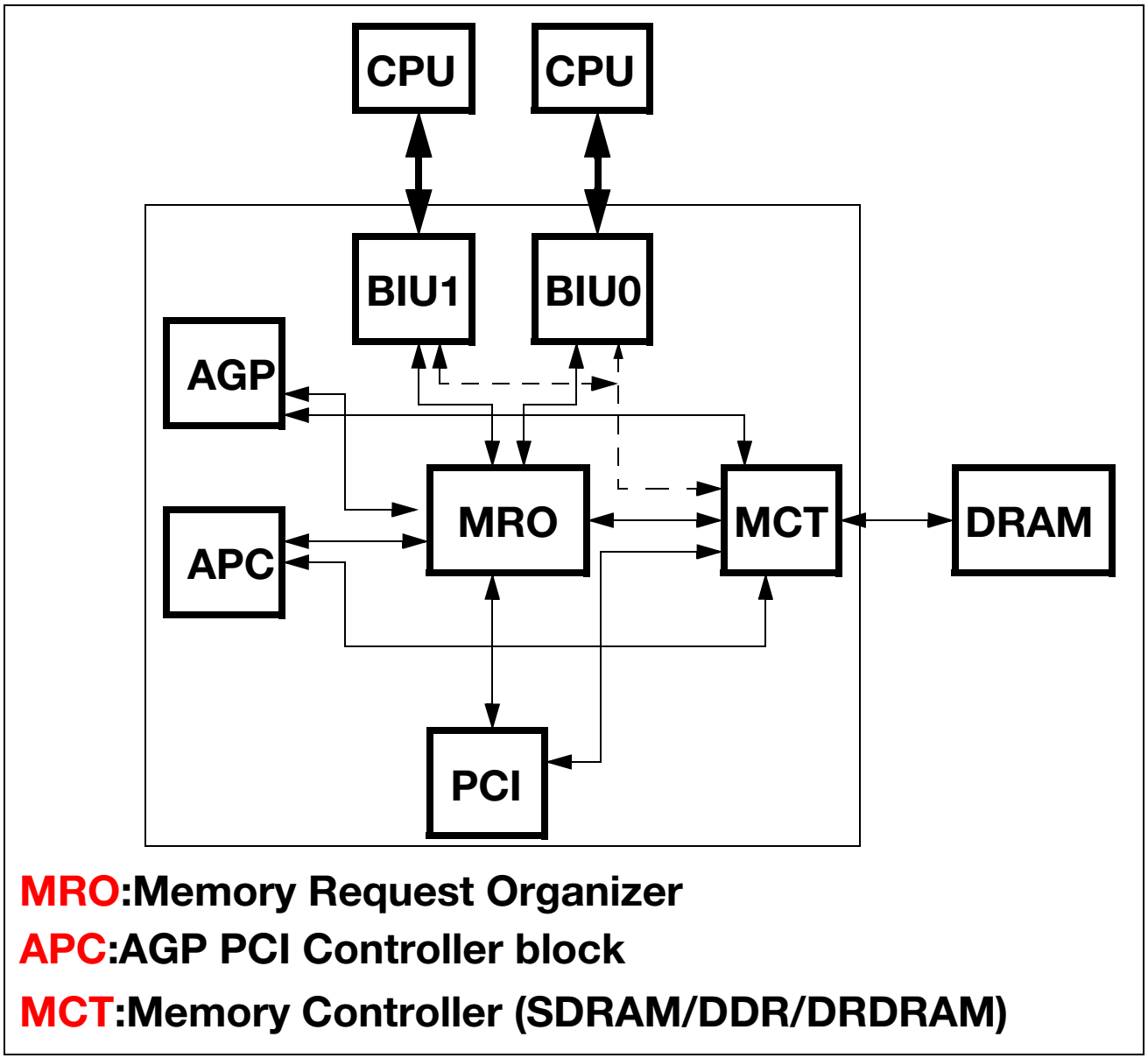
(AMD Athlon/Alpha EV6/PPC 970)

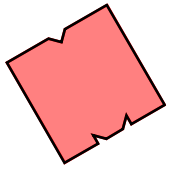


Integrated system controller system topology
(AMD Opteron/Alpha EV7 etc.)

represents point of synchronization*. (for local access)

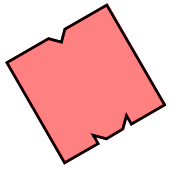
System Controller: Athlon





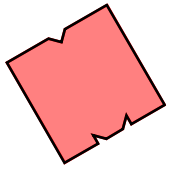
MRO: Memory Request Organizer

- Request crossbar responsible for scheduling memory read and write requests from BIU, PCI, AGP
- Serves as the coherence point
- Requests are reordered to minimize page conflict and maximize page hits
- Anti-starvation mechanism by aging of entries
- Arbitration bypassed during idle conditions to improve latency

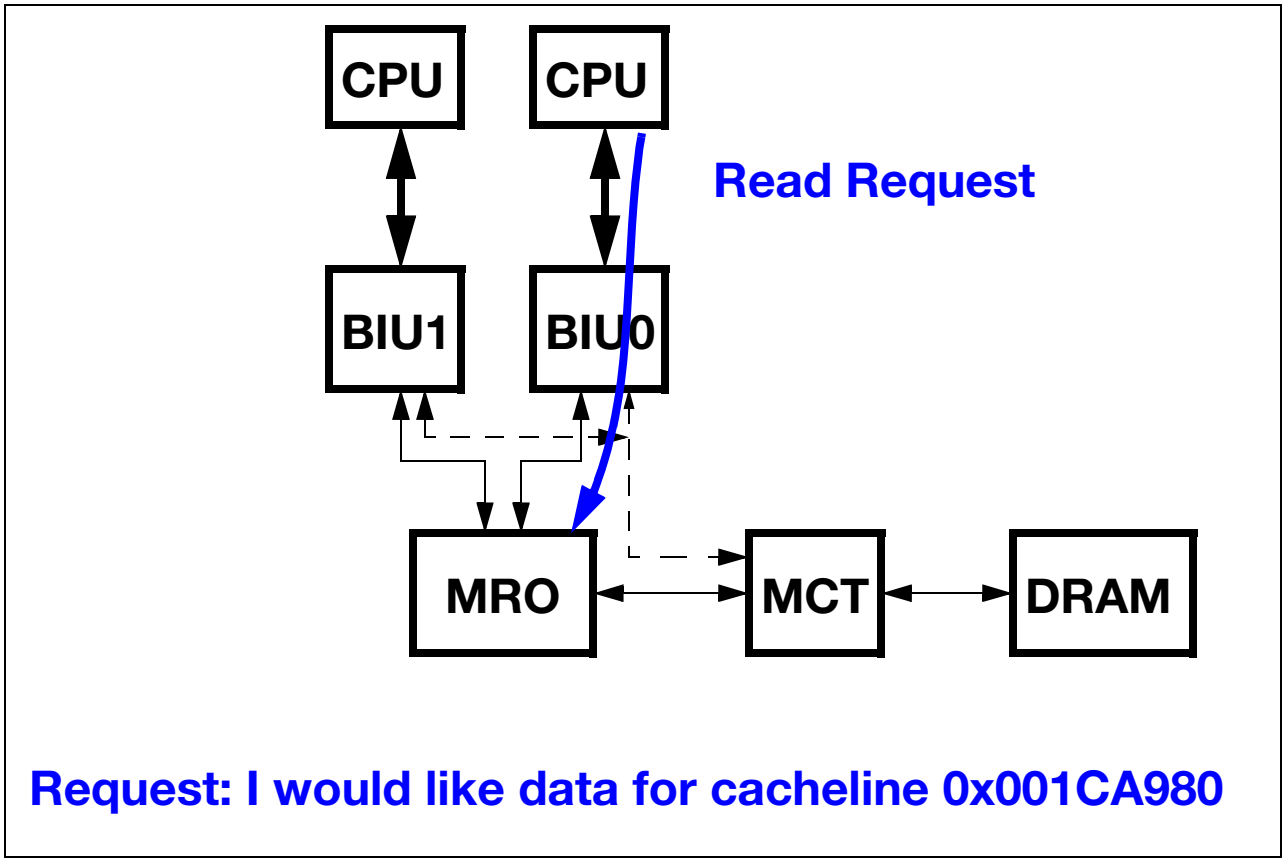


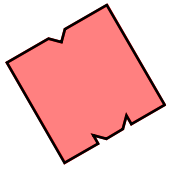
AMD Athlon Controller:

Chip Version	Tech & Voltage	Max Core Speed	Die Size (pad limited)	No. of pins
SDRAM 1P, 2xAGP	0.35um, 3.3V	100 MHz	107 mm^2	492
SDRAM, 2P, 2xAGP	0.35um, 3.3V	100 MHz	130 mm^2	656
DDR, 1P, 4xAGP	0.25um, 2.5V	133 MHz	133 mm^2	553
DDR, 2P, 4xAGP	0.25um, 2.5V	133 MHz		
RDRAM, 1P, 4xAGP	0.25um, 2.5V	133 MHz	107 mm^2	492

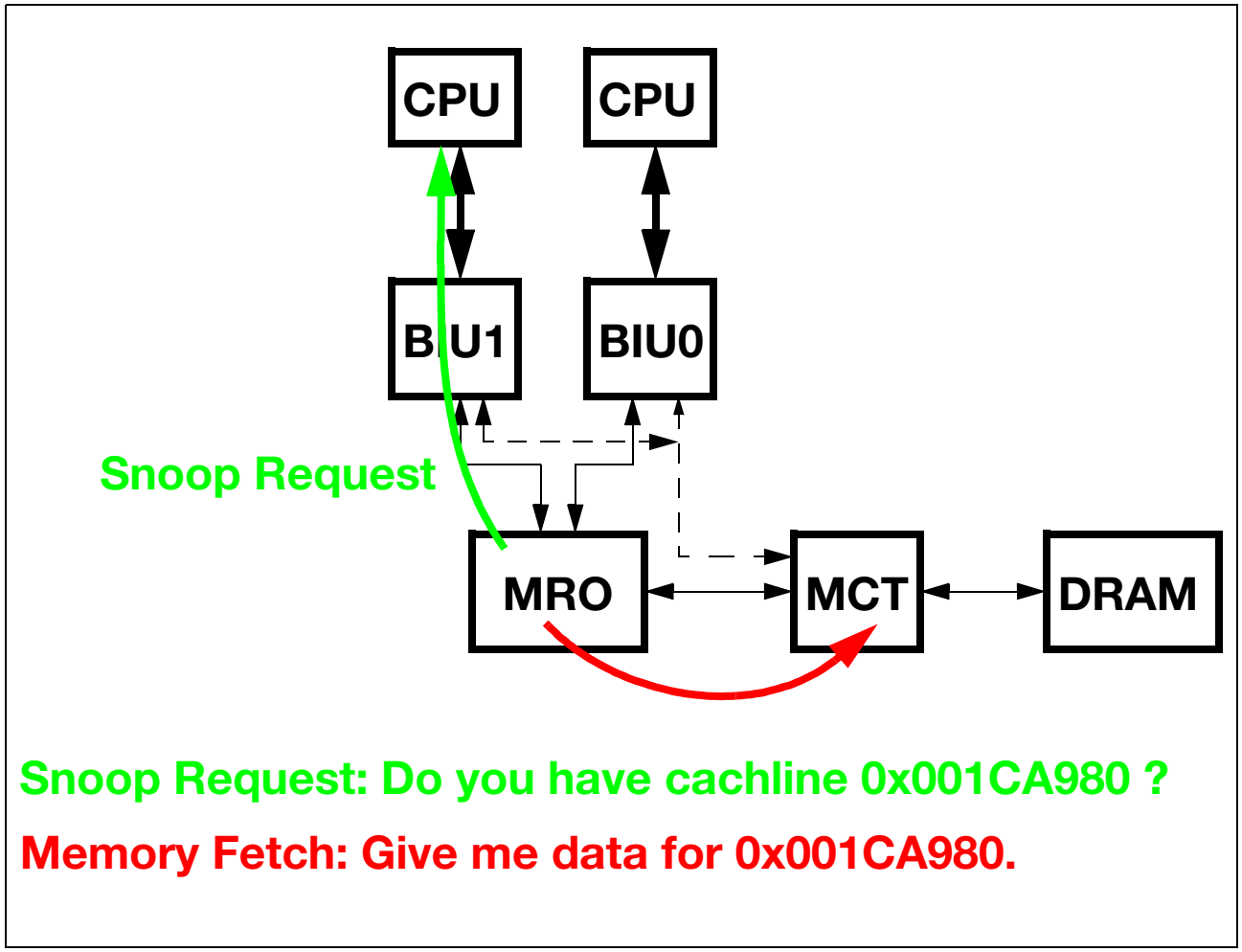


Cache Coherency I

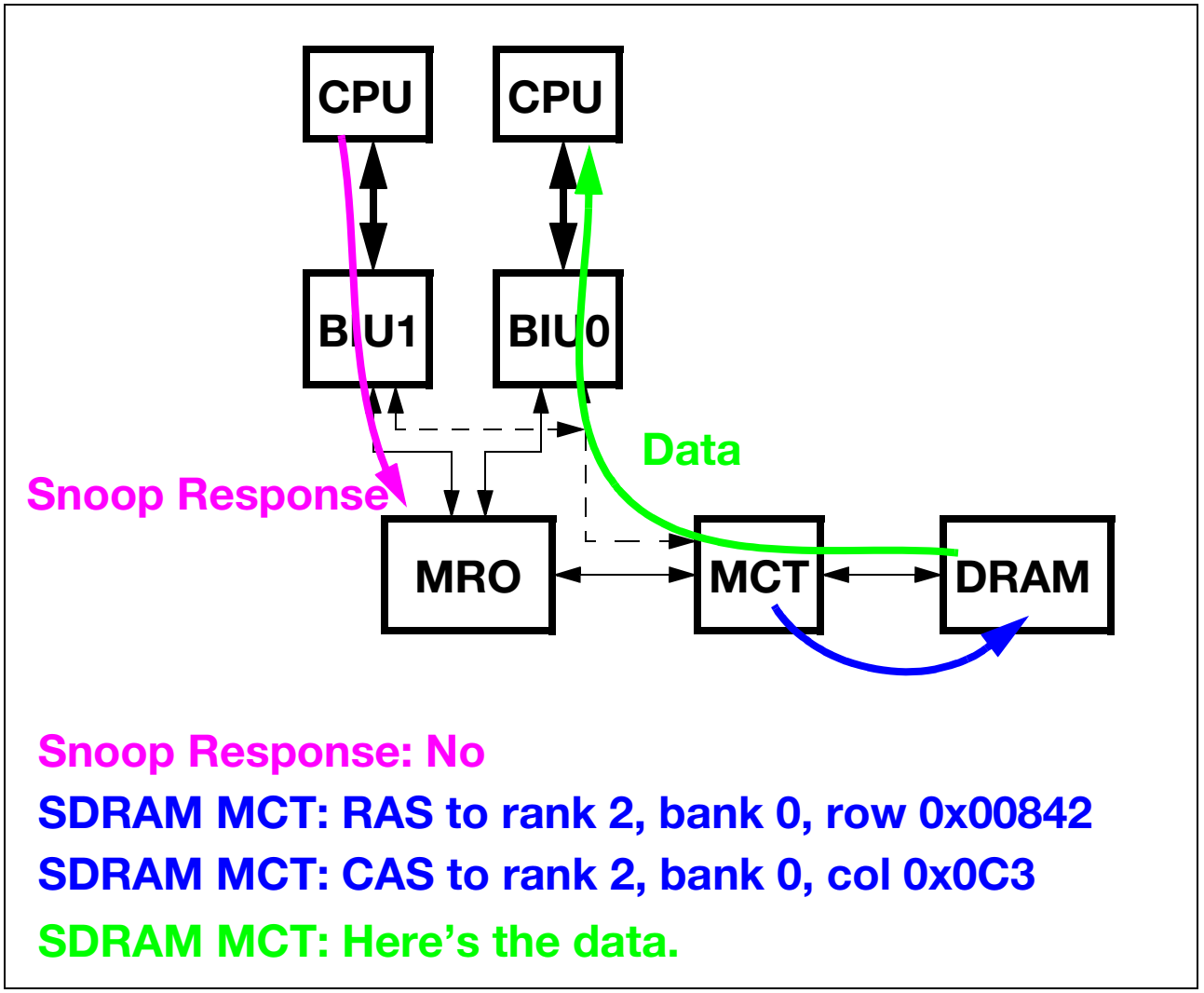


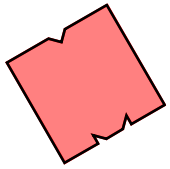


Cache Coherency II

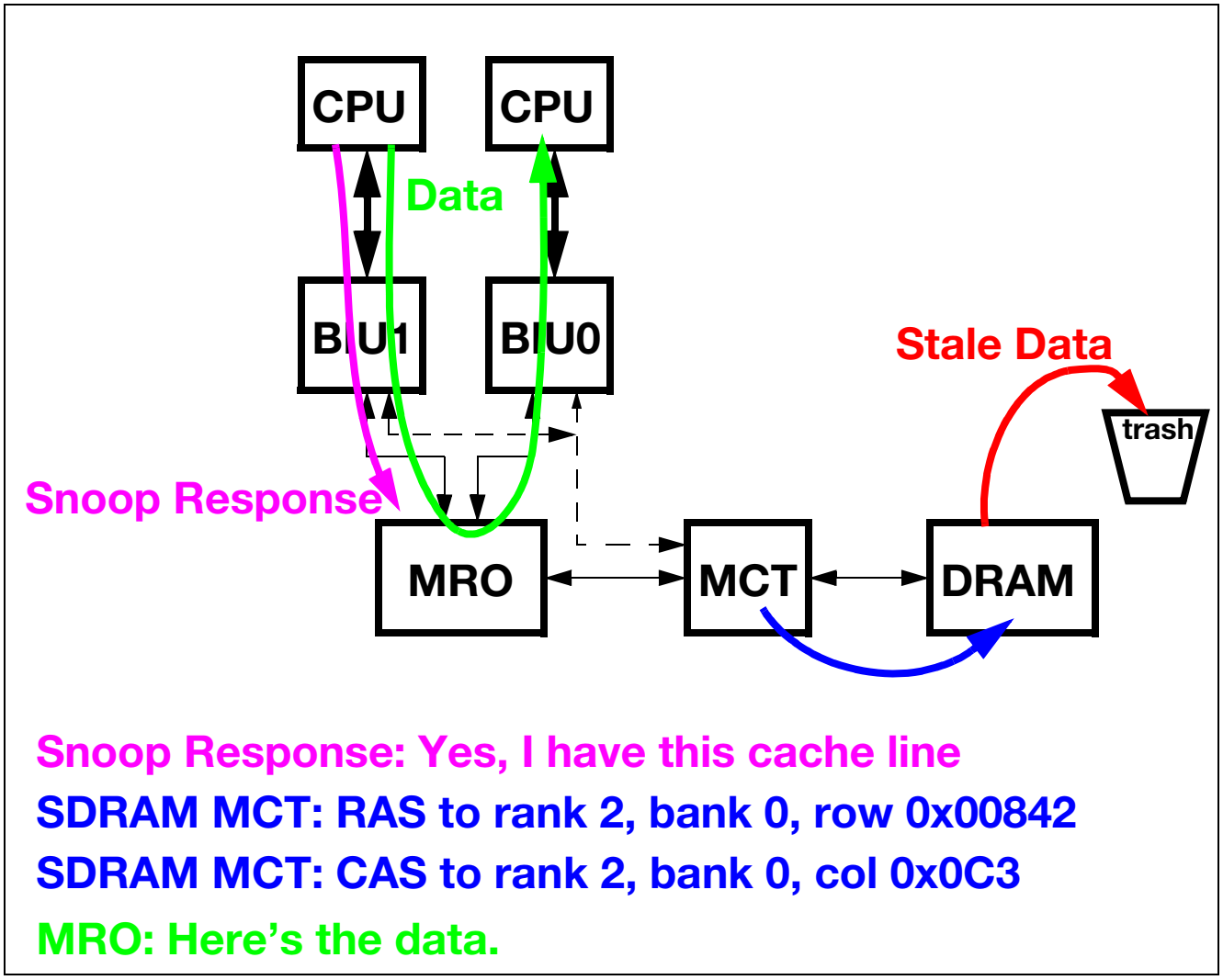


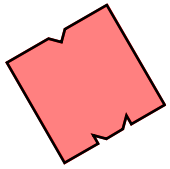
Cache Coherency IIIa



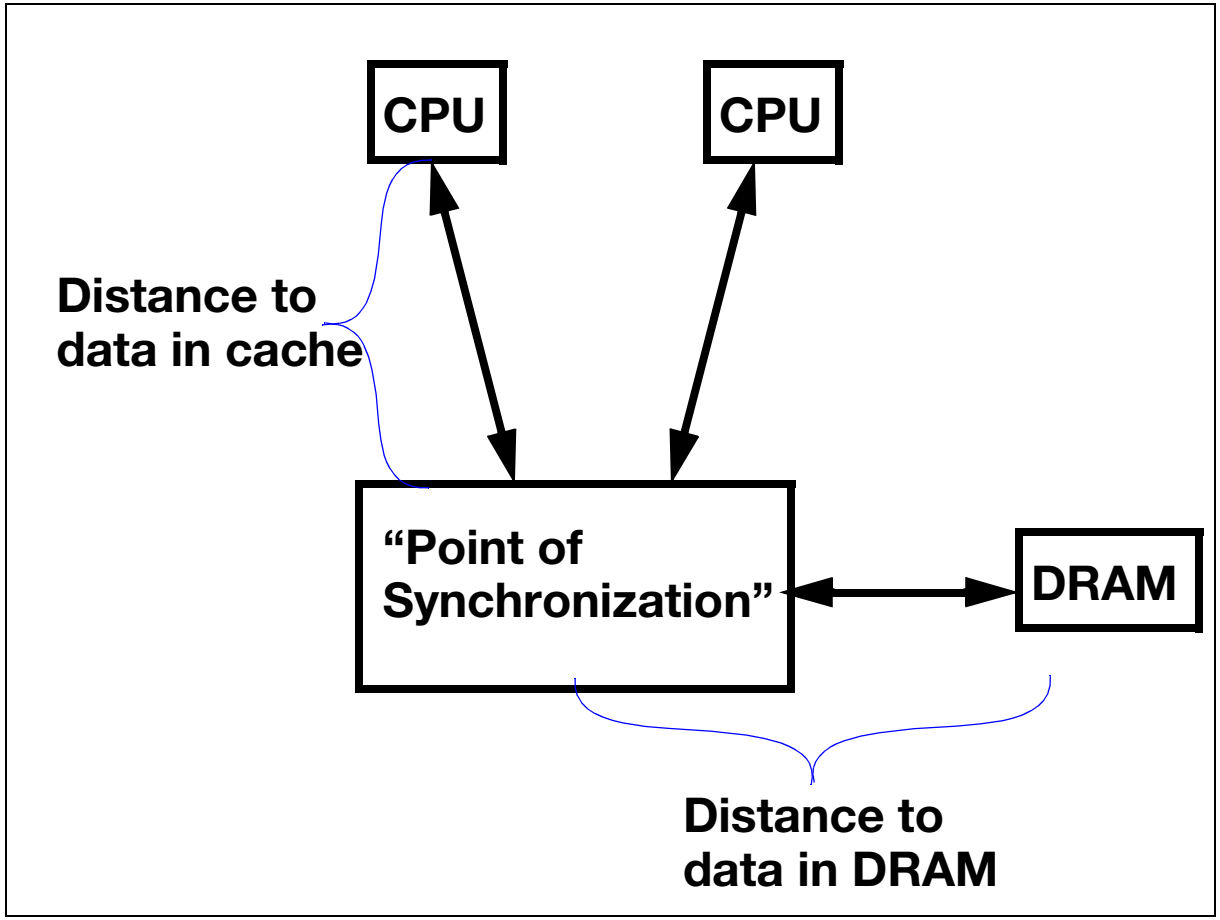


Cache Coherency IIIb

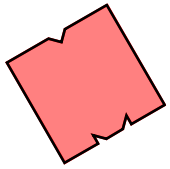




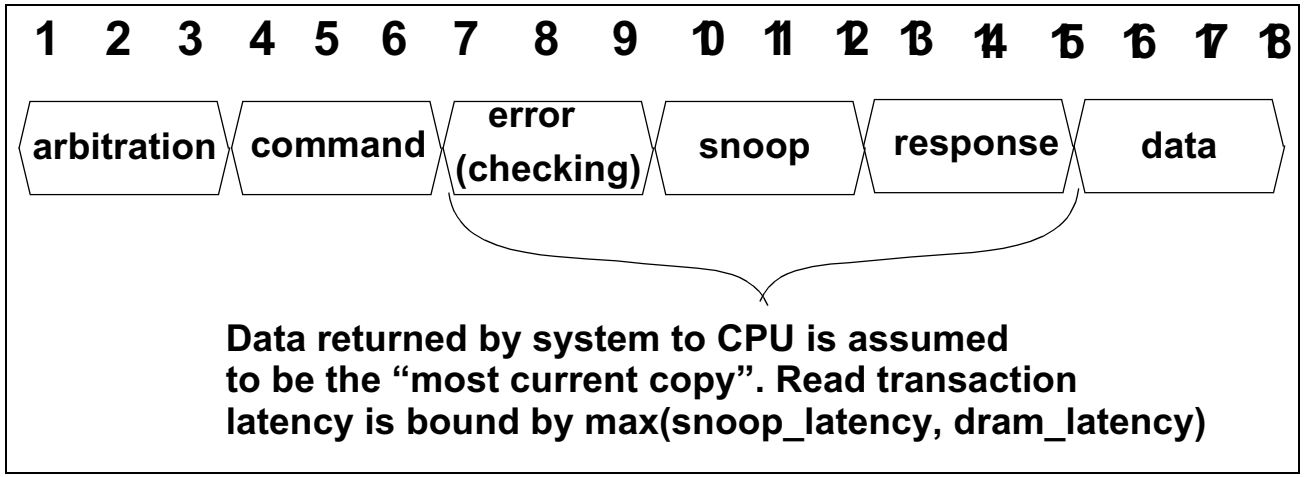
Why worry about CC? Part 1



What if distance to DRAM is shorter than distance to cache (in another CPU)?

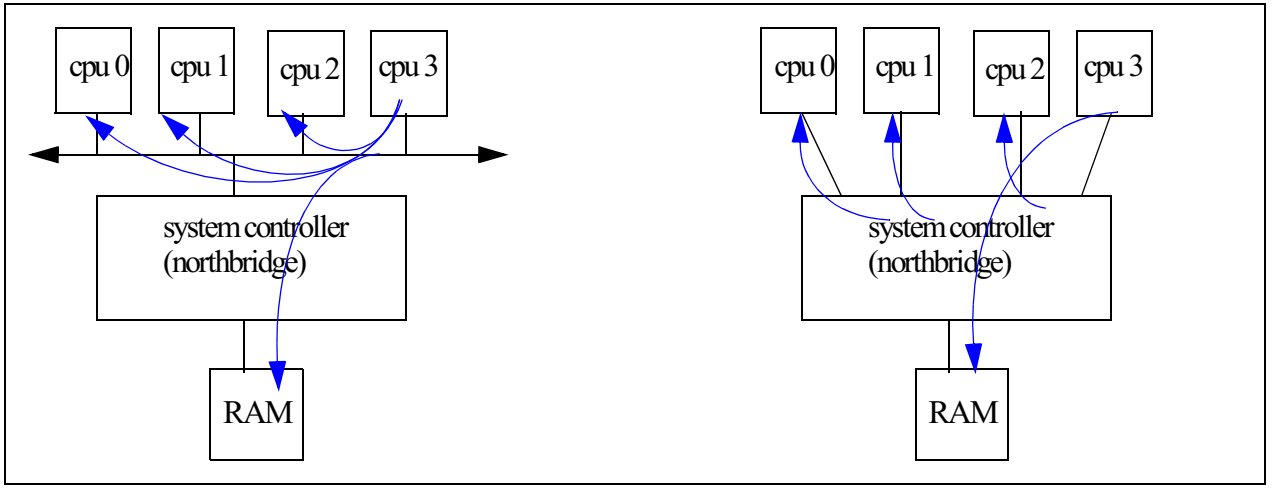


Why worry about CC? Part 2



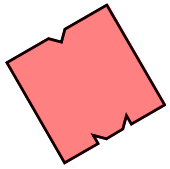
Data returned by system to CPU is assumed to be the "most current copy". Read transaction latency is bound by max(snoop_latency, dram_latency)

Intel P6 system bus read transaction latency breakdown

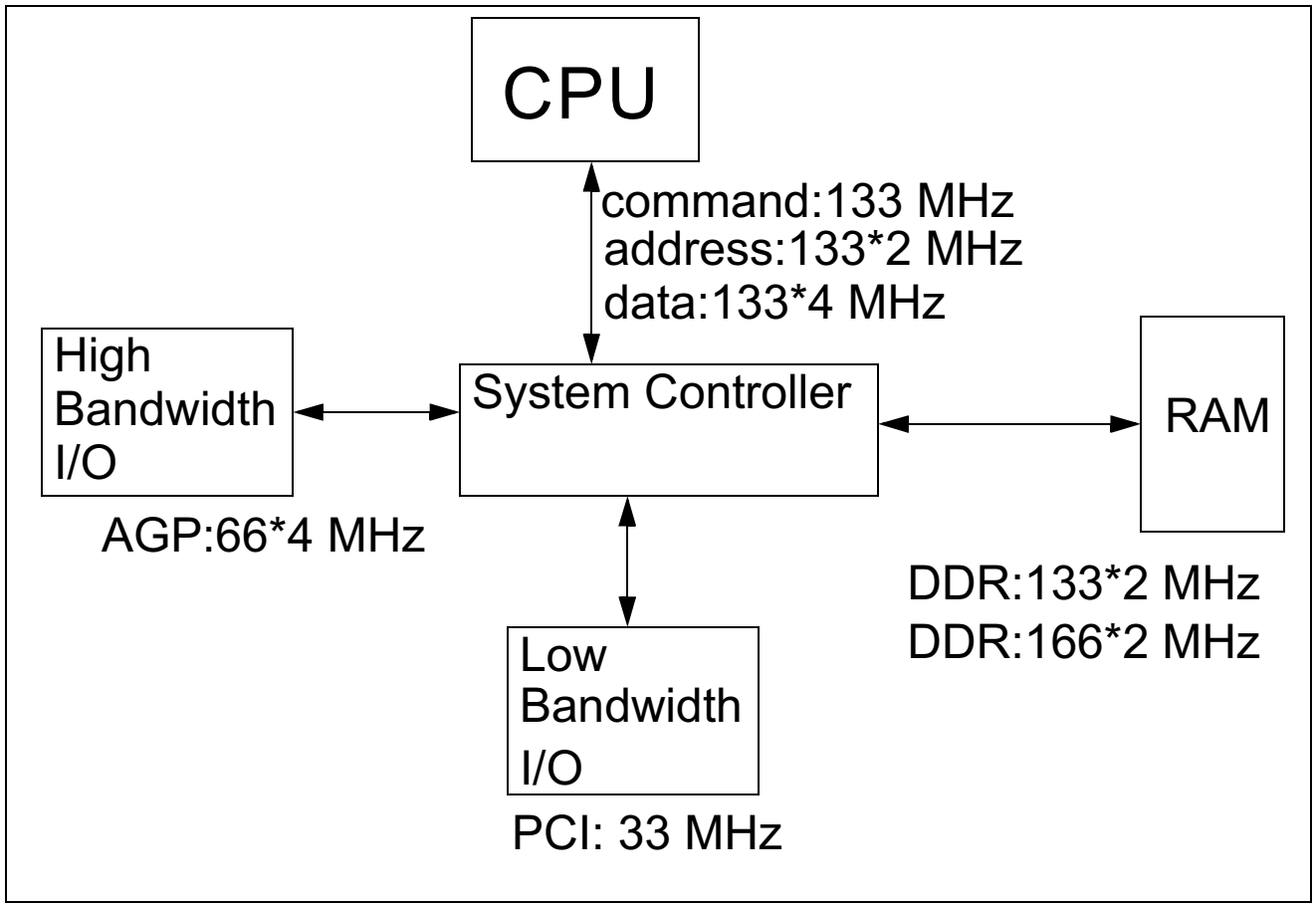


Processors can grab request address off of shared bus in shared multi-drop topology

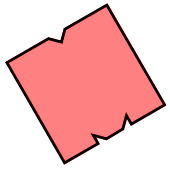
System controller rebroadcast request address to aid in snoop for point-to-point topology



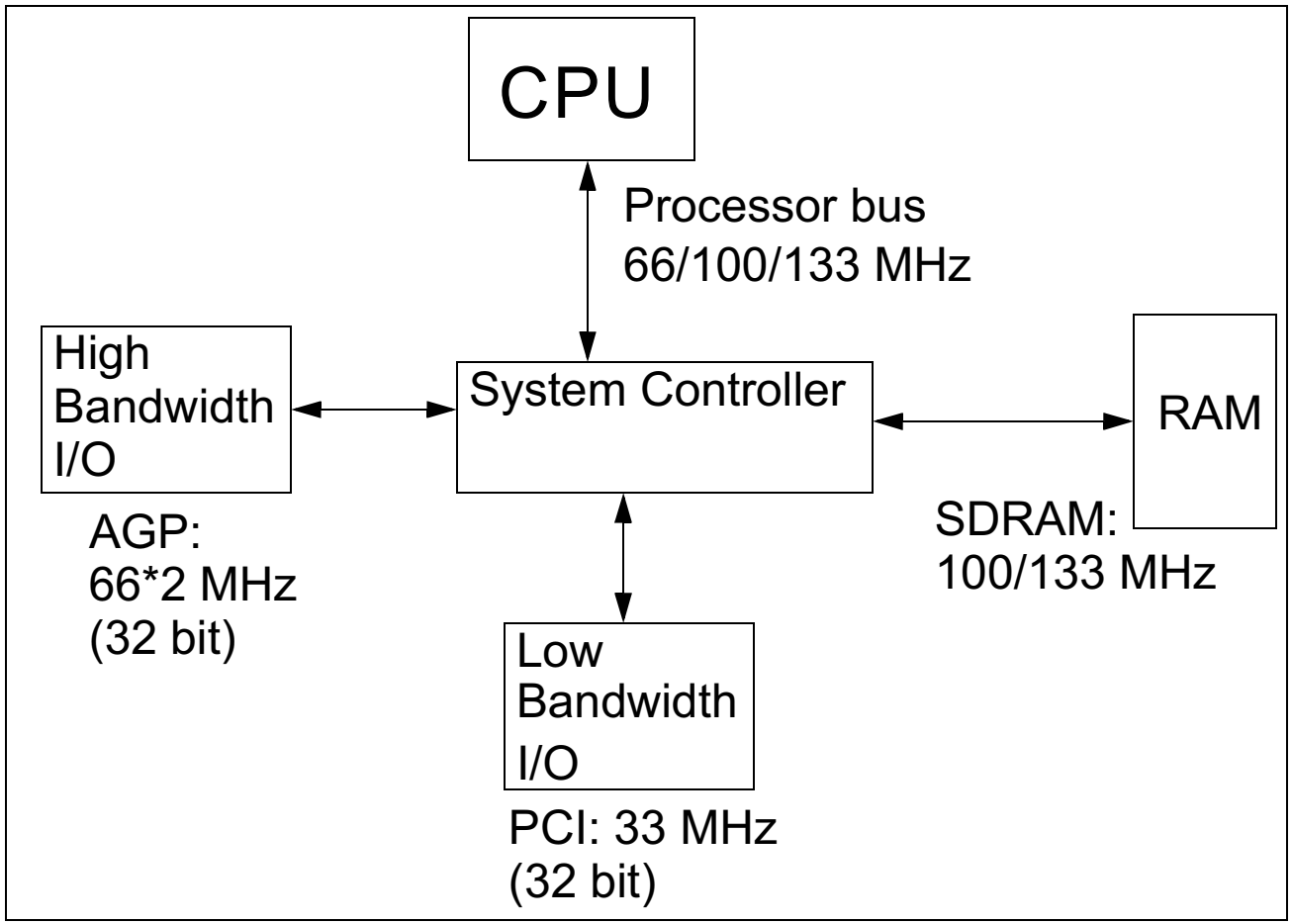
Multiple Clock Domains I



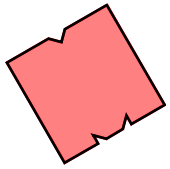
Most clock domains are integer multiples of each other



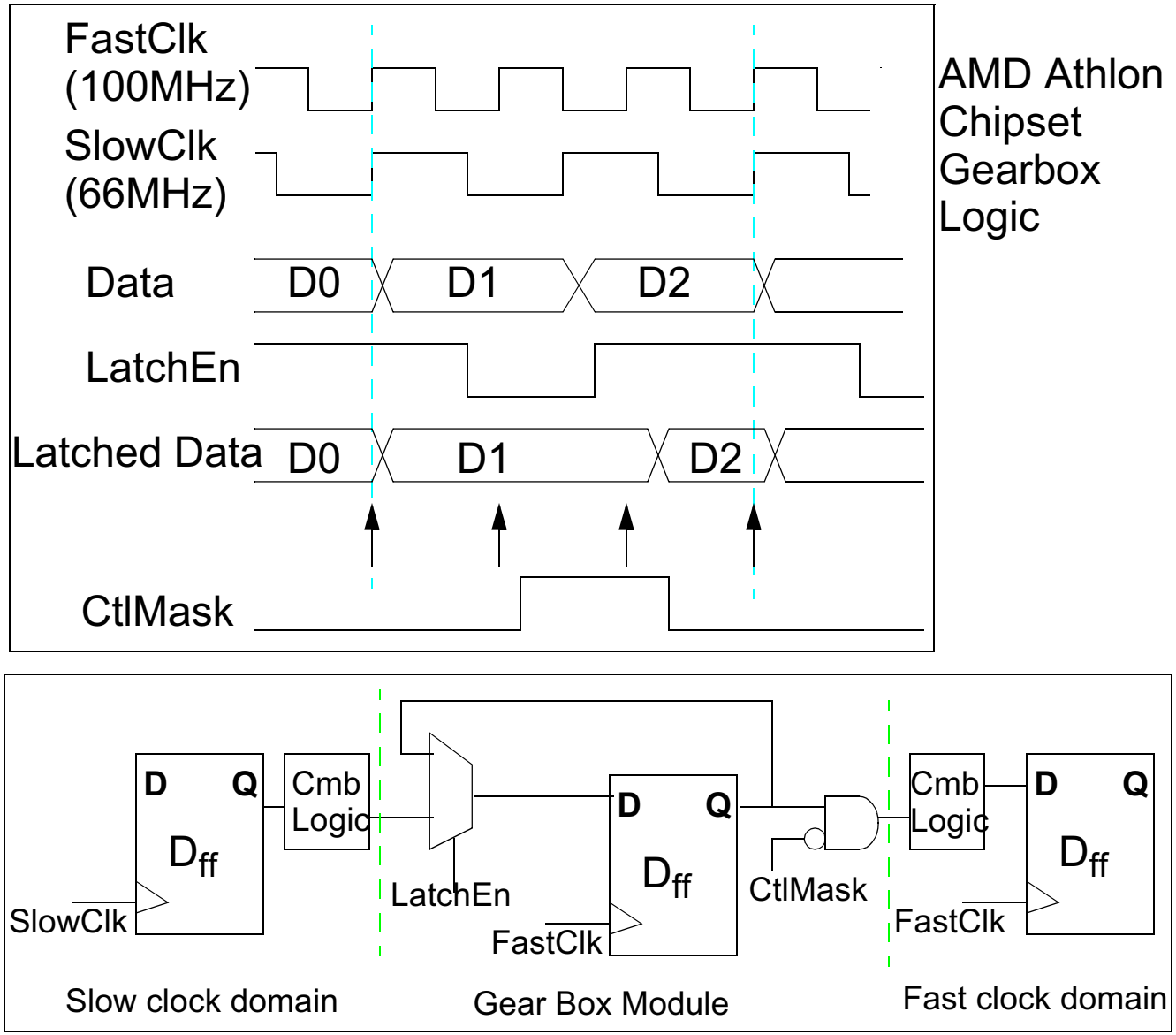
Multiple Clock Domains II



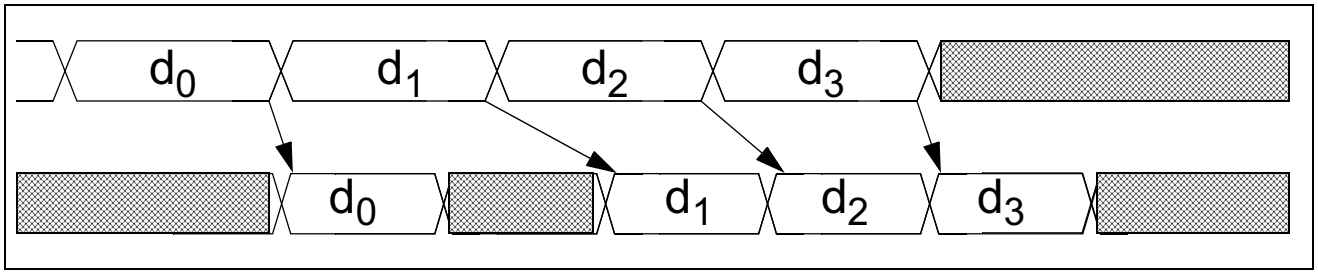
What if clock domains are not integer multiples of each other?



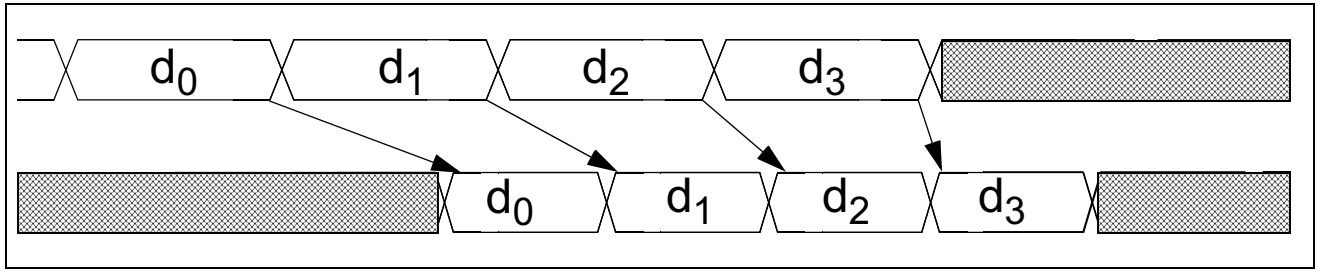
Multiple Clock Domains III



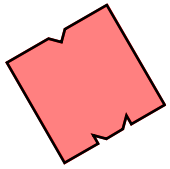
Multiple Clock Domains IV



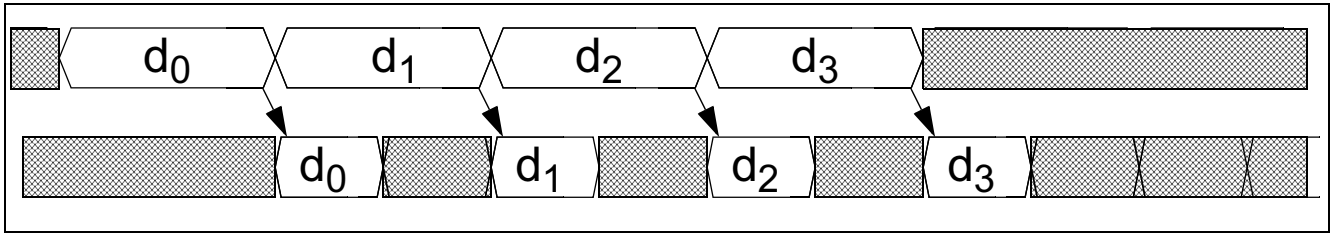
Data transfer from 100 MHz clock domain to 133 MHz clock domain (Latency Optimal)



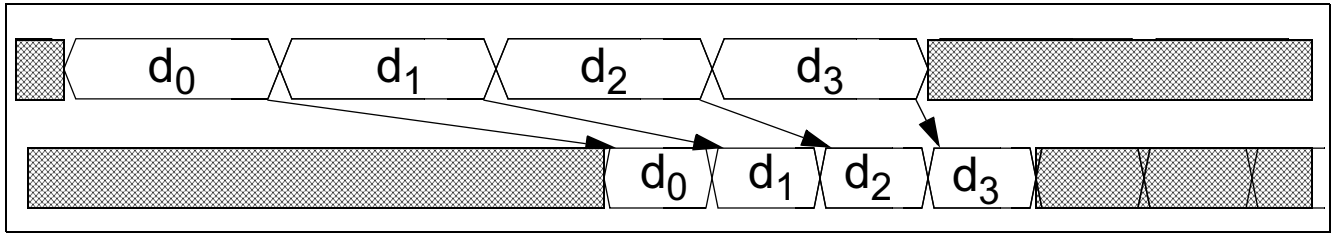
Data transfer from 100 MHz clock domain to 133 MHz clock domain (Bandwidth Optimal)



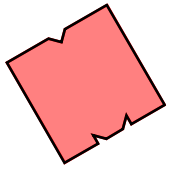
Multiple Clock Domains V



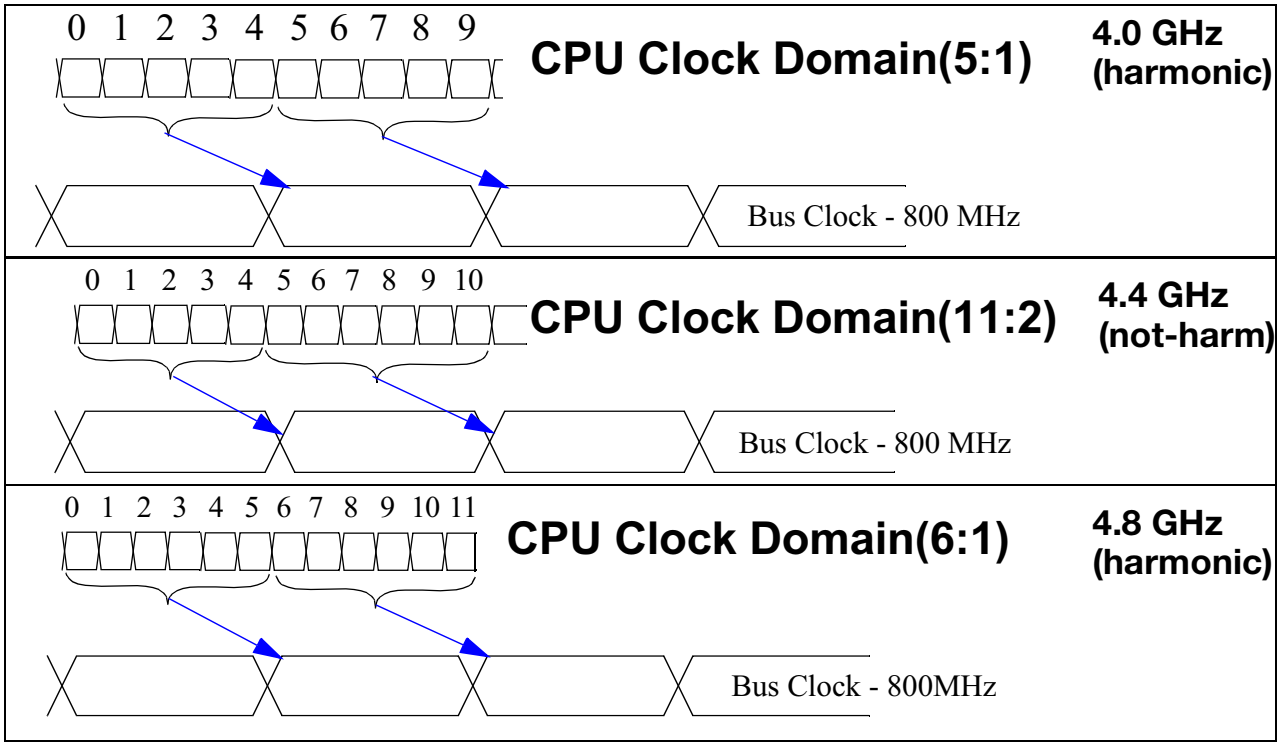
Data transfer from 400 MHz clock domain to 800 MHz clock domain (Latency Optimal)



Data transfer from 400 MHz clock domain to 800 MHz clock domain (Bandwidth Optimal)

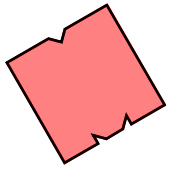


Multiple Clock Domains VI

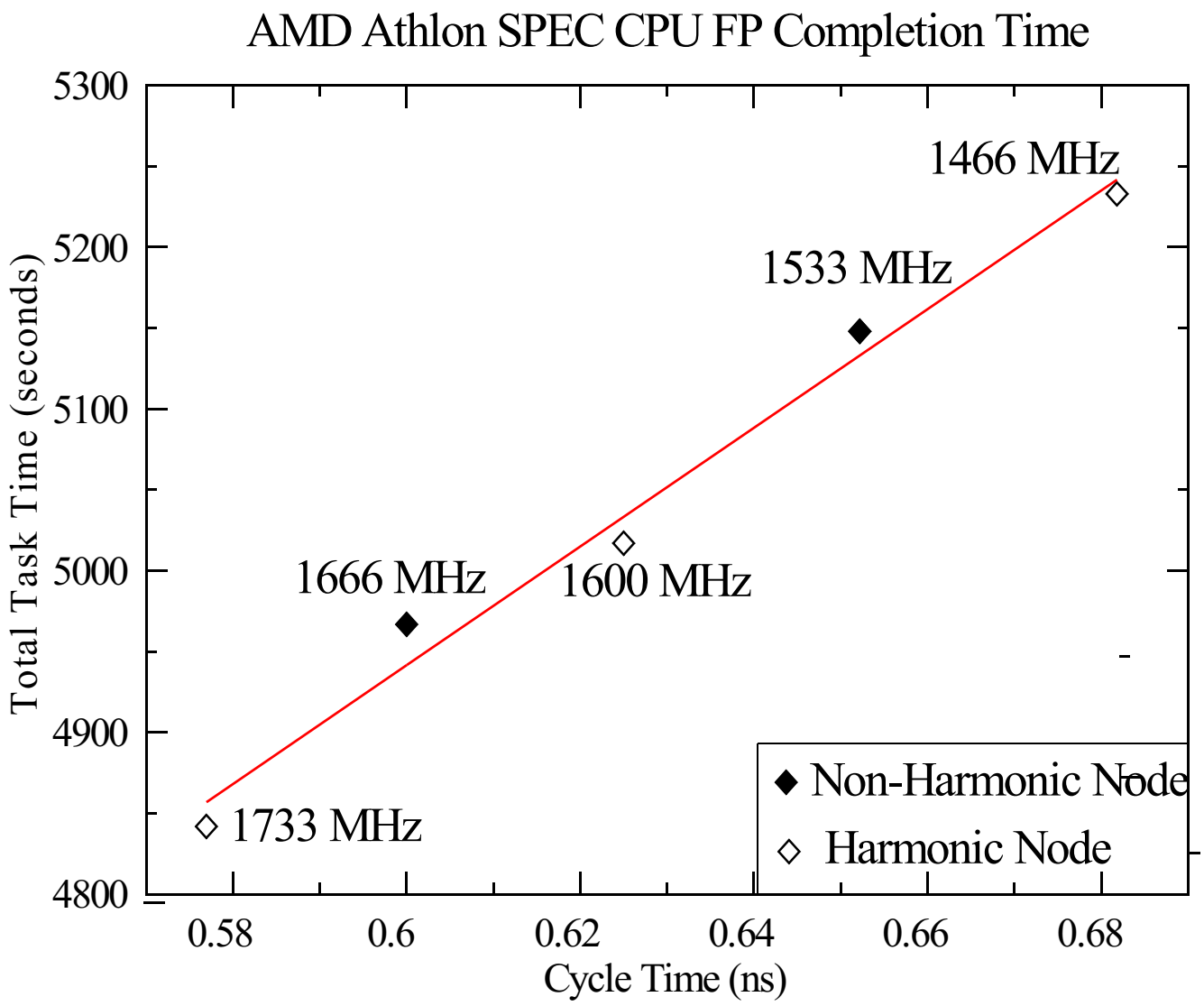


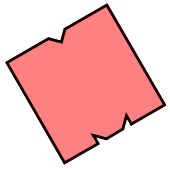
Processor to Processor Bus Interface

Fractional multipliers could impact performance, but we may not have a choice

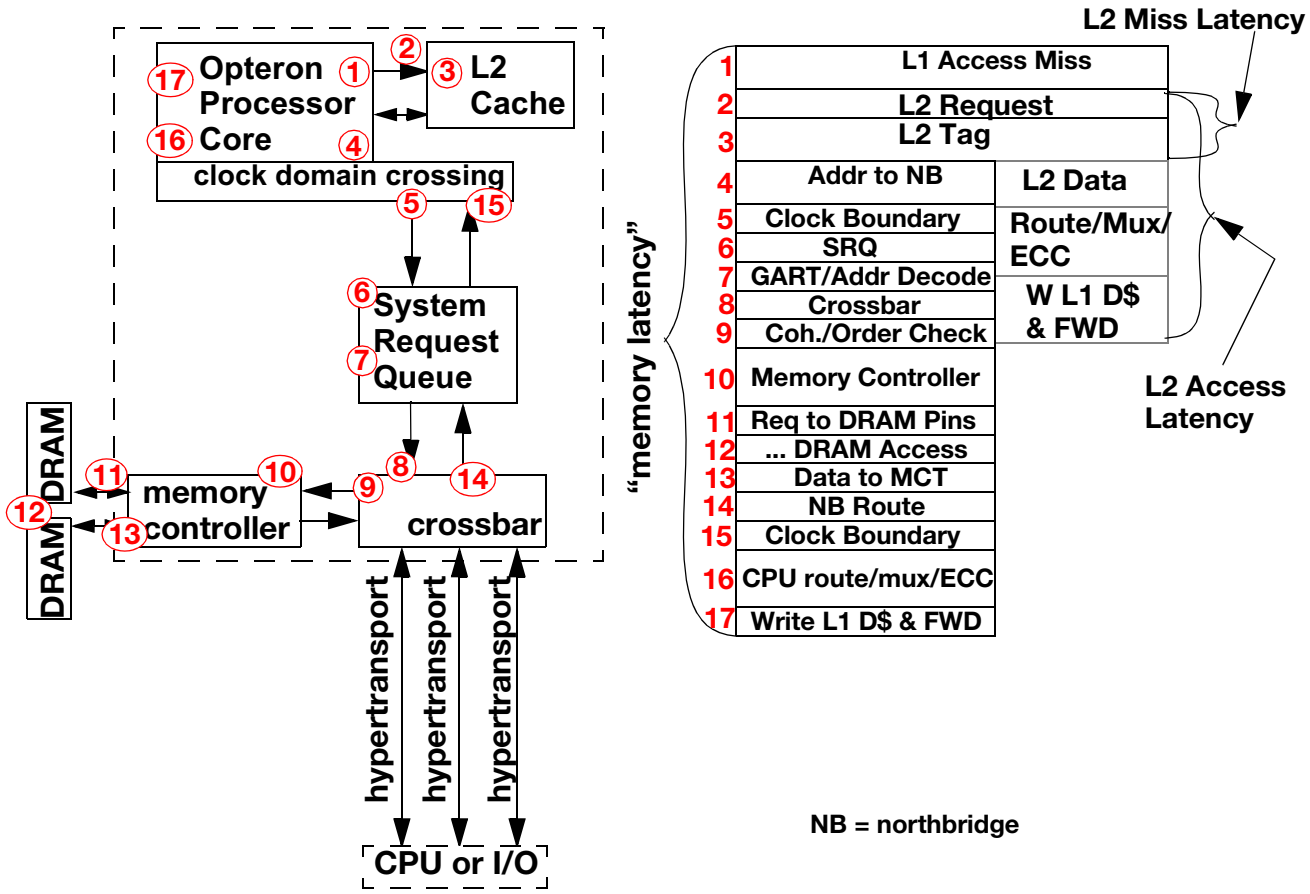


Multiple Clock Domains VII

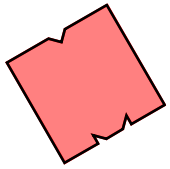




AMD Opteron



Same steps, just all inside the same chip



Summary

- System Controller is a “traffic cop”
- Traffic cop may have to deal with clock domain synchronization issue
- Handles Cache Coherency for small scale SMP configuration
- “Memory Latency” depends on lots of little things, not just speed of DRAM.