

15-213

"The course that gives CMU its Zip!"

Main Memory and Caches

Sept. 23, 2008

Topics

- DRAM as main memory
- Locality of reference
- Caches

Lecture08.ppt

Announcements

Exam Thursday (two days from now)

- In class
- See exam page on class website for info and old exams

Calculator policy

- Calculators will not be needed on the exam; hence forbidden

Collaboration reminder

- Writing code together counts as "sharing code" - forbidden
- Talking through a problem can include pictures (not code)

2

15-213, F08

Byte-Oriented Memory Organization



- **Programs Refer to Virtual Memory Addresses**
 - Conceptually very large array of bytes
 - Actually implemented with hierarchy of different memory types
 - System provides address space private to particular "process"
 - Program being executed
 - Program can clobber its own data, but not that of others
- **Compiler + Run-Time System Control Allocation**
 - Where different program objects should be stored
 - All allocation within single virtual address space

3

From class02.ppt

15-213, F08

Simple Addressing Modes

- **Normal (R) Mem[Reg[R]]**
 - Register R specifies memory address

```
movl (%eax), %eax
```
- **Displacement D(R) Mem[Reg[R]+D]**
 - Register R specifies start of memory region
 - Constant displacement D specifies offset

```
movl 8(%ebp), %edx
```

4

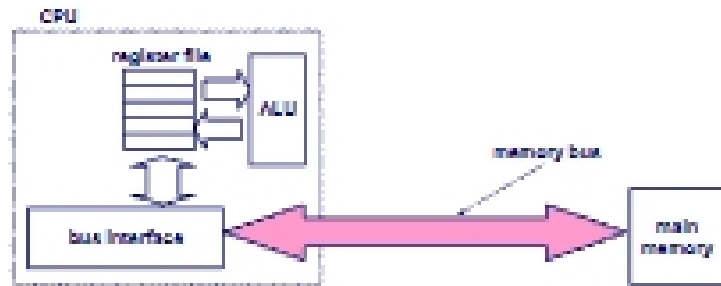
From class02.ppt

15-213, F08

Traditional Bus Structure Connecting CPU and Memory

A **bus** is a collection of parallel wires that carry address, data, and control signals.

Buses are typically shared by multiple devices.



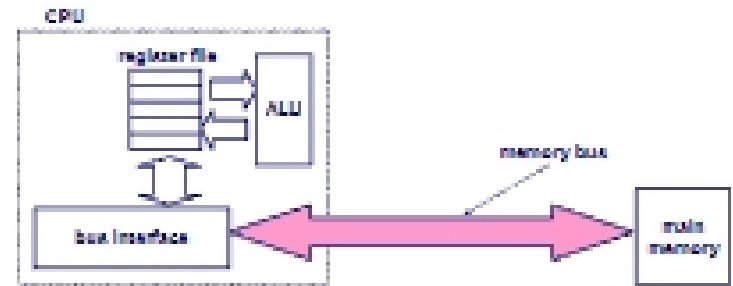
5

15-213, F08

Traditional Bus Structure Connecting CPU and Memory

A **bus** is a collection of parallel wires that carry address, data, and control signals.

Buses are typically shared by multiple devices.

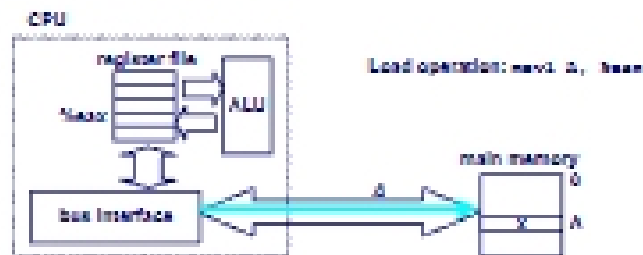


6

15-213, F08

Memory Read Transaction (1)

Step 1: CPU places address *A* on the memory bus with signal indicating "read"

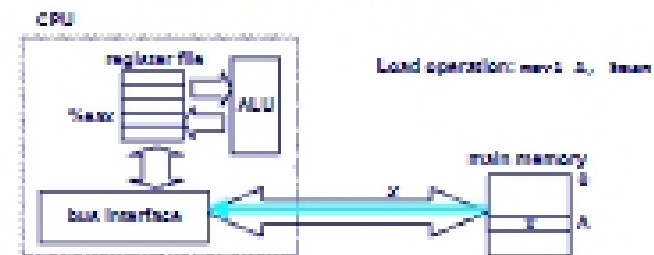


7

15-213, F08

Memory Read Transaction (2)

Steps 2-4: Main memory reads *A* from the memory bus, retrieves word *x*, and places it on the bus

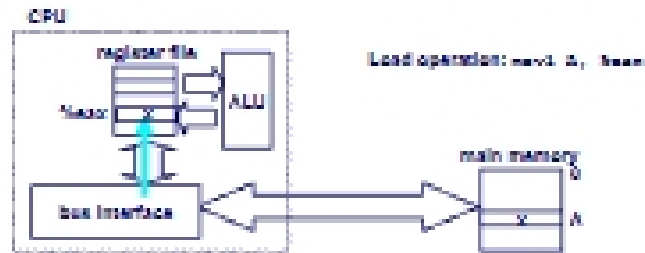


8

15-213, F08

Memory Read Transaction (3)

Step 5: CPU reads word x from the bus and copies it into register $\%eax$



9

15-213, F08

Memory Write Transaction (1)

Step 1: CPU places address A on the memory bus with signal indicating "write"

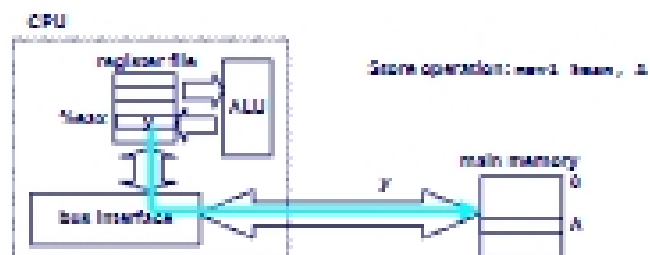


10

15-213, F08

Memory Write Transaction (2)

Step 2: CPU places data word y on the memory bus

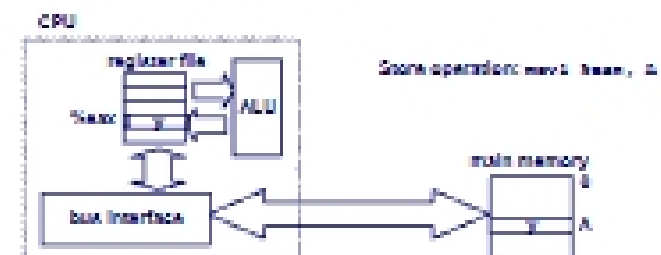


11

15-213, F08

Memory Write Transaction (3)

Steps 3-4: Main memory reads data word y from the bus and stores it at address A



12

15-213, F08