System Buses
Ch 3

Computer Function
Interconnection
Structures
Bus Interconnection
PCI Bus

Computer Function

• von Neumann architecture
  – memory contains both instruction and data

• Fetch-Execute Cycle

Figs 3.3, 3.9
I/O control

- CPU executes instructions and with those instructions guides I/O modules
  - control and data registers in I/O modules
  - I/O modules give feedback to CPU with control and data registers, but only when CPU is reading them!

I/O Control

- Interrupts allow I/O modules to give feedback to CPU even when CPU is doing something else
- DMA allows I/O modules to access memory without CPU’s help
von Neumann Bottleneck

- All components communicate via system bus
- Each component has its own inputs/outputs
  - System bus must support them all

System Bus

- 50-100 lines (wires)
  - address
  - data
  - control
  - other: power, ground, clock

- Performance
  - bandwidth, how many bits per sec?
  - propagation delay?
Bus Configurations

- One bus alone
  - might be very long
  - serious von Neumann bottleneck
  - all devices use similar speeds
  - slowest device dominates?

- Hierarchy of buses
  - can maximize speed for limited access
    (closer to CPU)
  - lower speed general access
    I/O (far from CPU)

Hierarchy of Buses

(Tanenbaum, Structured Computer Organization, 4th Ed.)
Bus Design Features (3)

- **Bus type**
  - dedicated, multiplexed

- **Arbitration method**
  - centralised, distributed
  - bus controller, arbiter

- **Timing**
  - synchronous: all same speed
  - asynchronous: also different speed devices
  - See examples on next slides

Synchronous Timing

CPU writes

- Clock
- Start
- Read
- Address Lines
- Data Lines
- Acknowledge

CPU reads

Memory reads, finds, and writes

Fig. 3.19 (a)
Asynchronous Timing (no anim)

CPU writes address & read request until all OK

CPU tells that request is waiting

Mem writes data and signals ready

Memory reads req and locates data

CPU reads data

CPU read OK, drops request

mem done, drops all

CPU done, drops all

Asynchronous Timing (9)

CPU writes address & read request until all OK

CPU tells that request is waiting

Mem writes data and signals ready

Memory reads req and locates data

CPU reads data

CPU read OK, drops request

mem done, drops all

CPU done, drops all

Fig. 3.19 (b)
Bus Design Features (cont)

- Bus width
  - address, data

- Data transfer types
  - read, write
    - multiplexed & non-multiplexed operations
  - read-modify-write
    - E.g., for indivisible increments (multiproc. env.)
  - read-after-write
    - E.g., for check that write succeeds (multiproc. env.)
  - block
    - long delay for interrupt handling?

Example Bus: Industry Standard Architecture (ISA, or PC-AT)

- Bus type: dedicated
- Arbitration method: single bus master
- Timing: asynchronous
  - own 8.33 MHz clock,
  - 15.9 MBps max data rate, 5.3 MBps in practice
- Bus width: address 32, data 16
- Data transfer type
  - read, write, read block, write block
Example: Peripheral Component Interconnect (PCI) Bus

- Bus type: multiplexed
- Arbitration method: centralised arbiter
- Timing: synchronous, own 33 MHz clock
  - 2.122 Gbps (265 MBps) max data rate
- Bus width: address/data 32 (64), signal 17
- Data transfer type
  - read, write, read block, write block
- max 16 slots (devices)
PCI Configurations

- Hierarchy
- Bridge to internal/system bus allows them to be faster
- Bridge to expansion buses allows them to slower

PCI Bus

49 Mandatory Signals

- 32 pins for address/data, time multiplexed
  - 1 parity pin
- 4 pins for command type/byte enable
  - E.g., 0110/1111 = memory read/all 4 bytes
- System pins (2): clock, reset
- Transaction timing & coordination pins (6)
- Arbitration pins (2 for each device) to PCI bus arbiter: REQ, GNT
- Error pins (2): parity, system
PCI Bus

41 Optional Signals (4)

- Request interrupt pins (4 pins for each dev)
- Cache support pins (2) for snoopy cache protocols
- 32 pins for additional multiplexed address/data
  - plus 7 control/parity pins
- 5 test pins

PCI Bus Transaction (4)

- Bus activity is in separate transactions
- Each transaction preceded by arbitration
  - central arbiter (e.g., First-In-First-Out)
  - determines initiator/master for transaction
- Transaction is executed
- Bus is marked “ready” for next transaction
PCI Transaction Types (5)

- **Interrupt Acknowledge**
  - READ interrupt parameter (e.g., subtype) for interrupt handler

- **Special Cycle**
  - broadcast message to many targets

- **Configuration Read/Write**
  - Read/Update (Write) device configuration data

- **Dual Address Cycle**
  - use 64 bit addresses in this transaction

- **I/O or memory read/write (line, multiple)**

PCI Read Transaction (no anim)

Fig. 3.22
### Arbitration: A and B want bus

Mostly just arbitration signals shown here

---

**Fig. 3.24**
Arbitration:

A and B want bus (10)

A action

B action

Arbitrator action

a) A wants bus
b) B wants bus
c) A granted bus

d) starts frame, requests for next transaction
e) Grants bus to B for next trans.
g) starts frame no more req.

knows that it has bus and bus is available

Sees that both still want it

A’s target reads data

Sees that only A wants it

b) B wants bus

c) A granted bus

A knows that it has bus and bus is available

B sees that both still want it

A’s target reads data

Sees that only A wants it

d) starts frame, requests for next transaction

A knows that it has bus and bus is available

B sees that both still want it

A’s target reads data

Sees that only A wants it

e) Grants bus to B for next trans.
f) marks last frame transfer, marks data ready
g) starts frame no more req.

All ready for new trans

All ready for new trans, granted for B, B knows that it has bus

3GIO - New Bus to Replace PCI

- Code name “Arapahoe” or 3GIO
- Prevent bus bottleneck between fast CPU and memory of the future
- Arapahoe Work Group
  - Compaq, Dell, IBM, Intel and Microsoft
- Will replace PCI as industry standard
- PCI devices will work with Arapahoe
- Speedup 50x as compared to std PCI
  - E.g., 100 MB/s/pin vs. 1.58 MB/s/pin
- Scalable capacity per device (pin count, speed)

http://www.pcisig.com
-- End of Chapter 3: System Buses --

(PCI card - connectors also on other side, some pins not used by this card)